Blackfin Embedded Processor

ADSP-BF539/ADSP-BF539F

FEATURES

- Up to 533 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
- Wide range of operating voltages; see Operating Conditions on Page 26
- Programmable on-chip voltage regulator
- 316-ball Pb-free CSP_BGA package

MEMORY

- 148K bytes of on-chip memory
  - 16K bytes of instruction SRAM/cache
  - 64K bytes of instruction SRAM
  - 32K bytes of data SRAM
  - 32K bytes of data SRAM/cache
  - 4K bytes of scratchpad SRAM
- 512K × 16-bit or 256K × 16-bit flash memory (ADSP-BF539F only)
- Memory management unit providing memory protection
- External memory controller with glueless support for SDRAM, SRAM, flash, and ROM
- Flexible memory booting options from SPI and external memory

PERIPHERALS

- Parallel peripheral interface (PPI), supporting ITU-R 656 video data formats
- 4 dual-channel, full-duplex synchronous serial ports, supporting 16 stereo I²S channels
- 2 DMA controllers supporting 26 peripheral DMAs
- 4 memory-to-memory DMAs
- Controller area network (CAN) 2.0B controller
- Media transceiver (MXVR) for connection to a MOST network
- 3 SPI-compatible ports
- Three 32-bit timer/counters with PWM support
- 3 UARTs with support for IrDA
- 2 TWI controllers compatible with I²C industry standard
- Up to 38 general-purpose I/O pins (GPIO)
- Up to 16 general-purpose flag pins (GPF)
- Real-time clock, watchdog timer, and 32-bit core timer
- On-chip PLL capable of 0.5× to 64× frequency multiplication
- Debug/JTAG interface

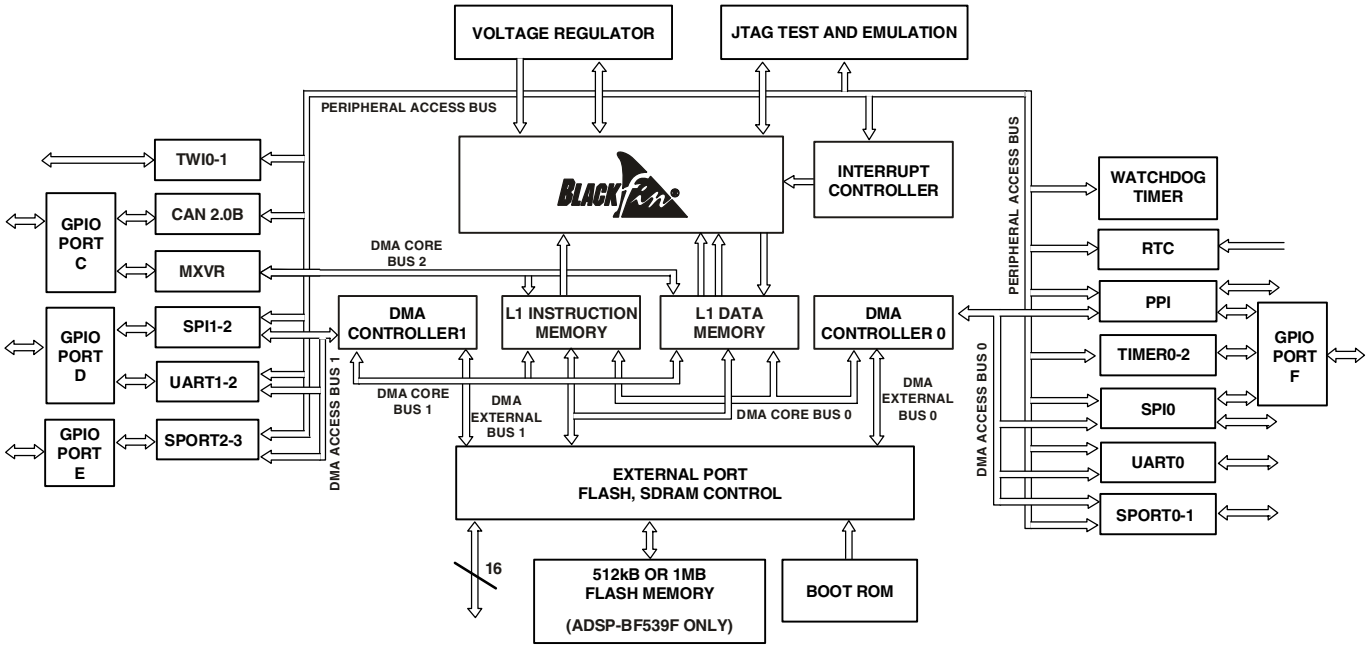


Figure 1. Functional Block Diagram

Blackfin and the Blackfin logo are registered trademarks of Analog Devices, Inc.

Rev. B

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

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 **©2008 Analog Devices, Inc. All rights reserved.**

TABLE OF CONTENTS

Features 1
Memory 1
Peripherals 1
Table Of Contents 2
Revision History 2
General Description 3
Low Power Architecture 3
System Integration 3
ADSP-BF539/ADSP-BF539F Processor Peripherals 3
Blackfin Processor Core 4
Memory Architecture 5
DMA Controllers 8
Real-Time Clock 9
Watchdog Timer 9
Timers 9
Serial Ports (SPORTs) 10
Serial Peripheral Interface (SPI) Ports 10
2-Wire Interface 10
UART Ports 10
Programmable I/O Pins 11
Parallel Peripheral Interface 12
Controller Area Network (CAN) Interface 13
Media Transceiver MAC layer (MXVR) 13
Dynamic Power Management 13
Voltage Regulation 15
Clock Signals 15
Booting Modes 16
Instruction Set Description 17
Development Tools 17
Designing an Emulator Compatible Processor Board 18
Example Connections and Layout Considerations 18
MXVR Board Layout Guidelines 18
Voltage Regulator Layout Guidelines 20
Related Documents 20
Pin Descriptions 21
Specifications 26
Operating Conditions 26
Electrical Characteristics 27
Absolute Maximum Ratings 30
ESD Sensitivity 30
Package Information 30
Timing Specifications 31
Output Drive Currents 51
Test Conditions 52
Thermal Characteristics 55
316-Ball CSP_BGA Ball Assignment 56
Outline Dimensions 59
Surface-Mount Design 60
Automotive Products 60
Ordering Guide 60

REVISION HISTORY

11/08—Rev. A to Rev. B

Corrected document errata associated with Pin Descriptions 21

Corrected document errata associated with $I_{DDHIBERNATE}$ in Electrical Characteristics 27

Removed the Power Dissipation section. See *Estimating Power for the ADSP-BF538/BF539 Blackfin Processors (EE-298)* and Table 14, Table 15, and Table 16 28

Revised SPI master timing specifications and diagram. See Serial Peripheral Interface Ports—Master Timing 45

Revised SPI slave timing specifications and diagram. See Serial Peripheral Interface Ports—Slave Timing 46

Replaced package drawing. See Outline Dimensions 59

Revised Automotive Products 60

GENERAL DESCRIPTION

The ADSP-BF539/ADSP-BF539F processors are members of the Blackfin® family of products, incorporating the Analog Devices, Inc./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-BF539/ADSP-BF539F processors are completely code compatible with other Blackfin processors, differing only with respect to performance, peripherals, and on-chip memory. These features are shown in Table 1.

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.

Table 1. Processor Features

Feature	ADSP-BF539	ADSP-BF539F4	ADSP-BF539F8
SPORTs	4	4	4
UARTs	3	3	3
SPI	3	3	3
TWI	2	2	2
CAN	1	1	1
MXVR	1	1	1
PPI	1	1	1
Instruction SRAM/Cache	16K bytes	16K bytes	16K bytes
Instruction SRAM	64K bytes	64K bytes	64K bytes
Data SRAM/Cache	32K bytes	32K bytes	32K bytes
Data SRAM	32K bytes	32K bytes	32K bytes
Scratchpad	4K bytes	4K bytes	4K bytes
Flash	Not applicable	256K × 16-bit	512K × 16-bit
Maximum Speed Grade	533 MHz 1066 MMACS	533 MHz 1066 MMACS	533 MHz 1066 MMACS
Package Option	BC-316	BC-316	BC-316

LOW POWER ARCHITECTURE

Blackfin processors provide world class power management and performance. Blackfin processors are designed in a low power and low voltage design methodology and feature dynamic power management, the ability to vary both the voltage and frequency of operation to significantly lower overall power consumption. Varying the voltage and frequency can result in a substantial reduction in power consumption, compared with simply varying the frequency of operation. This translates into longer battery life and lower heat dissipation.

SYSTEM INTEGRATION

The ADSP-BF539/ADSP-BF539F processors are highly integrated system-on-a-chip solutions for the next generation of industrial and automotive applications including audio and video signal processing. By combining advanced memory configurations, such as on-chip flash memory, with industry-standard interfaces with a high performance signal processing core, users can develop cost-effective solutions quickly without the need for costly external components. The system peripherals include a MOST Network Media Transceiver (MXVR), three UART ports, three SPI ports, four serial ports (SPORT), one CAN interface, two 2-wire interfaces (TWI), four general-purpose timers (three with PWM capability), a real-time clock, a watchdog timer, a parallel peripheral interface, general-purpose I/O, and general-purpose flag pins.

ADSP-BF539/ADSP-BF539F PROCESSOR PERIPHERALS

The ADSP-BF539/ADSP-BF539F 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 Figure 1 on Page 1). The general-purpose peripherals include functions such as UART, timers with PWM (pulse-width modulation) and pulse measurement capability, general-purpose flag I/O pins, a real-time clock, and a watchdog timer. This set of functions satisfies a wide variety of typical system support needs and is augmented by the system expansion capabilities of the device. In addition to these general-purpose peripherals, the processors contain high speed serial and parallel ports for interfacing to a variety of audio, video, and modem codec functions. An MXVR transceiver transmits and receives audio and video data and control information on a MOST automotive multimedia network. A CAN 2.0B controller is provided for automotive control networks. An interrupt controller manages interrupts from the on-chip peripherals or external sources. And power management control functions tailor the performance and power characteristics of the processor and system to many application scenarios.

All of the peripherals, GPIO, CAN, TWI, real-time clock, and timers, are supported by a flexible DMA structure. There are also four 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-BF539/ADSP-BF539F 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 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-bit, 16-bit, 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.

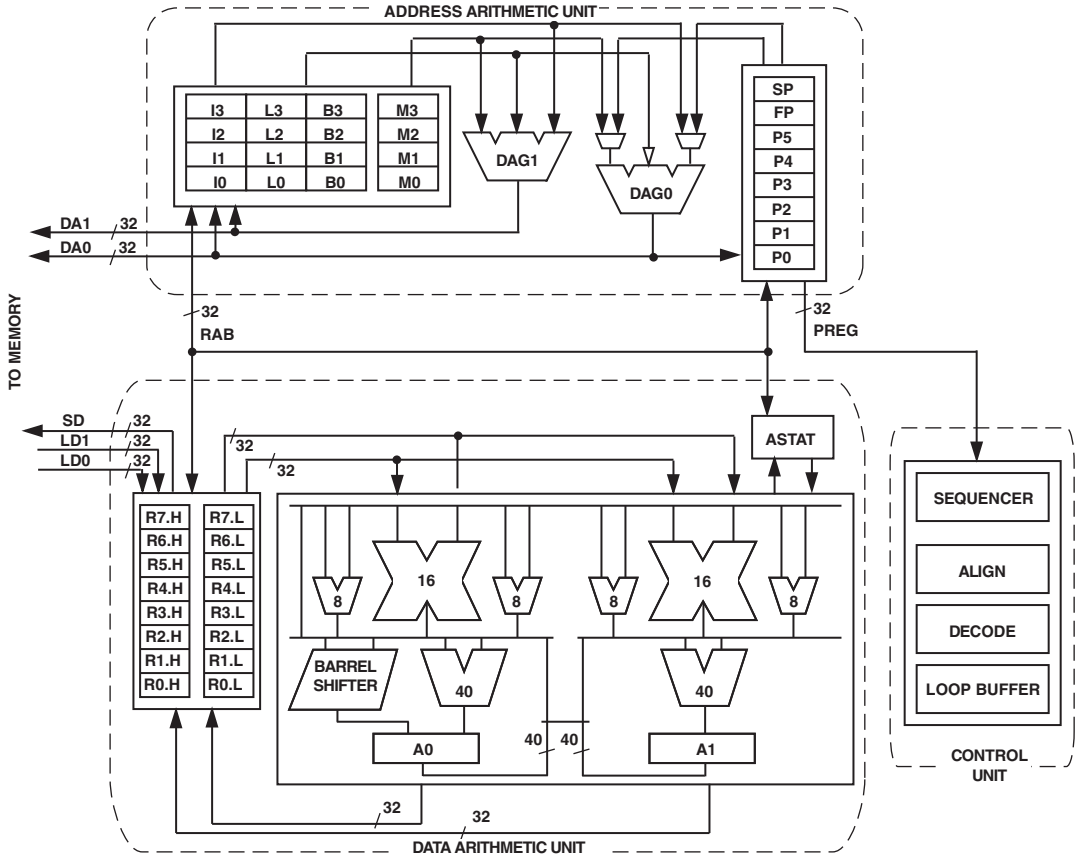


Figure 2. Blackfin Processor Core

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). By also using the second ALU, 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 can 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 ADSP-BF539/ADSP-BF539F processors view 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 L1 memory system is the primary 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 132M bytes of physical memory.

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

Internal (On-Chip) Memory

The ADSP-BF539/ADSP-BF539F processor has three blocks of on-chip memory, providing high bandwidth access to the core.

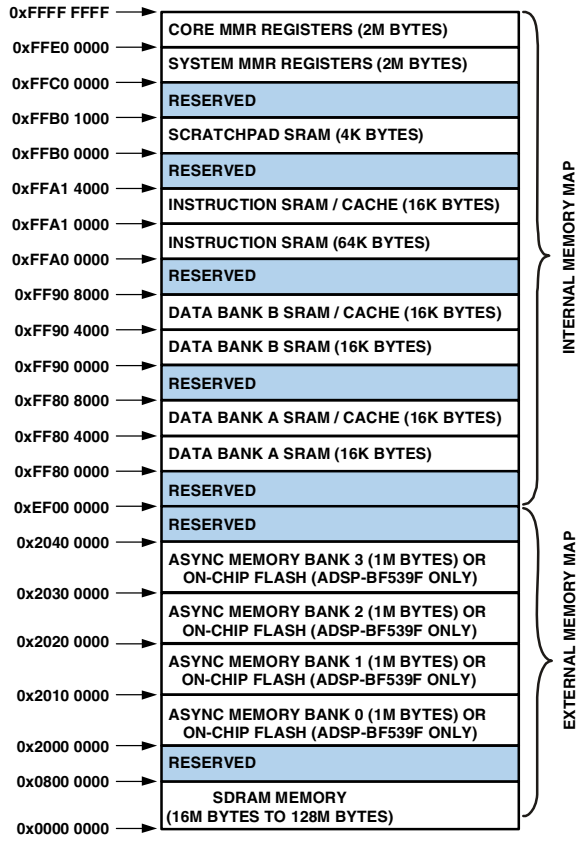


Figure 3. ADSP-BF539/ADSP-BF539F Internal/External Memory Map

The first is the L1 instruction memory, consisting of 80K 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 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 scratch pad 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 PC133-compliant SDRAM controller can be programmed to interface to up to 128M bytes of SDRAM. The SDRAM controller allows one row to be open for each internal SDRAM bank, for up to four internal SDRAM banks, improving overall system 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 will only be contiguous if each is fully populated with 1M byte of memory.

Flash Memory (ADSP-BF539F Only)

The ADSP-BF539F4 and ADSP-BF539F8 processors contain a separate flash die, connected to the EBIU bus, within the package of the ADSP-BF539F processors. Figure 4 shows how the flash memory die and Blackfin processor die are connected.

The ADSP-BF539F4 contains a 4M bit (256K × 16-bit) bottom boot sector Spansion S29AL004D known good die flash memory†. The ADSP-BF539F8 contains an 8M bit (512K × 16-bit) bottom boot sector Spansion S29AL008D known good die flash memory. Features include the following:

- Access times as fast as 70 ns (EBIU registers be set appropriately)
- Sector protection
- One million write cycles per sector
- 20 year data retention

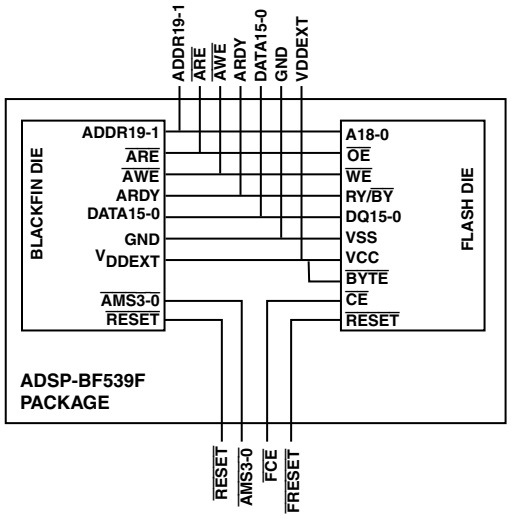


Figure 4. Internal Connection of Flash Memory (ADSP-BF539Fx)

The Blackfin processor connects to the flash memory die with address, data, chip enable, write enable, and output enable controls as if it were an external memory device.

The flash chip enable pin $\overline{\text{FCE}}$ must be connected to $\overline{\text{AMS0}}$ or $\overline{\text{AMS3–1}}$ through a printed circuit board trace. When connected to $\overline{\text{AMS0}}$, the Blackfin processor can boot from the flash die. When connected to $\overline{\text{AMS3–1}}$, the flash memory will appear as nonvolatile memory in the processor memory map, shown in Figure 3.

Flash Memory Programming

The ADSP-BF539F4 and ADSP-BF539F8 flash memory can be programmed before or after mounting on the printed circuit board.

To program the flash prior to mounting on the printed circuit board, use a hardware programming tool that can provide the data, address, and control stimuli to the flash die through the external pins on the package. During this programming, V_{DDEXT} and GND must be provided to the package and the Blackfin must be held in reset with bus request ($\overline{\text{BR}}$) asserted and a CLKIN provided.

The VisualDSP++‡ tools can be used to program the flash memory after the device is mounted on a printed circuit board.

Flash Memory Sector Protection

To use the sector protection feature, a high voltage (+12 V nominal) must be applied to the flash $\overline{\text{FRESET}}$ pin. Refer to the flash data sheet for details.

I/O Memory Space

Blackfin processors do 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 of which contains the control MMRs for all core functions, and the other of 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 ADSP-BF539/ADSP-BF539F processor contains a small boot kernel, which configures the appropriate peripheral for booting. If the ADSP-BF539/ADSP-BF539F 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 16.

Event Handling

The event controller on the ADSP-BF539/ADSP-BF539F processor handles all asynchronous and synchronous events to the processor. The ADSP-BF539/ADSP-BF539F 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.
- Reset – This event resets the processor.

† Refer to the Spansion website for the appropriate data sheets.

‡ VisualDSP++ is a registered trademark of Analog Devices, Inc.

- 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 (i.e., the exception will be 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 pins, 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 ADSP-BF539/ADSP-BF539F processor's 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 ADSP-BF539/ADSP-BF539F processors. Table 2 describes the inputs to the CEC, identifies their names in the event vector table (EVT), and lists their priorities.

System Interrupt Controller (SIC)

The system interrupt controller (SIC) 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 ADSP-BF539/ADSP-BF539F processors provide 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.

Event Control

The ADSP-BF539/ADSP-BF539F processors provide the user with 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) – The ILAT register indicates when events have been latched. The appropriate bit is set when the processor has latched the event and is cleared when the event has been accepted into the system. This register is updated automatically by the controller, but it can also be written to clear (cancel) latched events. This register may be read while in supervisor mode and may only be written while in supervisor mode when the corresponding IMASK bit is cleared.
- CEC interrupt mask register (IMASK) – The IMASK register controls the masking and unmasking of individual events. When a bit is set in the IMASK register, that event is unmasked and will be 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 can be latched in the ILAT register. This register can be read or written while in supervisor mode. 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 whether the event is currently active or nested at some level. This register is updated automatically by the controller but can be read while in supervisor mode.

The SIC allows further control of event processing by providing three 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 8.

- SIC interrupt mask registers (SIC_IMASKx) – These registers control the masking and unmasking of each peripheral interrupt event. When a bit is set in these registers, that peripheral event is unmasked and will be processed by the system when asserted. A cleared bit in these registers 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 that the peripheral is asserting the interrupt, and a cleared bit indicates that the peripheral is not asserting the event.
- SIC interrupt wake-up 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 13.)

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 will recognize and queue 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.

Table 2. Core Event Controller (CEC)

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

Table 3. System and Core Event Mapping

Event Source	Core Event Name
PLL Wake-Up Interrupt	IVG7
DMA Controller 0 Error	IVG7
DMA Controller 1 Error	IVG7
PPI Error Interrupt	IVG7
SPORT0 Error Interrupt	IVG7
SPORT1 Error Interrupt	IVG7
SPORT2 Error Interrupt	IVG7
SPORT3 Error Interrupt	IVG7
MXVR Synchronous Data Interrupt	IVG7
SPI0 Error Interrupt	IVG7
SPI1 Error Interrupt	IVG7
SPI2 Error Interrupt	IVG7
UART0 Error Interrupt	IVG7
UART1 Error Interrupt	IVG7
UART2 Error Interrupt	IVG7
CAN Error Interrupt	IVG7
Real-Time Clock Interrupt	IVG8
DMA0 Interrupt (PPI)	IVG8
DMA1 Interrupt (SPORT0 Rx)	IVG9
DMA2 Interrupt (SPORT0 Tx)	IVG9
DMA3 Interrupt (SPORT1 Rx)	IVG9
DMA4 Interrupt (SPORT1 Tx)	IVG9
DMA8 Interrupt (SPORT2 Rx)	IVG9
DMA9 Interrupt (SPORT2 Tx)	IVG9
DMA10 Interrupt (SPORT3 Rx)	IVG9
DMA11 Interrupt (SPORT3 Tx)	IVG9
DMA5 Interrupt (SPI0)	IVG10
DMA14 Interrupt (SPI1)	IVG10
DMA15 Interrupt (SPI2)	IVG10
DMA6 Interrupt (UART0 Rx)	IVG10
DMA7 Interrupt (UART0 Tx)	IVG10
DMA16 Interrupt (UART1 Rx)	IVG10
DMA17 Interrupt (UART1 Tx)	IVG10
DMA18 Interrupt (UART2 Rx)	IVG10
DMA19 Interrupt (UART2 Tx)	IVG10
Timer0, Timer1, Timer2 Interrupts	IVG11
TWI0 Interrupt	IVG11
TWI1 Interrupt	IVG11
CAN Receive Interrupt	IVG11
CAN Transmit Interrupt	IVG11
MXVR Status Interrupt	IVG11
MXVR Control Message Interrupt	IVG11
MXVR Asynchronous Packet Interrupt	IVG11
Programmable Flags Interrupts	IVG12
MDMA0 Stream 0 Interrupt	IVG13
MDMA0 Stream 1 Interrupt	IVG13
MDMA1 Stream 0 Interrupt	IVG13
MDMA1 Stream 1 Interrupt	IVG13
Software Watchdog Timer	IVG13

DMA CONTROLLERS

The ADSP-BF539/ADSP-BF539F processor has multiple, independent DMA controllers that support automated data transfers with minimal overhead for the processor core. DMA transfers can occur between the ADSP-BF539/ADSP-BF539F processor 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 SPORTs, SPI ports, UARTs, and PPI. Each individual DMA capable peripheral has at least one dedicated DMA channel. In addition, the MXVR peripheral has its own dedicated DMA controller, which supports its own unique set of operating modes.

The DMA controllers support both 1-dimensional (1-D) and 2-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 deinterleaved on the fly.

Examples of DMA types supported by the processor's 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 four memory DMA channels provided for transfers between the various memories of the ADSP-BF539/ADSP-BF539F 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.

REAL-TIME CLOCK

The ADSP-BF539/ADSP-BF539F processor 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 ADSP-BF539/ADSP-BF539F processors. 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.

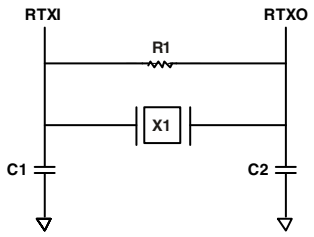
The 32.768 kHz input clock frequency is divided down to a 1 Hz signal by a prescaler. The counter function of the timer consists of four counters: a 60-second counter, a 60-minute counter, a 24-hour counter, and an 32,768-day counter.

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 wake-up event. Additionally, an RTC wake-up event can wake up the processor from deep sleep mode, and wake up the on-chip internal voltage regulator from a powered down state.

Connect RTC pins RTXI and RTXO with external components as shown in Figure 5.



SUGGESTED COMPONENTS:
ECLIPTEK EC38J (THROUGH-HOLE PACKAGE)
EPSON MC405 12 pF LOAD (SURFACE-MOUNT PACKAGE)
C1 = 22pF
C2 = 22pF
R1 = 10MΩ

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 3pF.

Figure 5. External Components for RTC

WATCHDOG TIMER

The ADSP-BF539/ADSP-BF539F processors include 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, and 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 ADSP-BF539/ADSP-BF539F 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 four general-purpose programmable timer units in the ADSP-BF539/ADSP-BF539F processors. Three 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 peri-

ods of external events. These timers can be synchronized to an external clock input to the PF1 pin (TACLK), an external clock input to the PPI_CLK pin (TMRCLK), or to the internal SCLK.

The timer units can be used in conjunction with UART0 to measure the width of the pulses in the data stream to provide an auto-baud detect function for a serial channel.

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 three general-purpose programmable timers, a fourth 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.

SERIAL PORTS (SPORTs)

The ADSP-BF539/ADSP-BF539F processors incorporate four dual-channel synchronous serial ports 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 16 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 bits 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) PORTS

The ADSP-BF539/ADSP-BF539F processors incorporate three SPI-compatible ports that enable 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, MOSIx, and master input-slave output, MISOx) and a clock pin (serial clock, SCKx). An SPI chip select input pin ($\overline{\text{SPIxSS}}$) lets other SPI devices select the processor. For SPI0, seven SPI chip select output pins ($\overline{\text{SPI0SEL7–1}}$) let the processor select other SPI devices. SPI1 and SPI2 each have a single SPI chip select output pin ($\overline{\text{SPI1SEL1}}$ and $\overline{\text{SPI2SEL1}}$) for SPI point-to-point communication. Each of the SPI select pins is a reconfigured GPIO pin. Using these pins, the SPI ports provide a full-duplex, synchronous serial interface, which supports both master/slave modes and multimaster environments.

The SPI ports' baud rate and clock phase/polarities are programmable, and they each have an integrated DMA controller, configurable to support transmit or receive data streams. Each SPI DMA controller can only service unidirectional accesses at any given time.

The SPI port clock rate is calculated as:

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

where the 16-bit *SPIx_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.

2-WIRE INTERFACE

The ADSP-BF539/ADSP-BF539F processors incorporate two 2-wire interface (TWI) modules that are compatible with the Philips Inter-IC bus standard. The TWI modules offer the capabilities of simultaneous master and slave operation, support for 7-bit addressing, and multimedia data arbitration. The TWI also includes master clock synchronization and support for clock low extension.

The TWI interface uses two pins for transferring clock (SCLx) and data (SDAx) and supports the protocol at speeds up to 400 kbps.

The TWI interface pins are compatible with 5 V logic levels.

UART PORTS

The ADSP-BF539/ADSP-BF539F processor incorporates three full-duplex universal asynchronous receiver/transmitter (UART) ports, which are fully compatible with PC standard UARTs. The UART ports provide a simplified UART interface to other peripherals or hosts, supporting full-duplex, DMA supported, asynchronous transfers of serial data. The UART ports include support for 5 data bits to 8 data bits, 1 stop bit or 2 stop bits, and none, even, or odd parity. The UART ports support 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. Each 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 7 bits to 12 bits per frame.
- Both transmit and receive operations can be configured to generate maskable interrupts to the processor.

Each 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 UARTx_DLH register (most significant 8 bits) and UARTx_DLL register (least significant 8 bits).

In conjunction with the general-purpose timer functions, autobaud detection is supported on UART0.

The capabilities of the UARTs are further extended with support for the Infrared Data Association (IrDA®) Serial Infrared Physical Layer Link Specification (SIR) protocol.

PROGRAMMABLE I/O PINS

The ADSP-BF539/ADSP-BF539F processor has numerous peripherals that may not all be required for every application. Therefore, many of the pins have a secondary function as programmable I/O pins. There are two types of programmable I/O pins on the ADSP-BF539/ADSP-BF539F processor, with slightly different functionality: programmable flags and general-purpose I/O.

Programmable Flags (GPIO Port F)

The ADSP-BF539/ADSP-BF539F processors have 16 bidirectional, general-purpose programmable flag (PF15–0) pins on GPIO Port F. Each programmable flag can be individually controlled by manipulation of the flag control, status, and interrupt registers:

- Flag direction control register – Specifies the direction of each individual PFx pin as input or output.
- Flag control and status registers – The processors employ a "write one to modify" mechanism that allows any combination of individual flags to be modified in a single instruction, without affecting the level of any other flags. Four control registers are provided. One register is written in order to set flag values, one register is written in order to clear flag values, one register is written in order to toggle flag values, and one register is written in order to specify a flag value. Reading the flag status register allows software to interrogate the sense of the flags.
- Flag interrupt mask registers – The two flag interrupt mask registers allow each individual PFx pin to function as an interrupt to the processor. Similar to the two flag control registers that are used to set and clear individual flag values, one flag interrupt mask register sets bits to enable interrupt function, and the other flag interrupt mask register clears bits to disable interrupt function. PFx pins defined as inputs can be configured to generate hardware interrupts, while output PFx pins can be triggered by software interrupts.
- Flag interrupt sensitivity registers – The two flag interrupt sensitivity registers specify whether individual PFx 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.

The PFx pins can also be used by the SPI0 and PPI ports as shown in Table 4 on Page 11, depending on how the peripherals are configured. Care must be taken so that these pins are not used for multiple purposes simultaneously.

General-Purpose I/O Ports C, D, and E

The ADSP-BF539/ADSP-BF539F processors have 38 general-purpose I/O pins that are multiplexed with other peripherals. They are arranged into Ports C, D, and E as shown in Table 4 on Page 11. The GPIO differ from the programmable flags on Port F in that the GPIO pins cannot generate interrupts to the processor.

Table 4. Programmable Flag / GPIO Ports

Peripheral	Alternate Programmable Flag/GPIO Port Function
PPI	PF15–3
SPORT2	PE7–0
SPORT3	PE15–8
SPI0	PF7–0
SPI1	PD4–0
SPI2	PD9–5
UART1	PD11–10
UART2	PD13–12
CAN	PC1–0[1]
MXVR	PC9–4[1]

[1] PC1 and PC4 are open-drain when configured as GPIO outputs.

The general-purpose I/O pins can be individually controlled by manipulation of the control and status registers. These pins will not cause interrupts to be generated to the processor but can be polled to determine their status.

- GPIO direction control register – Specifies the direction of each individual GPIOx pin as input or output.
- GPIO control and status registers – The processors employ 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 pin. Four control registers and a data register are provided for each GPIO port. One register is written in order to set GPIO pin values, one register is written in order to clear GPIO pin values, one register is written in order to toggle GPIO pin values, and one register is written in order to specify a GPIO input or output. Reading the GPIO data register allows software to determine the state of the input GPIO pins.

Note that the GP pin is used to specify the status of the GPIO pins PC9–PC4 at power up. If GP is tied high, then pins PC9–PC4 are configured as GPIO after reset. The pins cannot be reconfigured through software, and special care must be taken with the MLF pin. If the GP pin is tied low, then the pins are configured as MXVR pins after reset but can be reconfigured as GPIO pins through software.

PARALLEL PERIPHERAL INTERFACE

The ADSP-BF539/ADSP-BF539F processors provide a parallel peripheral interface (PPI) that can connect directly to parallel ADC and DAC converters, video encoders and decoders, and other general-purpose peripherals. The PPI consists of a dedicated input clock pin, up to 3 frame synchronization pins, and up to 16 data pins. The input clock supports parallel data rates up to $f_{SCLK}/2$ MHz, 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 3 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 are 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:

- Input Mode – Frame syncs and data are inputs into the PPI.
- Frame Capture Mode – Frame syncs are outputs from the PPI, but data are inputs.
- Output Mode – Frame syncs and data are outputs from the PPI.

Input Mode

This 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 and are programmable in the PPI_CONTROL register.

Frame Capture Mode

This mode allows the video source(s) to act as a slave (e.g., for frame capture). The ADSP-BF539/ADSP-BF539F 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

This 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:

- Active Video Only Mode
- Vertical Blanking Only Mode
- Entire Field Mode

Active Video Only Mode

This mode is used when only the active video portion of a field is of interest and not any of the blanking intervals. The PPI will 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 will ignore 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 can be embedded in horizontal and vertical blanking intervals. Data transfer starts immediately after synchronization to Field 1.

CONTROLLER AREA NETWORK (CAN) INTERFACE

The ADSP-BF539/ADSP-BF539F processors provide a CAN controller that is a communication controller implementing the controller area network (CAN) V2.0B protocol. This protocol is an asynchronous communications protocol used in both industrial and automotive control systems. CAN is well suited for control applications due to its ability to communicate reliably over a network since the protocol incorporates CRC checking, message error tracking, and fault node confinement.

The CAN controller is based on a 32-entry mailbox RAM and supports both the standard and extended identifier (ID) message formats specified in the CAN protocol specification, Revision 2.0, Part B.

Each mailbox consists of eight 16-bit data words. The data is divided into fields, which includes a message identifier, a time stamp, a byte count, up to 8 bytes of data, and several control bits. Each node monitors the messages being passed on the network. If the identifier in the transmitted message matches an identifier in one of its mailboxes, then the module knows that the message was meant for it, passes the data into its appropriate mailbox, and signals the processor of message arrival with an interrupt.

The CAN controller can wake up the processor from sleep mode upon generation of a wake-up event, such that the processor can be maintained in a low power mode during idle conditions. Additionally, a CAN wake-up event can wake up the on-chip internal voltage regulator from the hibernate state.

The electrical characteristics of each network connection are very stringent; therefore, the CAN interface is typically divided into two parts: a controller and a transceiver. This allows a single controller to support different drivers and CAN networks. The ADSP-BF539/ADSP-BF539F CAN module represents the controller part of the interface. This module's network I/O is a single transmit output and a single receive input, which connect to a line transceiver.

The CAN clock is derived from the processor system clock (SCLK) through a programmable divider and therefore does not require an additional crystal.

MEDIA TRANSCEIVER MAC LAYER (MXVR)

The ADSP-BF539/ADSP-BF539F processors provide a media transceiver (MXVR) MAC layer, allowing the processor to be connected directly to a MOST network through just an FOT or electrical PHY.

The MXVR is fully compatible with industry standard standalone MOST controller devices, supporting 22.579 Mbps or 24.576 Mbps data transfer. It offers faster lock times, greater jitter immunity, and a sophisticated DMA scheme for data transfers. The high speed internal interface to the core and L1 memory allows the full bandwidth of the network to be utilized. The MXVR can operate as either the network master or as a network slave.

Synchronous data is transferred to or from the synchronous data channels through eight programmable DMA engines. The synchronous data DMA engines can operate in various modes, including modes that trigger DMA operation when data patterns are detected in the receive data stream. Furthermore, two DMA engines support asynchronous traffic and control message traffic.

Interrupts are generated when a user-defined amount of synchronous data has been sent or received by the processor or when asynchronous packets or control messages have been sent or received.

The MXVR peripheral can wake up the processor from sleep mode when a wake-up preamble is received over the network or based on any other MXVR interrupt event. Additionally, detection of network activity by the MXVR can be used to wake up the processor from sleep mode and wake up the on-chip internal voltage regulator from the powered-down hibernate state. These features allow the processor to operate in a low-power state when there is no network activity or when data is not currently being received or transmitted by the MXVR.

The MXVR clock is provided through a dedicated external crystal or crystal oscillator. For 44.1 kHz frame syncs, use a 45.1584 MHz crystal or oscillator; for 48 kHz frame syncs, use a 49.152 MHz crystal or oscillator. If using a crystal to provide the MXVR clock, use a parallel-resonant, fundamental mode, microprocessor-grade crystal.

DYNAMIC POWER MANAGEMENT

The ADSP-BF539/ADSP-BF539F processors provide 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. Control of clocking to each of the ADSP-BF539/ADSP-BF539F processor peripherals also reduces power consumption. See Table 5 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 5. 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
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, assertion of a wake-up event enabled in the SIC_IWRx register 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 will transition to the active mode. When in the sleep mode, system DMA access to L1 memory is not supported.

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 will not be able to 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 the synchronous peripherals (SCLK). The internal voltage regulator for the processor can be shut off by writing b#00 to the FREQ bits of the VR_CTL register. This sets the internal power supply voltage (V_{DDINT}) to 0 V to provide the lowest static power dissipation. Any critical information stored internally (memory contents, register contents, etc.) must be written to a nonvolatile storage device prior to removing power if the processor state is to be preserved. Since V_{DDEXT} 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 internal supply regulator can be woken up either by a real-time clock wake-up, by CAN bus traffic, by asserting the $\overline{\text{RESET}}$ pin, or by an external source via the $\overline{\text{GPW}}$ pin.

Power Savings

As shown in Table 6, the ADSP-BF539/ADSP-BF539F processors support five different power domains. The use of multiple power domains maximizes flexibility, while maintaining compliance with industry standards and conventions:

- The 3.3 V V_{DDRTC} power domain supplies the RTC I/O and logic so that the RTC can remain functional when the rest of the chip is powered off.
- The 3.3 V MXEVDD power domain supplies the MXVR crystal and is separate to provide noise isolation.
- The 1.25 V MPIVDD power domain supplies the MXVR PLL and is separate to provide noise isolation.
- The 1.25 V V_{DDINT} power domain supplies all internal logic except for the RTC logic and the MXVR PLL.
- The 3.3 V V_{DDEXT} power domain supplies all I/O except for the RTC and MXVR crystals.

There are no sequencing requirements for the various power domains.

Table 6. Power Domains

Power Domain	VDD Range
RTC Crystal I/O and Logic	V_{DDRTC}
MXVR Crystal I/O	MXEVDD
MXVR PLL Analog and Logic	MPIVDD
All Internal Logic Except RTC and MXVR PLL	V_{DDINT}
All I/O Except RTC and MXVR Crystals	V_{DDEXT}

The V_{DDRTC} should either be connected to an isolated supply such as a battery (if the RTC is to operate while the rest of the chip is powered down) or should be connected to the V_{DDEXT} plane on the board. The V_{DDRTC} should remain powered when the processor is in hibernate state and should also remain powered even if the RTC functionality is not being used in an application. The MXEVDD should be connected to the V_{DDEXT} plane on the board at a single location with local bypass capacitors. The MXEVDD should remain powered when the processor is in hibernate state and should also remain powered even when the MXVR functionality is not being used in an application. The MPIVDD should be connected to the V_{DDINT} plane on the board at a single location through a ferrite bead with local bypass capacitors.

The power dissipated by a processor is largely a function of the clock frequency of the processor 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.

The dynamic power management feature of the ADSP-BF539/ADSP-BF539F processors allow both the processor input voltage (V_{DDINT}) and clock frequency (f_{CCLK}) to be dynamically controlled.

The savings in power dissipation can be modeled using the power savings factor and % power savings calculations.

The power savings factor is calculated as

$$\text{Power Savings Factor} = \frac{f_{CCLKRED}}{f_{CCLKNOM}} \times \left(\frac{V_{DDINTRED}}{V_{DDINTNOM}}\right)^2 \times \left(\frac{t_{RED}}{t_{NOM}}\right)$$

where:

$f_{CCLKNOM}$ 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* $f_{CCLKRED}$.

The Power Savings Factor is calculated as

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

VOLTAGE REGULATION

The Blackfin processors provide an on-chip voltage regulator that can generate appropriate V_{DDINT} voltage levels from the V_{DDEXT} supply. See Operating Conditions on Page 26 for regulator tolerances and acceptable V_{DDEXT} ranges for specific models.[†]

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 I/O power (V_{DDRTC}, MXEVDD, V_{DDEXT}) is still supplied. While in the hibernate state, I/O power is still being applied, eliminating the need for external buffers. The voltage regulator can be activated from this power-down state through an RTC wake-up, a CAN wake-up, an MXVR wake-up, a general-purpose wake-up, or by asserting $\overline{\text{RESET}}$, all of which will then initiate a boot sequence. The regulator can also be disabled and bypassed at the user's discretion.

CLOCK SIGNALS

The ADSP-BF539/ADSP-BF539F processors 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 ADSP-BF539/ADSP-BF539F processors include an on-chip oscillator circuit, an external crystal can be used. For fundamental frequency operation, use the circuit shown in Figure 7 on Page 16. A parallel-resonant, fundamental frequency, microprocessor-grade crystal is connected across the

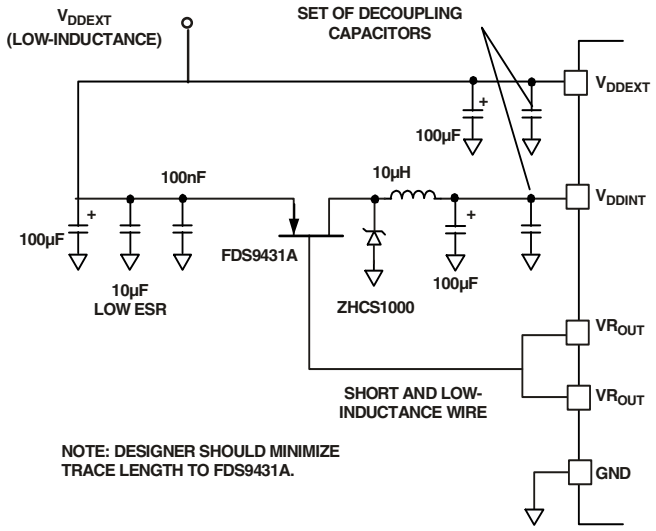


Figure 6. Voltage Regulator Circuit

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 7 on Page 16, fine tune the phase and amplitude of the sine frequency. The capacitor and resistor values, shown in Figure 7 on Page 16, are typical values only. The capacitor values are dependent upon the crystal manufacturer's load capacitance recommendations and the physical PCB layout. The resistor value depends on the drive level specified by the crystal manufacturer. System designs should verify the customized values based on careful investigation on multiple devices over the allowed temperature range.

A third-overtone crystal can be used at 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 7.

As shown in Figure 8, 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 user 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.

[†] See *Switching Regulator Design Considerations for ADSP-BF533 Blackfin Processors (EE-228).*

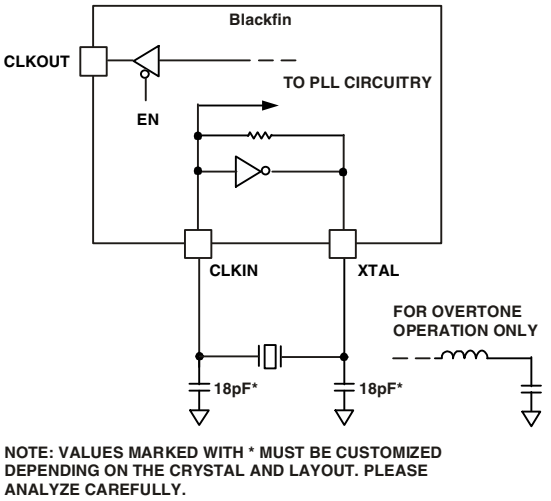


Figure 7. External Crystal Connections

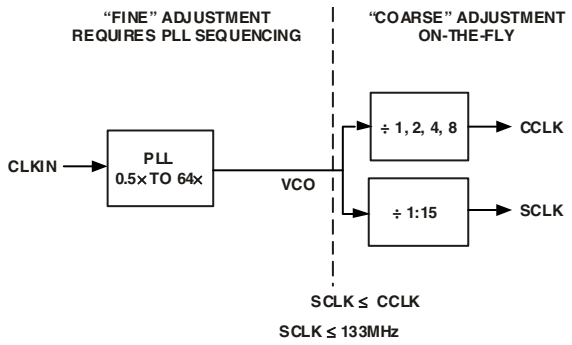


Figure 8. Frequency Modification Methods

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 7 illustrates typical system clock ratios.

Table 7. 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 maximum frequency of the system clock is f_{SCLK}. 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).

Note that when the SSEL value is changed, it will affect all the peripherals that derive their clock signals from the SCLK signal.

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 8. This programmable core clock capability is useful for fast core frequency modifications.

Table 8. Core Clock Ratios

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

BOOTING MODES

The ADSP-BF539/ADSP-BF539F processors have three mechanisms (listed in Table 9) for automatically loading internal L1 instruction memory after a reset. A fourth mode is provided to execute from external memory, bypassing the boot sequence.

Table 9. Booting Modes

BMODE1–0	Description
00	Execute from 16-bit external memory (bypass boot ROM)
01	Boot from 8-bit or 16-bit flash or boot from on-chip flash (ADSP-BF539F only)
10	Boot from SPI serial master connected to SPI0
11	Boot from SPI serial slave EEPROM/flash (8-,16-, or 24-bit address range, or Atmel AT45DB041, AT45DB081, or AT45DB161serial flash) connected to SPI0

The BMODE pins of the reset configuration register, sampled during power-on resets and software initiated resets, implement the following modes:

- Execute from 16-bit external memory – Execution starts from address 0x2000 0000 with 16-bit packing. The boot ROM is bypassed in this mode. All configuration settings are set for the slowest device possible (3-cycle hold time; 15-cycle R/W access times; 4-cycle setup).
- Boot from 8-bit or 16-bit external flash memory – The 8-bit flash boot routine located in boot ROM memory space is set up using asynchronous memory bank 0. For ADSP-BF539F processors, if $\overline{FCE}$ is connected to $\overline{AMS0}$, then the on-chip flash is booted. All configuration settings are set for the slowest device possible (3-cycle hold time; 15-cycle R/W access times; 4-cycle setup).
- Boot from SPI serial EEPROM/flash (8-, 16-, or 24-bit addressable, or Atmel AT45DB041, AT45DB081, or AT45DB161) connected to SPI0 – The SPI0 port uses the

PF2 output pin to select a single SPI EEPROM/flash device, submits a read command and successive address bytes (0x00) until a valid 8-, 16-, or 24-bit, or Atmel addressable device is detected, and begins clocking data into the beginning of the L1 instruction memory.

- Boot from SPI host device connected to SPI0 – The Blackfin processor operates in SPI slave mode and is configured to receive the bytes of the .LDR file from an SPI host (master) agent. To hold off the host device from transmitting while the boot ROM is busy, the Blackfin processor asserts a GPIO pin, called host wait (HWAIT), to signal the host device not to send any more bytes until the flag is deasserted. The flag is chosen by the user and this information is transferred to the Blackfin processor via bits 10:5 of the FLAG header in the .LDR image.

For each of the boot modes, a 10-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 can be loaded by any boot sequence. Once all blocks are loaded, program execution commences from the start of L1 instruction SRAM.

In addition, Bit 4 of the reset configuration register can be set by application code to bypass the normal boot sequence during a software reset. For this case, the processor jumps directly to the beginning of L1 instruction memory.

To augment the boot modes, a secondary software loader is provided that adds additional booting mechanisms. This secondary loader provides the ability to boot from 16-bit flash memory, fast flash, variable baud rate, and other sources. In all boot modes except bypass, program execution starts from on-chip L1 memory address 0xFFA0 0000.

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/CPU 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 plus two load/store plus 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 ADSP-BF539/ADSP-BF539F processors are supported by 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-BF539/ADSP-BF539F processor.

The VisualDSP++ project management environment lets programmers develop and debug an application. This environment includes an easy to use assembler (which is based on an algebraic syntax), an archiver (librarian/library builder), a linker, a loader, a cycle-accurate instruction-level simulator, a C/C++ compiler, and a C/C++ runtime library that includes DSP and mathematical functions. A key point for these tools is C/C++ code efficiency. The compiler has been developed for efficient translation of C/C++ code to processor assembly. The processor has architectural features that improve the efficiency of compiled C/C++ code.

The VisualDSP++ debugger has a number of important features. Data visualization is enhanced by a plotting package that offers a significant level of flexibility. This graphical representation of user data enables the programmer to quickly determine the performance of an algorithm. As algorithms grow in complexity, this capability can have increasing significance on the designer's development schedule, increasing productivity. Statistical profiling enables the programmer to nonintrusively poll the processor as it is running the program. This feature, unique to VisualDSP++, enables the software developer to passively gather important code execution metrics without interrupting the real-time characteristics of the program. Essentially, the developer can identify bottlenecks in software quickly and efficiently. By using the profiler, the programmer can focus on those areas in the program that impact performance and take corrective action.

Debugging both C/C++ and assembly programs with the VisualDSP++ debugger, programmers can

- View mixed C/C++ and assembly code (interleaved source and object information).
- Insert breakpoints.

[†] CROSSCORE is a registered trademark of Analog Devices, Inc.

- Set conditional breakpoints on registers, memory, and stacks.
- Trace instruction execution.
- Perform linear or statistical profiling of program execution.
- Fill, dump, and graphically plot the contents of memory.
- Perform source level debugging.
- Create custom debugger windows.

The VisualDSP++ IDDE lets programmers define and manage software development. Its dialog boxes and property pages let programmers configure and manage all of the Blackfin development tools, including the color syntax highlighting in the VisualDSP++ editor. This capability permits programmers to

- Control how the development tools process inputs and generate outputs.
- Maintain a one-to-one correspondence with the tool's command line switches.

The VisualDSP++ Kernel (VDK) incorporates scheduling and resource management tailored specifically to address the memory and timing constraints of DSP programming. These capabilities enable engineers to develop code more effectively, eliminating the need to start from the very beginning, when developing new application code. The VDK features include threads, critical and unscheduled regions, semaphores, events, and device flags. The VDK also supports priority-based, preemptive, cooperative, and time-sliced scheduling approaches. In addition, the VDK was designed to be scalable. If the application does not use a specific feature, the support code for that feature is excluded from the target system.

Because the VDK is a library, a developer can decide whether to use it or not. The VDK is integrated into the VisualDSP++ development environment, but can also be used via standard command line tools. When the VDK is used, the development environment assists the developer with many error prone tasks and assists in managing system resources, automating the generation of various VDK-based objects, and visualizing the system state, when debugging an application that uses the VDK.

Use the Expert Linker to visually manipulate the placement of code and data on the embedded system. View memory utilization in a color-coded graphical form, easily move code and data to different areas of the processor or external memory with the drag of the mouse and examine run-time stack and heap usage. The Expert Linker is fully compatible with existing Linker Definition File (LDF), allowing the developer to move between the graphical and textual environments.

Analog Devices emulators use the IEEE 1149.1 JTAG Test Access Port of the ADSP-BF539/ADSP-BF539F processors to monitor and control the target board processor during emulation. The emulator provides full-speed emulation, allowing inspection and modification of memory, registers, and processor stacks. Nonintrusive in-circuit emulation is assured by the use of the processor's JTAG interface—the emulator does not affect target system loading or timing.

In addition to the software and hardware development tools available from Analog Devices third parties provide a wide range of tools supporting the Blackfin processor family. Hardware tools include Blackfin processor PC plug-in cards. Third party software tools include DSP libraries, real-time operating systems, and block diagram design tools.

DESIGNING AN EMULATOR COMPATIBLE PROCESSOR BOARD

The Analog Devices family of emulators are tools that every system developer needs 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 *Analog Devices JTAG Emulation Technical Reference (EE-68)* on the Analog Devices web site (www.analog.com)—use site search on "EE-68." This document is updated regularly to keep pace with improvements to emulator support.

EXAMPLE CONNECTIONS AND LAYOUT CONSIDERATIONS

Figure 9 shows an example circuit connection of the ADSP-BF539/ADSP-BF539F to a MOST network. This diagram is intended as an example, and exact connections and recommended circuit values should be obtained from Analog Devices.

MXVR BOARD LAYOUT GUIDELINES

MLF pin

- Capacitors:

  C1: 0.1 µF (PPS type, 2% tolerance recommended)

  C2: 0.01 µF (PPS type, 2% tolerance recommended)
- Resistor:

  R1: 220 W (1% tolerance)
- The RC network connected to the MLF pin should be located physically close to the MLF pin on the board.
- The RC network should be wired up and connected to the MLF pin using wide traces.
- The capacitors in the RC network should be grounded to MXEGND.

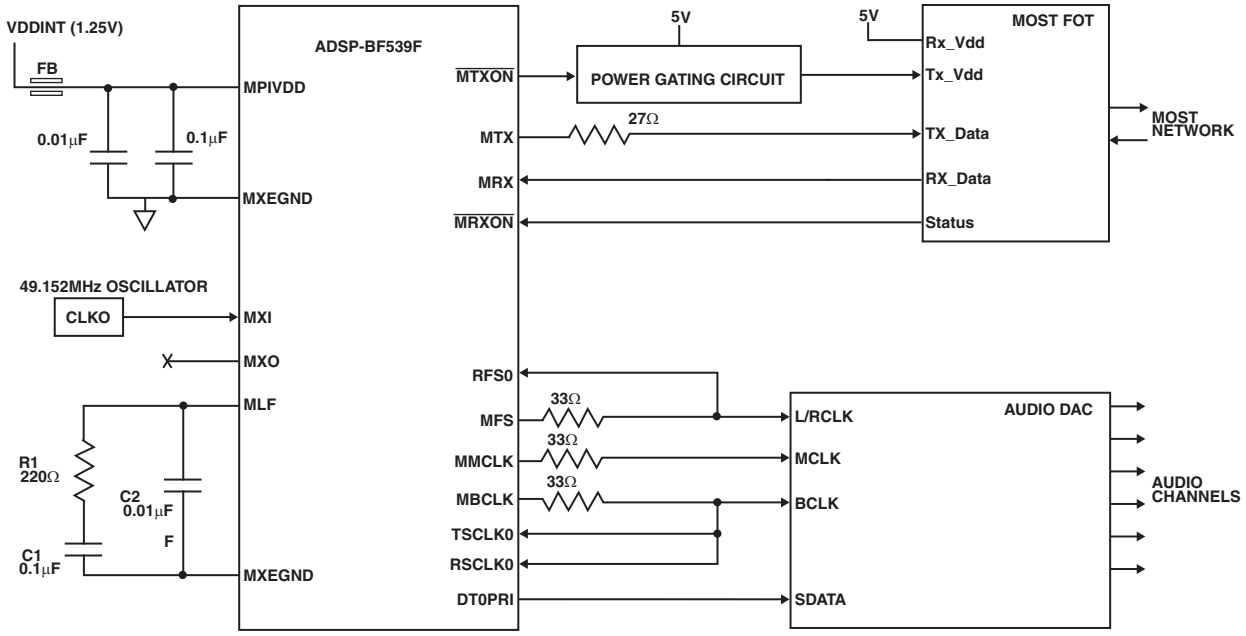


Figure 9. Example Connections of ADSP-BF539/ADSP-BF539F to MOST Network

- The RC network should be shielded using MXEGND traces.
- Avoid routing other switching signals near the RC network to avoid crosstalk.

MXI driven with external clock oscillator IC (recommended)

- MXI should be driven with the clock output of a 49.152 MHz or 45.1584 MHz clock oscillator IC.
- MXO should be left unconnected.
- Avoid routing other switching signals near the oscillator and clock output trace to avoid crosstalk. When not possible, shield traces with ground.

MXI/MXO with external crystal

- The crystal must be a 49.152 MHz or 45.1584 MHz fundamental mode crystal.
- The crystal and load capacitors should be placed physically close to the MXI and MXO pins on the board.
- The load capacitors should be grounded to MXEGND.
- The crystal and load capacitors should be wired up using wide traces.
- Board trace capacitance on each lead should not be more than 3 pF.
- Trace capacitance plus load capacitance should equal the load capacitance specification for the crystal.
- Avoid routing other switching signals near the crystal and components to avoid crosstalk. When not possible, shield traces and components with ground.

MXEGND–MXVR crystal oscillator and MXVR PLL ground

- Should be routed with wide traces or as ground plane.
- Should be tied together to other board grounds at only one point on the board.
- Avoid routing other switching signals near to MXEGND to avoid crosstalk.

MXEVDD–MXVR crystal oscillator 3.3 V power

- Should be routed with wide traces or as power plane.
- Locally bypass MXEVDD with 0.1 µF and 0.01 µF decoupling capacitors to MXEGND.
- Avoid routing other switching signals near to MXEVDD to avoid crosstalk.

MPIVDD–MXVR PLL 1.25 V power

- Should be routed with wide traces or as power plane.
- A ferrite bead should be placed between the 1.25 V VDDINT power plane and the MPIVDD pin for noise isolation.
- Locally bypass MPIVDD with 0.1 µF and 0.01 µF decoupling capacitors to MXEGND.
- Avoid routing other switching signals near to MPIVDD to avoid crosstalk.

Fiber optic transceiver (FOT) connections

- The traces between the ADSP-BF539/ADSP-BF539F processor and the FOT should be kept as short as possible.
- The receive data trace connecting the FOT receive data output pin to the ADSP-BF539/ADSP-BF539F MRX input pin should not have a series termination resistor. The edge rate of the FOT receive data signal driven by the FOT is typically very slow, and further degradation of the edge rate is not desirable.
- The transmit data trace connecting the processor's MTX output pin to the FOT Transmit Data input pin should have a 27 W series termination resistor placed close to the ADSP-BF539/ADSP-BF539F MTX pin.
- The receive data trace and the transmit data trace between the processor and the FOT should not be routed close to each other in parallel over long distances to avoid crosstalk.

VOLTAGE REGULATOR LAYOUT GUIDELINES

Regulator external component placement, board routing, and bypass capacitors all have a significant effect on noise injected into the other analog circuits on-chip. The VROUT1-0 traces and voltage regulator external components should be considered as noise sources when doing board layout and should not be routed or placed near sensitive circuits or components on the board. All internal and I/O power supplies should be well bypassed with bypass capacitors placed as close to the ADSP-BF539/ADSP-BF539F as possible.

For further details on the on-chip voltage regulator and related board design guidelines, see the *Switching Regulator Design Considerations for ADSP-BF533 Blackfin Processors (EE-228)* applications note on the Analog Devices website (www.analog.com)—use site search on "EE-228".

RELATED DOCUMENTS

The following publications that describe the ADSP-BF539/ADSP-BF539F 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-BF539 Blackfin Processor Hardware Reference*
- *ADSP-BF53x/ADSP-BF56x Blackfin Processor Programming Reference*
- *ADSP-BF539 Blackfin Processor Anomaly List*

PIN DESCRIPTIONS

ADSP-BF539/ADSP-BF539F processor pin definitions are listed in Table 10.

All pins are three-stated during and immediately after reset, except the memory interface, asynchronous memory control, and synchronous memory control pins. These pins are all driven high, with the exception of CLKOUT, which toggles at the system clock rate. If BR is active, then the memory pins are also three-stated. All unused I/O pins have their input buffers disabled with the exception of the pins that need pull-ups or pull-downs, as noted in the table.

In order to maintain maximum functionality and reduce package size and pin count, some pins have dual, multiplexed functionality. In cases where pin functionality is reconfigurable, the default state is shown in plain text, while alternate functionality is shown in italics.

Table 10. Pin Descriptions

Pin Name	Type	Description	Driver Type[1]
Memory Interface			
ADDR19–1	O	Address Bus for Async/Sync Access	A
DATA15–0	I/O	Data Bus for Async/Sync Access	A
$\overline{\text{ABE1–0}}$/SDQM1–0	O	Byte Enables/Data Masks for Async/Sync Access	A
$\overline{\text{BR}}$	I	Bus Request (This pin should be pulled high when not used.)	
$\overline{\text{BG}}$	O	Bus Grant	A
$\overline{\text{BGH}}$	O	Bus Grant Hang	A
Asynchronous Memory Control			
$\overline{\text{AMS3–0}}$	O	Bank Select	A
ARDY	I	Hardware Ready Control (This pin should always be pulled low when not used.)	
$\overline{\text{AOE}}$	O	Output Enable	A
$\overline{\text{ARE}}$	O	Read Enable	A
$\overline{\text{AWE}}$	O	Write Enable	A
Flash Control			
$\overline{\text{FCE}}$	I	Flash Enable (This pin is internally connected to GND on the ADSP-BF539.)	
$\overline{\text{FRESET}}$	I	Flash Reset (This pin is internally connected to GND on the ADSP-BF539.)	
Synchronous Memory Control			
$\overline{\text{SRAS}}$	O	Row Address Strobe	A
$\overline{\text{SCAS}}$	O	Column Address Strobe	A
$\overline{\text{SWE}}$	O	Write Enable	A
SCKE	O	Clock Enable	A
CLKOUT	O	Clock Output	B
SA10	O	A10 Pin	A
$\overline{\text{SMS}}$	O	Bank Select	A
Timers			
TMR0	I/O	Timer 0	C
TMR1/*PPI_FS1*	I/O	Timer 1/*PPI Frame Sync1*	C
TMR2/*PPI_FS2*	I/O	Timer 2/*PPI Frame Sync2*	C

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
Parallel Peripheral Interface Port/GPIO			
PF0/*SPI0SS*	I/O	Programmable Flag 0/*SPI0 Slave Select Input*	C
PF1/*SPI0SEL1*/*TACLK*	I/O	Programmable Flag 1/*SPI0 Slave Select Enable 1/Timer Alternate Clock*	C
PF2/*SPI0SEL2*	I/O	Programmable Flag 2/*SPI0 Slave Select Enable 2*	C
PF3/*SPI0SEL3*/*PPI_FS3*	I/O	Programmable Flag 3/*SPI0 Slave Select Enable 3/PPI Frame Sync 3*	C
PF4/*SPI0SEL4*/*PPI15*	I/O	Programmable Flag 4/*SPI0 Slave Select Enable 4/PPI 15*	C
PF5/*SPI0SEL5*/*PPI14*	I/O	Programmable Flag 5/*SPI0 Slave Select Enable 5/PPI 14*	C
PF6/*SPI0SEL6*/*PPI13*	I/O	Programmable Flag 6/*SPI0 Slave Select Enable 6/PPI 13*	C
PF7/*SPI0SEL7*/*PPI12*	I/O	Programmable Flag 7/*SPI0 Slave Select Enable 7/PPI 12*	C
PF8/*PPI11*	I/O	Programmable Flag 8/*PPI 11*	C
PF9/*PPI10*	I/O	Programmable Flag 9/*PPI 10*	C
PF10/*PPI9*	I/O	Programmable Flag 10/*PPI 9*	C
PF11/*PPI8*	I/O	Programmable Flag 11/*PPI 8*	C
PF12/*PPI7*	I/O	Programmable Flag 12/*PPI 7*	C
PF13/*PPI6*	I/O	Programmable Flag 13/*PPI 6*	C
PF14/*PPI5*	I/O	Programmable Flag 14/*PPI 5*	C
PF15/*PPI4*	I/O	Programmable Flag 15/*PPI 4*	C
PPI3–0	I/O	PPI3–0	C
PPI_CLK/*TMRCLK*	I	PPI Clock/*External Timer Reference*	
Controller Area Network			
CANTX/*PC0*	I/O 5 V	CAN Transmit/*GPIO*	C
CANRX/*PC1*	I/OD 5 V	CAN Receive/*GPIO*	C[2]
Media Transceiver (MXVR)/ General-Purpose I/O			
MTX/*PC5*	I/O	MXVR Transmit Data/*GPIO*	C
$\overline{\text{MTXON}}$/*PC9*	I/O	MXVR Transmit FOT On/*GPIO*	C
MRX/*PC4*	I/OD 5 V	MXVR Receive Data/*GPIO* (This pin should be pulled low when not used.)	C[2]
$\overline{\text{MRXON}}$	I 5 V	MXVR FOT Receive On (This pin should be pulled high when not used.)	C
MXI	I	MXVR Crystal Input (This pin should be pulled low when not used.)	
MXO	O	MXVR Crystal Output (This pin should be left unconnected when not used.)	
MLF	A I/O	MXVR Loop Filter (This pin should be pulled low when not used.)	
MMCLK/*PC6*	I/O	MXVR Master Clock/*GPIO*	C
MBCLK/*PC7*	I/O	MXVR Bit Clock/*GPIO*	C
MFS/*PC8*	I/O	MXVR Frame Sync/*GPIO*	C
GP	I	GPIO PC4–9 Enable (This pin should be pulled low when MXVR is used.)	

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
2-Wire Interface Ports		These pins are open-drain and require a pull-up resistor. See version 2.1 of the I^2C specification for proper resistor values.	
SDA0	I/O 5 V	TWI0 Serial Data	E
SCL0	I/O 5 V	TWI0 Serial Clock	E
SDA1	I/O 5 V	TWI1 Serial Data	E
SCL1	I/O 5 V	TWI1 Serial Clock	E
Serial Port0			
RSCLK0	I/O	SPORT0 Receive Serial Clock	D
RFS0	I/O	SPORT0 Receive Frame Sync	C
DR0PRI	I	SPORT0 Receive Data Primary	
DR0SEC	I	SPORT0 Receive Data Secondary	
TSCLK0	I/O	SPORT0 Transmit Serial Clock	D
TFS0	I/O	SPORT0 Transmit Frame Sync	C
DT0PRI	O	SPORT0 Transmit Data Primary	C
DT0SEC	O	SPORT0 Transmit Data Secondary	C
Serial Port1			
RSCLK1	I/O	SPORT1 Receive Serial Clock	D
RFS1	I/O	SPORT1 Receive Frame Sync	C
DR1PRI	I	SPORT1 Receive Data Primary	
DR1SEC	I	SPORT1 Receive Data Secondary	
TSCLK1	I/O	SPORT1 Transmit Serial Clock	D
TFS1	I/O	SPORT1 Transmit Frame Sync	C
DT1PRI	O	SPORT1 Transmit Data Primary	C
DT1SEC	O	SPORT1 Transmit Data Secondary	C
Serial Port2			
RSCLK2/*PE0*	I/O	SPORT2 Receive Serial Clock/*GPIO*	D
RFS2/*PE1*	I/O	SPORT2 Receive Frame Sync/*GPIO*	C
DR2PRI/*PE2*	I/O	SPORT2 Receive Data Primary/*GPIO*	C
DR2SEC/*PE3*	I/O	SPORT2 Receive Data Secondary/*GPIO*	C
TSCLK2/*PE4*	I/O	SPORT2 Transmit Serial Clock/*GPIO*	D
TFS2/*PE5*	I/O	SPORT2 Transmit Frame Sync/*GPIO*	C
DT2PRI /*PE6*	I/O	SPORT2 Transmit Data Primary/*GPIO*	C
DT2SEC/*PE7*	I/O	SPORT2 Transmit Data Secondary/*GPIO*	C

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
Serial Port3			
RSCLK3/*PE8*	I/O	SPORT3 Receive Serial Clock/*GPIO*	D
RFS3/*PE9*	I/O	SPORT3 Receive Frame Sync/*GPIO*	C
DR3PRI/*PE10*	I/O	SPORT3 Receive Data Primary/*GPIO*	C
DR3SEC/*PE11*	I/O	SPORT3 Receive Data Secondary/*GPIO*	C
TSCLK3/*PE12*	I/O	SPORT3 Transmit Serial Clock/*GPIO*	D
TFS3/*PE13*	I/O	SPORT3 Transmit Frame Sync/*GPIO*	C
DT3PRI /*PE14*	I/O	SPORT3 Transmit Data Primary/*GPIO*	C
DT3SEC/*PE15*	I/O	SPORT3 Transmit Data Secondary/*GPIO*	C
SPI0 Port			
MOSI0	I/O	SPI0 Master Out Slave In	C
MISO0	I/O	SPI0 Master In Slave Out (This pin should always be pulled high through a 4.7 kΩ resistor if booting via the SPI port.)	C
SCK0	I/O	SPI0 Clock	D
SPI1 Port			
MOSI1/*PD0*	I/O	SPI1 Master Out Slave In/*GPIO*	C
MISO1/*PD1*	I/O	SPI1 Master In Slave Out/*GPIO*	C
SCK1/*PD2*	I/O	SPI1 Clock/*GPIO*	D
$\overline{\text{SPI1SS}}$/*PD3*	I/O	SPI1 Slave Select Input/*GPIO*	D
$\overline{\text{SPI1SEL1}}$/*PD4*	I/O	SPI1 Slave Select Enable/*GPIO*	D
SPI2 Port			
MOSI2 /*PD5*	I/O	SPI2 Master Out Slave In/*GPIO*	C
MISO2/*PD6*	I/O	SPI2 Master In Slave Out/*GPIO*	C
SCK2/*PD7*	I/O	SPI2 Clock/*GPIO*	D
$\overline{\text{SPI2SS}}$/*PD8*	I/O	SPI2 Slave Select Input/*GPIO*	D
$\overline{\text{SPI2SEL1}}$/*PD9*	I/O	SPI2 Slave Select Enable/*GPIO*	D
UART0 Port			
RX0	I	UART Receive	
TX0	O	UART Transmit	C
UART1 Port			
RX1/*PD10*	I/O	UART1 Receive/*GPIO*	D
TX1/*PD11*	I/O	UART1 Transmit/*GPIO*	D
UART2 Port			
RX2 /*PD12*	I/O	UART2 Receive/*GPIO*	D
TX2/*PD13*	I/O	UART2 Transmit/*GPIO*	D
Real-Time Clock			
RTXI	I	RTC Crystal Input (This pin should be pulled low when not used.)	
RTXO	O	RTC Crystal Output	

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
JTAG Port			
TCK	I	JTAG Clock	
TDO	O	JTAG Serial Data Out	C
TDI	I	JTAG Serial Data In	
TMS	I	JTAG Mode Select	
$\overline{\text{TRST}}$	I	JTAG Reset (This pin should be pulled low if the JTAG port will not be used.)	
$\overline{\text{EMU}}$	O	Emulation Output	C
Clock			
CLKIN	I	Clock/Crystal Input	
XTAL	O	Crystal Output	
Mode Controls			
$\overline{\text{RESET}}$	I	Reset	
$\overline{\text{NMI}}$	I	Nonmaskable Interrupt (This pin should be pulled high when not used.)	
BMODE1–0	I	Boot Mode Strap	
Voltage Regulator			
VROUT0	O	External FET Drive 0 (This pin should be left unconnected when not used.)	
VROUT1	O	External FET Drive 1 (This pin should be left unconnected when not used.)	
Supplies			
V_{DDEXT}	P	I/O Power Supply	
V_{DDINT}	P	Internal Power Supply	
V_{DDRTC}	P	Real-Time Clock Power Supply (This pin should be connected to V_{DDEXT} when not used and should remain powered at all times.)	
MPIVDD	P	MXVR Internal Power Supply	
MXEVDD	P	MXVR External Power Supply	
MXEGND	G	MXVR Ground	
GND	G	Ground	

[1] Refer to Figure 30 on Page 51 to Figure 39 on Page 52.

[2] This pin is 5 V-tolerant when configured as an input and an open-drain when configured as an output; therefore, only the VOL curves in Figure 34 on Page 51 and Figure 35 on Page 51 and the fall time curves in Figure 47 on Page 54 and Figure 48 on Page 54 apply when configured as an output.

SPECIFICATIONS

Component specifications are subject to change without notice.

OPERATING CONDITIONS

Parameter		Conditions	Min	Nom	Max	Unit
V_{DDINT}	Internal Supply Voltage[1, 2]		0.95	1.25	1.375	V
V_{DDEXT}	External Supply Voltage[3]		2.7	3.3	3.6	V
V_{DDRTC}	Real-Time Clock Power Supply Voltage		2.7	3.3	3.6	V
V_{IH}	High Level Input Voltage[4]	V_{DDEXT} = Maximum	2.0		3.6	V
V_{IH5V}	High Level Input Voltage[5]	V_{DDEXT} = Maximum	2.0		5.5	V
$V_{IHCLKIN}$	High Level Input Voltage[6]	V_{DDEXT} = Maximum	2.2		3.6	V
V_{IL}	Low Level Input Voltage[4, 7]	V_{DDEXT} = Minimum	–0.3		+0.6	V
V_{IL5V}	Low Level Input Voltage[5]	V_{DDEXT} = Minimum	–0.3		+0.8	V
T_J	Junction Temperature	316-Ball Chip Scale Ball Grid Array Package (CSP_BGA) 533 MHz @ $T_{AMBIENT}$ = –40°C to +85°C	–40		+105	°C

[1] Parameter value applies also to MPIVDD.
[2] The regulator can generate V_{DDINT} at levels of 1.0 V to 1.2 V with –5% to +10% tolerance and 1.25 V with –4% to +10% tolerance.
[3] Parameter value applies also to MXEVDD.
[4] The 3.3 V tolerant pins are capable of accepting up to 3.6 V maximum V_{IH} The following bidirectional pins are 3.3 V tolerant: DATA15–0, SCK2–0, MISO2–0, MOSI2–0, PF15–0, PPI3–0, $\overline{\text{MTXON}}$, MMCLK, MBCLK, MFS, MTX, $\overline{\text{SPI1SS}}$, $\overline{\text{SPI1SEL1}}$, $\overline{\text{SPI2SS}}$, $\overline{\text{SPI2SEL1}}$, RX2–1, TX2–1, DT2PRI, DT2SEC, TSCLK3–0, DR2PRI, DR2SEC, DT3PRI, DT3SEC, RSCLK3–0, TFS3–0, RFS3–0, DR3PRI, DR3SEC, and TMR2–0. The following input-only pins are 3.3 V tolerant: $\overline{\text{RESET}}$, RX0, TCK, TDI, TMS, $\overline{\text{TRST}}$, ARDY, BMODE1–0, $\overline{\text{BR}}$, DR0PRI, DR0SEC, DR1PRI, DR1SEC, $\overline{\text{NMI}}$, PPI_CLK, RTXI, and GP.
[5] The 5 V tolerant pins are capable of accepting up to 5.5 V maximum V_{IH}. The following bidirectional pins are 5 V tolerant: SCL0, SCL1, SDA0, SDA1, and CANTX. The following input-only pins are 5 V tolerant: CANRX, MRX, $\overline{\text{MRXON}}$.
[6] Parameter value applies to the CLKIN and MXI input pins.
[7] Parameter value applies to all input and bidirectional pins.

The following tables describe the voltage/frequency requirements for the ADSP-BF538/ADSP-BF538F processor clocks. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock (Table 11) and system clock (Table 13) specifications. Table 12 describes phase-locked loop operating conditions.

Table 11. Core Clock (CCLK) Requirements

Parameter	Description	Internal Regulator Setting	Max	Unit
f_{CCLK}	CLK Frequency (V_{DDINT} = 1.2 V Minimum)	1.25 V	533	MHz
f_{CCLK}	CLK Frequency (V_{DDINT} = 1.14 V Minimum)	1.20 V	500	MHz
f_{CCLK}	CLK Frequency (V_{DDINT} = 1.045 V Minimum)	1.10 V	444	MHz
f_{CCLK}	CLK Frequency (V_{DDINT} = 0.95 V Minimum)	1.00 V	400	MHz

Table 12. Phase-Locked Loop Operating Conditions

Parameter	Description	Min	Max	Unit
f_{VCO}	Voltage Controlled Oscillator (VCO) Frequency	50	Max f_{CCLK}	MHz

Table 13. System Clock (SCLK) Requirements

Parameter[1]	Description	Max	Unit
f_{SCLK}	CLKOUT/SCLK Frequency ($V_{DDINT} \geq$ 1.14 V)	133[2]	MHz
f_{SCLK}	CLKOUT/SCLK Frequency (V_{DDINT} < 1.14 V)	100	MHz

[1] t_{SCLK} (= $1/f_{SCLK}$) must be greater than or equal to t_{CCLK}
[2] Guaranteed to t_{SCLK} = 7.5 ns. See Table 24 on Page 36.

ELECTRICAL CHARACTERISTICS

Parameter[1]		Test Conditions	Min	Typ	Max	Unit
V_{OH}	High Level Output Voltage[2]	V_{DDEXT} = +3.0 V, I_{OH} = –0.5 mA	2.4			V
V_{OL}	Low Level Output Voltage[2]	V_{DDEXT} = 3.0 V, I_{OL} = 2.0 mA			0.4	V
I_{IH}	High Level Input Current[3]	V_{DDEXT}= Maximum, V_{IN} = V_{DD} Maximum			10.0	µA
I_{IHP}	High Level Input Current JTAG[4]	V_{DDEXT} = Maximum, V_{IN} = V_{DD} Maximum			50.0	µA
I_{IL}	Low Level Input Current[3]	V_{DDEXT} = Maximum, V_{IN} = 0 V			10.0	µA
I_{OZH}	Three-State Leakage Current[5]	V_{DDEXT} = Maximum, V_{IN} = V_{DD} Maximum			10.0	µA
I_{OZL}	Three-State Leakage Current[5]	V_{DDEXT} = Maximum, V_{IN} = 0 V			10.0	µA
C_{IN}	Input Capacitance[6, 7]	f_{CCLK} = 1 MHz, $T_{AMBIENT}$ = 25°C, V_{IN} = 2.5 V		4	8	pF
$I_{DDDEEPSLEEP}$[8]	V_{DDINT} Current in Deep Sleep Mode	V_{DDINT} = 1.0 V, f_{CCLK} = 0 MHz, T_J = 25°C, ASF = 0.00		7.5		mA
$I_{DDSLEEP}$	V_{DDINT} Current in Sleep Mode	V_{DDINT} = 0.8 V, T_J = 25°C, SCLK = 25 MHz			10	mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.14 V, f_{CCLK} = 400 MHz, T_J = 25°C		130		mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.2 V, f_{CCLK} = 500 MHz, T_J = 25°C		168		mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.2 V, f_{CCLK} = 533 MHz, T_J = 25°C		180		mA
$I_{DDHIBERNATE}$[8]	V_{DDEXT} Current in Hibernate State	V_{DDEXT} = 3.6 V, CLKIN=0 MHz, T_J = Maximum, voltage regulator off (V_{DDINT} = 0 V)		50	100	µA
I_{DDRTC}	V_{DDRTC} Current	V_{DDRTC} = 3.3 V, T_J = 25°C		20		µA
I_{DDINT}[9]	V_{DDINT} Current	f_{CCLK} > 0 MHz			Table 14 + (Table 16 × ASF)	mA
$I_{DDDEEPSLEEP}$[8]	V_{DDINT} Current in Deep Sleep Mode	f_{CCLK} = 0 MHz		6	Table 14	mA

[1] Specifications subject to change without notice.
[2] Applies to output and bidirectional pins.
[3] Applies to input pins except JTAG inputs.
[4] Applies to JTAG input pins (TCK, TDI, TMS, $\overline{TRST}$).
[5] Applies to three-statable pins.
[6] Applies to all signal pins.
[7] Guaranteed, but not tested.
[8] See the *ADSP-BF539 Blackfin Processor Hardware Reference Manual* for definitions of sleep, deep sleep, and hibernate operating modes.
[9] See Table 15 for the list of I_{DDINT} power vectors covered by various activity scaling factors (ASF).

System designers should refer to *Estimating Power for the ADSP-BF538/BF539 Blackfin Processors (EE-298)*, which provides detailed information for optimizing designs for lowest power. All topics discussed in this section are described in detail in EE-298. Total power dissipation has two components:

1. Static, including leakage current
2. Dynamic, due to transistor switching characteristics

Many operating conditions can also affect power dissipation, including temperature, voltage, operating frequency, and processor activity. Electrical Characteristics on Page 27 shows the current dissipation for internal circuitry (V_{DDINT}). $I_{DDDEEPSLEEP}$ specifies static power dissipation as a function of voltage (V_{DDINT}) and temperature (see Table 14), and I_{DDINT} specifies the total power specification for the listed test conditions, including the dynamic component as a function of voltage (V_{DDINT}) and frequency (Table 16).

The dynamic component is also subject to an Activity Scaling Factor (ASF) which represents application code running on the processor (Table 15).

Table 14. Static Current (mA)[1]

	V_{DDINT} (V)												
T_J (°C)	0.80 V	0.85 V	0.90 V	0.95 V	1.00 V	1.05 V	1.10 V	1.15 V	1.20 V	1.25 V	1.30 V	1.32 V	1.375 V
-40	6.4	7.7	8.8	10.4	12.0	14.0	16.1	18.9	21.9	25.2	28.7	30.6	35.9
-25	9.2	10.9	12.5	14.5	16.7	19.3	22.1	25.6	29.5	33.7	38.1	40.5	47.2
0	16.8	18.9	21.5	24.4	27.7	31.7	35.8	40.5	45.8	51.6	58.2	61.0	69.8
25	32.9	37.2	41.4	46.2	51.8	57.4	64.2	72.3	80.0	89.3	98.9	103.3	116.4
40	48.4	54.8	60.5	67.1	74.7	82.9	91.6	101.5	112.4	123.2	136.2	142.0	158.7
55	71.2	78.6	86.5	95.8	104.9	115.7	127.1	139.8	153.6	168.0	183.7	191.0	211.8
70	102.3	112.2	122.1	133.5	146.1	159.2	173.9	189.8	206.7	225.5	245.6	254.1	279.6
85	140.7	153.0	167.0	182.5	198.0	216.0	234.3	254.0	276.0	299.1	324.3	334.8	366.6
100	190.6	207.1	224.6	244.0	265.6	285.7	309.0	333.7	360.0	387.8	417.3	431.1	469.3
105	210.2	228.1	245.1	265.6	285.8	309.2	334.0	360.1	385.6	417.2	448.0	461.5	501.1

[1] Values are guaranteed maximum $I_{DDDEEPSLEEP}$ specifications.

Table 15. Activity Scaling Factors

I_{DDINT} Power Vector[1]	Activity Scaling Factor (ASF)[2]
$I_{DD-PEAK-MXVR}$	1.36
$I_{DD-HIGH-MXVR}$	1.32
$I_{DD-PEAK}$	1.30
$I_{DD-HIGH}$	1.28
$I_{DD-TYP-MXVR}$	1.07
I_{DD-TYP}	1.00
$I_{DD-APP-MXVR}$	0.92
I_{DD-APP}	0.88
$I_{DD-NOP-MXVR}$	0.76
I_{DD-NOP}	0.74
$I_{DD-IDLE-MXVR}$	0.50
$I_{DD-IDLE}$	0.48

[1] See EE-298 for power vector definitions.
[2] All ASF values determined using a 10:1 CCLK:SCLK ratio.

Table 16. Dynamic Current (mA, with ASF = 1.0)[1]

Frequency (MHz)	V_{DDINT}									
	0.95 V	1.00 V	1.05 V	1.10 V	1.15 V	1.20 V	1.25 V	1.30 V	1.32 V	1.375 V
50	17.5	19.1	20.5	22.0	23.5	25.4	27.1	29.1	29.7	31.6
100	30.1	32.3	34.4	37.0	39.2	41.7	44.3	46.4	47.6	50.3
200	54.8	58.4	61.8	65.6	69.7	74.3	76.2	82.2	83.4	87.8
250	66.8	71.2	75.7	79.9	84.5	89.8	94.2	99.4	101.2	106.5
300	79.3	84.5	89.0	94.7	100.0	105.5	111.6	116.8	119.3	125.5
375	97.9	103.9	109.9	116.5	122.2	129.7	136.0	142.9	145.9	153.6
400	103.8	110.3	116.9	123.7	130.0	137.5	144.2	151.2	154.5	162.4
425	N/A	116.6	123.7	130.9	137.2	144.7	152.7	159.9	163.3	171.8
475	N/A	N/A	N/A	145.0	151.8	161.4	169.4	177.8	181.1	190.4
500	N/A	N/A	N/A	N/A	159.9	168.9	177.8	186.3	190.0	199.6
533	N/A	N/A	N/A	N/A	N/A	179.8	188.9	198.8	202.2	212.5

[1] The values are not guaranteed as standalone maximum specifications, they must be combined with static current per the equations of Electrical Characteristics on Page 27.

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 (Core) Supply Voltage (V_{DDINT})[1]	–0.3 V to +1.4 V
External (I/O) Supply Voltage (V_{DDEXT})[2]	–0.3 V to +3.8 V
Input Voltage[3, 4]	–0.5 V to +3.6 V
Input Voltage[4, 5]	–0.5 V to +5.5 V
Output Voltage Swing	–0.5 V to V_{DDEXT} + 0.5 V
Load Capacitance[6]	200 pF
Junction Temperature Under Bias	+125°C
Storage Temperature Range	–65°C to +150°C

[1] Parameter value applies also to MPIVDD.
[2] Parameter value applies also to MXEVDD and VDDRTC.
[3] Applies to 100% transient duty cycle. For other duty cycles, see Table 17.
[4] Applies only when V_{DDEXT} is within specifications. When V_{DDEXT} is outside specifications, the range is V_{DDEXT} ± 0.2 V.
[5] Applies to pins designated as 5 V tolerant only.
[6] For proper SDRAM controller operation, the maximum load capacitance is 50 pF for ADDR19–1, DATA15–0, ABE1–0/SDQM1–0, CLKOUT, SCKE, SA10, SRAS, SCAS, SWE, and SMS.

Table 17. Maximum Duty Cycle for Input Transient Voltage[1]

V_{IN} Min (V)	V_{IN} Max (V)[2]	Maximum Duty Cycle
–0.50	+3.80	100%
–0.70	+4.00	40%
–0.80	+4.10	25%
–0.90	+4.20	15%
–1.00	+4.30	10%

[1] Applies to all signal pins with the exception of CLKIN, MXI, MXO, MLF, VROUT1–0, XTAL, RTXI, and RTXO
[2] Only one of the listed options can apply to a particular design.

ESD SENSITIVITY



ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection 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 10 and Table 18 provides information about how to read the package brand and relate it to specific product features. For a complete listing of product offerings, see the Ordering Guide on Page 60.



Figure 10. Product Information on Package

Table 18. Package Brand Information

Brand Key	Field Description
Fx	On-Chip Flash Option: x = 4 or 8 (M bit)
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Part
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
#	RoHS Compliant Designation
yyww	Date Code

TIMING SPECIFICATIONS

Clock and Reset Timing

Table 19 and Figure 11 describe clock and reset operations. Per Absolute Maximum Ratings on Page 30, combinations of CLKIN and clock multipliers must not select core/peripheral clocks that exceed maximum operating conditions.

Table 19. Clock and Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{CKIN}	CLKIN Period[1, 2, 3]	20.0	100.0	ns
t_{CKINL}	CLKIN Low Pulse	8.0		ns
t_{CKINH}	CLKIN High Pulse	8.0		ns
t_{WRST}	$\overline{\text{RESET}}$ Asserted Pulse Width Low[4]	11 t_{CKIN}		ns

[1] Applies to PLL bypass mode and PLL non-bypass mode.
[2] If the DF bit in the PLL_CTL register is set, then the maximum t_{CKIN} period is 50 ns.
[3] CLKIN frequency must not change on the fly.
[4] 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 startup time of external clock oscillator).

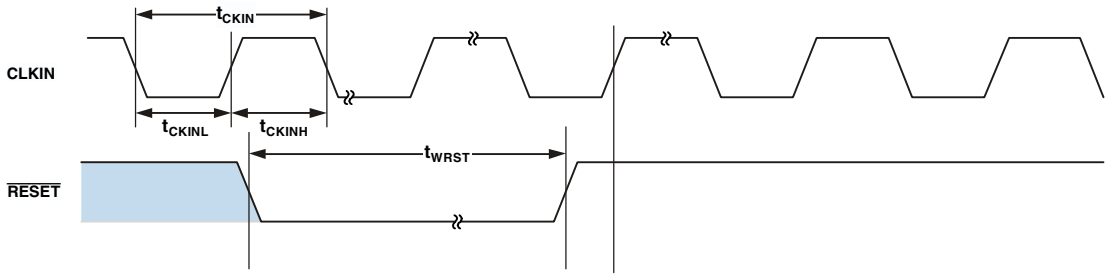


Figure 11. Clock and Reset Timing

Asynchronous Memory Read Cycle Timing

Table 20 and Table 21 on Page 33 and Figure 12 and Figure 13 on Page 33 describe asynchronous memory read cycle operations for synchronous and for asynchronous ARDY.

Table 20. Asynchronous Memory Read Cycle Timing with Synchronous ARDY

Parameter		Min	Max	Unit
Timing Requirements				
t_{SDAT}	DATA15–0 Setup Before CLKOUT	2.1		ns
t_{HDAT}	DATA15–0 Hold After CLKOUT	0.8		ns
t_{SARDY}	ARDY Setup Before the Falling Edge of CLKOUT	4.0		ns
t_{HARDY}	ARDY Hold After the Falling Edge of CLKOUT	0.0		ns
t_{DO}	Output Delay After CLKOUT[1]		6.0	ns
t_{HO}	Output Hold After CLKOUT[1]	0.8		ns

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

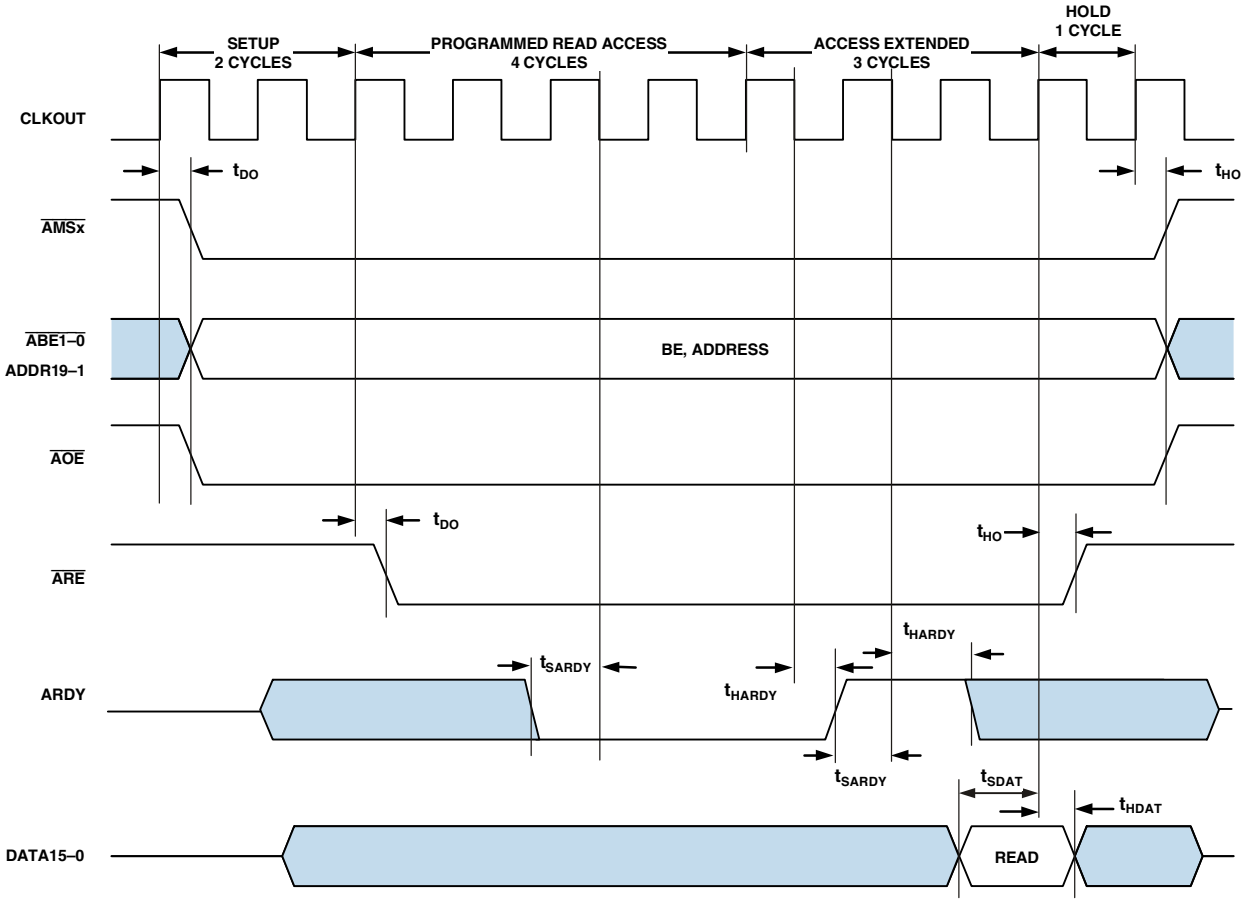


Figure 12. Asynchronous Memory Read Cycle Timing with Synchronous ARDY

Table 21. Asynchronous Memory Read Cycle Timing with Asynchronous ARDY

Parameter		Min	Max	Unit
Timing Requirements				
t_{SDAT}	DATA15–0 Setup Before CLKOUT	2.1		ns
t_{HDAT}	DATA15–0 Hold After CLKOUT	0.8		ns
t_{DANR}	ARDY Negated Delay from $\overline{AMSx}$ Asserted[1]		$(S + RA - 2) \times t_{SCLK}$	ns
t_{HAA}	ARDY Asserted Hold After $\overline{ARE}$ Negated	0.0		ns
t_{DO}	Output Delay After CLKOUT[2]		6.0	ns
t_{HO}	Output Hold After CLKOUT[2]	0.8		ns

[1] S = number of programmed setup cycles, RA = number of programmed read access cycles.
[2] Output pins include $\overline{AMS3-0}$, $\overline{ABE1-0}$, ADDR19–1, $\overline{AOE}$, $\overline{ARE}$.

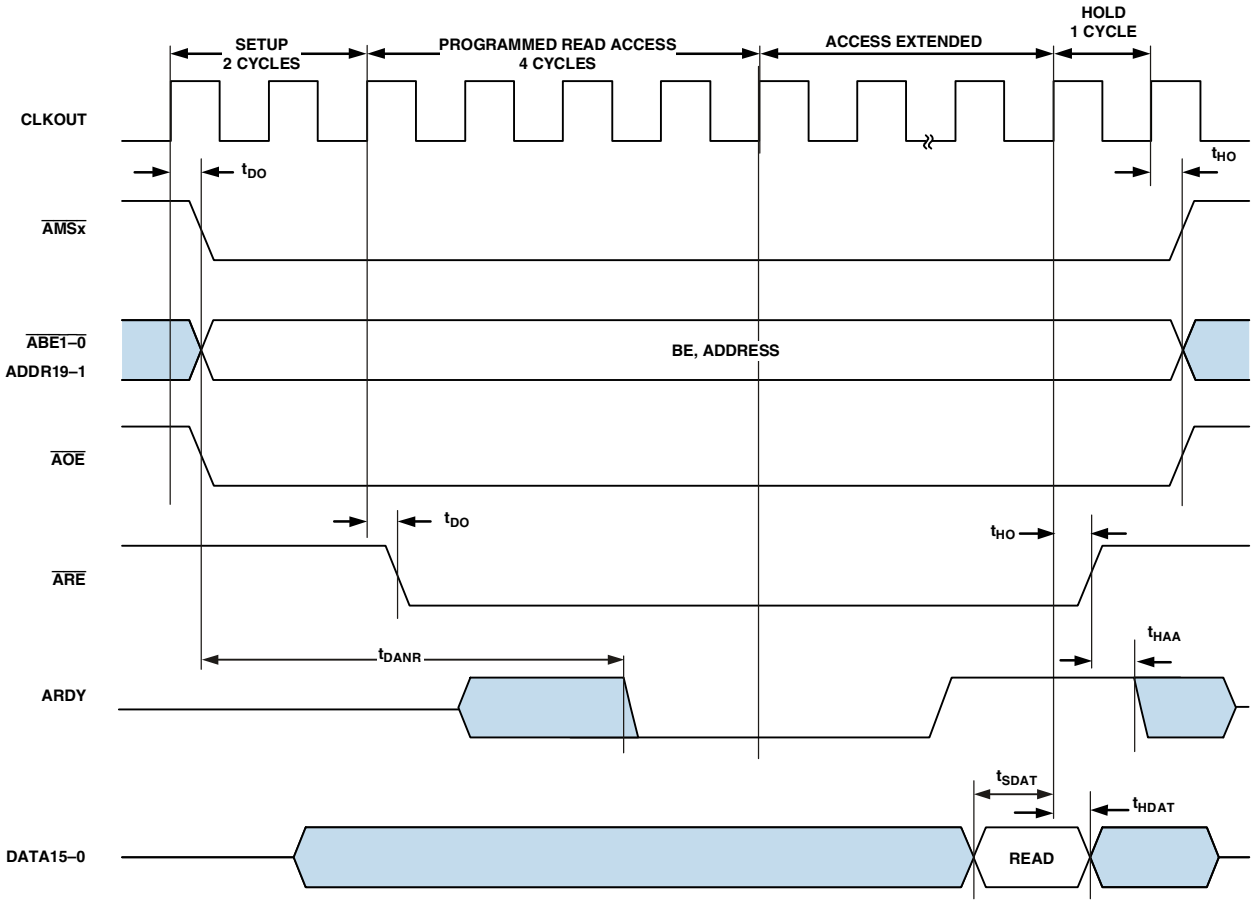


Figure 13. Asynchronous Memory Read Cycle Timing with Asynchronous ARDY

Asynchronous Memory Write Cycle Timing

Table 22 and Table 23 and Figure 14 and Figure 15 describe asynchronous memory write cycle operations for synchronous and for asynchronous ARDY.

Table 22. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

Parameter		Min	Max	Unit
Timing Requirements				
t_{SARDY}	ARDY Setup Before the Falling Edge of CLKOUT	4.0		ns
t_{HARDY}	ARDY Hold After the Falling Edge of CLKOUT	0.0		ns
Switching Characteristics				
t_{DDAT}	DATA15–0 Disable After CLKOUT		6.0	ns
t_{ENDAT}	DATA15–0 Enable After CLKOUT	1.0		ns
t_{DO}	Output Delay After CLKOUT[1]		6.0	ns
t_{HO}	Output Hold After CLKOUT[1]	0.8		ns

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

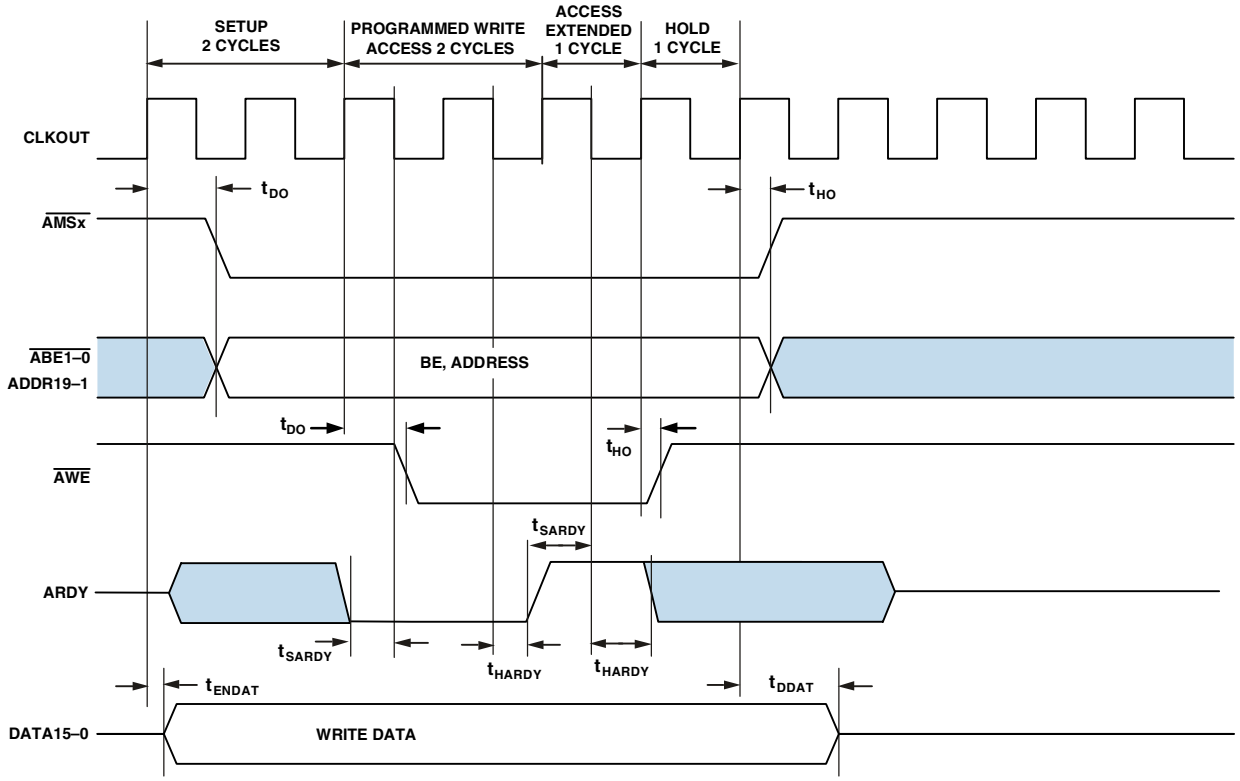


Figure 14. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

Table 23. Asynchronous Memory Write Cycle Timing with Asynchronous ARDY

Parameter		Min	Max	Unit
Timing Requirements				
t_{DANR}	ARDY Negated Delay from $\overline{AMSx}$ Asserted[1]		$(S + WA - 2) \times t_{SCLK}$	ns
t_{HAA}	ARDY Asserted Hold After $\overline{ARE}$ Negated	0.0		ns
Switching Characteristics				
t_{DDAT}	DATA15–0 Disable After CLKOUT		6.0	ns
t_{ENDAT}	DATA15–0 Enable After CLKOUT	1.0		ns
t_{DO}	Output Delay After CLKOUT[2]		6.0	ns
t_{HO}	Output Hold After CLKOUT[2]	0.8		ns

[1] S = Number of programmed setup cycles, WA = Number of programmed write access cycles.
[2] Output pins include $\overline{AMS3-0}$, $\overline{ABE1-0}$, ADDR19–1, DATA15–0, $\overline{AOE}$, $\overline{AWE}$.

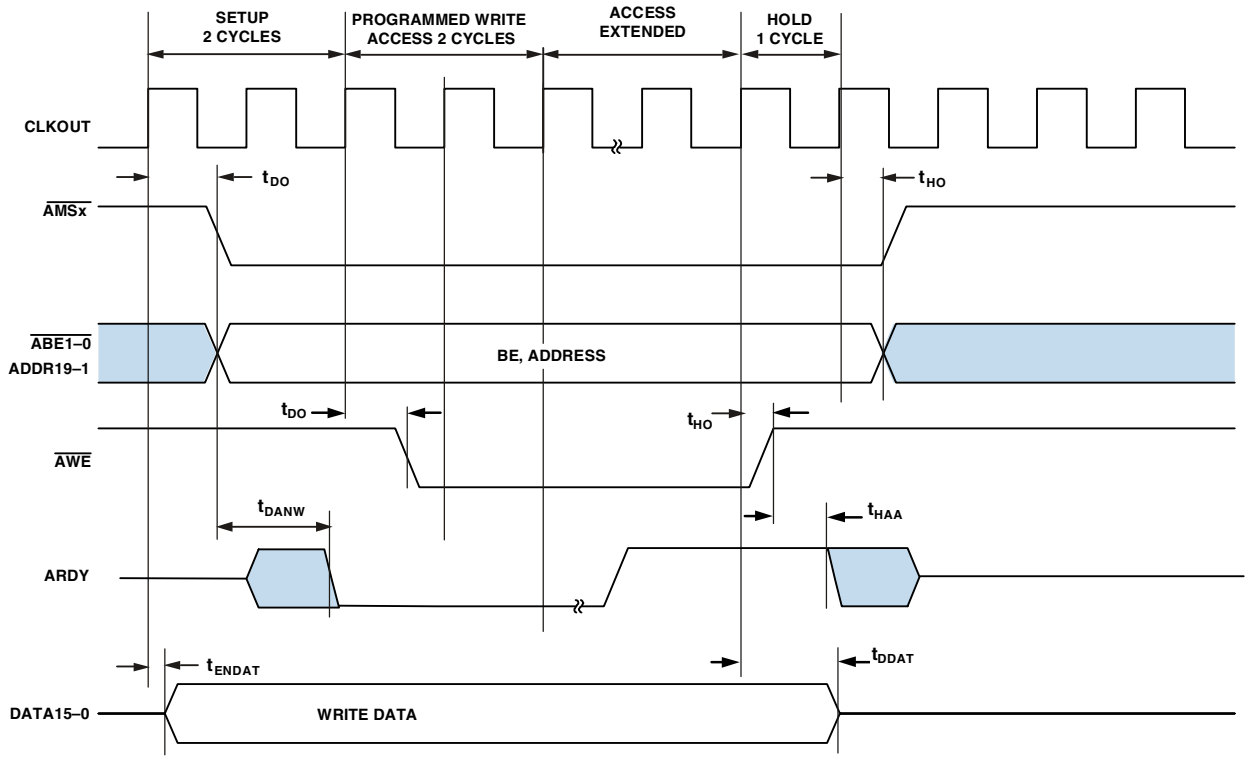


Figure 15. Asynchronous Memory Write Cycle Timing with Asynchronous ARDY

SDRAM Interface Timing

Table 24. SDRAM Interface Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{SSDAT}	DATA Setup Before CLKOUT	2.1		ns
t_{HSDAT}	DATA Hold After CLKOUT	0.8		ns
Switching Characteristics				
t_{SCLK}	CLKOUT Period[1]	7.5		ns
t_{SCLKH}	CLKOUT Width High	2.5		ns
t_{SCLKL}	CLKOUT Width Low	2.5		ns
t_{DCAD}	Command, ADDR, Data Delay After CLKOUT[2]		6.0	ns
t_{HCAD}	Command, ADDR, Data Hold After CLKOUT[2]	0.8		ns
t_{DSDAT}	Data Disable After CLKOUT		6.0	ns
t_{ENSDAT}	Data Enable After CLKOUT	1.0		ns

[1] SDRAM timing for $T_{JUNCTION}$ = 125°C is limited to 100 MHz.
[2] Command pins include: $\overline{SRAS}$, $\overline{SCAS}$, $\overline{SWE}$, SDQM, $\overline{SMS}$, SA10, SCKE.

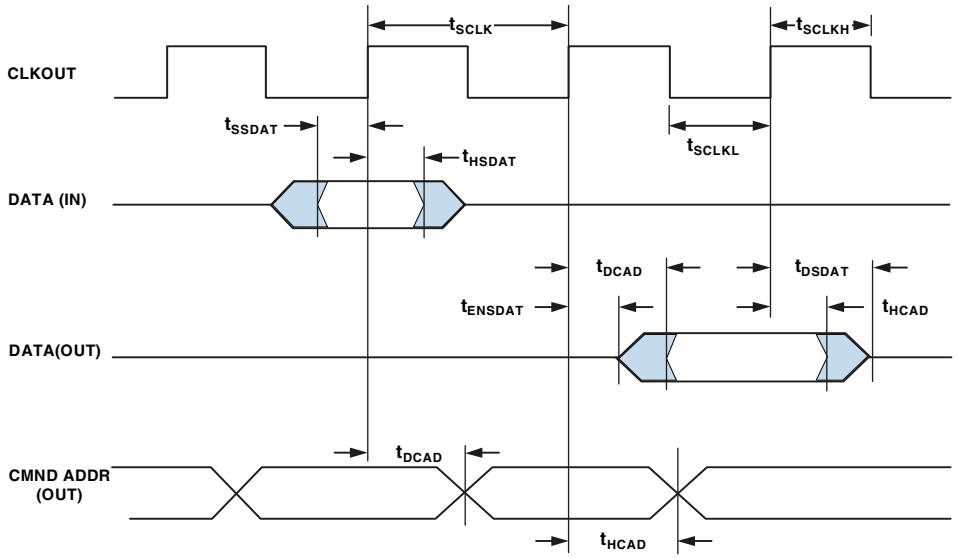


Figure 16. SDRAM Interface Timing

External Port Bus Request and Grant Cycle Timing

Table 25 and Table 26 and Figure 17 and Figure 18 describe external port bus request and grant cycle operations for synchronous and for asynchronous $\overline{BR}$.

Table 25. External Port Bus Request and Grant Cycle Timing with Synchronous $\overline{BR}$

Parameter		Min	Max	Unit
Timing Requirements				
t_{BS}	$\overline{BR}$ Setup to Falling Edge of CLKOUT	4.6		ns
t_{BH}	Falling Edge of CLKOUT to $\overline{BR}$ Deasserted Hold Time	1.0		ns
Switching Characteristics				
t_{SD}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}$/$\overline{AWE}$ Disable		4.5	ns
t_{SE}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}$/$\overline{AWE}$ Enable		4.5	ns
t_{DBG}	CLKOUT High to $\overline{BG}$ High Setup		4.0	ns
t_{EBG}	CLKOUT High to $\overline{BG}$ Deasserted Hold Time		4.0	ns
t_{DBH}	CLKOUT High to $\overline{BGH}$ High Setup		3.6	ns
t_{EBH}	CLKOUT High to $\overline{BGH}$ Deasserted Hold Time		3.6	ns

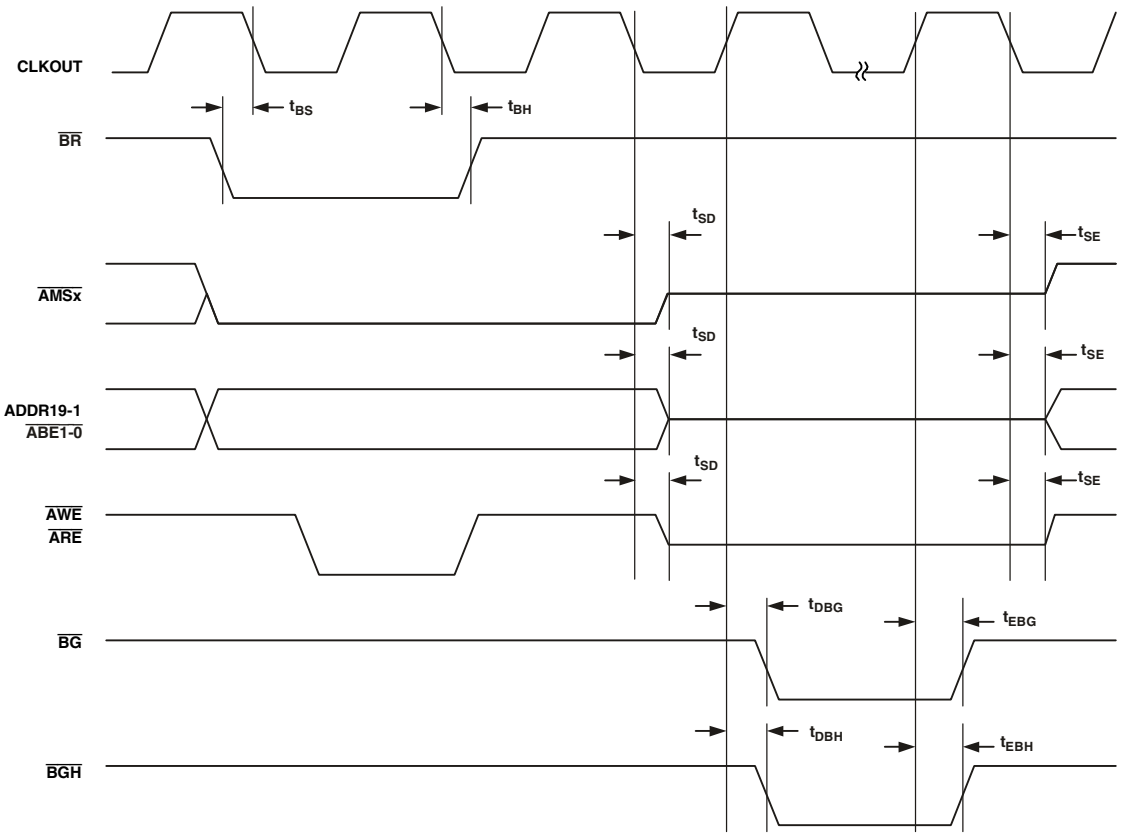


Figure 17. External Port Bus Request and Grant Cycle Timing with Synchronous $\overline{BR}$

Table 26. External Port Bus Request and Grant Cycle Timing with Asynchronous $\overline{BR}$

Parameter		Min	Max	Unit
Timing Requirement				
t_{WBR}	$\overline{BR}$ Pulse Width	$2 \times t_{SCLK}$		ns
Switching Characteristics				
t_{SD}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}$/$\overline{AWE}$ Disable		4.5	ns
t_{SE}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}$/$\overline{AWE}$ Enable		4.5	ns
t_{DBG}	CLKOUT High to $\overline{BG}$ High Setup		3.6	ns
t_{EBG}	CLKOUT High to $\overline{BG}$ Deasserted Hold Time		3.6	ns
t_{DBH}	CLKOUT High to $\overline{BGH}$ High Setup		3.6	ns
t_{EBH}	CLKOUT High to $\overline{BGH}$ Deasserted Hold Time		3.6	ns

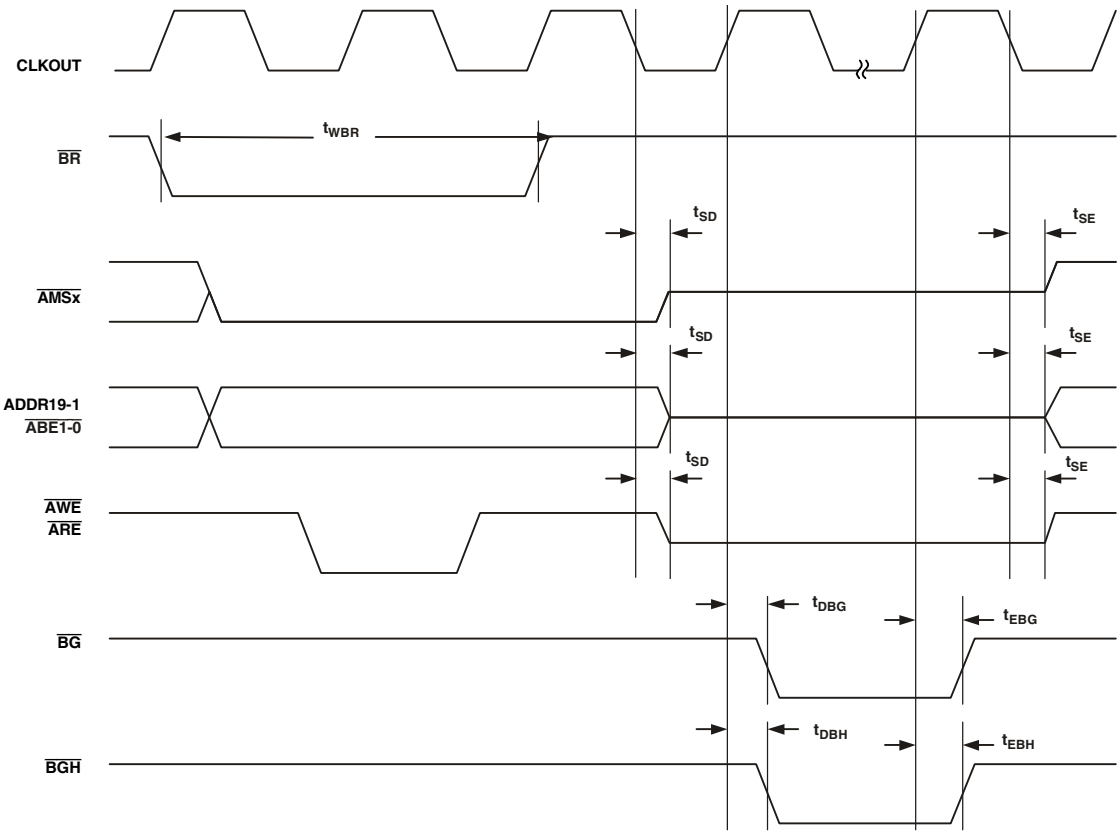


Figure 18. External Port Bus Request and Grant Cycle Timing with Asynchronous $\overline{BR}$

Parallel Peripheral Interface Timing

Table 27 and Figure 19, Figure 20, Figure 21, and Figure 22 describe Parallel Peripheral Interface operations.

Table 27. Parallel Peripheral Interface Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{PCLKW}	PPI_CLK Width	6.0		ns
t_{PCLK}	PPI_CLK Period[1]	15.0		ns
t_{SFSPE}	External Frame Sync Setup Before PPI_CLK	5.0		ns
t_{HFSPE}	External Frame Sync Hold After PPI_CLK	1.0		ns
t_{SDRPE}	Receive Data Setup Before PPI_CLK	2.0		ns
t_{HDRPE}	Receive Data Hold After PPI_CLK	4.0		ns
Switching Characteristics—GP Output and Frame Capture Modes				
t_{DFSPE}	Internal Frame Sync Delay After PPI_CLK		10.0	ns
t_{HOFSPE}	Internal Frame Sync Hold After PPI_CLK	0.0		ns
t_{DDTPE}	Transmit Data Delay After PPI_CLK		10.0	ns
t_{HDTPE}	Transmit Data Hold After PPI_CLK	0.0		ns

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

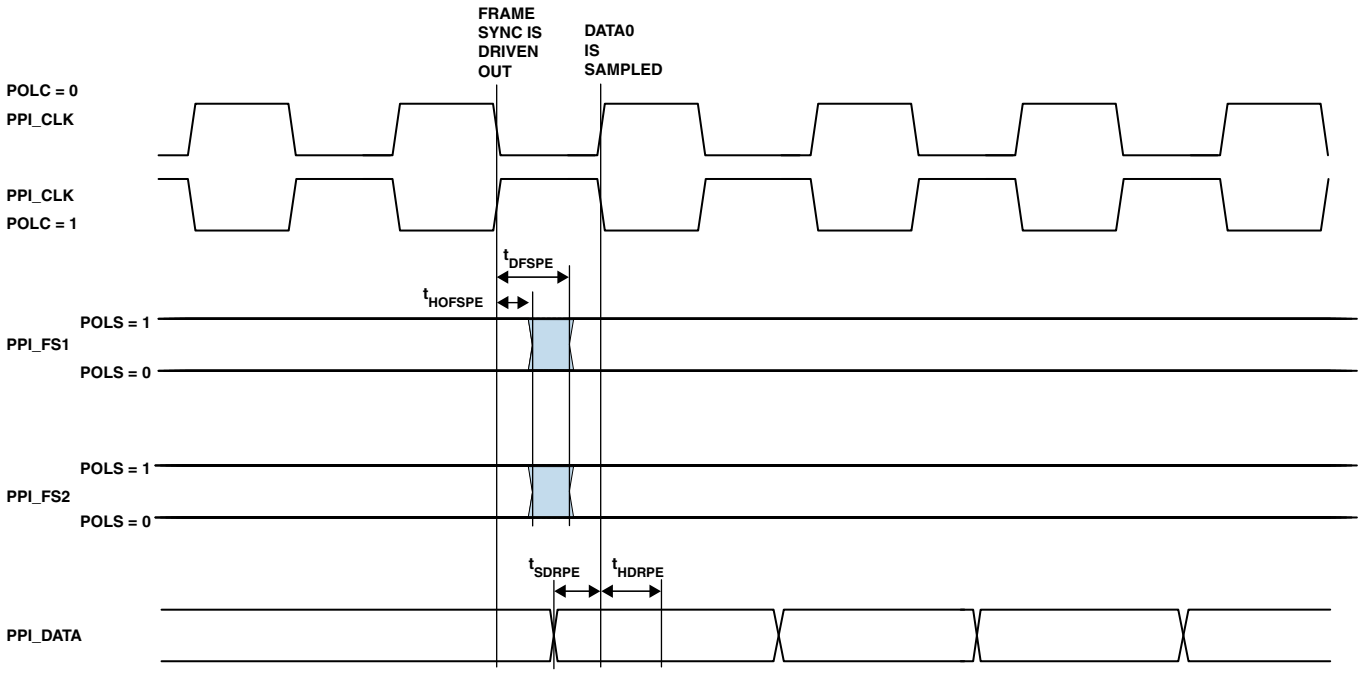


Figure 19. PPI GP Rx Mode with Internal Frame Sync Timing

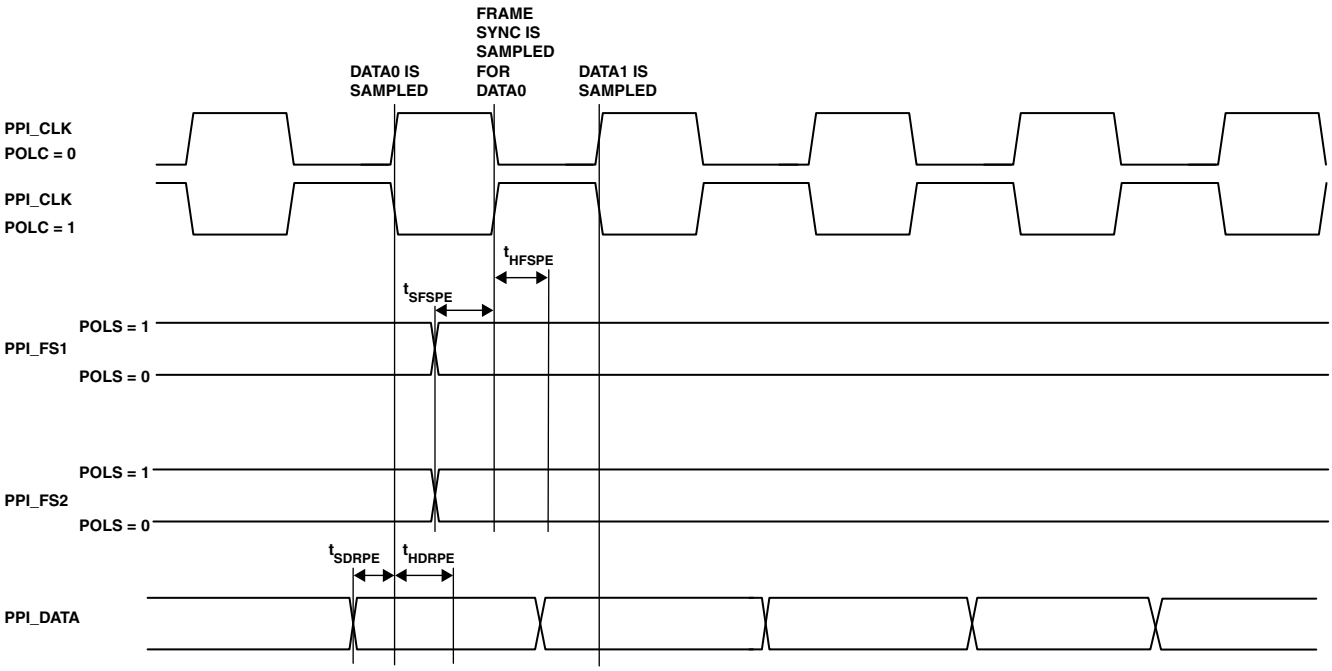


Figure 20. PPI GP Rx Mode with External Frame Sync Timing

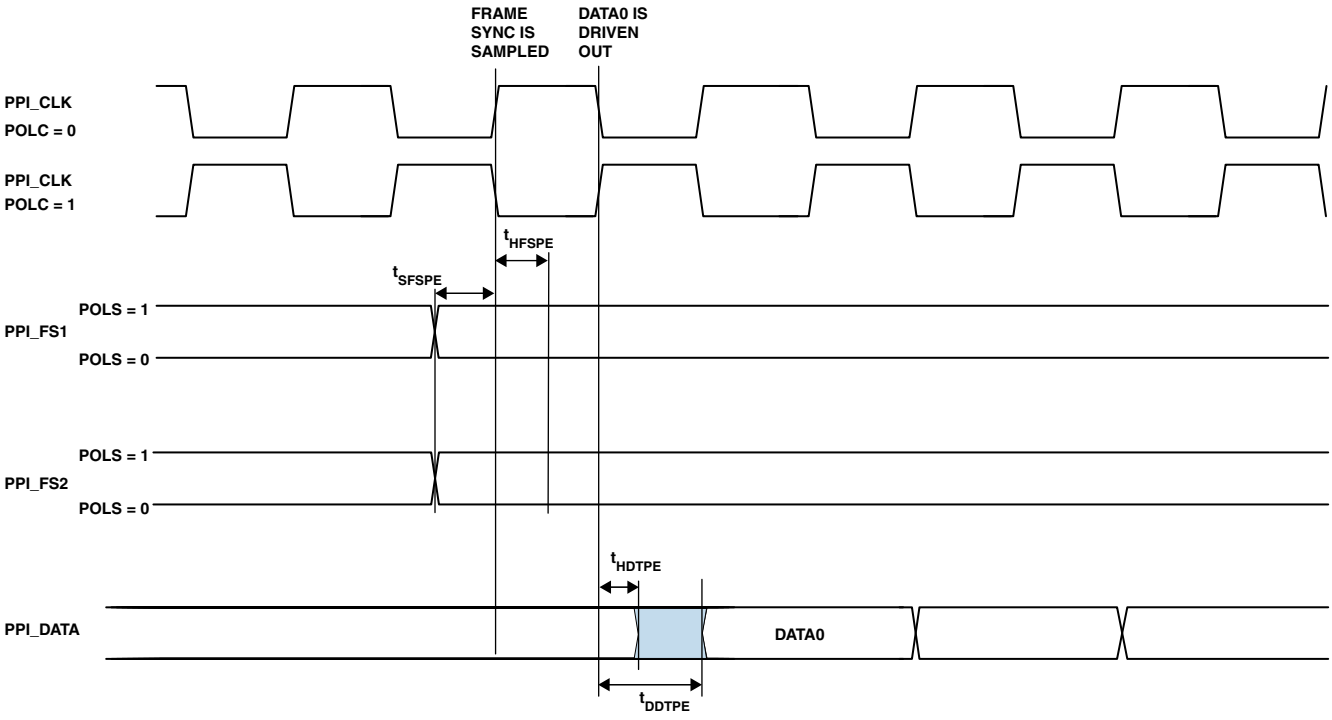


Figure 21. PPI GP Tx Mode with External Frame Sync Timing

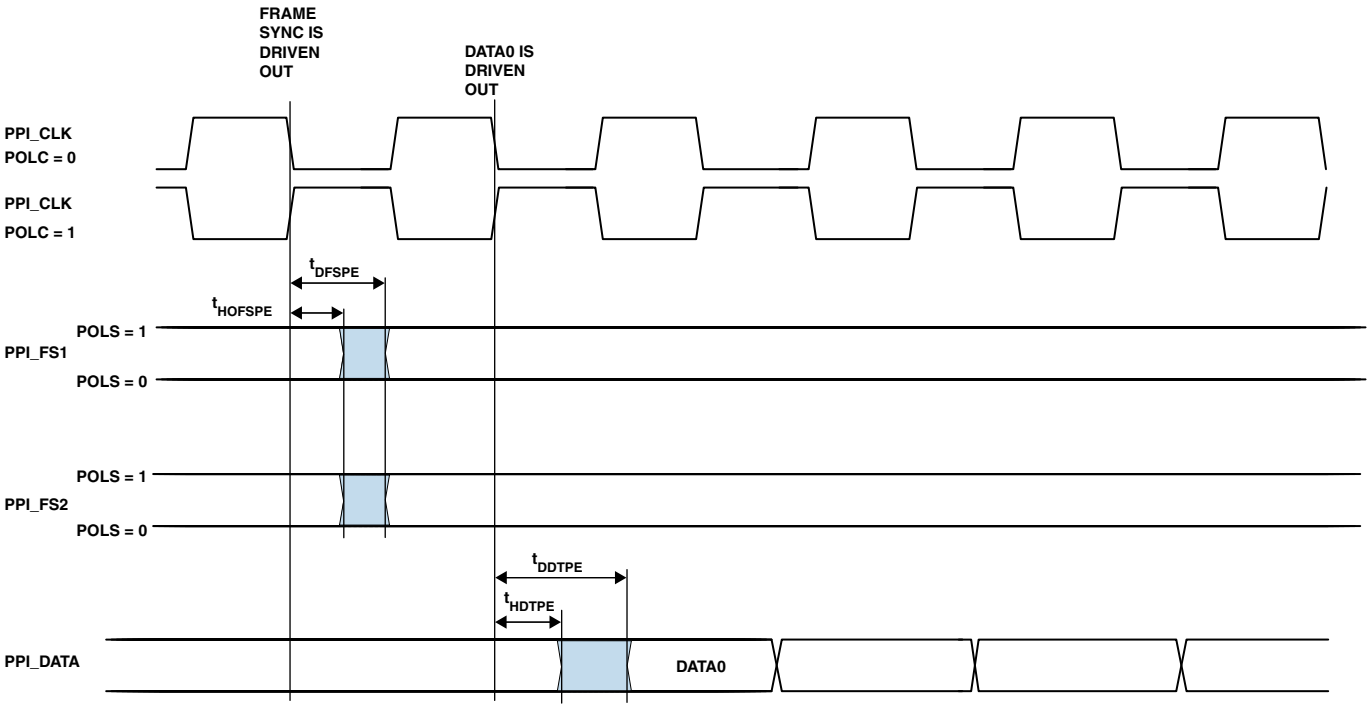


Figure 22. PPI GP Tx Mode with Internal Frame Sync Timing

Serial Ports Timing

Table 28 through Table 31 and Figure 23 through Figure 24 describe Serial Port operations.

Table 28. Serial Ports—External Clock

Parameter		Min	Max	Unit
Timing Requirements				
t_{SFSE}	TFSx/RFSx Setup Before TSCLKx/RSCLKx (Externally Generated TFSx/RFSx)[1]	3.0		ns
t_{HFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Externally Generated TFSx/RFSx)[1]	3.0		ns
t_{SDRE}	Receive Data Setup Before RSCLKx[1]	3.0		ns
t_{HDRE}	Receive Data Hold After RSCLKx[1]	3.0		ns
t_{SCLKEW}	TSCLKx/RSCLKx Width	4.5		ns
t_{SCLKE}	TSCLKx/RSCLKx Period	15.0		ns
Switching Characteristics				
t_{DFSE}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]		10.0	ns
t_{HOFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]	0.0		ns
t_{DDTE}	Transmit Data Delay After TSCLKx[2]		10.0	ns
t_{HDTE}	Transmit Data Hold After TSCLKx[2]	0.0		ns

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

Table 29. Serial Ports—Internal Clock

Parameter		Min	Max	Unit
Timing Requirements				
t_{SFSI}	TFSx/RFSx Setup Before TSCLKx/RSCLKx (Externally Generated TFSx/RFSx)[1]	8.0		ns
t_{HFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Externally Generated TFSx/RFSx)[1]	–1.5		ns
t_{SDRI}	Receive Data Setup Before RSCLKx[1]	9.0		ns
t_{HDRI}	Receive Data Hold After RSCLKx[1]	–1.5		ns
Switching Characteristics				
t_{DFSI}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]		3.0	ns
t_{HOFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]	–1.0		ns
t_{DDTI}	Transmit Data Delay After TSCLKx[2]		3.0	ns
t_{HDTI}	Transmit Data Hold After TSCLKx[2]	–2.0		ns
t_{SCLKIW}	TSCLKx/RSCLKx Width	4.5		ns

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

Table 30. Serial Ports—Enable and Three-State

Parameter		Min	Max	Unit
Switching Characteristics				
t_{DTENE}	Data Enable Delay from External TSCLKx[1]	0		ns
t_{DDTTE}	Data Disable Delay from External TSCLKx[1]		10.0	ns
t_{DTENI}	Data Enable Delay from Internal TSCLKx[1]	–2.0		ns
t_{DDTTI}	Data Disable Delay from Internal TSCLKx[1]		3.0	ns

[1] Referenced to drive edge.

Table 31. External Late Frame Sync

Parameter		Min	Max	Unit
Switching Characteristics				
$t_{DDTLFSE}$	Data Delay from Late External TFSx or External RFSx in multichannel mode, MFD = 0[1, 2]		10.0	ns
$t_{DTENLFS}$	Data Enable from Late FS or multichannel mode, MFD = 0[1, 2]	0		ns

[1] In multichannel mode, TFSx enable and TFSx valid follow $t_{DTENLFS}$ 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.

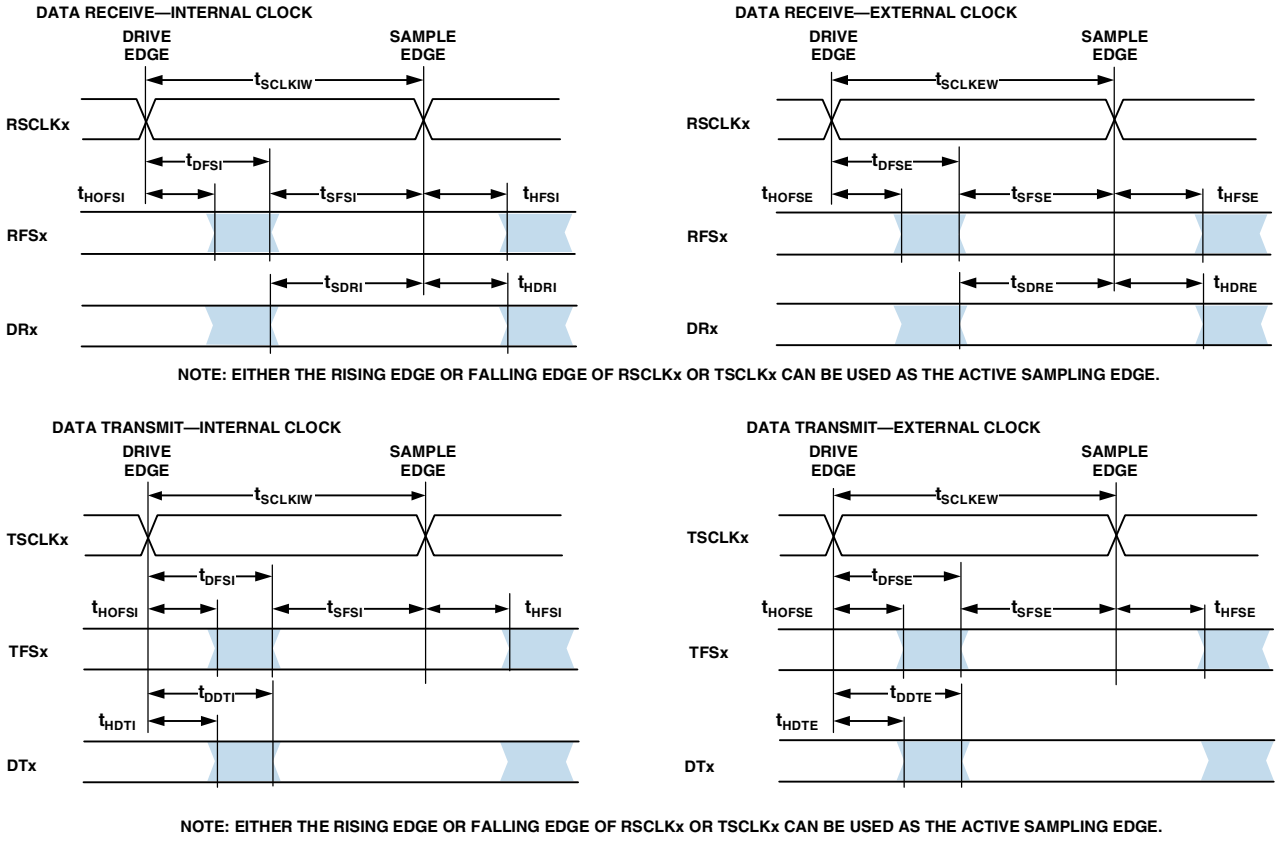


Figure 23. Serial Ports

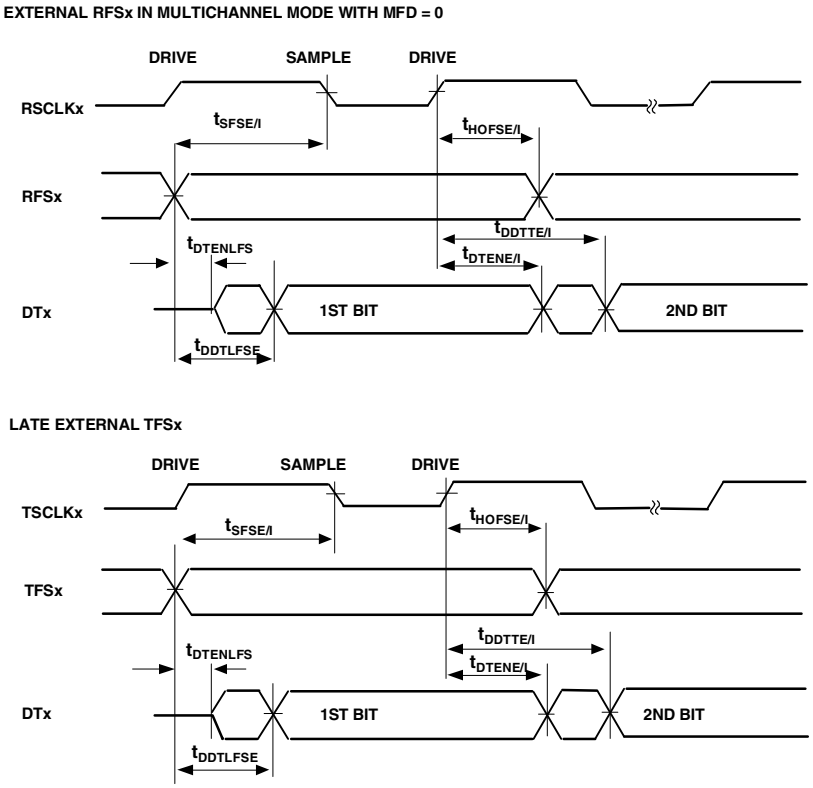


Figure 24. External Late Frame Sync

Serial Peripheral Interface Ports—Master Timing

Table 32 and Figure 25 describe SPI ports master operations.

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

Parameter		Min	Max	Unit
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SCKx Edge (Data Input Setup)	9.0		ns
t_{HSPIDM}	SCKx Sampling Edge to Data Input Invalid	–1.5		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{SPIxSELy}$ Low to First SCKx edge	$2t_{SCLK}$		ns
t_{SPICHM}	Serial Clock High Period	$2t_{SCLK} - 1.5$		ns
t_{SPICLM}	Serial Clock Low Period	$2t_{SCLK} - 1.5$		ns
t_{SPICLK}	Serial Clock Period	$4t_{SCLK}$		ns
t_{HDSM}	Last SCKx Edge to $\overline{SPIxSELy}$ High	$2t_{SCLK}$		ns
t_{SPITDM}	Sequential Transfer Delay	$2t_{SCLK}$		ns
$t_{DDSPIDM}$	SCKx Edge to Data Out Valid (Data Out Delay)		5	ns
$t_{HDSPIDM}$	SCKx Edge to Data Out Invalid (Data Out Hold)	–1.0		ns

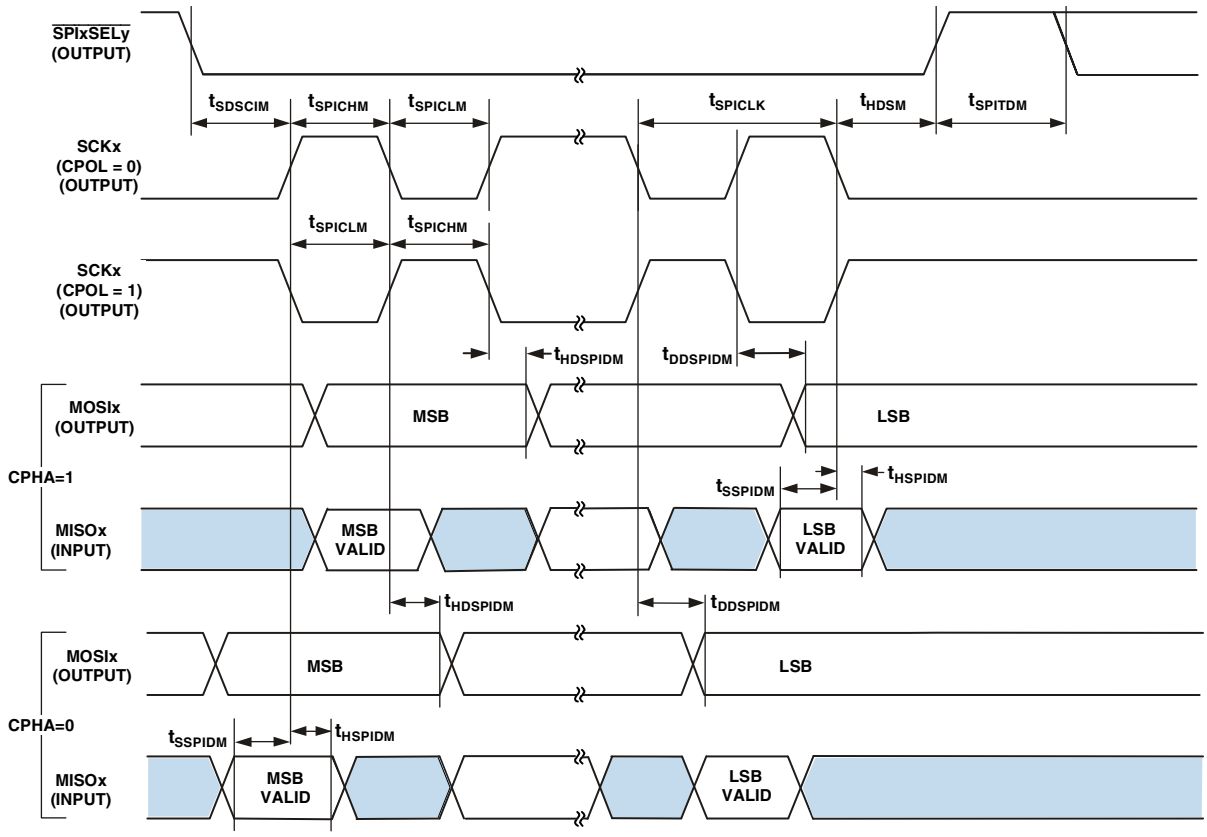


Figure 25. Serial Peripheral Interface (SPI) Ports—Master Timing

Serial Peripheral Interface Ports—Slave Timing

Table 33 and Figure 26 describe SPI ports slave operations.

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

Parameter		Min	Max	Unit
Timing Requirements				
t_{SPICHS}	Serial Clock High Period	$2t_{SCLK}-1.5$		ns
t_{SPICLS}	Serial Clock Low Period	$2t_{SCLK}-1.5$		ns
t_{SPICLK}	Serial Clock Period	$4t_{SCLK}$		ns
t_{HDS}	Last SCKx Edge to $\overline{SPIxSS}$ Not Asserted	$2t_{SCLK}$		ns
t_{SPITDS}	Sequential Transfer Delay	$2t_{SCLK}$		ns
t_{SDSCI}	$\overline{SPIxSS}$ Assertion to First SCKx Edge	$2t_{SCLK}$		ns
t_{SSPID}	Data Input Valid to SCKx Edge (Data Input Setup)	2.0		ns
t_{HSPID}	SCKx Sampling Edge to Data Input Invalid	2.0		ns
Switching Characteristics				
t_{DSOE}	$\overline{SPIxSS}$ Assertion to Data Out Active	0	8	ns
t_{DSDHI}	$\overline{SPIxSS}$ Deassertion to Data High impedance	0	8	ns
t_{DDSPID}	SCKx Edge to Data Out Valid (Data Out Delay)		10	ns
t_{HDSPID}	SCKx Edge to Data Out Invalid (Data Out Hold)	0		ns

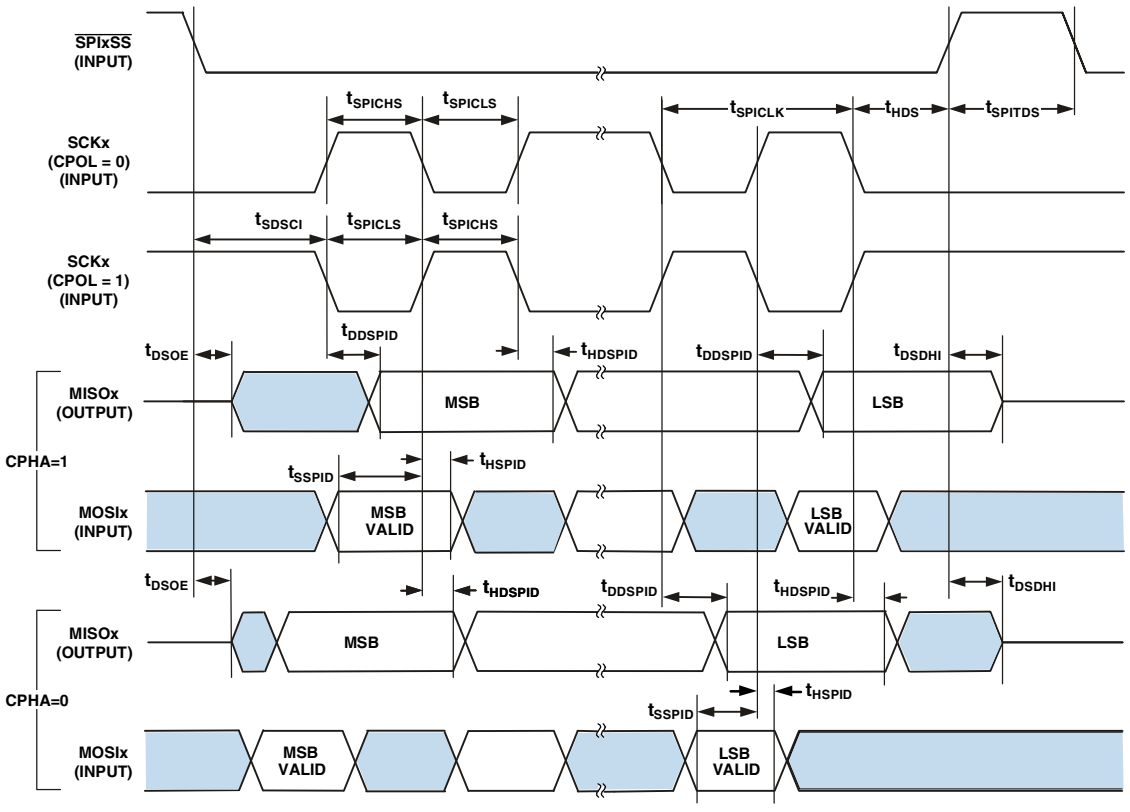


Figure 26. Serial Peripheral Interface (SPI) Ports—Slave Timing

General-Purpose Port Timing

Table 34 and Figure 27 describe general-purpose operations.

Table 34. General-Purpose Port Timing

Parameter		Min	Max	Unit
Timing Requirement				
t_{WFI}	GP Port Pin Input Pulse Width	t_{SCLK} + 1		ns
Switching Characteristic				
t_{GPOD}	GP Port Pin Output Delay from CLKOUT Low		6	ns

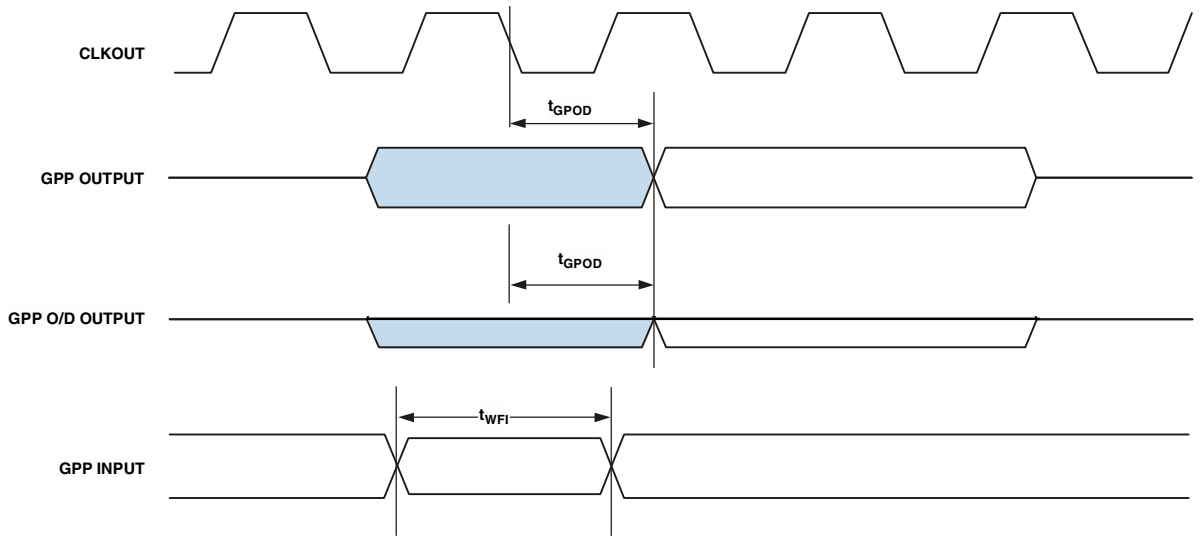


Figure 27. General-Purpose Port Cycle Timing

Timer Cycle Timing

Table 35 and Figure 28 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		Min	Max	Unit
Timing Characteristics				
t_{WL}	Timer Pulse Width Input Low[1] (Measured in SCLK Cycles)	1		SCLK
t_{WH}	Timer Pulse Width Input High[1] (Measured in SCLK Cycles)	1		SCLK
Switching Characteristic				
t_{HTO}	Timer Pulse width Output (measured in SCLK Cycles)	1	$(2^{32} - 1)$	SCLK

[1] The minimum pulse widths apply for TMRx input pins in width capture and external clock modes. They also apply to the PF1 or PPI_CLK input pins in PWM output mode.

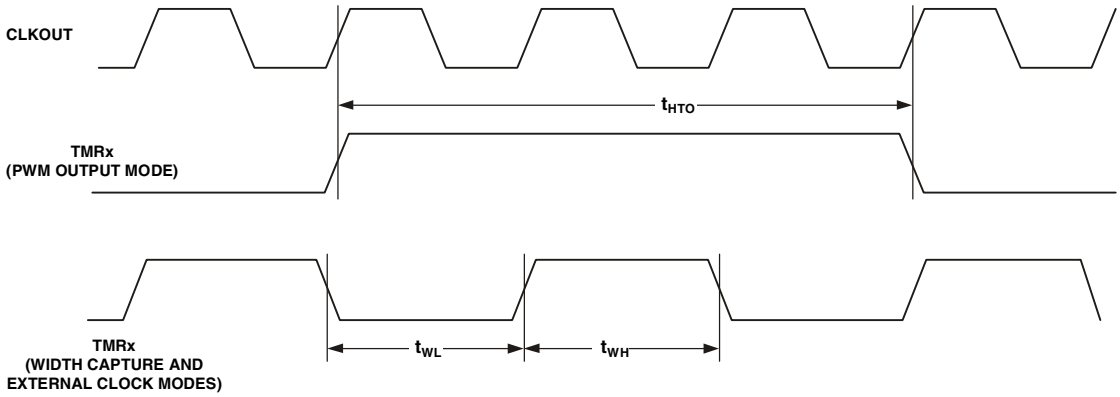


Figure 28. Timer PWM_OUT Cycle Timing

JTAG Test And Emulation Port Timing

Table 36 and Figure 29 describe JTAG port operations.

Table 36. JTAG Port Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{TCK}	TCK Period	20		ns
t_{STAP}	TDI, TMS Setup Before TCK High	4		ns
t_{HTAP}	TDI, TMS Hold After TCK High	4		ns
t_{SSYS}	System Inputs Setup Before TCK High[1]	4		ns
t_{HSYS}	System Inputs Hold After TCK High[1]	5		ns
t_{TRSTW}	$\overline{\text{TRST}}$ Pulse Width[2] (Measured in TCK Cycles)	4		TCK
Switching Characteristics				
t_{DTDO}	TDO Delay from TCK Low		10	ns
t_{DSYS}	System Outputs Delay After TCK Low[3]	0	12	ns

[1] System Inputs = ARDY, BMODE1–0, $\overline{\text{BR}}$, DATA15–0, $\overline{\text{NMI}}$, PF15–0, PPI_CLK, PPI3–0, SCL1–0, SDA1–0, $\overline{\text{MTXON}}$, $\overline{\text{MRXON}}$, MMCLK, MBCLK, MFS, MTX, MRX, $\overline{\text{SPI1SS}}$, $\overline{\text{SPI1SEL1}}$, SCK2–0, MISO2–0, MOSI2–0, $\overline{\text{SPI2SS}}$, $\overline{\text{SPI2SEL1}}$, RX2–0, TX2–1, DR0PRI, DR0SEC, DR1PRI, DR1SEC, DT2PRI, DT2SEC, DR2PRI, DR2SEC, TSCLK3–0, RSCLK3–0, TFS3–0, RFS3–0, DT3PRI, DT3SEC, DR3PRI, DR3SEC, CANTX, CANRX, $\overline{\text{RESET}}$, and TMR2–0.

[2] 50 MHz maximum

[3] System Outputs = $\overline{\text{AMS}}$, $\overline{\text{AOE}}$, $\overline{\text{ARE}}$, $\overline{\text{AWE}}$, $\overline{\text{ABE}}$, $\overline{\text{BG}}$, DATA15–0, PF15–0, PPI3–0, $\overline{\text{MTXON}}$, MMCLK, MBCLK, MFS, MTX, $\overline{\text{SPI1SS}}$, $\overline{\text{SPI1SEL1}}$, SCK2–0, MISO2–0, MOSI2–0, $\overline{\text{SPI2SS}}$, $\overline{\text{SPI2SEL1}}$, RX2–1, TX2–0, DT2PRI, DT2SEC, DR2PRI, DR2SEC, DT3PRI, DT3SEC, DR3PRI, DR3SEC, TSCLK3–0, TFS3–0, RSCLK3–0, RFS3–0, CLKOUT, CANTX, SA10, $\overline{\text{SCAS}}$, SCKE, $\overline{\text{SMS}}$, $\overline{\text{SRAS}}$, $\overline{\text{SWE}}$, and TMR2–0.

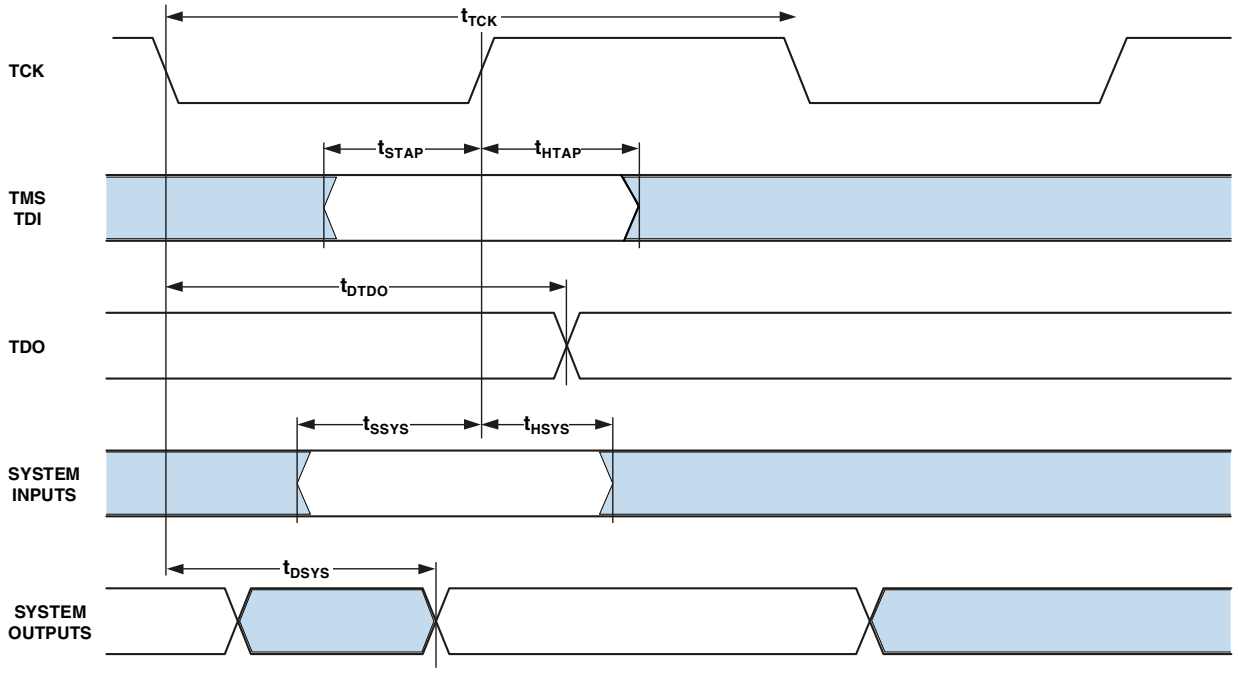


Figure 29. JTAG Port Timing

MXVR Timing

Table 37 and Table 38 describe the MXVR timing requirements.

Table 37. MXVR Timing—MXI Center Frequency Requirements

Parameter		f_S = 38 kHz	f_S = 44.1 kHz	f_S = 48 kHz	Unit
f_{MXI}	MXI Center Frequency	38.912	45.1584	49.152	MHz

Table 38. MXVR Timing— MXI Clock Requirements

Parameter		Min	Max	Unit
Timing Requirements				
FS_{MXI}	MXI Clock Frequency Stability	–50	+50	ppm
FT_{MXI}	MXI Frequency Tolerance Over Temperature	–300	+300	ppm
DC_{MXI}	MXI Clock Duty Cycle	40	60	%

OUTPUT DRIVE CURRENTS

The following figures show typical current-voltage characteristics for the output drivers of the ADSP-BF539/ADSP-BF539F processor. The curves represent the current drive capability of the output drivers as a function of output voltage.

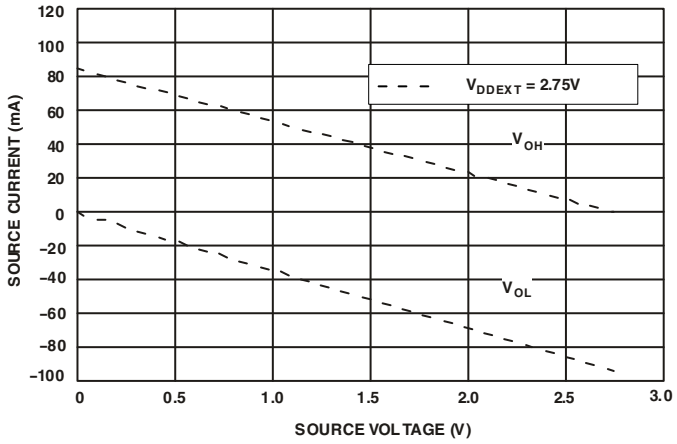


Figure 30. Drive Current A (Low V_{DDEXT})

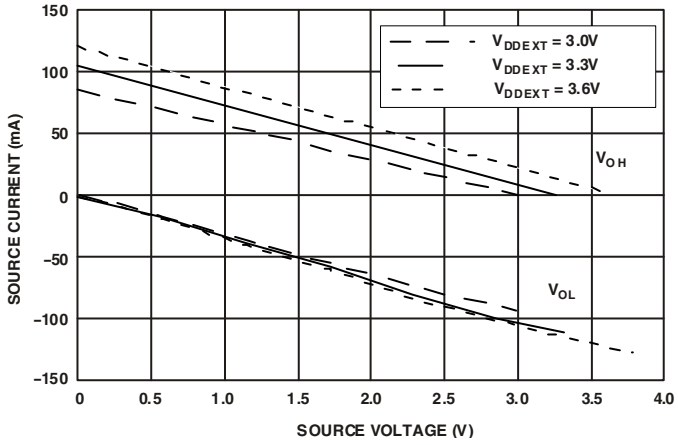


Figure 31. Drive Current A (High V_{DDEXT})

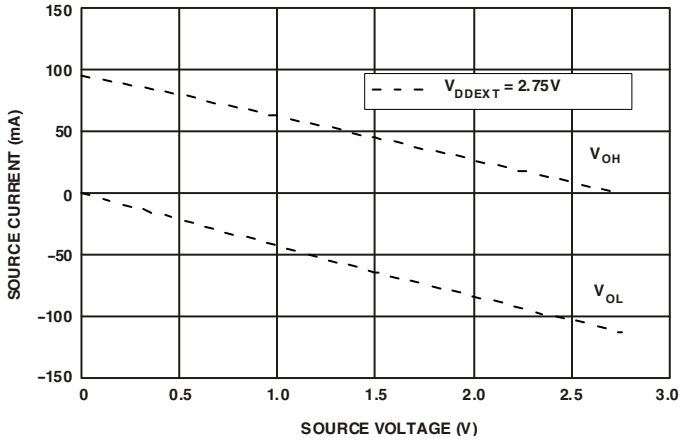


Figure 32. Drive Current B (Low V_{DDEXT})

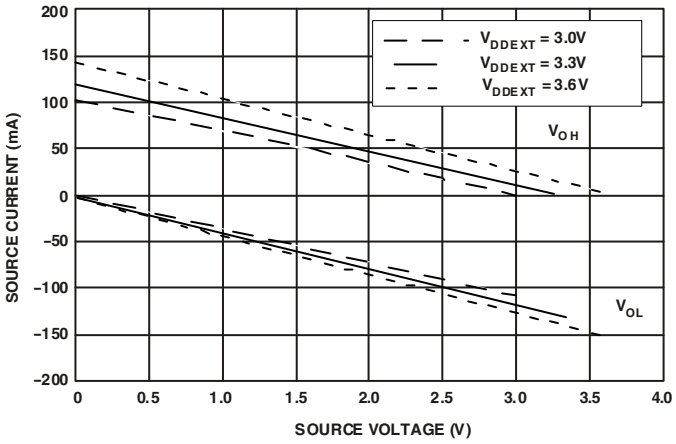


Figure 33. Drive Current B (High V_{DDEXT})

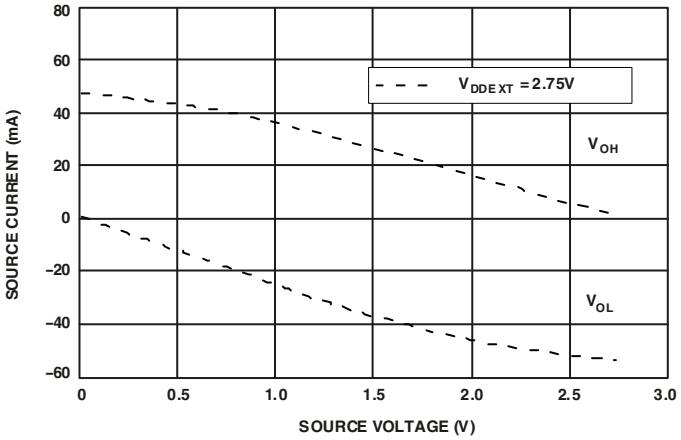


Figure 34. Drive Current C (Low V_{DDEXT})

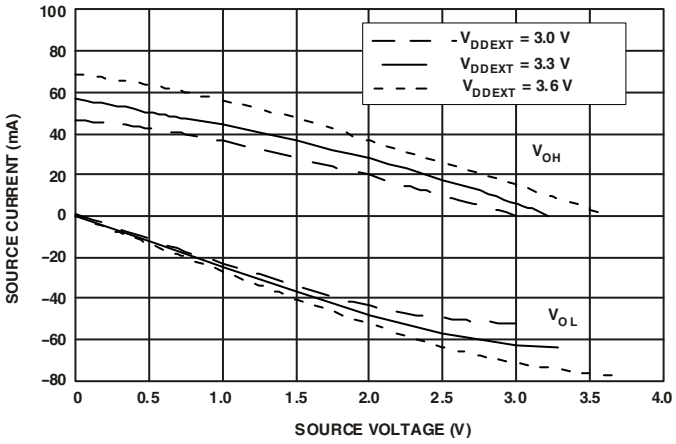


Figure 35. Drive Current C (High V_{DDEXT})

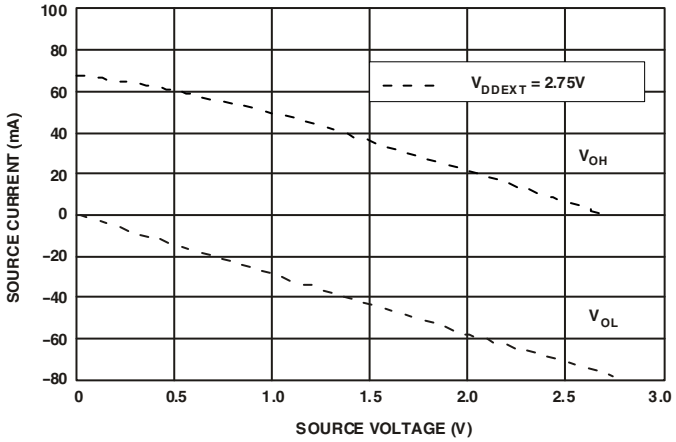


Figure 36. Drive Current D (Low V_{DDEXT})

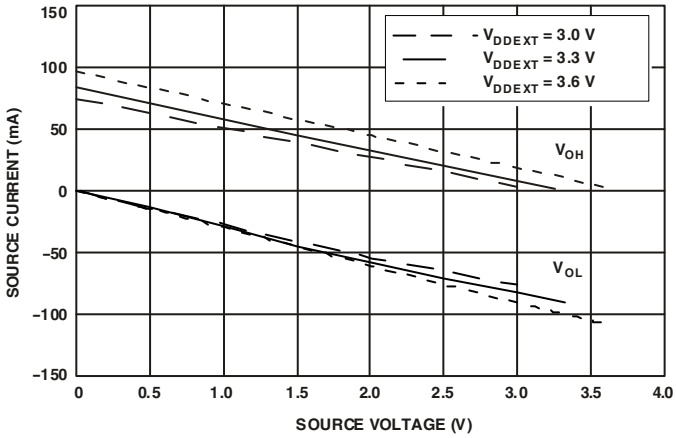


Figure 37. Drive Current D (High V_{DDEXT})

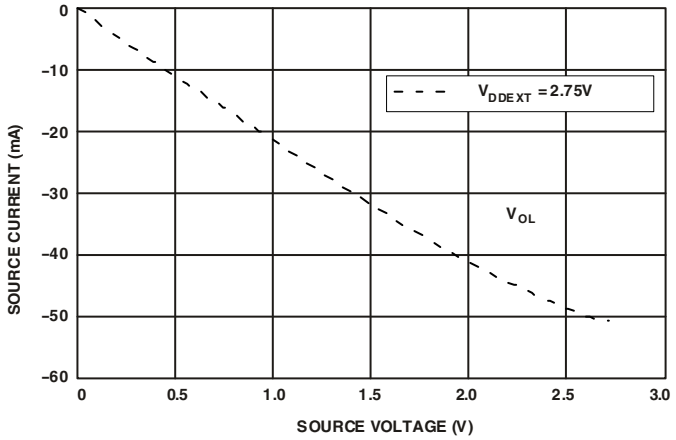


Figure 38. Drive Current E (Low V_{DDEXT})

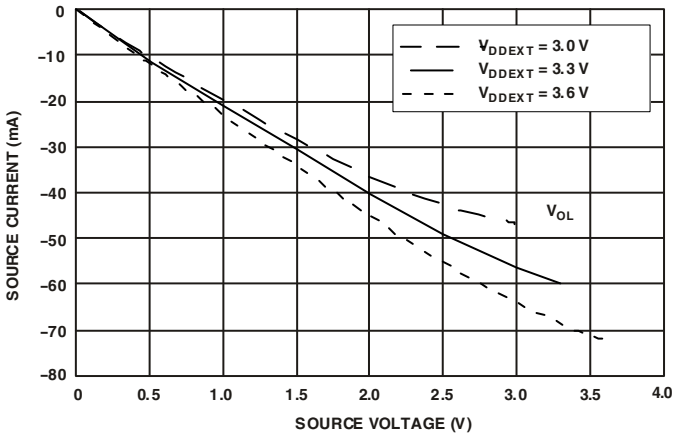


Figure 39. Drive Current E (High V_{DDEXT})

TEST CONDITIONS

All timing parameters appearing in this data sheet were measured under the conditions described in this section. Figure 40 shows the measurement point for ac measurements (except output enable/disable). The measurement point V_{MEAS} is 1.5 V for V_{DDEXT} (nominal) = 3.3 V.

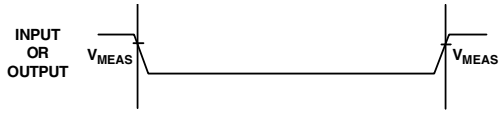


Figure 40. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time Measurement

Output pins 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 41, "Output Enable/Disable," on Page 53.

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} (nominal) = 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 pins (such as the data bus) are enabled, the measurement value is that of the first pin to start driving.

Output Disable Time Measurement

Output pins 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 41.

$$t_{DIS} = t_{DIS_MEASURED} - t_{DECAY}$$

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} (nominal) = 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 ADSP-BF539/ADSP-BF539F processor 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 is t_{DECAY} plus the various output disable times as specified in the Timing Specifications on Page 31 (for example, t_{DSDAT} for an SDRAM write cycle as shown in Table 24 on Page 36).

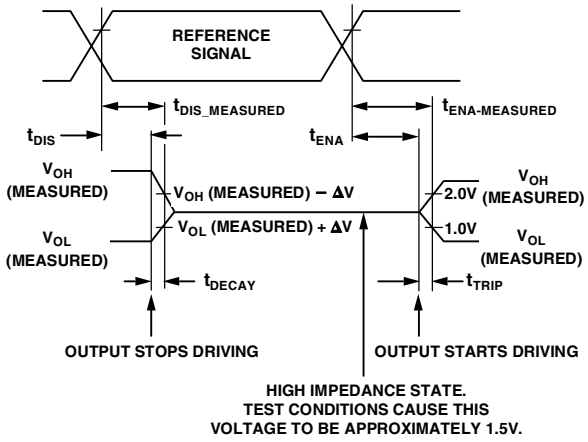


Figure 41. Output Enable/Disable

Capacitive Loading

Output delays and holds are based on standard capacitive loads: 30 pF on all pins (see Figure 42). V_{LOAD} is 1.5 V for V_{DDEXT} (nominal) = 3.3 V. Figure 43 on Page 53 through Figure 52 on Page 55 show how output rise and fall times vary 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.

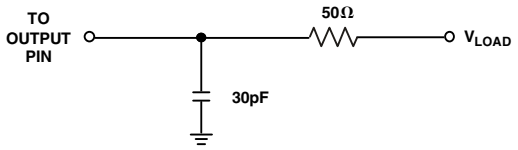


Figure 42. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

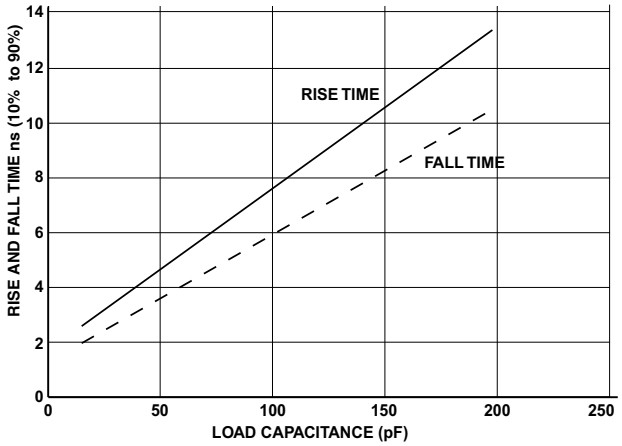


Figure 43. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver A at V_{DDEXT} = 2.7 V (Min)

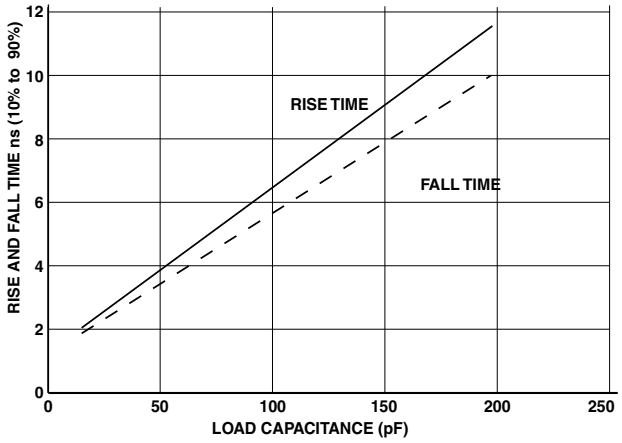


Figure 44. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver A at V_{DDEXT} = 3.65 V (Max)

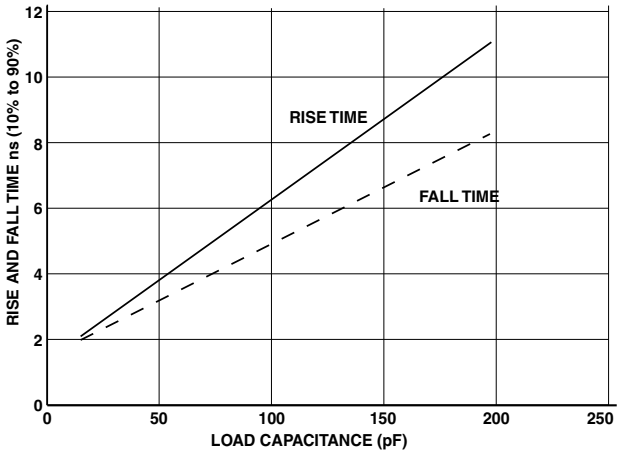


Figure 45. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver B at V_{DDEXT} = 2.7 V (Min)

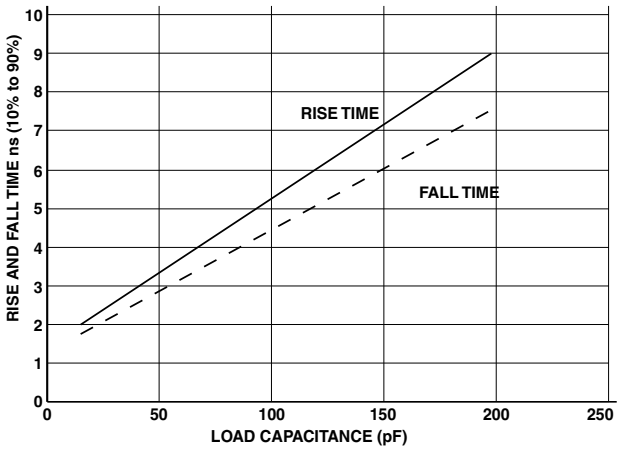


Figure 46. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver B at V_{DDEXT} = 3.65 V (Max)

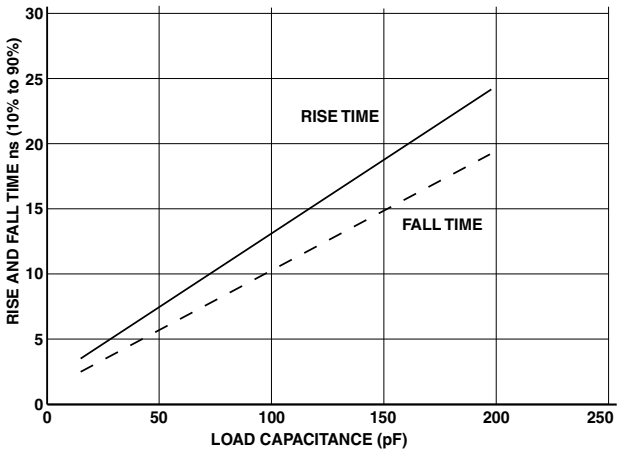


Figure 47. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver C at V_{DDEXT} = 2.7 V (Min)

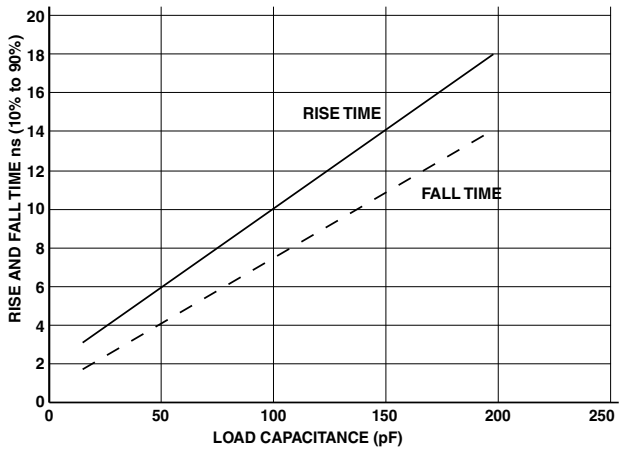


Figure 48. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver C at V_{DDEXT} = 3.65 V (Max)

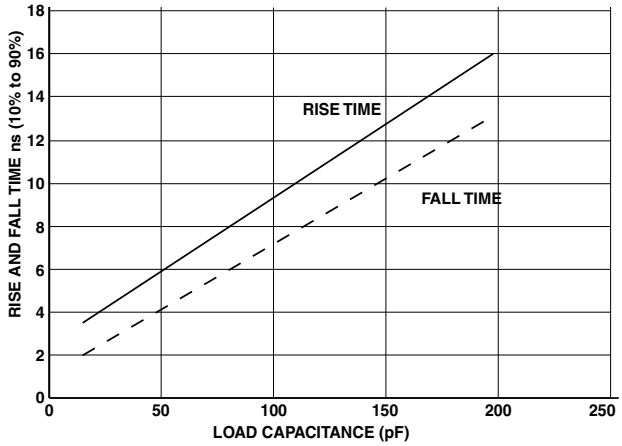


Figure 49. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver D at V_{DDEXT} = 2.7 V (Min)

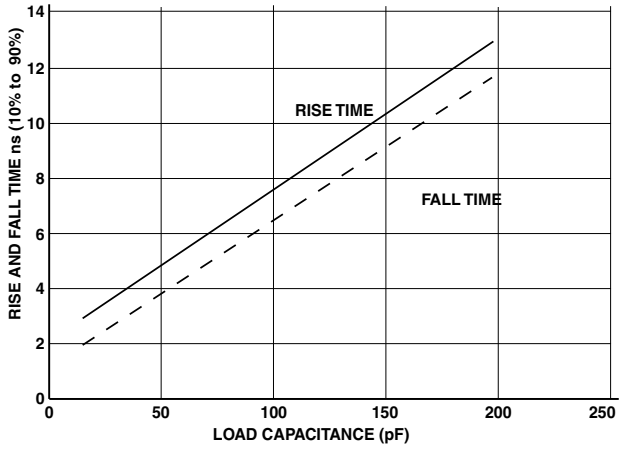


Figure 50. Typical Rise and Fall Times (10% to 90%) vs. Load Capacitance for Driver D at V_{DDEXT} = 3.65 V (Max)

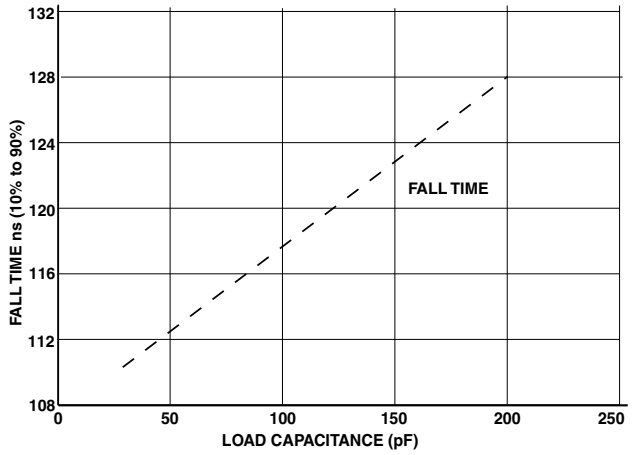


Figure 51. Typical Fall Time (10% to 90%) vs. Load Capacitance for Driver E at V_{DDEXT} = 2.7 V (Min)

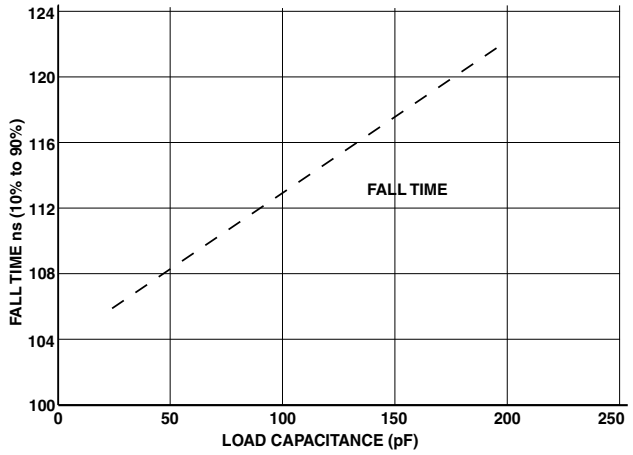


Figure 52. Typical Fall Time (10% to 90%) vs. Load Capacitance for Driver E at V_{DDEXT} = 3.65 V (Max)

THERMAL CHARACTERISTICS

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 39 or Table 40

P_D = power dissipation (see Electrical Characteristics on Page 27 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 heatsink is required.

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

In Table 39 and Table 40, 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 39. Thermal Characteristics BC-316 Without Flash

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	21.6	°C/W
θ_{JMA}	1 linear m/s air flow	18.8	°C/W
θ_{JMA}	2 linear m/s air flow	18.1	°C/W
θ_{JC}		5.36	°C/W
Ψ_{JT}	0 linear m/s air flow	0.13	°C/W
Ψ_{JT}	1 linear m/s air flow	0.25	°C/W
Ψ_{JT}	2 linear m/s air flow	0.25	°C/W

Table 40. Thermal Characteristics BC-316 With Flash

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	20.9	°C/W
θ_{JMA}	1 linear m/s air flow	18.1	°C/W
θ_{JMA}	2 linear m/s air flow	17.4	°C/W
θ_{JC}		5.01	°C/W
Ψ_{JT}	0 linear m/s air flow	0.12	°C/W
Ψ_{JT}	1 linear m/s air flow	0.24	°C/W
Ψ_{JT}	2 linear m/s air flow	0.24	°C/W

316-BALL CSP_BGA BALL ASSIGNMENT

Figure 53 lists the top view of the CSP_BGA ball assignment. Figure 54 lists the bottom view of the CSP_BGA ball assignment.

Table 41 on Page 57 lists the CSP_BGA ball assignment by ball number. Table 42 on Page 58 lists the CSP_BGA ball assignment by signal.

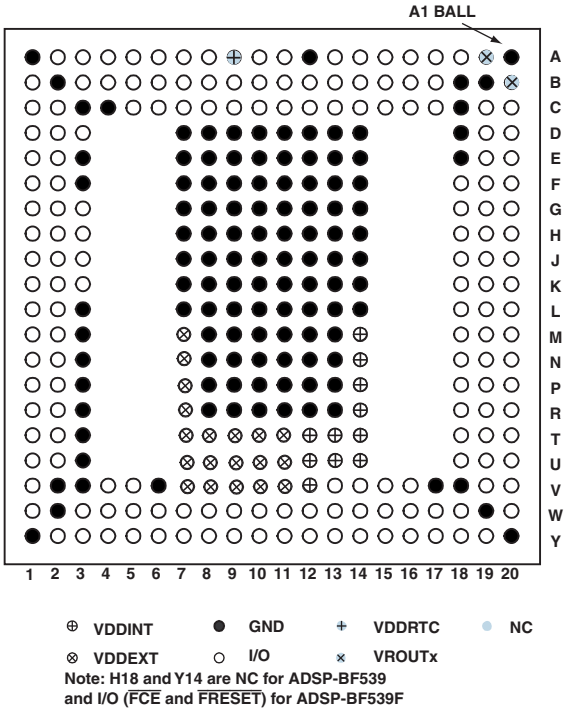


Figure 53. 316-Ball CSP_BGA Ball Assignment (Top View)

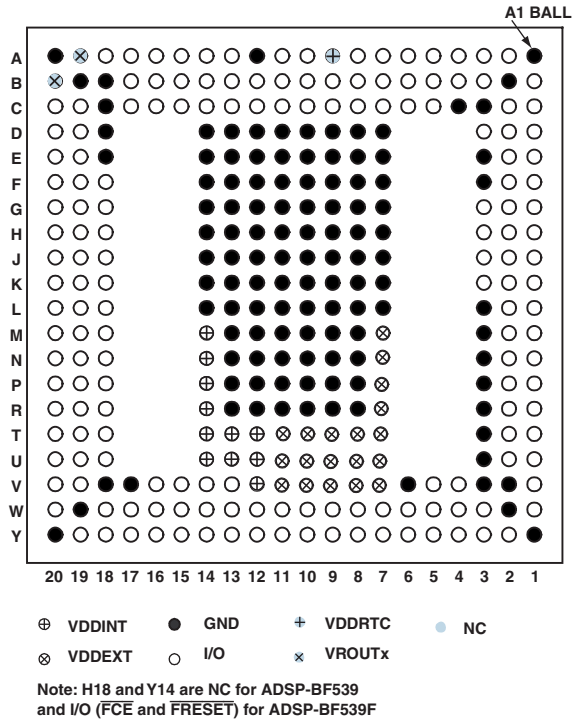


Figure 54. 316-Ball CSP_BGA Ball Assignment (Bottom View)

Table 41. 316-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	C7	$\overline{\text{SPI2SEL1}}$	F8	GND	J12	GND	M19	$\overline{\text{ABE0}}$	T3	GND	W1	TCK
A2	PF10	C8	$\overline{\text{SPI2SS}}$	F9	GND	J13	GND	M20	$\overline{\text{ABE1}}$	T7	V_{DDEXT}	W2	GND
A3	PF11	C9	MOSI2	F10	GND	J14	GND	N1	TFS0	T8	V_{DDEXT}	W3	DATA15
A4	PPI_CLK	C10	MISO2	F11	GND	J18	$\overline{\text{AMS0}}$	N2	DR0PRI	T9	V_{DDEXT}	W4	DATA13
A5	PPI0	C11	SCK2	F12	GND	J19	$\overline{\text{AMS2}}$	N3	GND	T10	V_{DDEXT}	W5	DATA11
A6	PPI2	C12	MPIVDD	F13	GND	J20	SA10	N7	V_{DDEXT}	T11	V_{DDEXT}	W6	DATA9
A7	PF15	C13	$\overline{\text{SPI1SEL1}}$	F14	GND	K1	RFS1	N8	GND	T12	V_{DDINT}	W7	DATA7
A8	PF13	C14	MISO1	F18	DT3PRI	K2	TMR2	N9	GND	T13	V_{DDINT}	W8	DATA5
A9	V_{DDRTC}	C15	$\overline{\text{SPI1SS}}$	F19	MRX	K3	GP	N10	GND	T14	V_{DDINT}	W9	DATA3
A10	RTXO	C16	MOSI1	F20	MFS	K7	GND	N11	GND	T18	RFS3	W10	DATA1
A11	RTXI	C17	SCK1	G1	SCK0	K8	GND	N12	GND	T19	ADDR7	W11	RSCLK2
A12	GND	C18	GND	G2	MOSI0	K9	GND	N13	GND	T20	ADDR8	W12	DR2PRI
A13	CLKIN	C19	MMCLK	G3	DT0SEC	K10	GND	N14	V_{DDINT}	U1	$\overline{\text{TRST}}$	W13	DT2PRI
A14	XTAL	C20	SCKE	G7	GND	K11	GND	N18	DT3SEC	U2	TMS	W14	RX2
A15	MLF	D1	PF4	G8	GND	K12	GND	N19	ADDR1	U3	GND	W15	TX2
A16	MXO	D2	PF5	G9	GND	K13	GND	N20	ADDR2	U7	V_{DDEXT}	W16	ADDR18
A17	MXI	D3	DT1SEC	G10	GND	K14	GND	P1	TSCLK0	U8	V_{DDEXT}	W17	ADDR15
A18	$\overline{\text{MRXON}}$	D7	GND	G11	GND	K18	$\overline{\text{AMS3}}$	P2	RFS0	U9	V_{DDEXT}	W18	ADDR13
A19	VROUT1	D8	GND	G12	GND	K19	$\overline{\text{AMS1}}$	P3	GND	U10	V_{DDEXT}	W19	GND
A20	GND	D9	GND	G13	GND	K20	$\overline{\text{AOE}}$	P7	V_{DDEXT}	U11	V_{DDEXT}	W20	ADDR14
B1	PF8	D10	GND	G14	GND	L1	RSCLK1	P8	GND	U12	V_{DDINT}	Y1	GND
B2	GND	D11	GND	G18	$\overline{\text{BR}}$	L2	TMR1	P9	GND	U13	V_{DDINT}	Y2	TDO
B3	PF9	D12	GND	G19	CLKOUT	L3	GND	P10	GND	U14	V_{DDINT}	Y3	DATA14
B4	PF3	D13	GND	G20	$\overline{\text{SRAS}}$	L7	GND	P11	GND	U18	RSCLK3	Y4	DATA12
B5	PPI1	D14	GND	H1	DT1PRI	L8	GND	P12	GND	U19	ADDR9	Y5	DATA10
B6	PPI3	D18	GND	H2	TSCLK1	L9	GND	P13	GND	U20	ADDR10	Y6	DATA8
B7	PF14	D19	MBCLK	H3	DR1SEC	L10	GND	P14	V_{DDINT}	V1	TDI	Y7	DATA6
B8	PF12	D20	$\overline{\text{SMS}}$	H7	GND	L11	GND	P18	DR3SEC	V2	GND	Y8	DATA4
B9	SCL0	E1	PF1	H8	GND	L12	GND	P19	ADDR3	V3	GND	Y9	DATA2
B10	SDA0	E2	PF2	H9	GND	L13	GND	P20	ADDR4	V4	BMODE1	Y10	DATA0
B11	CANRX	E3	GND	H10	GND	L14	GND	R1	TX0	V5	BMODE0	Y11	RFS2
B12	CANTX	E7	GND	H11	GND	L18	TSCLK3	R2	RSCLK0	V6	GND	Y12	TSCLK2
B13	$\overline{\text{NMI}}$	E8	GND	H12	GND	L19	$\overline{\text{ARE}}$	R3	GND	V7	V_{DDEXT}	Y13	TFS2
B14	$\overline{\text{RESET}}$	E9	GND	H13	GND	L20	$\overline{\text{AWE}}$	R7	V_{DDEXT}	V8	V_{DDEXT}	Y14	$\overline{\text{FRESET}}$
B15	MXEVDD	E10	GND	H14	GND	M1	DT0PRI	R8	GND	V9	V_{DDEXT}	Y15	SCL1
B16	MXEGND	E11	GND	H18	$\overline{\text{FCE}}$	M2	TMR0	R9	GND	V10	V_{DDEXT}	Y16	SDA1
B17	$\overline{\text{MTXON}}$	E12	GND	H19	$\overline{\text{SCAS}}$	M3	GND	R10	GND	V11	V_{DDEXT}	Y17	ADDR19
B18	GND	E13	GND	H20	$\overline{\text{SWE}}$	M7	V_{DDEXT}	R11	GND	V12	V_{DDINT}	Y18	ADDR17
B19	GND	E14	GND	J1	TFS1	M8	GND	R12	GND	V13	DR2SEC	Y19	ADDR16
B20	VROUT0	E18	GND	J2	DR1PRI	M9	GND	R13	GND	V14	$\overline{\text{BG}}$	Y20	GND
C1	PF6	E19	MTX	J3	DR0SEC	M10	GND	R14	V_{DDINT}	V15	$\overline{\text{BGH}}$		
C2	PF7	E20	ARDY	J7	GND	M11	GND	R18	DR3PRI	V16	DT2SEC		
C3	GND	F1	PF0	J8	GND	M12	GND	R19	ADDR5	V17	GND		
C4	GND	F2	MISO0	J9	GND	M13	GND	R20	ADDR6	V18	GND		
C5	RX1	F3	GND	J10	GND	M14	V_{DDINT}	T1	RX0	V19	ADDR11		
C6	TX1	F7	GND	J11	GND	M18	TFS3	T2	$\overline{\text{EMU}}$	V20	ADDR12		

Table 42. 316-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.
$\overline{ABE0}$	M19	DATA8	Y6	GND	E7	GND	K11	GND	V17	PPI2	A6	TSCLK1	H2
$\overline{ABE1}$	M20	DATA9	W6	GND	E8	GND	K12	GND	V18	PPI3	B6	TSCLK2	Y12
ADDR1	N19	DATA10	Y5	GND	E9	GND	K13	GND	W2	$\overline{RESET}$	B14	TSCLK3	L18
ADDR2	N20	DATA11	W5	GND	F8	GND	L13	GND	W19	RFS0	P2	TX0	R1
ADDR3	P19	DATA12	Y4	GND	F9	GND	L14	GND	Y1	RFS1	K1	TX1	C6
ADDR4	P20	DATA13	W4	GND	F10	GND	M3	GND	Y20	RFS2	Y11	TX2	W15
ADDR5	R19	DATA14	Y3	GND	F11	GND	M8	GP	K3	RFS3	T18	V_{DDEXT}	T8
ADDR6	R20	DATA15	W3	GND	F12	GND	M9	MBCLK	D19	RSCLK0	R2	V_{DDEXT}	T9
ADDR7	T19	DR0PRI	N2	GND	F13	GND	M10	MFS	F20	RSCLK1	L1	V_{DDEXT}	T10
ADDR8	T20	DR0SEC	J3	GND	F14	GND	M11	MISO0	F2	RSCLK2	W11	V_{DDEXT}	T11
ADDR9	U19	DR1PRI	J2	GND	G7	GND	M12	MISO1	C14	RSCLK3	U18	V_{DDEXT}	U7
ADDR10	U20	DR1SEC	H3	GND	G8	GND	M13	MISO2	C10	RTXI	A11	V_{DDEXT}	U8
ADDR11	V19	DR2PRI	W12	GND	G9	GND	N3	MLF	A15	RTXO	A10	V_{DDEXT}	U9
ADDR12	V20	DR2SEC	V13	GND	E10	GND	K14	MMCLK	C19	RX0	T1	V_{DDEXT}	U10
ADDR13	W18	DR3PRI	R18	GND	E11	GND	L3	MOSI0	G2	RX1	C5	V_{DDEXT}	U11
ADDR14	W20	DR3SEC	P18	GND	E12	GND	L7	MOSI1	C16	RX2	W14	V_{DDEXT}	V7
ADDR15	W17	DT0PRI	M1	GND	E13	GND	L8	MOSI2	C9	SA10	J20	V_{DDEXT}	M7
ADDR16	Y19	DT0SEC	G3	GND	E14	GND	L9	MPIVDD	C12	$\overline{SCAS}$	H19	V_{DDEXT}	N7
ADDR17	Y18	DT1PRI	H1	GND	E18	GND	L10	$\overline{MRXON}$	A18	SCK0	G1	V_{DDEXT}	P7
ADDR18	W16	DT1SEC	D3	GND	F3	GND	L11	MRX	F19	SCK1	C17	V_{DDEXT}	R7
ADDR19	Y17	DT2PRI	W13	GND	F7	GND	L12	MTX	E19	SCK2	C11	V_{DDEXT}	T7
$\overline{AMS0}$	J18	DT2SEC	V16	GND	G10	GND	N8	$\overline{MTXON}$	B17	SCKE	C20	V_{DDEXT}	V8
$\overline{AMS1}$	K19	DT3PRI	F18	GND	G11	GND	N9	MXEGND	B16	SCL0	B9	V_{DDEXT}	V9
$\overline{AMS2}$	J19	DT3SEC	N18	GND	G12	GND	N10	MXEVDD	B15	SCL1	Y15	V_{DDEXT}	V10
$\overline{AMS3}$	K18	$\overline{EMU}$	T2	GND	G13	GND	N11	MXI	A17	SDA0	B10	V_{DDEXT}	V11
$\overline{AOE}$	K20	$\overline{FCE}$	H18	GND	G14	GND	N12	MXO	A16	SDA1	Y16	V_{DDINT}	M14
ARDY	E20	$\overline{FRESET}$	Y14	GND	H7	GND	N13	$\overline{NMI}$	B13	$\overline{SMS}$	D20	V_{DDINT}	N14
$\overline{ARE}$	L19	GND	A1	GND	H8	GND	P3	PF0	F1	$\overline{SPI1SEL1}$	C13	V_{DDINT}	P14
$\overline{AWE}$	L20	GND	A12	GND	H9	GND	P8	PF1	E1	$\overline{SPI1SS}$	C15	V_{DDINT}	R14
$\overline{BG}$	V14	GND	A20	GND	H10	GND	P9	PF2	E2	$\overline{SPI2SEL1}$	C7	V_{DDINT}	T12
$\overline{BGH}$	V15	GND	B2	GND	H11	GND	P10	PF3	B4	$\overline{SPI2SS}$	C8	V_{DDINT}	T13
BMODE0	V5	GND	B18	GND	H12	GND	P11	PF4	D1	$\overline{SRAS}$	G20	V_{DDINT}	T14
BMODE1	V4	GND	B19	GND	H13	GND	P12	PF5	D2	$\overline{SWE}$	H20	V_{DDINT}	U12
$\overline{BR}$	G18	GND	C3	GND	H14	GND	P13	PF6	C1	TCK	W1	V_{DDINT}	U13
CANRX	B11	GND	C4	GND	J7	GND	R3	PF7	C2	TDI	V1	V_{DDINT}	U14
CANTX	B12	GND	C18	GND	J8	GND	R8	PF8	B1	TDO	Y2	V_{DDINT}	V12
CLKIN	A13	GND	D7	GND	J9	GND	R9	PF9	B3	TFS0	N1	V_{DDRTC}	A9
CLKOUT	G19	GND	D8	GND	J10	GND	R10	PF10	A2	TFS1	J1	VROUT0	B20
DATA0	Y10	GND	D9	GND	J11	GND	R11	PF11	A3	TFS2	Y13	VROUT1	A19
DATA1	W10	GND	D10	GND	J12	GND	R12	PF12	B8	TFS3	M18	XTAL	A14
DATA2	Y9	GND	D11	GND	J13	GND	R13	PF13	A8	TMR0	M2		
DATA3	W9	GND	D12	GND	J14	GND	T3	PF14	B7	TMR1	L2		
DATA4	Y8	GND	D13	GND	K7	GND	U3	PF15	A7	TMR2	K2		
DATA5	W8	GND	D14	GND	K8	GND	V2	PPI_CLK	A4	TMS	U2		
DATA6	Y7	GND	D18	GND	K9	GND	V3	PPI0	A5	$\overline{TRST}$	U1		
DATA7	W7	GND	E3	GND	K10	GND	V6	PPI1	B5	TSCLK0	P1		

OUTLINE DIMENSIONS

Dimensions in the outline dimensions figures are shown in millimeters.

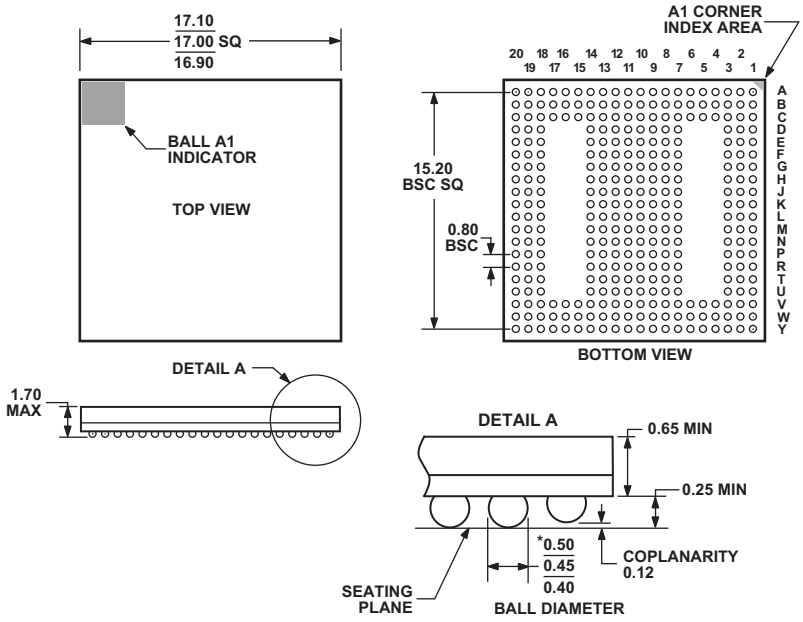


Figure 55. 316-Ball Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-316)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 43 is provided as an aid to PCB design. For industry-standard design recommendations, refer to IPC-7351, *Generic Requirements for Surface Mount Design and Land Pattern Standard*.

Table 43. BGA Data for Use with Surface-Mount Design

Package	Ball Attach Type	Solder Mask Opening	Ball Pad Size
316-Ball CSP_BGA (BC-316)	Solder Mask Defined	0.40 mm diameter	0.50 mm diameter

AUTOMOTIVE PRODUCTS

Some ADSP-BF539/ADSP-BF539F models are available for automotive applications with controlled manufacturing. Note that these special models may have specifications that differ from the general release models.

The automotive grade products shown in Table 44 are available for use in automotive applications. Contact your local ADI account representative or authorized ADI product distributor for specific product ordering information. Note that all automotive products are RoHS compliant.

Table 44. Automotive Products

Product Family [1]	Temperature Range[2]	Instruction Rate (Max)	Flash Memory	Operating Voltage (Nominal)	Package Description	Package Option
ADBF539WBBCZ5xx	–40°C to +85°C	533 MHz	N/A	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316
ADBF539WBBCZ5F4xx	–40°C to +85°C	533 MHz	512K byte	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316
ADBF539WBBCZ5F8xx	–40°C to +85°C	533 MHz	1M byte	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316

[1] x denotes silicon revision.
[2] Referenced temperature is ambient temperature.

ORDERING GUIDE

Model	Temperature Range[1]	Instruction Rate (Max)	Flash Memory	Operating Voltage (Nominal)	Package Description	Package Option
ADSP-BF539BBCZ-5A[2]	–40°C to +85°C	533 MHz	N/A	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316
ADSP-BF539BBCZ-5F4	–40°C to +85°C	533 MHz	512K byte	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316
ADSP-BF539BBCZ-5F8	–40°C to +85°C	533 MHz	1M byte	1.25 V internal/ 3.3 V I/O	316-Ball CSP_BGA	BC-316

[1] Referenced temperature is ambient temperature.
[2] Z = RoHS compliant part.

©2008 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.
D06699-0-11/08(B)