

FEATURES

- 1.0 V to 1.2 V core V_{DD} with on-chip voltage regulation
- 3.3 V tolerant I/O with specific 5 V tolerant pins
- 316-ball Pb-free mini-BGA package
- Up to 500 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

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 x 16-bits or 256K x 16-bits of flash memory (ADSP-BF539F only)
- Four dual-channel memory DMA controllers
- Memory management unit providing memory protection

- External memory controller with glueless support for SDRAM, SRAM, flash, and ROM
- Flexible memory booting options from SPI®, external memory

PERIPHERALS

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

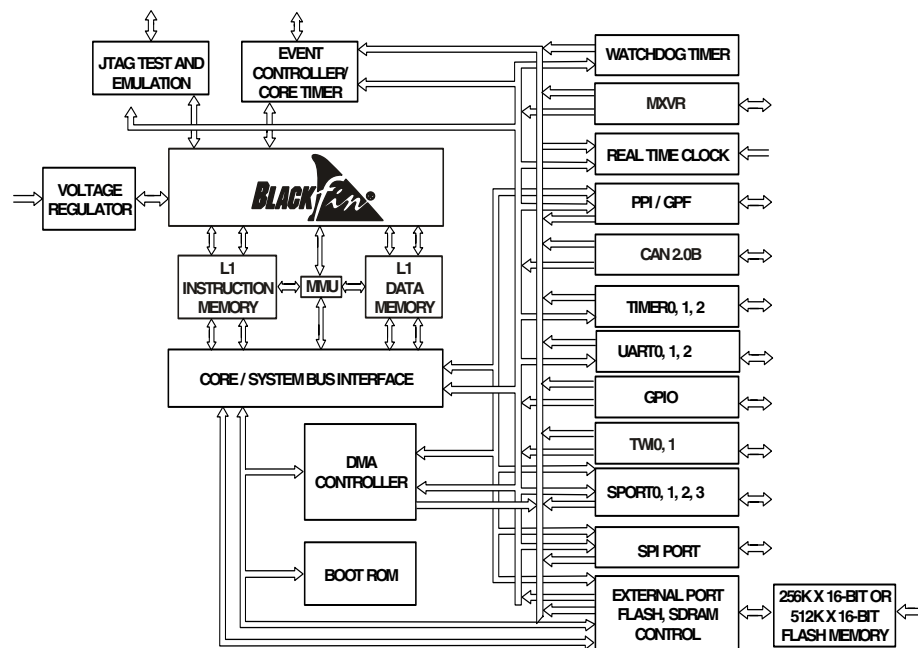


Figure 1. Functional Block Diagram

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

Rev. PrE

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.

TABLE OF CONTENTS

General Description	4	Electrical Characteristics	27
Low Power Architecture	4	Absolute Maximum Ratings	28
System Integration	4	Package Information	28
ADSP-BF539/ADSP-BF539F Processor Peripherals	4	ESD Sensitivity	28
Blackfin Processor Core	5	Timing Specifications	29
Memory Architecture	5	Clock and Reset Timing	30
DMA Controllers	10	Asynchronous Memory Read Cycle Timing	31
Real Time Clock	10	Asynchronous Memory Write Cycle Timing	35
Watchdog Timer	11	SDRAM Interface Timing	37
Timers	11	External Port Bus Request and Grant Cycle Timing ..	38
Serial Ports (SPORTs)	11	Parallel Peripheral Interface Timing	41
Serial Peripheral Interface (SPI) Ports	11	Serial Ports Timing	44
Two Wire Interface	12	Serial Peripheral Interface (SPI) Port	
UART Port	12	—Master Timing	48
Programmable I/O Pins	12	Serial Peripheral Interface (SPI) Port	
Parallel Peripheral Interface	13	—Slave Timing	49
Controller Area Network (CAN) Interface	14	Universal Asynchronous Receiver-Transmitter	
Media Transceiver MAC layer (MXVR)	14	(UART) Port Timing	50
Dynamic Power Management	15	Programmable Flags Cycle Timing	51
Voltage Regulation	16	Timer Cycle Timing	52
Clock Signals	16	JTAG Test And Emulation Port Timing	53
Bootling Modes	17	TWI Controller Timing	54
Instruction Set Description	18	MXVR Timing	58
Development Tools	18	CAN Timing	59
Designing an Emulator Compatible Processor Board ...	19	Output Drive Currents	60
Example Connections and Layout Considerations	19	Power Dissipation	62
Voltage Regulator Layout Guidelines	19	Test Conditions	62
MXVR Board Layout Guidelines	19	Environmental Conditions	65
Pin Descriptions	22	316-Ball Mini-BGA Pinout	66
Specifications	27	Outline Dimensions	69
Recommended Operating Conditions	27	Ordering Guide	70
Recommended Operating Conditions			
—Applies to 5V Tolerant pins	27		

REVISION HISTORY

5/06—Revision PrE:

Added Maximum Duty Cycle for Input Transient Voltage	28
Added Package Information	28
Added values in Power Dissipation	61
Added graphs in Output Drive Currents	59
Added graphs in Capacitive Loading	62

Changes Ordering Guide	67
------------------------------	----

12/05—Revision PrD:

Changes number of GPIO and GPF pins Peripherals	1
Changes wording in System Integration	4
Changes voltage regulator range in ADSP-BF539 processor Peripherals	4

Changes voltage regulator range in Voltage Regulation 15

Changes to Figure 8 and MXVR Board Layout Guidelines 19

Modified Example connections of ADSP-BF539 to MOST
Network 19

Changes to CAN, MXVR., and UART pullup in Pin
Descriptions 20

Adds footnotes to Recommended Operating Conditions .. 25

Adds footnotes to Absolute Maximum Ratings 26

Adds die temperature specification Absolute Maximum Ratings
26

Adds table and figure to Asynchronous Memory Read Cycle
Timing 29

Adds table and figure to Asynchronous Memory Write Cycle
Timing 31

Deletes footnotes in External Port Bus Request and Grant Cycle
Timing 34

Adds t_{WBR} to External Port Bus Request and Grant Cycle Timing
34

Replaced Figure 17, Figure 18, Figure 19, and Figure 20 ... 38

Added footnotes in Serial Ports—External Clock and Serial
Ports—Internal Clock 39

Changes to lower diagram in Serial Ports 40

Changes to External Late Frame Sync (Frame Sync Setup <
 $t_{SCLKE}/2$) 41

Replace section Power Dissipation 57

Added overbar to AOE in 316-Ball Sparse Mini-BGA Pin
Assignment (Alphabetically by Signal) 63

Adds section Surface Mount Design 65

GENERAL DESCRIPTION

The ADSP-BF539/ADSP-BF539F processors are a members of the Blackfin family of products, incorporating the Analog Devices/Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like micro-processor 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. Specific performance, peripherals, and memory configurations are shown in [Table 1 on Page 4](#).

Table 1. Processor Features

	ADSP-BF539	ADSP-BF539F4	ADSP-BF539F8
Maximum Performance	500 MHz 1000 MMACs	500 MHz 1000 MMACs	500 MHz 1000 MMACs
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 x 16-bit	512K x 16-bit
SPORTs	4	4	4
SPI	3	3	3
TWI	2	2	2
UARTs	3	3	3
PPI	1	1	1
CAN	1	1	1
MXVR	1	1	1
Package option	316	316	316

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.

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 just varying the frequency of operation. This translates into longer battery life and lower heat dissipation.

SYSTEM INTEGRATION

The ADSP-BF539/ADSP-BF539F processor is a highly integrated system-on-a-chip solution 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 Two-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 processor contains a rich set of peripherals connected to the core via several high bandwidth buses, providing flexibility in system configuration as well as excellent overall system performance (see the block diagram [on Page 1](#)). 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 ADSP-BF539/ADSP-BF539F processor contains 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, except for general purpose I/O, CAN, TWI, Real Time Clock, and timers, are supported by a flexible DMA structure. There are also four separate memory DMA controllers 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 processor includes an on-chip voltage regulator in support of the ADSP-BF539/ADSP-BF539F processor Dynamic Power Management capability. The voltage

regulator provides a range of core voltage levels from a single 2.7 V to 3.6 V input. The voltage regulator can be bypassed at the user's discretion.

BLACKFIN PROCESSOR CORE

As shown in [Figure 2 on Page 6](#), 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.

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 may be operating on the core and can protect system registers from unintended access.

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

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

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

MEMORY ARCHITECTURE

The ADSP-BF539/ADSP-BF539F processor views memory as a single unified 4G byte address space, using 32-bit addresses. All resources, including internal memory, external memory, and I/O control registers, occupy separate sections of this common address space. The memory portions of this address space are arranged in a hierarchical structure to provide a good cost/performance balance of some very fast, low latency on-chip memory as cache or SRAM, and larger, lower cost and performance off-chip memory systems. See [Figure 3 on Page 7](#).

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 can perform 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.

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.

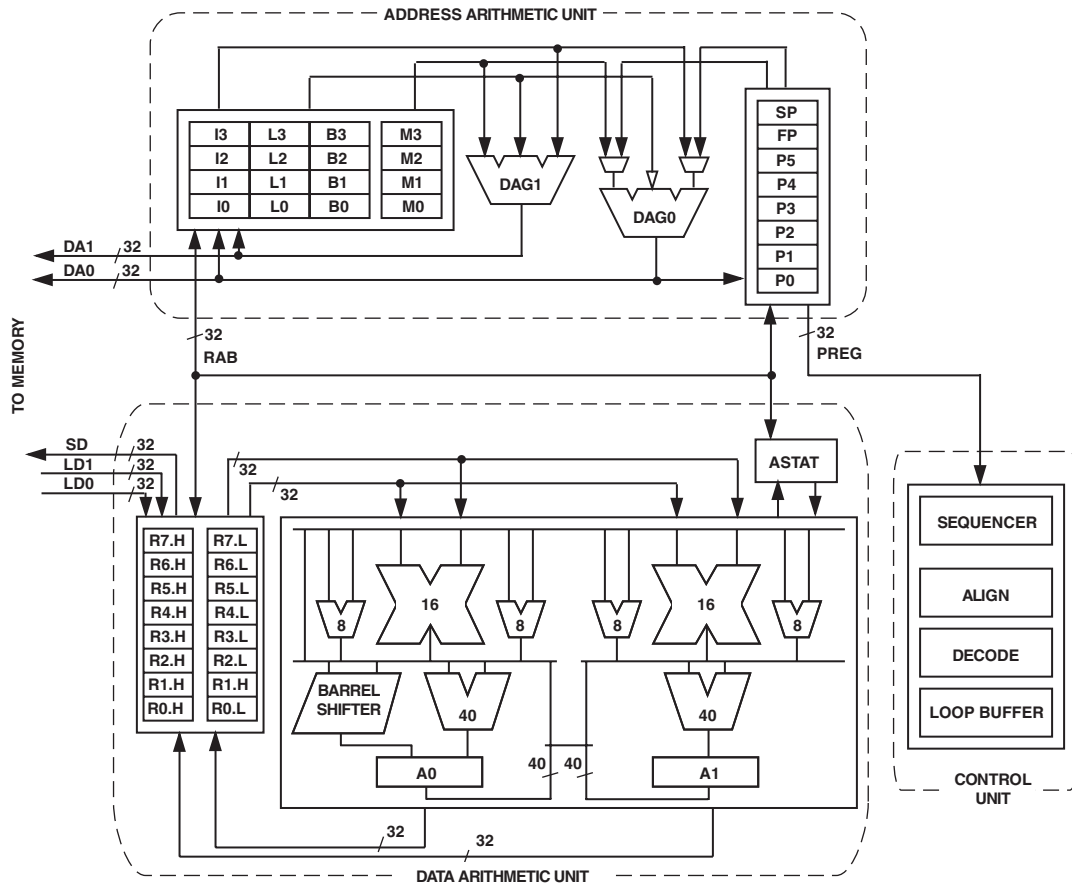


Figure 2. Blackfin Processor Core

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 scratchpad SRAM which runs at the same speed as the L1 memories, but is only accessible as data SRAM and cannot be configured as cache memory.

External (Off-Chip) Memory

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

The 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 on Page 7 shows how the flash memory die and Blackfin processor die are connected.

The ADSP-BF539F4 contains a 4 Mbit (256K x 16-bits) bottom boot sector flash memory. The ADSP-BF539F8 contains an 8 Mbit (512K x 16-bits) bottom boot sector 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

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.

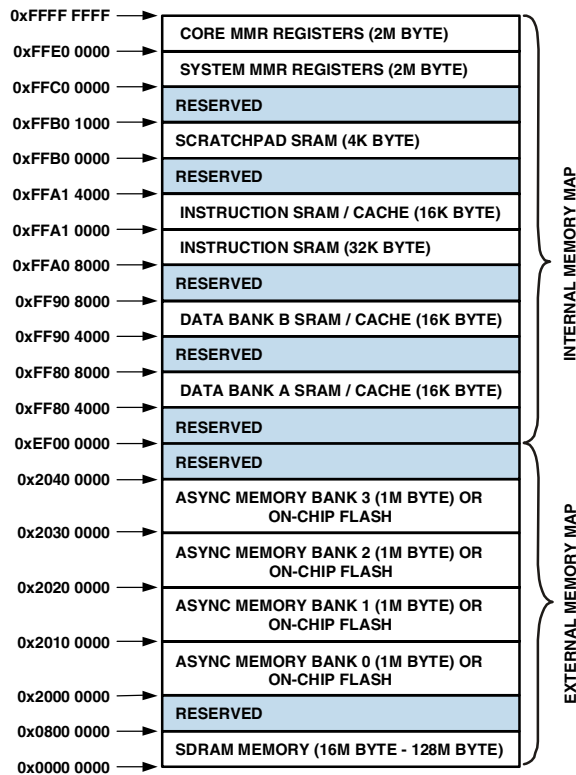


Figure 3. ADSP-BF539/ADSP-BF539F Internal/External Memory Map

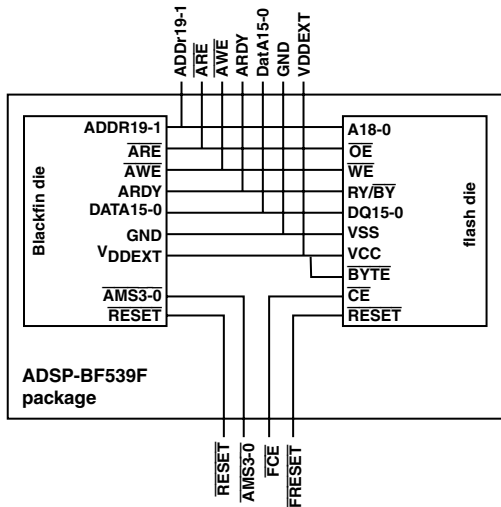


Figure 4. Internal Connection of Flash Memory

The flash chip enable pin \overline{FCE} must be connected to $\overline{AMS0}$ or $\overline{AMS3-1}$ through a printed circuit board trace. When connected to $\overline{AMS0}$ the Blackfin processor can boot from the flash die. When connected to $\overline{AMS3-1}$ the flash memory will appear as non-volatile memory in the processor memory map shown in [Figure 3 on Page 7](#).

Flash Memory Programming

The ADSP-BF539F4 and ADSP-BF539F8 flash memory may 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{BR}) asserted and a CLKIN provided.

The VDSP++ tools may 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{FRESET} pin. Refer to the flash datasheet 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 which contains the control MMRs for all core functions, and the other which contains the registers needed for setup and control of the on-chip peripherals outside of the core. The MMRs are accessible only in supervisor mode and appear as reserved space to on-chip peripherals.

Booting

The 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 17](#).

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.

- Non-Maskable Interrupt (NMI) – The NMI event can be generated by the software watchdog timer or by the NMI input signal to the processor. The NMI event is frequently used as a power-down indicator to initiate an orderly shut-down 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 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 pro-

cessor. [Table 2](#) describes the inputs to the CEC, identifies their names in the Event Vector Table (EVT), and lists their priorities.

Table 2. Core Event Controller (CEC)

Priority (0 is Highest)	Event Class	EVT Entry
0	Emulation/Test Control	EMU
1	Reset	RST
2	Non-Maskable 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

System Interrupt Controller (SIC)

The System Interrupt Controller provides the mapping and routing of events from the many peripheral interrupt sources to the prioritized general purpose interrupt inputs of the CEC. Although the ADSP-BF539/ADSP-BF539F processor provides a default mapping, the user can alter the mappings and priorities of interrupt events by writing the appropriate values into the Interrupt Assignment Registers (IAR). [Table 3](#) describes the inputs into the SIC and the default mappings into the CEC.

Table 3. System and Core Event Mapping

Event Source	Core Event Name
PLL Wakeup 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

Table 3. System and Core Event Mapping (Continued)

Event Source	Core Event Name
UART0 Error Interrupt	IVG7
UART1 Error Interrupt	IVG7
UART2 Error Interrupt	IVG7
CAN Error Interrupt	IVG7
Real Time Clock Interrupts	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
DMA12 Interrupt (SPORT2 RX)	IVG9
DMA13 Interrupt (SPORT2 TX)	IVG9
DMA14 Interrupt (SPORT3 RX)	IVG9
DMA15 Interrupt (SPORT3 TX)	IVG9
DMA5 Interrupt (SPI0)	IVG10
DMA18 Interrupt (SPI1)	IVG10
DMA19 Interrupt (SPI2)	IVG10
DMA6 Interrupt (UART0 RX)	IVG10
DMA7 Interrupt (UART0 TX)	IVG10
DMA20 Interrupt (UART1 RX)	IVG10
DMA21 Interrupt (UART1 TX)	IVG10
DMA22 Interrupt (UART2 RX)	IVG10
DMA23 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 Interrupt	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

Event Control

The ADSP-BF539/ADSP-BF539F processor provides 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 cleared when the event has been accepted into the system. This register is updated automatically by the controller, but it may 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 may be latched in the ILAT register. This register may be read or written while in supervisor mode. (Note that general purpose interrupts can be globally enabled and disabled with the STI and CLI instructions, respectively.)
- CEC Interrupt Pending Register (IPEND) – The IPEND register keeps track of all nested events. A set bit in the IPEND register indicates the event is currently active or nested at some level. This register is updated automatically by the controller but may be read while in supervisor mode.

The SIC allows further control of event processing by providing three 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 the peripheral is asserting the interrupt, and a cleared bit indicates the peripheral is not asserting the event.
- SIC Interrupt Wakeup Enable Registers (SIC_IWRx) – By enabling the corresponding bit in these registers, a peripheral can be configured to wake up the processor, should the core be idled when the event is generated. (For more information, see Dynamic Power Management on Page 15.)

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.

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 port, UART, and PPI. Each individual DMA capable peripheral has at least one dedicated DMA channel. The MXVR peripheral has its own dedicated DMA controller, which supports its own unique set of operating modes.

The ADSP-BF539/ADSP-BF539F processor DMA controllers support both 1-dimensional (1D) and 2-dimensional (2D) DMA transfers. DMA transfer initialization can be implemented from registers or from sets of parameters called descriptor blocks.

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

Examples of DMA types supported by the ADSP-BF539/ADSP-BF539F processor DMA controller include:

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

In addition to the dedicated peripheral DMA channels, there are 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 processor. The RTC peripheral has dedicated power supply pins so that it can remain powered up and clocked even when the rest of the processor is in a low power state. The RTC provides several programmable interrupt options, including interrupt per second, minute, hour, or day clock ticks, interrupt on programmable stopwatch countdown, or interrupt at a programmed alarm time.

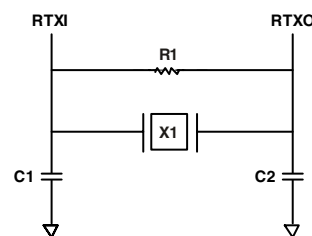
The 32.768 KHz input clock frequency is divided down to a 1 Hz signal by a prescaler. The counter function of the timer consists of four counters: a 60 second counter, a 60 minute counter, a 24 hour counter, and an 32,768 day counter.

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

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

Like the other peripherals, the RTC can wake up the ADSP-BF539/ADSP-BF539F processor from Sleep mode upon generation of any RTC wakeup event. Additionally, an RTC wakeup event can wake up the ADSP-BF539/ADSP-BF539F 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 = 22 pF
 C2 = 22 pF
 R1 = 10 MΩ

NOTE: C1 AND C2 ARE SPECIFIC TO CRYSTAL SPECIFIED FOR X1. CONTACT CRYSTAL MANUFACTURER FOR DETAILS. C1 AND C2 SPECIFICATIONS ASSUME BOARD TRACE CAPACITANCE OF 3 pF.

Figure 5. External Components for RTC

WATCHDOG TIMER

The ADSP-BF539/ADSP-BF539F processor includes a 32-bit timer that can be used to implement a software watchdog function. A software watchdog can improve system availability by forcing the processor to a known state through generation of a hardware reset, non-maskable interrupt (NMI), or general purpose interrupt, if the timer expires before being reset by software. The programmer initializes the count value of the timer, enables the appropriate interrupt, then enables the timer. Thereafter, the software must reload the counter before it counts to zero from the programmed value. This protects the system from remaining in an unknown state where software, which would normally reset the timer, has stopped running due to an external noise condition or software error.

If configured to generate a hardware reset, the watchdog timer resets both the core and the 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 processor. 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 periods of external events. These timers can be synchronized to an external clock input to the PF1 pin, an external clock input to the PPI_CLK pin, or to the internal SCLK.

The timer units can be used in conjunction with the UART 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 processor incorporates four dual-channel synchronous serial ports for serial and multiprocessor communications. The SPORTs support the following features:

- I²S capable operation.
- Bidirectional operation – Each SPORT has two sets of independent transmit and receive pins, enabling eight channels of I²S stereo audio.
- Buffered (8-deep) transmit and receive ports – Each port has a data register for transferring data words to and from other processor components and shift registers for shifting data in and out of the data registers.

- Clocking – Each transmit and receive port can either use an external serial clock or generate its own, in frequencies ranging from ($f_{SCLK}/131,070$) Hz to ($f_{SCLK}/2$) Hz.
- Word length – Each SPORT supports serial data words from 3 to 32 bits in length, transferred most significant bit first or least significant bit first.
- Framing – Each transmit and receive port can run with or without frame sync signals for each data word. Frame sync signals can be generated internally or externally, active high or low, and with either of two pulsewidths 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{SPIxSS}) lets other SPI devices select the processor. For SPI0, seven SPI chip select output pins ($SPI0SEL7-1$) let the processor select other SPI devices. The SPI select pins are reconfigured GPIO pins. SPI1 and SPI2 have a single SPI select for SPI point-to-point communication. 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 it has an integrated DMA controller, configurable to support transmit or receive data streams. Each SPI's DMA controller can only service unidirectional accesses at any given time.

The SPI port clock rate is calculated as:

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

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

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

TWO WIRE INTERFACE

The ADSP-BF539/ADSP-BF539F processor incorporates two Two 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 (SCL) and data (SDA) and supports the protocol at speeds up to 400 kbits/sec.

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

UART PORT

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 port includes support for 5 to 8 data bits, 1 or 2 stop bits, and none, even, or odd parity. The UART port supports two modes of operation:

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

The 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 to 12 bits per frame.
- Both transmit and receive operations can be configured to generate maskable interrupts to the processor.

The UART port's clock rate is calculated as:

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

Where the 16-bit UART_Divisor comes from the DLH register (most significant 8 bits) and DLL register (least significant 8 bits).

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

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

PROGRAMMABLE I/O PINS

The ADSP-BF539/ADSP-BF539F processor has numerous peripherals that may not all be required for every application. Many of the pins thus have a secondary function, as general purpose 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 (PFx)

The ADSP-BF539/ADSP-BF539F processor has 16 bi-directional, general purpose Programmable Flag (PF15–0) pins. 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 ADSP-BF539/ADSP-BF539F processor employs 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.

General Purpose I/O

The ADSP-BF539/ADSP-BF539F has 38 General Purpose I/O pins that are multiplexed with other peripherals. They are arranged into Port C, D, E as shown in [Table 4 on Page 13](#). The GPIO differ from the Programmable Flags in that the GPIO pins cannot generate interrupts to the processor.

The general purpose I/O pins may be individually controlled by manipulation of the control and status registers. These pins will not cause interrupts to be generated to the processor, but may 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 ADSP-BF539/ADSP-BF539F processor employs a “write one to modify” mechanism that allows any combination of individual GPIO to be modified in a single instruction, without affecting the level of any other GPIO. Four control registers and a Data register are provided for each GPIO port. One register is written in order to set GPIO values, one register is written in order to clear GPIO values, one register is written in order to toggle GPIO values, and one register is written in order to specify a GPIO input or output. Reading the GPIO Data 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 ports C9–C4 at power up. If GP is tied high, then ports C9–C4 are configured as GPIO pins after reset. The ports cannot be reconfigured through software and special care must be taken with the MLF pin. If the GP pin is tied low, then the ports are configured as MXVR pins after reset, but may be reconfigured as GPIO pins through software.

Table 4. Programmable Flag / GPIO Ports

Peripheral	Alternate Programmable Flag / GPIO Port Function
PPI	PF15–0
SPORT0	
SPORT1	
SPORT2	GPIO Port E7–0
SPORT3	GPIO Port E15–8
SPI0	
SPI1	GPIO Port D4–0
SPI2	GPIO Port D9–5
UART0	
UART1	GPIO Port D11–10
UART2	GPIO Port D13–12
CAN	GPIO Port C1–0
MXVR	GPIO Port C9–4
TWIO	
TWI1	

PARALLEL PERIPHERAL INTERFACE

The ADSP-BF539/ADSP-BF539F processor provides a Parallel Peripheral Interface (PPI) that can connect directly to parallel A/D and D/A converters, video encoders and decoders, and other general purpose peripherals. The PPI consists of a dedicated input clock pin, up to 3 frame synchronization pins, and

up to 16 data pins. The input clock supports parallel data rates up to $f_{CLK}/2$ 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, bi-directional 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, bi-directional transfer of 8- or 10-bit video data. Additionally, on-chip decode of embedded start-of-line (SOL) and start-of-field (SOF) preamble packets is supported.

General Purpose Mode Descriptions

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

- 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. Data widths of 8, 10, 11, 12, 13, 14, 15 and 16-bits are supported, as programmed by 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 processor controls 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 may be embedded in horizontal and vertical blanking intervals. Data transfer starts immediately after synchronization to Field 1. The MXVR supports synchronous data, asynchronous packets and control messages. Data is transferred to or from the synchronous channels through eight DMA engines that work autonomously from the processor core.

CONTROLLER AREA NETWORK (CAN) INTERFACE

The ADSP-BF539/ADSP-BF539F processor provides 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 capability 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 ADSP-BF539/ADSP-BF539F processor from Sleep mode upon generation of a wakeup event, such that the processor can be maintained in a low power mode during idle conditions. Additionally, a CAN wakeup event can wake up the ADSP-BF539/ADSP-BF539F processor from Deep Sleep or Hibernate mode, and wake up the on-chip internal voltage regulator from a powered-down state.

The electrical characteristics of each network connection are very stringent, therefore the CAN interface is typically divided into 2 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 processor provides 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 the industry standard standalone MOST controller devices, supporting 22.579 Mbps or 24.576 Mbps data transfer. It offers faster lock times, greater jitter immunity, a sophisticated DMA scheme for data transfers, and 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.

The MXVR supports synchronous data, asynchronous packets, and control messages using dedicated DMA engines which operate autonomously from the processor core moving data to and from L1 memory. 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 which trigger DMA operation when data patterns are detected in the receive data stream. Furthermore two DMA engines support asynchronous traffic and a further two support 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 ADSP-BF539/ADSP-BF539F processor from Sleep mode when a wakeup 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 ADSP-BF539/ADSP-BF539F processor from Deep Sleep or Hibernate mode, and wake up the on-chip internal voltage regulator from a powered-down state. These features allow the ADSP-BF539/ADSP-BF539F 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, a 45.1584 MHz crystal or oscillator should be used; or for 48 KHz frame syncs, a 49.152 MHz crystal or oscillator should be used.

DYNAMIC POWER MANAGEMENT

The ADSP-BF539/ADSP-BF539F processor provides five 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 powerup 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 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. In this mode, the CLKIN to CCLK multiplier ratio can be changed, although the changes are not realized until the Full-On mode is entered. 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	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 will wake up the processor. When in the Sleep mode, assertion of wakeup will cause the processor to sense the value of the BYPASS bit in the PLL Control register (PLL_CTL). If BYPASS is disabled, the processor will transition 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 (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 Operating Mode—Maximum Static Power Savings

The Hibernate mode 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 disables both CCLK and SCLK. Furthermore, it 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 non-volatile storage device prior to removing power if the processor state is to be preserved. Since V_{DDEXT} is still 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 have power still applied without drawing unwanted current. The internal supply regulator can be woken up either by a Real Time Clock wakeup, by CAN bus traffic, by asserting the $\overline{\text{RESET}}$ pin or by MOST bus traffic causing the $\overline{\text{MRXON}}$ pin to assert.

Power Savings

As shown in Table 6, the ADSP-BF539/ADSP-BF539F processor supports three different power domains. The use of multiple power domains maximizes flexibility, while maintaining compliance with industry standards and conventions. By isolating the internal logic of the ADSP-BF539/ADSP-BF539F processor into its own power domain, separate from the RTC and other I/O, the processor can take advantage of Dynamic Power Management, without affecting the RTC or other I/O devices. There are no sequencing requirements for the various power domains.

Table 6. Power Domains

Power Domain	VDD Range
RTC crystal I/O and logic	VDDRTC
MXVR crystal I/O	MXEVDD
MXVR PLL analog and logic	MPIVDD
All internal logic except RTC and MXVR PLL	VDDINT
All I/O except RTC and MXVR crystals	VDEEXT

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 power dissipation, while reducing the voltage by 25% reduces 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 processor allows both the processor's input voltage (V_{DDINT}) and clock frequency (f_{CLK}) 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:

$$\text{Power Savings Factor} = \frac{f_{CLKRED}}{f_{CLKNOM}} \times \left(\frac{V_{DDINTRED}}{V_{DDINTNOM}} \right)^2 \times \left(\frac{T_{RED}}{T_{NOM}} \right)$$

where the variables in the equations are:

- f_{CLKNOM} is the nominal core clock frequency
- f_{CLKRED} 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_{CLKNOM}
- T_{RED} is the duration running at f_{CLKRED}

The Power Savings Factor is calculated as:

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

VOLTAGE REGULATION

The Blackfin processor provides an on-chip voltage regulator that can generate processor core voltage levels 1.0 V (-5%/+10%) to 1.2V (-5% / +10%) from an external 2.7 V to 3.6 V supply. Figure 6 shows the typical external components required to complete the power management system.[†] The regulator controls the internal logic voltage levels and is programmable with the Voltage Regulator Control Register (VR_CTL) in increments of 50 mV. To reduce standby power consumption, the internal voltage regulator can be programmed to remove power to the processor core while I/O power (VDDRTC, MXEVDD, VDDEXT) is still supplied. While in Hibernate mode, I/O power is still being applied, eliminating the need for external buffers. The voltage regulator can be activated from this power-down state either through an RTC wakeup, a CAN wakeup, an MXVR wakeup, or by asserting RESET, 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 processor can be clocked by an external crystal, a sine wave input, or a buffered, shaped clock derived from an external clock oscillator.

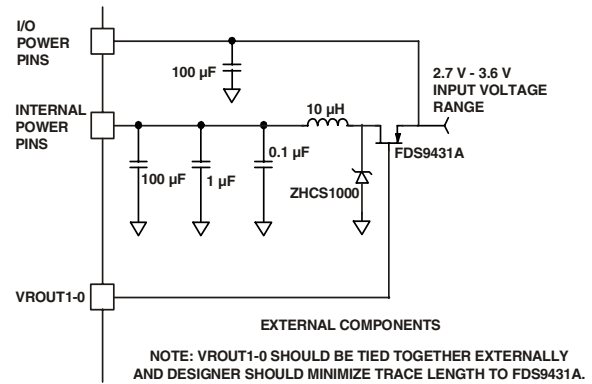


Figure 6. Voltage Regulator Circuit

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 processor includes an on-chip oscillator circuit, an external crystal may be used. The crystal should be connected across the CLKIN and XTAL pins, with two capacitors connected as shown in Figure 7. Capacitor values are dependent on crystal type and should be specified by the crystal manufacturer. A parallel-resonant, fundamental frequency, microprocessor-grade crystal should be used.

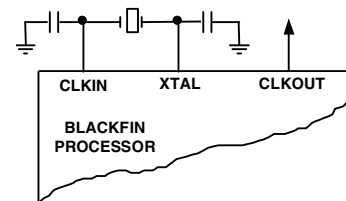


Figure 7. External Crystal Connections

As shown in Figure 8 on Page 17, 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 0.5x to 64x multiplication factor (bounded by specified minimum and maximum VCO frequencies). The default multiplier is 10x, 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.

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

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

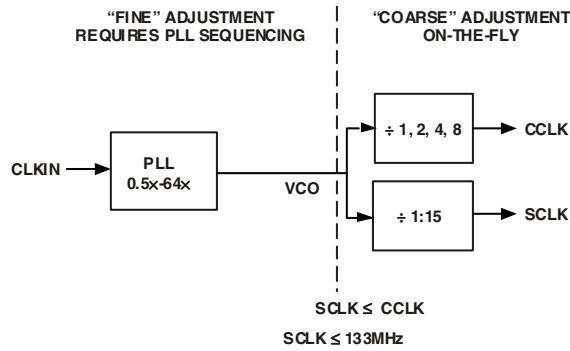


Figure 8. Frequency Modification Methods

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 processor has 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 (ADSP-BF539 only) or Boot from onboard flash (ADSP-BF539F only)
10	Boot from SPI serial master
11	Boot from SPI serial slave EEPROM /flash (8-, 16-, or 24-bit address range, or Atmel AT45DB041, AT45DB081, or AT45DB161 serial flash)

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 external flash memory – The 8-bit flash boot routine located in boot ROM memory space is set up using asynchronous memory bank 0. If FCE is connected to AMS0, then the on-chip flash is booted from the ADSP-BF539F. 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) – The SPI 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 – 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.

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 may 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 capability 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- 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 processor is supported with a complete set of CROSSCORE[†] software and hardware development tools, including Analog Devices emulators and

VisualDSP++[‡] development environment. The same emulator hardware that supports other Blackfin processors also fully emulates the ADSP-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 non intrusively 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.
- 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.

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

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

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.

VCSE is Analog Devices technology for creating, using, and reusing software components (independent modules of substantial functionality) to quickly and reliably assemble software applications. Download components from the Web and drop them into the application. Publish component archives from within VisualDSP++. VCSE supports component implementation in C/C++ or assembly language.

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, 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 processor 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. Non intrusive 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 *EE-68: Analog Devices JTAG Emulation Technical Reference* 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 connecting to a MOST network. This diagram is intended as an example and exact connections and recommended circuit values should be obtained from Analog Devices.

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 *EE-228: Switching Regulator Design Considerations for ADSP-BF533 Blackfin Processors* applications note on the Analog Devices web site (www.analog.com)—use site search on “EE-228.”

MXVR BOARD LAYOUT GUIDELINES

MLF pin

- Capacitors:
 - C1: 0.1 μ F (PPS type, 2% tolerance recommended)
 - C2: 0.01 μ F (PPS type, 2% tolerance recommended)
- Resistor:
 - R1: 220 ohm (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.

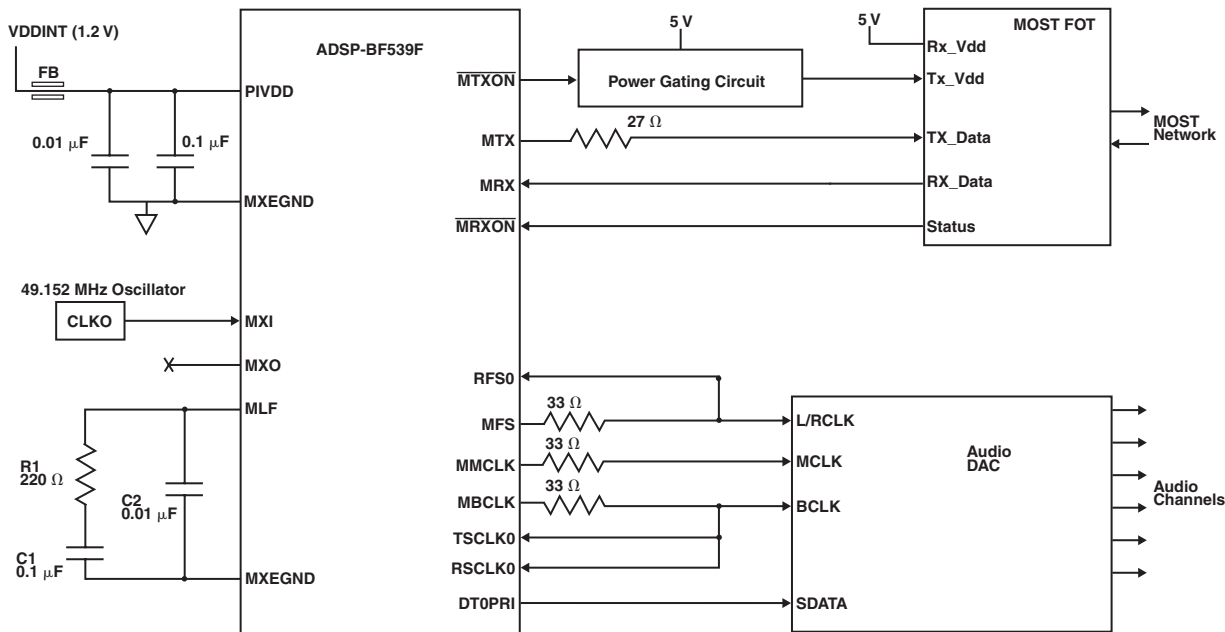


Figure 9. Example connections of ADSP-BF539/ADSP-BF539F to MOST Network

- 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 49.152MHz or 45.1584MHz 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 49.152MHz or 45.1584MHz 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 3pF.
- 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 3.3V Power

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

MPIVDD - MXVR PLL 1.2V Power

- Should be routed with wide traces or as power plane.
- A ferrite bead should be placed between the 1.2V VDDINT power plane and the MPIVDD pin for noise isolation.
- Locally bypass MPIVDD with 0.1μF and 0.01μF decoupling capacitors to MXEGND.
- Avoid routing other switching signals near to MPIVDD to avoid crosstalk.

Fiber Optic Transceiver (FOT) Connections

- The traces between ADSP-BF539/ADSP-BF539F 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 ADSP-BF539/ADSP-BF539F MTX output pin to the FOT Transmit Data input pin should have a 27 Ohm series termination resistor placed close to the ADSP-BF539/ADSP-BF539F MTX pin.
- The receive data trace and the transmit data trace between the ADSP-BF539/ADSP-BF539F and the FOT should not be routed close to each other in parallel over long distances to avoid crosstalk.

PIN DESCRIPTIONS

ADSP-BF539/ADSP-BF539F processor pin definitions are listed in Table 10. In order to maintain maximum functionality and reduce package size and pin count, some pins have dual, multi-

plexed 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	Function	Driver Type ¹	Pull-Up/Down Requirement
<i>Memory Interface</i>				
ADDR19-1	O	Address Bus for Async/Sync Access	A	Pull High When Not Used
DATA15-0	I/O	Data Bus for Async/Sync Access	A	
$\overline{\text{ABE1-0}}/\text{SDQM1-0}$	O	Byte Enables/Data Masks for Async/Sync Access	A	
$\overline{\text{BR}}$	I	Bus Request		
$\overline{\text{BG}}$	O	Bus Grant	A	
$\overline{\text{BGH}}$	O	Bus Grant Hang	A	
<i>Asynchronous Memory Control</i>				
$\overline{\text{AMS3-0}}$	O	Bank Select	A	Pull LOW when not used.
ARDY	I	Hardware Ready Control		
$\overline{\text{AOE}}$	O	Output Enable	A	
$\overline{\text{ARE}}$	O	Read Enable	A	
$\overline{\text{AWE}}$	O	Write Enable	A	
<i>Flash Control</i>				
$\overline{\text{FCE}}$	I	Flash Enable (ADSP-BF539F only)		Leave unconnected for ADSP-BF539
$\overline{\text{FRESET}}$	I	Flash Reset (ADSP-BF539F only)		Leave unconnected for ADSP-BF539
<i>Synchronous Memory Control</i>				
$\overline{\text{SRAS}}$	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	
<i>Timers</i>				
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	
<i>Parallel Peripheral Interface Port/GPIO</i>				
PF0/SPI0SS	I/O	Programmable Flag 0/SPI Slave Select Input	C	
PF1/SPI0SEL1/TMRCLK	I/O	Programmable Flag 1/SPI Slave Select Enable 1/ External Timer Reference	C	
PF2/SPI0SEL2	I/O	Programmable Flag 2/SPI Slave Select Enable 2	C	
PF3/SPI0SEL3/PPI_FS3	I/O	Programmable Flag 3/SPI Slave Select Enable 3/ PPI Frame Sync 3	C	

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type ¹	Pull-Up/Down Requirement
PF4/SPIOSEL4/PPI15	I/O	Programmable Flag 4/ SPI Slave Select Enable 4/PPI 15	C	
PF5/SPIOSEL5/PPI14	I/O	Programmable Flag 5/ SPI Slave Select Enable 5/PPI 14	C	
PF6/SPIOSEL6/PPI13	I/O	Programmable Flag 6/ SPI Slave Select Enable 6/PPI 13	C	
PF7/SPIOSEL7/PPI12	I/O	Programmable Flag 7/ SPI 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	I	PPI Clock		
Controller Area Network				
CANTX/GPIOC0	I/O 5V	CAN Transmit/GPIO_C_0	C	
CANRX/GPIOC1	I/O 5V	CAN Receive/GPIO_C_1	C	
Media Transceiver (MXVR)/ General Purpose I/O				
MTX/GPIOC5	I/O	MXVR Transmit Data/GPIO_C_5	C	
$\overline{\text{MTXON}}$ /GPIOC9	I/O	MXVR Transmit FOT On/GPIO_C_9	C	
MRX/GPIOC4	I/O 5V	MXVR Receive Data/GPIO_C_4	C	Pull LOW when not used.
$\overline{\text{MRXON}}$	I 5V	MXVR FOT Receive On		Pull High When Not Used
MXI	I	MXVR Crystal Input		Pull LOW when not used.
MXO	O	MXVR Crystal Output		Leave unconnected when not used.
MLF	I	MXVR Loop Filter		Pull LOW when not used.
MMCLK/GPIOC6	I/O	MXVR Master Clock/GPIO_C_6	C	
MBCLK/GPIOC7	I/O	MXVR Bit Clock/GPIO_C_7	C	
MFS/GPIOC8	I/O	MXVR Frame Sync/GPIO_C_8	C	
GP	I	GPIO_C_4-9 Enable		Pull LOW when not used.
Two Wire Interface Port				
SDA0	I/O 5V	TWI0 Serial Data	E ²	
SCL0	I/O 5V	TWI0 Serial Clock	E ²	
SDA1	I/O 5V	TWI1 Serial Data	E ²	
SCL1	I/O 5V	TWI1 Serial Clock	E ²	

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type ¹	Pull-Up/Down Requirement
<i>Serial Port0</i>				
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	
<i>Serial Port1</i>				
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	
<i>Serial Port2</i>				
RSCLK2 / GPIOE0	I/O	SPORT2 Receive Serial Clock/GPIO_E_0	D	
RFS2 / GPIOE1	I/O	SPORT2 Receive Frame Sync/GPIO_E_1	C	
DR2PRI / GPIOE2	I/O	SPORT2 Receive Data Primary/GPIO_E_2		
DR2SEC / GPIOE3	I/O	SPORT2 Receive Data Secondary/GPIO_E_3		
TSCLK2 / GPIOE4	I/O	SPORT2 Transmit Serial Clock/GPIO_E_4	D	
TFS2 / GPIOE5	I/O	SPORT2 Transmit Frame Sync/GPIO_E_5	C	
DT2PRI / GPIOE6	I/O	SPORT2 Transmit Data Primary/GPIO_E_6	C	
DT2SEC / GPIOE7	I/O	SPORT2 Transmit Data Secondary/GPIO_E_7	C	
<i>Serial Port3</i>				
RSCLK3 / GPIOE8	I/O	SPORT3 Receive Serial Clock/GPIO_E_8	D	
RFS3 / GPIOE9	I/O	SPORT3 Receive Frame Sync/GPIO_E_9	C	
DR3PRI / GPIOE10	I/O	SPORT3 Receive Data Primary/GPIO_E_10		
DR3SEC / GPIOE11	I/O	SPORT3 Receive Data Secondary/GPIO_E_11		
TSCLK3 / GPIOE12	I/O	SPORT3 Transmit Serial Clock/GPIO_E_12	D	
TFS3 / GPIOE13	I/O	SPORT3 Transmit Frame Sync/GPIO_E_13	C	
DT3PRI / GPIOE14	I/O	SPORT3 Transmit Data Primary/GPIO_E_14	C	
DT3SEC / GPIOE15	I/O	SPORT3 Transmit Data Secondary/GPIO_E_15	C	

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type ¹	Pull-Up/Down Requirement
<i>SPI0 Port</i>				
MOSI0	I/O	SPI0 Master Out Slave In	C	
MISO0	I/O	SPI0 Master In Slave Out	C ³	
SCK0	I/O	SPI0 Clock	D	
<i>SPI1 Port</i>				
MOSI1 / GPIOD0	I/O	SPI1 Master Out Slave In/GPIO_D_0	C	
MISO1 / GPIOD1	I/O	SPI1 Master In Slave Out/GPIO_D_1	C	
SCK1 / GPIOD2	I/O	SPI1 Clock/GPIO_D_2	D	
SPI1SS / GPIOD3	I/O	SPI1 Slave Select Input/GPIO_D_3	C	
SPI1SEL / GPIOD4	I/O	SPI1 Slave Select Enable/GPIO_D_4	C	
<i>SPI2 Port</i>				
MOSI2 / GPIOD5	I/O	SPI2 Master Out Slave In/GPIO_D_5	C	
MISO2 / GPIOD6	I/O	SPI2 Master In Slave Out/GPIO_D_6	C	
SCK2 / GPIOD7	I/O	SPI2 Clock/GPIO_D_7	D	
SPI2SS / GPIOD8	I/O	SPI2 Slave Select Input/GPIO_D_8	C	
SPI2SEL / GPIOD9	I/O	SPI2 Slave Select Enable/GPIO_D_9	C	
<i>UART0 Port</i>				
RX0	I	UART Receive		
TX0	O	UART Transmit	C	
<i>UART1 Port</i>				
RX1 / GPIOD10	I/O	UART1 Receive/GPIO_D_10	C	
TX1 / GPIOD11	I/O	UART1 Transmit/GPIO_D_11	C	
<i>UART2 Port</i>				
RX2 / GPIOD12	I/O	UART2 Receive/GPIO_D_12	C	
TX2 / GPIOD13	I/O	UART2 Transmit/GPIO_D_13	C	
<i>Real Time Clock</i>				
RTXI	I	RTC Crystal Input		Pull LOW when not used.
RTXO	O	RTC Crystal Output		
<i>JTAG Port</i>				
TCK	I	JTAG Clock		Pull LOW when JTAG port not used.
TDO	O	JTAG Serial Data Out	C	
TDI	I	JTAG Serial Data In		
TMS	I	JTAG Mode Select		
$\overline{\text{TRST}}$	I	JTAG Reset		
$\overline{\text{EMU}}$	O	Emulation Output	C	
<i>Clock</i>				
CLKIN	I	Clock/Crystal Input		
XTAL	O	Crystal Output		

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type ¹	Pull-Up/Down Requirement
<i>Mode Controls</i>				
$\overline{\text{RESET}}$	I	Reset		Pull High When Not Used
$\overline{\text{NMI}}$	I	Non-maskable Interrupt		
BMODE1-0	I	Boot Mode Strap		
<i>Voltage Regulator</i>				
VROUT0	O	External FET Drive 0		Leave unconnected when not used.
VROUT1	O	External FET Drive 1		Leave unconnected when not used.
<i>Supplies</i>				
VDEXT	P	I/O Power Supply		
VDDINT	P	Internal Power Supply		
VDDRTC	P	Real Time Clock Power Supply		
MPIVDD	P	MXVR Internal Power Supply		
MXEVDD	P	MXVR External Power Supply		
MXEGND	G	MXVR Ground		
GND	G	Ground		

¹ Refer to [Figure 38 on Page 59](#) to [Figure 49 on Page 60](#).² Open drain output. See version 2.1 of I²C specification for proper resistor value.³ Pull HIGH through a 4.7 k Ω resistor if booting via the SPI port.

SPECIFICATIONS

Component specifications are subject to change without notice.

RECOMMENDED OPERATING CONDITIONS

Parameter		Min	Nominal	Max	Unit
V _{DDINT}	Internal Supply Voltage ¹	0.95	1.2	1.32	V
V _{DDEXT}	External Supply Voltage ²	2.7	3.3	3.6	V
V _{DDRTC}	Real Time Clock Power Supply Voltage	2.25		3.6	V
V _{IH}	High Level Input Voltage ³ , @ V _{DDEXT} = maximum	2.0		3.6	V
V _{IHCLKIN}	High Level Input Voltage ⁴ , @ V _{DDEXT} = maximum	2.2		3.6	V
V _{IL}	Low Level Input Voltage ^{3,5} , @ V _{DDEXT} = minimum	-0.3		0.6	V
T _{AMBIENT}	Automotive Ambient Operating Temperature	-40		85	°C

¹ Parameter value applies also to MPIVDD.

² Parameter value applies also to MXEVDD.

³ The 3.3 V tolerant pins are capable of accepting up to 3.6 V maximum V_{IH}. The following bi-directional pins are 3.3 V tolerant: DATA15-0, MISO0, MOSI0, PF15-0, PPI3-0, MTXON, MMCLK, MBCLK, MFS, MTX, SCK1, MISO1, MOSI1, SPI1SS, SPI1SEL, SCK2, MISO2, MOSI2, SPI2SS, SPI2SEL, RX1, TX1, RX2, TX2, DT2PRI, DT2SEC, TSCLK2, DR2PRI, DT2SEC, RSCLK2, RFS2, TFS2, DT3PRI, DT3SEC, DR3PRI, DR3SEC, RSCLK3, RFS3, TFS3, RFS0, RFS1, RSCLK0, RSCLK1, TSCLK0, TSCLK1, SCK0, TFS0, TFS1, and TMR2-0. The following input-only pins are 3.3 V tolerant: RESET, RX0, TCK, TDI, TMS, TRST, ARDY, BMODE1-0, BR, DR0PRI, DR0SEC, DR1PRI, DR1SEC, NMI, PPI_CLK, RTXI, and GP.

⁴ Parameter value applies to the CLKIN and MXI input pins.

⁵ Parameter value applies to all input and bi-directional pins.

RECOMMENDED OPERATING CONDITIONS

—APPLIES TO 5V TOLERANT PINS

Parameter ¹		Min	Nominal	Max	Unit
V _{IH5V} ²	High Level Input Voltage, @ V _{DDEXT} = maximum	2.0		5.5	V
V _{IL5V} ²	Low Level Input Voltage, @ V _{DDEXT} = minimum	-0.3		0.8	V

¹ Specifications subject to change without notice.

² The 5 V tolerant pins are capable of accepting up to 5.5 V maximum V_{IH}. The following bi-directional pins are 5 V tolerant: SCL0, SCL1, SDA0, SDA1, MRX, CANRX, CANTX. The following input-only pins are 5 V tolerant: MRXON.

ELECTRICAL CHARACTERISTICS

Parameter ¹	Test Conditions	Min	Max	Unit
V _{OH}	High Level Output Voltage ²	@ V _{DDEXT} = 3.0V, I _{OH} = -0.5 mA	2.4	V
V _{OL}	Low Level Output Voltage ²	@ V _{DDEXT} = 3.0V, I _{OL} = 2.0 mA	0.4	V
I _{IH}	High Level Input Current ³	@ V _{DDEXT} = maximum, V _{IN} = V _{DD} maximum	TBD	μA
I _{IL}	Low Level Input Current ⁴	@ V _{DDEXT} = maximum, V _{IN} = 0 V	TBD	μA
I _{OZH}	Three-State Leakage Current ⁴	@ V _{DDEXT} = maximum, V _{IN} = V _{DD} maximum	TBD	μA
I _{OZL}	Three-State Leakage Current ⁵	@ V _{DDEXT} = maximum, V _{IN} = 0 V	TBD	μA
C _{IN}	Input Capacitance ^{5,6}	f _{IN} = 1 MHz, T _{AMBIENT} = 25°C, V _{IN} = 2.5 V	TBD	pF

¹ Specifications subject to change without notice.

² Applies to output and bidirectional pins.

³ Applies to input pins.

⁴ Applies to three-statable pins.

⁵ Applies to all signal pins.

⁶ Guaranteed, but not tested.

ABSOLUTE MAXIMUM RATINGS

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

Parameter	Rating
Internal (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	-0.5 V to 3.6 V
Input Voltage ⁴	-0.5 V to 5.5 V
Output Voltage Swing	-0.5 V to $V_{DDEXT} + 0.5$ V
Load Capacitance ⁵	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 11.

⁴ Applies to pins designated as 5V tolerant only.

⁵ For proper SDRAM controller operation, the maximum load capacitance is 50 pF (at 3.3V) or 30 pF (at 2.5V) for ADDR19-1, DATA15-0, ABE1-0/SDQM1-0, CLKOUT, SCKE, SA10, SRAS, SCAS, SWE, and SMS.

Table 11. Maximum Duty Cycle for Input Transient Voltage¹

V_{IN} Min (V)	V_{IN} Max (V)	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.

ESD SENSITIVITY

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADSP-BF539/ADSP-BF539F processor feature proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



PACKAGE INFORMATION

The information presented in Figure 10 and Table 12 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 69.



Figure 10. Product Information on Package

Table 12. Package Brand Information

Brand Key	Field Description
W	Automotive Grade (Optional)
t	Temperature Range
pp	Package Type
Z	Lead Free Option (Optional)
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
yyww	Date Code

TIMING SPECIFICATIONS

Table 13 describes the timing requirements for the ADSP-BF539/ADSP-BF539F processor clocks. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock, system clock and Voltage Controlled Oscillator

(VCO) operating frequencies, as described in [Absolute Maximum Ratings on Page 28](#). Table 14 describes Phase Locked Loop operating conditions.

Table 13. Core Clock Requirements—ADSP-BF539/ADSP-BF539F—500 MHz

Parameter		Minimum	Maximum	Unit
t _{CCLK}	Core Cycle Period (V _{DDINT} =1.20 V minimum)	2.00		ns
t _{CCLK}	Core Cycle Period (V _{DDINT} =1.045 V minimum)	2.25		ns
t _{CCLK}	Core Cycle Period (V _{DDINT} =0.95 V minimum)	2.50		ns
t _{CCLK}	Core Cycle Period (V _{DDINT} =0.85 V minimum)	3.00		ns
t _{CCLK}	Core Cycle Period (V _{DDINT} =0.80 V minimum)	4.00		ns

Table 14. Phase Locked Loop Operating Conditions

Parameter		Minimum	Maximum	Unit
f _{VCO}	Voltage Controlled Oscillator (VCO) Frequency	50	Max CCLK	MHz

Table 15. Maximum SCLK Conditions

Parameter ¹		V _{DDEXT} = 3.3 V	V _{DDEXT} = 2.5 V	Unit
f _{SCLK}	CLKOUT/SCLK Frequency (V _{DDINT} ≥ 1.14 V)	133	133	MHz
f _{SCLK}	CLKOUT/SCLK Frequency (V _{DDINT} < 1.14 V)	100	100	MHz

¹ t_{SCLK} (= 1/f_{SCLK}) must be greater than or equal to t_{CCLK}.

Clock and Reset Timing

Table 16 and Figure 11 describe clock and reset operations. Per Absolute Maximum Ratings on Page 28, combinations of CLKIN and clock multipliers must not select core/peripheral clocks in excess of 500/133 MHz.

Table 16. Clock and Reset Timing

Parameter	Minimum	Maximum	Unit
<i>Timing Requirements</i>			
t_{CKIN} CLKIN Period	20.0	100.0	ns
t_{CKINL} CLKIN Low Pulse ¹	8.0		ns
t_{CKINH} CLKIN High Pulse ¹	8.0		ns
t_{WRST} \overline{RESET} Asserted Pulsewidth Low ²	11 t_{CKIN}		ns

¹ Applies to bypass mode and non-bypass mode.

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

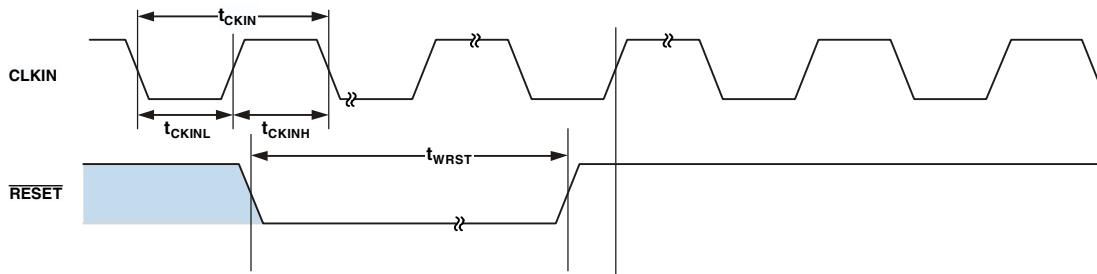


Figure 11. Clock and Reset Timing

Asynchronous Memory Read Cycle Timing

Table 17 and Table 18 on Page 31 and Figure 12 and Figure 13 on Page 33 describe asynchronous memory read cycle operations for synchronous and for asynchronous ARDY.

Table 17. Asynchronous Memory Read Cycle Timing with Synchronous ARDY

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
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	tbd		ns
t _{HARDY}	ARDY Hold After the Falling Edge of CLKOUT	tbd		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 $\overline{\text{AMS3-0}}$, $\overline{\text{ABE1-0}}$, ADDR19–1, AO $\overline{\text{E}}$, $\overline{\text{ARE}}$.

Table 18. Asynchronous Memory Read Cycle Timing with Asynchronous ARDY

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
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 $\overline{\text{AMSx}}$ Asserted ¹		(S+RA–2)*t _{SCLK}	ns
t _{HAA}	ARDY Asserted Hold After $\overline{\text{ARE}}$ 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 $\overline{\text{AMS3-0}}$, $\overline{\text{ABE1-0}}$, ADDR19–1, AO $\overline{\text{E}}$, $\overline{\text{ARE}}$.

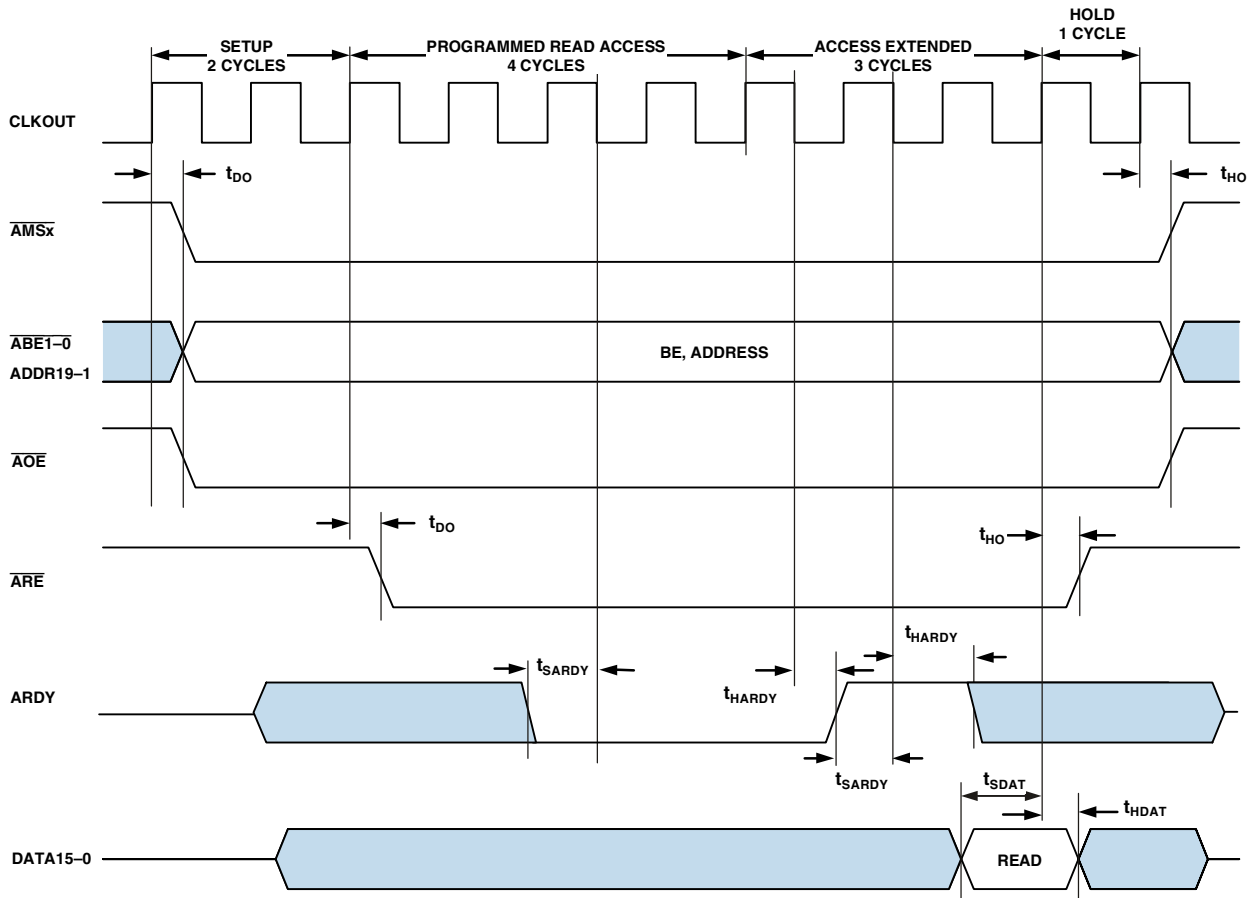


Figure 12. Asynchronous Memory Read Cycle Timing with Synchronous ARDY

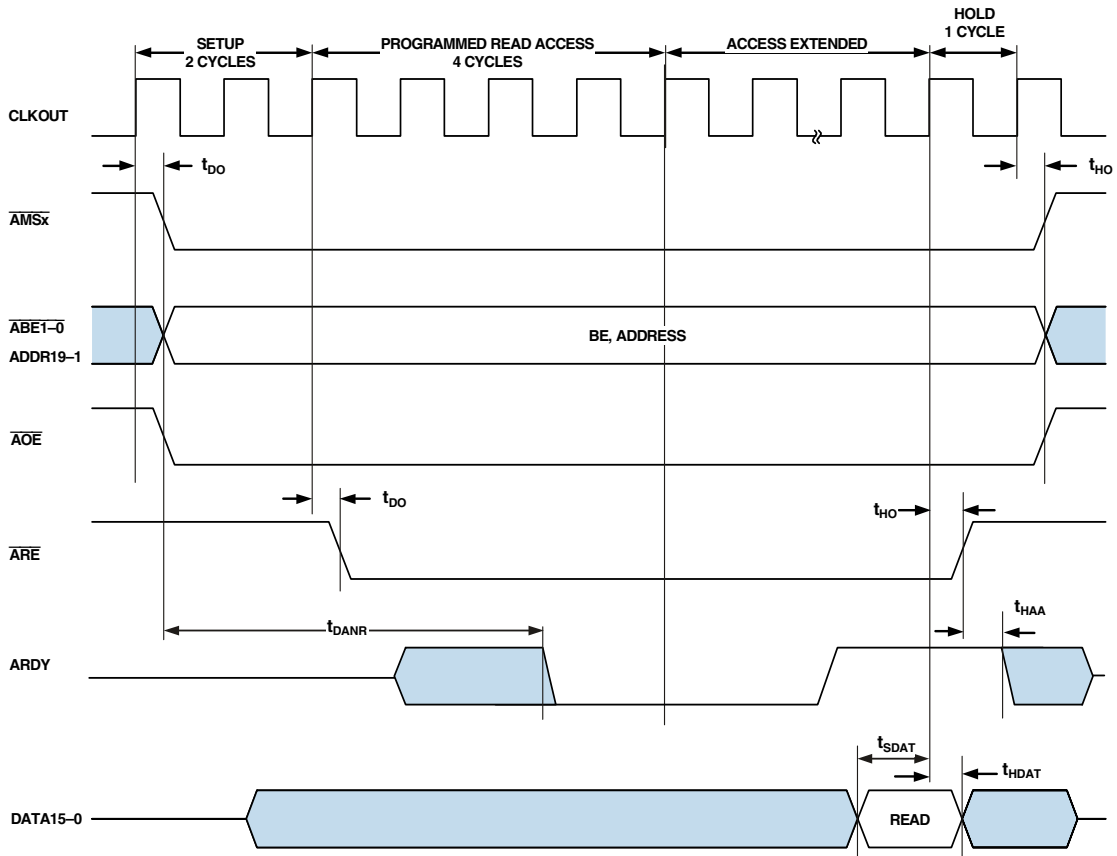


Figure 13. Asynchronous Memory Read Cycle Timing with Asynchronous ARDY

Asynchronous Memory Write Cycle Timing

Table 19 and Table 20 on Page 34 and Figure 14 and Figure 15 on Page 35 describe asynchronous memory write cycle operations for synchronous and for asynchronous ARDY.

Table 19. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
t_{SARDY}	ARDY Setup Before the Falling Edge of CLKOUT	tbd		ns
t_{HARDY}	ARDY Hold After the Falling Edge of CLKOUT	tbd		ns
<i>Switching Characteristics</i>				
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 $\overline{AMS3-0}$, $\overline{ABE1-0}$, $\overline{ADDR19-1}$, $\overline{DATA15-0}$, \overline{AOE} , \overline{AWE} .

Table 20. Asynchronous Memory Write Cycle Timing with Asynchronous ARDY

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
t_{DANR}	ARDY Negated Delay from \overline{AMSx} Asserted ¹		$(S+WA-2)*t_{SCLK}$	ns
t_{HAA}	ARDY Asserted Hold After \overline{ARE} Negated	0.0		ns
<i>Switching Characteristics</i>				
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 $\overline{AMS3-0}$, $\overline{ABE1-0}$, $\overline{ADDR19-1}$, $\overline{DATA15-0}$, \overline{AOE} , \overline{AWE} .

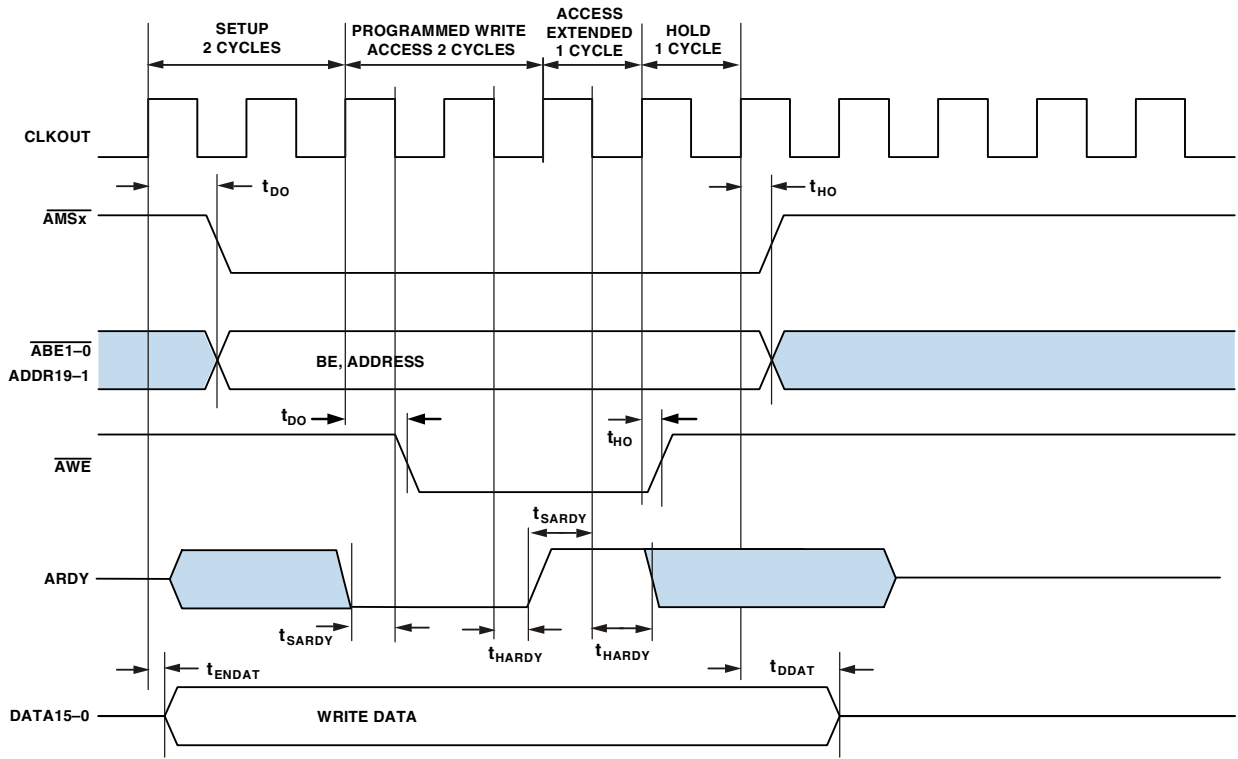


Figure 14. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

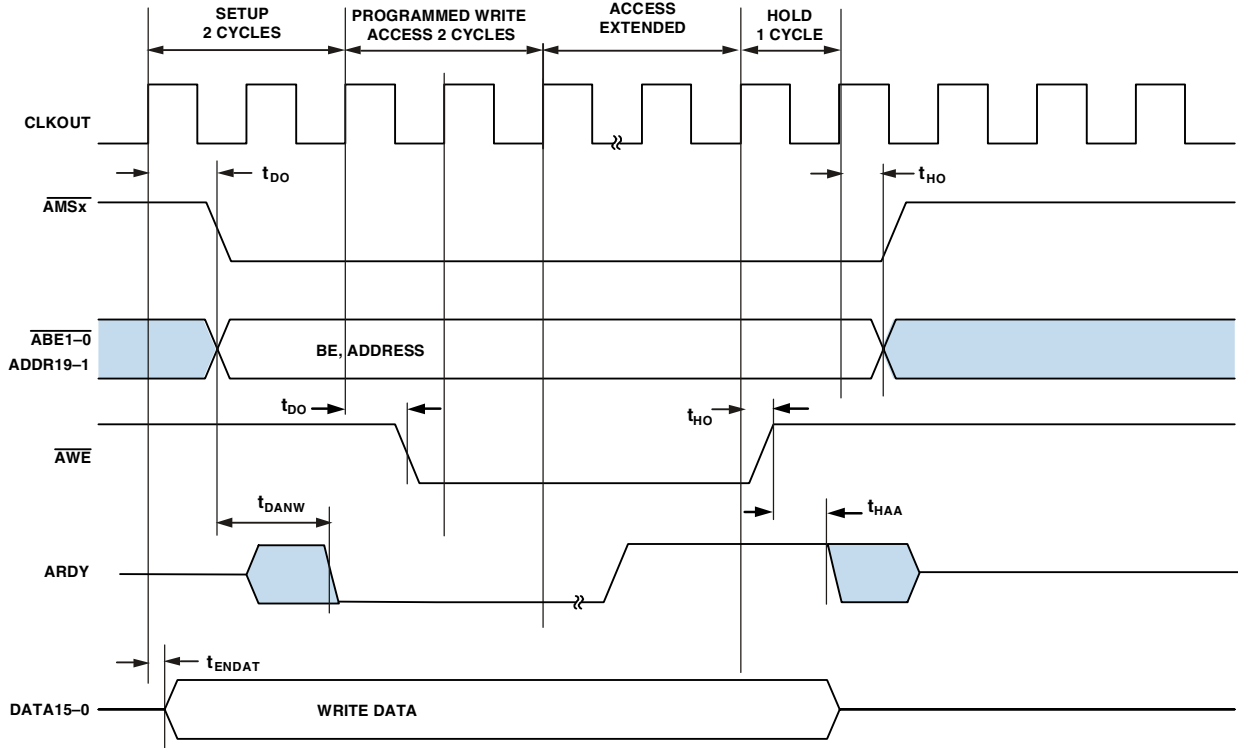


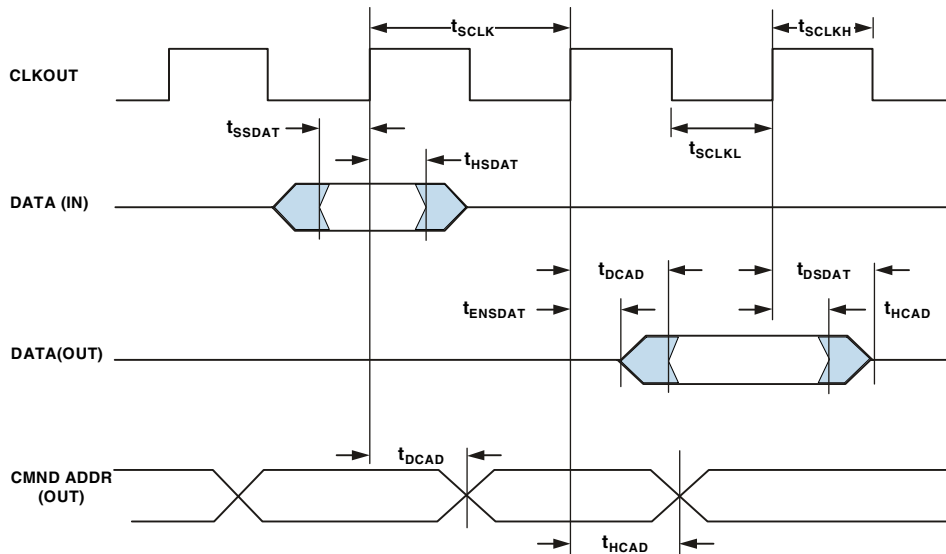
Figure 15. Asynchronous Memory Write Cycle Timing with Asynchronous ARDY

SDRAM Interface Timing

Table 21. SDRAM Interface Timing

Parameter	Minimum	Maximum	Unit
<i>Timing Requirements</i>			
t_{SSDAT} DATA Setup Before CLKOUT	2.1		ns
t_{HSDAT} DATA Hold After CLKOUT	0.8		ns
<i>Switching Characteristics</i>			
t_{SCLK} CLKOUT Period	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 ¹		6.0	ns
t_{HCAD} Command, ADDR, Data Hold After CLKOUT ¹	0.8		ns
t_{DSDAT} Data Disable After CLKOUT		6.0	ns
t_{ENSDAT} Data Enable After CLKOUT	1.0		ns

¹ Command pins include: \overline{SRAS} , \overline{SCAS} , \overline{SWE} , \overline{SDQM} , \overline{SMS} , SA10, \overline{SCKE} .



NOTE: COMMAND = \overline{SRAS} , \overline{SCAS} , \overline{SWE} , \overline{SDQM} , \overline{SMS} , SA10, \overline{SCKE} .

Figure 16. SDRAM Interface Timing

External Port Bus Request and Grant Cycle Timing

Table 22 and Table 23 on Page 37 and Figure 17 and Figure 18 on Page 39 describe external port bus request and grant cycle operations for synchronous and for asynchronous \overline{BR} .

Table 22. External Port Bus Request and Grant Cycle Timing with Synchronous \overline{BR}

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
t_{BS}	\overline{BR} Setup to Falling Edge of CLKOUT	tbd		ns
t_{BH}	Falling Edge of CLKOUT to \overline{BR} Deasserted Hold Time	tbd		ns
<i>Switching Characteristics</i>				
t_{SD}	CLKOUT Low to \overline{xMS} , Address, and $\overline{RD}/\overline{WR}$ disable		4.5	ns
t_{SE}	CLKOUT Low to \overline{xMS} , Address, and $\overline{RD}/\overline{WR}$ 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

Table 23. External Port Bus Request and Grant Cycle Timing with Asynchronous \overline{BR}

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
t_{WBR}	\overline{BR} Pulsewidth	$2 \times t_{SCLK}$		ns
<i>Switching Characteristics</i>				
t_{SD}	CLKOUT Low to \overline{xMS} , Address, and $\overline{RD}/\overline{WR}$ disable		4.5	ns
t_{SE}	CLKOUT Low to \overline{xMS} , Address, and $\overline{RD}/\overline{WR}$ 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

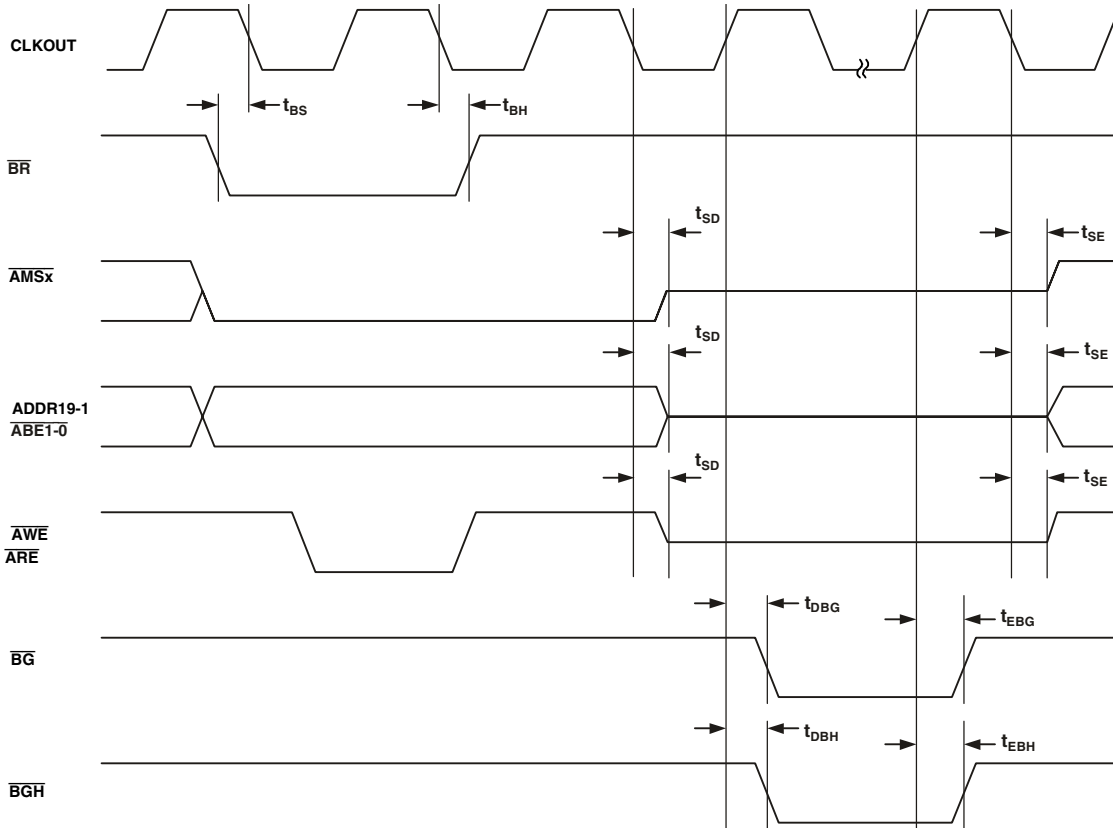


Figure 17. External Port Bus Request and Grant Cycle Timing with Synchronous \overline{BR}

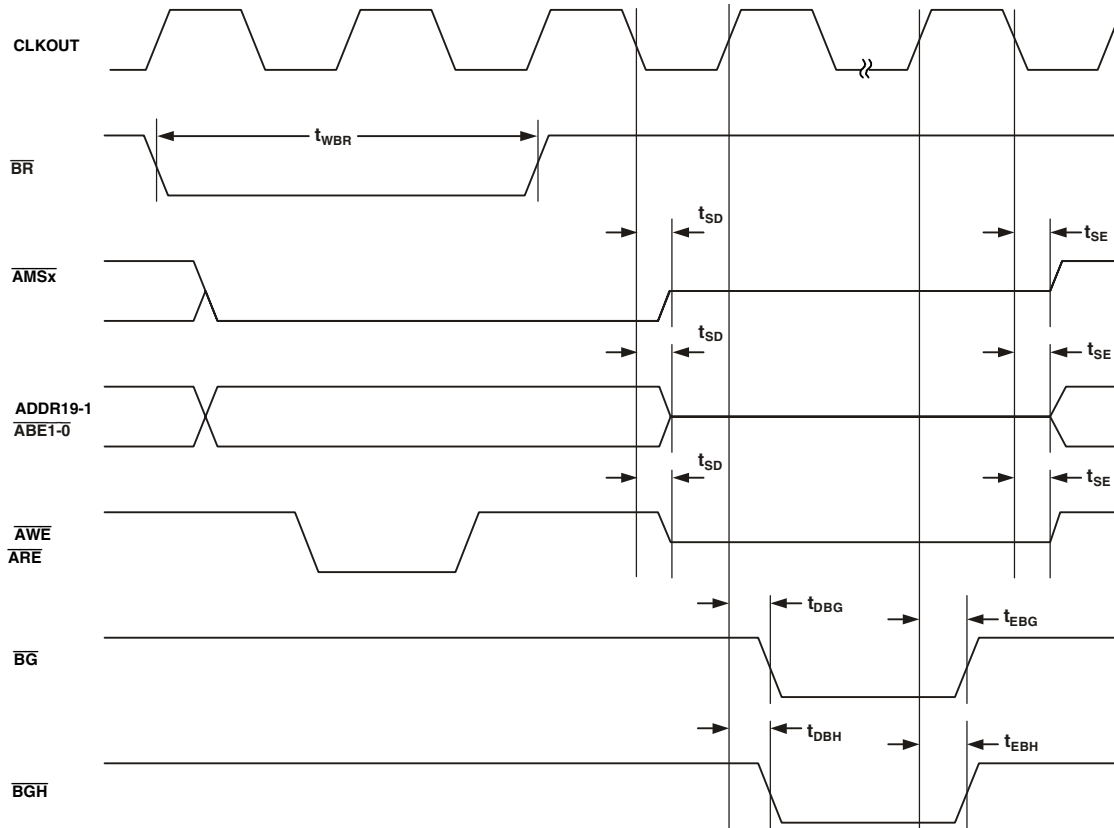


Figure 18. External Port Bus Request and Grant Cycle Timing with Asynchronous \overline{BR}

Parallel Peripheral Interface Timing

Table 24 and Figure 19, Figure 20, Figure 21, and Figure 22 describe Parallel Peripheral Interface operations.

Table 24. Parallel Peripheral Interface Timing

Parameter	Min	Max	Unit
<i>Timing Requirements</i>			
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	3.0		ns
t_{HFSPE} External Frame Sync Hold After PPI_CLK	3.0		ns
t_{SDRPE} Receive Data Setup Before PPI_CLK	2.0		ns
t_{HDRPE} Receive Data Hold After PPI_CLK	4.0		ns
<i>Switching Characteristics — GP Output and Frame Capture Modes</i>			
t_{DFSPE} Internal Frame Sync Delay After PPI_CLK		10.0	ns
$t_{HOFSPPE}$ 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 $f_{SCLK}/2$

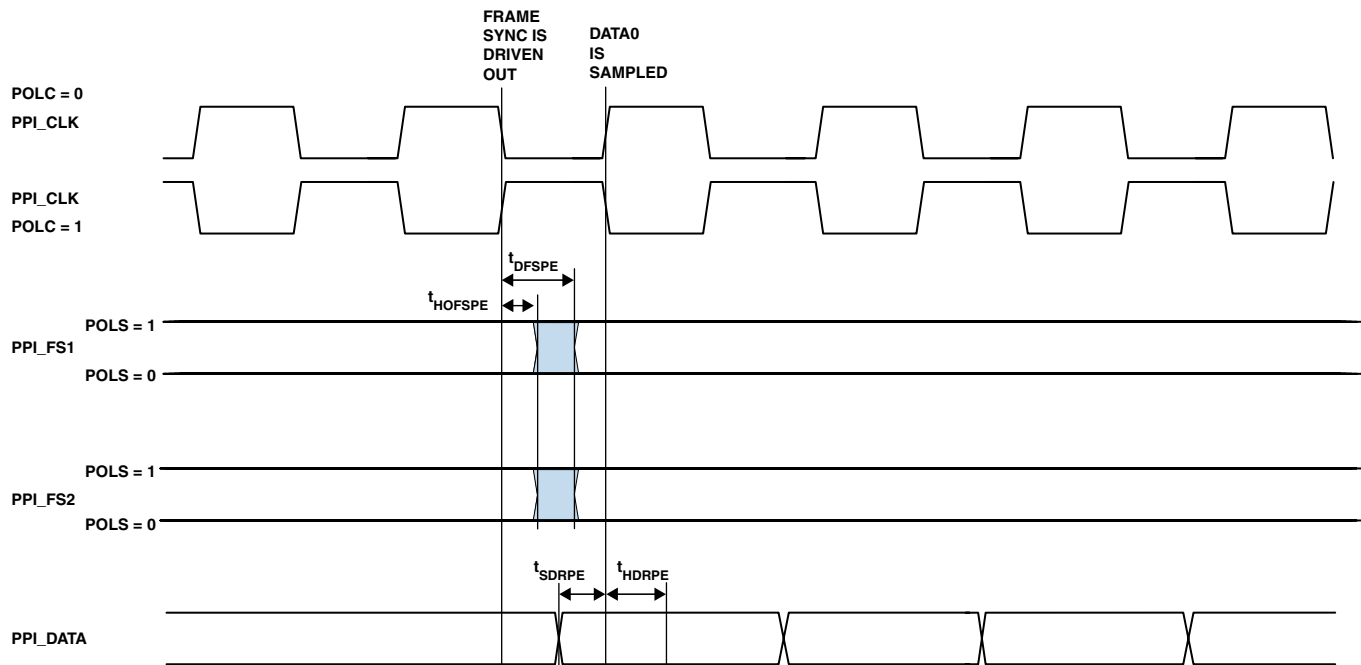


Figure 19. PPI GP Rx Mode with Internal Frame Sync Timing

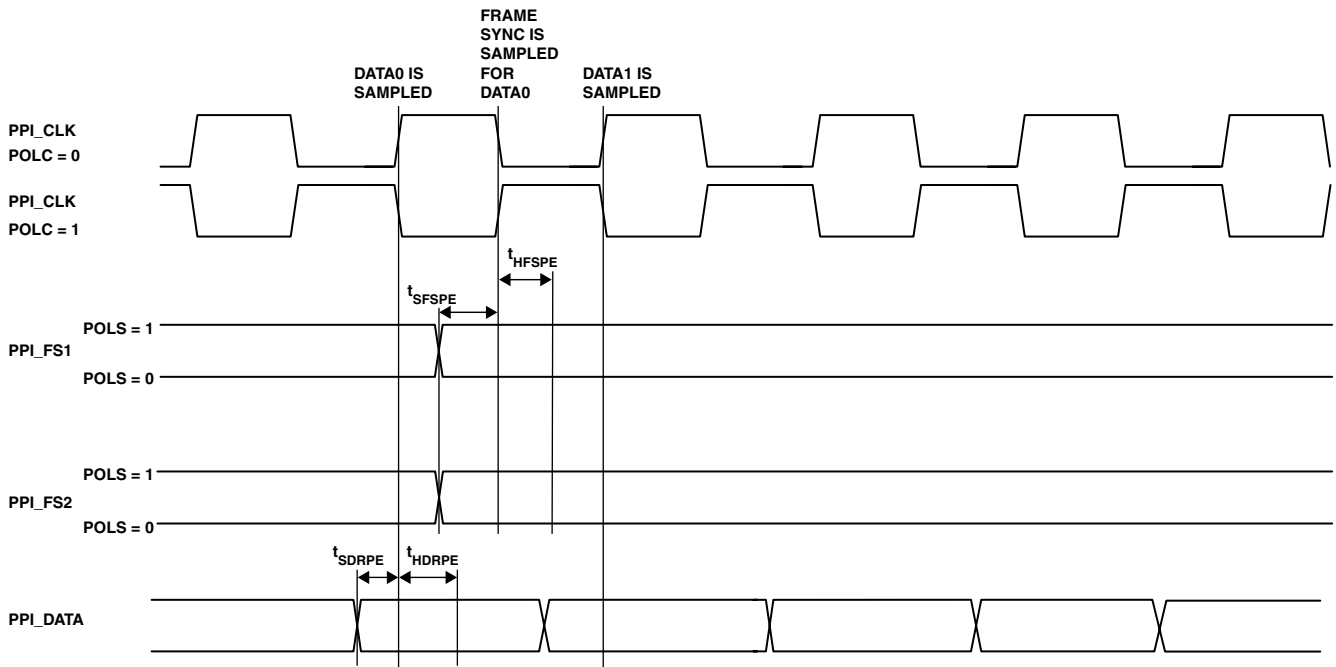


Figure 20. PPI GP Rx Mode with External Frame Sync Timing

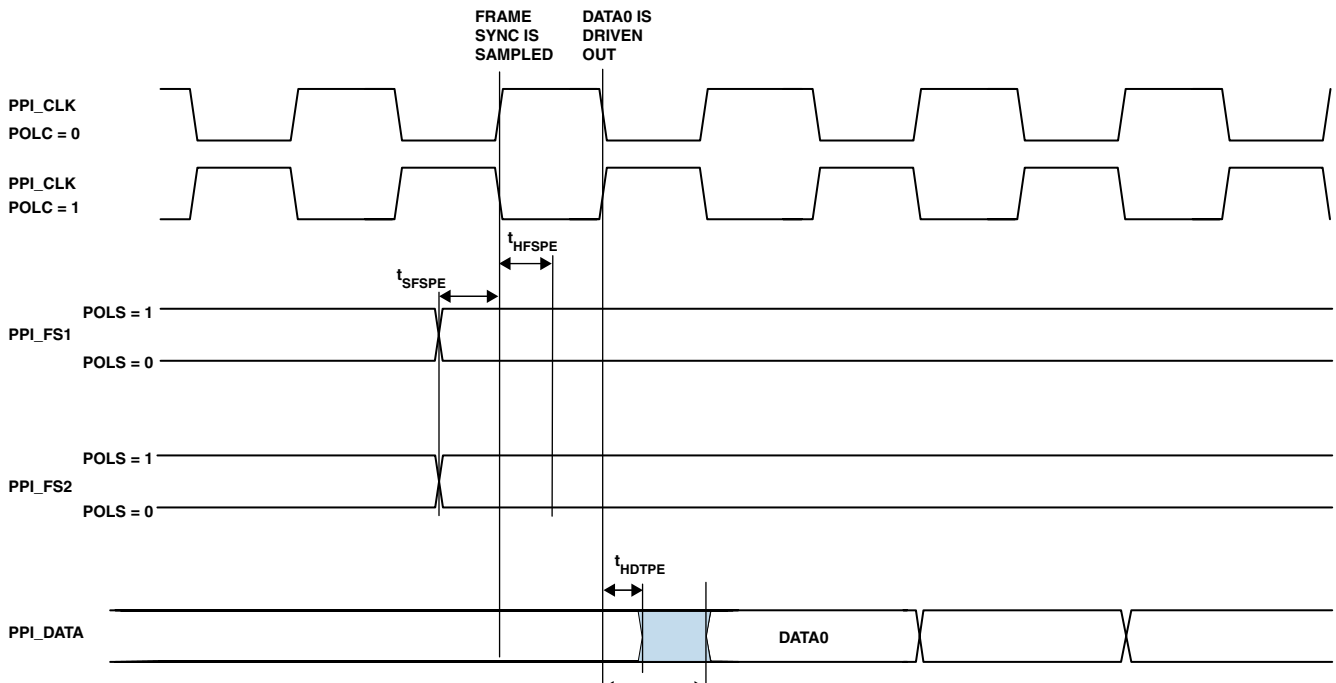


Figure 21. PPI GP Tx Mode with External Frame Sync Timing

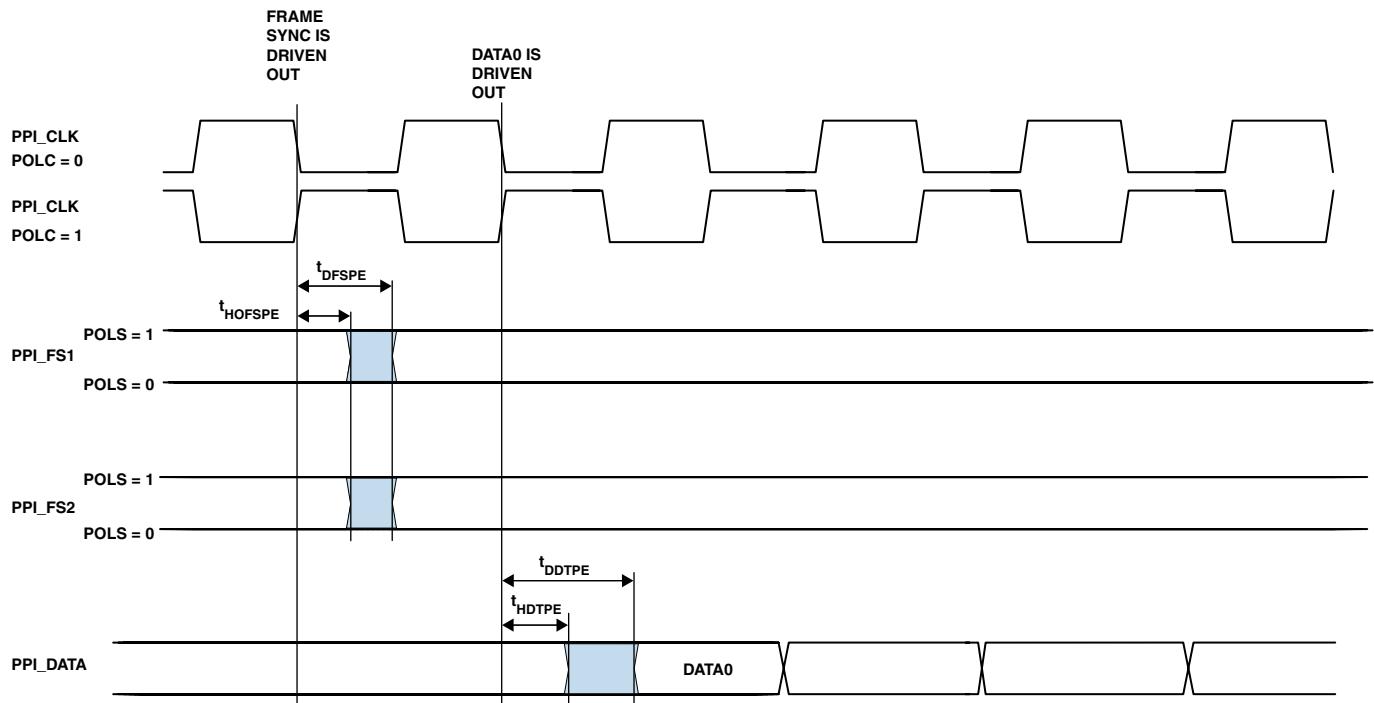


Figure 22. PPI GP Tx Mode with Internal Frame Sync Timing

Serial Ports Timing

Table 25 through Table 28 on Page 44 and Figure 23 on Page 44 through Figure 25 on Page 46 describe Serial Port operations.

Table 25. Serial Ports—External Clock

Parameter	Min	Max	Unit
<i>Timing Requirements</i>			
t_{SFSE} TFS/RFS Setup Before TSCLK/RSCLK (externally generated TFS/RFS) ¹	3.0		ns
t_{HFSE} TFS/RFS Hold After TSCLK/RSCLK (externally generated TFS/RFS) ¹	3.0		ns
t_{SDRE} Receive Data Setup Before RSCLK ¹	3.0		ns
t_{HDRE} Receive Data Hold After RSCLK ¹	3.0		ns
t_{SCLKEW} TSCLK/RSCLK Width	4.5		ns
t_{SCLKE} TSCLK/RSCLK Period	15.0		ns
<i>Switching Characteristics</i>			
t_{DFSE} TFS/RFS Delay After TSCLK/RSCLK (Internally Generated TFS/RFS) ²		10.0	ns
t_{HOFSE} TFS/RFS Hold After TSCLK/RSCLK (Internally Generated TFS/RFS) ²	0.0		ns
t_{DDTE} Transmit Data Delay After TSCLK ²		10.0	ns
t_{HDTE} Transmit Data Hold After TSCLK ²	0.0		ns

¹ Referenced to sample edge.

² Referenced to drive edge.

Table 26. Serial Ports—Internal Clock

Parameter	Min	Max	Unit
<i>Timing Requirements</i>			
t_{SFSI} TFS/RFS Setup Before TSCLK/RSCLK (externally generated TFS/RFS) ¹	8.0		ns
t_{HFSI} TFS/RFS Hold After TSCLK/RSCLK (externally generated TFS/RFS) ¹	-2.0		ns
t_{SDRI} Receive Data Setup Before RSCLK ¹	6.0		ns
t_{HDRI} Receive Data Hold After RSCLK ¹	0.0		ns
t_{SCLKEW} TSCLK/RSCLK Width	4.5		ns
t_{SCLKE} TSCLK/RSCLK Period	15.0		ns
<i>Switching Characteristics</i>			
t_{DFSI} TFS/RFS Delay After TSCLK/RSCLK (Internally Generated TFS/RFS) ²		3.0	ns
t_{HOFSI} TFS/RFS Hold After TSCLK/RSCLK (Internally Generated TFS/RFS) ²	-1.0		ns
t_{DDTI} Transmit Data Delay After TSCLK ²		3.0	ns
t_{HDTI} Transmit Data Hold After TSCLK ²	-2.0		ns
t_{SCLKIW} TSCLK/RSCLK Width	4.5		ns

¹ Referenced to sample edge.

² Referenced to drive edge.

Table 27. Serial Ports—Enable and Three-State

Parameter	Min	Max	Unit
<i>Switching Characteristics</i>			
t_{DTENE} Data Enable Delay from External TSCLK ¹	0		ns
t_{DDTTE} Data Disable Delay from External TSCLK ¹		10.0	ns
t_{DTENI} Data Enable Delay from Internal TSCLK ¹	-2.0		ns
t_{DDTTI} Data Disable Delay from Internal TSCLK ¹		3.0	ns

¹ Referenced to drive edge.

Table 28. External Late Frame Sync

Parameter	Min	Max	Unit
<i>Switching Characteristics</i>			
$t_{DDTLFSE}$		10.0	ns
$t_{DTENLFS}$	0		ns

¹ MCE = 1, TFS enable and TFS valid follow $t_{DTENLFS}$ and $t_{DDTLFSE}$.

² If external RFS/TFS setup to RSCLK/TSCLK > $t_{SCLKE}/2$, then $t_{DDTE/I}$ and $t_{DTENE/I}$ apply; otherwise $t_{DDTLFSE}$ and $t_{DTENLFS}$ apply.

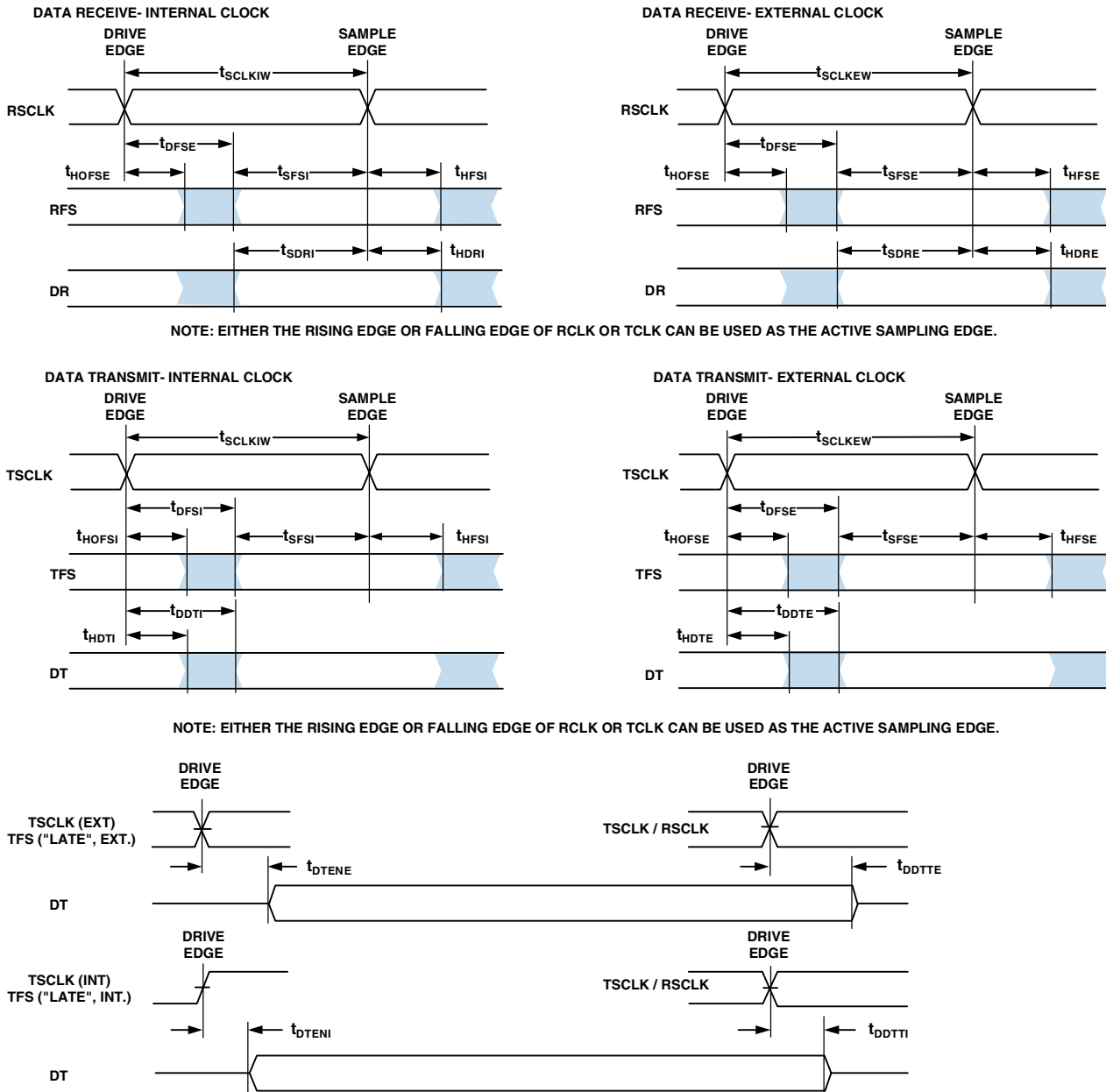
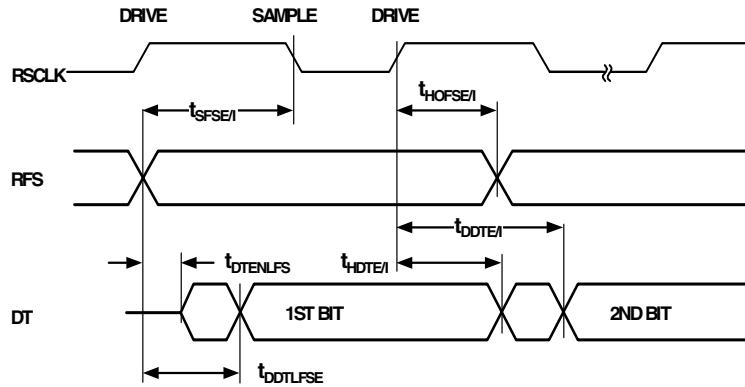


Figure 23. Serial Ports

EXTERNAL RFS WITH MCE = 1, MFD = 0 (INTERNAL OR EXTERNAL CLOCK)



LATE EXTERNAL TFS (INTERNAL OR EXTERNAL CLOCK)

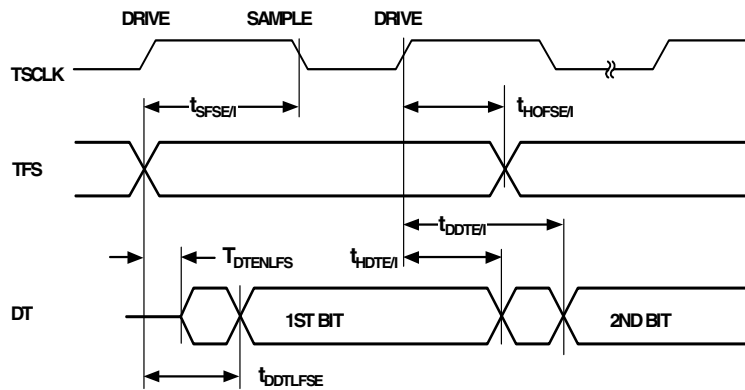
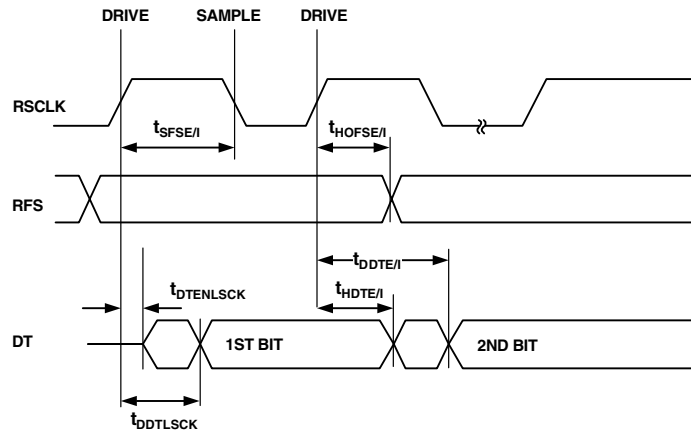


Figure 24. External Late Frame Sync (Frame Sync Setup < $t_{SCLK}/2$)

EXTERNAL RFS WITH MCE = 1, MFD = 0



LATE EXTERNAL TFS

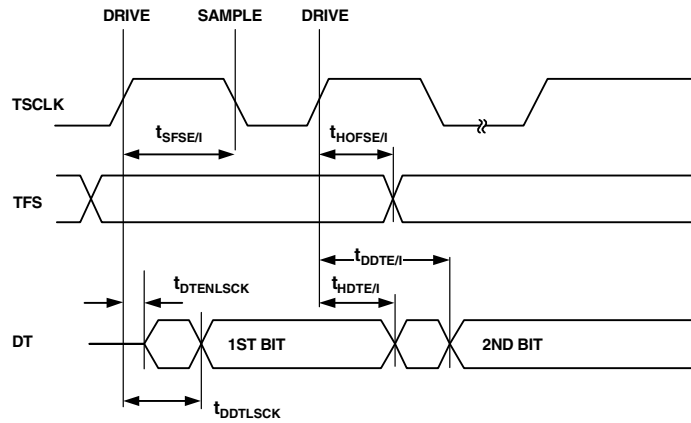


Figure 25. External Late Frame Sync (Frame Sync Setup > $t_{SCLK}/2$)

Serial Peripheral Interface (SPI) Port
—Master Timing

Table 29 and Figure 26 describe SPI port master operations.

Table 29. Serial Peripheral Interface (SPI) Port—Master Timing

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
t_{SSPIDM}	Data Input Valid to SCK Edge (Data Input Setup)	7.5		ns
t_{HSPIDM}	SCK Sampling Edge to Data Input Invalid	-1.5		ns
<i>Switching Characteristics</i>				
t_{SDSCIM}	$\overline{SPIOSELx}$ Low to First SCK edge (x=0 or 1)	$2t_{SCLK} - 1.5$		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	$4t_{SCLK} - 1.5$		ns
t_{HDSM}	Last SCK Edge to $\overline{SPIOSELx}$ High (x=0 or 1)	$2t_{SCLK} - 1.5$		ns
t_{SPITDM}	Sequential Transfer Delay	$2t_{SCLK} - 1.5$		ns
$t_{DDSPIDM}$	SCK Edge to Data Out Valid (Data Out Delay)	0	6	ns
$t_{HDSPIDM}$	SCK Edge to Data Out Invalid (Data Out Hold)	-1.0	4.0	ns

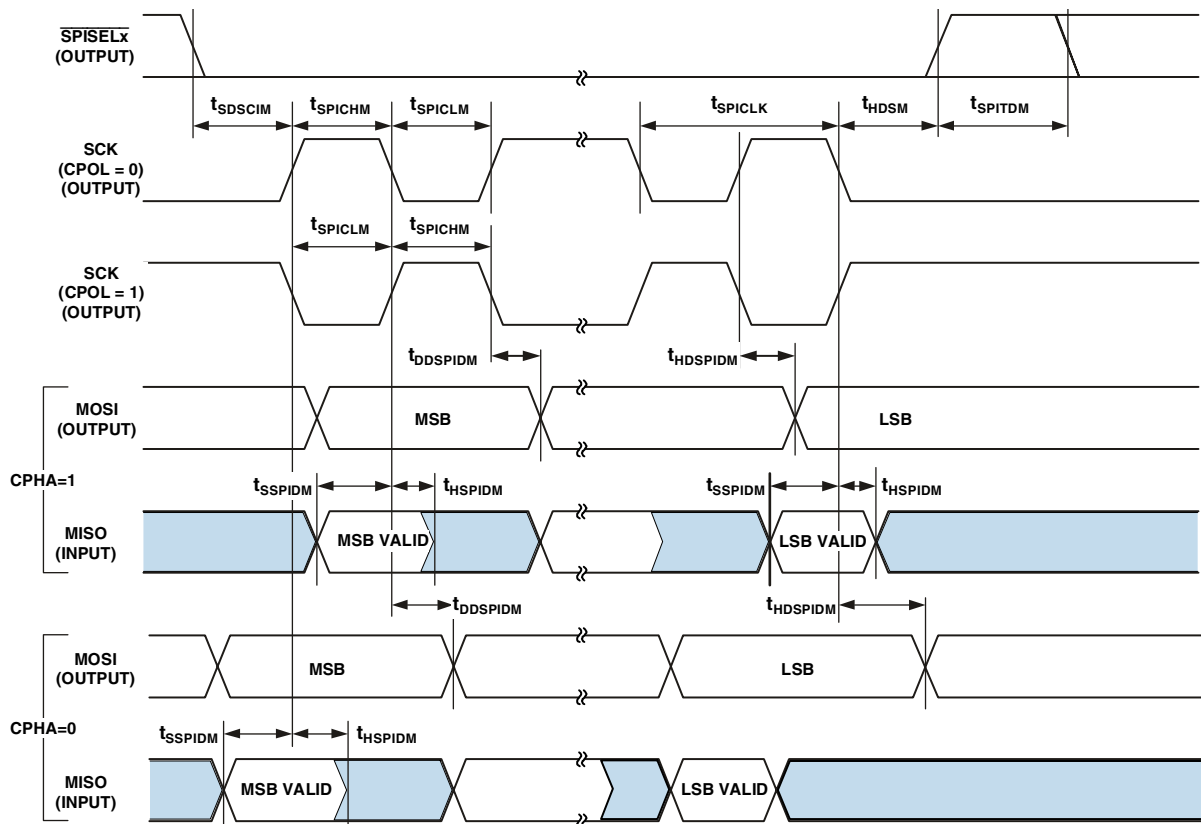


Figure 26. Serial Peripheral Interface (SPI) Port—Master Timing

Serial Peripheral Interface (SPI) Port —Slave Timing

Table 30 and Figure 27 describe SPI port slave operations.

Table 30. Serial Peripheral Interface (SPI) Port—Slave Timing

Parameter		Min	Max	Unit
<i>Timing Requirements</i>				
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	$4t_{SCLK} - 1.5$		ns
t_{HDS}	Last SCK Edge to \overline{SPIOSS} Not Asserted	$2t_{SCLK} - 1.5$		ns
t_{SPITDS}	Sequential Transfer Delay	$2t_{SCLK} - 1.5$		ns
t_{SDSCI}	\overline{SPIOSS} Assertion to First SCK Edge	$2t_{SCLK} - 1.5$		ns
t_{SSPID}	Data Input Valid to SCK Edge (Data Input Setup)	1.6		ns
t_{HSPID}	SCK Sampling Edge to Data Input Invalid	1.6		ns
<i>Switching Characteristics</i>				
t_{DSOE}	\overline{SPIOSS} Assertion to Data Out Active	0	8	ns
t_{DSDHI}	\overline{SPIOSS} Deassertion to Data High impedance	0	8	ns
t_{DDSPID}	SCK Edge to Data Out Valid (Data Out Delay)	0	10	ns
t_{HDSPID}	SCK Edge to Data Out Invalid (Data Out Hold)	0	10	ns

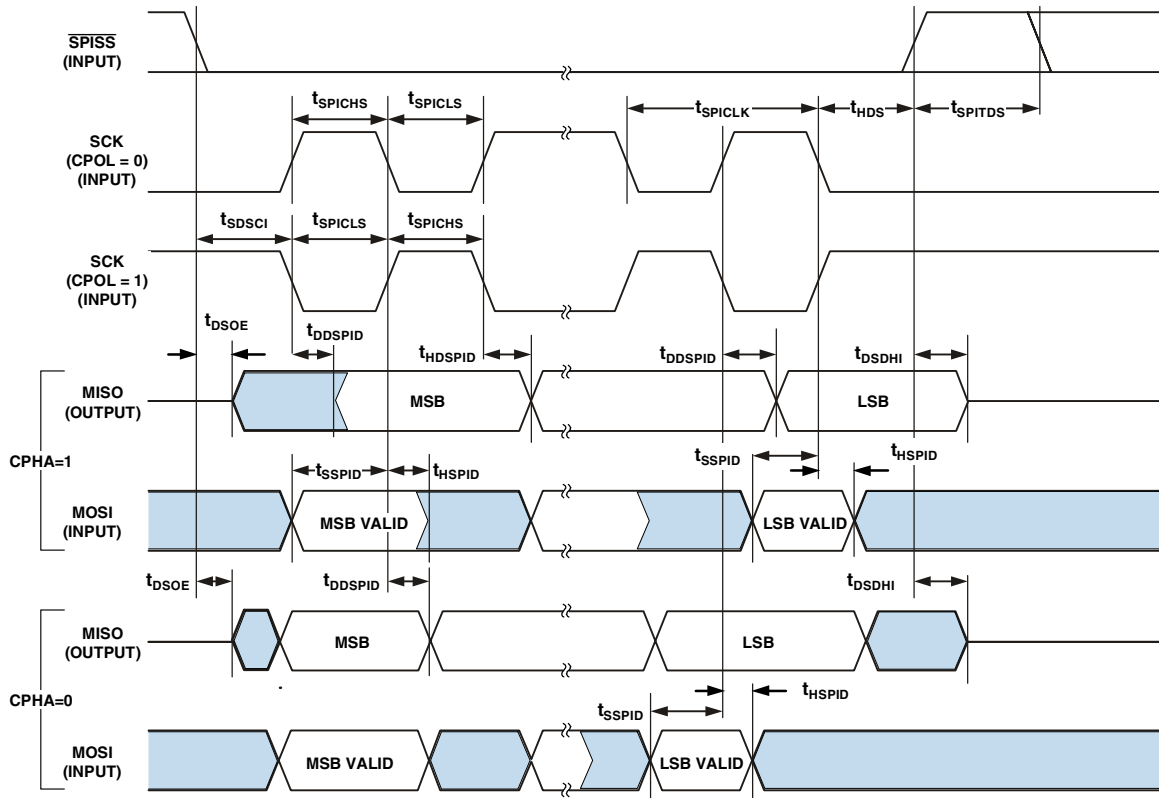


Figure 27. Serial Peripheral Interface (SPI) Port—Slave Timing

**Universal Asynchronous Receiver-Transmitter
(UART) Port Timing**

Figure 28 describes UART port receive and transmit operations. The maximum baud rate is SCLK/16. As shown in Figure 28 there is some latency between the generation internal UART

interrupts and the external data operations. These latencies are negligible at the data transmission rates for the UART.

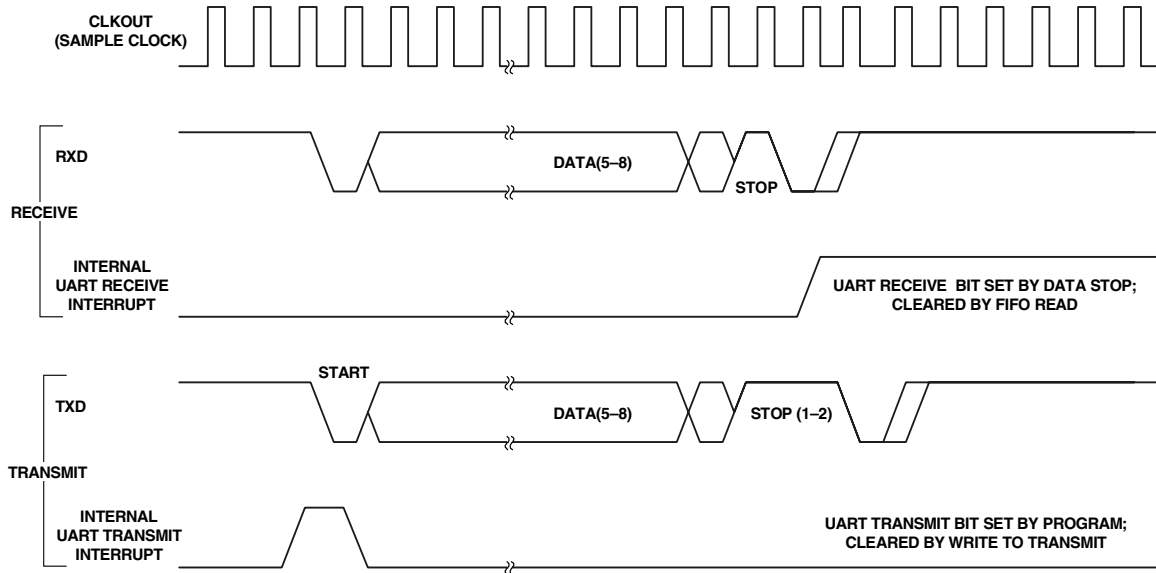


Figure 28. UART Port—Receive and Transmit Timing

Programmable Flags Cycle Timing

Table 31 and Figure 29 describe programmable flag operations.

Table 31. Programmable Flags Cycle Timing

Parameter	Minimum	Maximum	Unit
<i>Timing Requirement</i>			
t_{WFI} Flag input pulsewidth	$t_{SCLK} + 1$		ns
<i>Switching Characteristic</i>			
t_{DFO} Flag output delay from CLKOUT low		6	ns

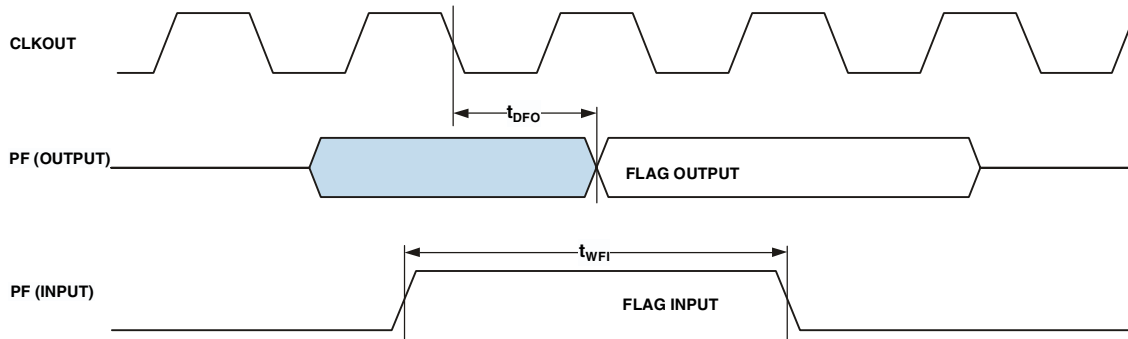


Figure 29. Programmable Flags Cycle Timing

Timer Cycle Timing

Table 32 and Figure 30 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 32. Timer Cycle Timing

Parameter	Minimum	Maximum	Unit
<i>Timing Characteristics</i>			
t_{WL} Timer Pulsewidth Input Low ¹ (measured in SCLK cycles)	1		SCLK
t_{WH} Timer Pulsewidth Input High ¹ (measured in SCLK cycles)	1		SCLK
<i>Switching Characteristic</i>			
t_{HTO} Timer Pulsewidth Output ² (measured in SCLK cycles)	1	$(2^{32}-1)$	SCLK

¹ The minimum pulsewidths 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.

² The minimum time for t_{HTO} is one cycle, and the maximum time for t_{HTO} equals $(2^{32}-1)$ cycles.

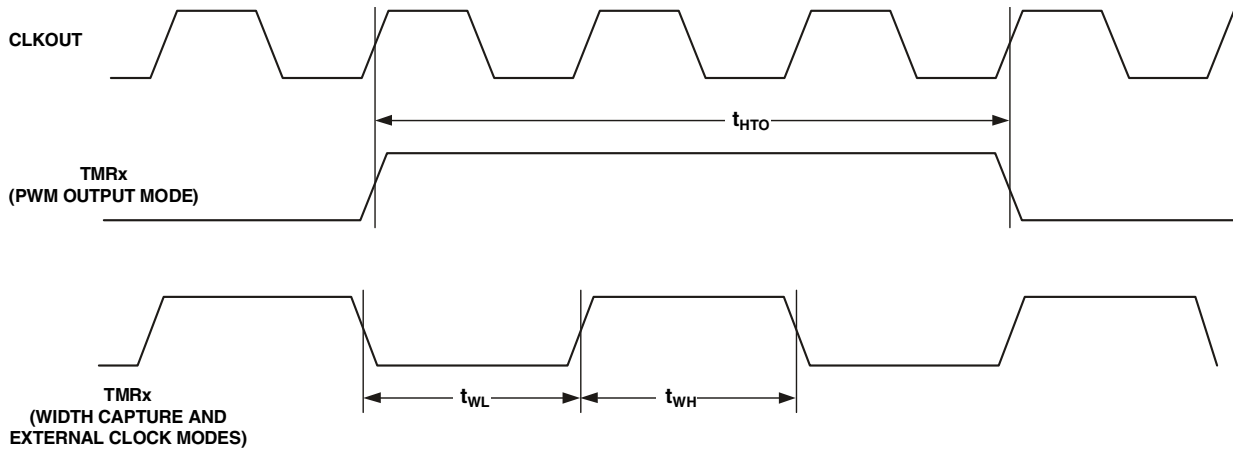


Figure 30. Timer PWM_OUT Cycle Timing

JTAG Test And Emulation Port Timing

Table 33 and Figure 31 describe JTAG port operations.

Table 33. JTAG Port Timing

Parameter	Minimum	Maximum	Unit
<i>Timing Requirements</i>			
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{TRST} Pulsewidth ² (measured in TCK cycles)	4		TCK
<i>Switching Characteristics</i>			
t_{DIDO} 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{BR} , DATA15-0, DR0PRI, DR0SEC, DR1PRI, DR1SEC, MISO0, MOSI0, NMI, PF15-0, PPI_CLK, PPI3-0, SCL0, SCL1, SDA0, SDA1, SCK, MTXON, MRXON, MMCLK, MBCLK, MFS, MTX, MRX, SCK1, MISO1, MOSI1, SPI1SS, SPI1SEL, SCK2, MISO2, MOSI2, SPI2SS, SPI2SEL, RX1, TX1, RX2, TX2, DT2PRI, DT2SEC, TSCLK2, DR2PRI, DT2SEC, RSCLK2, RFS2, TFS2, DT3PRI, DT3SEC, DR3PRI, DR3SEC, RSCLK3, RFS3, TFS3, CANTX, CANRX, RESET, RFS0, RFS1, RSCLK0, RSCLK1, TSCLK0, TSCLK1, RX0, SCK0, TFS0, TFS1, and TMR2-0,

² 50 MHz Maximum

³ System Outputs = AMS, AO \overline{E} , ARE, AWE, AB \overline{E} , BG, DATA15-0, DT0PRI, DT0SEC, DT1PRI, DT1SEC, MISO0, MOSI0, PF15-0, PPI3-0, MTXON, MRXON, MMCLK, MBCLK, MFS, MTX, MRX, SCK1, MISO1, MOSI1, SPI1SS, SPI1SEL, SCK2, MISO2, MOSI2, SPI2SS, SPI2SEL, RX1, TX1, RX2, TX2, DT2PRI, DT2SEC, TSCLK2, DR2PRI, DR2SEC, RSCLK2, RFS2, TFS2, DT3PRI, DT3SEC, TSCLK3, DR3PRI, DR3SEC, RSCLK3, RFS3, TFS3, CANTX, CANRX, RFS0, RFS1, RSCLK0, RSCLK1, TSCLK0, TSCLK1, CLKOUT, TX0, SA10, SCAS, SCK0, SCKE, SMS, SRAS, SWE, and TMR2-0.

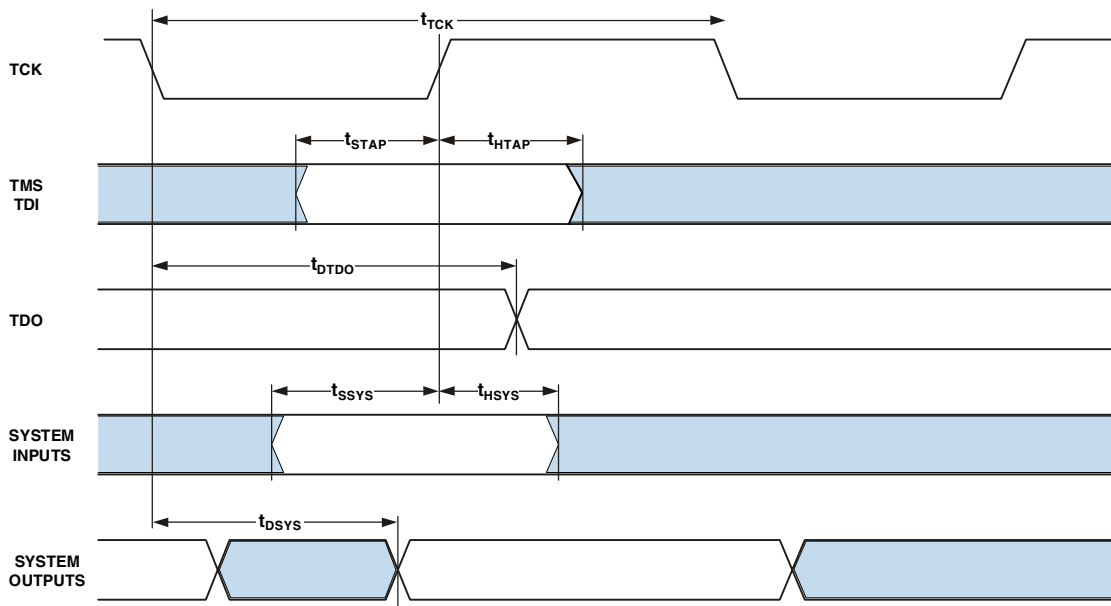


Figure 31. JTAG Port Timing

TWI Controller Timing

Table 34 through Table 41 and Figure 32 through Figure 35 describe the TWI Controller operations.

Table 34. TWI Controller Timing: Bus Start/Stop Bits, Slave Mode, 100 kHz

Parameter		Min	Max	Unit
$t_{SU:STA}$	Start condition setup time	4.7		μ s
$t_{HD:STA}$	Start condition hold time	4.0		μ s
$t_{SU:STO}$	Stop condition setup time	4.0		μ s
t_{BUF}	Bus free time between STOP and START condition	4.7		μ s

Table 35. TWI Controller Timing: Bus Data Requirements, Slave Mode, 100 kHz

Parameter		Min	Max	Unit
t_{HIGH}	Clock high time	4.0		μ s
t_{LOW}	Clock low time	4.7		μ s
t_R	SDA and SCL rise time		1000	ns
t_F	SDA and SCL fall time		300	ns
$t_{HD:DAT}$	Data input hold time	0		ns
$t_{SU:DAT}$	Data input setup time ¹	250		ns
C_B	Bus capacitive loading		400	pF

¹ As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. TBD ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

Table 36. TWI Controller Timing: Bus Start/Stop Bits, Slave Mode, 400 kHz

Parameter		Min	Max	Unit
$t_{SU:STA}$	Start condition setup time	0.6		μ s
$t_{HD:STA}$	Start condition hold time	0.6		μ s
$t_{SU:STO}$	Stop condition setup time	0.6		μ s
t_{BUF}	Bus free time between STOP and START condition	1.3		μ s

Table 37. TWI Controller Timing: Bus Data Requirements, Slave Mode, 400 kHz

Parameter		Min	Max	Unit
t_{HIGH}	Clock high time	0.6		μ s
t_{LOW}	Clock low time	1.3		μ s
t_R	SDA and SCL rise time	TBD	300	ns
t_F	SDA and SCL fall time	TBD	300	ns
$t_{HD:DAT}$	Data input hold time	0	0.9	μ s
$t_{SU:DAT}$	Data input setup time ¹	100		ns
C_B	Bus capacitive loading		400	pF

¹ As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. TBD ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

Table 38. TWI Controller Timing: Bus Start/Stop Bits, Master Mode, 100 kHz

Parameter		Min	Max	Unit
t _{SU:STA}	Start condition setup time	4.7		μs
t _{HD:STA}	Start condition hold time	4.0		μs
t _{SU:STO}	Stop condition setup time	4.0		μs
t _{BUF}	Bus free time between STOP and START condition	4.7		μs

Table 39. TWI Controller Timing: Bus Data Requirements, Master Mode, 100 kHz

Parameter		Min	Max	Unit
t _{HIGH}	Clock high time	4.0		μs
t _{LOW}	Clock low time	4.7		μs
t _R	SDA and SCL rise time		1000	ns
t _F	SDA and SCL fall time		300	ns
t _{HD:DAT}	Data input hold time	0		ns
t _{SU:DAT}	Data input setup time ¹	250		ns
C _B	Bus capacitive loading		400	pF

¹ A Fast mode I²C bus device can be used in a Standard mode I²C bus system, but the requirement T_{SU:DAT} ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line. Before the SCL line is released, T_R max. + T_{SU:DAT} = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification).

Table 40. TWI Controller Timing: Bus Start/Stop Bits, Master Mode, 400 kHz

Parameter		Min	Max	Unit
t _{SU:STA}	Start condition setup time	0.6		μs
t _{HD:STA}	Start condition hold time	0.6		μs
t _{SU:STO}	Stop condition setup time	0.6		μs
t _{BUF}	Bus free time between STOP and START condition	1.3		μs

Table 41. TWI Controller Timing: Bus Data Requirements, Master Mode, 400 kHz

Parameter		Min	Max	Unit
t _{HIGH}	Clock high time	0.6		μs
t _{LOW}	Clock low time	1.3		μs
t _R	SDA and SCL rise time	TBD	300	ns
t _F	SDA and SCL fall time	TBD	300	ns
t _{HD:DAT}	Data input hold time	0	0.9	μs
t _{SU:DAT}	Data input setup time ¹	100		ns
C _B	Bus capacitive loading		400	pF

¹ A Fast mode I²C bus device can be used in a Standard mode I²C bus system, but the requirement T_{SU:DAT} ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line. Before the SCL line is released, T_R max. + T_{SU:DAT} = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification).

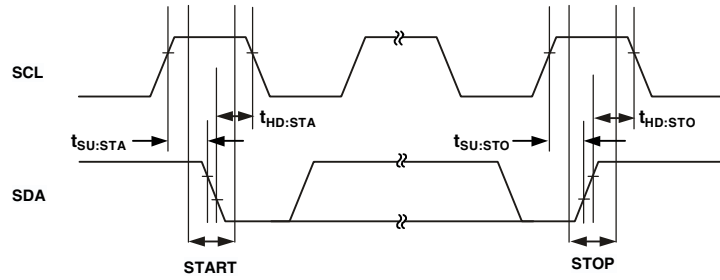


Figure 32. TWI Controller Timing: Bus Start/Stop Bits, Slave Mode

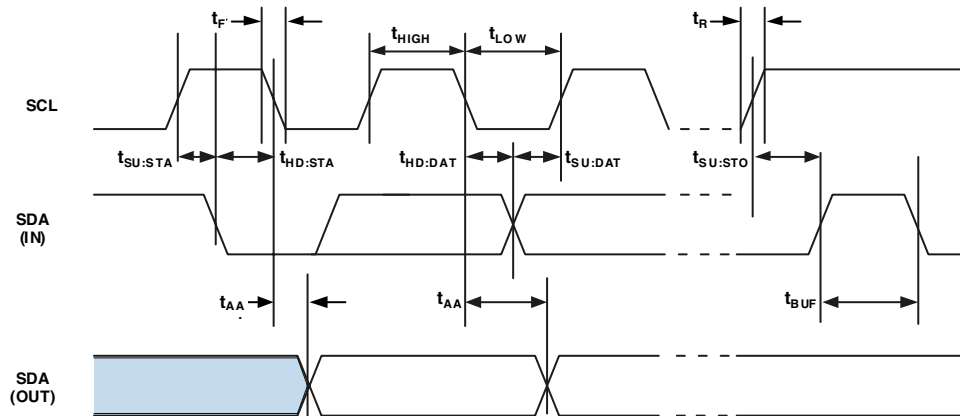


Figure 33. TWI Controller Timing: Bus Data, Slave Mode

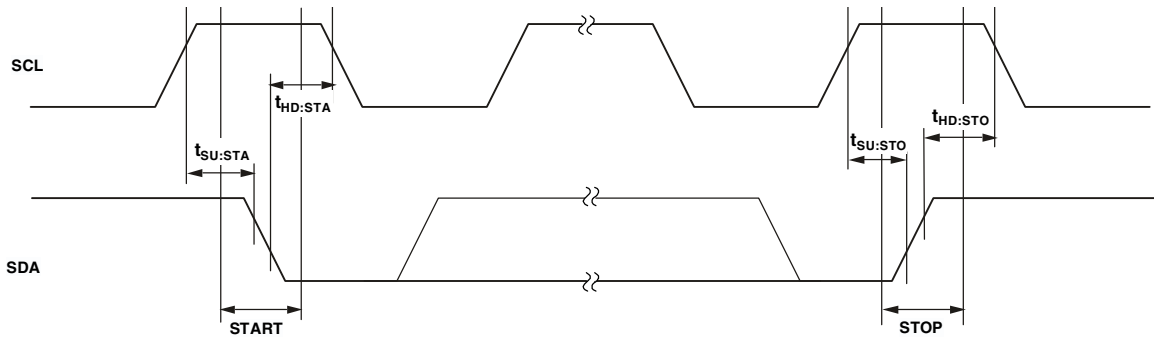


Figure 34. TWI Controller Timing: Bus Start/Stop Bits, Master Mode

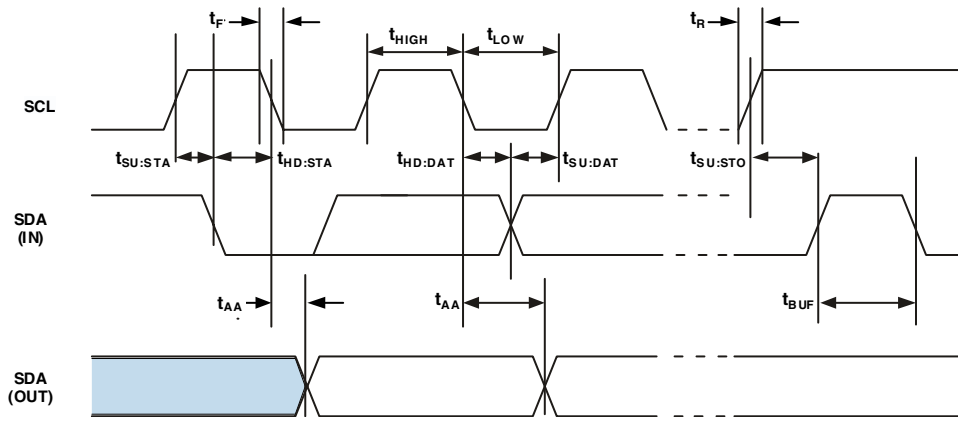


Figure 35. TWI Controller Timing: Bus Data, Master Mode

MXVR Timing

Table 42 through Table 43 and Figure 36 describe the MXVR operations.

Table 42. MXVR Timing—Input Clock Requirements

Parameter	Min	Max	Unit
<i>Timing Requirements</i>			
t_{CKIN} MXI Period ¹	20		ns
t_{CKINL} MXI Low Pulse	8		ns
t_{CKINH} MXI High Pulse	8		ns

¹ For $F_S=48$ KHz, f_{MXI} should be 49.152 MHz. For $F_S=44.1$ KHz, f_{MXI} should be 45.1584 MHz. For $F_S=38$ KHz, f_{MXI} should be 38.912 MHz.

Table 43. MXVR Timing—Rise and Fall Time

Parameter	Min	Max	Unit
t_R MTX/MMCLK/MBCLK/MFS/ \overline{MTXON} Rise Time	-	TBD	ns
t_F MTX/MMCLK/MBCLK/MFS/ \overline{MTXON} Fall Time	-	TBD	ns

TBD

Figure 36. MXVR Timing

CAN Timing

Table 44 and Figure 37 describe the CAN operations.

Table 44. CAN Timing—Rise and Fall Time

Parameter		Minimum	Maximum	Unit
t_R	CANTX rise time	-	TBD	ns
t_F	CANTX fall time	-	TBD	ns

TBD

Figure 37. CAN Timing

OUTPUT DRIVE CURRENTS

Figure 38 shows 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.

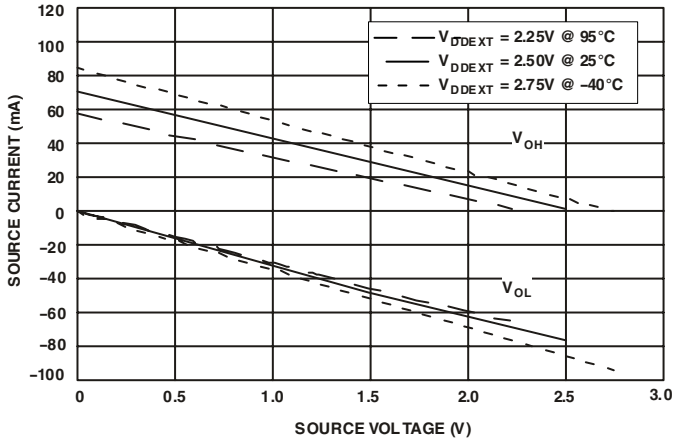


Figure 38. Drive Current A (Low V_{DDEXT})

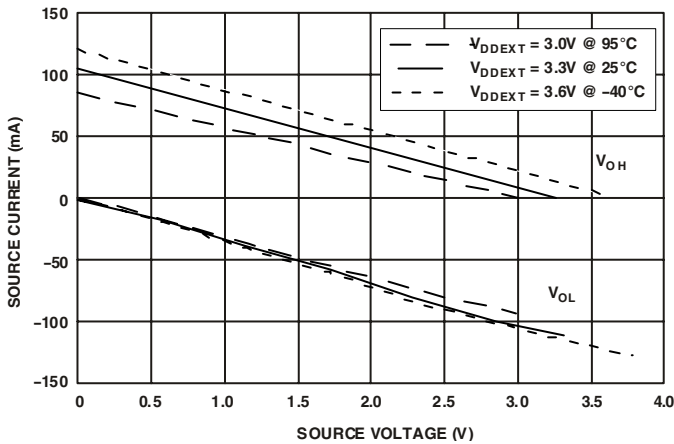


Figure 39. Drive Current A (High V_{DDEXT})

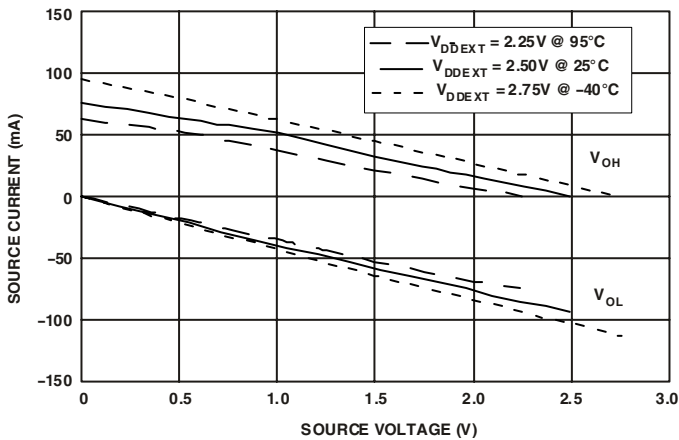


Figure 40. Drive Current B (Low V_{DDEXT})

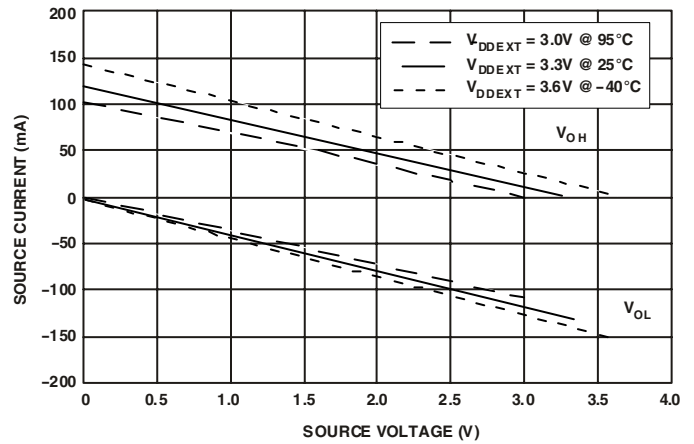


Figure 41. Drive Current B (High V_{DDEXT})

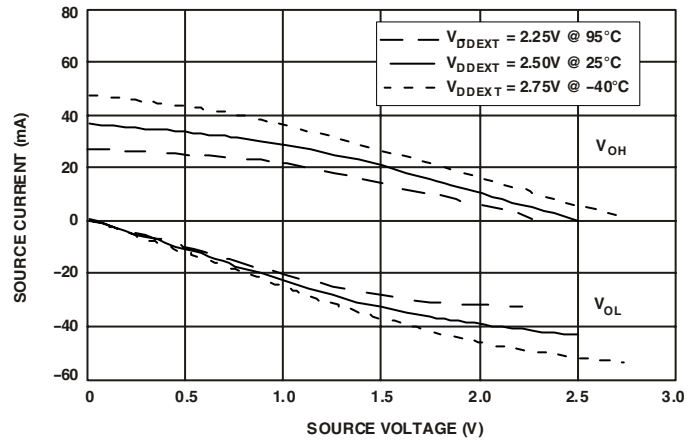


Figure 42. Drive Current C (Low V_{DDEXT})

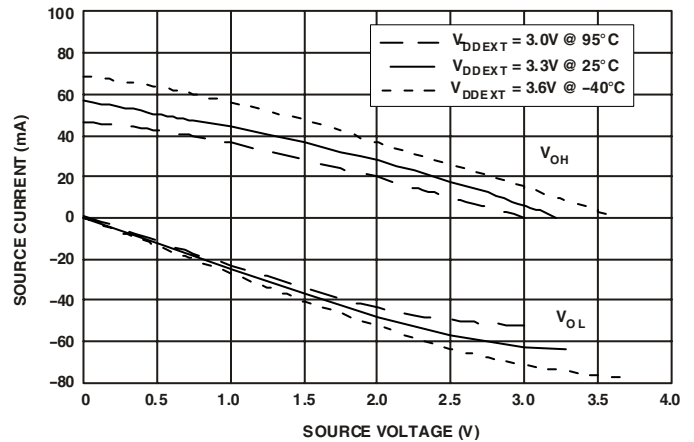


Figure 43. Drive Current C (High V_{DDEXT})

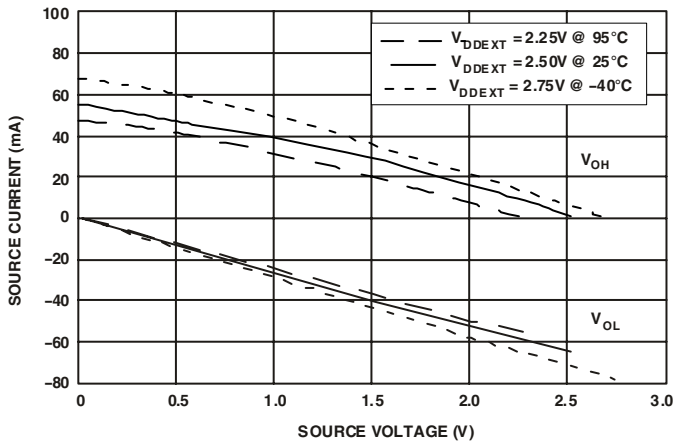


Figure 44. Drive Current D (Low V_{DDEXT})

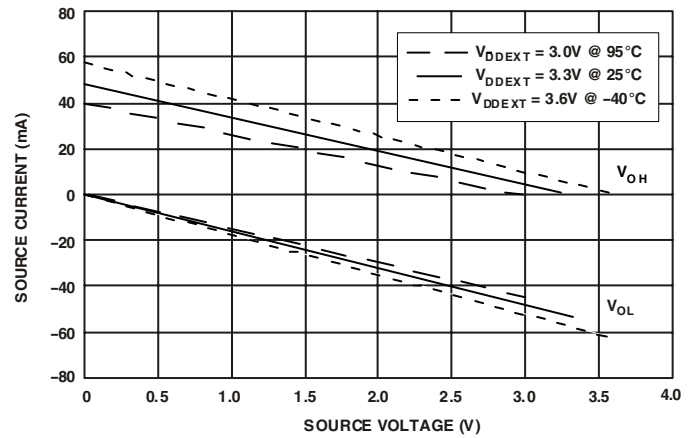


Figure 47. Drive Current E (High V_{DDEXT})

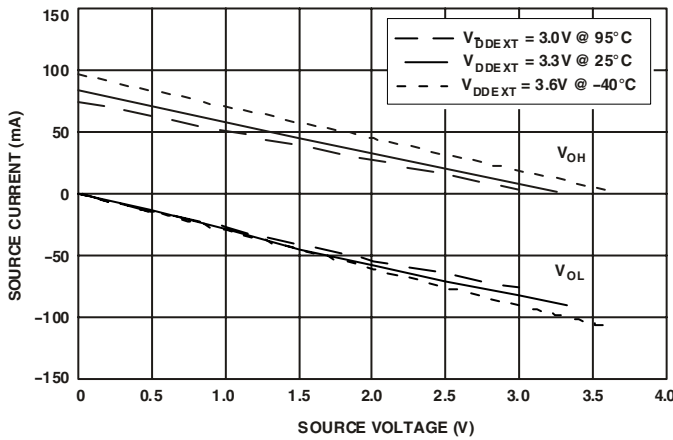


Figure 45. Drive Current D (High V_{DDEXT})

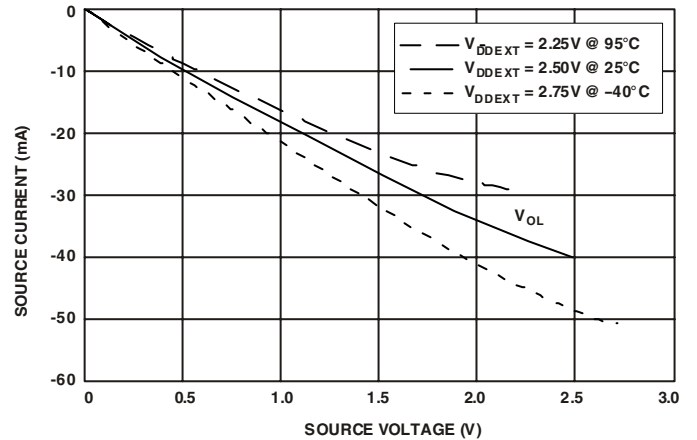


Figure 48. Drive Current F (Low V_{DDEXT})

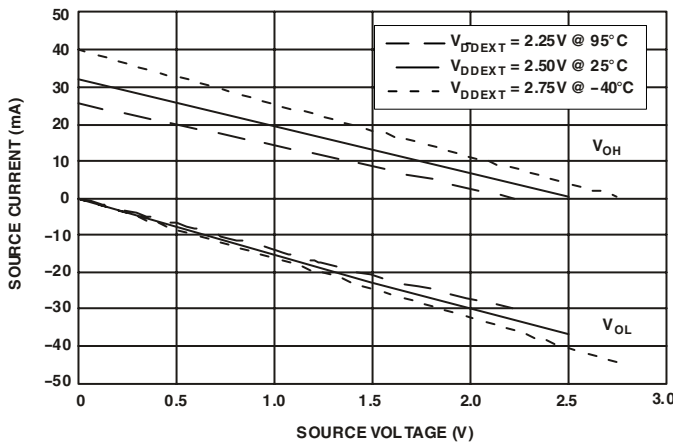


Figure 46. Drive Current E (Low V_{DDEXT})

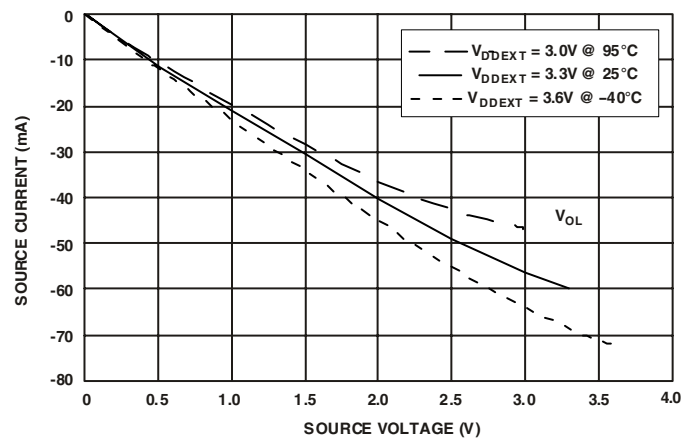


Figure 49. Drive Current F (High V_{DDEXT})

POWER DISSIPATION

Total power dissipation has two components: one due to internal circuitry (P_{INT}) and one due to the switching of external output drivers (P_{EXT}). Table 45 through Table 47 show the power dissipation for internal circuitry (V_{DDINT}).

See the *ADSP-BF53x Blackfin Processor Hardware Reference Manual* for definitions of the various operating -modes and for instructions on how to minimize system power.

Many operating conditions can affect power dissipation. System designers should refer to *EE-229: Estimating Power for ADSP-BF531/ADSP-BF532 Blackfin Processors* on the Analog Devices website (www.analog.com)—use site search on “EE-229.” This document provides detailed information for optimizing your design for lowest power.

Table 45. Internal Power Dissipation (Hibernate mode)

	I_{DD} (nominal) ¹	Unit
$I_{DDHIBERNATE}$ ²	50	μA
I_{DDRTC} ³	20	μA

¹ Nominal assumes an operating temperature of 25°C.
² Measured at $V_{DDEXT} = 3.65$ V with voltage regulator off ($V_{DDINT} = 0$ V).
³ Measured at $V_{DDRTC} = 3.3$ V at 25°C.

Table 46. Internal Power Dissipation (Deep Sleep mode)

V_{DDINT} ¹	I_{DD} (nominal) ²	Unit
0.95	28.00	mA
1.00	32.00	mA
1.10	40.00	mA
1.26	54.00	mA

¹ Assumes V_{DDINT} is regulated externally.
² Nominal assumes an operating temperature of 25°C.

Table 47. Internal Power Dissipation (Full On¹ mode)

V_{DDINT} ² @ f_{CLK} (MHz)	I_{DD} (nominal) ³	Unit
0.95 @ 50 MHz	46.00	mA
0.95 @ 250 MHz	98.00	mA
0.95 @ 300 MHz	110.00	mA
1.00 @ 350 MHz	132.00	mA
1.10 @ 444 MHz	180.00	mA
1.26 @ 500 MHz	235.00	mA

¹ Processor executing 75% dual MAC, 25% ADD with moderate data bus activity.
² Assumes V_{DDINT} is regulated externally.
³ Nominal assumes an operating temperature of 25°C.

TEST CONDITIONS

All timing parameters appearing in this data sheet were measured under the conditions described in this section. Figure 50 shows the measurement point for AC measurements (except output enable/disable). The measurement point V_{MEAS} is 1.5 V for V_{DDEXT} (nominal) = 2.5/3.3 V.

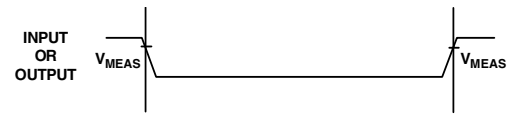


Figure 50. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

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 51, “Output Enable/Disable,” on page 62.

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) = 2.5/3.3 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 51.

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

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) = 2.5/3.3 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's output voltage and the input threshold for the device requiring the hold time. C_L is the total bus capacitance (per data line), and I_L is the total leakage or three-state current (per data line). The hold time will be t_{DECAY} plus the various output disable times as specified in the [Timing Specifications on Page 29](#) (for example t_{DSDAT} for an SDRAM write cycle as shown in [Table 21 on Page 36](#)).

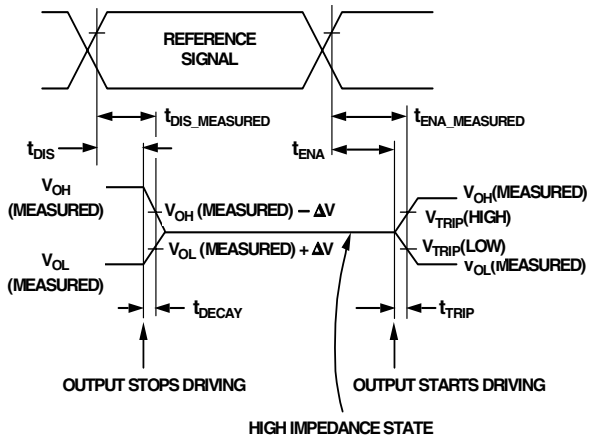


Figure 51. Output Enable/Disable

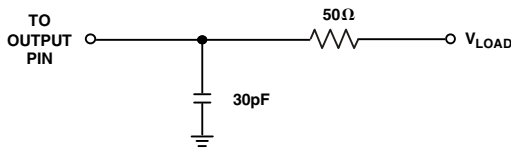


Figure 52. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

Capacitive Loading

Output delays and holds are based on standard capacitive loads: 30 pF on all pins (see [Figure 52](#)). V_{LOAD} is 1.5 V for V_{DDEXT} (nominal) = 2.5/3.3 V. [Figure 53 on Page 62](#) through [Figure 60 on Page 63](#) show how output rise time varies with capacitance.

The delay and hold specifications given should be derated by a factor derived from these figures. The graphs in these figures may not be linear outside the ranges shown.

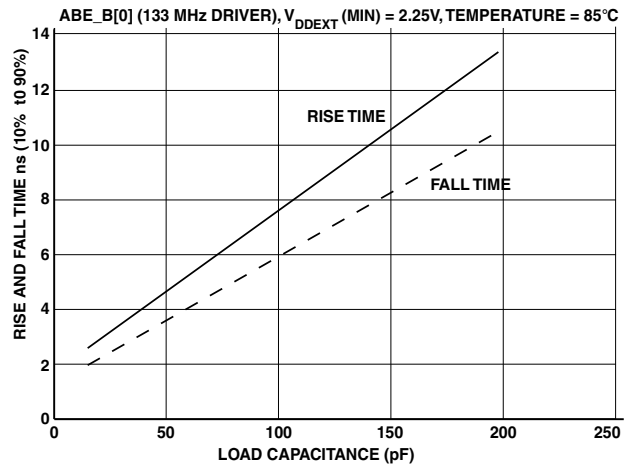


Figure 53. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver A at V_{DDEXT} (min)

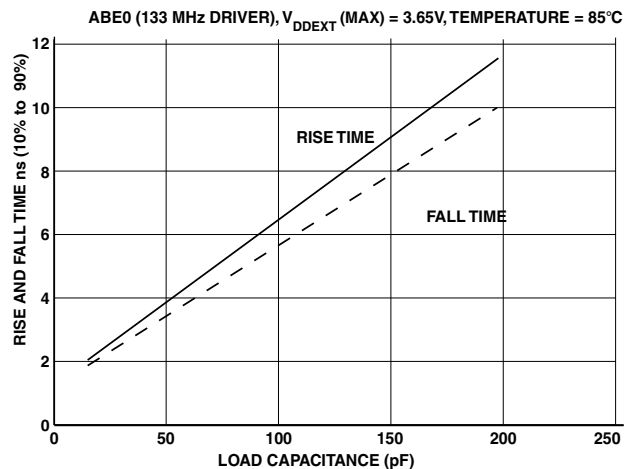


Figure 54. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver A at V_{DDEXT} (max)

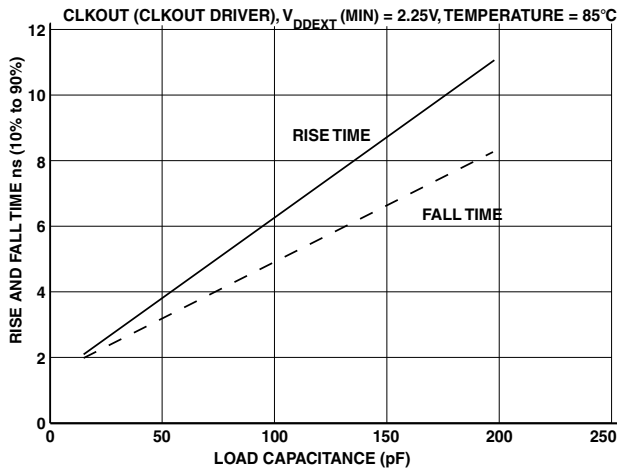


Figure 55. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver B at $V_{DDEXT}(\text{min})$

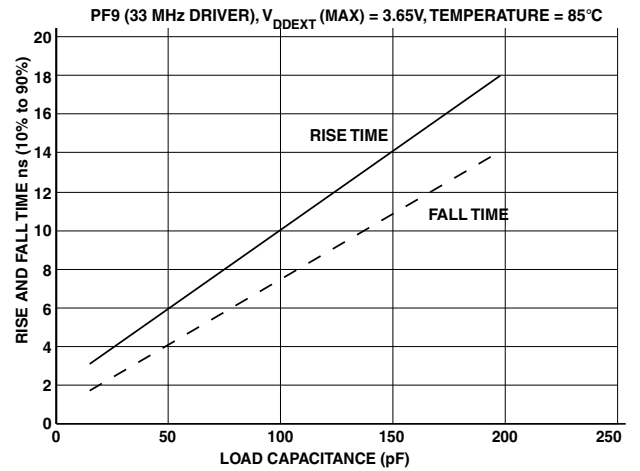


Figure 58. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver C at $V_{DDEXT}(\text{max})$

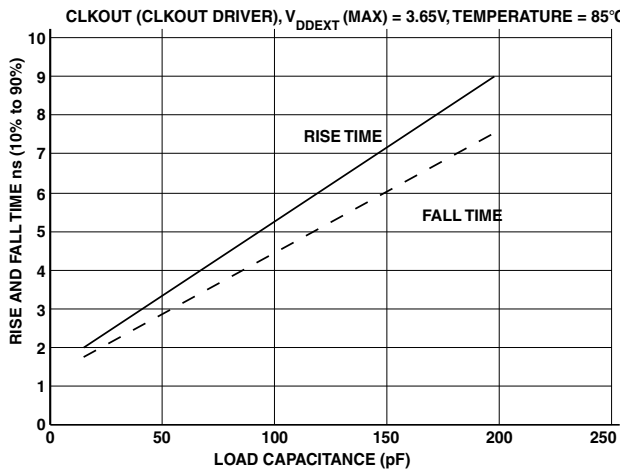


Figure 56. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver B at $V_{DDEXT}(\text{max})$

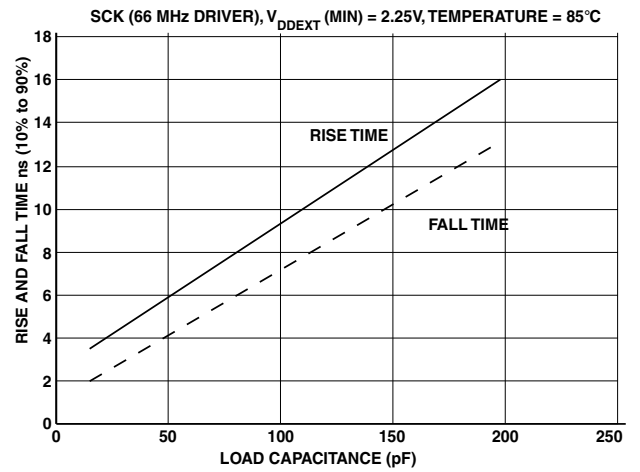


Figure 59. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver D at $V_{DDEXT}(\text{min})$

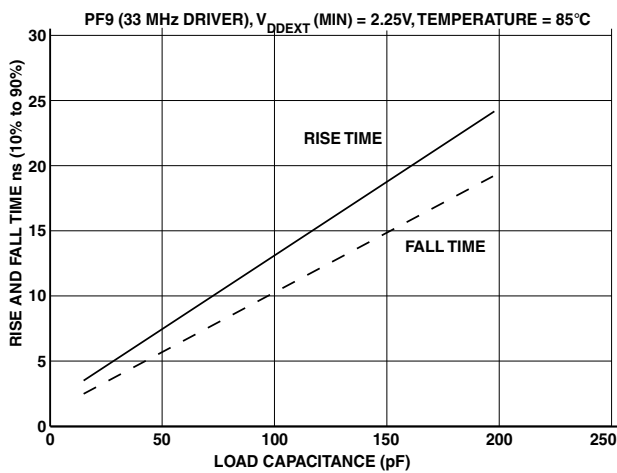


Figure 57. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver C at $V_{DDEXT}(\text{min})$

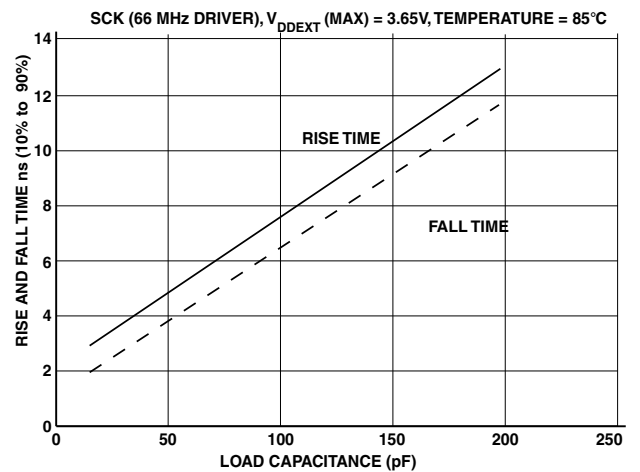


Figure 60. Typical Rise and Fall Times (10%–90%) versus Load Capacitance for Driver D at $V_{DDEXT}(\text{max})$

ENVIRONMENTAL CONDITIONS

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

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

where:

T_J = Junction temperature (°C)

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

Ψ_{JT} = From [Table 48](#) or [Table 49](#)

P_D = Power dissipation (see [Power Dissipation on Page 61](#) 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 48](#) and [Table 49](#), 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 48. 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 49. 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 MINI-BGA PINOUT

Figure 61 lists the top view of the mini-BGA pin configuration. Figure 62 lists the bottom view of the mini-BGA pin configuration.

Table 50 on Page 66 lists the mini-BGA pinout by pin number. Table 51 on Page 67 lists the mini-BGA pinout by signal.

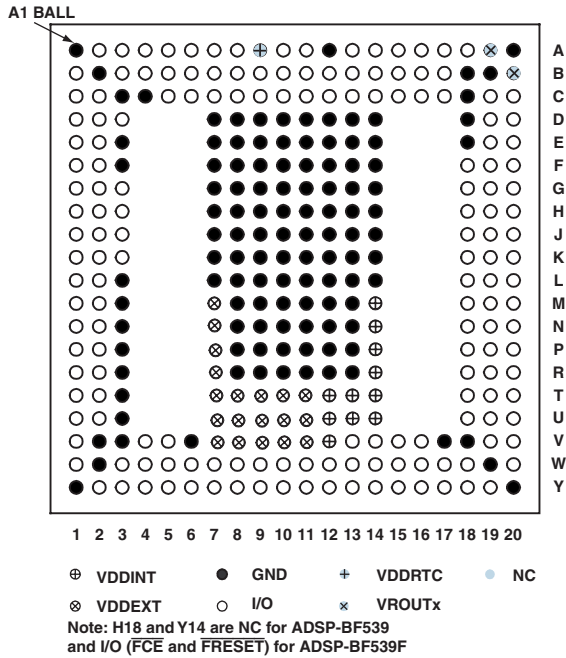


Figure 61. 316-Ball Mini-BGA Pin Configuration (Top View)

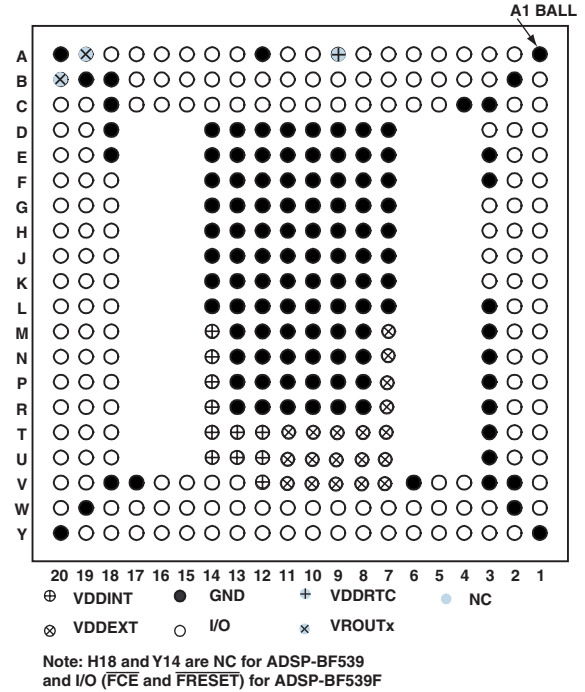


Figure 62. 316-Ball Mini-BGA Pin Configuration (Bottom View)

Table 50. 316-Ball Mini-BGA Pin Assignment (Numerically by Pin Number)

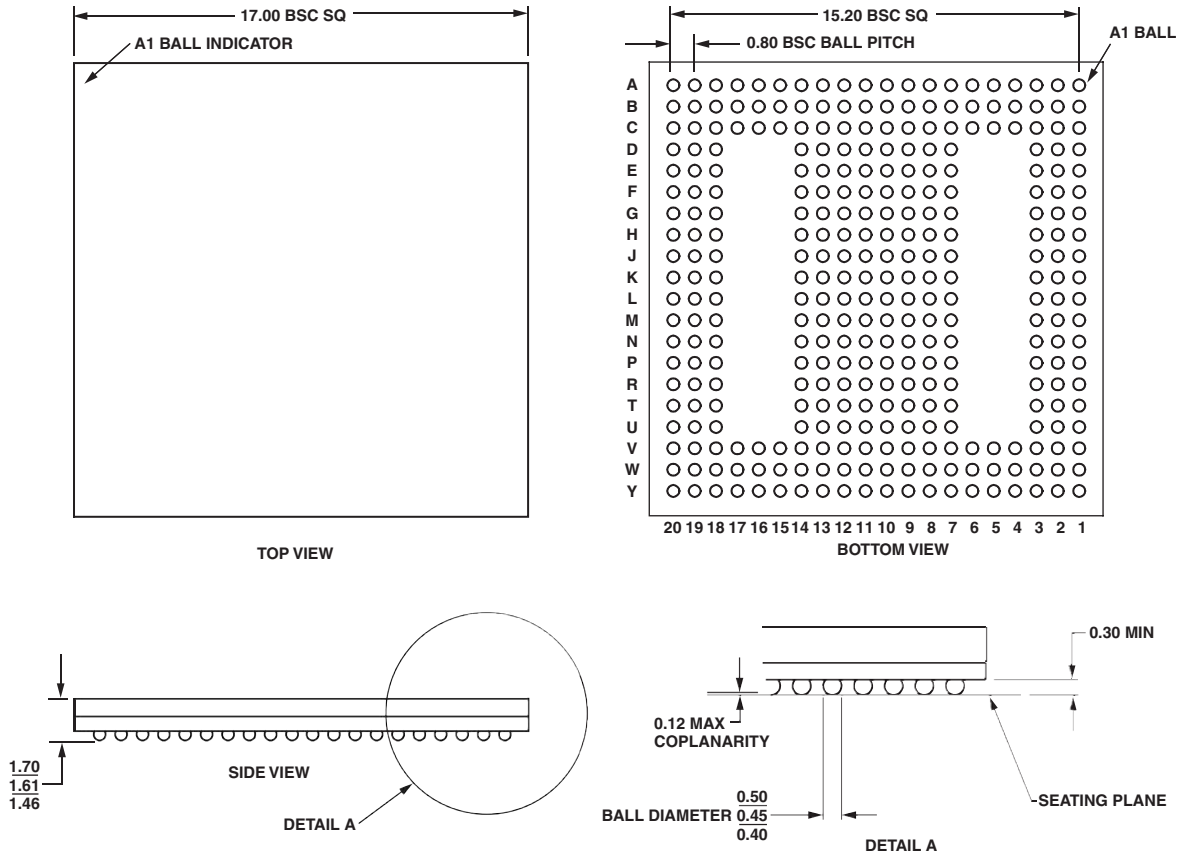
Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal
A1	GND	C7	SPI2SEL	F8	GND	J12	GND	M19	ABE0	T3	GND	W1	TCK
A2	PF10	C8	SPI2SS	F9	GND	J13	GND	M20	ABE1	T7	VDDEXT	W2	GND
A3	PF11	C9	MOSI2	F10	GND	J14	GND	N1	TFS0	T8	VDDEXT	W3	DATA15
A4	PPI_CLK	C10	MISO2	F11	GND	J18	AMS0	N2	DR0PRI	T9	VDDEXT	W4	DATA13
A5	PPI0	C11	SCK2	F12	GND	J19	AMS2	N3	GND	T10	VDDEXT	W5	DATA11
A6	PPI2	C12	MPIVDD	F13	GND	J20	SA10	N7	VDDEXT	T11	VDDEXT	W6	DATA9
A7	PF15	C13	SPI1SEL	F14	GND	K1	RFS1	N8	GND	T12	VDDINT	W7	DATA7
A8	PF13	C14	MISO1	F18	DT3PRI	K2	TMR2	N9	GND	T13	VDDINT	W8	DATA5
A9	VDDRTC	C15	SPI1SS	F19	MRX	K3	GP	N10	GND	T14	VDDINT	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	RSCK2
A12	GND	C18	GND	G2	MOSI0	K9	GND	N13	GND	T20	ADDR8	W12	DR2PRI
A13	CLKIN	C19	MMCLK	G3	DT0SEC	K10	GND	N14	VDDINT	U1	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	VDDEXT	W16	ADDR18
A17	MXI	D3	DT1SEC	G10	GND	K14	GND	P1	TSCK0	U8	VDDEXT	W17	ADDR15
A18	MRXON	D7	GND	G11	GND	K18	AMS3	P2	RFS0	U9	VDDEXT	W18	ADDR13
A19	VROUT1	D8	GND	G12	GND	K19	AMS1	P3	GND	U10	VDDEXT	W19	GND
A20	GND	D9	GND	G13	GND	K20	AOE	P7	VDDEXT	U11	VDDEXT	W20	ADDR14
B1	PF8	D10	GND	G14	GND	L1	RSCK1	P8	GND	U12	VDDINT	Y1	GND
B2	GND	D11	GND	G18	BR	L2	TMR1	P9	GND	U13	VDDINT	Y2	TDO
B3	PF9	D12	GND	G19	CLKOUT	L3	GND	P10	GND	U14	VDDINT	Y3	DATA14
B4	PF3	D13	GND	G20	SRAS	L7	GND	P11	GND	U18	RSCK3	Y4	DATA12
B5	PPI1	D14	GND	H1	DT1PRI	L8	GND	P12	GND	U19	ADDR9	Y5	DATA10
B6	PPI3	D18	GND	H2	TSCK1	L9	GND	P13	GND	U20	ADDR10	Y6	DATA8
B7	PF14	D19	MBCLK	H3	DR1SEC	L10	GND	P14	VDDINT	V1	TDI	Y7	DATA6
B8	PF12	D20	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	TSCK3	R2	RSCK0	V6	GND	Y12	TSCK2
B13	NMI	E8	GND	H12	GND	L19	ARE	R3	GND	V7	VDDEXT	Y13	TFS2
B14	RESET	E9	GND	H13	GND	L20	AWE	R7	VDDEXT	V8	VDDEXT	Y14	FRESET
B15	MXEVDD	E10	GND	H14	GND	M1	DT0PRI	R8	GND	V9	VDDEXT	Y15	SCL1
B16	MXEGND	E11	GND	H18	FCE	M2	TMRO	R9	GND	V10	VDDEXT	Y16	SDA1
B17	MTXON	E12	GND	H19	SCAS	M3	GND	R10	GND	V11	VDDEXT	Y17	ADDR19
B18	GND	E13	GND	H20	SWE	M7	VDDEXT	R11	GND	V12	VDDINT	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	BG	Y20	GND
C1	PF6	E19	MTX	J3	DR0SEC	M10	GND	R14	VDDINT	V15	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	VDDINT	T1	RX0	V19	ADDR11		
C6	TX1	F7	GND	J11	GND	M18	TFS3	T2	EMU	V20	ADDR12		

Table 51. 316-Ball Mini-BGA Pin 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.
$\overline{\text{ABE0}}$	M19	DATA8	Y6	GND	E7	GND	K11	GND	V17	PPI2	A6	TSCK1	H2
$\overline{\text{ABE1}}$	M20	DATA9	W6	GND	E8	GND	K12	GND	V18	PPI3	B6	TSCK2	Y12
ADDR1	N19	DATA10	Y5	GND	E9	GND	K13	GND	W2	$\overline{\text{RESET}}$	B14	TSCK3	L18
ADDR2	N20	DATA11	W5	GND	E10	GND	K14	GND	W19	RFS0	P2	TX0	R1
ADDR3	P19	DATA12	Y4	GND	E11	GND	L3	GND	Y1	RFS1	K1	TX1	C6
ADDR4	P20	DATA13	W4	GND	E12	GND	L7	GND	Y20	RFS2	Y11	TX2	W15
ADDR5	R19	DATA14	Y3	GND	E13	GND	L8	GP	K3	RFS3	T18	VDDEXT	M7
ADDR6	R20	DATA15	W3	GND	E14	GND	L9	MBCLK	D19	RSCK0	R2	VDDEXT	N7
ADDR7	T19	DR0PRI	N2	GND	E18	GND	L10	MFS	F20	RSCK1	L1	VDDEXT	P7
ADDR8	T20	DR0SEC	J3	GND	F3	GND	L11	MMCLK	C19	RSCK2	W11	VDDEXT	R7
ADDR9	U19	DR1PRI	J2	GND	F7	GND	L12	MLF	A15	RSCK3	U18	VDDEXT	T7
ADDR10	U20	DR1SEC	H3	GND	F8	GND	L13	MISO0	F2	RTXI	A11	VDDEXT	T8
ADDR11	V19	DR2PRI	W12	GND	F9	GND	L14	MISO1	C14	RTXO	A10	VDDEXT	T9
ADDR12	V20	DR2SEC	V13	GND	F10	GND	M3	MISO2	C10	RX0	T1	VDDEXT	T10
ADDR13	W18	DR3PRI	R18	GND	F11	GND	M8	MOSI0	G2	RX1	C5	VDDEXT	T11
ADDR14	W20	DR3SEC	P18	GND	F12	GND	M9	MOSI1	C16	RX2	W14	VDDEXT	U7
ADDR15	W17	DT0PRI	M1	GND	F13	GND	M10	MOSI2	C9	SA10	J20	VDDEXT	U8
ADDR16	Y19	DT0SEC	G3	GND	F14	GND	M11	MPIVDD	C12	$\overline{\text{SCAS}}$	H19	VDDEXT	U9
ADDR17	Y18	DT1PRI	H1	GND	G7	GND	M12	$\overline{\text{MRXON}}$	A18	SCK0	G1	VDDEXT	U10
ADDR18	W16	DT1SEC	D3	GND	G8	GND	M13	MRX	F19	SCK1	C17	VDDEXT	U11
ADDR19	Y17	DT2PRI	W13	GND	G9	GND	N3	MTX	E19	SCK2	C11	VDDEXT	V7
$\overline{\text{AMS0}}$	J18	DT2SEC	V16	GND	G10	GND	N8	$\overline{\text{MTXON}}$	B17	SCKE	C20	VDDEXT	V8
$\overline{\text{AMS1}}$	K19	DT3PRI	F18	GND	G11	GND	N9	MXEGND	B16	SCL0	B9	VDDEXT	V9
$\overline{\text{AMS2}}$	J19	DT3SEC	N18	GND	G12	GND	N10	MXEVDD	B15	SCL1	Y15	VDDEXT	V10
$\overline{\text{AMS3}}$	K18	$\overline{\text{EMU}}$	T2	GND	G13	GND	N11	MXI	A17	SDA0	B10	VDDEXT	V11
$\overline{\text{AOE}}$	K20	$\overline{\text{FCE}}$	H18	GND	G14	GND	N12	MXO	A16	SDA1	Y16	VDDINT	M14
ARDY	E20	$\overline{\text{FRESET}}$	Y14	GND	H7	GND	N13	$\overline{\text{NMI}}$	B13	$\overline{\text{SMS}}$	D20	VDDINT	N14
$\overline{\text{ARE}}$	L19	GND	A1	GND	H8	GND	P3	PF0	F1	SPI1SEL	C13	VDDINT	P14
$\overline{\text{AWE}}$	L20	GND	A12	GND	H9	GND	P8	PF1	E1	SPI1SS	C15	VDDINT	R14
$\overline{\text{BG}}$	V14	GND	A20	GND	H10	GND	P9	PF2	E2	SPI2SEL	C7	VDDINT	T12
$\overline{\text{BGH}}$	V15	GND	B2	GND	H11	GND	P10	PF3	B4	SPI2SS	C8	VDDINT	T13
BMODE0	V5	GND	B18	GND	H12	GND	P11	PF4	D1	$\overline{\text{SRAS}}$	G20	VDDINT	T14
BMODE1	V4	GND	B19	GND	H13	GND	P12	PF5	D2	$\overline{\text{SWE}}$	H20	VDDINT	U12
$\overline{\text{BR}}$	G18	GND	C3	GND	H14	GND	P13	PF6	C1	TCK	W1	VDDINT	U13
CANRX	B11	GND	C4	GND	J7	GND	R3	PF7	C2	TDI	V1	VDDINT	U14
CANTX	B12	GND	C18	GND	J8	GND	R8	PF8	B1	TDO	Y2	VDDINT	V12
CLKIN	A13	GND	D7	GND	J9	GND	R9	PF9	B3	TFS0	N1	VDDRTC	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	TMRO	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	$\overline{\text{TRST}}$	U1		
DATA7	W7	GND	E3	GND	K10	GND	V6	PPI1	B5	TSCK0	P1		

OUTLINE DIMENSIONS

Dimensions in Figure 63—316-Ball Mini Ball Grid Array (BC-316) are shown in millimeters.



- NOTES:
1. ALL DIMENSIONS ARE IN MILLIMETERS.
 2. COMPLIANT TO JEDEC REGISTERED OUTLINE MO-205, VARIATION AM, WITH THE EXCEPTION OF BALL DIAMETER.
 3. CENTER DIMENSIONS ARE NOMINAL.

Figure 63. 316-Ball Mini Ball Grid Array (BC-316)

SURFACE MOUNT DESIGN

Table 52 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 52. BGA Data for Use with Surface Mount Design

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

ORDERING GUIDE

Model ^{1,2}	Temperature Range ³	Package Description	Package Option	Instruction Rate (Max)	Operating Voltage (Nominal)
ADSP-BF539BBCZ-5A	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WBBCZ-5A	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539YBCZ-4A	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WYBCZ-4A	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539BBCZ-5F4 ⁴	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WBBCZ5F4 ⁴	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539BBCZ-5F8 ⁵	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WBBCZ5F8 ⁵	-40°C to +85°C	316-Ball Mini Ball Grid Array	BC-316	500 MHz	1.2 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539YBCZ-4F4 ⁴	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WYBCZ4F4 ⁴	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539YBCZ-4F8 ⁵	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O
ADSP-BF539WYBCZ4F8 ⁵	-40°C to +105°C	316-Ball Mini Ball Grid Array	BC-316	400 MHz	1.0 V internal, 2.5 V or 3.3 V I/O

¹ W indicates this product is available for use in specific automotive applications. Contact your local ADI sales office for specific ordering information.

² Z indicates Pb-free part.

³ Referenced temperature is ambient temperature.

⁴ F4 indicates 256K x 16-bit flash memory.

⁵ F8 indicates 512K x 16-bit flash memory.

