SHARC+ Single Core High Performance DSP (Up to 1 GHz)

Preliminary Technical Data

ADSP-21562/21563/21565

SYSTEM FEATURES

Enhanced SHARC+ high performance floating-point core
- **Up to 1 GHz**
- **Up to 5 Mb (640 kB) Level 1 (L1) SRAM memory with parity (optional ability to configure as cache)**
- **32-bit, 40-bit, and 64-bit floating-point support**
- **32-bit fixed point**
- **Byte, short-word, word, long-word addressed**

Powerful DMA system

On-chip memory protection

Integrated safety features

17 mm × 17 mm 400 CSP_BGA (0.8 mm pitch), RoHS compliant

120-lead LQFP_EP (0.4 mm pitch), RoHS compliant

Low system power across automotive temperature range

MEMORY

Large on-chip Level 2 (L2) SRAM with ECC protection, up to 8 Mb (1 MB)

One Level 3 (L3) interface optimized for low system power, providing 16-bit interface to DDR3/ DDR3L SDRAM devices

ADDITIONAL FEATURES

Security and Protection
- **Crypto hardware accelerators**
- **Fast secure boot with IP protection**

Enhanced FIR and IIR accelerators running up to 1 GHz

APPLICATIONS

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

Consumer: speakers, sound bars, AVRs, conferencing systems, mixing consoles, microphone arrays, headphones

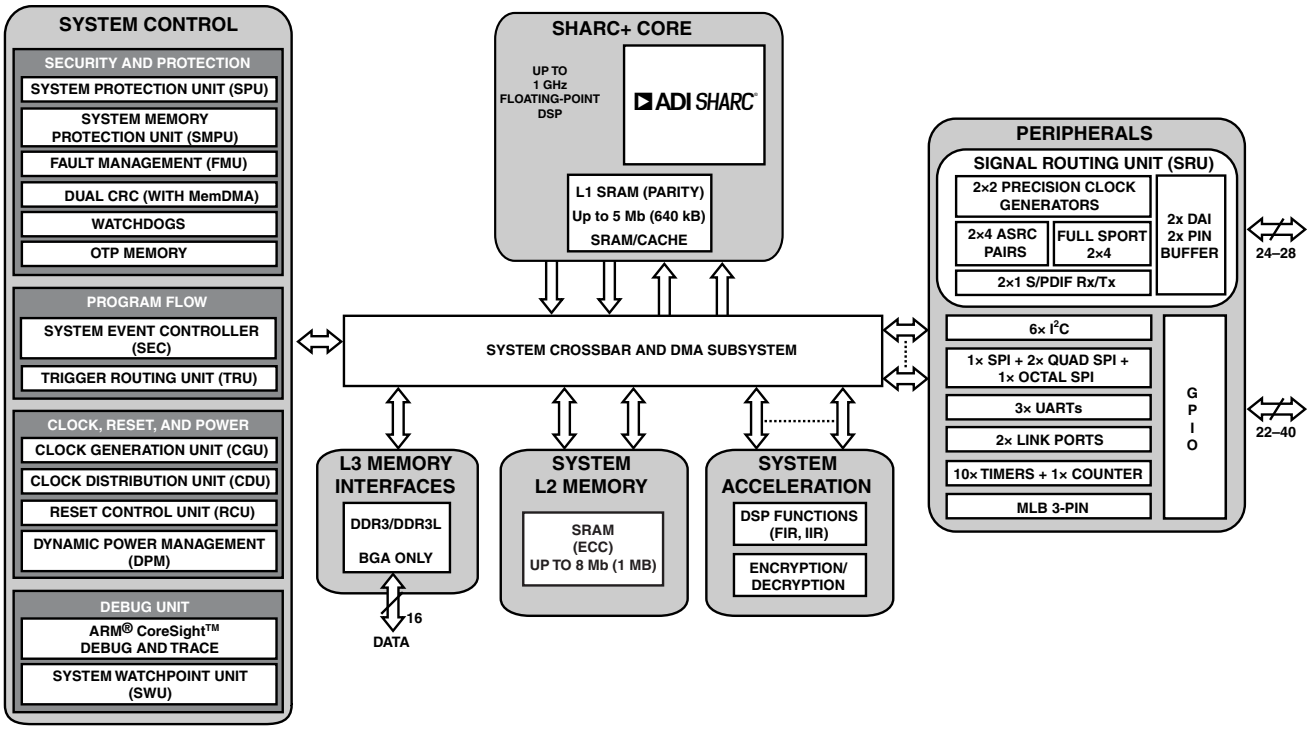


Figure 1. Processor Block Diagram

Rev. PrG **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

6/2020—Revision PrF to Revision PrG

Changes to Memory 1
Changes to Processor Features 3
Changes to Dynamic Memory Controller (DMC) 13
Changes to Preliminary Clock Operating Conditions 44

GENERAL DESCRIPTION

Reaching speeds up to 1 GHz, the ADSP-2156x processors are members of the SHARC® family of products. The ADSP-2156x processor is based on the SHARC+® single core. The ADSP-2156x SHARC processors are members of the SIMD SHARC family of digital signal processors (DSPs) that feature Analog Devices Super Harvard Architecture. These 32-bit/40-bit/64-bit floating-point processors are optimized for high performance audio/floating-point applications with 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 rich set of industry-leading system peripherals and memory (see Table 1), the SHARC+ processor is the platform of choice for applications that require programmability similar to RISC (reduced instruction set computing), multimedia support, and leading edge signal processing in one integrated package. These applications span a wide array of markets, including automotive, pro audio, and industrial-based applications that require high floating-point performance.

Table 1. Processor Features[1]

Processor Feature		ADSP-21562	ADSP-21563	ADSP-21565
SHARC+ Core (MHz max)[2]		400	600, 800	800, 1000
SHARC L1 SRAM (kB)		640	640	640
System Memory	L2 SRAM (kB)	256	512	1024
	DDR3 and DDR3L Controller (16-bit)	N/A	N/A	N/A
DAI (includes SRU)		2	2	2
	Full SPORTs	2 × 4	2 × 4	2 × 4
	S/PDIF Rx/Tx	2 × 1	2 × 1	2 × 1
	ASRCs	2 × 4	2 × 4	2 × 4
	Precision Clock Generators	2 × 2	2 × 2	2 × 2
	Buffers	2 × 12	2 × 12	2 × 12
Hardware Accelerators				
	FIR/IIR	Yes	Yes	Yes
	Security Crypto Engine	Yes	Yes	Yes
I^2C (TWI)		4	4	4
SPI		1	1	1
Quad SPI		2	2	2
Octal SPI		1	1	1
UARTs		2	2	2
Link Port		0	0	0
General-Purpose Timer[3]		6	6	6
General-Purpose Counter		1	1	1
Watchdog Timer		2	2	2
MLB 3-pin		Automotive Models Only		
GPIO Ports		Port A to B		
GPIO + DAI Pins		22 + 24		
Package Options		120-Lead LQFP_EP		

[1] N/A means not applicable.
[2] The values refer to different speed grades.
[3] Refer to Table 14 for internal timer signal routing.

SHARC PROCESSOR

Figure 2 shows the SHARC processor integrates a SHARC+ SIMD core, L1 memory crossbar, I/D cache controller, L1 memory blocks, and the master/slave ports. Figure 3 shows the SHARC+ SIMD core block diagram.

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.

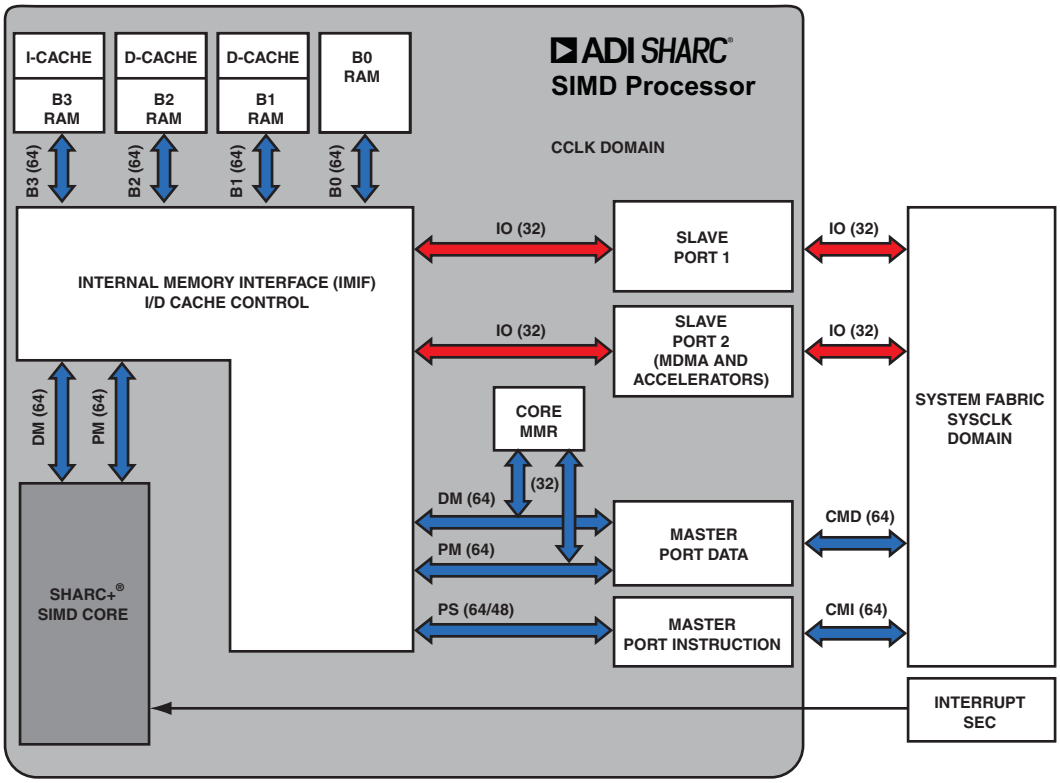


Figure 2. SHARC Processor Block Diagram

L1 Memory

Figure 4 shows the ADSP-2156x memory map. The SHARC+ core has a tightly coupled L1 5 Mb SRAM. The SHARC+ core can access code and data in a single cycle from this memory space.

In the SHARC+ core private address space, the core has L1 memory.

SHARC+ core memory-mapped register (CMMR) address space is 0x 0000 0000 through 0x0003 FFFF 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 1024 Kb can be configured for data memory (DM), program memory (PM), and instruction cache each. 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. While each memory block can store combinations of code and data, accesses are most efficient when one block stores data using the DM bus for transfers, and the other block stores instructions and data using the PM bus for transfers.

Using the DM 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

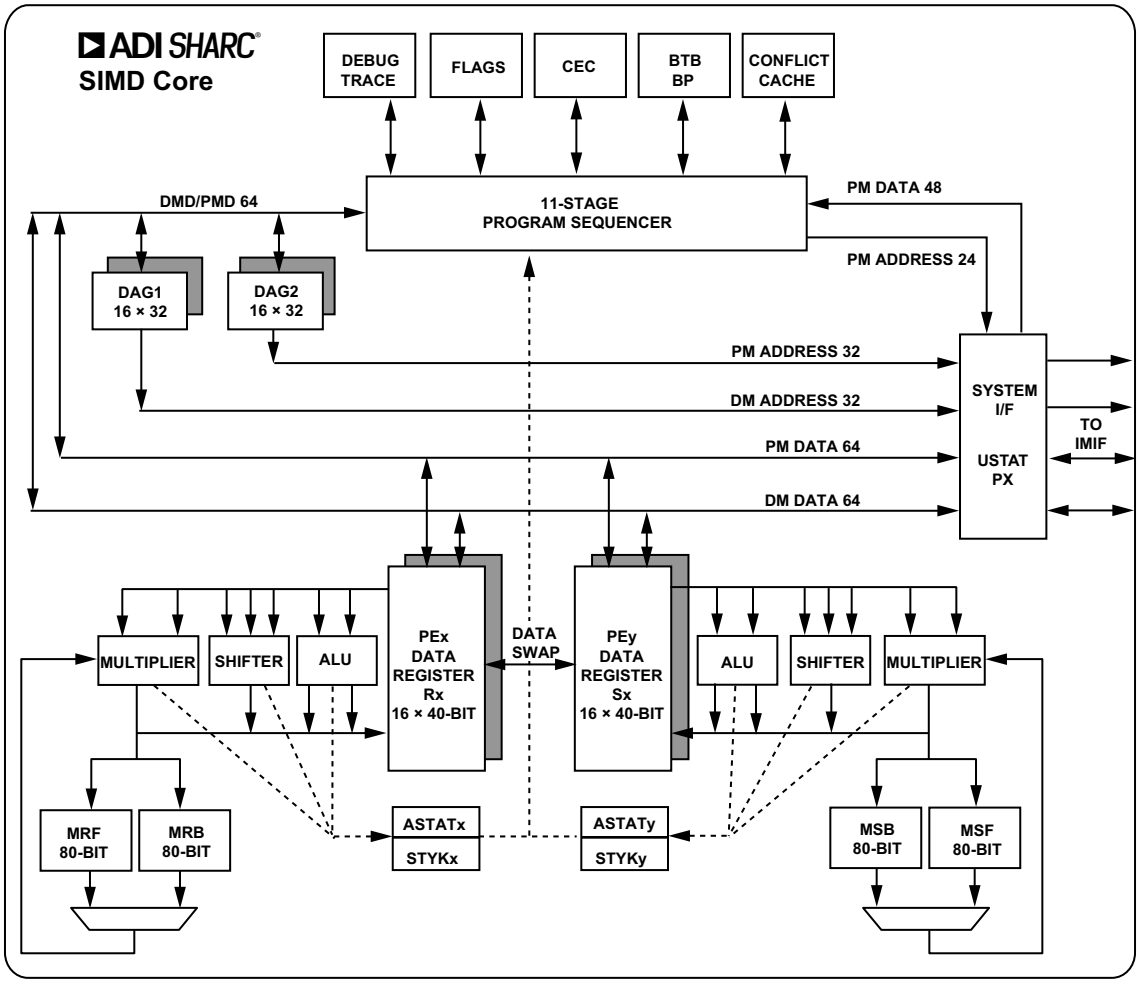


Figure 3. SHARC+ SIMD Core Block Diagram

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 Master and Slave Ports

The SHARC+ core has two master/slave ports to and from the system fabric. One master port fetches instructions. The second master port drives data to the system world. Slave port 1 together with slave port 2 (MDMA) run conflict free access to the individual memory blocks. For the slave port addresses, 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-2156x processors also include a traditional instruction cache (I-cache) and two data caches (D-cache) (PM/DM caches) 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 Kb to a maximum of 1024 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.

System Event Controller (SEC) Input

The output of the system event controller (SEC) controller is forwarded to the core event controller (CEC) to respond directly to all unmasked system-based interrupts. The SEC also supports nesting including various SEC interrupt channel

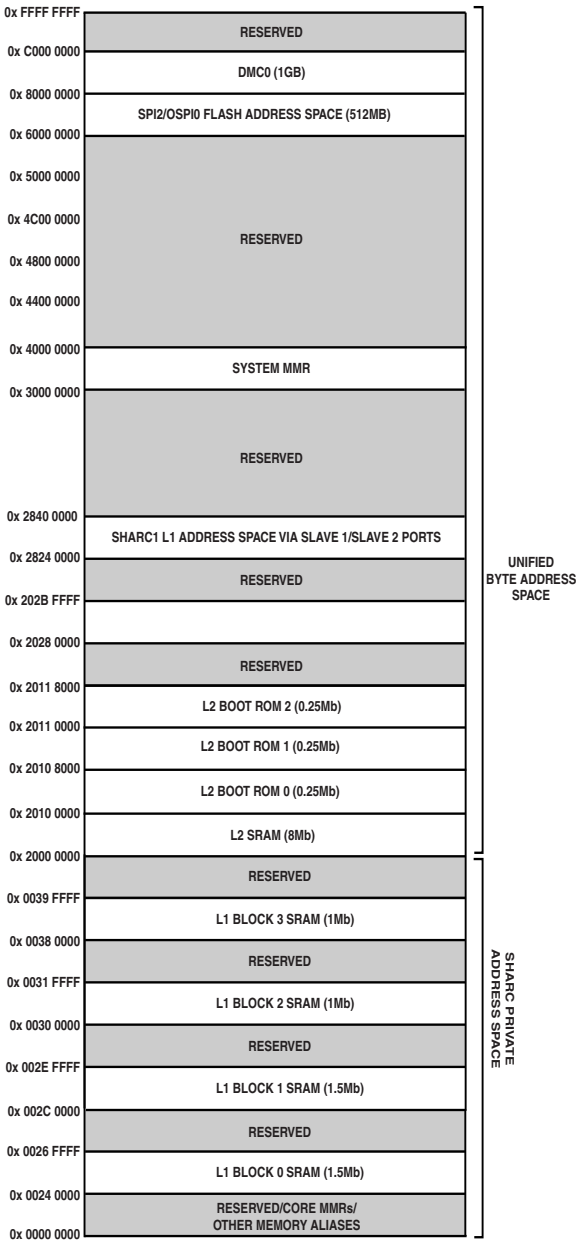


Figure 4. ADSP-2156x Memory Map

arbitration options. For all SEC channels, the processor automatically stacks the arithmetic status (ASTATx and ASTATy) registers and mode (MODE1) register in parallel with the interrupt servicing.

Core Memory-Mapped Registers (CMMR)

The core memory-mapped registers control the L1 instruction and data cache, BTB, parity error, system control, debug, and monitor functions.

SHARC+ CORE ARCHITECTURE

The ADSP-2156x processors are code compatible at the assembly level with the ADSP-2158x, ADSP-2157x, ADSP-2148x, ADSP-2147x, ADSP-2146x, ADSP-2137x, ADSP-2136x, ADSP-2126x, ADSP-2116x, and with the first-generation ADSP-2106x SHARC processors.

The ADSP-2156x processors share architectural features with the ADSP-2126x, ADSP-2136x, ADSP-2137x, ADSP-214xx, ADSP-2157x, ADSP-2158x, and ADSP-2116x SIMD SHARC processors, shown in Figure 3 and detailed 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 data registers and each contain 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 also affects the way data transfers between memory and the processing elements because to sustain computational operation in the processing elements requires twice the data bandwidth. Therefore, entering SIMD mode doubles the bandwidth between memory and the processing elements. When using the DAGs to transfer data in SIMD mode, two data values transfer with each memory or register file access.

Independent Parallel Computation Units

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

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

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

Core Timer

The SHARC+ processor core also has a 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, while control bits in MODE1 activate the secondary registers.

Universal Registers

General-purpose tasks use the universal registers. The four 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-2156x processor uses the two data address generators (DAGs). Circular buffers allow efficient programming of delay lines and other data structures required in digital signal processing, and are commonly used in digital filters and fast Fourier transforms (FFT). The two DAGs of the processors 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 wrap-around, 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 an addition to the SHARC+ core.

Variable Instruction Set Architecture (VISA)

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

Single-Cycle Fetch of Instructional Four Operands

The ADSP-2156x processors feature an enhanced Harvard architecture in which the DM bus transfers data and 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 generates various core interrupts (including arithmetic and circular buffer instruction flow exceptions) and SEC events (debug or monitor and software). The core only responds to unmasked interrupts (enabled in the IMASK register).

Instruction Conflict Cache

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

Additional Features

To enhance the reliability of the application, L1 data RAMs support parity error detection logic for every byte. Additionally, the processors detect illegal opcodes. Core interrupts flag both errors. Master ports of the core also detect for failed external accesses.

SYSTEM INFRASTRUCTURE

The following sections describe the system infrastructure of the ADSP-2156x processors.

System L2 Memory

A system L2 SRAM memory of up to 8 Mb (1 MB) is available to the SHARC+ core and the system DMA channels (see Table 3). The L2 SRAM block is subdivided into eight banks to support concurrent access to the L2 memory ports. Memory accesses to the L2 memory space are multicycle accesses by the SHARC+ core.

The memory space is used for various situations including

- 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 the SHARC+ core to avoid external memory latencies and reduce external memory bandwidth
- Storage for data coefficient tables cached by the SHARC+ core

See System Memory Protection Unit (SMPU) section for options in limiting access by the core and DMA masters.

One Time Programmable Memory (OTP)

The processors feature 7 Kb of one time programmable (OTP) memory which is memory-map accessible. This memory can be programmed with custom keys and it supports secure boot and secure operation.

I/O Memory Space

Mapped I/Os include SPI2 or OSPI0 memory address spaces (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 (1.5 Mb)	0x00048000–0x0004DFFF	0x00090000–0x00097FFF	0x00090000–0x0009BFFF	0x00120000–0x00137FFF	0x00240000–0x0026FFFF
L1 Block 1 SRAM (1.5 Mb)	0x00058000–0x0005DFFF	0x000B0000–0x000B7FFF	0x000B0000–0x000BBFFF	0x00160000–0x00177FFF	0x002C0000–0x002EFFFF
L1 Block 2 SRAM (1 Mb)	0x00060000–0x00063FFF	0x000C0000–0x000C5554	0x000C0000–0x000C7FFF	0x00180000–0x0018FFFF	0x00300000–0x0031FFFF
L1 Block 3 SRAM (1 Mb)	0x00070000–0x00073FFF	0x000E0000–0x000E5554	0x000E0000–0x000E7FFF	0x001C0000–0x001CFFFF	0x00380000–0x0039FFFF

Table 3. L2 Memory Addressing Map

Memory[1]	Byte Address Space SHARC+ Data Access	Normal Word Address Space SHARC+ Data Address	VISA Address Space SHARC+ Instruction Fetch	ISA Address Space SHARC+ Instruction Fetch
L2 RAM (2 Mb)	0x200C0000–0x200FFFFF	0x08030000–0x0803FFFF	0x00BE0000–0x00BFFFFF	0x005E0000–0x005EAAAA
L2 RAM (4 Mb)	0x20080000–0x200FFFFF	0x08020000–0x0803FFFF	0x00BC0000–0x00BFFFFF	0x005D5556–0x005EAAAA
L2 RAM (8 Mb)	0x20000000–0x200FFFFF	0x08000000–0x0803FFFF	0x00B80000–0x00BFFFFF	0x005C0000–0x005EAAAA
L2 Boot ROM0	SHARC/DMA: 0x20100000–0x20107FFF	0x08040000–0x08041FFF	0x00B20000–0x00B23FFF	0x00580000–0x00581555
L2 Boot ROM1	0x20108000–0x2010FFFF	0x08042000–0x08043FFF	0x00B00000–0x00B03FFF	0x00500000–0x00501555
L2 Boot ROM2	0x20110000–0x20117FFF	0x08044000–0x08045FFF	0x00B40000–0x00B43FFF	0x00540000–0x00541555

[1] The L2 RAM blocks are subdivided into banks—the 8 Mb L2 models have eight banks, the 4 Mb models have four banks, and there are two banks for the 2 Mb models.

Table 4. SHARC+® L1 Memory Space

	Memory Block	Byte Address Space SHARC+	Normal Word Address Space SHARC+
L1 Memory Space Address via Slave1/Slave2 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

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

	Byte Address Space 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)	0x60000000–0x600FFFFF	0x04000000–0x07FFFFFF	0x00F80000–0x00FFFFFF	0x00780000–0x007FFFFF
	0x60100000–0x602FFFFF		Not applicable	
	0x60300000–0x6FFFFFFF		Not applicable	Not applicable
	0x70000000–0x7FFFFFFF	Not applicable	Not applicable	Not applicable

[1] 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.

Table 6. DMC Memory Map[1]

	Byte Address Space 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 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 masters to system bus slaves, providing concurrent data transfer between multiple bus masters and multiple bus slaves. A hierarchical model—built from multiple SCBs—provides a power and area efficient system 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 masters to access bus slaves simultaneously
- Protection model (privileged/secure) support for selective bus interconnect protection

Direct Memory Access (DMA)

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

DMA transfers can occur between memory and a peripheral or between one memory and another memory. Each memory to memory DMA stream uses two channels: 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 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 SYCLK)

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 Redundant Code (CRC) Protection

The cyclic redundant codes (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, every 100 ms the system software initiates the signature calculation of the entire memory contents and compares these contents 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 a 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 exceptions event occurs synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions triggered on the one side by the SHARC+ core, such as data alignment (SIMD or long word) or compute violations (fixed or floating point), and illegal instructions cause core exceptions. Conditions triggered on the other side by the SEC, such as error correcting codes (ECC), parity, watchdog, or system clock, cause system exceptions.
- An interrupts event occurs asynchronously to program flow. They are caused by input signals, timers, and other peripherals, as well as by an explicit software instruction.

System Event Controller (SEC)

The SHARC+ core features a system event controller. The SEC features include the following:

- Comprehensive system event source management, including interrupt enable, fault enable, priority, 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 slave 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 masters (generators of triggers) to trigger slaves (receivers of triggers). Slave endpoints can be configured to respond to triggers in various ways. Common applications enabled by the TRU include,

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

SECURITY FEATURES

The following sections describe the security features of the ADSP-2156x processors.

Cryptographic Hardware Accelerators

The ADSP-2156x 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 three system MMR masters (SHARC+ core, memory DMA, and CoreSight debug) the peripherals are guarded against.

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

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-2156x processors for each memory space, except for SHARC L1 memory.

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 masters 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-2156x processors are designed to support functional safety applications. While the level of safety is mainly dominated by the system concept, the following primitives are provided by the 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.

Error Correcting Codes (ECC) Protected L2 Memories

Error correcting codes (ECC) correct 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-Protected Peripheral Memories

Parity protection is added to all peripheral memories:

- ASRC
- IIR
- FIR
- CRYPTO
- MLB
- TRACE

Cyclic Redundant Code (CRC) Protected Memories

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

CRC checksums can be calculated or compared automatically during memory transfers, or one or multiple memory regions can be continuously scrubbed by a single DMA work unit as per DMA descriptor chain instructions. The CRC engine also protects data loaded during the boot process.

Signal Watchdogs

The ten general-purpose timers feature modes to monitor off-chip signals. The watchdog period mode monitors whether external signals toggle with a period within an expected range. The watchdog width mode monitors whether the pulse widths of external signals are within an expected range. Both modes help 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-2156x 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 mirrored 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 the DAI 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 Buffers 20 and 19 can change the polarity of the input signals. Most signals of the peripherals belonging to different DAIs cannot be interconnected, with few exceptions.

The DAI_PINx pin buffers may 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 hardware reference manual for complete information on the use of the DAIs and SRUs.

Serial Port (SPORT)

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

An individual full SPORT module consists of two independently configurable SPORT halves with identical functionality. Two bidirectional data lines—primary (0) and secondary (1)—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 signals, while the other half SPORT provides the two receive 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 without converting them to an analog signal. 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 compatible 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. The S/PDIF transmitter receives audio data in serial format and converts it into a biphase encoded signal. The serial data input to the transmitter can be formatted as left justified, I^2S, or right justified with word widths of 16, 18, 20, or 24 bits. The S/PDIF receiver converts a biphase encoded signal into I^2S serial format.The serial data, clock, and frame sync outputs/inputs from/to the S/PDIF receiver/transmitter are routed through the signal routing unit (SRU). They can be connected to various peripherals, 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 four units: units A/B located in the DAI0 block, and units C/D located in the DAI1 block. The PCG can generate a pair of signals (clock and frame sync) derived from a clock input signal (SCLK0, SYS_CLKIN0, or DAI pin buffer). Each unit can also output to the pin buffers of the opposite DAI unit. All 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.

Universal Asynchronous Receiver/Transmitter (UART) Ports

The processors provide 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 8 bits in parallel). The OSPI port supports DDR modes of operation, which enable the transfer of up to 16 bits of data in each clock. The OSPI port provides overall data throughput and performance improvement, including faster boot time.

Feature of the OSPI port include:

- Support for single, dual, quad, or octal I/O transfers
- Multiple modes of operation including direct and STIG (software triggered instruction generator)
- Support for XIP (execute in place): continuous mode
- Programmable page and block sizes
- Programmable write protected regions
- Programmable memory timing
- Support for DDR commands

Link Port (LP)

Two 8-bit wide link ports (LP) for the BGA package 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.

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 ten general-purpose programmable timers. Each timer has an external pin that can be configured either 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 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/or stopped by any TRU master without core intervention.

Watchdog Timer (WDT)

Two on-chip software watchdog timers (WDT) can be used by the SHARC+ core. 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 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 either controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can 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.

Media Local Bus (MediaLB)

The automotive models have a MediaLB (MLB) slave interface that allows the processors to function as a media local bus device. It includes support for 3-pin media local bus protocols. The MLB 3-pin configuration supports speeds up to 1024 × 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 slave mode only.

2-Wire Controller Interface (TWI)

The processors include 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 master and slave operation and support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (TWI_SCL) and data (TWI_SDA) and supports the protocol at speeds up to 400 kb/sec. 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:

- 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, while output pins can be triggered by software interrupts.
- GPIO interrupt sensitivity registers specify whether individual pins are level or edge sensitive and specify, if edge sensitive, whether 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. Three system-level interrupt channels (PINT0–PINT2) are reserved for this purpose. Each of these interrupt channels can manage up to 32 interrupt pins. The assignment from pin to interrupt is not performed on a pin by pin basis. Rather, groups of eight pins (half ports) can be flexibly assigned to interrupt channels.

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

SYSTEM ACCELERATION

The following sections describe the system acceleration blocks of the ADSP-2156x 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 MAC units. A controller manages the accelerator. The FIR accelerator runs at the core clock frequency. The FIR accelerator can access all memory spaces and can run concurrently with the IIR accelerator 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 core clock frequency. The IIR accelerator can access all memory spaces and run concurrently with the other accelerators on the processor.

In addition to operating at core clock, the FIR/IIR accelerators support various enhanced features, including the ability to halt the accelerator for dynamic queuing of unlimited FIR/IIR channels, selective interrupt generation for each channel, and trigger master/slave support.

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 starts 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 causes resources to stall. This 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 master.

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—can be triggered by software (writing to the RCU_CTL register).
- Trigger request (peripheral).

Clock Generation Unit (CGU)

The ADSP-2156x processors support two independent PLLs. Each PLL is part of a clock generation unit (CGU). Each CGU is driven externally by the same clock source. This provides flexibility in determining the internal clocking frequencies for each clock domain.

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

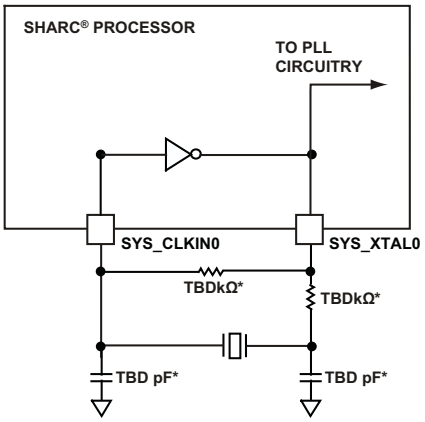
The CGU generates all on-chip clocks and synchronization signals. Multiplication factors are programmed to define the PLLCLK frequency.

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 hardware reference manual.

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

System Crystal Oscillator

The processor can be clocked by an external crystal (see Figure 5), 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. When using an external clock, the clock signal is connected to the SYS_CLKIN0 pin of the processor and the SYS_XTAL0 pin must be left unconnected. Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal can be used.



NOTE: VALUES MARKED WITH * MUST BE CUSTOMIZED, DEPENDING ON THE CRYSTAL AND LAYOUT. ANALYZE CAREFULLY. VALID FREQUENCY RANGE IS 20 MHz TO 30 MHz FOR SYS_CLKIN0.

Figure 5. External Crystal Connection

For fundamental frequency operation, use the circuit shown in Figure 5. A parallel resonant, fundamental frequency, microprocessor grade crystal is connected across the SYS_CLKIN0 pin and the SYS_XTAL0 pin.

The two capacitors and the series resistor, shown in Figure 5, fine tune phase and amplitude of the sine frequency. The capacitor and resistor values shown in Figure 5 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.

Clock Distribution Unit (CDU)

The two CGUs each provide outputs which feed a clock distribution unit (CDU). The clock outputs CLKO0–CLKO1 and the CGU outputs are connected to various targets. For more information, refer to the hardware reference manual.

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 hardware reference manual to change the default mapping of clocks.

Booting

The processors have several mechanisms for automatically loading internal and external memory after a reset. The boot mode is defined by the SYS_BMODE[n] input pins. There are two categories of boot modes. In master boot mode, the processors actively load data from serial memories. In slave boot modes, the processors receive data from external host devices.

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.

Table 7. Boot Modes

SYS_BMODE[n] Setting[1]	Boot Mode
000	No boot
001	SPI2 master
010	SPI2 slave
011	UART0 slave
100	Link0 slave
101[2]	Octal SPI master
110	Reserved
111	Reserved

[1] SYS_BMODE2 pin is applicable only for the BGA package.
[2] Though octal SPI master boot is not supported on the LQFP package, it is available through the ROM API.

In the ADSP-2156x processors, the SHARC+ core controls the boot process, including loading all internal and external memory. The option for secure boot is available on all models.

Power Supplies

The processors have separate power supply connections for:

- Internal (VDD_INT)
- External (VDD_EXT)
- External (VDD_REF)
- DMC (VDD_DMC)

Power Management

As shown in Table 8, the processors support five 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}
All other I/O (includes SYS, JTAG, and ports pins except SYS_CLKIN0)	V_{DD_EXT}
SYS_CLKIN0	V_{DD_REF}

Power-Up and Power-Down Sequencing

The VDD_REF and VDD_EXT supplies must be turned on, along with other power supplies, such that the absolute voltage difference between VDD_EXT and VDD_REF does not exceed $V_{DELTA\_EXT\_REF}$. SYS_XTAL0 oscillations (SYS_CLKIN0) 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 oscillations are valid. For further details and information, see Power-Up Reset Timing.

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 slave. The SWU provides ports for all system bus address channel signals. Each SWU contains four match groups of registers with associated hardware. These four SWU match groups operate independently but share common event (for example, interrupt and trigger) outputs.

Debug Access Port (DAP)

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 (CrossCore® Embedded Studio), evaluation products, emulators, and a variety of software add ins.

Integrated Development Environments (IDEs)

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

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

EZ-KIT Evaluation Board

For processor evaluation, Analog Devices provides a wide range of EZ-KIT® evaluation boards. Including the processor and key peripherals, the evaluation board also supports on-chip emulation capabilities and other evaluation and development features. Various EZ-Extenders® are also available, which are daughter cards that deliver additional specialized functionality, including audio and video processing. For more information visit SHARC Processors Software and Tools.

EZ-KIT Evaluation Kits

For a cost-effective way to learn more about developing with Analog Devices processors, Analog Devices offer a range of EZ-KIT evaluation kits. Each evaluation kit includes an EZ-KIT evaluation board, directions for downloading an evaluation version of the available IDE(s), a USB cable, and a power supply.

The USB controller on the EZ-KIT board connects to the USB port of the user PC, enabling the chosen IDE evaluation suite to emulate the on-board processor in-circuit.

This permits users to download, execute, and debug programs for the EZ-KIT system. It also supports in circuit programming of the on-board Flash device to store user specific boot code, enabling standalone operation. With the full version of CrossCore Embedded Studio installed (sold separately), engineers can develop software for supported EZ-KITs or any custom system utilizing supported Analog Devices processors.

Software Add Ins for CrossCore Embedded Studio

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

Board Support Packages for Evaluation Hardware

Software support for the EZ-KIT evaluation boards and EZ-Extender daughter cards is provided by software add ins called board support packages (BSPs). The BSPs contain the required drivers, pertinent release notes, and select example code for the given evaluation hardware. A download link for a specific BSP is located on the web page for the associated EZ-KIT or EZ-Extender product.

Middleware Packages

Analog Devices offers middleware add ins for real-time operating systems. For more information, see the following web pages:

- www.analog.com/ucos3
- www.analog.com/FreeRTOS

Algorithmic Modules

To speed development, Analog Devices offers add ins that perform popular audio and video processing algorithms. These are available for use with CrossCore Embedded Studio. For more information visit www.analog.com/cces.

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 once an operation is completed by the emulator, the DSP system is set to run at full speed with no impact on system timing. The emulators require the target board to include a header that supports connection of the 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-2156x 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 well. A tool for viewing relationships between specific applications and related components is available on the www.analog.com\circuits website.

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-2156x DETAILED SIGNAL DESCRIPTIONS

Table 9 provides a detailed description of each pin.

Table 9. ADSP-2156x Detailed Signal Descriptions

Signal Name	Direction	Description
C0_FLG[n]	InOut	**SHARC Core 0 Flag Pin.**
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).
DMC_A[nn]	Output	**Address n.** Address bus.
DMC_BA[n]	Output	**Bank Address 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.
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.
DMC_VREF	Input	**Voltage Reference.** Connects to half of the VDD_DMC voltage. Applies to the DMC0_VREF pin.
$\overline{\text{DMC\_WE}}$	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.
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.

Table 9. ADSP-2156x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
JTG_TRST	Input	**JTAG Reset.** JTAG test access port reset.
LP_ACK	InOut	**Acknowledge.** Provides handshaking. When the link port is configured as a receiver, ACK is an output. When the link port is configured as a transmitter, ACK is an input.
LP_CLK	InOut	**Clock.** When the link port is configured as a receiver, CLK is an input. When the link port is configured as a transmitter, CLK is an output.
LP_D[n]	InOut	**Data n.** Data bus. Input when receiving, output when transmitting.
MLB_CLK	InOut	**Single-Ended Clock.**
MLB_DAT	InOut	**Single-Ended Data.**
MLB_SIG	InOut	**Single-Ended Signal.**
OSPI_CLK	Output	**SPI Master Clock Output.** SPI master clock output.
OSPI_D2	InOut	**Data 2.** Transfers serial data in quad and octal modes.
OSPI_D3	InOut	**Data 3.** Transfers serial data in quad and octal modes.
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_MISO	InOut	**Master In, Slave Out.** Transfers serial data. Operates in the same direction as SPI_MOSI in dual and quad modes.
OSPI_MOSI	InOut	**Master Out, Slave In.** Transfers serial data. Operates in the same direction as SPI_MISO in dual and quad modes.
OSPI_SEL[n]	Output	**Slave Select Output n.** Used in master mode to enable the desired slave.
P_[nn]	InOut	**Position n.** General-purpose input/output. See the ADSP-2156x SHARC+ Processor Hardware Reference manual for programming information.
SPI_CLK	Output	**SPI Master Clock Output.** SPI master clock output.
SPI_D2	InOut	**Data 2.** Transfers serial data in quad modes.
SPI_D3	InOut	**Data 3.** Transfers serial data in quad modes. 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.
SPI_SEL[n]	Output	**Slave Select Output n.** Used in master mode to enable the desired slave.
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.
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.

Table 9. ADSP-2156x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
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_CLKOUT	Output	**Processor Clock Output.** Outputs internal clocks. Clocks may be divided down. See the ADSP-2156x SHARC+ Processor Hardware Reference manual for programming information.
SYS_FAULT	InOut	**Active-High Fault Output.** Indicates indicates internal faults or senses external faults depending on the operating mode.
$\overline{\text{SYS\_FAULT}}$	InOut	**Active-Low Fault Output.** Indicates 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.**
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 master, clock input when slave.
TWI_SDA	InOut	**Serial Data.** Receives or transmits data.
$\overline{\text{UART\_CTS}}$	Input	**Clear to Send.** Flow control signal.
$\overline{\text{UART\_RTS}}$	Output	**Request to Send.** Flow control signal.
$\overline{\text{UART\_RX}}$	Input	**Receive.** 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.

400-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processor pin definitions are shown in Table 10 for the 400-ball CSP_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 general-purpose I/O 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 general-purpose I/O 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 ADSP-2156x SHARC+ Processor Hardware Reference manual for complete information on the use of the DAI and SRUs.

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
C0_FLG0	SHARC Core 0 Flag Pin	A	PA_12
C0_FLG1	SHARC Core 0 Flag Pin	A	PA_13
C0_FLG2	SHARC Core 0 Flag Pin	B	PB_03
C0_FLG3	SHARC Core 0 Flag Pin	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

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC0_A00	DMC0 Address 0	Not Muxed	DMC0_A00
DMC0_A01	DMC0 Address 1	Not Muxed	DMC0_A01
DMC0_A02	DMC0 Address 2	Not Muxed	DMC0_A02
DMC0_A03	DMC0 Address 3	Not Muxed	DMC0_A03
DMC0_A04	DMC0 Address 4	Not Muxed	DMC0_A04
DMC0_A05	DMC0 Address 5	Not Muxed	DMC0_A05
DMC0_A06	DMC0 Address 6	Not Muxed	DMC0_A06
DMC0_A07	DMC0 Address 7	Not Muxed	DMC0_A07
DMC0_A08	DMC0 Address 8	Not Muxed	DMC0_A08
DMC0_A09	DMC0 Address 9	Not Muxed	DMC0_A09
DMC0_A10	DMC0 Address 10	Not Muxed	DMC0_A10
DMC0_A11	DMC0 Address 11	Not Muxed	DMC0_A11
DMC0_A12	DMC0 Address 12	Not Muxed	DMC0_A12
DMC0_A13	DMC0 Address 13	Not Muxed	DMC0_A13
DMC0_A14	DMC0 Address 14	Not Muxed	DMC0_A14
DMC0_A15	DMC0 Address 15	Not Muxed	DMC0_A15
DMC0_BA0	DMC0 Bank Address Input 0	Not Muxed	DMC0_BA0
DMC0_BA1	DMC0 Bank Address Input 1	Not Muxed	DMC0_BA1
DMC0_BA2	DMC0 Bank Address Input 2	Not Muxed	DMC0_BA2
$\overline{\text{DMC0\_CAS}}$	DMC0 Column Address Strobe	Not Muxed	$\overline{\text{DMC0\_CAS}}$
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 (DDR3 only)	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

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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_VREF	DMC0 Voltage Reference	Not Muxed	DMC0_VREF
$\overline{\text{DMC0\_WE}}$	DMC0 Write Enable	Not Muxed	$\overline{\text{DMC0\_WE}}$
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_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_MISO/D1	OSPI0 Master In, Slave Out	A	PA_00
OSPI0_MOSI/D0	OSPI0 Master Out, Slave In	A	PA_01
$\overline{\text{OSPI0\_SEL1}}$	OSPI0 Slave Select Output 1	A	PA_05
$\overline{\text{OSPI0\_SEL2}}$	OSPI0 Slave Select Output 2	C	PC_04
$\overline{\text{OSPI0\_SEL3}}$	OSPI0 Slave Select Output 3	C	PC_05
$\overline{\text{OSPI0\_SEL4}}$	OSPI0 Slave Select Output 4	C	PC_07
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

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control pin	Not Muxed	SYS_BMODE1
SYS_BMODE2	Boot Mode Control pin	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-High Fault Output	C	PC_07
SYS_FAULT	Active-Low Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL0	Crystal Output	Not Muxed	SYS_XTAL0
TM0_ACI0	TIMER0 Alternate Capture Input 0	A	PA_07
TM0_ACI1	TIMER0 Alternate Capture Input 1	A	PA_14
TM0_ACI2	TIMER0 Alternate Capture Input 2	B	PB_11
TM0_ACI3	TIMER0 Alternate Capture Input 3	B	PB_00

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TM0_ACI4	TIMER0 Alternate Capture Input 4	A	PA_11
TM0_ACLK1	TIMER0 Alternate Clock 1	A	PA_06
TM0_ACLK2	TIMER0 Alternate Clock 2	A	PA_08
TM0_ACLK3	TIMER0 Alternate Clock 3	A	PA_02
TM0_ACLK4	TIMER0 Alternate Clock 4	B	PB_02
TM0_CLK	TIMER0 Clock	B	PB_01
TM0_TMR0	TIMER0 Timer 0	A	PA_10
TM0_TMR1	TIMER0 Timer 1	A	PA_12
TM0_TMR2	TIMER0 Timer 2	A	PA_13
TM0_TMR3	TIMER0 Timer 3	B	PB_03
TM0_TMR4	TIMER0 Timer 4	B	PB_04
TM0_TMR5	TIMER0 Timer 5	B	PB_05
TM0_TMR6	TIMER0 Timer 6	B	PB_08
TM0_TMR7	TIMER0 Timer 7	B	PB_09
TM0_TMR8	TIMER0 Timer 8	C	PC_05
TM0_TMR9	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	B	PB_08
TRACE0_D02	TRACE0 Trace Data	B	PB_09
TRACE0_D03	TRACE0 Trace Data 3	B	PB_10
TRACE0_D04	TRACE0 Trace Data	C	PC_00
TRACE0_D05	TRACE0 Trace Data	C	PC_01
TRACE0_D06	TRACE0 Trace Data	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
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	A	PA_09
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	A	PA_08
$\overline{\text{UART0\_RX}}$	UART0 Receive	A	PA_07
$\overline{\text{UART0\_TX}}$	UART0 Transmit	A	PA_06
$\overline{\text{UART1\_CTS}}$	UART1 Clear to Send	B	PB_01
$\overline{\text{UART1\_RTS}}$	UART1 Request to Send	B	PB_00
$\overline{\text{UART1\_RX}}$	UART1 Receive	A	PA_14
$\overline{\text{UART1\_TX}}$	UART1 Transmit	A	PA_15
$\overline{\text{UART2\_CTS}}$	UART2 Clear to Send	B	PB_14

Table 10. ADSP-2156x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
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 FOR 400-BALL CSP_BGA PACKAGE

Table 11 through Table 13 identify the pin functions that are multiplexed on the general-purpose I/O pins of the 400-ball CSP_BGA package.

Table 11. 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/D1			
PA_01	SPI2_MOSI	OSPI0_MOSI/D0			
PA_02	SPI2_D2	OSPI0_D2	TWI3_SCL		TM0_ACLK3
PA_03	SPI2_D3	OSPI0_D3	TWI3_SDA		
PA_04	SPI2_CLK	OSPI0_CLK			
PA_05	SPI2_SEL1	OSPI0_SEL1			SPI2_SS
PA_06	SPI0_CLK	UART0_TX	OSPI0_D4		TM0_ACLK1
PA_07	SPI0_MISO	UART0_RX	OSPI0_D5		TM0_ACI0
PA_08	SPI0_MOSI	UART0_RTS	OSPI0_D6		TM0_ACLK2
PA_09	SPI0_SEL1	UART0_CTS	OSPI0_D7		SPI0_SS
PA_10	TWI0_SCL	SPI1_CLK	TM0_TMR0		
PA_11	TWI0_SDA	SPI1_MISO			TM0_ACI4
PA_12	C0_FLG0	SPI1_MOSI	TM0_TMR1		
PA_13	C0_FLG1	SPI1_SEL1	TM0_TMR2		SPI1_SS
PA_14	TWI2_SCL	SPI1_D2	UART1_RX		TM0_ACI1
PA_15	TWI2_SDA	SPI1_D3	UART1_TX		

Table 12. 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	UART1_RTS		TM0_ACI3
PB_01	MLB0_SIG	TWI1_SDA	UART1_CTS		TM0_CLK
PB_02	MLB0_CLK	C0_FLG3	LP1_ACK		TM0_ACLK4
PB_03	TM0_TMR3	C0_FLG2	SPI2_SEL2		CNT0_UD
PB_04	TM0_TMR4	SPI1_RDY	LP0_ACK		CNT0_ZM
PB_05	TM0_TMR5	SPI2_RDY	SPI0_SEL2		CNT0_DG
PB_06	LP0_CLK	SPI1_SEL5		TRACE0_CLK	
PB_07	LP0_D0	SPI2_SEL5		TRACE0_D00	
PB_08	LP0_D1	SPI1_SEL7	TM0_TMR6	TRACE0_D01	
PB_09	LP0_D2	SPI2_SEL7	TM0_TMR7	TRACE0_D02	
PB_10	LP0_D3	SPI1_SEL2		TRACE0_D03	
PB_11	LP0_D4	SPI0_RDY		UART2_RX	TM0_ACI2
PB_12	LP0_D5	SPI2_SEL3		UART2_TX	
PB_13	LP0_D6	SPI1_SEL3		UART2_RTS	
PB_14	LP0_D7	SPI0_SEL3		UART2_CTS	
PB_15	LP1_D0	SPI0_SEL4			

Table 13. 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	SPI1_SEL4	
PC_01	LP1_D2	TWI4_SDA	TRACE0_D05	SPI2_SEL4	
PC_02	LP1_D3	TWI5_SCL	TRACE0_D06	SPI1_SEL6	
PC_03	LP1_D4	TWI5_SDA	TRACE0_D07	SPI2_SEL6	
PC_04	LP1_D5	OSPI0_SEL2			
PC_05	LP1_D6	OSPI0_SEL3	TM0_TMR8		
PC_06	LP1_D7	SPI1_RDY			
PC_07	LP1_CLK	OSPI0_SEL4	TM0_TMR9	SYS_FAULT[1]	

[1] The default PC_07 pin function is SYS_FAULT until the PORTC_FER and PORTC_MUX registers are explicitly programmed to change it.

Table 14 shows the internal timer signal routing. This table applies to both the 400-ball CSP_BGA and 120-lead LQFP packages.

Table 14. Internal Timer Signal Routing

Timer Input Signal	Internal Source
TM0_ACLK0	SYS_CLKIN0
TM0_ACI5	DAI0_PB04_O
TM0_ACLK5	DAI0_PB03_O
TM0_ACI6	DAI1_PB04_O
TM0_ACLK6	DAI1_PB03_O
TM0_ACI7	CNT0_TO signal
TM0_ACLK7	SYS_CLKIN0
TM0_ACI8	DAI0_PB06_O
TM0_ACLK8	DAI0_PB05_O
TM0_ACI9	DAI1_PB06_O
TM0_ACLK9	DAI1_PB05_O

120-LEAD LQFP SIGNAL DESCRIPTIONS

The processor pin definitions are shown Table 15 for the 120-lead LQFP 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 general-purpose I/O 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 general-purpose I/O 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 ADSP-2156x SHARC+ Processor Hardware Reference manual for complete information on the use of the DAI and SRUs.

Table 15. ADSP-2156x 120-Lead LQFP Signal Descriptions

Signal Name	Description	Port	Pin Name
C0_FLG0	SHARC Core 0 Flag Pin	A	PA_12
C0_FLG1	SHARC Core 0 Flag Pin	A	PA_13
C0_FLG2	SHARC Core 0 Flag Pin	B	PB_03
C0_FLG3	SHARC Core 0 Flag Pin	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_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_PIN19	DAI1 Pin 19	Not Muxed	DAI1_PIN19
DAI1_PIN20	DAI1 Pin 20	Not Muxed	DAI1_PIN20
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

Table 15. ADSP-2156x 120-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
JTG_TRST	JTAG Reset	Not Muxed	JTG_TRST
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_MISO/D1	OSPI0 Master In, Slave Out	A	PA_00
OSPI0_MOSI/D0	OSPI0 Master Out, Slave In	A	PA_01
OSPI0_SEL1	OSPI0 Slave Select Output 1	A	PA_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_SEL1	SPI0 Slave Select Output 1	A	PA_09
SPI0_SEL2	SPI0 Slave Select Output 2	B	PB_05
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	B	PB_04
SPI1_SEL1	SPI1 Slave Select Output 1	A	PA_13
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_SS	SPI2 Slave Select Input	A	PA_05
SYS_BMODE0	Boot Mode Control pin	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control pin	Not Muxed	SYS_BMODE1
SYS_CLKIN0	Clock/Crystal Input	Not Muxed	SYS_CLKIN0
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_FAULT	Active-Low Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL0	Crystal Output	Not Muxed	SYS_XTAL0
TM0_ACI0	TIMER0 Alternate Capture Input 0	A	PA_07
TM0_ACI1	TIMER0 Alternate Capture Input 1	A	PA_14
TM0_ACI3	TIMER0 Alternate Capture Input 3	B	PB_00

Table 15. ADSP-2156x 120-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TM0_ACI4	TIMER0 Alternate Capture Input 4	A	PA_11
TM0_ACLK1	TIMER0 Alternate Clock 1	A	PA_06
TM0_ACLK2	TIMER0 Alternate Clock 2	A	PA_08
TM0_ACLK3	TIMER0 Alternate Clock 3	A	PA_02
TM0_ACLK4	TIMER0 Alternate Clock 4	B	PB_02
TM0_CLK	TIMER0 Clock	B	PB_01
TM0_TMR0	TIMER0 Timer 0	A	PA_10
TM0_TMR1	TIMER0 Timer 1	A	PA_12
TM0_TMR2	TIMER0 Timer 2	A	PA_13
TM0_TMR3	TIMER0 Timer 3	B	PB_03
TM0_TMR4	TIMER0 Timer 4	B	PB_04
TM0_TMR5	TIMER0 Timer 5	B	PB_05
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
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	A	PA_09
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	A	PA_08
$\overline{\text{UART0\_RX}}$	UART0 Receive	A	PA_07
$\overline{\text{UART0\_TX}}$	UART0 Transmit	A	PA_06
$\overline{\text{UART1\_CTS}}$	UART1 Clear to Send	B	PB_01
$\overline{\text{UART1\_RTS}}$	UART1 Request to Send	B	PB_00
$\overline{\text{UART1\_RX}}$	UART1 Receive	A	PA_14
$\overline{\text{UART1\_TX}}$	UART1 Transmit	A	PA_15

GPIO MULTIPLEXING FOR 120-LEAD LQFP

Table 16 and Table 17 identify the pin functions that are multiplexed on the general-purpose I/O pins of the 120-lead LQFP package.

Table 16. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PA_00	SPI2_MISO	OSPI0_MISO/D1		
PA_01	SPI2_MOSI	OSPI0_MOSI/D0		
PA_02	SPI2_D2	OSPI0_D2	TWI3_SCL	TM0_ACLK3
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_ACLK1
PA_07	SPI0_MISO	$\overline{\text{UART0\_RX}}$	OSPI0_D5	TM0_ACI0
PA_08	SPI0_MOSI	$\overline{\text{UART0\_RTS}}$	OSPI0_D6	TM0_ACLK2
PA_09	$\overline{\text{SPI0\_SEL1}}$	$\overline{\text{UART0\_CTS}}$	OSPI0_D7	$\overline{\text{SPI0\_SS}}$
PA_10	TWI0_SCL	SPI1_CLK	TM0_TMR0	
PA_11	TWI0_SDA	SPI1_MISO		TM0_ACI4
PA_12	C0_FLG0	SPI1_MOSI	TM0_TMR1	
PA_13	C0_FLG1	$\overline{\text{SPI1\_SEL1}}$	TM0_TMR2	$\overline{\text{SPI1\_SS}}$
PA_14	TWI2_SCL	SPI1_D2	$\overline{\text{UART1\_RX}}$	TM0_ACI1
PA_15	TWI2_SDA	SPI1_D3	$\overline{\text{UART1\_TX}}$	

Table 17. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PB_00	MLB0_DAT	TWI1_SCL	$\overline{\text{UART1\_RTS}}$	TM0_ACI3
PB_01	MLB0_SIG	TWI1_SDA	$\overline{\text{UART1\_CTS}}$	TM0_CLK
PB_02	MLB0_CLK	C0_FLG3		TM0_ACLK4
PB_03	TM0_TMR3	C0_FLG2	$\overline{\text{SPI2\_SEL2}}$	CNT0_UD
PB_04	TM0_TMR4	SPI1_RDY		CNT0_ZM
PB_05	TM0_TMR5	SPI2_RDY	$\overline{\text{SPI0\_SEL2}}$	CNT0_DG

ADSP-2156x DESIGNER QUICK REFERENCE

Table 18 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 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 18. ADSP-2156x Designer Quick Reference

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
DAI0_PIN01	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 1 Notes: See note[2]
DAI0_PIN02	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 2 Notes: See note[2]
DAI0_PIN03	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 3 Notes: See note[2]
DAI0_PIN04	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 4 Notes: See note[2]
DAI0_PIN05	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 5 Notes: See note[2]
DAI0_PIN06	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 6 Notes: See note[2]
DAI0_PIN07	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 7 Notes: See note[2]
DAI0_PIN08	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 8 Notes: See note[2]
DAI0_PIN09	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 9 Notes: See note[2]
DAI0_PIN10	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 10 Notes: See note[2]
DAI0_PIN11	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 11 Notes: See note[2]
DAI0_PIN12	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 12 Notes: See note[2]
DAI0_PIN19	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 19 Notes: See note[2]
DAI0_PIN20	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI0 Pin 20 Notes: See note[2]
DAI1_PIN01	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 1 Notes: See note[2]

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
DAI1_PIN02	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 2 Notes: See note[2]
DAI1_PIN03	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 3 Notes: See note[2]
DAI1_PIN04	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 4 Notes: See note[2]
DAI1_PIN05	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 5 Notes: See note[2]
DAI1_PIN06	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 6 Notes: See note[2]
DAI1_PIN07	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 7 Notes: See note[2]
DAI1_PIN08	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 8 Notes: See note[2]
DAI1_PIN09	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 9 Notes: See note[2]
DAI1_PIN10	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 10 Notes: See note[2]
DAI1_PIN11	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 11 Notes: See note[2]
DAI1_PIN12	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 12 Notes: See note[2]
DAI1_PIN19	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: DAI1 Pin 19 Notes: See note[2]
DAI1_PIN20	InOut	A	Programmable PullUp/PullDown[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

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and 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
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

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and 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
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

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and 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 (complement) Notes: No notes
DMC0_ODT	Output	B	none	none	L	VDD_DMC	Desc: DMC0 On-die termination Notes: No notes
DMC0_RAS	Output	B	none	none	L	VDD_DMC	Desc: DMC0 Row Address Strobe Notes: No notes
DMC0_RESET	Output	B	none	none	L	VDD_DMC	Desc: DMC0 Reset (DDR3 only) Notes: No notes
DMC0_RZQ	a	B	none	none	none	VDD_DMC	Desc: DMC0 External calibration resistor connection Notes: 34 Ohm 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
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
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
JTG_TCK	Input		PullUp	PullUp	none	VDD_EXT	Desc: JTAG Clock Notes: See note[2]
JTG_TDI	Input		PullUp	PullUp	none	VDD_EXT	Desc: JTAG Serial Data In Notes: See note[2]
JTG_TDO	Output	A	none	none	High-Z when JTG_TRST is low, not affected by SYS_HWRST	VDD_EXT	Desc: JTAG Serial Data Out Notes: No notes
JTG_TMS	InOut	A	PullUp	PullUp	none	VDD_EXT	Desc: JTAG Mode Select Notes: See note[2]
JTG_TRST	Input		PullDown	PullDown	none	VDD_EXT	Desc: JTAG Reset Notes: See note[2]
PA_00	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 0 Notes: See note[2]
PA_01	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 1 Notes: See note[2]

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PA_02	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 2 Notes: See note[2]
PA_03	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 3 Notes: See note[2]
PA_04	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 4 Notes: See note[2]
PA_05	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 5 Notes: See note[2]
PA_06	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 6 Notes: See note[2]
PA_07	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 7 Notes: See note[2]
PA_08	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 8 Notes: See note[2]
PA_09	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 9 Notes: See note[2]
PA_10	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 10 Notes: See note[2]
PA_11	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 11 Notes: See note[2]
PA_12	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 12 Notes: See note[2]
PA_13	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 13 Notes: See note[2]
PA_14	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 14 Notes: See note[2]
PA_15	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTA Position 15 Notes: See note[2]
PB_00	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 0 Notes: See note[2]
PB_01	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 1 Notes: See note[2]
PB_02	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 2 Notes: See note[2]
PB_03	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 3 Notes: See note[2]
PB_04	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 4 Notes: See note[2]
PB_05	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 5 Notes: See note[2]
PB_06	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 6 Notes: See note[2]
PB_07	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 7 Notes: See note[2]
PB_08	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 8 Notes: See note[2]
PB_09	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 9 Notes: See note[2]

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
PB_10	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 10 Notes: See note[2]
PB_11	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 11 Notes: See note[2]
PB_12	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 12 Notes: See note[2]
PB_13	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 13 Notes: See note[2]
PB_14	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 14 Notes: See note[2]
PB_15	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTB Position 15 Notes: See note[2]
PC_00	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 0 Notes: See note[2]
PC_01	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 1 Notes: See note[2]
PC_02	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 2 Notes: See note[2]
PC_03	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 3 Notes: See note[2]
PC_04	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 4 Notes: See note[2]
PC_05	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 5 Notes: See note[2]
PC_06	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 6 Notes: See note[2]
PC_07	InOut	A	Programmable PullUp/PullDown[1]	none	none	VDD_EXT	Desc: PORTC Position 7 (default is SYS_FAULT) Notes: External pull-down required to keep signal in de-asserted state
SYS_BMODE0	Input	NA	none	none	none	VDD_EXT	Desc: Boot Mode Control 0 Notes: No connection not allowed
SYS_BMODE1	Input	NA	none	none	none	VDD_EXT	Desc: Boot Mode Control 1 Notes: No connection not allowed
SYS_BMODE2	Input	NA	none	none	none	VDD_EXT	Desc: Boot Mode Control 2 Notes: No connection not allowed
SYS_CLKIN0	a	NA	none	none	none	VDD_REF	Desc: Clock/Crystal Input Notes: No connection not allowed
SYS_CLKOUT	a	A	none	none	none		Desc: Processor Clock Output Notes: No notes
$\overline{\text{SYS\_FAULT}}$	InOut	A	none	none	none		Desc: Active-Low Fault Output Notes: External pull-up required to keep signal in de-asserted state.
$\overline{\text{SYS\_HWRST}}$	Input	NA	none	none	none	VDD_EXT	Desc: Processor Hardware Reset Control Notes: No connection not allowed
$\overline{\text{SYS\_RESOUT}}$	Output	A	none	none	L	VDD_EXT	Desc: Reset Output Notes: No notes

Table 18. ADSP-2156x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Internal Termination	Reset Termination	Reset Drive	Power Domain	Description and Notes
SYS_XTAL0	a	NA	none	none	none	VDD_REF	Desc: Crystal Output Notes: Leave unconnected if an oscillator is used to provide SYS_CLKIN0.
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_REF	s		none	none	none		Desc: External Voltage Domain Notes: No notes

[1] Disable by default.
[2] When present, the internal PullUp/PullDown 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.

PRELIMINARY SPECIFICATIONS

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

PRELIMINARY OPERATING CONDITIONS

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DD_INT}	Internal (Core) Supply Voltage	400 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_DMC}[1]	DDR3L Controller Supply Voltage		1.283	1.350	1.418	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 × V_{DD_DMC}	0.50 × V_{DD_DMC}	0.51 × V_{DD_DMC}	V
$V_{DELTA\_EXT\_REF}$[4]	V_{DD_EXT} – V_{DD_REF}		–1.89		+1.89	V
V_{IH}[5]	High Level Input Voltage	V_{DD_EXT} = 3.47 V	2.0			V
$V_{IHCLKIN}$[2]	High Level Clock Input Voltage		0.65 × V_{DD_REF}		V_{DD_REF}	V
V_{IL}[5]	Low Level Input Voltage	V_{DD_EXT} = 3.13 V			0.8	V
$V_{ILCLKIN}$[2]	Low Level Clock Input Voltage		–0.30		+0.35 × V_{DD_REF}	V
V_{IL_DDR3L}[6]	Low Level Input Voltage	V_{DD_DMC} = 1.283 V			V_{DDR_VREF} – 0.175	V
V_{IL_DDR3}[6]	Low Level Input Voltage	V_{DD_DMC} = 1.425 V			V_{DDR_VREF} – 0.175	V
V_{IH_DDR3L}[6]	High Level Input Voltage	V_{DD_DMC} = 1.418 V	V_{DDR_VREF} + 0.175			V
V_{IH_DDR3}[6]	High Level Input Voltage	V_{DD_DMC} = 1.575 V	V_{DDR_VREF} + 0.175			V
T_J	Junction Temperature (Commercial Grade)	$T_{AMBIENT}$ = 0°C to 70°C	0		110	°C
T_J	Junction Temperature (Industrial Grade)	$T_{AMBIENT}$ = –40°C to +85°C	–40		TBD	°C
AUTOMOTIVE USE ONLY						
T_J	Junction Temperature (Automotive Grade)[7]	$T_{AMBIENT}$ = –40°C to +TBD°C	–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 8.
[5] Parameter value applies to all input and bidirectional pins except the DMC pins.
[6] This parameter applies to all DMC0 pins.
[7] Automotive application use profile only. Not supported for nonautomotive use. Contact Analog Devices for more information.

Preliminary Clock Related Operating Conditions

Table 19 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 19. Preliminary Clock Operating Conditions

Parameter		Conditions	Min	Typ	Max	Unit
f_{CCLK}	Core Clock Frequency	$f_{CCLK} = 2 \times f_{SYSCLK}$	400		1000	MHz
f_{SYSCLK}	SYSCLK Frequency		200		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	$f_{SYSCLK} \geq f_{SCLK1}$			250	MHz
f_{DCLK}	DDR3 Clock Frequency[1]	All combinations are supported except for: [$f_{CCLK} > 800$ MHz and $T_j < 0°C$ and f_{CCLK}:f_{DCLK} = 2:1]	300		667	MHz
f_{OCLK}	Output Clock Frequency[2]				125	MHz
$f_{SYS\_CLKOUTJ}$	SYS_CLKOUT Period Jitter[3, 4]			±2		%
$f_{LCLKTPROG}$	Programmed Link Port Transmit Clock				125	MHz
$f_{LCLKREXT}$	External Link Port Receive Clock[5, 6]	$f_{LCLKEXT} \leq f_{OCLK\_0}$			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[5, 6]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			62.5	MHz
$f_{SPTCLKEXT}$	External SPT Clock Transmitting Data or Frame Sync[5, 6]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			31.25	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data	Clock ratio 1:1			75	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data	Clock ratio 1:1			75	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data	Clock ratio 1:2			62.5	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data	Clock ratio 1:2			62.5	MHz
$f_{SPICLKEXT}$	External SPI Clock When Receiving Data[5, 6]	$f_{SPICLKEXT} \leq f_{CDU\_CLKO0}$			62.5	MHz
$f_{SPICLKEXT}$	External SPI Clock When Transmitting Data[5, 6]	$f_{SPICLKEXT} \leq f_{CDU\_CLKO0}$			45	MHz

[1] In order to ensure proper operation of the DDR3/3L, all the DDR3/3L guidelines must be strictly followed. See ADSP-2156x Board Design Guidelines for Dynamic Memory Controller (EE-418).

[2] f_{OCLK} must not exceed f_{SCLK0} when selected as SYS_CLKOUT.

[3] 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.

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

[5] 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.

[6] The peripheral external clock frequency must also be less than or equal to the frequency that clocks the peripheral.

Table 20. Phase-Locked Loop (PLL) Operating Conditions

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

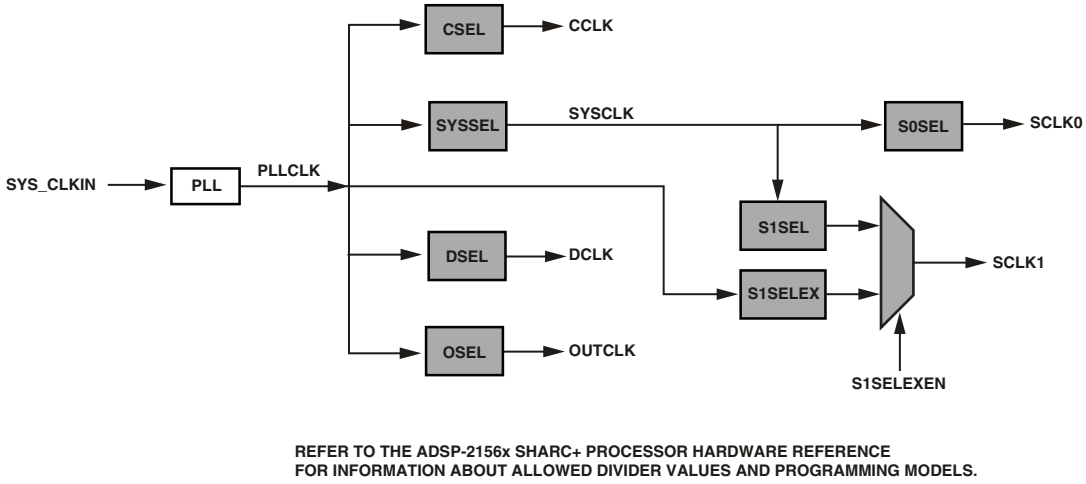


Figure 6. Clock Relationships and Divider Values

PRELIMINARY ELECTRICAL CHARACTERISTICS

Parameter		Conditions	Min	Typ	Max	Unit
V_{OH}[1]	High Level Output Voltage	At V_{DD_EXT} = minimum, I_{OH} = –1.0 mA[2]	2.4			V
V_{OL}[1]	Low Level Output Voltage	At V_{DD_EXT} = minimum, I_{OL} = 1.0 mA[2]			0.4	V
V_{OH_XTAL}	High Level Output Voltage	At V_{DD_EXT} = minimum, I_{OH} = –1.0 mA	1.26			V
V_{OL_XTAL}	Low Level Output Voltage	At V_{DD_EXT} = minimum, I_{OL} = 1.0 mA			0.45	V
V_{OH_DDR3L}[3]	High Level Output Voltage for DDR3L DS = 100 Ω	At V_{DD_DDR} = minimum, I_{OH} = –1.0 mA	0.963			V
V_{OL_DDR3L}[3]	Low Level Output Voltage for DDR3L DS = 100 Ω	At V_{DD_DDR} = minimum, I_{OL} = 1.0 mA			0.32	V
V_{OH_DDR3}[4]	High Level Output Voltage for DDR3 DS = 100 Ω	At V_{DD_DDR} = minimum, I_{OH} = –1.0 mA	1.105			V
V_{OL_DDR3}[4]	Low Level Output Voltage for DDR3 DS = 100 Ω	At V_{DD_DDR} = minimum, I_{OL} = 1.0 mA			0.32	V
I_{IH}[5, 6]	High Level Input Current	At V_{DD_EXT} = maximum, V_{IN} = V_{DD_EXT} maximum			10	µA
I_{IL}[5]	Low Level Input Current	At V_{DD_EXT} = maximum, V_{IN} = 0 V			10	µA
I_{IL_PU}[6]	Low Level Input Current Pull-Up	At V_{DD_EXT} = maximum, V_{IN} = 0 V			200	µA
I_{IH_PD}[7]	High Level Input Current Pull-Down	At V_{DD_EXT} = maximum, V_{IN} = V_{DD_EXT} maximum			200	µA
I_{OZH}[8]	Three-State Leakage Current	At V_{DD_EXT}/V_{DD_DDR} = maximum, V_{IN} = V_{DD_EXT}/V_{DD_DDR} maximum			10	µA
I_{OZL}[8]	Three-State Leakage Current	at V_{DD_EXT}/V_{DD_DDR} = maximum, V_{IN} = 0 V			10	µA
C_{IN}[9]	Input Capacitance	T_{CASE} = 25°C			5	pF

[1] Applies to all output and bidirectional pins except DMC.
[2] See the Output Drive Currents (TBD) section for typical drive current capabilities.
[3] Applies to all DMC output and bidirectional signals in DDR3L mode.
[4] Applies to all DMC output and bidirectional signals in DDR3 mode.
[5] Applies to input pins SYS_BMODE2-0, SYS_CLKIN, $\overline{\text{SYS\_HWRST}}$, JTG_TDI, and JTG_TMS.
[6] Applies to input pins with internal pull-ups including JTG_TDI, JTG_TMS, and JTG_TCK.
[7] Applies to signals JTAG_TRST.
[8] Applies to signals PA15-0, PB15-0, PC7-0, DAI0_PINx, DAI1_PINx, DMC0_DQx, DMC0_LDQS, DMC0_UDQS, $\overline{\text{DMC0\_LDQS}}$, $\overline{\text{DMC0\_UDQS}}$, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, and JTG_TDO.
[9] Applies to all signal pins.

ABSOLUTE MAXIMUM RATINGS

Stresses at or above those listed in Table 21 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 21. 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
$V_{DD\_EXT} – V_{DD\_REF}$ ($V_{DELTA\_EXT\_REF}$)	–1.89 V to +1.89 V
DDR3 Controller Supply Voltage (V_{DD_DMC})	–0.3 V to +1.60 V
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
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 the that power domain ± 0.2 V.
[2] Applies to 100% transient duty cycle.
[3] Applies to TWI_SCL and TWI_SDA.

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

Power-Up Reset Timing

Table 22 and Figure 7 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 7, the $V_{DD\_SUPPLIES}$ are V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, and V_{DD_REF}. The $V_{DELTA\_EXT\_REF}$ specification must be met at all times, including during power-up reset and when powering down the device (Figure 8).

Table 22. Power-Up Reset Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirement				
$t_{RST\_IN\_PWR}$	$\overline{SYS\_HWRST}$ Deasserted after $V_{DD\_SUPPLIES}$ (V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_REF}) and SYS_CLKIN0 are Stable and Within Specification	$11 \times t_{CKIN}$		ns

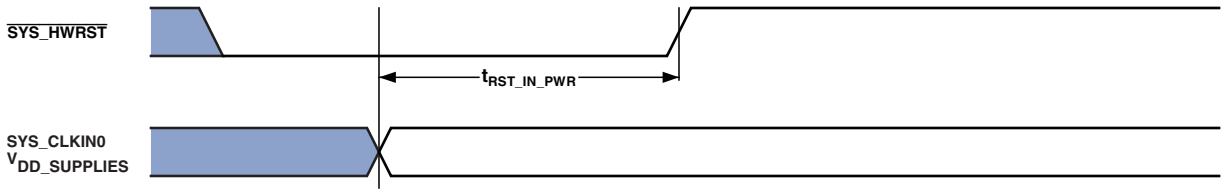


Figure 7. Power-Up Reset Timing

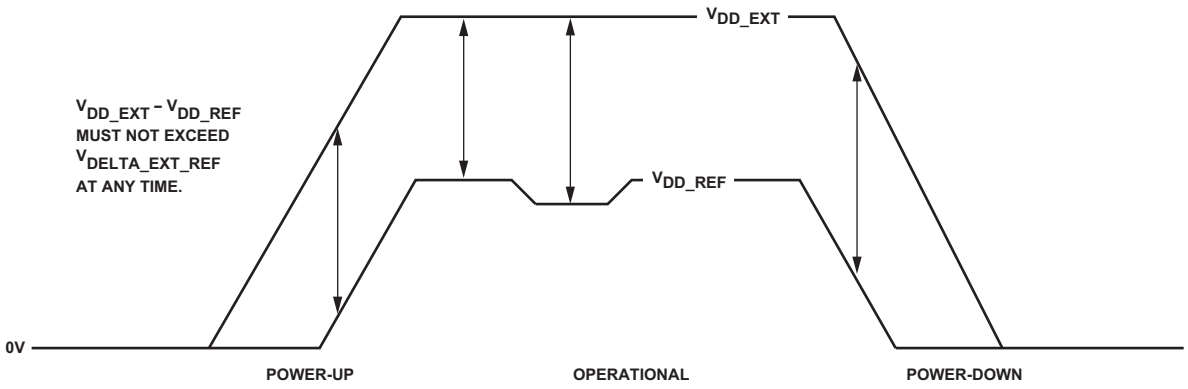


Figure 8. Power-Up and Power-Down Voltage Delta Requirement

Clock and Reset Timing

Table 23 and Figure 9 describe clock and reset operations related to the CGU and RCU. Per the CCLK, SYSCLK, SCLK, DCLK, and OCLK timing specifications in Table 19 (Preliminary 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 23. Clock and Reset Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
f_{CKIN}	SYS_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 9) equals $1/f_{CKIN}$.
[3] Applies after power-up sequence is complete. See Table 22 and Figure 7 for power-up reset timing.

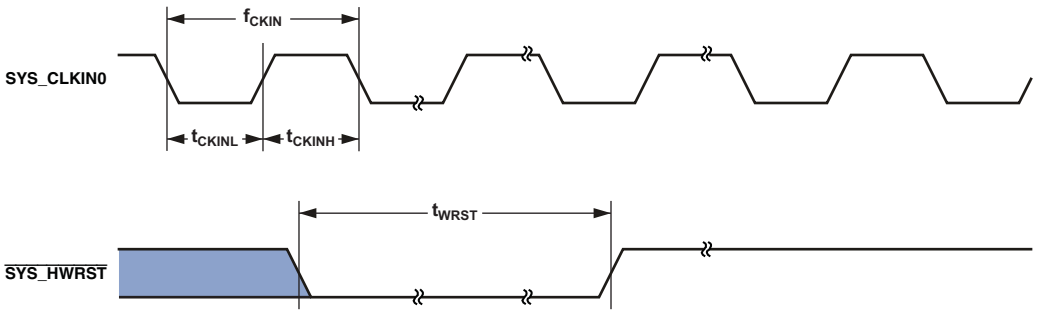


Figure 9. Clock and Reset Timing

Link Ports (LPs)

In LP receive mode, the LP clock is supplied externally and is called $f_{LCLKREXT}$, therefore the period can be represented by

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

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

$$f_{LCLKTPROG} = \frac{f_{SCLK0}}{(VALUE \times 2)}$$

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

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

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

Table 24. LPs—Receive[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
$f_{LCLKREXT}$	LPx_CLK Frequency		125	MHz
t_{SLDCL}	Data Setup Before LPx_CLK Low	1.5		ns
t_{HLDCL}	Data Hold After LPx_CLK Low	1.4		ns
t_{LCLKEW}	LPx_CLK Period[2]	$t_{LCLKREXT}$ − 1		ns
$t_{LCLKRWL}$	LPx_CLK Width Low[2]	0.5 × $t_{LCLKREXT}$		ns
$t_{LCLKRWH}$	LPx_CLK Width High[2]	0.5 × $t_{LCLKREXT}$		ns
Switching Characteristic				
t_{DLALC}	LPx_ACK Low Delay After LPx_CLK Low[3]	1.5 × t_{SCLK0} + 4	2.5 × t_{SCLK0} + 12	ns

[1] Specifications apply to LP0 and LP1.
[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external LPx_CLK. For the external LPx_CLK ideal maximum frequency, see the $f_{LCLKTEXT}$ specification in the Table 19.
[3] LPx_ACK goes low with t_{DLALC} relative to rise of LPx_CLK after first byte, but does not go low if the link buffer of the receiver is not about to fill.

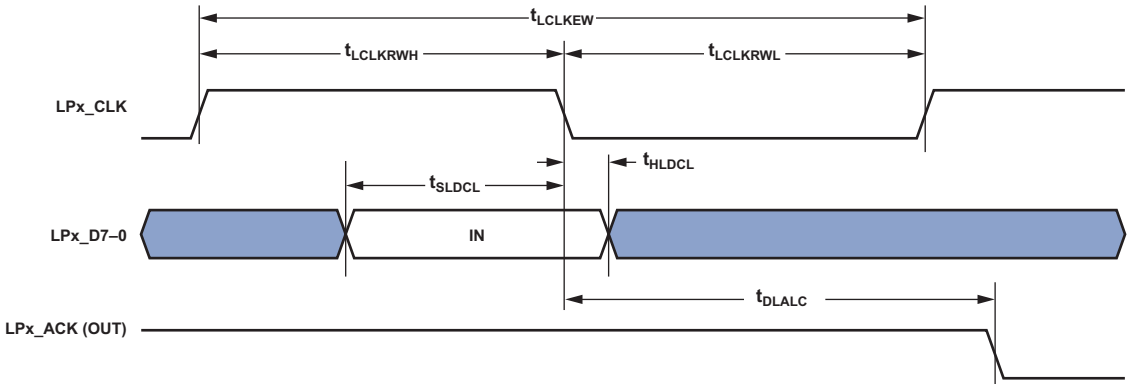
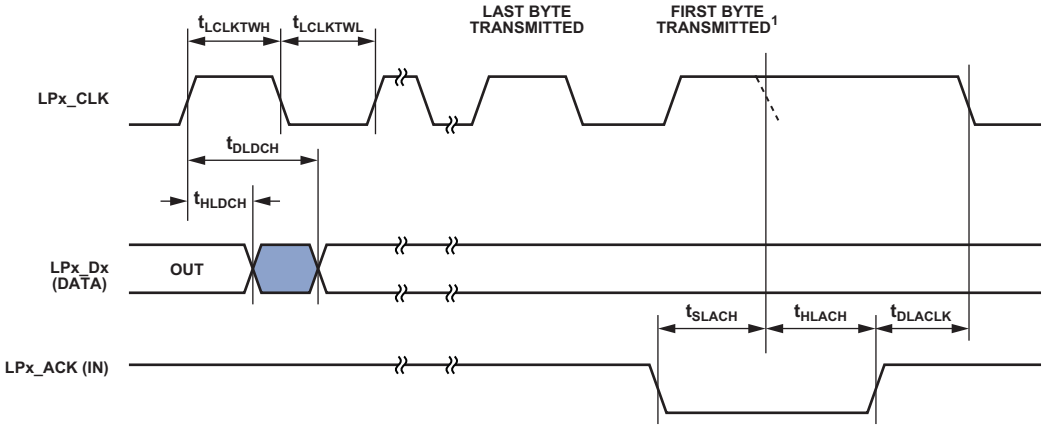


Figure 10. LPs—Receive

Table 25. LPs—Transmit[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SLACH}	LPx_ACK Setup Before LPx_CLK Low	$2 \times t_{SCLK0} + 13.5$		ns
t_{HLACH}	LPx_ACK Hold After LPx_CLK Low	–2.5		ns
Switching Characteristics				
t_{DLDCH}	Data Delay After LPx_CLK High		2.23	ns
t_{HLDCH}	Data Hold After LPx_CLK High	–1.04		ns
$t_{LCLKTWL}$[2]	LPx_CLK Width Low	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
$t_{LCLKTWH}$[2]	LPx_CLK Width High	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
t_{LCLKTW}[2]	LPx_CLK Period	$N \times t_{LCLKTPROG} - 0.5$		ns
t_{DLACLK}	LPx_CLK Low Delay After LPx_ACK High	$t_{SCLK0} + 4$	$2 \times t_{SCLK0} + 1 \times t_{LPCLK} + 10$	ns

[1] Specifications apply to LP0 and LP1.
[2] See Table 19 for details on the minimum period that can be programmed for $t_{LCLKTPROG}$.



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

Figure 11. LPs—Transmit

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 12, either the rising edge or the falling edge of SPTx_CLK (external or internal) can be used as the active sampling edge.

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

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

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

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

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

Table 26. SPORTs—External Clock[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SFSE}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[2]	2		ns
t_{HFSE}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[2]	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]		11	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]		11	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 external SPTx_CLK. For the external SPTx_CLK ideal maximum frequency, see the $f_{SPTCLKEXT}$ specification in Table 19.
[4] Referenced to drive edge.

Table 27. SPORTs—Internal Clock[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SFSI}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[2]	12		ns
t_{HFSI}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[2]	–0.5		ns
t_{SDRI}	Receive Data Setup Before SPTx_CLK[2]	3.4		ns
t_{HDRI}	Receive Data Hold After SPTx_CLK[2]	2		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]	$0.5 \times t_{SPTCLKPROG} - 2$		ns
t_{SPTCLK}	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 19 for details on the minimum period that can be programmed for $t_{SPTCLKPROG}$.

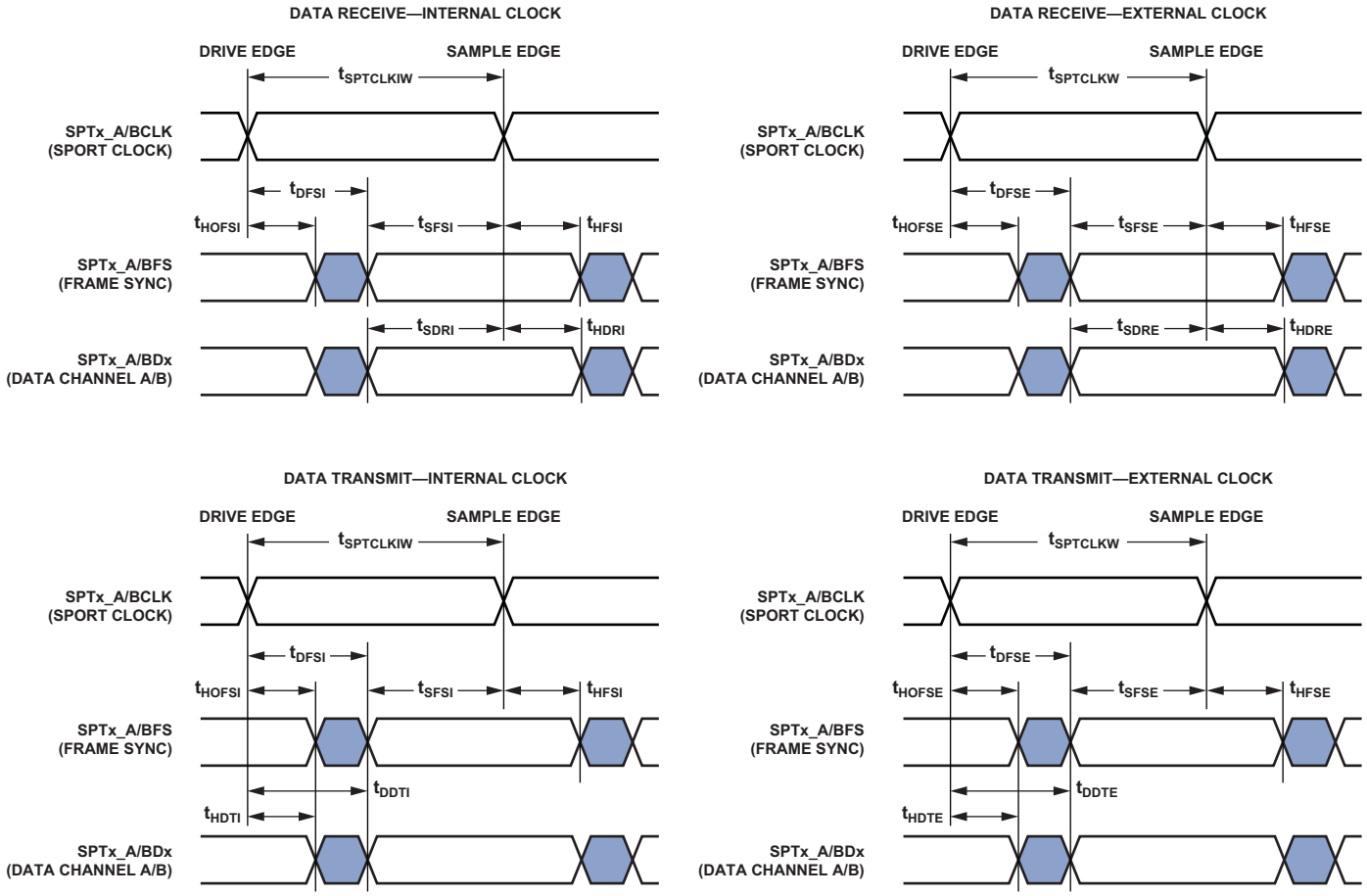


Figure 12. SPORTs

Table 28. SPORTs—Enable and Three-State[1]

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

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

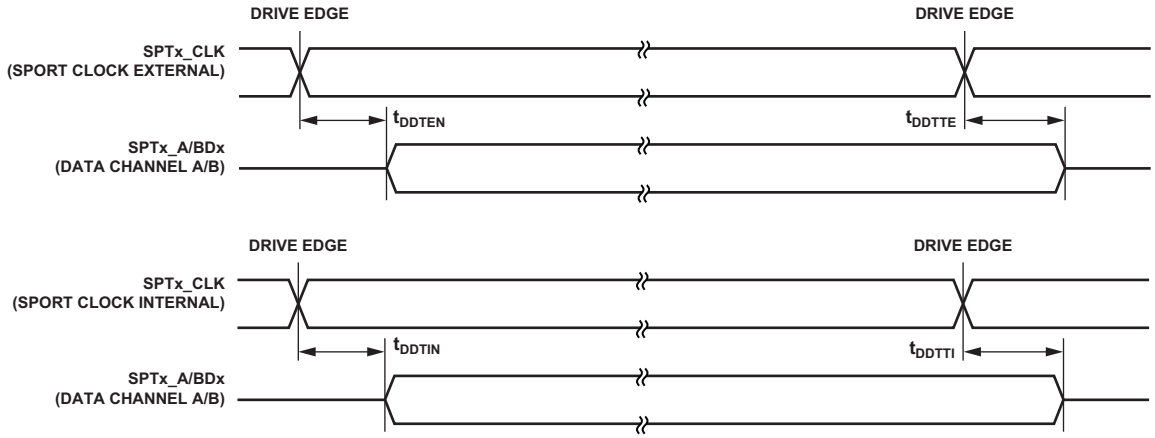


Figure 13. SPORTs—Enable and Three-State

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

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

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

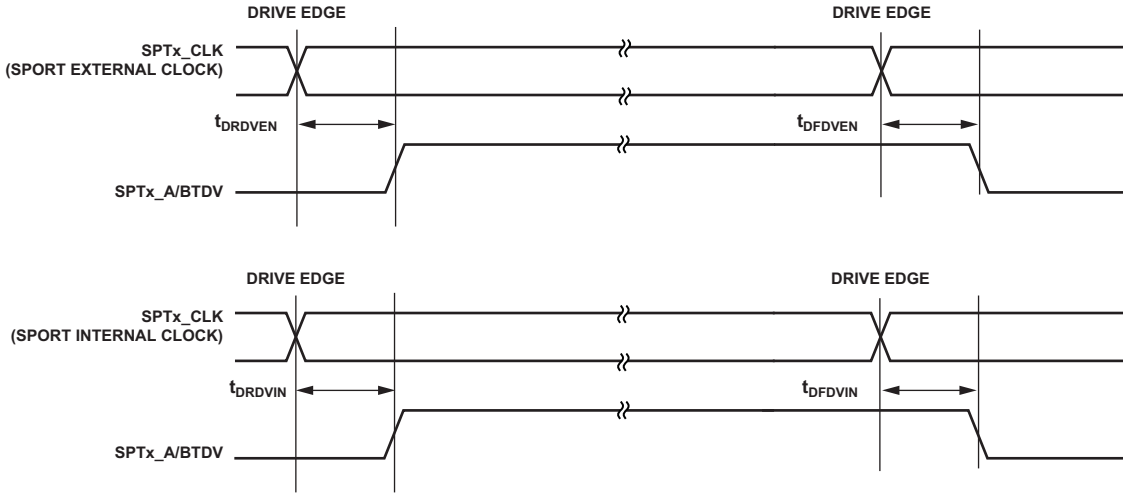


Figure 14. SPORTs—Transmit Data Valid Internal and External Clock

Table 30. SPORTs—External Late Frame Sync[1]

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

[1] Specifications apply to all 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.

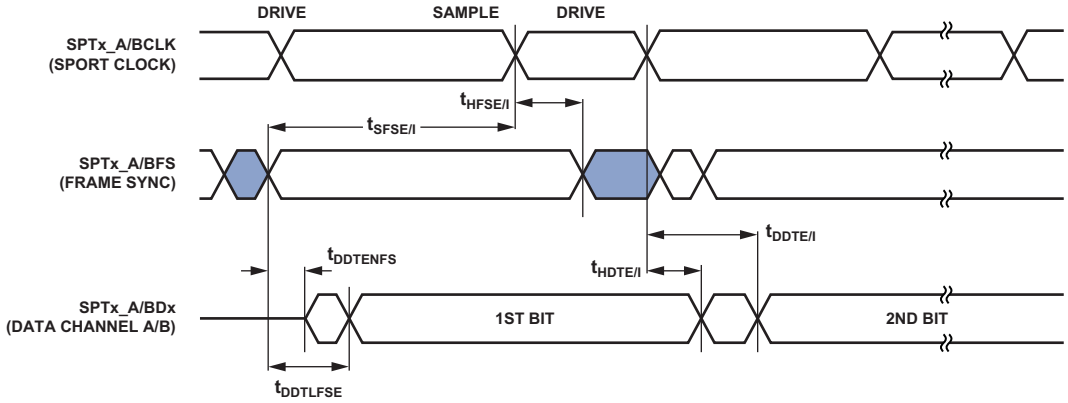


Figure 15. External Late Frame Sync

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 31 are valid at the DAI0_PINx pins.

Table 31. ASRC, Serial Input Port

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

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

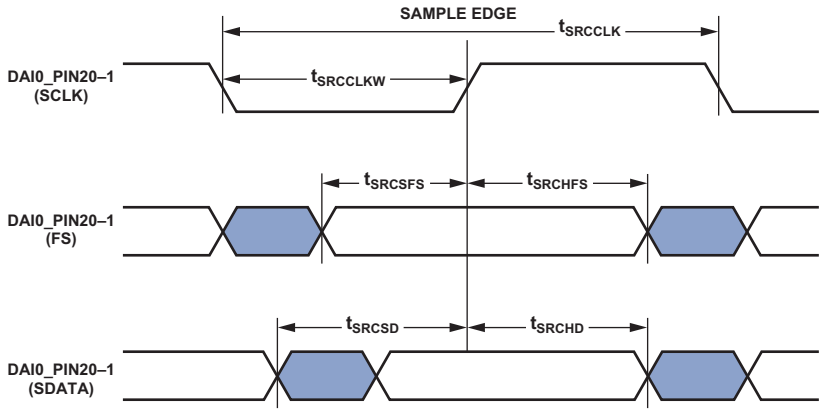


Figure 16. ASRC Serial Input Port Timing

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 SCLK on the output port. The serial data output has a hold time and delay specification with regard to serial clock. The serial clock rising edge is the sampling edge, and the falling edge is the drive edge.

Table 32. ASRC, Serial Output Port

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

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

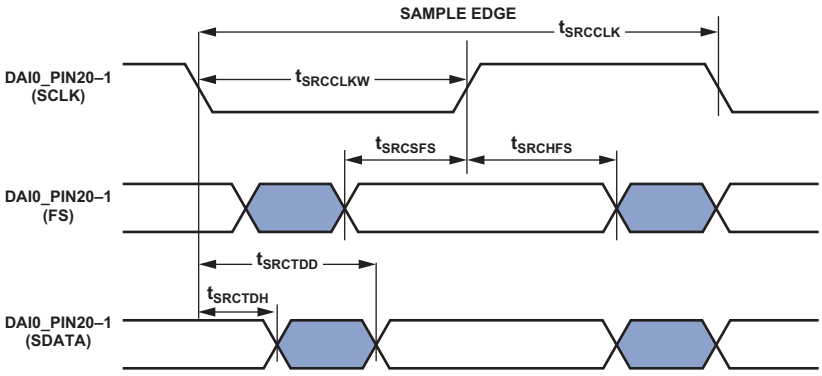


Figure 17. ASRC Serial Output Port Timing

SPI Port—Master Timing

SPI0, SPI1, and SPI2

Table 33 and Figure 18 describe the SPI port master operations.

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

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

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

Note that

- In dual-mode data transmit, the SPIx_MISO signal is also an output.
- In quad-mode data transmit, the SPIx_MISO, SPIx_D2, and SPIx_D3 signals are also outputs.
- In dual-mode data receive, the SPIx_MOSI signal is also an input.
- In quad-mode data receive, the SPIx_MOSI, SPIx_D2, and SPIx_D3 signals are also inputs.
- Quad mode is supported by SPI1 and SPI2.
- CPHA is a configuration bit in the SPI_CTL register.

Table 33. SPI Port—Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	3.5		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 19 for details on the minimum period that can be programmed for $t_{SPICLKPROG}$.
[4] Applies to sequential mode with STOP ≥ 1.

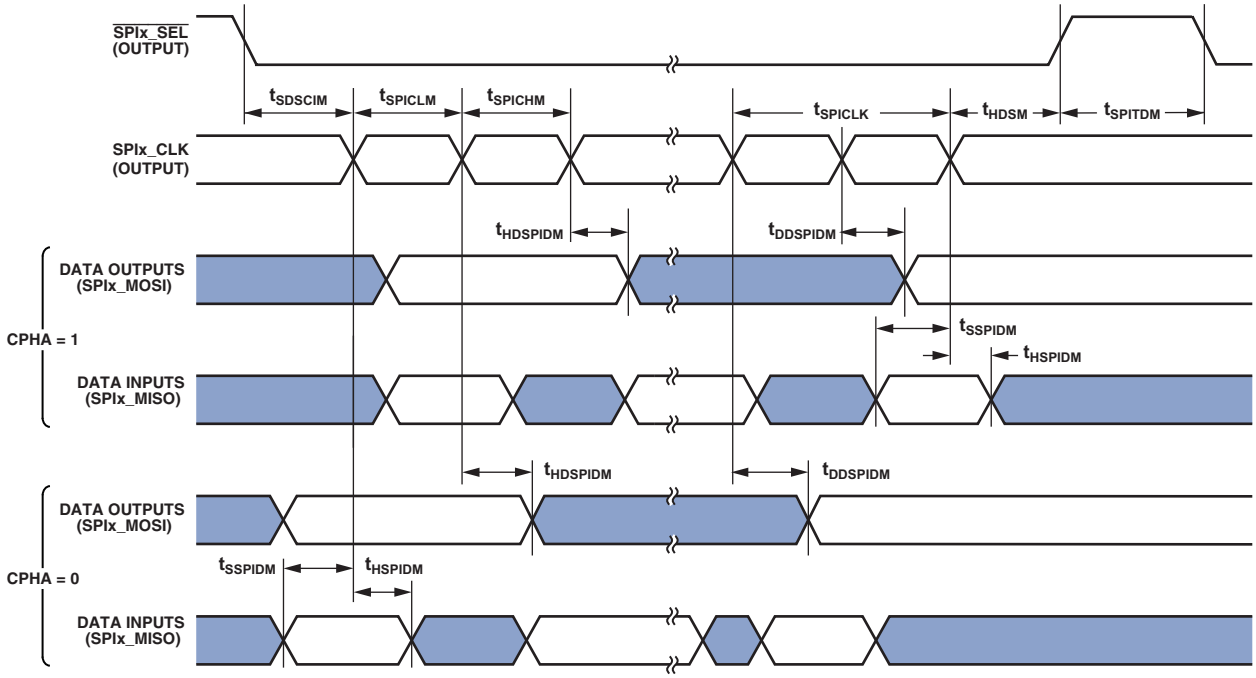


Figure 18. SPI Port—Master Timing

SPI Port—Slave Timing

SPI0, SPI1, and SPI2

Table 34 and Figure 19 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 34. SPI Port—Slave Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	0.5 × $t_{SPICLKEXT}$ – 1.5		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	0.5 × $t_{SPICLKEXT}$ – 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 19.

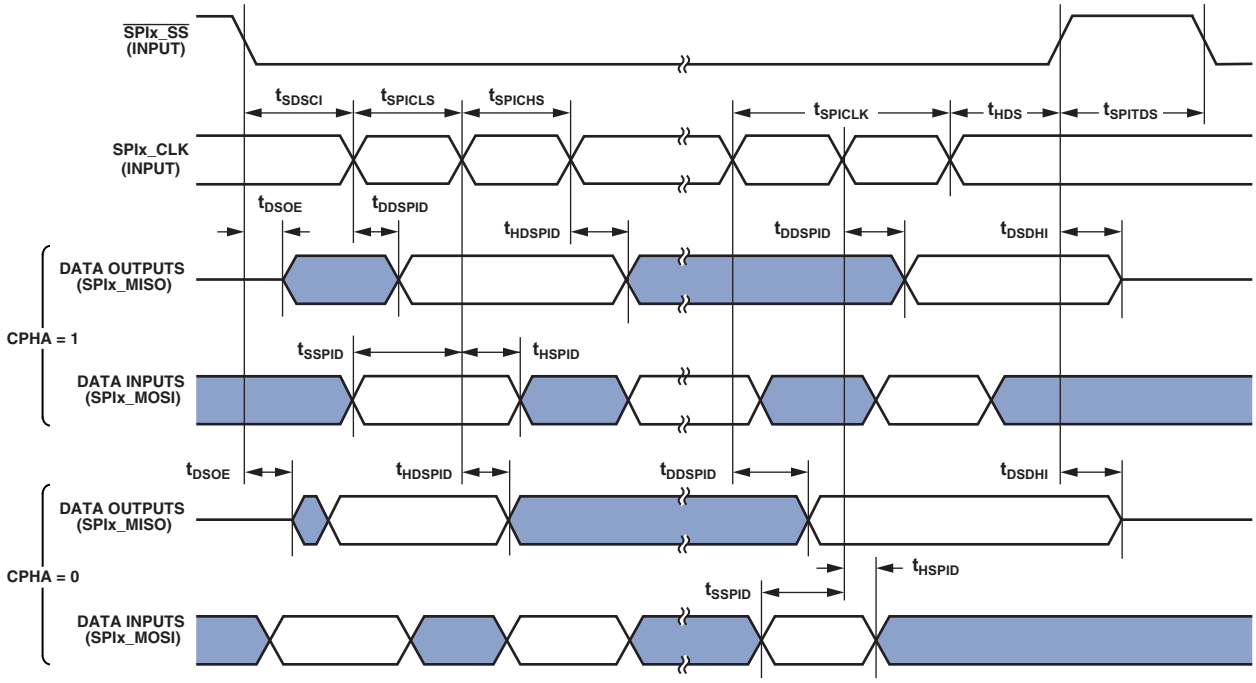


Figure 19. SPI Port—Slave Timing

SPI Port—SPIx_RDY Slave Timing

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

Table 35. SPI Port—SPIx_RDY Slave Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter	**Conditions**	**Min**	**Max**	**Unit**
Switching Characteristic				
$t_{DSPISCKRDYS}$ SPIx_RDY Deassertion From Last Valid Input SPIx_CLK Edge	FCCH = 0	$3 \times t_{SCLK1}$	$4 \times t_{SCLK1} + 10$	ns
	FCCH = 1	$4 \times t_{SCLK1}$	$5 \times t_{SCLK1} + 10$	ns

[1] All specifications apply to all three SPIs.

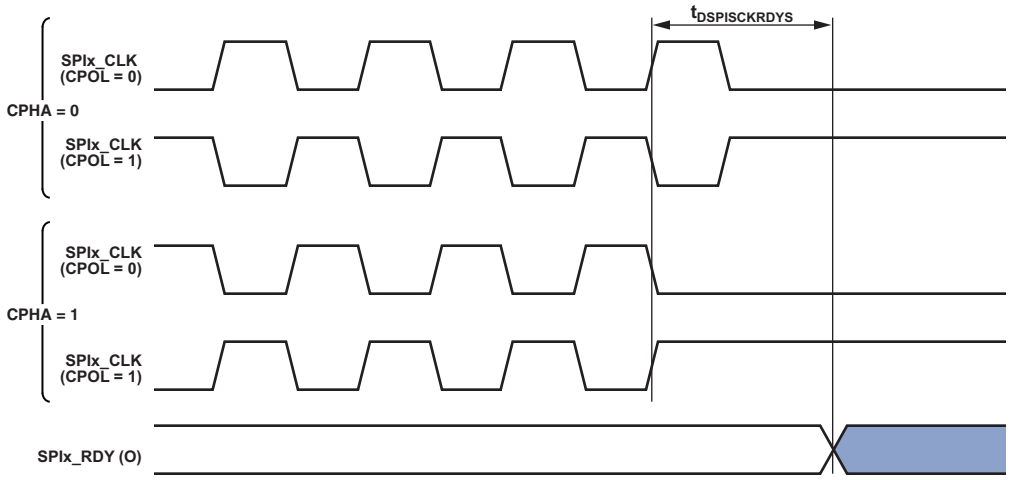


Figure 20. SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode

SPI Port—Open Drain Mode (ODM) Timing

In Figure 21 and Figure 22, 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 36. SPI Port—ODM Master Mode Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
$t_{HDSPIODMM}$	SPIx_CLK Edge to High Impedance From Data Out Valid	–1.5		ns
$t_{DDSPIODMM}$	SPIx_CLK Edge to Data Out Valid From High Impedance		6	ns

[1] All specifications apply to all three SPIs.

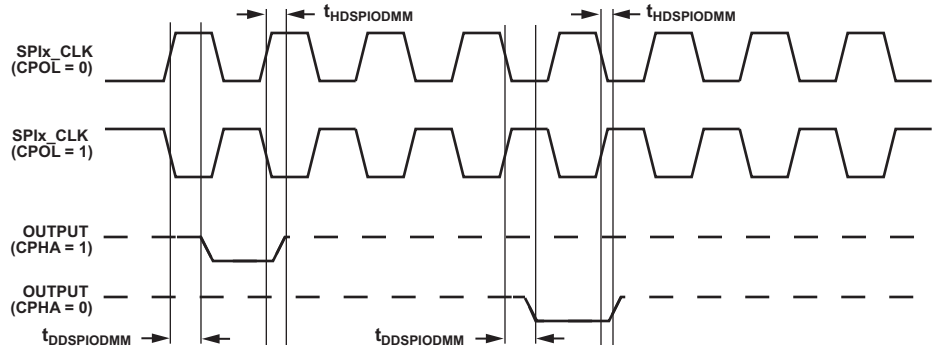


Figure 21. ODM Master Mode

Table 37. SPI Port—ODM Slave Mode[1]

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

[1] All specifications apply to all three SPIs.

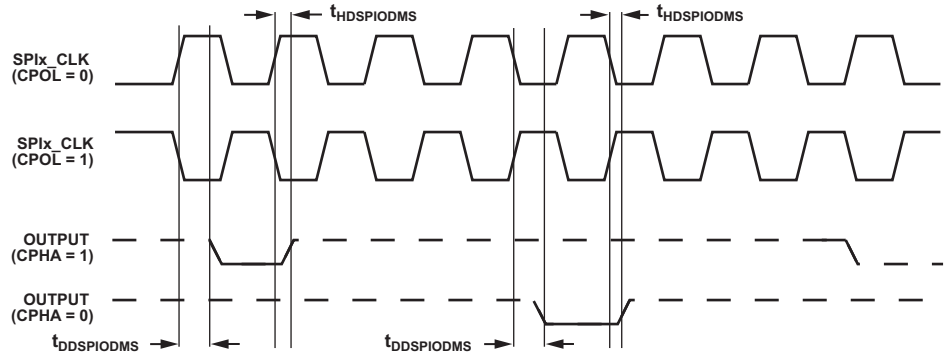


Figure 22. ODM Slave Mode

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, while LEADX, LAGX, and STOP are configuration bits in the SPIx_DLY register.

Table 38. SPI Port—SPIx_RDY Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.					
Parameter		**Conditions**	**Min**	**Max**	**Unit**
Timing Requirement					
$t_{SRDYSCKM}$	Setup Time for SPIx_RDY Deassertion Before Last Valid Data SPIx_CLK Edge		$(2 + 2 \times BAUD^{2}) \times t_{SCLK1} + 11$		ns
Switching Characteristic					
$t_{DRDYSCKM}$[3]	Assertion of SPIx_RDY to First SPIx_CLK Edge of Next Transfer	BAUD = 0, CPHA = 0	$4.5 \times t_{SCLK1}$	$5.5 \times t_{SCLK1} + 11$	ns
		BAUD = 0, CPHA = 1	$4 \times t_{SCLK1}$	$5 \times t_{SCLK1} + 11$	ns
		BAUD > 0, CPHA = 0	$(1 + 1.5 \times BAUD^{2}) \times t_{SCLK1}$	$(2 + 2.5 \times BAUD^{2}) \times t_{SCLK1} + 11$	ns
		BAUD > 0, CPHA = 1	$(1 + 1 \times BAUD^{2}) \times t_{SCLK1}$	$(2 + 2 \times BAUD^{2}) \times t_{SCLK1} + 11$	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.

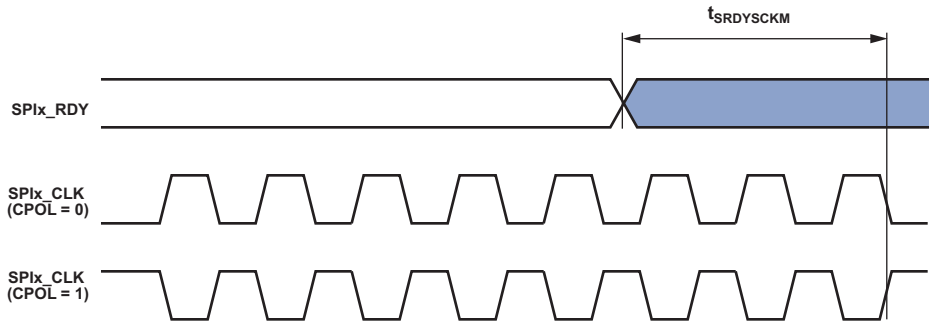


Figure 23. SPIx_RDY Setup Before SPIx_CLK

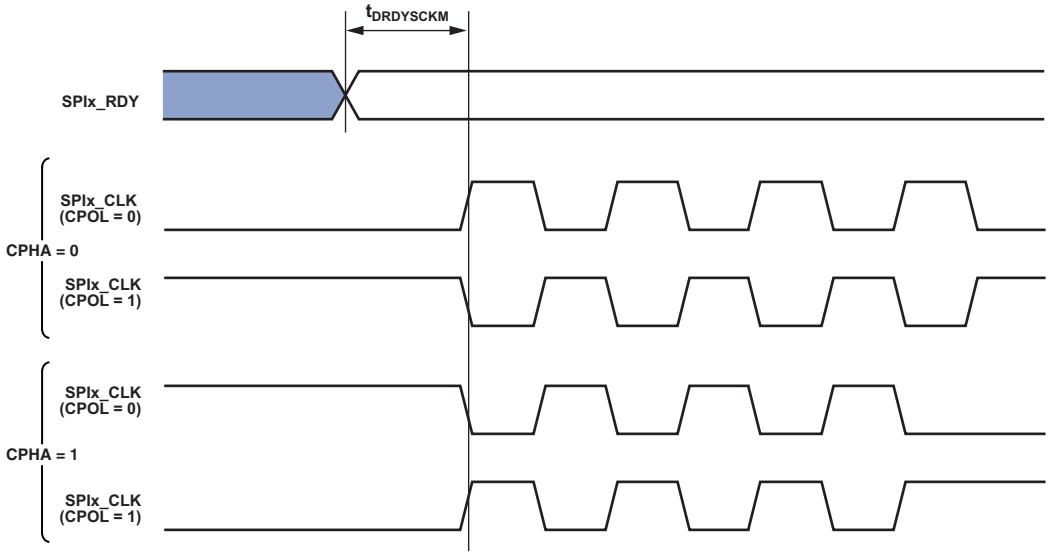


Figure 24. SPIx_CLK Switching Diagram after SPIx_RDY Assertion

OSPI Port—Master Timing

OSPI0

Table 39 and Figure 25 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 where PRG_MBD = master mode BAUD rate divisor.

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

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

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 39. OSPI Port—Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SSPIDM}	Data Input Valid to OSPI0_CLK Sampling Edge (Data Input Setup) [2]	t_{SYSCLK} + 2		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 OSPI only.
[2] t_{SSPIDM} and t_{HSPIDM} specifications are valid only for default RDCR (read data capture register) settings.
[3] PRG_CSSOT = chip select start of transfer (defined in the device delay register bit field [7:0]).
[4] See Table 19 for details on the minimum period that can be programmed for $t_{SPICLKPROG}$.
[5] PRG_CSEOT = chip select end of transfer (defined in the device delay register bit field [15:8]).
[6] Mode = clock phase and clock polarity bits (defined in the octal SPI configuration register bit field [2:1]).
[7] PRG_WRHLD = transmit delay to improve output hold (defined in the read data capture register bit field [19:16]).

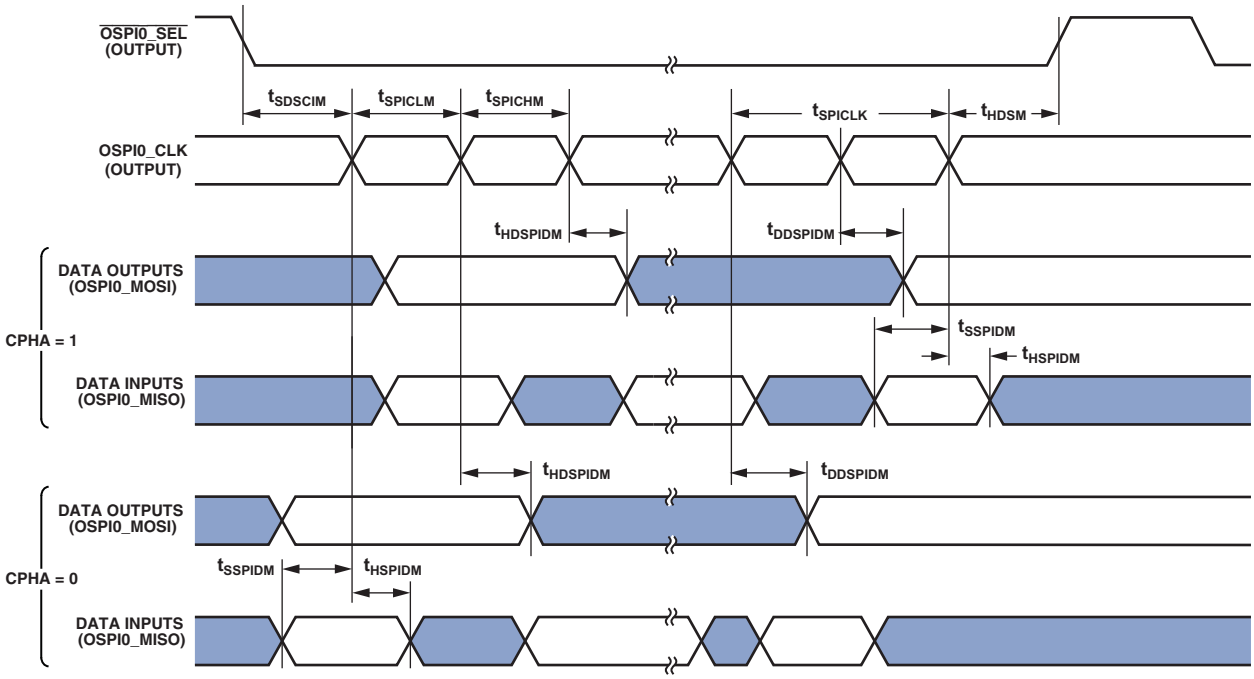


Figure 25. OSPI Port—Master Timing

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 40. PCG (Direct Pin Routing)

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{PCGIP}	Input Clock Period	$t_{SCLK} \times 2$		ns
t_{STRIG}	PCG Trigger Setup Before Falling Edge of PCG Input Clock	4.5		ns
t_{HTRIG}	PCG Trigger Hold After Falling Edge of PCG Input Clock	3		ns
Switching Characteristics				
t_{DPCGIO}	PCG Output Clock and Frame Sync Active Edge Delay After PCG Input Clock	2	10	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-2156x SHARC+ Processor Hardware Reference.
[2] Normal mode of operation.

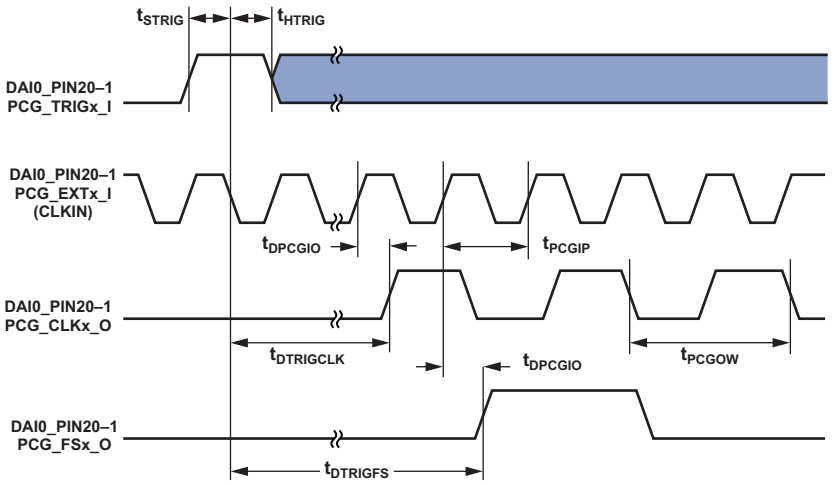


Figure 26. PCG (Direct Pin Routing)

General-Purpose IO Port Timing

Table 41 and Figure 27 describe I/O timing, related to the general-purpose ports (PORT).

Table 41. General-Purpose Port Timing

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

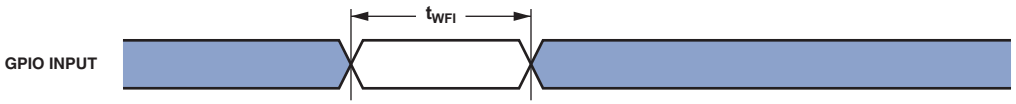


Figure 27. General-Purpose Port Timing

General-Purpose I/O Timer Cycle Timing

Table 42, Table 43, and Figure 28 describe timer expired operations related to the general-purpose timer (TIMER). The input signal is asynchronous in Width Capture Mode and External Clock Mode and has an absolute maximum input frequency of $f_{SCLK}/4$ MHz. The Width Value value is the timer period assigned in the TMx_TMRn_WIDTH register and can range from 1 to $2^{32} - 1$. When externally generated, the TMx_CLK clock is called $f_{TMRCLKEXT}$:

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

Table 42. Timer Cycle Timing—Internal Mode

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

[1] The minimum pulse width applies for 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 43. Timer Cycle Timing—External Mode

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

[1] The minimum pulse width applies for 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 19.
[3] WIDTH refers to the value in the TMRx_WIDTH register (it can vary from 1 to $2^{32} - 1$).

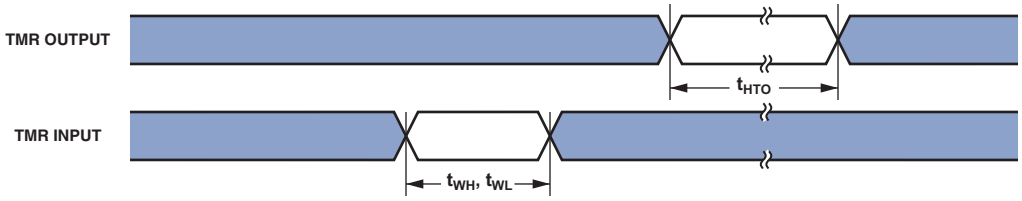


Figure 28. Timer Cycle Timing

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

Table 44 and Figure 29 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 44. DAI Pin to DAI Pin Routing

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

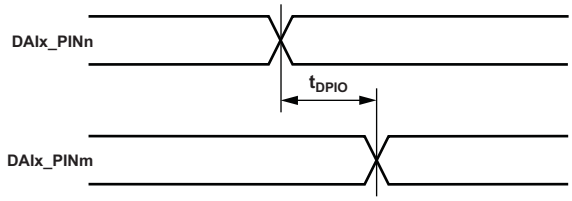


Figure 29. DAI Pin to DAI Pin Direct Routing

Up/Down Counter/Rotary Encoder Timing

Table 45 and Figure 30 describe timing related to the general-purpose counter (CNT).

Table 45. Up/Down Counter/Rotary Encoder Timing

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

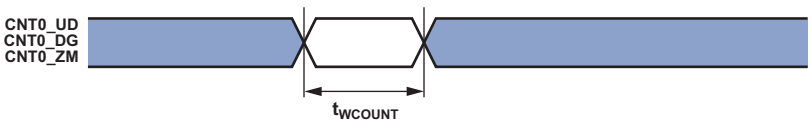


Figure 30. Up/Down Counter/Rotary Encoder Timing

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

The UART ports receive and transmit operations are described in the ADSP-2156x SHARC+ Processor Hardware Reference.

Sony/Philips Digital Interface (S/PDIF) Transmitter

Serial data input to the S/PDIF transmitter can be formatted as left justified, I^2S, or right justified with word widths of 16, 18, 20, or 24 bits. The following sections provide timing for the transmitter.

S/PDIF Transmitter Serial Input Waveforms

Table 46 and Figure 31 show the right justified mode. Frame sync is high for the left channel and low for the right channel. Data is valid on the rising edge of serial clock. The MSB is delayed the minimum in 24-bit output mode or the maximum in 16-bit output mode from a frame sync transition, so that when there are 64 serial clock periods per frame sync period, the LSB of the data is right justified to the next frame sync transition.

Table 46. S/PDIF Transmitter Right Justified Mode

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

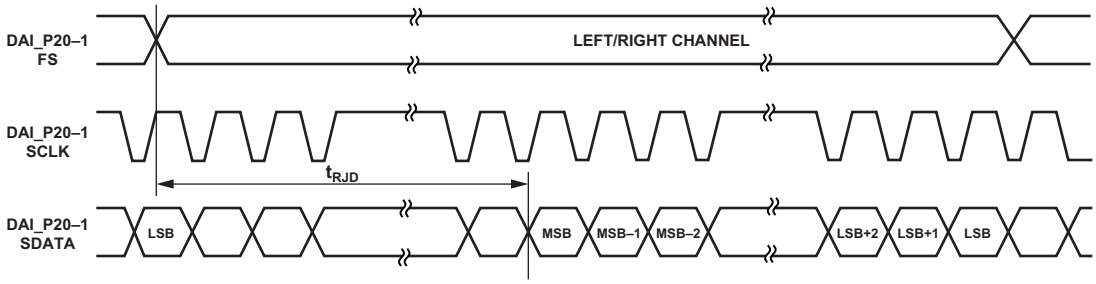


Figure 31. Right Justified Mode

Table 47 and Figure 32 show the default I²S justified mode. The frame sync is low for the left channel and high for the right channel. Data is valid on the rising edge of serial clock. The MSB is left justified to the frame sync transition but with a delay.

Table 47. S/PDIF Transmitter I²S Mode

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

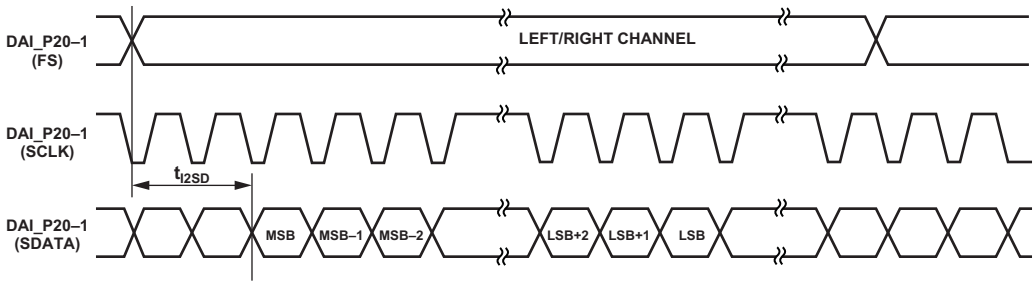


Figure 32. I²S Justified Mode

Table 48 and Figure 33 show the left justified mode. The frame sync is high for the left channel and low for the right channel. Data is valid on the rising edge of serial clock. The MSB is left justified to the frame sync transition with no delay.

Table 48. S/PDIF Transmitter Left Justified Mode

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

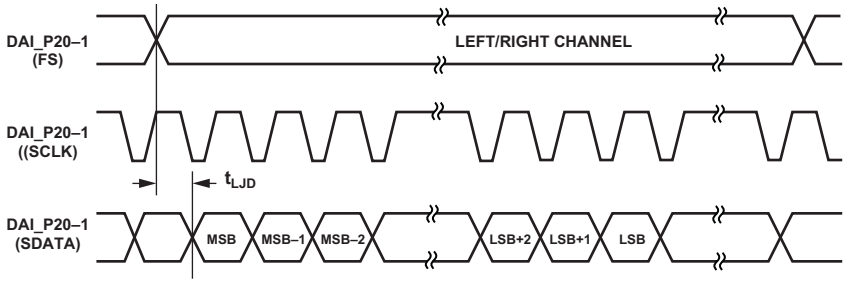


Figure 33. Left Justified Mode

S/PDIF Transmitter Input Data Timing

The timing requirements for the S/PDIF transmitter are given in Table 49. Input signals are routed to the DAI0_PINx pins using the SRU. Therefore, the timing specifications provided below are valid at the DAI0_PINx pins.

Table 49. S/PDIF Transmitter Input Data Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SISFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	3.4		ns
t_{SIHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	3		ns
t_{SISD}[1]	Data Setup Before Serial Clock Rising Edge	3		ns
t_{SIHD}[1]	Data Hold After Serial Clock Rising Edge	3		ns
$t_{SITXCLKW}$	Transmit Clock Width	9		ns
$t_{SITXCLK}$	Transmit Clock Period	20		ns
$t_{SISCLKW}$	Clock Width	36		ns
t_{SISCLK}	Clock Period	80		ns

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

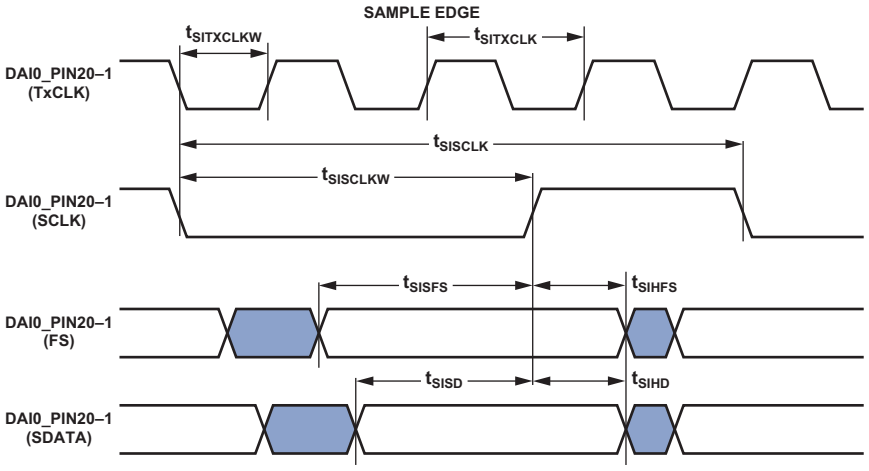


Figure 34. S/PDIF Transmitter Input Timing

Oversampling Clock (TxCLK) Switching Characteristics

The S/PDIF transmitter requires an oversampling clock input. This high frequency clock (TxCLK) input is divided down to generate the internal biphase clock.

Table 50. Oversampling Clock (TxCLK) Switching Characteristics

All specifications are based on simulation data and are subject to change without notice.			
Parameter		**Max**	**Unit**
Switching Characteristics			
f_{TXCLK_384}	Frequency for TxCLK = 384 × Frame Sync	Oversampling ratio × frame sync ≤ $1/t_{SITXCLK}$	MHz
f_{TXCLK_256}	Frequency for TxCLK = 256 × Frame Sync	49.2	MHz
f_{FS}	Frame Rate (FS)	192.0	kHz

S/PDIF Receiver

The following section describes timing as it relates to the S/PDIF receiver.

Internal Digital PLL Mode

In the internal digital PLL mode, the internal digital PLL generates the 512 × FS clock.

Table 51. S/PDIF Receiver Internal Digital PLL Mode Timing

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

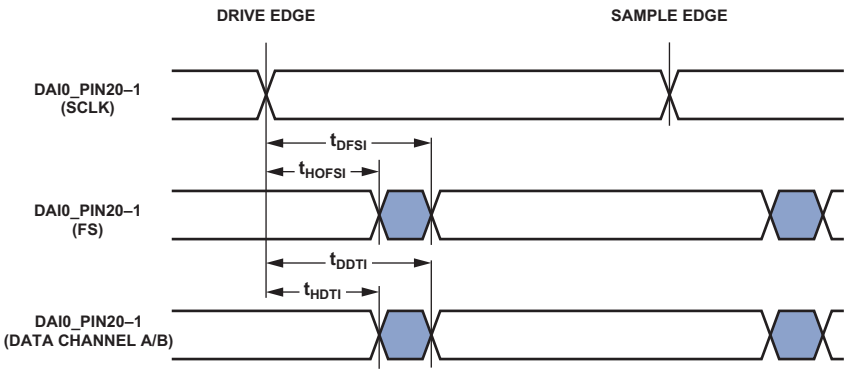


Figure 35. S/PDIF Receiver Internal Digital PLL Mode Timing

MediaLB (MLB)

All the numbers shown in Table 52 are applicable for all MLB speed modes (1024 FS, 512 FS, and 256 FS) for the 3-pin protocol, unless otherwise specified. Refer to the *Media Local Bus Specification version 4.2* for more details.

Table 52. 3-Pin MLB Interface Specifications

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

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

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

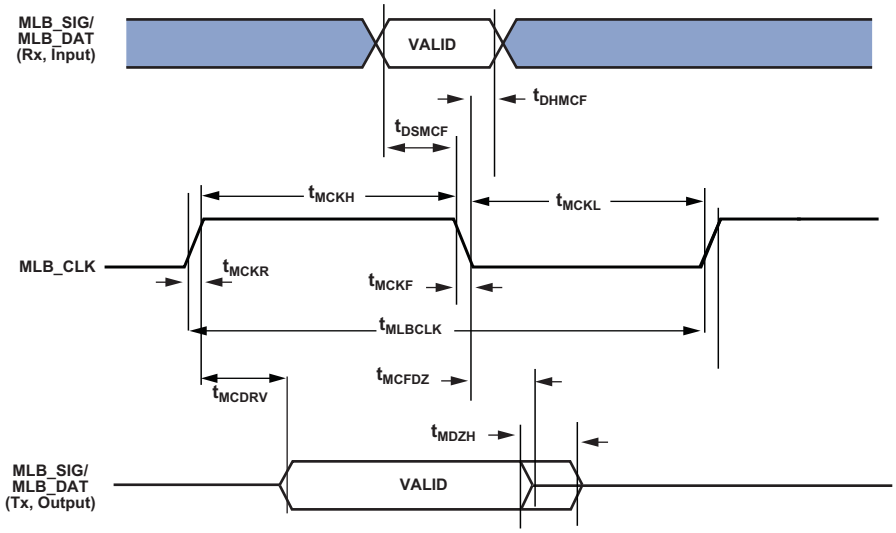


Figure 36. MLB Timing (3-Pin Interface)

Program Trace Macrocell (PTM) Timing

Table 53 and Figure 37 provide I/O timing related to the PTM.

Table 53. Trace Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DTRD}	TRACE Data Delay From Trace Clock Maximum		$0.5 \times t_{SCLK0} + 3$	ns
t_{HTRD}	TRACE Data Hold From Trace Clock Minimum	$0.5 \times t_{SCLK0} - 2$		ns
t_{PTRCK}	TRACE Clock Period Minimum	$2 \times t_{SCLK0} - 1$		ns

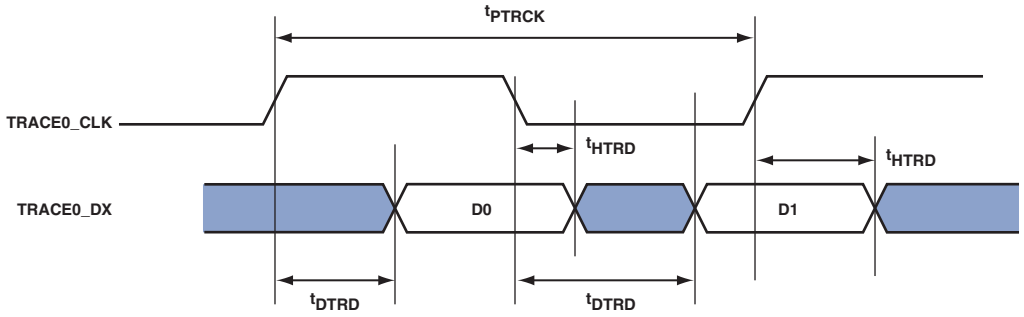


Figure 37. Trace Timing

Debug Interface (JTAG Emulation Port) Timing

Table 54 and Figure 38 provide I/O timing related to the debug interface (JTAG emulator port).

Table 54. JTAG Emulation Port Timing

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

[1] System Inputs = 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, SYS_FAULT, $\overline{\text{SYS\_RESOUT}}$.

[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_BMODE2-0, SYS_CLKOUT, SYS_FAULT, $\overline{\text{SYS\_RESOUT}}$.

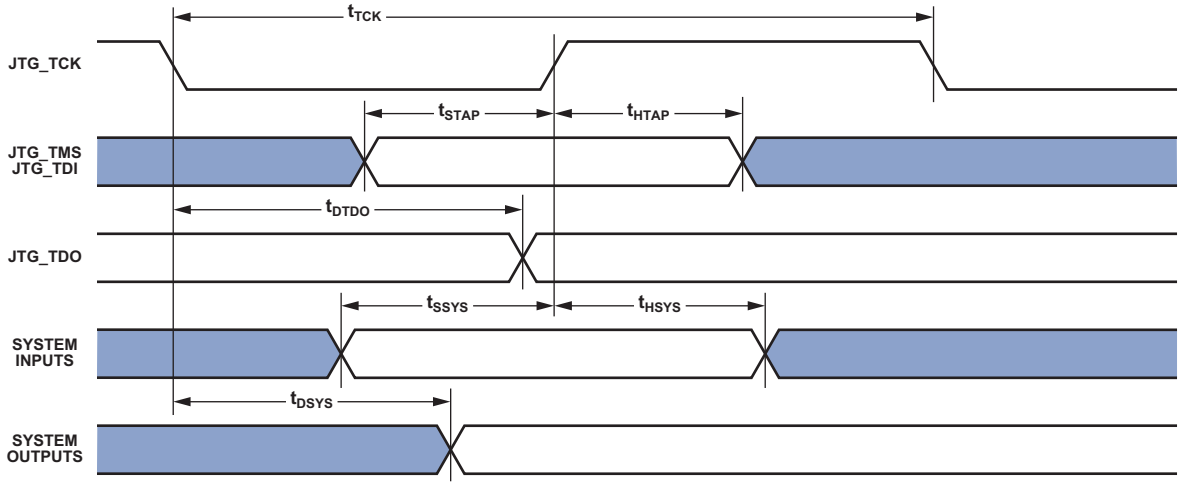


Figure 38. JTAG Port Timing

ADSP-2156x 400-BALL BGA BALL ASSIGNMENTS

The ADSP-2156x 400-Ball BGA Ball Assignments (Numerical by Ball Number) table lists the 400-ball BGA package by ball number.

The ADSP-2156x 400-Ball BGA Ball Assignments (Alphabetical by Pin Name) table lists the 400-ball BGA package by pin name.

ADSP-2156x 400-BALL BGA BALL ASSIGNMENTS (NUMERICAL BY BALL NUMBER)

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
A01	GND	C01	DMC0_LDM	E01	DMC0_DQ05	G01	DMC0_DQ04
A02	DMC0_DQ11	C02	DMC0_DQ08	E02	DMC0_DQ06	G02	DMC0_DQ02
A03	DMC0_DQ10	C03	GND	E03	GND	G03	GND
A04	$\overline{\text{DMC0\_UDQS}}$	C04	GND	E04	VDD_INT	G04	VDD_INT
A05	DMC0_DQ15	C05	GND	E05	VDD_INT	G05	VDD_INT
A06	DMC0_UDM	C06	GND	E06	VDD_INT	G06	VDD_DMC
A07	PB_00	C07	PB_03	E07	VDD_INT	G07	VDD_DMC
A08	SYS_BMODE2	C08	PB_05	E08	VDD_INT	G08	VDD_DMC
A09	$\overline{\text{SYS\_HWRST}}$	C09	PB_01	E09	VDD_INT	G09	VDD_DMC
A10	$\overline{\text{JTG\_TRST}}$	C10	$\overline{\text{SYS\_RESOUT}}$	E10	VDD_INT	G10	VDD_REF
A11	$\overline{\text{SYS\_FAULT}}$	C11	JTG_TDO	E11	VDD_INT	G11	VDD_REF
A12	$\overline{\text{DMC0\_WE}}$	C12	JTG_TMS	E12	VDD_INT	G12	VDD_DMC
A13	DMC0_A14	C13	JTG_TDI	E13	VDD_INT	G13	VDD_DMC
A14	DMC0_A13	C14	GND	E14	VDD_INT	G14	VDD_DMC
A15	DMC0_A10	C15	GND	E15	VDD_INT	G15	VDD_DMC
A16	DMC0_A08	C16	GND	E16	VDD_INT	G16	VDD_INT
A17	DMC0_BA1	C17	GND	E17	VDD_INT	G17	VDD_INT
A18	DMC0_A07	C18	GND	E18	GND	G18	GND
A19	DMC0_A04	C19	DMC0_A03	E19	DMC0_BA2	G19	DMC0_RZQ
A20	GND	C20	DMC0_A02	E20	$\overline{\text{DMC0\_CAS}}$	G20	DMC0_VREF0
B01	GND	D01	DMC0_DQ07	F01	$\overline{\text{DMC0\_LDQS}}$	H01	DMC0_DQ03
B02	GND	D02	GND	F02	DMC0_LDQS	H02	GND
B03	DMC0_DQ12	D03	DMC0_DQ09	F03	GND	H03	GND
B04	DMC0_UDQS	D04	GND	F04	VDD_INT	H04	VDD_INT
B05	DMC0_DQ14	D05	GND	F05	VDD_INT	H05	VDD_INT
B06	DMC0_DQ13	D06	GND	F06	VDD_DMC	H06	VDD_DMC
B07	PB_04	D07	GND	F07	VDD_DMC	H07	VDD_DMC
B08	PB_02	D08	VDD_EXT	F08	VDD_DMC	H08	GND
B09	SYS_BMODE1	D09	VDD_EXT	F09	VDD_DMC	H09	GND
B10	SYS_BMODE0	D10	VDD_EXT	F10	VDD_DMC	H10	GND
B11	JTG_TCK	D11	VDD_EXT	F11	VDD_DMC	H11	GND
B12	$\overline{\text{DMC0\_RESET}}$	D12	GND	F12	VDD_DMC	H12	GND
B13	DMC0_A15	D13	GND	F13	VDD_DMC	H13	GND
B14	DMC0_A12	D14	GND	F14	VDD_DMC	H14	VDD_DMC
B15	DMC0_A11	D15	GND	F15	VDD_DMC	H15	VDD_DMC
B16	DMC0_A09	D16	GND	F16	VDD_INT	H16	VDD_INT
B17	DMC0_BA0	D17	GND	F17	VDD_INT	H17	VDD_INT
B18	DMC0_A06	D18	GND	F18	GND	H18	GND
B19	GND	D19	DMC0_A00	F19	$\overline{\text{DMC0\_RAS}}$	H19	DMC0_ODT
B20	DMC0_A05	D20	DMC0_A01	F20	$\overline{\text{DMC0\_CS0}}$	H20	DMC0_CKE

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
J01	DMC0_DQ01	L09	GND	N17	GND	T05	VDD_EXT
J02	DMC0_DQ00	L10	GND	N18	GND	T06	VDD_REF
J03	GND	L11	GND	N19	GND	T07	VDD_REF
J04	DMC0_VREF1	L12	GND	N20	GND	T08	VDD_REF
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	GND	P06	VDD_INT	T14	VDD_REF
J11	GND	L19	VDD_REF	P07	VDD_INT	T15	VDD_REF
J12	GND	L20	VDD_REF	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_INT	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_INT	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	GND	R07	VDD_INT	U15	VDD_EXT
K12	GND	M20	GND	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_INT	N14	VDD_INT	T02	PA_05	V10	DAI0_PIN06
L07	VDD_INT	N15	VDD_REF	T03	PA_06	V11	DAI0_PIN09
L08	GND	N16	VDD_EXT	T04	GND	V12	DAI0_PIN20

Ball No.	Pin Name
V13	PC_00
V14	PC_05
V15	GND
V16	GND
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-2156x 400-BALL BGA BALL ASSIGNMENTS (ALPHABETICAL BY PIN NAME)

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
DAI0_PIN01	Y08	DMC0_BA2	E19	GND	C16	GND	K13
DAI0_PIN02	V09	$\overline{\text{DMC0\_CAS}}$	E20	GND	C17	GND	K17
DAI0_PIN03	W09	$\overline{\text{DMC0\_CK}}$	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	$\overline{\text{DMC0\_CS0}}$	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	L18
DAI1_PIN03	R18	DMC0_DQ10	A03	GND	E18	GND	M01
DAI1_PIN04	T20	DMC0_DQ11	A02	GND	F03	GND	M02
DAI1_PIN05	U20	DMC0_DQ12	B03	GND	F18	GND	M08
DAI1_PIN06	T19	DMC0_DQ13	B06	GND	G03	GND	M09
DAI1_PIN07	U19	DMC0_DQ14	B05	GND	G18	GND	M10
DAI1_PIN08	R19	DMC0_DQ15	A05	GND	H02	GND	M11
DAI1_PIN09	T18	DMC0_LDM	C01	GND	H03	GND	M12
DAI1_PIN10	U18	$\overline{\text{DMC0\_LDQS}}$	F01	GND	H08	GND	M13
DAI1_PIN11	V19	DMC0_LDQS	F02	GND	H09	GND	M17
DAI1_PIN12	V20	DMC0_ODT	H19	GND	H10	GND	M18
DAI1_PIN19	V17	$\overline{\text{DMC0\_RAS}}$	F19	GND	H11	GND	M19
DAI1_PIN20	W18	$\overline{\text{DMC0\_RESET}}$	B12	GND	H12	GND	M20
DMC0_A00	D19	DMC0_RZQ	G19	GND	H13	GND	N02
DMC0_A01	D20	DMC0_UDM	A06	GND	H18	GND	N08
DMC0_A02	C20	$\overline{\text{DMC0\_UDQS}}$	A04	GND	J03	GND	N09
DMC0_A03	C19	DMC0_UDQS	B04	GND	J08	GND	N10
DMC0_A04	A19	DMC0_VREF0	G20	GND	J09	GND	N11
DMC0_A05	B20	DMC0_VREF1	J04	GND	J10	GND	N12
DMC0_A06	B18	$\overline{\text{DMC0\_WE}}$	A12	GND	J11	GND	N13
DMC0_A07	A18	GND	A01	GND	J12	GND	N17
DMC0_A08	A16	GND	A20	GND	J13	GND	N18
DMC0_A09	B16	GND	B01	GND	J18	GND	N19
DMC0_A10	A15	GND	B02	GND	K01	GND	N20
DMC0_A11	B15	GND	B19	GND	K02	GND	P01
DMC0_A12	B14	GND	C03	GND	K03	GND	P17
DMC0_A13	A14	GND	C04	GND	K08	GND	P19
DMC0_A14	A13	GND	C05	GND	K09	GND	P20
DMC0_A15	B13	GND	C06	GND	K10	GND	R02
DMC0_BA0	B17	GND	C14	GND	K11	GND	R17
DMC0_BA1	A17	GND	C15	GND	K12	GND	T01

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
GND	T04	PB_03	C07	VDD_DMC	H06	VDD_INT	F05
GND	T17	PB_04	B07	VDD_DMC	H07	VDD_INT	F16
GND	U01	PB_05	C08	VDD_DMC	H14	VDD_INT	F17
GND	U04	PB_06	V05	VDD_DMC	H15	VDD_INT	G04
GND	U05	PB_07	W04	VDD_DMC	J06	VDD_INT	G05
GND	U16	PB_08	W03	VDD_DMC	J07	VDD_INT	G16
GND	U17	PB_09	V06	VDD_DMC	K07	VDD_INT	G17
GND	V03	PB_10	Y03	VDD_EXT	D08	VDD_INT	H04
GND	V04	PB_11	Y04	VDD_EXT	D09	VDD_INT	H05
GND	V15	PB_12	V07	VDD_EXT	D10	VDD_INT	H16
GND	V16	PB_13	W05	VDD_EXT	D11	VDD_INT	H17
GND	V18	PB_14	Y14	VDD_EXT	K04	VDD_INT	J05
GND	W01	PB_15	W14	VDD_EXT	L04	VDD_INT	J14
GND	W02	PC_00	V13	VDD_EXT	M04	VDD_INT	J15
GND	W19	PC_01	W15	VDD_EXT	M16	VDD_INT	J16
GND	W20	PC_02	Y15	VDD_EXT	N04	VDD_INT	J17
GND	Y01	PC_03	W16	VDD_EXT	N16	VDD_INT	K05
GND	Y02	PC_04	Y17	VDD_EXT	P04	VDD_INT	K06
GND	Y05	PC_05	V14	VDD_EXT	P16	VDD_INT	K14
GND	Y07	PC_06	W17	VDD_EXT	R04	VDD_INT	K15
GND	Y13	PC_07	Y18	VDD_EXT	R16	VDD_INT	K16
GND	Y16	SYS_BMODE0	B10	VDD_EXT	T05	VDD_INT	L05
GND	Y19	SYS_BMODE1	B09	VDD_EXT	T16	VDD_INT	L06
GND	Y20	SYS_BMODE2	A08	VDD_EXT	U06	VDD_INT	L07
JTG_TCK	B11	SYS_CLKIN0	N01	VDD_EXT	U07	VDD_INT	L14
JTG_TDI	C13	SYS_CLKOUT	M03	VDD_EXT	U08	VDD_INT	L15
JTG_TDO	C11	$\overline{\text{SYS\_FAULT}}$	A11	VDD_EXT	U09	VDD_INT	M05
JTG_TMS	C12	$\overline{\text{SYS\_HWRST}}$	A09	VDD_EXT	U10	VDD_INT	M06
$\overline{\text{JTG\_TRST}}$	A10	$\overline{\text{SYS\_RESOUT}}$	C10	VDD_EXT	U11	VDD_INT	M07
PA_00	N03	SYS_XTAL0	L01	VDD_EXT	U12	VDD_INT	M14
PA_01	P03	VDD_DMC	F06	VDD_EXT	U13	VDD_INT	N06
PA_02	P02	VDD_DMC	F07	VDD_EXT	U14	VDD_INT	N07
PA_03	R03	VDD_DMC	F08	VDD_EXT	U15	VDD_INT	N14
PA_04	R01	VDD_DMC	F09	VDD_INT	E04	VDD_INT	P06
PA_05	T02	VDD_DMC	F10	VDD_INT	E05	VDD_INT	P07
PA_06	T03	VDD_DMC	F11	VDD_INT	E06	VDD_INT	P08
PA_07	V01	VDD_DMC	F12	VDD_INT	E07	VDD_INT	P09
PA_08	U03	VDD_DMC	F13	VDD_INT	E08	VDD_INT	P10
PA_09	V02	VDD_DMC	F14	VDD_INT	E09	VDD_INT	P11
PA_10	U02	VDD_DMC	F15	VDD_INT	E10	VDD_INT	P12
PA_11	V08	VDD_DMC	G06	VDD_INT	E11	VDD_INT	P13
PA_12	W06	VDD_DMC	G07	VDD_INT	E12	VDD_INT	P14
PA_13	Y06	VDD_DMC	G08	VDD_INT	E13	VDD_INT	R06
PA_14	W07	VDD_DMC	G09	VDD_INT	E14	VDD_INT	R07
PA_15	W08	VDD_DMC	G12	VDD_INT	E15	VDD_INT	R08
PB_00	A07	VDD_DMC	G13	VDD_INT	E16	VDD_INT	R09
PB_01	C09	VDD_DMC	G14	VDD_INT	E17	VDD_INT	R10
PB_02	B08	VDD_DMC	G15	VDD_INT	F04	VDD_INT	R11

Pin Name	Ball No.
VDD_INT	R12
VDD_INT	R13
VDD_INT	R14
VDD_REF	G10
VDD_REF	G11
VDD_REF	L19
VDD_REF	L20
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 400-BALL CSP_BGA

Figure 39 shows an overview of signal placement on the 400-ball CSP_BGA.

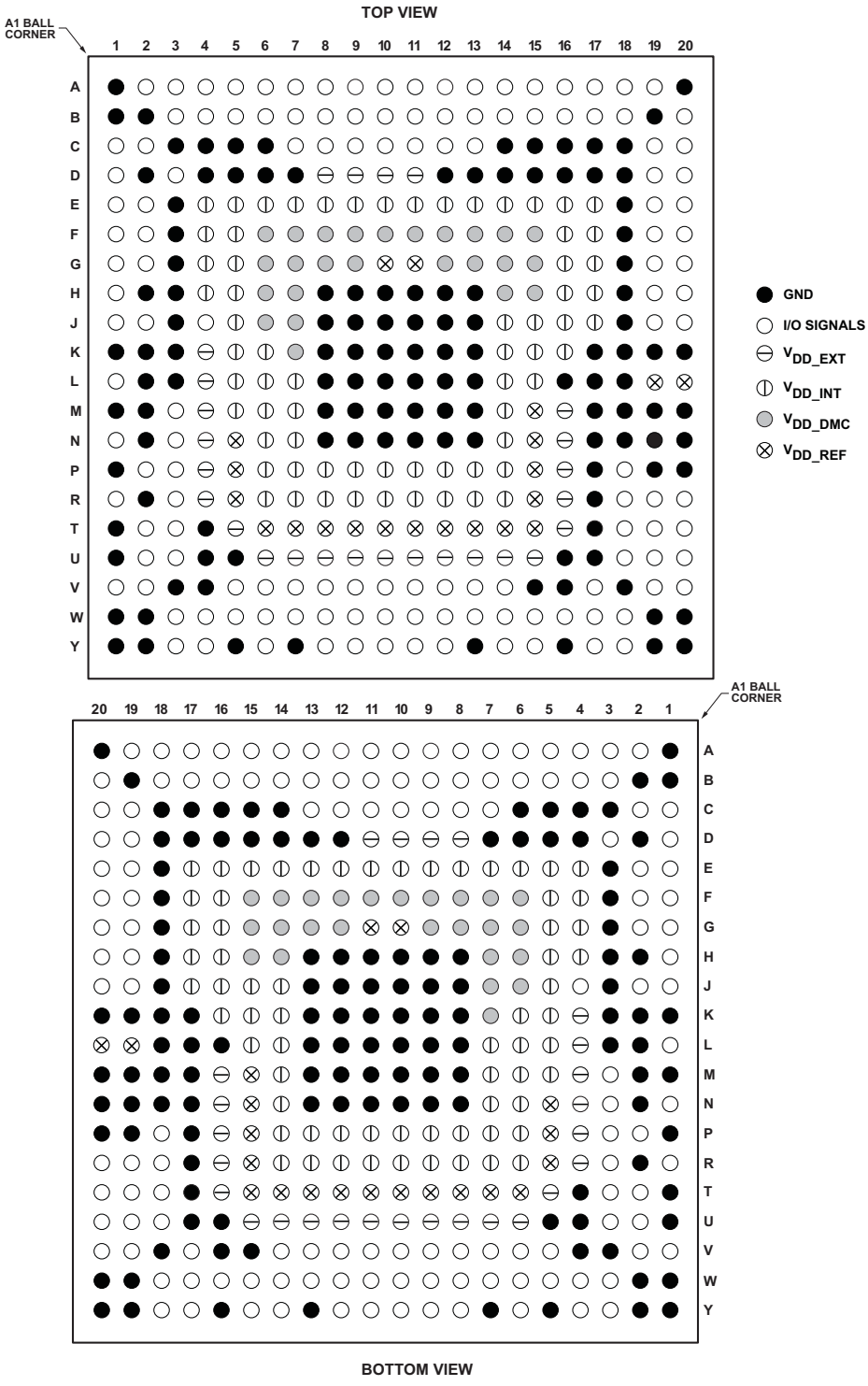


Figure 39. 400-Ball CSP_BGA Configuration

ADSP-2156x 120-LEAD LQFP LEAD ASSIGNMENTS

The ADSP-2156x 120-Lead LQFP Lead Assignments (Numerical by Lead Number) table lists the 120-lead LQFP package by lead number.

The ADSP-2156x 120-Lead LQFP Lead Assignments (Alphabetical by Pin Name) table lists the 120-lead LQFP package by pin name.

ADSP-2156x 120-LEAD LQFP LEAD ASSIGNMENTS (NUMERICAL BY LEAD NUMBER)

Lead No.	Pin Name	Lead No.	Pin Name	Lead No.	Pin Name	Lead No.	Pin Name
01	VDD_INT	41	DAI0_PIN01	81	GND	121[1]	GND
02	GND	42	DAI0_PIN02	82	GND		
03	VDD_INT	43	DAI0_PIN03	83	VDD_REF		
04	VDD_INT	44	VDD_INT	84	GND		
05	SYS_CLKIN0	45	VDD_INT	85	VDD_REF		
06	SYS_XTAL0	46	DAI0_PIN04	86	GND		
07	VDD_REF	47	DAI0_PIN05	87	VDD_INT		
08	VDD_INT	48	DAI0_PIN06	88	VDD_INT		
09	GND	49	DAI0_PIN07	89	VDD_INT		
10	SYS_CLKOUT	50	VDD_EXT	90	GND		
11	VDD_REF	51	VDD_REF	91	VDD_INT		
12	VDD_EXT	52	DAI0_PIN08	92	GND		
13	VDD_INT	53	DAI0_PIN09	93	VDD_INT		
14	PA_00	54	DAI0_PIN10	94	VDD_INT		
15	PA_01	55	DAI0_PIN19	95	VDD_INT		
16	PA_02	56	DAI0_PIN20	96	JTG_TDI		
17	VDD_INT	57	VDD_INT	97	JTG_TCK		
18	PA_03	58	GND	98	JTG_TMS		
19	PA_04	59	VDD_INT	99	JTG_TDO		
20	PA_05	60	VDD_INT	100	VDD_REF		
21	PA_06	61	VDD_INT	101	VDD_EXT		
22	VDD_INT	62	GND	102	$\overline{\text{SYS\_FAULT}}$		
23	VDD_REF	63	DAI1_PIN20	103	$\overline{\text{JTG\_TRST}}$		
24	VDD_EXT	64	DAI1_PIN19	104	$\overline{\text{SYS\_HWRST}}$		
25	PA_07	65	DAI1_PIN10	105	SYS_BMODE0		
26	PA_08	66	DAI1_PIN09	106	SYS_BMODE1		
27	PA_09	67	VDD_REF	107	$\overline{\text{SYS\_RESOUT}}$		
28	PA_10	68	VDD_EXT	108	PB_00		
29	VDD_INT	69	VDD_INT	109	PB_01		
30	GND	70	DAI1_PIN08	110	VDD_INT		
31	GND	71	DAI1_PIN07	111	VDD_EXT		
32	VDD_INT	72	DAI1_PIN06	112	VDD_REF		
33	VDD_INT	73	DAI1_PIN05	113	PB_02		
34	PA_11	74	DAI1_PIN04	114	PB_03		
35	PA_12	75	DAI1_PIN03	115	PB_04		
36	PA_13	76	DAI1_PIN02	116	PB_05		
37	PA_14	77	DAI1_PIN01	117	VDD_INT		
38	VDD_EXT	78	VDD_EXT	118	VDD_INT		
39	VDD_REF	79	VDD_REF	119	VDD_INT		
40	PA_15	80	VDD_INT	120	GND		

[1] Pin121 is the GND supply (see Figure 41) for the processor; this pad must connect to GND.

ADSP-2156x 120-LEAD LQFP LEAD ASSIGNMENTS (ALPHABETICAL BY PIN NAME)

Pin Name	Lead No.	Pin Name	Lead No.	Pin Name	Lead No.
DAI0_PIN01	41	PA_03	18	VDD_INT	45
DAI0_PIN02	42	PA_04	19	VDD_INT	57
DAI0_PIN03	43	PA_05	20	VDD_INT	59
DAI0_PIN04	46	PA_06	21	VDD_INT	60
DAI0_PIN05	47	PA_07	25	VDD_INT	61
DAI0_PIN06	48	PA_08	26	VDD_INT	69
DAI0_PIN07	49	PA_09	27	VDD_INT	80
DAI0_PIN08	52	PA_10	28	VDD_INT	87
DAI0_PIN09	53	PA_11	34	VDD_INT	88
DAI0_PIN10	54	PA_12	35	VDD_INT	89
DAI0_PIN19	55	PA_13	36	VDD_INT	91
DAI0_PIN20	56	PA_14	37	VDD_INT	93
DAI1_PIN01	77	PA_15	40	VDD_INT	94
DAI1_PIN02	76	PB_00	108	VDD_INT	95
DAI1_PIN03	75	PB_01	109	VDD_INT	110
DAI1_PIN04	74	PB_02	113	VDD_INT	117
DAI1_PIN05	73	PB_03	114	VDD_INT	118
DAI1_PIN06	72	PB_04	115	VDD_INT	119
DAI1_PIN07	71	PB_05	116	VDD_REF	07
DAI1_PIN08	70	SYS_BMODE0	105	VDD_REF	11
DAI1_PIN09	66	SYS_BMODE1	106	VDD_REF	23
DAI1_PIN10	65	SYS_CLKIN0	05	VDD_REF	39
DAI1_PIN19	64	SYS_CLKOUT	10	VDD_REF	51
DAI1_PIN20	63	$\overline{\text{SYS\_FAULT}}$	102	VDD_REF	67
GND	02	$\overline{\text{SYS\_HWRST}}$	104	VDD_REF	79
GND	09	$\overline{\text{SYS\_RESOUT}}$	107	VDD_REF	83
GND	30	SYS_XTAL0	06	VDD_REF	85
GND	31	VDD_EXT	12	VDD_REF	100
GND	58	VDD_EXT	24	VDD_REF	112
GND	62	VDD_EXT	38		
GND	81	VDD_EXT	50		
GND	82	VDD_EXT	68		
GND	84	VDD_EXT	78		
GND	86	VDD_EXT	101		
GND	90	VDD_EXT	111		
GND	92	VDD_INT	01		
GND	120	VDD_INT	03		
GND	121[1]	VDD_INT	04		
JTG_TCK	97	VDD_INT	08		
JTG_TDI	96	VDD_INT	13		
JTG_TDO	99	VDD_INT	17		
JTG_TMS	98	VDD_INT	22		
$\overline{\text{JTG\_TRST}}$	103	VDD_INT	29		
PA_00	14	VDD_INT	32		
PA_01	15	VDD_INT	33		
PA_02	16	VDD_INT	44		

[1] Pin121 is the GND supply (see Figure 41) for the processor; this pad must connect to GND.

CONFIGURATION OF THE 120-LEAD LQFP LEAD CONFIGURATION

Figure 40 shows the top view of the 120-lead LQFP lead configuration and Figure 41 shows the bottom view of the 120-lead LQFP lead configuration.

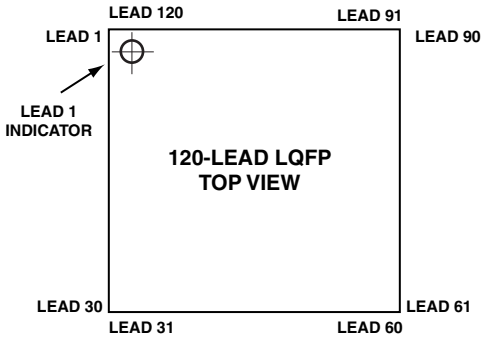


Figure 40. 120-Lead LQFP Lead Configuration (Top View)

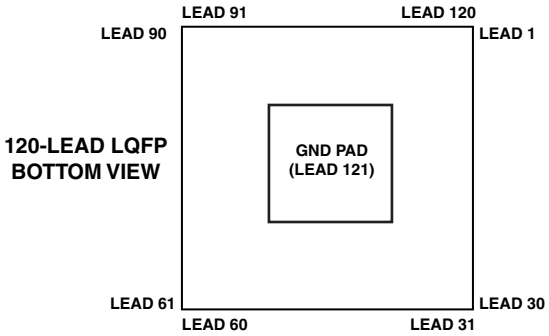
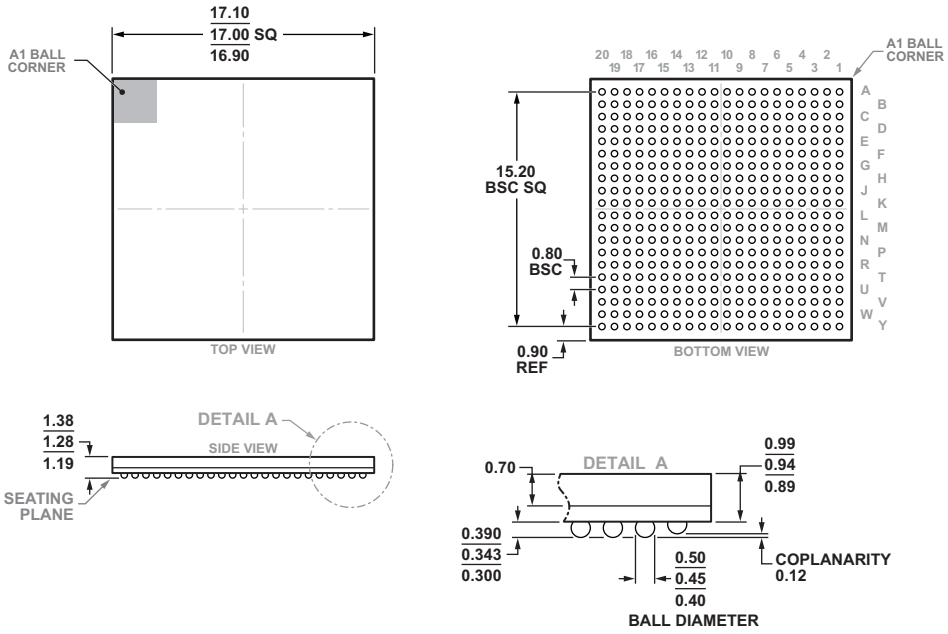


Figure 41. 120-Lead LQFP Lead Configuration (Bottom View)

OUTLINE DIMENSIONS

Dimensions in Figure 42 (for the 400-ball BGA) and Figure 43 (for the 120-lead LQFP) are shown in millimeters.



COMPLIANT TO JEDEC STANDARDS MO-275-MMAB-1

Figure 42. 400-Ball Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-400-3)
Dimensions shown in millimeters

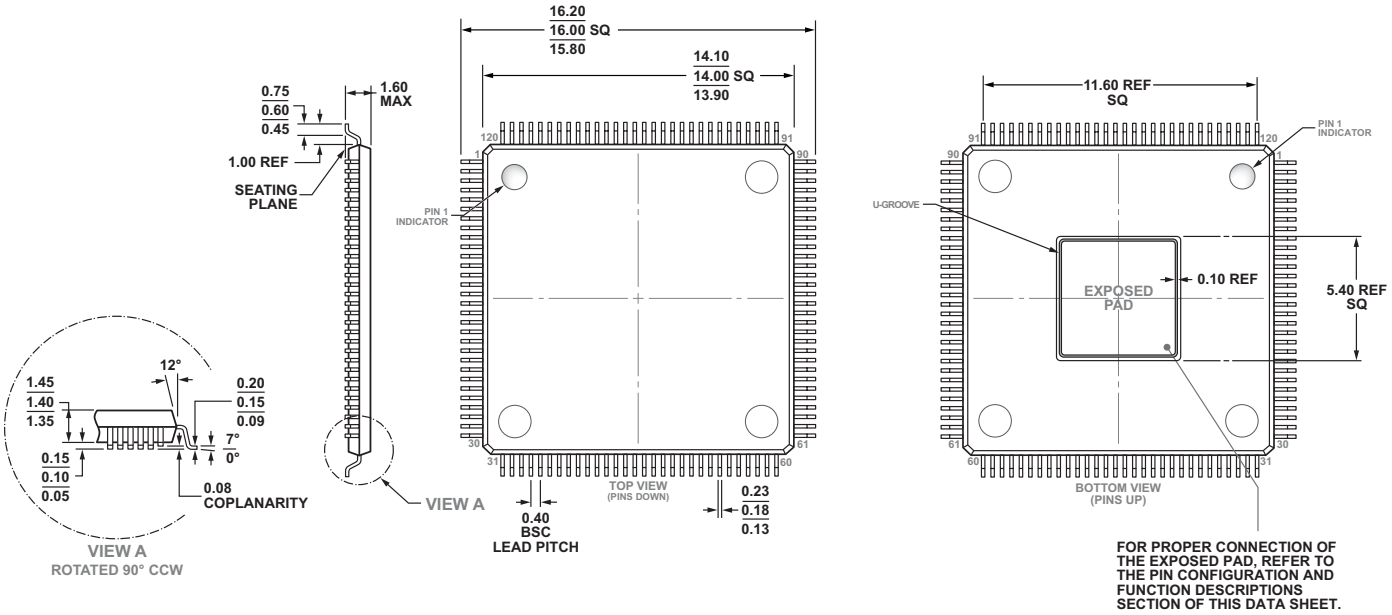


Figure 43. 120-Lead Low Profile Quad Flat Package, Exposed Pad [LQFP]
(SW-120-4)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 55 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 55. CSP_BGA Data for Use with Surface-Mount Design

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

PLANNED AUTOMOTIVE PRODUCTION PRODUCTS

Model[1]	Processor Instruction Rate (Max)	Temperature Range[2]	External Memory Port	Package Description	Package Option
ADSP-21562WCSWZ4	400 MHz	–40°C to +105°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21563WCSWZ6	600 MHz	–40°C to +105°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21563WCSWZ8	800 MHz	–40°C to +105°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565WCSWZ8	800 MHz	–40°C to +105°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565WCSWZ10	1000 MHz	–40°C to +105°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21566WCBCZ4	400 MHz	–40°C to +105°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21567WCBCZ6	600 MHz	–40°C to +105°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21567WCBCZ8	800 MHz	–40°C to +105°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21569WCBCZ8	800 MHz	–40°C to +105°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21569WCBCZ10	1000 MHz	–40°C to +105°C	1	Pad 400-Ball CSP_BGA	BC-400-3

[1] Z =RoHS compliant part.
[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification.

PLANNED PRODUCTION PRODUCTS

Model[1]	Processor Instruction Rate (Max)	Temperature Range[2]	External Memory Port	Package Description	Package Option
ADSP-21562KSWZ4	400 MHz	0°C to +70°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21562BSWZ4	400 MHz	–40°C to +85°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21563KSWZ8	800 MHz	0°C to +70°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21563BSWZ8	800 MHz	–40°C to +85°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565KSWZ8	800 MHz	0°C to +70°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565BSWZ8	800 MHz	–40°C to +85°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565KSWZ10	1000 MHz	0°C to +70°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21565BSWZ10	1000 MHz	–40°C to +85°C	0	120-Lead LQFP_EP, Exposed	SW-120-4
ADSP-21566BBCZ4	400 MHz	–40°C to +85°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21567BBCZ6	600 MHz	–40°C to +85°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21567BBCZ8	800 MHz	–40°C to +85°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21569BBCZ8	800 MHz	–40°C to +85°C	1	Pad 400-Ball CSP_BGA	BC-400-3
ADSP-21569BBCZ10	1000 MHz	–40°C to +85°C	1	Pad 400-Ball CSP_BGA	BC-400-3

[1] Z =RoHS compliant part.
[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification.

PRE RELEASE PRODUCTS

Model[1]	Processor Instruction Rate (Max)	Temperature Range[2, 3, 4]	Package Description	Package Option
ADSP-21565-SWZ8ENG	800 MHz	N/A	120-Lead LQFP_EP, Exposed	SW-120-4

[1] Z =RoHS compliant part.
[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification.
[3] These are pre production parts. See ENG-Grade agreement for details.
[4] N/A means not available.

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

PR20695-6/20(PrG)

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Analog Devices Inc.:

ADSP-21563KSWZ8 ADSP-21565KSWZ8 ADSP-21565KSWZ10