Blackfin® Embedded Processor

Preliminary Technical Data

ADSP-BF522/523/524/525/526/527

FEATURES

- Up to 600 MHz high-performance Blackfin processor
  - Two 16-bit MACs, two 40-bit ALUs, four 8-bit video ALUs, 40-bit shifter
  - RISC-like register and instruction model for ease of programming and compiler-friendly support
  - Advanced debug, trace, and performance monitoring
- Accepts a wide range of supply voltages for internal and I/O operations. See Operating Conditions for ADSP-BF523/525/527 on Page 29 and Operating Conditions for ADSP-BF522/524/526 on Page 27
- Programmable on-chip voltage regulator (ADSP-BF523/525/527 processors only)
- 289-ball (12 mm x 12 mm) and 208-ball (17 mm x 17 mm) CSP_BGA packages

MEMORY

- 132K bytes of on-chip memory:
  - (See Table 1 on Page 3 for L1 and L3 memory size details)
- External memory controller with glueless support for SDRAM and asynchronous 8-bit and 16-bit memories
- Flexible booting options from external flash, SPI, and TWI memory or from host devices including SPI, TWI, and UART
- Code Security with Lockbox™ Secure Technology
  - One-Time-Programmable (OTP) Memory
- Memory management unit providing memory protection

PERIPHERALS

- USB 2.0 high speed on-the-go (OTG) with Integrated PHY
- IEEE 802.3-compliant 10/100 Ethernet MAC
- Parallel peripheral interface (PPI), supporting ITU-R 656 video data formats
- Host DMA port (HOSTDP)
- Two dual-channel, full-duplex synchronous serial ports (SPORTs), supporting eight stereo I^2S channels
- 12 peripheral DMAs, 2 mastered by the Ethernet MAC
- Two memory-to-memory DMAs with external request lines
- Event handler with 54 interrupt inputs
- Serial peripheral interface (SPI) compatible port
- Two UARTs with IrDA® support
- Two-wire interface (TWI) controller
- Eight 32-bit timers/counters with PWM support
- 32-bit up/down counter with rotary support
- Real-time clock (RTC) and watchdog timer
- 32-bit core timer
- 48 general-purpose I/Os (GPIOs), with programmable hysteresis
- NAND flash controller (NFC)
- Debug/JTAG interface
- On-chip PLL capable of 0.5× to 64× frequency multiplication

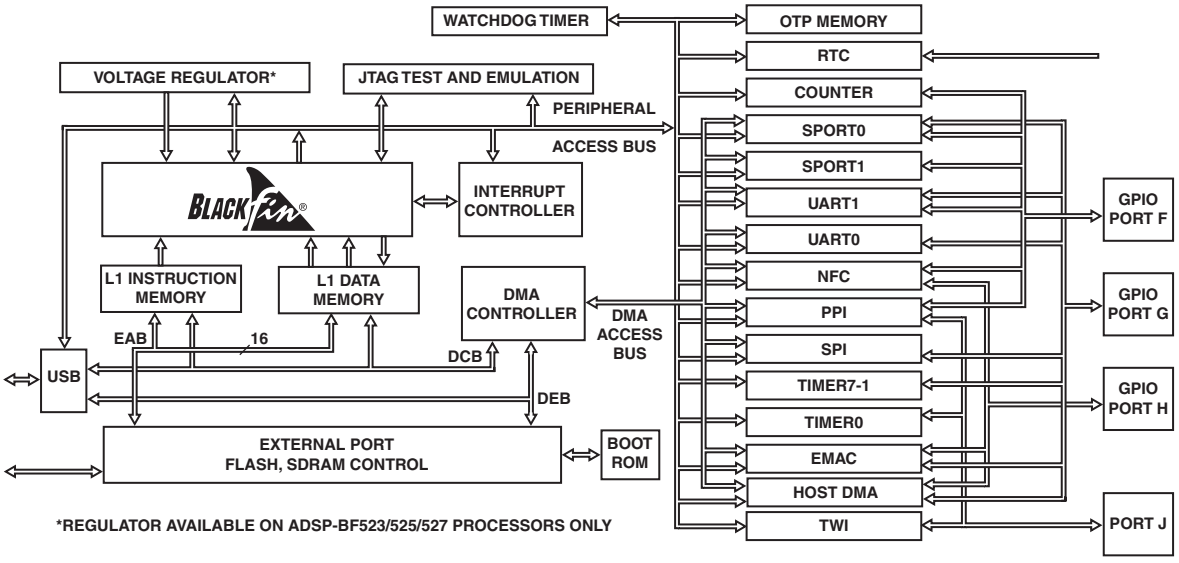


Figure 1. Processor Block Diagram

Rev. PrE

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106 U.S.A.
Tel: 781.329.4700 **www.analog.com**
Fax: 781.461.3113

TABLE OF CONTENTS

General Description 3
Portable Low-Power Architecture 3
System Integration 3
Processor Peripherals 3
Blackfin Processor Core 4
Memory Architecture 5
DMA Controllers 9
Host DMA Port 9
Real-Time Clock 10
Watchdog Timer 10
Timers 10
Up/Down Counter and Thumbwheel Interface 10
Serial Ports 11
Serial Peripheral Interface (SPI) Port 11
UART Ports 11
USB On-The-Go Dual-Role Device Controller 12
TWI Controller Interface 12
10/100 Ethernet MAC 12
Ports 12
Parallel Peripheral Interface (PPI) 13
Code Security with Lockbox Secure Technology 14
Dynamic Power Management 14
ADSP-BF523/525/527 Voltage Regulation 15
ADSP-BF522/524/526 Voltage Regulation 16
Clock Signals 16
Booting Modes 18
Instruction Set Description 20
Development Tools 21
Designing an Emulator-Compatible Processor Board (Target) 21
Related Documents 21
Lockbox Secure Technology Disclaimer 21
Signal Descriptions 22
Specifications 26
Operating Conditions for ADSP-BF522/524/526 26
Operating Conditions for ADSP-BF523/525/527 28
Electrical Characteristics 30
Absolute Maximum Ratings 31
ESD Sensitivity 31
Package Information 31
Timing Specifications 32
Output Drive Currents 57
Power Dissipation 59
Test Conditions 59
Environmental Conditions 62
289-Ball CSP_BGA Ball assignment 63
208-Ball CSP_BGA Ball assignment 66
Outline Dimensions 69
Surface Mount Design 70
Ordering Guide 71

REVISION HISTORY

06/08—Revision PrE:

Numerous small clarifications and corrections throughout document.

Changes to voltage regulator in Block Diagram ..... Page 1

Changes to processor comparison data ..... Table 1 on Page 3

Changes to hibernate state description ..... Page 15

Changes to voltage regulator description ..... Page 16

Changes to booting modes description ..... Page 18

Changes to signal descriptions ..... Table 10 on Page 23

Added Lockbox Secure Technology Disclaimer ..... Page 21

Changes to processor specifications (starting on Page 27). Major changes include:

- Added NFC timing ..... Page 36
- Changes to SPI timing ..... Page 47
- Added UART timing ..... Page 49
- Added Up/Down Counter timing ..... Page 51
- Changes to HOSTDP timing ..... Page 52 and Page 53

Changes to ball assignment tables ..... Page 64 and Page 67

GENERAL DESCRIPTION

The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors are members of the Blackfin family of products, incorporating the Analog Devices/Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC 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.

The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors are completely code compatible with other Blackfin processors. The ADSP-BF523/525/527 processors offer performance up to 600 MHz. The ADSP-BF522/524/526 processors offer performance up to 400 MHz and reduced static power consumption. Differences with respect to peripheral combinations are shown in Table 1.

Table 1. Processor Comparison

Feature		ADSP-BF522	ADSP-BF524	ADSP-BF526	ADSP-BF523	ADSP-BF525	ADSP-BF527
Host DMA		1	1	1	1	1	1
USB		–	1	1	–	1	1
Ethernet MAC		–	–	1	–	–	1
Internal Voltage Regulator		–	–	–	1	1	1
TWI		1	1	1	1	1	1
SPORTs		2	2	2	2	2	2
UARTs		2	2	2	2	2	2
SPI		1	1	1	1	1	1
GP Timers		8	8	8	8	8	8
Watchdog Timers		1	1	1	1	1	1
RTC		1	1	1	1	1	1
Parallel Peripheral Interface		1	1	1	1	1	1
GPIOs		48	48	48	48	48	48
Memory (bytes)	L1 Instruction SRAM	48K	48K	48K	48K	48K	48K
	L1 Instruction SRAM/Cache	16K	16K	16K	16K	16K	16K
	L1 Data SRAM	32K	32K	32K	32K	32K	32K
	L1 Data SRAM/Cache	32K	32K	32K	32K	32K	32K
	L1 Scratchpad	4K	4K	4K	4K	4K	4K
	L3 Boot ROM	32K	32K	32K	32K	32K	32K
Maximum Speed Grade[1]		400 MHz			600 MHz		
Maximum System Clock Speed		80 MHz			133 MHz		
Package Options		289-Ball CSP_BGA					
		208-Ball CSP_BGA					

[1] Maximum speed grade is not available with every possible SCLK selection.

By integrating a rich set of industry-leading system peripherals and memory, Blackfin processors are the platform of choice for next-generation applications that require RISC-like programmability, multimedia support, and leading-edge signal processing in one integrated package.

PORTABLE LOW-POWER ARCHITECTURE

Blackfin processors provide world-class power management and performance. They are produced with a low power and low voltage design methodology and feature on-chip dynamic power management, which is the ability to vary both the voltage and frequency of operation to significantly lower overall power consumption. This capability can result in a substantial reduction in power consumption, compared with just varying the frequency of operation. This allows longer battery life for portable appliances.

SYSTEM INTEGRATION

The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors are highly integrated system-on-a-chip solutions for the next generation of embedded network connected applications. By combining industry-standard interfaces with a high performance signal processing core, cost-effective applications can be developed quickly, without the need for costly external components. The system peripherals include an IEEE-compliant 802.3 10/100 Ethernet MAC, a USB 2.0 high speed OTG controller, a TWI controller, a NAND flash controller, two UART ports, an SPI port, two serial ports (SPORTs), eight general purpose 32-bit timers with PWM capability, a core timer, a real-time clock, a watchdog timer, a Host DMA (HOSTDP) interface, and a parallel peripheral interface (PPI).

PROCESSOR PERIPHERALS

The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors contain a rich set of peripherals connected to the core via several high bandwidth buses, providing flexibility in system configuration as well as excellent overall system performance (see the block diagram on Page 1). These Blackfin processors contain dedicated network communication modules and high speed serial and parallel ports, an interrupt controller for flexible management of interrupts from the on-chip peripherals or external sources, and power management control functions to tailor the performance and power characteristics of the processor and system to many application scenarios.

All of the peripherals, except for the general-purpose I/O, TWI, real-time clock, and timers, are supported by a flexible DMA structure. There are also separate memory DMA channels dedicated to data transfers between the processor's various memory spaces, including external SDRAM and asynchronous memory. Multiple on-chip buses running at up to 133 MHz provide enough bandwidth to keep the processor core running along with activity on all of the on-chip and external peripherals.

The ADSP-BF523/525/527 processors include an on-chip voltage regulator in support of the processor's dynamic power management capability. The voltage regulator provides a range of core voltage levels when supplied from V_{DDEXT}. The voltage regulator can be bypassed at the user's discretion.

BLACKFIN PROCESSOR CORE

As shown in Figure 2, the Blackfin processor core contains two 16-bit multipliers, two 40-bit accumulators, two 40-bit ALUs, four video ALUs, and a 40-bit shifter. The computation units process 8-, 16-, or 32-bit data from the register file.

The compute register file contains eight 32-bit registers. When performing compute operations on 16-bit operand data, the register file operates as 16 independent 16-bit registers. All operands for compute operations come from the multiported register file and instruction constant fields.

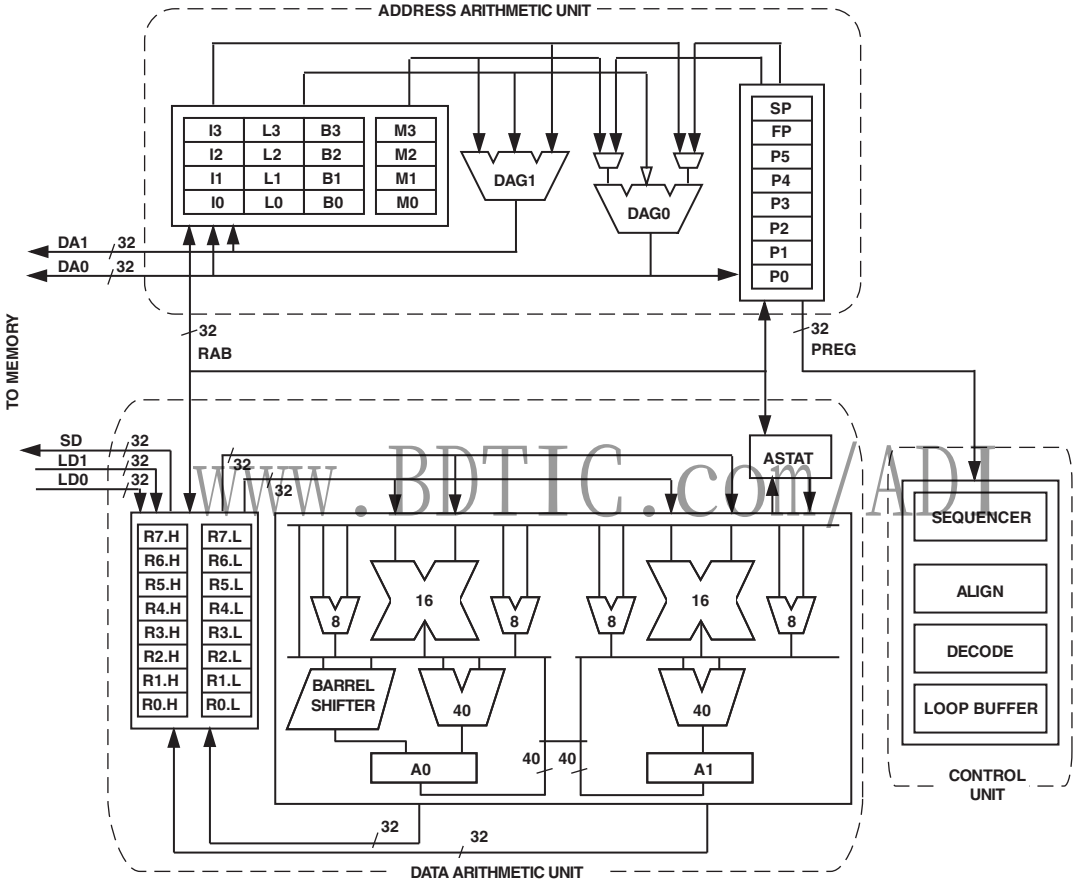


Figure 2. Blackfin Processor Core

Each MAC can perform a 16-bit by 16-bit multiply in each cycle, accumulating the results into the 40-bit accumulators. Signed and unsigned formats, rounding, and saturation are supported.

The ALUs perform a traditional set of arithmetic and logical operations on 16-bit or 32-bit data. In addition, many special instructions are included to accelerate various signal processing tasks. These include bit operations such as field extract and population count, modulo 2^{32} multiply, divide primitives, saturation and rounding, and sign/exponent detection. The set of video instructions include byte alignment and packing operations, 16-bit and 8-bit adds with clipping, 8-bit average operations, and 8-bit subtract/absolute value/accumulate (SAA) operations. Also provided are the compare/select and vector search instructions.

For certain instructions, two 16-bit ALU operations can be performed simultaneously on register pairs (a 16-bit high half and 16-bit low half of a compute register). If the second ALU is used, quad 16-bit operations are possible.

The 40-bit shifter can perform shifts and rotates and is used to support normalization, field extract, and field deposit instructions.

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 static branch prediction), and subroutine calls. Hardware is provided to support 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).

Blackfin processors support 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 two data memories hold 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.

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. Blackfin processors support 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 assembly language uses an algebraic syntax for ease of coding and readability. The architecture has been optimized for use in conjunction with the C/C++ compiler, resulting in fast and efficient software implementations.

MEMORY ARCHITECTURE

The Blackfin processor views memory as a single unified 4G byte address space, using 32-bit addresses. All resources, including internal memory, external memory, and I/O control registers, occupy separate sections of this common address space. The memory portions of this address space are arranged in a hierarchical structure to provide a good cost/performance balance of some very fast, low-latency on-chip memory as cache or SRAM, and larger, lower-cost and performance off-chip memory systems. See Figure 3.

The on-chip L1 memory system is the highest-performance memory available to the Blackfin processor. The off-chip memory system, accessed through the external bus interface unit (EBIU), provides expansion with SDRAM, flash memory, and SRAM, optionally accessing up to 516M bytes of physical memory.

The memory DMA controller provides high-bandwidth data-movement capability. It can perform block transfers of code or data between the internal memory and the external memory spaces.

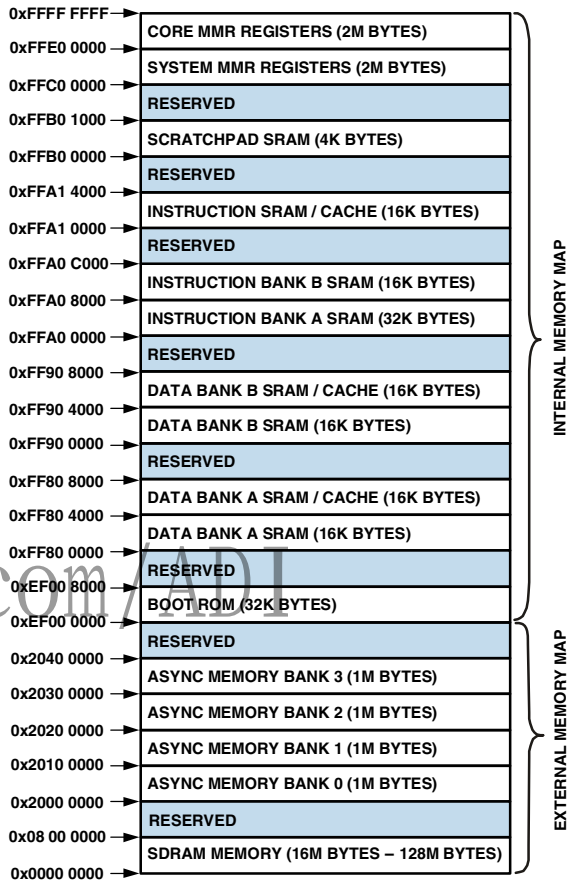


Figure 3. Internal/External Memory Map

Internal (On-Chip) Memory

The processor has three blocks of on-chip memory providing high-bandwidth access to the core.

The first block is the L1 instruction memory, consisting of 64K bytes SRAM, of which 16K bytes can be configured as a four-way set-associative cache. This memory is accessed at full processor speed.

The second on-chip memory block is the L1 data memory, consisting of up to two banks of up to 32K bytes each. Each memory bank is configurable, offering both cache and SRAM functionality. This memory block is accessed at full processor speed.

The third memory block is a 4K byte scratchpad SRAM which runs at the same speed as the L1 memories, but is only accessible as data SRAM and cannot be configured as cache memory.

External (Off-Chip) Memory

External memory is accessed via the EBIU. This 16-bit interface provides a glueless connection to a bank of synchronous DRAM (SDRAM) as well as up to four banks of asynchronous memory devices including flash, EPROM, ROM, SRAM, and memory mapped I/O devices.

The SDRAM controller can be programmed to interface to up to 128M bytes of SDRAM. A separate row can be open for each SDRAM internal bank and the SDRAM controller supports up to 4 internal SDRAM banks, improving overall performance.

The asynchronous memory controller can be programmed to control up to four banks of devices with very flexible timing parameters for a wide variety of devices. Each bank occupies a 1M byte segment regardless of the size of the devices used, so that these banks are only contiguous if each is fully populated with 1M byte of memory.

NAND Flash Controller (NFC)

The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors provide a NAND flash controller (NFC). NAND flash devices provide high-density, low-cost memory. However, NAND flash devices also have long random access times, invalid blocks, and lower reliability over device lifetimes. Because of this, NAND flash is often used for read-only code storage. In this case, all DSP code can be stored in NAND flash and then transferred to a faster memory (such as SDRAM or SRAM) before execution. Another common use of NAND flash is for storage of multimedia files or other large data segments. In this case, a software file system may be used to manage reading and writing of the NAND flash device. The file system selects memory segments for storage with the goal of avoiding bad blocks and equally distributing memory accesses across all address locations. Hardware features of the NFC include:

- Support for page program, page read, and block erase of NAND flash devices, with accesses aligned to page boundaries.
- Error checking and correction (ECC) hardware that facilitates error detection and correction.
- A single 8-bit external bus interface for commands, addresses and data.
- Support for SLC (single level cell) NAND flash devices unlimited in size, with page sizes of 256 and 512 bytes. Larger page sizes can be supported in software.
- Capability of releasing external bus interface pins during long accesses.
- Support for internal bus requests of 16-bits
- DMA engine to transfer data between internal memory and NAND flash device.

One-Time Programmable Memory

The processor has 64K bits of one-time programmable non-volatile memory that can be programmed by the developer only one time. It includes the array and logic to support read access and programming. Additionally, its pages can be write protected.

OTP enables developers to store both public and private data on-chip. In addition to storing public and private key data for applications requiring security, it also allows developers to store completely user-definable data such as customer ID, product ID, MAC address, etc. Hence generic parts can be shipped which are then programmed and protected by the developer within this non-volatile memory.

I/O Memory Space

The processor does not define a separate I/O space. All resources are mapped through the flat 32-bit address space. On-chip I/O devices have their control registers mapped into memory-mapped registers (MMRs) at addresses near the top of the 4G byte address space. These are separated into two smaller blocks, one which contains the control MMRs for all core functions, and the other which contains the registers needed for setup and control of the on-chip peripherals outside of the core. The MMRs are accessible only in supervisor mode and appear as reserved space to on-chip peripherals.

Booting

The processor contains a small on-chip boot kernel, which configures the appropriate peripheral for booting. If the processor is configured to boot from boot ROM memory space, the processor starts executing from the on-chip boot ROM. For more information, see Booting Modes on Page 18.

Event Handling

The event controller on the processor handles all asynchronous and synchronous events to the processor. The processor provides event handling that supports both nesting and prioritization. Nesting allows multiple event service routines to be active simultaneously. Prioritization ensures that servicing of a higher-priority event takes precedence over servicing of a lower-priority event. The controller provides support for five different types of events:

- Emulation – An emulation event causes the processor to enter emulation mode, allowing command and control of the processor via the JTAG interface.
- $\overline{\text{RESET}}$ – This event resets the processor.
- Nonmaskable Interrupt (NMI) – The NMI event can be generated by the software watchdog timer or by the $\overline{\text{NMI}}$ input signal to the processor. The NMI event is frequently used as a power-down indicator to initiate an orderly shutdown of the system.
- Exceptions – Events that occur synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions such as data alignment violations and undefined instructions cause exceptions.
- Interrupts – Events that occur asynchronously to program flow. They are caused by input signals, timers, and other peripherals, as well as by an explicit software instruction.

Each event type has an associated register to hold the return address and an associated return-from-event instruction. When an event is triggered, the state of the processor is saved on the supervisor stack.

The processor event controller consists of two stages, the core event controller (CEC) and the system interrupt controller (SIC). The core event controller works with the system interrupt controller to prioritize and control all system events. Conceptually, interrupts from the peripherals enter into the SIC and are then routed directly into the general-purpose interrupts of the CEC.

Core Event Controller (CEC)

The CEC supports nine general-purpose interrupts (IVG15–7), in addition to the dedicated interrupt and exception events. Of these general-purpose interrupts, the two lowest-priority interrupts (IVG15–14) are recommended to be reserved for software interrupt handlers, leaving seven prioritized interrupt inputs to support the peripherals of the processor. Table 2 describes the inputs to the CEC, identifies their names in the event vector table (EVT), and lists their priorities.

Table 2. Core Event Controller (CEC)

Priority (0 is Highest)	Event Class	EVT Entry
0	Emulation/Test Control	EMU
1	$\overline{\text{RESET}}$	RST
2	Nonmaskable Interrupt	$\overline{\text{NMI}}$
3	Exception	EVX
4	Reserved	—
5	Hardware Error	IVHW
6	Core Timer	IVTMR
7	General-Purpose Interrupt 7	IVG7
8	General-Purpose Interrupt 8	IVG8
9	General-Purpose Interrupt 9	IVG9
10	General-Purpose Interrupt 10	IVG10
11	General-Purpose Interrupt 11	IVG11
12	General-Purpose Interrupt 12	IVG12
13	General-Purpose Interrupt 13	IVG13
14	General-Purpose Interrupt 14	IVG14
15	General-Purpose Interrupt 15	IVG15

System Interrupt Controller (SIC)

The system interrupt controller provides the mapping and routing of events from the many peripheral interrupt sources to the prioritized general-purpose interrupt inputs of the CEC. Although the processor provides a default mapping, the user can alter the mappings and priorities of interrupt events by writing the appropriate values into the interrupt assignment registers (SIC_IARx). Table 3 describes the inputs into the SIC and the default mappings into the CEC.

Table 3. System Interrupt Controller (SIC)

Peripheral Interrupt Event	General Purpose Interrupt (at $\overline{\text{RESET}}$)	Peripheral Interrupt ID	Default Core Interrupt ID	SIC Registers	
PLL Wakeup Interrupt	IVG7	0	0	IAR0	IMASK0, ISR0, IWR0
DMA Error 0 (generic)	IVG7	1	0	IAR0	IMASK0, ISR0, IWR0
DMAR0 Block Interrupt	IVG7	2	0	IAR0	IMASK0, ISR0, IWR0
DMAR1 Block Interrupt	IVG7	3	0	IAR0	IMASK0, ISR0, IWR0
DMAR0 Overflow Error	IVG7	4	0	IAR0	IMASK0, ISR0, IWR0
DMAR1 Overflow Error	IVG7	5	0	IAR0	IMASK0, ISR0, IWR0
PPI Error	IVG7	6	0	IAR0	IMASK0, ISR0, IWR0
MAC Status	IVG7	7	0	IAR0	IMASK0, ISR0, IWR0
SPORT0 Status	IVG7	8	0	IAR1	IMASK0, ISR0, IWR0
SPORT1 Status	IVG7	9	0	IAR1	IMASK0, ISR0, IWR0
Reserved	IVG7	10	0	IAR1	IMASK0, ISR0, IWR0
Reserved	IVG7	11	0	IAR1	IMASK0, ISR0, IWR0
UART0 Status	IVG7	12	0	IAR1	IMASK0, ISR0, IWR0
UART1 Status	IVG7	13	0	IAR1	IMASK0, ISR0, IWR0
RTC	IVG8	14	1	IAR1	IMASK0, ISR0, IWR0
DMA Channel 0 (PPI/NFC)	IVG8	15	1	IAR1	IMASK0, ISR0, IWR0
DMA 3 Channel (SPORT0 RX)	IVG9	16	2	IAR2	IMASK0, ISR0, IWR0
DMA 4 Channel (SPORT0 TX)	IVG9	17	2	IAR2	IMASK0, ISR0, IWR0
DMA 5 Channel (SPORT1 RX)	IVG9	18	2	IAR2	IMASK0, ISR0, IWR0
DMA 6 Channel (SPORT1 TX)	IVG9	19	2	IAR2	IMASK0, ISR0, IWR0
TWI	IVG10	20	3	IAR2	IMASK0, ISR0, IWR0

Table 3. System Interrupt Controller (SIC) (Continued)

Peripheral Interrupt Event	General Purpose Interrupt (at RESET)	Peripheral Interrupt ID	Default Core Interrupt ID	SIC Registers	
DMA 7 Channel (SPI)	IVG10	21	3	IAR2	IMASK0, ISR0, IWR0
DMA8 Channel (UART0 RX)	IVG10	22	3	IAR2	IMASK0, ISR0, IWR0
DMA9 Channel (UART0 TX)	IVG10	23	3	IAR2	IMASK0, ISR0, IWR0
DMA10 Channel (UART1 RX)	IVG10	24	3	IAR3	IMASK0, ISR0, IWR0
DMA11 Channel (UART1 TX)	IVG10	25	3	IAR3	IMASK0, ISR0, IWR0
OTP Memory Interrupt	IVG11	26	4	IAR3	IMASK0, ISR0, IWR0
GP Counter	IVG11	27	4	IAR3	IMASK0, ISR0, IWR0
DMA1 Channel (MAC RX/HOSTDP)	IVG11	28	4	IAR3	IMASK0, ISR0, IWR0
Port H Interrupt A	IVG11	29	4	IAR3	IMASK0, ISR0, IWR0
DMA2 Channel (MAC TX/NFC)	IVG11	30	4	IAR3	IMASK0, ISR0, IWR0
Port H Interrupt B	IVG11	31	4	IAR3	IMASK0, ISR0, IWR0
Timer 0	IVG12	32	5	IAR4	IMASK1, ISR1, IWR1
Timer 1	IVG12	33	5	IAR4	IMASK1, ISR1, IWR1
Timer 2	IVG12	34	5	IAR4	IMASK1, ISR1, IWR1
Timer 3	IVG12	35	5	IAR4	IMASK1, ISR1, IWR1
Timer 4	IVG12	36	5	IAR4	IMASK1, ISR1, IWR1
Timer 5	IVG12	37	5	IAR4	IMASK1, ISR1, IWR1
Timer 6	IVG12	38	5	IAR4	IMASK1, ISR1, IWR1
Timer 7	IVG12	39	5	IAR4	IMASK1, ISR1, IWR1
Port G Interrupt A	IVG12	40	5	IAR5	IMASK1, ISR1, IWR1
Port G Interrupt B	IVG12	41	5	IAR5	IMASK1, ISR1, IWR1
MDMA Stream 0	IVG13	42	6	IAR5	IMASK1, ISR1, IWR1
MDMA Stream 1	IVG13	43	6	IAR5	IMASK1, ISR1, IWR1
Software Watchdog Timer	IVG13	44	6	IAR5	IMASK1, ISR1, IWR1
Port F Interrupt A	IVG13	45	6	IAR5	IMASK1, ISR1, IWR1
Port F Interrupt B	IVG13	46	6	IAR5	IMASK1, ISR1, IWR1
SPI Status	IVG7	47	0	IAR5	IMASK1, ISR1, IWR1
NFC Status	IVG7	48	0	IAR6	IMASK1, ISR1, IWR1
HOSTDP Status	IVG7	49	0	IAR6	IMASK1, ISR1, IWR1
Host Read Done	IVG7	50	0	IAR6	IMASK1, ISR1, IWR1
USB_EINT Interrupt	IVG10	51	3	IAR6	IMASK1, ISR1, IWR1
USB_INT0 Interrupt	IVG10	52	3	IAR6	IMASK1, ISR1, IWR1
USB_INT1 Interrupt	IVG10	53	3	IAR6	IMASK1, ISR1, IWR1
USB_INT2 Interrupt	IVG10	54	3	IAR6	IMASK1, ISR1, IWR1
USB_DMAINT Interrupt	IVG10	55	3	IAR6	IMASK1, ISR1, IWR1

Event Control

The processor provides a very flexible mechanism to control the processing of events. In the CEC, three registers are used to coordinate and control events. Each register is 16 bits wide.

- CEC interrupt latch register (ILAT) – Indicates when events have been latched. The appropriate bit is set when the processor has latched the event and cleared when the event has been accepted into the system. This register is updated automatically by the controller, but it may be written only when its corresponding IMASK bit is cleared.
- CEC interrupt mask register (IMASK) – Controls the masking and unmasking of individual events. When a bit is set in the IMASK register, that event is unmasked and is processed by the CEC when asserted. A cleared bit in the IMASK register masks the event, preventing the processor from servicing the event even though the event may be latched in the ILAT register. This register may be read or

written while in supervisor mode. (Note that general-purpose interrupts can be globally enabled and disabled with the STI and CLI instructions, respectively.)

- CEC interrupt pending register (IPEND) – The IPEND register keeps track of all nested events. A set bit in the IPEND register indicates the event is currently active or nested at some level. This register is updated automatically by the controller but may be read while in supervisor mode.

The SIC allows further control of event processing by providing three pairs of 32-bit interrupt control and status registers. Each register contains a bit corresponding to each of the peripheral interrupt events shown in Table 3 on Page 7.

- SIC interrupt mask registers (SIC_IMASKx) – Control the masking and unmasking of each peripheral interrupt event. When a bit is set in these registers, that peripheral event is unmasked and is processed by the system when asserted. A cleared bit in the register masks the peripheral event, preventing the processor from servicing the event.
- SIC interrupt status registers (SIC_ISRx) – As multiple peripherals can be mapped to a single event, these registers allow the software to determine which peripheral event source triggered the interrupt. A set bit indicates the peripheral is asserting the interrupt, and a cleared bit indicates the peripheral is not asserting the event.
- SIC interrupt wakeup enable registers (SIC_IWRx) – By enabling the corresponding bit in these registers, a peripheral can be configured to wake up the processor, should the core be idled or in sleep mode when the event is generated. For more information see Dynamic Power Management on Page 14.

Because multiple interrupt sources can map to a single general-purpose interrupt, multiple pulse assertions can occur simultaneously, before or during interrupt processing for an interrupt event already detected on this interrupt input. The IPEND register contents are monitored by the SIC as the interrupt acknowledgement.

The appropriate ILAT register bit is set when an interrupt rising edge is detected (detection requires two core clock cycles). The bit is cleared when the respective IPEND register bit is set. The IPEND bit indicates that the event has entered into the processor pipeline. At this point the CEC recognizes and queues the next rising edge event on the corresponding event input. The minimum latency from the rising edge transition of the general-purpose interrupt to the IPEND output asserted is three core clock cycles; however, the latency can be much higher, depending on the activity within and the state of the processor.

DMA CONTROLLERS

The processor has multiple, independent DMA channels that support automated data transfers with minimal overhead for the processor core. DMA transfers can occur between the processor's internal memories and any of its DMA-capable peripherals. Additionally, DMA transfers can be accomplished between any of the DMA-capable peripherals and external devices connected to the external memory interfaces, including the SDRAM controller and the asynchronous memory controller. DMA-capable peripherals include the Ethernet MAC, NFC, HOSTDP, USB, SPORTs, SPI port, UARTs, and PPI. Each individual DMA-capable peripheral has at least one dedicated DMA channel.

The processor DMA controller supports both one-dimensional (1-D) and two-dimensional (2-D) DMA transfers. DMA transfer initialization can be implemented from registers or from sets of parameters called descriptor blocks.

The 2-D DMA capability supports arbitrary row and column sizes up to 64K elements by 64K elements, and arbitrary row and column step sizes up to ±32K elements. Furthermore, the column step size can be less than the row step size, allowing implementation of interleaved data streams. This feature is especially useful in video applications where data can be de-interleaved on the fly.

Examples of DMA types supported by the processor DMA controller include:

- A single, linear buffer that stops upon completion
- A circular, auto-refreshing buffer that interrupts on each full or fractionally full buffer
- 1-D or 2-D DMA using a linked list of descriptors
- 2-D DMA using an array of descriptors, specifying only the base DMA address within a common page

In addition to the dedicated peripheral DMA channels, there are two memory DMA channels provided for transfers between the various memories of the processor system. This enables transfers of blocks of data between any of the memories—including external SDRAM, ROM, SRAM, and flash memory—with minimal processor intervention. Memory DMA transfers can be controlled by a very flexible descriptor-based methodology or by a standard register-based autobuffer mechanism.

The processor also has an external DMA controller capability via dual external DMA request pins when used in conjunction with the external bus interface unit (EBIU). This functionality can be used when a high speed interface is required for external FIFOs and high bandwidth communications peripherals such as USB 2.0. It allows control of the number of data transfers for memory DMA. The number of transfers per edge is programmable. This feature can be programmed to allow memory DMA to have an increased priority on the external bus relative to the core.

HOST DMA PORT

The host port interface allows an external host to be a DMA master to transfer data in and out of the device. The host device masters the transactions and the Blackfin is the DMA slave.

The host port is enabled through the PAB interface. Once enabled, the DMA is controlled by the external host, which can then program the DMA to send/receive data to any valid internal or external memory location.

The host port interface controller has the following features.

- Allows external master to configure DMA read/write data transfers and read port status.
- Uses asynchronous memory protocol for external interface.

- 8-/16-bit external data interface to host device.
- Half duplex operation
- Little-/big-endian data transfer.
- Acknowledge mode allows flow control on host transactions.
- Interrupt mode guarantees a burst of FIFO depth host transactions.

REAL-TIME CLOCK

The real-time clock (RTC) provides a robust set of digital watch features, including current time, stopwatch, and alarm. The RTC is clocked by a 32.768 kHz crystal external to the Blackfin processor. The RTC peripheral has dedicated power supply pins so that it can remain powered up and clocked even when the rest of the processor is in a low-power state. The RTC provides several programmable interrupt options, including interrupt per second, minute, hour, or day clock ticks, interrupt on programmable stopwatch countdown, or interrupt at a programmed alarm time.

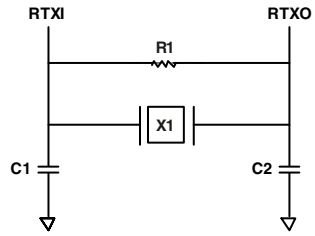
The 32.768 kHz input clock frequency is divided down to a 1 Hz signal by a prescaler. The counter function of the timer consists of four counters: a 60-second counter, a 60-minute counter, a 24-hour counter, and an 32,768-day counter.

When enabled, the alarm function generates an interrupt when the output of the timer matches the programmed value in the alarm control register. There are two alarms: The first alarm is for a time of day. The second alarm is for a day and time of that day.

The stopwatch function counts down from a programmed value, with one-second resolution. When the stopwatch is enabled and the counter underflows, an interrupt is generated.

Like the other peripherals, the RTC can wake up the processor from sleep mode upon generation of any RTC wakeup event. Additionally, an RTC wakeup event can wake up the processor from deep sleep mode or cause a transition from the hibernate state.

Connect RTC pins RTXI and RTXO with external components as shown in Figure 4.



SUGGESTED COMPONENTS:
X1 = ECLIPTEK EC38J (THROUGH-HOLE PACKAGE) OR
EPSON MC405 12 pF LOAD (SURFACE-MOUNT PACKAGE)
C1 = 22 pF
C2 = 22 pF
R1 = 10 MΩ
NOTE: C1 AND C2 ARE SPECIFIC TO CRYSTAL SPECIFIED FOR X1. CONTACT CRYSTAL MANUFACTURER FOR DETAILS. C1 AND C2 SPECIFICATIONS ASSUME BOARD TRACE CAPACITANCE OF 3 pF.

Figure 4. External Components for RTC

WATCHDOG TIMER

The processor includes a 32-bit timer that can 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 to zero from the programmed value. This protects the system from remaining in an unknown state where software, which would normally reset the timer, has stopped running due to an external noise condition or software error.

If configured to generate a hardware reset, the watchdog timer resets both the core and the processor peripherals. After a reset, software can determine if the watchdog was the source of the hardware reset by interrogating a status bit in the watchdog timer control register.

The timer is clocked by the system clock (SCLK), at a maximum frequency of f_{SCLK}.

TIMERS

There are nine general-purpose programmable timer units in the processors. Eight timers have 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 to the several other associated PF pins, an external clock input to the PPI_CLK input pin, or to the internal SCLK.

The timer units can be used in conjunction with the two UARTs 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 timers can generate interrupts to the processor core providing periodic events for synchronization, either to the system clock or to a count of external signals.

In addition to the eight general-purpose programmable timers, a ninth timer is also provided. This extra timer is clocked by the internal processor clock and is typically used as a system tick clock for generation of operating system periodic interrupts.

UP/DOWN COUNTER AND THUMBWHEEL INTERFACE

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

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

An internal signal forwarded to the timer unit enables one timer to measure the intervals between count events. Boundary registers enable auto-zero operation or simple system warning by interrupts when programmable count values are exceeded.

SERIAL PORTS

The processors incorporate two dual-channel synchronous serial ports (SPORT0 and SPORT1) for serial and multiprocessor communications. The SPORTs support the following features:

- I^2S capable operation.
- Bidirectional operation – Each SPORT has two sets of independent transmit and receive pins, enabling eight channels of I^2S stereo audio.
- Buffered (8-deep) transmit and receive ports – Each port has a data register for transferring data words to and from other processor components and shift registers for shifting data in and out of the data registers.
- Clocking – Each transmit and receive port can either use an external serial clock or generate its own, in frequencies ranging from (f_{SCLK}/131,070) Hz to (f_{SCLK}/2) Hz.
- Word length – Each SPORT supports serial data words from 3 to 32 bits in length, transferred most-significant-bit first or least-significant-bit first.
- Framing – Each transmit and receive port can run with or without frame sync signals for each data word. Frame sync signals can be generated internally or externally, active high or low, and with either of two pulse widths and early or late frame sync.
- Companding in hardware – Each SPORT can perform A-law or μ-law companding according to ITU recommendation G.711. Companding can be selected on the transmit and/or receive channel of the SPORT without additional latencies.
- DMA operations with single-cycle overhead – Each SPORT can automatically receive and transmit multiple buffers of memory data. The processor can link or chain sequences of DMA transfers between a SPORT and memory.
- Interrupts – Each transmit and receive port generates an interrupt upon completing the transfer of a data word or after transferring an entire data buffer, or buffers, through DMA.
- Multichannel capability – Each SPORT supports 128 channels out of a 1024-channel window and is compatible with the H.100, H.110, MVIP-90, and HMVIP standards.

SERIAL PERIPHERAL INTERFACE (SPI) PORT

The processors have an SPI-compatible port that enables the processor to communicate with multiple SPI-compatible devices.

The SPI interface uses three pins for transferring data: two data pins (Master Output-Slave Input, MOSI, and Master Input-Slave Output, MISO) and a clock pin (serial clock, SCK). An SPI chip select input pin ($\overline{\text{SPISS}}$) lets other SPI devices select the processor, and seven SPI chip select output pins ($\overline{\text{SPISEL7–1}}$) let the processor select other SPI devices. The SPI select pins are reconfigured general-purpose I/O pins. Using these pins, the SPI port provides a full-duplex, synchronous serial interface, which supports both master/slave modes and multimaster environments.

The SPI port's baud rate and clock phase/polarities are programmable, and it has an integrated DMA channel, configurable to support transmit or receive data streams. The SPI's DMA channel can only service unidirectional accesses at any given time.

The SPI port's clock rate is calculated as:

$$SPI\ Clock\ Rate = \frac{f_{SCLK}}{2 \times SPI\_BAUD}$$

Where the 16-bit SPI_BAUD register contains a value of 2 to 65,535.

During transfers, the SPI port simultaneously transmits and receives by serially shifting data in and out on its two serial data lines. The serial clock line synchronizes the shifting and sampling of data on the two serial data lines.

UART PORTS

The processors provide two full-duplex universal asynchronous receiver/transmitter (UART) ports, which are 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, one or two stop bits, and none, even, or odd parity. Each UART port supports two modes of operation:

- PIO (programmed I/O) – The processor sends or receives data by writing or reading I/O mapped UART registers. The data is double-buffered on both transmit and receive.
- DMA (direct memory access) – The DMA controller transfers both transmit and receive data. This reduces the number and frequency of interrupts required to transfer data to and from memory. The UART has two dedicated DMA channels, one for transmit and one for receive. These DMA channels have lower default priority than most DMA channels because of their relatively low service rates.

Each UART port's baud rate, serial data format, error code generation and status, and interrupts are programmable:

- Supporting bit rates ranging from (f_{SCLK}/1,048,576) to (f_{SCLK}/16) bits per second.
- Supporting data formats from seven to 12 bits per frame.
- Both transmit and receive operations can be configured to generate maskable interrupts to the processor.

The UART port's clock rate is calculated as:

$$UART\ Clock\ Rate = \frac{f_{SCLK}}{16 \times UART\_Divisor}$$

Where the 16-bit UART_Divisor comes from the UART_DLH (most significant 8 bits) and UART_DLL (least significant 8 bits) registers.

In conjunction with the general-purpose timer functions, autobaud detection is supported.

The capabilities of the UARTs are further extended with support for the infrared data association (IrDA®) serial infrared physical layer link specification (SIR) protocol.

USB ON-THE-GO DUAL-ROLE DEVICE CONTROLLER

The USB OTG controller provides a low-cost connectivity solution for consumer mobile devices such as cell phones, digital still cameras and MP3 players, allowing these devices to transfer data using a point-to-point USB connection without the need for a PC host. The USBDRC module can operate in a traditional USB peripheral-only mode as well as the host mode presented in the On-The-Go (OTG) supplement to the USB 2.0 Specification. In host mode, the USB module supports transfers at high-speed (480Mbps), full-speed (12Mbps), and low-speed (1.5Mbps) rates. Peripheral-only mode supports the high- and full-speed transfer rates.

TWI CONTROLLER INTERFACE

The processors include a two wire interface (TWI) module for providing a simple exchange method of control data between multiple devices. The TWI is compatible with the widely used I^2C® bus standard. The TWI module offers the capabilities of simultaneous master and slave operation, support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (SCL) and data (SDA) and supports the protocol at speeds up to 400k bits/sec. The TWI interface pins are compatible with 5 V logic levels.

Additionally, the TWI module is fully compatible with serial camera control bus (SCCB) functionality for easier control of various CMOS camera sensor devices.

10/100 ETHERNET MAC

The ADSP-BF526 and ADSP-BF527 processors offer the capability to directly connect to a network by way of an embedded Fast Ethernet Media Access Controller (MAC) that supports both 10-BaseT (10M bits/sec) and 100-BaseT (100M bits/sec) operation. The 10/100 Ethernet MAC peripheral on the processor is fully compliant to the IEEE 802.3-2002 standard and it provides programmable features designed to minimize supervision, bus use, or message processing by the rest of the processor system.

Some standard features are:

- Support of MII and RMII protocols for external PHYs.
- Full duplex and half duplex modes.
- Data framing and encapsulation: generation and detection of preamble, length padding, and FCS.
- Media access management (in half-duplex operation): collision and contention handling, including control of retransmission of collision frames and of back-off timing.
- Flow control (in full-duplex operation): generation and detection of PAUSE frames.
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers.
- SCLK operating range down to 25 MHz (active and sleep operating modes).
- Internal loopback from Tx to Rx.

Some advanced features are:

- Buffered crystal output to external PHY for support of a single crystal system.
- Automatic checksum computation of IP header and IP payload fields of Rx frames.
- Independent 32-bit descriptor-driven Rx and Tx DMA channels.
- Frame status delivery to memory via DMA, including frame completion semaphores, for efficient buffer queue management in software.
- Tx DMA support for separate descriptors for MAC header and payload to eliminate buffer copy operations.
- Convenient frame alignment modes support even 32-bit alignment of encapsulated Rx or Tx IP packet data in memory after the 14-byte MAC header.

- Programmable Ethernet event interrupt supports any combination of:
  - Any selected Rx or Tx frame status conditions.
  - PHY interrupt condition.
  - Wakeup frame detected.
  - Any selected MAC management counter(s) at half-full.
  - DMA descriptor error.
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value.
- Programmable Rx address filters, including a 64-bin address hash table for multicast and/or unicast frames, and programmable filter modes for broadcast, multicast, unicast, control, and damaged frames.
- Advanced power management supporting unattended transfer of Rx and Tx frames and status to/from external memory via DMA during low-power sleep mode.
- System wakeup from sleep operating mode upon magic packet or any of four user-definable wakeup frame filters.
- Support for 802.3Q tagged VLAN frames.
- Programmable MDC clock rate and preamble suppression.
- In RMII operation, seven unused pins may be configured as GPIO pins for other purposes.

PORTS

Because of the rich set of peripherals, the processor groups the many peripheral signals to four ports—Port F, Port G, Port H, and Port J. Most of the associated pins are shared by multiple signals. The ports function as multiplexer controls.

General-Purpose I/O (GPIO)

The processor has 48 bidirectional, general-purpose I/O (GPIO) pins allocated across three separate GPIO modules—PORTFIO, PORTGIO, and PORTHIO, associated with Port F, Port G, and Port H, respectively. Port J does not provide GPIO functionality. Each GPIO-capable pin shares functionality with other processor peripherals via a multiplexing scheme; however, the GPIO functionality is the default state of the device upon power-up. Neither GPIO output nor input drivers are active by default. Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers:

- GPIO direction control register – Specifies the direction of each individual GPIO pin as input or output.
- GPIO control and status registers – The processor employs a "write one to modify" mechanism that allows any combination of individual GPIO pins to be modified in a single instruction, without affecting the level of any other GPIO pins. Four control registers are provided. One register is written in order to set pin values, one register is written in order to clear pin values, one register is written in order to toggle pin values, and one register is written in order to specify a pin value. Reading the GPIO status register allows software to interrogate the sense of the pins.
- GPIO interrupt mask registers – The two GPIO interrupt mask registers allow each individual GPIO pin to function as an interrupt to the processor. Similar to the two GPIO control registers that are used to set and clear individual pin values, one GPIO interrupt mask register sets bits to enable interrupt function, and the other GPIO interrupt mask register clears bits to disable interrupt function. GPIO pins defined as inputs can be configured to generate hardware interrupts, while output pins can be triggered by software interrupts.
- GPIO interrupt sensitivity registers – The two GPIO interrupt sensitivity registers specify whether individual pins are level- or edge-sensitive and specify—if edge-sensitive—whether just the rising edge or both the rising and falling edges of the signal are significant. One register selects the type of sensitivity, and one register selects which edges are significant for edge-sensitivity.

PARALLEL PERIPHERAL INTERFACE (PPI)

The processor provides a parallel peripheral interface (PPI) that can connect directly to parallel A/D and D/A converters, video encoders and decoders, and other general-purpose peripherals. The PPI consists of a dedicated input clock pin, up to three frame synchronization pins, and up to 16 data pins. The input clock supports parallel data rates up to half the system clock rate and the synchronization signals can be configured as either inputs or outputs.

The PPI supports a variety of general-purpose and ITU-R 656 modes of operation. In general-purpose mode, the PPI provides half-duplex, bidirectional data transfer with up to 16 bits of data. Up to three frame synchronization signals are also provided. In ITU-R 656 mode, the PPI provides half-duplex bidirectional transfer of 8- or 10-bit video data. Additionally, on-chip decode of embedded start-of-line (SOL) and start-of-field (SOF) preamble packets is supported.

General-Purpose Mode Descriptions

The general-purpose modes of the PPI are intended to suit a wide variety of data capture and transmission applications. Three distinct submodes are supported:

1. Input mode – Frame syncs and data are inputs into the PPI.
2. Frame capture mode – Frame syncs are outputs from the PPI, but data are inputs.
3. Output mode – Frame syncs and data are outputs from the PPI.

Input Mode

Input mode is intended for ADC applications, as well as video communication with hardware signaling. In its simplest form, PPI_FS1 is an external frame sync input that controls when to read data. The PPI_DELAY MMR allows for a delay (in PPI_CLK cycles) between reception of this frame sync and the initiation of data reads. The number of input data samples is

user programmable and defined by the contents of the PPI_COUNT register. The PPI supports 8-bit and 10-bit through 16-bit data, programmable in the PPI_CONTROL register.

Frame Capture Mode

Frame capture mode allows the video source(s) to act as a slave (for frame capture for example). The ADSP-BF522/524/526 and ADSP-BF523/525/527 processors control when to read from the video source(s). PPI_FS1 is an HSYNC output and PPI_FS2 is a VSYNC output.

Output Mode

Output mode is used for transmitting video or other data with up to three output frame syncs. Typically, a single frame sync is appropriate for data converter applications, whereas two or three frame syncs could be used for sending video with hardware signaling.

ITU-R 656 Mode Descriptions

The ITU-R 656 modes of the PPI are intended to suit a wide variety of video capture, processing, and transmission applications. Three distinct submodes are supported:

1. Active video only mode
2. Vertical blanking only mode
3. Entire field mode

Active Video Mode

Active video only mode is used when only the active video portion of a field is of interest and not any of the blanking intervals. The PPI does not read in any data between the end of active video (EAV) and start of active video (SAV) preamble symbols, or any data present during the vertical blanking intervals. In this mode, the control byte sequences are not stored to memory; they are filtered by the PPI. After synchronizing to the start of Field 1, the PPI ignores incoming samples until it sees an SAV code. The user specifies the number of active video lines per frame (in PPI_COUNT register).

Vertical Blanking Interval Mode

In this mode, the PPI only transfers vertical blanking interval (VBI) data.

Entire Field Mode

In this mode, the entire incoming bit stream is read in through the PPI. This includes active video, control preamble sequences, and ancillary data that may be embedded in horizontal and vertical blanking intervals. Data transfer starts immediately after synchronization to Field 1. Data is transferred to or from the synchronous channels through eight DMA engines that work autonomously from the processor core.

CODE SECURITY WITH LOCKBOX SECURE TECHNOLOGY

A security system consisting of a blend of hardware and software provides customers with a flexible and rich set of code security features with Lockbox secure technology. Key features include:

- OTP memory
- Unique chip ID
- Code authentication
- Secure mode of operation

The security scheme is based upon the concept of authentication of digital signatures using standards-based algorithms and provides a secure processing environment in which to execute code and protect assets. See Lockbox Secure Technology Disclaimer on Page 21.

DYNAMIC POWER MANAGEMENT

The processor provides four operating modes, each with a different performance/power profile. In addition, dynamic power management provides the control functions to dynamically alter the processor core supply voltage, further reducing power dissipation. When configured for a 0 volt core supply voltage, the processor enters the hibernate state. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 4 for a summary of the power settings for each mode.

Full-On Operating Mode—Maximum Performance

In the full-on mode, the PLL is enabled and is not bypassed, providing capability for maximum operational frequency. This is the power-up default execution state in which maximum performance can be achieved. The processor core and all enabled peripherals run at full speed.

Active Operating Mode—Moderate Dynamic Power Savings

In the active mode, the PLL is enabled but bypassed. Because the PLL is bypassed, the processor's core clock (CCLK) and system clock (SCLK) run at the input clock (CLKIN) frequency. DMA access is available to appropriately configured L1 memories.

In the active mode, it is possible to disable the PLL through the PLL control register (PLL_CTL). If disabled, the PLL must be re-enabled before transitioning to the full-on or sleep modes.

Table 4. Power Settings

Mode/State	PLL	PLL Bypassed	Core Clock (CCLK)	System Clock (SCLK)	Core Power
Full On	Enabled	No	Enabled	Enabled	On
Active	Enabled/ Disabled	Yes	Enabled	Enabled	On

Table 4. Power Settings (Continued)

Mode/State	PLL	PLL Bypassed	Core Clock (CCLK)	System Clock (SCLK)	Core Power
Sleep	Enabled	—	Disabled	Enabled	On
Deep Sleep	Disabled	—	Disabled	Disabled	On
Hibernate	Disabled	—	Disabled	Disabled	Off

Sleep Operating Mode—High Dynamic Power Savings

The sleep mode reduces dynamic power dissipation by disabling the clock to the processor core (CCLK). The PLL and system clock (SCLK), however, continue to operate in this mode. Typically an external event or RTC activity wakes up the processor. When in the sleep mode, asserting a wakeup enabled in the SIC_IWRx registers causes the processor to sense the value of the BYPASS bit in the PLL control register (PLL_CTL). If BYPASS is disabled, the processor transitions to the full on mode. If BYPASS is enabled, the processor transitions to the active mode.

System DMA access to L1 memory is not supported in sleep mode.

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core (CCLK) and to all synchronous peripherals (SCLK). Asynchronous peripherals, such as the RTC, may still be running but cannot access internal resources or external memory. This powered-down mode can only be exited by assertion of the reset interrupt ($\overline{\text{RESET}}$) or by an asynchronous interrupt generated by the RTC. When in deep sleep mode, an RTC asynchronous interrupt causes the processor to transition to the Active mode. Assertion of $\overline{\text{RESET}}$ while in deep sleep mode causes the processor to transition to the full on mode.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor core (CCLK) and to all of the synchronous peripherals (SCLK). The internal voltage regulator (ADSP-BF523/525/527 only) for the processor can be shut off by writing b#00 to the FREQ bits of the VR_CTL register. This setting sets the internal power supply voltage (V_{DDINT}) to 0 V to provide the lowest static power dissipation. Any critical information stored internally (for example, memory contents, register contents, and other information) must be written to a non-volatile storage device prior to removing power if the processor state is to be preserved. Writing b#00 to the FREQ bits also causes EXT_WAKE0 and EXT_WAKE1 to transition low, which can be used to signal an external voltage regulator to shut down.

Since V_{DDEXT} and V_{DDMEM} can still be supplied in this mode, all of the external pins three-state, unless otherwise specified. This allows other devices that may be connected to the processor to still have power applied without drawing unwanted current.

The Ethernet or USB modules can wake up the internal supply regulator (ADSP-BF525 and ADSP-BF527 only) or signal an external regulator to wake up using EXT_WAKE0 or EXT_WAKE1. If PG15 does not connect as a $\overline{\text{PHYINT}}$ signal to an external PHY device, PG15 can be pulled low by any other device to wake the processor up. The processor can also be woken up by a real-time clock wakeup event or by asserting the $\overline{\text{RESET}}$ pin. All hibernate wakeup events initiate the hardware reset sequence. Individual sources are enabled by the VR_CTL register. The EXT_WAKEx signals are provided to indicate the occurrence of wakeup events.

As long as V_{DDEXT} is applied, the VR_CTL register maintains its state during hibernation. All other internal registers and memories, however, lose their content in the hibernate state. State variables may be held in external SRAM or SDRAM. The SCKELOW bit in the VR_CTL register controls whether or not SDRAM operates in self-refresh mode, which allows it to retain its content while the processor is in hibernate and through the subsequent reset sequence.

Power Savings

As shown in Table 5, the processor supports six different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions. By isolating the internal logic of the processor into its own power domain, separate from the RTC and other I/O, the processor can take advantage of dynamic power management without affecting the RTC or other I/O devices. There are no sequencing requirements for the various power domains, but all domains must be powered according to the appropriate Specifications table for processor Operating Conditions; even if the feature/peripheral is not used.

Table 5. Power Domains

Power Domain	V_{DD} Range
All internal logic, except RTC, Memory, USB, OTP	V_{DDINT}
RTC internal logic and crystal I/O	V_{DDRTC}
Memory logic	V_{DDMEM}
USB PHY logic	V_{DDUSB}
OTP logic	V_{DDOTP}
All other I/O	V_{DDEXT}

The dynamic power management feature of the processor allows both the processor's input voltage (V_{DDINT}) and clock frequency (f_{CCLK}) to be dynamically controlled.

The power dissipated by a processor is largely a function of its clock frequency and the square of the operating voltage. For example, reducing the clock frequency by 25% results in a 25% reduction in dynamic power dissipation, while reducing the voltage by 25% reduces dynamic power dissipation by more than 40%. Further, these power savings are additive, in that if the clock frequency and supply voltage are both reduced, the power savings can be dramatic, as shown in the following equations.

Power Savings Factor

$$= \frac{f_{CCLKRED}}{f_{CCLKNOM}} \times \left(\frac{V_{DDINTRED}}{V_{DDINTNOM}}\right)^2 \times \left(\frac{T_{RED}}{T_{NOM}}\right)$$

$$\%\ Power\ Savings\ =\ (1 - Power\ Savings\ Factor) \times 100\%$$

where the variables in the equations are:

$f_{CCLKNOM}$ is the nominal core clock frequency

$f_{CCLKRED}$ is the reduced core clock frequency

$V_{DDINTNOM}$ is the nominal internal supply voltage

$V_{DDINTRED}$ is the reduced internal supply voltage

T_{NOM} is the duration running at $f_{CCLKNOM}$

T_{RED} is the duration running at $f_{CCLKRED}$

ADSP-BF523/525/527 VOLTAGE REGULATION

The ADSP-BF523/525/527 provides an on-chip voltage regulator that can generate processor core voltage levels from an external supply. Figure 5 shows the typical external components required to complete the power management system. The regulator controls the internal logic voltage levels and is programmable with the voltage regulator control register (VR_CTL) in increments of 50 mV. To reduce standby power consumption, the internal voltage regulator can be programmed to remove power to the processor core while keeping I/O power supplied. While in the hibernate state, all external supplies (V_{DDEXT}, V_{DDMEM}, V_{DDUSB}, V_{DDOTP}) can still be applied, eliminating the need for external buffers. V_{DDRTC} must be applied at all times for correct hibernate operation. The voltage regulator can be activated from this power down state either through an RTC wakeup, a USB wakeup, an ethernet wakeup, or by asserting the $\overline{\text{RESET}}$ pin, each of which then initiates a boot sequence. The regulator can also be disabled and bypassed at the user's discretion.

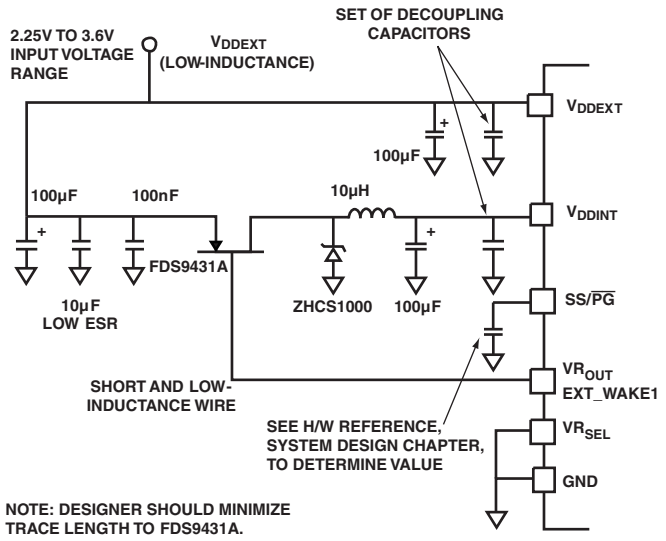


Figure 5. ADSP-BF523/525/527 Voltage Regulator Circuit

The voltage regulator has two modes set by the VR_{SEL} pin—the normal pulse width control of an external FET and the external supply mode which can signal a power down during hibernate to an external regulator. Set VR_{SEL} to V_{DDEXT} to use an external regulator or set VR_{SEL} to GND to use the internal regulator. In the external mode VR_{OUT} becomes EXT_WAKE1. If the internal regulator is used, EXT_WAKE0 can control other power sources in the system during the hibernate state. Both signals are high-true for power-up and may be connected directly to the low-true shut down input of many common regulators. The mode of the SS/$\overline{\text{PG}}$ (Soft Start/Power Good) signal also changes according to the state of VR_{SEL}. When using an internal regulator, the SS/$\overline{\text{PG}}$ pin is Soft Start, and when using an external regulator, it is Power Good. The Soft Start feature is recommended to reduce the inrush currents and to reduce V_{DDINT} voltage overshoot when coming out of hibernate or changing voltage levels. The Power Good ($\overline{\text{PG}}$) input signal allows the processor to start only after the internal voltage has reached a chosen level. In this way, the startup time of the external regulator is detected after hibernation. For a complete description of Soft Start and Power Good functionality, refer to the *ADSP-BF52x Blackfin Processor Hardware Reference*.

ADSP-BF522/524/526 VOLTAGE REGULATION

The ADSP-BF522/524/526 processor requires an external voltage regulator to power the V_{DDINT} domain. To reduce standby power consumption, the external voltage regulator can be signaled through EXT_WAKE0 or EXT_WAKE1 to remove power from the processor core. These identical signals are high-true for power-up and may be connected directly to the low-true shut down input of many common regulators. While in the hibernate state, all external supplies (V_{DDEXT}, V_{DDMEM}, V_{DDUSB}, V_{DDOTP}) can still be applied, eliminating the need for external buffers. V_{DDRTC} must be applied at all times for correct hibernate operation. The external voltage regulator can be activated from this power down state either through an RTC wakeup, a USB wakeup, an ethernet wakeup, or by asserting the $\overline{\text{RESET}}$ pin, each of which then initiates a boot sequence. EXT_WAKE0 or EXT_WAKE1 indicate a wakeup to the external voltage regulator. The Power Good ($\overline{\text{PG}}$) input signal allows the processor to start only after the internal voltage has reached a chosen level. In this way, the startup time of the external regulator is detected after hibernation. For a complete description of the Power Good functionality, refer to the *ADSP-BF52x Blackfin Processor Hardware Reference*.

CLOCK SIGNALS

The processor can be clocked by an external crystal, a sine wave input, or a buffered, shaped clock derived from an external clock oscillator.

If an external clock is used, it should be a TTL compatible signal and must not be halted, changed, or operated below the specified frequency during normal operation. This signal is connected to the processor's CLKIN pin. When an external clock is used, the XTAL pin must be left unconnected.

Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal may be used. For fundamental frequency operation, use the circuit shown in Figure 6. A parallel-resonant, fundamental frequency, microprocessor-grade crystal is connected across the CLKIN and XTAL pins. The on-chip resistance between CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended. The two capacitors and the series resistor shown in Figure 6 fine tune phase and amplitude of the sine frequency.

The capacitor and resistor values shown in Figure 6 are typical values only. The capacitor values are dependent upon the crystal manufacturers' load capacitance recommendations and the PCB physical layout. The resistor value depends on the drive level specified by the crystal manufacturer. The user should verify the customized values based on careful investigations on multiple devices over temperature range.

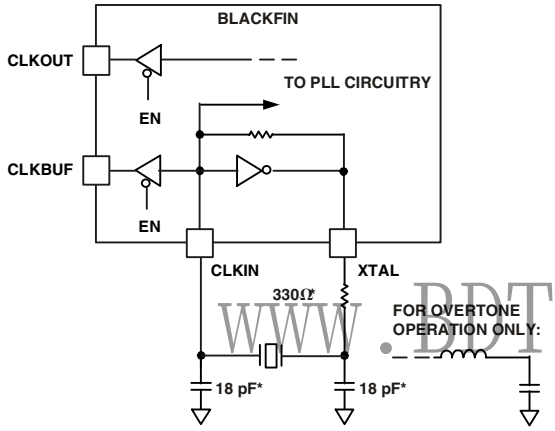


Figure 6. External Crystal Connections

A third-overtone crystal can be used for frequencies above 25 MHz. The circuit is then modified to ensure crystal operation only at the third overtone, by adding a tuned inductor circuit as shown in Figure 6. A design procedure for third-overtone operation is discussed in detail in application note *(EE-168) Using Third Overtone Crystals with the ADSP-218x DSP* on the Analog Devices website (www.analog.com)—use site search on "EE-168."

The CLKBUF pin is an output pin, which is a buffered version of the input clock. This pin is particularly useful in Ethernet applications to limit the number of required clock sources in the system. In this type of application, a single 25 MHz or 50 MHz crystal may be applied directly to the processor. The 25 MHz or 50 MHz output of CLKBUF can then be connected to an external Ethernet MII or RMII PHY device. If instead of a crystal, an external oscillator is used at CLKIN, CLKBUF will not have the 40/60 duty cycle required by some devices. The CLKBUF output is active by default and can be disabled for power savings reasons using the VR_CTL register.

The Blackfin core runs at a different clock rate than the on-chip peripherals. As shown in Figure 7, the core clock (CCLK) and system peripheral clock (SCLK) are derived from the input clock (CLKIN) signal. An on-chip PLL is capable of multiplying the CLKIN signal by a programmable 0.5× to 64× multiplication factor (bounded by specified minimum and maximum VCO frequencies). The default multiplier is 10×, but it can be modified by a software instruction sequence.

On-the-fly frequency changes can be effected by simply writing to the PLL_DIV register. The maximum allowed CCLK and SCLK rates depend on the applied voltages V_{DDINT}, V_{DDEXT}, and V_{DDMEM}; the VCO is always permitted to run up to the frequency specified by the part's speed grade. The CLKOUT pin reflects the SCLK frequency to the off-chip world. It is part of the SDRAM interface, but it functions as a reference signal in other timing specifications as well. While active by default, it can be disabled using the EBIU_SDGCTL and EBIU_AMGCTL registers.

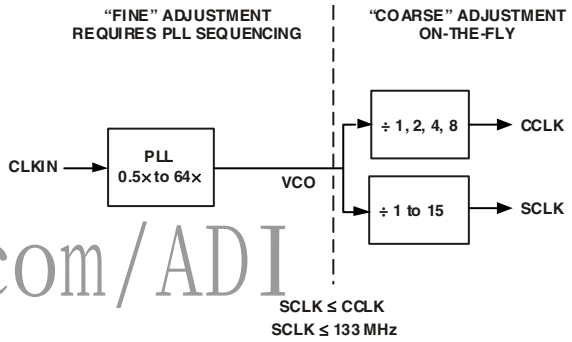


Figure 7. Frequency Modification Methods

All on-chip peripherals are clocked by the system clock (SCLK). The system clock frequency is programmable by means of the SSEL3–0 bits of the PLL_DIV register. The values programmed into the SSEL fields define a divide ratio between the PLL output (VCO) and the system clock. SCLK divider values are 1 through 15. Table 6 illustrates typical system clock ratios.

Note that the divisor ratio must be chosen to limit the system clock frequency to its maximum of f_{SCLK}. The SSEL value can be changed dynamically without any PLL lock latencies by writing the appropriate values to the PLL divisor register (PLL_DIV).

Table 6. Example System Clock Ratios

Signal Name SSEL3–0	Divider Ratio VCO/SCLK	Example Frequency Ratios (MHz)	
		VCO	SCLK
0001	1:1	100	100
0110	6:1	300	50
1010	10:1	500	50

The core clock (CCLK) frequency can also be dynamically changed by means of the CSEL1–0 bits of the PLL_DIV register. Supported CCLK divider ratios are 1, 2, 4, and 8, as shown in Table 7. This programmable core clock capability is useful for fast core frequency modifications.

Table 7. Core Clock Ratios

Signal Name CSEL1–0	Divider Ratio VCO/CCLK	Example Frequency Ratios (MHz)	
		VCO	CCLK
00	1:1	300	300
01	2:1	300	150
10	4:1	500	125
11	8:1	200	25

The maximum CCLK frequency not only depends on the part's speed grade (see Page 72), it also depends on the applied V_{DDINT} voltage. See Table 12 and Table 15 for details. The maximal system clock rate (SCLK) depends on the chip package and the applied V_{DDINT}, V_{DDEXT}, and V_{DDMEM} voltages (see Table 14 and Table 17).

BOOTING MODES

The processor has several mechanisms (listed in Table 8) for automatically loading internal and external memory after a reset. The boot mode is defined by four BMODE input pins dedicated to this purpose. There are two categories of boot modes. In master boot modes the processor actively loads data from parallel or serial memories. In slave boot modes the processor receives data from external host devices.

The boot modes listed in Table 8 provide a number of mechanisms for automatically loading the processor's internal and external memories after a reset. By default, all boot modes use the slowest meaningful configuration settings. Default settings can be altered via the initialization code feature at boot time or by proper OTP programming at pre-boot time. The BMODE pins of the reset configuration register, sampled during power-on resets and software-initiated resets, implement the modes shown in Table 8.

Table 8. Booting Modes

BMODE3–0	Description
0000	Idle - No boot
0001	Boot from 8- or 16-bit external flash memory
0010	Boot from 16-bit asynchronous FIFO
0011	Boot from serial SPI memory (EEPROM or flash)
0100	Boot from SPI host device
0101	Boot from serial TWI memory (EEPROM/flash)
0110	Boot from TWI host
0111	Boot from UART0 Host
1000	Boot from UART1 Host
1001	Reserved
1010	Boot from SDRAM
1011	Boot from OTP memory
1100	Boot from 8-bit NAND flash via NFC using PORTF data pins
1101	Boot from 8-bit NAND flash via NFC using PORTH data pins
1110	Boot from 16-Bit Host DMA
1111	Boot from 8-Bit Host DMA

- Idle/no boot mode (BMODE = 0x0) — In this mode, the processor goes into idle. The idle boot mode helps recover from illegal operating modes, such as when the OTP memory has been misconfigured.
- Boot from 8-bit or 16-bit external flash memory (BMODE = 0x1) — In this mode, the boot kernel loads the first block header from address 0x2000 0000, and (depending on instructions contained in the header) the boot kernel performs an 8- or 16-bit boot or starts program execution at the address provided by the header. By default, all configuration settings are set for the slowest device possible (3-cycle hold time, 15-cycle R/W access times, 4-cycle setup).

  The ARDY is not enabled by default, but it can be enabled through OTP programming. Similarly, all interface behavior and timings can be customized through OTP programming. This includes activation of burst-mode or page-mode operation. In this mode, all asynchronous interface signals are enabled at the port muxing level.
- Boot from 16-bit asynchronous FIFO (BMODE = 0x2) — In this mode, the boot kernel starts booting from address 0x2030 0000. Every 16-bit word that the boot kernel has to read from the FIFO must be requested by placing a low pulse on the DMAR1 pin.
- Boot from serial SPI memory, EEPROM or flash (BMODE = 0x3) — 8-, 16-, 24-, or 32-bit addressable devices are supported. The processor uses the PG1 GPIO pin to select a single SPI EEPROM/flash device and sub-

mits a read command and successive address bytes (0x00) until a valid 8-, 16-, 24-, or 32-bit addressable device is detected. Pull-up resistors are required on the SPISEL1 and MISO pins. By default, a value of 0x85 is written to the SPI_BAUD register.

- Boot from SPI host device (BMODE = 0x4) — The processor operates in SPI slave mode and is configured to receive the bytes of the LDR file from an SPI host (master) agent. The HWAIT signal must be interrogated by the host before every transmitted byte. A pull-up resistor is required on the SPISS input. A pull-down on the serial clock (SCK) may improve signal quality and booting robustness.
- Boot from serial TWI memory, EEPROM/flash (BMODE = 0x5) — The processor operates in master mode and selects the TWI slave connected to the TWI with the unique ID 0xA0.

  The processor submits successive read commands to the memory device starting at internal address 0x0000 and begins clocking data into the processor. The TWI memory device should comply with the Philips I²C® Bus Specification version 2.1 and should be able to auto-increment its internal address counter such that the contents of the memory device can be read sequentially. By default, a PRESCALE value of 0xA and a TWI_CLKDIV value of 0x0811 are used. Unless altered by OTP settings, an I²C memory that takes two address bytes is assumed. The development tools ensure that data booted to memories that cannot be accessed by the Blackfin core is written to an intermediate storage location and then copied to the final destination via memory DMA.
- Boot from TWI host (BMODE = 0x6) — The TWI host selects the slave with the unique ID 0x5F.

  The processor replies with an acknowledgement and the host then downloads the boot stream. The TWI host agent should comply with the Philips I²C Bus Specification version 2.1. An I²C multiplexer can be used to select one processor at a time when booting multiple processors from a single TWI.
- Boot from UART0 host on Port G (BMODE = 0x7) — Using an autobaud handshake sequence, a boot-stream formatted program is downloaded by the host. The host selects a bit rate within the UART clocking capabilities.

  When performing the autobaud, the UART expects a "@" (0x40) character (eight bits data, one start bit, one stop bit, no parity bit) on the UART0RX pin to determine the bit rate. The UART then replies with an acknowledgement composed of 4 bytes (0xBF, the value of UART0_DLL, the value of UART0_DLH, then 0x00). The host can then download the boot stream. To hold off the host the Blackfin processor signals the host with the boot host wait (HWAIT) signal. Therefore, the host must monitor HWAIT before every transmitted byte.
- Boot from UART1 host on Port F (BMODE = 0x8). Same as BMODE = 0x7 except that the UART1 port is used.
- Boot from SDRAM (BMODE = 0xA) This is a warm boot scenario, where the boot kernel starts booting from address 0x0000 0010. The SDRAM is expected to contain a valid boot stream and the SDRAM controller must be configured by the OTP settings.
- Boot from OTP memory (BMODE = 0xB) — This provides a stand-alone booting method. The boot stream is loaded from on-chip OTP memory. By default, the boot stream is expected to start from OTP page 0x40 and can occupy all public OTP memory up to page 0xDF. This is 2560 bytes. Since the start page is programmable, the maximum size of the boot stream can be extended to 3072 bytes.

Table 9. Fourth Byte for Large Page Devices

Bit	Parameter	Value	Meaning
D1:D0	Page Size (excluding spare area)	00 01 10 11	1K byte 2K byte 4K byte 8K byte
D2	Spare Area Size	00 01	8 byte/512 byte 16 byte/512 byte
D5:D4	Block Size (excluding spare area)	00 01 10 11	64K byte 128K byte 256K byte 512K byte
D6	Bus width	00 01	x8 not supported
D3, D7	Not Used for configuration		

- Boot from 8-bit external NAND flash memory (BMODE = 0xC and BMODE = 0xD) — In this mode, auto detection of the NAND flash device is performed.

  BMODE = 0xC, the processor configures PORTF GPIO pins PF7:0 for the NAND data pins and PORTH pins PH15:10 for the NAND control signals.

  BMODE = 0xD, the processor configures PORTH GPIO pins PH7:0 for the NAND data pins and PORTH pins PH15:10 for the NAND control signals.

  For correct device operation pull-up resistors are required on both ND_CE (PH10) and ND_BUSY (PH13) signals. By default, a value of 0x0033 is written to the NFC_CTL register. The booting procedure always starts by booting from byte 0 of block 0 of the NAND flash device.

NAND flash boot supports the following features:

—Device Auto Detection

—Error Detection & Correction for maximum reliability

—No boot stream size limitation

—Peripheral DMA providing efficient transfer of all data (excluding the ECC parity data)

—Software-configurable boot mode for booting from boot streams spanning multiple blocks, including bad blocks

—Software-configurable boot mode for booting from multiple copies of the boot stream, allowing for handling of bad blocks and uncorrectable errors

—Configurable timing via OTP memory

Small page NAND flash devices must have a 512-byte page size, 32 pages per block, a 16-byte spare area size, and a bus configuration of 8 bits. By default, all read requests from the NAND flash are followed by four address cycles. If the NAND flash device requires only three address cycles, the device must be capable of ignoring the additional address cycles.

The small page NAND flash device must comply with the following command set:

—Reset: 0xFF

—Read lower half of page: 0x00

—Read upper half of page: 0x01

—Read spare area: 0x50

For large-page NAND-flash devices, the four-byte electronic signature is read in order to configure the kernel for booting, which allows support for multiple large-page devices. The fourth byte of the electronic signature must comply with the specification in Table 9.

Any NAND flash array configuration from Table 9, excluding 16-bit devices, that also complies with the command set listed below are directly supported by the boot kernel. There are no restrictions on the page size or block size as imposed by the small-page boot kernel.

For devices consisting of a five-byte signature, only four are read. The fourth must comply as outlined above.

Large page devices must support the following command set:

—Reset: 0xFF

—Read Electronic Signature: 0x90

—Read: 0x00, 0x30 (confirm command)

Large-page devices must not support or react to NAND flash command 0x50. This is a small-page NAND flash command used for device auto detection.

By default, the boot kernel will always issue five address cycles; therefore, if a large page device requires only four cycles, the device must be capable of ignoring the additional address cycles.

- Boot from 16-Bit Host DMA (BMODE = 0xE) — In this mode, the host DMA port is configured in 16-bit Acknowledge mode, with little endian data formatting. Unlike other modes, the host is responsible for interpreting the boot stream. It writes data blocks individually into the Host DMA port. Before configuring the DMA settings for each block, the host may either poll the ALLOW_CONFIG bit in HOST_STATUS or wait to be interrupted by the HWAIT signal. When using HWAIT, the host must still check ALLOW_CONFIG at least once before beginning to configure the Host DMA Port. After completing the configuration, the host is required to poll the READY bit in HOST_STATUS before beginning to transfer data. When the host sends an HIRQ control command, the boot kernel issues a CALL instruction to address 0xFFA0 0000. It is the host's responsibility to ensure that valid code has been placed at this address. The routine at 0xFFA0 0000 can be a simple initialization routine to configure internal resources, such as the SDRAM controller, which then returns using an RTS instruction. The routine may also by the final application, which will never return to the boot kernel.
- Boot from 8-Bit Host DMA (BMODE = 0xF) — In this mode, the Host DMA port is configured in 8-bit interrupt mode, with little endian data formatting. Unlike other modes, the host is responsible for interpreting the boot stream. It writes data blocks individually into the Host DMA port. Before configuring the DMA settings for each block, the host may either poll the ALLOW_CONFIG bit in HOST_STATUS or wait to be interrupted by the HWAIT signal. When using HWAIT, the host must still check ALLOW_CONFIG at least once before beginning to configure the Host DMA Port. The host will receive an interrupt from the HOST_ACK signal every time it is allowed to send the next FIFO depths worth (sixteen 32-bit words) of information. When the host sends an HIRQ control command, the boot kernel issues a CALL instruction to address 0xFFA0 0000. It is the host's responsibility to ensure valid code has been placed at this address. The routine at 0xFFA0 0000 can be a simple initialization routine to configure internal resources, such as the SDRAM controller, which then returns using an RTS instruction. The routine may also by the final application, which will never return to the boot kernel.

For each of the boot modes, a 16-byte header is first read from an external memory device. The header specifies the number of bytes to be transferred and the memory destination address. Multiple memory blocks may be loaded by any boot sequence. Once all blocks are loaded, program execution commences from the address stored in the EVT1 register.

Prior to booting, the pre-boot routine interrogates the OTP memory. Individual boot modes can be customized or even disabled based on OTP programming. External hardware, especially booting hosts, may watch the HWAIT signal to determine when the pre-boot has finished and the boot kernel starts the boot process. By programming OTP memory, the user can

also instruct the pre-boot routine to customize the PLL, Internal Voltage Regulator (ADSP-BF523/525/527 only), SDRAM Controller, and Asynchronous Memory Controller.

The boot kernel differentiates between a regular hardware reset and a wakeup-from-hibernate event to speed up booting in the later case. Bits 6-4 in the system reset configuration (SYSCR) register can be used to bypass the pre-boot routine and/or boot kernel in case of a software reset. They can also be used to simulate a wakeup-from-hibernate boot in the software reset case.

The boot process can be further customized by "initialization code." This is a piece of code that is loaded and executed prior to the regular application boot. Typically, this is used to configure the SDRAM controller or to speed up booting by managing the PLL, clock frequencies, wait states, or serial bit rates.

The boot ROM also features C-callable function that can be called by the user application at run time. This enables second-stage boot or boot management schemes to be implemented with ease.

INSTRUCTION SET DESCRIPTION

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.
- 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.

DEVELOPMENT TOOLS

The processor is supported with a complete set of CROSSCORE® software and hardware development tools, including Analog Devices emulators and VisualDSP++® development environment. The same emulator hardware that supports other Blackfin processors also fully emulates the ADSP-BF522/524/526 and ADSP-BF523/525/527 processors.

EZ-KIT Lite® Evaluation Board

For evaluation of ADSP-BF523/525/527 processors, use the ADSP-BF527 EZ-KIT Lite board available from Analog Devices. Order part number ADZS-BF527-EZLITE. The board comes with on-chip emulation capabilities and is equipped to enable software development. Multiple daughter cards are available. An ADSP-BF526 EZ-KIT Lite board is under development.

DESIGNING AN EMULATOR-COMPATIBLE PROCESSOR BOARD (TARGET)

The Analog Devices family of emulators are tools that every system developer needs in order to test and debug hardware and software systems. Analog Devices has supplied an IEEE 1149.1 JTAG Test Access Port (TAP) on each JTAG processor. The emulator uses the TAP to access the internal features of the processor, allowing the developer to load code, set breakpoints, observe variables, observe memory, and examine registers. The processor must be halted to send data and commands, but once an operation has been completed by the emulator, the processor system is set running at full speed with no impact on system timing.

To use these emulators, the target board must include a header that connects the processor's JTAG port to the emulator.

For details on target board design issues including mechanical layout, single processor connections, multiprocessor scan chains, signal buffering, signal termination, and emulator pod logic, see *(EE-68) Analog Devices JTAG Emulation Technical Reference* on the Analog Devices website (www.analog.com)—use site search on "EE-68." This document is updated regularly to keep pace with improvements to emulator support.

RELATED DOCUMENTS

The following publications that describe the ADSP-BF522/524/526 and ADSP-BF523/525/527 processors (and related processors) can be ordered from any Analog Devices sales office or accessed electronically on our website:

- *Getting Started With Blackfin Processors*
- *ADSP-BF52x Blackfin Processor Hardware Reference*
- *Blackfin Processor Programming Reference*
- *ADSP-BF522/524/526 Blackfin Processor Anomaly List*
- *ADSP-BF523/525/527 Blackfin Processor Anomaly List*

LOCKBOX SECURE TECHNOLOGY DISCLAIMER

Analog Devices products containing Lockbox Secure Technology are warranted by Analog Devices as detailed in the Analog Devices Standard Terms and Conditions of Sale. To our knowledge, the Lockbox Secure Technology, when used in accordance

with the data sheet and hardware reference manual specifications, provides a secure method of implementing code and data safeguards. However, Analog Devices does not guarantee that this technology provides absolute security.

ACCORDINGLY, ANALOG DEVICES HEREBY DISCLAIMS ANY AND ALL EXPRESS AND IMPLIED WARRANTIES THAT THE LOCKBOX SECURE TECHNOLOGY CANNOT BE BREACHED, COMPROMISED OR OTHERWISE CIRCUMVENTED AND IN NO EVENT SHALL ANALOG DEVICES BE LIABLE FOR ANY LOSS, DAMAGE DESTRUCTION OR RELEASE OF DATA, INFORMATION, PHYSICAL PROPERTY OR INTELLECTUAL PROPERTY.

SIGNAL DESCRIPTIONS

Signal definitions for the ADSP-BF522/524/526 and ADSP-BF523/525/527 processors are listed in Table 10. In order to maintain maximum function and reduce package size and ball count, some balls have dual, multiplexed functions. In cases where ball function is reconfigurable, the default state is shown in plain text, while the alternate function is shown in italics.

All pins are three-stated during and immediately after reset, with the exception of the external memory interface, asynchronous and synchronous memory control, and the buffered XTAL output pin (CLKBUF). On the external memory interface, the control and address lines are driven high, with the exception of CLKOUT, which toggles at the system clock rate. If, however, the $\overline{BR}$ pin is asserted, then the memory pins are also three-stated.

All I/O pins have their input buffers disabled with the exception of the pins that need pull-ups or pull-downs, as noted in Table 10.

Table 10. Signal Descriptions

Signal Name	Type	Function	Driver Type[1]
EBIU			
ADDR19–1	O	Address Bus	A
DATA15–0	I/O	Data Bus	A
$\overline{ABE1–0}$/*SDQM1–0*	O	Byte Enables/*Data Mask*	A
$\overline{AMS3–0}$	O	Bank Select	A
ARDY	I	Hardware Ready Control	
$\overline{AOE}$	O	Output Enable	A
$\overline{ARE}$	O	Read Enable	A
$\overline{AWE}$	O	Write Enable	A
$\overline{SRAS}$	O	SDRAM Row Address Strobe	A
$\overline{SCAS}$	O	SDRAM Column Address Strobe	A
$\overline{SWE}$	O	SDRAM Write Enable	A
SCKE	O	SDRAM Clock Enable	A
CLKOUT	O	SDRAM Clock Output	B
SA10	O	SDRAM A10 Signal	A
$\overline{SMS}$	O	SDRAM Bank Select	A
USB 2.0 HS OTG			
USB_DP	I/O	Data + (This ball should be pulled low when USB is unused or not present.)	F
USB_DM	I/O	Data - (This ball should be pulled low when USB is unused or not present.)	F
USB_XI	I	USB Crystal Input (This ball should be pulled low when USB is unused or not present.)	
USB_XO	O	USB Crystal Output (This ball should be left unconnected when USB is unused or not present.)	F
USB_ID	I	USB OTG mode (This ball should be pulled low when USB is unused or not present.)	
USB_VREF	I	USB voltage reference (Connect to GND through a 0.1 µF capacitor, or leave unconnected if USB is unused or not present.)	
USB_RSET	I	USB resistance set. (This ball should be left unconnected when USB is unused or not present.)	
USB_VBUS	I/O 5V	USB VBUS (USB_VBUS is an output only during initialization of USB OTG session request pulses. Host mode or OTG type A mode require that an external voltage source of 5V, at 8mA or more–per the OTG specification–be applied to VBUS. Other OTG modes require that this external voltage be disabled. This ball should be pulled low when USB is unused or not present.)	F

Table 10. Signal Descriptions (Continued)

Signal Name	Type	Function	Driver Type[1]
Port F: *GPIO and Multiplexed Peripherals*			
PF0/*PPI D0/DR0PRI /ND_D0A*	I/O	GPIO/*PPI Data 0/SPORT0 Primary Receive Data /NAND Alternate Data 0*	C
PF1/*PPI D1/RFS0/ND_D1A*	I/O	GPIO/*PPI Data 1/SPORT0 Receive Frame Sync /NAND Alternate Data 1*	C
PF2/*PPI D2/RSCLK0/ND_D2A*	I/O	GPIO/*PPI Data 2/SPORT0 Receive Serial Clock /NAND Alternate Data 2/Alternate Capture Input 0*	D
PF3/*PPI D3/DT0PRI/ND_D3A*	I/O	GPIO/*PPI Data 3/SPORT0 Transmit Primary Data /NAND Alternate Data 3*	C
PF4/*PPI D4/TFS0/ND_D4A/TACLK0*	I/O	GPIO/*PPI Data 4/SPORT0 Transmit Frame Sync /NAND Alternate Data 4/Alternate Timer Clock 0*	C
PF5/*PPI D5/TSCLK0/ND_D5A/TACLK1*	I/O	GPIO/*PPI Data 5/SPORT0 Transmit Serial Clock /NAND Alternate Data 5/Alternate Timer Clock 1*	D
PF6/*PPI D6/DT0SEC/ND_D6A/TACI0*	I/O	GPIO/*PPI Data 6/SPORT0 Transmit Secondary Data /NAND Alternate Data 6/Alternate Capture Input 0*	C
PF7/*PPI D7/DR0SEC/ND_D7A/TACI1*	I/O	GPIO/*PPI Data 7/SPORT0 Receive Secondary Data /NAND Alternate Data 7/Alternate Capture Input 1*	C
PF8/*PPI D8/DR1PRI*	I/O	GPIO/*PPI Data 8/SPORT1 Primary Receive Data*	C
PF9/*PPI D9/RSCLK1/*$\overline{SPISEL6}$	I/O	GPIO/*PPI Data 9/SPORT1 Receive Serial Clock/SPI Slave Select 6*	D
PF10/*PPI D10/RFS1/*$\overline{SPISEL7}$	I/O	GPIO/*PPI Data 10/SPORT1 Receive Frame Sync/SPI Slave Select 7*	C
PF11/*PPI D11/TFS1/CZM*	I/O	GPIO/*PPI Data 11/SPORT1 Transmit Frame Sync/Counter Zero Marker*	C
PF12/*PPI D12/DT1PRI/*$\overline{SPISEL2}$/*CDG*	I/O	GPIO/*PPI Data 12/SPORT1 Transmit Primary Data/SPI Slave Select 2/Counter Down Gate*	C
PF13/*PPI D13/TSCLK1/*$\overline{SPISEL3}$/*CUD*	I/O	GPIO/*PPI Data 13/SPORT1 Transmit Serial Clock/SPI Slave Select 3/Counter Up Direction*	D
PF14/*PPI D14/DT1SEC/UART1TX*	I/O	GPIO/*PPI Data 14/SPORT1 Transmit Secondary Data/UART1 Transmit*	C
PF15/*PPI D15/DR1SEC/UART1RX/TACI3*	I/O	GPIO/*PPI Data 15/SPORT1 Receive Secondary Data /UART1 Receive /Alternate Capture Input 3*	C
Port G: *GPIO and Multiplexed Peripherals*			
PG0/*HWAIT*	I/O	GPIO/*Boot Host Wait*[2]	C
PG1/$\overline{SPISS}$/$\overline{SPISEL1}$	I/O	GPIO/*SPI Slave Select Input/SPI Slave Select 1*	C
PG2/*SCK*	I/O	GPIO/*SPI Clock*	D
PG3/*MISO/DR0SECA*	I/O	GPIO/*SPI Master In Slave Out/Sport 0 Alternate Receive Data Secondary*	C
PG4/*MOSI/DT0SECA*	I/O	GPIO/*SPI Master Out Slave In/Sport 0 Alternate Transmit Data Secondary*	C
PG5/*TMR1/PPI_FS2*	I/O	GPIO/*Timer1/PPI Frame Sync2*	C
PG6/*DT0PRIA/TMR2/PPI_FS3*	I/O	GPIO/*SPORT0 Alternate Primary Transmit Data / Timer2 / PPI Frame Sync3*	C
PG7/*TMR3/DR0PRIA/UART0TX*	I/O	GPIO/*Timer3/Sport 0 Alternate Receive Data Primary/UART0 Transmit*	C
PG8/*TMR4/RFS0A/UART0RX/TACI4*	I/O	GPIO/*Timer 4/Sport 0 Alternate Receive Clock/Frame Sync /UART0 Receive/Alternate Capture Input 4*	C
PG9/*TMR5/RSCLK0A/TACI5*	I/O	GPIO/*Timer5/Sport 0 Alternate Receive Clock /Alternate Capture Input 5*	D
PG10/*TMR6/TSCLK0A/TACI6*	I/O	GPIO/*Timer 6 /Sport 0 Alternate Transmit /Alternate Capture Input 6*	D
PG11/*TMR7/*$\overline{HOST\_WR}$	I/O	GPIO/*Timer7/Host DMA Write Enable*	C
PG12/*DMAR1/UART1TXA/HOST_ACK*	I/O	GPIO/*DMA Request 1/Alternate UART1 Transmit/Host DMA Acknowledge*	C

Table 10. Signal Descriptions (Continued)

Signal Name	Type	Function	Driver Type[1]
PG13/*DMAR0/UART1RXA/HOST_ADDR/TACI2*	I/O	GPIO/*DMA Request 0/Alternate UART1 Receive/Host DMA Address/Alternate Capture Input 2*	C
PG14/*TSCLK0A1/MDC/*$\overline{HOST\_RD}$	I/O	GPIO/*SPORT0 Alternate 1 Transmit/Ethernet Management Channel Clock /Host DMA Read Enable*	D
PG15[3]/*TFS0A/*$\overline{MII\ PHYINT}$/$\overline{RMII\ MDINT}$/$\overline{HOST\_CE}$	I/O	GPIO/*SPORT0 Alternate Transmit Frame Sync/Ethernet/MII PHY Interrupt/RMII Management Channel Data Interrupt/Host DMA Chip Enable*	C
Port H: *GPIO and Multiplexed Peripherals*			
PH0/*ND_D0/MIICRS/RMIICRSDV/HOST_D0*	I/O	GPIO/*NAND D0/Ethernet MII or RMII Carrier Sense/Host DMA D0*	C
PH1/*ND_D1/ERxER/HOST_D1*	I/O	GPIO/*NAND D1/Ethernet MII or RMII Receive Error/Host DMA D1*	C
PH2/*ND_D2/MDIO/HOST_D2*	I/O	GPIO/*NAND D2/Ethernet Management Channel Serial Data/Host DMA D2*	C
PH3/*ND_D3/ETxEN/HOST_D3*	I/O	GPIO/*NAND D3/Ethernet MII Transmit Enable/Host DMA D3*	C
PH4/*ND_D4/MIITxCLK/RMIIREF_CLK/HOST_D4*	I/O	GPIO/*NAND D4/Ethernet MII or RMII Reference Clock/Host D4*	C
PH5/*ND_D5/ETxD0/HOST_D5*	I/O	GPIO/*NAND D5/Ethernet MII or RMII Transmit D0/Host DMA D5*	C
PH6/*ND_D6/ERxD0/HOST_D6*	I/O	GPIO/*NAND D6/Ethernet MII or RMII Receive D0/Host DMA D6*	C
PH7/*ND_D7/ETxD1/HOST_D7*	I/O	GPIO/*NAND D7/Ethernet MII or RMII Transmit D1/Host DMA D7*	C
PH8/$\overline{SPISEL4}$/*ERxD1/HOST_D8/TACLK2*	I/O	GPIO/*Alternate Capture Input 2/Ethernet MII or RMII Receive D1/Host DMA D8 /SPI Slave Select 4*	C
PH9/$\overline{SPISEL5}$/*ETxD2/HOST_D9/TACLK3*	I/O	GPIO/*SPI Slave Select 5/Ethernet MII Transmit D2/Host DMA D9 /Alternate Timer Clock 3*	C
PH10/$\overline{ND\_CE}$/*ERxD2/HOST_D10*	I/O	GPIO/*NAND Chip Enable/Ethernet MII Receive D2/Host DMA D10*	C
PH11/$\overline{ND\_WE}$/*ETxD3/HOST_D11*	I/O	GPIO/*NAND Write Enable/Ethernet MII Transmit D3/Host DMA D11*	C
PH12/$\overline{ND\_RE}$/*ERxD3/HOST_D12*	I/O	GPIO/*NAND Read Enable/Ethernet MII Receive D3/Host DMA D12*	C
PH13/$\overline{ND\_BUSY}$/*ERxCLK/HOST_D13*	I/O	GPIO/*NAND Busy/Ethernet MII Receive Clock/Host DMA D13*	C
PH14/*ND_CLE/ERxDV/HOST_D14*	I/O	GPIO/*NAND Command Latch Enable/Ethernet MII or RMII Receive Data Valid/Host DMA D14*	C
PH15/*ND_ALE/COL/HOST_D15*	I/O	GPIO/*NAND Address Latch Enable/Ethernet MII Collision/Host DMA Data 15*	C
Port J: Multiplexed Peripherals			
PJ0: PPI_FS1/*TMR0*	I/O	PPI Frame Sync1/*Timer0*	C
PJ1: PPI_CLK/*TMRCLK*	I	PPI Clock/*Timer Clock*	
PJ2: SCL	I/O 5V	TWI Serial Clock (This pin is an open-drain output and requires a pull-up resistor.[4])	E
PJ3: SDA	I/O 5V	TWI Serial Data (This pin is an open-drain output and requires a pull-up resistor.[4])	E
Real Time Clock			
RTXI	I	RTC Crystal Input (This ball should be pulled low when not used.)	
RTXO	O	RTC Crystal Output	
JTAG Port			
TCK	I	JTAG Clock	
TDO	O	JTAG Serial Data Out	C
TDI	I	JTAG Serial Data In	
TMS	I	JTAG Mode Select	
$\overline{TRST}$	I	JTAG Reset (This ball should be pulled low if the JTAG port is not used.)	
$\overline{EMU}$	O	Emulation Output	C

Table 10. Signal Descriptions (Continued)

Signal Name	Type	Function	Driver Type[1]
Clock			
CLKIN	I	Clock/Crystal Input	
XTAL	O	Crystal Output	
CLKBUF	O	Buffered XTAL Output	C
Mode Controls			
$\overline{\text{RESET}}$	I	Reset	
$\overline{\text{NMI}}$	I	Nonmaskable Interrupt (This ball should be pulled high when not used.)	
BMODE3–0	I	Boot Mode Strap 3-0	
ADSP-BF523/525/527 Voltage Regulator			
VR_{SEL}	I	External/Internal Voltage Regulator Select	
VR_{OUT}/EXT_WAKE1	O	External FET Drive/Wake up Indication 1	G
EXT_WAKE0	O	Wake up Indication 0	C
SS/$\overline{\text{PG}}$	I	Soft Start/Power Good	
ADSP-BF522/524/526 Voltage Regulation I/F			
EXT_WAKE1	O	Wake up Indication 1	C
EXT_WAKE0	O	Wake up Indication 0	C
$\overline{\text{PG}}$	I	Power Good	
Power Supplies		**ALL SUPPLIES MUST BE POWERED** See Operating Conditions for ADSP-BF523/525/527 on Page 29, and see Operating Conditions for ADSP-BF522/524/526 on Page 27.	
V_{DDEXT}	P	I/O Power Supply	
V_{DDINT}	P	Internal Power Supply	
V_{DDRTC}	P	Real Time Clock Power Supply	
V_{DDUSB}	P	3.3 V USB Phy Power Supply	
V_{DDMEM}	P	MEM Power Supply	
V_{DDOTP}	P	OTP Power Supply	
V_{PPOTP}	P	OTP Programming Voltage	
V_{SS}	G	Ground for All Supplies	

[1] See Output Drive Currents on Page 58 for more information about each driver type.
[2] HWAIT must be pulled high or low to configure polarity. See Booting Modes on Page 18.
[3] When driven low, this ball can be used to wake up the processor from the hibernate state, either in normal GPIO mode or in Ethernet mode as $\overline{\text{MII PHYINT}}$. If the ball is used for wake up, enable the feature with the PHYWE bit in the VR_CTL register, and pull-up the ball with a resistor.
[4] Consult version 2.1 of the I^2C specification for the proper resistor value.

SPECIFICATIONS

Specifications are subject to change without notice.

OPERATING CONDITIONS FOR ADSP-BF522/524/526

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DDINT}	Internal Supply Voltage		tbd	tbd	tbd	V
V_{DDEXT}[1]	External Supply Voltage		1.70	1.8, 2.5 or 3.3	3.6	V
V_{DDRTC}[2]	RTC Power Supply Voltage		2.25		3.6	V
V_{DDMEM}[3]	MEM Supply Voltage		1.70	1.8, 2.5 or 3.3	3.6	V
V_{DDOTP}	OTP Supply Voltage		2.25	2.5	2.75	V
V_{PPOTP}	OTP Programming Voltage					
	For Reads		2.25	2.5	2.75	V
	For Writes[4]		6.9	7.0	7.1	V
V_{DDUSB}	USB Supply Voltage[5]		3.0	3.3	3.6	V
V_{IH}	High Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 1.90 V	1.1		3.6	V
V_{IH}	High Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 2.75 V	1.7		3.6	V
V_{IH}	High Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 3.6 V	2.0		3.6	V
V_{IHTWI}	High Level Input Voltage	V_{DDEXT} = 1.90 V/2.75 V/3.6 V	0.7 x V_{BUSTWI}		V_{BUSTWI}[8]	V
V_{IL}	Low Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 1.7 V	–0.3		0.6	V
V_{IL}	Low Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 2.25 V	–0.3		0.7	V
V_{IL}	Low Level Input Voltage[6, 7]	V_{DDEXT}/V_{DDMEM} = 3.0 V	–0.3		0.8	V
V_{ILTWI}	Low Level Input Voltage	V_{DDEXT} = minimum	–0.3		0.3 x V_{BUSTWI}[9]	V
T_J	Junction Temperature	289-Ball CSP_BGA @ $T_{AMBIENT}$ = 0°C to +70°C	0		+105	°C
T_J	Junction Temperature	208-Ball CSP_BGA @ $T_{AMBIENT}$ = 0°C to +70°C	0		+105	°C
T_J	Junction Temperature	208-Ball CSP_BGA @ $T_{AMBIENT}$ = –40°C to +85°C	–40		+105	°C

[1] Must remain powered (even if the associated function is not used).

[2] If not used, power with V_{DDEXT}.

[3] Balls that use V_{DDMEM} are DATA15–0, ADDR19–1, $\overline{ABE1–0}$, $\overline{ARE}$, $\overline{AWE}$, $\overline{AOE}$, $\overline{AMS3–0}$, ARDY, SA10, $\overline{SWE}$, $\overline{SCAS}$, CLKOUT, $\overline{SRAS}$, $\overline{SMS}$, SCKE. These balls are not tolerant to voltages higher than V_{DDMEM}.

[4] The V_{DDOTP} voltage for writes must only be applied when programming OTP memory. There is a finite amount of cumulative time that this voltage may be applied (dependent on voltage and junction temperature) over the lifetime of the part. Please see Table 19 on Page 32 for details.

[5] When not using the USB peripheral on the ADSP-BF524/BF526 or terminating V_{DDUSB} on the ADSP-BF522, V_{DDUSB} must be powered by V_{DDEXT}.

[6] Bidirectional balls (PF15–0, PG15–0, PH15–0) and input balls (RTXI, TCK, TDI, TMS, $\overline{TRST}$, CLKIN, $\overline{RESET}$, $\overline{NMI}$, and BMODE3–0) of the ADSP-BF522/523/524/525/526/527 processors are 3.3 V tolerant (always accept up to 3.6 V maximum V_{IH}). Voltage compliance (on outputs, V_{OH}) is limited by the V_{DDEXT} supply voltage.

[7] Parameter value applies to all input and bidirectional balls, except USB_DP, USB_DM, USB_VBUS, SDA, and SCL.

[8] The V_{IHTWI} min and max value vary with the selection in the TWI_DT field of the NONGPIO_DRIVE register. See V_{BUSTWI} min and max values in Table 11.

[9] SDA and SCL are pulled up to V_{BUSTWI}. See Table 11.

Table 11 shows settings for TWI_DT in the NONGPIO_DRIVE register. Set this register prior to using the TWI port.

Table 11. TWI_DT Field Selections and V_{DDEXT}/V_{BUSTWI}

TWI_DT	V_{DDEXT} Nominal	V_{BUSTWI} Minimum	V_{BUSTWI} Nominal	V_{BUSTWI} Maximum	Unit
000 (default)	3.3	2.97	3.3	3.63	V
001	1.8	1.27	1.8	2.35	V
010	2.5	2.97	3.3	3.63	V
011	1.8	2.97	3.3	3.63	V
100	3.3	4.5	5	5.5	V
101	1.8	2.25	2.5	2.75	V
110	2.5	2.25	2.5	2.75	V
111 (reserved)	–	–	–	–	–

ADSP-BF522/524/526 Clock Related Operating Conditions

Table 12 describes the core clock timing requirements for the ADSP-BF522/524/526 processors. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock and system clock (see Table 14). Table 13 describes phase-locked loop operating conditions.

Table 12. Core Clock (CCLK) Requirements—ADSP-BF522/524/526 Processors—All Speed Grades[1]

Parameter		Max	Unit
f_{CCLK}	Core Clock Frequency (V_{DDINT} =tbd[2] V minimum)	400[3]	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} =tbd[4] V minimum)	350	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = tbd[5] V minimum)	300	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = tbd V minimum)	TBD	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = tbd V minimum)	TBD	MHz

[1] See the Ordering Guide on Page 72.
[2] Preliminary data indicates a value of 1.33 V.
[3] Applies only to 400 MHz speed grade only. See the Ordering Guide on Page 72.
[4] Preliminary data indicates a value of 1.235 V.
[5] Preliminary data indicates a value of 1.14 V.

Table 13. Phase-Locked Loop Operating Conditions

Parameter		Minimum	Maximum	Unit
f_{VCO}	Voltage Controlled Oscillator (VCO) Frequency	50	Speed Grade[1]	MHz

[1] See the Ordering Guide on Page 72.

Table 14. ADSP-BF522/524/526 Processors Maximum SCLK Conditions

Parameter		V_{DDEXT}/V_{DDMEM} = 1.8 V/2.5 V/3.3 V Nominal	Unit
f_{SCLK}	CLKOUT/SCLK Frequency ($V_{DDINT} \geq$ tbd V)[1]	80	MHz
f_{SCLK}	CLKOUT/SCLK Frequency (V_{DDINT} < tbd V)	tbd	MHz

[1] f_{SCLK} must be less than or equal to f_{CCLK} and is subject to additional restrictions for SDRAM interface operation. See Table 25 on Page 39.

OPERATING CONDITIONS FOR ADSP-BF523/525/527

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DDINT}	Internal Supply Voltage[1]		0.95		1.26	V
V_{DDEXT}	External Supply Voltage[2, 3]	Internal Voltage Regulator Disabled	1.70	1.8, 2.5 or 3.3	3.6	V
V_{DDEXT}	External Supply Voltage[2, 3]	Internal Voltage Regulator Enabled	2.25	2.5 or 3.3	3.6	V
V_{DDRTC}	RTC Power Supply Voltage[4]		2.25		3.6	V
V_{DDMEM}	MEM Supply Voltage[2, 5]		1.70	1.8, 2.5 or 3.3	3.6	V
V_{DDOTP}	OTP Supply Voltage[2]		2.25	2.5	2.75	V
V_{PPOTP}	OTP Programming Voltage[2]		2.25	2.5	2.75	V
V_{DDUSB}	USB Supply Voltage[6]		3.0	3.3	3.6	V
V_{IH}	High Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 1.90 V	1.1		3.6	V
V_{IH}	High Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 2.75 V	1.7		3.6	V
V_{IH}	High Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 3.6 V	2.0		3.6	V
V_{IHTWI}	High Level Input Voltage	V_{DDEXT} = 1.90 V/2.75 V/3.6 V	0.7 x V_{BUSTWI}		V_{BUSTWI}[9]	V
V_{IL}	Low Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 1.7 V	–0.3		0.6	V
V_{IL}	Low Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 2.25 V	–0.3		0.7	V
V_{IL}	Low Level Input Voltage[7, 8]	V_{DDEXT}/V_{DDMEM} = 3.0 V	–0.3		0.8	V
V_{ILTWI}	Low Level Input Voltage	V_{DDEXT} = minimum	–0.3		0.3 x V_{BUSTWI}[10]	V
T_J	Junction Temperature	289-Ball CSP_BGA @ $T_{AMBIENT}$ = 0°C to +70°C	0		+105	°C
T_J	Junction Temperature	208-Ball CSP_BGA @ $T_{AMBIENT}$ = 0°C to +70°C	0		+105	°C
T_J	Junction Temperature	208-Ball CSP_BGA @ $T_{AMBIENT}$ = –40°C to +85°C	–40		+105	°C

[1] The voltage regulator can generate V_{DDINT} at levels of tbd V to tbd V with tbd% to +tbd% tolerance.
[2] Must remain powered (even if the associated function is not used).
[3] V_{DDEXT} is the supply to the voltage regulator and GPIO.
[4] If not used, power with V_{DDEXT}.
[5] Balls that use V_{DDMEM} are DATA15–0, ADDR19–1, $\overline{ABE1–0}$, $\overline{ARE}$, $\overline{AWE}$, $\overline{AOE}$, $\overline{AMS3–0}$, ARDY, SA10, $\overline{SWE}$, $\overline{SCAS}$, CLKOUT, $\overline{SRAS}$, $\overline{SMS}$, SCKE. These balls are not tolerant to voltages higher than V_{DDMEM}.
[6] When not using the USB peripheral on the ADSP-BF525/BF527 or terminating V_{DDUSB} on the ADSP-BF523, V_{DDUSB} must be powered by V_{DDEXT}.
[7] Bidirectional balls (PF15–0, PG15–0, PH15–0) and input balls (RTXI, TCK, TDI, TMS, $\overline{TRST}$, CLKIN, $\overline{RESET}$, $\overline{NMI}$, and BMODE3–0) of the ADSP-BF522/523/524/525/526/527 processors are 3.3 V tolerant (always accept up to 3.6 V maximum V_{IH}). Voltage compliance (on outputs, V_{OH}) is limited by the V_{DDEXT} supply voltage.
[8] Parameter value applies to all input and bidirectional balls, except USB_DP, USB_DM, USB_VBUS, SDA, and SCL.
[9] The V_{IHTWI} min and max value vary with the selection in the TWI_DT field of the NONGPIO_DRIVE register. See V_{BUSTWI} min and max values in Table 11 on Page 27.
[10] SDA and SCL are pulled up to V_{BUSTWI}. See Table 11 on Page 27.

ADSP-BF523/525/527 Clock Related Operating Conditions

Table 15 describes the core clock timing requirements for the ADSP-BF523/525/527 processors. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock and system clock (see Table 17). Table 16 describes phase-locked loop operating conditions.

Table 15. Core Clock (CCLK) Requirements—ADSP-BF523/525/527 Processors—All Speed Grades[1]

Parameter		Internal Regulator Setting	Max	Unit
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 1.14 V minimum)[2]	1.20 V	600	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 1.093 V minimum)[3]	1.15 V	533	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.95 V minimum)	1.0 V	400	MHz

[1] See the Ordering Guide on Page 72.
[2] Applies only to 600 MHz speed grades. See the Ordering Guide on Page 72.
[3] Applies only to 533 MHz and 600 MHz speed grades. See the Ordering Guide on Page 72.

Table 16. Phase-Locked Loop Operating Conditions

Parameter		Minimum	Maximum	Unit
f_{VCO}	Voltage Controlled Oscillator (VCO) Frequency	50	Speed Grade[1]	MHz

[1] See the Ordering Guide on Page 72.

Table 17. ADSP-BF523/525/527 Processors Maximum SCLK Conditions

Parameter		V_{DDEXT}/V_{DDMEM} = 1.8 V Nominal[1]	V_{DDEXT}/V_{DDMEM} = 2.5 V/3.3 V Nominal	Unit
f_{SCLK}	CLKOUT/SCLK Frequency ($V_{DDINT} \geq 1.14$ V)[2]	100	133[3]	MHz
f_{SCLK}	CLKOUT/SCLK Frequency ($V_{DDINT} < 1.14$ V)[2]	100	100	MHz

[1] If either V_{DDEXT} or V_{DDMEM} are operating at 1.8V nominal, f_{SCLK} is constrained to 100MHz.
[2] f_{SCLK} must be less than or equal to f_{CCLK} and is subject to additional restrictions for SDRAM interface operation. See Table 25 on Page 39.
[3] Rounded number. Actual test specification is SCLK period of 7.5 ns. See Table 25 on Page 39.

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	Min	Typical	Max	Unit
V_{OH}	High Level Output Voltage	V_{DDEXT}/V_{DDMEM} = 1.7 V, I_{OH} = –0.5 mA	1.35			V
V_{OH}	High Level Output Voltage	V_{DDEXT}/V_{DDMEM} = 2.25 V, I_{OH} = –0.5 mA	2.0			V
V_{OH}	High Level Output Voltage	V_{DDEXT}/V_{DDMEM} = 3.0 V, I_{OH} = –0.5 mA	2.4			V
V_{OL}	Low Level Output Voltage	V_{DDEXT}/V_{DDMEM} = 1.7 V/2.25 V/3.0 V, I_{OL} = 2.0 mA			0.4	V
V_{OLTWI}	Low Level Output Voltage	V_{DDEXT}/V_{DDMEM} = 1.7 V/2.25 V/3.0 V, I_{OL} = 2.0 mA			TBD	V V
I_{IH}	High Level Input Current[1]	V_{DDEXT}/V_{DDMEM} = 3.6 V, V_{IN} = 3.6 V			10.0	µA
I_{IL}	Low Level Input Current[1]	V_{DDEXT}/V_{DDMEM} = 3.6 V, V_{IN} = 0 V			10.0	µA
I_{IHP}	High Level Input Current JTAG[2]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			50.0	µA
I_{OZH}	Three-State Leakage Current[3]	V_{DDEXT}/V_{DDMEM} = 3.6 V, V_{IN} = 3.6 V			10.0	µA
I_{OZHTWI}	Three-State Leakage Current[4]	V_{DDEXT} = 3.0 V, V_{IN} = 5.5 V			10.0	µA
I_{OZL}	Three-State Leakage Current[3]	V_{DDEXT}/V_{DDMEM} = 3.6 V, V_{IN} = 0 V			10.0	µA
C_{IN}	Input Capacitance[5]	f_{IN} = 1 MHz, $T_{AMBIENT}$ = 25°C, V_{IN} = 2.5 V		TBD	TBD[6]	pF
$I_{DDHIBERNATE}$	Total Current for All Domains in Hibernate State	V_{DDEXT}=V_{DDMEM}=V_{DDRTC}=V_{DDUSB} = 3.3 V, V_{DDOTP}=V_{PPOTP}= 2.5 V, V_{DDINT} = 0 V, CLKIN=0 MHz, @T_J = 25°C			TBD	µA
I_{DDRTC}	Total Current for V_{DDRTC} in Hibernate State	V_{DDRTC} = 3.3 V, @T_J = 25°C			TBD	µA

[1] Applies to input balls.
[2] Applies to JTAG input balls (TCK, TDI, TMS, $\overline{TRST}$).
[3] Applies to three-statable balls.
[4] Applies to bidirectional balls SCL and SDA.
[5] Applies to all signal balls.
[6] Guaranteed, but not tested.

ABSOLUTE MAXIMUM RATINGS

Stresses greater than those listed in the table may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions greater than those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Parameter	Rating
Internal Supply Voltage (V_{DDINT})	tbd V to +tbd V
External (I/O) Supply Voltage (V_{DDEXT}/V_{DDMEM})	–0.3 V to +3.8 V
Input Voltage[1, 2]	–0.5 V to +3.6 V
Input Voltage[1, 2, 3]	–0.5 V to +5.5 V
Input Voltage[1, 2, 4]	–0.5 V to +5.25 V
Output Voltage Swing	–0.5 V to V_{DDEXT}/V_{DDMEM}+0.5 V
Load Capacitance[5]	200 pF
Storage Temperature Range	–65°C to +150°C
Junction Temperature Underbias	+110°C

[1] Applies to 100% transient duty cycle. For other duty cycles see Table 18.
[2] Applies only when V_{DDEXT} is within specifications. When V_{DDEXT} is outside specifications, the range is V_{DDEXT} ± 0.2 Volts.
[3] Applies to balls SCL and SDA.
[4] Applies to balls USB_DP, USB_DM, and USB_VBUS.
[5] For proper SDRAM controller operation, the maximum load capacitance is 50 pF (at 3.3 V) or 30 pF (at 2.5 V) for ADDR19–1, DATA15–0, ABE1–0/SDQM1–0, CLKOUT, SCKE, SA10, SRAS, SCAS, SWE, and SMS.

Table 18. Maximum Duty Cycle for Input Transient Voltage[1]

V_{IN} Min (V)	V_{IN} Max (V)	Maximum Duty Cycle
TBD	TBD	100 %
TBD	TBD	40%
TBD	TBD	25%
TBD	TBD	15%
TBD	TBD	10%

[1] Applies to all signal balls with the exception of CLKIN, XTAL, VR_{OUT}/EXT_WAKE1.

When programming OTP memory on the ADSP-BF522/524/526 processors, the VPPOTP ball must be set to the write value specified in the Operating Conditions for ADSP-BF522/524/526 on Page 27. There is a finite amount of cumulative time that the write voltage may be applied (dependent on voltage and junction temperature) to VPPOTP over the lifetime of the part. Therefore, maximum OTP memory programming time for the ADSP-BF522/524/526 processors is shown in Table 19. The ADSP-BF523/525/527 processors do not have a similar restriction.

Table 19. Maximum OTP Memory Programming Time for ADSP-BF522/524/526 Processors

	Temperature (T_J)			
VPPOTP Voltage (V)	25°C	85°C	110°C	125°C
6.9	tbd sec	tbd sec	tbd sec	tbd sec
7.0	2400 sec	tbd sec	tbd sec	tbd sec
7.1	1000 sec	tbd sec	tbd sec	tbd sec

ESD SENSITIVITY

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PACKAGE INFORMATION

The information presented in Figure 8 and Table 20 provides details about the package branding for the ADSP-BF522/524/526 and ADSP-BF523/525/527 processors. For a complete listing of product availability, see Ordering Guide on Page 72.

Figure 8. Product Information on Package

Table 20. Package Brand Information

Brand Key	Field Description
ADSP-BF52x	Product Name[1]
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Designation
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
yyww	Date Code

[1] See product names in the Ordering Guide on Page 72.

TIMING SPECIFICATIONS

Clock and Reset Timing

Table 21 and Figure 9 describe clock and reset operations. Per the CCLK and SCLK timing specifications in Table 12 to Table 17, combinations of CLKIN and clock multipliers must not select core/peripheral clocks in excess of the processor's speed grade.

Table 21. Clock and Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{CKIN}	CLKIN Period	20.0	100.0	ns
t_{CKINL}	CLKIN Low Pulse[1]	10.0		ns
t_{CKINH}	CLKIN High Pulse[1]	10.0		ns
$t_{BUFDLAY}$	CLKIN to CLKBUF Delay		10	ns
t_{WRST}	$\overline{\text{RESET}}$ Asserted Pulse Width Low[2]	11 t_{CKIN}		ns

[1] Applies to bypass mode and non-bypass mode.
[2] Applies after power-up sequence is complete. At power-up, the processor's internal phase-locked loop requires no more than 2000 CLKIN cycles, while $\overline{\text{RESET}}$ is asserted, assuming stable power supplies and CLKIN (not including start-up time of external clock oscillator).

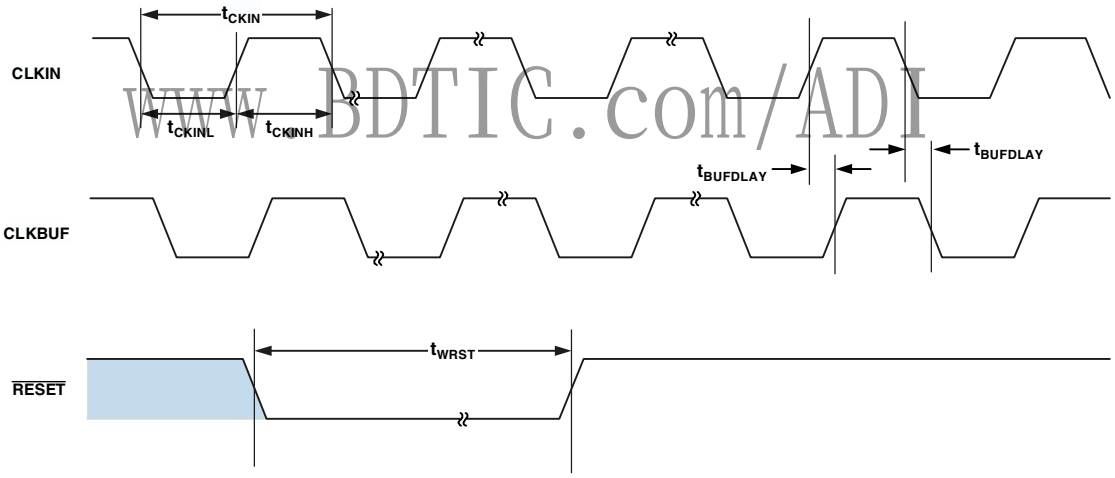


Figure 9. Clock and Reset Timing

Asynchronous Memory Read Cycle Timing

Table 22. Asynchronous Memory Read Cycle Timing

Parameter		V_{DDMEM} = 1.8 V Min	V_{DDMEM} = 1.8 V Max	V_{DDMEM} = 2.5/3.3 V Min	V_{DDMEM} = 2.5/3.3 V Max	Unit
Timing Requirements						
t_{SDAT}	DATA15–0 Setup Before CLKOUT	2.1		2.1		ns
t_{HDAT}	DATA15–0 Hold After CLKOUT	0.8		0.8		ns
t_{SARDY}	ARDY Setup Before CLKOUT	4.0		4.0		ns
t_{HARDY}	ARDY Hold After CLKOUT	0.0		0.0		ns
Switching Characteristics						
t_{DO}	Output Delay After CLKOUT[1]		6.0		6.0	ns
t_{HO}	Output Hold After CLKOUT [1]	0.8		0.8		ns

[1] Output balls include $\overline{AMS3-0}$, $\overline{ABE1-0}$, ADDR19–1, $\overline{AOE}$, $\overline{ARE}$.

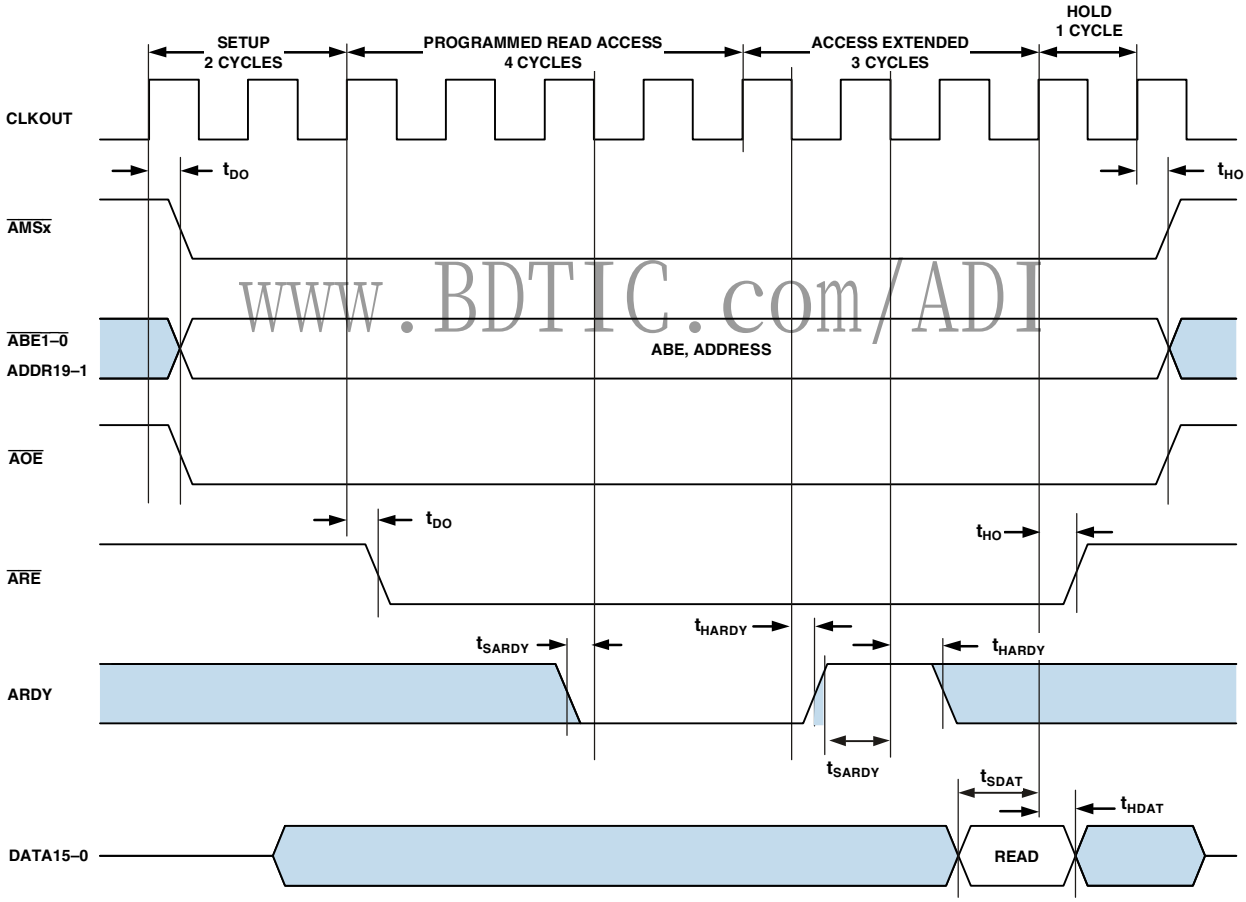


Figure 10. Asynchronous Memory Read Cycle Timing

Asynchronous Memory Write Cycle Timing

Table 23. Asynchronous Memory Write Cycle Timing

Parameter		V_{DDMEM} = 1.8 V Min	V_{DDMEM} = 1.8 V Max	V_{DDMEM} = 2.5/3.3 V Min	V_{DDMEM} = 2.5/3.3 V Max	Unit
Timing Requirements						
t_{SARDY}	ARDY Setup Before CLKOUT	4.0		4.0		ns
t_{HARDY}	ARDY Hold After CLKOUT	0.0		0.0		ns
Switching Characteristics						
t_{DDAT}	DATA15–0 Disable After CLKOUT		6.0		6.0	ns
t_{ENDAT}	DATA15–0 Enable After CLKOUT	0.0		0.0		ns
t_{DO}	Output Delay After CLKOUT[1]		6.0		6.0	ns
t_{HO}	Output Hold After CLKOUT [1]	0.8		0.8		ns

[1] Output balls include $\overline{AMS3-0}$, $\overline{ABE1-0}$, ADDR19–1, DATA15–0, $\overline{AOE}$, $\overline{AWE}$.

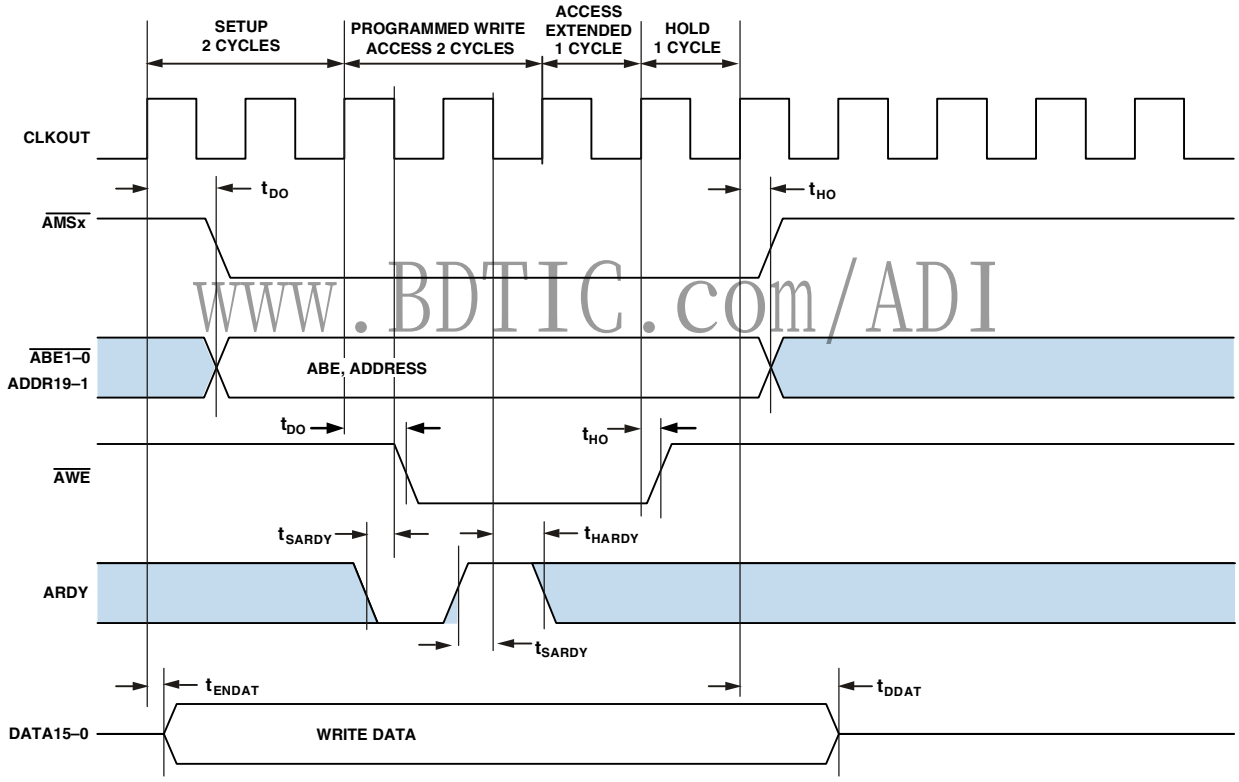


Figure 11. Asynchronous Memory Write Cycle Timing

NAND Flash Controller Interface Timing

Table 24 and Figure 12 on Page 36 through Figure 16 on Page 38 describe NAND Flash Controller Interface operations.

Table 24. NAND Flash Controller Interface Timing

Parameter		Min	Max	Unit
Write Cycle				
Switching Characteristics				
t_{CWL}	$\overline{ND\_CE}$ Setup Time to $\overline{AWE}$ Low	$1.0 \times t_{SCLK} - 4$		ns
t_{CH}	$\overline{ND\_CE}$ Hold Time From $\overline{AWE}$ High	$3.0 \times t_{SCLK} - 4$		ns
t_{CLHWL}	ND_CLE Setup Time High to $\overline{AWE}$ Low	0.0		ns
t_{CLH}	ND_CLE Hold Time From $\overline{AWE}$ high	$2.5 \times t_{SCLK} - 4$		ns
t_{ALLWL}	ND_ALE Setup Time Low to $\overline{AWE}$ Low	0.0		ns
t_{ALH}	ND_ALE Hold Time From $\overline{AWE}$ High	$2.5 \times t_{SCLK} - 4$		ns
t_{WP}[1]	$\overline{AWE}$ Low to $\overline{AWE}$ high	$(WR\_DLY + 1.0) \times t_{SCLK} - 4$		ns
t_{WHWL}	$\overline{AWE}$ High to $\overline{AWE}$ Low	$4.0 \times t_{SCLK} - 4$		ns
t_{WC}[1]	$\overline{AWE}$ Low to $\overline{AWE}$ Low	$(WR\_DLY + 5.0) \times t_{SCLK} - 4$		ns
t_{DWS}[1]	Data Setup Time for a Write Access	$(WR\_DLY + 1.5) \times t_{SCLK} - 4$		ns
t_{DWH}	Data Hold Time for a Write Access	$2.5 \times t_{SCLK} - 4$		ns
Read Cycle				
Switching Characteristics				
t_{CRL}	$\overline{ND\_CE}$ Setup Time to $\overline{ARE}$ Low	$1.0 \times t_{SCLK} - 4$		ns
t_{CRH}	$\overline{ND\_CE}$ Hold Time From $\overline{ARE}$ High	$3.0 \times t_{SCLK} - 4$		ns
t_{RP}[1]	$\overline{ARE}$ Low to $\overline{ARE}$ High	$(RD\_DLY + 1.0) \times t_{SCLK} - 4$		ns
t_{RHRL}	$\overline{ARE}$ High to $\overline{ARE}$ Low	$4.0 \times t_{SCLK} - 4$		ns
t_{RC}[1]	$\overline{ARE}$ Low to $\overline{ARE}$ Low	$(RD\_DLY + 5.0) \times t_{SCLK} - 4$		ns
Timing Requirements				
t_{DRS}	Data Setup Time for a Read Transaction	8.0[2]		ns
t_{DRH}	Data Hold Time for a Read Transaction	0.0		ns
Write Followed by Read				
Switching Characteristics				
t_{WHRL}	$\overline{AWE}$ High to $\overline{ARE}$ Low	$5.0 \times t_{SCLK} - 4$		ns

[1] WR_DLY and RD_DLY are defined in the NFC_CTL register.
[2] The only parameter that differs from 1.8V to 2.5/3.3V operation is t_{DRS}, which is 8.0ns at 2.5/3.3V and is 11ns at 1.8V.

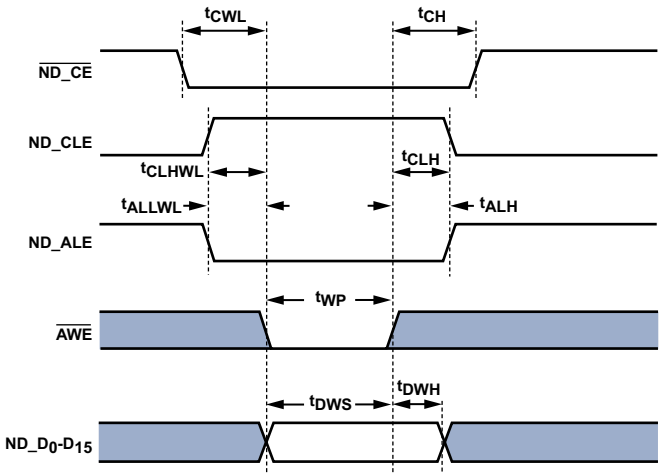


Figure 12. NAND Flash Controller Interface Timing - Command Write Cycle

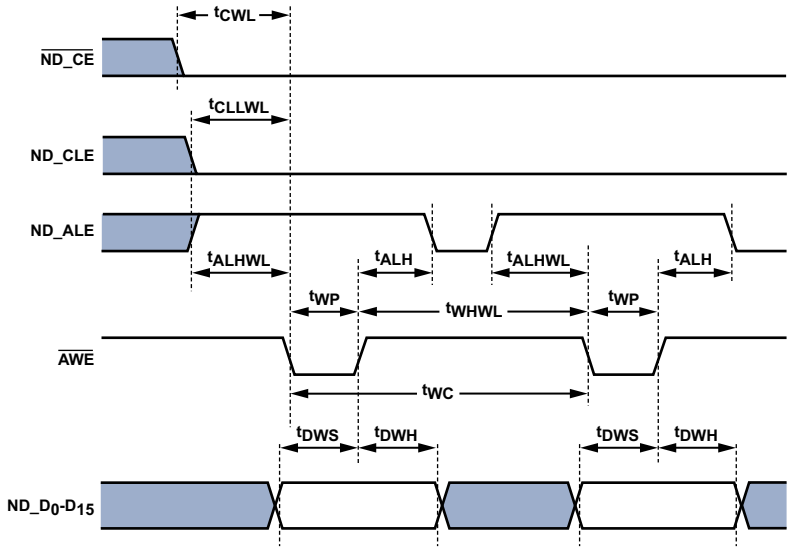


Figure 13. NAND Flash Controller Interface Timing - Address Write Cycle

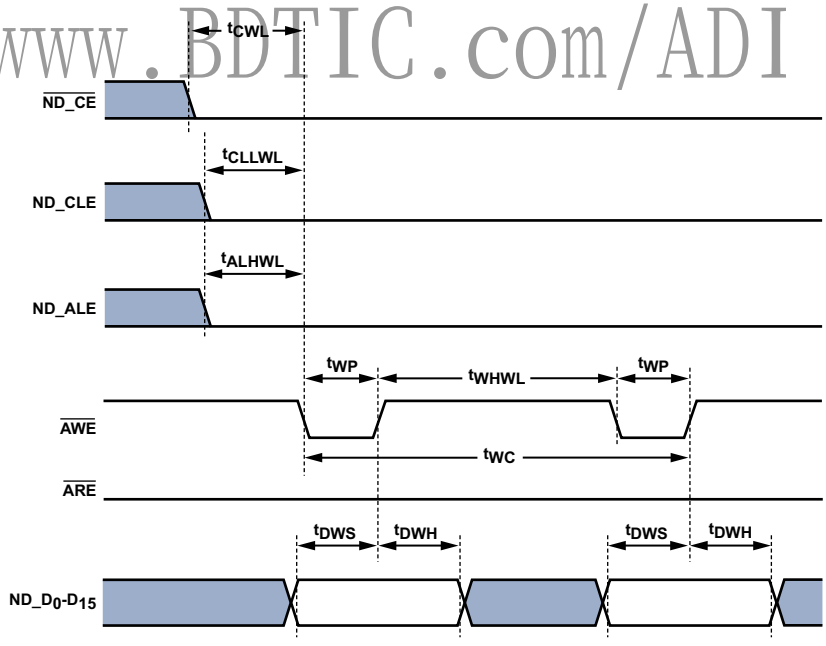


Figure 14. NAND Flash Controller Interface Timing - Data Write Operation

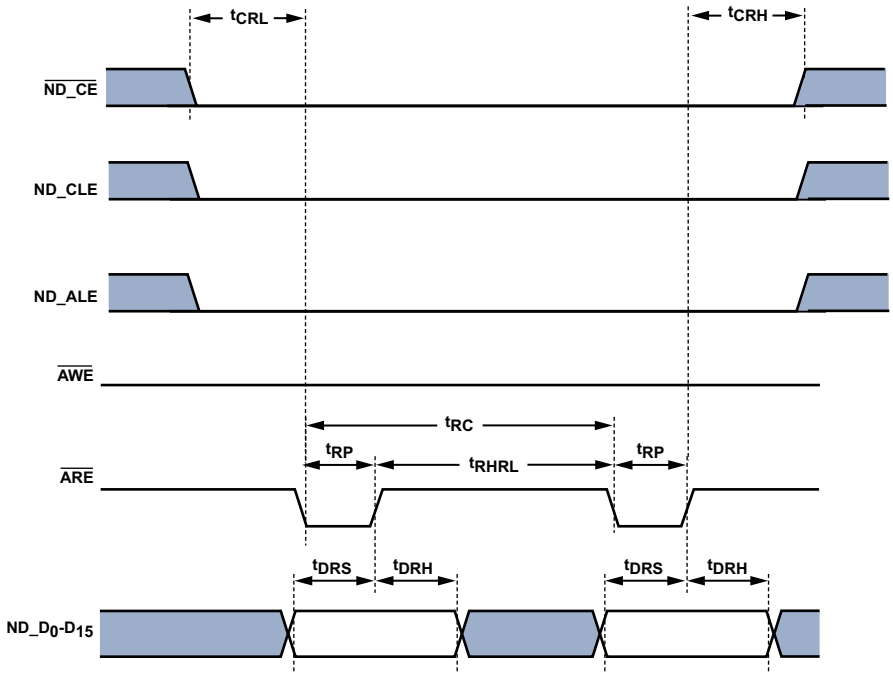


Figure 15. NAND Flash Controller Interface Timing - Data Read Operation

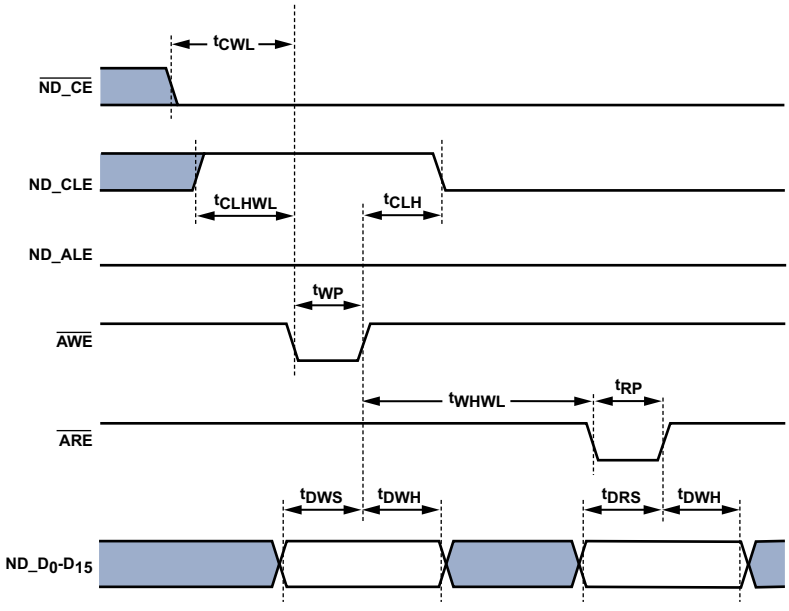


Figure 16. NAND Flash Controller Interface Timing - Write Followed by Read Operation

SDRAM Interface Timing

Table 25. SDRAM Interface Timing

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDMEM} = 1.8 V		V_{DDMEM} = 2.5/3.3 V		V_{DDMEM} = 1.8 V		V_{DDMEM} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Timing Requirements										
t_{SSDAT}	Data Setup Before CLKOUT	1.5		1.5		1.5		1.5		ns
t_{HSDAT}	Data Hold After CLKOUT	0.8		0.8		0.8		0.8		ns
Switching Characteristics										
t_{SCLK}	CLKOUT Period[1]	12.5		12.5		7.5		7.5		ns
t_{SCLKH}	CLKOUT Width High	2.5		2.5		2.5		2.5		ns
t_{SCLKL}	CLKOUT Width Low	2.5		2.5		2.5		2.5		ns
t_{DCAD}	Command, Address, Data Delay After CLKOUT[2]		4.4		4.4		4.0		4.0	ns
t_{HCAD}	Command, Address, Data Hold After CLKOUT[2]	1.0		1.0		1.0		1.0		ns
t_{DSDAT}	Data Disable After CLKOUT		5.0		5.0		5.0		5.0	ns
t_{ENSDAT}	Data Enable After CLKOUT	0.0		0.0		0.0		0.0		ns

[1] The t_{SCLK} value is the inverse of the f_{SCLK} specification discussed in Table 14 and Table 17. Package type and reduced supply voltages affect the best-case values listed here.
[2] Command balls include: $\overline{SRAS}$, $\overline{SCAS}$, $\overline{SWE}$, SDQM, $\overline{SMS}$, SA10, SCKE.

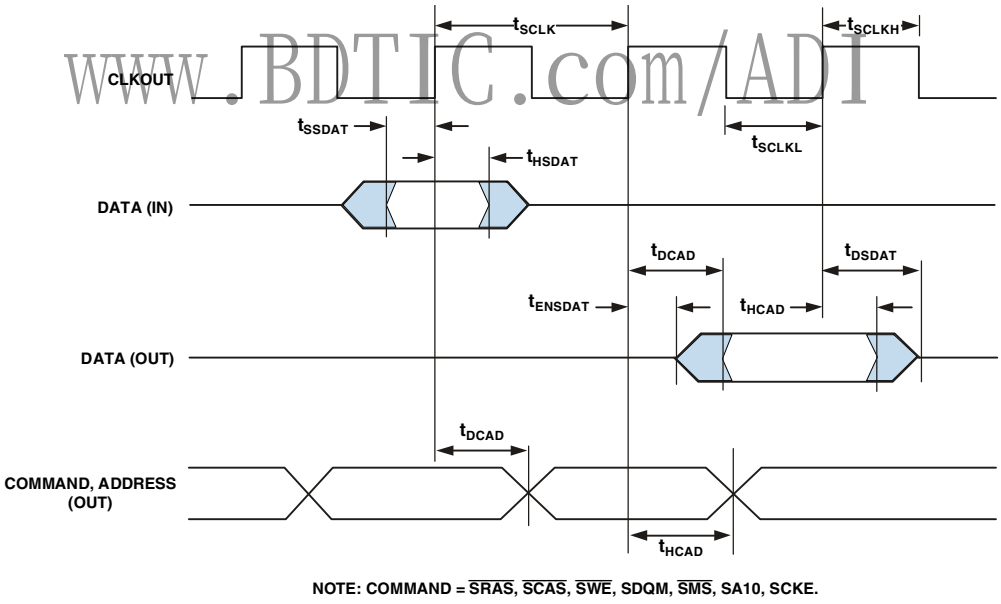


Figure 17. SDRAM Interface Timing

External DMA Request Timing

Table 26 and Figure 18 describe the External DMA Request operations.

Table 26. External DMA Request Timing

Parameter		V_{DDEXT}/V_{DDMEM} = 1.8 V[1] Min	Max	V_{DDEXT}/V_{DDMEM} = 2.5/3.3 V Min	Max	Unit
Timing Parameters						
t_{DR}	DMARx Asserted to CLKOUT High Setup	6.0		6.0		ns
t_{DH}	CLKOUT High to DMARx Deasserted Hold Time	0.0		0.0		ns
$t_{DMARACT}$	DMARx Active Pulse Width	$1.0 \times t_{SCLK}$		$1.0 \times t_{SCLK}$		ns
$t_{DMARINACT}$	DMARx Inactive Pulse Width	$1.75 \times t_{SCLK}$		$1.75 \times t_{SCLK}$		ns

[1] Because the external DMA control pins are part of the V_{DDEXT} power domain and the CLKOUT signal is part of the V_{DDMEM} power domain, systems in which V_{DDEXT} and V_{DDMEM} are NOT equal may require level shifting logic for correct operation.

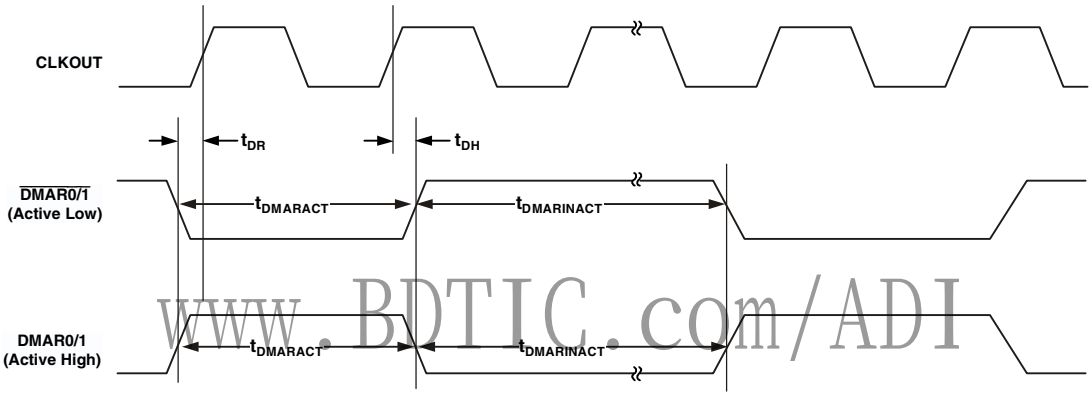


Figure 18. External DMA Request Timing

Parallel Peripheral Interface Timing

Table 27 and Figure 19 on Page 41, Figure 23 on Page 46, and Figure 24 on Page 46 describe parallel peripheral interface operations.

Table 27. Parallel Peripheral Interface Timing

Parameter		ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Min	ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Max	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Requirements										
t_{PCLKW}	PPI_CLK Width[1]	6.4		6.4		6.0		6.0		ns
t_{PCLK}	PPI_CLK Period[1]	16.0		16.0		15.0		15.0		ns
Timing Requirements - GP Input and Frame Capture Modes										
t_{SFSPE}	External Frame Sync Setup Before PPI_CLK (Nonsampling Edge for Rx, Sampling Edge for Tx)	6.7		6.7		6.7		6.7		ns
t_{HFSPE}	External Frame Sync Hold After PPI_CLK	1.0		1.0		1.0		1.0		ns
t_{SDRPE}	Receive Data Setup Before PPI_CLK	3.5		3.5		3.5		3.5		ns
t_{HDRPE}	Receive Data Hold After PPI_CLK	1.5		1.5		1.5		1.5		ns
Switching Characteristics - GP Output and Frame Capture Modes										
t_{DFSPE}	Internal Frame Sync Delay After PPI_CLK		8.8		8.8		8.0		8.0	ns
t_{HOFSPE}	Internal Frame Sync Hold After PPI_CLK	1.7		1.7		1.7		1.7		ns
t_{DDTPE}	Transmit Data Delay After PPI_CLK		8.8		8.8		8.0		8.0	ns
t_{HDTPE}	Transmit Data Hold After PPI_CLK	1.8		1.8		1.8		1.8		ns

[1] PPI_CLK frequency cannot exceed $f_{SCLK}/2$

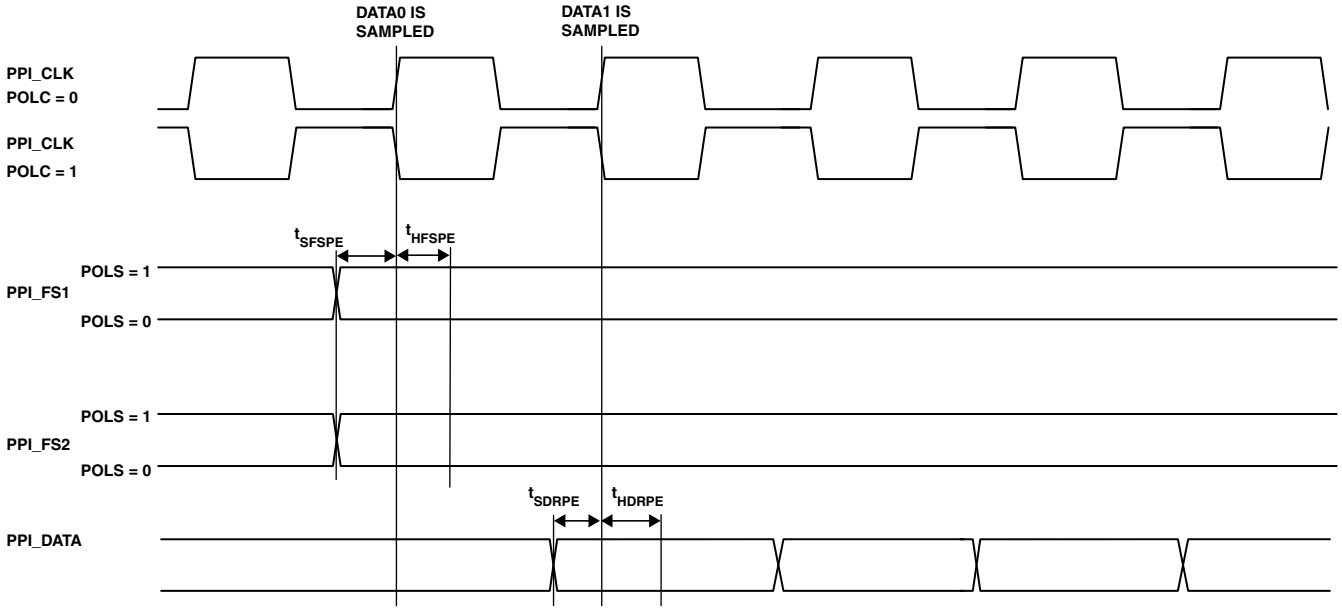


Figure 19. PPI GP Rx Mode with External Frame Sync Timing

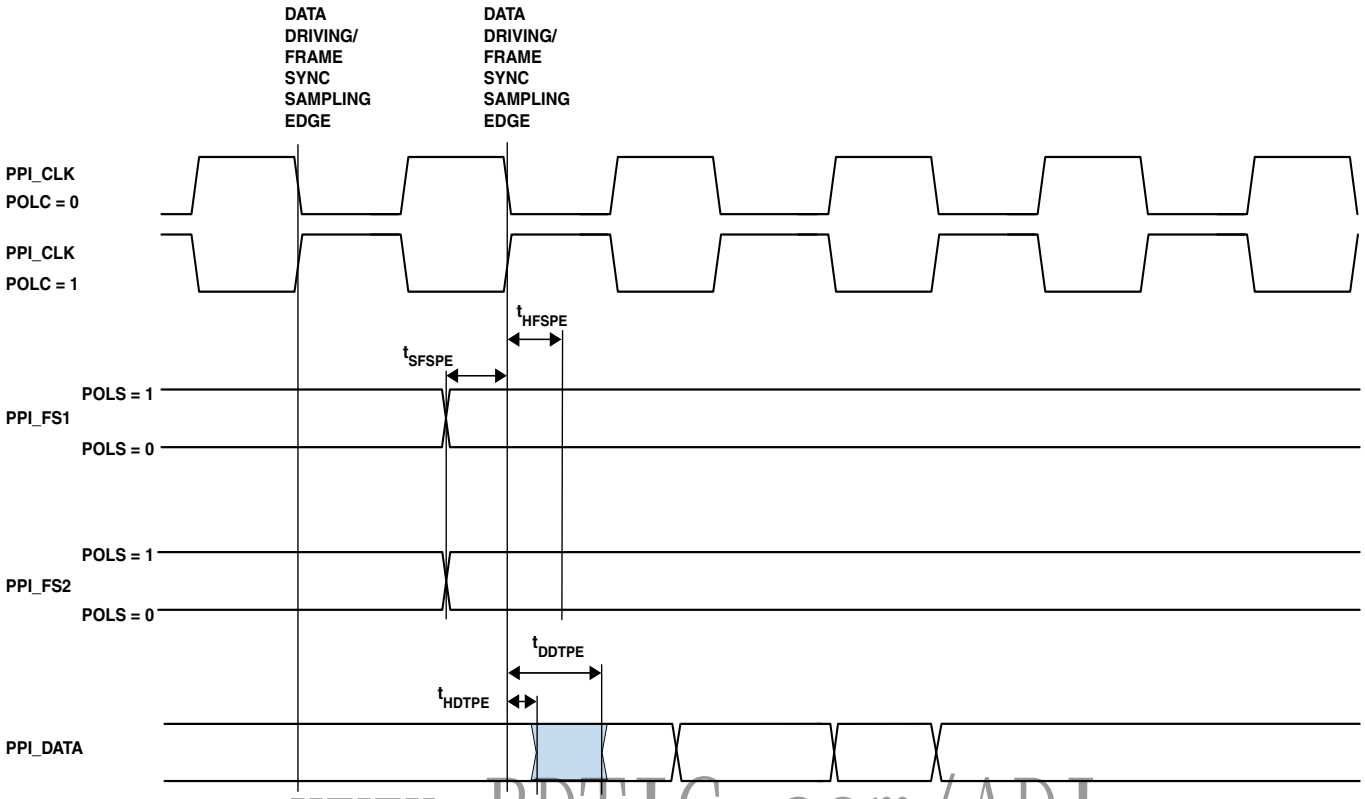


Figure 20. PPI GP Tx Mode with External Frame Sync Timing

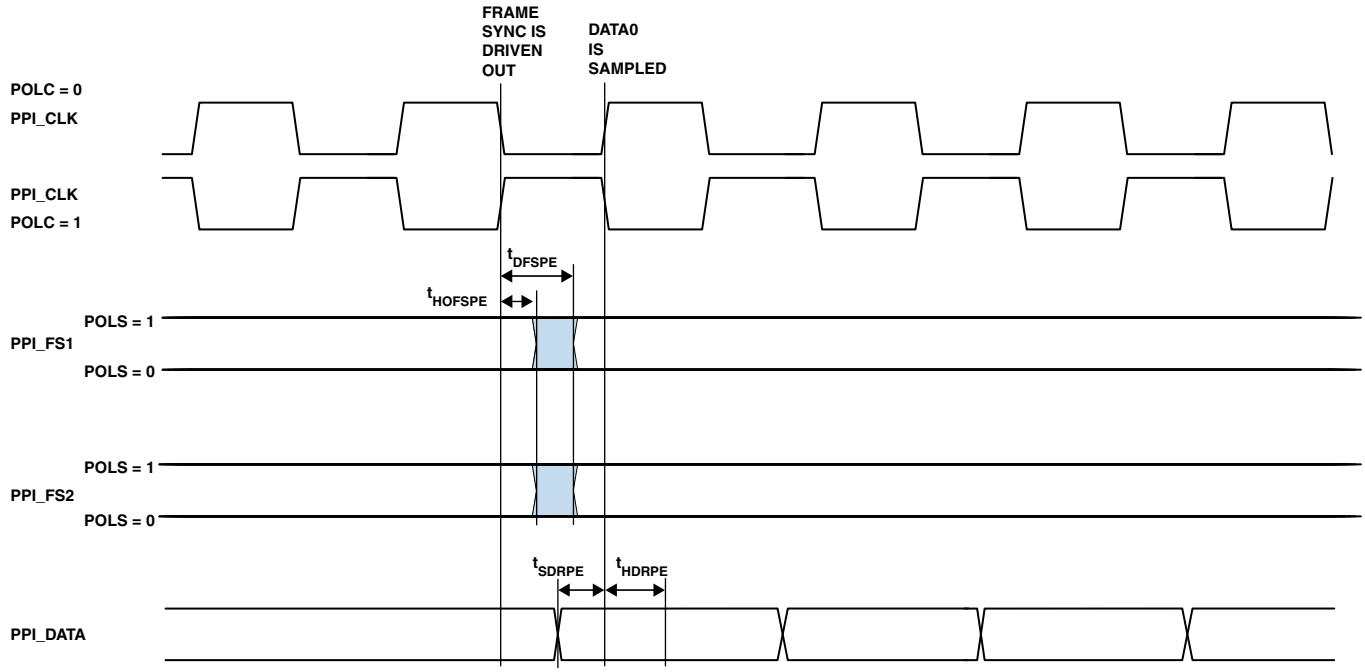


Figure 21. PPI GP Rx Mode with Internal Frame Sync Timing

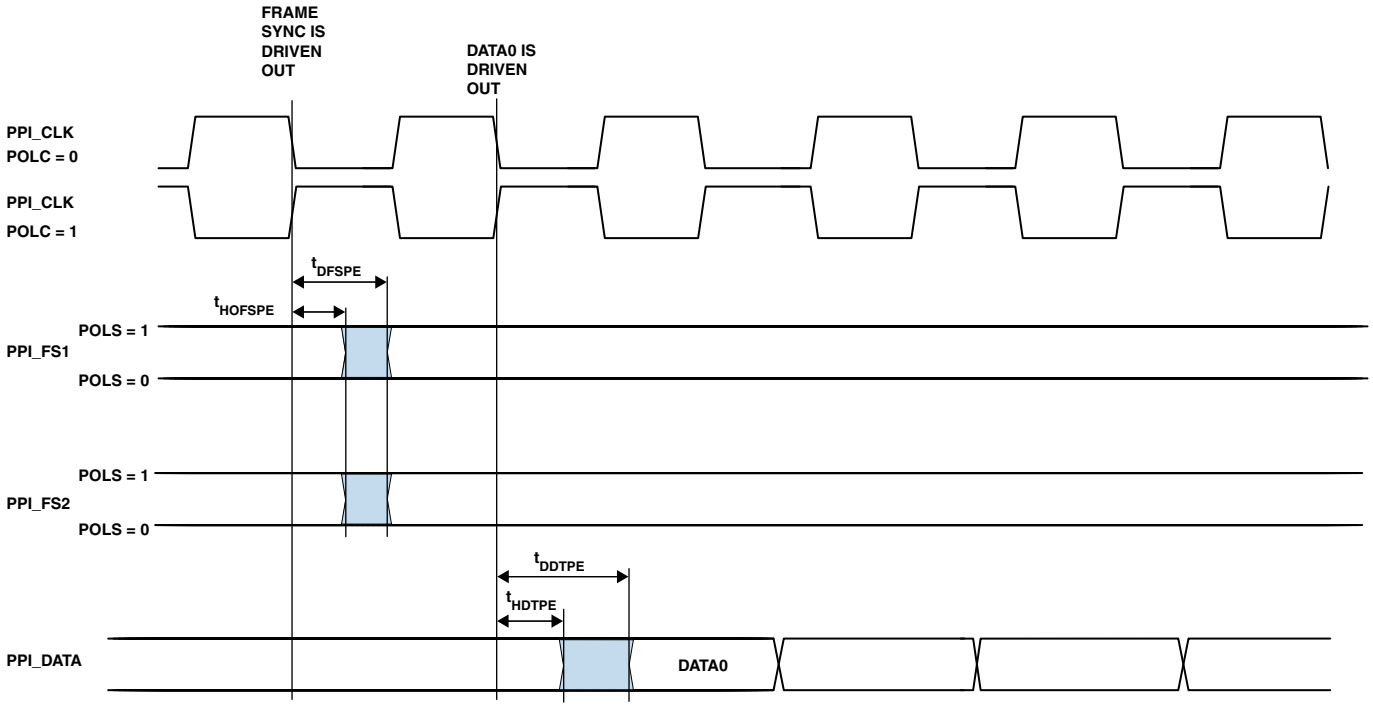


Figure 22. PPI GP Tx Mode with Internal Frame Sync Timing

Serial Ports

Table 28 through Table 31 on Page 45 and Figure 23 on Page 46 through Figure 24 on Page 46 describe serial port operations.

Table 28. Serial Ports—External Clock

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Timing Requirements										
t_{SFSE}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	3.0		3.0		3.0		3.0		ns
t_{HFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	3.0		3.0		3.0		3.0		ns
t_{SDRE}	Receive Data Setup Before RSCLKx[1]	3.0		3.0		3.0		3.0		ns
t_{HDRE}	Receive Data Hold After RSCLKx[1]	3.6		3.6		3.0		3.0		ns
t_{SCLKEW}	TSCLKx/RSCLKx Width	5.4		5.4		4.5		4.5		ns
t_{SCLKE}	TSCLKx/RSCLKx Period	18.0		18.0		15.0		15.0		ns
Switching Characteristics										
t_{DFSE}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]		12.0		12.0		10.0		10.0	ns
t_{HOFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[1]	0.0		0.0		0.0		0.0		ns
t_{DDTE}	Transmit Data Delay After TSCLKx[1]		12.0		12.0		10.0		10.0	ns
t_{HDTE}	Transmit Data Hold After TSCLKx[1]	0.0		0.0		0.0		0.0		ns

[1] Referenced to sample edge.
[2] Referenced to drive edge.

Table 29. Serial Ports—Internal Clock

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Timing Requirements										
t_{SFSI}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	11.3		11.3		9.6		9.6		ns
t_{HFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	–1.5		–1.5		–1.5		–1.5		ns
t_{SDRI}	Receive Data Setup Before RSCLKx[1]	11.3		11.3		9.6		9.6		ns
t_{HDRI}	Receive Data Hold After RSCLKx[1]	–1.5		–1.5		–1.5		–1.5		ns
Switching Characteristics										
t_{SCLKIW}	TSCLKx/RSCLKx Width	5.4		5.4		4.5		4.5		ns
t_{SCLKI}	TSCLKx/RSCLKx Period	18.0		18.0		15.0		15.0		ns
t_{DFSI}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]		3.0		3.0		3.0		3.0	ns
t_{HOFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[1]	–4.0		–4.0		–1.0		–1.0		ns
t_{DDTI}	Transmit Data Delay After TSCLKx[1]		3.0		3.0		3.0		3.0	ns
t_{HDTI}	Transmit Data Hold After TSCLKx[1]	–1.8		–1.8		–1.0		–1.0		ns

[1] Referenced to sample edge.
[2] Referenced to drive edge.

Table 30. Serial Ports—Enable and Three-State

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Switching Characteristics										
t_{DTENE}	Data Enable Delay from External TSCLKx[1]	0.0		0.0		0.0		0.0		ns
t_{DDTTE}	Data Disable Delay from External TSCLKx[1]		10.0		10.0		10.0		10.0	ns
t_{DTENI}	Data Enable Delay from Internal TSCLKx[1]	–2.0		–2.0		–2.0		–2.0		ns
t_{DDTTI}	Data Disable Delay from Internal TSCLKx[1]		3.0		3.0		3.0		3.0	ns

[1] Referenced to drive edge.

Table 31. External Late Frame Sync

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Switching Characteristics										
$t_{DDTLFSE}$	Data Delay from Late External TFSx or External RFSx in multi-channel mode with MFD = 0[1, 2]		10.0		10.0		10.0		10.0	ns
$t_{DTENLFSE}$	Data Enable from Late FS or in multi-channel mode with MFD = 0[1, 2]	0.0		0.0		0.0		0.0		ns

[1] When in multi-channel mode, TFSx enable and TFSx valid follow $t_{DDTENFS}$ and $t_{DDTLFSE}$.
[2] If external RFSx/TFSx setup to RSCLKx/TSCLKx > $t_{SCLKE}/2$ then $t_{DDTTE/I}$ and $t_{DTENE/I}$ apply, otherwise $t_{DDTLFSE}$ and $t_{DTENLFS}$ apply.

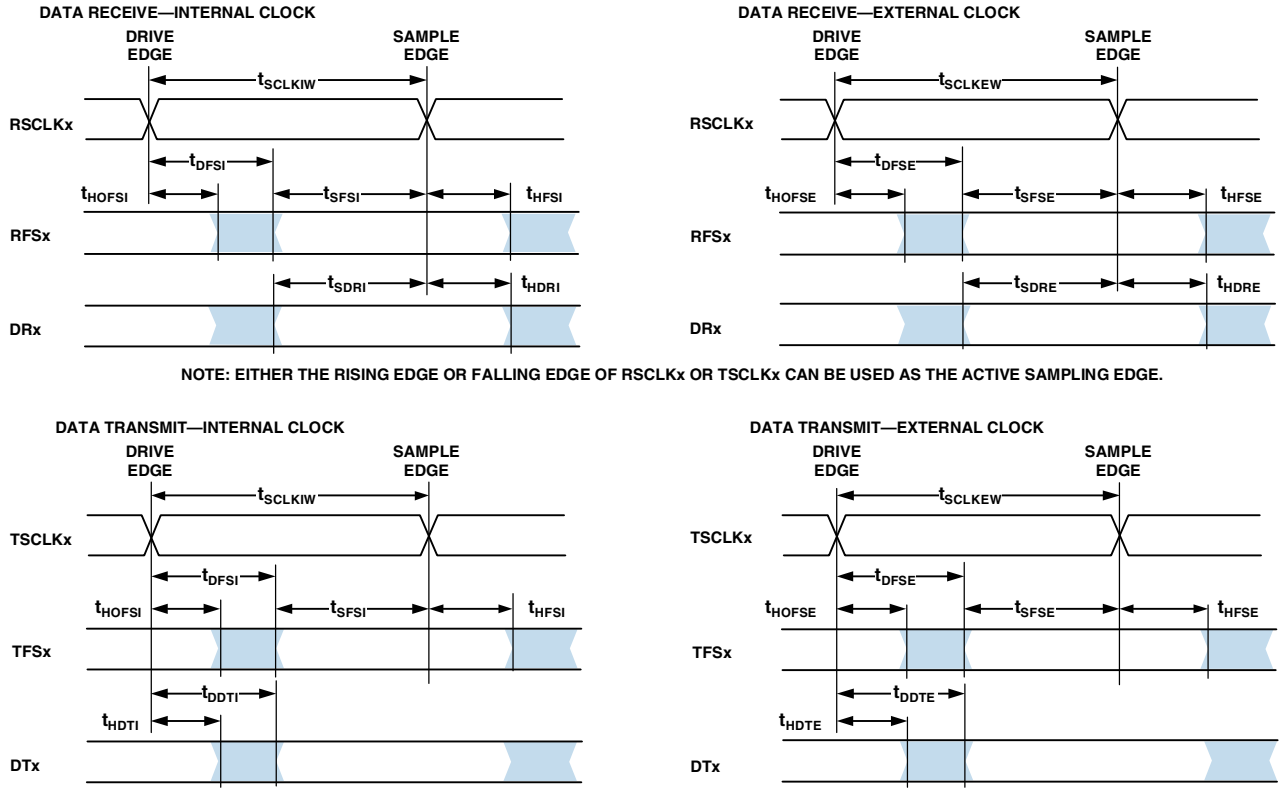


Figure 23. Serial Ports

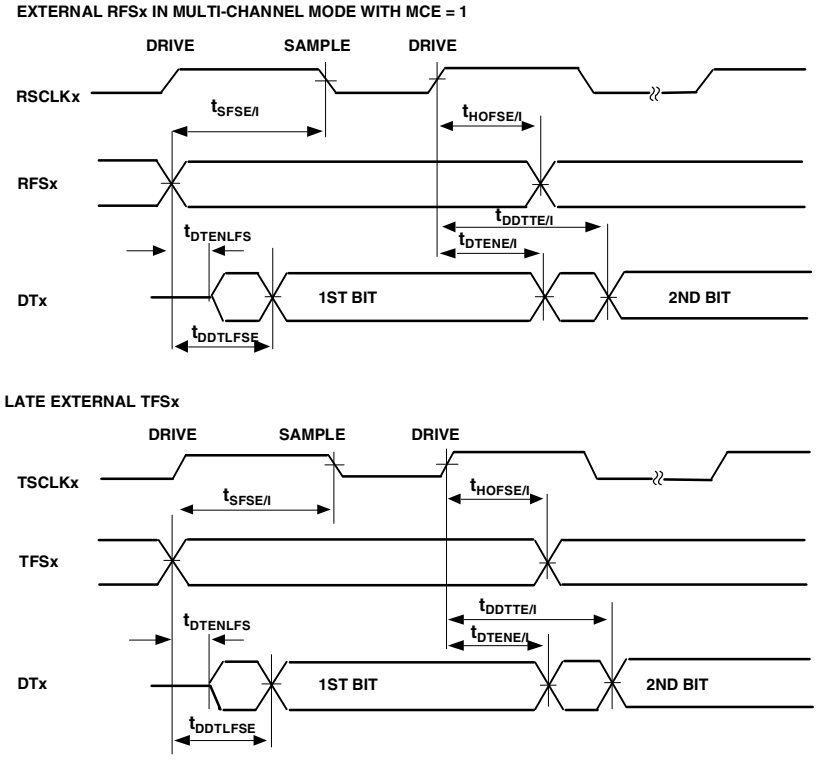


Figure 24. External Late Frame Sync

Serial Peripheral Interface (SPI) Port—Master Timing

Table 32 and Figure 25 describe SPI port master operations.

Table 32. Serial Peripheral Interface (SPI) Port—Master Timing

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Timing Requirements										
t_{SSPIDM}	Data Input Valid to SCK Edge (Data Input Setup)	11.6		11.6		9.6		9.6		ns
t_{HSPIDM}	SCK Sampling Edge to Data Input Invalid	–1.5		–1.5		–1.5		–1.5		ns
Switching Characteristics										
t_{SDSCIM}	$\overline{SPISELx}$ low to First SCK Edge	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SPICHM}	Serial Clock High Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLM}	Serial Clock Low Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLK}	Serial Clock Period	$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		ns
t_{HDSM}	Last SCK Edge to $\overline{SPISELx}$ High	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SPITDM}	Sequential Transfer Delay	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
$t_{DDSPIDM}$	SCK Edge to Data Out Valid (Data Out Delay)		6		6		6		6	ns
$t_{HDSPIDM}$	SCK Edge to Data Out Invalid (Data Out Hold)	–1.0		–1.0		–1.0		–1.0		ns

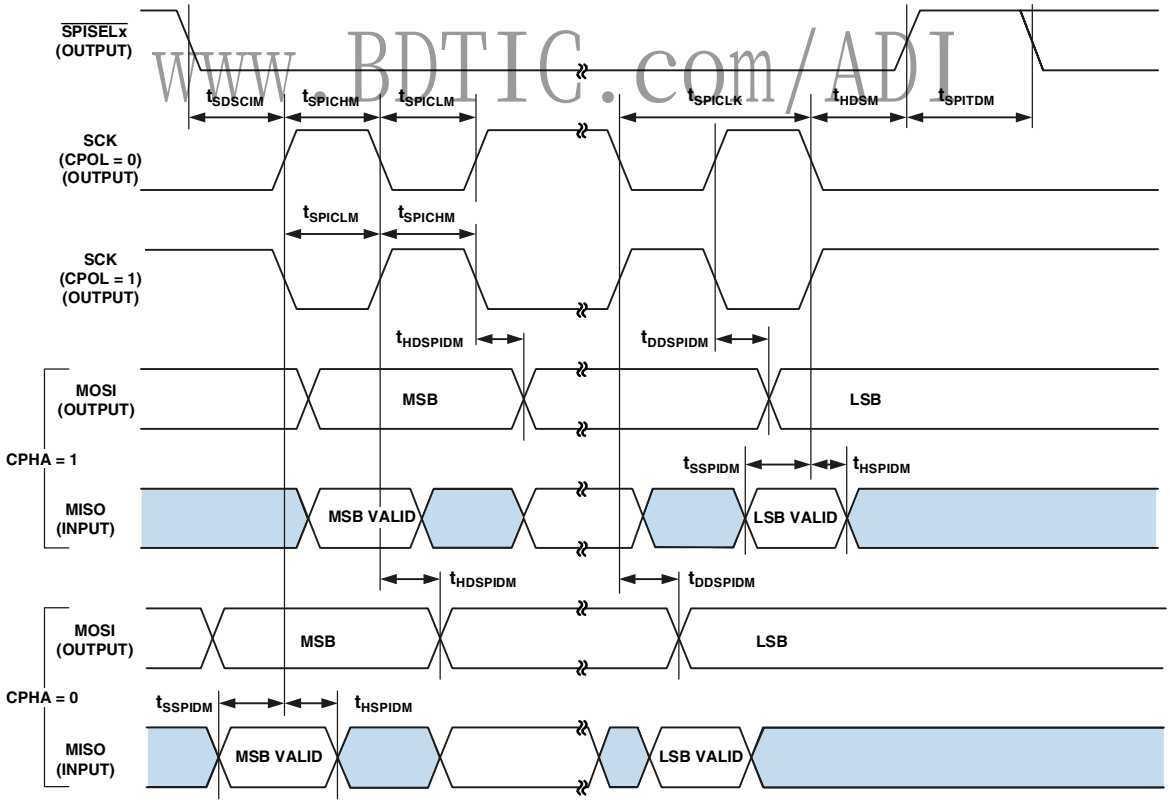


Figure 25. Serial Peripheral Interface (SPI) Port—Master Timing

Serial Peripheral Interface (SPI) Port—Slave Timing

Table 33 and Figure 26 describe SPI port slave operations.

Table 33. Serial Peripheral Interface (SPI) Port—Slave Timing

Parameter		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
Timing Requirements										
t_{SPICHS}	Serial Clock High Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLS}	Serial Clock Low Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLK}	Serial Clock Period	$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		ns
t_{HDS}	Last SCK Edge to $\overline{SPISS}$ Not Asserted	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SPITDS}	Sequential Transfer Delay	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SDSCI}	$\overline{SPISS}$ Assertion to First SCK Edge	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SSPID}	Data Input Valid to SCK Edge (Data Input Setup)	1.6		1.6		1.6		1.6		ns
t_{HSPID}	SCK Sampling Edge to Data Input Invalid	1.6		1.6		1.6		1.6		ns
Switching Characteristics										
t_{DSOE}	$\overline{SPISS}$ Assertion to Data Out Active	0	12.0	0	12.0	0	10.3	0	10.3	ns
t_{DSDHI}	$\overline{SPISS}$ Deassertion to Data High Impedance	0	8.5	0	8.5	0	8	0	8	ns
t_{DDSPID}	SCK Edge to Data Out Valid (Data Out Delay)		10		10		10		10	ns
t_{HDSPID}	SCK Edge to Data Out Invalid (Data Out Hold)	0		0		0		0		ns

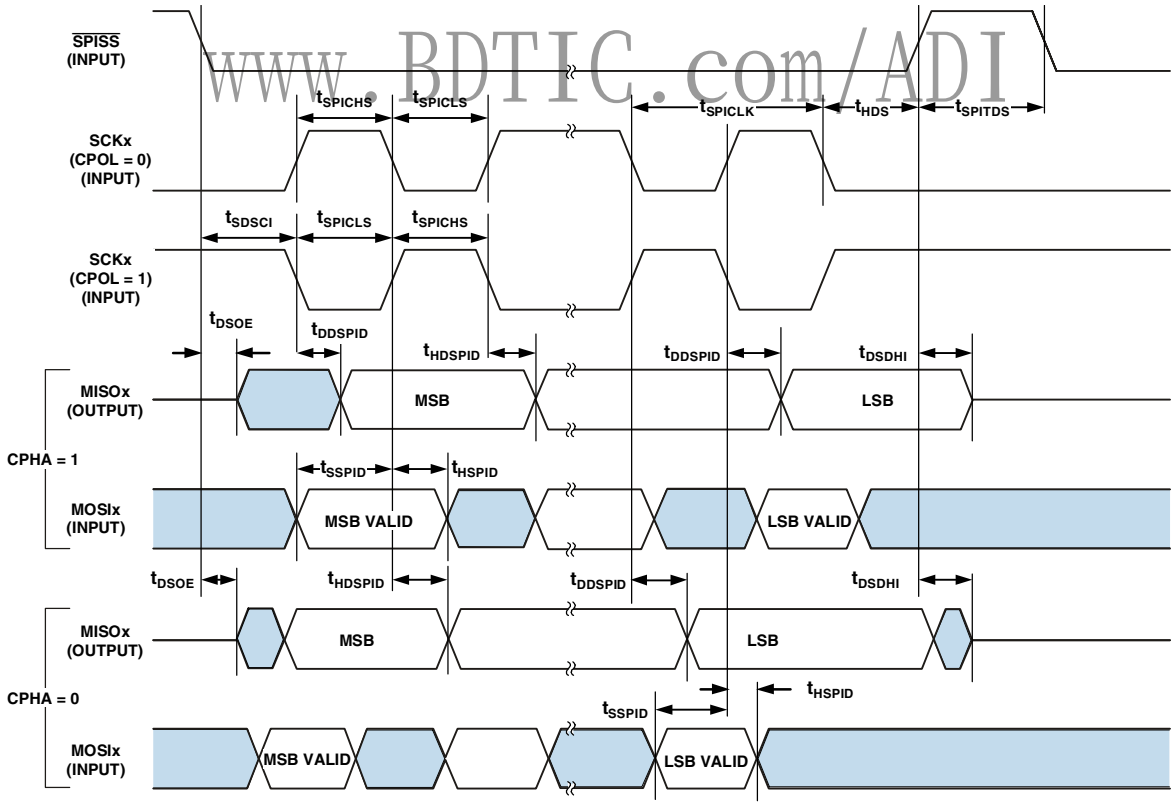


Figure 26. Serial Peripheral Interface (SPI) Port—Slave Timing

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

Figure 27 describes the UART ports receive and transmit operations. The maximum baud rate is SCLK/16. There is some latency between the generation of internal UART interrupts and the external data operations. These latencies are negligible at the data transmission rates for the UART.

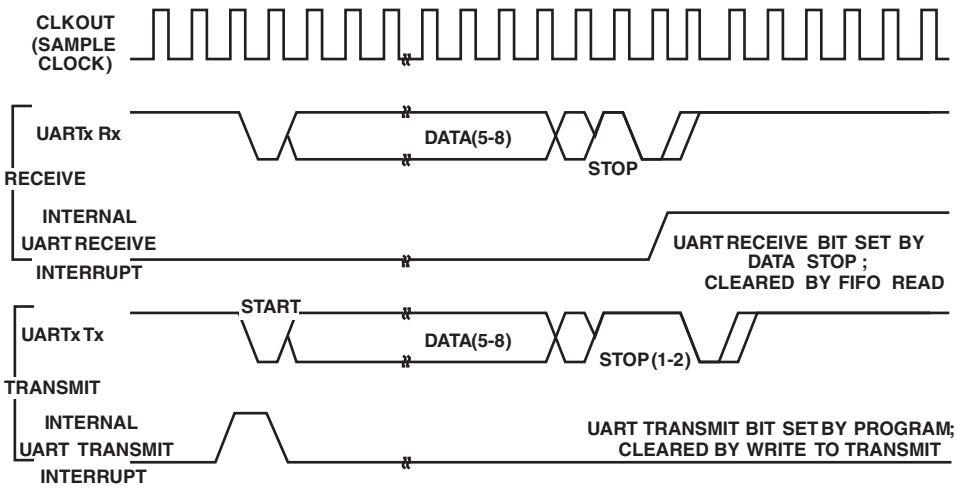


Figure 27. UART Ports—Receive and Transmit Timing

General-Purpose Port Timing

Table 34 and Figure 28 describe general-purpose port operations.

Table 34. General-Purpose Port Timing

Parameter		ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Min	Max	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Min	Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	Max	Unit
Timing Requirement										
t_{WFI}	General-Purpose Port Ball Input Pulse Width	t_{SCLK} + 1		t_{SCLK} + 1		t_{SCLK} + 1		t_{SCLK} + 1		ns
Switching Characteristics										
t_{GPOD}	General-Purpose Port Ball Output Delay from CLKOUT Low	0	9.66	0	9.66	0	6	0	6	ns

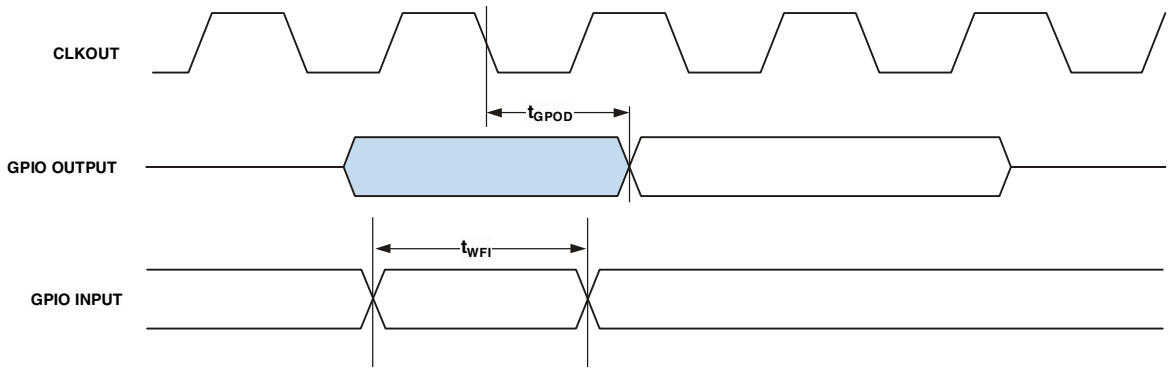


Figure 28. General-Purpose Port Timing

Timer Cycle Timing

Table 35 and Figure 29 describe timer expired operations. The input signal is asynchronous in "width capture mode" and "external clock mode" and has an absolute maximum input frequency of (f_{SCLK}/2) MHz.

Table 35. Timer Cycle Timing

Parameter		ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Min	ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Max	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Characteristics										
t_{WL}	Timer Pulse Width Input Low (Measured In SCLK Cycles)[1]	$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		ns
t_{WH}	Timer Pulse Width Input High (Measured In SCLK Cycles)[1]	$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		ns
t_{TIS}	Timer Input Setup Time Before CLKOUT Low[2]	5		5		5		5		ns
t_{TIH}	Timer Input Hold Time After CLKOUT Low[2]	–2		–2		–2		–2		ns
Switching Characteristics										
t_{HTO}	Timer Pulse Width Output (Measured In SCLK Cycles)	$1 \times t_{SCLK}$	$(2^{32}-1)t_{SCLK}$	$1 \times t_{SCLK}$	$(2^{32}-1)t_{SCLK}$	$1 \times t_{SCLK}$	$(2^{32}-1)t_{SCLK}$	$1 \times t_{SCLK}$	$(2^{32}-1)t_{SCLK}$	ns
t_{TOD}	Timer Output Update Delay After CLKOUT High		8.1		8.1		6		6	ns

[1] The minimum pulse widths apply for TMRx signals in width capture and external clock modes. They also apply to the PF15 or PPI_CLK signals in PWM output mode.
[2] Either a valid setup and hold time or a valid pulse width is sufficient. There is no need to resynchronize programmable flag inputs.

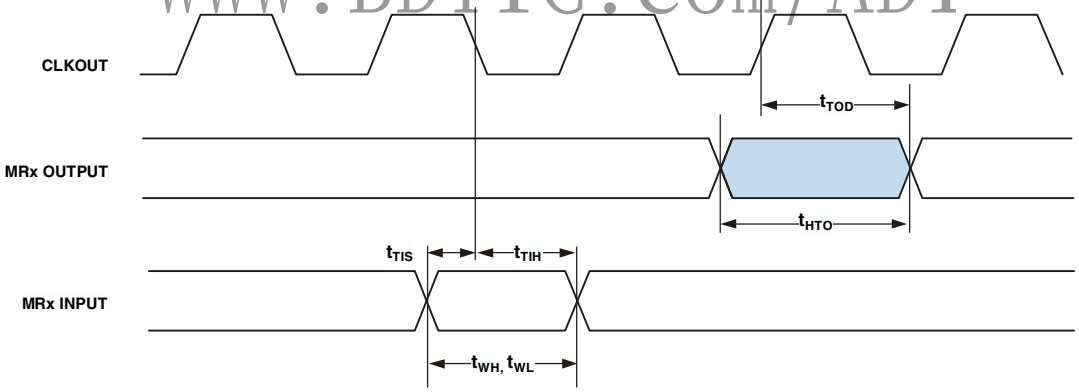


Figure 29. Timer Cycle Timing

Timer Clock Timing

Table 36 and Figure 30 describe timer clock timing.

Table 36. Timer Clock Timing

Parameter		V_{DDEXT} = 1.8 V Min	V_{DDEXT} = 1.8 V Max	V_{DDEXT} = 2.5/3.3 V Min	V_{DDEXT} = 2.5/3.3 V Max	Unit
Switching Characteristic						
t_{TODP}	Timer Output Update Delay After PPI_CLK High		12.64		12.64	ns

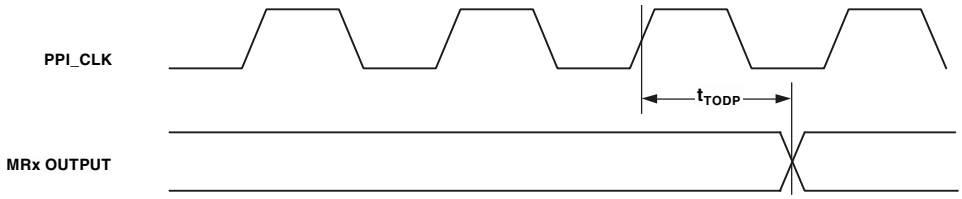


Figure 30. Timer Clock Timing

Up/Down Counter/Rotary Encoder Timing

Table 37. Up/Down Counter/Rotary Encoder Timing

Parameter		V_{DDEXT} = 1.8 V Min	V_{DDEXT} = 1.8 V Max	V_{DDEXT} = 2.5/3.3 V Min	V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Requirements						
t_{WCOUNT}	Up/Down Counter/Rotary Encoder Input Pulse Width	t_{SCLK} + 1		t_{SCLK} + 1		ns
t_{CIS}	Counter Input Setup Time Before CLKOUT Low[1]	4.0		4.0		ns
t_{CIH}	Counter Input Hold Time After CLKOUT Low1	4.0		4.0		ns

[1] Either a valid setup and hold time or a valid pulse width is sufficient. There is no need to resynchronize counter inputs.

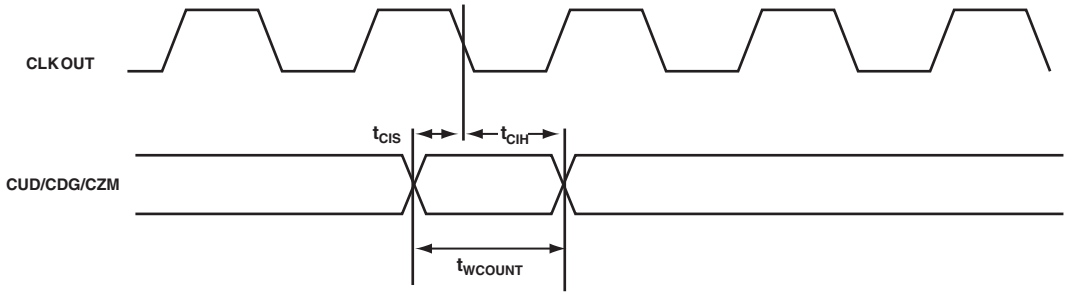


Figure 31. Up/Down Counter/Rotary Encoder Timing

HOSTDP A/C Timing- Host Read Cycle

Table 38 describe the HOSTDP A/C Host Read Cycle timing requirements.

Table 38. Host Read Cycle Timing Requirements

Parameter		ADSP-BF522/524/526, V_{DDEXT} = 1.8 V Min	ADSP-BF522/524/526, V_{DDEXT} = 1.8 V Max	ADSP-BF522/524/526, V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF522/524/526, V_{DDEXT} = 2.5/3.3 V Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Requirements										
t_{SADRDL}	HOST_ADDR and $\overline{HOST\_CE}$ Setup before $\overline{HOST\_RD}$ falling edge	4		4		4		4		ns
t_{HADRDH}	HOST_ADDR and $\overline{HOST\_CE}$ Hold after $\overline{HOST\_RD}$ rising edge	2.5		2.5		2.5		2.5		ns
t_{RDWL}	$\overline{HOST\_RD}$ pulse width low (ACK mode)	$t_{DRDYRDL} + t_{RDYPRD} + t_{DRDHRDY}$		$t_{DRDYRDL} + t_{RDYPRD} + t_{DRDHRDY}$		$t_{DRDYRDL} + t_{RDYPRD} + t_{DRDHRDY}$		$t_{DRDYRDL} + t_{RDYPRD} + t_{DRDHRDY}$		ns
t_{RDWL}	$\overline{HOST\_RD}$ pulse width low (INT mode)	$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		ns
t_{RDWH}	$\overline{HOST\_RD}$ pulse width high or time between $\overline{HOST\_RD}$ rising edge and $\overline{HOST\_WR}$ falling edge	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
$t_{DRDHRDY}$	$\overline{HOST\_RD}$ rising edge delay after HOST_ACK rising edge (ACK mode)	0		0		0		0		ns
Switching Characteristics										
$t_{SDATRDY}$	Data valid prior HOST_ACK rising edge (ACK mode)	4.5		3.5		4.5		3.5		ns
$t_{DRDYRDL}$	Host_ACK assertion delay after $\overline{HOST\_RD}$/$\overline{HOST\_CE}$ (ACK mode)		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$	ns
t_{RDYPRD}	HOST_ACK low pulse-width for Read access (ACK mode)		NM[1]		NM[1]		NM[1]		NM[1]	ns
t_{DDARWH}	Data disable after $\overline{HOST\_RD}$		9.0		9.0		9.0		9.0	ns
t_{ACC}	Data valid after $\overline{HOST\_RD}$ falling edge (INT mode)		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$	ns
t_{HDARWH}	Data hold after $\overline{HOST\_RD}$ rising edge	1.0		1.0		1.0		1.0		ns

[1] NM (Not Measured) — This parameter is not measured, because the time for which HOST_ACK is low is system design dependent.

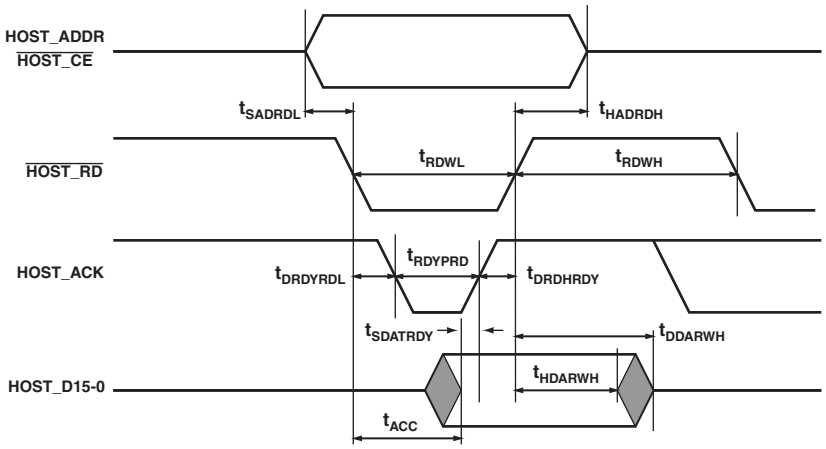


Figure 32. HOSTDP A/C- Host Read Cycle

HOSTDP A/C Timing- Host Write Cycle

Table 39 describes the HOSTDP A/C Host Write Cycle timing requirements.

Table 39. Host Write Cycle Timing Requirements

Parameter		ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Min	ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Max	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Requirements										
t_{SADWRL}	HOST_ADDR/$\overline{HOST\_CE}$ Setup before $\overline{HOST\_WR}$ falling edge	4		4		4		4		ns
t_{HADWRH}	HOST_ADDR/$\overline{HOST\_CE}$ Hold after $\overline{HOST\_WR}$ rising edge	2.5		2.5		2.5		2.5		ns
t_{WRWL}	$\overline{HOST\_WR}$ pulse width low (ACK mode)	$t_{DRDYWRL} + t_{RDYPRD} + t_{DWRHRDY}$		$t_{DRDYWRL} + t_{RDYPRD} + t_{DWRHRDY}$		$t_{DRDYWRL} + t_{RDYPRD} + t_{DWRHRDY}$		$t_{DRDYWRL} + t_{RDYPRD} + t_{DWRHRDY}$		ns
	$\overline{HOST\_WR}$ pulse width low (INT mode)	$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		$1.5 \times t_{SCLK} + 8.7$		ns
t_{WRWH}	$\overline{HOST\_WR}$ pulse width high or time between $\overline{HOST\_WR}$ rising edge and $\overline{HOST\_RD}$ falling edge	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
$t_{DWRHRDY}$	$\overline{HOST\_WR}$ rising edge delay after HOST_ACK rising edge (ACK mode)	0		0		0		0		ns
t_{HDATWH}	Data Hold after $\overline{HOST\_WR}$ rising edge	2.5		2.5		2.5		2.5		ns
t_{SDATWH}	Data Setup before $\overline{HOST\_WR}$ rising edge	2.5		2.5		2.5		2.5		ns
Switching Characteristics										
$t_{DRDYWRL}$	HOST_ACK low delay after $\overline{HOST\_WR}$/$\overline{HOST\_CE}$ asserted (ACK mode)		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$		$1.5 \times t_{SCLK}$	ns
t_{RDYPWR}	HOST_ACK low pulse-width for Write access (ACK mode)		NM[1]		NM[1]		NM[1]		NM[1]	ns

[1] NM (Not Measured) — This parameter is not measured, because the time for which HOST_ACK is low is system design dependent.

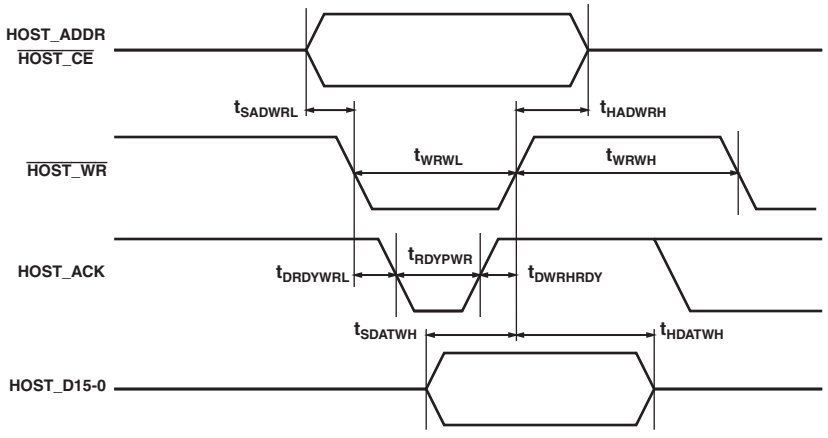


Figure 33. HOSTDP A/C- Host Write Cycle

10/100 Ethernet MAC Controller Timing

Table 40 through Table 45 and Figure 34 through Figure 39 describe the 10/100 Ethernet MAC Controller operations.

Table 40. 10/100 Ethernet MAC Controller Timing: MII Receive Signal

Parameter[1]		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		Unit
		Min	Max	Min	Max	
$t_{ERXCLKF}$	ERxCLK Frequency (f_{SCLK} = SCLK Frequency)	None	25 + 1% f_{SCLK} + 1%	None	25 + 1% f_{SCLK} + 1%	MHz
$t_{ERXCLKW}$	ERxCLK Width (t_{ERxCLK} = ERxCLK Period)	t_{ERxCLK} x 35%	t_{ERxCLK} x 65%	t_{ERxCLK} x 35%	t_{ERxCLK} x 65%	ns
$t_{ERXCLKIS}$	Rx Input Valid to ERxCLK Rising Edge (Data In Setup)	7.5		7.5		ns
$t_{ERXCLKIH}$	ERxCLK Rising Edge to Rx Input Invalid (Data In Hold)	7.5		7.5		ns

[1] MII inputs synchronous to ERxCLK are ERxD3–0, ERxDV, and ERxER.

Table 41. 10/100 Ethernet MAC Controller Timing: MII Transmit Signal

Parameter[1]		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		Unit
		Min	Max	Min	Max	
$t_{ETXCLKF}$	ETxCLK Frequency (f_{SCLK} = SCLK Frequency)	None	25 + 1% f_{SCLK} + 1%	None	25 + 1% f_{SCLK} + 1%	MHz
$t_{ETXCLKW}$	ETxCLK Width (t_{ETxCLK} = ETxCLK Period)	t_{ETxCLK} x 35%	t_{ETxCLK} x 65%	t_{ETxCLK} x 35%	t_{ETxCLK} x 65%	ns
$t_{ETXCLKOV}$	ETxCLK Rising Edge to Tx Output Valid (Data Out Valid)		20		20	ns
$t_{ETXCLKOH}$	ETxCLK Rising Edge to Tx Output Invalid (Data Out Hold)	0		0		ns

[1] MII outputs synchronous to ETxCLK are ETxD3–0.

Table 42. 10/100 Ethernet MAC Controller Timing: RMII Receive Signal

Parameter[1]		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		Unit
		Min	Max	Min	Max	
$t_{EREFCLKF}$	REF_CLK Frequency (f_{SCLK} = SCLK Frequency)	None	50 + 1% 2 x f_{SCLK} + 1%	None	50 + 1% 2 x f_{SCLK} + 1%	MHz
$t_{EREFCLKW}$	EREF_CLK Width ($t_{EREFCLK}$ = EREFCLK Period)	$t_{EREFCLK}$ x 35%	$t_{EREFCLK}$ x 65%	$t_{EREFCLK}$ x 35%	$t_{EREFCLK}$ x 65%	ns
$t_{EREFCLKIS}$	Rx Input Valid to RMII REF_CLK Rising Edge (Data In Setup)	4		4		ns
$t_{EREFCLKIH}$	RMII REF_CLK Rising Edge to Rx Input Invalid (Data In Hold)	2		2		ns

[1] RMII inputs synchronous to RMII REF_CLK are ERxD1–0, RMII CRS_DV, and ERxER.

Table 43. 10/100 Ethernet MAC Controller Timing: RMII Transmit Signal

Parameter[1]		ADSP-BF522/524/526				ADSP-BF523/525/527				Unit
		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		V_{DDEXT} = 1.8 V		V_{DDEXT} = 2.5/3.3 V		
		Min	Max	Min	Max	Min	Max	Min	Max	
$t_{EREFCLKOV}$	RMII REF_CLK Rising Edge to Tx Output Valid (Data Out Valid)		8.1		8.1		7.5		7.5	ns
$t_{EREFCLKOH}$	RMII REF_CLK Rising Edge to Tx Output Invalid (Data Out Hold)	2		2		2		2		ns

[1] RMII outputs synchronous to RMII REF_CLK are ETxD1–0.

Table 44. 10/100 Ethernet MAC Controller Timing: MII/RMII Asynchronous Signal

Parameter[1, 2]		V_{DDEXT} = 1.8 V Min	Max	V_{DDEXT} = 2.5/3.3 V Min	Max	Unit
t_{ECOLH}	COL Pulse Width High	t_{ETxCLK} x 1.5 t_{ERxCLK} x 1.5		t_{ETxCLK} x 1.5 t_{ERxCLK} x 1.5		ns
t_{ECOLL}	COL Pulse Width Low	t_{ETxCLK} x 1.5 t_{ERxCLK} x 1.5		t_{ETxCLK} x 1.5 t_{ERxCLK} x 1.5		ns
t_{ECRSH}	CRS Pulse Width High	t_{ETxCLK} x 1.5		t_{ETxCLK} x 1.5		ns
t_{ECRSL}	CRS Pulse Width Low	t_{ETxCLK} x 1.5		t_{ETxCLK} x 1.5		ns

[1] MII/RMII asynchronous signals are COL and CRS. These signals are applicable in both MII and RMII modes. The asynchronous COL input is synchronized separately to both the ETxCLK and the ERxCLK, and the COL input must have a minimum pulse width high or low at least 1.5 times the period of the slower of the two clocks.
[2] The asynchronous CRS input is synchronized to the ETxCLK, and the CRS input must have a minimum pulse width high or low at least 1.5 times the period of ETxCLK.

Table 45. 10/100 Ethernet MAC Controller Timing: MII Station Management

Parameter[1]		ADSP-BF522/524/526 V_{DDEXT} = 1.8 V Min	Max	ADSP-BF522/524/526 V_{DDEXT} = 2.5/3.3 V Min	Max	ADSP-BF523/525/527 V_{DDEXT} = 1.8 V Min	Max	ADSP-BF523/525/527 V_{DDEXT} = 2.5/3.3 V Min	Max	Unit
t_{MDIOS}	MDIO Input Valid to MDC Rising Edge (Setup)	11.5		11.5		10		10		ns
t_{MDCIH}	MDC Rising Edge to MDIO Input Invalid (Hold)	11.5		11.5		10		10		ns
t_{MDCOV}	MDC Falling Edge to MDIO Output Valid	25		25		25		25		ns
t_{MDCOH}	MDC Falling Edge to MDIO Output Invalid (Hold)	–1		–1		–1		–1		ns

[1] MDC/MDIO is a 2-wire serial bidirectional port for controlling one or more external PHYs. MDC is an output clock whose minimum period is programmable as a multiple of the system clock SCLK. MDIO is a bidirectional data line.

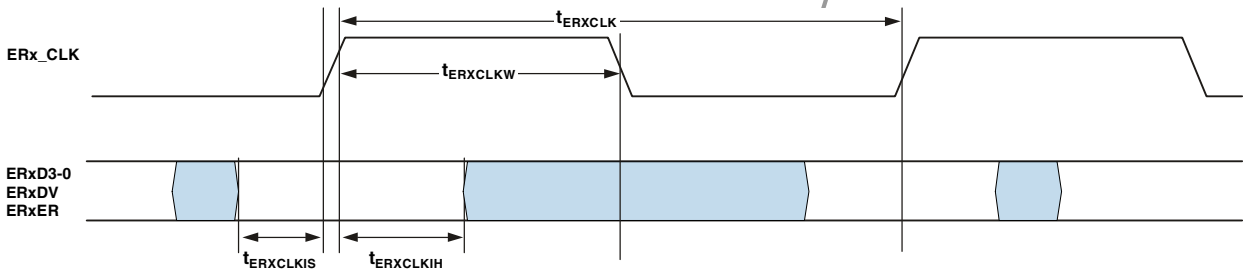


Figure 34. 10/100 Ethernet MAC Controller Timing: MII Receive Signal

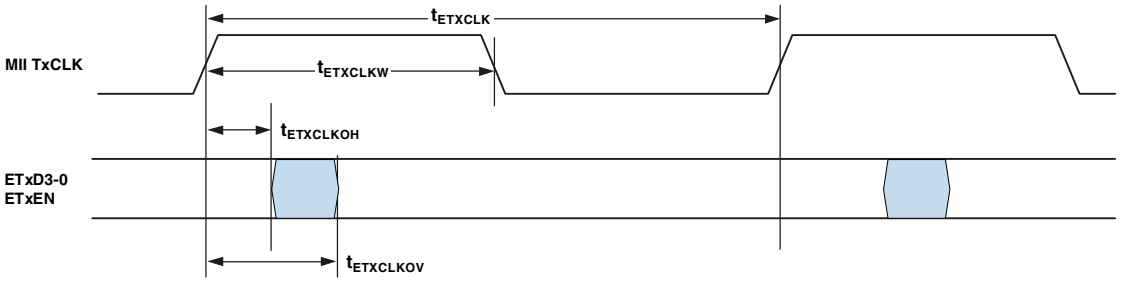


Figure 35. 10/100 Ethernet MAC Controller Timing: MII Transmit Signal

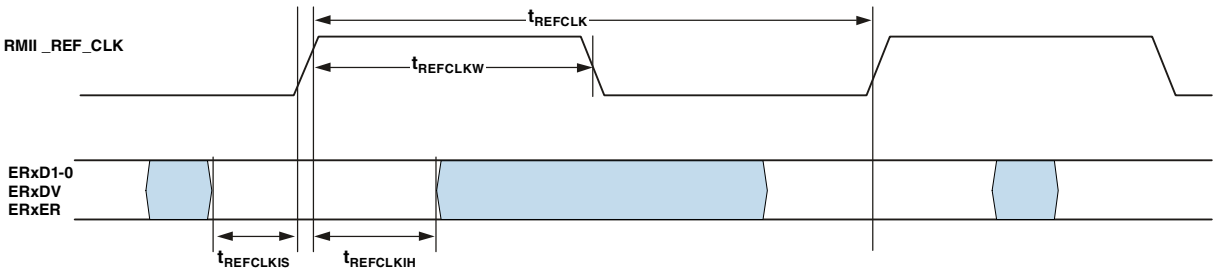


Figure 36. 10/100 Ethernet MAC Controller Timing: RMII Receive Signal

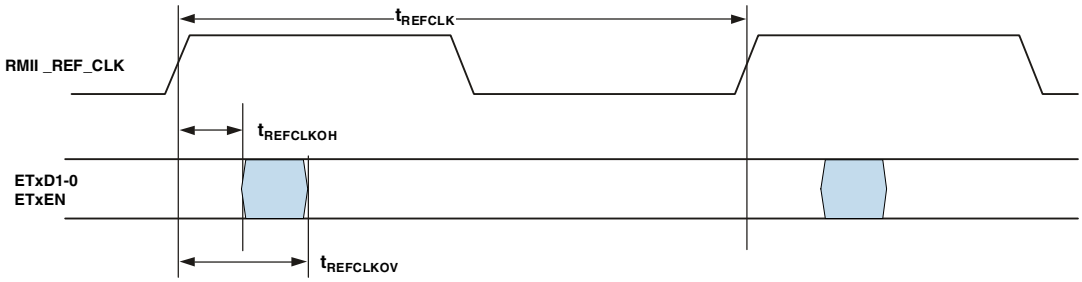


Figure 37. 10/100 Ethernet MAC Controller Timing: RMII Transmit Signal

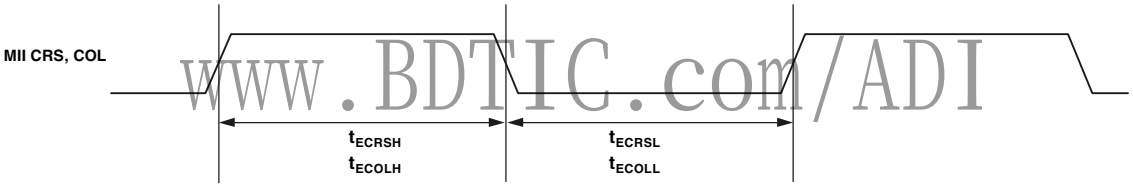


Figure 38. 10/100 Ethernet MAC Controller Timing: Asynchronous Signal

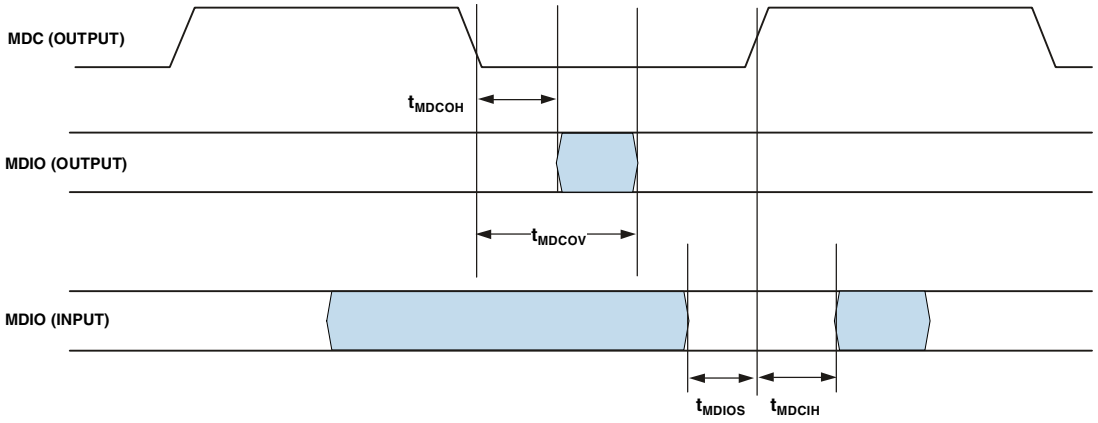


Figure 39. 10/100 Ethernet MAC Controller Timing: MII Station Management

JTAG Test And Emulation Port Timing

Table 46 and Figure 40 describe JTAG port operations.

Table 46. JTAG Port Timing

Parameter		V_{DDEXT} = 1.8 V Min	V_{DDEXT} = 1.8 V Max	V_{DDEXT} = 2.5/3.3 V Min	V_{DDEXT} = 2.5/3.3 V Max	Unit
Timing Parameters						
t_{TCK}	TCK Period	20		20		ns
t_{STAP}	TDI, TMS Setup Before TCK High	4		4		ns
t_{HTAP}	TDI, TMS Hold After TCK High	4		4		ns
t_{SSYS}	System Inputs Setup Before TCK High[1]	4		4		ns
t_{HSYS}	System Inputs Hold After TCK High[1]	5		5		ns
t_{TRSTW}	$\overline{TRST}$ Pulse Width[2] (measured in TCK cycles)	4		4		TCK
Switching Characteristics						
t_{DTDO}	TDO Delay from TCK Low		10		10	ns
t_{DSYS}	System Outputs Delay After TCK Low[3]		12		12	ns

[1] System Inputs = DATA15–0, ARDY, SCL, SDA, PF15–0, PG15–0, PH15–0, TCK, $\overline{TRST}$, $\overline{RESET}$, $\overline{NMI}$, BMODE3–0.
[2] 50 MHz Maximum
[3] System Outputs = DATA15–0, ADDR19–1, $\overline{ABE1–0}$, $\overline{AOE}$, $\overline{ARE}$, $\overline{AWE}$, $\overline{AMS3–0}$, $\overline{SRAS}$, $\overline{SCAS}$, $\overline{SWE}$, SCKE, CLKOUT, SA10, $\overline{SMS}$, SCL, SDA, PF15–0, PG15–0, PH15–0, TDO, $\overline{EMU}$.

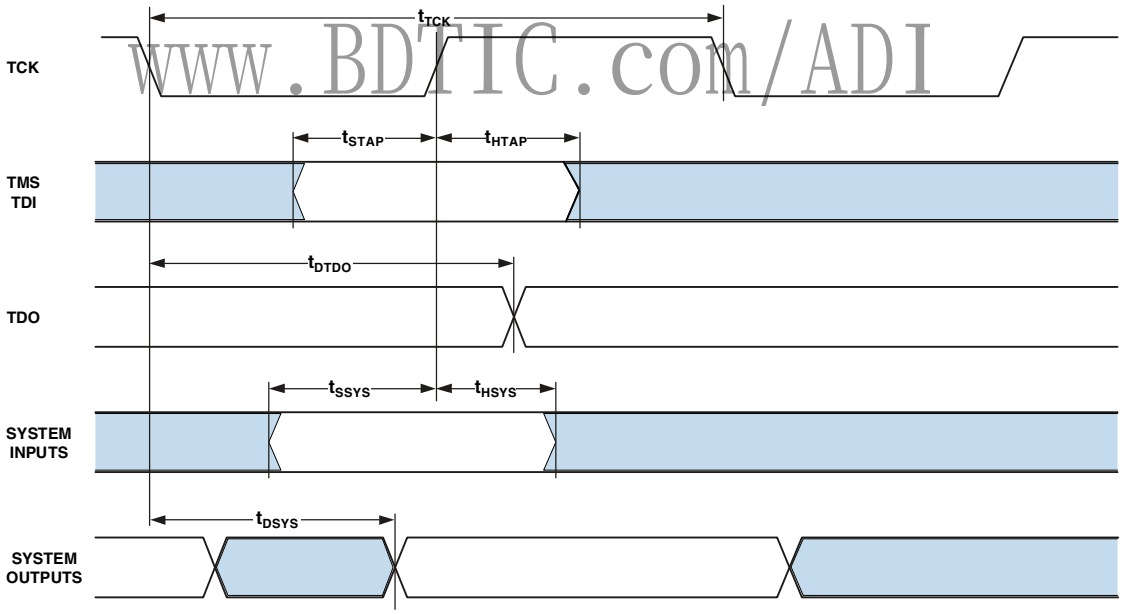


Figure 40. JTAG Port Timing

OUTPUT DRIVE CURRENTS

Figure 41 through Figure 52 show typical current-voltage characteristics for the output drivers of the ADSP-BF523/525/527 and ADSP-BF522/524/526 processors. The curves represent the current drive capability of the output drivers as a function of output voltage. See Table 10 on Page 23 for information about which driver type corresponds to a particular ball.

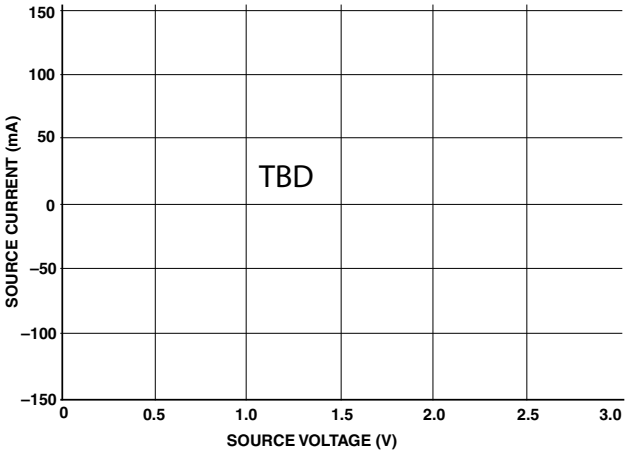


Figure 41. Drive Current A (Low V_{DDEXT}/V_{DDMEM})

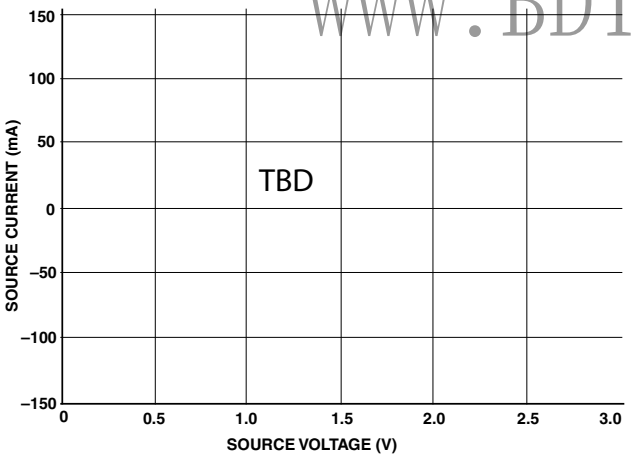


Figure 42. Drive Current A (High V_{DDEXT}/V_{DDMEM})

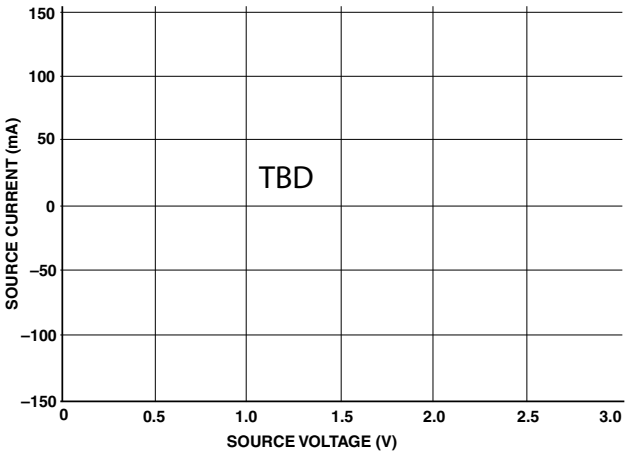


Figure 43. Drive Current B (Low V_{DDEXT}/V_{DDMEM})

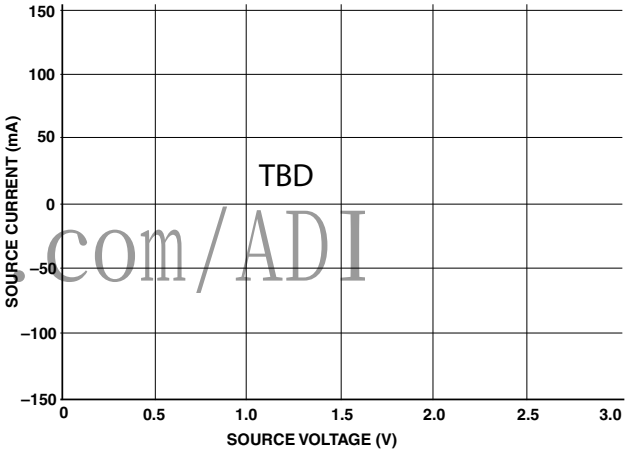


Figure 44. Drive Current B (High V_{DDEXT}/V_{DDMEM})

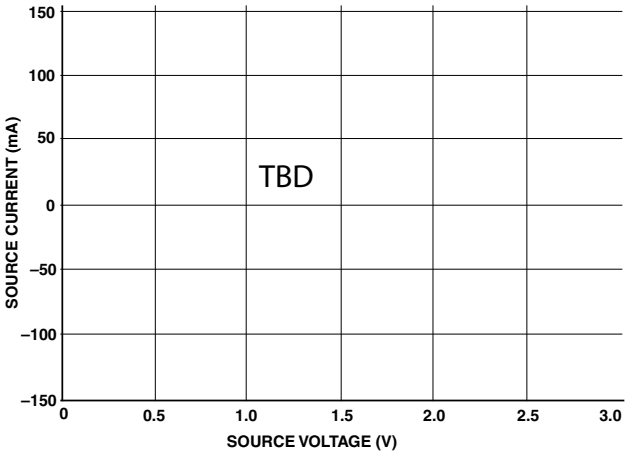


Figure 45. Drive Current C (Low V_{DDEXT}/V_{DDMEM})

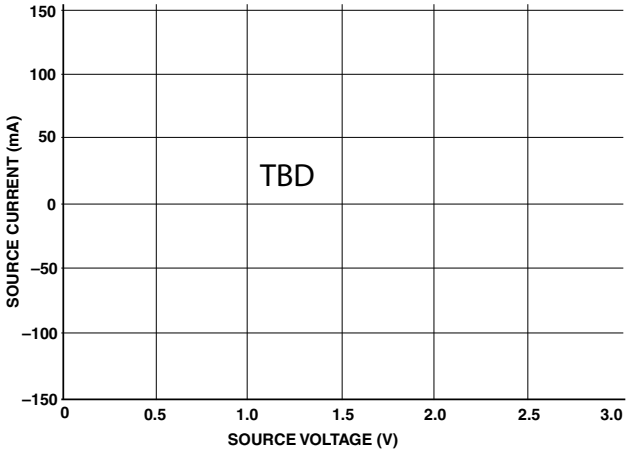


Figure 46. Drive Current C (High V_{DDEXT}/V_{DDMEM})

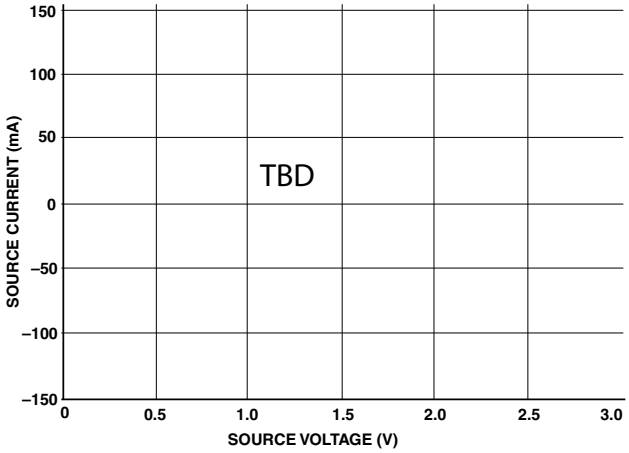


Figure 49. Drive Current E (Low V_{DDEXT}/V_{DDMEM})

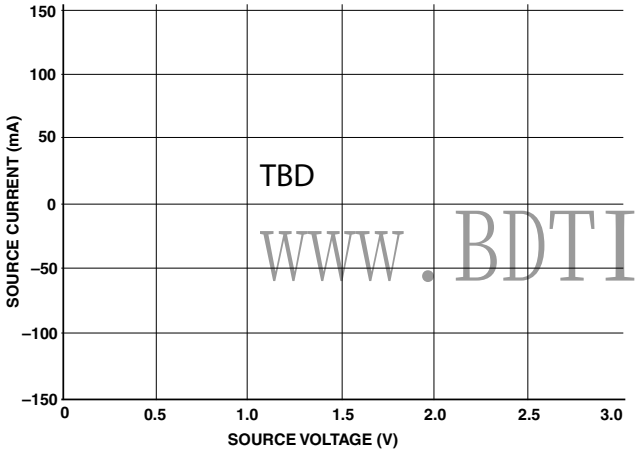


Figure 47. Drive Current D (Low V_{DDEXT}/V_{DDMEM})

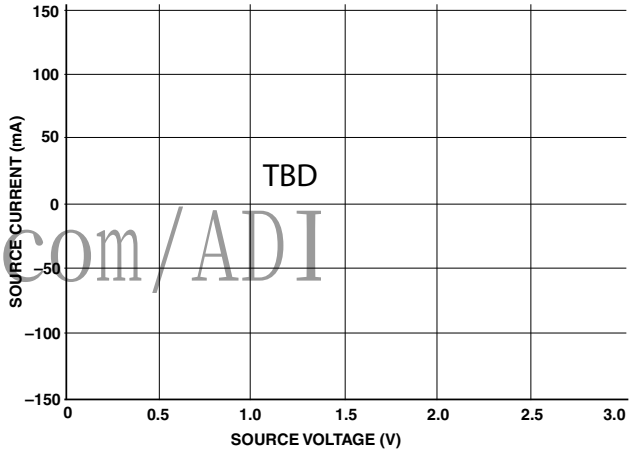


Figure 50. Drive Current E (High V_{DDEXT}/V_{DDMEM})

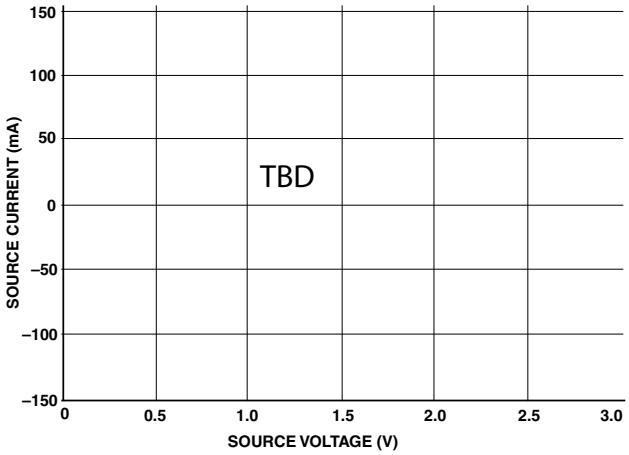


Figure 48. Drive Current D (High V_{DDEXT}/V_{DDMEM})

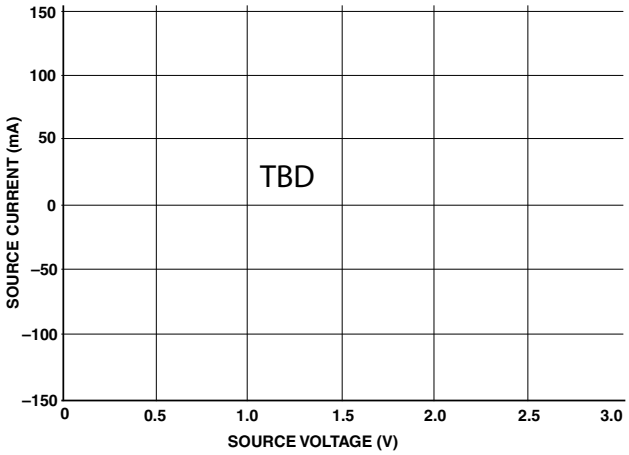


Figure 51. Drive Current F (Low V_{DDEXT}/V_{DDMEM})

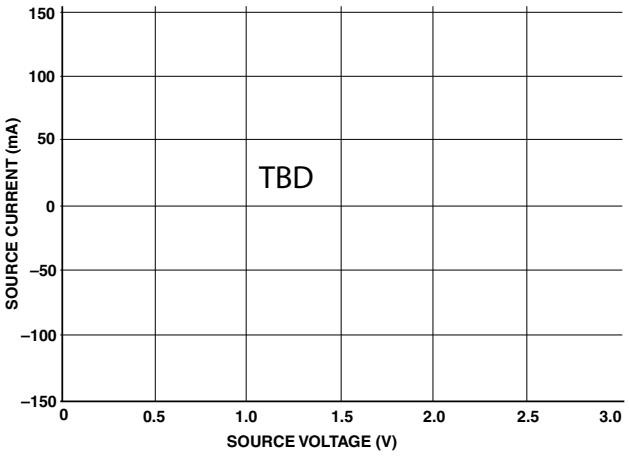


Figure 52. Drive Current F (High V_{DDEXT}/V_{DDMEM})

POWER DISSIPATION

Total power dissipation has two components: one due to internal circuitry (P_{INT}) and one due to the switching of external output drivers (P_{EXT}).

See the *ADSP-BF52x Blackfin Processor Hardware Reference Manual* for definitions of the various operating modes and for instructions on how to minimize system power.

Power dissipation specifications for the ADSP-BF522/523/524/525/526/527 processors are TBD.

TEST CONDITIONS

All timing parameters appearing in this data sheet were measured under the conditions described in this section. Figure 53 shows the measurement point for AC measurements (except output enable/disable). The measurement point V_{MEAS} is V_{DDEXT}/2 or V_{DDMEM}/2 for V_{DDEXT}/V_{DDMEM} (nominal) = 1.8 V/2.5 V/3.3 V.

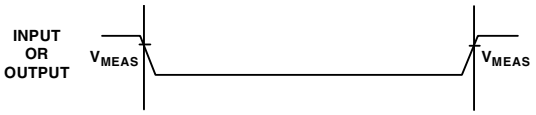


Figure 53. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

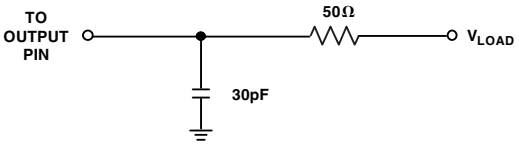


Figure 54. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

Output Enable Time Measurement

Output balls are considered to be enabled when they have made a transition from a high impedance state to the point when they start driving.

The output enable time t_{ENA} is the interval from the point when a reference signal reaches a high or low voltage level to the point when the output starts driving as shown on the right side of Figure 55.

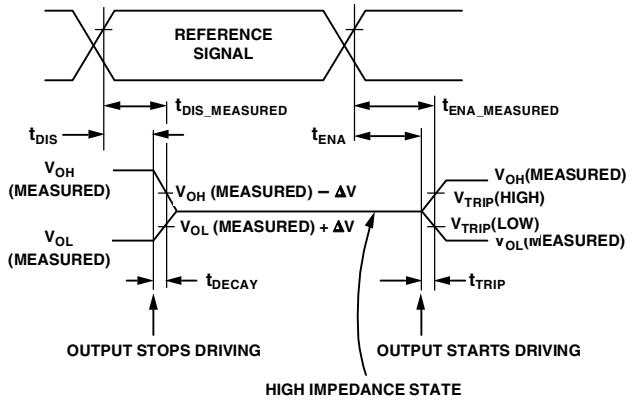


Figure 55. Output Enable/Disable

The time $t_{ENA\_MEASURED}$ is the interval, from when the reference signal switches, to when the output voltage reaches V_{TRIP}(high) or V_{TRIP}(low). V_{TRIP}(high) is 2.0 V and V_{TRIP}(low) is 1.0 V for V_{DDEXT}/V_{DDMEM} (nominal) = 2.5 V/3.3 V. Time t_{TRIP} is the interval from when the output starts driving to when the output reaches the V_{TRIP}(high) or V_{TRIP}(low) trip voltage.

Time t_{ENA} is calculated as shown in the equation:

$$t_{ENA} = t_{ENA\_MEASURED} - t_{TRIP}$$

If multiple balls (such as the data bus) are enabled, the measurement value is that of the first ball to start driving.

Output Disable Time Measurement

Output balls are considered to be disabled when they stop driving, go into a high impedance state, and start to decay from their output high or low voltage. The output disable time t_{DIS} is the difference between $t_{DIS\_MEASURED}$ and t_{DECAY} as shown on the left side of Figure 55.

$$t_{DIS} = t_{DIS\_MEASURED} - t_{DECAY}$$

The time for the voltage on the bus to decay by ΔV is dependent on the capacitive load C_L and the load current I_L. This decay time can be approximated by the equation:

$$t_{DECAY} = (C_L \Delta V)/I_L$$

The time t_{DECAY} is calculated with test loads C_L and I_L, and with ΔV equal to 0.5 V for V_{DDEXT}/V_{DDMEM} (nominal) = 2.5 V/3.3 V.

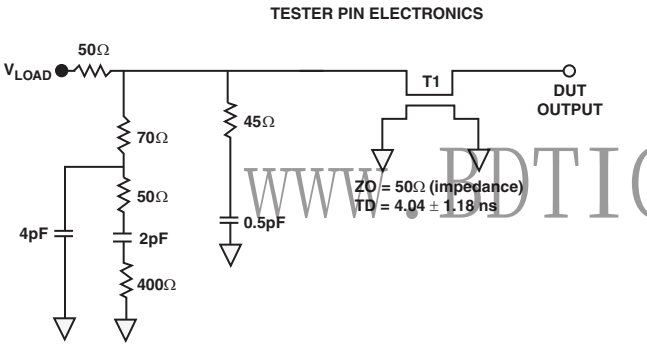
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches, to when the output voltage decays ΔV from the measured output high or output low voltage.

Example System Hold Time Calculation

To determine the data output hold time in a particular system, first calculate t_{DECAY} using the equation given above. Choose ΔV to be the difference between the processor's output voltage and the input threshold for the device requiring the hold time. C_L is the total bus capacitance (per data line), and I_L is the total leakage or three-state current (per data line). The hold time will be t_{DECAY} plus the various output disable times as specified in the Timing Specifications on Page 33 (for example t_{DSDAT} for an SDRAM write cycle as shown in SDRAM Interface Timing on Page 39).

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all balls (see Figure 56). V_{LOAD} is equal to (V_{DDEXT}/V_{DDMEM}) /2. The graphs of Figure 57 through Figure 64 show how output rise time varies with capacitance. The delay and hold specifications given should be derated by a factor derived from these figures. The graphs in these figures may not be linear outside the ranges shown.



NOTES:
THE WORST CASE TRANSMISSION LINE DELAY IS SHOWN AND CAN BE USED FOR THE OUTPUT TIMING ANALYSIS TO REFELECT THE TRANSMISSION LINE EFFECT AND MUST BE CONSIDERED. THE TRANSMISSION LINE (TD), IS FOR LOAD ONLY AND DOES NOT AFFECT THE DATA SHEET TIMING SPECIFICATIONS.

ANALOG DEVICES RECOMMENDS USING THE IBIS MODEL TIMING FOR A GIVEN SYSTEM REQUIREMENT. IF NECESSARY, A SYSTEM MAY INCORPORATE EXTERNAL DRIVERS TO COMPENSATE FOR ANY TIMING DIFFERENCES.

Figure 56. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

Figure 57. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver A at V_{DDEXT}/V_{DDMEM} = Min

Figure 58. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver A at V_{DDEXT}/V_{DDMEM} = Max

Figure 59. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver B at V_{DDEXT}/V_{DDMEM} = Min

Figure 60. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver B at V_{DDEXT}/V_{DDMEM} = Max

Figure 61. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver C at V_{DDEXT}/V_{DDMEM} = Min

Figure 62. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver C at V_{DDEXT}/V_{DDMEM} = Max

Figure 63. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver D at V_{DDEXT}/V_{DDMEM} = Min

Figure 64. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver D at V_{DDEXT}/V_{DDMEM} = Max

ENVIRONMENTAL CONDITIONS

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

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

where:

T_J = Junction temperature (°C)

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

Ψ_{JT} = From Table 48

P_D = Power dissipation (see Power Dissipation on Page 60 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.

Values of θ_{JB} are provided for package comparison and printed circuit board design considerations.

In Table 48, airflow measurements comply with JEDEC standards JESD51-2 and JESD51-6, and the junction-to-board measurement complies with JESD51-8. The junction-to-case measurement complies with MIL-STD-883 (Method 1012.1). All measurements use a 2S2P JEDEC test board.

Table 47. Thermal Characteristics (BC-208-1)

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	23.20	°C/W
θ_{JMA}	1 linear m/s air flow	20.20	°C/W
θ_{JMA}	2 linear m/s air flow	19.20	°C/W
θ_{JB}		13.05	°C/W
θ_{JC}		6.92	°C/W

Table 48. Thermal Characteristics (BC-289-2)

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	34.5	°C/W
θ_{JMA}	1 linear m/s air flow	31.1	°C/W
θ_{JMA}	2 linear m/s air flow	29.8	°C/W
θ_{JB}		20.3	°C/W
θ_{JC}		8.8	°C/W

289-BALL CSP_BGA BALL ASSIGNMENT

Table 49 lists the CSP_BGA balls by signal mnemonic. Table 50 on Page 65 lists the CSP_BGA by ball number.

Table 49. 289-Ball CSP_BGA Ball Assignment (Alphabetically by Signal)

Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.
$\overline{ABE0}$/SDQM0	AB9	DATA9	P1	GND	N9	VPPOTP	AB11	PH12	M23	V_{DDEXT}	N17	V_{DDMEM}	T7
$\overline{ABE1}$/SDQM1	AC9	DATA10	P2	GND	N10	PF0	A7	PH13	N22	V_{DDEXT}	P17	V_{DDMEM}	U7
ADDR1	AB8	DATA11	R2	GND	N11	PF1	B8	PH14	N23	V_{DDEXT}	R17	V_{DDMEM}	U8
ADDR2	AC8	DATA12	N1	GND	N12	PF2	A8	PH15	P22	V_{DDEXT}	T17	V_{DDMEM}	U9
ADDR3	AB7	DATA13	N2	GND	N13	PF3	B9	PPI_CLK/TMRCLK	A6	V_{DDEXT}	U17	V_{DDMEM}	U10
ADDR4	AC7	DATA14	M2	GND	N14	PF4	B11	PPI_FS1/TMR0	B7	V_{DDINT}	B5	V_{DDMEM}	U11
ADDR5	AC6	DATA15	M1	GND	N15	PF5	B10	$\overline{RESET}$	V22	V_{DDINT}	H8	V_{DDMEM}	U12
ADDR6	AB6	EMU	J2	GND	P9	PF6	B12	RTXI	U23	V_{DDINT}	H9	V_{DDMEM}	U13
ADDR7	AB4	EXT_WAKE0	AC19	GND	P10	PF7	B13	RTXO	V23	V_{DDINT}	H10	V_{DDMEM}	U14
ADDR8	AB5	GND	A1	GND	P11	PF8	B16	SA10	AC10	V_{DDINT}	H11	V_{DDMEM}	U15
ADDR9	AC5	GND	A23	GND	P12	PF9	A20	$\overline{SCAS}$	AC11	V_{DDINT}	H12	V_{DDMEM}	U16
ADDR10	AC4	GND	B6	GND	P13	PF10	B15	SCKE	AB13	V_{DDINT}	H13	V_{DDOTP}	AC12
ADDR11	AB3	GND[1]	G16	GND	P14	PF11	B17	SCL	B22	V_{DDINT}	H14	V_{DDRTC}	W23
ADDR12	AC3	GND	G17	GND	P15	PF12	B18	SDA	C22	V_{DDINT}	H15	V_{DDUSB}	W22
ADDR13	AB2	GND[1]	H17	GND	R9	PF13	B19	$\overline{SMS}$	AC13	V_{DDINT}	H16	V_{DDUSB}	Y23
ADDR14	AC2	GND	H22	GND	R10	PF14	A9	$\overline{SRAS}$	AB12	V_{DDINT}	J8	NC	G23
ADDR15	AA2	GND[1]	J22	GND	R11	PF15	A10	SS/**$\overline{PG}$**	AC20	V_{DDINT}	J16	VR_{OUT}/**EXT_WAKE1**	AC18
ADDR16	W2	GND	J9	GND	R12	PG0	H2	$\overline{SWE}$	AB10	V_{DDINT}	K8	VR_{SEL}/**V_{DDEXT}**	AB22
ADDR17	Y2	GND	J10	GND	R13	PG1	G1	TCK	L1	V_{DDINT}	K16	XTAL	P23
ADDR18	AA1	GND	J11	GND	R14	PG2	H1	TDI	J1	V_{DDINT}	L8		
ADDR19	AB1	GND	J12	GND	R15	PG3	F1	TDO	K1	V_{DDINT}	L16		
$\overline{AMS0}$	AC17	GND	J13	GND	T22	PG4	D1	TMS	L2	V_{DDINT}	M8		
$\overline{AMS1}$	AB16	GND	J14	GND	AC1	PG5	D2	TRST	K2	V_{DDINT}	M16		
$\overline{AMS2}$	AC16	GND	J15	GND	AC23	PG6	C2	USB_DM	AB21	V_{DDINT}	N8		
$\overline{AMS3}$	AB15	GND	K9	NC	A15	PG7	B1	USB_DP	AA22	V_{DDINT}	N16		
$\overline{AOE}$	AC15	GND	K10	NC	A16	PG8	C1	USB_ID	Y22	V_{DDINT}	P8		
ARDY	AC14	GND	K11	NC	A17	PG9	B2	USB_RSET	AC21	V_{DDINT}	P16		
$\overline{ARE}$	AB17	GND	K12	NC	A18	PG10	B4	USB_VBUS	AB20	V_{DDINT}	R8		
$\overline{AWE}$	AB14	GND	K13	NC	A19	PG11	B3	USB_VREF	AC22	V_{DDINT}	R16		
BMODE0	G2	GND	K14	NC	A21	PG12	A2	USB_XI	AB23	V_{DDINT}	T8		
BMODE1	F2	GND	K15	NC	A22	PG13	A3	USB_XO	AA23	V_{DDINT}	T9		
BMODE2	E1	GND	L9	NC	B20	PG14	A4	V_{DDEXT}	G7	V_{DDINT}	T10		
BMODE3	E2	GND	L10	NC	B21	PG15	A5	V_{DDEXT}	G8	V_{DDINT}	T11		
CLKBUF	AB19	GND	L11	NC	B23	PH0	A11	V_{DDEXT}	G9	V_{DDINT}	T12		
CLKIN	R23	GND	L12	NC	C23	PH1	A12	V_{DDEXT}	G10	V_{DDINT}	T13		
CLKOUT	AB18	GND	L13	NC	D22	PH2	A13	V_{DDEXT}	G11	V_{DDINT}	T14		
DATA0	Y1	GND	L14	NC	D23	PH3	B14	V_{DDEXT}	G12	V_{DDINT}	T15		
DATA1	V2	GND	L15	NC	E22	PH4	A14	V_{DDEXT}	G13	V_{DDINT}	T16		
DATA2	W1	GND	M9	NC	E23	PH5	K23	V_{DDEXT}	G14	V_{DDMEM}	J7		
DATA3	U2	GND	M10	NC	F22	PH6	K22	V_{DDEXT}	G15	V_{DDMEM}	K7		
DATA4	V1	GND	M11	NC	F23	PH7	L23	V_{DDEXT}	H7	V_{DDMEM}	L7		
DATA5	U1	GND	M12	NC	G22	PH8	L22	V_{DDEXT}	J17	V_{DDMEM}	M7		
DATA6	T2	GND	M13	NC	H23	PH9	T23	V_{DDEXT}	K17	V_{DDMEM}	N7		
DATA7	T1	GND	M14	NC	J23	PH10	M22	V_{DDEXT}	L17	V_{DDMEM}	P7		
DATA8	R1	GND	M15	$\overline{NMI}$	U22	PH11	R22	V_{DDEXT}	M17	V_{DDMEM}	R7		

NOTE: In this table, **BOLD TYPE** indicates the sole signal/function for that ball on ADSP-BF522/524/526 processors.

[1] For ADSP-BF52xC compatibility, connect this ball to V_{DDEXT}.

Table 50. 289-Ball CSP_BGA Ball Assignment (Numerically by Ball Number)

Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal
A1	GND	B23	NC	H22	GND	L22	PH8	P22	PH15	U22	NMI	AC5	ADDR9
A2	PG12	C1	PG8	H23	NC	L23	PH7	P23	XTAL	U23	RTXI	AC6	ADDR5
A3	PG13	C2	PG6	J1	TDI	M1	DATA15	R1	DATA8	V1	DATA4	AC7	ADDR4
A4	PG14	C22	SDA	J2	EMU	M2	DATA14	R2	DATA11	V2	DATA1	AC8	ADDR2
A5	PG15	C23	NC	J7	V_{DDMEM}	M7	V_{DDMEM}	R7	V_{DDMEM}	V22	$\overline{RESET}$	AC9	$\overline{ABE1}$/SDQM1
A6	PPI_CLK/TMRCLK	D1	PG4	J8	V_{DDINT}	M8	V_{DDINT}	R8	V_{DDINT}	V23	RTXO	AC10	SA10
A7	PF0	D2	PG5	J9	GND	M9	GND	R9	GND	W1	DATA2	AC11	$\overline{SCAS}$
A8	PF2	D22	NC	J10	GND	M10	GND	R10	GND	W2	ADDR16	AC12	V_{DDOTP}
A9	PF14	D23	NC	J11	GND	M11	GND	R11	GND	W22	V_{DDUSB}	AC13	$\overline{SMS}$
A10	PF15	E1	BMODE2	J12	GND	M12	GND	R12	GND	W23	V_{DDRTC}	AC14	ARDY
A11	PH0	E2	BMODE3	J13	GND	M13	GND	R13	GND	Y1	DATA0	AC15	$\overline{AOE}$
A12	PH1	E22	NC	J14	GND	M14	GND	R14	GND	Y2	ADDR17	AC16	$\overline{AMS2}$
A13	PH2	E23	NC	J15	GND	M15	GND	R15	GND	Y22	USB_ID	AC17	$\overline{AMS0}$
A14	PH4	F1	PG3	J16	V_{DDINT}	M16	V_{DDINT}	R16	V_{DDINT}	Y23	V_{DDUSB}	AC18	VR_{OUT}/**EXT_WAKE1**
A15	NC	F2	BMODE1	J17	V_{DDEXT}	M17	V_{DDEXT}	R17	V_{DDEXT}	AA1	ADDR18	AC19	EXT_WAKE0
A16	NC	F22	NC	J22	GND[1]	M22	PH10	R22	PH11	AA2	ADDR15	AC20	SS/**$\overline{PG}$**
A17	NC	F23	NC	J23	NC	M23	PH12	R23	CLKIN	AA22	USB_DP	AC21	USB_RSET
A18	NC	G1	PG1	K1	TDO	N1	DATA12	T1	DATA7	AA23	USB_XO	AC22	USB_VREF
A19	NC	G2	BMODE0	K2	TRST	N2	DATA13	T2	DATA6	AB1	ADDR19	AC23	GND
A20	PF9	G7	V_{DDEXT}	K7	V_{DDMEM}	N7	V_{DDMEM}	T7	V_{DDMEM}	AB2	ADDR13		
A21	NC	G8	V_{DDEXT}	K8	V_{DDINT}	N8	V_{DDINT}	T8	V_{DDINT}	AB3	ADDR11		
A22	NC	G9	V_{DDEXT}	K9	GND	N9	GND	T9	V_{DDINT}	AB4	ADDR7		
A23	GND	G10	V_{DDEXT}	K10	GND	N10	GND	T10	V_{DDINT}	AB5	ADDR8		
B1	PG7	G11	V_{DDEXT}	K11	GND	N11	GND	T11	V_{DDINT}	AB6	ADDR6		
B2	PG9	G12	V_{DDEXT}	K12	GND	N12	GND	T12	V_{DDINT}	AB7	ADDR3		
B3	PG11	G13	V_{DDEXT}	K13	GND	N13	GND	T13	V_{DDINT}	AB8	ADDR1		
B4	PG10	G14	V_{DDEXT}	K14	GND	N14	GND	T14	V_{DDINT}	AB9	$\overline{ABE0}$/SDQM0		
B5	V_{DDINT}	G15	V_{DDEXT}	K15	GND	N15	GND	T15	V_{DDINT}	AB10	$\overline{SWE}$		
B6	GND	G16	GND[1]	K16	V_{DDINT}	N16	V_{DDINT}	T16	V_{DDINT}	AB11	VPPOTP		
B7	PPI_FS1/TMR0	G17	GND	K17	V_{DDEXT}	N17	V_{DDEXT}	T17	V_{DDEXT}	AB12	$\overline{SRAS}$		
B8	PF1	G22	NC	K22	PH6	N22	PH13	T22	GND	AB13	SCKE		
B9	PF3	G23	NC	K23	PH5	N23	PH14	T23	PH9	AB14	$\overline{AWE}$		
B10	PF5	H1	PG2	L1	TCK	P1	DATA9	U1	DATA5	AB15	$\overline{AMS3}$		
B11	PF4	H2	PG0	L2	TMS	P2	DATA10	U2	DATA3	AB16	$\overline{AMS1}$		
B12	PF6	H7	V_{DDEXT}	L7	V_{DDMEM}	P7	V_{DDMEM}	U7	V_{DDMEM}	AB17	$\overline{ARE}$		
B13	PF7	H8	V_{DDINT}	L8	V_{DDINT}	P8	V_{DDINT}	U8	V_{DDMEM}	AB18	CLKOUT		
B14	PH3	H9	V_{DDINT}	L9	GND	P9	GND	U9	V_{DDMEM}	AB19	CLKBUF		
B15	PF10	H10	V_{DDINT}	L10	GND	P10	GND	U10	V_{DDMEM}	AB20	USB_VBUS		
B16	PF8	H11	V_{DDINT}	L11	GND	P11	GND	U11	V_{DDMEM}	AB21	USB_DM		
B17	PF11	H12	V_{DDINT}	L12	GND	P12	GND	U12	V_{DDMEM}	AB22	VR_{SEL}/**V_{DDEXT}**		
B18	PF12	H13	V_{DDINT}	L13	GND	P13	GND	U13	V_{DDMEM}	AB23	USB_XI		
B19	PF13	H14	V_{DDINT}	L14	GND	P14	GND	U14	V_{DDMEM}	AC1	GND		
B20	NC	H15	V_{DDINT}	L15	GND	P15	GND	U15	V_{DDMEM}	AC2	ADDR14		
B21	NC	H16	V_{DDINT}	L16	V_{DDINT}	P16	V_{DDINT}	U16	V_{DDMEM}	AC3	ADDR12		
B22	SCL	H17	GND[1]	L17	V_{DDEXT}	P17	V_{DDEXT}	U17	V_{DDEXT}	AC4	ADDR10		

NOTE: In this table, **BOLD TYPE** indicates the sole signal/function for that ball on ADSP-BF522/524/526 processors.

[1] For ADSP-BF52xC compatibility, connect this ball to V_{DDEXT}.

Figure 66 shows the top view of the BC-289-2 CSP_BGA ball configuration. Figure 65 shows the bottom view of the BC-289-2 CSP_BGA ball configuration.

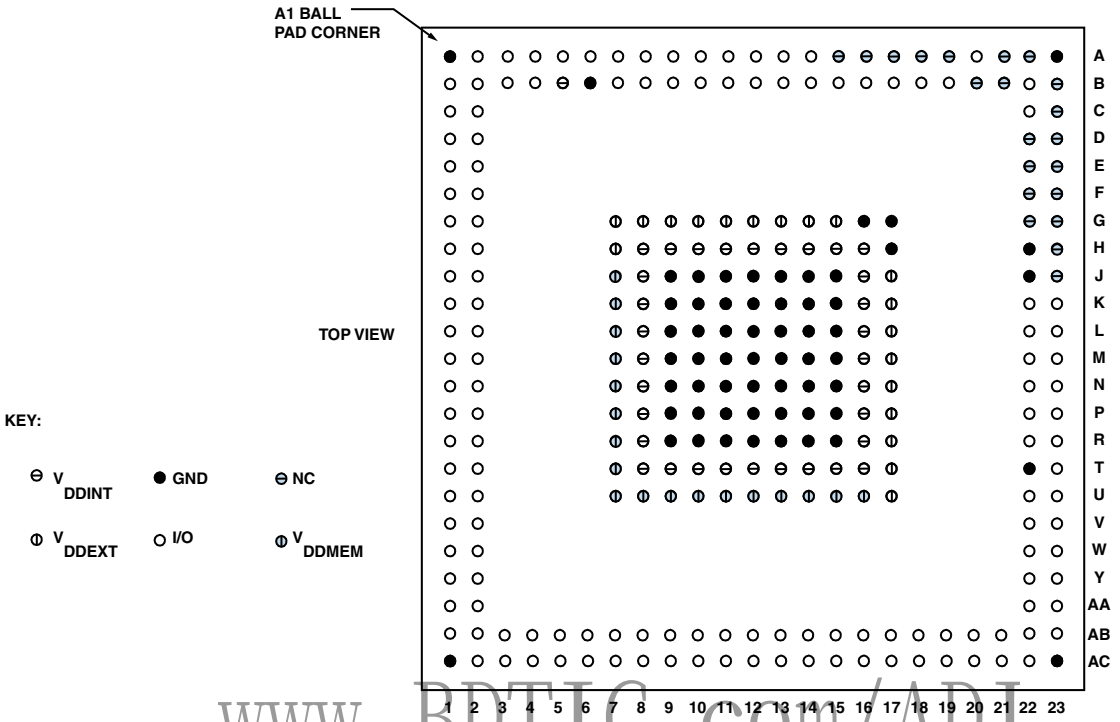


Figure 65. 289-Ball CSP_BGA Ball Configuration (Top View)

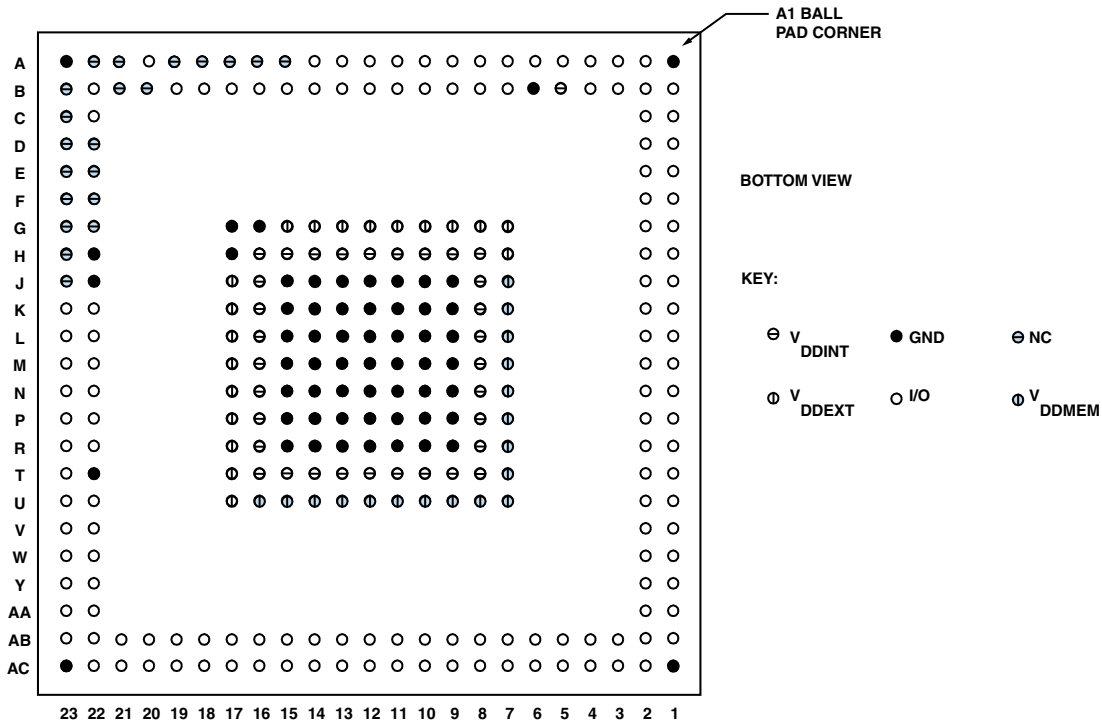


Figure 66. 289-Ball CSP_BGA Ball Configuration (Bottom View)

208-BALL CSP_BGA BALL ASSIGNMENT

Table 51 lists the CSP_BGA balls by signal mnemonic. Table 52 on Page 68 lists the CSP_BGA by ball number.

Table 51. 208-Ball CSP_BGA Ball Assignment (Alphabetically by Signal)

Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.
$\overline{ABE0}$/SDQM0	V19	DATA2	Y7	GND	L12	PG6	M2	SS/**$\overline{PG}$**	G19	V_{DDINT}	P14
$\overline{ABE1}$/SDQM1	V20	DATA3	W7	GND	L13	PG7	L1	$\overline{SWE}$	T20	V_{DDMEM}	L8
ADDR01	W20	DATA4	Y6	GND	M9	PG8	L2	TCK	V2	V_{DDMEM}	M7
ADDR02	W19	DATA5	W6	GND	M10	PG9	K1	TDI	R1	V_{DDMEM}	M8
ADDR03	Y19	DATA6	Y5	GND	M11	PG10	K2	TDO	T1	V_{DDMEM}	N7
ADDR04	W18	DATA7	W5	GND	M12	PG11	J1	TMS	U2	V_{DDMEM}	N8
ADDR05	Y18	DATA8	Y4	GND	M13	PG12	J2	TRST	U1	V_{DDMEM}	P7
ADDR06	W17	DATA9	W4	GND	N9	PG13	H1	USB_DM	F20	V_{DDMEM}	P8
ADDR07	Y17	DATA10	Y3	GND	N10	PG14	H2	USB_DP	E20	V_{DDMEM}	P9
ADDR08	W16	DATA11	W3	GND	N11	PG15	G1	USB_ID	C20	V_{DDMEM}	P10
ADDR09	Y16	DATA12	Y2	GND	N12	PH0	A7	USB_RSET	D20	V_{DDMEM}	P11
ADDR10	W15	DATA13	W2	GND	N13	PH1	B7	USB_VBUS	E19	V_{DDOTP}	R20
ADDR11	Y15	DATA14	W1	GND	Y1	PH2	A8	USB_VREF	H19	V_{DDRTC}	A16
ADDR12	W14	DATA15	V1	GND	Y20	PH3	B8	USB_XI	A19	V_{DDUSB}	D19
ADDR13	Y14	EMU	T2	$\overline{NMI}$	B19	PH4	A9	USB_XO	A18	V_{DDUSB}	G20
ADDR14	W13	EXT_WAKE1	J20	VPPOTP	L19	PH5	B9	V_{DDEXT}	G7	VR_{OUT}/**EXT_WAKE0**	H20
ADDR15	Y13	GND	A1	PF0	F1	PH6	B10	V_{DDEXT}	G8	VR_{SEL}/**V_{DDEXT}**	F19
ADDR16	W12	GND	A17	PF1	E1	PH7	B11	V_{DDEXT}	G9	XTAL	A10
ADDR17	Y12	GND	A20	PF2	E2	PH8	A12	V_{DDEXT}	G10		
ADDR18	W11	GND	B20	PF3	D1	PH9	B12	V_{DDEXT}	G11		
ADDR19	Y11	GND	H9	PF4	D2	PH10	A13	V_{DDEXT}	H7		
$\overline{AMS0}$	J19	GND	H10	PF5	C1	PH11	B13	V_{DDEXT}	H8		
$\overline{AMS1}$	K19	GND	H11	PF6	C2	PH12	B14	V_{DDEXT}	J7		
$\overline{AMS2}$	M19	GND	H12	PF7	B1	PH13	B15	V_{DDEXT}	J8		
$\overline{AMS3}$	L20	GND	H13	PF8	B2	PH14	B16	V_{DDEXT}	K7		
$\overline{AOE}$	N20	GND	J9	PF9	A2	PH15	B17	V_{DDEXT}	K8		
ARDY	P19	GND	J10	PF10	B3	PPI_CLK/TMRCLK	G2	V_{DDEXT}	L7		
$\overline{ARE}$	M20	GND	J11	PF11	A3	PPI_FS1/TMR0	F2	V_{DDINT}	G12		
$\overline{AWE}$	N19	GND	J12	PF12	B5	$\overline{RESET}$	B18	V_{DDINT}	G13		
BMODE0	Y10	GND	J13	PF13	A5	RTXI	A14	V_{DDINT}	G14		
BMODE1	W10	GND	K9	PF14	B6	RTXO	A15	V_{DDINT}	H14		
BMODE2	Y9	GND	K10	PF15	A6	SA10	U19	V_{DDINT}	J14		
BMODE3	W9	GND	K11	PG0	R2	$\overline{SCAS}$	U20	V_{DDINT}	K14		
CLKBUF	C19	GND	K12	PG1	P1	SCKE	P20	V_{DDINT}	L14		
CLKIN	A11	GND	K13	PG2	P2	SCL	A4	V_{DDINT}	M14		
CLKOUT	K20	GND	L9	PG3	N1	SDA	B4	V_{DDINT}	N14		
DATA0	Y8	GND	L10	PG4	N2	$\overline{SMS}$	R19	V_{DDINT}	P12		
DATA1	W8	GND	L11	PG5	M1	$\overline{SRAS}$	T19	V_{DDINT}	P13		

NOTE: In this table, **BOLD TYPE** indicates the sole signal/function for that ball on ADSP-BF522/524/526 processors.

Table 52. 208-Ball CSP_BGA Ball Assignment (Numerically by Ball Number)

Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal
A1	GND	B19	$\overline{NMI}$	H13	GND	L19	VPPOTP	R1	TDI	Y3	DATA10
A2	PF9	B20	GND	H14	V_{DDINT}	L20	$\overline{AMS3}$	R2	PG0	Y4	DATA8
A3	PF11	C1	PF5	H19	USB_VREF	M1	PG5	R19	$\overline{SMS}$	Y5	DATA6
A4	SCL	C2	PF6	H20	VR_{OUT}/**EXT_WAKE0**	M2	PG6	R20	V_{DDOTP}	Y6	DATA4
A5	PF13	C19	CLKBUF	J1	PG11	M7	V_{DDMEM}	T1	TDO	Y7	DATA2
A6	PF15	C20	USB_ID	J2	PG12	M8	V_{DDMEM}	T2	EMU	Y8	DATA0
A7	PH0	D1	PF3	J7	V_{DDEXT}	M9	GND	T19	$\overline{SRAS}$	Y9	BMODE2
A8	PH2	D2	PF4	J8	V_{DDEXT}	M10	GND	T20	$\overline{SWE}$	Y10	BMODE0
A9	PH4	D19	V_{DDUSB}	J9	GND	M11	GND	U1	TRST	Y11	ADDR19
A10	XTAL	D20	USB_RSET	J10	GND	M12	GND	U2	TMS	Y12	ADDR17
A11	CLKIN	E1	PF1	J11	GND	M13	GND	U19	SA10	Y13	ADDR15
A12	PH8	E2	PF2	J12	GND	M14	V_{DDINT}	U20	$\overline{SCAS}$	Y14	ADDR13
A13	PH10	E19	USB_VBUS	J13	GND	M19	$\overline{AMS2}$	V1	DATA15	Y15	ADDR11
A14	RTXI	E20	USB_DP	J14	V_{DDINT}	M20	$\overline{ARE}$	V2	TCK	Y16	ADDR9
A15	RTXO	F1	PF0	J19	$\overline{AMS0}$	N1	PG3	V19	$\overline{ABE0}$/SDQM0	Y17	ADDR7
A16	V_{DDRTC}	F2	PPI_FS1/TMR0	J20	EXT_WAKE1	N2	PG4	V20	$\overline{ABE1}$/SDQM1	Y18	ADDR5
A17	GND	F19	VR_{SEL}/**V_{DDEXT}**	K1	PG9	N7	V_{DDMEM}	W1	DATA14	Y19	ADDR3
A18	USB_XO	F20	USB_DM	K2	PG10	N8	V_{DDMEM}	W2	DATA13	Y20	GND
A19	USB_XI	G1	PG15	K7	V_{DDEXT}	N9	GND	W3	DATA11		
A20	GND	G2	PPI_CLK/TMRCLK	K8	V_{DDEXT}	N10	GND	W4	DATA9		
B1	PF7	G7	V_{DDEXT}	K9	GND	N11	GND	W5	DATA7		
B2	PF8	G8	V_{DDEXT}	K10	GND	N12	GND	W6	DATA5		
B3	PF10	G9	V_{DDEXT}	K11	GND	N13	GND	W7	DATA3		
B4	SDA	G10	V_{DDEXT}	K12	GND	N14	V_{DDINT}	W8	DATA1		
B5	PF12	G11	V_{DDEXT}	K13	GND	N19	$\overline{AWE}$	W9	BMODE3		
B6	PF14	G12	V_{DDINT}	K14	V_{DDINT}	N20	$\overline{AOE}$	W10	BMODE1		
B7	PH1	G13	V_{DDINT}	K19	$\overline{AMS1}$	P1	PG1	W11	ADDR18		
B8	PH3	G14	V_{DDINT}	K20	CLKOUT	P2	PG2	W12	ADDR16		
B9	PH5	G19	SS/**$\overline{PG}$**	L1	PG7	P7	V_{DDMEM}	W13	ADDR14		
B10	PH6	G20	V_{DDUSB}	L2	PG8	P8	V_{DDMEM}	W14	ADDR12		
B11	PH7	H1	PG13	L7	V_{DDEXT}	P9	V_{DDMEM}	W15	ADDR10		
B12	PH9	H2	PG14	L8	V_{DDMEM}	P10	V_{DDMEM}	W16	ADDR8		
B13	PH11	H7	V_{DDEXT}	L9	GND	P11	V_{DDMEM}	W17	ADDR6		
B14	PH12	H8	V_{DDEXT}	L10	GND	P12	V_{DDINT}	W18	ADDR4		
B15	PH13	H9	GND	L11	GND	P13	V_{DDINT}	W19	ADDR2		
B16	PH14	H10	GND	L12	GND	P14	V_{DDINT}	W20	ADDR1		
B17	PH15	H11	GND	L13	GND	P19	ARDY	Y1	GND		
B18	$\overline{RESET}$	H12	GND	L14	V_{DDINT}	P20	SCKE	Y2	DATA12		

NOTE: In this table, **BOLD TYPE** indicates the sole signal/function for that ball on ADSP-BF522/524/526 processors.

Figure 67 shows the top view of the CSP_BGA ball configuration. Figure 68 shows the bottom view of the CSP_BGA ball configuration.

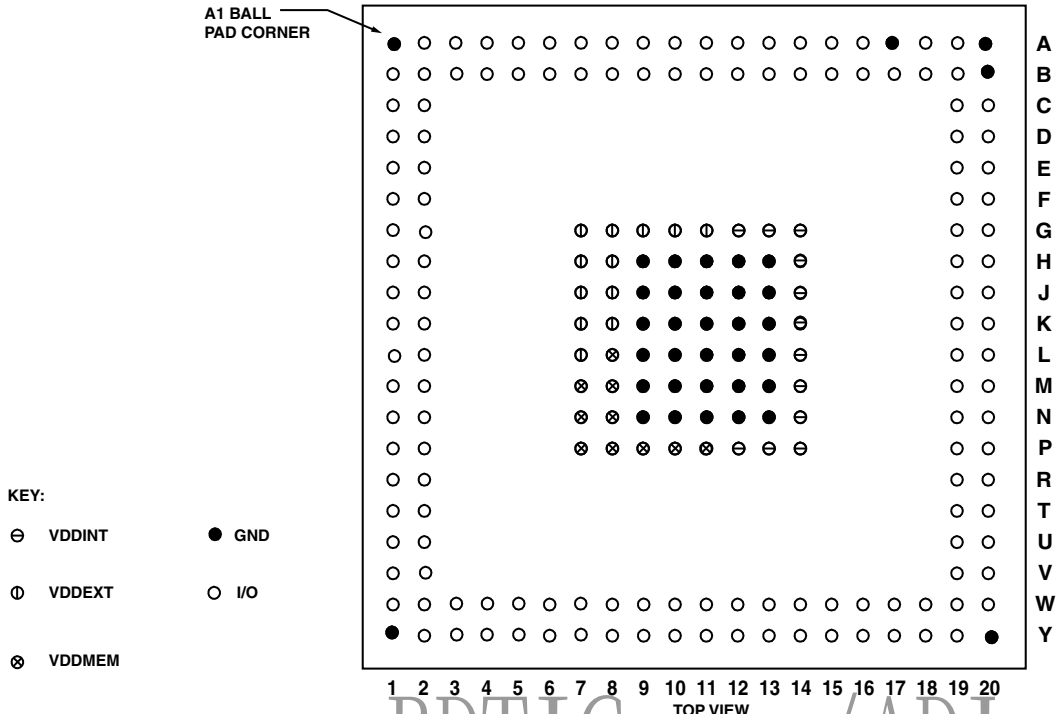


Figure 67. 208-Ball CSP_BGA Ball Configuration (Top View)

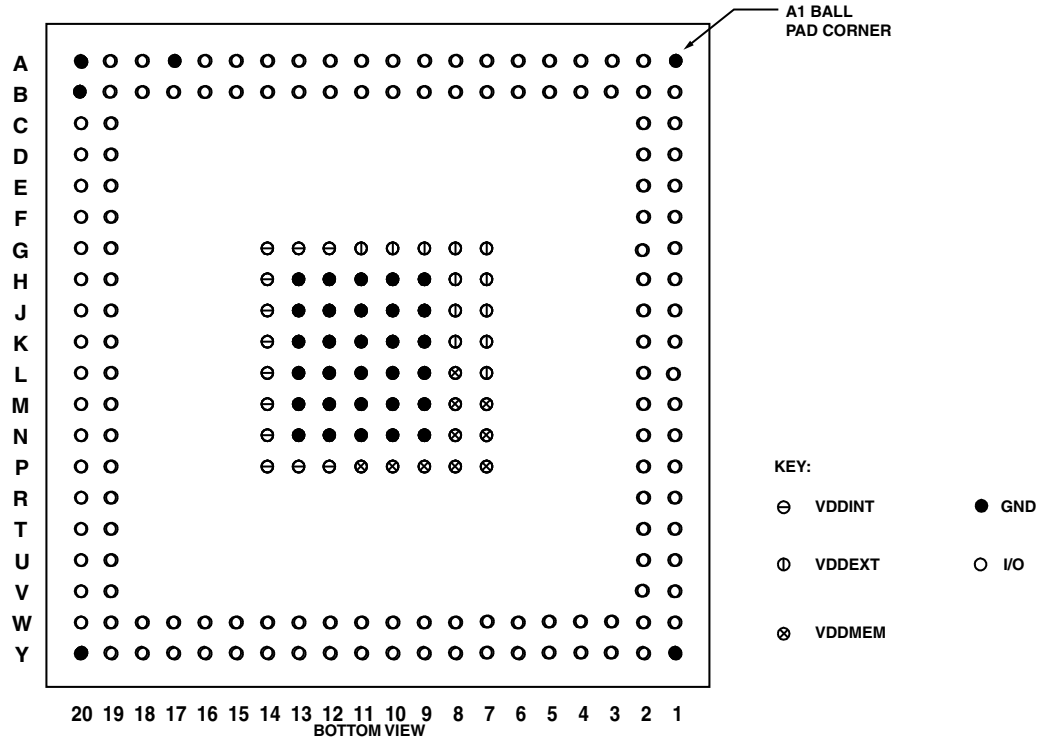


Figure 68. 208-Ball CSP_BGA Ball Configuration (Bottom View)

OUTLINE DIMENSIONS

Dimensions in Figure 69, 289-Ball CSP_BGA (BC-289-2) are shown in millimeters.

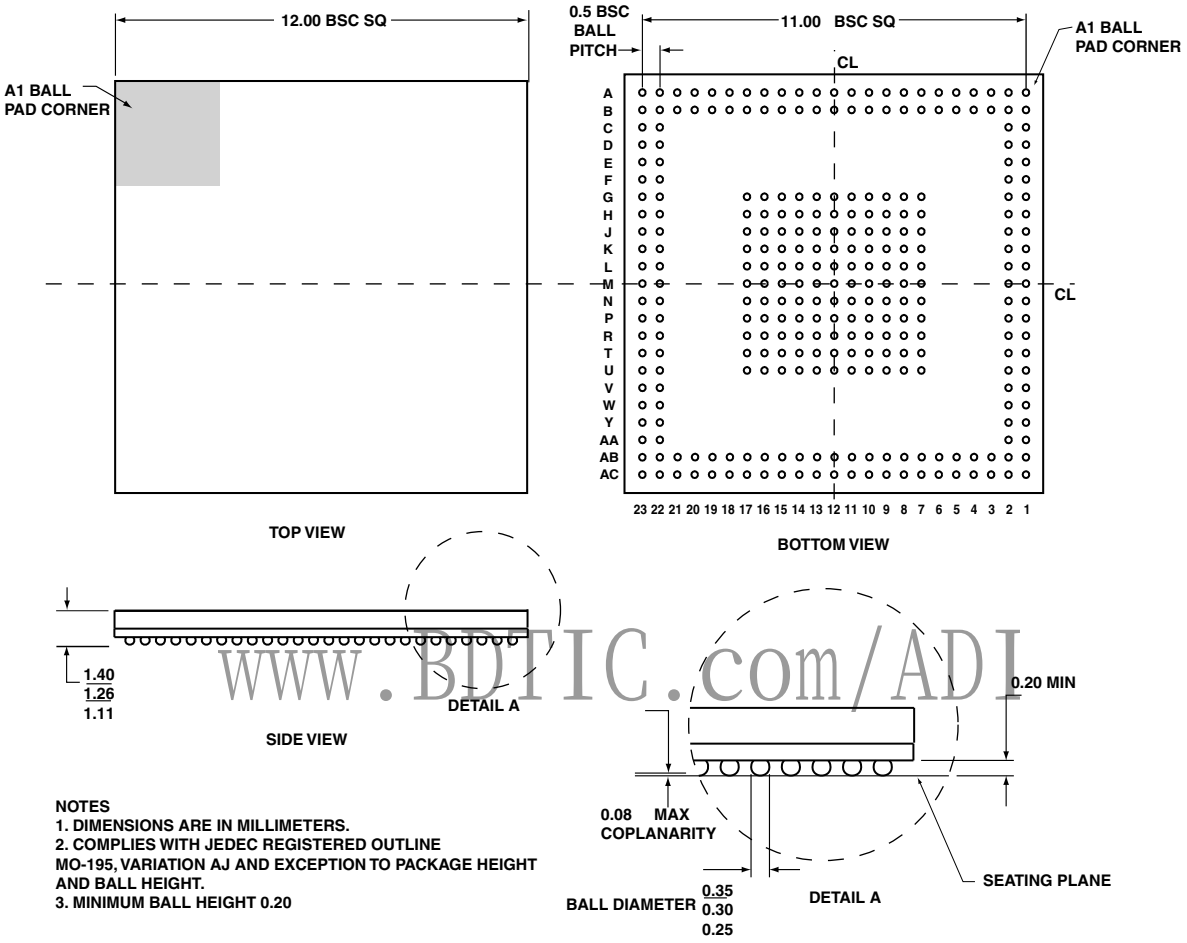


Figure 69. 289-Ball CSP_BGA (BC-289-2)

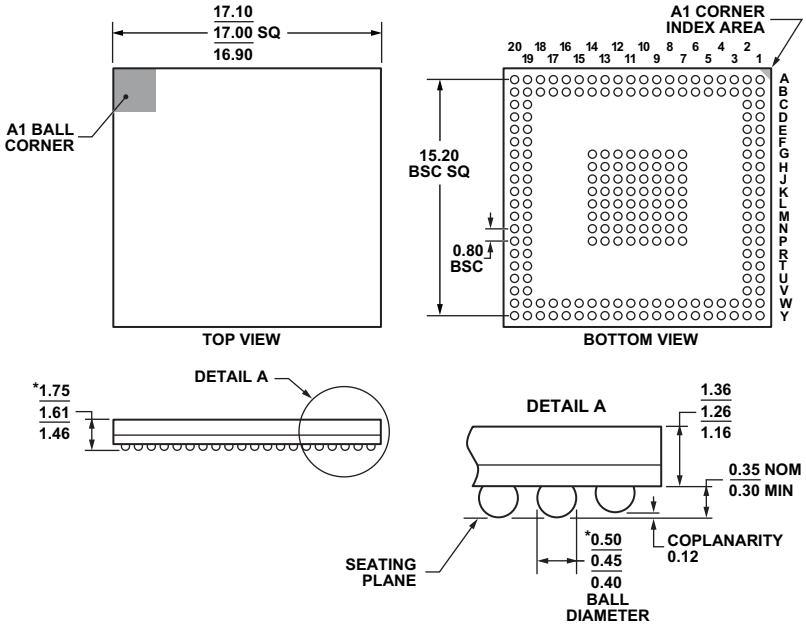


Figure 70. 208-Ball CSP_BGA (BC-208-2)

SURFACE MOUNT DESIGN

Table 53 is provided as an aide to PCB design. For industry-standard design recommendations, refer to IPC-7351, Generic Requirements for Surface Mount Design and Land Pattern Standard.

Table 53. Surface Mount Design Supplement

Package	Ball Attach Type	Solder Mask Opening	Ball Pad Size
289-Ball CSP_BGA	Solder Mask Defined	0.26 mm diameter	0.35 mm diameter
208-Ball CSP_BGA	Solder Mask Defined	0.40 mm diameter	0.50 mm diameter

ORDERING GUIDE

Table 54. ADSP-BF522/524/526 Processors

Model	Temperature Range[1]	Package Description	Package Option	Instruction Rate (Max)	Operating Voltage (Nom)
ADSP-BF526KBCZ-4X	0°C to +70°C	289-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-289-2	400 MHz	tbd V internal, 1.8 V, 2.5 V, or 3.3 V I/O
ADSP-BF526BBCZ-4AX	–40°C to +85°C	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-208-2	400 MHz	tbd V internal, 1.8 V, 2.5 V, or 3.3 V I/O
ADSP-BF526BBCZ-3AX	–40°C to +85°C	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-208-2	400 MHz	tbd V internal, 1.8 V, 2.5 V, or 3.3 V I/O

[1] Referenced temperature is ambient temperature.

Table 55. ADSP-BF523/525/527 Processors

Model	Temperature Range[1]	Package Description	Package Option	Instruction Rate (Max)	Operating Voltage (Nom)
ADSP-BF527KBCZ-6X	0°C to +70°C	289-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-289-2	600 MHz	1.2 V internal[2], 1.8 V, 2.5 V, or 3.3 V I/O
ADSP-BF527KBCZ-6AX	0°C to +70°C	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-208-2	600 MHz	1.2 V internal[2], 1.8 V, 2.5 V, or 3.3 V I/O
ADSP-BF527BBCZ-5AX	–40°C to +85°C	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA)	BC-208-2	533 MHz	1.15 V internal[2], 1.8 V, 2.5 V, or 3.3 V I/O

[1] Referenced temperature is ambient temperature.
[2] This is the voltage required to run at the maximum instruction rate. Lesser frequencies may require lower operating voltages. Please see Table 12 and Table 15 for details.

PR06675-0-9/08(PrE)