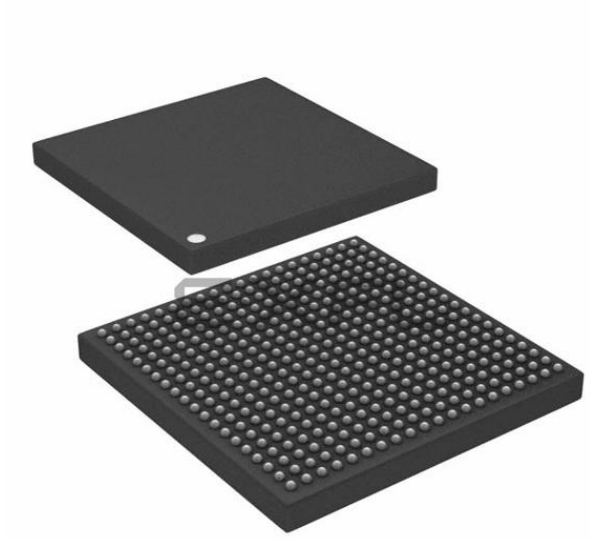

Welcome to E-XFL.COM

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	Fixed/Floating Point
Interface	CAN, EBI/EMI, Ethernet, DAI, I²C, MMC/SD/SDIO, SPI, SPORT, UART/USART, USB OTG
Clock Rate	450MHz, 450MHz
Non-Volatile Memory	External
On-Chip RAM	2MB
Voltage - I/O	3.30V
Voltage - Core	1.10V
Operating Temperature	-40°C ~ 100°C (TA)
Mounting Type	Surface Mount
Package / Case	400-LFBGA, CSPBGA
Supplier Device Package	400-CSPBGA (17x17)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-sc573cbcz-4

TABLE OF CONTENTS

REVISION HISTORY

6/2018—Rev. A to Rev. B

Changes to System Features 1
Changes to Additional Features 1
Changes to Table 2 and Table 3, General Description 3
Changes to Operating Conditions 56
Deleted Package Information from Specifications 56
Changes to Table 27 and Table 28, Clock Related Operating Conditions 58
Changes to Electrical Characteristics 60
Changes to Table 29, Table 32, and Table 33, Total Internal Power Dissipation 62
Changes to Table 37, HADC Timing Specifications 64
Changes to Program Trace Macrocell (PTM) Timing 120
Changes to Test Conditions 124
Changes to Automotive Products 140
Changes to Ordering Guide 141

GENERAL DESCRIPTION

The ADSP-SC57x/ADSP-2157x processors are members of the SHARC® family of products. The ADSP-SC57x processor is based on the SHARC+® dual-core and the ARM® Cortex®-A5 core. The ADSP-SC57x/ADSP-2157x SHARC processors are members of the single-instruction, multiple data (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 reduced instruction set computing (RISC), multimedia support, and leading edge signal processing in one integrated package. These applications span a wide array of markets, including automotive, professional audio, and industrial-based applications that require high floating-point performance.

Table 2 provides comparison information for features that vary across the standard processors.

Table 3 provides comparison information for features that vary across the automotive processors.

Table 1. Common Product Features

Product Features	ADSP-SC57x/ADSP-2157x
DAI (includes SRU)	1
Full SPORTs	4
S/PDIF receive/transmit	1
ASRCs	4
PCGs	2
Pin buffers	20
I^2C (TWI)	3
Quad-data bit SPI	1
Dual-data bit SPI	2
CAN2.0	2
UARTs	3
Enhanced PPI Up to 16-bit on BGA 12-bit on LQFP	1
GP timer	8
GP counter	1
Watchdog timers	3
ADC control module	Yes
Hardware accelerators	
FIR/IIR	Yes
Security cryptographic engine	Yes
Multichannel 12-bit ADC	8-channel BGA; 4-channel LQFP

periodic communication message objects. Dedicated hardware circuitry compares the signature with precalculated values and triggers appropriate fault events.

For example, every 100 ms the system software initiates the signature calculation of the entire memory contents and compares these contents with expected, precalculated values. If a mismatch occurs, a fault condition is generated through the processor core or the trigger routing unit.

The CRC is a hardware module based on a CRC32 engine that computes the CRC value of the 32-bit data-words presented to it. The source channel of the memory to memory DMA (in memory scan mode) provides data. The data can be optionally forwarded to the destination channel (memory transfer mode). The main features of the CRC peripheral are as follows:

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

Event Handling

The processors provide event handling that supports both nesting and prioritization. Nesting allows multiple event service routines to be active simultaneously. Prioritization ensures that servicing a higher priority event takes precedence over servicing a lower priority event.

The processors provide support for four different types of events:

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

System Event Controller (SEC)

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 and fault services
- Global locking that supports a register level protection model to prevent writes to locked registers
- Fault management including fault action configuration, time out, external indication, and system reset

Trigger Routing Unit (TRU)

The trigger routing unit (TRU) provides system level sequence control without core intervention. The TRU maps trigger masters (generators of triggers) to trigger slaves (receivers of triggers). Slave endpoints can be configured to respond to triggers in various ways. Common applications enabled by the TRU include,

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

SECURITY FEATURES

The following sections describe the security features of the ADSP-SC57x/ADSP-2157x processors.

ARM TrustZone

The ADSP-SC57x processors provide TrustZone technology that is integrated into the ARM Cortex-A5 processors. The TrustZone technology enables a secure state that is extended throughout the system fabric.

Cryptographic Hardware Accelerators

The ADSP-SC57x/ADSP-2157x processors support standards-based hardware accelerated encryption, decryption, authentication, and true random number generation.

Support for the hardware accelerated cryptographic ciphers includes the following:

- AES in ECB, CBC, ICM, and CTR modes with 128-bit, 192-bit, and 256-bit keys
- DES in ECB and CBC mode with 56-bit key
- 3DES in ECB and CBC mode with 3x 56-bit key
- ARC4 in stateful, stateless mode, up to 128-bit key

synchronous or asynchronous sample rate conversion across independent stereo channels, without using internal processor resources. The ASRC blocks can also be configured to operate together to convert multichannel audio data without phase mismatches. Finally, the ASRC can clean up audio data from jittery clock sources such as the S/PDIF receiver.

S/PDIF-Compatible Digital Audio Receiver/Transmitter

The Sony/Philips Digital Interface Format (S/PDIF) is a standard audio data transfer format that allows the transfer of digital audio signals from one device to another without converting them to an analog signal. There is one S/PDIF transmit/receive block 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 two units located in the DAI block. The PCG can generate a pair of signals (clock and frame sync) derived from a clock input signal (CLKIN, SCLK0, or DAI pin buffer). Both units are identical in functionality and operate independently of each other. The two signals generated by each unit are normally used as a serial bit clock/frame sync pair.

Enhanced Parallel Peripheral Interface (EPPI)

The processors provide an enhanced parallel peripheral interface (EPPI) that supports data widths up to 16 bits for the BGA package and 12 bits for the LQFP package. The EPPI supports direct connection to thin film transistor (TFT) LCD panels, parallel ADCs and DACs, video encoders and decoders, image sensor modules, and other general-purpose peripherals.

The features supported in the EPPI module include the following:

- Programmable data length of 8 bits, 10 bits, 12 bits, 14 bits, and 16 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 and 16 bits. If packing/unpacking is enabled, configure endianness to change the order of packing/unpacking of bytes or words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various deinterleaving/interleaving modes for receiving or transmitting 4:2:2 YCrCb data.
- Configurable LCD data enable output available on Frame Sync 3.

Universal Asynchronous Receiver/Transmitter (UART) Ports

The processors provide 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, 4-wire interface consisting of two data pins, one device select pin, and a gated clock pin. The two data pins allow full-duplex operation to other SPI-compatible devices. An extra two (optional) data pins are provided to support quad-SPI operation. Enhanced modes of operation, such as flow control, fast mode, and dual-I/O mode (DIOM), are also supported. DMA mode allows for transferring several words with minimal central processing unit (CPU) interaction.

With a range of configurable options, the SPI ports provide a glueless hardware interface with other SPI-compatible devices in master mode, slave mode, and multimaster environments. The SPI peripheral includes programmable baud rates, clock phase, and clock polarity. The peripheral can operate in a multimaster environment by interfacing with several other devices,

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DAI0_PIN16	DAI0 Pin 16	Not Muxed	DAI0_PIN16
DAI0_PIN17	DAI0 Pin 17	Not Muxed	DAI0_PIN17
DAI0_PIN18	DAI0 Pin 18	Not Muxed	DAI0_PIN18
DAI0_PIN19	DAI0 Pin 19	Not Muxed	DAI0_PIN19
DAI0_PIN20	DAI0 Pin 20	Not Muxed	DAI0_PIN20
DMC0_A00	DMC0 Address 0	Not Muxed	DMC0_A00
DMC0_A01	DMC0 Address 1	Not Muxed	DMC0_A01
DMC0_A02	DMC0 Address 2	Not Muxed	DMC0_A02
DMC0_A03	DMC0 Address 3	Not Muxed	DMC0_A03
DMC0_A04	DMC0 Address 4	Not Muxed	DMC0_A04
DMC0_A05	DMC0 Address 5	Not Muxed	DMC0_A05
DMC0_A06	DMC0 Address 6	Not Muxed	DMC0_A06
DMC0_A07	DMC0 Address 7	Not Muxed	DMC0_A07
DMC0_A08	DMC0 Address 8	Not Muxed	DMC0_A08
DMC0_A09	DMC0 Address 9	Not Muxed	DMC0_A09
DMC0_A10	DMC0 Address 10	Not Muxed	DMC0_A10
DMC0_A11	DMC0 Address 11	Not Muxed	DMC0_A11
DMC0_A12	DMC0 Address 12	Not Muxed	DMC0_A12
DMC0_A13	DMC0 Address 13	Not Muxed	DMC0_A13
DMC0_A14	DMC0 Address 14	Not Muxed	DMC0_A14
DMC0_A15	DMC0 Address 15	Not Muxed	DMC0_A15
DMC0_BA0	DMC0 Bank Address Input 0	Not Muxed	DMC0_BA0
DMC0_BA1	DMC0 Bank Address Input 1	Not Muxed	DMC0_BA1
DMC0_BA2	DMC0 Bank Address Input 2	Not Muxed	DMC0_BA2
$\overline{\text{DMC0\_CAS}}$	DMC0 Column Address Strobe	Not Muxed	$\overline{\text{DMC0\_CAS}}$
DMC0_CK	DMC0 Clock	Not Muxed	DMC0_CK
$\overline{\text{DMC0\_CK}}$	DMC0 Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
DMC0_CKE	DMC0 Clock enable	Not Muxed	DMC0_CKE
$\overline{\text{DMC0\_CS0}}$	DMC0 Chip Select 0	Not Muxed	$\overline{\text{DMC0\_CS0}}$
DMC0_DQ00	DMC0 Data 0	Not Muxed	DMC0_DQ00
DMC0_DQ01	DMC0 Data 1	Not Muxed	DMC0_DQ01
DMC0_DQ02	DMC0 Data 2	Not Muxed	DMC0_DQ02
DMC0_DQ03	DMC0 Data 3	Not Muxed	DMC0_DQ03
DMC0_DQ04	DMC0 Data 4	Not Muxed	DMC0_DQ04
DMC0_DQ05	DMC0 Data 5	Not Muxed	DMC0_DQ05
DMC0_DQ06	DMC0 Data 6	Not Muxed	DMC0_DQ06
DMC0_DQ07	DMC0 Data 7	Not Muxed	DMC0_DQ07
DMC0_DQ08	DMC0 Data 8	Not Muxed	DMC0_DQ08
DMC0_DQ09	DMC0 Data 9	Not Muxed	DMC0_DQ09
DMC0_DQ10	DMC0 Data 10	Not Muxed	DMC0_DQ10
DMC0_DQ11	DMC0 Data 11	Not Muxed	DMC0_DQ11
DMC0_DQ12	DMC0 Data 12	Not Muxed	DMC0_DQ12
DMC0_DQ13	DMC0 Data 13	Not Muxed	DMC0_DQ13
DMC0_DQ14	DMC0 Data 14	Not Muxed	DMC0_DQ14
DMC0_DQ15	DMC0 Data 15	Not Muxed	DMC0_DQ15
DMC0_LDM	DMC0 Data Mask for Lower Byte	Not Muxed	DMC0_LDM
DMC0_LDQS	DMC0 Data Strobe for Lower Byte	Not Muxed	DMC0_LDQS
$\overline{\text{DMC0\_LDQS}}$	DMC0 Data Strobe for Lower Byte (complement)	Not Muxed	$\overline{\text{DMC0\_LDQS}}$

GPIO MULTIPLEXING FOR 400-BALL CSP_BGA PACKAGE

Table 13 through Table 18 identify the pin functions that are multiplexed on the GPIO pins of the 400-ball CSP_BGA package.

Table 13. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PA_00					
PA_01					
PA_02					
PA_03					
PA_04					
PA_05					
PA_06					
PA_07					
PA_08					
PA_09	ETH0_PTPPPS0	LP1_D6	$\overline{\text{SPI0\_SEL4}}$		
PA_10	ETH0_MDIO	$\overline{\text{UART2\_RTS}}$	$\overline{\text{SPI2\_SEL6}}$		
PA_11	ETH0_MDC	$\overline{\text{UART2\_CTS}}$			
PA_12	ETH0_RXD1				
PA_13	ETH0_RXD0				
PA_14	ETH0_RXD2	ACM0_A3	$\overline{\text{SPI1\_SEL4}}$		
PA_15	ETH0_RXD3	ACM0_T0	$\overline{\text{SPI2\_SEL5}}$		

Table 14. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PB_00	ETH0_RXCLK_REFCLK	C2_FLG0			
PB_01	ETH0_CRS	ACM0_A4	LP1_ACK	TM0_TMR3	
PB_02	ETH0_RXCTL_RXDV		$\overline{\text{SPI1\_SEL5}}$		
PB_03	ETH0_RXERR	MLB0_CLKOUT	LP1_CLK	TM0_TMR4	
PB_04	ETH0_TXCLK	MLB0_DAT			
PB_05	ETH0_TXD3	MLB0_SIG			
PB_06	ETH0_TXD2	MLB0_CLK			
PB_07	ETH0_TXD0		$\overline{\text{SPI2\_SEL7}}$		
PB_08	ETH0_TXD1				
PB_09	ETH0_TXCTL_TXEN				
PB_10	SPI2_MISO				
PB_11	SPI2_MOSI				
PB_12	SPI2_D2				
PB_13	SPI2_D3				
PB_14	SPI2_CLK				
PB_15	$\overline{\text{SPI2\_SEL1}}$				$\overline{\text{SPI2\_SS}}$

GPIO MULTIPLEXING FOR 176-LEAD LQFP PACKAGE

Table 21 through Table 24 identify the pin functions that are multiplexed on the GPIO pins of the 176-lead LQFP package.

Table 21. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PA_00	TRACE0_CLK				TM0_ACLK1
PA_01	TRACE0_D00				
PA_02	TRACE0_D01				
PA_03	TRACE0_D02				
PA_04	TRACE0_D03				
PA_05	UART0_TX				
PA_06	UART0_RX				TM0_ACI0
PA_07	UART1_TX	SPI2_SEL2			
PA_08	UART1_RX	ACM0_A0	SPI1_SEL3		TM0_ACI1
PA_09	ETH0_PTPPPS0	LP1_D6	SPI0_SEL4		
PA_10	ETH0_MDIO	UART2_RTS	SPI2_SEL6		
PA_11	ETH0_MDC	UART2_CTS			
PA_12	ETH0_RXD1				
PA_13	ETH0_RXD0				
PA_14	ETH0_RXD2	ACM0_A3	SPI1_SEL4		
PA_15	ETH0_RXD3	ACM0_T0	SPI2_SEL5		

Table 22. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PB_00	ETH0_RXCLK_REFCLK	C2_FLG0			
PB_01	ETH0_CRS	ACM0_A4	LP1_ACK	TM0_TMR3	
PB_02	ETH0_RXCTL_RXDV		SPI1_SEL5		
PB_03	ETH0_RXERR	MLB0_CLKOUT	LP1_CLK	TM0_TMR4	
PB_04	ETH0_TXCLK	MLB0_DAT			
PB_05	ETH0_TXD3	MLB0_SIG			
PB_06	ETH0_TXD2	MLB0_CLK			
PB_07	ETH0_TXD0		SPI2_SEL7		
PB_08	ETH0_TXD1				
PB_09	ETH0_TXCTL_TXEN				
PB_10	SPI2_MISO				
PB_11	SPI2_MOSI				
PB_12	SPI2_D2				
PB_13	SPI2_D3				
PB_14	SPI2_CLK				
PB_15	SPI2_SEL1				SPI2_SS

Parameter		Conditions	Min	Nominal	Max	Unit
AUTOMOTIVE USE ONLY						
T_J	Junction Temperature 400-Ball CSP_BGA (Automotive Grade)	$T_{AMBIENT}$ = –40°C to +105°C CCLK ≤ 450 MHz	–40		+130[9]	°C
T_J	Junction Temperature 176-Lead LQFP-EP (Automotive Grade)	$T_{AMBIENT}$ = –40°C to +105°C CCLK ≤ 450 MHz	–40		+125[9]	°C
T_J	Junction Temperature 400-Ball CSP_BGA (Automotive Grade)	$T_{AMBIENT}$ = –40°C to +105°C CCLK ≤ 500 MHz	–40		+133[9]	°C
T_J	Junction Temperature 176-Lead LQFP-EP (Automotive Grade)	$T_{AMBIENT}$ = –40°C to +105°C CCLK ≤ 500 MHz	–40		+130[9]	°C

[1] Applies to DDR2/DDR3/LPDDR signals.
[2] If not used, V_{DD_USB} must be connected to 3.3 V.
[3] V_{HADC_VREF} must always be less than V_{DD_HADC}.
[4] Parameter value applies to all input and bidirectional pins except the TWI, DMC, USB, and MLB pins.
[5] Parameter applies to TWI signals.
[6] TWI signals are pulled up to V_{BUSTWI}. See Table 26.
[7] This parameter applies to all DMC0 signals in DDR2/DDR3 mode. V_{REF} is the voltage applied to the V_{REF_DMC} pin, nominally $V_{DD\_DMC}/2$.
[8] This parameter applies to DMC0 signals in LPDDR mode.
[9] Automotive application use profile only. Not supported for nonautomotive use. Contact Analog Devices for more information.

Table 26. TWI_VSEL Selections and V_{DD_EXT}/V_{BUSTWI}

TWI_VSEL Selections	V_{DD_EXT} Nominal	V_{BUSTWI}			Unit
		Min	Nominal	Max	
TWI000[1]	3.30	3.13	3.30	3.47	V
TWI100	3.30	4.75	5.00	5.25	V

[1] Designs must comply with the V_{DD_EXT} and V_{BUSTWI} voltages specified for the default TWI_DT setting for correct JTAG boundary scan operation during reset.

Table 32. Dynamic Current for Each SHARC+®Core (mA, with ASF = 1.00)[1]

f_{CCLK} (MHz)	Voltage (V_{DD_INT})			
	1.05	1.10	1.15	1.20
500	N/A	347	362	378
450	298	312	326	340
400	265	277	290	302
350	232	243	254	265
300	198	208	217	227
250	165	173	181	189
200	132	139	145	151
150	99	104	109	113
100	66	69	72	76

[1] N/A means not applicable.

Table 33. Dynamic Current for the ARM® Cortex®-A5 Core (mA, with ASF = 1.00)[1]

f_{CCLK} (MHz)	Voltage (V_{DD_INT})			
	1.05	1.10	1.15	1.20
500	N/A	88	92	96
450	76	79	83	86
400	67	70	74	77
350	59	62	64	67
300	50	53	55	58
250	42	44	46	48
200	34	35	37	39
150	25	26	28	29
100	17	18	18	19

[1] N/A means not applicable.

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) = 0.52 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK0\_DYN}$ (mA) = 0.28 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_SCLK1\_DYN}$ (mA) = 0.013 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_DCLK\_DYN}$ (mA) = 0.08 × f_{DCLK} (MHz) × V_{DD_INT} (V)

$I_{DD\_INT\_OCLK\_DYN}$ (mA) = 0.015 × 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.

$I_{DD\_INT\_USB\_DYN}$ = 9.6 mA (if USB is enabled in HS mode)

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

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

Data Transmission Current

The data transmission current represents the power dissipated when moving data throughout the system via DMA. This current is proportional to the data rate. Refer to the power calculator available with "Estimating Power for ADSP-SC57x/2157x SHARC+ Processors" (EE-397) to estimate $I_{DD\_INT\_DMA\_DR\_DYN}$ based on the bandwidth of the data transfer.

HADC

HADC Electrical Characteristics

Table 34. 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	60	µA

HADC DC Accuracy

Table 35. HADC DC Accuracy for CSP_BGA[1]

Parameter	Typ	Unit[2]
Resolution	9	Bits
No Missing Codes (NMC) – Unrestricted	9	Bits
No Missing Codes (NMC) – Pin Restrictions[3]	10	Bits
Integral Nonlinearity (INL)	±2	LSB
Differential Nonlinearity (DNL)	±2	LSB
Offset Error	±5	LSB
Offset Error Matching	±6	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 ÷ 512.
[3] Pin restrictions required: pins DAI18, DAI19, and DAI20 must be programmed to inputs and a static (non-switching) signal applied to the pins.

Table 36. HADC DC Accuracy for LQFP_EP[1]

Parameter	Typ	Unit[2]
Resolution	7	Bits
No Missing Codes (NMC) – Unrestricted	7	Bits
No Missing Codes (NMC) – Pin Restrictions[3]	9	Bits
Integral Nonlinearity (INL)	±2	LSB
Differential Nonlinearity (DNL)	±2	LSB
Offset Error	±5	LSB
Offset Error Matching	±6	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 ÷ 128.
[3] Pin restrictions required: pins DAI18, DAI19, and DAI20 must be programmed to inputs and a static (non-switching) signal applied to the pins.

HADC Timing Specifications

Table 37. HADC Timing Specifications

Parameter	Typ	Max	Unit
Conversion Time[1]	20 × T_{SAMPLE}		µs
Throughput Range		1	MSPS
T_{WAKEUP}		100	µs

[1] Refer to the ADSP-SC57x/ADSP-2157x SHARC+ Processor Hardware Reference for additional information about T_{SAMPLE}.

TMU

TMU Characteristics

Table 38. TMU Characteristics

Parameter	Typ	Unit
Resolution	1	°C
Accuracy	±8	°C

Table 39. TMU Gain and Offset

Junction Temperature Range	TMU_GAIN	TMU_OFFSET
–40°C to +40°C	Contact Analog Devices, Inc.	
40°C to 85°C	Contact Analog Devices, Inc.	
85°C to 133°C	Contact Analog Devices, Inc.	

DDR2 SDRAM Clock and Control Cycle Timing

Table 43 and Figure 10 show DDR2 SDRAM clock and control cycle timing, related to the DMC.

Table 43. DDR2 SDRAM Clock and Control Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		400 MHz[1] Min	400 MHz[1] Max	Unit
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	2.5		ns
$t_{CH(abs)}$[2]	Minimum Clock Pulse Width	0.48	0.52	t_{CK}
$t_{CL(abs)}$[2]	Maximum Clock Pulse Width	0.48	0.52	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	175		ps
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	250		ps

[1] To ensure proper operation of DDR2, all the DDR2 requirements must be strictly followed. See "Interfacing DDR3/DDR2/LPDDR Memory to ADSP-SC5xx/215xx Processors" (EE-387).
[2] As per JESD79-2E definition.

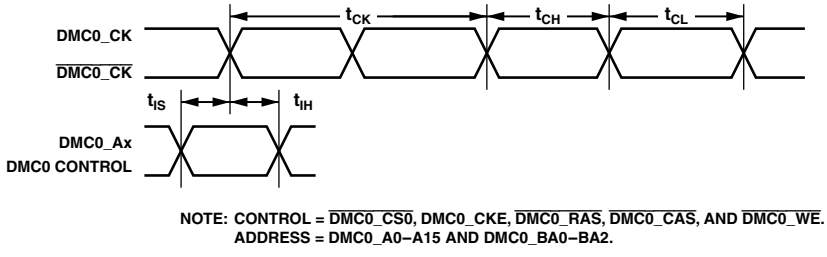

Figure 10. DDR2 SDRAM Clock and Control Cycle Timing

Enhanced Parallel Peripheral Interface (EPPI) Timing

Table 52 and Table 53 and Figure 19 through Figure 27 describe enhanced parallel peripheral interface (EPPI) timing operations. In Figure 19 through Figure 27, POLC[1:0] represents the setting of the EPPI_CTL register, which sets the sampling/driving edges of the EPPI clock.

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

$$f_{PCLKPROG} = \frac{f_{SCLK0}}{(VALUE + 1)}$$

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

When externally generated, the EPPI_CLK is called $f_{PCLKEXT}$:

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

Table 52. Enhanced Parallel Peripheral Interface (EPPI)—Internal Clock

Parameter		Min	Max	Unit
Timing Requirements				
t_{SFSPI}	External FS Setup Before EPPI_CLK	6.5		ns
t_{HFSPI}	External FS Hold After EPPI_CLK	0		ns
t_{SDRPI}	Receive Data Setup Before EPPI_CLK	6.5		ns
t_{HDRPI}	Receive Data Hold After EPPI_CLK	0		ns
t_{SFS3GI}	External FS3 Input Setup Before EPPI_CLK Fall Edge in Clock Gating Mode	14		ns
t_{HFS3GI}	External FS3 Input Hold Before EPPI_CLK Fall Edge in Clock Gating Mode	0		ns
Switching Characteristics				
t_{PCLKW}	EPPI_CLK Width[1]	$0.5 \times t_{PCLKPROG} - 1.5$		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKPROG} - 1.5$		ns
t_{DFSPI}	Internal FS Delay After EPPI_CLK		3.6	ns
t_{HOFSPI}	Internal FS Hold After EPPI_CLK	–0.72		ns
t_{DDTPI}	Transmit Data Delay After EPPI_CLK		3.5	ns
t_{HDTPI}	Transmit Data Hold After EPPI_CLK	–0.5		ns

[1] See Table 27 for details on the minimum period that can be programmed for $t_{PCLKPROG}$.

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

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

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

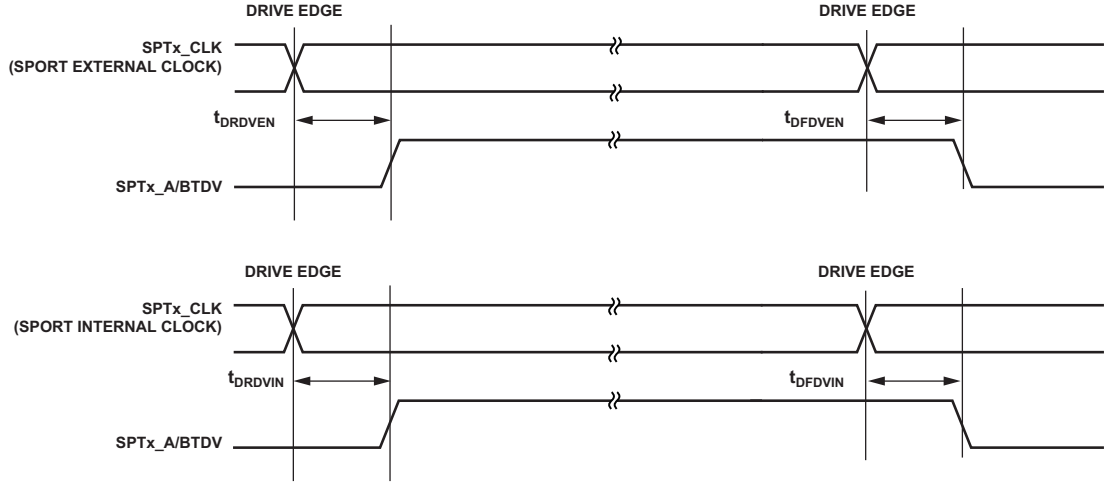

Figure 32. SPORTs—Transmit Data Valid Internal and External Clock

Table 64. SPI2 Port—Master Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	2.7		ns
t_{HSPIDM}	SPIx_CLK Sampling Edge to Data Input Invalid	0.75		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 1[2]	$t_{SPICLKPROG} - 5$		ns
	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 0[2]	$1.5 \times t_{SPICLKPROG} - 5$		ns
t_{SPICHM}	SPIx_CLK High Period[3]	$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLM}	SPIx_CLK Low Period[3]	$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLK}	SPIx_CLK Period[3]	$t_{SPICLKPROG} - 1.5$		ns
t_{HDSM}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 1[2]	$1.5 \times t_{SPICLKPROG} - 5$		ns
	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 0[2]	$t_{SPICLKPROG} - 5$		ns
t_{SPITDM}	Sequential Transfer Delay[2, 4]	$t_{SPICLKPROG} - 1.5$		ns
$t_{DDSPIDM}$	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		3.17	ns
$t_{HDSPIDM}$	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	–2.4		ns

[1] All specifications apply to SPI2 only.
[2] Specification assumes the LEADX and LAGX bits in the SPI_DLY register are 1.
[3] See Table 27 for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.
[4] Applies to sequential mode with STOP ≥ 1.

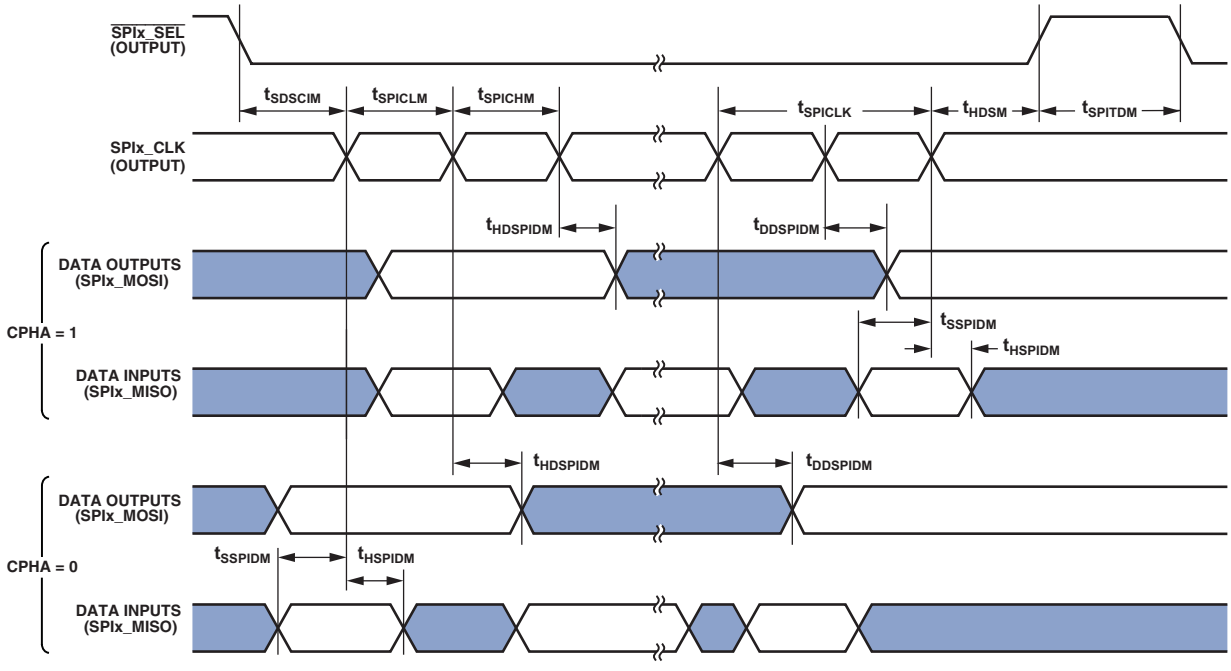

Figure 36. SPI Port—Master Timing

SPI Port—Slave Timing

SPI0, SPI1, and SPI2

Table 65, Table 66, and Figure 37 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 65. SPI0, SPI1 Port—Slave Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLK}	SPIx_CLK Period[2]	$t_{SPICLKEXT} - 1.5$		ns
t_{HDS}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SS}}$ Not Asserted	5		ns
t_{SPITDS}	Sequential Transfer Delay	$t_{SPICLKEXT} - 1.5$		ns
t_{SDSCI}	$\overline{\text{SPIx\_SS}}$ Assertion to First SPIx_CLK Edge	11.7		ns
t_{SSPID}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	2		ns
t_{HSPID}	SPIx_CLK Sampling Edge to Data Input Invalid	1.6		ns
Switching Characteristics				
t_{DSOE}	$\overline{\text{SPIx\_SS}}$ Assertion to Data Out Active	0	14.12	ns
t_{DSDHI}	$\overline{\text{SPIx\_SS}}$ Deassertion to Data High Impedance	0	12.6	ns
t_{DDSPID}	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		14.16	ns
t_{HDSPID}	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	1.5		ns

[1] All specifications apply to SPI0 and SPI1.
[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 27.

Table 66. SPI2 Port—Slave Timing[1]

Parameter		Min	Max	Unit
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	$0.5 \times t_{SPICLKEXT} - 1.5$		ns
t_{SPICLK}	SPIx_CLK Period[2]	$t_{SPICLKEXT} - 1.5$		ns
t_{HDS}	Last SPIx_CLK Edge to $\overline{SPIx\_SS}$ Not Asserted	5		ns
t_{SPITDS}	Sequential Transfer Delay	$t_{SPICLKEXT} - 1.5$		ns
t_{SDSCI}	$\overline{SPIx\_SS}$ 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}	$\overline{SPIx\_SS}$ Assertion to Data Out Active	0	14	ns
t_{DSDHI}	$\overline{SPIx\_SS}$ Deassertion to Data High Impedance	0	11.5	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)	1.5		ns

[1] All specifications apply to SPI2 only.
[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 27.

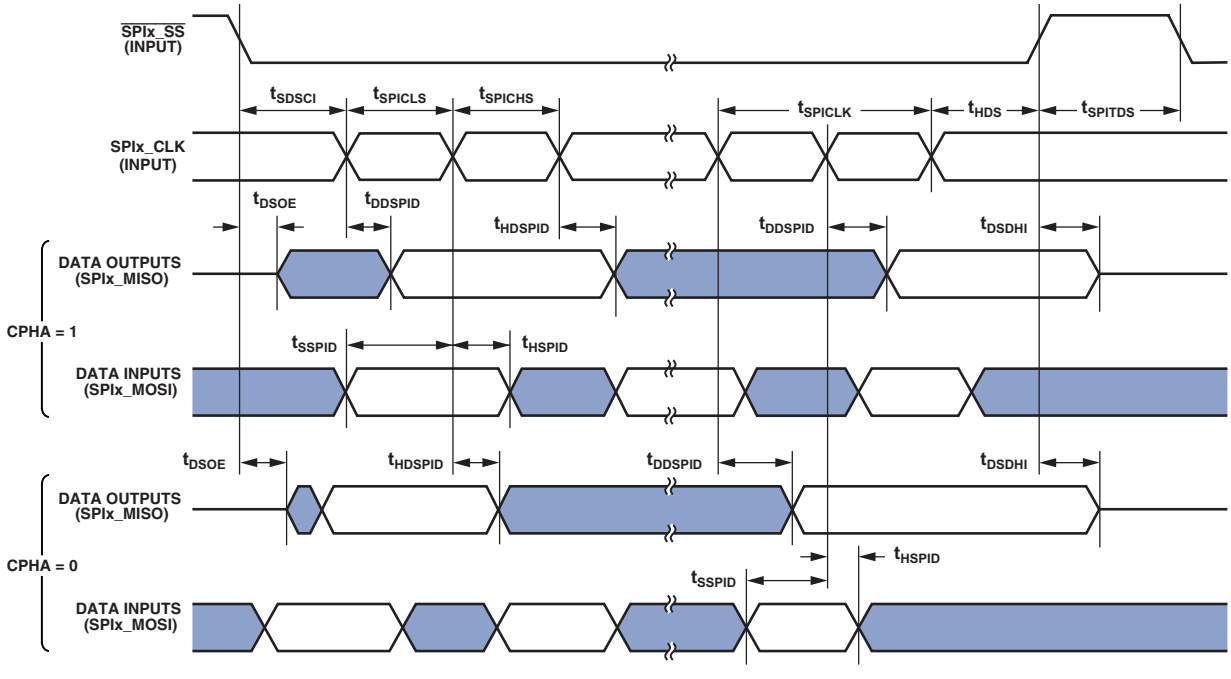

Figure 37. SPI Port—Slave Timing

Table 81. 10/100 EMAC Timing—RMII Receive Signal

Parameter[1]		Min	Max	Unit
Timing Requirements				
$t_{REFCLKF}$	ETH0_RXCLK_REFCLK Frequency (f_{SCLK0} = SCLK0 Frequency)	None	50 + 1%	MHz
$t_{REFCLKW}$	ETH0_RXCLK_REFCLK Width ($t_{REFCLKF}$ = ETH0_RXCLK_REFCLK Period)	$t_{REFCLK} \times 35\%$	$t_{REFCLK} \times 65\%$	ns
$t_{REFCLKIS}$	Rx Input Valid to RMII ETH0_RXCLK_REFCLK Rising Edge (Data In Setup)	1.75		ns
$t_{REFCLKIH}$	RMII ETH0_RXCLK_REFCLK Rising Edge to Rx Input Invalid (Data In Hold)	1.6		ns

[1] RMII inputs synchronous to RMII ETH0_RXCLK_REFCLK are ETH0_RXD1–0, RMII ETH0_RXCTL_RXDV, and ETH0_RXERR.

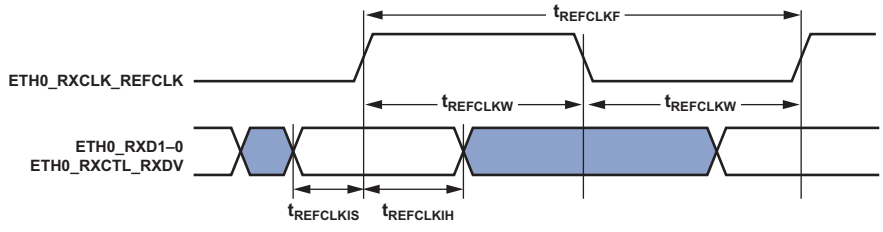

Figure 51. 10/100 EMAC Controller Timing—RMII Receive Signal

Table 82. 10/100 EMAC Timing—RMII Transmit Signal

Parameter[1]		Min	Max	Unit
Switching Characteristics				
$t_{REFCLKOV}$	RMII ETH0_RXCLK_REFCLK Rising Edge to Transmit Output Valid (Data Out Valid)		11.9	ns
$t_{REFCLKOH}$	RMII ETH0_RXCLK_REFCLK Rising Edge to Transmit Output Invalid (Data Out Hold)	2		ns

[1] RMII outputs synchronous to RMII ETH0_RXCLK_REFCLK are ETH0_TXD1–0.

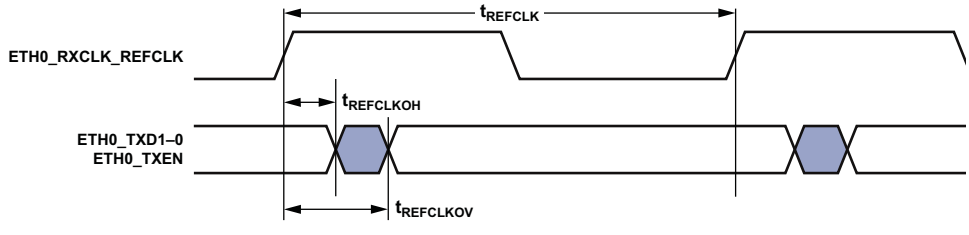

Figure 52. 10/100 EMAC Controller Timing—RMII Transmit Signal

S/PDIF Transmitter Input Data Timing

The timing requirements for the S/PDIF transmitter are given in Table 88. Input signals are routed to the DAI0_PINx pins using the SRU. Therefore, the timing specifications provided below are valid at the DAI0_PINx pins.

Table 88. S/PDIF Transmitter Input Data Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{SISFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	3		ns
t_{SIHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	3		ns
t_{SISD}[1]	Data Setup Before Serial Clock Rising Edge	3		ns
t_{SIHD}[1]	Data Hold After Serial Clock Rising Edge	3		ns
$t_{SITXCLKW}$	Transmit Clock Width	9		ns
$t_{SITXCLK}$	Transmit Clock Period	20		ns
$t_{SISCLKW}$	Clock Width	36		ns
t_{SISCLK}	Clock Period	80		ns

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

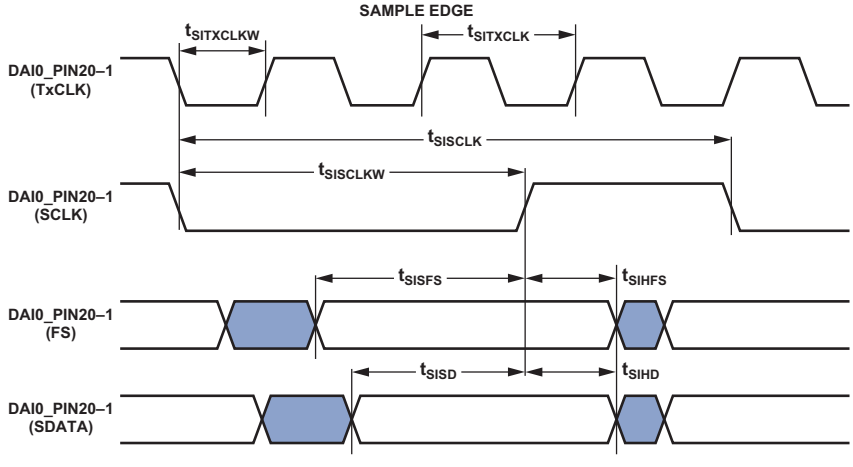

Figure 58. S/PDIF Transmitter Input Timing

Oversampling Clock (TxCLK) Switching Characteristics

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

Table 89. Oversampling Clock (TxCLK) Switching Characteristics

Parameter		Max	Unit
Switching Characteristics			
f_{TXCLK_384}	Frequency for TxCLK = 384 × Frame Sync	Oversampling ratio × frame sync ≤ $1/t_{SITXCLK}$	MHz
f_{TXCLK_256}	Frequency for TxCLK = 256 × Frame Sync	49.2	MHz
f_{FS}	Frame Rate (FS)	192.0	kHz

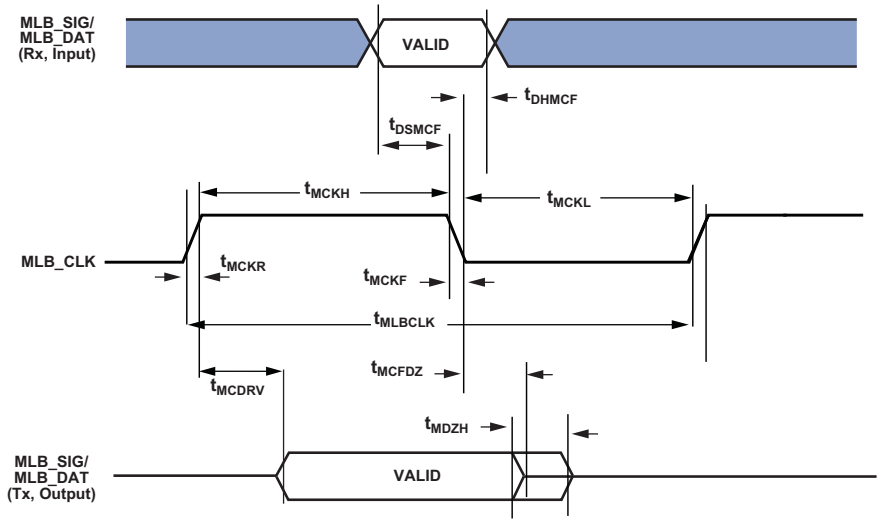

Figure 60. MLB Timing (3-Pin Interface)

The ac timing specifications of the 6-pin MLB interface is detailed in Table 92. Refer to the *Media Local Bus Specification version 4.2* for more details.

Table 92. 6-Pin MLB Interface Specifications

Parameter		Conditions	Min	Typ	Max	Unit
t_{MT}	Differential Transition Time at the Input Pin (See Figure 61)	20% to 80% $V_{IN}+/V_{IN}-$ 80% to 20% $V_{IN}+/V_{IN}-$			1	ns
f_{MCKE}	MLBCP/N External Clock Operating Frequency (See Figure 62)[1]	2048 × FS at 44.0 kHz	90.112			MHz
		2048 × FS at 50.0 kHz			102.4	MHz
f_{MCKR}	Recovered Clock Operating Frequency (Internal, Not Observable at Pins, Only for Timing References) (See Figure 62)	2048 × FS at 44.0 kHz	90.112			MHz
		2048 × FS at 50.0 kHz			102.4	MHz
t_{DELAY}	Transmitter MLBSP/N (MLBDP/N) Output Valid From Transition of MLBCP/N (Low to High) (See Figure 63)	f_{MCKR} = 2048 × FS	0.6		5	ns
t_{PHZ}	Disable Turnaround Time From Transition of MLBCP/N (Low to High) (See Figure 64)	f_{MCKR} = 2048 × FS	0.6		7	ns
t_{PLZ}	Enable Turnaround Time From Transition of MLBCP/N (Low to High) (See Figure 64)	f_{MCKR} = 2048 × FS	0.6		11.2	ns
t_{SU}	MLBSP/N (MLBDP/N) Valid to Transition of MLBCP/N (Low to High) (See Figure 63)	f_{MCKR} = 2048 × FS	1			ns
t_{HD}	MLBSP/N (MLBDP/N) Hold From Transition of MLBCP/N (Low to High) (See Figure 63)[2]		0.6			ns

[1] f_{MCKE} (maximum) and f_{MCKR} (maximum) include maximum cycle to cycle system jitter (t_{JITTER}) of 600 ps for a bit error rate of 10E-9.
[2] Receivers must latch MLBSP/N (MLBDP/N) data within t_{HD} (minimum) of the rising edge of MLBCP/N.

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
JTG_TDI	C16	PC_05	E01	PF_04	E18	VDD_DMC	U09
JTG_TDO	B14	PC_06	D01	PF_05	A19	VDD_DMC	U10
JTG_TMS	B15	PC_07	J02	PF_06	B18	VDD_DMC	U11
$\overline{\text{JTG\_TRST}}$	B10	PC_08	D02	PF_07	B20	VDD_DMC	U12
MLB0_CLKN	A06	PC_09	H02	PF_08	C19	VDD_DMC	U13
MLB0_CLKP	A07	PC_10	J03	PF_09	F18	VDD_DMC	U14
MLB0_DATN	B07	PC_11	C01	PF_10	D18	VDD_EXT	E07
MLB0_DATP	B08	PC_12	B01	PF_11	C20	VDD_EXT	E08
MLB0_SIGN	A08	PC_13	C02	SYS_BMODE0	Y18	VDD_EXT	E09
MLB0_SIGP	B09	PC_14	G02	SYS_BMODE1	W18	VDD_EXT	E10
PA_00	V12	PC_15	H03	SYS_BMODE2	Y19	VDD_EXT	E11
PA_01	V11	PD_00	W07	SYS_CLKIN0	A11	VDD_EXT	E12
PA_02	V10	PD_01	V06	SYS_CLKIN1	A14	VDD_EXT	E17
PA_03	V09	PD_02	W06	SYS_CLKOUT	D11	VDD_EXT	G04
PA_04	W10	PD_03	V04	SYS_FAULT	V14	VDD_EXT	H04
PA_05	V08	PD_04	V05	$\overline{\text{SYS\_FAULT}}$	W15	VDD_EXT	H17
PA_06	W09	PD_05	T01	$\overline{\text{SYS\_HWRST}}$	C11	VDD_EXT	J04
PA_07	W08	PD_06	R01	$\overline{\text{SYS\_RESOUT}}$	V13	VDD_EXT	J17
PA_08	V07	PD_07	R02	SYS_XTAL0	A10	VDD_EXT	K04
PA_09	A03	PD_08	R03	SYS_XTAL1	A13	VDD_EXT	K17
PA_10	A02	PD_09	M03	TWI0_SCL	T03	VDD_EXT	L04
PA_11	A04	PD_10	L02	TWI0_SDA	T02	VDD_EXT	L17
PA_12	C04	PD_11	L01	TWI1_SCL	U05	VDD_EXT	M04
PA_13	B03	PD_12	L03	TWI1_SDA	U03	VDD_EXT	M17
PA_14	C05	PD_13	K02	TWI2_SCL	W01	VDD_EXT	N17
PA_15	B04	PD_14	K01	TWI2_SDA	V01	VDD_EXT	T04
PB_00	E06	PD_15	K03	USB0_DM	A17	VDD_EXT	U06
PB_01	B05	PE_00	J01	USB0_DP	A16	VDD_HADC	R20
PB_02	C07	PE_01	H01	USB0_ID	C12	VDD_INT	F05
PB_03	C06	PE_02	E04	USB0_VBC	D13	VDD_INT	F07
PB_04	B06	PE_03	E03	USB0_VBUS	B11	VDD_INT	F08
PB_05	C10	PE_04	D03	USB0_CLKIN	C13	VDD_INT	F09
PB_06	C09	PE_05	E02	USB0_XTAL	B12	VDD_INT	F10
PB_07	D09	PE_06	F04	VDD_DMC	N04	VDD_INT	F11
PB_08	D08	PE_07	A05	VDD_DMC	T06	VDD_INT	F12
PB_09	D10	PE_08	D05	VDD_DMC	T07	VDD_INT	F13
PB_10	B13	PE_09	D07	VDD_DMC	T08	VDD_INT	F14
PB_11	D12	PE_10	C08	VDD_DMC	T09	VDD_INT	F16
PB_12	C14	PE_11	D06	VDD_DMC	T10	VDD_INT	G05
PB_13	C15	PE_12	D16	VDD_DMC	T11	VDD_INT	G16
PB_14	D14	PE_13	D15	VDD_DMC	T12	VDD_INT	H05
PB_15	G17	PE_14	C17	VDD_DMC	T13	VDD_INT	H16
PC_00	G01	PE_15	E15	VDD_DMC	T14	VDD_INT	J05
PC_01	G03	PF_00	B16	VDD_DMC	T15	VDD_INT	J16
PC_02	F01	PF_01	B17	VDD_DMC	T17	VDD_INT	K05
PC_03	F02	PF_02	F17	VDD_DMC	U07	VDD_INT	K16
PC_04	F03	PF_03	A18	VDD_DMC	U08	VDD_INT	L05