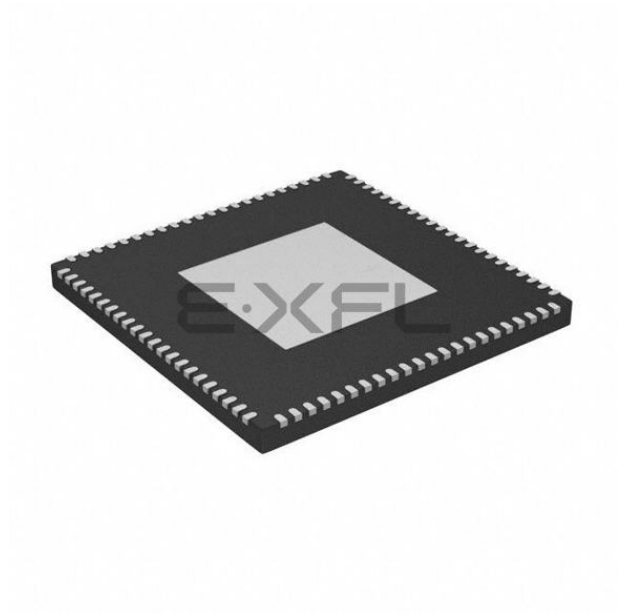


Welcome to E-XFL.COM

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	Blackfin+
Interface	CAN, DSPI, EBI/EMI, I²C, PPI, QSPI, SD/SDIO, SPI, SPORT, UART/USART, USB OTG
Clock Rate	400MHz
Non-Volatile Memory	ROM (512kB)
On-Chip RAM	512kB
Voltage - I/O	1.8V, 3.3V
Voltage - Core	1.10V
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	88-VFQFN Exposed Pad, CSP
Supplier Device Package	88-LFCSP-VQ (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adbf704wccpz411

GENERAL DESCRIPTION

The ADSP-BF70x processor is a member of the Blackfin® family of products. The Blackfin processor combines a dual-MAC 16-bit state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like microprocessor instruction set, and single-instruction, multiple-data (SIMD) multimedia capabilities into a single instruction-set architecture. New enhancements to the Blackfin+ core add 32-bit MAC and 16-bit complex MAC support, cache enhancements, branch prediction and other instruction set improvements—all while maintaining instruction set compatibility to previous Blackfin products.

The processor offers performance up to 400 MHz, as well as low static power consumption. Produced with a low-power and low-voltage design methodology, they provide world-class power management and performance.

By integrating a rich set of industry-leading system peripherals and memory (shown in Table 1), the Blackfin processor is the platform of choice for next-generation applications that require RISC-like programmability, multimedia support, and leading-edge signal processing in one integrated package. These applications span a wide array of markets, from automotive systems to embedded industrial, instrumentation, video/image analysis, biometric and power/motor control applications.

Table 1. Processor Comparison

Processor Feature		ADSP-BF700	ADSP-BF701	ADSP-BF702	ADSP-BF703	ADSP-BF704	ADSP-BF705	ADSP-BF706	ADSP-BF707
Maximum Speed Grade (MHz)[1]		200		400					
Maximum SYSCLK (MHz)		100		200					
Package Options		88-Lead LFCSP	184-Ball CSP_BGA	88-Lead LFCSP	184-Ball CSP_BGA	88-Lead LFCSP	184-Ball CSP_BGA	88-Lead LFCSP	184-Ball CSP_BGA
GPIOs		43	47	43	47	43	47	43	47
Memory (bytes)	L1 Instruction SRAM	48K							
	L1 Instruction SRAM/Cache	16K							
	L1 Data SRAM	32K							
	L1 Data SRAM/Cache	32K							
	L1 Scratchpad (L1 Data C)	8K							
	L2 SRAM	128K		256K		512K		1024K	
	L2 ROM	512K							
	DDR2/LPDDR (16-bit)	No	Yes	No	Yes	No	Yes	No	Yes
I^2C		1							
Up/Down/Rotary Counter		1							
GP Timer		8							
Watchdog Timer		1							
GP Counter		1							
SPORTs		2							
Quad SPI		2							
Dual SPI		1							
SPI Host Port		1							
USB 2.0 HS OTG		1							
Parallel Peripheral Interface		1							
CAN		2							
UART		2							
Real-Time Clock		1							
Static Memory Controller (SMC)		Yes							
Security Crypto Engine		Yes							
SD/SDIO (MSI)		4-bit	8-bit	4-bit	8-bit	4-bit	8-bit	4-bit	8-bit
4-Channel 12-Bit ADC		No	Yes	No	Yes	No	Yes	No	Yes

[1] Other speed grades available.

The program sequencer controls the flow of instruction execution, including instruction alignment and decoding. For program flow control, the sequencer supports PC relative and indirect conditional jumps (with dynamic branch prediction), and subroutine calls. Hardware supports zero-overhead looping. The architecture is fully interlocked, meaning that the programmer need not manage the pipeline when executing instructions with data dependencies.

The address arithmetic unit provides two addresses for simultaneous dual fetches from memory. It contains a multiported register file consisting of four sets of 32-bit index, modify, length, and base registers (for circular buffering), and eight additional 32-bit pointer registers (for C-style indexed stack manipulation).

The Blackfin processor supports a modified Harvard architecture in combination with a hierarchical memory structure. Level 1 (L1) memories are those that typically operate at the full processor speed with little or no latency. At the L1 level, the instruction memory holds instructions only. The data memory holds data, and a dedicated scratchpad data memory stores stack and local variable information.

In addition, multiple L1 memory blocks are provided, offering a configurable mix of SRAM and cache. The memory management unit (MMU) provides memory protection for individual tasks that may be operating on the core and can protect system registers from unintended access.

The architecture provides three modes of operation: user mode, supervisor mode, and emulation mode. User mode has restricted access to certain system resources, thus providing a protected software environment, while supervisor mode has unrestricted access to the system and core resources.

INSTRUCTION SET DESCRIPTION

The Blackfin processor instruction set has been optimized so that 16-bit opcodes represent the most frequently used instructions, resulting in excellent compiled code density. Complex DSP instructions are encoded into 32-bit opcodes, representing fully featured multifunction instructions. The Blackfin processor supports a limited multi-issue capability, where a 32-bit instruction can be issued in parallel with two 16-bit instructions, allowing the programmer to use many of the core resources in a single instruction cycle.

The Blackfin processor family assembly language instruction set employs an algebraic syntax designed for ease of coding and readability. The instructions have been specifically tuned to provide a flexible, densely encoded instruction set that compiles to a very small final memory size. The instruction set also provides fully featured multifunction instructions that allow the programmer to use many of the processor core resources in a single instruction. Coupled with many features more often seen on microcontrollers, this instruction set is very efficient when compiling C and C++ source code. In addition, the architecture supports both user (algorithm/application code) and supervisor (O/S kernel, device drivers, debuggers, ISRs) modes of operation, allowing multiple levels of access to core processor resources.

The assembly language, which takes advantage of the processor's unique architecture, offers the following advantages:

- Seamlessly integrated DSP/MCU features are optimized for both 8-bit and 16-bit operations.
- A multi-issue load/store modified-Harvard architecture, which supports two 16-bit MAC or four 8-bit ALU + two load/store + two pointer updates per cycle.
- All registers, I/O, and memory are mapped into a unified 4G byte memory space, providing a simplified programming model.
- Control of all asynchronous and synchronous events to the processor is handled by two subsystems: the core event controller (CEC) and the system event controller (SEC).
- Microcontroller features, such as arbitrary bit and bit-field manipulation, insertion, and extraction; integer operations on 8-, 16-, and 32-bit data-types; and separate user and supervisor stack pointers.
- Code density enhancements, which include intermixing of 16-bit and 32-bit instructions (no mode switching, no code segregation). Frequently used instructions are encoded in 16 bits.

PROCESSOR INFRASTRUCTURE

The following sections provide information on the primary infrastructure components of the ADSP-BF70x processor.

DMA Controllers

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

DMA transfers can occur between memory and a peripheral or between one memory and another memory. Each memory-to-memory DMA stream uses two channels, where one channel is the source channel, and the second is the destination channel.

All DMAs can transport data to and from all on-chip and off-chip memories. Programs can use two types of DMA transfers, descriptor-based or register-based. Register-based DMA allows the processor to directly program DMA control registers to initiate a DMA transfer. On completion, the control registers may be automatically updated with their original setup values for continuous transfer. Descriptor-based DMA transfers require a set of parameters stored within memory to initiate a DMA sequence. Descriptor-based DMA transfers allow multiple DMA sequences to be chained together and a DMA channel can be programmed to automatically set up and start another DMA transfer after the current sequence completes.

The DMA controller supports the following DMA operations.

- A single linear buffer that stops on completion.
- A linear buffer with negative, positive, or zero stride length.
- A circular, auto-refreshing buffer that interrupts when each buffer becomes full.

General-Purpose Timers

There is one GP timer unit, and it provides eight general-purpose programmable timers. Each timer has an external pin that can be configured either as a pulse width modulator (PWM) or timer output, as an input to clock the timer, or as a mechanism for measuring pulse widths and periods of external events. These timers can be synchronized to an external clock input on the TIMER_TMRx pins, an external TIMER_CLK input pin, or to the internal SCLK0.

These timer units can be used in conjunction with the UARTs and the CAN controller to measure the width of the pulses in the data stream to provide a software auto-baud detect function for the respective serial channels.

The GP timers can generate interrupts to the processor core, providing periodic events for synchronization to either the system clock or to external signals. Timer events can also trigger other peripherals through the TRU (for instance, to signal a fault). Each timer may also be started and/or stopped by any TRU master without core intervention.

Core Timer

The processor core also has its own dedicated timer. This extra timer is clocked by the internal processor clock and is typically used as a system tick clock for generating periodic operating system interrupts.

Watchdog Timer

The core includes a 32-bit timer, which may be used to implement a software watchdog function. A software watchdog can improve system availability by forcing the processor to a known state, through generation of a hardware reset, nonmaskable interrupt (NMI), or general-purpose interrupt, if the timer expires before being reset by software. The programmer initializes the count value of the timer, enables the appropriate interrupt, then enables the timer. Thereafter, the software must reload the counter before it counts down to zero from the programmed value. This protects the system from remaining in an unknown state where software that would normally reset the timer has stopped running due to an external noise condition or software error.

After a reset, software can determine if the watchdog was the source of the hardware reset by interrogating a status bit in its timer control register that is set only upon a watchdog-generated reset.

Serial Ports (SPORTs)

Two synchronous serial ports (comprised of four half-SPORTs) provide an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices such as Analog Devices' audio codecs, ADCs, and DACs. Each half-SPORT is made up of two data lines, a clock, and frame sync. The data lines can be programmed to either transmit or receive and each data line has a dedicated DMA channel.

Serial port data can be automatically transferred to and from on-chip memory/external memory through dedicated DMA channels. Each of the serial ports can work in conjunction with another serial port to provide TDM support. In this configuration, one SPORT provides two transmit signals while the other SPORT provides the two receive signals. The frame sync and clock are shared.

Serial ports operate in six modes:

- Standard DSP serial mode
- Multichannel (TDM) mode
- I^2S mode
- Packed I^2S mode
- Left-justified mode
- Right-justified mode

General-Purpose Counters

A 32-bit counter is provided that can operate in general-purpose up/down count modes and can sense 2-bit quadrature or binary codes as typically emitted by industrial drives or manual thumbwheels. Count direction is either controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can alternatively be used to input the push-button signal of thumbwheel devices. All three pins have a programmable debouncing circuit.

Internal signals forwarded to a GP timer enable this timer to measure the intervals between count events. Boundary registers enable auto-zero operation or simple system warning by interrupts when programmed count values are exceeded.

Parallel Peripheral Interface (PPI)

The processor provides a parallel peripheral interface (PPI) that supports data widths up to 18 bits. The PPI supports direct connection to TFT LCD panels, parallel analog-to-digital and digital-to-analog converters, video encoders and decoders, image sensor modules, and other general-purpose peripherals.

The following features are supported in the PPI module:

- Programmable data length: 8 bits, 10 bits, 12 bits, 14 bits, 16 bits, and 18 bits per clock.
- Various framed, non-framed, 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 decode.
- Optional packing and unpacking of data to/from 32 bits from/to 8 bits, 16 bits and 24 bits. If packing/unpacking is enabled, endianness can be configured to change the order of packing/unpacking of bytes/words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various de-interleaving/interleaving modes for receiving/transmitting 4:2:2 YCrCb data.
- Configurable LCD data enable (DEN) output available on Frame Sync 3.

184-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processor's pin definitions are shown in Table 7. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the signal name for every pin and (where applicable) the GPIO multiplexed pin function for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- General-Purpose Port: The Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 7. ADSP-BF70x 184-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
CAN0_RX	CAN0 Receive	C	PC_02
CAN0_TX	CAN0 Transmit	C	PC_03
CAN1_RX	CAN1 Receive	A	PA_12
CAN1_TX	CAN1 Transmit	A	PA_13
CNT0_DG	CNT0 Count Down and Gate	A	PA_07
CNT0_UD	CNT0 Count Up and Direction	A	PA_15
CNT0_ZM	CNT0 Count Zero Marker	A	PA_13
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_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
DMC0_CKE	DMC0 Clock enable	Not Muxed	DMC0_CKE
$\overline{\text{DMC0\_CK}}$	DMC0 Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
$\overline{\text{DMC0\_CS0}}$	DMC0 Chip Select 0	Not Muxed	$\overline{\text{DMC0\_CS0}}$
DMC0_DQ00	DMC0 Data 0	Not Muxed	DMC0_DQ00
DMC0_DQ01	DMC0 Data 1	Not Muxed	DMC0_DQ01
DMC0_DQ02	DMC0 Data 2	Not Muxed	DMC0_DQ02
DMC0_DQ03	DMC0 Data 3	Not Muxed	DMC0_DQ03
DMC0_DQ04	DMC0 Data 4	Not Muxed	DMC0_DQ04
DMC0_DQ05	DMC0 Data 5	Not Muxed	DMC0_DQ05
DMC0_DQ06	DMC0 Data 6	Not Muxed	DMC0_DQ06

Table 7. ADSP-BF70x 184-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC0_DQ07	DMC0 Data 7	Not Muxed	DMC0_DQ07
DMC0_DQ08	DMC0 Data 8	Not Muxed	DMC0_DQ08
DMC0_DQ09	DMC0 Data 9	Not Muxed	DMC0_DQ09
DMC0_DQ10	DMC0 Data 10	Not Muxed	DMC0_DQ10
DMC0_DQ11	DMC0 Data 11	Not Muxed	DMC0_DQ11
DMC0_DQ12	DMC0 Data 12	Not Muxed	DMC0_DQ12
DMC0_DQ13	DMC0 Data 13	Not Muxed	DMC0_DQ13
DMC0_DQ14	DMC0 Data 14	Not Muxed	DMC0_DQ14
DMC0_DQ15	DMC0 Data 15	Not Muxed	DMC0_DQ15
DMC0_LDM	DMC0 Data Mask for Lower Byte	Not Muxed	DMC0_LDM
DMC0_LDQS	DMC0 Data Strobe for Lower Byte	Not Muxed	DMC0_LDQS
$\overline{\text{DMC0\_LDQS}}$	DMC0 Data Strobe for Lower Byte (complement)	Not Muxed	$\overline{\text{DMC0\_LDQS}}$
DMC0_ODT	DMC0 On-die termination	Not Muxed	DMC0_ODT
$\overline{\text{DMC0\_RAS}}$	DMC0 Row Address Strobe	Not Muxed	$\overline{\text{DMC0\_RAS}}$
DMC0_UDM	DMC0 Data Mask for Upper Byte	Not Muxed	DMC0_UDM
DMC0_UDQS	DMC0 Data Strobe for Upper Byte	Not Muxed	DMC0_UDQS
$\overline{\text{DMC0\_UDQS}}$	DMC0 Data Strobe for Upper Byte (complement)	Not Muxed	$\overline{\text{DMC0\_UDQS}}$
DMC0_VREF	DMC0 Voltage Reference	Not Muxed	DMC0_VREF
$\overline{\text{DMC0\_WE}}$	DMC0 Write Enable	Not Muxed	$\overline{\text{DMC0\_WE}}$
GND	Ground	Not Muxed	GND
GND_HADC	Ground HADC	Not Muxed	GND_HADC
HADC0_VIN0	HADC0 Analog Input at channel 0	Not Muxed	HADC0_VIN0
HADC0_VIN1	HADC0 Analog Input at channel 1	Not Muxed	HADC0_VIN1
HADC0_VIN2	HADC0 Analog Input at channel 2	Not Muxed	HADC0_VIN2
HADC0_VIN3	HADC0 Analog Input at channel 3	Not Muxed	HADC0_VIN3
HADC0_VREFN	HADC0 Ground Reference for ADC	Not Muxed	HADC0_VREFN
HADC0_VREFP	HADC0 External Reference for ADC	Not Muxed	HADC0_VREFP
JTG_SWCLK	TAPC0 Serial Wire Clock	Not Muxed	JTG_TCK_SWCLK
JTG_SWDIO	TAPC0 Serial Wire DIO	Not Muxed	JTG_TMS_SWDIO
JTG_SWO	TAPC0 Serial Wire Out	Not Muxed	JTG_TDO_SWO
JTG_TCK	TAPC0 JTAG Clock	Not Muxed	JTG_TCK_SWCLK
JTG_TDI	TAPC0 JTAG Serial Data In	Not Muxed	JTG_TDI
JTG_TDO	TAPC0 JTAG Serial Data Out	Not Muxed	JTG_TDO_SWO
JTG_TMS	TAPC0 JTAG Mode Select	Not Muxed	JTG_TMS_SWDIO
$\overline{\text{JTG\_TRST}}$	TAPC0 JTAG Reset	Not Muxed	$\overline{\text{JTG\_TRST}}$
$\overline{\text{MSI0\_CD}}$	MSI0 Card Detect	A	PA_08
MSI0_CLK	MSI0 Clock	C	PC_09
MSI0_CMD	MSI0 Command	C	PC_05
MSI0_D0	MSI0 Data 0	C	PC_08
MSI0_D1	MSI0 Data 1	C	PC_04
MSI0_D2	MSI0 Data 2	C	PC_07
MSI0_D3	MSI0 Data 3	C	PC_06
MSI0_D4	MSI0 Data 4	C	PC_10
MSI0_D5	MSI0 Data 5	C	PC_11
MSI0_D6	MSI0 Data 6	C	PC_12
MSI0_D7	MSI0 Data 7	C	PC_13

Table 7. ADSP-BF70x 184-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SMC0_ARE	SMC0 Read Enable	A	PA_13
SMC0_AWE	SMC0 Write Enable	A	PA_14
SMC0_D00	SMC0 Data 0	B	PB_07
SMC0_D01	SMC0 Data 1	B	PB_06
SMC0_D02	SMC0 Data 2	B	PB_05
SMC0_D03	SMC0 Data 3	B	PB_04
SMC0_D04	SMC0 Data 4	B	PB_03
SMC0_D05	SMC0 Data 5	B	PB_02
SMC0_D06	SMC0 Data 6	B	PB_01
SMC0_D07	SMC0 Data 7	B	PB_00
SMC0_D08	SMC0 Data 8	B	PB_08
SMC0_D09	SMC0 Data 9	B	PB_09
SMC0_D10	SMC0 Data 10	B	PB_10
SMC0_D11	SMC0 Data 11	B	PB_11
SMC0_D12	SMC0 Data 12	B	PB_12
SMC0_D13	SMC0 Data 13	B	PB_13
SMC0_D14	SMC0 Data 14	B	PB_14
SMC0_D15	SMC0 Data 15	B	PB_15
SPI0_CLK	SPI0 Clock	B	PB_00
SPI0_CLK	SPI0 Clock	C	PC_04
SPI0_D2	SPI0 Data 2	B	PB_03
SPI0_D2	SPI0 Data 2	C	PC_08
SPI0_D3	SPI0 Data 3	B	PB_07
SPI0_D3	SPI0 Data 3	C	PC_09
SPI0_MISO	SPI0 Master In, Slave Out	B	PB_01
SPI0_MISO	SPI0 Master In, Slave Out	C	PC_06
SPI0_MOSI	SPI0 Master Out, Slave In	B	PB_02
SPI0_MOSI	SPI0 Master Out, Slave In	C	PC_07
SPI0_RDY	SPI0 Ready	A	PA_06
SPI0_SEL1	SPI0 Slave Select Output 1	A	PA_05
SPI0_SEL2	SPI0 Slave Select Output 2	A	PA_06
SPI0_SEL3	SPI0 Slave Select Output 3	C	PC_11
SPI0_SEL4	SPI0 Slave Select Output 4	B	PB_04
SPI0_SEL5	SPI0 Slave Select Output 5	B	PB_05
SPI0_SEL6	SPI0 Slave Select Output 6	B	PB_06
SPI0_SS	SPI0 Slave Select Input	A	PA_05
SPI1_CLK	SPI1 Clock	A	PA_00
SPI1_MISO	SPI1 Master In, Slave Out	A	PA_01
SPI1_MOSI	SPI1 Master Out, Slave In	A	PA_02
SPI1_RDY	SPI1 Ready	A	PA_03
SPI1_SEL1	SPI1 Slave Select Output 1	A	PA_04
SPI1_SEL2	SPI1 Slave Select Output 2	A	PA_03
SPI1_SEL3	SPI1 Slave Select Output 3	C	PC_10
SPI1_SEL4	SPI1 Slave Select Output 4	A	PA_14
SPI1_SS	SPI1 Slave Select Input	A	PA_04
SPI2_CLK	SPI2 Clock	B	PB_10

Table 7. ADSP-BF70x 184-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
UART0_RX	UART0 Receive	B	PB_09
UART0_TX	UART0 Transmit	B	PB_08
UART1_CTS	UART1 Clear to Send	B	PB_14
UART1_RTS	UART1 Request to Send	B	PB_13
UART1_RX	UART1 Receive	C	PC_01
UART1_TX	UART1 Transmit	C	PC_00
USB0_CLKIN	USB0 Clock/Crystal Input	Not Muxed	USB0_CLKIN
USB0_DM	USB0 Data –	Not Muxed	USB0_DM
USB0_DP	USB0 Data +	Not Muxed	USB0_DP
USB0_ID	USB0 OTG ID	Not Muxed	USB0_ID
USB0_VBC	USB0 VBUS Control	Not Muxed	USB0_VBC
USB0_VBUS	USB0 Bus Voltage	Not Muxed	USB0_VBUS
USB0_XTAL	USB0 Crystal	Not Muxed	USB0_XTAL
VDD_DMC	VDD for DMC	Not Muxed	VDD_DMC
VDD_EXT	External VDD	Not Muxed	VDD_EXT
VDD_HADC	VDD for HADC	Not Muxed	VDD_HADC
VDD_INT	Internal VDD	Not Muxed	VDD_INT
VDD_OTP	VDD for OTP	Not Muxed	VDD_OTP
VDD_RTC	VDD for RTC	Not Muxed	VDD_RTC
VDD_USB	VDD for USB	Not Muxed	VDD_USB

Table 15. ADSP-BF70x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
DMC0_DQ08	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 8 Notes: No notes.
DMC0_DQ09	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 9 Notes: No notes.
DMC0_DQ10	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 10 Notes: No notes.
DMC0_DQ11	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 11 Notes: No notes.
DMC0_DQ12	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 12 Notes: No notes.
DMC0_DQ13	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 13 Notes: No notes.
DMC0_DQ14	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 14 Notes: No notes.
DMC0_DQ15	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 15 Notes: No notes.
DMC0_LDM	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Lower Byte Notes: No notes.
DMC0_LDQS	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte Notes: For LPDDR, a pull-down is required.
$\overline{\text{DMC0\_LDQS}}$	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte (complement) Notes: For single ended DDR2, connect to DMC0_VREF. For LPDDR, leave unconnected.
DMC0_ODT	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 On-die termination Notes: For LPDDR, leave unconnected.
$\overline{\text{DMC0\_RAS}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Row Address Strobe Notes: No notes.
DMC0_UDM	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Upper Byte Notes: No notes.
DMC0_UDQS	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte Notes: For LPDDR, a pull-down is required.
$\overline{\text{DMC0\_UDQS}}$	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte (complement) Notes: For single ended DDR2, connect to DMC0_VREF. For LPDDR, leave unconnected.
DMC0_VREF	a	na	none	none	none	none	none	VDD_DMC	Desc: DMC0 Voltage Reference Notes: For LPDDR, leave unconnected. If the DMC is not used, connect to ground.
$\overline{\text{DMC0\_WE}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Write Enable Notes: No notes.
GND	g	na	none	none	none	none	none	na	Desc: Ground Notes: No notes.

Table 15. ADSP-BF70x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PA_02	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SPI1 Master Out, Slave In | TRACE0 Trace Data 5 | SMC0 Memory Select 1 Notes: May require a pull-up if used as an SMC memory select. Check the data sheet requirements of the IC it connects to.
PA_03	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SPI1 Slave Select Output 2 | SPI1 Ready | SMC0 Asynchronous Ready Notes: May require a pull-up or pull-down if used as an SMC asynchronous ready. Check the data sheet requirements of the IC it connects to and the programmed polarity.
PA_04	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SPI1 Slave Select Output 1 | TM0 Timer 7 | SPI2 Ready | SMC0 Address 8 | SPI1 Slave Select Input Notes: SPI slave select outputs require a pull-up when used.
PA_05	I/O	A	none	none	none	none	none	VDD_EXT	Desc: TM0 Timer 0 | SPI0 Slave Select Output 1 | SMC0 Address 7 | SPI0 Slave Select Input Notes: SPI slave select outputs require a pull-up when used.
PA_06	I/O	A	none	none	none	none	none	VDD_EXT	Desc: TM0 Timer 1 | SPI0 Slave Select Output 2 | SPI0 Ready | SMC0 Address 6 Notes: SPI slave select outputs require a pull-up when used.
PA_07	I/O	A	none	none	none	none	none	VDD_EXT	Desc: TM0 Timer 2 | SPT1 Channel B Transmit Data Valid | SPT1 Channel A Transmit Data Valid | SMC0 Address 5 | CNT0 Count Down and Gate Notes: No notes.
PA_08	I/O	A	none	none	none	none	none	VDD_EXT	Desc: PPI0 Data 11 | MSI0 Card Detect | SPT1 Channel A Clock | SMC0 Address 1 Notes: An external pull-up may be required for MSI modes, see the MSI chapter in the hardware reference for details.
PA_09	I/O	A	none	none	none	none	none	VDD_EXT	Desc: PPI0 Data 10 | TM0 Timer 4 | SPT1 Channel A Frame Sync | SMC0 Address 2 Notes: No notes.
PA_10	I/O	A	none	none	none	none	none	VDD_EXT	Desc: PPI0 Data 9 | TM0 Timer 5 | SPT1 Channel A Data 0 | SMC0 Address 3 Notes: No notes.
PA_11	I/O	A	none	none	none	none	none	VDD_EXT	Desc: PPI0 Data 8 | TM0 Timer 6 | SPT1 Channel A Data 1 | SMC0 Address 4 Notes: No notes.

Total Internal Power Dissipation

Total power dissipation has two components:

1. Static, including leakage current (deep sleep)
2. Dynamic, due to transistor switching characteristics for each clock domain

Many operating conditions can also affect power dissipation, including temperature, voltage, operating frequency, and processor activity. The following equation describes the internal current consumption.

$$I_{DDINT\_TOT} = I_{DDINT\_DEEPSLEEP} + I_{DDINT\_CCLK\_DYN} + I_{DDINT\_PLLCLK\_DYN} + I_{DDINT\_SYSCLK\_DYN} + I_{DDINT\_SCLK0\_DYN} + I_{DDINT\_SCLK1\_DYN} + I_{DDINT\_DCLK\_DYN} + I_{DDINT\_DMA\_DR\_DYN} + I_{DDINT\_USBCLK\_DYN}$$

$I_{DDINT\_DEEPSLEEP}$ is the only item present that is part of the static power dissipation component. $I_{DDINT\_DEEPSLEEP}$ is specified as a function of voltage (V_{DD_INT}) and temperature (see Table 21).

There are eight different items that contribute to the dynamic power dissipation. These components fall into three broad categories: application-dependent currents, clock currents, and data transmission currents.

Application-Dependent Current

The application-dependent currents include the dynamic current in the core clock domain.

Core clock (CCLK) use is subject to an activity scaling factor (ASF) that represents application code running on the processor cores and L1/L2 memories (Table 22). The ASF is combined with the CCLK frequency and V_{DD_INT} dependent data in Table 23 to calculate this portion.

$I_{DDINT\_CCLK\_DYN}$ (mA) = Table 23 × *ASF*

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.

$I_{DDINT\_PLLCLK\_DYN}$ (mA) = 0.012 × f_{PLLCLK} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SYSCLK\_DYN}$ (mA) = 0.120 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK0\_DYN}$ (mA) = 0.110 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK1\_DYN}$ (mA) = 0.068 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_DCLK\_DYN}$ (mA) = 0.055 × f_{DCLK} (MHz) × V_{DD_INT} (V)

The dynamic component of the USB clock is a unique case. The USB clock contributes a near constant current value when used.

Table 20. $I_{DDINT\_USBCLK\_DYN}$ Current

Is USB Enabled?	$I_{DDINT\_USBCLK\_DYN}$ (mA)
Yes – High-Speed Mode	13.94
Yes – Full-Speed Mode	10.83
Yes – Suspend Mode	5.2
No	0.34

Data Transmission Current

The data transmission current represents the power dissipated when transmitting data. This current is expressed in terms of data rate. The calculation is performed by adding the data rate (MB/s) of each DMA-driven access to peripherals, L1, L2, and external memory. This number is then multiplied by a weighted data-rate coefficient and V_{DD_INT}:

$I_{DDINT\_DMADR\_DYN}$ (mA) = *Weighted DRC* × *Total Data Rate* (MB/s) × V_{DD_INT} (V)

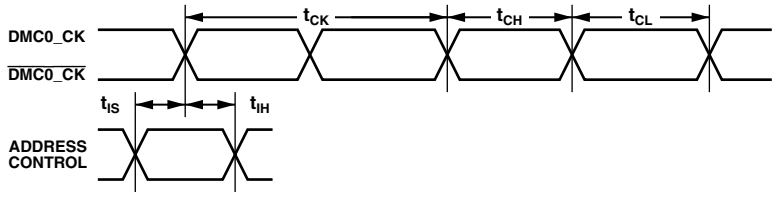
A weighted data-rate coefficient is used because different coefficients exist depending on the source and destination of the transfer. For details on using this equation and calculating the weighted DRC, see the related Engineer Zone material. For a quick maximum calculation, the weighted DRC can be assumed to be 0.0497, which is the coefficient for L1 to L1 transfers.

Mobile DDR SDRAM Clock and Control Cycle Timing

Table 42 and Figure 20 show mobile DDR SDRAM clock and control cycle timing, related to the dynamic memory controller (DMC).

Table 42. Mobile DDR SDRAM Clock and Control Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		200 MHz Min	200 MHz Max	Unit
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	5		ns
t_{CH}	Minimum Clock Pulse Width	0.45	0.55	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.45	0.55	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	1.5		ns
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	1.5		ns



NOTE: CONTROL = $\overline{\text{DMC0\_CS0}}$, DMC0_CKE, $\overline{\text{DMC0\_RAS}}$, $\overline{\text{DMC0\_CAS}}$, AND $\overline{\text{DMC0\_WE}}$.
ADDRESS = DMC0_A00–13, AND DMC0_BA0–2.

Figure 20. Mobile DDR SDRAM Clock and Control Cycle Timing

Table 50. Serial Ports—Internal Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSI}	Frame Sync Setup Before SPT_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[1]	17		14.5		ns
t_{HFSI}	Frame Sync Hold After SPT_CLK (Externally Generated Frame Sync in Either Transmit or Receive Mode)[1]	–0.5		–0.5		ns
t_{SDRI}	Receive Data Setup Before SPT_CLK[1]	6.5		5		ns
t_{HDRI}	Receive Data Hold After SPT_CLK[1]	1.5		1		ns
Switching Characteristics						
t_{DFSI}	Frame Sync Delay After SPT_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[2]		2		2	ns
t_{HOFSI}	Frame Sync Hold After SPT_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[2]	–4.5		–3.5		ns
t_{DDTI}	Transmit Data Delay After SPT_CLK[2]		2		2	ns
t_{HDTI}	Transmit Data Hold After SPT_CLK[2]	–5		–3.5		ns
t_{SCLKIW}	SPT_CLK Width[3]	$0.5 \times t_{SPTCLKPROG} - 1.5$		$0.5 \times t_{SPTCLKPROG} - 1.5$		ns
$t_{SPTCLKI}$	SPT_CLK Period[3]	$t_{SPTCLKPROG} - 1.5$		$t_{SPTCLKPROG} - 1.5$		ns

[1] Referenced to the sample edge.
[2] Referenced to drive edge.
[3] See Table 18 on Page 52 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{SPTCLKPROG}$.

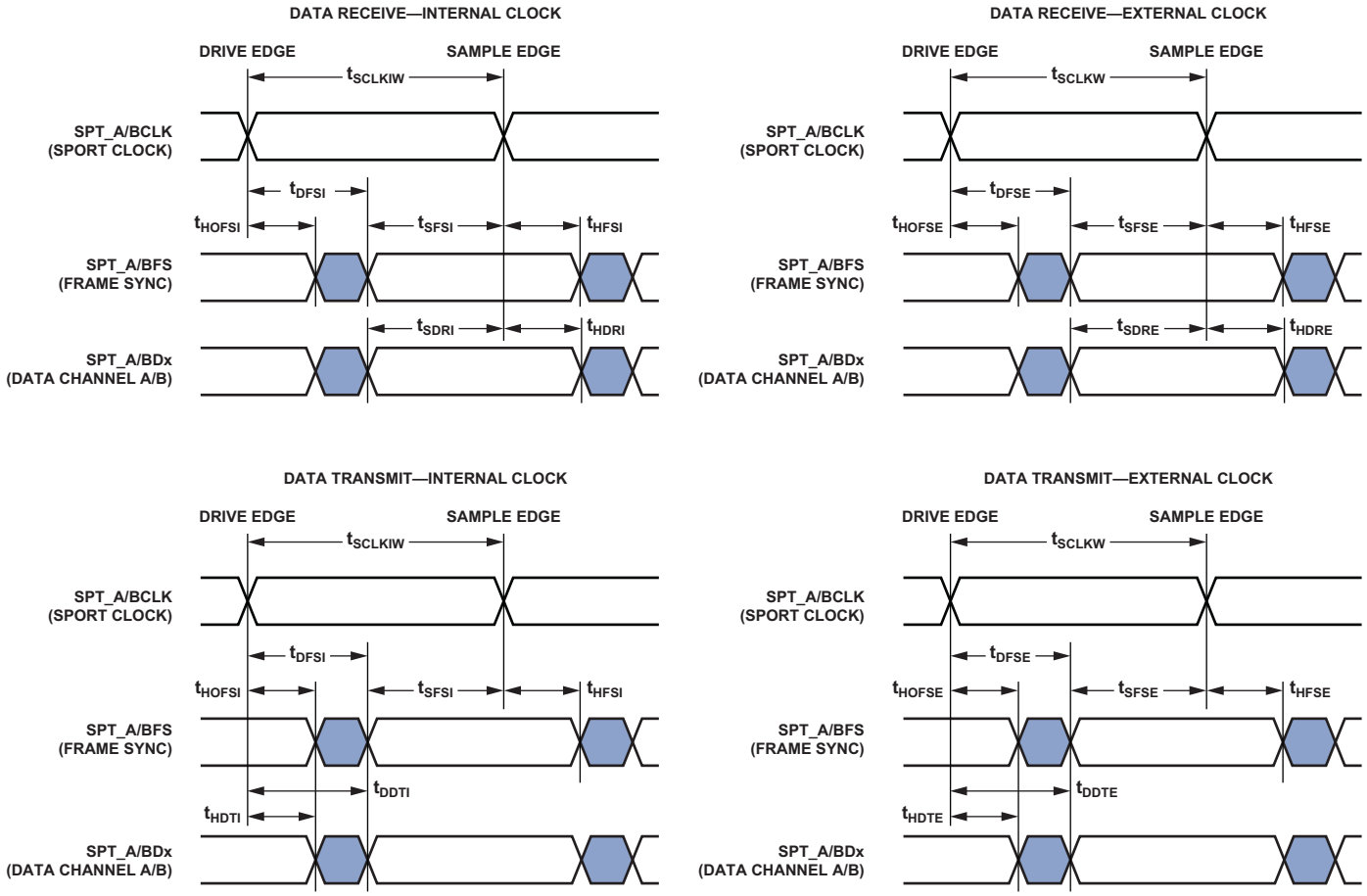


Figure 27. Serial Ports

Table 51. Serial Ports—Enable and Three-State

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Switching Characteristics						
t_{DDTEN}	Data Enable from External Transmit SPT_CLK[1]	1		1		ns
t_{DDTTE}	Data Disable from External Transmit SPT_CLK[1]		14		14	ns
t_{DDTIN}	Data Enable from Internal Transmit SPT_CLK[1]	–1.12		–1		ns
t_{DDTTI}	Data Disable from Internal Transmit SPT_CLK[1]		2.8		2.8	ns

[1] Referenced to drive edge.

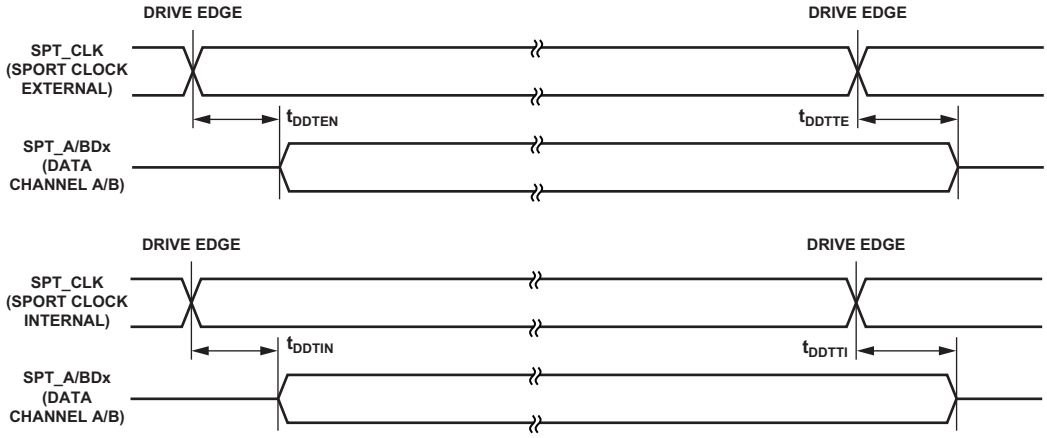


Figure 28. Serial Ports—Enable and Three-State

Serial Peripheral Interface (SPI) Port—Master Timing

Table 54 and Figure 31 describe serial peripheral interface (SPI) port master operations.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in MHz is set by the following equation where BAUD is a field in the SPI_CLK register that can be set from 0 to 65,535:

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

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

Note that:

- In dual mode data transmit, the SPI_MISO signal is also an output.
- In quad mode data transmit, the SPI_MISO, SPI_D2, and SPI_D3 signals are also outputs.
- In dual mode data receive, the SPI_MOSI signal is also an input.
- In quad mode data receive, the SPI_MOSI, SPI_D2, and SPI_D3 signals are also inputs.
- To add additional frame delays, see the documentation for the SPI_DLY register in the hardware reference manual.

Table 54. Serial Peripheral Interface (SPI) Port—Master Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SSPIDM}	Data Input Valid to SPI_CLK Edge (Data Input Setup)	6.5		5.5		ns
t_{HSPIDM}	SPI_CLK Sampling Edge to Data Input Invalid	1		1		ns
Switching Characteristics						
t_{SDSCIM}	$\overline{SPI\_SEL}$ low to First SPI_CLK Edge	$0.5 \times t_{SCLK0} - 2.5$		$0.5 \times t_{SCLK0} - 1.5$		ns
t_{SPICHM}	SPI_CLK High Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLM}	SPI_CLK Low Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLK}	SPI_CLK Period[1]	$t_{SPICLKPROG} - 1.5$		$t_{SPICLKPROG} - 1.5$		ns
t_{HDSM}	Last SPI_CLK Edge to $\overline{SPI\_SEL}$ High	$(0.5 \times t_{SCLK0}) - 2.5$		$(0.5 \times t_{SCLK0}) - 1.5$		ns
t_{SPITDM}	Sequential Transfer Delay[2]	$(STOP \times t_{SPICLK}) - 1.5$		$(STOP \times t_{SPICLK}) - 1.5$		ns
$t_{DDSPIDM}$	SPI_CLK Edge to Data Out Valid (Data Out Delay)		2.5		2	ns
$t_{HDSPIDM}$	SPI_CLK Edge to Data Out Invalid (Data Out Hold)	–4.5		–3.5		ns

[1] See Table 18 on Page 52 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.
[2] STOP value set using the SPI_DLY.STOP bits.

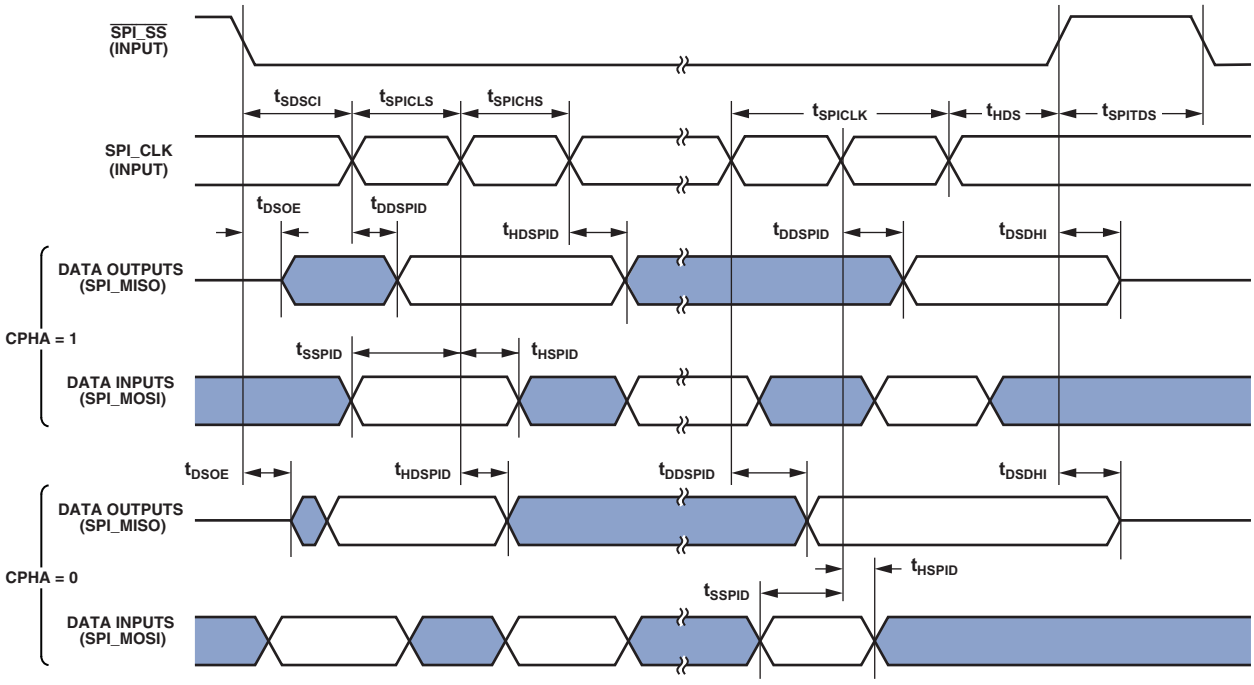


Figure 32. Serial Peripheral Interface (SPI) Port—Slave Timing

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

The universal asynchronous receiver-transmitter (UART) ports receive and transmit operations are described in the *ADSP-BF70x Blackfin+ Processor Hardware Reference*.

Controller Area Network (CAN) Interface

The controller area network (CAN) interface timing is described in the *ADSP-BF70x Blackfin+ Processor Hardware Reference*.

Universal Serial Bus (USB) On-The-Go—Receive and Transmit Timing

Table 62 describes the universal serial bus (USB) on-the-go receive and transmit operations.

Table 62. USB On-The-Go—Receive and Transmit Timing

		V_{DD_USB} 3.3 V Nominal		
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
f_{USBS}	USB_XI Frequency	24	24	MHz
fs_{USB}	USB_XI Clock Frequency Stability	–50	+50	ppm

ENVIRONMENTAL CONDITIONS

To determine the junction temperature on the application printed circuit board, use the following equation:

$$T_J = T_{CASE} + (\Psi_{JT} \times P_D)$$

where:

T_J = Junction temperature (°C).

T_{CASE} = Case temperature (°C) measured by customer at top center of package.

Ψ_{JT} = From Table 65 and Table 66.

P_D = Power dissipation (see Total Internal Power Dissipation on Page 56 for the method to calculate P_D).

Values of θ_{JA} are provided for package comparison and printed circuit board design considerations. θ_{JA} can be used for a first order approximation of T_J by the equation:

$$T_J = T_A + (\theta_{JA} \times P_D)$$

where:

T_A = Ambient temperature (°C).

Values of θ_{JC} are provided for package comparison and printed circuit board design considerations when an external heat sink is required.

In Table 65 and Table 66, airflow measurements comply with JEDEC standards JESD51-2 and JESD51-6. The junction-to-case measurement complies with MIL-STD-883 (Method 1012.1). All measurements use a 2S2P JEDEC test board.

Table 65. Thermal Characteristics for CSP_BGA

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	28.7	°C/W
θ_{JMA}	1 linear m/s air flow	26.2	°C/W
θ_{JMA}	2 linear m/s air flow	25.2	°C/W
θ_{JC}		10.1	°C/W
Ψ_{JT}	0 linear m/s air flow	0.24	°C/W
Ψ_{JT}	1 linear m/s air flow	0.40	°C/W
Ψ_{JT}	2 linear m/s air flow	0.51	°C/W

Table 66. Thermal Characteristics for LFCSP (QFN)

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	22.9	°C/W
θ_{JMA}	1 linear m/s air flow	17.9	°C/W
θ_{JMA}	2 linear m/s air flow	16.4	°C/W
θ_{JC}		2.26	°C/W
Ψ_{JT}	0 linear m/s air flow	0.14	°C/W
Ψ_{JT}	1 linear m/s air flow	0.27	°C/W
Ψ_{JT}	2 linear m/s air flow	0.30	°C/W

Table 68. ADSP-BF70x 184-Ball CSP_BGA Ball Assignments (Alphabetical by Signal Name)

Signal Name	Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name	Ball No.
DMC0_A00	D01	DMC0_WE	B06	PA_08	D12	SYS_HWRST	C13
DMC0_A01	F01	GND	C08	PA_09	G12	SYS_NMI	C07
DMC0_A02	F02	GND	A01	PA_10	H12	SYS_RESOUT	J03
DMC0_A03	G01	GND	A14	PA_11	H13	SYS_XTAL	N14
DMC0_A04	D02	GND	F06	PA_12	K12	TWI0_SCL	L03
DMC0_A05	E02	GND	F07	PA_13	J12	TWI0_SDA	L02
DMC0_A06	E01	GND	F08	PA_14	P13	USB0_CLKIN	P06
DMC0_A07	B01	GND	F09	PA_15	N13	USB0_DM	P07
DMC0_A08	B02	GND	G05	PB_00	N10	USB0_DP	N07
DMC0_A09	A02	GND	G06	PB_01	M11	USB0_ID	N06
DMC0_A10	B04	GND	G07	PB_02	L12	USB0_VBC	M07
DMC0_A11	B03	GND	G08	PB_03	M12	USB0_VBUS	M06
DMC0_A12	B05	GND	G09	PB_04	M10	USB0_XTAL	P05
DMC0_A13	A08	GND	G10	PB_05	M09	VDD_DMC	D06
DMC0_BA0	A03	GND	H05	PB_06	N09	VDD_DMC	D07
DMC0_BA1	A04	GND	H06	PB_07	P08	VDD_DMC	D08
DMC0_BA2	A05	GND	H07	PB_08	N08	VDD_DMC	D09
DMC0_CAS	A06	GND	H08	PB_09	M08	VDD_DMC	E06
DMC0_CK	A10	GND	H09	PB_10	P03	VDD_DMC	E07
DMC0_CKE	B09	GND	H10	PB_11	N03	VDD_DMC	E08
DMC0_CK	A11	GND	J06	PB_12	M04	VDD_DMC	E09
DMC0_CS0	B07	GND	J07	PB_13	P02	VDD_DMC	F10
DMC0_DQ00	B10	GND	J08	PB_14	N02	VDD_DMC	F11
DMC0_DQ01	B12	GND	J09	PB_15	M03	VDD_DMC	G11
DMC0_DQ02	B11	GND	L14	PC_00	M01	VDD_DMC	H11
DMC0_DQ03	B14	GND	P01	PC_01	K02	VDD_EXT	K05
DMC0_DQ04	B13	GND	P14	PC_02	K03	VDD_EXT	K06
DMC0_DQ05	D14	GND_HADC	J10	PC_03	L01	VDD_EXT	K07
DMC0_DQ06	D13	HADC0_VIN0	P12	PC_04	K01	VDD_EXT	K08
DMC0_DQ07	E14	HADC0_VIN1	N12	PC_05	J01	VDD_EXT	K09
DMC0_DQ08	E13	HADC0_VIN2	N11	PC_06	J02	VDD_EXT	L07
DMC0_DQ09	F14	HADC0_VIN3	P11	PC_07	H01	VDD_EXT	L08
DMC0_DQ10	F13	HADC0_VREFN	P09	PC_08	G03	VDD_EXT	L09
DMC0_DQ11	G13	HADC0_VREFP	P10	PC_09	F03	VDD_HADC	K10
DMC0_DQ12	G14	JTG_TCK_SWCLK	C03	PC_10	H02	VDD_INT	E05
DMC0_DQ13	J13	JTG_TDI	E03	PC_11	N05	VDD_INT	F04
DMC0_DQ14	K14	JTG_TDO_SWO	C01	PC_12	M05	VDD_INT	F05
DMC0_DQ15	K13	JTG_TMS_SWDIO	C02	PC_13	P04	VDD_INT	G04
DMC0_LDM	M13	JTG_TRST	D03	PC_14	N04	VDD_INT	H04
DMC0_LDQS	A12	PA_00	G02	RTC0_CLKIN	M02	VDD_INT	J04
DMC0_LDQS	A13	PA_01	C04	RTC0_XTAL	N01	VDD_OTP	J11
DMC0_ODT	B08	PA_02	C06	SYS_BMODE0	E12	VDD_RTC	J05
DMC0_RAS	A07	PA_03	A09	SYS_BMODE1	C14	VDD_USB	L06
DMC0_UDM	L13	PA_04	C09	SYS_CLKIN	M14		
DMC0_UDQS	J14	PA_05	C10	SYS_CLKOUT	H03		
DMC0_UDQS	H14	PA_06	C11	SYS_EXTWAKE	C05		
DMC0_VREF	E10	PA_07	C12	SYS_FAULT	F12		

Table 70. ADSP-BF70x 12 mm × 12 mm 88 -Lead LFCSP (QFN) Lead Assignments (Alphabetical by Signal Name)

Signal Name	Lead No.	Signal Name	Lead No.	Signal Name	Lead No.	Signal Name	Lead No.
GND	76	PB_00	52	PC_07	5	USB0_VBUS	34
GND	89	PB_01	48	PC_08	3	USB0_XTAL	31
JTG_TCK_SWCLK	85	PB_02	47	PC_09	2	VDD_EXT	4
JTG_TDI	86	PB_03	46	PC_10	1	VDD_EXT	11
JTG_TDO_SWO	83	PB_04	45	RTC0_CLKIN	21	VDD_EXT	17
JTG_TMS_SWDIO	84	PB_05	44	RTC0_XTAL	20	VDD_EXT	26
$\overline{\text{JTG\_TRST}}$	87	PB_06	43	SYS_BMODE0	66	VDD_EXT	41
PA_00	88	PB_07	42	SYS_BMODE1	67	VDD_EXT	50
PA_01	80	PB_08	40	SYS_CLKIN	57	VDD_EXT	55
PA_02	78	PB_09	39	SYS_CLKOUT	12	VDD_EXT	62
PA_03	75	PB_10	29	SYS_EXTWAKE	79	VDD_EXT	72
PA_04	74	PB_11	28	$\overline{\text{SYS\_FAULT}}$	65	VDD_EXT	82
PA_05	73	PB_12	27	$\overline{\text{SYS\_HWRST}}$	68	VDD_INT	14
PA_06	71	PB_13	25	$\overline{\text{SYS\_NMI}}$	77	VDD_INT	30
PA_07	70	PB_14	24	$\overline{\text{SYS\_RESOUT}}$	15	VDD_INT	51
PA_08	69	PB_15	23	SYS_XTAL	56	VDD_INT	61
PA_09	64	PC_00	16	TWI0_SCL	19	VDD_INT	81
PA_10	63	PC_01	13	TWI0_SDA	18	VDD_OTP	49
PA_11	60	PC_02	10	USB0_CLKIN	32	VDD_RTC	22
PA_12	59	PC_03	9	USB0_DM	37	VDD_USB	36
PA_13	58	PC_04	8	USB0_DP	35		
PA_14	54	PC_05	7	USB0_ID	33		
PA_15	53	PC_06	6	USB0_VBC	38		