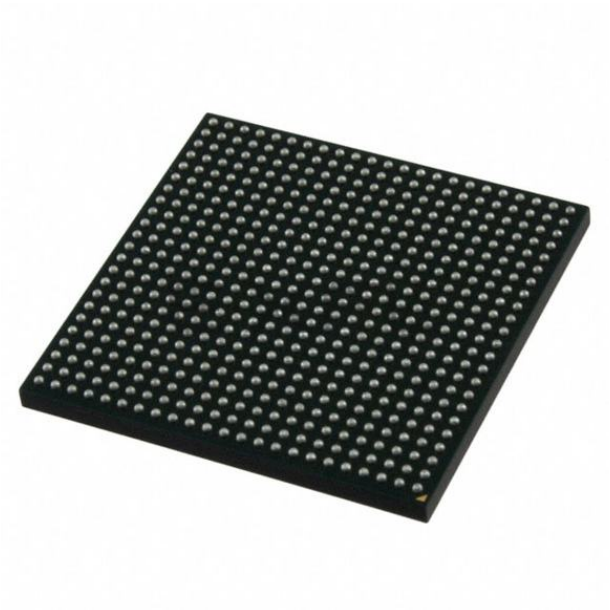

Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Floating Point
Interface	CAN, EBI/EMI, Ethernet, DAI, I²C, MMC/SD/SDIO, SPI, SPORT, UART/USART, USB OTG
Clock Rate	450MHz
Non-Volatile Memory	ROM (512kB)
On-Chip RAM	640kB
Voltage - I/O	3.30V
Voltage - Core	1.10V
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	529-LFBGA, CSPBGA
Supplier Device Package	529-CSPBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-sc587bbcz-4b

Table 2. Comparison of ADSP-SC58x/ADSP-2158x Processor Features

Processor Feature	ADSP-SC582	ADSP-SC583	ADSP-SC584	ADSP-SC587	ADSP-SC589	ADSP-21583	ADSP-21584	ADSP-21587
ARM Cortex-A5 (MHz, Max)	450	450	450	450	450	N/A	N/A	N/A
ARM Core L1 Cache (I, D kB)	32, 32	32, 32	32, 32	32, 32	32, 32	N/A	N/A	N/A
ARM Core L2 Cache (kB)	256	256	256	256	256	N/A	N/A	N/A
SHARC+ Core1 (MHz, Max)	450	450	450	450	450	450	450	450
SHARC+ Core2 (MHz, Max)	N/A	450	450	450	450	450	450	450
SHARC L1 SRAM/Core (kB)	640	384	640	640	640	384	640	640
System Memory: L2 SRAM (Shared) (kB)	256	256	256	256	256	256	256	256
System Memory: L2 ROM (Shared) (kB)	512	512	512	512	512	512	512	512
System Memory: DDR3/DDR2/LPDDR1 Controller (16-bit)	1	1	1	2	2	1	1	2
USB 2.0 HS + PHY (Host/Device/OTG)	1	1	1	1	1	N/A	N/A	N/A
USB 2.0 HS + PHY (Host/Device)	N/A	N/A	N/A	1	1	N/A	N/A	N/A
10/100 Std EMAC	N/A	N/A	N/A	1	1	N/A	N/A	N/A
10/100/1000 /AVB EMAC + Timer IEEE 1588	1	1	1	1	1	N/A	N/A	N/A
SDIO/eMMC	N/A	N/A	N/A	1	1	N/A	N/A	N/A
PCIe 2.0 (1 Lane)	N/A	N/A	N/A	N/A	1	N/A	N/A	N/A
RTC	N/A	N/A	N/A	1	1	N/A	N/A	1
GPIO Ports	Port A to E	Port A to E	Port A to E	Port A to G	Port A to G	Port A to E	Port A to E	Port A to G
GPIO + DAI Pins	80 + 28	80 + 28	80 + 28	102 + 40	102 + 40	80 + 28	80 + 28	102 + 40
19 mm × 19 mm Package Options	349-BGA	349-BGA	349-BGA	529-BGA	529-BGA	349-BGA	349-BGA	529-BGA

Table 3. Comparison of ADSP-SC58x/ADSP-2158x Processor Features for Automotive

Processor Feature	ADSP-SC582W	ADSP-SC583W	ADSP-SC584W	ADSP-SC587W	ADSP-21583W	ADSP-21584W
ARM Cortex-A5 (MHz, Max)	450	450	450	450	N/A	N/A
ARM Core L1 Cache (I, D kB)	32, 32	32, 32	32, 32	32, 32	N/A	N/A
ARM Core L2 Cache (kB)	256	256	256	256	N/A	N/A
SHARC+ Core1 (MHz, Max)	450	450	450	450	450	450
SHARC+ Core2 (MHz, Max)	N/A	450	450	450	450	450
SHARC L1 SRAM/Core (kB)	640	384	640	640	384	640
System Memory: L2 SRAM (Shared) (kB)	256	256	256	256	256	256
System Memory: L2 ROM (Shared) (kB)	512	512	512	512	512	512
System Memory: DDR3/DDR2/LPDDR1 Controller (16-bit)	1	1	1	2	1	1
USB 2.0 HS + PHY (Host/Device/OTG)	1	1	1	1	N/A	N/A
USB 2.0 HS + PHY (Host/Device)	N/A	N/A	N/A	1	N/A	N/A
10/100 Std EMAC	N/A	N/A	N/A	1	N/A	N/A
10/100/1000/AVB EMAC + Timer IEEE 1588	1	1	1	1	N/A	N/A
SDIO/eMMC	N/A	N/A	N/A	1	N/A	N/A
PCIe 2.0 (1 Lane)	N/A	N/A	N/A	N/A	N/A	N/A
MLB 3-Pin/6-Pin	1	1	1	1	1	1
RTC	N/A	N/A	N/A	1	N/A	N/A
GPIO Ports	Port A to E	Port A to E	Port A to E	Port A to G	Port A to E	Port A to E
GPIO + DAI Pins	80 + 28	80 + 28	80 + 28	102 + 40	80 + 28	80 + 28
19 mm × 19 mm Package Options	349-BGA	349-BGA	349-BGA	529-BGA	349-BGA	349-BGA

Generic Interrupt Controller (GIC), PL390 (ADSP-SC58x Only)

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

Generic Interrupt Controller Port0 (GICPORT0)

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

Generic Interrupt Controller Port1 (GICPORT1)

The GICPORT1 CPU interface block performs priority masking and preemption handling for a connected processor in the system. GICPORT1 supports 8 software generated interrupts (SGIs) and 254 shared peripheral interrupts (SPIs).

L2 Cache Controller, PL310 (ADSP-SC58x Only)

The L2 cache controller, PL310 (see Figure 2), works efficiently with the ARM Cortex-A5 processors that implement system fabric. The cache controller directly interfaces on the data and instruction interface. The internal pipelining of the cache controller is optimized to enable the processors to operate at the same clock frequency. The cache controller supports the following:

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

SHARC PROCESSOR

Figure 3 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 4 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.

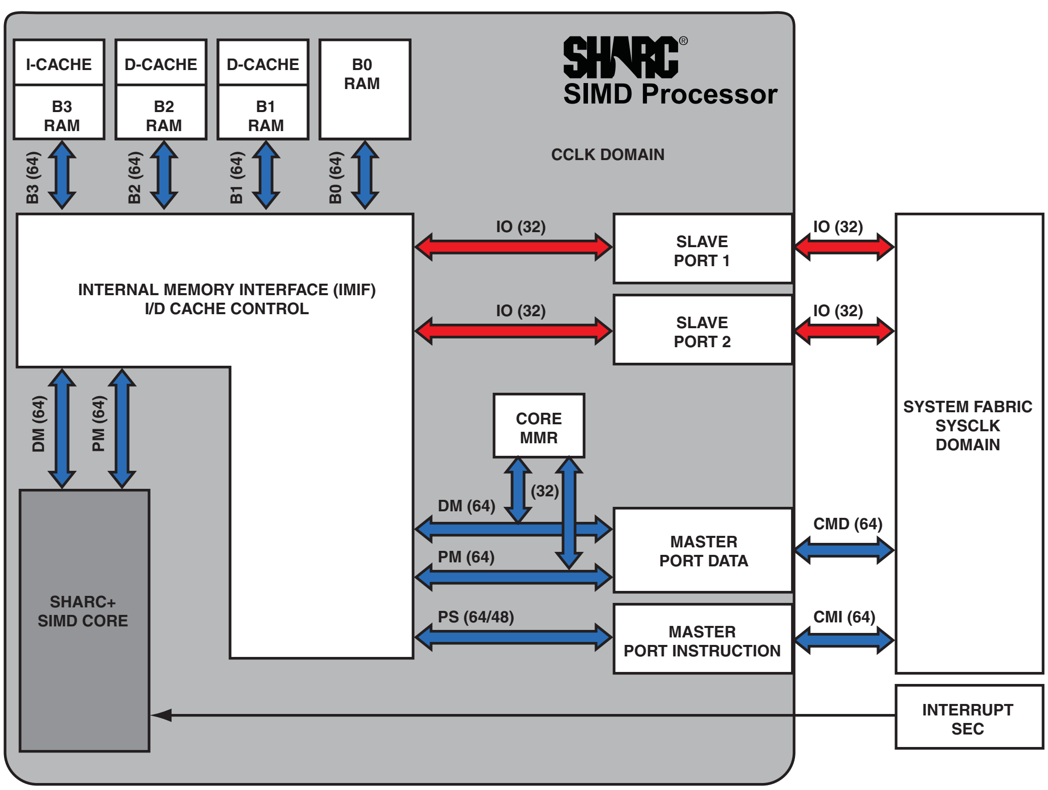

Figure 3. SHARC Processor Block Diagram

Memory Direct Memory Access (MDMA)

The processor supports various MDMA operations, including,

- Standard bandwidth MDMA channels with CRC protection (32-bit bus width, runs on SCLK0)
- Enhanced bandwidth MDMA channel (32-bit bus width, runs on SYSCLK)
- Maximum bandwidth MDMA channels (64-bit bus width, run on SYCLK, one channel can be assigned to the FFT accelerator)

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 C ode (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/byte mirroring option (endianness)
- Fault/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 five 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 occur synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions triggered on the one side by the SHARC+ core, such as data alignment (SIMD/long word) or compute violations (fixed or floating point), and illegal instructions cause core exceptions. Conditions triggered on the other side by the SEC, such as error correcting codes (ECC)/parity/watchdog/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)

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

- Comprehensive system event source management including interrupt enable, fault enable, priority, core mapping, and source grouping
- A distributed programming model where each system event source control and all status fields are independent of each other
- Determinism where all system events have the same propagation delay and provide unique identification of a specific system event source
- A slave control port that provides access to all SEC registers for configuration, status, and interrupt/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

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

Software Add-Ins for CrossCore Embedded Studio

Analog Devices offers software add-ins which seamlessly integrate with CrossCore Embedded Studio to extend 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 Lite evaluation boards and EZ-Extender daughter cards is provided by software add-ins called board support packages (BSPs). The BSPs contain the required drivers, pertinent release notes, and select example code for the given evaluation hardware. A download link for a specific BSP is located on the web page for the associated EZ-KIT or EZ-Extender product.

Middleware Packages

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

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

Algorithmic Modules

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

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-SC58x/ADSP-2158x 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 website.

The application signal chains page in the Circuits from the Lab® site (http:\\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

SECURITY FEATURES DISCLAIMER

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

Table 11. ADSP-SC58x/ADSP-2158x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
$\overline{\text{DMC\_UDQS}}$	InOut	**Data Strobe for Upper Byte (Complement).** Complement of $\overline{\text{UDQS}}$. Not used in single-ended mode.
DMC_VREF	Input	**Voltage Reference.** Externally driven to VDD_DMC/2.
$\overline{\text{DMC\_WE}}$	Output	**Write Enable.** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the $\overline{\text{WE}}$ input of dynamic memory.
ETH_CRS	Input	**Carrier Sense/RMII Receive Data Valid.** Multiplexed on alternate clock cycles. CRS— asserted by the PHY when either the transmit or receive medium is not idle. Deasserted when both are idle. RXDV—asserted by the PHY when the data on RXDn is valid.
ETH_MDC	Output	**Management Channel Clock.** Clocks the MDC input of the PHY.
ETH_MDIO	InOut	**Management Channel Serial Data.** Bidirectional data bus for PHY control.
ETH_PTPAUXIN[n]	Input	**PTP Auxiliary Trigger Input.** Assert this signal to take an auxiliary snapshot of the time and store it in the auxiliary time stamp FIFO.
ETH_PTPCLKIN[n]	Input	**PTP Clock Input.** Optional external PTP clock input.
ETH_PTPPPS[n]	Output	**PTP Pulse Per Second Output.** When the advanced time stamp feature enables, this signal is asserted based on the PPS mode selected. Otherwise, PTPPPS is asserted every time the seconds counter is incremented.
ETH_REFCLK	Input	**Reference Clock.** Externally supplied Ethernet clock.
ETH_RXCLK_REFCLK	Input	**RXCLK (GigE) or REFCLK (10/100).**
ETH_RXCTL_CRS	Input	**RXCTL (GigE) or CRS (10/100).**
ETH_RXD[n]	Input	**Receive Data n.** Receive data bus.
ETH_TXCLK	Output	**Transmit Clock.**
ETH_TXCTL_TXEN	Output	**TXCTL (GigE) or TXEN (10/100).**
ETH_TXD[n]	Output	**Transmit Data n.** Transmits data bus.
ETH_TXEN	Output	**Transmit Enable.** When asserted, signal indicates the data on TXDn is valid.
HADC_EOC_DOUT	Output	**End of Conversion/Serial Data Out.** Transitions high for one cycle of the HADC internal clock at the end of every conversion. Alternatively, HADC serial data out can be seen by setting the appropriate bit in HADC_CTL.
HADC_MUX[n]	Input	**Controls to External Multiplexer.** Allows additional input channels when connected to an external multiplexer.
HADC_VIN[n]	Input	**Analog Input at Channel n.** Analog voltage inputs for digital conversion.
HADC_VREFN	Input	**Ground Reference for ADC.** Connect to an external voltage reference that meets data sheet specifications.
HADC_VREFP	Input	**External Reference for ADC.** Connect to an external voltage reference that meets data sheet specifications.
JTG_TCK	Input	**JTAG Clock.** JTAG test access port clock.
JTG_TDI	Input	**JTAG Serial Data In.** JTAG test access port data input.
JTG_TDO	Output	**JTAG Serial Data Out.** JTAG test access port data output.
JTG_TMS	Input	**JTAG Mode Select.** JTAG test access port mode select.
$\overline{\text{JTG\_TRST}}$	Input	**JTAG Reset.** JTAG test access port reset.
LP_ACK	InOut	**Acknowledge.** Provides handshaking. When the link port is configured as a receiver, ACK is an output. When the link port is configured as a transmitter, ACK is an input.
LP_CLK	InOut	**Clock.** When the link port is configured as a receiver, CLK is an input. When the link port is configured as a transmitter, CLK is an output.
LP_D[n]	InOut	**Data n.** Data bus. Input when receiving, output when transmitting.
MLB_CLKN	Input	**Differential Clock (–).**
MLB_CLKP	Input	**Differential Clock (+).**
MLB_DATN	InOut	**Differential Data (–).**
MLB_DATP	InOut	**Differential Data (+).**
MLB_SIGN	InOut	**Differential Signal (–).**

Table 11. ADSP-SC58x/ADSP-2158x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
MLB_SIGP	InOut	**Differential Signal (+).**
MLB_CLK	Input	**Single-Ended Clock.**
MLB_DAT	InOut	**Single-Ended Data.**
MLB_SIG	InOut	**Single-Ended Signal.**
MLB_CLKOUT	Output	**Single-Ended Clock Out.**
MSI_CD	Input	**Card Detect.** Connects to a pull-up resistor and to the card detect output of an SD socket.
MSI_CLK	Output	**Clock.** The clock signal applied to the connected device from the MSI.
MSI_CMD	InOut	**Command.** Sends commands to and receives responses from the connected device.
MSI_D[n]	InOut	**Data n.** Bidirectional data bus.
MSI_INT	Input	**eSDIO Interrupt Input.** Used only for eSDIO. Connects to an eSDIO card interrupt output. An interrupt may be sampled even when the MSI clock to the card is switched off.
PCIE_CLKM	Input	**CLK –.**
PCIE_CLKP	Input	**CLK +.**
PCIE_REF	InOut	**Reference Resistor.** Attach a 200 Ω, 1%, 100-ppm/C precision resistor to ground on the board.
PCIE_RXM	Input	**RX –.**
PCIE_RXP	Input	**RX +.**
PCIE_TXM	Output	**TX –.**
PCIE_TXP	Output	**TX +.**
PPI_CLK	InOut	**Clock.** Input in external clock mode, output in internal clock mode.
PPI_D[nn]	InOut	**Data n.** Bidirectional data bus.
PPI_FS1	InOut	**Frame Sync 1 (HSYNC).** Behavior depends on EPPI mode. See the EPPI chapter of the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference for more details.
PPI_FS2	InOut	**Frame Sync 2 (VSYNC).** Behavior depends on EPPI mode. See the EPPI chapter of the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference for more details.
PPI_FS3	InOut	**Frame Sync 3 (FIELD).** Behavior depends on EPPI mode. See the EPPI chapter of the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference for more details.
PWM_AH	Output	**Channel A High Side.** High side drive signal.
PWM_AL	Output	**Channel A Low Side.** Low side drive signal.
PWM_BH	Output	**Channel B High Side.** High side drive signal.
PWM_BL	Output	**Channel B Low Side.** Low side drive signal.
PWM_CH	Output	**Channel C High Side.** High side drive signal.
PWM_CL	Output	**Channel C Low Side.** Low side drive signal.
PWM_DH	Output	**Channel D High Side.** High side drive signal.
PWM_DL	Output	**Channel D Low Side.** Low side drive signal.
PWM_SYNC	Input	**PWMTMR Grouped.** This input is for an externally generated sync signal. If the sync signal is internally generated, no connection is necessary.
PWM_TRIP[n]	Input	**Shutdown Input n.** When asserted, the selected PWM channel outputs are shut down immediately.
P_[nn]	InOut	**Position n.** General-purpose input/output. See the GP Ports chapter of the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference for more details.
RTC_CLKIN	Input	**Crystal Input/External Oscillator Connection.** Connect to an external clock source or crystal.
RTC_XTAL	Output	**Crystal Output.** Drives an external crystal. Must be left unconnected if an external clock is driving RTC_CLKIN.
SINC_CLK0	Output	**Clock 0.**
SINC_D0	Input	**Data 0.**
SINC_D1	Input	**Data 1.**
SINC_D2	Input	**Data 2.**
SINC_D3	Input	**Data 3.**

529-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processor pin definitions are shown Table 19 for the 529-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 Digital Audio Interface (DAI) chapter of the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference for complete information on the use of the DAIs and SRUs.

Table 19. ADSP-SC58x/ADSP-2158x 529-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
ACM0_A0	ACM0 ADC Control Signals	C	PC_13
ACM0_A1	ACM0 ADC Control Signals	C	PC_14
ACM0_A2	ACM0 ADC Control Signals	C	PC_15
ACM0_A3	ACM0 ADC Control Signals	D	PD_00
ACM0_A4	ACM0 ADC Control Signals	D	PD_01
ACM0_T0	ACM0 External Trigger n	C	PC_12
C1_FLG0	SHARC Core 1 Flag Pin	E	PE_01
C1_FLG1	SHARC Core 1 Flag Pin	E	PE_03
C1_FLG2	SHARC Core 1 Flag Pin	E	PE_05
C1_FLG3	SHARC Core 1 Flag Pin	E	PE_07
C2_FLG0	SHARC Core 2 Flag Pin	E	PE_02
C2_FLG1	SHARC Core 2 Flag Pin	E	PE_04
C2_FLG2	SHARC Core 2 Flag Pin	E	PE_06
C2_FLG3	SHARC Core 2 Flag Pin	E	PE_08
CAN0_RX	CAN0 Receive	C	PC_07
CAN0_TX	CAN0 Transmit	C	PC_08
CAN1_RX	CAN1 Receive	B	PB_10
CAN1_TX	CAN1 Transmit	B	PB_09
CNT0_DG	CNT0 Count Down and Gate	B	PB_14
CNT0_UD	CNT0 Count Up and Direction	B	PB_12
CNT0_ZM	CNT0 Count Zero Marker	B	PB_11
DAI0_PIN01	DAI0 Pin 1	Not Muxed	DAI0_PIN01
DAI0_PIN02	DAI0 Pin 2	Not Muxed	DAI0_PIN02
DAI0_PIN03	DAI0 Pin 3	Not Muxed	DAI0_PIN03
DAI0_PIN04	DAI0 Pin 4	Not Muxed	DAI0_PIN04
DAI0_PIN05	DAI0 Pin 5	Not Muxed	DAI0_PIN05
DAI0_PIN06	DAI0 Pin 6	Not Muxed	DAI0_PIN06
DAI0_PIN07	DAI0 Pin 7	Not Muxed	DAI0_PIN07
DAI0_PIN08	DAI0 Pin 8	Not Muxed	DAI0_PIN08
DAI0_PIN09	DAI0 Pin 9	Not Muxed	DAI0_PIN09
DAI0_PIN10	DAI0 Pin 10	Not Muxed	DAI0_PIN10
DAI0_PIN11	DAI0 Pin 11	Not Muxed	DAI0_PIN11
DAI0_PIN12	DAI0 Pin 12	Not Muxed	DAI0_PIN12
DAI0_PIN13	DAI0 Pin 13	Not Muxed	DAI0_PIN13
DAI0_PIN14	DAI0 Pin 14	Not Muxed	DAI0_PIN14

Table 19. ADSP-SC58x/ADSP-2158x 529-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
$\overline{\text{DMC0\_CK}}$	DMC0 Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
$\overline{\text{DMC0\_CS0}}$	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
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	$\overline{\text{DMC0\_WE}}$
DMC1_A00	DMC1 Address 0	Not Muxed	DMC1_A00
DMC1_A01	DMC1 Address 1	Not Muxed	DMC1_A01
DMC1_A02	DMC1 Address 2	Not Muxed	DMC1_A02
DMC1_A03	DMC1 Address 3	Not Muxed	DMC1_A03
DMC1_A04	DMC1 Address 4	Not Muxed	DMC1_A04
DMC1_A05	DMC1 Address 5	Not Muxed	DMC1_A05
DMC1_A06	DMC1 Address 6	Not Muxed	DMC1_A06
DMC1_A07	DMC1 Address 7	Not Muxed	DMC1_A07
DMC1_A08	DMC1 Address 8	Not Muxed	DMC1_A08
DMC1_A09	DMC1 Address 9	Not Muxed	DMC1_A09
DMC1_A10	DMC1 Address 10	Not Muxed	DMC1_A10
DMC1_A11	DMC1 Address 11	Not Muxed	DMC1_A11
DMC1_A12	DMC1 Address 12	Not Muxed	DMC1_A12
DMC1_A13	DMC1 Address 13	Not Muxed	DMC1_A13
DMC1_A14	DMC1 Address 14	Not Muxed	DMC1_A14
DMC1_A15	DMC1 Address 15	Not Muxed	DMC1_A15
DMC1_BA0	DMC1 Bank Address 0	Not Muxed	DMC1_BA0
DMC1_BA1	DMC1 Bank Address 1	Not Muxed	DMC1_BA1

Table 27. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PA_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 15 | EMAC0 PTP Pulse-Per-Second Output 2 | SINC0 Data 1 | SMC0 Address 9 Notes: No notes
PB_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 0 | EMAC0 PTP Pulse-Per-Second Output 1 | EPPI0 Data 14 | SINC0 Data 2 | SMC0 Address 8 | TIMER0 Alternate Clock 3 Notes: No notes
PB_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 1 | EMAC0 PTP Pulse-Per-Second Output 0 | EPPI0 Data 15 | SINC0 Clock 0 | SMC0 Address 7 | TIMER0 Alternate Clock 4 Notes: No notes
PB_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 2 | EMAC0 PTP Clock Input 0 | EPPI0 Data 16 | SMC0 Address 4 | UART1 Transmit Notes: No notes
PB_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 3 | EMAC0 PTP Auxiliary Trigger Input 0 | EPPI0 Data 17 | SMC0 Address 3 | UART1 Receive | TIMER0 Alternate Capture Input 1 Notes: No notes
PB_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 4 | EPPI0 Data 12 | MLB0 Single-Ended Clock | SINC0 Data 3 | SMC0 Asynchronous Ready | EMAC0 PTP Auxiliary Trigger Input 1 Notes: No notes
PB_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 5 | EPPI0 Data 13 | MLB0 Single-Ended Signal | SMC0 Address 1 | EMAC0 PTP Auxiliary Trigger Input 2 Notes: No notes
PB_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 6 | MLB0 Single-Ended Data | PWM0 Channel B High Side | SMC0 Address 2 | EMAC0 PTP Auxiliary Trigger Input 3 Notes: No notes
PB_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 7 | LP1 Data 0 | PWM0 Channel A High Side | SMC0 Data 15 | TIMER0 Timer 3 Notes: No notes
PB_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 8 | LP1 Data 1 | PWM0 Channel A Low Side | SMC0 Data 14 | TIMER0 Timer 4 Notes: No notes

Table 27. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PB_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 9 | CAN1 Transmit | LP1 Data 2 | SMC0 Data 13 Notes: No notes
PB_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 10 | CAN1 Receive | LP1 Data 3 | SMC0 Data 12 | TIMER0 Timer 2 | TIMER0 Alternate Capture Input 4 Notes: No notes
PB_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 11 | LP1 Data 4 | PWM0 Channel D High Side | SMC0 Data 11 | CNT0 Count Zero Marker Notes: No notes
PB_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 12 | LP1 Data 5 | PWM0 Channel D Low Side | SMC0 Data 10 | CNT0 Count Up and Direction Notes: No notes
PB_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 13 | LP1 Data 6 | PWM0 Channel C High Side | SMC0 Data 9 Notes: No notes
PB_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 14 | LP1 Data 7 | PWM0 Channel C Low Side | SMC0 Data 8 | TIMER0 Timer 5 | CNT0 Count Down and Gate Notes: No notes
PB_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 15 | LP1 Acknowledge | PWM0 Shutdown Input 0 | SMC0 Write Enable | TIMER0 Timer 1 Notes: No notes
PCIE0_CLKM	Input	NA	PullDown	none	none	VDD_PCIE	Desc: PCIE0 CLK - Notes: No notes
PCIE0_CLKP	Input	NA	PullDown	none	none	VDD_PCIE	Desc: PCIE0 CLK + Notes: No notes
PCIE0_REF	a	NA	PullDown	none	none	VDD_PCIE	Desc: PCIE0 Reference Notes: No notes
PCIE0_RXM	Input	NA	PullDown	none	none	VDD_PCIE_RX	Desc: PCIE0 RX - Notes: No notes
PCIE0_RXP	Input	NA	PullDown	none	none	VDD_PCIE_RX	Desc: PCIE0 RX + Notes: No notes
PCIE0_TXM	InOut	J	PullDown	none	none	VDD_PCIE_TX	Desc: PCIE0 TX - Notes: No notes
PCIE0_TXP	InOut	J	PullDown	none	none	VDD_PCIE_TX	Desc: PCIE0 TX + Notes: No notes

Table 27. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PC_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 13 | ACM0 ADC Control Signals | SPI1 Slave Select Output 1 | UART0 Transmit Notes: No notes
PC_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 14 | ACM0 ADC Control Signals | UART0 Receive | TIMER0 Alternate Capture Input 0 Notes: No notes
PC_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 15 | ACM0 ADC Control Signals | EPPI0 Frame Sync 3 (FIELD) | SMC0 Memory Select 0 | UART0 Request to Send Notes: No notes
PD_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 0 | ACM0 ADC Control Signals | EPPI0 Data 23 | SMC0 Data 7 | UART0 Clear to Send Notes: No notes
PD_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 1 | ACM0 ADC Control Signals | SMC0 Output Enable | SPI0 Slave Select Output 2 | SPI0 Slave Select Input Notes: No notes
PD_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 2 | LP0 Data 0 | PWM1 Shutdown Input 0 | TRACE0 Trace Data 0 Notes: No notes
PD_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 3 | LP0 Data 1 | PWM1 Channel A High Side | TRACE0 Trace Data 1 Notes: No notes
PD_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 4 | LP0 Data 2 | PWM1 Channel A Low Side | TRACE0 Trace Data 2 Notes: No notes
PD_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 5 | LP0 Data 3 | PWM1 Channel B High Side | TRACE0 Trace Data 3 Notes: No notes
PD_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 6 | LP0 Data 4 | PWM1 Channel B Low Side | TRACE0 Trace Data 4 Notes: No notes
PD_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 7 | LP0 Data 5 | PWM1 Channel C High Side | TRACE0 Trace Data 5 Notes: No notes

Clock Related Operating Conditions

Table 29 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 29. Clock Operating Conditions

Parameter		Restriction	Min	Typ	Max	Unit
f_{CCLK}	Core Clock Frequency	$f_{CCLK} \geq f_{SYSCLK}$			450	MHz
f_{SYSCLK}	SYSCLK Frequency				225	MHz
f_{SCLK0}	SCLK0 Frequency[1]	$f_{SYSCLK} \geq f_{SCLK0}$	30		112.5	MHz
f_{SCLK1}	SCLK1 Frequency	$f_{SYSCLK} \geq f_{SCLK1}$			112.5	MHz
f_{DCLK}	LPDDR Clock Frequency				200	MHz
f_{DCLK}	DDR2 Clock Frequency				400	MHz
f_{DCLK}	DDR3 Clock Frequency				450	MHz
f_{OCLK}	Output Clock Frequency[2]				225	MHz
$f_{SYS\_CLKOUTJ}$	SYS_CLKOUT Period Jitter[3, 4]			±2		%
$f_{PCLKPROG}$	Programmed PPI Clock When Transmitting Data and Frame Sync				75	MHz
$f_{PCLKPROG}$	Programmed PPI Clock When Receiving Data or Frame Sync				45	MHz
$f_{PCLKEXT}$	External PPI Clock When Receiving Data and Frame Sync[5]	$f_{PCLKEXT} \leq f_{SCLK1}$			75	MHz
$f_{PCLKEXT}$	External PPI Clock Transmitting Data or Frame Sync[5, 6]	$f_{PCLKEXT} \leq f_{SCLK1}$			45	MHz
$f_{LCLKTPROG}$	Programmed Link Port Transmit Clock				150	MHz
$f_{LCLKREXT}$	External Link Port Receive Clock[5, 6]	$f_{LCLKEXT} \leq f_{CLKO8}$			150	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Transmitting Data and Frame Sync				56.25	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Receiving Data or Frame Sync				28.125	MHz
$f_{SPTCLKEXT}$	External SPT Clock When Receiving Data and Frame Sync[5, 6]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			56.25	MHz
$f_{SPTCLKEXT}$	External SPT Clock Transmitting Data or Frame Sync[5, 6]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			28.125	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data				75	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data				75	MHz
$f_{SPICLKEXT}$	External SPI Clock When Receiving Data[5, 6]	$f_{SPICLKEXT} \leq f_{SCLK1}$			75	MHz
$f_{SPICLKEXT}$	External SPI Clock When Transmitting Data[5, 6]	$f_{SPICLKEXT} \leq f_{SCLK1}$			45	MHz
$f_{ACLKPROG}$	Programmed ACM Clock				56.25	MHz

[1] The minimum frequency for SCLK0 applies only when using the USB.
[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 typical value is the percentage of the SYS_CLKOUT period.
[5] The maximum achievable frequency for any peripheral in external clock mode is dependent on the ability 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 f_{SCLK} (f_{SCLK0} or f_{SCLK1}) that clocks the peripheral.

Table 30. Phase-Locked Loop (PLL) Operating Conditions

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

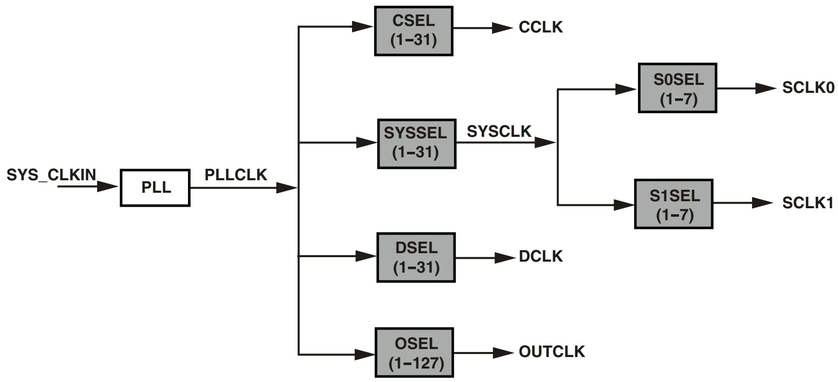

Figure 8. Clock Relationships and Divider Values

Asynchronous Page Mode Read

Table 47 and Figure 14 show asynchronous memory page mode read timing, related to the SMC.

Table 47. Asynchronous Page Mode Read

Parameter		Min	Max	Unit
Switching Characteristics				
t_{AV}	SMC0_Axx (Address) Valid for First Address Minimum Width[1]	$(PREST + RST + PREAT + RAT) \times t_{SCLK0} - 2$		ns
t_{AV1}	SMC0_Axx (Address) Valid for Subsequent SMC0_Ax (Address) Minimum Width	$PGWS \times t_{SCLK0} - 2$		ns
t_{WADV}	SMC0_NORDV Active Low Width[2]	$RST \times t_{SCLK0} - 2$		ns
t_{HARE}	Output[3] Hold After $\overline{\text{SMC0\_ARE}}$ High[4]	$RHT \times t_{SCLK0} - 2$		ns
t_{WARE}[5]	$\overline{\text{SMC0\_ARE}}$ Active Low Width[6, 7]	$(RAT + (Nw - 1) \times PGWS) \times t_{SCLK0} - 2$		ns

[1] PREST, RST, PREAT and RAT values set using the SMC_BxETIM.PREST bits, SMC_BxTIM.RST bits, SMC_BxETIM.PREAT bits, and the SMC_BxTIM.RAT bits.
[2] RST value set using the SMC_BxTIM.RST bits.
[3] Output signals are SMC0_Ax, $\overline{\text{SMC0\_AMSx}}$, $\overline{\text{SMC0\_AOE}}$.
[4] RHT value set using the SMC_BxTIM.RHT bits.
[5] SMC_BxCTL.ARDYEN bit = 0.
[6] RAT value set using the SMC_BxTIM.RAT bits.
[7] Nw = Number of 16-bit data words read.

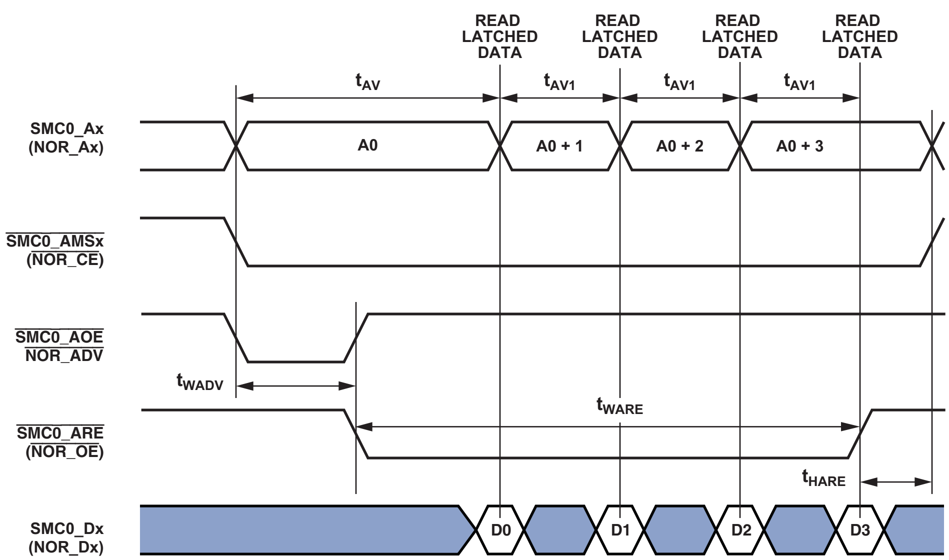

Figure 14. Asynchronous Page Mode Read

Asynchronous Write

Table 48 and Figure 15 show asynchronous memory write timing, related to the SMC.

Table 48. Asynchronous Memory Write

Parameter		Min	Max	Unit
Timing Requirement				
$t_{DARDYAWE}$[1]	SMC0_ARDY Valid After $\overline{\text{SMC0\_AWE}}$ Low [2]		(WAT – 2.5) × t_{SCLK0} – 17.5	ns
Switching Characteristics				
t_{ENDAT}	DATA Enable After $\overline{\text{SMC0\_AMSx}}$ Assertion	–3.5		ns
t_{DDAT}	DATA Disable After $\overline{\text{SMC0\_AMSx}}$ Deassertion		2.5	ns
t_{AMSAWE}	ADDR/$\overline{\text{SMC0\_AMSx}}$ Assertion Before $\overline{\text{SMC0\_AWE}}$ Low[3]	(PREST + WST + PREAT) × t_{SCLK0} – 2		ns
t_{HAWE}	Output[4] Hold After $\overline{\text{SMC0\_AWE}}$ High[5]	WHT × t_{SCLK0} – 3.5		ns
t_{WAWE}[6]	$\overline{\text{SMC0\_AWE}}$ Active Low Width[2]	WAT × t_{SCLK0} – 2		ns
$t_{DAWEARDY}$[1]	$\overline{\text{SMC0\_AWE}}$ High Delay After SMC0_ARDY Assertion	2.5 × t_{SCLK0}	3.5 × t_{SCLK0} + 17.5	ns

[1] SMC_BxCTL.ARDYEN bit = 1.
[2] WAT value set using the SMC_BxTIM.WAT bits.
[3] PREST, WST, PREAT values set using the SMC_BxETIM.PREST bits, SMC_BxTIM.WST bits, SMC_BxETIM.PREAT bits, and the SMC_BxTIM.RAT bits.
[4] Output signals are DATA, SMC0_Ax, $\overline{\text{SMC0\_AMSx}}$, $\overline{\text{SMC0\_ABEx}}$.
[5] WHT value set using the SMC_BxTIM.WHT bits.
[6] SMC_BxCTL.ARDYEN bit = 0.

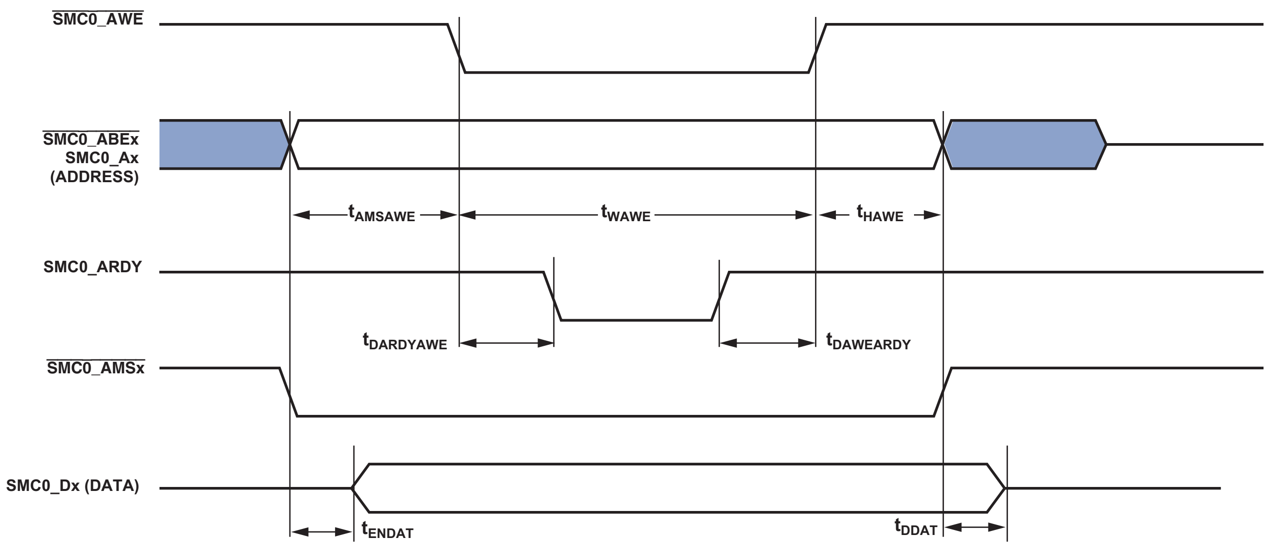

Figure 15. Asynchronous Write

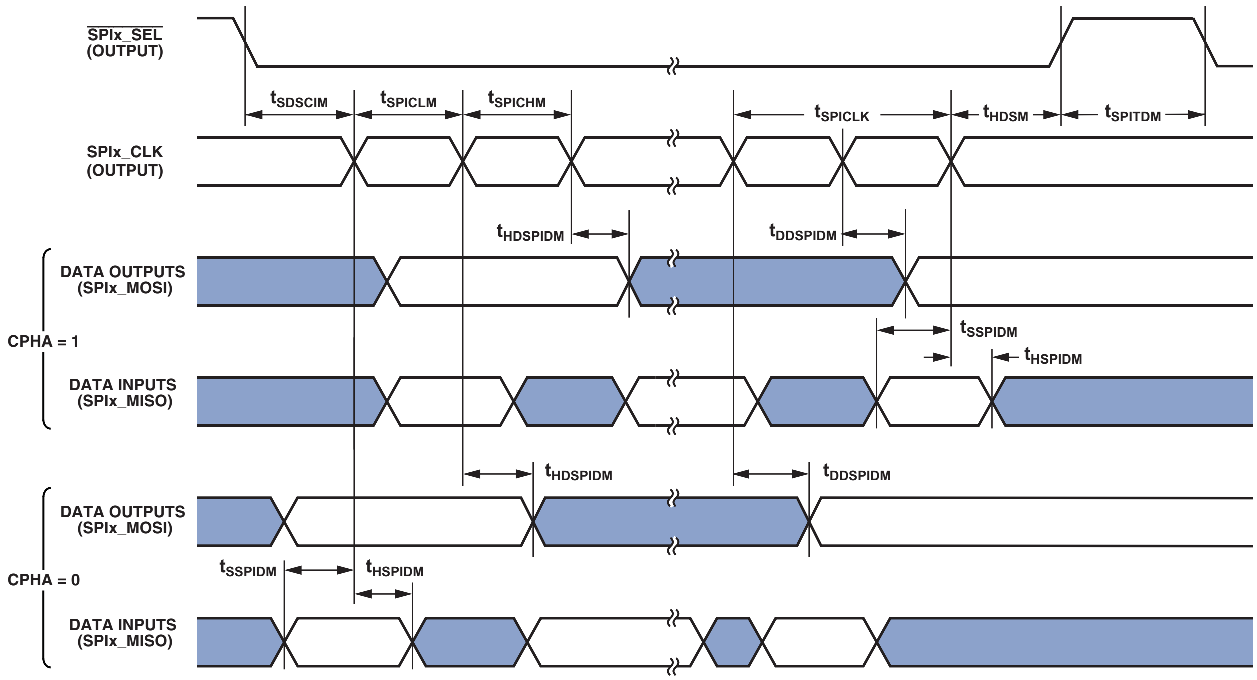

Figure 43. SPI Port—Master Timing

SPI Port—Open Drain Mode (ODM) Timing

In Figure 46 and Figure 47 and Table 75 and Table 76, 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 74. SPI Port ODM Master Mode Timing[1]

Parameter		Min	Max	Unit
Switching Characteristics				
$t_{HDSPIODMM}$	SPIx_CLK Edge to High Impedance from Data Out Valid	–1		ns
$t_{DDSPIODMM}$	SPIx_CLK Edge to Data Out Valid from High Impedance	–1	+6	ns

[1] All specifications apply to all three SPIs.

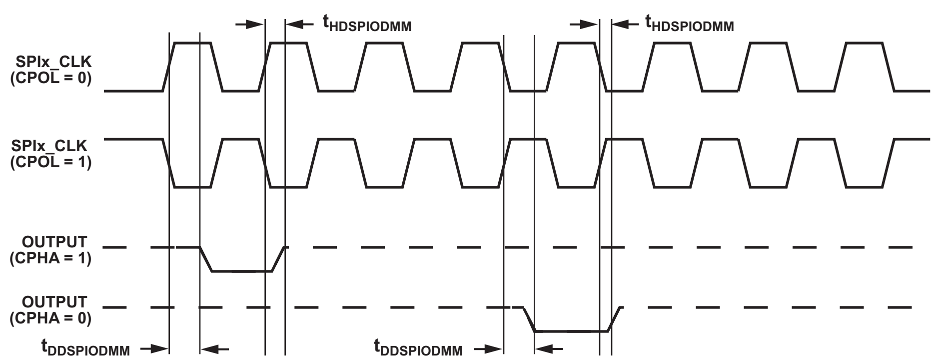

Figure 46. ODM Master Mode

Table 75. SPI Port—ODM Slave Mode[1]

Parameter		Min	Max	Unit
Timing Requirements				
$t_{HDSPIODMS}$	SPIx_CLK Edge to High Impedance from Data Out Valid	0		ns
$t_{DDSPIODMS}$	SPIx_CLK Edge to Data Out Valid from High Impedance		11	ns

[1] All specifications apply to all three SPIs.

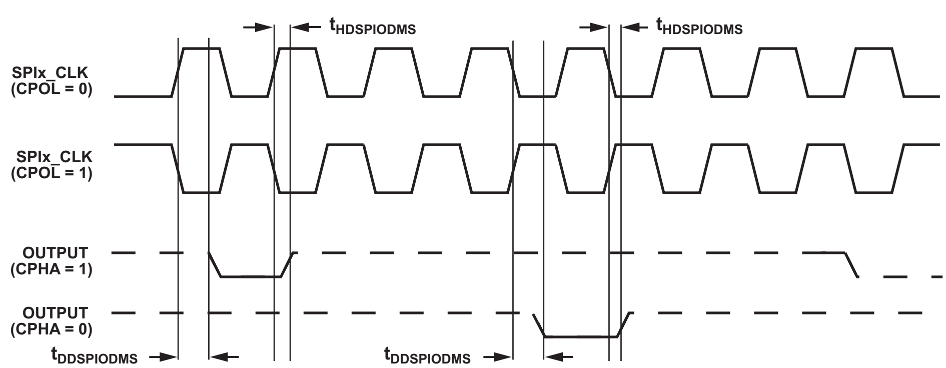

Figure 47. ODM Slave Mode

Pulse Width Modulator (PWM) Timing

Table 83 and Figure 55 describe timing, related to the PWM.

Table 83. PWM Timing[1]

Parameter		Min	Max	Unit
Timing Requirement				
t_{ES}	External Sync Pulse Width	$2 \times t_{SCLK0}$		ns
Switching Characteristics				
t_{DODIS}	Output Inactive (off) After Trip Input[2]		15	ns
t_{DOE}	Output Delay After External Sync[2, 3]	$2 \times t_{SCLK0} + 5.5$	$5 \times t_{SCLK0} + 14$	ns

[1] All specifications apply to all three PWMs.
[2] PWM outputs are PWMx_AH, PWMx_AL, PWMx_BH, PWMx_BL, PWMx_CH, and PWMx_CL.
[3] When the external sync signal is synchronous to the peripheral clock, it takes fewer clock cycles for the output to appear compared to when the external sync signal is asynchronous to the peripheral clock.

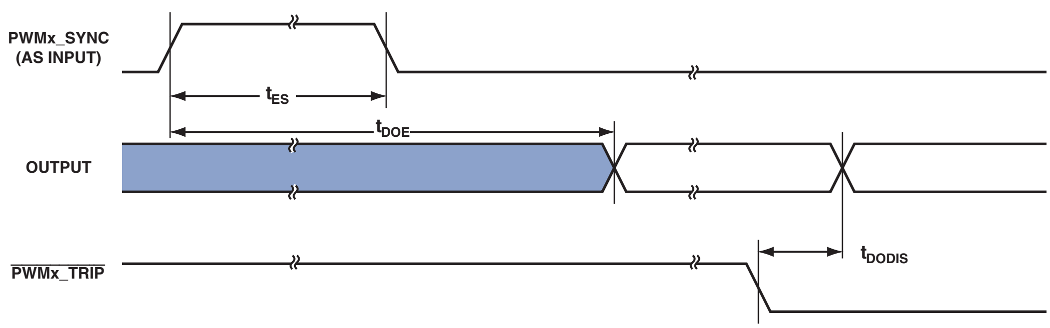

Figure 55. PWM Timing

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

The UART ports receive and transmit operations are described in the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference.

Controller Area Network (CAN) Interface

The CAN interface timing is described in the ADSP-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference.

Universal Serial Bus (USB) OTG—Receive and Transmit Timing

Table 87 describes the USB OTG receive and transmit operations.

Table 87. USB OTG—Receive and Transmit Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
f_{USBS}	USB_XI Frequency	24	24	MHz
fs_{USB}	USB_XI Clock Frequency Stability	–50	+50	ppm

[1] This specification is supported by USB0.

PCI Express (PCIe)

The PCIe interface complies with the Gen1 and Gen2 x1 lane data rate specification and supports up to 3.0 PCIe base functionality.

For more information about PCIe, see the following standards:

- *PCI Express Base 3.0 Specification*, Revision 1.0, PCI-SIG
- *PCI Express 2.0 Card Electromechanical Specification*, Revision 2.0, PCI-SIG
- *PHY Interface for the PCI Express Architecture*, Revision 2.0, Intel Corporation
- *PCI-SIG Engineering Change Request: L1 Substates*, February 1, 2012, PCI-SIG
- *IEEE Standard 1149.1-2001*, IEEE
- *IEEE Standard 1149.6-2003*, IEEE

Debug Interface (JTAG Emulation Port) Timing

Table 103 and Figure 76 provide I/O timing related to the debug interface (JTAG Emulator Port).

Table 103. JTAG Emulation Port Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{TCK}	JTG_TCK Period	20		ns
t_{STAP}	JTG_TDI, JTG_TMS Setup Before JTG_TCK High	4		ns
t_{HTAP}	JTG_TDI, JTG_TMS Hold After JTG_TCK High	4		ns
t_{SSYS}	System Inputs Setup Before JTG_TCK High[1]	12		ns
t_{HSYS}	System Inputs Hold After JTG_TCK High[1]	5		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		13.5	ns
t_{DSYS}	System Outputs Delay After JTG_TCK Low[3]		17	ns

[1] System Inputs = MLB0_CLKP, MLB0_DATP, MLB0_SIGP, DAI0_PIN20-01, DAI1_PIN20-01, DMC0_A15-0, DMC1_A15-0, DMC0_DQ15-0, DMC1_DQ15-0, DMC0_RESET, DMC1_RESET, PA_15-0, PB_15-0, PC_15-0, PD_15-0, PE_15-0, PF_15-0, PG_5-0, SYS_BMODE2-0, SYS_FAULT, SYS_FAULT, SYS_RESOUT, TWI2-0_SCL, TWI2-0_SDA2.

[2] 50 MHz maximum.

[3] System Outputs = DMC0_A15-0, DMC0_BA2-0, DMC0_CAS, DMC0_CK, DMC0_CKE, DMC0_CS0, DMC0_DQ15-0, DMC0_LDM, DMC0_LDQS, DMC0_ODT, DMC0_RAS, DMC0_RESET, DMC0_UDM, DMC0_UDQS, DMC0_WE, DMC1_A15-0, DMC1_BA2-0, DMC1_CAS, DMC1_CK, DMC1_CKE, DMC1_CS0, DMC1_DQ15-0, DMC1_LDM, DMC1_LDQS, DMC1_ODT, DMC1_RAS, DMC1_RESET, DMC1_UDM, DMC1_UDQS, DMC1_WE, MLB0_DATP, MLB0_SIGP, PA_15-0, PB_15-0, PC_15-0, PCIE_TXP, PD_15-0, PE_15-0, PF_15-0, PG_5-0, SYS_BMODE2-0, SYS_CLKOUT, SYS_FAULT, SYS_FAULT, SYS_RESOUT.

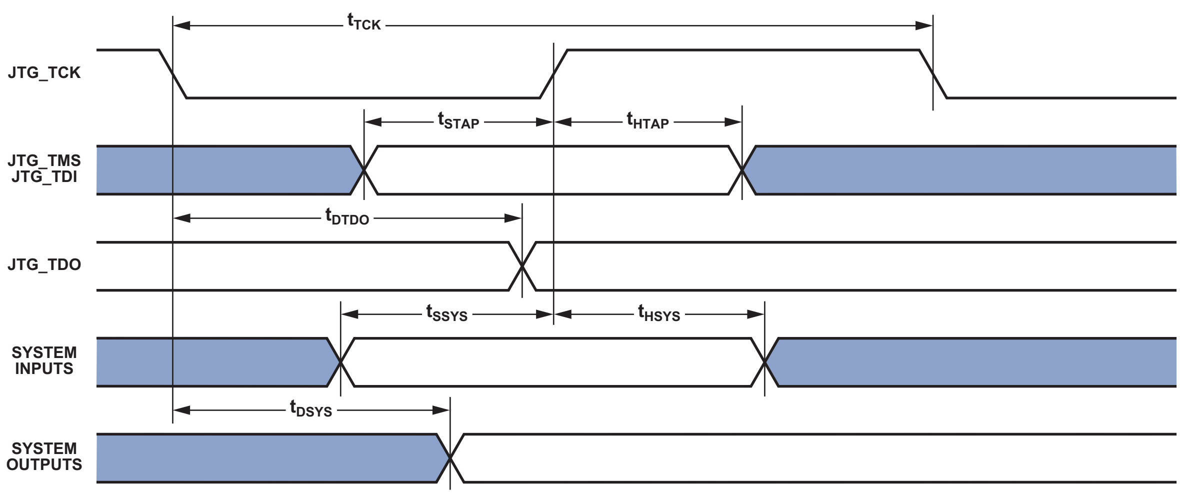

Figure 76. JTAG Port Timing