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	500MHz
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-sc589bbcz-5b

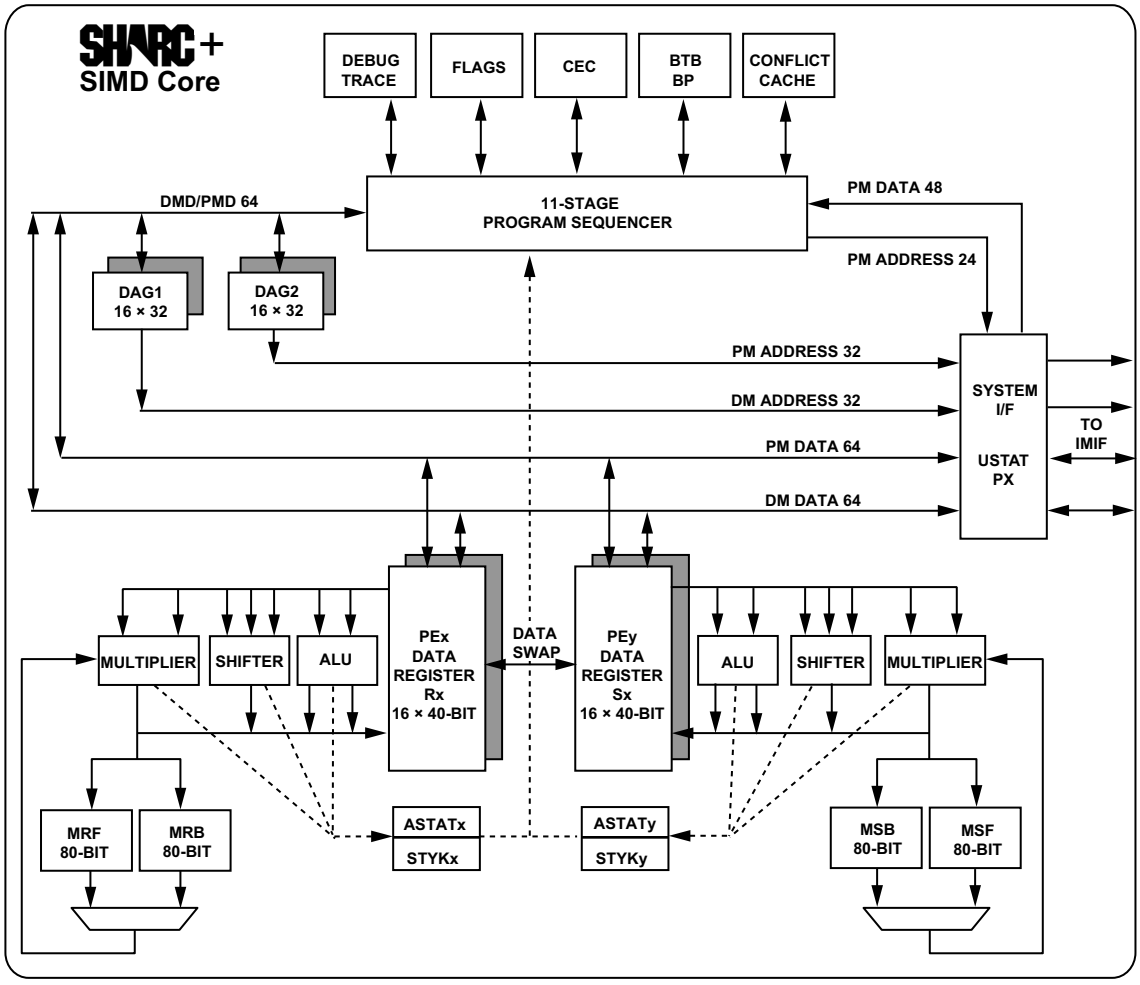

Figure 4. SHARC+ SIMD Core Block Diagram

L1 Memory

Figure 5 shows the ADSP-SC58x/ADSP-2158x memory map. Each SHARC+ core has a tightly coupled L1 SRAM of up to 5 Mb. Each SHARC+ core can access code and data in a single cycle from this memory space. The ARM Cortex-A5 core can also access this memory space with multicycle accesses.

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

SHARC+ core memory-mapped register (CMMR) address space is 0x 0000 0000 through 0x 0003 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 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 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 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.

Single-Cycle Fetch of Instructional Four Operands

The ADSP-SC58x/ADSP-2158x 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/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 fast Fourier transforms (FFT) butterfly processing. The conflict cache serves for on-chip bus conflicts only.

Branch Target Buffer/Branch Predictor

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

The enhanced ISA/VISA of the ADSP-SC58x/ADSP-2158x processors also provides a memory barrier instruction for data synchronization, exclusive data access support for multicore data sharing, and exclusive data access to enable multiprocessor programming. To enhance the reliability of the application, L1 data RAMs support parity error detection 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-SC58x/ADSP-2158x processors.

System L2 Memory

A system L2 SRAM memory of 2 Mb (256 kB) and two ROM memories, each 2 Mb (256 kB), are available to both SHARC+ cores, the ARM Cortex-A5 core, and the system DMA channels (see Table 5). All L2 SRAM/ROM blocks are subdivided into eight banks to support concurrent access to the L2 memory ports. Memory accesses to the L2 memory space are multicycle accesses by both the ARM Cortex-A5 and SHARC+ cores.

The memory space is used for various cases including:

- ARM Cortex-A5 to SHARC+ core data sharing and inter-core communications
- Accelerator and peripheral sources and destination memory to avoid accessing data in the external memory
- A location for DMA descriptors
- Storage for additional data for either the ARM Cortex-A5 or SHARC+ cores to avoid external memory latencies and reduce external memory bandwidth
- Storage for incoming Ethernet traffic to improve performance
- Storage for data coefficient tables cached by the SHARC+ core

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

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

SHARC+ Core L1 Memory in Multiprocessor Space

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

One Time Programmable Memory (OTP)

The processors feature 7 Kb of one time programmable (OTP) memory which is memory-map accessible. This memory stores a unique chip identification and supports secure boot and secure operation.

I/O Memory Space

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

blocks on the processor. The digital audio interface carries three types of information: audio data, nonaudio data (compressed data), and timing information.

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

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

Precision Clock Generators (PCG)

The precision clock generators (PCG) consist of 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 (CLKIN1-0, SCLK0, or DAI pin buffer). Each unit can also access 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.

Enhanced Parallel Peripheral Interface (EPPI)

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

The features supported in the EPPI module include the following:

- Programmable data length of 8 bits, 10 bits, 12 bits, 14 bits, 16 bits, 18 bits, and 24 bits per clock.
- Various framed, nonframed, and general-purpose operating modes. Frame syncs can be generated internally or can be supplied by an external device.
- ITU-656 status word error detection and correction for ITU-656 receive modes and ITU-656 preamble and status word decoding.
- Optional packing and unpacking of data to/from 32 bits from/to 8 bits, 16 bits, and 24 bits. If packing/unpacking is enabled, configure endianness to change the order of packing/unpacking of the bytes/words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various deinterleaving/interleaving modes for receiving/transmitting 4:2:2 YCrCb data.
- Configurable LCD data enable output available on Frame Sync 3.

Universal Asynchronous Receiver/Transmitter (UART) Ports

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

The baseline SPI peripheral is a synchronous, four-wire interface consisting of two data pins, one device select pin, and a gated clock pin. The two data pins allow full-duplex operation to other SPI-compatible devices. An 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. A direct memory access (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/polarities of the SPI port are programmable. The port has integrated DMA channels for both transmit and receive data streams.

Link Ports (LP)

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

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

Signal Name	Description	Port	Pin Name
PWM2_DL	PWM2 Channel D Low Side	E	PE_10
PWM2_SYNC	PWM2 PWMTMR Grouped	E	PE_05
$\overline{\text{PWM2\_TRIP0}}$	PWM2 Shutdown Input 0	D	PD_14
GND	Ground	Not Muxed	GND
VDD_EXT	External Voltage Domain	Not Muxed	VDD_EXT
VDD_INT	Internal Voltage Domain	Not Muxed	VDD_INT
SINC0_CLK0	SINC0 Clock 0	B	PB_01
SINC0_D0	SINC0 Data 0	A	PA_14
SINC0_D1	SINC0 Data 1	A	PA_15
SINC0_D2	SINC0 Data 2	B	PB_00
SINC0_D3	SINC0 Data 3	B	PB_04
SMC0_A01	SMC0 Address 1	B	PB_05
SMC0_A02	SMC0 Address 2	B	PB_06
SMC0_A03	SMC0 Address 3	B	PB_03
SMC0_A04	SMC0 Address 4	B	PB_02
SMC0_A05	SMC0 Address 5	D	PD_13
SMC0_A06	SMC0 Address 6	D	PD_12
SMC0_A07	SMC0 Address 7	B	PB_01
SMC0_A08	SMC0 Address 8	B	PB_00
SMC0_A09	SMC0 Address 9	A	PA_15
SMC0_A10	SMC0 Address 10	A	PA_14
SMC0_A11	SMC0 Address 11	A	PA_09
SMC0_A12	SMC0 Address 12	A	PA_08
SMC0_A13	SMC0 Address 13	A	PA_13
SMC0_A14	SMC0 Address 14	A	PA_12
SMC0_A15	SMC0 Address 15	A	PA_11
SMC0_A16	SMC0 Address 16	A	PA_07
SMC0_A17	SMC0 Address 17	A	PA_06
SMC0_A18	SMC0 Address 18	A	PA_05
SMC0_A19	SMC0 Address 19	A	PA_04
SMC0_A20	SMC0 Address 20	A	PA_01
SMC0_A21	SMC0 Address 21	A	PA_00
SMC0_A22	SMC0 Address 22	A	PA_10
SMC0_A23	SMC0 Address 23	A	PA_03
SMC0_A24	SMC0 Address 24	A	PA_02
SMC0_A25	SMC0 Address 25	C	PC_12
$\overline{\text{SMC0\_ABE0}}$	SMC0 Byte Enable 0	E	PE_14
$\overline{\text{SMC0\_ABE1}}$	SMC0 Byte Enable 1	E	PE_15
$\overline{\text{SMC0\_AMS0}}$	SMC0 Memory Select 0	C	PC_15
$\overline{\text{SMC0\_AMS1}}$	SMC0 Memory Select 1	E	PE_13
$\overline{\text{SMC0\_AMS2}}$	SMC0 Memory Select 2	C	PC_07
$\overline{\text{SMC0\_AMS3}}$	SMC0 Memory Select 3	C	PC_08
$\overline{\text{SMC0\_AOE}}$	SMC0 Output Enable	D	PD_01
SMC0_ARDY	SMC0 Asynchronous Ready	B	PB_04
$\overline{\text{SMC0\_ARE}}$	SMC0 Read Enable	C	PC_00
$\overline{\text{SMC0\_AWE}}$	SMC0 Write Enable	B	PB_15
SMC0_D00	SMC0 Data 0	E	PE_12
SMC0_D01	SMC0 Data 1	E	PE_11

Table 17. Signal Multiplexing for Port E

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PE_00	PPI0_D09	PWM2_CL		SMC0_D04	
PE_01	PPI0_FS2	SPI0_SEL5	UART1_CTS	C1_FLG0	
PE_02	PPI0_FS1	SPI0_SEL6	UART1_RTS	C2_FLG0	
PE_03	PPI0_CLK	SPI0_SEL7	SPI2_SEL2	C1_FLG1	
PE_04	PPI0_D08	PWM2_DH	SPI2_SEL3	C2_FLG1	
PE_05	PPI0_D07	PWM2_SYNC	SPI2_SEL4	C1_FLG2	
PE_06	PPI0_D06		SPI2_SEL5	C2_FLG2	
PE_07	PPI0_D05		SPI1_SEL2	C1_FLG3	
PE_08	PPI0_D04	SPI1_SEL5	SPI1_RDY	C2_FLG3	
PE_09	PPI0_D03	PWM0_SYNC	TM0_TMR0	SMC0_D03	
PE_10	PPI0_D02	PWM2_DL	UART2_RTS	SMC0_D02	
PE_11	PPI0_D01	SPI1_SEL3	UART2_CTS	SMC0_D01	SPI1_SS
PE_12	PPI0_D00	SPI1_SEL4	SPI2_RDY	SMC0_D00	
PE_13	SPI1_CLK		PPI0_D20	SMC0_AMS1	
PE_14	SPI1_MISO		PPI0_D21	SMC0_ABE0	
PE_15	SPI1_MOSI		PPI0_D22	SMC0_ABE1	

Table 18 shows the internal timer signal routing. This table applies to both the 349-ball and 529-ball CSP_BGA packages.

Table 18. Internal Timer Signal Routing

Timer Input Signal	Internal Source
TM0_ACLK0	SYS_CLKIN1
TM0_ACI5	DAI0_CRS_PB04_O
TM0_ACLK5	DAI0_CRS_PB03_O
TM0_ACI6	DAI1_CRS_PB04_O
TM0_ACLK6	DAI1_CRS_PB03_O
TM0_ACI7	CNT0_TO
TM0_ACLK7	SYS_CLKIN0

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

Signal Name	Description	Port	Pin Name
LP0_D7	LP0 Data 7	D	PD_09
LP1_ACK	LP1 Acknowledge	B	PB_15
LP1_CLK	LP1 Clock	C	PC_00
LP1_D0	LP1 Data 0	B	PB_07
LP1_D1	LP1 Data 1	B	PB_08
LP1_D2	LP1 Data 2	B	PB_09
LP1_D3	LP1 Data 3	B	PB_10
LP1_D4	LP1 Data 4	B	PB_11
LP1_D5	LP1 Data 5	B	PB_12
LP1_D6	LP1 Data 6	B	PB_13
LP1_D7	LP1 Data 7	B	PB_14
MLB0_CLKN	MLB0 Differential Clock (–)	Not Muxed	MLB0_CLKN
MLB0_CLKP	MLB0 Differential Clock (+)	Not Muxed	MLB0_CLKP
MLB0_DATN	MLB0 Differential Data (–)	Not Muxed	MLB0_DATN
MLB0_DATP	MLB0 Differential Data (+)	Not Muxed	MLB0_DATP
MLB0_SIGN	MLB0 Differential Signal (–)	Not Muxed	MLB0_SIGN
MLB0_SIGP	MLB0 Differential Signal (+)	Not Muxed	MLB0_SIGP
MLB0_CLK	MLB0 Single-Ended Clock	B	PB_04
MLB0_DAT	MLB0 Single-Ended Data	B	PB_06
MLB0_SIG	MLB0 Single-Ended Signal	B	PB_05
MLB0_CLKOUT	MLB0 Single-Ended Clock Out	D	PD_14
$\overline{\text{MSI0\_CD}}$	MSI0 Card Detect	F	PF_12
MSI0_CLK	MSI0 Clock	F	PF_11
MSI0_CMD	MSI0 Command	F	PF_10
MSI0_D0	MSI0 Data 0	F	PF_02
MSI0_D1	MSI0 Data 1	F	PF_03
MSI0_D2	MSI0 Data 2	F	PF_04
MSI0_D3	MSI0 Data 3	F	PF_05
MSI0_D4	MSI0 Data 4	F	PF_06
MSI0_D5	MSI0 Data 5	F	PF_07
MSI0_D6	MSI0 Data 6	F	PF_08
MSI0_D7	MSI0 Data 7	F	PF_09
$\overline{\text{MSI0\_INT}}$	MSI0 eSDIO Interrupt Input	F	PF_13
PA_00-15	PORTA Position 00 through Position 15	A	PA_00-15
PB_00-15	PORTB Position 00 through Position 15	B	PB_00-15
PCIE0_CLKM	PCIE0 CLK -	Not Muxed	PCIE0_CLKM
PCIE0_CLKP	PCIE0 CLK +	Not Muxed	PCIE0_CLKP
PCIE0_REF	PCIE0 Reference	Not Muxed	PCIE0_REF
PCIE0_RXM	PCIE0 RX -	Not Muxed	PCIE0_RXM
PCIE0_RXP	PCIE0 RX +	Not Muxed	PCIE0_RXP
PCIE0_TXM	PCIE0 TX -	Not Muxed	PCIE0_TXM
PCIE0_TXP	PCIE0 TX +	Not Muxed	PCIE0_TXP
PC_00-15	PORTC Position 00 through Position 15	C	PC_00-15
PD_00-15	PORTD Position 00 through Position 15	D	PD_00-15
PE_00-15	PORTE Position 00 through Position 15	E	PE_00-15
PF_00-15	PORTF Position 00 through Position 15	F	PF_00-15
PG_00-5	PORTG Position 00 through Position 5	G	PG_00-5
PPI0_CLK	EPPI0 Clock	E	PE_03

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
PE_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 12 | EPPI0 Data 0 | SMC0 Data 0 | SPI1 Slave Select Output 4 | SPI2 Ready Notes: No notes
PE_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 13 | EPPI0 Data 20 | SMC0 Memory Select 1 | SPI1 Clock Notes: No notes
PE_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 14 | EPPI0 Data 21 | SMC0 Byte Enable 0 | SPI1 Master In, Slave Out Notes: No notes
PE_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 15 | EPPI0 Data 22 | SMC0 Byte Enable 1 | SPI1 Master Out, Slave In Notes: No notes
PF_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 0 | SPI1 Slave Select Output 6 | TIMER0 Timer 6 Notes: No notes
PF_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 1 | SPI1 Slave Select Output 7 | TIMER0 Timer 7 Notes: No notes
PF_02	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 2 | HADC0 End of Conversion / Serial Data Out | MSI0 Data 0 Notes: No notes
PF_03	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 3 | HADC0 Controls to external multiplexer | MSI0 Data 1 Notes: No notes
PF_04	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 4 | HADC0 Controls to external multiplexer | MSI0 Data 2 Notes: No notes
PF_05	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 5 | HADC0 Controls to external multiplexer | MSI0 Data 3 Notes: No notes
PF_06	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 6 | MSI0 Data 4 | PWM2 Channel A Low Side Notes: No notes
PF_07	InOut	A	PullDown/ Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 7 | MSI0 Data 5 | PWM2 Channel A High Side Notes: No notes

Table 34. Dynamic Current for Each SHARC+ Core (mA, with ASF = 1.00)

	Voltage (V_{DD_INT})		
f_{CCLK} (MHz)	1.05	1.10	1.15
450	321.3	336.6	351.9
400	285.6	299.2	312.8
350	249.9	261.8	273.7
300	214.2	224.4	234.6
250	178.5	187.0	195.5
200	142.8	149.6	156.4
150	107.1	112.2	117.3
100	71.4	74.8	78.2

Table 35. Dynamic Current for the ARM Cortex-A5 Core (mA, with ASF = 1.00)

	Voltage (V_{DD_INT})		
f_{CCLK} (MHz)	1.05	1.10	1.15
450	70.88	74.25	77.63
400	63.00	66.00	69.00
350	55.13	57.75	60.38
300	47.25	49.50	51.75
250	39.38	41.25	43.13
200	31.50	33.00	34.50
150	23.63	24.75	25.88
100	15.75	16.50	17.25

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

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

Table 36. Activity Scaling Factors for the FFT Accelerator (ASF_{ACCL})

I_{DD_INT} Power Vector	ASF_{ACCL}
Unused	0.0
I_{DD-TYP}	0.32

Clock Current

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

$I_{DD\_INT\_SYSCLK\_DYN}$ (mA) = 0.78 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK0\_DYN}$ (mA) = 0.44 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK1\_DYN}$ (mA) = 0.06 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

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

$I_{DD\_INT\_OCLK\_DYN}$ (mA) = 0.02 × f_{OCLK} (MHz) × V_{DD_INT} (V)

Current from High-Speed Peripheral Operation

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

$I_{DD\_INT\_USB\_DYN}$ = 20 mA (if both USBs are enabled in HS mode)

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

$I_{DD\_INT\_GIGE\_DYN}$ = 10 mA (if gigabit EMAC is enabled)

$I_{DD\_INT\_PCIE\_DYN}$ = 240 mA (if PCIe is enabled in 5 Gbps mode)

Data Transmission Current

The data transmission current represents the power dissipated when moving data throughout the system via direct memory access (DMA). This current is proportional to the data rate. Refer to the power calculator available with "Estimating Power for ADSP-SC58x/2158x SHARC+ Processors" (EE-392) to estimate $I_{DD\_INT\_DMA\_DR\_DYN}$ based on the bandwidth of the data transfer.

HADC

HADC Electrical Characteristics

Table 37. HADC Electrical Characteristics

Parameter	Conditions	Typ	Unit
$I_{DD\_HADC\_IDLE}$	Current consumption on V_{DD_HADC}. HADC is powered on, but not converting.	2.0	mA
$I_{DD\_HADC\_ACTIVE}$	Current consumption on V_{DD_HADC} during a conversion.	2.5	mA
$I_{DD\_HADC\_POWERDOWN}$	Current consumption on V_{DD_HADC}. Analog circuitry of the HADC is powered down.	10	µA

HADC DC Accuracy

Table 38. HADC DC Accuracy[1]

Parameter	Typ	Unit[2]
Resolution	12	Bits
No Missing Codes (NMC)	10	Bits
Integral Nonlinearity (INL)	±2	LSB
Differential Nonlinearity (DNL)	±2	LSB
Offset Error	±8	LSB
Offset Error Matching	±10	LSB
Gain Error	±4	LSB
Gain Error Matching	±4	LSB

[1] See the Operating Conditions section for the HADC0_VINx specification.
[2] LSB = HADC0_VREFP ÷ 4096.

HADC Timing Specifications

Table 39. HADC Timing Specifications

Parameter	Typ	Max	Unit
Conversion Time	$20 \times T_{SAMPLE}$		µs
Throughput Range		1	MSPS
T_{WAKEUP}		100	µs

TMU

TMU Characteristics

Table 40. TMU Characteristics

Parameter	Typ	Unit
Resolution	1	°C
Accuracy	±6	°C

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}.

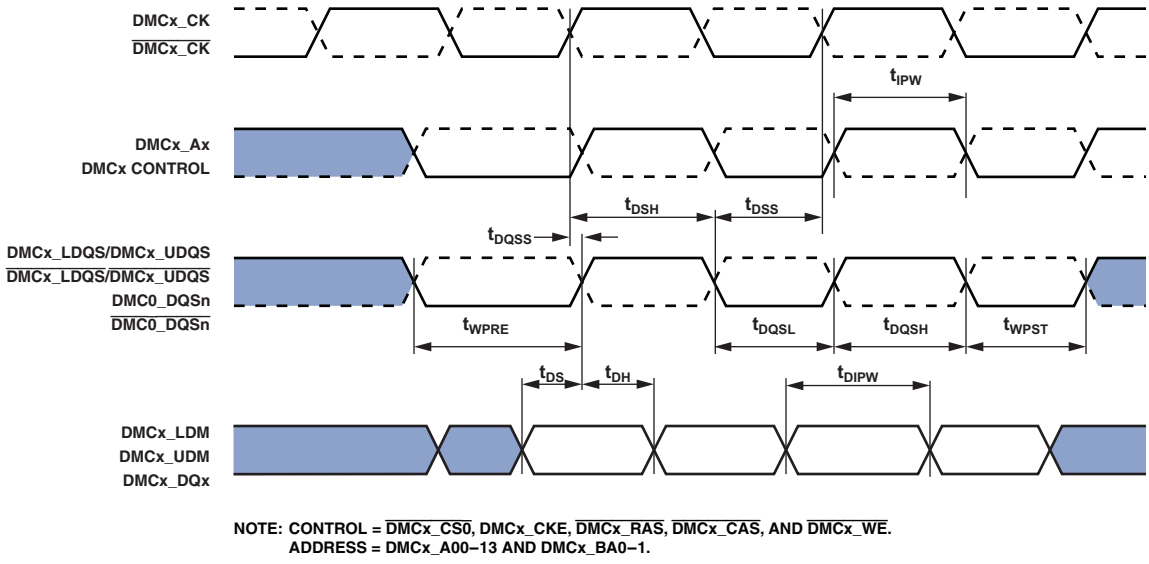

Figure 19. DDR2 SDRAM Controller Output AC Timing

DDR3 SDRAM Write Cycle Timing

Table 59 and Figure 25 show mobile DDR3 SDRAM output ac timing, related to the DMC.

Table 59. DDR3 SDRAM Write Cycle Timing VDD_DMCx Nominal 1.5 V[1]

Parameter		450 MHz[2] Min	450 MHz[2] Max	Unit
Switching Characteristics				
t_{DQSS}	DMCx_DQS Latching Rising Transitions to Associated Clock Edges[3]	–0.25	0.25	t_{CK}
t_{DS}	Last Data Valid to DMCx_DQS Delay (Slew > 1 V/ns)	0.125		ns
t_{DH}	DMCx_DQS to First Data Invalid Delay (Slew > 1 V/ns)	0.150		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	0.45	0.55	t_{CK}
t_{DQSL}	DMCx_DQS Input Low Pulse Width	0.45	0.55	t_{CK}
t_{WPRE}	Write Preamble	0.9		t_{CK}
t_{WPST}	Write Postamble	0.3		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	0.840		ns
t_{DIPW}	DMCx_DQ and DMCx_DM Output Pulse Width	0.550		ns

[1] Specifications apply to both DMC0 and DMC1.
[2] To ensure proper operation of the DDR3, all the DDR3 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}.

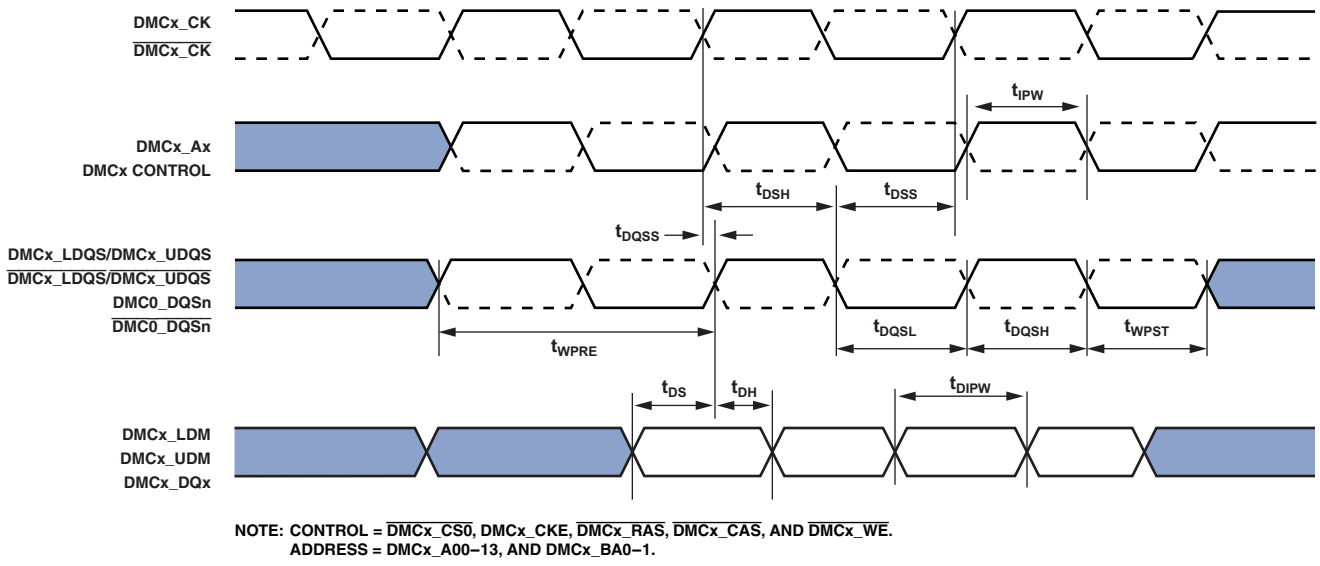

Figure 25. DDR3 SDRAM Controller Output AC Timing

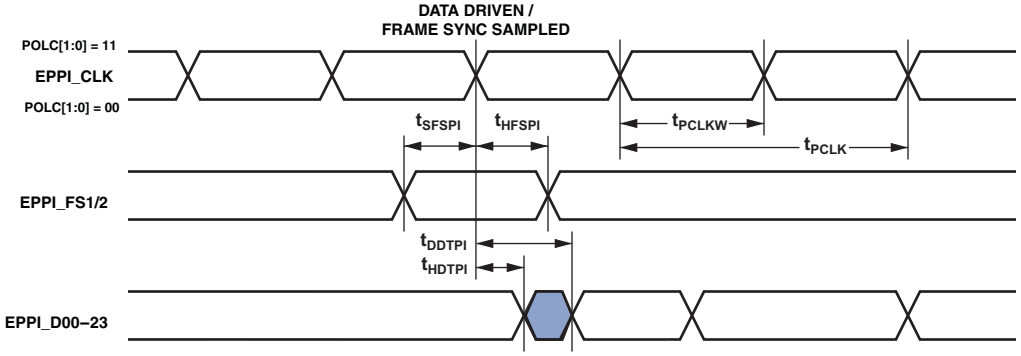

Figure 29. EPPI Internal Clock GP Transmit Mode with External Frame Sync Timing

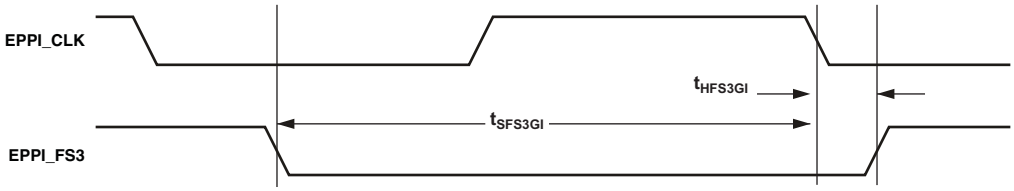

Figure 30. Clock Gating Mode with Internal Clock and External Frame Sync Timing

Table 61. Enhanced Parallel Peripheral Interface (EPPI)—External Clock

Parameter		Min	Max	Unit
Timing Requirements				
t_{PCLKW}	EPPI_CLK Width[1]	0.5 × $t_{PCLKEXT}$ – 0.5		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKEXT}$ – 1		ns
t_{SFSPE}	External FS Setup Before EPPI_CLK	2		ns
t_{HFSPE}	External FS Hold After EPPI_CLK	3.7		ns
t_{SDRPE}	Receive Data Setup Before EPPI_CLK	2		ns
t_{HDRPE}	Receive Data Hold After EPPI_CLK	3.7		ns
Switching Characteristics				
t_{DFSPE}	Internal FS Delay After EPPI_CLK		15.3	ns
t_{HOFSPE}	Internal FS Hold After EPPI_CLK	2.4		ns
t_{DDTPE}	Transmit Data Delay After EPPI_CLK		15.3	ns
t_{HDTPE}	Transmit Data Hold After EPPI_CLK	2.4		ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external EPPI_CLK. For the external EPPI_CLK ideal maximum frequency see the $f_{PCLKEXT}$ specification in Table 29.

Serial Ports (SPORT)

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

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

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

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

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

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

Table 64. Serial Ports—External Clock[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SFSE}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	2		ns
t_{HFSE}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	2.7		ns
t_{SDRE}	Receive Data Setup Before Receive SPTx_CLK[2]	2		ns
t_{HDRE}	Receive Data Hold After SPTx_CLK[2]	2.7		ns
$t_{SPTCLKW}$	SPTx_CLK Width[3]	$0.5 \times t_{SPTCLKEXT} - 1.5$		ns
t_{SPTCLK}	SPTx_CLK Period[3]	$t_{SPTCLKEXT} - 1.5$		ns
Switching Characteristics				
t_{DFSE}	Frame Sync Delay After SPTx_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[4]		14.5	ns
t_{HOFSE}	Frame Sync Hold After SPTx_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[4]	2		ns
t_{DDTE}	Transmit Data Delay After Transmit SPTx_CLK[4]		14	ns
t_{HDTE}	Transmit Data Hold After Transmit SPTx_CLK[4]	2		ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to sample edge.
[3] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPTx_CLK. For the external SPTx_CLK ideal maximum frequency see the $f_{SPTCLKEXT}$ specification in Table 29.
[4] Referenced to drive edge.

Table 65. Serial Ports—Internal Clock[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SFSI}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	12		ns
t_{HFSI}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	–0.5		ns
t_{SDRI}	Receive Data Setup Before SPTx_CLK[2]	3.4		ns
t_{HDRI}	Receive Data Hold After SPTx_CLK[2]	1.5		ns
Switching Characteristics				
t_{DFSI}	Frame Sync Delay After SPTx_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[3]		3.5	ns
t_{HOFSI}	Frame Sync Hold After SPTx_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[3]	–2.5		ns
t_{DDTI}	Transmit Data Delay After SPTx_CLK[3]		3.5	ns
t_{HDTI}	Transmit Data Hold After SPTx_CLK[3]	–2.5		ns
t_{SCLKIW}	SPTx_CLK Width[4]	$0.5 \times t_{SPTCLKPROG} - 1.5$		ns
t_{SPTCLK}	SPTx_CLK Period[4]	$t_{SPTCLKPROG} - 1.5$		ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to the sample edge.
[3] Referenced to drive edge.
[4] See Table 29 for details on the minimum period that can be programmed for $t_{SPTCLKPROG}$.

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

Parameter		Min	Max	Unit
Switching Characteristics				
t_{DRDVEN}	Data Valid Enable Delay from Drive Edge of External Clock[2]	2		ns
t_{DFDVEN}	Data Valid Disable Delay from Drive Edge of External Clock[2]		14	ns
t_{DRDVIN}	Data Valid Enable Delay from Drive Edge of Internal Clock[2]	–2.5		ns
t_{DFDVIN}	Data Valid Disable Delay from Drive Edge of Internal Clock[2]		3.5	ns

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

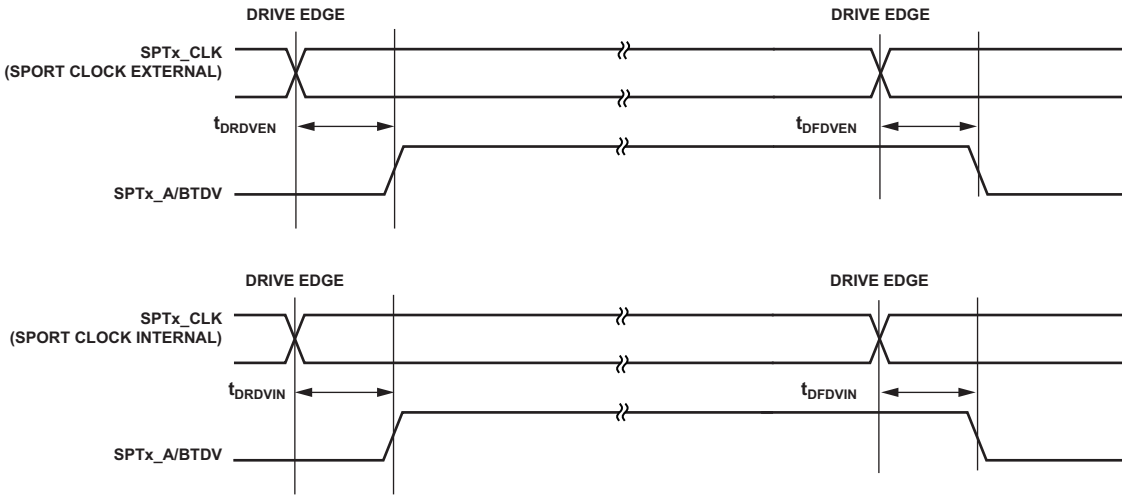

Figure 39. Serial Ports—Transmit Data Valid Internal and External Clock

SPI Port—Slave Timing

Table 72 and Figure 44 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 SPI2 only.
- CPHA is a configuration bit in the SPI_CTL register.

Table 72. SPI Port—Slave Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	$0.5 \times t_{SPICLKEXT} - 1$		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	$0.5 \times t_{SPICLKEXT} - 1$		ns
t_{SPICLK}	SPIx_CLK Period[2]	$t_{SPICLKEXT} - 1$		ns
t_{HDS}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SS}}$ Not Asserted	5		ns
t_{SPITDS}	Sequential Transfer Delay	$t_{SPICLK} - 1$		ns
t_{SDSCI}	$\overline{\text{SPIx\_SS}}$ Assertion to First SPIx_CLK Edge	10.5		ns
t_{SSPID}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	2		ns
t_{HSPID}	SPIx_CLK Sampling Edge to Data Input Invalid	1.6		ns
Switching Characteristics				
t_{DSOE}	$\overline{\text{SPIx\_SS}}$ Assertion to Data Out Active	0	14	ns
t_{DSDHI}	$\overline{\text{SPIx\_SS}}$ Deassertion to Data High Impedance	0	12.5	ns
t_{DDSPID}	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		14	ns
t_{HDSPID}	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	0		ns

[1] All specifications apply to all three SPIs.
[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPIx_CLK. For the external SPIx_CLK ideal maximum frequency see the $f_{SPICLKTEXT}$ specification in Table 29.

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 76. SPI Port—SPIx_RDY Master Timing[1]

Parameter		Conditions	Min	Max	Unit
Timing Requirement					
$t_{SRDYSCKM}$	Setup Time for SPIx_RDY Deassertion Before Last Valid Data SPIx_CLK Edge		$(2 + 2 \times BAUD^{2}) \times t_{SCLK1} + 10$		ns
Switching Characteristic					
$t_{DRDYSCKM}$[3]	Assertion of SPIx_RDY to First SPIx_CLK Edge of Next Transfer	Baud = 0, CPHA = 0	$4.5 \times t_{SCLK1}$	$5.5 \times t_{SCLK1} + 10$	ns
		Baud = 0, CPHA = 1	$4 \times t_{SCLK1}$	$5 \times t_{SCLK1} + 10$	ns
		Baud > 0, CPHA = 0	$(1 + 1.5 \times BAUD^{2}) \times t_{SCLK1}$	$(2 + 2.5 \times BAUD^{2}) \times t_{SCLK1} + 10$	ns
		Baud > 0, CPHA = 1	$(1 + 1 \times BAUD^{2}) \times t_{SCLK1}$	$(2 + 2 \times BAUD^{2}) \times t_{SCLK1} + 10$	ns

[1] All specifications apply to all three SPIs.
[2] BAUD value 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.

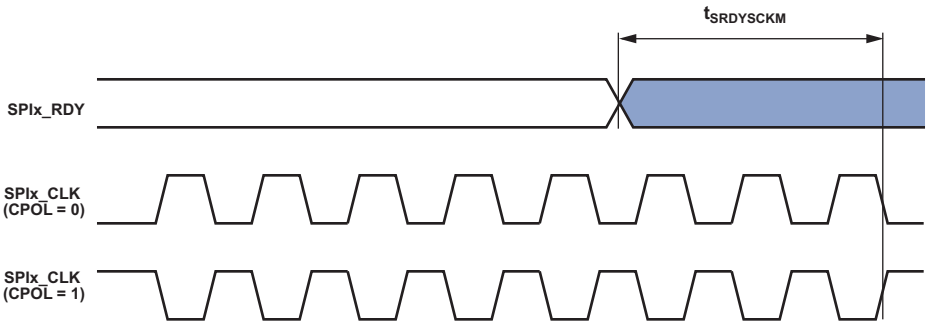

Figure 48. SPIx_RDY Setup Before SPIx_CLK

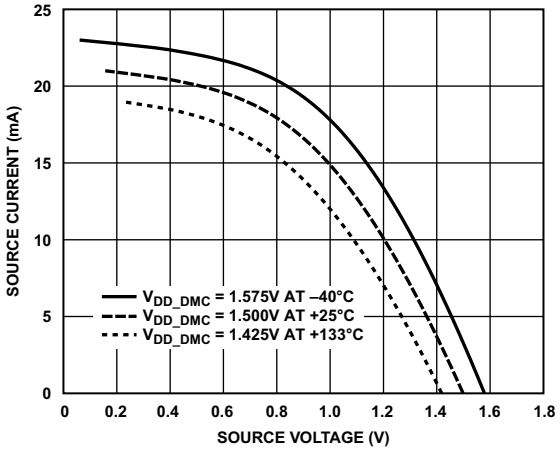

Figure 82. Driver Type B and Driver Type C (DDR3 Drive Strength 40 Ω)

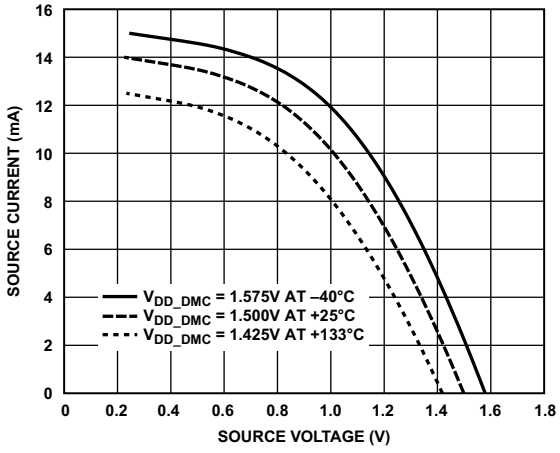

Figure 83. Driver Type B and Driver Type C (DDR3 Drive Strength 60 Ω)

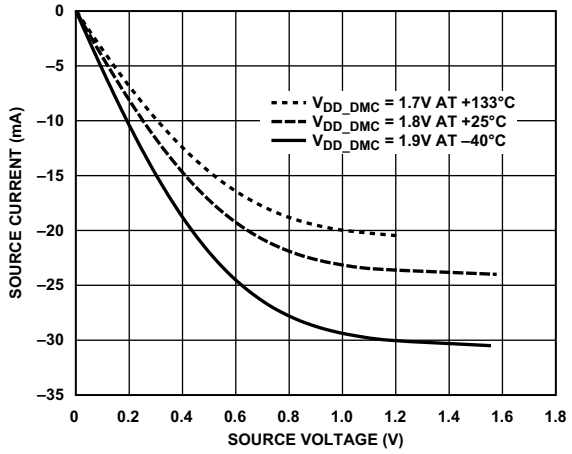

Figure 84. Driver Type B and Driver Type C (DDR2 Drive Strength 40 Ω)

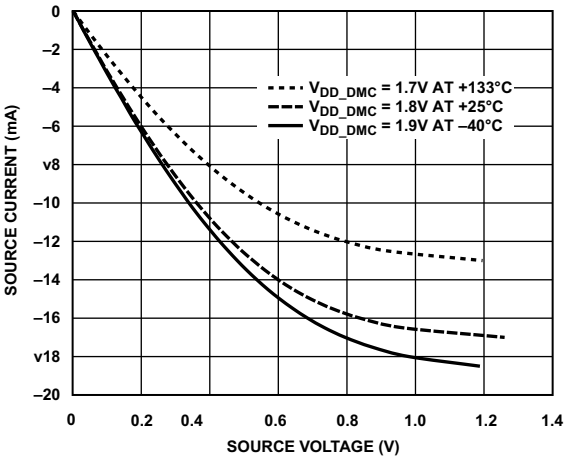

Figure 85. Driver Type B and Driver Type C (DDR2 Drive Strength 60Ω)

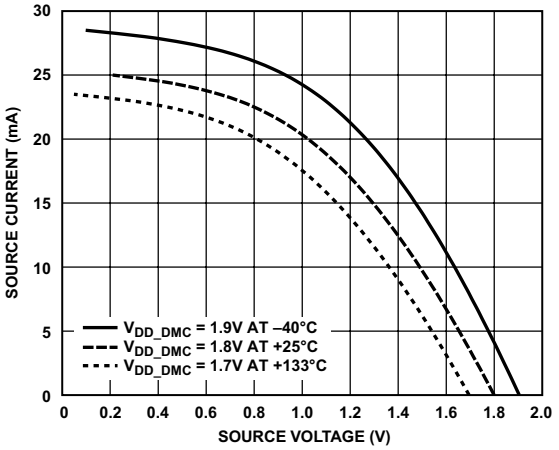

Figure 86. Driver Type B and Driver Type C (DDR2 Drive Strength 40Ω)

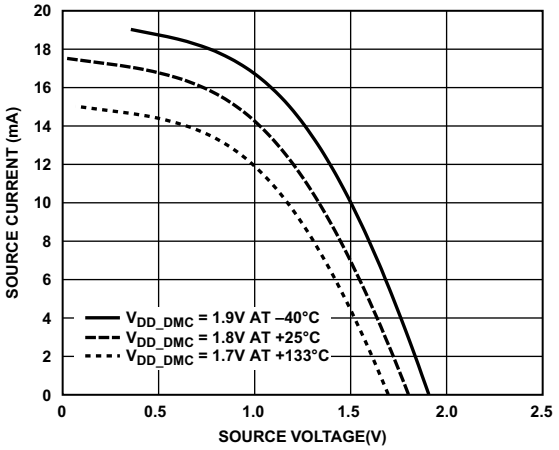

Figure 87. Driver Type B and Driver Type C (DDR2 Drive Strength 60 Ω)

ADSP-SC58x/ADSP-2158x 529-BALL BGA BALL ASSIGNMENTS

The ADSP-SC58x/ADSP-2158x 529-Ball BGA Ball Assignments (Numerical by Ball Number) table lists the 529-ball BGA package by ball number.

The ADSP-SC58x/ADSP-2158x 529-Ball BGA Ball Assignments (Alphabetical by Pin Name) table lists the 529-ball BGA package by pin name.

ADSP-SC58x/ADSP-2158x 529-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	B19	DMC1_DQ11	D14	DMC1_BA2	F09	GND
A02	DMC0_UDQS	B20	DMC1_DQ12	D15	DMC1_CAS	F10	VDD_INT
A03	DMC0_CK	B21	DMC1_DQ14	D16	DMC1_RAS	F11	VDD_INT
A04	DMC0_CK	B22	PD_00	D17	DMC1_A09	F12	VDD_INT
A05	DMC0_DQ09	B23	PD_04	D18	DMC1_A15	F13	VDD_INT
A06	DMC0_LDQS	C01	DMC0_DQ14	D19	DMC1_A10	F14	VDD_INT
A07	DMC0_LDQS	C02	DMC0_DQ13	D20	DMC1_A11	F15	VDD_INT
A08	DMC0_DQ05	C03	DMC0_CS0	D21	PC_14	F16	GND
A09	DMC0_DQ03	C04	DMC0_CKE	D22	PD_10	F17	VDD_INT
A10	DMC0_DQ01	C05	DMC0_LDM	D23	PD_09	F18	VDD_INT
A11	DMC1_DQ03	C06	DMC1_RESET	E01	DMC0_A04	F19	VDD_INT
A12	DMC1_DQ00	C07	DMC1_A03	E02	DMC0_RAS	F20	PE_06
A13	DMC1_LDQS	C08	DMC1_A00	E03	DMC0_BA1	F21	PD_02
A14	DMC1_LDQS	C09	DMC1_A01	E04	DMC0_WE	F22	PD_13
A15	DMC1_VREF	C10	DMC1_A04	E05	DMC0_RZQ	F23	PD_12
A16	DMC1_CK	C11	DMC1_A06	E06	GND	G01	DMC0_A13
A17	DMC1_CK	C12	DMC1_BA1	E07	GND	G02	DMC0_A09
A18	DMC1_DQ09	C13	DMC1_ODT	E08	GND	G03	DMC0_A03
A19	DMC1_UDQS	C14	DMC1_CS0	E09	GND	G04	DMC0_A11
A20	DMC1_UDQS	C15	DMC1_LDM	E10	VDD_INT	G05	VDD_INT
A21	DMC1_DQ13	C16	DMC1_UDM	E11	VDD_INT	G06	VDD_DMC
A22	DMC1_DQ15	C17	DMC1_A14	E12	VDD_INT	G07	VDD_DMC
A23	GND	C18	DMC1_A12	E13	VDD_INT	G08	VDD_DMC
B01	DMC0_UDQS	C19	DMC1_A13	E14	VDD_INT	G09	VDD_DMC
B02	DMC0_DQ12	C20	PC_13	E15	VDD_INT	G10	VDD_DMC
B03	DMC0_DQ11	C21	PD_01	E16	VDD_INT	G11	VDD_DMC
B04	DMC0_DQ10	C22	PD_06	E17	VDD_INT	G12	VDD_DMC
B05	DMC0_DQ08	C23	PD_05	E18	VDD_INT	G13	VDD_DMC
B06	DMC0_DQ06	D01	DMC0_VREF	E19	DMC1_RZQ	G14	VDD_DMC
B07	DMC0_DQ07	D02	DMC0_DQ15	E20	PC_15	G15	VDD_DMC
B08	DMC0_DQ04	D03	DMC0_BA0	E21	PD_08	G16	VDD_DMC
B09	DMC0_DQ02	D04	DMC0_BA2	E22	PD_14	G17	VDD_DMC
B10	DMC0_DQ00	D05	DMC0_ODT	E23	PD_11	G18	VDD_DMC
B11	DMC1_DQ01	D06	DMC0_UDM	F01	DMC0_A01	G19	VDD_INT
B12	DMC1_DQ02	D07	DMC1_A05	F02	DMC0_A06	G20	PE_04
B13	DMC1_DQ04	D08	DMC1_WE	F03	DMC0_CAS	G21	PE_13
B14	DMC1_DQ05	D09	DMC1_A07	F04	DMC0_A02	G22	PE_01
B15	DMC1_DQ06	D10	DMC1_A02	F05	DMC0_A07	G23	PE_00
B16	DMC1_DQ07	D11	DMC1_BA0	F06	GND	H01	DMC0_A14
B17	DMC1_DQ08	D12	DMC1_A08	F07	VDD_INT	H02	DMC0_A12
B18	DMC1_DQ10	D13	DMC1_CKE	F08	VDD_INT	H03	DMC0_A05