THE DATASHEET OF

ADSP-21591KBPZ8

SHARC+ Dual-Core DSP with Arm Cortex-A5

ADSP-21591/ADSP-21593

SYSTEM FEATURES

Dual-enhanced SHARC+ floating-point cores
- **High performance SHARC+ cores (up to 1 GHz each)**
- **Up to 5 Mb (640 kB) L1 SRAM 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 support**
- **Byte, short word, word, and long word addressability**

Arm Cortex-A5 core
- **Up to 1 GHz/1600 DMIPS with NEON/VFPv4-D16**
- **32 kB L1 instruction and data caches with parity**
- **256 kB L2 cache with parity**

Powerful DMA system with 8 MemDMAs

On-chip memory protection

Integrated safety features

17 mm × 17 mm, 400-ball BGA_ED (0.8 mm pitch), RoHS compliant

MEMORY

Large on-chip Level 2 (L2) SRAM with ECC protection, up to 2 MB

One Level 3 (L3) interface providing 16-bit interface to DDR3/ DDR3L SDRAM devices

ADDITIONAL FEATURES

ADSP-2156x pin-compatible package options

Enhanced FIR and IIR accelerators running up to 1 GHz

Security and protection
- **Cryptographic hardware accelerators**
- **Fast secure boot with IP protection**
- **Support for Arm TrustZone**

APPLICATIONS

Automotive: audio amplifier, head unit, ANC/RNC, rear seat entertainment, digital cockpit, ADAS

Consumer: AVRs, mixing consoles, microphone arrays, conferencing systems

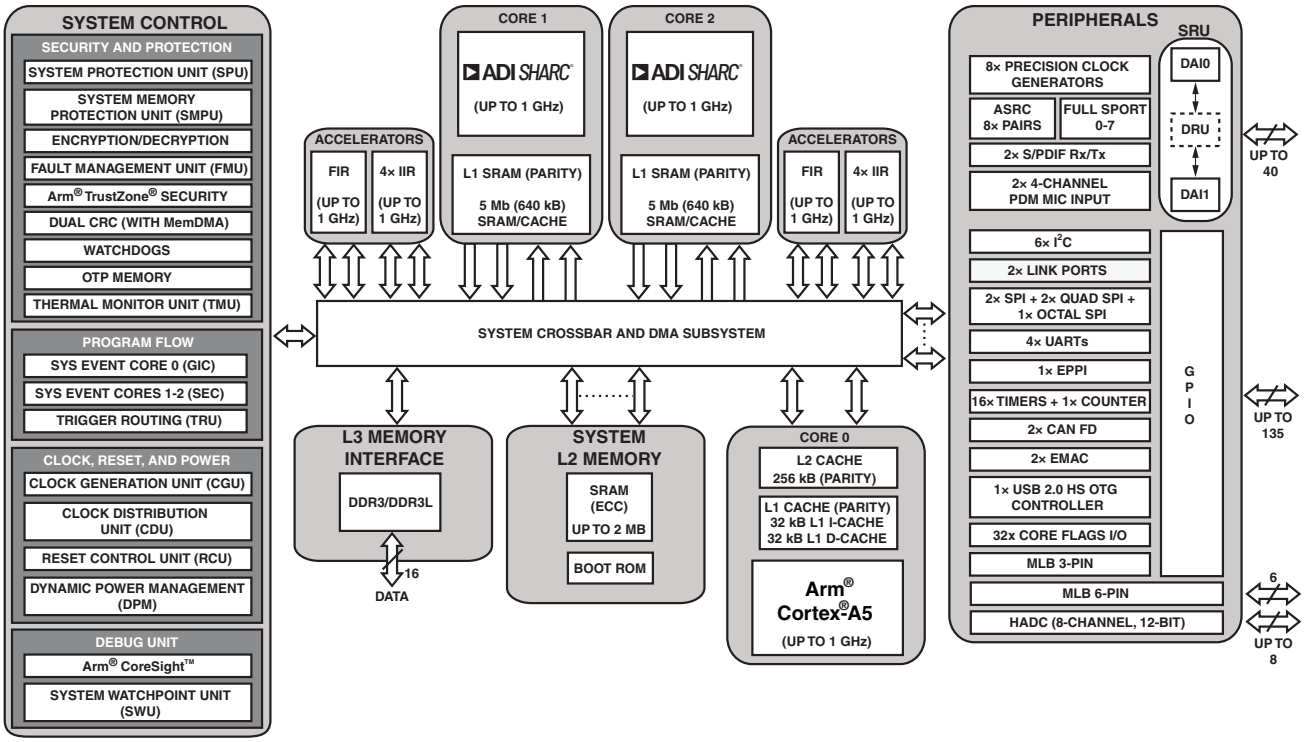


Figure 1. ADSP-SC594 (Full-Featured Model) Processor Block Diagram

Rev. 0 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

REVISION HISTORY

8/2021—Revision 0: Initial Version

Analog Devices is in the process of updating documentation to provide terminology and language that is culturally appropriate. This is a process with a wide scope and will be phased in as quickly as possible. Thank you for your patience.

GENERAL DESCRIPTION

The ADSP-2159x/ADSP-SC59x processors are members of the SHARC® family of products. The ADSP-SC59x processors are based on the SHARC+® dual-core and the Arm® Cortex®-A5 core. The ADSP-2159x/ADSP-SC59x SHARC processors are members of the single-instruction, multiple data (SIMD) SHARC family of digital signal processors (DSPs) that feature Analog Devices, Inc., Super Harvard Architecture. These 32-bit/40-bit/64-bit floating-point processors are optimized for high performance audio/floating-point applications with large on-chip static random-access memory (SRAM), multiple internal buses that eliminate input/output (I/O) bottlenecks, and innovative digital audio interfaces (DAI). New additions to the SHARC+ core include cache enhancements and branch prediction, while maintaining instruction set compatibility to previous SHARC products.

By integrating a set of industry leading system peripherals and memory, the Arm Cortex-A5 and SHARC processor is the platform of choice for applications that require programmability similar to reduced instruction set computing (RISC), multimedia support, and leading edge signal processing in one integrated package. These applications span a wide array of markets, including automotive, professional audio, and industrial-based applications that require high floating-point performance.

Table 1 provides comparison information for features that vary across the standard processors.

Table 1. Processor Features[1]

Processor Feature	ADSP-21591	ADSP-21593
Arm Cortex-A5 (MHz, Maximum)[2]	N/A	N/A
Arm Core L1 Cache (I, D kB)	N/A	N/A
Arm Core L2 Cache (kB)	N/A	N/A
SHARC+ Core1 (MHz, Maximum)[2]	800	800, 1000
SHARC+ Core2 (MHz, Maximum)[2]	800	800, 1000
SHARC L1 SRAM (kB)	2 × 640	2 × 640
System Memory		
L2 SRAM (Shared) (MB)	1	2
DDR3/DDR3L Controller (16-Bit)	1	1
Hardware Accelerators		
FIRs Per SHARC+ Core	1	1
IIRs Per SHARC+ Core	4	4
Security Cryptographic Engine	Yes	Yes
DAI (Includes SRU and DRU)	2	2
Full SPORTs	8 (4 per DAI)	8 (4 per DAI)
S/PDIF Receive/Transmit	2 (1 per DAI)	2 (1 per DAI)
ASRCs	8 (4 per DAI)	8 (4 per DAI)
PCGs	8 (4 per DAI)	8 (4 per DAI)
4-Channel PDM MIC Input	2 (1 per DAI)	2 (1 per DAI)
Pin Buffers	28 (14 per DAI)	28 (14 per DAI)
Multiplexed Peripherals		
MLB 3-Pin	Yes[3]	Yes[3]
Link Ports	2	2
GP Counter	1	1
I^2C (TWI)	6	6
Watchdog Timers	3	3
GP Timers	16[4]	16[4]
Octal SPI	1	1
Quad-Data Bit SPI	2	2
Dual-Data Bit SPI	1	1
UARTs	3	3
ePPI	N/A	N/A
USB 2.0 HS OTG Controller	N/A	N/A
EMAC Std	N/A	N/A
EMAC Std/AVB + Timer IEEE 1588	N/A	N/A
CAN FD	N/A	N/A
MLB 6-Pin	N/A	N/A
Multichannel 12-Bit ADC	4-channel	4-channel
GPIO Ports	Port A to Port C	Port A to Port C
GPIO + DAI Pins	40 + 28	40 + 28
Package Options	400- ball BGA_ED	400- ball BGA_ED
ADSP-2156x Pin-Compatible	Yes[5]	Yes

[1] N/A means not applicable.
[2] Multiple values indicate various speed grades. See Planned Production Products and Ordering Guide.
[3] Applies to automotive models only. See Planned Automotive Production Products.
[4] For the ADSP-21591 and ADSP-21593 models, GP timer instances TIMER10-TIMER15 are not brought out to package pins and must only be configured for internal modes of operation.
[5] Pin compatible with the ADSP-21566, ADSP-21567, and ADSP-21569.

ARM CORTEX-A5 PROCESSOR (ADSP-SC59x ONLY)

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

- Instruction cache unit (32 kB) and data Level 1 (L1) cache unit (32 kB)
- In order pipeline with dynamic branch prediction
- Arm, Thumb®, and ThumbEE instruction set support
- Arm TrustZone® security extensions
- Harvard L1 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

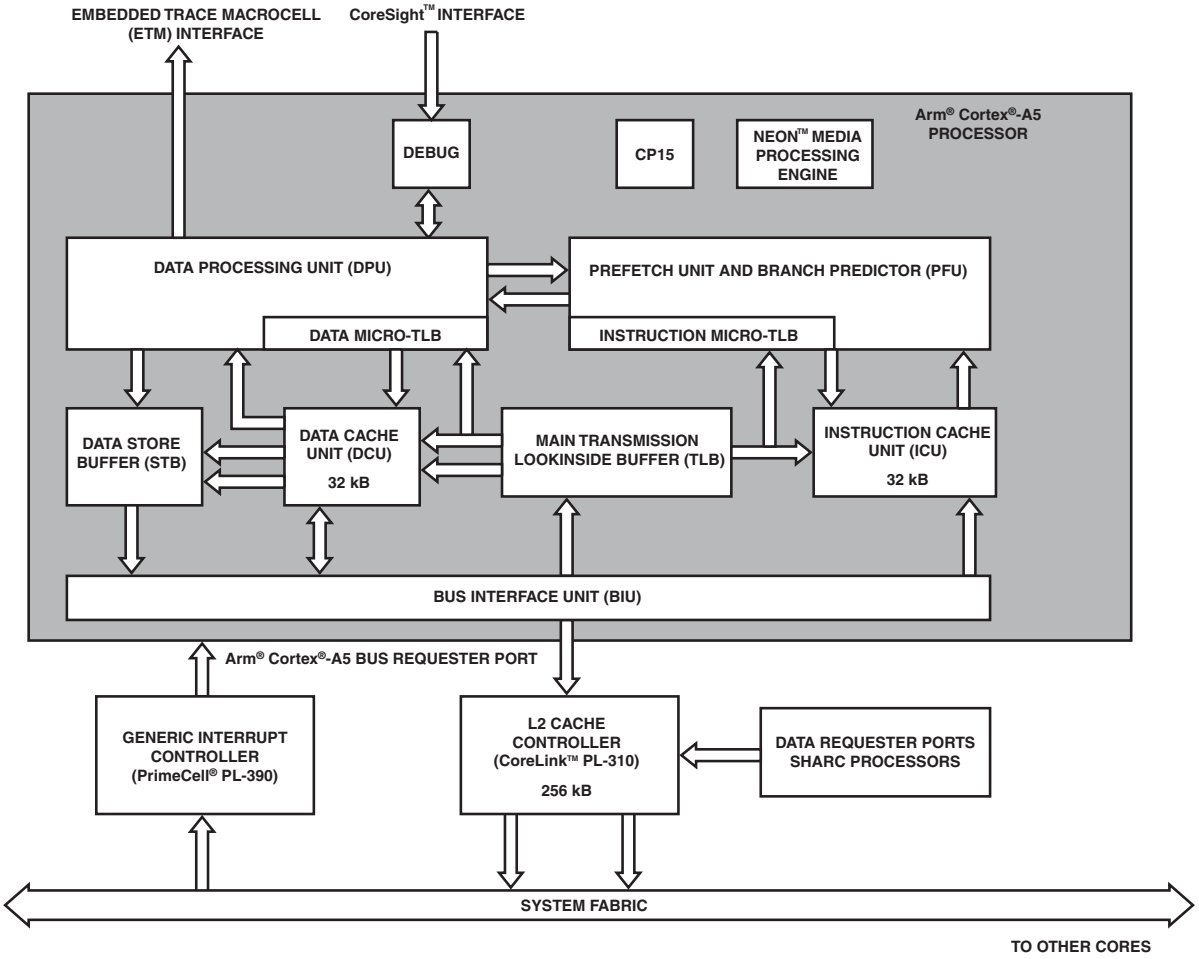


Figure 2. Arm Cortex-A5 Processor Block Diagram

Generic Interrupt Controller (GIC), PL390 (ADSP-SC59x Only)

The generic interrupt controller (GIC) is a centralized resource for supporting and managing interrupts. The GIC splits into the distributor block (GICPORT0) and the central processing unit (CPU) interface block (GICPORT1).

Generic Interrupt Controller Port0 (GICPORT0)

The GICPORT0 distributor block 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)

The GICPORT1 CPU interface block performs priority masking and preemption handling for a connected processor in the system. GICPORT1 supports 8 software generated interrupts (SGIs) and 326 shared peripheral interrupts (SPIs).

L2 Cache Controller, PL310 (ADSP-SC59x Only)

The Level 2 (L2) cache controller, PL310 (see Figure 2), works efficiently with the Arm Cortex-A5 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 the following:

- Two read/write 64-bit completer ports, one connected to the Arm Cortex-A5 instruction and data interfaces, and one connecting the Arm Cortex-A5 and SHARC+ cores for data coherency
- Two read/write 64-bit requester ports for interfacing with the system fabric

SHARC PROCESSOR

The SHARC processor integrates a SHARC+ SIMD core, L1 memory crossbar, I-cache/D-cache controller, L1 memory blocks, and the requester/completer ports, as shown in Figure 3. The SHARC+ SIMD core block diagram is shown in Figure 4.

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

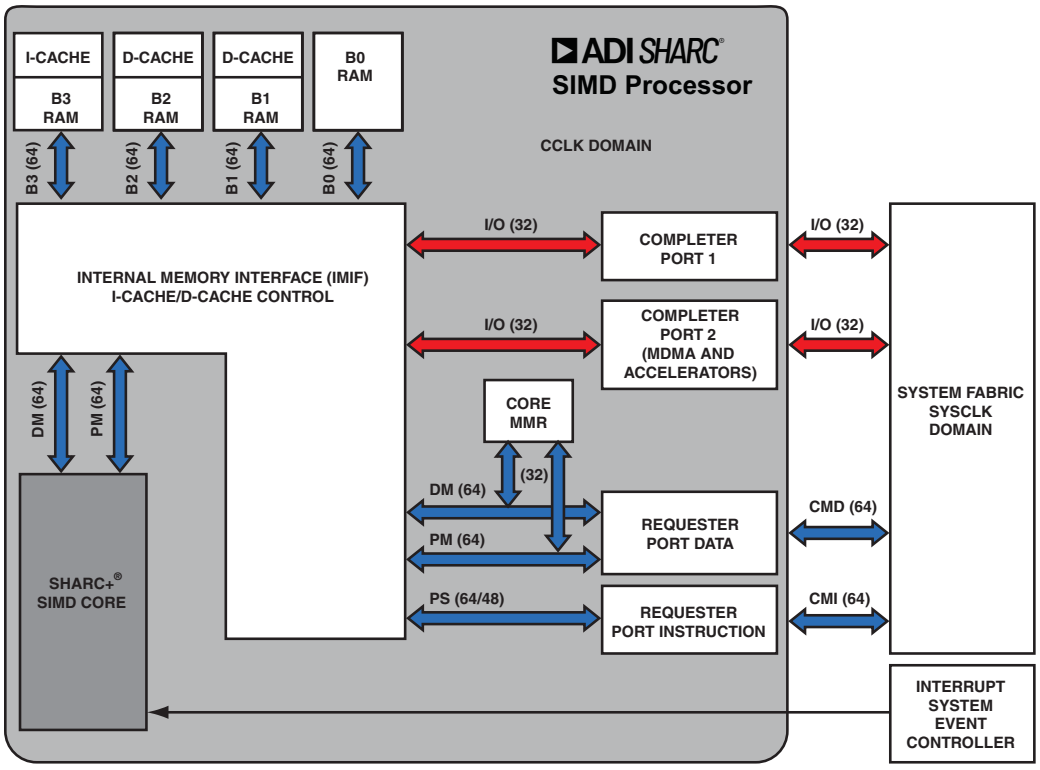


Figure 3. SHARC Processor Block Diagram

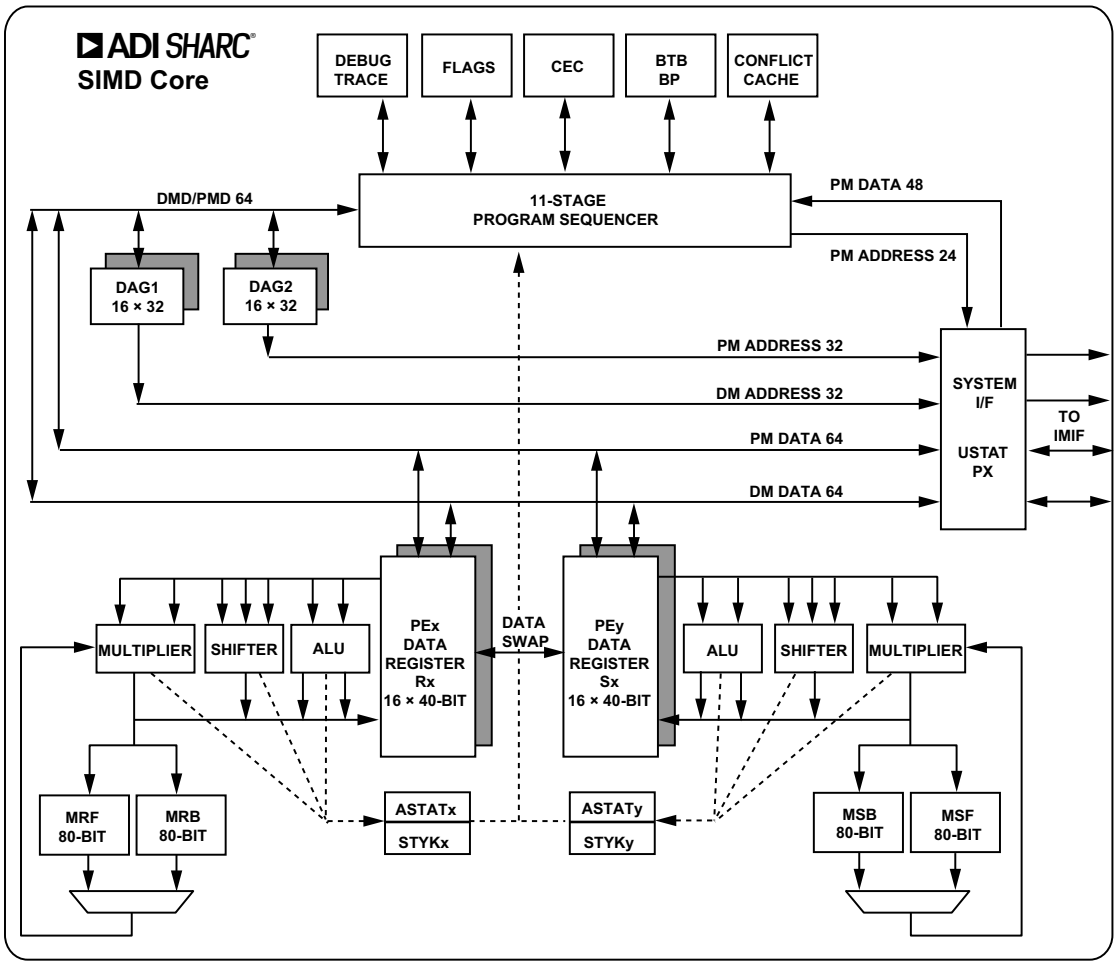


Figure 4. SHARC+ SIMD Core Block Diagram

L1 Memory

Figure 5 shows the ADSP-2159x/ADSP-SC59x memory map. Each SHARC+ core has a tightly coupled 5 Mb L1 SRAM. Each SHARC+ core can access code and data in a single cycle from this memory space. The Arm Cortex-A5 core can also access this memory space with multicycle accesses.

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

SHARC+ core memory-mapped register (CMMR) address space is 0x00000000 through 0x0003FFFF in normal word (32-bit). Each block can be configured for different combinations of code and data storage. Of the 5 Mb SRAM, up to 1 Mb can be configured for data memory (DM), program memory (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 direct memory access (DMA) engine in a single cycle.

The SRAM of the processor can be configured as a maximum of 160k words of 32-bit data, 320k words of 16-bit data, 106.7k words of 48-bit instructions (or 40-bit data), or combinations of different word sizes up to 5 Mb. All of the memory can be accessed as 8-bit, 16-bit, 32-bit, 48-bit, or 64-bit words. Support of a 16-bit floating-point storage format doubles the amount of data that can be stored on chip.

Conversion between the 32-bit floating-point and 16-bit floating-point formats is performed in a single instruction. Whereas 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 and PM buses, with each 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 typical configuration is 512 kb DM, 128 kb PM, and 128 kb of instruction cache, with the remaining L1 memory configured as SRAM. Each addressable memory space outside the L1 memory can be accessed either directly or via cache.

The memory map in Table 2 gives the L1 memory address space and shows multiple L1 memory blocks offering a configurable mix of SRAM and cache.

L1 Requester and Completer Ports

Each SHARC+ core has two requester/completer ports to and from the system fabric. One requester port fetches instructions. The second requester port drives data to the system world. Completer Port 1 together with Completer Port 2 memory direct memory access (high speed MDMA and accelerators) run conflict free access to the individual memory blocks. For the completer port address, refer to the L1 memory address map in Table 2.

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 both the DMD and PMD buses (2 × 64-bits CCLK speed and 2 × 32-bit SYSCLK speed).

Instruction and Data Cache

The ADSP-2159x/ADSP-SC59x processors also include a traditional instruction cache (I-cache) and two data caches (D-caches, one each for PM/DM) with parity support for all caches. These caches support one instruction access and two data accesses over the DM and PM buses per CCLK cycle. The cache controllers automatically manage the configured 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 0 to 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 full and partial locking, range bound invalidation, and flushing.

Core Memory-Mapped Registers (CMMR)

The core memory-mapped registers (CMMR) control the L1 instruction and data cache, branch target buffer (BTB), L2 cache, parity error, system control, debug, and monitor functions.

SHARC+ CORE ARCHITECTURE

The ADSP-2159x/ADSP-SC59x processors are assembly code compatible with all previous SHARC processors featuring the SHARC or SHARC+ core, beginning with the first generation ADSP-2106x SHARC processors and including the ADSP-2116x, ADSP-2126x, ADSP-213xx, ADSP-214xx, and ADSP-SC5xx/ADSP-215xx processors.

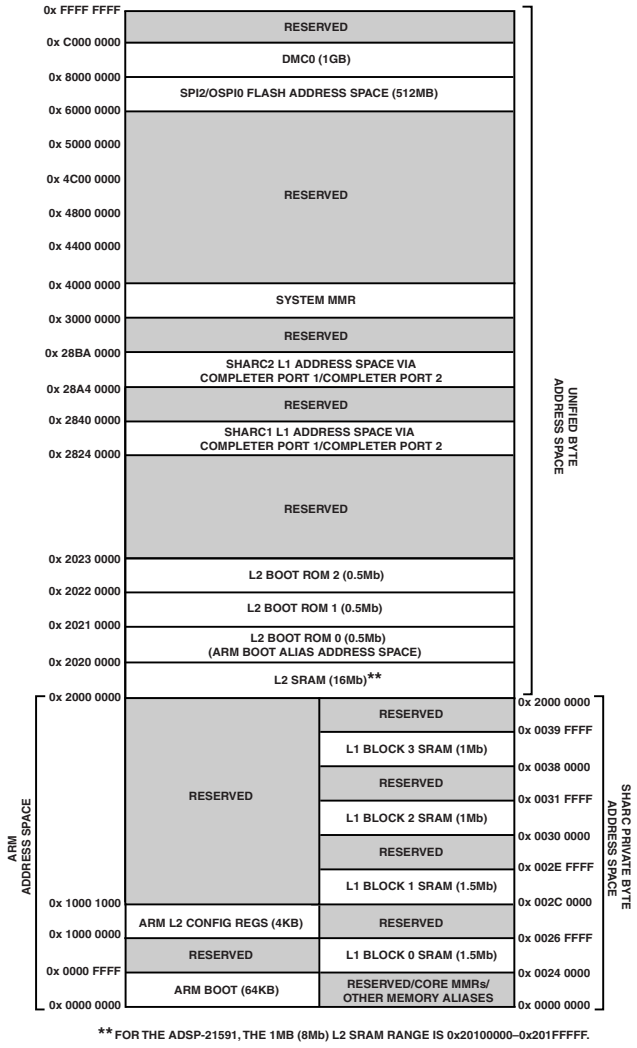


Figure 5. ADSP-2159x/ADSP-SC59x Memory Map

The SIMD architecture featured on the ADSP-2159x/ADSP-SC59x processors is identical to all previous SIMD SHARC processors, namely the ADSP-2116x, ADSP-2126x, ADSP-213xx, ADSP-214xx, and ADSP-SC5xx/ADSP-215xx processors, as shown in Figure 4 and as described in the following sections.

Single-Instruction, Multiple Data (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, each containing an arithmetic logic unit (ALU), multiplier, shifter, and register file. PEx is always active, and PEy is enabled by setting the PEYEN mode bit in the mode control register (MODE1).

SIMD mode allows the processors to execute the same instruction in both processing elements, but each processing element operates on different data. This architecture efficiently executes math intensive DSP algorithms. In addition to all the features of previous generation SHARC cores, the SHARC+ core also provides a new and simpler way to execute an instruction only on the PEy data register.

SIMD mode doubles the bandwidth between memory and the processing elements, as required for sustained computational operation of two processing elements. When using the data address generators (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 a multiplier, an ALU, and a 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.

A multifunction instruction set supports parallel execution of the 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 for an interlocked pipeline plus data dependency check.

Core Timer

Each SHARC+ processor core includes an extra 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 register files (16 primary, 16 secondary), combined with the enhanced Harvard architecture of the processor, allow unconstrained data flow between computation units and internal memory. The registers in the PEx data register file are referred to as R0–R15 and in the PEy data register file 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, DAG, and multiplier result registers have secondary registers. The primary registers are active at reset, whereas control bits in MODE1 activate the secondary registers.

Universal Registers

General-purpose tasks use the universal registers. The four universal status (USTAT) registers allow easy bit manipulations (set, clear, toggle, test, XOR) for all control and status peripheral registers.

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 or DM data bus. These registers contain hardware to handle the data width difference.

Data Address Generators (DAG) With Zero Overhead Hardware Circular Buffer Support

For indirect addressing and implementing circular data buffers in hardware, the ADSP-2159x/ADSP-SC59x processors use 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 fast Fourier transforms (FFT). The DAGs contain sufficient registers to allow the creation of up to 32 circular buffers (16 primary register sets and 16 secondary sets). 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 Instruction Set Architecture (ISA)

The flexible instruction set architecture (ISA), a 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 new to the SHARC+ core, as compared with the previous SHARC core.

Variable Instruction Set Architecture (VISA)

In addition to supporting the standard 48-bit instructions from previous SHARC core processors, the SHARC+ core processors support 16-bit and 32-bit opcodes for many instructions, formerly 48-bit in the ISA. This variable instruction set architecture (VISA) feature drops redundant or 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; rather, it is address dependent (refer to the ISA/VISA address spaces in Table 5). Finally, the processor allows jumps between ISA and VISA instruction fetches.

Single-Cycle Fetch of Instructional Four Operands

The ADSP-2159x/ADSP-SC59x processors feature an enhanced Harvard architecture in which the DM bus transfers data and the PM bus transfers both instructions and data.

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

Core Event Controller (CEC)

The SHARC+ core event controller (CEC) can be configured to service various interrupts generated by the core (including arithmetic and circular buffer instruction flow exceptions) and system event controller (SEC) events (peripheral interrupt request, debug or monitor, and software-raised), responding only to interrupts enabled in the IMASK register. The output of the SEC is forwarded to the CEC to respond directly to any enabled system interrupts. For all SEC channels, the processor automatically stacks the arithmetic status (ASTATx and ASTATy) registers and mode (MODE1) register in parallel with interrupt servicing.

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 that require fetches conflict with the PM bus data access 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 (BTB)/Branch Predictor (BP)

Implementation of a hardware-based branch predictor (BP) and branch target buffer (BTB) reduce branch delay. The program sequencer supports efficient branching using the 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 converting word addresses to byte addresses and byte addresses to word addresses.

SHARC Fabric

The FIR/IIR accelerators on the ADSP-2159x/ADSP-SC59x processors are integrated closely with the SHARC+ core with the help of a dedicated SHARC fabric and run at CCLK speed. This allows the FIR/IIR accelerator requester ports to directly access the SHARC L1 memory with reduced latency, as these accesses do not go through the main system fabric. These accesses are arbitrated between both the SHARC+ core completer ports. The SHARC+ core can also access the FIR/IIR accelerator MMR registers directly.

Additional Features

The enhanced ISA/VISA of the ADSP-2159x/ADSP-SC59x processors provides a memory barrier instruction for data synchronization, exclusive data access support for multicore data sharing, and exclusive data access to enable multiprocessor programming. To enhance the reliability of the application, L1 data RAMs support parity error detection for every byte, and illegal opcodes are also detected (core interrupts flag both errors). Requester ports of the core also detect failed external accesses.

SYSTEM INFRASTRUCTURE

The following sections describe the system infrastructure of the ADSP-2159x/ADSP-SC59x processors.

System L2 Memory

A system L2 SRAM memory of up to 16 Mb (2 MB) is available to both SHARC+ cores, the Arm Cortex-A5 core, and the system DMA channels (see Table 3). The L2 SRAM block is subdivided into up to eight banks to support concurrent access to the L2 memory ports. Memory accesses to the L2 memory space are multicycle accesses by both the Arm Cortex-A5 and SHARC+ cores.

The memory space is used for various situations including

- Arm Cortex-A5 to SHARC+ core data sharing and intercore communications
- Accelerator and peripheral sources and destination memory to avoid accessing data in the external memory
- A location for DMA descriptors
- Storage for additional data for either the Arm Cortex-A5 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 the System Memory Protection Unit (SMPU) section for options in limiting access by specific cores and DMA requesters.

The Arm Cortex-A5 core has an L1 instruction and data cache, each of which is 32 kB in size. The core also has an L2 cache controller of 256 kB. 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 4 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 7 kb of one time programmable (OTP) memory that is memory-map accessible. This memory can be programmed with custom keys and supports secure boot and secure operation.

I/O Memory Space

Mapped I/Os include SPI2/OSPI0 memory address space (see Table 5).

SYSTEM MEMORY MAP

Table 2. L1 Block 0, Block 1, Block 2, and Block 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 (192 KB)	0x00048000–0x0004DFFF	0x00090000–0x00097FFF	0x00090000–0x0009BFFF	0x00120000–0x00137FFF	0x00240000–0x0026FFFF
L1 Block 1 SRAM (192 KB)	0x00058000–0x0005DFFF	0x000B0000–0x000B7FFF	0x000B0000–0x000BBFFF	0x00160000–0x00177FFF	0x002C0000–0x002EFFFF
L1 Block 2 SRAM (128 KB)	0x00060000–0x00063FFF	0x000C0000–0x000C5554	0x000C0000–0x000C7FFF	0x00180000–0x0018FFFF	0x00300000–0x0031FFFF
L1 Block 3 SRAM (128 KB)	0x00070000–0x00073FFF	0x000E0000–0x000E5554	0x000E0000–0x000E7FFF	0x001C0000–0x001CFFFF	0x00380000–0x0039FFFF

Table 3. L2 Memory Addressing Map

Memory	Byte Address Space Arm Cortex-A5: Data Access and Instruction Fetch SHARC+: Data Access	Normal Word Address Space SHARC+ Data Access	VISA Address Space SHARC+ Instruction Fetch	ISA Address Space SHARC+ Instruction Fetch
L2 Boot ROM0[1]	Arm: 0x00000000–0x0000FFFF SHARC+/DMA: 0x20200000–0x2020FFFF	0x08080000–0x08083FFF	0x00C20000–0x00C27FFF	0x00520000–0x00522AA9
L2 RAM (2 MB)[2]	0x20000000–0x201FFFFF	0x08000000–0x0807FFFF	0x00B00000–0x00BFFFFF	0x00580000–0x005D5554
L2 RAM (1 MB)[2]	0x20100000–0x201FFFFF	0x08040000–0x0807FFFF	0x00B80000–0x00BFFFFF	0x005AAAAB–0x005D5554
L2 Boot ROM1	0x20210000–0x2021FFFF	0x08084000–0x08087FFF	0x00C00000–0x00C07FFF	0x00500000–0x00502AA9
L2 Boot ROM2	0x20220000–0x2022FFFF	0x08088000–0x0808BFFF	0x00C40000–0x00C47FFF	0x00540000–0x00542AA9

[1] For ADSP-SC59x products, the L2 Boot ROM0 byte address space is 0x00000000–0x0000FFFF.
[2] All L2 RAM blocks are subdivided into 256 KB banks. All models feature 2 MB of L2 SRAM (8 banks) except for the ADSP-21591/ADSP-SC591 models, which feature 1 MB (4 banks). See Planned Automotive Production Products, Planned Production Products, and Ordering Guide.

Table 4. SHARC+® L1 Memory in Multiprocessor Space

		Memory Block	Byte Address Space Arm Cortex-A5 and SHARC+	Normal Word Address Space SHARC+
L1 memory of SHARC1 in multiprocessor space	Address via Completer 1 Port	Block 0	0x28240000–0x2826FFFF	0x0A090000–0x0A09BFFF
		Block 1	0x282C0000–0x282EFFFF	0x0A0B0000–0x0A0BBFFF
		Block 2	0x28300000–0x2831FFFF	0x0A0C0000–0x0A0C7FFF
		Block 3	0x28380000–0x2839FFFF	0x0A0E0000–0x0A0E7FFF
L1 memory of SHARC2 in multiprocessor space	Address via Completer 1 Port	Block 0	0x28A40000–0x28A6FFFF	0x0A290000–0x0A29BFFF
		Block 1	0x28AC0000–0x28AEFFFF	0x0A2B0000–0x0A2BBFFF
		Block 2	0x28B00000–0x28B1FFFF	0x0A2C0000–0x0A2C7FFF
		Block 3	0x28B80000–0x28B9FFFF	0x0A2E0000–0x0A2E7FFF

Table 5. Memory Map of Mapped I/Os[1]

	Byte Address Space Arm Cortex-A5: Data Access and Instruction Fetch SHARC+: Data Access	Normal Word Address Space SHARC+ Data Access	VISA Address Space SHARC+ Instruction Fetch	ISA Address Space SHARC+ Instruction Fetch
SPI2/OSPI0 Memory (512 MB)[2]	0x60000000–0x600FFFFF	0x04000000–0x07FFFFFF	0x00F80000–0x00FFFFFF	0x00780000–0x007AAAAA
	0x60100000–0x602FFFFF		Not available	0x007AAAAB–0x007FFFFF
	0x60300000–0x603FFFFF		0x00E80000–0x00EFFFFF	0x00680000–0x006AAAAA
	0x60400000–0x605FFFFF		Not available	0x006AAAAB–0x006FFFFF
	0x60600000–0x6FFFFFFF		Not available	Not available
	0x70000000–0x7FFFFFFF	Not available	Not available	Not available

[1] The Arm Cortex-A5 can access the entire byte address space. The SHARC+ VISA/ISA address space for instruction fetch and the normal word address space for data access do not cover the entire byte address space.
[2] For the ADSP-SC592/SC594/21594 processors, the SPI2/OSPI0 memory-mapped I/O space can be configured to be shared between the OSPI0 and SPI2 peripherals using the REG_SCB5_REMAP.REMAP field. See the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for details.

Table 6. DMC Memory Map[1]

	Byte Address Space Arm Cortex-A5: Data Access and Instruction Fetch SHARC+: Data Access	Normal Word Address Space SHARC+ Data Access	VISA Address Space SHARC+ Instruction Fetch	ISA Address Space SHARC+ Instruction Fetch
DMC0 (1 GB)	0x80000000–0x805FFFFF	0x10000000–0x17FFFFFF	Not applicable	0x00400000–0x004FFFFF
	0x80600000–0x809FFFFF		Not applicable	Not applicable
	0x80A00000–0x80FFFFFF		0x00800000–0x00AFFFFF	Not applicable
	0x81000000–0x9FFFFFFF		Not applicable	Not applicable
	0xA0000000–0xBFFFFFFF	Not applicable	Not applicable	Not applicable

[1] The Arm Cortex-A5 can access the entire byte address space. The SHARC+ VISA/ISA address space for instruction fetch and the normal word address space for data access do not cover the entire byte address space.

System Crossbars (SCBs)

The system crossbars (SCBs) are the fundamental building blocks of a switch fabric style for on-chip system bus interconnection. The SCBs connect system bus requesters to system bus completers, providing concurrent data transfer between multiple bus requesters and multiple bus completers. A hierarchical model—built from multiple SCBs—provides a power and area efficient system interconnection.

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 requesters to access bus completers 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: the source channel and the destination channel.

All DMA channels 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 DMA control registers automatically update with 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 autorefreshing buffer that interrupts when each buffer becomes full
- A similar circular buffer that interrupts on fractional buffers, such as at the halfway point
- The 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
- The 1D DMA uses a linked list of four-word descriptor sets containing a link pointer, an address, a length, and a configuration
- The 2D DMA uses an array of one-word descriptor sets, specifying only the base DMA address
- The 2D DMA uses a linked list of multiword descriptor sets, specifying all configurable parameters

Memory Direct Memory Access (MDMA)

The processor supports various memory direct memory access (MDMA) operations, including,

- Enhanced bandwidth MDMA channels with cyclic redundancy check (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, runs on SYSCLK)

Extended Memory DMA

Extended memory DMA supports various operating modes, such as delay line (which allows processor reads and writes to external delay line buffers and to the external memory), with limited core interaction and scatter/gather DMA (writes to and from noncontiguous memory blocks).

Cyclic Redundancy Check (CRC) Protection

The cyclic redundancy check (CRC) protection modules allow system software to calculate the signature of code, data, or both in memory, the content of memory-mapped registers, or periodic communication message objects. Dedicated hardware circuitry compares the signature with precalculated values and triggers appropriate fault events.

For example, the system software initiates the signature calculation of the entire memory contents every 100 ms and compares this with expected, precalculated 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 can be optionally forwarded to the destination channel (memory transfer mode). The main features of the CRC peripheral are as follows:

- Memory scan mode
- Memory transfer mode
- Data verify mode
- Data fill mode
- User-programmable CRC32 polynomial
- Bit and byte mirroring option (endianness)
- Fault and error interrupt mechanisms
- 1D and 2D fill block to initialize an array with constants
- 32-bit CRC signature of a block of memory or an 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 a higher priority event takes precedence over servicing a lower priority event.

The processors provide support for four different types of events:

- An emulation event causes the processors to enter emulation mode, allowing command and control of the processors through the JTAG interface.
- A reset event resets the processors.
- An exception event occurs synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions triggered by the SHARC+ core, such as data alignment (SIMD or long word) or compute violations (fixed or floating point) and illegal instructions, cause core exceptions. Conditions triggered by the SEC, such as error correcting code (ECC), parity, watchdog, or system clock, cause system exceptions.
- An interrupt event occurs asynchronously to program flow. The interrupts are caused by input signals, timers, and other peripherals, as well as by an explicit software instruction.

System Event Controller (SEC)

Each SHARC+ core event controller receives interrupt requests from the system event controller (SEC). The SEC features include the following:

- Comprehensive system event source management, including interrupt enable, fault enable, priority, core mapping, and source grouping
- A distributed programming model where each system event source control and all status fields are independent of each other
- Determinism where all system events have the same propagation delay and provide unique identification of a specific system event source
- A completer control port that provides access to all SEC registers for configuration, status, and interrupt and fault services
- Global locking that 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 trigger routing unit (TRU) provides system level sequence control without core intervention. The TRU maps trigger generators to trigger receivers. Trigger receivers 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-2159x/ADSP-SC59x processors.

Arm TrustZone

The ADSP-SC59x processors provide TrustZone technology that is integrated into the Arm Cortex-A5 processors. The TrustZone technology enables a secure state that is extended throughout the system fabric.

Cryptographic Hardware Accelerators

The ADSP-2159x/ADSP-SC59x processors support standards-based hardware accelerated encryption, decryption, authentication, and true random number generation.

Support for the hardware accelerated cryptographic ciphers includes the following:

- 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 hardware accelerated hash functions includes the following:

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

Public key accelerator (PKA) is available to offload computation intensive public key cryptography operations.

Both a hardware-based nondeterministic random number generator and pseudorandom number generator are available.

Secure boot is also available with 224-bit and 256-bit elliptic curve digital signatures ensuring integrity and authenticity of the boot stream. Optionally, ensuring confidentiality through AES-128 encryption is available.

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

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.

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 can choose and configure the protected peripherals as well as configure which of the six system MMR requesters (Arm Cortex-A5, two SHARC+ cores, two memory DMA, and Arm® 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 nonsecure as well as block access to secure resources from nonsecure requesters.

System Memory Protection Unit (SMPU)

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-2159x/ADSP-SC59x processors for each memory space, except for SHARC L1.

The SMPU is also part of the security infrastructure. It allows the user to protect against arbitrary read and/or write transactions and allows regions of memory to be defined as secure and prevent nonsecure requesters from accessing those memory regions.

SECURITY FEATURES DISCLAIMER

Analog Devices does not guarantee that the Security Features described herein provide 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.

SAFETY FEATURES

The ADSP-2159x/ADSP-SC59x processors are designed to support functional safety applications. Whereas the level of safety is mainly dominated by the system concept, the following primitives are provided by the processors to build a robust safety concept.

Multiparity Bit Protected SHARC+ Core L1 Memories

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

Parity Protected Arm L1 Cache

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

Error Correcting Code (ECC) Protected L2 Memories

Error correcting code (ECC) corrects single event upsets. A single error correct/double error detect (SEC/DED) code protects the L2 memory. By default, ECC is enabled, but it can be disabled on a per bank basis. Single-bit errors correct transparently. If enabled, dual-bit errors can issue a system event or fault. ECC protection is fully transparent to the user, even if L2 memory is read or written by 8-bit or 16-bit entities.

Parity and ECC Protected Peripheral Memories

Parity protection is added to the following peripheral memories:

- ASRC
- IIR
- FIR
- USB
- CRYPTO
- EMAC
- MLB
- TRACE

CAN FD memory is ECC protected.

Cyclic Redundancy Check (CRC) Protected Memories

Whereas parity bit and ECC protection mainly protect against random soft errors in L1 and L2 memory cells, the CRC engines can protect against systematic errors (pointer errors) and static content (instruction code) of L1, L2, and even Level 3 (L3) memories (DDR3, DDR3L). The processors feature four CRC engines that are embedded in the memory to memory DMA controllers.

CRC checksums can be calculated or compared automatically during memory transfers. Alternatively, single 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 16 general-purpose (GP) 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 detect undesired toggling or lack of toggling of system level signals.

System Event Controller (SEC)

Besides system events, the system event controller (SEC) further supports fault management, including fault action configuration as timeout, internal indication by system interrupt, or external indication through the $\overline{\text{SYS\_FAULT}}$ pin and system reset.

Memory Error Controller (MEC)

The memory error controller (MEC) manages memory parity/ECC errors and warnings from the cores and peripherals and sends out interrupts and triggers.

PROCESSOR PERIPHERALS

The following sections describe the peripherals of the ADSP-2159x/ADSP-SC59x processors.

Dynamic Memory Controller (DMC)

The 16-bit dynamic memory controller (DMC) interfaces to

- DDR3 (JESD79-3), 512 Mb to 8 Gb
- DDR3L (JESD79-3-1A), 512 Mb to 8 Gb

See Table 6 for the DMC memory map.

Digital Audio Interface (DAI)

The processors support two identical digital audio interface (DAI) units. The DAI can connect various peripherals to any of the DAI pins.

The application code makes these connections using the signal routing unit (SRU), shown in Figure 1.

The SRU is a matrix routing unit (or group of multiplexers) that enables the peripherals provided by each DAI instance to interconnect under software control. This functionality allows easy use of the DAI associated peripherals for a 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 (SPORTs, ASRC, S/PDIF, and PCG). DAI Pin Buffer 20 and DAI Pin Buffer 19 can change the polarity of the input signals.

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

See the Digital Audio Interface (DAI) chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for complete information on the use of the DAIs and SRUs.

DAI Routing Unit (DRU)

The DAI routing unit (DRU) provides flexibility when routing signals across the two DAI units. All DAI0 SRU source signals are available as source signals for the DAI1 SRU, and all DAI1 SRU source signals are available as source signals for the DAI0 SRU.

Serial Port (SPORT)

The processors feature eight synchronous serial ports (SPORTs), providing an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices. These devices include Analog Devices AD19xx and ADAU19xx families of audio codecs, analog-to-digital converters (ADCs) and digital-to-analog converters (DACs). Two data lines, a clock, and a frame sync comprise a SPORT half. The data lines can be programmed to either transmit or receive data, and each data line has a dedicated DMA channel.

An individual SPORT module consists of two independently configurable SPORT halves with identical functionality. Two bidirectional data lines—primary (0) and secondary (1)—are available per SPORT half and are configurable as either transmitters or receivers. Therefore, each SPORT half permits two unidirectional streams into or out of the same SPORT. This bidirectional functionality provides greater flexibility for serial communications. For full-duplex configuration, one half SPORT provides two transmit data signals, and the other half SPORT provides two receive data signals. The frame sync and clock are shared.

Serial ports operate in the following six modes:

- Standard DSP serial mode
- Multichannel time division multiplexing (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 eight ASRC blocks. The ASRC provides up to 140 dB signal-to-noise ratio (SNR). The ASRC block performs synchronous or asynchronous sample rate conversion across independent stereo channels, without using internal processor resources. The ASRC blocks can also be configured to operate together to convert multichannel audio data without phase mismatches. Finally, the ASRC can 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 Format (S/PDIF) is a standard audio data transfer format that allows the transfer of digital audio signals from one device to another. There are two S/PDIF transmit/receive blocks on the processor. The digital audio interface carries three types of information: audio data, nonaudio data (compressed data), and timing information.

The S/PDIF interface supports one stereo channel or compressed audio streams. The S/PDIF transmitter and receiver are AES3 compliant and support the sample rate from 24 kHz to 192 kHz. The S/PDIF receiver supports professional jitter standards.

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 SRU. They can come from various 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 eight units located in the two DAI blocks. The PCG can generate a pair of signals (clock and frame sync) derived from a clock input signal (CLKIN, SCLK0, or 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.

Pulse Density Modulation (PDM) Microphone Interface

The pulse density modulation (PDM) interface is used to convert digital PDM microphone data to I^2S/TDM format. The microphone data in I^2S/TDM format is then routed internally to the serial port/ASRC or externally via the DAI pins. The PDM microphone inputs include an internal decimation filter. Up to eight PDM microphones can be connected to the two dedicated digital microphone interfaces (one per DAI). Each PDM interface consists of one clock line and two data lines. Two microphones can share a single data line and be used along with a clock line to create a dual-input microphone port. Two dual-input lines can share a single clock line to support four microphone inputs.

Enhanced Parallel Peripheral Interface (EPPI)

The processors provide an enhanced parallel peripheral interface (EPPI) that supports data widths up to 24 bits. The EPPI supports direct connection to thin film transistor (TFT) LCD panels, parallel ADCs and DACs, video encoders and decoders, image sensor modules, and other general-purpose peripherals.

The features supported in the EPPI module include the following:

- Programmable data length of 8 bits, 10 bits, 12 bits, 14 bits, 16 bits, 18 bits, and 24 bits per clock
- Various framed, nonframed, 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 decoding

- Optional packing and unpacking of data to/from 32 bits from/to 8 bits, 16 bits, and 24 bits. If packing/unpacking is enabled, configure endianness to change the order of packing/unpacking of bytes or words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various deinterleaving/interleaving modes for receiving or transmitting 4:2:2 YCrCb data
- Configurable LCD data enable output available on Frame Sync 3

Universal Asynchronous Receiver/Transmitter (UART) Ports

The processors provide four full-duplex universal asynchronous receiver/transmitter (UART) ports, 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 as well as no parity, even parity, or odd parity.

Optionally, an additional address bit can be transferred to interrupt only addressed nodes in multidrop 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 first in, first out (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 interframe space.

Serial Peripheral Interface (SPI) Ports

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

The baseline SPI peripheral is a synchronous, 4-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 extra 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. DMA mode allows for transferring several words with minimal central processing unit (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 (SPI_RDY), which flexibly controls the transfers.

The baud rate and clock phase and polarities of the SPI port are programmable. The port has integrated DMA channels for both transmit and receive data streams.

Octal Serial Peripheral Interface (OSPI) Port

The octal serial peripheral interface (OSPI) port provides an increased external memory data bus width (up to eight bits in parallel). The OSPI port supports dual data rate (DDR) modes of operation, which enable the transfer of up to 16 bits of data each clock cycle. The OSPI port provides overall data throughput and performance improvement, including faster boot time.

Features of the OSPI port include:

- Support for single-, dual-, quad-, or octal-I/O transfers
- Multiple modes of operation including direct and software triggered instruction generator (STIG)
- Support for execute in place (XIP): continuous mode
- Programmable page and block sizes
- Programmable write protected regions
- Programmable memory timing
- Support for DDR commands
- Support for PHY mode of operation to enable high speed transfers
- Support for DQS to increase robustness of data sampling at higher speeds

Link Port (LP)

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

Link ports can operate in reduced pin mode, thereby reducing the number of pins required to interface between two processors. For example, two processors can be connected using the link port in 4-bit single data rate (SDR) and dual data rate (DDR) modes.

Ethernet Media Access Controller (EMAC)

The processor features an ethernet media access controller (EMAC): 10/100/1000 AVB Ethernet with precision time protocol (IEEE 1588).

The processors can directly connect to a network through embedded fast EMAC that supports 10Base-T (10 Mb/sec), 100Base-T (100 Mb/sec) and 1000Base-T (1 Gb/sec) operations.

Some standard features of the EMAC are as follows:

- Support of MII/RMII/RGMII protocols for external PHYs
- Full-duplex and half-duplex modes
- Media access management (in half-duplex operation)
- Flow control
- Station management, including the generation of MDC/MDIO frames for read/write access to PHY registers

Some advanced features of the EMAC include the following:

- Automatic checksum computation of IP header and IP payload fields of receive 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
- Transmit DMA support for separate descriptors for MAC header and payload fields 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 bridging (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, transmit and receive FIFO for AVB latency class
- Programmable control to route received VLAN tagged non AV packets to channels or queues

Precision Time Protocol (PTP) 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 include the following:

- 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, and external clock)
- Programmable pulse per second (PPS) output
- Auxiliary snapshot to time stamp external events

Controller Area Network with Flexible Data-Rate (CAN FD)

There are two controller area network (CAN) modules. A CAN controller implements the CAN with flexible data-rate (CAN FD) and the CAN 2.0B protocol supporting both standard and extended message frames and long payloads up to 64 bytes, transferred at rates of up to 8 Mbps. 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 the capability to communicate reliably over a network. This is because the protocol incorporates CRC checking, message error tracking, and fault node confinement.

The CAN FD controller offers the following features:

- Flexible mailboxes configurable to store 0 to 8, 16, 32, or 64 bytes
- Dedicated receiver masks for each mailbox
- Flexible message buffers up to 64 buffers of 8 bytes length each, configurable as receive or transmit
- Programmable transmission priority scheme
- Transceiver delay compensation when transmitting CAN FD messages at faster data rates
- Memory read accesses error detection and correction

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

Timers

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

General-Purpose (GP) Timers (TIMER)

There is one general-purpose (GP) timer unit, providing 16 GP programmable timers. Each timer has an external pin that can be configured as 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 TM_TMR[n] pins, an external TM_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 autobaud 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 can also be started and stopped by any trigger generator without core intervention. While the ADSP-21591 and ADSP-21593 processors feature 16 GP timers, timer units TIMER10-TIMER15 have no external pins and can only be configured for internal modes of operation.

Watchdog Timer (WDT)

Three on-chip software watchdog timers (WDT) can be used by the Arm Cortex-A5 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, protecting the system from remaining in an unknown state where software that normally resets the timer stops running due to an external noise condition or software error.

General-Purpose Counters (CNT)

A 32-bit general-purpose counter (CNT) 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 controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can input the push button signal of thumbwheel devices. All three CNT0 pins have a programmable debouncing circuit.

Internal signals forwarded to a GP timer enable the 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 housekeeping analog-to-digital converter (HADC) provides a general-purpose, multichannel, successive approximation ADC. The following baseline HADC features apply to all models:

- 12-bit ADC core with built in sample and hold
- Throughput rates up to 1 MSPS
- Single external reference with analog inputs between 0 V and 1.8 V
- Selectable ADC clock frequency including the ability to program a prescaler
- Adaptable conversion type; allows single or continuous conversion with option of autoscan
- Four single-ended input channels
- Autosequencing capability with up to four autoconversions in a single session. Each conversion can be programmed to select one to four input channels.
- Four data registers (individually addressable) to store conversion values

For the ADSP-SC592 and ADSP-SC594 processors, the following features extend the baseline features above:

- Total of eight single-ended input channels that can be further extended to 15 channels by connecting the HADC_MUX_PIN_NAME pin(s) to an external channel multiplexer
- Autosequencing capability with up to a total of eight autoconversions in a single session. Each conversion can be programmed to select one to fifteen input channels.
- 16 data registers (individually addressable) to store conversion values

USB 2.0 High Speed (HS) On the Go (OTG) Controller

The USB supports high speed/full speed/low speed (HS/FS/LS) USB2.0 on the go (OTG) and UTMI+ low pin interface (USBC).

The USB 2.0 OTG dual-role device controller provides a low cost connectivity solution 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 OTG supplement to the USB 2.0 specification.

The USB controller does not have an integrated on-chip PHY and must connect to an external PHY on the board through an USBC 8-bit interface supported by the USB controller.

Media Local Bus (MediaLB)

The automotive model has a Microchip MediaLB (MLB) device interface that 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. The MLB 3-pin configuration supports speeds up to 1024 × FS. The MLB 6-pin configuration supports speed of 2048 × FS. The MLB also supports up to 64 logical channels with up to 468 bytes of data per MLB frame.

The MLB interface supports MOST25, MOST50, and MOST150 data rates and operates in device mode only.

2-Wire Controller Interface (TWI)

The processors include six 2-wire interface (TWI) modules that provide 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 controller and target 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 400 kbps. The TWI interface pins are compatible with 3.3 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 manipulating the port control, status, and interrupt registers:

- The GPIO direction control register specifies the direction of each individual GPIO pin as input or output.
- GPIO control and status registers have a write-one-to-modify mechanism that 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, whereas 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 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. Eight system level interrupt channels (PINT0–PINT7) 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) are 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 functionality 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. Most control registers feature multiple MMR address entries to write one to set or write one to clear them individually.

Core Flags I/O Pins

The processor features 32 flag I/O pins (16 per SHARC+ core), which allow for external control and monitoring of the SHARC+ core FLAGS register. User code can write to bits in this register to be driven to pins configured as outputs, and code execution can be made conditional based on the settings of the pins configured as inputs.

SYSTEM ACCELERATION

The following sections describe the system acceleration blocks of the ADSP-2159x/ADSP-SC59x processors.

Finite Impulse Response (FIR) Accelerator

The finite impulse response (FIR) accelerator consists of a 1024 word coefficient memory, a 1024 word deep delay line for the data, and four multiplier-accumulator (MAC) units. A controller manages the accelerator. The FIR accelerator runs at the SHARC core clock frequency. The FIR accelerator can access all memory spaces and can run concurrently with the other accelerators on the processor.

Infinite Impulse Response (IIR) Accelerator

The infinite impulse response (IIR) 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 SHARC core clock frequency. The IIR accelerator can access all memory spaces and run concurrently with the other accelerators on the processor.

Note: There are four IIR accelerators per SHARC core.

SYSTEM DESIGN

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

Clock Management

The processors provide three operating modes, each with a different performance and power profile. Control of clocking to each of the processor peripherals 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 default values and functional units are idle. Exiting a full system reset begins with the core 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 put the system into an undefined state or cause resources to stall. This requirement is particularly important when the core resets (programs must ensure that there is no pending system activity involving the core when it is reset).

From a system perspective, reset is defined by both the reset target and the reset source.

The reset target is defined as the following:

- System reset—all functional units except the RCU are set to default states.
- Hardware reset—all functional units are set to default states without exception. History is lost.
- Core only reset—affects the core only. When in reset state, the core is not accessed by any bus requester.

The reset source is defined as the following:

- System reset—can 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 SEC, TRU, or emulator inputs.

- Hardware reset—the $\overline{\text{SYS\_HWRST}}$ input signal asserts active (pulled down).
- Core only reset—affects only the core. The core is not accessed by any bus requester when in reset state.
- Trigger request (peripheral).

Clock Generation Unit (CGU)

The ADSP-2159x/ADSP-SC59x processors support two independent PLLs. Each PLL is part of a clock generation unit (CGU). Each CGU can be either driven externally by the same clock source or driven by separate sources, thus providing 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 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 DDR3/DDR3L clock (DCLK), and the output clock (OCLK). For more information on clocking, see the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference.

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 to ensure smooth transitions from the current conditions to the new conditions.

System Crystal Oscillator

The processor can be clocked by an external crystal (see Figure 6), a sine wave input, or a buffered, shaped clock derived from an external clock oscillator. If using an external clock, it must be compatible with the $V_{IHCLKIN}$ and $V_{ILCLKIN}$ specifications and must not be halted, changed, or operated below the specified frequency during normal operation (see the Operating Conditions section). This signal is connected to the SYS_CLKINx pin of the processor. When using an external clock, the SYS_XTALx pin must be left unconnected. Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal can 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_CLKINx pin and the SYS_XTALx pin.

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 physical layout of the printed circuit board (PCB). The resistor value depends on the drive level specified by the crystal manufacturer. The user must verify the customized values based on careful investigations on multiple devices over the required temperature range.

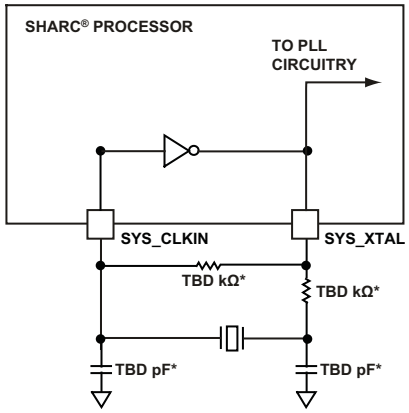


Figure 6. External Crystal Connection

Clock Distribution Unit (CDU)

The two clock generation units each provide outputs that feed a clock distribution unit (CDU). The clock outputs CLKO0–CLKO12 are connected to various targets. For more information, refer to the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference.

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 ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference 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[n] input pins. There are two categories of boot modes. In flash boot modes, the processors actively load data from serial memories. In external host boot modes, the processors receive data over a serial interface from an external host device.

The boot modes are shown in Table 7. These modes are implemented by the SYS_BMODE[n] bits of the reset configuration register and are sampled during power-on resets and software initiated resets.

In the ADSP-SC59x processors, the Arm Cortex-A5 (Core 0) controls the boot process, including loading all internal and external memory. Likewise, in the ADSP-2159x processors, the SHARC+ (Core 1) controls the boot function. The option for secure boot is available on all models.

Table 7. Boot Modes

SYS_BMODE[n] Setting	Boot Mode
000	No boot
001	SPI2 flash
010	External SPI2 host
011	External UART0 host
100	External LP0 host
101	Octal SPI flash
110	Reserved

Thermal Monitoring Unit (TMU)

The thermal monitoring unit (TMU) provides on-chip temperature measurement for 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 the following:

- On-chip temperature sensing
- Programmable over temperature and under temperature limits
- Programmable conversion rate
- Averaging feature available

Power Supplies

The processors have separate power supply connections for

- Internal (VDD_INT)
- External (VDD_EXT)
- External (VDD_REF)
- HADC/TMU (VDD_ANA)
- DMC (VDD_DMC)
- PLL (VDD_PLL)

All power supplies must meet the specifications provided in the Operating Conditions section. All external supply pins must be connected to the same power supply.

Power Management

As shown in Table 8, the processors support six different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions.

The power dissipated by a processor is largely a function of the 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.

Table 8. Power Domains

Power Domain	V_{DD} Range
All Internal Logic	V_{DD_INT}
DDR3/DDR3L	V_{DD_DMC}
HADC/TMU	V_{DD_ANA}
SYS_CLKIN0/1	V_{DD_REF}[1]
PLL0/1	V_{DD_PLL}
All Other I/O (Includes SYS, JTAG, and Ports Pins)	V_{DD_EXT}

[1] V_{DD_REF} requires a minimum of 10 nF and 100 nF decoupling capacitance to meet source/sink requirements.

Power-Up and Power-Down Sequencing

At all times (including during power-up/power-down sequencing), the VDD_REF, VDD_ANA, and VDD_EXT supplies must stay within the $V_{DELTA\_EXT\_REF}$ specification listed in the Operating Conditions table. SYS_XTAL0 and SYS_XTAL1 oscillations (SYS_CLKIN0 and SYS_CLKIN1) start when power is applied to the VDD_REF pins. The rising edge of $\overline{\text{SYS\_HWRST}}$ initiates the PLL locking sequence. The rising edge of $\overline{\text{SYS\_HWRST}}$ must occur after all voltage supplies and SYS_CLKIN0 and SYS_CLKIN1 oscillations are valid. For further details and information, see the Power-Up Reset Timing section.

Target Board JTAG Emulator Connector

The Analog Devices DSP tools product line of JTAG emulators uses the IEEE 1149.1 JTAG test access port of the processors to monitor and control the target board processor during emulation. The 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 JTAG interface ensures the emulator does not affect target system loading or timing.

For information on JTAG emulator operation, see the appropriate emulator hardware user's guide at SHARC Processors Software and Tools.

SYSTEM DEBUG

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

System Watchpoint Unit (SWU)

The system watchpoint unit (SWU) is a single module that connects to a single system bus and provides transaction monitoring. One SWU is attached to the bus going to each system completer. 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 (for example, interrupt and trigger) outputs.

Debug Access Port (DAP)

The debug access port (DAP) provides IEEE 1149.1 JTAG interface support through the 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, evaluation products, emulators, and a variety of software add-ins.

Integrated Development Environment

For C/C++ software writing and editing, code generation, and debug support, Analog Devices offers the CrossCore® Embedded Studio (CCES) integrated development environment (IDE).

CCES is based on the Eclipse framework. Supporting most Analog Devices processor families, CCES is the IDE of choice for processors, including multicore devices.

CCES 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 (BSPs). For more information, visit www.analog.com/cces.

EZ-KIT Evaluation Systems

For processor evaluation, Analog Devices provides EZ-KIT® evaluation systems (EV-SC594-EZKIT and EV-21593-EZKIT), which are bundles comprised of a System on Module (SOM) board (EV-SC594-SOM or EV-21593-SOM) and an EZ-KIT SOM carrier board (EV-SOMCRR-EZKIT).

SOM boards are small and low cost. The EV-SC594-SOM and EV-21593-SOM boards either have the ADSP-SC594 or ADSP-21593 processor, SDRAM and QSPI flash memories, JTAG debug connection, FTDI USB-to-UART, and USB power. SOM boards can be used alone or in combination with a SOM carrier board. SOM carrier boards have high speed connectors for the SOM, a comprehensive set of peripherals, and an on-board emulator. Each SOM carrier board also comes with a power supply.

The USB controller on the SOM carrier board connects to the USB port of the user's PC, enabling CCES to emulate the on-board processor in circuit. This permits users to download, execute, and debug programs. It also supports in-circuit programming of the on-board flash memory device to store user-specific boot code, enabling standalone operation.

Each EZ-KIT purchased includes an evaluation license for CCES. The CCES evaluation license type restricts CCES features to specific evaluation systems. With the full CCES license type (sold separately), engineers can develop software for any of the CCES-supported evaluation boards (including the SOM when used standalone or when connected to a different carrier board) or any custom system designed around supported Analog Devices processors. The full CCES license type also enables use of ICE-2000 or ICE-1000 emulators for higher performance debugging via JTAG.

When a SOM board is mounted to a SOM carrier board, embedded system evaluation can be performed. The EZ-KIT SOM carrier board provides an evaluation platform for various system peripherals including audio, S/PDIF, CAN, MLB, USB, 10/100/1000 Ethernet, and A^2B, as well as an OSPI flash memory. The EZ-KIT SOM carrier board also features the USB Debug Agent, which allows for debug and evaluation of the full system (including the processor/memory on the SOM module) without an emulator.

For further information, see:

- www.analog.com/cces
- www.analog.com/EV-21593-SOM
- www.analog.com/EV-SC594-SOM
- www.analog.com/EV-SOMCRR-EZKIT

Software Add-Ins for CCES

Analog Devices offers software add-ins which seamlessly integrate with CCES to extend the capabilities and reduce development time. Add-ins include BSPs 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 CCES IDE upon add-in installation.

Board Support Packages (BSPs) for Evaluation Hardware

Software support for the EZ-KIT evaluation systems 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 SOM product.

Middleware Packages

Analog Devices offers middleware add-ins such as real-time operating systems, file systems, USB stacks, and TCP/IP stacks. For more information, see the Operating Systems and Middleware page.

Algorithmic Modules

To speed development, Analog Devices offers add-ins that perform popular audio and video processing algorithms. These are available for use with CCES. For more information, visit the Design Center.

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 internal features of the processor via the 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 after 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 JTAG port of the DSP 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 Analog Devices JTAG Emulation Technical Reference (EE-68).

ADDITIONAL INFORMATION

This data sheet provides a general overview of the ADSP-2159x/ADSP-SC59x architecture and functionality. For detailed information on the core architecture and instruction set, refer to the SHARC+ Core Programming Reference.

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. A tool for viewing relationships between specific applications and related components is available at www.analog.com\circuits.

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

- 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

ADSP-2159x/ADSP-SC59x DETAILED SIGNAL DESCRIPTIONS

Table 9 provides a detailed description of each pin.

Table 9. ADSP-2159x/ADSP-SC59x Detailed Signal Descriptions

Signal Name	Direction	Description
C1_FLG[n]	InOut	**Core 1 FLAGS I/O n.** External pins associated with the core FLAGS register on SHARC+ core 1.
C2_FLG[n]	InOut	**Core 2 FLAGS I/O n.** External pins associated with the core FLAGS register on SHARC+ core 2.
CANFD_RX	Input	**Receive.** Typically an external CAN transceiver RX output.
CANFD_TX	Output	**Transmit.** Typically an external CAN transceiver 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. Count Down—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.** The digital applications interface (DAI0) connects various peripherals to any of the DAI0_PINxx pins. Programs make these connections using the signal routing unit (SRU/DRU). DRU allows routing of any signal across the DAIs.
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 applied to on the dynamic memory. Bank Address n also defines which mode registers (MR, EMR, EMR2, and/or EMR3) load 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.
DMC_CK	Output	**Clock.** Outputs DCLK to external dynamic memory.
$\overline{\text{DMC\_CK}}$	Output	**Clock (Complement).** Complement of DMC_CK.
DMC_CKE	Output	**Clock Enable.** Active high clock enables. Connects to the CKE input of the dynamic memory.
$\overline{\text{DMC\_CS[n]}}$	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.
DMC_LDQS	InOut	**Data Strobe for Lower Byte.** DMC_DQ07:DMC_DQ00 data strobe. Output with write data. Input with read data. Can be single-ended or differential depending on register settings.
$\overline{\text{DMC\_LDQS}}$	InOut	**Data Strobe for Lower Byte (Complement).** Complement of DMC_LDQS. Not used in single-ended mode.
DMC_ODT	Output	**On Die Termination.** Enables dynamic memory termination resistances when driven high (assuming the memory is properly configured). ODT is enabled or 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.**
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.
DMC_UDQS	InOut	**Data Strobe for Upper Byte.** DMC_DQ15:DMC_DQ08 data strobe. Output with write data. Input with read data. Can be single-ended or differential depending on register settings.
$\overline{\text{DMC\_UDQS}}$	InOut	**Data Strobe for Upper Byte (Complement).** Complement of DMC_UDQS. Not used in single-ended mode.

Table 9. ADSP-2159x/ADSP-SC59x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
DMC_VREF[n]	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 $\overline{\text{WE}}$ input of dynamic memory.
ETH_COL	Input	**MII Collision Detect.** Collision detect input signal valid only in MII.
ETH_CRS	Input	**MII Carrier Sense.** Asserted by the PHY when either the transmit or receive medium is not idle. Deasserted when both are idle. This signal is not used in RMII/RGMII modes.
ETH_MDC	Output	**Management Channel Clock.** Clocks the MDC input of the PHY for RMII/RGMII.
ETH_MDIO	InOut	**Management Channel Serial Data.** Bidirectional data bus for PHY control for RMII/RGMII.
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 enables, this signal is asserted based on the PPS mode selected. Otherwise, this signal is asserted every time the seconds counter is incremented.
ETH_REFCLK	Input	**Reference Clock.** Externally supplied Ethernet clock.
ETH_RXCLK_REFCLK	Input	**RXCLK (10/100/1000) or REFCLK (10/100).**
ETH_RXCTL_CRSRX_DV	InOut	**RXCTL (10/100/1000) or CRSRX_DV (10/100).** In RGMII mode, RXCTL multiplexes receive data valid and receiver error. In RMII mode, CRSRX_DV is carrier sense and receive data valid (CRS_DV), multiplexed on alternating clock cycles. In MII mode, CRSRX_DV is receive data valid (RX_DV), asserted by the PHY when the data on ETH_RXD[n] is valid.
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	Output	**TXCTL (10/100/1000) or TXEN (10/100).**
ETH_TXD[n]	Output	**Transmit Data n.** Transmit data bus.
ETH_TXEN	Output	**Transmit Enable.** When asserted, this signal indicates the data on ETH_TXD[n] is valid.
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_MUX[n]	Output	**Controls to External Multiplexer.** Allows additional input channels when connected to an external multiplexer.
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	Input	**Single Ended Clock.**
MLB_CLKN	Input	**Differential Clock (–).**
MLB_CLKOUT	Output	**Single Ended Clock Out.**

Table 9. ADSP-2159x/ADSP-SC59x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
MLB_CLKP	Input	**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 (+).**
OSPI_CLK	Output	**Clock Output.** SPI clock output.
OSPI_D2	InOut	**Data 2.** Transfers serial data in quad and octal mode.
OSPI_D3	InOut	**Data 3.** Transfers serial data in quad and octal mode.
OSPI_D4	InOut	**Data 4.** Transfers serial data in octal mode.
OSPI_D5	InOut	**Data 5.** Transfers serial data in octal mode.
OSPI_D6	InOut	**Data 6.** Transfers serial data in octal mode.
OSPI_D7	InOut	**Data 7.** Transfers serial data in octal mode.
OSPI_DQS	Input	**Data Strobe.** Data strobe input from an external flash device.
OSPI_MISO	InOut	**Master In, Slave Out.** Transfers serial data. Operates in the same direction as SPI_MOSI in dual, quad, and octal modes.
OSPI_MOSI	InOut	**Master Out, Slave Input.** Transfers serial data. Operates in the same direction as SPI_MISO in dual, quad, and octal modes.
$\overline{\text{OSPI\_SEL[n]}}$	Output	**Slave Select Output n.** Used in master mode to enable the desired slave.
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 chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for more details.
PPI_FS2	InOut	**Frame Sync 2 (VSYNC).** Behavior depends on EPPI mode. See the EPPI chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for more details.
PPI_FS3	InOut	**Frame Sync 3 (FIELD).** Behavior depends on EPPI mode. See the EPPI chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for more details.
P_[nn]	InOut	**Position n.** General-purpose input/output. See the GP Ports chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for more details.
SPI_CLK	InOut	**Clock.** Input in slave mode, output in master mode.
SPI_D2	InOut	**Data 2.** Transfers serial data in quad mode. Open-drain when ODM mode is enabled.
SPI_D3	InOut	**Data 3.** Transfers serial data in quad mode. Open-drain when ODM mode is enabled.
SPI_MISO	InOut	**Master In, Slave Out.** Transfers 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.** Transfers 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 or 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.

Table 9. ADSP-2159x/ADSP-SC59x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
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 multichannel transmit mode. It is asserted during enabled slots.
SPT_BCLK	InOut	**Channel B Clock.** Data and frame sync are driven or 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 multichannel 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 ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for more details.
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 an individual timer.
TM_CLK	Input	**Clock.** Provides an additional global time base for all GP timers.
TM_TMR[n]	InOut	**Timer n.** The 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 controller, clock input when target.
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.** Receives input. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
$\overline{\text{UART\_TX}}$	Output	**Transmit.** Transmits output. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
USBC_CLK	Input	**USBC Clock.**
USBC_DATA[n]	InOut	**USBC Data.**
USBC_DIR	Input	**USBC Data Bus Control.** Controls the direction of data bus.
USBC_NXT	Input	**USBC Next Data Control.**
USBC_STOP	Output	**USBC Stop Output Control.**

400-BALL HIGH PERIPHERAL COUNT (HPC) BGA SIGNAL DESCRIPTIONS

The processor pin definitions are shown in Table 10 for the 400-ball HPC BGA package. The columns in this table provide the following information:

- The signal name column includes the signal name for every pin and the GPIO multiplexed pin function, where applicable.
- The description column provides a descriptive name for each signal.
- The port column shows whether or not a signal is multiplexed with other signals on a GPIO port pin.
- The pin name column 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 GPIO pin).
- The DAI pins and their associated signal routing units (SRUs) connect inputs and outputs of the DAI peripherals (SPORT, ASRC, S/PDIF, and PCG). See the Digital Audio Interface (DAI) chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for complete information on the use of the DAI and SRUs.

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
C1_FLG00	SHARC+ Core 1 FLAGS I/O 0	A	PA_12
C1_FLG01	SHARC+ Core 1 FLAGS I/O 1	H	PH_02
C1_FLG02	SHARC+ Core 1 FLAGS I/O 2	B	PB_03
C1_FLG03	SHARC+ Core 1 FLAGS I/O 3	B	PB_02
C1_FLG04	SHARC+ Core 1 FLAGS I/O 4	I	PI_03
C1_FLG05	SHARC+ Core 1 FLAGS I/O 5	I	PI_04
C1_FLG06	SHARC+ Core 1 FLAGS I/O 6	F	PF_02
C1_FLG07	SHARC+ Core 1 FLAGS I/O 7	F	PF_01
C1_FLG08	SHARC+ Core 1 FLAGS I/O 8	E	PE_12
C1_FLG09	SHARC+ Core 1 FLAGS I/O 9	F	PF_09
C1_FLG10	SHARC+ Core 1 FLAGS I/O 10	F	PF_03
C1_FLG11	SHARC+ Core 1 FLAGS I/O 11	D	PD_03
C1_FLG12	SHARC+ Core 1 FLAGS I/O 12	F	PF_13
C1_FLG13	SHARC+ Core 1 FLAGS I/O 13	F	PF_12
C1_FLG14	SHARC+ Core 1 FLAGS I/O 14	G	PG_09
C1_FLG15	SHARC+ Core 1 FLAGS I/O 15	I	PI_05
C2_FLG00	SHARC+ Core 2 FLAGS I/O 0	I	PI_01
C2_FLG01	SHARC+ Core 2 FLAGS I/O 1	I	PI_02
C2_FLG02	SHARC+ Core 2 FLAGS I/O 2	F	PF_06
C2_FLG03	SHARC+ Core 2 FLAGS I/O 3	F	PF_07
C2_FLG04	SHARC+ Core 2 FLAGS I/O 4	F	PF_10
C2_FLG05	SHARC+ Core 2 FLAGS I/O 5	F	PF_11
C2_FLG06	SHARC+ Core 2 FLAGS I/O 6	G	PG_13
C2_FLG07	SHARC+ Core 2 FLAGS I/O 7	E	PE_11
C2_FLG08	SHARC+ Core 2 FLAGS I/O 8	F	PF_08
C2_FLG09	SHARC+ Core 2 FLAGS I/O 9	D	PD_14
C2_FLG10	SHARC+ Core 2 FLAGS I/O 10	D	PD_02
C2_FLG11	SHARC+ Core 2 FLAGS I/O 11	G	PG_12
C2_FLG12	SHARC+ Core 2 FLAGS I/O 12	F	PF_14
C2_FLG13	SHARC+ Core 2 FLAGS I/O 13	E	PE_13
C2_FLG14	SHARC+ Core 2 FLAGS I/O 14	G	PG_10
C2_FLG15	SHARC+ Core 2 FLAGS I/O 15	G	PG_11
CANFD0_RX	CANFD0 Receive	F	PF_15
CANFD0_TX	CANFD0 Transmit	G	PG_00
CANFD1_RX	CANFD1 Receive	G	PG_01
CANFD1_TX	CANFD1 Transmit	G	PG_02

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
CNT0_DG	CNT0 Count Down and Gate	B	PB_05
CNT0_UD	CNT0 Count Up and Direction	B	PB_03
CNT0_ZM	CNT0 Count Zero Marker	B	PB_04
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
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
DAI1_PIN01	DAI1 Pin 1	Not Muxed	DAI1_PIN01
DAI1_PIN02	DAI1 Pin 2	Not Muxed	DAI1_PIN02
DAI1_PIN03	DAI1 Pin 3	Not Muxed	DAI1_PIN03
DAI1_PIN04	DAI1 Pin 4	Not Muxed	DAI1_PIN04
DAI1_PIN05	DAI1 Pin 5	Not Muxed	DAI1_PIN05
DAI1_PIN06	DAI1 Pin 6	Not Muxed	DAI1_PIN06
DAI1_PIN07	DAI1 Pin 7	Not Muxed	DAI1_PIN07
DAI1_PIN08	DAI1 Pin 8	Not Muxed	DAI1_PIN08
DAI1_PIN09	DAI1 Pin 9	Not Muxed	DAI1_PIN09
DAI1_PIN10	DAI1 Pin 10	Not Muxed	DAI1_PIN10
DAI1_PIN11	DAI1 Pin 11	Not Muxed	DAI1_PIN11
DAI1_PIN12	DAI1 Pin 12	Not Muxed	DAI1_PIN12
DAI1_PIN13	DAI1 Pin 13	Not Muxed	DAI1_PIN13
DAI1_PIN14	DAI1 Pin 14	Not Muxed	DAI1_PIN14
DAI1_PIN15	DAI1 Pin 15	Not Muxed	DAI1_PIN15
DAI1_PIN16	DAI1 Pin 16	Not Muxed	DAI1_PIN16
DAI1_PIN17	DAI1 Pin 17	Not Muxed	DAI1_PIN17
DAI1_PIN18	DAI1 Pin 18	Not Muxed	DAI1_PIN18
DAI1_PIN19	DAI1 Pin 19	Not Muxed	DAI1_PIN19
DAI1_PIN20	DAI1 Pin 20	Not Muxed	DAI1_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

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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	$\overline{\text{DMC0\_CAS}}$
DMC0_CK	DMC0 Clock	Not Muxed	DMC0_CK
$\overline{\text{DMC0\_CK}}$	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
DMC0_LDQS	DMC0 Data Strobe for Lower Byte	Not Muxed	DMC0_LDQS
$\overline{\text{DMC0\_LDQS}}$	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	Not Muxed	$\overline{\text{DMC0\_RESET}}$
DMC0_RZQ	DMC0 External Calibration Resistor Connection	Not Muxed	DMC0_RZQ
DMC0_UDM	DMC0 Data Mask for Upper Byte	Not Muxed	DMC0_UDM
DMC0_UDQS	DMC0 Data Strobe for Upper Byte	Not Muxed	DMC0_UDQS
$\overline{\text{DMC0\_UDQS}}$	DMC0 Data Strobe for Upper Byte (Complement)	Not Muxed	$\overline{\text{DMC0\_UDQS}}$
DMC0_VREF0	DMC0 Voltage Reference	Not Muxed	DMC0_VREF0
$\overline{\text{DMC0\_WE}}$	DMC0 Write Enable	Not Muxed	$\overline{\text{DMC0\_WE}}$
ETH0_COL	EMAC0 MII Collision Detect	D	PD_07

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
ETH0_CRS	EMAC0 MII Carrier Sense	D	PD_02
ETH0_MDC	EMAC0 Serial Management Clock	H	PH_03
ETH0_MDIO	EMAC0 Serial Management Bidirectional Data	H	PH_04
ETH0_PTPAUXIN0	EMAC0 PTP Auxiliary Trigger Input 0	I	PI_02
ETH0_PTPAUXIN1	EMAC0 PTP Auxiliary Trigger Input 1	D	PD_05
ETH0_PTPAUXIN2	EMAC0 PTP Auxiliary Trigger Input 2	D	PD_03
ETH0_PTPAUXIN3	EMAC0 PTP Auxiliary Trigger Input 3	E	PE_09
ETH0_PTPCLKIN0	EMAC0 PTP Clock Input 0	I	PI_01
ETH0_PTPPPS0	EMAC0 Pulse Per Second Output 0	I	PI_04
ETH0_PTPPPS1	EMAC0 Pulse Per Second Output 1	I	PI_03
ETH0_PTPPPS2	EMAC0 Pulse Per Second Output 2	I	PI_05
ETH0_PTPPPS3	EMAC0 Pulse Per Second Output 3	I	PI_06
ETH0_RXCLK_REFCLK	EMAC0 RXCLK (10/100/1000) or REFCLK (10/100)	H	PH_07
ETH0_RXCTL_CRSRX_DV	EMAC0 RXCTL (RGMII) or CRS_DV (GMII) or RX_DV (MII)	H	PH_08
ETH0_RXD0	EMAC0 Receive Data 0	H	PH_05
ETH0_RXD1	EMAC0 Receive Data 1	H	PH_06
ETH0_RXD2	EMAC0 Receive Data 2	H	PH_11
ETH0_RXD3	EMAC0 Receive Data 3	H	PH_12
ETH0_RXERR	EMAC0 Receive Error	D	PD_06
ETH0_TXCLK	EMAC0 Transmit Clock	H	PH_14
ETH0_TXCTL_TXEN	EMAC0 TXCTL (10/100/1000) or TXEN (10/100)	H	PH_13
ETH0_TXD0	EMAC0 Transmit Data 0	H	PH_09
ETH0_TXD1	EMAC0 Transmit Data 1	H	PH_10
ETH0_TXD2	EMAC0 Transmit Data 2	H	PH_15
ETH0_TXD3	EMAC0 Transmit Data 3	I	PI_00
ETH1_CRS	EMAC1 Carrier Sense	F	PF_03
ETH1_MDC	EMAC1 Serial Management Clock	F	PF_02
ETH1_MDIO	EMAC1 Serial Management Bidirectional Data	F	PF_01
ETH1_REFCLK	EMAC1 Reference Clock	E	PE_11
ETH1_RXD0	EMAC1 Receive Data 0	E	PE_15
ETH1_RXD1	EMAC1 Receive Data 1	F	PF_00
ETH1_TXD0	EMAC1 Transmit Data 0	E	PE_13
ETH1_TXD1	EMAC1 Transmit Data 1	E	PE_14
ETH1_TXEN	EMAC1 Transmit Enable	E	PE_12
HADC0_EOC_DOUT	HADC0 End of Conversion	A	PA_11
HADC0_MUX0	HADC0 MUX0	E	PE_02
HADC0_MUX1	HADC0 MUX1	E	PE_04
HADC0_MUX2	HADC0 MUX2	E	PE_03
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

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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	$\overline{\text{JTG\_TRST}}$
LP0_ACK	LP0 Acknowledge	B	PB_04
LP0_CLK	LP0 Clock	B	PB_06
LP0_D0	LP0 Data 0	B	PB_07
LP0_D1	LP0 Data 1	B	PB_08
LP0_D2	LP0 Data 2	B	PB_09
LP0_D3	LP0 Data 3	B	PB_10
LP0_D4	LP0 Data 4	B	PB_11
LP0_D5	LP0 Data 5	B	PB_12
LP0_D6	LP0 Data 6	B	PB_13
LP0_D7	LP0 Data 7	B	PB_14
LP1_ACK	LP1 Acknowledge	B	PB_02
LP1_CLK	LP1 Clock	C	PC_07
LP1_D0	LP1 Data 0	B	PB_15
LP1_D1	LP1 Data 1	C	PC_00
LP1_D2	LP1 Data 2	C	PC_01
LP1_D3	LP1 Data 3	C	PC_02
LP1_D4	LP1 Data 4	C	PC_03
LP1_D5	LP1 Data 5	C	PC_04
LP1_D6	LP1 Data 6	C	PC_05
LP1_D7	LP1 Data 7	C	PC_06
MLB0_CLK	MLB0 Single-Ended Clock	B	PB_02
MLB0_CLKN	MLB0 Differential Clock (–)	Not Muxed	MLB0_CLKN
MLB0_CLKOUT	MLB0 Clock Single-Ended Clock Out	F	PF_05
MLB0_CLKP	MLB0 Differential Clock (+)	Not Muxed	MLB0_CLKP
MLB0_DAT	MLB0 Single-Ended Data	B	PB_00
MLB0_DATN	MLB0 Differential Data (–)	Not Muxed	MLB0_DATN
MLB0_DATP	MLB0 Differential Clock (+)	Not Muxed	MLB0_DATP
MLB0_SIG	MLB0 Single-Ended Signal	B	PB_01
MLB0_SIGN	MLB0 Differential Signal (–)	Not Muxed	MLB0_SIGN
MLB0_SIGP	MLB0 Differential Signal (+)	Not Muxed	MLB0_SIGP
OSPI0_CLK	OSPI0 Clock	C	PC_08
OSPI0_D2	OSPI0 Data 2	A	PA_02
OSPI0_D3	OSPI0 Data 3	A	PA_03
OSPI0_D4	OSPI0 Data 4	D	PD_00
OSPI0_D5	OSPI0 Data 5	C	PC_15
OSPI0_D6	OSPI0 Data 6	A	PA_08
OSPI0_D7	OSPI0 Data 7	C	PC_13
OSPI0_DQS	OSPI0 Data Strobe	D	PD_04
OSPI0_MISO	OSPI0 Master In, Slave Out	C	PC_12
OSPI0_MOSI	OSPI0 Master Out, Slave In	C	PC_11
$\overline{\text{OSPI0\_SEL1}}$	OSPI0 Slave Select Output 1	A	PA_05
$\overline{\text{OSPI0\_SEL2}}$	OSPI0 Slave Select Output 2	I	PI_05
$\overline{\text{OSPI0\_SEL3}}$	OSPI0 Slave Select Output 3	G	PG_12

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
$\overline{\text{OSPI0\_SEL4}}$	OSPI0 Slave Select Output 4	G	PG_13
PPI0_CLK	EPPI0 Clock	E	PE_04
PPI0_D00	EPPI0 Data 0	E	PE_05
PPI0_D01	EPPI0 Data 1	E	PE_06
PPI0_D02	EPPI0 Data 2	E	PE_07
PPI0_D03	EPPI0 Data 3	E	PE_08
PPI0_D04	EPPI0 Data 4	E	PE_09
PPI0_D05	EPPI0 Data 5	E	PE_10
PPI0_D06	EPPI0 Data 6	D	PD_01
PPI0_D07	EPPI0 Data 7	D	PD_04
PPI0_D08	EPPI0 Data 8	D	PD_05
PPI0_D09	EPPI0 Data 9	D	PD_10
PPI0_D10	EPPI0 Data 10	D	PD_11
PPI0_D11	EPPI0 Data 11	D	PD_12
PPI0_D12	EPPI0 Data 12	D	PD_13
PPI0_D13	EPPI0 Data 13	D	PD_14
PPI0_D14	EPPI0 Data 14	D	PD_15
PPI0_D15	EPPI0 Data 15	E	PE_00
PPI0_D16	EPPI0 Data 16	C	PC_08
PPI0_D17	EPPI0 Data 17	C	PC_09
PPI0_D18	EPPI0 Data 18	C	PC_10
PPI0_D19	EPPI0 Data 19	C	PC_11
PPI0_D20	EPPI0 Data 20	C	PC_12
PPI0_D21	EPPI0 Data 21	C	PC_13
PPI0_D22	EPPI0 Data 22	C	PC_14
PPI0_D23	EPPI0 Data 23	C	PC_15
PPI0_FS1	EPPI0 Frame Sync 1 (HSYNC)	E	PE_01
PPI0_FS2	EPPI0 Frame Sync 2 (VSYNC)	E	PE_02
PPI0_FS3	EPPI0 Frame Sync 3 (FIELD)	E	PE_03
SPI0_CLK	SPI0 Clock	A	PA_06
SPI0_MISO	SPI0 Mater In, Slave Out	A	PA_07
SPI0_MOSI	SPI0 Master Out, Slave In	A	PA_08
SPI0_RDY	SPI0 Ready	B	PB_11
$\overline{\text{SPI0\_SEL1}}$	SPI0 Slave Select Output 1	A	PA_09
$\overline{\text{SPI0\_SEL2}}$	SPI0 Slave Select Output 2	B	PB_05
$\overline{\text{SPI0\_SEL3}}$	SPI0 Slave Select Output 3	B	PB_14
$\overline{\text{SPI0\_SEL4}}$	SPI0 Slave Select Output 4	B	PB_15
$\overline{\text{SPI0\_SEL5}}$	SPI0 Slave Select Output 5	G	PG_02
$\overline{\text{SPI0\_SEL6}}$	SPI0 Slave Select Output 6	E	PE_15
$\overline{\text{SPI0\_SEL7}}$	SPI0 Slave Select Output 7	F	PF_00
$\overline{\text{SPI0\_SS}}$	SPI0 Slave Select Input	A	PA_09
SPI1_CLK	SPI1 Clock	A	PA_10
SPI1_D2	SPI1 Data 2	A	PA_14
SPI1_D3	SPI1 Data 3	A	PA_15
SPI1_MISO	SPI1 Master In, Slave Out	A	PA_11
SPI1_MOSI	SPI1 Master Out, Slave In	A	PA_12
SPI1_RDY	SPI1 Ready	C	PC_06
$\overline{\text{SPI1\_SEL1}}$	SPI1 Slave Select Output 1	A	PA_13

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SPI1_SEL2	SPI1 Slave Select Output 2	B	PB_10
SPI1_SEL3	SPI1 Slave Select Output 3	B	PB_13
SPI1_SEL4	SPI1 Slave Select Output 4	E	PE_02
SPI1_SEL5	SPI1 Slave Select Output 5	B	PB_06
SPI1_SEL6	SPI1 Slave Select Output 6	G	PG_09
SPI1_SEL7	SPI1 Slave Select Output 7	B	PB_08
SPI1_SS	SPI1 Slave Select Input	A	PA_13
SPI2_CLK	SPI2 Clock	A	PA_04
SPI2_D2	SPI2 Data 2	A	PA_02
SPI2_D3	SPI2 Data 3	A	PA_03
SPI2_MISO	SPI2 Master In, Slave Out	A	PA_00
SPI2_MOSI	SPI2 Master Out, Slave In	A	PA_01
SPI2_RDY	SPI2 Ready	B	PB_05
SPI2_SEL1	SPI2 Slave Select Output 1	A	PA_05
SPI2_SEL2	SPI2 Slave Select Output 2	H	PH_02
SPI2_SEL3	SPI2 Slave Select Output 3	B	PB_12
SPI2_SEL4	SPI2 Slave Select Output 4	G	PG_12
SPI2_SEL5	SPI2 Slave Select Output 5	B	PB_07
SPI2_SEL6	SPI2 Slave Select Output 6	G	PG_01
SPI2_SEL7	SPI2 Slave Select Output 7	E	PE_14
SPI2_SS	SPI2 Slave Select Input	A	PA_05
SPI3_CLK	SPI3 Clock	G	PG_05
SPI3_MISO	SPI3 Master In, Slave Out	G	PG_06
SPI3_MOSI	SPI3 Master Out, Slave In	G	PG_07
SPI3_RDY	SPI3 Ready	F	PF_00
SPI3_SEL1	SPI3 Slave Select Output 1	G	PG_08
SPI3_SEL2	SPI3 Slave Select Output 2	F	PF_07
SPI3_SEL3	SPI3 Slave Select Output 3	E	PE_00
SPI3_SEL4	SPI3 Slave Select Output 4	E	PE_01
SPI3_SEL5	SPI3 Slave Select Output 5	G	PG_15
SPI3_SEL6	SPI3 Slave Select Output 6	F	PF_08
SPI3_SEL7	SPI3 Slave Select Output 7	H	PH_00
SPI3_SS	SPI3 Slave Select Input	G	PG_08
SYS_BMODE0	Boot Mode Control Pin 0	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control Pin 1	Not Muxed	SYS_BMODE1
SYS_BMODE2	Boot Mode Control Pin 2	Not Muxed	SYS_BMODE2
SYS_CLKIN0	Clock/Crystal Input 0	Not Muxed	SYS_CLKIN0
SYS_CLKIN1	Clock/Crystal Input 1	Not Muxed	SYS_CLKIN1
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
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 0	Not Muxed	SYS_XTAL0
SYS_XTAL1	Crystal Output 1	Not Muxed	SYS_XTAL1
TM0_ACI00	TIMER0 Alternate Capture Input 0	D	PD_08
TM0_ACI01	TIMER0 Alternate Capture Input 1	D	PD_04
TM0_ACI02	TIMER0 Alternate Capture Input 2	B	PB_11
TM0_ACI03	TIMER0 Alternate Capture Input 3	B	PB_00

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TM0_ACI04	TIMER0 Alternate Capture Input 4	A	PA_11
TM0_ACI10	TIMER0 Alternate Capture Input 10	G	PG_14
TM0_ACI11	TIMER0 Alternate Capture Input 11	G	PG_01
TM0_ACI12	TIMER0 Alternate Capture Input 12	H	PH_00
TM0_ACI13	TIMER0 Alternate Capture Input 13	H	PH_01
TM0_ACLK01	TIMER0 Alternate Clock 1	A	PA_06
TM0_ACLK02	TIMER0 Alternate Clock 2	A	PA_08
TM0_ACLK03	TIMER0 Alternate Clock 3	G	PG_10
TM0_ACLK04	TIMER0 Alternate Clock 4	B	PB_02
TM0_ACLK10	TIMER0 Alternate Clock 10	G	PG_00
TM0_ACLK11	TIMER0 Alternate Clock 11	G	PG_05
TM0_ACLK12	TIMER0 Alternate Clock 12	G	PG_07
TM0_ACLK13	TIMER0 Alternate Clock 13	F	PF_04
TM0_ACLK14	TIMER0 Alternate Clock 14	I	PI_06
TM0_ACLK15	TIMER0 Alternate Clock 15	E	PE_01
TM0_CLK	TIMER0 Clock	F	PF_05
TM0_TMR00	TIMER0 Timer 0	A	PA_10
TM0_TMR01	TIMER0 Timer 1	A	PA_12
TM0_TMR02	TIMER0 Timer 2	E	PE_10
TM0_TMR03	TIMER0 Timer 3	B	PB_03
TM0_TMR04	TIMER0 Timer 4	B	PB_04
TM0_TMR05	TIMER0 Timer 5	B	PB_05
TM0_TMR06	TIMER0 Timer 6	B	PB_08
TM0_TMR07	TIMER0 Timer 7	B	PB_09
TM0_TMR08	TIMER0 Timer 8	C	PC_05
TM0_TMR09	TIMER0 Timer 9	C	PC_07
TM0_TMR10	TIMER0 Timer 10	G	PG_14
TM0_TMR11	TIMER0 Timer 11	G	PG_15
TM0_TMR12	TIMER0 Timer 12	H	PH_00
TM0_TMR13	TIMER0 Timer 13	H	PH_01
TM0_TMR14	TIMER0 Timer 14	H	PH_02
TM0_TMR15	TIMER0 Timer 15	D	PD_15
TRACE0_CLK	TRACE0 Trace Clock	B	PB_06
TRACE0_D00	TRACE0 Trace Data 0	B	PB_07
TRACE0_D01	TRACE0 Trace Data 1	B	PB_08
TRACE0_D02	TRACE0 Trace Data 2	B	PB_09
TRACE0_D03	TRACE0 Trace Data 3	B	PB_10
TRACE0_D04	TRACE0 Trace Data 4	C	PC_00
TRACE0_D05	TRACE0 Trace Data 5	C	PC_01
TRACE0_D06	TRACE0 Trace Data 6	C	PC_02
TRACE0_D07	TRACE0 Trace Data 7	C	PC_03
TRACE0_D08	TRACE0 Trace Data 8	H	PH_03
TRACE0_D09	TRACE0 Trace Data 9	H	PH_04
TRACE0_D10	TRACE0 Trace Data 10	H	PH_05
TRACE0_D11	TRACE0 Trace Data 11	H	PH_06
TRACE0_D12	TRACE0 Trace Data 12	H	PH_07
TRACE0_D13	TRACE0 Trace Data 13	H	PH_08
TRACE0_D14	TRACE0 Trace Data 14	H	PH_09

Table 10. ADSP-SC59x 400-Ball HPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TRACE0_D15	TRACE0 Trace Data 15	H	PH_10
TWI0_SCL	TWI0 Serial Clock	E	PE_02
TWI0_SDA	TWI0 Serial Data	E	PE_03
TWI1_SCL	TWI1 Serial Clock	B	PB_00
TWI1_SDA	TWI1 Serial Data	B	PB_01
TWI2_SCL	TWI2 Serial Clock	E	PE_04
TWI2_SDA	TWI2 Serial Data	E	PE_05
TWI3_SCL	TWI3 Serial Clock	A	PA_02
TWI3_SDA	TWI3 Serial Data	I	PI_02
TWI4_SCL	TWI4 Serial Clock	D	PD_14
TWI4_SDA	TWI4 Serial Data	C	PC_01
TWI5_SCL	TWI5 Serial Clock	C	PC_02
TWI5_SDA	TWI5 Serial Data	E	PE_01
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	D	PD_06
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	D	PD_07
$\overline{\text{UART0\_RX}}$	UART0 Receive	A	PA_07
$\overline{\text{UART0\_TX}}$	UART0 Transmit	D	PD_09
$\overline{\text{UART1\_CTS}}$	UART1 Clear to Send	D	PD_03
$\overline{\text{UART1\_RTS}}$	UART1 Request to Send	B	PB_00
$\overline{\text{UART1\_RX}}$	UART1 Receive	D	PD_04
$\overline{\text{UART1\_TX}}$	UART1 Transmit	D	PD_05
$\overline{\text{UART2\_CTS}}$	UART2 Clear to Send	B	PB_14
$\overline{\text{UART2\_RTS}}$	UART2 Request to Send	D	PD_12
$\overline{\text{UART2\_RX}}$	UART2 Receive	D	PD_10
$\overline{\text{UART2\_TX}}$	UART2 Transmit	D	PD_11
$\overline{\text{UART3\_CTS}}$	UART3 Clear to Send	G	PG_10
$\overline{\text{UART3\_RTS}}$	UART3 Request to Send	G	PG_09
$\overline{\text{UART3\_RX}}$	UART3 Receive	G	PG_04
$\overline{\text{UART3\_TX}}$	UART3 Transmit	G	PG_03
USBC0_CLK	USBC0 Clock Signal	F	PF_14
USBC0_DATA0	USBC0 Data 0	F	PF_13
USBC0_DATA1	USBC0 Data 1	F	PF_12
USBC0_DATA2	USBC0 Data 2	F	PF_11
USBC0_DATA3	USBC0 Data 3	F	PF_10
USBC0_DATA4	USBC0 Data 4	F	PF_07
USBC0_DATA5	USBC0 Data 5	F	PF_06
USBC0_DATA6	USBC0 Data 6	F	PF_05
USBC0_DATA7	USBC0 Data 7	F	PF_04
USBC0_DIR	USBC0 Data Direction Control	F	PF_09
USBC0_NXT	USBC0 Next Data Control	F	PF_08
USBC0_STOP	USBC0 Stop Output Control	F	PF_03

GPIO MULTIPLEXING FOR 400-BALL HIGH PERIPHERAL COUNT (HPC) BGA PACKAGE

Table 11 through Table 19 identify the pin functions that are multiplexed on the GPIO pins of the 400-ball HPC BGA package.

Table 11. ADSP-SC59x 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	SPI2_MISO	OSPI0_MISO[1]			
PA_01	SPI2_MOSI	OSPI0_MOSI[1]			
PA_02	SPI2_D2	OSPI0_D2[1]	TWI3_SCL[1]		TM0_ACLK03
PA_03	SPI2_D3	OSPI0_D3[1]	TWI3_SDA[1]		
PA_04	SPI2_CLK	OSPI0_CLK[1]			
PA_05	$\overline{\text{SPI2\_SEL1}}$	$\overline{\text{OSPI0\_SEL1}}$[2]			$\overline{\text{SPI2\_SS}}$
PA_06	SPI0_CLK	$\overline{\text{UART0\_TX}}$[1]	OSPI0_D4[1]		TM0_ACLK01
PA_07	SPI0_MISO	$\overline{\text{UART0\_RX}}$[1]	OSPI0_D5[1]		TM0_ACI00
PA_08	SPI0_MOSI	$\overline{\text{UART0\_RTS}}$[1]	OSPI0_D6[1]		TM0_ACLK02
PA_09	$\overline{\text{SPI0\_SEL1}}$	$\overline{\text{UART0\_CTS}}$[1]	OSPI0_D7[1]		$\overline{\text{SPI0\_SS}}$
PA_10	TWI0_SCL[1]	SPI1_CLK	TM0_TMR00		
PA_11	TWI0_SDA[1]	SPI1_MISO	HADC0_EOC_DOUT		TM0_ACI04
PA_12	C1_FLG00	SPI1_MOSI	TM0_TMR01		
PA_13	C1_FLG01	$\overline{\text{SPI1\_SEL1}}$	TM0_TMR02		$\overline{\text{SPI1\_SS}}$
PA_14	TWI2_SCL[1]	SPI1_D2	$\overline{\text{UART1\_RX}}$[1]		TM0_ACI01
PA_15	TWI2_SDA[1]	SPI1_D3	$\overline{\text{UART1\_TX}}$[1]		

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other.

[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 12. ADSP-SC59x 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	MLB0_DAT	TWI1_SCL[1]	$\overline{\text{UART1\_RTS}}$[1]		TM0_ACI03
PB_01	MLB0_SIG	TWI1_SDA[1]	$\overline{\text{UART1\_CTS}}$[1]		TM0_CLK
PB_02	MLB0_CLK	C1_FLG03	LP1_ACK		TM0_ACLK04
PB_03	TM0_TMR03	C1_FLG02	$\overline{\text{SPI2\_SEL2}}$[2]		CNT0_UD
PB_04	TM0_TMR04	SPI1_RDY[2]	LP0_ACK		CNT0_ZM
PB_05	TM0_TMR05	SPI2_RDY[2]	$\overline{\text{SPI0\_SEL2}}$		CNT0_DG
PB_06	LP0_CLK	$\overline{\text{SPI1\_SEL5}}$		TRACE0_CLK	
PB_07	LP0_D0	$\overline{\text{SPI2\_SEL5}}$		TRACE0_D00	
PB_08	LP0_D1	$\overline{\text{SPI1\_SEL7}}$[2]	TM0_TMR06	TRACE0_D01	
PB_09	LP0_D2	$\overline{\text{SPI2\_SEL7}}$[2]	TM0_TMR07	TRACE0_D02	
PB_10	LP0_D3	$\overline{\text{SPI1\_SEL2}}$		TRACE0_D03	
PB_11	LP0_D4	SPI0_RDY[2]		$\overline{\text{UART2\_RX}}$[1]	TM0_ACI02
PB_12	LP0_D5	$\overline{\text{SPI2\_SEL3}}$[2]		$\overline{\text{UART2\_TX}}$[1]	
PB_13	LP0_D6	$\overline{\text{SPI1\_SEL3}}$	OSPI0_DQS[1]	$\overline{\text{UART2\_RTS}}$[1]	
PB_14	LP0_D7	$\overline{\text{SPI0\_SEL3}}$		$\overline{\text{UART2\_CTS}}$[1]	
PB_15	LP1_D0	$\overline{\text{SPI0\_SEL4}}$			

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other

[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 13. ADSP-SC59x 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	LP1_D1	TWI4_SCL[1]	TRACE0_D04	$\overline{\text{SPI1\_SEL4}}$	
PC_01	LP1_D2	TWI4_SDA[1]	TRACE0_D05	$\overline{\text{SPI2\_SEL4}}$[2]	
PC_02	LP1_D3	TWI5_SCL[1]	TRACE0_D06	$\overline{\text{SPI1\_SEL6}}$[2]	
PC_03	LP1_D4	TWI5_SDA[1]	TRACE0_D07	$\overline{\text{SPI2\_SEL6}}$[2]	
PC_04	LP1_D5	$\overline{\text{OSPI0\_SEL2}}$[2]			
PC_05	LP1_D6	$\overline{\text{OSPI0\_SEL3}}$[2]	TM0_TMR08		
PC_06	LP1_D7	SPI1_RDY			
PC_07	LP1_CLK		TM0_TMR09	SYS_FAULT	
PC_08	OSPI0_CLK[1]			PPI0_D16	
PC_09	OSPI0_D3[1]			PPI0_D17	
PC_10	OSPI0_D2[1]			PPI0_D18	
PC_11	OSPI0_MOSI[1]			PPI0_D19	
PC_12	OSPI0_MISO[1]			PPI0_D20	
PC_13	OSPI0_D7[1]			PPI0_D21	
PC_14	OSPI0_D6[1]			PPI0_D22	
PC_15	OSPI0_D5[1]			PPI0_D23	

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other

[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 14. ADSP-SC59x 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	OSPI0_D4[1]				
PD_01	$\overline{\text{OSPI0\_SEL1}}$[2]			PPI0_D06	
PD_02	$\overline{\text{UART1\_RTS}}$[1]	C2_FLG10	ETH0_CRS		
PD_03	$\overline{\text{UART1\_CTS}}$[1]	C1_FLG11		LP0_ACK	ETH0_PTPAUXIN2
PD_04	$\overline{\text{UART1\_RX}}$[1]		OSPI0_DQS[1]	PPI0_D07	TM0_ACI01
PD_05	$\overline{\text{UART1\_TX}}$[1]			PPI0_D08	ETH0_PTPAUXIN1
PD_06	$\overline{\text{UART0\_CTS}}$[1]	ETH0_RXERR			
PD_07	$\overline{\text{UART0\_RTS}}$[1]	ETH0_COL			
PD_08	$\overline{\text{UART0\_RX}}$[1]				TM0_ACI00
PD_09	$\overline{\text{UART0\_TX}}$[1]				
PD_10		$\overline{\text{UART2\_RX}}$[1]		PPI0_D09	TM0_ACI02
PD_11		$\overline{\text{UART2\_TX}}$[1]		PPI0_D10	
PD_12	TM0_TMR06	$\overline{\text{UART2\_RTS}}$[1]		PPI0_D11	
PD_13	TM0_TMR07	$\overline{\text{UART2\_CTS}}$[1]		PPI0_D12	
PD_14	TWI4_SCL[1]	LP1_ACK	C2_FLG09	PPI0_D13	
PD_15	TWI4_SDA[1]		TM0_TMR15	PPI0_D14	

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other

[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 15. ADSP-SC59x 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	TWI5_SCL[1]		SPI3_SEL3	PPI0_D15	
PE_01	TWI5_SDA[1]		SPI3_SEL4	PPI0_FS1	TM0_ACLK15
PE_02	TWI0_SCL[1]	SPI1_SEL4	HADC0_MUX0	PPI0_FS2	
PE_03	TWI0_SDA[1]	SPI2_SEL3[2]	HADC0_MUX2	PPI0_FS3	TM0_ACLK04
PE_04	TWI2_SCL[1]		HADC0_MUX1	PPI0_CLK	
PE_05	TWI2_SDA[1]			PPI0_D00	
PE_06	TM0_TMR08	C1_FLG02		PPI0_D01	
PE_07	TM0_TMR09	C1_FLG03	SPI1_RDY[2]	PPI0_D02	
PE_08	TM0_TMR00			PPI0_D03	
PE_09	TM0_TMR01			PPI0_D04	ETH0_PTPAUXIN3
PE_10	TM0_TMR02	SPI0_SEL4		PPI0_D05	
PE_11	ETH1_REFCLK		C2_FLG07		
PE_12	ETH1_TXEN		C1_FLG08		
PE_13	ETH1_TXD0		C2_FLG13		
PE_14	ETH1_TXD1		SPI2_SEL7[2]		
PE_15	ETH1_RXD0		SPI0_SEL6		

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other.
[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 16. ADSP-SC59x 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	ETH1_RXD1	SPI3_RDY	SPI0_SEL7		
PF_01	ETH1_MDIO		C1_FLG07		
PF_02	ETH1_MDC		C1_FLG06		
PF_03	ETH1_CRS	C1_FLG10	USBC0_STOP		
PF_04			USBC0_DATA7		TM0_ACLK13
PF_05	MLB0_CLKOUT		USBC0_DATA6		TM0_CLK
PF_06	C2_FLG02	SPI1_SEL7[1]	USBC0_DATA5		
PF_07	C2_FLG03	SPI3_SEL2	USBC0_DATA4		
PF_08	SPI3_SEL6	C2_FLG08	USBC0_NXT	TM0_TMR11	
PF_09		C1_FLG09	USBC0_DIR		
PF_10		C2_FLG04	USBC0_DATA3		
PF_11		C2_FLG05	USBC0_DATA2		
PF_12		C1_FLG13	USBC0_DATA1		
PF_13		C1_FLG12	USBC0_DATA0		
PF_14		C2_FLG12	USBC0_CLK		
PF_15	CANFD0_RX				TM0_ACI04

[1] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 17. ADSP-SC59x Signal Multiplexing for Port G

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PG_00	CANFD0_TX				TM0_ACLK10
PG_01	CANFD1_RX	SPI2_SEL6[1]			TM0_ACI11
PG_02	CANFD1_TX	SPI0_SEL5			
PG_03	UART3_TX				
PG_04	UART3_RX				TM0_ACI03
PG_05	SPI3_CLK				TM0_ACLK11
PG_06	SPI3_MISO				
PG_07	SPI3_MOSI				TM0_ACLK12
PG_08	SPI3_SEL1				SPI3_SS
PG_09	UART3_RTS	SPI1_SEL6[1]	C1_FLG14		TM0_ACLK01
PG_10	UART3_CTS		C2_FLG14		TM0_ACLK03
PG_11			C2_FLG15		
PG_12	TM0_TMR03	SPI2_SEL4[1]	C2_FLG11	OSPI0_SEL3[1]	
PG_13	C1_FLG00		C2_FLG06	OSPI0_SEL4[1]	
PG_14	TM0_TMR10	SPI0_SEL2			TM0_ACI10
PG_15	TM0_TMR11	SPI2_RDY[1]	SPI3_SEL5		

[1] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 18. ADSP-SC59x Signal Multiplexing for Port H

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PH_00	TM0_TMR12		SPI3_SEL7		TM0_ACI12
PH_01	TM0_TMR13	SPI0_RDY[1]			TM0_ACI13
PH_02	C1_FLG01	SPI2_SEL2[1]	TM0_TMR14		
PH_03	ETH0_MDC	TRACE0_D08			
PH_04	ETH0_MDIO	TRACE0_D09			
PH_05	ETH0_RXD0	TRACE0_D10			
PH_06	ETH0_RXD1	TRACE0_D11			
PH_07	ETH0_RXCLK_REFCLK	TRACE0_D12			
PH_08	ETH0_RXCTL_RXDV	TRACE0_D13			
PH_09	ETH0_TXD0	TRACE0_D14			
PH_10	ETH0_TXD1	TRACE0_D15			
PH_11	ETH0_RXD2				
PH_12	ETH0_RXD3				
PH_13	ETH0_TXCTL_TXEN				
PH_14	ETH0_TXCLK				
PH_15	ETH0_TXD2				

[1] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 19. ADSP-SC59x Signal Multiplexing for Port I

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PI_00	ETH0_TXD3				
PI_01	C2_FLG00	ETH0_PTPCLKIN0	TWI3_SCL[1]		
PI_02	C2_FLG01	ETH0_PTPAUXIN0	TWI3_SDA[1]		
PI_03	TWI1_SCL[1]	C1_FLG04	ETH0_PTPPPS1		
PI_04	TWI1_SDA[1]	C1_FLG05	ETH0_PTPPPS0		
PI_05	ETH0_PTPPPS2	$\overline{\text{OSPI0\_SEL2}}$[2]	C1_FLG15		TM0_ACLK02
PI_06	ETH0_PTPPPS3				TM0_ACLK14

[1] To ensure proper timing, for peripherals whose signals are available at multiple places in the pin mux, care must be taken to select all needed signals of the peripheral so that their associated pins are as close as possible to each other

[2] These peripheral signals are available at multiple places in the pin mux. These signals can be selected from either of the two locations in pin mux, regardless of proximity to other associated signals of the same peripheral.

Table 20 shows the internal timer signal routing. This table applies to both the HPC and LPC 400-ball BGA packages.

Table 20. ADSP-2159x/ADSP-SC59x Internal Timer Signal Routing

Timer Input Signal	Internal Source
TM0_ACLK0	SYS_CLKIN0
TM0_ACI5	DAI0_PB04
TM0_ACLK5	DAI0_PB03
TM0_ACI6	DAI1_PB04
TM0_ACLK6	DAI1_PB03
TM0_ACI7	CNT0_TO
TM0_ACLK7	SYS_CLKIN1
TM0_ACI8	DAI0_PB06
TM0_ACLK8	DAI0_PB05
TM0_ACI9	DAI1_PB06
TM0_ACLK9	DAI1_PB05
TM0_ACI14	DAI0 Group C
TM0_ACI15	DAI1 Group C

400-BALL LOW PERIPHERAL COUNT (LPC) BGA SIGNAL DESCRIPTIONS

The processor pin definitions are shown in Table 21 for the 400-ball LPC BGA package. The columns in this table provide the following information:

- The signal name column includes the signal name for every pin and the GPIO multiplexed pin function, where applicable.
- The description column provides a descriptive name for each signal.
- The port column shows whether or not a signal is multiplexed with other signals on a GPIO port pin.
- The pin name column 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 GPIO pin).
- The DAI pins and their associated signal routing units (SRUs) connect inputs and outputs of the DAI peripherals (SPORT, ASRC, S/PDIF, and PCG). See the Digital Audio Interface (DAI) chapter of the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference for complete information on the use of the DAI and SRUs.

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
C1_FLG00	SHARC+ Core 1 FLAGS I/O 0	A	PA_12
C1_FLG01	SHARC+ Core 1 FLAGS I/O 1	A	PA_13
C1_FLG02	SHARC+ Core 1 FLAGS I/O 2	B	PB_03
C1_FLG03	SHARC+ Core 1 FLAGS I/O 3	B	PB_02
CNT0_DG	CNT0 Count Down and Gate	B	PB_05
CNT0_UD	CNT0 Count Up and Direction	B	PB_03
CNT0_ZM	CNT0 Count Zero Marker	B	PB_04
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_PIN19	DAI0 Pin 19	Not Muxed	DAI0_PIN19
DAI0_PIN20	DAI0 Pin 20	Not Muxed	DAI0_PIN20
DAI1_PIN01	DAI1 Pin 1	Not Muxed	DAI1_PIN01
DAI1_PIN02	DAI1 Pin 2	Not Muxed	DAI1_PIN02
DAI1_PIN03	DAI1 Pin 3	Not Muxed	DAI1_PIN03
DAI1_PIN04	DAI1 Pin 4	Not Muxed	DAI1_PIN04
DAI1_PIN05	DAI1 Pin 5	Not Muxed	DAI1_PIN05
DAI1_PIN06	DAI1 Pin 6	Not Muxed	DAI1_PIN06
DAI1_PIN07	DAI1 Pin 7	Not Muxed	DAI1_PIN07
DAI1_PIN08	DAI1 Pin 8	Not Muxed	DAI1_PIN08
DAI1_PIN09	DAI1 Pin 9	Not Muxed	DAI1_PIN09
DAI1_PIN10	DAI1 Pin 10	Not Muxed	DAI1_PIN10
DAI1_PIN11	DAI1 Pin 11	Not Muxed	DAI1_PIN11
DAI1_PIN12	DAI1 Pin 12	Not Muxed	DAI1_PIN12
DAI1_PIN19	DAI1 Pin 19	Not Muxed	DAI1_PIN19
DAI1_PIN20	DAI1 Pin 20	Not Muxed	DAI1_PIN20
DMC0_A00	DMC0 Address 0	Not Muxed	DMC0_A00

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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	$\overline{\text{DMC0\_CAS}}$
DMC0_CK	DMC0 Clock	Not Muxed	DMC0_CK
$\overline{\text{DMC0\_CK}}$	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
DMC0_LDQS	DMC0 Data Strobe for Lower Byte	Not Muxed	DMC0_LDQS
$\overline{\text{DMC0\_LDQS}}$	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	Not Muxed	$\overline{\text{DMC0\_RESET}}$
DMC0_RZQ	DMC0 External Calibration Resistor Connection	Not Muxed	DMC0_RZQ
DMC0_UDM	DMC0 Data Mask for Upper Byte	Not Muxed	DMC0_UDM
DMC0_UDQS	DMC0 Data Strobe for Upper Byte	Not Muxed	DMC0_UDQS

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC0_UDQS	DMC0 Data Strobe for Upper Byte (Complement)	Not Muxed	DMC0_UDQS
DMC0_VREF0	DMC0 Voltage Reference	Not Muxed	DMC0_VREF0
DMC0_VREF1	DMC0 Voltage Reference	Not Muxed	DMC0_VREF1
DMC0_WE	DMC0 Write Enable	Not Muxed	DMC0_WE
HADC0_EOC_DOUT	HADC0 End of Conversion	A	PA_11
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
JTG_TRST	JTAG Reset	Not Muxed	JTG_TRST
LP0_ACK	LP0 Acknowledge	B	PB_04
LP0_CLK	LP0 Clock	B	PB_06
LP0_D0	LP0 Data 0	B	PB_07
LP0_D1	LP0 Data 1	B	PB_08
LP0_D2	LP0 Data 2	B	PB_09
LP0_D3	LP0 Data 3	B	PB_10
LP0_D4	LP0 Data 4	B	PB_11
LP0_D5	LP0 Data 5	B	PB_12
LP0_D6	LP0 Data 6	B	PB_13
LP0_D7	LP0 Data 7	B	PB_14
LP1_ACK	LP1 Acknowledge	B	PB_02
LP1_CLK	LP1 Clock	C	PC_07
LP1_D0	LP1 Data 0	B	PB_15
LP1_D1	LP1 Data 1	C	PC_00
LP1_D2	LP1 Data 2	C	PC_01
LP1_D3	LP1 Data 3	C	PC_02
LP1_D4	LP1 Data 4	C	PC_03
LP1_D5	LP1 Data 5	C	PC_04
LP1_D6	LP1 Data 6	C	PC_05
LP1_D7	LP1 Data 7	C	PC_06
MLB0_CLK	MLB0 Single-Ended Clock	B	PB_02
MLB0_DAT	MLB0 Single-Ended Data	B	PB_00
MLB0_SIG	MLB0 Single-Ended Signal	B	PB_01
OSPI0_CLK	OSPI0 Clock	A	PA_04
OSPI0_D2	OSPI0 Data 2	A	PA_02
OSPI0_D3	OSPI0 Data 3	A	PA_03
OSPI0_D4	OSPI0 Data 4	A	PA_06
OSPI0_D5	OSPI0 Data 5	A	PA_07
OSPI0_D6	OSPI0 Data 6	A	PA_08
OSPI0_D7	OSPI0 Data 7	A	PA_09
OSPI0_DQS	OSPI0 Data Strobe	B	PB_13
OSPI0_MISO	OSPI0 Master In, Slave Out	A	PA_00

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
OSPI0_MOSI	OSPI0 Master Out, Slave In	A	PA_01
OSPI0_SEL1	OSPI0 Slave Select Output 1	A	PA_05
OSPI0_SEL2	OSPI0 Slave Select Output 2	C	PC_04
OSPI0_SEL3	OSPI0 Slave Select Output 3	C	PC_05
SPI0_CLK	SPI0 Clock	A	PA_06
SPI0_MISO	SPI0 Master In, Slave Out	A	PA_07
SPI0_MOSI	SPI0 Master Out, Slave In	A	PA_08
SPI0_RDY	SPI0 Ready	B	PB_11
SPI0_SEL1	SPI0 Slave Select Output 1	A	PA_09
SPI0_SEL2	SPI0 Slave Select Output 2	B	PB_05
SPI0_SEL3	SPI0 Slave Select Output 3	B	PB_14
SPI0_SEL4	SPI0 Slave Select Output 4	B	PB_15
SPI0_SS	SPI0 Slave Select Input	A	PA_09
SPI1_CLK	SPI1 Clock	A	PA_10
SPI1_D2	SPI1 Data 2	A	PA_14
SPI1_D3	SPI1 Data 3	A	PA_15
SPI1_MISO	SPI1 Master In, Slave Out	A	PA_11
SPI1_MOSI	SPI1 Master Out, Slave In	A	PA_12
SPI1_RDY	SPI1 Ready	C	PC_06
SPI1_SEL1	SPI1 Slave Select Output 1	A	PA_13
SPI1_SEL2	SPI1 Slave Select Output 2	B	PB_10
SPI1_SEL3	SPI1 Slave Select Output 3	B	PB_13
SPI1_SEL4	SPI1 Slave Select Output 4	C	PC_00
SPI1_SEL5	SPI1 Slave Select Output 5	B	PB_06
SPI1_SEL6	SPI1 Slave Select Output 6	C	PC_02
SPI1_SEL7	SPI1 Slave Select Output 7	B	PB_08
SPI1_SS	SPI1 Slave Select Input	A	PA_13
SPI2_CLK	SPI2 Clock	A	PA_04
SPI2_D2	SPI2 Data 2	A	PA_02
SPI2_D3	SPI2 Data 3	A	PA_03
SPI2_MISO	SPI2 Master In, Slave Out	A	PA_00
SPI2_MOSI	SPI2 Master Out, Slave In	A	PA_01
SPI2_RDY	SPI2 Ready	B	PB_05
SPI2_SEL1	SPI2 Slave Select Output 1	A	PA_05
SPI2_SEL2	SPI2 Slave Select Output 2	B	PB_03
SPI2_SEL3	SPI2 Slave Select Output 3	B	PB_12
SPI2_SEL4	SPI2 Slave Select Output 4	C	PC_01
SPI2_SEL5	SPI2 Slave Select Output 5	B	PB_07
SPI2_SEL6	SPI2 Slave Select Output 6	C	PC_03
SPI2_SEL7	SPI2 Slave Select Output 7	B	PB_09
SPI2_SS	SPI2 Slave Select Input	A	PA_05
SYS_BMODE0	Boot Mode Control Pin 0	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control Pin 1	Not Muxed	SYS_BMODE1
SYS_BMODE2	Boot Mode Control Pin 2	Not Muxed	SYS_BMODE2
SYS_CLKIN0	Clock/Crystal Input	Not Muxed	SYS_CLKIN0
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_FAULT	Active-Low Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL0	Crystal Output	Not Muxed	SYS_XTAL0
TM0_ACI00	TIMER0 Alternate Capture Input 0	A	PA_07
TM0_ACI01	TIMER0 Alternate Capture Input 1	A	PA_14
TM0_ACI02	TIMER0 Alternate Capture Input 2	B	PB_11
TM0_ACI03	TIMER0 Alternate Capture Input 3	B	PB_00
TM0_ACI04	TIMER0 Alternate Capture Input 4	A	PA_11
TM0_ACLK01	TIMER0 Alternate Clock 1	A	PA_06
TM0_ACLK02	TIMER0 Alternate Clock 2	A	PA_08
TM0_ACLK03	TIMER0 Alternate Clock 3	A	PA_02
TM0_ACLK04	TIMER0 Alternate Clock 4	B	PB_02
TM0_CLK	TIMER0 Timer Clock	B	PB_01
TM0_TMR00	TIMER0 Timer 0	A	PA_10
TM0_TMR01	TIMER0 Timer 1	A	PA_12
TM0_TMR02	TIMER0 Timer 2	A	PA_13
TM0_TMR03	TIMER0 Timer 3	B	PB_03
TM0_TMR04	TIMER0 Timer 4	B	PB_04
TM0_TMR05	TIMER0 Timer 5	B	PB_05
TM0_TMR06	TIMER0 Timer 6	B	PB_08
TM0_TMR07	TIMER0 Timer 7	B	PB_09
TM0_TMR08	TIMER0 Timer 8	C	PC_05
TM0_TMR09	TIMER0 Timer 9	C	PC_07
TRACE0_CLK	TRACE0 Trace Clock	B	PB_06
TRACE0_D00	TRACE0 Trace Data 0	B	PB_07
TRACE0_D01	TRACE0 Trace Data 1	B	PB_08
TRACE0_D02	TRACE0 Trace Data 2	B	PB_09
TRACE0_D03	TRACE0 Trace Data 3	B	PB_10
TRACE0_D04	TRACE0 Trace Data 4	C	PC_00
TRACE0_D05	TRACE0 Trace Data 5	C	PC_01
TRACE0_D06	TRACE0 Trace Data 6	C	PC_02
TRACE0_D07	TRACE0 Trace Data 7	C	PC_03
TWI0_SCL	TWI0 Serial Clock	A	PA_10
TWI0_SDA	TWI0 Serial Data	A	PA_11
TWI1_SCL	TWI1 Serial Clock	B	PB_00
TWI1_SDA	TWI1 Serial Data	B	PB_01
TWI2_SCL	TWI2 Serial Clock	A	PA_14
TWI2_SDA	TWI2 Serial Data	A	PA_15
TWI3_SCL	TWI3 Serial Clock	A	PA_02
TWI3_SDA	TWI3 Serial Data	A	PA_03
TWI4_SCL	TWI4 Serial Clock	C	PC_00
TWI4_SDA	TWI4 Serial Data	C	PC_01
TWI5_SCL	TWI5 Serial Clock	C	PC_02
TWI5_SDA	TWI5 Serial Data	C	PC_03
UART0_CTS	UART0 Clear to Send	A	PA_09
UART0_RTS	UART0 Request to Send	A	PA_08
UART0_RX	UART0 Receive	A	PA_07
UART0_TX	UART0 Transmit	A	PA_06
UART1_CTS	UART1 Clear to Send	B	PB_01

Table 21. ADSP-2159x 400-Ball LPC BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
UART1_RTS	UART1 Request to Send	B	PB_00
UART1_RX	UART1 Receive	A	PA_14
UART1_TX	UART1 Transmit	A	PA_15
UART2_CTS	UART2 Clear to Send	B	PB_14
UART2_RTS	UART2 Request to Send	B	PB_13
UART2_RX	UART2 Receive	B	PB_11
UART2_TX	UART2 Transmit	B	PB_12

GPIO MULTIPLEXING 400-BALL LOW PERIPHERAL COUNT (LPC) BGA PACKAGE

Table 22 through Table 24 identify the pin functions that are multiplexed on the GPIO pins of the 400-ball LPC BGA package.

Table 22. ADSP-2159x 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	SPI2_MISO	OSPI0_MISO			
PA_01	SPI2_MOSI	OSPI0_MOSI			
PA_02	SPI2_D2	OSPI0_D2	TWI3_SCL		TM0_ACLK03
PA_03	SPI2_D3	OSPI0_D3	TWI3_SDA		
PA_04	SPI2_CLK	OSPI0_CLK			
PA_05	$\overline{\text{SPI2\_SEL1}}$	$\overline{\text{OSPI0\_SEL1}}$			$\overline{\text{SPI2\_SS}}$
PA_06	SPI0_CLK	$\overline{\text{UART0\_TX}}$	OSPI0_D4		TM0_ACLK01
PA_07	SPI0_MISO	$\overline{\text{UART0\_RX}}$	OSPI0_D5		TM0_ACI00
PA_08	SPI0_MOSI	$\overline{\text{UART0\_RTS}}$	OSPI0_D6		TM0_ACLK02
PA_09	$\overline{\text{SPI0\_SEL1}}$	$\overline{\text{UART0\_CTS}}$	OSPI0_D7		$\overline{\text{SPI0\_SS}}$
PA_10	TWI0_SCL	SPI1_CLK	TM0_TMR00		
PA_11	TWI0_SDA	SPI1_MISO	HADC0_EOC_DOUT		TM0_ACI04
PA_12	C1_FLG00	SPI1_MOSI	TM0_TMR01		
PA_13	C1_FLG01	$\overline{\text{SPI1\_SEL1}}$	TM0_TMR02		$\overline{\text{SPI1\_SS}}$
PA_14	TWI2_SCL	SPI1_D2	$\overline{\text{UART1\_RX}}$		TM0_ACI01
PA_15	TWI2_SDA	SPI1_D3	$\overline{\text{UART1\_TX}}$		

Table 23. ADSP-2159x 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	MLB0_DAT	TWI1_SCL	$\overline{\text{UART1\_RTS}}$		TM0_ACI03
PB_01	MLB0_SIG	TWI1_SDA	$\overline{\text{UART1\_CTS}}$		TM0_CLK
PB_02	MLB0_CLK	C1_FLG03	LP1_ACK		TM0_ACLK04
PB_03	TM0_TMR03	C1_FLG02	$\overline{\text{SPI2\_SEL2}}$		CNT0_UD
PB_04	TM0_TMR04	SPI1_RDY	LP0_ACK		CNT0_ZM
PB_05	TM0_TMR05	SPI2_RDY	$\overline{\text{SPI0\_SEL2}}$		CNT0_DG
PB_06	LP0_CLK	$\overline{\text{SPI1\_SEL5}}$		TRACE0_CLK	
PB_07	LP0_D0	$\overline{\text{SPI2\_SEL5}}$		TRACE0_D00	
PB_08	LP0_D1	$\overline{\text{SPI1\_SEL7}}$	TM0_TMR06	TRACE0_D01	
PB_09	LP0_D2	$\overline{\text{SPI2\_SEL7}}$	TM0_TMR07	TRACE0_D02	
PB_10	LP0_D3	$\overline{\text{SPI1\_SEL2}}$		TRACE0_D03	
PB_11	LP0_D4	SPI0_RDY		$\overline{\text{UART2\_RX}}$	TM0_ACI02
PB_12	LP0_D5	$\overline{\text{SPI2\_SEL3}}$		$\overline{\text{UART2\_TX}}$	
PB_13	LP0_D6	$\overline{\text{SPI1\_SEL3}}$	OSPI0_DQS	$\overline{\text{UART2\_RTS}}$	
PB_14	LP0_D7	$\overline{\text{SPI0\_SEL3}}$		$\overline{\text{UART2\_CTS}}$	
PB_15	LP1_D0	$\overline{\text{SPI0\_SEL4}}$			

Table 24. ADSP-2159x 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	LP1_D1	TWI4_SCL	TRACE0_D04	$\overline{\text{SPI1\_SEL4}}$	
PC_01	LP1_D2	TWI4_SDA	TRACE0_D05	$\overline{\text{SPI2\_SEL4}}$	
PC_02	LP1_D3	TWI5_SCL	TRACE0_D06	$\overline{\text{SPI1\_SEL6}}$	
PC_03	LP1_D4	TWI5_SDA	TRACE0_D07	$\overline{\text{SPI2\_SEL6}}$	
PC_04	LP1_D5	$\overline{\text{OSPI0\_SEL2}}$			
PC_05	LP1_D6	$\overline{\text{OSPI0\_SEL3}}$	TM0_TMR08		
PC_06	LP1_D7	SPI1_RDY			
PC_07	LP1_CLK		TM0_TMR09	SYS_FAULT	

ADSP-2159x/ADSP-SC59x 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:

- The signal name column includes the signal name for every pin and the GPIO multiplexed pin function, where applicable.
- The type column identifies the I/O type or supply type of the pin. The abbreviations used in this column are analog (a), supply (s), ground (g) and Input, Output, and InOut.
- The driver type column identifies the driver type used by the corresponding pin. The driver types are defined in the Output Drive Currents section of this data sheet.
- The internal termination column specifies the termination present after the processor is powered up (both during reset and after reset).
- The reset termination column specifies the termination present when the processor is in the reset state.
- The reset drive column specifies the active drive on the signal when the processor is in the reset state.
- The power domain column specifies the power supply domain in which the signal resides.
- The description and notes column identifies any special requirements or characteristics for a signal. These recommendations apply whether or not the hardware block associated with the signal is featured on the product. If no special requirements are listed, the signal can be left unconnected if it is not used. For multiplexed GPIO pins, this column identifies the functions available on the pin.

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
DAI0_PIN01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 1 Notes: See note[2]
DAI0_PIN02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 2 Notes: See note[2]
DAI0_PIN03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 3 Notes: See note[2]
DAI0_PIN04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 4 Notes: See note[2]
DAI0_PIN05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 5 Notes: See note[2]
DAI0_PIN06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 6 Notes: See note[2]
DAI0_PIN07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 7 Notes: See note[2]
DAI0_PIN08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 8 Notes: See note[2]
DAI0_PIN09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 9 Notes: See note[2]
DAI0_PIN10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 10 Notes: See note[2]
DAI0_PIN11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 11 Notes: See note[2]
DAI0_PIN12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 12 Notes: See note[2]
DAI0_PIN13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 13 Notes: See note[2]
DAI0_PIN14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 14 Notes: See note[2]
DAI0_PIN15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 15 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
DAI0_PIN16	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 16 Notes: See note[2]
DAI0_PIN17	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 17 Notes: See note[2]
DAI0_PIN18	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 18 Notes: See note[2]
DAI0_PIN19	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 19 Notes: See note[2]
DAI0_PIN20	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI0 Pin 20 Notes: See note[2]
DAI1_PIN01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 1 Notes: See note[2]
DAI1_PIN02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 2 Notes: See note[2]
DAI1_PIN03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 3 Notes: See note[2]
DAI1_PIN04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 4 Notes: See note[2]
DAI1_PIN05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 5 Notes: See note[2]
DAI1_PIN06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 6 Notes: See note[2]
DAI1_PIN07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 7 Notes: See note[2]
DAI1_PIN08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 8 Notes: See note[2]
DAI1_PIN09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 9 Notes: See note[2]
DAI1_PIN10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 10 Notes: See note[2]
DAI1_PIN11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 11 Notes: See note[2]
DAI1_PIN12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 12 Notes: See note[2]
DAI1_PIN13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 13 Notes: See note[2]
DAI1_PIN14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 14 Notes: See note[2]
DAI1_PIN15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 15 Notes: See note[2]
DAI1_PIN16	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 16 Notes: See note[2]
DAI1_PIN17	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 17 Notes: See note[2]
DAI1_PIN18	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 18 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
DAI1_PIN19	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 19 Notes: See note[2]
DAI1_PIN20	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: DAI1 Pin 20 Notes: See note[2]
DMC0_A00	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 0 Notes: No notes
DMC0_A01	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 1 Notes: No notes
DMC0_A02	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 2 Notes: No notes
DMC0_A03	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 3 Notes: No notes
DMC0_A04	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 4 Notes: No notes
DMC0_A05	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 5 Notes: No notes
DMC0_A06	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 6 Notes: No notes
DMC0_A07	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 7 Notes: No notes
DMC0_A08	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 8 Notes: No notes
DMC0_A09	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 9 Notes: No notes
DMC0_A10	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 10 Notes: No notes
DMC0_A11	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 11 Notes: No notes
DMC0_A12	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 12 Notes: No notes
DMC0_A13	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 13 Notes: No notes
DMC0_A14	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 14 Notes: No notes
DMC0_A15	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Address 15 Notes: No notes
DMC0_BA0	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Bank Address Input 0 Notes: No notes
DMC0_BA1	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Bank Address Input 1 Notes: No notes
DMC0_BA2	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Bank Address Input 2 Notes: No notes
$\overline{\text{DMC0\_CAS}}$	Output	B	None	None	L	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

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and 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	H	VDD_DMC	Desc: DMC0 Clock (Complement) Notes: No notes
$\overline{\text{DMC0\_CS0}}$	Output	B	None	None	L	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
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

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and 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	L	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: No notes
$\overline{\text{DMC0\_LDQS}}$	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	L	VDD_DMC	Desc: DMC0 On-Die Termination Notes: No notes
$\overline{\text{DMC0\_RAS}}$	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Row Address Strobe Notes: No notes
$\overline{\text{DMC0\_RESET}}$	Output	B	None	None	L	VDD_DMC	Desc: DMC0 Reset Notes: No notes
DMC0_RZQ	a	B	None	None	None	VDD_DMC	Desc: DMC0 External Calibration Resistor Connection Notes: 34 Ω external pull-down must be added
DMC0_UDM	Output	B	None	None	L	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: No notes
$\overline{\text{DMC0\_UDQS}}$	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_VREF0	a		None	None	None	VDD_DMC	Desc: DMC0 Voltage Reference Notes: No notes
DMC0_VREF1	a		None	None	None	VDD_DMC	Desc: DMC0 Voltage Reference Notes: No notes
$\overline{\text{DMC0\_WE}}$	Output	B	None	None	L	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_ANA	Desc: HADC0 Analog Input 0 Notes: Connect to GND if not used
HADC0_VIN1	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 1 Notes: Connect to GND if not used
HADC0_VIN2	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 2 Notes: Connect to GND if not used
HADC0_VIN3	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 3 Notes: Connect to GND if not used
HADC0_VIN4	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 4 Notes: Connect to GND if not used
HADC0_VIN5	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 5 Notes: Connect to GND if not used

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
HADC0_VIN6	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 6 Notes: Connect to GND if not used
HADC0_VIN7	a	NA	None	None	None	VDD_ANA	Desc: HADC0 Analog Input 7 Notes: Connect to GND if not used
HADC0_VREFN	s	NA	None	None	None	VDD_ANA	Desc: HADC0 Ground Reference for ADC Notes: Connect to GND if HADC and TMU are not used
HADC0_VREFP	s	NA	None	None	None	VDD_ANA	Desc: HADC0 External Reference for ADC Notes: Connect to VDD_REF if HADC and TMU are not used
JTG_TCK	Input		Pull-up	Pull-up	None	VDD_EXT	Desc: JTAG Clock Notes: No notes
JTG_TDI	Input		Pull-up	Pull-up	None	VDD_EXT	Desc: JTAG Serial Data In Notes: No notes
JTG_TDO	Output	A	None	High-Z when $\overline{\text{JTG\_TRST}}$ is low, not affected by $\overline{\text{SYS\_HWRST}}$	None	VDD_EXT	Desc: JTAG Serial Data Out Notes: No notes
JTG_TMS	InOut	A	Pull-up	Pull-up	None	VDD_EXT	Desc: JTAG Mode Select Notes: No notes
$\overline{\text{JTG\_TRST}}$	Input		Pull-down	Pull-down	None	VDD_EXT	Desc: JTAG Reset Notes: No notes
MLB0_CLKN	Input	N/A	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Clock (–) Notes: No notes
MLB0_CLKP	Input	N/A	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Clock (+) Notes: No notes
MLB0_DATN	InOut	I	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Data (–) Notes: No notes
MLB0_DATP	InOut	I	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Data (+) Notes: No notes
MLB0_SIGN	InOut	I	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Signal (–) Notes: No notes
MLB0_SIGP	InOut	I	Internal logic ensures that input signal does not float	None	None	VDD_REF	Desc: MLB0 Differential Signal (+) Notes: No notes
PA_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 0 | OSPI0 MISO | SPI2 MISO Notes: See note[2]
PA_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position1 | OSPI0 MOSI | SPI2 MOSI Notes: See note[2]
PA_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 2 | OSPI0 D2 | SPI2 D2 | TWI3 Clock | TIMER0 Timer Alternate Clock 3 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PA_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 3 | OSPI0 D3 | SPI2 D3 | TWI3 Data Notes: See note[2]
PA_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 4 | OSPI0 Clock | SPI2 Clock Notes: See note[2]
PA_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 5 | OSPI0 Slave Select Output 1 | SPI2 Slave Select Output 1 | SPI2 Slave Select Notes: See note[2]
PA_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 6 | OSPI0 D4 | SPI0 Clock | UART0 TX | TIMER0 Timer Alternate Clock 1 Notes: See note[2]
PA_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 7 | OSPI0 D5 | SPI0 MISO | UART0 RX | TIMER0 Timer Alternate Input 0 Notes: See note[2]
PA_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 8 | OSPI0 D6 | SPI0 MOSI | UART0 RTS | TIMER0 Timer Alternate Clock 2 Notes: See note[2]
PA_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 9 | OSPI0 D7 | SPI0 Slave Select Output 1 | UART0 CTS | SPI0 Slave Select Notes: See note[2]
PA_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 10 | SPI1 Clock | TIMER0 Timer 0 | TWI0 Clock Notes: See note[2]
PA_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 11 | HADC0 End of Conversion | SPI1 MISO | TWI0 Data | TIMER0 Timer Alternate Input 4 Notes: See note[2]
PA_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 12 | SPI1 MOSI | SHARC1 Core Flag 0 | TIMER0 Timer 1 Notes: See note[2]
PA_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 13 | SPI1 Slave Select Output 1 | SHARC1 Core Flag 1 | TIMER0 Timer 2 | SPI1 Slave Select Notes: See note[2]
PA_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 14 | SPI1 D2 | TWI2 Clock | UART1 RX | TIMER0 Alternate Clock Input 1 Notes: See note[2]
PA_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTA Position 15 | SPI1 D3 | TWI2 Data | UART1 TX Notes: See note[2]
PB_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 0 | MLB0 Data | TWI1 Clock | UART1 RTS | TIMER0 Alternate Clock Input 3 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PB_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 1 | MLB0 Signal | TWI1 Data | UART1 CTS | TIMER0 Timer Clock Notes: See note[2]
PB_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 2 | LP1 Acknowledge | MLB0 Clock | SHARC1 Core Flag 3 | TIMER0 Timer Alternate Clock 4 Notes: See note[2]
PB_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 3 | SPI2 Slave Select Output 2 | SHARC1 Core Flag 2 | TIMER0 Timer 3 | CNT0 Count Up and Direction Notes: See note[2]
PB_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 4 | LP0 Acknowledge | SPI1 Ready | TIMER0 Timer 4 | CNT0 Zero Marker Notes: See note[2]
PB_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 5 | SPI0 Slave Select Output 2 | SPI2 Ready | TIMER0 Timer 5 | CNT0 Count Down and Gate Notes: See note[2]
PB_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 6 | LP0 Clock | SPI1 Slave Select Output 5 | TRACE0 Trace Clock Notes: See note[2]
PB_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 7 | LP0 D0 | SPI2 Slave Select Output 5 | TRACE0 Trace Data 00 Notes: See note[2]
PB_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 8 | LP0 D1 | SPI1 Slave Select Output 7 | TIMER0 Timer 6 | TRACE0 Trace Data 01 Notes: See note[2]
PB_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 9 | LP0 D2 | SPI2 Slave Select Output 7 | TIMER0 Timer 7 | TRACE0 Trace Data 02 Notes: See note[2]
PB_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 10 | LP0 D3 | SPI1 Slave Select Output 2 | TRACE0 Trace Data 03 Notes: See note[2]
PB_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 11 | LP0 D4 | SPI0 Ready | UART2 RX | TIMER0 Alternate Clock Input 2 Notes: See note[2]
PB_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 12 | LP0 D5 | SPI2 Slave Select Output 3 | UART2 TX Notes: See note[2]
PB_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 13 | LP0 D6 | OSPI0 DQS | SPI1 Slave Select Output 3 | UART2 RTS Notes: See note[2]
PB_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 14 | LP0 D7 | SPI0 Slave Select Output 3 | UART2 CTS Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PB_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTB Position 15 | LP1 D0 | SPI0 Slave Select Output 4 Notes: See note[2]
PC_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 0 | LP1 D1 | SPI1 Slave Select Output 4 | TRACE0 Trace Data 04 | TWI4 Clock Notes: See note[2]
PC_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 1 | LP1 D2 | SPI2 Slave Select Output 4 | TRACE0 Trace Data 05 | TWI4 Data Notes: See note[2]
PC_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 2 | LP1 D3 | SPI1 Slave Select Output 6 | TRACE0 Trace Data 06 | TWI5 Clock Notes: See note[2]
PC_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 3 | LP1 D4 | SPI2 Slave Select Output 6 | TRACE0 Trace Data 07 | TWI5 Data Notes: See note[2]
PC_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 4 | LP1 D5 | OSPI0 Slave Select Output 2 Notes: See note[2]
PC_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 5 | LP1 D6 | OSPI0 Slave Select Output 3 | TIMER0 Timer 8 Notes: See note[2]
PC_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 6 | LP1 D7 | SPI1 Ready Notes: See note[2]
PC_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 7 | LP1 Clock | OSPI0 Slave Select Output 4 | System Fault | TIMER0 Timer 9 Notes: Defaults to GPIO on HPC package. Defaults to SYS_FAULT on LPC package, so external pull-down required to keep signal in deasserted state.
PC_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 8 | EPPI0 D16 | OSPI0 Clock Notes: See note[2]
PC_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 9 | EPPI0 D17 | OSPI0 D3 Notes: See note[2]
PC_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 10 | EPPI0 D18 | OSPI0 D2 Notes: See note[2]
PC_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 11 | EPPI0 D19 | OSPI0 MOSI Notes: See note[2]
PC_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 12 | EPPI0 D20 | OSPI0 MISO Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PC_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 13 | EPPI0 D21 | OSPI0 D7 Notes: See note[2]
PC_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 14 | EPPI0 D22 | OSPI0 D6 Notes: See note[2]
PC_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTC Position 15 | EPPI0 D23 | OSPI0 D5 Notes: See note[2]
PD_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 0 | OSPI0 D4 Notes: See note[2]
PD_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 1 | EPPI0 D06 | OSPI0 Slave Select Output 1 Notes: See note[2]
PD_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 2 | EMAC0 Carrier Sense | SHARC2 Core Flag 10 | UART1 RTS Notes: See note[2]
PD_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 3 | LP0 Acknowl-edgment | SHARC1 Core Flag 11 | UART1 CTS | EMAC0 PTP Aux Input 2 Notes: See note[2]
PD_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 4 | EPPI0 D07 | OSPI0 DQS | UART1 RX | TIMER0 Alternate Clock Input 1 Notes: See note[2]
PD_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 5 | EPPI0 D08 | UART1 TX | EMAC0 PTP Aux Input 1 Notes: See note[2]
PD_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 6 | EMAC0 Receive Error | UART0 CTS Notes: See note[2]
PD_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 7 | EMAC0 Collision Detect | UART0 RTS Notes: See note[2]
PD_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 8 | UART0 RX | TIMER0 Alternate Clock Input 0 Notes: See note[2]
PD_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 9 | UART0 TX Notes: See note[2]
PD_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 10 | EPPI0 D09 | UART2 RX | TIMER0 Alternate Clock Input 2 Notes: See note[2]
PD_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 11 | EPPI0 D10 | UART2 TX Notes: See note[2]
PD_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 12 | EPPI0 D11 | TIMER0 Timer 6 | UART2 RTS Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PD_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 13 | EPPI0 D12 | TIMER0 Timer 7 | UART2 CTS Notes: See note[2]
PD_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 14 | EPPI0 D13 | LP1 Acknowledgment | SHARC2 Core Flag 9 | TWI4 Clock Notes: See note[2]
PD_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTD Position 15 | EPPI0 D14 | TIMER0 Timer 15 | TWI4 Data Notes: See note[2]
PE_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 0 | EPPI0 D15 | SPI3 Slave Select Output 3 | TWI5 Clock Notes: See note[2]
PE_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 1 | EPPI0 Frame Sync 1 | SPI3 Slave Select Output 4 | TWI5 Data | TIMER0 Timer Alternate Clock 15 Notes: See note[2]
PE_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 2 | EPPI0 Frame Sync 2 | HADC0 MUX0 | SPI1 Slave Select Output 4 | TWI0 Clock Notes: See note[2]
PE_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 3 | EPPI0 Frame Sync 3 | HADC0 MUX2 | SPI2 Slave Select Output 3 | TWI0 Data | TIMER0 Timer Alternate Clock 4 Notes: See note[2]
PE_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 4 | EPPI0 Clock | HADC0 MUX1 | TWI2 Clock Notes: See note[2]
PE_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 5 | EPPI0 D00 | TWI2 Data Notes: See note[2]
PE_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 6 | EPPI0 D01 | SHARC1 Core Flag 2 | TIMER0 Timer 8 Notes: See note[2]
PE_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 7 | EPPI0 D02 | SPI1 Ready | SHARC1 Core Flag 3 | TIMER0 Timer 9 Notes: See note[2]
PE_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 8 | EPPI0 D03 | TIMER0 Timer 0 Notes: See note[2]
PE_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 9 | EPPI0 D04 | TIMER0 Timer 1 | EMAC0 PTP Aux Input 3 Notes: See note[2]
PE_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 10 | EPPI0 D05 | SPI0 Slave Select Output 4 | TIMER0 Timer 2 Notes: See note[2]
PE_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 11 | EMAC1 Reference Clock | SHARC2 Core Flag 7 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PE_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 12 | EMAC1 Transmit Enable | SHARC1 Core Flag 8 Notes: See note[2]
PE_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 13 | EMAC1 Transmit Data D0 | SHARC2 Core Flag 13 Notes: See note[2]
PE_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 14 | EMAC1 Transmit Data D1 | SPI2 Slave Select Output 7 Notes: See note[2]
PE_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTE Position 15 | EMAC1 Receive Data D0 | SPI0 Slave Select Output 6 Notes: See note[2]
PF_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 0 | EMAC1 Receive Data D1 | SPI0 Slave Select Output 7 | SPI3 Ready Notes: See note[2]
PF_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 1 | EMAC1 Serial Management Bidirectional Data | SHARC1 Core Flag 7 Notes: See note[2]
PF_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 2 | EMAC1 Serial Management Clock | SHARC1 Core Flag 6 Notes: See note[2]
PF_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 3 | EMAC1 Carrier Sense | SHARC1 Core Flag 10 | USBC0 USBC Stop Output Control Notes: See note[2]
PF_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 4 | USBC0 USBC Data 7 | TIMER0 Timer Alternate Clock 13 Notes: See note[2]
PF_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 5 | MLB0 Clock Output | USBC0 USBC Data 6 | TIMER0 Timer Clock Notes: See note[2]
PF_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 6 | SPI1 Slave Select Output 7 | SHARC2 Core Flag 2 | USBC0 USBC Data 5 Notes: See note[2]
PF_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 7 | SPI3 Slave Select Output 2 | SHARC2 Core Flag 3 | USBC0 USBC Data 4 Notes: See note[2]
PF_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 8 | SPI3 Slave Select Output 6 | SHARC2 Core Flag 8 | TIMER0 Timer 11 | USBC0 USBC Next Data Control Notes: See note[2]
PF_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 9 | SHARC1 Core Flag 9 | USBC0 USBC Data Direction Control Notes: See note[2]
PF_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 10 | SHARC2 Core Flag 4 | USBC0 USBC Data 3 Notes: See note

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PF_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 11 | SHARC2 Core Flag 5 | USBC0 USBC Data 2 Notes: See note[2]
PF_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 12 | SHARC1 Core Flag 13 | USBC0 USBC Data 1 Notes: See note[2]
PF_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 13 | SHARC1 Core Flag 12 | USBC0 USBC Data 0 Notes: See note[2]
PF_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 14 | SHARC2 Core Flag 12 | USBC0 USBC Clock Signal Notes: See note[2]
PF_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTF Position 15 | CANFD0 Receive | TIMER0 Alternate Clock Input 4 Notes: See note[2]
PG_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 0 | CANFD0 Transmit | TIMER0 Alternate Clock 10 Notes: See note[2]
PG_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 1 | CANFD1 Receive | SPI2 Slave Select Output 6 | TIMER0 Alternate Clock Input 11 Notes: See note[2]
PG_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 2 | CANFD1 Transmit | SPI0 Slave Select Output 5 Notes: See note[2]
PG_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 3 | UART3 TX Notes: See note[2]
PG_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 4 | UART3 RX | TIMER0 Alternate Clock Input 3 Notes: See note[2]
PG_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 5 | SPI3 Clock | TIMER0 Alternate Clock 11 Notes: See note[2]
PG_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 6 | SPI3 MISO Notes: See note[2]
PG_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 7 | SPI3 MOSI | TIMER0 Alternate Clock 12 Notes: See note[2]
PG_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 7 | SPI3 Slave Select Output 1 | SPI3 Slave Select Input Notes: See note[2]
PG_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 9 | SPI1 Slave Select Output 6 | SHARC1 Core Flag14 | UART3 RTS | TIMER0 Alternate Clock 1 Notes: See note[2]
PG_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 10 | SHARC2 Core Flag14 | UART3 CTS | TIMER0 Alternate Clock 3 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PG_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 11 | SHARC2 Core Flag 15 Notes: See note[2]
PG_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 12 | OSPI0 Slave Select Output 3 | SPI2 Slave Select Output 4 | SHARC2 Core Flag 11 | TIMER0 Timer 3 Notes: See note[2]
PG_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 13 | OSPI0 Slave Select Output 4 | SHARC1 Core Flag 0 | SHARC2 Core Flag 6 Notes: See note[2]
PG_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 14 | SPI0 Slave Select Output 2 | TIMER0 Timer 10 | TIMER0 Alternate Clock Input 10 Notes: See note[2]
PG_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTG Position 15 | SPI2 Ready | SPI3 Slave Select Output 5 | TIMER0 Timer 11 Notes: See note[2]
PH_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 0 | SPI3 Slave Select Output 7 | TIMER0 Timer 12 | TIMER0 Alternate Clock Input 12 Notes: See note[2]
PH_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 1 | SPI0 Ready | TIMER0 Timer 13 | TIMER0 Alternate Clock Input 13 Notes: See note[2]
PH_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 2 | SPI2 Slave Select Output 2 | SHARC1 Core Flag 1 | TIMER0 Timer 14 Notes: See note[2]
PH_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 3 | EMAC0 Serial Management Clock | TRACE0 Trace D08 Notes: See note[2]
PH_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 4 | EMAC0 Serial Management Bidirectional Data | TRACE0 Trace D09 Notes: See note[2]
PH_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 5 | EMAC0 Receive Data D0 | TRACE0 Trace D10 Notes: See note[2]
PH_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 6 | EMAC0 Receive Data D1 | TRACE0 Trace D11 Notes: See note[2]
PH_07	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 7 | EMAC0 Receive Reference Clock | TRACE0 Trace D12 Notes: See note[2]
PH_08	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 8 | EMAC0 Receive Data Valid | TRACE0 Trace D13 Notes: See note[2]

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PH_09	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 9 | EMAC0 Transmit Data D0 | TRACE0 Trace D14 Notes: See note[2]
PH_10	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 10 | EMAC0 Transmit Data D1 | TRACE0 Trace D15 Notes: See note[2]
PH_11	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 11 | EMAC0 Receive Data D2 Notes: See note[2]
PH_12	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 12 | EMAC0 Receive Data D3 Notes: See note[2]
PH_13	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 13 | EMAC0 Transmit Enable Notes: See note[2]
PH_14	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 14 | EMAC0 Transmit Clock Notes: See note[2]
PH_15	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTH Position 15 | EMAC0 Transmit Data D2 Notes: See note[2]
PI_00	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 0 | EMAC0 Transmit Data D3 Notes: See note[2]
PI_01	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 1 | EMAC0 PTP Clock Input 0 | SHARC2 Core Flag 0 | TWI3 Clock Notes: See note[2]
PI_02	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 2 | EMAC0 PTP Aux Input 0 | SHARC2 Core Flag 1 | TWI3 Data Notes: See note[2]
PI_03	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 3 | EMAC0 Pulse Per Second Output | SHARC1 Core Flag 4 | TWI1 Clock Notes: See note[2]
PI_04	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 4 | EMAC0 Pulse Per Second Output | SHARC1 Core Flag 5 | TWI1 Data Notes: See note[2]
PI_05	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 5 | EMAC0 Pulse Per Second Output | OSPI0 Slave Select Output 2 | SHARC1 Core Flag 15 | TIMER0 Alternate Clock 2 Notes: See note[2]
PI_06	InOut	A	Programmable pull-up/pull-down[1]	None	None	VDD_EXT	Desc: PORTI Position 6 | EMAC0 Pulse Per Second Output | TIMER0 Alternate Clock 14 Notes: See note[2]
SYS_BMODE0	Input	NA	None	None	None	VDD_EXT	Desc: Boot Mode Control 0 Notes: Cannot be left unconnected
SYS_BMODE1	Input	NA	None	None	None	VDD_EXT	Desc: Boot Mode Control 1 Notes: Cannot be left unconnected

Table 25. ADSP-2159x/ADSP-SC59x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
SYS_BMODE2	Input	NA	None	None	None	VDD_EXT	Desc: Boot Mode Control 2 Notes: Cannot be left unconnected
SYS_CLKIN0	a	NA	None	None	None	VDD_REF	Desc: Clock/Crystal Input Notes: Cannot be left unconnected
SYS_CLKIN1	a	NA	None	None	None	VDD_REF	Desc: Clock/Crystal Input Notes: Cannot be left unconnected
SYS_CLKOUT	Output	A	None	None	L	VDD_EXT	Desc: Processor Clock Output Notes: No notes
SYS_FAULT	InOut	A	None	None	None	VDD_EXT	Desc: Active-High Fault Output Notes: External pull-down required to keep signal in deasserted state
$\overline{\text{SYS\_FAULT}}$	InOut	A	None	None	None	VDD_EXT	Desc: Active-Low Fault Output Notes: External pull-up required to keep signal in deasserted state
$\overline{\text{SYS\_HWRST}}$	Input	NA	None	None	None	VDD_EXT	Desc: Processor Hardware Reset Control Notes: Cannot be left unconnected
$\overline{\text{SYS\_RESOUT}}$	Output	A	None	None	L	VDD_EXT	Desc: Reset Output Notes: No notes
SYS_XTAL0	a	NA	None	None	None	VDD_REF	Desc: Crystal Output Notes: Leave unconnected if an oscillator provides SYS_CLKIN0
SYS_XTAL1	a		None	None	None	VDD_REF	Desc: Crystal Output Notes: Leave unconnected if an oscillator provides SYS_CLKIN1
VDD_ANA	s		None	None	None		Desc: Analog VDD 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_INT	s		None	None	None		Desc: Internal Voltage Domain Notes: No notes
VDD_PLL	s		None	None	None		Desc: PLL VDD Notes: No notes
VDD_REF	s		None	None	None		Desc: External Voltage Domain Notes: No notes

[1] Disabled by default.

[2] When present, the internal pull-up/pull-down design holds the internal path from the pins at the expected logic levels. To pull up or pull down the external pads to the expected logic levels, use external resistors.

SPECIFICATIONS

Specifications are subject to change without notice. For information about product specifications, contact your Analog Devices, Inc., representative.

OPERATING CONDITIONS

All specifications and characteristics apply across the entire operating conditions range unless otherwise noted.

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DD_INT}	Internal (Core) Supply Voltage	600 MHz ≤ CCLK ≤ 1 GHz	0.95	1.00	1.05	V
V_{DD_EXT}	External (I/O) Supply Voltage		3.13	3.30	3.47	V
V_{DD_ANA}	Analog Power Supply Voltage		1.71	1.80	1.89	V
V_{DD_DMC}[1]	DDR3L Controller Supply Voltage		1.34	1.39	1.44	V
	DDR3 Controller Supply Voltage		1.425	1.500	1.575	V
V_{DD_REF}[2]	External (I/O) Reference Supply Voltage		1.71	1.80	1.89	V
V_{DDR_VREF}[3]	DDR3 Reference Voltage		$0.49 \times V_{DD\_DMC}$	$0.50 \times V_{DD\_DMC}$	$0.51 \times V_{DD\_DMC}$	V
$V_{DELTA\_EXT\_REF}$[4]	($V_{DD\_EXT} - V_{DD\_REF}$) and ($V_{DD\_EXT} - V_{DD\_ANA}$)		−1.89		+1.89	V
V_{HADC_REF}[5]	HADC Reference Voltage		1.71	1.80	V_{DD_ANA}	V
V_{HADC0_VINx}	HADC Input Voltage		0		V_{HADC_REF} + 0.09	V
V_{IH}[6]	High Level Input Voltage	V_{DD_EXT} = 3.47 V	2.0			V
$V_{IHCLKIN}$[2]	High Level Clock Input Voltage	V_{DD_REF} = 1.89 V	$0.87 \times V_{DD\_REF}$		V_{DD_REF}	V
V_{IL}[6]	Low Level Input Voltage	V_{DD_EXT} = 3.13 V			0.8	V
$V_{ILCLKIN}$[2]	Low Level Clock Input Voltage	V_{DD_REF} = 1.71 V	−0.30		$+0.35 \times V_{DD\_REF}$	V
V_{IL_DDR3L}[7]	Low Level Input Voltage	V_{DD_DMC} = 1.34 V			V_{DDR_VREF} − 0.175	V
V_{IL_DDR3}[7]	Low Level Input Voltage	V_{DD_DMC} = 1.425 V			V_{DDR_VREF} − 0.175	V
V_{IH_DDR3L}[7]	High Level Input Voltage	V_{DD_DMC} = 1.44 V	V_{DDR_VREF} + 0.175			V
V_{IH_DDR3}[7]	High Level Input Voltage	V_{DD_DMC} = 1.575 V	V_{DDR_VREF} + 0.175			V
CONSUMER GRADE						
T_J	Junction Temperature 400-Ball BGA_ED		0		125	°C
INDUSTRIAL GRADE						
T_J	Junction Temperature 400-Ball BGA_ED		−40		+125	°C
AUTOMOTIVE GRADE[8]						
T_J	Junction Temperature 400-Ball BGA_ED		−40		+125	°C

[1] Applies to DDR3L/DDR3 signals.
[2] Applies to SYS_CLKIN0 pin.
[3] Applies to DMC0_VREF0 and DMC0_VREF1 pins.
[4] See Figure 9.
[5] V_{HADC_VREF} must always be less than V_{DD_ANA}.
[6] Parameter value applies to all input and bidirectional pins except the DMC pins.
[7] This parameter applies to all DMC0 pins.
[8] Automotive application use profile only. Not supported for nonautomotive use. Contact Analog Devices for more information.

Clock Related Operating Conditions

Table 26 describes the core clock, system clock, and peripheral clock timing requirements. The data presented in the table applies to all speed grades except where noted.

Table 26. Clock Operating Conditions

Parameter		Conditions	Min	Typ	Max	Unit
f_{CCLK}	Core Clock (CCLK) Frequency	$f_{CCLK} = 2 \times f_{SYSCLK}$	600		1000	MHz
f_{SYSCLK}	SYSCLK Frequency[1]		300		500	MHz
f_{SCLK0}	SCLK0 Frequency	$f_{SYSCLK} = N \times f_{SCLK0}$ where N = 2 or 4 or 6	30		125	MHz
f_{SCLK1}	SCLK1 Frequency	$f_{SYSCLK} \geq f_{SCLK1}$			333.33	MHz
f_{DCLK}	DDR3 Clock (DCLK) Frequency[2]	All combinations are supported except for: [f_{CCLK} > 800 MHz and T_j < 0°C and f_{CCLK}:f_{DCLK} = 2:1]	300		800	MHz
f_{OCLK}	Output Clock (OCLK) Frequency[3]				125	MHz
$f_{SYS\_CLKOUTJ}$	SYS_CLKOUT Period Jitter[4, 5]			±1		%
$f_{LCLKTPROG}$	Programmed Link Port Transmit Clock				125	MHz
$f_{LCLKREXT}$	External Link Port Receive Clock[6, 7]	$f_{LCLKREXT} \leq f_{OCLK\_0}$			125	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Transmitting Data and Frame Sync				62.5	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Receiving Data or Frame Sync				31.25	MHz
$f_{SPTCLKEXT}$	External SPT Clock When Receiving Data and Frame Sync[6, 7]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			62.5	MHz
$f_{SPTCLKEXT}$	External SPT Clock Transmitting Data or Frame Sync[6, 7]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			31.25	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data	f_{SPICLK}:f_{SCLK0} ratio = 1:1			75	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data	f_{SPICLK}:f_{SCLK0} ratio = 1:1			75	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data	f_{SPICLK}:f_{SCLK0} ratio = 1:2			62.5	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data	f_{SPICLK}:f_{SCLK0} ratio = 1:2			62.5	MHz
$f_{SPICLKEXT}$	External SPI Clock When Receiving Data[6, 7]	$f_{SPICLKEXT} \leq f_{CDU\_CLKO0}$			62.5	MHz
$f_{SPICLKEXT}$	External SPI Clock When Transmitting Data[6, 7]	$f_{SPICLKEXT} \leq f_{CDU\_CLKO0}$			45	MHz
$f_{TMRCLKEXT}$	External Timer Clock (TMx_CLK)	$f_{TMRCLKEXT} \leq f_{SCLK0} / 4$			31.25	MHz

[1] When using MLB, there is a requirement that the f_{SYSCLK} value must be a minimum of 100 MHz for 3-pin mode and for all supported speeds.
[2] To ensure proper operation of the DDR3/3L, all the DDR3/3L guidelines must be strictly followed.
[3] f_{OCLK} must not exceed f_{SCLK0} when selected as SYS_CLKOUT.
[4] 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.
[5] The value in the Typ field is the percentage of the SYS_CLKOUT period.
[6] 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.
[7] The peripheral external clock frequency must also be less than or equal to the frequency that clocks the peripheral.

Table 27. Phase-Locked Loop (PLL) Operating Conditions

Parameter		Min	Max	Unit
f_{PLLCLK}	PLL Clock Frequency	1.20	2.00	GHz

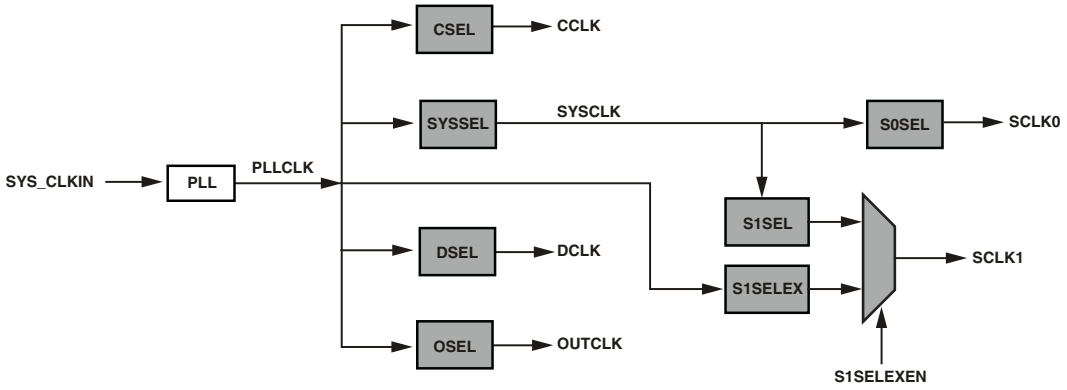


Figure 7. Clock Relationships and Divider Values

ELECTRICAL CHARACTERISTICS

All specifications and characteristics apply across the entire operating conditions range unless otherwise noted.

Parameter		Conditions	Min	Typ	Max	Unit
V_{OH}	High Level Output Voltage	V_{DD_EXT} = minimum, (I_{OH} = –2.0 mA, DS1)[1], (I_{OH} = –4.0 mA, DS2)[2]	2.4			V
V_{OL}	Low Level Output Voltage	V_{DD_EXT} = minimum, (I_{OL} = 2.0 mA, DS1)[1], (I_{OL} = 4.0 mA, DS2)[2]			0.4	V
V_{OH_XTAL}[3]	High Level Output Voltage	V_{DD_REF} = minimum, I_{OH} = –1.0 mA	1.26			V
V_{OL_XTAL}[3]	Low Level Output Voltage	V_{DD_REF} = minimum, I_{OL} = 1.0 mA			0.45	V
V_{OH_DDR3L}[4]	High Level Output Voltage for DDR3L Drive Strength =100 Ω	V_{DD_DDR} = minimum, I_{OH} = –1.0 mA	0.963			V
V_{OL_DDR3L}[4]	Low Level Output Voltage for DDR3L Drive Strength =100 Ω	V_{DD_DDR} = minimum, I_{OL} = 1.0 mA			0.32	V
V_{OH_DDR3}[5]	High Level Output Voltage for DDR3 Drive Strength = 100 Ω	V_{DD_DDR} = minimum, I_{OH} = –1.0 mA	1.105			V
V_{OL_DDR3}[5]	Low Level Output Voltage for DDR3 Drive Strength = 100 Ω	V_{DD_DDR} = minimum, I_{OL} = 1.0 mA			0.32	V
I_{IH}[6]	High Level Input Current	V_{DD_EXT} = maximum, V_{IN} = V_{DD_EXT} maximum			10	µA
I_{IL}[6]	Low Level Input Current	V_{DD_EXT} = maximum, V_{IN} = 0 V			10	µA
I_{IL_PU}[7]	Low Level Input Current Pull-Up	V_{DD_EXT} = maximum, V_{IN} = 0 V			200	µA
I_{IH_PD}[8]	High Level Input Current Pull-Down	V_{DD_EXT} = maximum, V_{IN} = V_{DD_EXT} maximum			200	µA
I_{OZH}[9]	Three-State Leakage Current	V_{DD_EXT}/V_{DD_DDR} = maximum, V_{IN} = V_{DD_EXT}/V_{DD_DDR} maximum			10	µA
I_{OZL}[9]	Three-State Leakage Current	V_{DD_EXT}/V_{DD_DDR} = maximum, V_{IN} = 0 V			10	µA
C_{IN}[10]	Input Capacitance	T_J = 25°C			5	pF
I_{DD_IDLE}	V_{DD_INT} Current in Idle	f_{CCLK} = 1000 MHz ASF_{SHARC1} = 0.41 ASF_{SHARC2} = 0.41 f_{SYSCLK} = 500 MHz f_{SCLK0} = 125 MHz f_{SCLK1} = 333.33 MHz (Other clocks are disabled) No peripheral or DMA activity T_J = 25°C V_{DD_INT} = 1.0 V		1376		mA
I_{DD_TYP}	V_{DD_INT} Current	f_{CCLK} = 1000 MHz ASF_{SHARC1} = 1.0 ASF_{SHARC2} = 1.0 f_{SYSCLK} = 500 MHz f_{SCLK0} = 125 MHz f_{SCLK1} = 333.33 MHz (Other clocks are disabled) No peripheral or DMA activity T_J = 25°C V_{DD_INT} = 1.0 V		2364		mA

Parameter		Conditions	Min	Typ	Max	Unit
I_{DD_INT}	V_{DD_INT} Current	$f_{CCLK} > 0$ MHz $f_{SCLK0/1} \geq 0$ MHz			$I_{DD\_INT\_TOT}$ See equation in the Total Internal Power Dissipation section.	mA

[1] Applies to all output and bidirectional pins operating at less than or equal to 62.5 MHz, except DMC and SYS_XTAL0.
[2] Applies to all output and bidirectional pins operating above 62.5 MHz and less than or equal to 125 MHz, except DMC.
[3] Applies to SYS_XTAL0 pin.
[4] Applies to all DMC output and bidirectional signals in DDR3L mode.
[5] Applies to all DMC output and bidirectional signals in DDR3 mode.
[6] Applies to input pins: SYS_BMODE2-0, SYS_CLKIN, and $\overline{\text{SYS\_HWRST}}$.
[7] Applies to input pins with internal pull-ups: JTG_TDI, JTG_TMS, and JTG_TCK.
[8] Applies to JTAG_TRST signal.
[9] Applies to signals: PA15 to PA0, PB15 to PB0, PC7 to PC0, DAI0_PINx, DAI1_PINx, DMC0_DQx, DMC0_LDQS, DMC0_UDQS, $\overline{\text{DMC0\_LDQS}}$, $\overline{\text{DMC0\_UDQS}}$, $\overline{\text{SYS\_FAULT}}$, and JTG_TDO.
[10] Applies to all signal pins.

Total Internal Power Dissipation

Total power dissipation has two components:

- Static, including leakage current
- 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.

$$I_{DD\_INT\_TOT} = I_{DD\_INT\_STATIC} + I_{DD\_INT\_CCLK\_SHARC1\_DYN} + I_{DD\_INT\_CCLK\_SHARC2\_DYN} + I_{DD\_INT\_DCLK\_DYN} + I_{DD\_INT\_SYSCLK\_DYN} + I_{DD\_INT\_SCLK0\_DYN} + I_{DD\_INT\_SCLK1\_DYN} + I_{DD\_INT\_OCLK\_DYN} + I_{DD\_INT\_ACCL\_DYN} + I_{DD\_INT\_DMA\_DR\_DYN}$$

where $I_{DD\_INT\_STATIC}$ is the sole contributor to the static power dissipation component and is specified as a function of voltage (V_{DD_INT}) and junction temperature (T_J) in Table 28.

Table 28. Static Current—$I_{DD\_INT\_STATIC}$ (mA)

	Voltage (V_{DD_INT})		
T_J (°C)	0.95 V	1.00 V	1.05 V
–40	17	23	32
–20	30	39	52
–10	40	52	68
0	55	70	89
+10	76	95	119
+25	123	149	183
+40	199	238	287
+55	314	371	440
+70	485	567	666
+85	740	859	1001
+100	1120	1293	1501
+105	1287	1485	1721
+115	1690	1951	2261
+125	2205	2551	2965

The other eight addends in the $I_{DD\_INT\_TOT}$ equation comprise the dynamic power dissipation component and fall into four broad categories: application dependent currents, clock currents, currents from high speed peripheral operation, and data transmission currents.

Application Dependent Current

The application dependent currents include the dynamic current in the core clock domain of the SHARC+ core, as well as the dynamic current in the accelerator block.

Dynamic current consumed by the core is subject to an activity scaling factor (ASF) that represents application code running on the processor core (see Table 29). The ASF is combined with the CCLK frequency and V_{DD_INT} dependent dynamic current data in Table 30 to calculate this portion of the total dynamic power dissipation component.

$$I_{DD\_INT\_CCLK\_SHARC\_DYN} = \text{Table 30} \times ASF_{SHARC}$$

Table 29. Activity Scaling Factors for the SHARC+® Core 1 and Core 2 (ASF_{SHARC1} and ASF_{SHARC2})

I_{DD_INT} Power Vector	ASF
$I_{DD\text{-}LS}$	0.32
$I_{DD\text{-}IDLE}$	0.41
$I_{DD\text{-}NOP}$	0.61
$I_{DD\text{-}TYP\_3070}$	0.79
$I_{DD\text{-}TYP\_5050}$	0.90
$I_{DD\text{-}TYP\_7030}$	1.00
$I_{DD\text{-}PEAK\_100}$	1.09

Table 30. Dynamic Current for SHARC+®Core (mA, with ASF = 1.00)

	Voltage (V_{DD_INT})		
f_{CCLK} (MHz)	0.95 V	1.00 V	1.05 V
600	481	506	532
650	521	549	576
700	561	591	620
750	601	633	665
800	641	675	709
850	682	717	753
900	722	760	798
950	762	802	842
1000	802	844	886

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.

$I_{DD\_INT\_SYSCLK\_DYN}$ (mA) = 0.910 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK0\_DYN}$ (mA) = 0.514 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK1\_DYN}$ (mA) = 0.023 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_DCLK\_DYN}$ (mA) = 0.142 × f_{DCLK} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_OCLK\_DYN}$ (mA) = 0.128 × f_{OCLK} (MHz) × V_{DD_INT} (V)

Data Transmission Current

The data transmission current represents the power dissipated when moving data throughout the system via DMA. This current is proportional to the data rate.

ABSOLUTE MAXIMUM RATINGS

Stresses at or above those listed in Table 31 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 31. Absolute Maximum Ratings

Parameter	Rating
Internal (Core) Supply Voltage (V_{DD_INT})	–0.3 V to +1.05 V
External (I/O) Supply Voltage (V_{DD_EXT})	–0.3 V to +3.47 V
External (I/O) Reference Supply Voltage (V_{DD_REF})	–0.3 V to +1.89 V
(V_{DD_EXT} – V_{DD_REF}) and (V_{DD_EXT} – V_{DD_ANA}) ($V_{DELTA\_EXT\_REF}$)	–1.89 V to +1.89 V
DDR3 Controller Supply Voltage (V_{DD_DMC})	–0.3 V to +1.60 V
Analog Supply Voltage (V_{DD_ANA})	–0.3 V to +1.89 V
HADC Reference Voltage (V_{HADC_REF})	–0.3 V to +1.89 V
DDR3 Input Voltage[1]	–0.3 V to +1.60 V
Digital Input Voltage[1, 2]	–0.3 V to +3.47 V
TWI Input Voltage[1, 3]	–0.3 V to +3.47 V
Output Voltage Swing	–0.3 V to V_{DD_EXT} +0.5 V
Analog Input Voltage[4]	–0.2 V to V_{DD_ANA} +0.09 V

Table 31. Absolute Maximum Ratings (Continued)

Parameter	Rating
I_{OH}/I_{OL} Current per Signal[2]	6 mA (maximum)
Storage Temperature Range	–65°C to +150°C
Junction Temperature While Biased	125°C

[1] Applies only when the related power supply (V_{DD_DMC} or V_{DD_EXT}) is within specification. When the power supply is below specification, the range is the voltage being applied to that power domain ± 0.2 V.
[2] Applies to 100% transient duty cycle.
[3] Applies to TWI_SCL and TWI_SDA.
[4] Applies only when V_{DD_ANA} is within specifications and ≤ 1.8 V. When V_{DD_ANA} is within specifications and > 1.8 V, the maximum rating is 1.89 V. When V_{DD_ANA} is below specifications, the range is V_{DD_ANA} ± 0.09 V.

Table 32. Maximum Duty Cycle for Input Transient Voltage for V_{DD_INT} and V_{DD_EXT}

V_{DD_INT} (V)[1]	V_{DD_EXT} (V)[1]	Maximum Duty Cycle[2]
1.120		5%
1.103		10%
1.086		20%
1.077		30%
1.065		50%
1.056		75%
1.050	3.470	100%

[1] 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 voltages specified and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.
[2] Duty cycle refers to the percentage of time the signal exceeds the value for the 100% case. This is equivalent to the measured duration of a single instance of overshoot or undershoot as a percentage of the period of occurrence.

Table 33. Maximum Duty Cycle for Input Transient Voltage

3.3 V V_{IN} Max (V)[1]	1.8 V V_{IN} Max (V)[1]	Maximum Duty Cycle[2]
3.47	1.89	100%

[1] 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 voltages specified and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.
[2] Duty cycle refers to the percentage of time the signal exceeds the value for the 100% case. This is equivalent to the measured duration of a single instance of overshoot or undershoot as a percentage of the period of occurrence.

ESD CAUTION

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.

TIMING SPECIFICATIONS

All specifications and characteristics apply across the entire operating conditions range unless otherwise noted.

Power-Up Reset Timing

Table 34 and Figure 8 show the relationship between power supply startup and processor reset timing, as relating to the clock generation unit (CGU) and the reset control unit (RCU).

In Figure 8, the $V_{DD\_SUPPLIES}$ are V_{DD_INT}, V_{DD_PLL}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_REF}, and V_{DD_ANA}. The $V_{DELTA\_EXT\_REF}$ specification must be met at all times, including during power-up reset and when powering down the device (Figure 9).

Table 34. Power-Up Reset Timing

Parameter		Min	Max	Unit
Timing Requirement				
$t_{RST\_IN\_PWR}$	$\overline{SYS\_HWRST}$ Deasserted after $V_{DD\_SUPPLIES}$ (V_{DD_INT}, V_{DD_PLL}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_REF}, V_{DD_ANA}) and SYS_CLKIN0 are Stable and Within Specification	11 × t_{CKIN}		ns

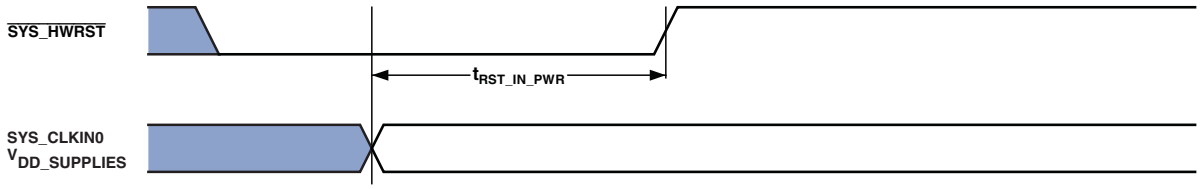


Figure 8. Power-Up Reset Timing

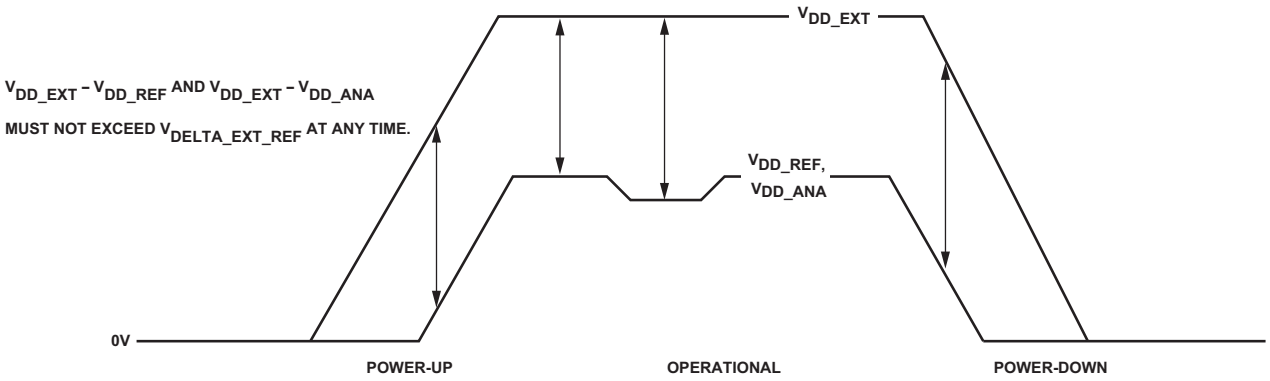


Figure 9. Power-Up and Power-Down Voltage Delta Requirement

Clock and Reset Timing

Table 35 and Figure 10 describe clock and reset operations related to the CGU and RCU. Per the CCLK, SYSCLK, SCLKx, DCLK, and OCLK timing specifications in Table 26 (Clock Operating Conditions), combinations of SYS_CLKIN0 and clock multipliers must not select clock rates in excess of the maximum instruction rate of the processor.

Table 35. Clock and Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
f_{CKIN}	SYS_CLKIN0 Frequency (Crystal)[1, 2]	20	30	MHz
	SYS_CLKIN0 Frequency (External SYS_CLKIN0)[1, 2]	20	30	MHz
t_{CKINL}	SYS_CLKIN0 Low Pulse[1]	16.67		ns
t_{CKINH}	SYS_CLKIN0 High Pulse[1]	16.67		ns
t_{WRST}	$\overline{\text{RESET}}$ Asserted Pulse Width Low[3]	11 × t_{CKIN}		ns

[1] Applies to PLL bypass mode and PLL nonbypass mode.
[2] The t_{CKIN} period (see Figure 10) equals $1/f_{CKIN}$.
[3] Applies after power-up sequence is complete. See Table 34 and Figure 8 for power-up reset timing.

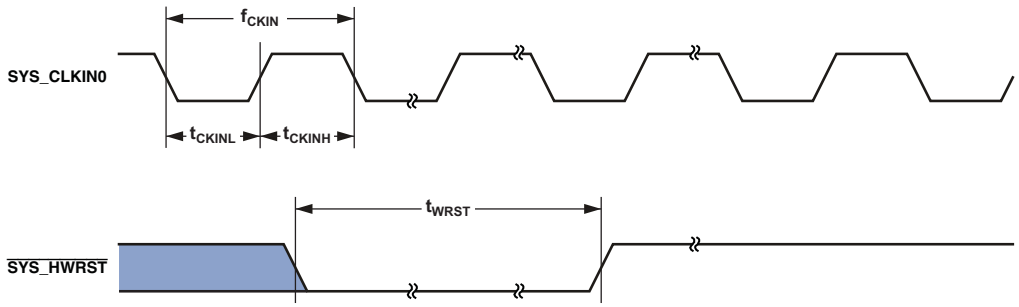


Figure 10. Clock and Reset Timing

Dynamic Memory Controller (DMC)—Clock, Control, Write and Read Cycle Timing

The DMC clock, control, write and read timings comply with the JEDEC standards. To ensure proper operation of the DDR3/3L, all DDR3/3L guidelines must be strictly followed.

Serial Ports (SPORTs)

To determine whether a device is compatible with the SPORT at clock speed n, the following specifications must be confirmed: frame sync delay and frame sync setup and hold; data delay and data setup and hold; and serial clock (SPTx_CLK) width. In Figure 11, either the rising edge or the falling edge of SPTx_A/BCLK (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 megahertz is set by the following equation:

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

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

where *CLKDIV* is a field in the SPORT_DIV register that can be set from 0 to 65,535.

Table 36. SPORTs—External Clock[1]

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]	3		ns
t_{SDRE}	Receive Data Setup Before Receive SPTx_CLK[2]	2		ns
t_{HDRE}	Receive Data Hold After SPTx_CLK[2]	3		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]		12	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]		12	ns
t_{HDTE}	Transmit Data Hold After Transmit SPTx_CLK[4]	2		ns

[1] Specifications apply to all four 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 ideal maximum external SPTx_CLK. For the external SPTx_CLK ideal maximum frequency, see the $f_{SPTCLKEXT}$ specification in Table 26.
[4] Referenced to drive edge.

Table 37. SPORTs—Internal Clock[1]

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]	11		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]	3.6		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]	–3		ns
t_{DDTI}	Transmit Data Delay After SPTx_CLK[3]		3.5	ns
t_{HDTI}	Transmit Data Hold After SPTx_CLK[3]	–3		ns
$t_{SPTCLKIW}$	SPTx_CLK Width[4]	$0.5 \times t_{SPTCLKPROG} - 1.5$		ns
$t_{SPTCLKW}$	SPTx_CLK Period[4]	$t_{SPTCLKPROG} - 1.5$		ns

[1] Specifications apply to all four SPORTs.
[2] Referenced to the sample edge.
[3] Referenced to drive edge.
[4] See Table 26 for details on the minimum period that can be programmed for $f_{SPTCLKPROG}$.

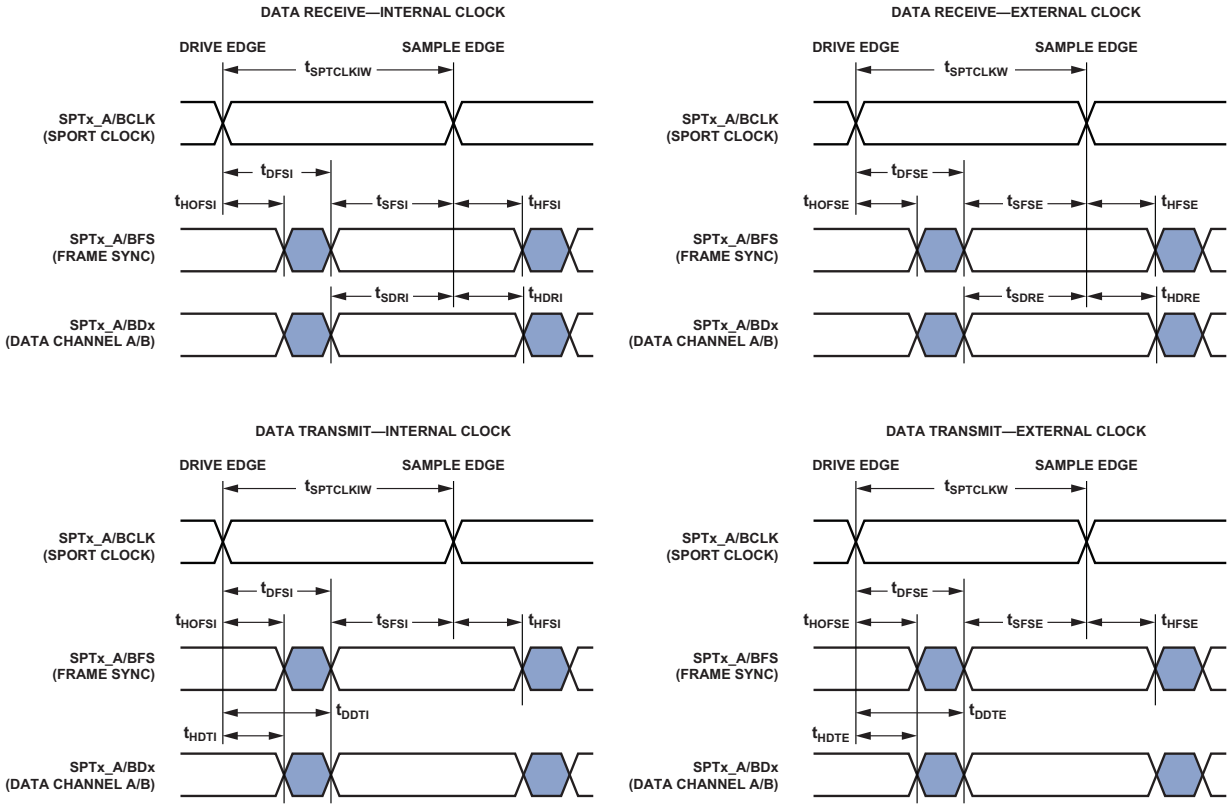


Figure 11. SPORTs

Table 38. SPORTs—Enable and Three-State[1]

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 four SPORTs.
[2] Referenced to drive edge.

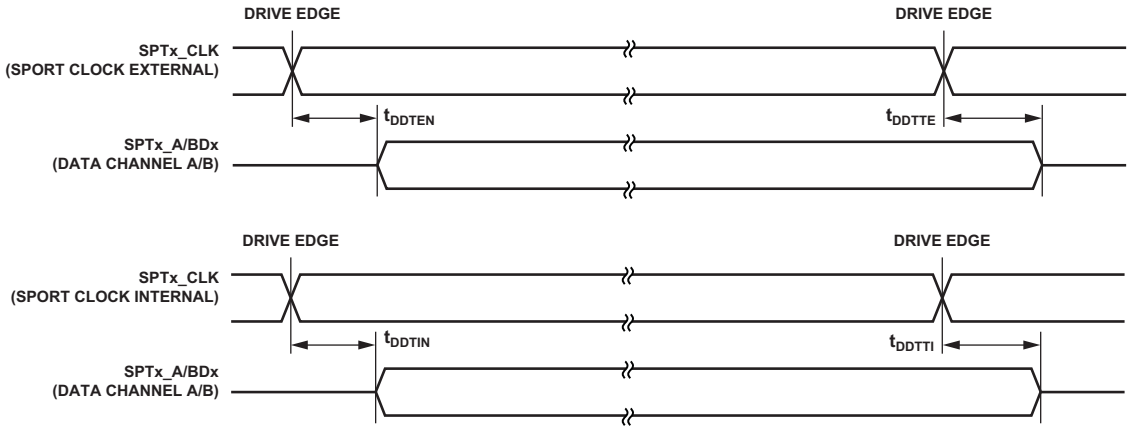


Figure 12. SPORTs—Enable and Three-State

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 39. SPORTs—Transmit Data Valid (TDV)[1]

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 four SPORTs.
[2] Referenced to drive edge.

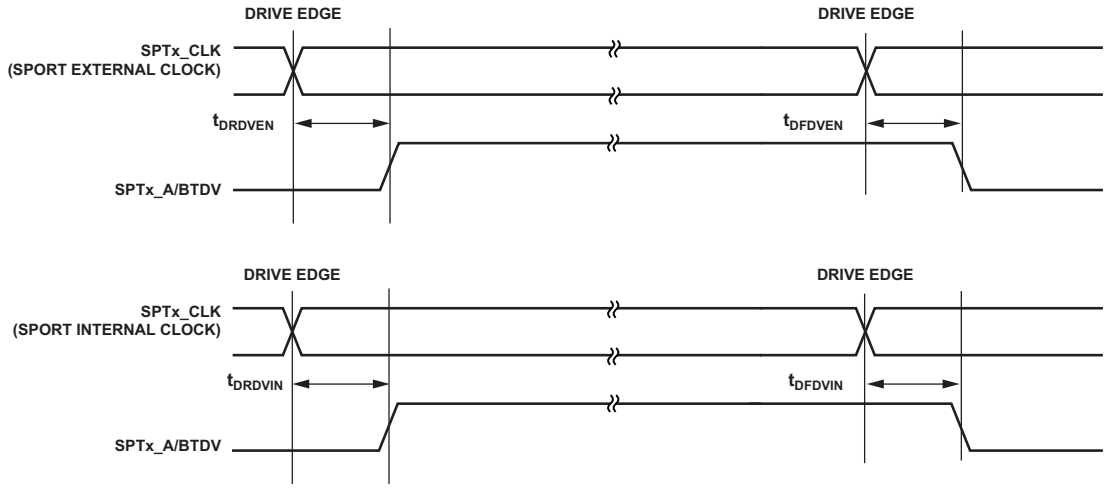


Figure 13. SPORTs—Transmit Data Valid Internal and External Clock

Table 40. SPORTs—External Late Frame Sync[1]

Parameter		Min	Max	Unit
Switching Characteristics				
$t_{DDTLFSE}$	Data Delay From Late External Transmit Frame Sync or External Receive Frame Sync with SPORT_MCTL_A/B bits MCE = 1, MFD = 0[2]		14	ns
$t_{DDTENFS}$	Data Enable for SPORT_MCTL_A/B bits MCE = 1, MFD = 0[2]	0.5		ns

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

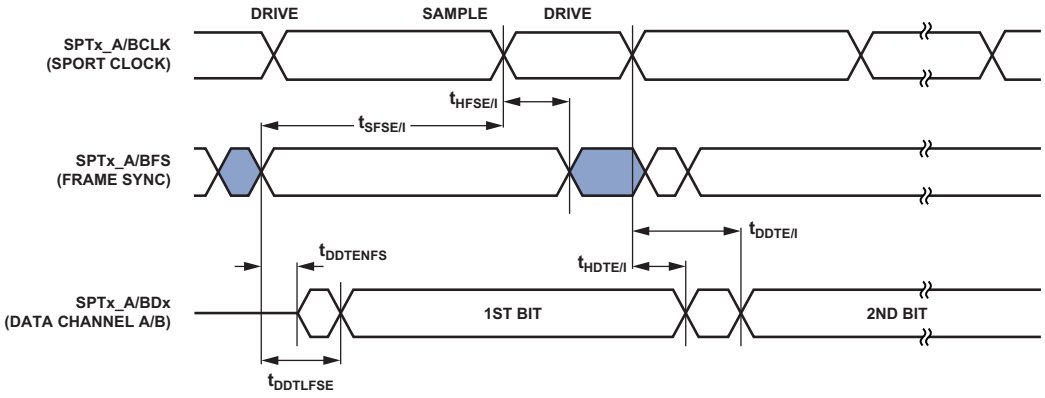


Figure 14. External Late Frame Sync

Asynchronous Sample Rate Converter (ASRC)—Serial Input Port

The ASRC input signals are routed from the DAI0_PINx pins using the SRU. Therefore, the timing specifications provided in Table 41 are valid at the DAI0_PINx pins.

Table 41. ASRC, Serial Input Port

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 originate from any of the DAI pins. The serial clock and frame sync signals can also originate via PCG or SPORTs. The input of the PCG can be either CLKIN or any of the DAI pins.

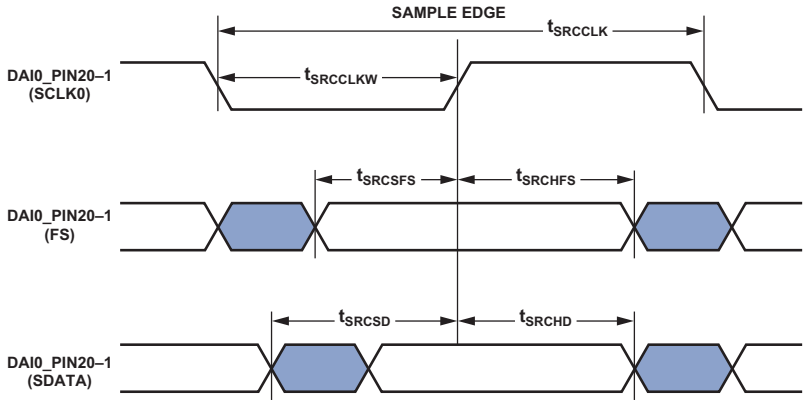


Figure 15. ASRC Serial Input Port Timing

Asynchronous Sample Rate Converter (ASRC)—Serial Output Port

For the serial output port, the frame sync is an input and it must meet setup and hold times with regard to SCLK0 on the output port. The serial data output has a hold time and delay specification with regard to the serial clock. In TDM mode, the ASRC drives at the rising edge and samples at the falling edge of the serial clock. In all other modes, the serial clock rising edge is the sampling edge, and the falling edge is the driving edge.

Table 42. ASRC, Serial Output Port

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		13.3	ns
t_{SRCTDH}[1]	Transmit Data Hold After Serial Clock Falling Edge	1		ns

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

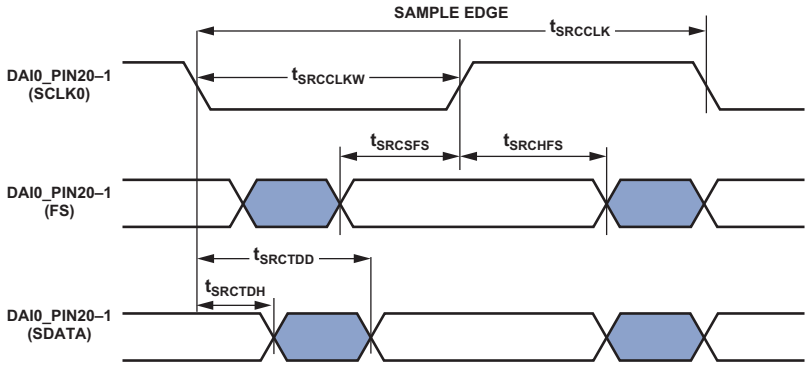


Figure 16. ASRC Serial Output Port Timing

SPI Port—Master Timing

SPI0, SPI1, and SPI2

Table 43 and Figure 17 describe the SPI port master operations.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in megahertz is set by the following equation:

$$f_{SPICLKPROG} = \frac{f_{CDU\_CLKO0}}{(BAUD + 1)}$$

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

where *BAUD* is a field in the SPIx_CLK register that can be set from 0 to 65,535.

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 SPI1 and SPI2.
- CPHA is a configuration bit in the SPI_CTL register.

Table 43. SPI Port—Master Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	3.5		ns
t_{HSPIDM}	SPIx_CLK Sampling Edge to Data Input Invalid	2		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{\text{SPIx\_SEL}}$ Low to First SPI_CLK Edge for CPHA = 1[2]	$t_{SPICLKPROG} - 5$		ns
	$\overline{\text{SPIx\_SEL}}$ Low to First SPI_CLK Edge for CPHA = 0[2]	$1.5 \times t_{SPICLKPROG} - 5$		ns
t_{SPICHM}	SPIx_CLK High Period[3]	$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLM}	SPIx_CLK Low Period[3]	$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLK}	SPIx_CLK Period[3]	$t_{SPICLKPROG} - 1.5$		ns
t_{HDSM}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 1[2]	$1.5 \times t_{SPICLKPROG} - 5$		ns
	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 0[2]	$t_{SPICLKPROG} - 5$		ns
t_{SPITDM}	Sequential Transfer Delay[2, 4]	$t_{SPICLKPROG} - 1.5$		ns
$t_{DDSPIDM}$	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		2.7	ns
$t_{HDSPIDM}$	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	–3.75		ns

[1] All specifications apply to SPI0, SPI1, and SPI2.
[2] Specification assumes the LEADX and LAGX bits in the SPI_DLY register are 1.
[3] See Table 26 for details on the minimum period that can be programmed for $t_{SPICLKPROG}$.
[4] Applies to sequential mode with STOP ≥ 1.

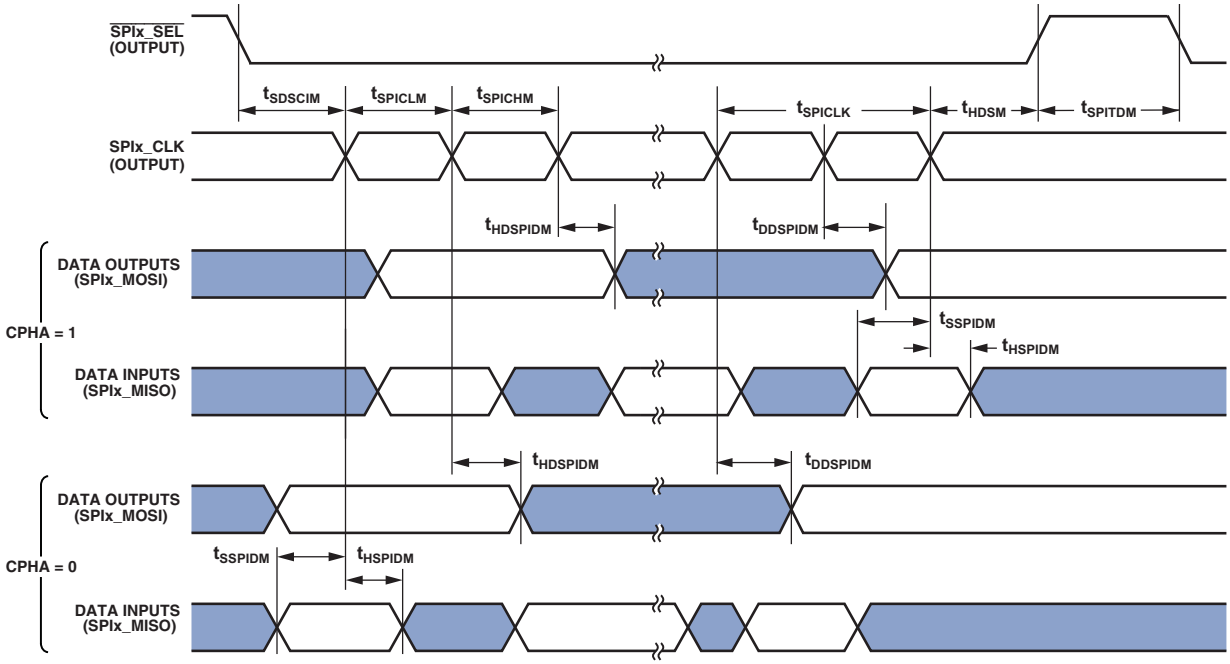


Figure 17. SPI Port—Master Timing

SPI Port—Slave Timing

SPI0, SPI1, and SPI2

Table 44 and Figure 18 describe 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 SPI1 and SPI2.
- CPHA is a configuration bit in the SPI_CTL register.

Table 44. SPI Port—Slave Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLK}	SPIx_CLK Period[2]	$t_{SPICLKEXT} - 1.5$		ns
t_{HDS}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SS}}$ Not Asserted	5		ns
t_{SPITDS}	Sequential Transfer Delay	$t_{SPICLKEXT} - 1.5$		ns
t_{SDSCI}	$\overline{\text{SPIx\_SS}}$ Assertion to First SPIx_CLK Edge	11.7		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	14.12	ns
t_{DSDHI}	$\overline{\text{SPIx\_SS}}$ Deassertion to Data High Impedance	0	12.6	ns
t_{DDSPID}	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		14.16	ns
t_{HDSPID}	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	1.5		ns

[1] All specifications apply to SPI0, SPI1, and SPI2.
[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 26.

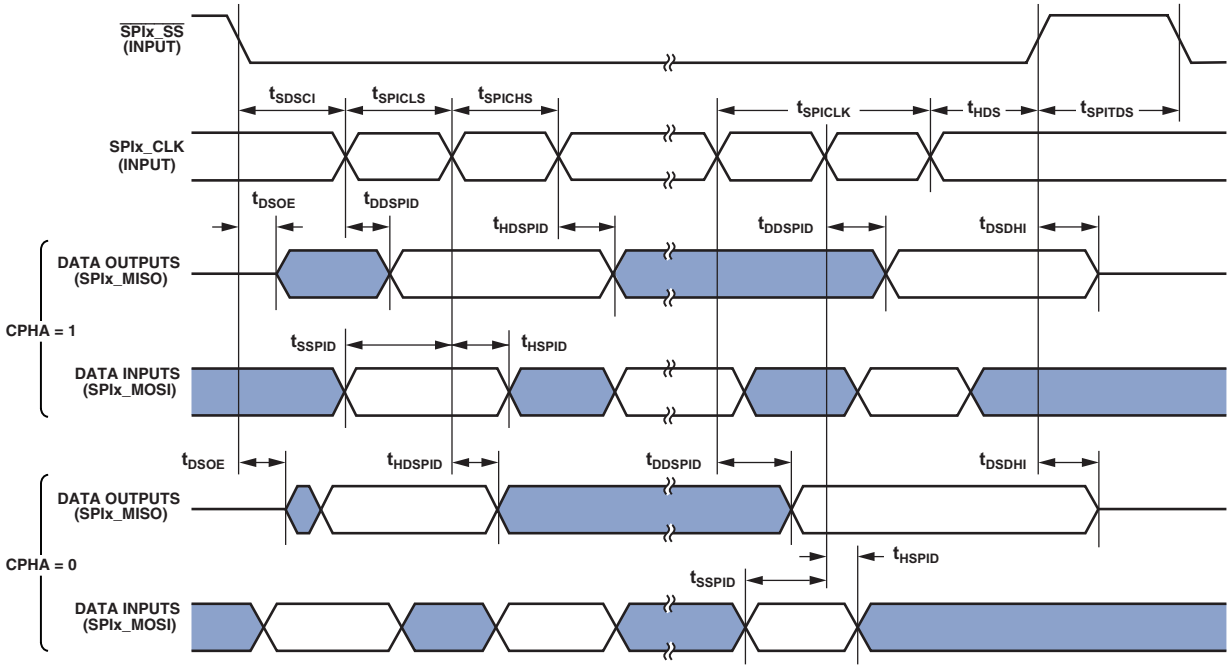


Figure 18. SPI Port—Slave Timing

SPI Port—SPIx_RDY Slave Timing

SPIx_RDY provides flow control. CPOL, CPHA, and FCCH are configuration bits in the SPIx_CTL register.

Table 45. SPI Port—SPIx_RDY Slave Timing[1]

Parameter	Conditions	Min	Max	Unit
Switching Characteristic				
$t_{DSPISCKRDYS}$ SPIx_RDY Deassertion From Last Valid Input SPIx_CLK Edge	FCCH = 0	$3 \times t_{CDU\_CLKO0}$	$4 \times t_{CDU\_CLKO0} + 10$	ns
	FCCH = 1	$4 \times t_{CDU\_CLKO0}$	$5 \times t_{CDU\_CLKO0} + 10$	ns

[1] All specifications apply to all three SPIs.

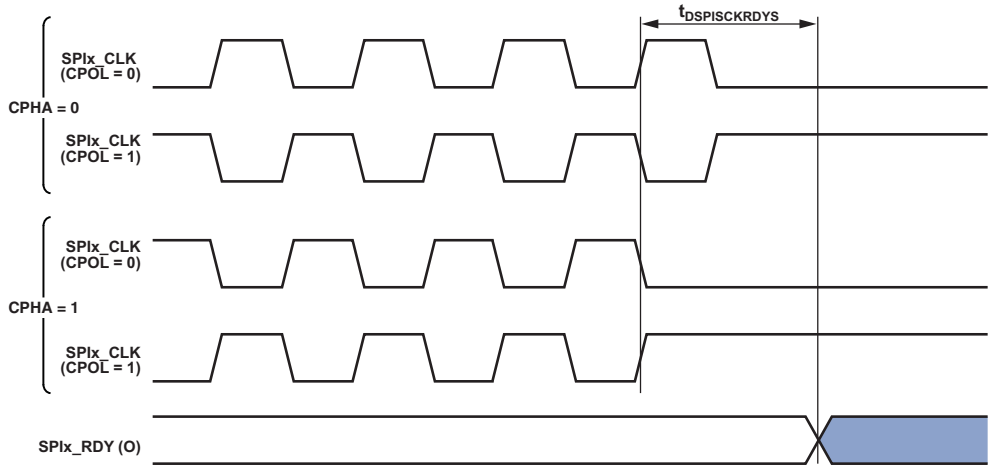


Figure 19. SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode

SPI Port—Open Drain Mode (ODM) Timing

In Figure 20 and Figure 21, the outputs can be SPIx_MOSI, SPIx_MISO, SPIx_D2, and/or SPIx_D3, depending on the mode of operation. CPOL and CPHA are configuration bits in the SPI_CTL register.

Table 46. SPI Port—ODM Master Mode Timing[1]

Parameter		Min	Max	Unit
Switching Characteristics				
$t_{HDSPIODMM}$	SPIx_CLK Edge to High Impedance From Data Out Valid	–1.5		ns
$t_{DDSPIODMM}$	SPIx_CLK Edge to Data Out Valid From High Impedance		6	ns

[1] All specifications apply to all three SPIs.

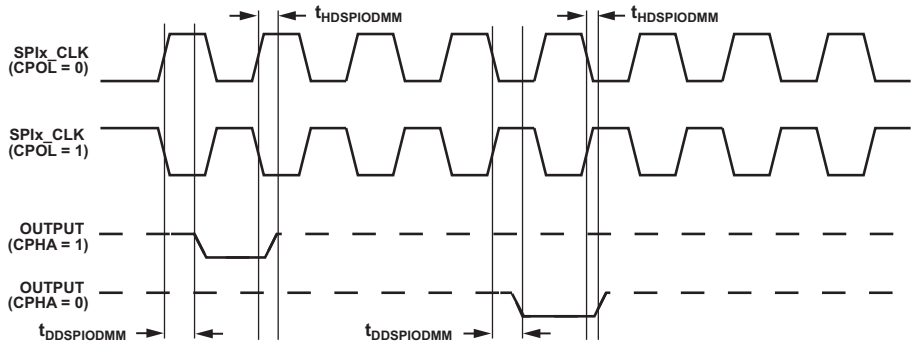


Figure 20. ODM Master Mode

Table 47. SPI Port—ODM Slave Mode[1]

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.

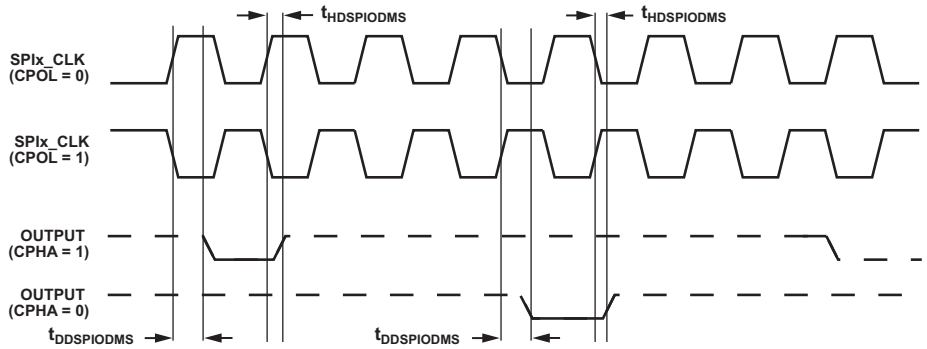


Figure 21. ODM Slave Mode

SPI Port—SPIx_RDY Master Timing

SPIx_RDY is used to provide flow control. CPOL and CPHA are configuration bits in the SPIx_CTL register, whereas LEADX, LAGX, and STOP are configuration bits in the SPIx_DLY register.

Table 48. SPI Port—SPIx_RDY Master Timing[1]

Parameter		Conditions	Min	Max	Unit
Timing Requirement					
$t_{SRDYSCKM}$	Setup Time for SPIx_RDY Deassertion Before Last Valid Data SPIx_CLK Edge		$(2 + 2 \times BAUD^{2}) \times t_{CDU\_CLKO0} + 11$		ns
Switching Characteristic					
$t_{DRDYSCKM}$[3]	Assertion of SPIx_RDY to First SPIx_CLK Edge of Next Transfer	BAUD = 0, CPHA = 0	$4.5 \times t_{CDU\_CLKO0}$	$5.5 \times t_{CDU\_CLKO0} + 12$	ns
		BAUD = 0, CPHA = 1	$4 \times t_{CDU\_CLKO0}$	$5 \times t_{CDU\_CLKO0} + 12$	ns
		BAUD > 0, CPHA = 0	$(1 + 1.5 \times BAUD^{2}) \times t_{CDU\_CLKO0}$	$(2 + 2.5 \times BAUD^{2}) \times t_{CDU\_CLKO0} + 12$	ns
		BAUD > 0, CPHA = 1	$(1 + 1 \times BAUD^{2}) \times t_{CDU\_CLKO0}$	$(2 + 2 \times BAUD^{2}) \times t_{CDU\_CLKO0} + 12$	ns

[1] All specifications apply to all three SPIs.
[2] BAUD value is set using the SPIx_CLK.BAUD bits. BAUD value = SPIx_CLK.BAUD bits + 1.
[3] Specification assumes the LEADX, LAGX, and STOP bits in the SPI_DLY register are zero.

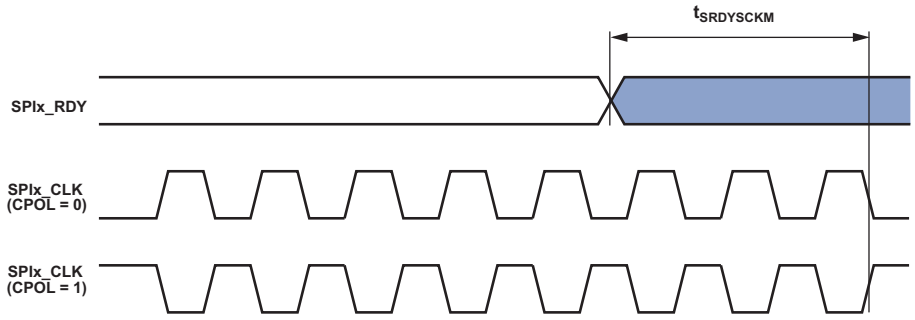


Figure 22. SPIx_RDY Setup Before SPIx_CLK

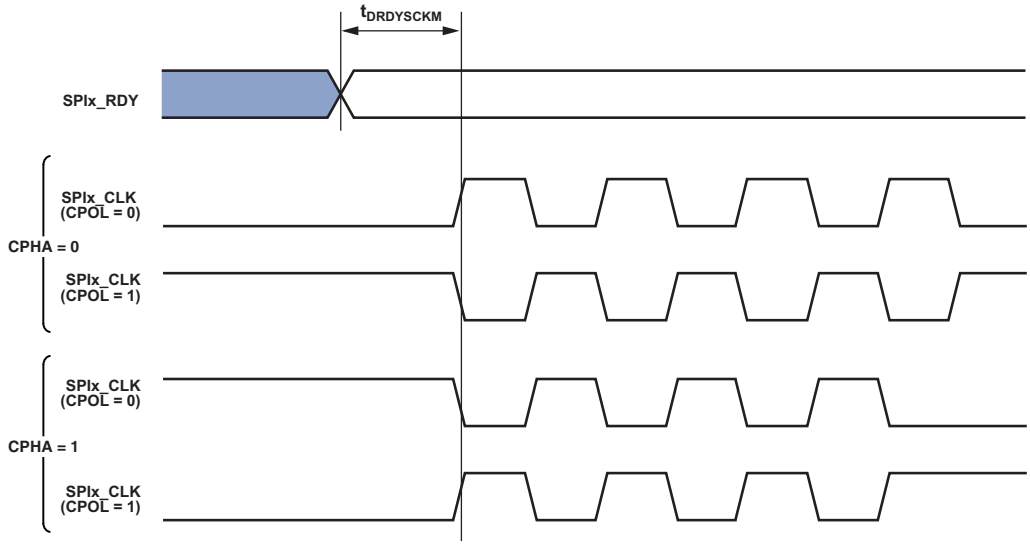


Figure 23. SPIx_CLK Switching Diagram after SPIx_RDY Assertion

OSPI Port—Master Timing

OSPI0

Table 49 and Figure 24 describe the OSPI port master operations. Slave mode is not supported for OSPI.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in megahertz is set by the following equation:

$$f_{SPICLKPROG} = \frac{f_{SYSCLK}}{PRG\_MBD}$$

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

where *PRG_MBD* is the master mode baud rate divisor.

Note that

- In dual-mode data transmit, the OSPI0_MISO signal is also an output.
- In quad-mode data transmit, the OSPI0_MISO, OSPI0_D2, and OSPI0_D3 signals are also outputs.
- In octal-mode data transmit, the OSPI0_MISO, OSPI0_D2, OSPI0_D3, OSPI0_D4, OSPI0_D5, OSPI0_D6, and OSPI0_D7 signals are also outputs.
- In dual-mode data receive, the OSPI0_MOSI signal is also an input.
- In quad-mode data receive, the OSPI0_MOSI, OSPI0_D2, and OSPI0_D3 signals are also inputs.
- In octal-mode data receive, the OSPI0_MISO, OSPI0_D2, OSPI0_D3, OSPI0_D4, OSPI0_D5, OSPI0_D6, and OSPI0_D7 signals are also outputs.
- CPHA is a configuration bit in the OSPI0_CTL register.

Table 49. OSPI0 Port—Master Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SSPIDM}	Data Input Valid to OSPI0_CLK Sampling Edge (Data Input Setup) [2]	t_{SYSCLK} + 2.6		ns
t_{HSPIDM}	OSPI0_CLK Sampling Edge to Data Input Invalid (Data Input Hold)[2]	1		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{OSPI0\_SEL}$ Low to First OSPI0_CLK Edge[3]	0.5 × $t_{SPICLKPROG}$ + PRG_CSSOT × t_{SYSCLK} – 1.5		ns
t_{SPICHM}	OSPI0_CLK High Period[4]	0.5 × $t_{SPICLKPROG}$ – 1.5		ns
t_{SPICLM}	OSPI0_CLK Low Period[4]	0.5 × $t_{SPICLKPROG}$ – 1.5		ns
t_{SPICLK}	OSPI0_CLK Period[4]	$t_{SPICLKPROG}$ – 1.5		ns
t_{HDSM}	Last OSPI0_CLK Edge to $\overline{OSPI0\_SEL}$ High for Mode = 0[5]	PRG_CSEOT × t_{SYSCLK} – 1		ns
	Last OSPI0_CLK Edge to $\overline{OSPI0\_SEL}$ High for Mode = 3[5, 6]	PRG_CSEOT × t_{SYSCLK} + 0.5 × $t_{SPICLKPROG}$ – 1		ns
$t_{DDSPIDM}$	OSPI0_CLK Edge to Data Out Valid to Driving Edge (Data Out Delay)[7]		(PRG_WRHLD + 1) × t_{SYSCLK} + 2.5	ns
$t_{HDSPIDM}$	OSPI0_CLK Edge to Data Out Invalid to Driving Edge (Data Out Hold)[7]	PRG_WRHLD × t_{SYSCLK} – 1		ns

[1] All specifications apply to OSPI0 only.
[2] t_{SSPIDM} and t_{HSPIDM} specifications are valid only for default OSPI0_RDC settings.
[3] PRG_CSSOT = chip select start of transfer (defined in OSPI0_DLY[7:0]).
[4] See Table 26 for details on the minimum period that can be programmed for $t_{SPICLKPROG}$.
[5] PRG_CSEOT = chip select end of transfer (defined in OSPI0_DLY[15:8]).
[6] Mode = clock phase and clock polarity bits (defined inOSPI0_CTL[2:1]).
[7] PRG_WRHLD = transmit delay to improve output hold (defined in OSPI0_RDC[19:16]).

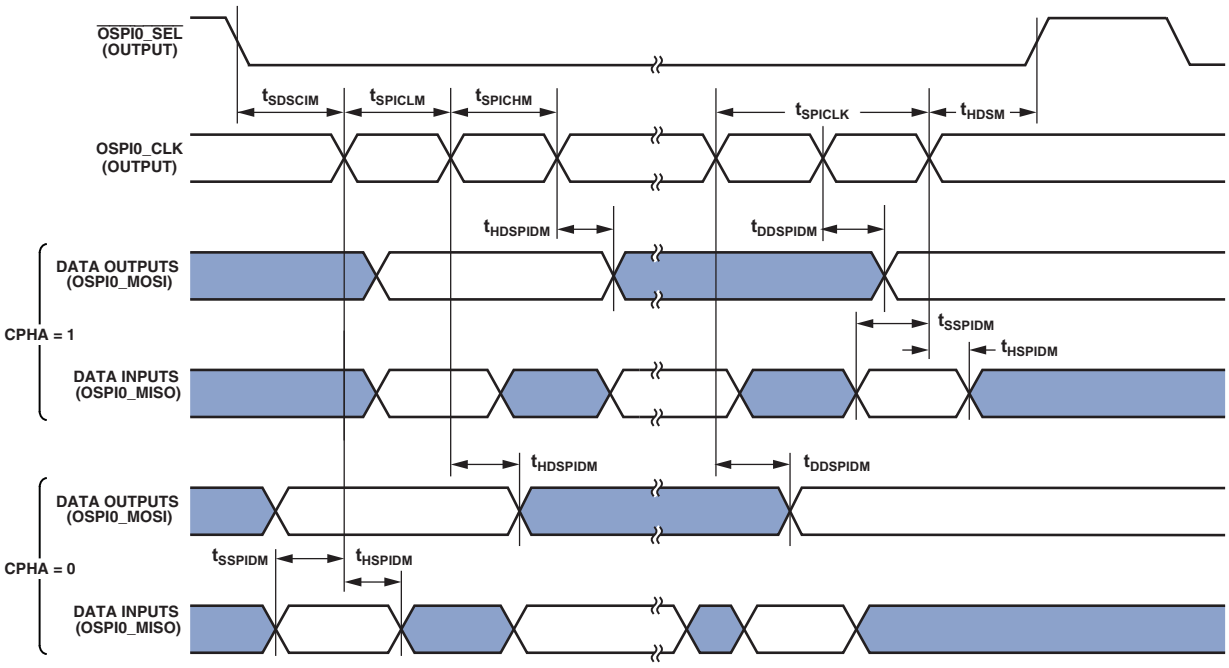


Figure 24. OSPI Port—Master Timing

Precision Clock Generator (PCG) (Direct Pin Routing)

This timing is only valid when the SRU is configured such that the precision clock generator (PCG) takes inputs directly from the DAI pins (via pin buffers) and sends outputs directly to the DAI pins. For the other cases, where the PCG 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_PINx).

Table 50. PCG (Direct Pin Routing)

Parameter		Min	Max	Unit
Timing Requirements				
t_{PCGIP}	Input Clock Period	$t_{SCLK0} \times 2$		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	11	ns
$t_{DTRIGCLK}$	PCG Output Clock Delay After PCG Trigger	$2 + (2.5 \times t_{PCGIP})$	$13.5 + (2.5 \times t_{PCGIP})$	ns
$t_{DTRIGFS}$[1]	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}[2]	Output Clock Period	$2 \times t_{PCGIP} - 1$		ns

[1] D = FSxDIV, PH = FSxPHASE. For more information, see the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference.
[2] Normal mode of operation.

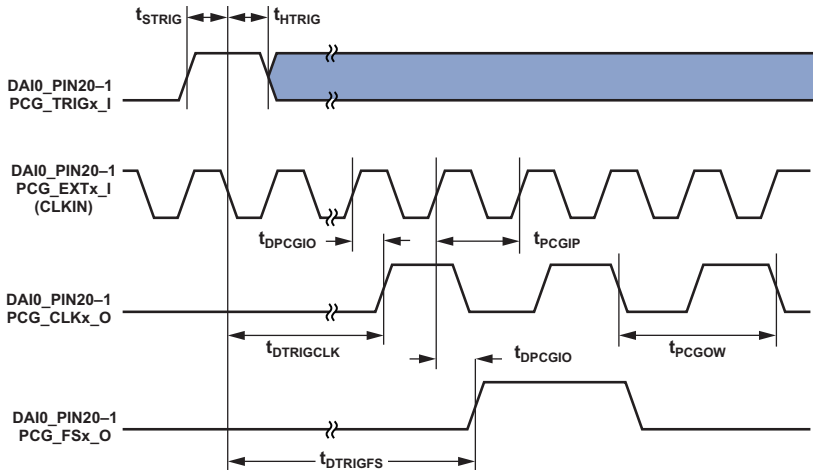


Figure 25. PCG (Direct Pin Routing)

General-Purpose IO Port Timing

Table 51 and Figure 26 describe I/O timing, related to the general-purpose ports (PORT).

Table 51. General-Purpose Port Timing

Parameter		Min	Max	Unit
Timing Requirement				
t_{WFI}	General-Purpose Port Pin Input Pulse Width	$2 \times t_{SCLK0} - 1.5$		ns

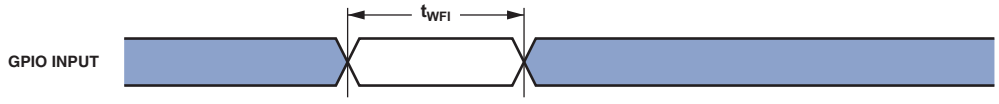


Figure 26. General-Purpose Port Timing

General-Purpose I/O Timer Cycle Timing

Table 52, Table 53, and Figure 27 describe timer expired operations related to the general-purpose timer (TIMER0). The width value is the timer period assigned in the TMx_TMRn_WIDTH register and can range from 1 to $2^{32} - 1$. When externally generated, the TMx_CLK clock is called $f_{TMRCLKEXT}$:

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

Table 52. Timer Cycle Timing—Internal Mode

Parameter		Min	Max	Unit
Timing Requirements				
t_{WL}	Timer Pulse Width Input Low (Measured In SCLK0 Cycles)[1]	$2 \times t_{SCLK0}$		ns
t_{WH}	Timer Pulse Width Input High (Measured In SCLK0 Cycles)[1]	$2 \times t_{SCLK0}$		ns
Switching Characteristic				
t_{HTO}	Timer Pulse Width Output (Measured In SCLK0 Cycles)[2]	$t_{SCLK0} \times$ WIDTH – 1.7	$t_{SCLK0} \times$ WIDTH + 1.5	ns

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

Table 53. Timer Cycle Timing—External Mode

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 timer 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 26.
[3] WIDTH refers to the value in the TMRx_WIDTH register (it can vary from 1 to $2^{32} - 1$).

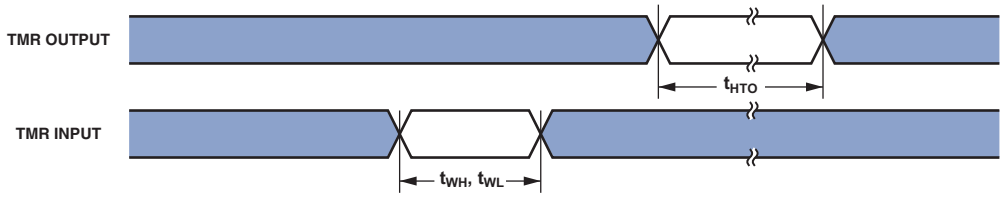


Figure 27. Timer Cycle Timing

DAIx Pin to DAIx Pin Direct Routing (DAI0 Block and DAI1 Block)

Table 54 and Figure 28 describe I/O timing related to the digital audio interface (DAI) for direct pin connections only (for example, DAIx_PB01_I to DAIx_PB02_O).

Table 54. DAI Pin to DAI Pin Routing

Parameter		Min	Max	Unit
Switching Characteristic				
t_{DPIO}	Delay DAI Pin Input Valid to DAI Output Valid	1.5	12	ns

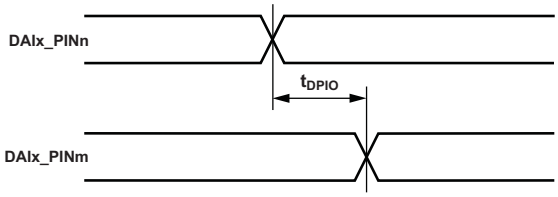


Figure 28. DAI Pin to DAI Pin Direct Routing

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

The UART ports receive and transmit operations are described in the ADSP-2159x/ADSP-SC591/592/594 SHARC+ Processor Hardware Reference.

Program Trace Macrocell (PTM) Timing

Table 55 and Figure 29 provide I/O timing related to the PTM.

Table 55. TRACE0 Timing

Parameter		Min	Max	Unit
Switching Characteristics				
t_{DTRD}	TRACE0 Data Delay From Trace Clock Maximum		$0.5 \times t_{SCLK0} + 3$	ns
t_{HTRD}	TRACE0 Data Hold From Trace Clock Minimum	$0.5 \times t_{SCLK0} - 2$		ns
t_{PTRCK}	TRACE0 Clock Period Minimum	$2 \times t_{SCLK0} - 1$		ns

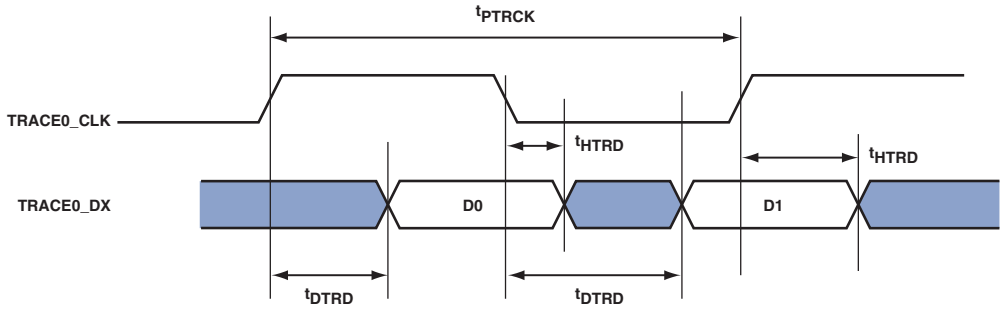


Figure 29. Trace Timing

Debug Interface (JTAG Emulation Port) Timing

Table 56 and Figure 30 provide I/O timing related to the debug interface (JTAG emulator port).

Table 56. JTAG Emulation Port Timing

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 = DAI0_PIN20-19, DAI0_PIN12-1, DAI1_PIN20-19, DAI1_PIN12-1, DMC0_A15-0, DMC0_DQ15-0, $\overline{\text{DMC0\_RESET}}$, PA_15-0, PB_15-0, PC_7-0, SYS_BMODE2-0, $\overline{\text{SYS\_FAULT}}$.

[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, $\overline{\text{DMC0\_WE}}$, PA_15-0, PB_15-0, PC_7-0, SYS_CLKOUT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_RESOUT}}$.

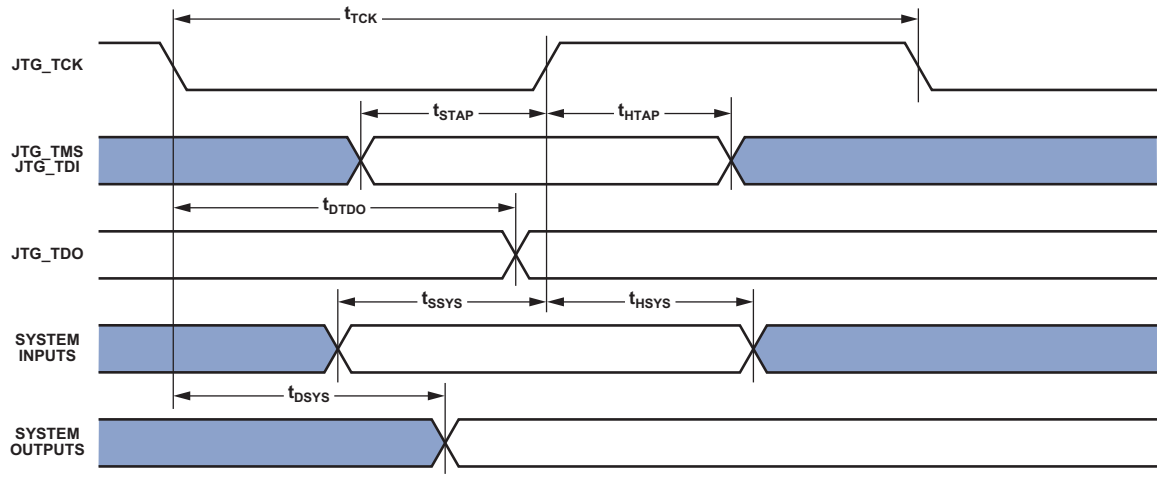


Figure 30. JTAG Port Timing

OUTPUT DRIVE CURRENTS

Figure 31 through Figure 36 show typical current voltage characteristics for the output drivers of the ADSP-2159x/ADSP-SC59x processors. The curves represent the current drive capability of the output drivers as a function of output voltage.

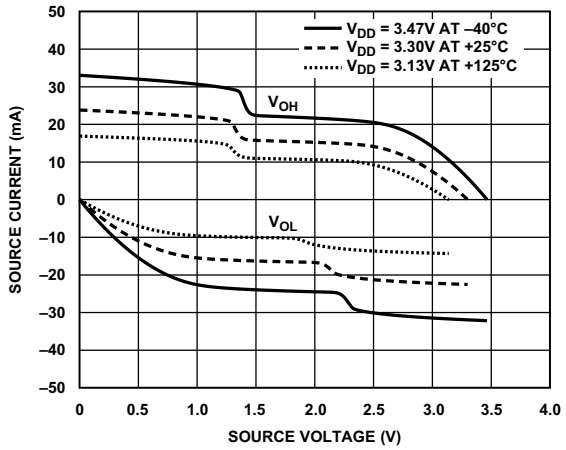


Figure 31. Driver Type A Current for All Pins Operating at Less Than or Equal to 62.5 MHz (3.3 V V_{DD_EXT})

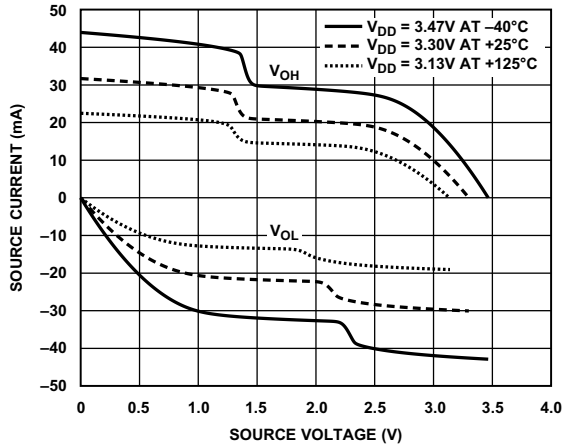


Figure 32. Driver Type A Current for All Pins Operating Above 62.5 MHz and Less Than or Equal to125 MHz (3.3 V V_{DD_EXT})

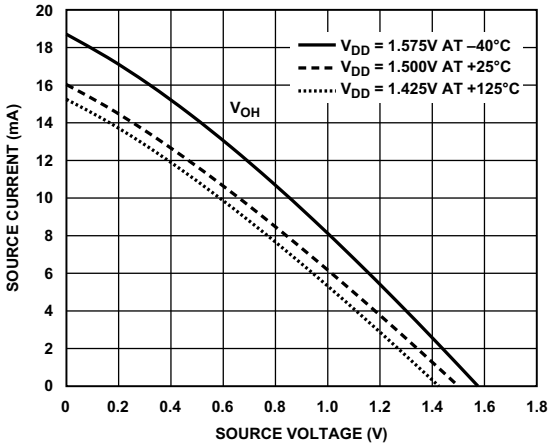


Figure 33. Driver Type B and Driver Type C (DDR3 Drive Strength 100 Ω)

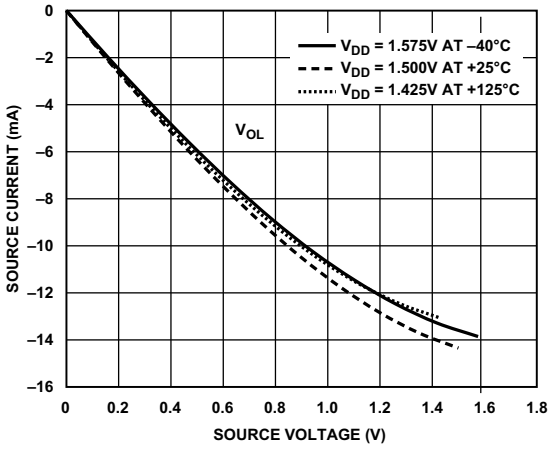


Figure 34. Driver Type B and Driver Type C (DDR3 Drive Strength 100 Ω)

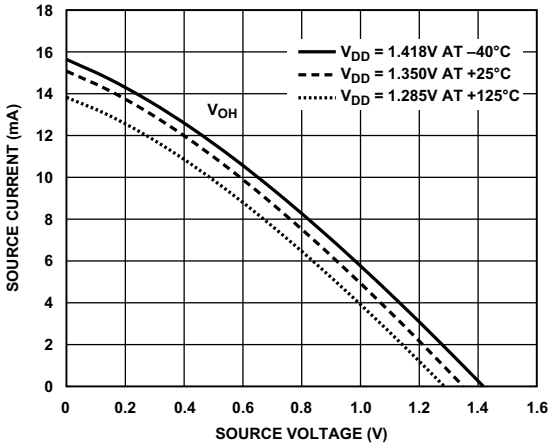


Figure 35. Driver Type B and Driver Type C (DDR3L Drive Strength100 Ω)

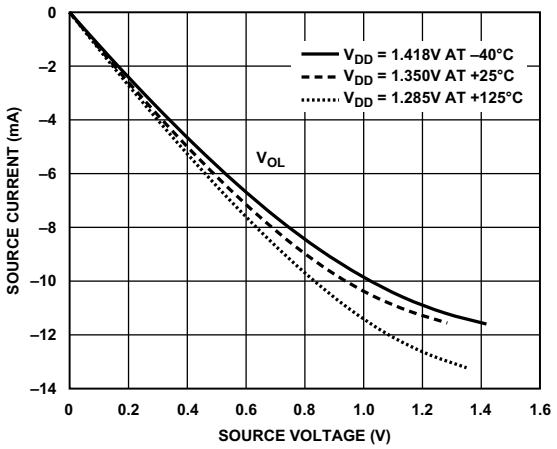


Figure 36. Driver Type B and Driver Type C (DDR3L Drive Strength 100 Ω)

TEST CONDITIONS

All timing parameters appearing in this data sheet were measured under the conditions described in this section. Figure 37 shows the measurement point for ac measurements (except output enable/disable). The measurement point, V_{MEAS}, is $V_{DD\_EXT}/2$ for V_{DD_EXT} (nominal) = 3.3 V.

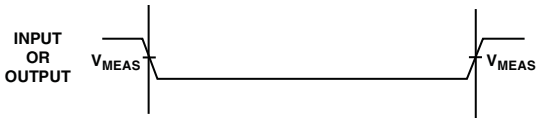


Figure 37. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time Measurement

Output pins are considered enabled when they make a transition from a high impedance state to the point when they start driving.

The output enable time, t_{ENA}, is the interval from the point when a reference signal reaches a high or low voltage level to the point when the output starts driving, as shown on the right side of Figure 38. If multiple pins are enabled, the measurement value is that of the first pin to start driving.

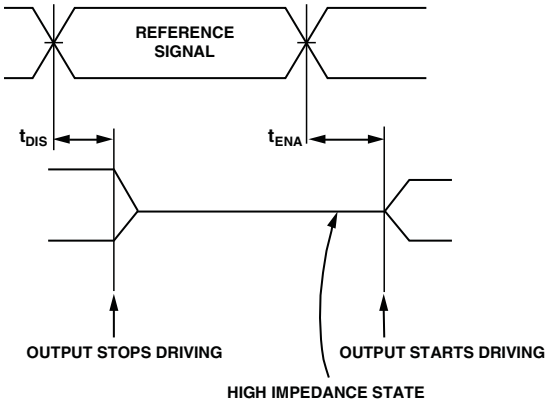


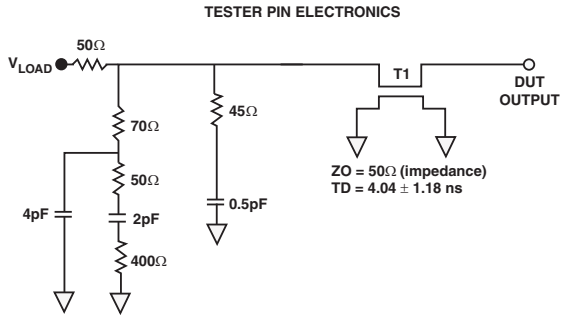
Figure 38. Output Enable/Disable

Output Disable Time Measurement

Output pins are considered disabled when they stop driving, enter a high impedance state, and start to decay from the output high or low voltage. The output disable time, t_{DIS}, is the interval from when a reference signal reaches a high or low voltage level to the point when the output stops driving, as shown on the left side of Figure 38.

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all pins (see Figure 39). V_{LOAD} is equal to $V_{DD\_EXT}/2$. Figure 40 through Figure 43 show how output rise time varies with capacitance. The delay and hold specifications given must be derated by a factor derived from these figures. The graphs in Figure 40 through Figure 43 cannot be linear outside the ranges shown.



NOTES:
THE WORST-CASE TRANSMISSION LINE DELAY IS SHOWN AND CAN BE USED FOR THE OUTPUT TIMING ANALYSIS TO REFLECT THE TRANSMISSION LINE EFFECT AND MUST BE CONSIDERED. THE TRANSMISSION LINE (TD) IS FOR LOAD ONLY AND DOES NOT AFFECT THE DATA SHEET TIMING SPECIFICATIONS.

ANALOG DEVICES RECOMMENDS USING THE IBIS MODEL TIMING FOR A GIVEN SYSTEM REQUIREMENT. IF NECESSARY, THE SYSTEM CAN INCORPORATE EXTERNAL DRIVERS TO COMPENSATE FOR ANY TIMING DIFFERENCES.

Figure 39. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

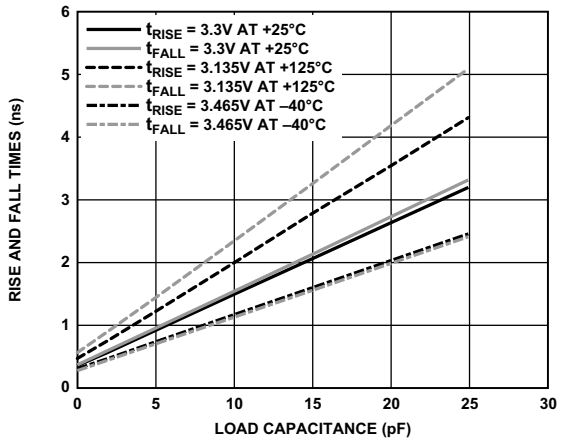


Figure 40. Driver Type A Rise and Fall Times (10% to 90%) vs. Load Capacitance for All Pins Operating Above 62.5 MHz and Less Than or Equal to 125 MHz

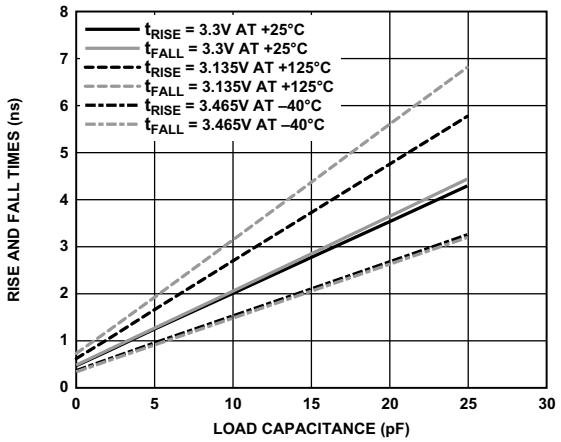


Figure 41. Driver Type A Rise and Fall Times (10% to 90%) vs. Load Capacitance for All Pins Operating at Less Than or Equal to 62.5 MHz

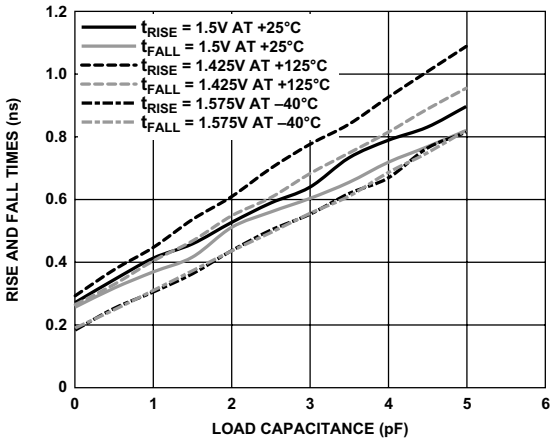


Figure 42. Driver Type B and Driver Type C Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for DDR3 at 100 Ω

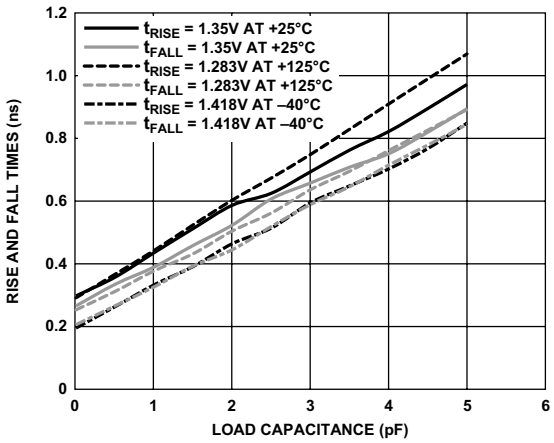


Figure 43. Driver Type B and Driver Type C Rise and Fall Times (10% to 90%) vs. Load Capacitance for DDR3L at 100 Ω

ENVIRONMENTAL CONDITIONS

The ADSP-2159x/ADSP-SC59x processors are rated for performance over the temperature range specified in the Operating Conditions section.

Application system thermal simulation is required for accurate temperature analysis. The thermal simulation must account for all specific 3D system design features, including, but not limited to other heat sources, use of heat sinks, use of thermal interface materials, and the system enclosure details. Thermal models of the package are available from Analog Devices under the Tools and Simulations tab of the product web page. The thermal model(s) are compatible with all major thermal simulation tools.

The use of JEDEC θ_{JA}, θ_{JC}, or Ψ_{JT} thermal parameters for application system thermal estimates is not recommended as indicated in the JEDEC51 specifications:

> "This methodology is not meant to and will not predict the performance of a package in an application-specific environment."

ADSP-2159x 400-BALL LOW PERIPHERAL COUNT (LPC) BGA BALL ASSIGNMENTS

The ADSP-2159x 400-Ball LPC BGA Ball Assignments (Numerical by Ball Number) table lists the package by ball number.

The ADSP-2159x 400-Ball LPC BGA Ball Assignments (Alphabetical by Pin Name) table lists the package by pin name.

ADSP-2159x 400-BALL LPC 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	DMC0_DQ08	E03	GND	G04	VDD_INT
A02	DMC0_DQ11	C03	GND	E04	VDD_INT	G05	VDD_INT
A03	DMC0_DQ10	C04	GND	E05	VDD_INT	G06	VDD_DMC
A04	$\overline{\text{DMC0\_UDQS}}$	C05	GND	E06	VDD_INT	G07	VDD_DMC
A05	DMC0_DQ15	C06	GND	E07	VDD_INT	G08	VDD_DMC
A06	DMC0_UDM	C07	PB_03	E08	VDD_INT	G09	VDD_DMC
A07	PB_00	C08	PB_05	E09	VDD_INT	G10	VDD_REF
A08	SYS_BMODE2	C09	PB_01	E10	VDD_INT	G11	VDD_REF
A09	$\overline{\text{SYS\_HWRST}}$	C10	$\overline{\text{SYS\_RESOUT}}$	E11	VDD_INT	G12	VDD_DMC
A10	$\overline{\text{JTG\_TRST}}$	C11	JTG_TDO	E12	VDD_INT	G13	VDD_DMC
A11	$\overline{\text{SYS\_FAULT}}$	C12	JTG_TMS	E13	VDD_INT	G14	VDD_DMC
A12	$\overline{\text{DMC0\_WE}}$	C13	JTG_TDI	E14	VDD_INT	G15	VDD_DMC
A13	DMC0_A14	C14	GND	E15	VDD_INT	G16	VDD_INT
A14	DMC0_A13	C15	GND	E16	VDD_INT	G17	VDD_INT
A15	DMC0_A10	C16	GND	E17	VDD_INT	G18	GND
A16	DMC0_A08	C17	GND	E18	GND	G19	DMC0_RZQ
A17	DMC0_BA1	C18	GND	E19	DMC0_BA2	G20	DMC0_VREF1
A18	DMC0_A07	C19	DMC0_A03	E20	$\overline{\text{DMC0\_CAS}}$	H01	DMC0_DQ03
A19	DMC0_A04	C20	DMC0_A02	F01	$\overline{\text{DMC0\_LDQS}}$	H02	GND
A20	GND	D01	DMC0_DQ07	F02	DMC0_LDQS	H03	GND
B01	GND	D02	GND	F03	GND	H04	VDD_INT
B02	GND	D03	DMC0_DQ09	F04	VDD_INT	H05	VDD_INT
B03	DMC0_DQ12	D04	GND	F05	VDD_INT	H06	VDD_DMC
B04	DMC0_UDQS	D05	GND	F06	VDD_DMC	H07	VDD_DMC
B05	DMC0_DQ14	D06	GND	F07	VDD_DMC	H08	GND
B06	DMC0_DQ13	D07	GND	F08	VDD_DMC	H09	GND
B07	PB_04	D08	VDD_EXT	F09	VDD_DMC	H10	GND
B08	PB_02	D09	VDD_EXT	F10	VDD_DMC	H11	GND
B09	SYS_BMODE1	D10	VDD_EXT	F11	VDD_DMC	H12	GND
B10	SYS_BMODE0	D11	VDD_EXT	F12	VDD_DMC	H13	GND
B11	JTG_TCK	D12	GND	F13	VDD_DMC	H14	VDD_DMC
B12	$\overline{\text{DMC0\_RESET}}$	D13	GND	F14	VDD_DMC	H15	VDD_DMC
B13	DMC0_A15	D14	GND	F15	VDD_DMC	H16	VDD_INT
B14	DMC0_A12	D15	GND	F16	VDD_INT	H17	VDD_INT
B15	DMC0_A11	D16	GND	F17	VDD_INT	H18	GND
B16	DMC0_A09	D17	GND	F18	GND	H19	DMC0_ODT
B17	DMC0_BA0	D18	GND	F19	$\overline{\text{DMC0\_RAS}}$	H20	DMC0_CKE
B18	DMC0_A06	D19	DMC0_A00	F20	$\overline{\text{DMC0\_CS0}}$	J01	DMC0_DQ01
B19	GND	D20	DMC0_A01	G01	DMC0_DQ04	J02	DMC0_DQ00
B20	DMC0_A05	E01	DMC0_DQ05	G02	DMC0_DQ02	J03	GND
C01	DMC0_LDM	E02	DMC0_DQ06	G03	GND	J04	DMC0_VREF0

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
J05	VDD_INT	L13	GND	P01	GND	T09	VDD_REF
J06	VDD_DMC	L14	VDD_INT	P02	PA_02	T10	VDD_REF
J07	VDD_DMC	L15	VDD_INT	P03	PA_01	T11	VDD_REF
J08	GND	L16	GND	P04	VDD_EXT	T12	VDD_REF
J09	GND	L17	GND	P05	VDD_REF	T13	VDD_REF
J10	GND	L18	HADC0_VREFN	P06	VDD_INT	T14	VDD_REF
J11	GND	L19	VDD_ANA	P07	VDD_INT	T15	VDD_REF
J12	GND	L20	HADC0_VREFP	P08	VDD_INT	T16	VDD_EXT
J13	GND	M01	GND	P09	VDD_INT	T17	GND
J14	VDD_INT	M02	GND	P10	VDD_INT	T18	DAI1_PIN09
J15	VDD_INT	M03	SYS_CLKOUT	P11	VDD_INT	T19	DAI1_PIN06
J16	VDD_INT	M04	VDD_EXT	P12	VDD_INT	T20	DAI1_PIN04
J17	VDD_INT	M05	VDD_INT	P13	VDD_INT	U01	GND
J18	GND	M06	VDD_INT	P14	VDD_INT	U02	PA_10
J19	$\overline{\text{DMC0\_CK}}$	M07	VDD_PLL	P15	VDD_REF	U03	PA_08
J20	DMC0_CK	M08	GND	P16	VDD_EXT	U04	GND
K01	GND	M09	GND	P17	GND	U05	GND
K02	GND	M10	GND	P18	DAI1_PIN02	U06	VDD_EXT
K03	GND	M11	GND	P19	GND	U07	VDD_EXT
K04	VDD_EXT	M12	GND	P20	GND	U08	VDD_EXT
K05	VDD_INT	M13	GND	R01	PA_04	U09	VDD_EXT
K06	VDD_PLL	M14	VDD_INT	R02	GND	U10	VDD_EXT
K07	VDD_DMC	M15	VDD_REF	R03	PA_03	U11	VDD_EXT
K08	GND	M16	VDD_EXT	R04	VDD_EXT	U12	VDD_EXT
K09	GND	M17	GND	R05	VDD_REF	U13	VDD_EXT
K10	GND	M18	GND	R06	VDD_INT	U14	VDD_EXT
K11	GND	M19	HADC0_VIN2	R07	VDD_INT	U15	VDD_EXT
K12	GND	M20	HADC0_VIN3	R08	VDD_INT	U16	GND
K13	GND	N01	SYS_CLKIN0	R09	VDD_INT	U17	GND
K14	VDD_INT	N02	GND	R10	VDD_INT	U18	DAI1_PIN10
K15	VDD_INT	N03	PA_00	R11	VDD_INT	U19	DAI1_PIN07
K16	VDD_INT	N04	VDD_EXT	R12	VDD_INT	U20	DAI1_PIN05
K17	GND	N05	VDD_REF	R13	VDD_INT	V01	PA_07
K18	GND	N06	VDD_INT	R14	VDD_INT	V02	PA_09
K19	GND	N07	VDD_INT	R15	VDD_REF	V03	GND
K20	GND	N08	GND	R16	VDD_EXT	V04	GND
L01	SYS_XTAL0	N09	GND	R17	GND	V05	PB_06
L02	GND	N10	GND	R18	DAI1_PIN03	V06	PB_09
L03	GND	N11	GND	R19	DAI1_PIN08	V07	PB_12
L04	VDD_EXT	N12	GND	R20	DAI1_PIN01	V08	PA_11
L05	VDD_INT	N13	GND	T01	GND	V09	DAI0_PIN02
L06	VDD_PLL	N14	VDD_INT	T02	PA_05	V10	DAI0_PIN06
L07	VDD_PLL	N15	VDD_REF	T03	PA_06	V11	DAI0_PIN09
L08	GND	N16	VDD_EXT	T04	GND	V12	DAI0_PIN20
L09	GND	N17	GND	T05	VDD_EXT	V13	PC_00
L10	GND	N18	GND	T06	VDD_REF	V14	PC_05
L11	GND	N19	HADC0_VIN0	T07	VDD_REF	V15	GND
L12	GND	N20	HADC0_VIN1	T08	VDD_REF	V16	GND

Ball No.	Pin Name
V17	DAI1_PIN19
V18	GND
V19	DAI1_PIN11
V20	DAI1_PIN12
W01	GND
W02	GND
W03	PB_08
W04	PB_07
W05	PB_13
W06	PA_12
W07	PA_14
W08	PA_15
W09	DAI0_PIN03
W10	DAI0_PIN05
W11	DAI0_PIN08
W12	DAI0_PIN12
W13	DAI0_PIN19
W14	PB_15
W15	PC_01
W16	PC_03
W17	PC_06
W18	DAI1_PIN20
W19	GND
W20	GND
Y01	GND
Y02	GND
Y03	PB_10
Y04	PB_11
Y05	GND
Y06	PA_13
Y07	GND
Y08	DAI0_PIN01
Y09	DAI0_PIN04
Y10	DAI0_PIN07
Y11	DAI0_PIN10
Y12	DAI0_PIN11
Y13	GND
Y14	PB_14
Y15	PC_02
Y16	GND
Y17	PC_04
Y18	PC_07
Y19	GND
Y20	GND

ADSP-2159x 400-BALL LPC 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	Y08	DMC0_BA2	E19	GND	C16	GND	K13
DAI0_PIN02	V09	DMC0_CAS	E20	GND	C17	GND	K17
DAI0_PIN03	W09	DMC0_CK	J19	GND	C18	GND	K18
DAI0_PIN04	Y09	DMC0_CK	J20	GND	D02	GND	K19
DAI0_PIN05	W10	DMC0_CKE	H20	GND	D04	GND	K20
DAI0_PIN06	V10	DMC0_CS0	F20	GND	D05	GND	L02
DAI0_PIN07	Y10	DMC0_DQ00	J02	GND	D06	GND	L03
DAI0_PIN08	W11	DMC0_DQ01	J01	GND	D07	GND	L08
DAI0_PIN09	V11	DMC0_DQ02	G02	GND	D12	GND	L09
DAI0_PIN10	Y11	DMC0_DQ03	H01	GND	D13	GND	L10
DAI0_PIN11	Y12	DMC0_DQ04	G01	GND	D14	GND	L11
DAI0_PIN12	W12	DMC0_DQ05	E01	GND	D15	GND	L12
DAI0_PIN19	W13	DMC0_DQ06	E02	GND	D16	GND	L13
DAI0_PIN20	V12	DMC0_DQ07	D01	GND	D17	GND	L16
DAI1_PIN01	R20	DMC0_DQ08	C02	GND	D18	GND	L17
DAI1_PIN02	P18	DMC0_DQ09	D03	GND	E03	GND	M01
DAI1_PIN03	R18	DMC0_DQ10	A03	GND	E18	GND	M02
DAI1_PIN04	T20	DMC0_DQ11	A02	GND	F03	GND	M08
DAI1_PIN05	U20	DMC0_DQ12	B03	GND	F18	GND	M09
DAI1_PIN06	T19	DMC0_DQ13	B06	GND	G03	GND	M10
DAI1_PIN07	U19	DMC0_DQ14	B05	GND	G18	GND	M11
DAI1_PIN08	R19	DMC0_DQ15	A05	GND	H02	GND	M12
DAI1_PIN09	T18	DMC0_LDM	C01	GND	H03	GND	M13
DAI1_PIN10	U18	DMC0_LDQS	F01	GND	H08	GND	M17
DAI1_PIN11	V19	DMC0_LDQS	F02	GND	H09	GND	M18
DAI1_PIN12	V20	DMC0_ODT	H19	GND	H10	GND	N02
DAI1_PIN19	V17	DMC0_RAS	F19	GND	H11	GND	N08
DAI1_PIN20	W18	DMC0_RESET	B12	GND	H12	GND	N09
DMC0_A00	D19	DMC0_RZQ	G19	GND	H13	GND	N10
DMC0_A01	D20	DMC0_UDM	A06	GND	H18	GND	N11
DMC0_A02	C20	DMC0_UDQS	A04	GND	J03	GND	N12
DMC0_A03	C19	DMC0_UDQS	B04	GND	J08	GND	N13
DMC0_A04	A19	DMC0_VREF0	J04	GND	J09	GND	N17
DMC0_A05	B20	DMC0_VREF1	G20	GND	J10	GND	N18
DMC0_A06	B18	DMC0_WE	A12	GND	J11	GND	P01
DMC0_A07	A18	GND	A01	GND	J12	GND	P17
DMC0_A08	A16	GND	A20	GND	J13	GND	P19
DMC0_A09	B16	GND	B01	GND	J18	GND	P20
DMC0_A10	A15	GND	B02	GND	K01	GND	R02
DMC0_A11	B15	GND	B19	GND	K02	GND	R17
DMC0_A12	B14	GND	C03	GND	K03	GND	T01
DMC0_A13	A14	GND	C04	GND	K08	GND	T04
DMC0_A14	A13	GND	C05	GND	K09	GND	T17
DMC0_A15	B13	GND	C06	GND	K10	GND	U01
DMC0_BA0	B17	GND	C14	GND	K11	GND	U04
DMC0_BA1	A17	GND	C15	GND	K12	GND	U05

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
GND	U16	PB_02	B08	VDD_DMC	G14	VDD_INT	E17
GND	U17	PB_03	C07	VDD_DMC	G15	VDD_INT	F04
GND	V03	PB_04	B07	VDD_DMC	H06	VDD_INT	F05
GND	V04	PB_05	C08	VDD_DMC	H07	VDD_INT	F16
GND	V15	PB_06	V05	VDD_DMC	H14	VDD_INT	F17
GND	V16	PB_07	W04	VDD_DMC	H15	VDD_INT	G04
GND	V18	PB_08	W03	VDD_DMC	J06	VDD_INT	G05
GND	W01	PB_09	V06	VDD_DMC	J07	VDD_INT	G16
GND	W02	PB_10	Y03	VDD_DMC	K07	VDD_INT	G17
GND	W19	PB_11	Y04	VDD_EXT	D08	VDD_INT	H04
GND	W20	PB_12	V07	VDD_EXT	D09	VDD_INT	H05
GND	Y01	PB_13	W05	VDD_EXT	D10	VDD_INT	H16
GND	Y02	PB_14	Y14	VDD_EXT	D11	VDD_INT	H17
GND	Y05	PB_15	W14	VDD_EXT	K04	VDD_INT	J05
GND	Y07	PC_00	V13	VDD_EXT	L04	VDD_INT	J14
GND	Y13	PC_01	W15	VDD_EXT	M04	VDD_INT	J15
GND	Y16	PC_02	Y15	VDD_EXT	M16	VDD_INT	J16
GND	Y19	PC_03	W16	VDD_EXT	N04	VDD_INT	J17
GND	Y20	PC_04	Y17	VDD_EXT	N16	VDD_INT	K05
HADC0_VIN0	N19	PC_05	V14	VDD_EXT	P04	VDD_INT	K14
HADC0_VIN1	N20	PC_06	W17	VDD_EXT	P16	VDD_INT	K15
HADC0_VIN2	M19	PC_07	Y18	VDD_EXT	R04	VDD_INT	K16
HADC0_VIN3	M20	SYS_BMODE0	B10	VDD_EXT	R16	VDD_INT	L05
HADC0_VREFN	L18	SYS_BMODE1	B09	VDD_EXT	T05	VDD_INT	L14
HADC0_VREFP	L20	SYS_BMODE2	A08	VDD_EXT	T16	VDD_INT	L15
JTG_TCK	B11	SYS_CLKIN0	N01	VDD_EXT	U06	VDD_INT	M05
JTG_TDI	C13	SYS_CLKOUT	M03	VDD_EXT	U07	VDD_INT	M06
JTG_TDO	C11	$\overline{\text{SYS\_FAULT}}$	A11	VDD_EXT	U08	VDD_INT	M14
JTG_TMS	C12	$\overline{\text{SYS\_HWRST}}$	A09	VDD_EXT	U09	VDD_INT	N06
$\overline{\text{JTG\_TRST}}$	A10	$\overline{\text{SYS\_RESOUT}}$	C10	VDD_EXT	U10	VDD_INT	N07
PA_00	N03	SYS_XTAL0	L01	VDD_EXT	U11	VDD_INT	N14
PA_01	P03	VDD_ANA	L19	VDD_EXT	U12	VDD_INT	P06
PA_02	P02	VDD_DMC	F06	VDD_EXT	U13	VDD_INT	P07
PA_03	R03	VDD_DMC	F07	VDD_EXT	U14	VDD_INT	P08
PA_04	R01	VDD_DMC	F08	VDD_EXT	U15	VDD_INT	P09
PA_05	T02	VDD_DMC	F09	VDD_INT	E04	VDD_INT	P10
PA_06	T03	VDD_DMC	F10	VDD_INT	E05	VDD_INT	P11
PA_07	V01	VDD_DMC	F11	VDD_INT	E06	VDD_INT	P12
PA_08	U03	VDD_DMC	F12	VDD_INT	E07	VDD_INT	P13
PA_09	V02	VDD_DMC	F13	VDD_INT	E08	VDD_INT	P14
PA_10	U02	VDD_DMC	F14	VDD_INT	E09	VDD_INT	R06
PA_11	V08	VDD_DMC	F15	VDD_INT	E10	VDD_INT	R07
PA_12	W06	VDD_DMC	G06	VDD_INT	E11	VDD_INT	R08
PA_13	Y06	VDD_DMC	G07	VDD_INT	E12	VDD_INT	R09
PA_14	W07	VDD_DMC	G08	VDD_INT	E13	VDD_INT	R10
PA_15	W08	VDD_DMC	G09	VDD_INT	E14	VDD_INT	R11
PB_00	A07	VDD_DMC	G12	VDD_INT	E15	VDD_INT	R12
PB_01	C09	VDD_DMC	G13	VDD_INT	E16	VDD_INT	R13

Pin Name	Ball No.
VDD_INT	R14
VDD_PLL	K06
VDD_PLL	L06
VDD_PLL	L07
VDD_PLL	M07
VDD_REF	G10
VDD_REF	G11
VDD_REF	M15
VDD_REF	N05
VDD_REF	N15
VDD_REF	P05
VDD_REF	P15
VDD_REF	R05
VDD_REF	R15
VDD_REF	T06
VDD_REF	T07
VDD_REF	T08
VDD_REF	T09
VDD_REF	T10
VDD_REF	T11
VDD_REF	T12
VDD_REF	T13
VDD_REF	T14
VDD_REF	T15

CONFIGURATION OF THE ADSP-2159x 400-BALL LOW PERIPHERAL COUNT (LPC) BGA

Figure 44 shows an overview of signal placement on the ADSP-2159x 400-ball LPC BGA.

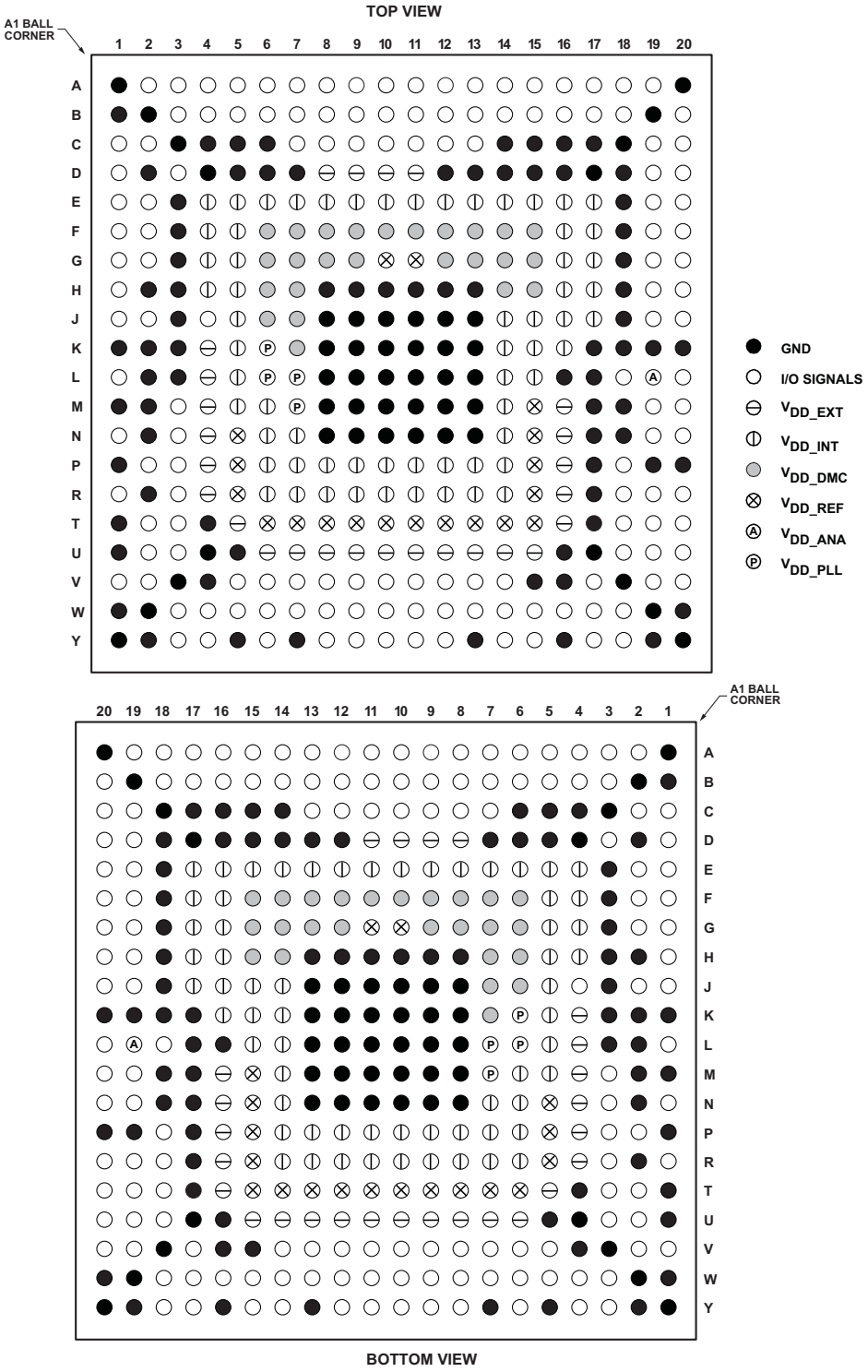


Figure 44. 400-Ball LPC BGA Configuration

OUTLINE DIMENSIONS

Dimensions for the 17 mm × 17 mm 400-ball BGA_ED package in Figure 45 are shown in millimeters.

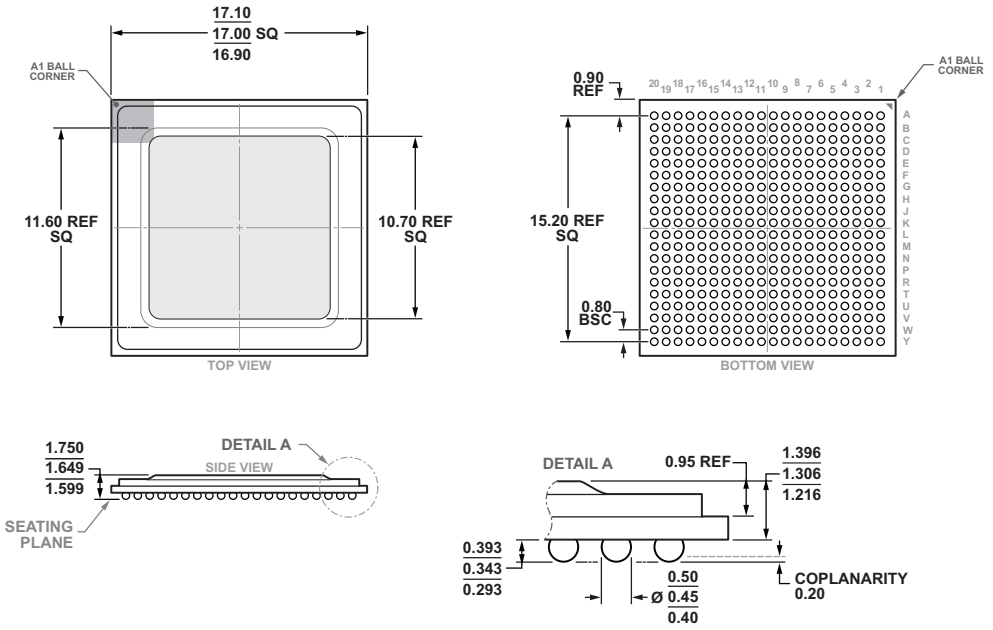


Figure 45. 400-Ball Ball Grid Array, Thermally Enhanced [BGA_ED]
(BP-400-3)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 57 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 57. BGA Data for Use with Surface-Mount Design

Package	Package Ball Attach Type	Package Solder Mask Opening	Package Ball Pad Size
BP-400-3	Solder Mask Defined	0.4 mm Diameter	0.5 mm Diameter

PLANNED AUTOMOTIVE PRODUCTION PRODUCTS

Model[1, 2]	Processor Instruction Rate (Max)	Arm Instruction Rate (Max)[3]	Temperature Range[4]	L2 SRAM	Arm Cores	SHARC+ Cores	Package Description	Package Option
ADSP-21591WCBPZ8	800 MHz	N/A	–40°C to +125°C	1 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21591WCBPZ8RL	800 MHz	N/A	–40°C to +125°C	1 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593WCBPZ8	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593WCBPZ8RL	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593WCBPZ10	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP21593WCBPZ10RL	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594WCBPZ8	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594WCBPZ8RL	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594WCBPZ10	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP21594WCBPZ10RL	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-SC592WCBPZ10	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	1	400-Ball BGA_ED	BP-400-3
ADSPSC592WCBPZ10RL	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	1	400-Ball BGA_ED	BP-400-3
ADSP-SC594WCBPZ8	800 MHz	800 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSP-SC594WCBPZ8RL	800 MHz	800 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSP-SC594WCBPZ10	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSPSC594WCBPZ10RL	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3

[1] Z =RoHS compliant part.
[2] RL = Supplied on Tape and Reel.
[3] N/A means not applicable.
[4] Referenced temperature is junction temperature. See Operating Conditions for junction temperature (T_J) specification.

PLANNED PRODUCTION PRODUCTS

Model[1]	Processor Instruction Rate (Max)	Arm Instruction Rate (Max)[2]	Temperature Range[3]	L2 SRAM	Arm Cores	SHARC+ Cores	Package Description	Package Option
ADSP-21591BBPZ8	800 MHz	N/A	–40°C to +125°C	1 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593BBPZ8	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593BBPZ10	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594BBPZ8	800 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594BBPZ10	1000 MHz	N/A	–40°C to +125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594KBPZ8	800 MHz	N/A	0°C to 125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21594KBPZ10	1000 MHz	N/A	0°C to 125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-SC592BBPZ10	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	1	400-Ball BGA_ED	BP-400-3
ADSP-SC592KBPZ10	1000 MHz	1000 MHz	0°C to 125°C	2 MB	1	1	400-Ball BGA_ED	BP-400-3
ADSP-SC594BBPZ8	800 MHz	800 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSP-SC594BBPZ10	1000 MHz	1000 MHz	–40°C to +125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSP-SC594KBPZ8	800 MHz	800 MHz	0°C to 125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3
ADSP-SC594KBPZ10	1000 MHz	1000 MHz	0°C to 125°C	2 MB	1	2	400-Ball BGA_ED	BP-400-3

[1] Z =RoHS compliant part.
[2] N/A means not applicable.
[3] Referenced temperature is junction temperature. See Operating Conditions for junction temperature (T_J) specification.

ORDERING GUIDE

Model[1]	Processor Instruction Rate (Max)	Arm Instruction Rate (Max)[2]	Temperature Range[3]	L2 SRAM	Arm Cores	SHARC+ Cores	Package Description	Package Option
ADSP-21591KBPZ8	800 MHz	N/A	0°C to 125°C	1 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593KBPZ8	800 MHz	N/A	0°C to 125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3
ADSP-21593KBPZ10	1000 MHz	N/A	0°C to 125°C	2 MB	0	2	400-Ball BGA_ED	BP-400-3

[1] Z =RoHS compliant part.
[2] N/A means not applicable.
[3] Referenced temperature is junction temperature. See Operating Conditions for junction temperature (T_J) specification.

I^2C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

D26463-8/21(0)

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View ADSP-21591KBPZ8 on WIN SOURCE
- Analog Devices Inc. Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management