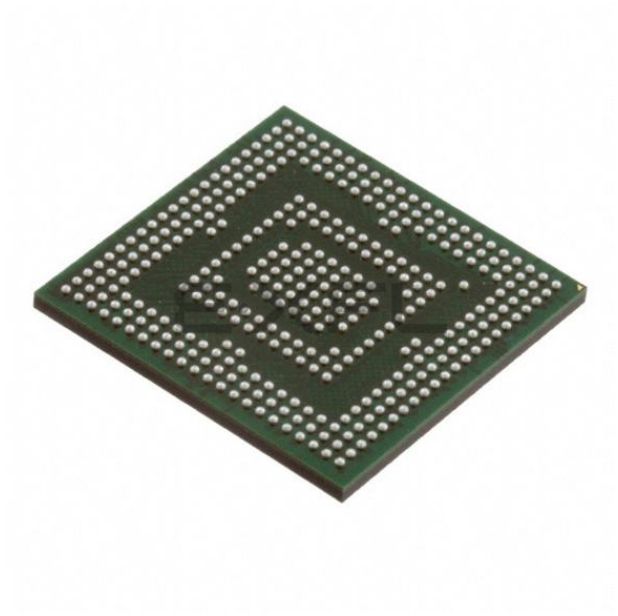


Welcome to E-XFL.COM

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	384kB
Voltage - I/O	3.30V
Voltage - Core	1.10V
Operating Temperature	-40°C ~ 95°C (TA)
Mounting Type	Surface Mount
Package / Case	349-LFBGA, CSPBGA
Supplier Device Package	349-CSPBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-sc583cbcz-4a

GENERAL DESCRIPTION

The ADSP-SC58x/ADSP-2158x processors are members of the SHARC® family of products. The ADSP-SC58x processor is based on the SHARC+ dual core and the ARM® Cortex®-A5 core. The ADSP-SC58x/ADSP-2158x 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 set of industry leading system peripherals and memory (see Table 1, Table 2, and Table 3), the ARM Cortex-A5 and 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 2 provides comparison information for features that vary across the standard processors. (N/A in the table means not applicable.)

Table 3 provides comparison information for features that vary across the automotive processors. (N/A in the table means not applicable.)

Table 1. Common Product Features

Product Features	ADSP-SC58x/ADSP-2158x
DAI (includes SRU)	2
Full SPORTs	4 per DAI
S/PDIF receive/transmit	1per DAI
ASRCs	4 pair per DAI
PCGs	2 per DAI
I^2C (TWI)	3
Quad-data bit SPI	1
Dual-data bit SPI	2
CAN2.0	2
UARTs	3
Link ports	2
Enhanced PPI	1
GP timer[1]	8
GP counter	1
Enhanced PWMs[2]	3
Watchdog timers	2
ADC control module	Yes
Static memory controller	Yes
Hardware accelerators	
High performance FFT/IFFT	Yes
FIR/IIR	Yes
Harmonic analysis engine	Yes
SINC filter	Yes
Security cryptographic engine	Yes
Multichannel 12-bit ADC	8-channel

[1] Eight timers are available in the 529-BGA package only. The 349-BGA package does not include Timer 6 and 7.

[2] Three 3ePWMs are available in the 529-BGA package only. The 349-BGA package does not include PWM 2.

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.

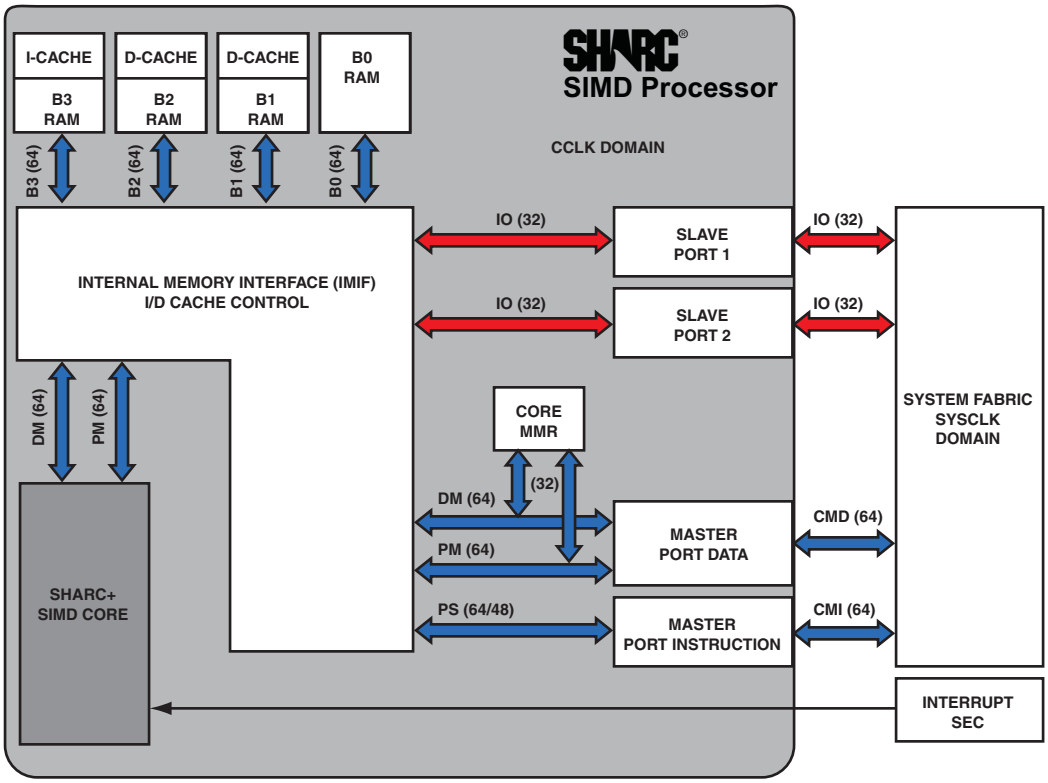


Figure 3. SHARC Processor Block Diagram

ADC Control Module (ACM) Interface

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

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

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

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

3-Phase Pulse Width Modulator (PWM) Units

The pulse width modulator (PWM) module is a flexible and programmable waveform generator. With minimal CPU intervention, the PWM generates complex waveforms for motor control, pulse coded modulation (PCM), DAC conversions, power switching, and power conversion. The PWM module has four PWM pairs capable of 3-phase PWM generation for source inverters for ac induction and dc brushless motors.

Each of the three 3-phase PWM generation units features the following:

- 16-bit center-based PWM generation unit
- Programmable PWM pulse width
- Single update mode with an option for asymmetric duty
- Programmable dead time and switching frequency
- Programmable dead time per channel
- Twos complement implementation which permits smooth transition to full on and full off states
- Dedicated asynchronous PWM shutdown signal

Ethernet Media Access Controller (EMAC)

The processor features two ethernet media access controllers (EMACs): 10/100 Ethernet and 10/100/1000/AVB Ethernet with precision time protocol IEEE 1588.

The processors can directly connect to a network through embedded fast EMAC that supports 10-BaseT (10 Mb/sec), 100-BaseT (100 Mb/sec) and 1000-BaseT (1 Gb/sec) operations. The 10/100 EMAC peripheral on the processors is fully compliant to the IEEE 802.3-2002 standard. The peripheral provides programmable features designed to minimize supervision, bus use, or message processing by the rest of the processor system.

Some standard features of the EMAC are as follows:

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

Some advanced features of the EMAC are as follows:

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

Audio Video Bridging (AVB) Support (10/100/1000 EMAC Only)

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

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

Precision Time Protocol (PTP) IEEE 1588 Support

The IEEE 1588 standard is a precision clock synchronization protocol for networked measurement and control systems. The processors include hardware support for IEEE 1588 with an integrated precision time protocol synchronization engine (PTP_TSYNC).

This engine provides hardware assisted time stamping to improve the accuracy of clock synchronization between PTP nodes. The main features of the engine are as follows:

- Support for both IEEE 1588-2002 and IEEE 1588-2008 protocol standards
- Hardware assisted time stamping capable of up to 12.5 ns resolution
- Lock adjustment

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

Signal Name	Direction	Description
SYS_FAULT	InOut	**Active-High Fault Output.** Indicates internal faults or senses external faults depending on the operating mode.
$\overline{\text{SYS\_FAULT}}$	InOut	**Active-Low Fault Output.** Indicates internal faults or senses external faults depending on the operating mode.
$\overline{\text{SYS\_HWRST}}$	Input	**Processor Hardware Reset Control.** Resets the device when asserted.
$\overline{\text{SYS\_RESOUT}}$	Output	**Reset Output.** Indicates the device is in the reset state.
SYS_XTAL0	Output	**Crystal Output.**
SYS_XTAL1	Output	**Crystal Output.**
TM_ACI[n]	Input	**Alternate Capture Input n.** Provides an additional input for WIDCAP, WATCHDOG, and PININT modes.
TM_ACLK[n]	Input	**Alternate Clock n.** Provides an additional time base for 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.
USB_CLKIN	Input	**Clock/Crystal Input.** This clock input is multiplied by a PLL to form the USB clock. See data sheet specifications for frequency/tolerance information.
USB_DM	InOut	**Data –.** Bidirectional differential data line.
USB_DP	InOut	**Data +.** Bidirectional differential data line.
USB_ID	Input	**OTG ID.** Senses whether the controller is a host or device. This signal is pulled low when an A-type plug is sensed (signifying that the USB controller is the A device).The input is high when a B-type plug is sensed (signifying that the USB controller is the B device).
USB_VBC	Output	**VBUS Control.** Controls an external voltage source to supply VBUS when in host mode. Can be configured as open-drain. Polarity is configurable as well.
USB_VBUS	InOut	**Bus Voltage.** Connects to bus voltage in host and device modes.
USB_XTAL	Output	**Crystal.** Drives an external crystal. Must be left unconnected if an external clock is driving USB_CLKIN.

Table 12. ADSP-SC58x/ADSP-2158x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SMC0_D02	SMC0 Data 2	E	PE_10
SMC0_D03	SMC0 Data 3	E	PE_09
SMC0_D04	SMC0 Data 4	E	PE_00
SMC0_D05	SMC0 Data 5	D	PD_15
SMC0_D06	SMC0 Data 6	D	PD_14
SMC0_D07	SMC0 Data 7	D	PD_00
SMC0_D08	SMC0 Data 8	B	PB_14
SMC0_D09	SMC0 Data 9	B	PB_13
SMC0_D10	SMC0 Data 10	B	PB_12
SMC0_D11	SMC0 Data 11	B	PB_11
SMC0_D12	SMC0 Data 12	B	PB_10
SMC0_D13	SMC0 Data 13	B	PB_09
SMC0_D14	SMC0 Data 14	B	PB_08
SMC0_D15	SMC0 Data 15	B	PB_07
SPI0_CLK	SPI0 Clock	C	PC_09
SPI0_MISO	SPI0 Master In, Slave Out	C	PC_10
SPI0_MOSI	SPI0 Master Out, Slave In	C	PC_11
SPI0_RDY	SPI0 Ready	C	PC_12
$\overline{\text{SPI0\_SEL1}}$	SPI0 Slave Select Output 1	C	PC_07
$\overline{\text{SPI0\_SEL2}}$	SPI0 Slave Select Output 2	D	PD_01
$\overline{\text{SPI0\_SEL3}}$	SPI0 Slave Select Output 3	C	PC_12
$\overline{\text{SPI0\_SEL4}}$	SPI0 Slave Select Output 4	C	PC_00
$\overline{\text{SPI0\_SEL5}}$	SPI0 Slave Select Output 5	E	PE_01
$\overline{\text{SPI0\_SEL6}}$	SPI0 Slave Select Output 6	E	PE_02
$\overline{\text{SPI0\_SEL7}}$	SPI0 Slave Select Output 7	E	PE_03
$\overline{\text{SPI0\_SS}}$	SPI0 Slave Select Input	D	PD_01
SPI1_CLK	SPI1 Clock	E	PE_13
SPI1_MISO	SPI1 Master In, Slave Out	E	PE_14
SPI1_MOSI	SPI1 Master Out, Slave In	E	PE_15
SPI1_RDY	SPI1 Ready	E	PE_08
$\overline{\text{SPI1\_SEL1}}$	SPI1 Slave Select Output 1	C	PC_13
$\overline{\text{SPI1\_SEL2}}$	SPI1 Slave Select Output 2	E	PE_07
$\overline{\text{SPI1\_SEL3}}$	SPI1 Slave Select Output 3	E	PE_11
$\overline{\text{SPI1\_SEL4}}$	SPI1 Slave Select Output 4	E	PE_12
$\overline{\text{SPI1\_SEL5}}$	SPI1 Slave Select Output 5	E	PE_08
$\overline{\text{SPI1\_SS}}$	SPI1 Slave Select Input	E	PE_11
SPI2_CLK	SPI2 Clock	C	PC_01
SPI2_D2	SPI2 Data 2	C	PC_04
SPI2_D3	SPI2 Data 3	C	PC_05
SPI2_MISO	SPI2 Master In, Slave Out	C	PC_02
SPI2_MOSI	SPI2 Master Out, Slave In	C	PC_03
SPI2_RDY	SPI2 Ready	E	PE_12
$\overline{\text{SPI2\_SEL1}}$	SPI2 Slave Select Output 1	C	PC_06
$\overline{\text{SPI2\_SEL2}}$	SPI2 Slave Select Output 2	E	PE_03
$\overline{\text{SPI2\_SEL3}}$	SPI2 Slave Select Output 3	E	PE_04
$\overline{\text{SPI2\_SEL4}}$	SPI2 Slave Select Output 4	E	PE_05
$\overline{\text{SPI2\_SEL5}}$	SPI2 Slave Select Output 5	E	PE_06
$\overline{\text{SPI2\_SS}}$	SPI2 Slave Select Input	C	PC_06

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

Signal Name	Description	Port	Pin Name
DMC0_CK	DMC0 Clock (complement)	Not Muxed	DMC0_CK
DMC0_CS0	DMC0 Chip Select 0	Not Muxed	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
DMC0_LDQS	DMC0 Data Strobe for Lower Byte (complement)	Not Muxed	DMC0_LDQS
DMC0_ODT	DMC0 On-die termination	Not Muxed	DMC0_ODT
DMC0_RAS	DMC0 Row Address Strobe	Not Muxed	DMC0_RAS
DMC0_RESET	DMC0 Reset (DDR3 only)	Not Muxed	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
DMC0_UDQS	DMC0 Data Strobe for Upper Byte (complement)	Not Muxed	DMC0_UDQS
DMC0_VREF	DMC0 Voltage Reference	Not Muxed	DMC0_VREF
DMC0_WE	DMC0 Write Enable	Not Muxed	DMC0_WE
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 19. ADSP-SC58x/ADSP-2158x 529-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC1_BA2	DMC1 Bank Address 2	Not Muxed	DMC1_BA2
$\overline{\text{DMC1\_CAS}}$	DMC1 Column Address Strobe	Not Muxed	$\overline{\text{DMC1\_CAS}}$
DMC1_CK	DMC1 Clock	Not Muxed	DMC1_CK
DMC1_CKE	DMC1 Clock enable	Not Muxed	DMC1_CKE
$\overline{\text{DMC1\_CK}}$	DMC1 Clock (complement)	Not Muxed	$\overline{\text{DMC1\_CK}}$
$\overline{\text{DMC1\_CS0}}$	DMC1 Chip Select 0	Not Muxed	$\overline{\text{DMC1\_CS0}}$
DMC1_DQ00	DMC1 Data 0	Not Muxed	DMC1_DQ00
DMC1_DQ01	DMC1 Data 1	Not Muxed	DMC1_DQ01
DMC1_DQ02	DMC1 Data 2	Not Muxed	DMC1_DQ02
DMC1_DQ03	DMC1 Data 3	Not Muxed	DMC1_DQ03
DMC1_DQ04	DMC1 Data 4	Not Muxed	DMC1_DQ04
DMC1_DQ05	DMC1 Data 5	Not Muxed	DMC1_DQ05
DMC1_DQ06	DMC1 Data 6	Not Muxed	DMC1_DQ06
DMC1_DQ07	DMC1 Data 7	Not Muxed	DMC1_DQ07
DMC1_DQ08	DMC1 Data 8	Not Muxed	DMC1_DQ08
DMC1_DQ09	DMC1 Data 9	Not Muxed	DMC1_DQ09
DMC1_DQ10	DMC1 Data 10	Not Muxed	DMC1_DQ10
DMC1_DQ11	DMC1 Data 11	Not Muxed	DMC1_DQ11
DMC1_DQ12	DMC1 Data 12	Not Muxed	DMC1_DQ12
DMC1_DQ13	DMC1 Data 13	Not Muxed	DMC1_DQ13
DMC1_DQ14	DMC1 Data 14	Not Muxed	DMC1_DQ14
DMC1_DQ15	DMC1 Data 15	Not Muxed	DMC1_DQ15
DMC1_LDM	DMC1 Data Mask for Lower Byte	Not Muxed	DMC1_LDM
DMC1_LDQS	DMC1 Data Strobe for Lower Byte	Not Muxed	DMC1_LDQS
$\overline{\text{DMC1\_LDQS}}$	DMC1 Data Strobe for Lower Byte (complement)	Not Muxed	$\overline{\text{DMC1\_LDQS}}$
DMC1_ODT	DMC1 On-die termination	Not Muxed	DMC1_ODT
$\overline{\text{DMC1\_RAS}}$	DMC1 Row Address Strobe	Not Muxed	$\overline{\text{DMC1\_RAS}}$
$\overline{\text{DMC1\_RESET}}$	DMC1 Reset (DDR3 only)	Not Muxed	$\overline{\text{DMC1\_RESET}}$
DMC1_RZQ	DMC1 External calibration resistor connection	Not Muxed	DMC1_RZQ
DMC1_UDM	DMC1 Data Mask for Upper Byte	Not Muxed	DMC1_UDM
DMC1_UDQS	DMC1 Data Strobe for Upper Byte	Not Muxed	DMC1_UDQS
$\overline{\text{DMC1\_UDQS}}$	DMC1 Data Strobe for Upper Byte (complement)	Not Muxed	$\overline{\text{DMC1\_UDQS}}$
DMC1_VREF	DMC1 Voltage Reference	Not Muxed	DMC1_VREF
$\overline{\text{DMC1\_WE}}$	DMC1 Write Enable	Not Muxed	$\overline{\text{DMC1\_WE}}$
ETH0_CRS	ETH0 Carrier Sense/RMII Receive Data Valid	A	PA_07
ETH0_MDC	ETH0 Management Channel Clock	A	PA_02
ETH0_MDIO	ETH0 Management Channel Serial Data	A	PA_03
ETH0_PTPAUXIN0	ETH0 PTP Auxiliary Trigger Input 0	B	PB_03
ETH0_PTPAUXIN1	ETH0 PTP Auxiliary Trigger Input 1	B	PB_04
ETH0_PTPAUXIN2	ETH0 PTP Auxiliary Trigger Input 2	B	PB_05
ETH0_PTPAUXIN3	ETH0 PTP Auxiliary Trigger Input 3	B	PB_06
ETH0_PTPCLKIN0	ETH0 PTP Clock Input 0	B	PB_02
ETH0_PTPPPS0	ETH0 PTP Pulse-Per-Second Output 0	B	PB_01
ETH0_PTPPPS1	ETH0 PTP Pulse-Per-Second Output 1	B	PB_00
ETH0_PTPPPS2	ETH0 PTP Pulse-Per-Second Output 2	A	PA_15
ETH0_PTPPPS3	ETH0 PTP Pulse-Per-Second Output 3	A	PA_14
ETH0_RXCLK_REFCLK	ETH0 RXCLK (GigE) or REFCLK (10/100)	A	PA_06
ETH0_RXCTL_CRS	ETH0 RXCTL (GigE) or CRS (10/100)	A	PA_07

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
DMC0_UDQS	InOut	C	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte (complement) Notes: No notes
DMC0_VREF	a		none	none	none	VDD_DMC	Desc: DMC0 Voltage Reference Notes: No notes
DMC0_WE	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Write Enable Notes: No notes
DMC1_A00	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 0 Notes: No notes
DMC1_A01	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 1 Notes: No notes
DMC1_A02	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 2 Notes: No notes
DMC1_A03	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 3 Notes: No notes
DMC1_A04	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 4 Notes: No notes
DMC1_A05	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 5 Notes: No notes
DMC1_A06	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 6 Notes: No notes
DMC1_A07	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 7 Notes: No notes
DMC1_A08	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 8 Notes: No notes
DMC1_A09	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 9 Notes: No notes
DMC1_A10	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 10 Notes: No notes
DMC1_A11	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 11 Notes: No notes
DMC1_A12	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 12 Notes: No notes
DMC1_A13	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 13 Notes: No notes
DMC1_A14	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 14 Notes: No notes
DMC1_A15	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Address 15 Notes: No notes
DMC1_BA0	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Bank Address Input 0 Notes: No notes
DMC1_BA1	Output	B	none	none	none	VDD_DMC	Desc: DMC1 Bank Address Input 1 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
PE_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 2 | EPPI0 Frame Sync 1 (HSYNC) | SPI0 Slave Select Output 6 | SHARC Core 2 Flag Pin | UART1 Request to Send Notes: No notes
PE_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 3 | EPPI0 Clock | SPI0 Slave Select Output 7 | SPI2 Slave Select Output 2 | SHARC Core 1 Flag Pin Notes: No notes
PE_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 4 | EPPI0 Data 8 | PWM2 Channel D High Side | SPI2 Slave Select Output 3 | SHARC Core 2 Flag Pin Notes: No notes
PE_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 5 | EPPI0 Data 7 | PWM2 PWMTMR Grouped | SPI2 Slave Select Output 4 | SHARC Core 1 Flag Pin Notes: No notes
PE_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 6 | EPPI0 Data 6 | SPI2 Slave Select Output 5 | SHARC Core 2 Flag Pin Notes: No notes
PE_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 7 | EPPI0 Data 5 | SPI1 Slave Select Output 2 | SHARC Core 1 Flag Pin Notes: No notes
PE_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 8 | EPPI0 Data 4 | SPI1 Ready | SPI1 Slave Select Output 5 | SHARC Core 2 Flag Pin Notes: No notes
PE_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 9 | EPPI0 Data 3 | PWM0 PWMTMR Grouped | SMC0 Data 3 | TIMER0 Timer 0 Notes: No notes
PE_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 10 | EPPI0 Data 2 | PWM2 Channel D Low Side | SMC0 Data 2 | UART2 Request to Send Notes: No notes
PE_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 11 | EPPI0 Data 1 | SMC0 Data 1 | SPI1 Slave Select Output 3 | UART2 Clear to Send | SPI1 Slave Select Input Notes: No notes

Asynchronous Flash Read

Table 46 and Figure 13 show asynchronous flash memory read timing, related to the SMC.

Table 46. Asynchronous Flash Read

Parameter		Min	Max	Unit
Switching Characteristics				
t_{AMSADV}	SMC0_Ax (Address)/$\overline{\text{SMC0\_AMSx}}$ Assertion Before SMC0_NORDV Low[1]	PREST × t_{SCLK0} – 2		ns
t_{WADV}	SMC0_NORDV Active Low Width[2]	RST × t_{SCLK0} – 2		ns
$t_{DADVARE}$	$\overline{\text{SMC0\_ARE}}$ Low Delay From SMC0_NORDV High[3]	PREAT × t_{SCLK0} – 2		ns
t_{HARE}	Output[4] Hold After $\overline{\text{SMC0\_ARE}}$ High[5]	RHT × t_{SCLK0} – 2		ns
t_{WARE}[6]	$\overline{\text{SMC0\_ARE}}$ Active Low Width[7]	RAT × t_{SCLK0} – 2		ns

[1] PREST value set using the SMC_BxETIM.PREST bits.
[2] RST value set using the SMC_BxTIM.RST bits.
[3] PREAT value set using the SMC_BxETIM.PREAT bits.
[4] Output signals are SMC0_Ax, $\overline{\text{SMC0\_AMS}}$, $\overline{\text{SMC0\_AOE}}$.
[5] RHT value set using the SMC_BxTIM.RHT bits.
[6] SMC0_BxCTL.ARDYEN bit = 0.
[7] RAT value set using the SMC_BxTIM.RAT bits.

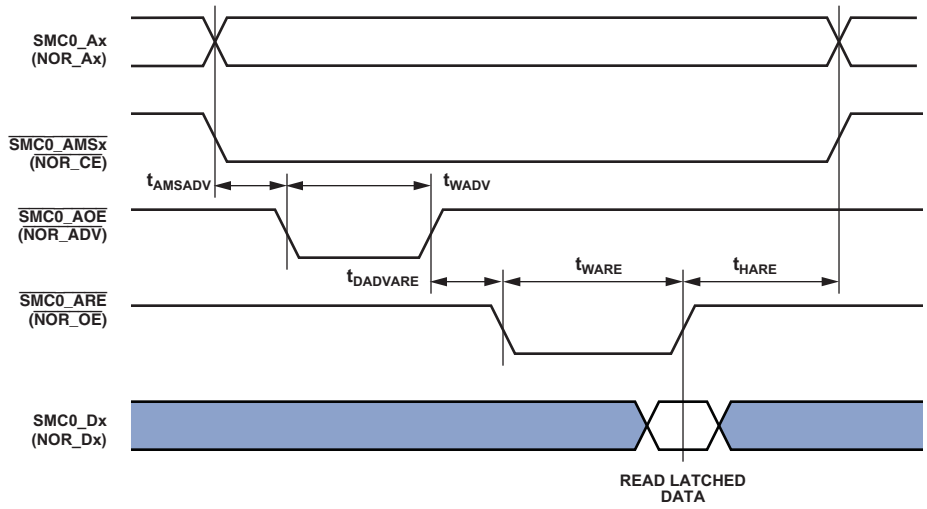


Figure 13. Asynchronous Flash Read

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.

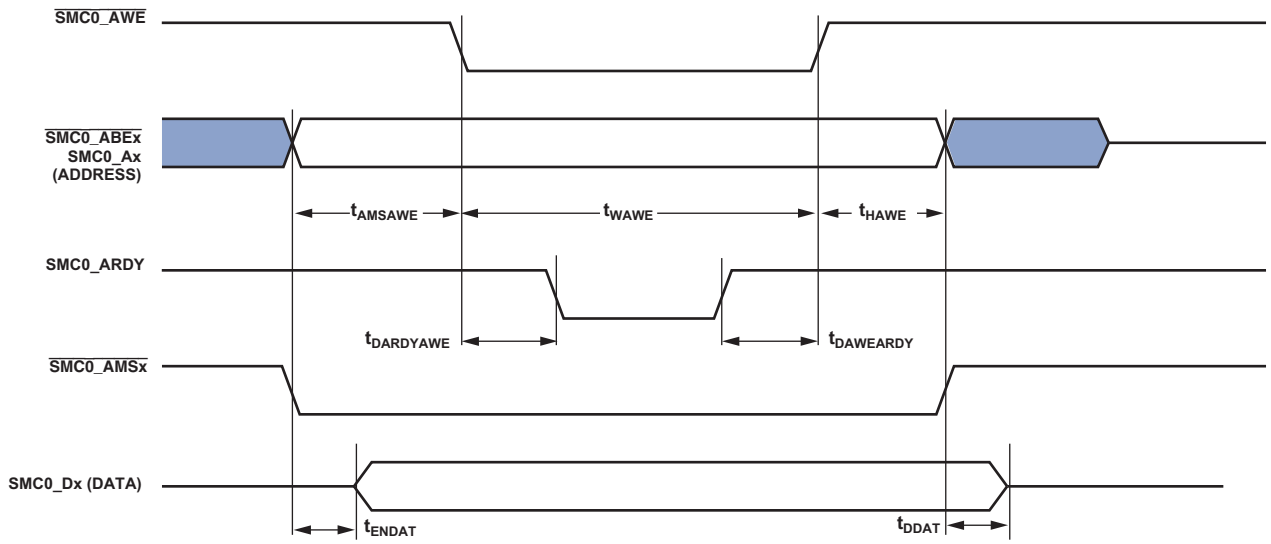


Figure 15. Asynchronous Write

DDR2 SDRAM Clock and Control Cycle Timing

Table 51 and Figure 17 show DDR2 SDRAM clock and control cycle timing, related to the DMC.

Table 51. DDR2 SDRAM Clock and Control Cycle Timing, V_{DD_DMCx} Nominal 1.8 V[1]

Parameter		400 MHz[2] Min	400 MHz[2] Max	Unit
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	2.5		ns
$t_{CH\ (abs)}$[3]	Minimum Clock Pulse Width	0.44	0.56	t_{CK}
$t_{CL\ (abs)}$[3]	Maximum Clock Pulse Width	0.44	0.56	t_{CK}
t_{IS}	Control/Address Setup Relative to DMCx_CK Rise	175		ps
t_{IH}	Control/Address Hold Relative to DMCx_CK Rise	250		ps

[1] Specifications apply to both DMC0 and DMC1.
[2] In order to ensure proper operation of the DDR2, all the DDR2 guidelines must be strictly followed. See "Interfacing DDR3/DDR2/LPDDR Memory to ADSP-SC5xx/215xx Processors" (EE-387).
[3] As per JESD79-2E definition.

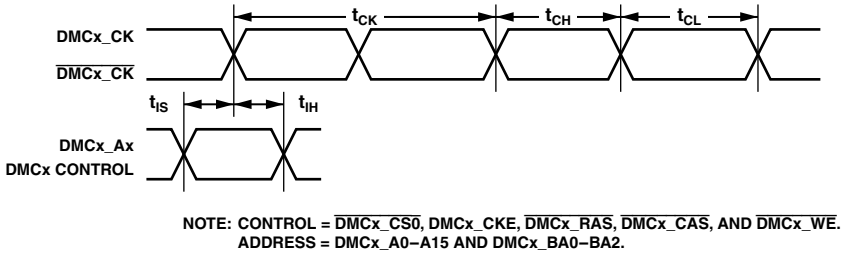


Figure 17. DDR2 SDRAM Clock and Control Cycle Timing

DDR2 SDRAM Write Cycle Timing

Table 53 and Figure 19 show DDR2 SDRAM write cycle timing, related to the DMC.

Table 53. DDR2 SDRAM Write Cycle Timing, V_{DD_DMCx} Nominal 1.8 V[1]

Parameter		400 MHz[2] Min	400 MHz[2] Max	Unit
Switching Characteristics				
t_{DQSS}	DMCx_DQS Latching Rising Transitions to Associated Clock Edges[3]	–0.15	+0.15	t_{CK}
t_{DS}	Last Data Valid to DMCx_DQS Delay	0.1		ns
t_{DH}	DMCx_DQS to First Data Invalid Delay	0.15		ns
t_{DSS}	DMCx_DQS Falling Edge to Clock Setup Time	0.2		t_{CK}
t_{DSH}	DMCx_DQS Falling Edge Hold Time From DMCx_CK	0.2		t_{CK}
t_{DQSH}	DMCx_DQS Input High Pulse Width	0.35		t_{CK}
t_{DQSL}	DMCx_DQS Input Low Pulse Width	0.35		t_{CK}
t_{WPRE}	Write Preamble	0.35		t_{CK}
t_{WPST}	Write Postamble	0.4		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	0.6		t_{CK}
t_{DIPW}	DMCx_DQ and DMCx_DM Output Pulse Width	0.35		t_{CK}

[1] Specifications apply to both DMC0 and DMC1.
[2] To ensure proper operation of the DDR2, all the DDR2 guidelines must be strictly followed. See "Interfacing DDR3/DDR2/LPDDR Memory to ADSP-SC5xx/215xx Processors" (EE-387).
[3] Write command to first DMCx_DQS delay = WL × t_{CK} + t_{DQSS}.

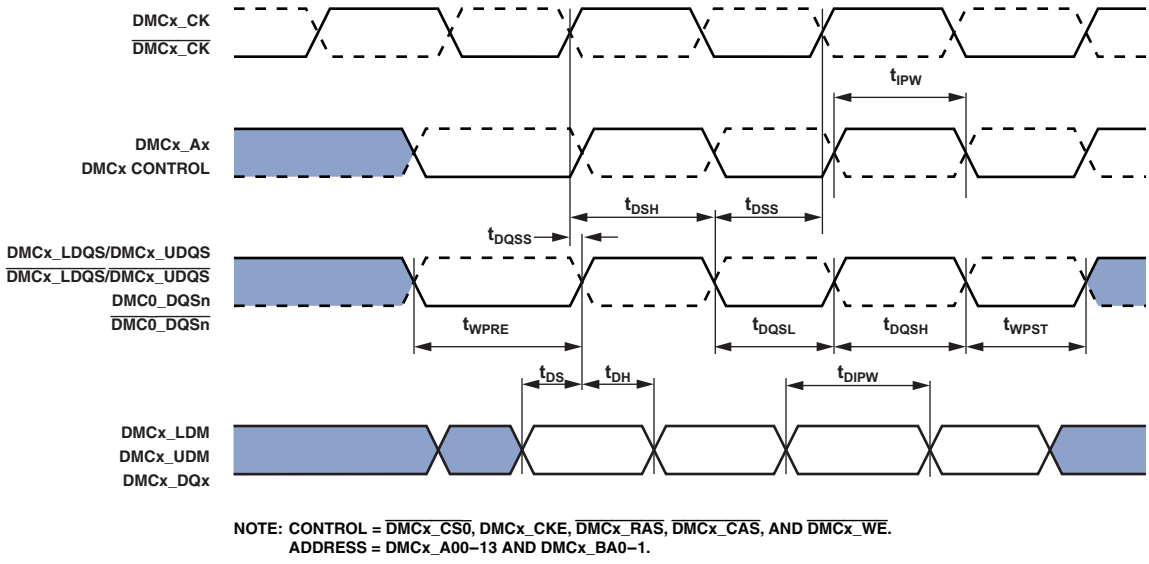
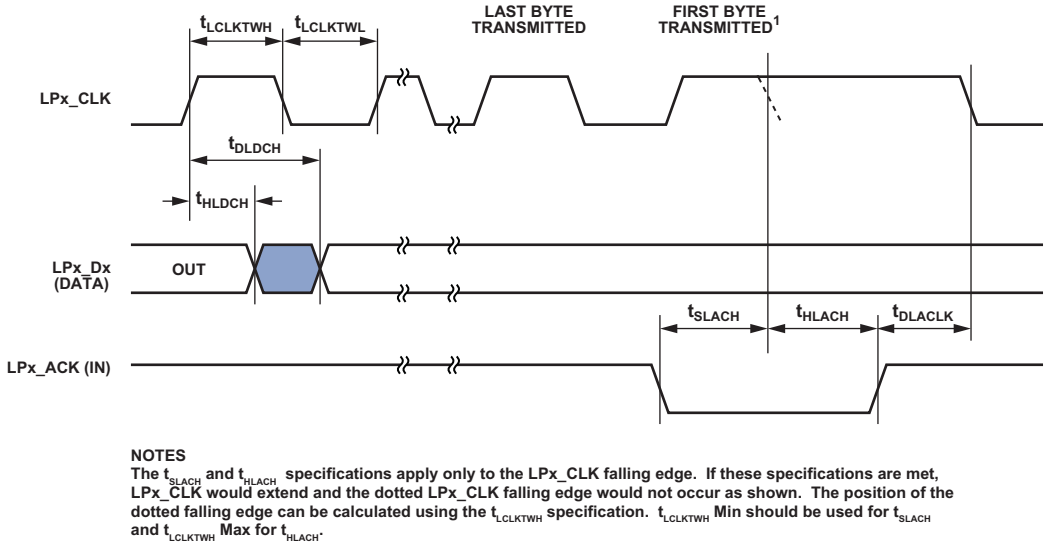


Figure 19. DDR2 SDRAM Controller Output AC Timing

Table 63. Link Ports—Transmit[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SLACH}	LPx_ACK Setup Before LPx_CLK Low	$2 \times t_{CLKO8} + 13.5$		ns
t_{HLACH}	LPx_ACK Hold After LPx_CLK Low	–5.5		ns
Switching Characteristics				
t_{DLDCH}	Data Delay After LPx_CLK High		1.6	ns
t_{HLDCH}	Data Hold After LPx_CLK High	–0.8		ns
$t_{LCLKTWL}$[2]	LPx_CLK Width Low	$0.33 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
$t_{LCLKTWH}$[2]	LPx_CLK Width High	$0.45 \times t_{LCLKTPROG}$	$0.66 \times t_{LCLKTPROG}$	ns
t_{LCLKTW}[2]	LPx_CLK Period	$N \times t_{LCLKTPROG} - 0.5$		ns
t_{DLACLK}	LPx_CLK Low Delay After LPx_ACK High	$t_{CLKO8} + 4$	$2 \times t_{CLKO8} + 1 \times t_{LPCLK} + 10$	ns

[1] Specifications apply to LP0 and LP1.
[2] See Table 29 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 would extend and the dotted LPx_CLK falling edge would not occur as shown. The position of the dotted falling edge can be calculated using the $t_{LCLKTWH}$ specification. $t_{LCLKTWH}$ Min should be used for t_{SLACH} and $t_{LCLKTWH}$ Max for t_{HLACH}.

Figure 36. Link Ports—Transmit

Table 66. Serial Ports—Enable and Three-State[1]

Parameter		Min	Max	Unit
Switching Characteristics				
t_{DDTEN}	Data Enable from External Transmit SPTx_CLK[2]	1		ns
t_{DDTTE}	Data Disable from External Transmit SPTx_CLK[2]		14	ns
t_{DDTIN}	Data Enable from Internal Transmit SPTx_CLK[2]	–2.5		ns
t_{DDTTI}	Data Disable from Internal Transmit SPTx_CLK[2]		2.8	ns

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

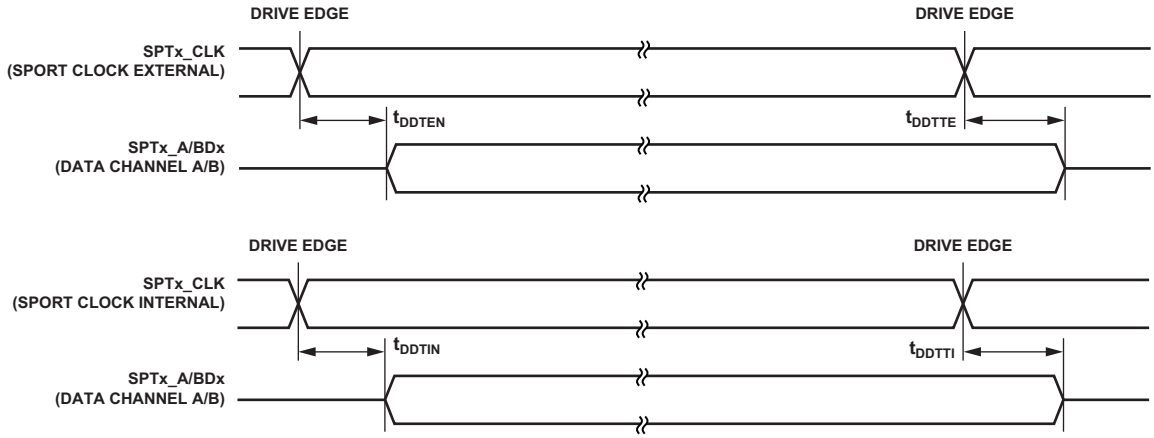


Figure 38. Serial Ports—Enable and Three-State

Sample Rate Converter—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 70. ASRC, Serial Output Port

Parameter		Min	Max	Unit
Timing Requirements				
t_{SRCSFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	4		ns
t_{SRCHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	5.5		ns
$t_{SRCCLKW}$	Clock Width	$t_{SCLK0} - 1$		ns
t_{SRCCLK}	Clock Period	$2 \times t_{SCLK0}$		ns
Switching Characteristics				
t_{SRCTDD}[1]	Transmit Data Delay After Serial Clock Falling Edge		13	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.

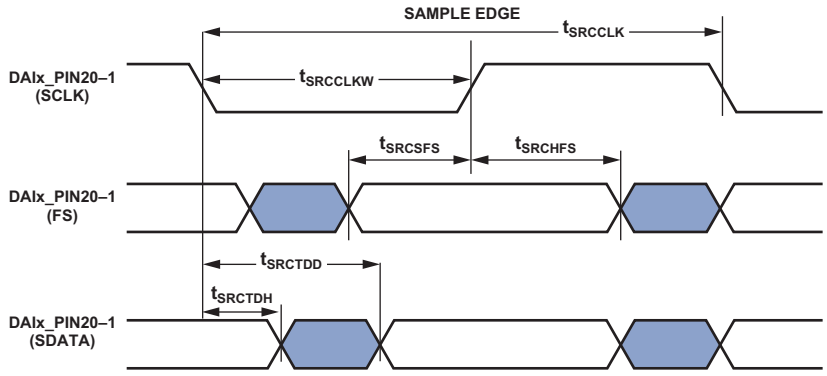


Figure 42. ASRC Serial Output Port Timing

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 (DAIx_PINx).

Table 77. Precision Clock Generator (Direct Pin Routing)

Parameter		Min	Max	Unit
Timing Requirements				
t_{PCGIP}	Input Clock Period	$t_{SCLK} \times 2$		ns
t_{STRIG}	PCG Trigger Setup Before Falling Edge of PCG Input Clock	4.5		ns
t_{HTRIG}	PCG Trigger Hold After Falling Edge of PCG Input Clock	3		ns
Switching Characteristics				
t_{DPCGIO}	PCG Output Clock and Frame Sync Active Edge Delay After PCG Input Clock	2.5	13.5	ns
$t_{DTRIGCLK}$	PCG Output Clock Delay After PCG Trigger	$2.5 + (2.5 \times t_{PCGIP})$	$13.5 + (2.5 \times t_{PCGIP})$	ns
$t_{DTRIGFS}$[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-SC58x/ADSP-2158x SHARC+ Processor Hardware Reference.
[2] Normal mode of operation.

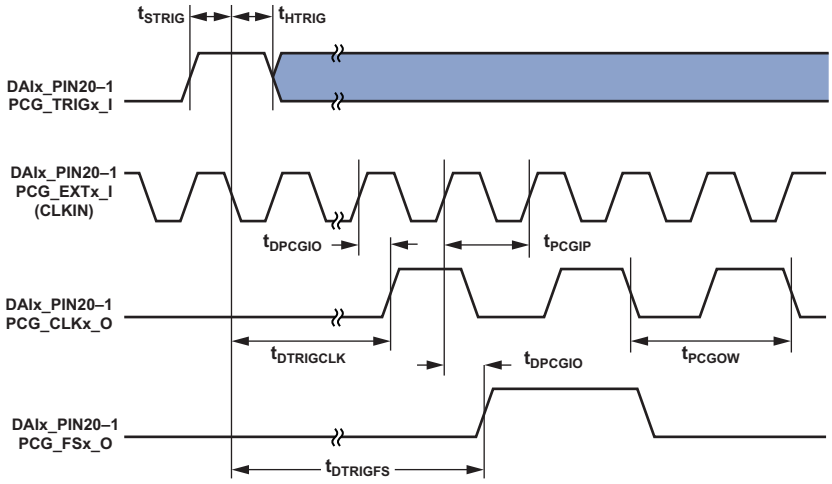


Figure 50. PCG (Direct Pin Routing)

Gigabit EMAC Timing (ETH0 Only)

Table 91 and Figure 62 describe the gigabit EMAC timing.

Table 91. Gigabit Ethernet MAC Controller (EMAC) Timing: RGMII [1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SETUPR}	Data to Clock Input Setup at Receiver	1		ns
t_{HOLDR}	Data to Clock Input Hold at Receiver	1		ns
$t_{GREFCLKF}$	RGMII Receive Clock Period	8		ns
$t_{GREFCLKW}$	RGMII Receive Clock Pulse Width	4		ns
Switching Characteristics				
t_{SKEWT}	Data to Clock Output Skew at Transmitter	–0.5	0.5	ns
t_{CYC}	Clock Cycle Duration	7.2	8.8	ns
t_{DUTY_G}	Duty Cycle for Gigabit Minimum	$t_{GREFCLKF} \times 45\%$	$t_{GREFCLKF} \times 55\%$	ns

[1] This specification is supported by ETH0 only (10/100/1000 EMAC controller).

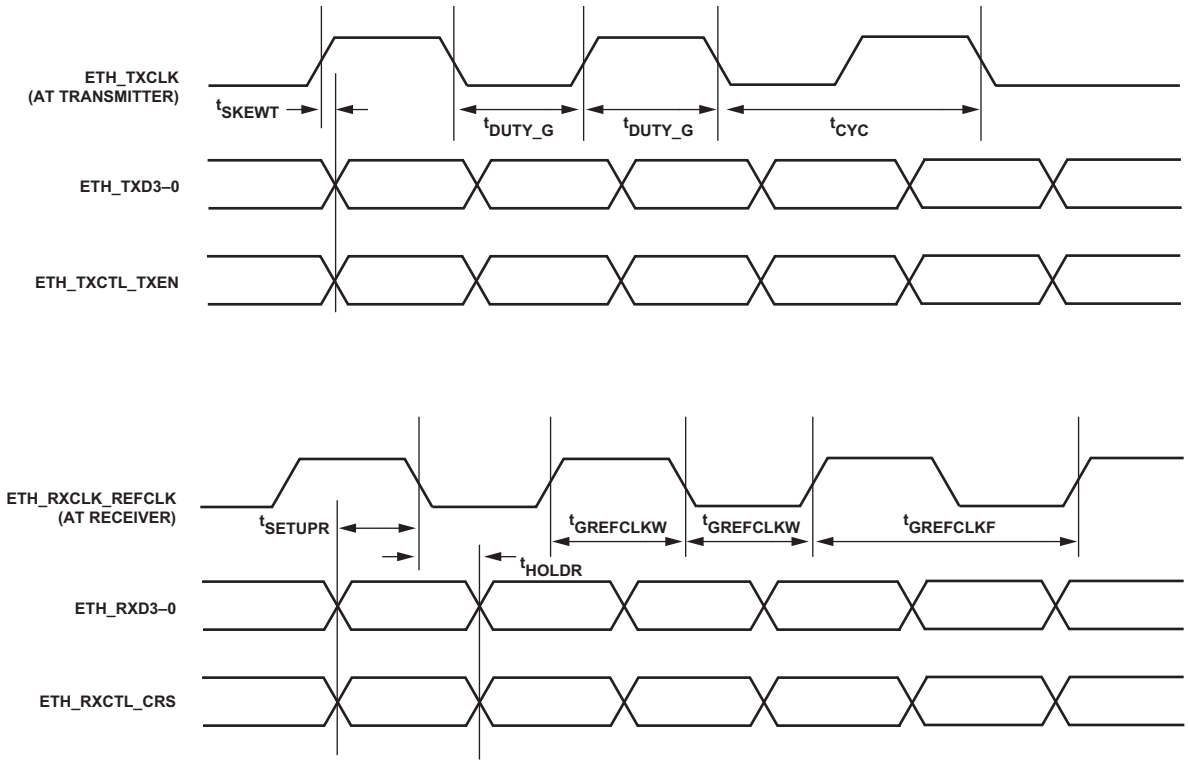


Figure 62. Gigabit EMAC Controller Timing—RGMII

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, $\overline{\text{DMC0\_RESET}}$, $\overline{\text{DMC1\_RESET}}$, 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, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_RESOUT}}$, TWI2-0_SCL, TWI2-0_SDA2.

[2] 50 MHz maximum.

[3] System Outputs = DMC0_A15-0, DMC0_BA2-0, $\overline{\text{DMC0\_CAS}}$, DMC0_CK, DMC0_CKE, $\overline{\text{DMC0\_CS0}}$, DMC0_DQ15-0, DMC0_LDM, DMC0_LDQS, DMC0_ODT, $\overline{\text{DMC0\_RAS}}$, $\overline{\text{DMC0\_RESET}}$, DMC0_UDM, DMC0_UDQS, $\overline{\text{DMC0\_WE}}$, DMC1_A15-0, DMC1_BA2-0, $\overline{\text{DMC1\_CAS}}$, DMC1_CK, DMC1_CKE, $\overline{\text{DMC1\_CS0}}$, DMC1_DQ15-0, DMC1_LDM, DMC1_LDQS, DMC1_ODT, $\overline{\text{DMC1\_RAS}}$, $\overline{\text{DMC1\_RESET}}$, DMC1_UDM, DMC1_UDQS, $\overline{\text{DMC1\_WE}}$, 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, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_RESOUT}}$.

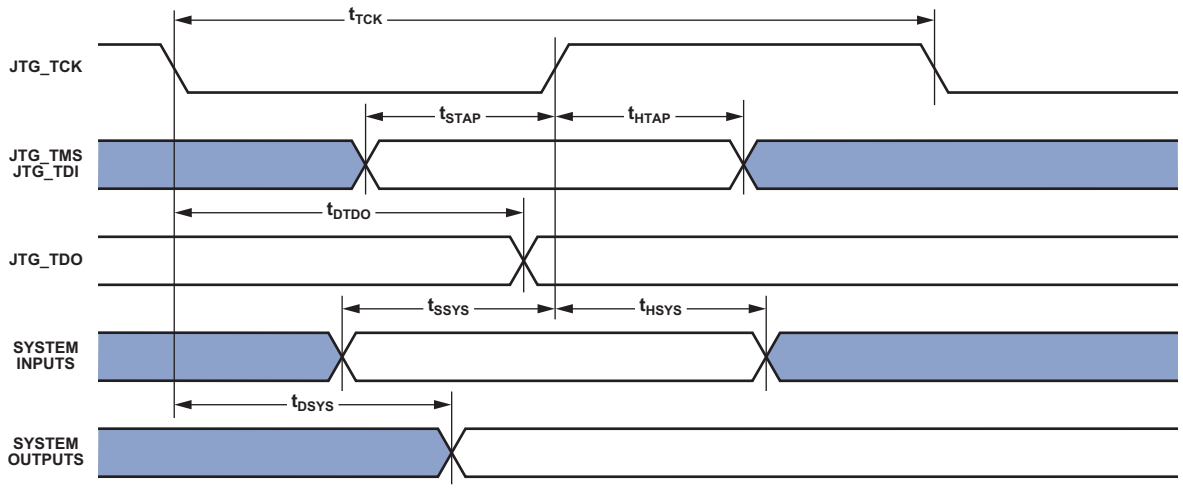


Figure 76. JTAG Port Timing

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
GND	L16	GND	T08	PA_02	AB23	PC_11	K02
GND	L17	GND	T09	PA_03	AC22	PC_12	L02
GND	M07	GND	T10	PA_04	AB22	PC_13	C20
GND	M08	GND	T11	PA_05	AA20	PC_14	D21
GND	M09	GND	T12	PA_06	AB21	PC_15	E20
GND	M10	GND	T13	PA_07	AC20	PD_00	B22
GND	M11	GND	T14	PA_08	AC21	PD_01	C21
GND	M12	GND	T15	PA_09	AA19	PD_02	F21
GND	M13	GND	T16	PA_10	Y19	PD_03	J19
GND	M14	GND	T17	PA_11	AB20	PD_04	B23
GND	M15	GND	U07	PA_12	Y18	PD_05	C23
GND	M16	GND	U08	PA_13	Y17	PD_06	C22
GND	M17	GND	U09	PA_14	Y16	PD_07	J20
GND	N07	GND	U10	PA_15	AB19	PD_08	E21
GND	N08	GND	U11	PB_00	AA18	PD_09	D23
GND	N09	GND	U12	PB_01	AC19	PD_10	D22
GND	N10	GND	U13	PB_02	AA15	PD_11	E23
GND	N11	GND	U14	PB_03	AA17	PD_12	F23
GND	N12	GND	U15	PB_04	AA16	PD_13	F22
GND	N13	GND	U16	PB_05	Y15	PD_14	E22
GND	N14	GND	U17	PB_06	AA14	PD_15	K20
GND	N15	GND	Y14	PB_07	AA02	PE_00	G23
GND	N16	GND	AC01	PB_08	AA01	PE_01	G22
GND	N17	GND	AC14	PB_09	W02	PE_02	H23
GND	P07	GND	AC23	PB_10	Y02	PE_03	L20
GND	P08	HADC0_VIN0	Y12	PB_11	Y01	PE_04	G20
GND	P09	HADC0_VIN1	AA12	PB_12	W01	PE_05	H22
GND	P10	HADC0_VIN2	AB13	PB_13	V02	PE_06	F20
GND	P11	HADC0_VIN3	AB14	PB_14	T04	PE_07	J23
GND	P12	HADC0_VIN4	V12	PB_15	T02	PE_08	M19
GND	P13	HADC0_VIN5	AA13	PCIE0_CLKM	AC04	PE_09	L22
GND	P14	HADC0_VIN6	W12	PCIE0_CLKP	AC05	PE_10	K23
GND	P15	HADC0_VIN7	Y13	PCIE0_REF	AA07	PE_11	M20
GND	P16	HADC0_VREFN	AB12	PCIE0_RXM	AC03	PE_12	H21
GND	P17	HADC0_VREFP	AC12	PCIE0_RXP	AC02	PE_13	G21
GND	R07	JTG_TCK	P04	PCIE0_TXM	AC07	PE_14	L23
GND	R08	JTG_TDI	P02	PCIE0_TXP	AC06	PE_15	N20
GND	R09	JTG_TDO	P01	PC_00	U03	PF_00	M22
GND	R10	JTG_TMS	N01	PC_01	M01	PF_01	J22
GND	R11	$\overline{\text{JTG\_TRST}}$	N02	PC_02	M03	PF_02	M23
GND	R12	MLB0_CLKN	AB18	PC_03	N04	PF_03	M21
GND	R13	MLB0_CLKP	AC18	PC_04	L01	PF_04	N21
GND	R14	MLB0_DATN	AB17	PC_05	M02	PF_05	N22
GND	R15	MLB0_DATP	AC17	PC_06	K03	PF_06	K22
GND	R16	MLB0_SIGN	AB16	PC_07	L03	PF_07	N23
GND	R17	MLB0_SIGP	AC16	PC_08	J04	PF_08	P20
GND	T03	PA_00	Y20	PC_09	K04	PF_09	L21
GND	T07	PA_01	AA21	PC_10	L04	PF_10	P19