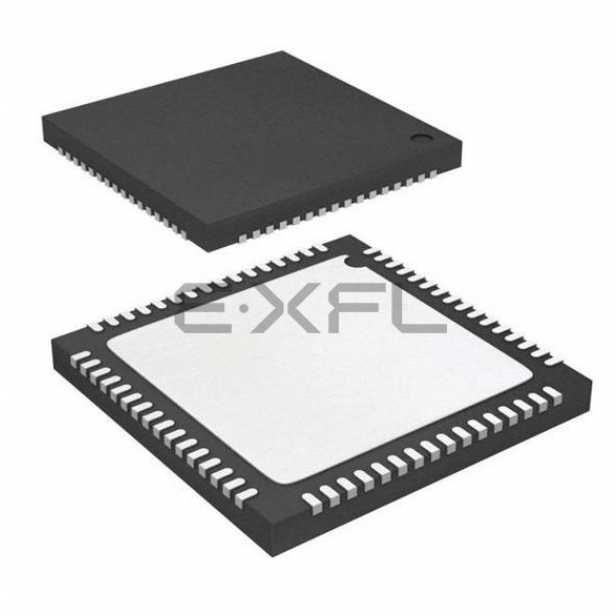


Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Fixed Point
Interface	I²C, I²S, IrDA, PPI, SPI, SPORT, UART
Clock Rate	200MHz
Non-Volatile Memory	ROM (64kB)
On-Chip RAM	64kB
Voltage - I/O	1.8V, 2.5V, 3.3V
Voltage - Core	1.29V
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad, CSP
Supplier Device Package	64-LFCSP-VQ (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-bf592kcpz-2

- EE-302: Interfacing ADSP-BF53x Blackfin® Processors to NAND FLASH Memory
- EE-303: Using VisualDSP++® Thread-Safe Libraries with a Third-Party RTOS
- EE-304: Using the Blackfin® Processor SPORT to Emulate a SPI Interface
- EE-306: PGO Linker - A Code Layout Tool for Blackfin Processors
- EE-312: Building Complex VDK/LwIP Applications Using Blackfin® Processors
- EE-323: Implementing Dynamically Loaded Software Modules
- EE-325: Interfacing Atmel Fingerprint Sensor AT77C104B with Blackfin® Processors
- EE-326: Blackfin® Processor and SDRAM Technology
- EE-330: Windows Vista Compatibility in VisualDSP++ 5.0 Development Tools
- EE-332: Cycle Counting and Profiling
- EE-333: Interfacing Blackfin® Processors to Winbond W25X16 SPI Flash Devices
- EE-334: Using Blackfin® Processor Hibernate State for Low Standby Power
- EE-336: Putting ADSP-BF54x Blackfin® Processor Booting into Practice
- EE-339: Using External Switching Regulators with Blackfin® Processors
- EE-340: Connecting SHARC® and Blackfin® Processors over SPI
- EE-341: Expert Pin Multiplexing Plug-in for Blackfin® Processors
- EE-347: Formatted Print to a UART Terminal with Blackfin® Processors
- EE-350: Seamlessly Interfacing MEMS Microphones with Blackfin Processors
- EE-351: Using the ADSP-BF592 Blackfin® Processor Tools Utility ROM
- EE-352: Soldering Considerations for Exposed-Pad Packages
- EE-356: Emulator and Evaluation Hardware Troubleshooting Guide for CCES Users

Data Sheet

- ADSP-BF592 Blackfin Embedded Processor Data Sheet

Emulator Manuals

- HPUSB, USB, and HPPCI Emulator User's Guide
- ICE-1000/ICE-2000 Emulator User's Guide
- ICE-100B Emulator User's Guide

Evaluation Kit Manuals

- ADSP-BF592 EZ-KIT Lite® Evaluation System Manual
- Blackfin®/SHARC® USB EZ-Extender® Manual

Integrated Circuit Anomalies

- ADSP-BF592 Blackfin Silicon Anomaly List for Revisions 0.1, 0.2

Processor Manuals

- ADSP-BF59x Blackfin ® Processor Hardware Reference
- ADSP-BF5xx/ADSP-BF60x Blackfin® Processor Programming Reference
- Blackfin Processors: Manuals

Software Manuals

- CrossCore® Embedded Studio 2.5.0 Assembler and Preprocessor Manual
- CrossCore® Embedded Studio 2.5.0 C/C++ Compiler and Library Manual for Blackfin Processors
- CrossCore® Embedded Studio 2.5.0 Linker and Utilities Manual
- CrossCore® Embedded Studio 2.5.0 Loader and Utilities Manual
- CrossCore® Software Licensing Guide
- lwIP for CrossCore® Embedded Studio 1.0.0 User's Guide
- VisualDSP++® 5.0 Assembler and Preprocessor Manual
- VisualDSP++® 5.0 C/C++ Compiler and Library Manual for Blackfin Processors
- VisualDSP++® 5.0 Device Drivers and System Services Manual for Blackfin Processors
- VisualDSP++® 5.0 Kernel (VDK) Users Guide
- VisualDSP++® 5.0 Licensing Guide
- VisualDSP++® 5.0 Linker and Utilities Manual
- VisualDSP++® 5.0 Loader and Utilities Manual
- VisualDSP++® 5.0 Product Release Bulletin
- VisualDSP++® 5.0 Quick Installation Reference Card
- VisualDSP++® 5.0 Users Guide

SOFTWARE AND SYSTEMS REQUIREMENTS

- Software and Tools Anomalies Search

TOOLS AND SIMULATIONS

- ADSP-BF592 Blackfin Processor BSDL File 64-Lead LFCSP (11/2011)
- Blackfin Processors Software and Tools
- ADSP-BF592 Blackfin Processor IBIS Datafile 64 Lead LFCSP Package (11/2011)

TABLE OF CONTENTS

REVISION HISTORY

7/13—Rev. A to Rev. B

Corrected Processor Block Diagram 1

Updated Development Tools 12

Updated text in Signal Descriptions 14

Corrected V_{DDINT} rating in Table 14, Absolute Maximum Ratings 20

head 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. Data memory holds data, and a dedicated scratchpad data memory stores stack and local variable information.

Multiple L1 memory blocks are provided. 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.

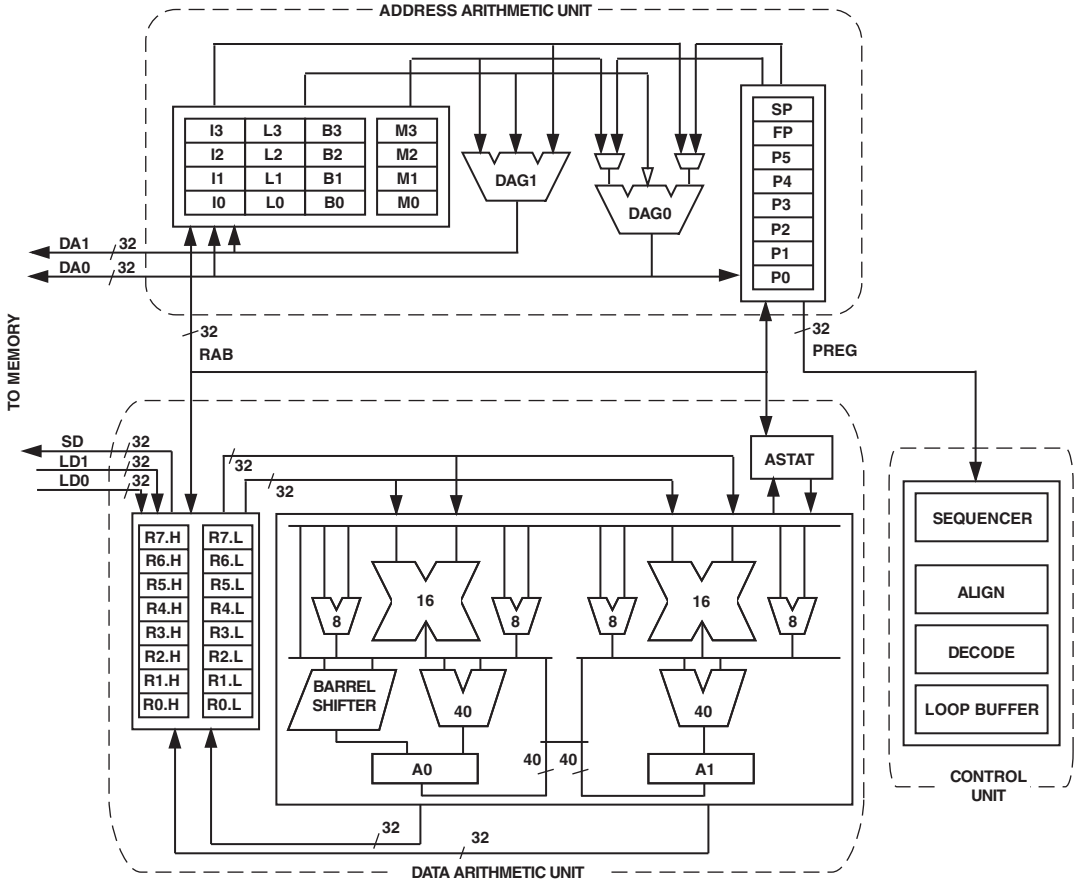


Figure 2. Blackfin Processor Core

MEMORY ARCHITECTURE

The Blackfin processor views memory as a single unified 4G byte address space, using 32-bit addresses. All resources, including internal memory and I/O control registers, occupy separate sections of this common address space. See Figure 3.

The core-accessible L1 memory system is high performance internal memory that operates at the core clock frequency. The external bus interface unit (EBIU) provides access to the boot ROM.

The memory DMA controller provides high bandwidth data-movement capability. It can perform block transfers of code or data between the L1 Instruction SRAM and L1 Data SRAM memory spaces.

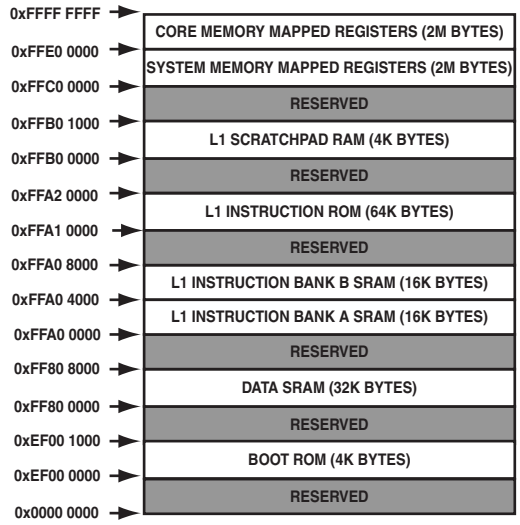


Figure 3. Internal/External Memory Map

Internal (Core-Accessible) Memory

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

The first block is the L1 instruction memory, consisting of 32K bytes SRAM. This memory is accessed at full processor speed.

The second core-accessible memory block is the L1 data memory, consisting of 32K bytes. This memory block is accessed at full processor speed.

The third memory block is a 4K byte L1 scratchpad SRAM, which runs at the same speed as the other L1 memories.

L1 Utility ROM

The L1 instruction ROM contains utility ROM code. This includes the TMK (VDK core), C run-time libraries, and DSP libraries. See the VisualDSP++ documentation for more information.

Custom ROM (Optional)

The on-chip L1 Instruction ROM on the ADSP-BF592 may be customized to contain user code with the following features:

- 64K bytes of L1 Instruction ROM available for custom code
- Ability to restrict access to all or specific segments of the on-chip ROM

Customers wishing to customize the on-chip ROM for their own application needs should contact ADI sales for more information on terms and conditions and details on the technical implementation.

I/O Memory Space

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

Booting from ROM

The processor contains a small on-chip boot kernel, which configures the appropriate peripheral for booting. If the processor is configured to boot from boot ROM memory space, the processor starts executing from the on-chip boot ROM. For more information, see Booting Modes on Page 11.

EVENT HANDLING

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

- Emulation – An emulation event causes the processor to enter emulation mode, allowing command and control of the processor via the JTAG interface.
- $\overline{\text{RESET}}$ – This event resets the processor.
- Nonmaskable Interrupt (NMI) – The NMI event can be generated by the software watchdog timer or by the $\overline{\text{NMI}}$ input signal to the processor. The NMI event is frequently used as a power-down indicator to initiate an orderly shutdown of the system.

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. Input mode is intended for ADC applications, as well as video communication with hardware signaling.
- Frame capture mode – Frame syncs are outputs from the PPI, but data are inputs. This mode allows the video source(s) to act as a slave (for frame capture for example).
- Output mode – Frame syncs and data are outputs from the PPI. Output mode is used for transmitting video or other data with up to three output frame syncs.

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 – Active video only mode is used when only the active video portion of a field is of interest and not any of the blanking intervals.
- Vertical blanking only 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.

TWI Controller Interface

The processor includes a 2-wire interface (TWI) module for providing a simple exchange method of control data between multiple devices. The TWI is functionally compatible with the widely used I²C® bus standard. The TWI module offers the capabilities of simultaneous master and slave operation and support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (SCL) and data (SDA) and supports the protocol at speeds up to 400K bits/sec.

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

Ports

The processor groups the many peripheral signals to two ports—Port F and Port G. Most of the associated pins are shared by multiple signals. The ports function as multiplexer controls.

General-Purpose I/O (GPIO)

The processor has 32 bidirectional, general-purpose I/O (GPIO) pins allocated across two separate GPIO modules—PORTFIO and PORTGIO, associated with Port F and Port G respectively. Each GPIO-capable pin shares functionality with other processor peripherals via a multiplexing scheme; however, the GPIO functionality is the default state of the device upon power-up. Neither GPIO output nor input drivers are active by default. Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers.

DYNAMIC POWER MANAGEMENT

The 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. When configured for a 0 V core supply voltage, the processor enters the hibernate state. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 2 for a summary of the power settings for each mode.

Table 2. Power Settings

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

Full-On Operating Mode—Maximum Performance

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

Active Operating Mode—Moderate Dynamic Power Savings

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

For more information about PLL controls, see the "Dynamic Power Management" chapter in the *ADSP-BF59x Blackfin Processor Hardware Reference*.

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 wakes up the processor.

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

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core (CCLK) and to all synchronous peripherals (SCLK). Asynchronous peripherals may still be running but cannot access internal resources or external memory. This powered-down mode can only be exited by assertion of the reset interrupt ($\overline{\text{RESET}}$) or by an asynchronous interrupt generated by a GPIO pin.

Note that when a GPIO pin is used to trigger wake from deep sleep, the programmed wake level must linger for at least 10ns to guarantee detection.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling clocks to the processor core (CCLK) and to all of the peripherals (SCLK), as well as signaling an external voltage regulator that V_{DDINT} can be shut off. Any critical information stored internally (for example, memory contents, register contents, and other information) must be written to a nonvolatile storage device prior to removing power if the processor state is to be preserved. Writing b#0 to the $\overline{\text{HIBERNATE}}$ bit causes EXT_WAKE to transition low, which can be used to signal an external voltage regulator to shut down.

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

As long as V_{DDEXT} is applied, the VR_CTL register maintains its state during hibernation. All other internal registers and memories, however, lose their content in the hibernate state.

Power Savings

As shown in Table 3, the processor supports two different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions. By isolating the internal logic of the processor into its own power domain, separate from other I/O, the processor can take advantage of dynamic power management without affecting the other I/O devices. There are no sequencing requirements for the various power domains, but all domains must be powered according to the appropriate Specifications table for processor operating conditions, even if the feature/peripheral is not used.

Table 3. Power Domains

Power Domain	V_{DD} Range
All internal logic and memories	V_{DDINT}
All other I/O	V_{DDEXT}

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

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

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

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

where:

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

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

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

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

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

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

VOLTAGE REGULATION

The ADSP-BF592 processor requires an external voltage regulator to power the V_{DDINT} domain. To reduce standby power consumption, the external voltage regulator can be signaled through EXT_WAKE to remove power from the processor core. This signal is high-true for power-up and may be connected directly to the low-true shut-down input of many common regulators.

While in the hibernate state, the external supply, V_{DDEXT}, can still be applied, eliminating the need for external buffers. The external voltage regulator can be activated from this power-down state by asserting the $\overline{\text{RESET}}$ pin, which then initiates a boot sequence. EXT_WAKE indicates a wakeup to the external voltage regulator.

The power good ($\overline{\text{PG}}$) input signal allows the processor to start only after the internal voltage has reached a chosen level. In this way, the startup time of the external regulator is detected after hibernation. For a complete description of the power-good functionality, refer to the *ADSP-BF59x Blackfin Processor Hardware Reference*.

CLOCK SIGNALS

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

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

Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal may be used. For fundamental frequency operation, use the circuit shown in Figure 4. A parallel-resonant, fundamental frequency, microprocessor-grade crystal is connected across the CLKIN and XTAL pins. The on-chip resistance between CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not

INSTRUCTION SET DESCRIPTION

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

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

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

DEVELOPMENT TOOLS

Analog Devices supports its processors with a complete line of software and hardware development tools, including integrated development environments (which include CrossCore® Embedded Studio and/or VisualDSP++®), evaluation products, emulators, and a wide variety of software add-ins.

Integrated Development Environments (IDEs)

For C/C++ software writing and editing, code generation, and debug support, Analog Devices offers two IDEs.

The newest IDE, CrossCore Embedded Studio, is based on the Eclipse™ framework. Supporting most Analog Devices processor families, it is the IDE of choice for future processors, including multicore devices. CrossCore Embedded Studio seamlessly integrates available software add-ins to support real time operating systems, file systems, TCP/IP stacks, USB stacks, algorithmic software modules, and evaluation hardware board support packages. For more information, visit www.analog.com/cces.

The other Analog Devices IDE, VisualDSP++, supports processor families introduced prior to the release of CrossCore Embedded Studio. This IDE includes the Analog Devices VDK real time operating system and an open source TCP/IP stack. For more information visit www.analog.com/visualdsp. Note that VisualDSP++ will not support future Analog Devices processors.

EZ-KIT Lite Evaluation Board

For processor evaluation, Analog Devices provides wide range of EZ-KIT Lite® evaluation boards. Including the processor and key peripherals, the evaluation board also supports on-chip emulation capabilities and other evaluation and development features. Also available are various EZ-Extenders®, which are daughter cards delivering additional specialized functionality, including audio and video processing. For more information visit www.analog.com and search on "ezkit" or "ezextender".

EZ-KIT Lite Evaluation Kits

For a cost-effective way to learn more about developing with Analog Devices processors, Analog Devices offer a range of EZ-KIT Lite evaluation kits. Each evaluation kit includes an EZ-KIT Lite evaluation board, directions for downloading an evaluation version of the available IDE(s), a USB cable, and a power supply. The USB controller on the EZ-KIT Lite board connects to the USB port of the user's PC, enabling the chosen IDE evaluation suite to emulate the on-board processor in-circuit. This permits the customer to download, execute, and debug programs for the EZ-KIT Lite system. It also supports in-circuit programming of the on-board Flash device to store user-specific boot code, enabling standalone operation. With the full version of CrossCore Embedded Studio or VisualDSP++ installed (sold separately), engineers can develop software for supported EZ-KITs or any custom system utilizing supported Analog Devices processors.

Software Add-Ins for CrossCore Embedded Studio

Analog Devices offers software add-ins which seamlessly integrate with CrossCore Embedded Studio to extend its capabilities and reduce development time. Add-ins include board support packages for evaluation hardware, various middleware packages, and algorithmic modules. Documentation, help, configuration dialogs, and coding examples present in these add-ins are viewable through the CrossCore Embedded Studio IDE once the add-in is installed.

Board Support Packages for Evaluation Hardware

Software support for the EZ-KIT Lite evaluation boards and EZ-Extender daughter cards is provided by software add-ins called Board Support Packages (BSPs). The BSPs contain the required drivers, pertinent release notes, and select example code for the given evaluation hardware. A download link for a specific BSP is located on the web page for the associated EZ-KIT or EZ-Extender product. The link is found in the *Product Download* area of the product web page.

Middleware Packages

Analog Devices separately offers middleware add-ins such as real time operating systems, file systems, USB stacks, and TCP/IP stacks. For more information see the following web pages:

- www.analog.com/ucos3
- www.analog.com/ucfs
- www.analog.com/ucusbd
- www.analog.com/lwip

Algorithmic Modules

To speed development, Analog Devices offers add-ins that perform popular audio and video processing algorithms. These are available for use with both CrossCore Embedded Studio and VisualDSP++. For more information visit www.analog.com and search on "Blackfin software modules" or "SHARC software modules".

Designing an Emulator-Compatible DSP Board (Target)

For embedded system test and debug, Analog Devices provides a family of emulators. On each JTAG DSP, Analog Devices supplies an IEEE 1149.1 JTAG Test Access Port (TAP). In-circuit emulation is facilitated by use of this JTAG interface. The emulator accesses the processor's internal features via the processor's TAP, allowing the developer to load code, set breakpoints, and view variables, memory, and registers. The processor must be halted to send data and commands, but once an operation is completed by the emulator, the DSP system is set to run at full speed with no impact on system timing. The emulators require the target board to include a header that supports connection of the DSP's JTAG port to the emulator.

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

ADDITIONAL INFORMATION

The following publications that describe the ADSP-BF592 processor (and related processors) can be ordered from any Analog Devices sales office or accessed electronically on our website:

- *Getting Started With Blackfin Processors*
- *ADSP-BF59x Blackfin Processor Hardware Reference*
- *Blackfin Processor Programming Reference*
- *ADSP-BF592 Blackfin Processor Anomaly List*

RELATED SIGNAL CHAINS

A *signal chain* is a series of signal conditioning electronic components that receive input (data acquired from sampling either real-time phenomena or from stored data) in tandem, with the output of one portion of the chain supplying input to the next. Signal chains are often used in signal processing applications to gather and process data or to apply system controls based on analysis of real-time phenomena. For more information about this term and related topics, see the "signal chain" entry in the Glossary of EE Terms on the Analog Devices website.

Analog Devices eases signal processing system development by providing signal processing components that are designed to work together well. A tool for viewing relationships between specific applications and related components is available on the www.analog.com website.

The Circuits from the Lab™ site (www.analog.com\circuits) provides:

- Graphical circuit block diagram presentation of signal chains for a variety of circuit types and applications
- Drill down links for components in each chain to selection guides and application information
- Reference designs applying best practice design techniques

SIGNAL DESCRIPTIONS

Signal definitions for the ADSP-BF592 processor are listed in Table 7. In order to maintain maximum function and reduce package size and pin count, some pins have dual, multiplexed functions. In cases where pin function is reconfigurable, the default state is shown in plain text, while the alternate function is shown in italics.

During and immediately after reset, all processor signals are three-stated with the following exceptions: EXT_WAKE is driven high and XTAL is driven in conjunction with CLKIN to create a crystal oscillator circuit. During hibernate, all signals are three-stated with the following exceptions: EXT_WAKE is driven low and XTAL is driven to a solid logic level.

During and immediately after reset, all I/O pins have their input buffers disabled with the exception of the pins that need pull-ups or pull-downs, as noted in Table 7.

Adding a parallel termination to EXTCLK may prove useful in further enhancing signal integrity. Be sure to verify over-shoot/undershoot and signal integrity specifications on actual hardware.

Table 7. Signal Descriptions

Signal Name	Type	Function	Driver Type
Port F: *GPIO and Multiplexed Peripherals*			
PF0–GPIO/*DR1SEC/PPI_D8/WAKEN1*	I/O	GPIO/*SPORT1 Receive Data Secondary/PPI Data 8/Wake Enable 1*	A
PF1–GPIO/*DR1PRI/PPI_D9*	I/O	GPIO/*SPORT1 Receive Data Primary/PPI Data 9*	A
PF2–GPIO/*RSCLK1/PPI_D10*	I/O	GPIO/*SPORT1 Receive Serial Clock/PPI Data 10*	A
PF3–GPIO/*RFS1/PPI_D11*	I/O	GPIO/*SPORT1 Receive Frame Sync/PPI Data 11*	A
PF4–GPIO/*DT1SEC/PPI_D12*	I/O	GPIO/*SPORT1 Transmit Data Secondary/PPI Data 12*	A
PF5–GPIO/*DT1PRI/PPI_D13*	I/O	GPIO/*SPORT1 Transmit Data Primary/PPI Data 13*	A
PF6–GPIO/*TSCLK1/PPI_D14*	I/O	GPIO/*SPORT1 Transmit Serial Clock/PPI Data 14*	A
PF7–GPIO/*TFS1/PPI_D15*	I/O	GPIO/*SPORT1 Transmit Frame Sync/PPI Data 15*	A
PF8–GPIO/*TMR2/$\overline{\text{SPI0\_SSEL2}}$/WAKEN0*	I/O	GPIO/*Timer 2/SPI0 Slave Select Enable 2/Wake Enable 0*	A
PF9–GPIO/*TMR0/PPI_FS1/*$\overline{\text{SPI0\_SSEL3}}$	I/O	GPIO/*Timer 0/PPI Frame Sync 1/SPI0 Slave Select Enable 3*	A
PF10–GPIO/*TMR1/PPI_FS2*	I/O	GPIO/*Timer 1/PPI Frame Sync 2*	A
PF11–GPIO/*UA_TX/$\overline{\text{SPI0\_SSEL4}}$*	I/O	GPIO/*UART Transmit/SPI0 Slave Select Enable 4*	A
PF12–GPIO/*UA_RX/$\overline{\text{SPI0\_SSEL7}}$/TACI2–0*	I/O	GPIO/*UART Receive/SPI0 Slave Select Enable 7/Timers 2–0 Alternate Input Capture*	A
PF13–GPIO/*SPI0_MOSI/$\overline{\text{SPI1\_SSEL3}}$*	I/O	GPIO/*SPI0 Master Out Slave In/SPI1 Slave Select Enable 3*	A
PF14–GPIO/*SPI0_MISO/$\overline{\text{SPI1\_SSEL4}}$*	I/O	GPIO/*SPI0 Master In Slave Out/SPI1 Slave Select Enable 4* (This pin should always be pulled high through a 4.7 kΩ resistor, if booting via the SPI port.)	A
PF15–GPIO/*SPI0_SCK/$\overline{\text{SPI1\_SSEL5}}$*	I/O	GPIO/*SPI0 Clock/SPI1 Slave Select Enable 5*	A
Port G: *GPIO and Multiplexed Peripherals*			
PG0–GPIO/*DR0SEC/$\overline{\text{SPI0\_SSEL1}}$/$\overline{\text{SPI0\_SS}}$*	I/O	GPIO/*SPORT0 Receive Data Secondary/SPI0 Slave Select Enable 1/SPI0 Slave Select Input*	A
PG1–GPIO/*DR0PRI/$\overline{\text{SPI1\_SSEL1}}$/WAKEN3*	I/O	GPIO/*SPORT0 Receive Data Primary/SPI1 Slave Select Enable 1/Wake Enable 3*	A
PG2–GPIO/*RSCLK0/$\overline{\text{SPI0\_SSEL5}}$*	I/O	GPIO/*SPORT0 Receive Serial Clock/SPI0 Slave Select Enable 5*	A
PG3–GPIO/*RFS0/PPI_FS3*	I/O	GPIO/*SPORT0 Receive Frame Sync/PPI Frame Sync 3*	A
PG4–GPIO(HWAIT)/*DT0SEC/$\overline{\text{SPI0\_SSEL6}}$*	I/O	GPIO (HWAIT output for Slave Boot Modes)/*SPORT0 Transmit Data Secondary/SPI0 Slave Select Enable 6*	A
PG5–GPIO/*DT0PRI/$\overline{\text{SPI1\_SSEL6}}$*	I/O	GPIO/*SPORT0 Transmit Data Primary/SPI1 Slave Select Enable 6*	A
PG6–GPIO/*TSCLK0*	I/O	GPIO/*SPORT0 Transmit Serial Clock*	A
PG7–GPIO/*TFS0/$\overline{\text{SPI1\_SSEL7}}$*	I/O	GPIO/*SPORT0 Transmit Frame Sync/SPI1 Slave Select Enable 7*	A
PG8–GPIO/*SPI1_SCK/PPI_D0*	I/O	GPIO/*SPI1 Clock/PPI Data 0*	A
PG9–GPIO/*SPI1_MOSI/PPI_D1*	I/O	GPIO/*SPI1 Master Out Slave In/PPI Data 1*	A

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	Min	Typical	Max	Unit
V_{OH}	High Level Output Voltage	V_{DDEXT} = 1.7 V, I_{OH} = –0.5 mA	1.35			V
V_{OH}	High Level Output Voltage	V_{DDEXT} = 2.25 V, I_{OH} = –0.5 mA	2.0			V
V_{OH}	High Level Output Voltage	V_{DDEXT} = 3.0 V, I_{OH} = –0.5 mA	2.4			V
V_{OL}	Low Level Output Voltage	V_{DDEXT} = 1.7 V/2.25 V/3.0 V, I_{OL} = 2.0 mA			0.4	V
V_{OLTWI}	Low Level Output Voltage	V_{DDEXT} = 1.7 V/2.25 V/3.0 V, I_{OL} = 2.0 mA			0.4	V V
I_{IH}	High Level Input Current[1]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			10	µA
I_{IL}	Low Level Input Current[1]	V_{DDEXT} = 3.6 V, V_{IN} = 0 V			10	µA
I_{IHP}	High Level Input Current JTAG[2]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V	10		50	µA
I_{OZH}	Three-State Leakage Current[3]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			10	µA
I_{OZHTWI}	Three-State Leakage Current[4]	V_{DDEXT} = 3.0 V, V_{IN} = 3.6 V			10	µA
I_{OZL}	Three-State Leakage Current[3]	V_{DDEXT} = 3.6 V, V_{IN} = 0 V			10	µA
C_{IN}	Input Capacitance[5]	f_{IN} = 1 MHz, $T_{AMBIENT}$ = 25°C, V_{IN} = 2.5 V		4	8[6]	pF
$I_{DDDEEPSLEEP}$[7]	V_{DDINT} Current in Deep Sleep Mode	V_{DDINT} = 1.2 V, f_{CCLK} = 0 MHz, f_{SCLK} = 0 MHz, T_J = 25°C, ASF = 0.00		0.8		mA
$I_{DDSLEEP}$	V_{DDINT} Current in Sleep Mode	V_{DDINT} = 1.2 V, f_{SCLK} = 25 MHz, T_J = 25°C		4		mA
$I_{DD-IDLE}$	V_{DDINT} Current in Idle	V_{DDINT} = 1.2 V, f_{CCLK} = 50 MHz, T_J = 25°C, ASF = 0.35		6		mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.3 V, f_{CCLK} = 200 MHz, T_J = 25°C, ASF = 1.00		40		mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.3 V, f_{CCLK} = 300 MHz, T_J = 25°C, ASF = 1.00		66		mA
I_{DD-TYP}	V_{DDINT} Current	V_{DDINT} = 1.4 V, f_{CCLK} = 400 MHz, T_J = 25°C, ASF = 1.00		91		mA
$I_{DDHIBERNATE}$[7]	Hibernate State Current	V_{DDEXT} = 3.3 V, T_J = 25°C, CLKIN = 0 MHz with voltage regulator off (V_{DDINT} = 0 V)		20		µA
$I_{DDDEEPSLEEP}$[7]	V_{DDINT} Current in Deep Sleep Mode	f_{CCLK} = 0 MHz, f_{SCLK} = 0 MHz			Table 12	mA
I_{DDINT}[8]	V_{DDINT} Current	f_{CCLK} > 0 MHz, f_{SCLK} ≥ 0 MHz			Table 12 + (Table 13 × ASF)	mA

[1] Applies to input pins.
[2] Applies to JTAG input pins (TCK, TDI, TMS, $\overline{TRST}$).
[3] Applies to three-statable pins.
[4] Applies to bidirectional pins SCL and SDA.
[5] Applies to all signal pins.
[6] Guaranteed, but not tested.
[7] See the *ADSP-BF59x Blackfin Processor Hardware Reference Manual* for definitions of sleep, deep sleep, and hibernate operating modes.
[8] See Table 11 for the list of I_{DDINT} power vectors covered.

ABSOLUTE MAXIMUM RATINGS

Stresses greater than those listed in Table 14 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.

Table 14. Absolute Maximum Ratings

Parameter	Rating
Internal Supply Voltage (V_{DDINT})	–0.3 V to +1.50 V
External (I/O) Supply Voltage (V_{DDEXT})	–0.3 V to +3.8 V
Input Voltage[1, 2]	–0.5 V to +3.6 V
Output Voltage Swing	–0.5 V to V_{DDEXT}+0.5 V
I_{OH}/I_{OL} Current per Pin Group	55 mA (Max)
I_{OH}/I_{OL} Current per Individual Pin	25 mA (Max)
Storage Temperature Range	–65°C to +150°C
Junction Temperature While Biased (Non-Automotive Models)	+110°C
Junction Temperature While Biased (Automotive Models)	+115°C

[1] Applies to 100% transient duty cycle. For other duty cycles see Table 15.
[2] Applies only when V_{DDEXT} is within specifications. When V_{DDEXT} is outside specifications, the range is V_{DDEXT} ± 0.2 Volts.

Table 15. Maximum Duty Cycle for Input Transient Voltage[1]

V_{IN} Min (V)[2]	V_{IN} Max (V)[2]	Maximum Duty Cycle[3]
–0.5	+3.8	100%
–0.7	+4.0	40%
–0.8	+4.1	25%
–0.9	+4.2	15%
–1.0	+4.3	10%

[1] Applies to all signal pins with the exception of CLKIN, XTAL, EXT_WAKE.
[2] The individual values cannot be combined for analysis of a single instance of overshoot or undershoot. The worst case observed value must fall within one of the voltages specified, and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.
[3] Duty cycle refers to the percentage of time the signal exceeds the value for the 100% case. The is equivalent to the measured duration of a single instance of overshoot or undershoot as a percentage of the period of occurrence.

Table 14 specifies the maximum total source/sink (I_{OH}/I_{OL}) current for a group of pins and for individual pins. Permanent damage can occur if this value is exceeded. To understand this specification, if pins PF0 and PF1 from Group 1 in Table 16 were sourcing or sinking 10 mA each, the total current for those pins would be 20 mA. This would allow up to 35 mA total that could be sourced or sunk by the remaining pins in the group without damaging the device. It should also be noted that the maximum source or sink current for an individual pin cannot exceed 25 mA. The list of all groups and their pins are shown in Table 16. Note that the V_{OH} and V_{OL} specifications have separate per-pin maximum current requirements, see the Electrical Characteristics table.

Table 16. Total Current Pin Groups–V_{DDEXT} Groups

Group	Pins in Group
1	PF0, PF1, PF2, PF3
2	PF4, PF5, PF6, PF7
3	PF8, PF9, PF10, PF11
4	PF12, PF13, PF14, PF15
5	PG3, PG2, PG1, PG0
6	PG7, PG6, PG5, PG4
7	PG11, PG10, PG9, PG8
8	PG15, PG14, PG13, PG12
9	TDI, TDO, EMU, TCK, $\overline{TRST}$, TMS
10	BMODE2, BMODE1, BMODE0
11	EXT_WAKE, $\overline{PG}$, $\overline{RESET}$, $\overline{NMI}$, PPI_CLK, EXTCLK
12	SDA, SCL, CLKIN, XTAL

ESD SENSITIVITY

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

Table 19. Power-Up Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
$t_{RST\_IN\_PWR}$	$\overline{\text{RESET}}$ Deasserted after the V_{DDINT}, V_{DDEXT}, and CLKIN Pins are Stable and within Specification	$3500 \times t_{CKIN}$		µs

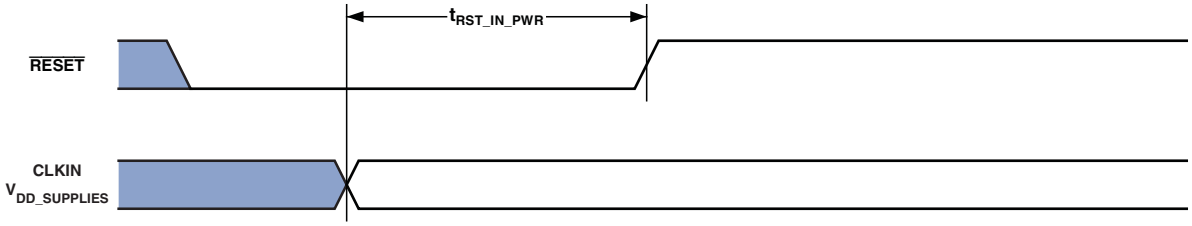


Figure 8. Power-Up Reset Timing

Parallel Peripheral Interface Timing

Table 20 and Figure 9 through Figure 13 describe parallel peripheral interface operations.

Table 20. Parallel Peripheral Interface Timing

Parameter		V_{DDEXT} = 1.8 V Min	V_{DDEXT} = 1.8 V Max	V_{DDEXT} = 2.5 V/3.3 V Min	V_{DDEXT} = 2.5 V/3.3 V Max	Unit
Timing Requirements						
t_{PCLKW}	PPI_CLK Width[1]	t_{SCLK} – 1.5		t_{SCLK} – 1.5		ns
t_{PCLK}	PPI_CLK Period[1]	2 × t_{SCLK} – 1.5		2 × t_{SCLK} – 1.5		ns
Timing Requirements—GP Input and Frame Capture Modes						
t_{PSUD}	External Frame Sync Startup Delay[2]	4 × t_{PCLK}		4 × t_{PCLK}		ns
t_{SFSPE}	External Frame Sync Setup Before PPI_CLK (Nonsampling Edge for Rx, Sampling Edge for Tx)	6.7		6.7		ns
t_{HFSPE}	External Frame Sync Hold After PPI_CLK	1.8		1.6		ns
t_{SDRPE}	Receive Data Setup Before PPI_CLK	4.1		3.5		ns
t_{HDRPE}	Receive Data Hold After PPI_CLK	2		1.6		ns
Switching Characteristics—GP Output and Frame Capture Modes						
t_{DFSPE}	Internal Frame Sync Delay After PPI_CLK		9.0		8.0	ns
t_{HOFSPE}	Internal Frame Sync Hold After PPI_CLK	1.7		1.7		ns
t_{DDTPE}	Transmit Data Delay After PPI_CLK		8.7		8.0	ns
t_{HDTPE}	Transmit Data Hold After PPI_CLK	2.3		1.9		ns

[1] PPI_CLK frequency cannot exceed $f_{SCLK}/2$.
[2] The PPI port is fully enabled 4 PPI clock cycles after the PAB write to the PPI port enable bit. Only after the PPI port is fully enabled are external frame syncs and data words guaranteed to be received correctly by the PPI peripheral.

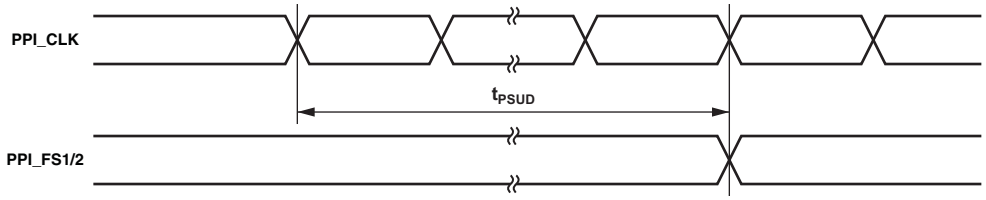


Figure 9. PPI with External Frame Sync Timing

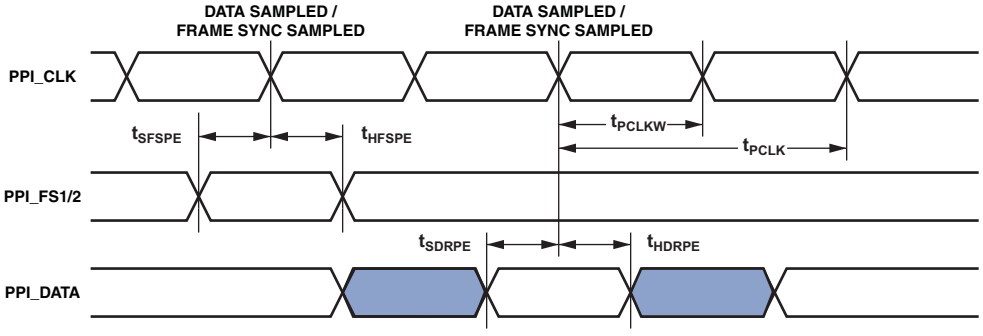


Figure 10. PPI GP Rx Mode with External Frame Sync Timing

Serial Ports

Table 21 through Table 25 and Figure 14 through Figure 18 describe serial port operations.

Table 21. Serial Ports—External Clock

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSE}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	3		3		ns
t_{HFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	3		3		ns
t_{SDRE}	Receive Data Setup Before RSCLKx[1]	3		3		ns
t_{HDRE}	Receive Data Hold After RSCLKx[1]	3.5		3		ns
t_{SCLKEW}	TSCLKx/RSCLKx Width	4.5		4.5		ns
t_{SCLKE}	TSCLKx/RSCLKx Period	$2 \times t_{SCLK}$		$2 \times t_{SCLK}$		ns
t_{SUDTE}	Start-Up Delay From SPORT Enable To First External TFSx[2]	$4 \times t_{TSCLKE}$		$4 \times t_{TSCLKE}$		ns
t_{SUDRE}	Start-Up Delay From SPORT Enable To First External RFSx[2]	$4 \times t_{RSCLKE}$		$4 \times t_{RSCLKE}$		ns
Switching Characteristics						
t_{DFSE}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[3]		10		10	ns
t_{HOFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[1]	0		0		ns
t_{DDTE}	Transmit Data Delay After TSCLKx[1]		11		10	ns
t_{HDTE}	Transmit Data Hold After TSCLKx[1]	0		0		ns

[1] Referenced to sample edge.
[2] Verified in design but untested.
[3] Referenced to drive edge.

Table 22. Serial Ports—Internal Clock

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSI}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	11.5		9.6		ns
t_{HFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	–1.5		–1.5		ns
t_{SDRI}	Receive Data Setup Before RSCLKx[1]	11.5		11.3		ns
t_{HDRI}	Receive Data Hold After RSCLKx[1]	–1.5		–1.5		ns
Switching Characteristics						
t_{SCLKIW}	TSCLKx/RSCLKx Width	7		8		ns
t_{DFSI}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[2]		4		3	ns
t_{HOFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[1]	–2		–2		ns
t_{DDTI}	Transmit Data Delay After TSCLKx[1]		4		3	ns
t_{HDTI}	Transmit Data Hold After TSCLKx[1]	–1.8		–1.5		ns

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

Table 25. Serial Ports—Gated Clock Mode

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{SDRI}	Receive Data Setup Before TSCLKx	11.3		8.7		ns
t_{HDRI}	Receive Hold After TSCLKx	0		0		ns
Switching Characteristics						
t_{DDTI}	Transmit Data Delay After TSCLKx		3		3	ns
t_{HDTI}	Transmit Data Hold After TSCLKx	–1.8		–1.8		ns
$t_{DFTSCLKCNV}$	First TSCLKx edge delay after TFSx/TMR1 Low	$0.5 \times t_{TSCLK} - 3$		$0.5 \times t_{TSCLK} - 3$		ns
$t_{DCNVLTSCLK}$	TFSx/TMR1 High Delay After Last TSCLKx Edge	$t_{TSCLK} - 3$		$t_{TSCLK} - 3$		ns

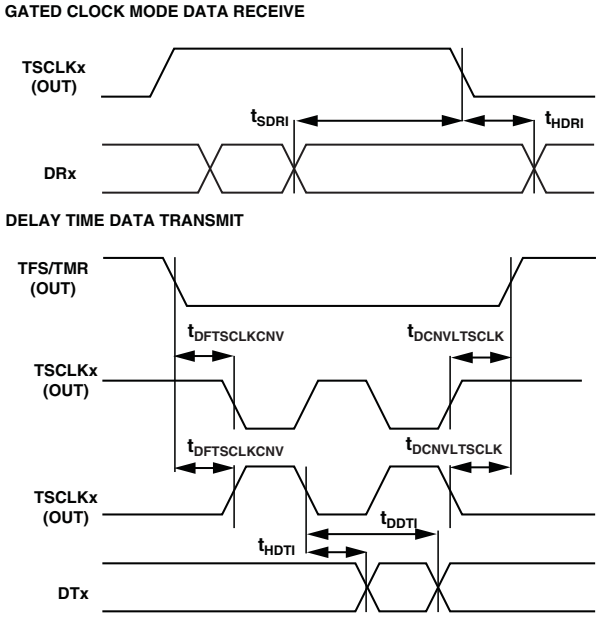


Figure 18. Serial Ports Gated Clock Mode

Serial Peripheral Interface (SPI) Port—Master Timing

Table 26 and Figure 19 describe SPI port master operations.

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

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{SSPIDM}	Data Input Valid to SCK Edge (Data Input Setup)	11.6		9.6		ns
t_{HSPIDM}	SCK Sampling Edge to Data Input Invalid	–1.5		–1.5		ns
Switching Characteristics						
t_{SDSCIM}	$\overline{SPI\_SELx}$ low to First SCK Edge	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICHM}	Serial Clock High Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLM}	Serial Clock Low Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLK}	Serial Clock Period	$4 \times t_{SCLK} - 1.5$		$4 \times t_{SCLK} - 1.5$		ns
t_{HDSM}	Last SCK Edge to $\overline{SPI\_SELx}$ High	$2 \times t_{SCLK} - 2$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPITDM}	Sequential Transfer Delay	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
$t_{DDSPIDM}$	SCK Edge to Data Out Valid (Data Out Delay)	0	6	0	6	ns
$t_{HDSPIDM}$	SCK Edge to Data Out Invalid (Data Out Hold)	–1		–1		ns

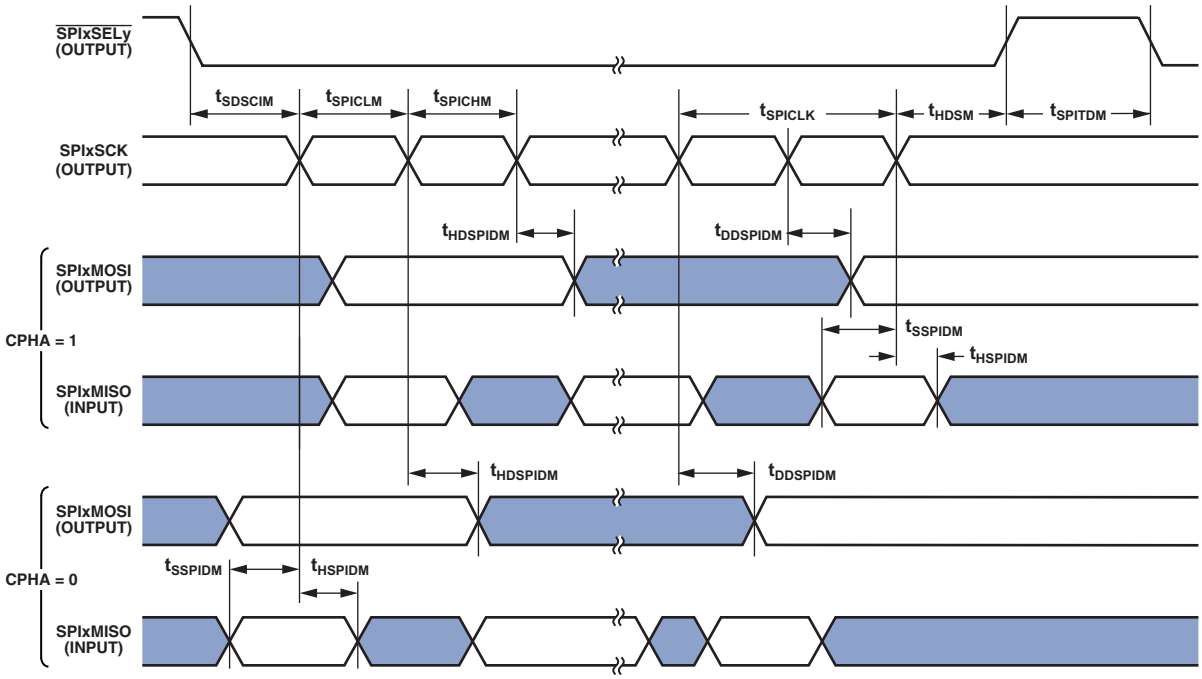


Figure 19. Serial Peripheral Interface (SPI) Port—Master Timing

Serial Peripheral Interface (SPI) Port—Slave Timing

Table 27 and Figure 20 describe SPI port slave operations.

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

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{SPICHS}	Serial Clock High Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLS}	Serial Clock Low Period	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPICLK}	Serial Clock Period	$4 \times t_{SCLK}$		$4 \times t_{SCLK}$		ns
t_{HDS}	Last SCK Edge to $\overline{SPI\_SS}$ Not Asserted	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SPITDS}	Sequential Transfer Delay	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SDSCI}	$\overline{SPI\_SS}$ Assertion to First SCK Edge	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
t_{SSPID}	Data Input Valid to SCK Edge (Data Input Setup)	1.6		1.6		ns
t_{HSPID}	SCK Sampling Edge to Data Input Invalid	2		1.6		ns
Switching Characteristics						
t_{DSOE}	$\overline{SPI\_SS}$ Assertion to Data Out Active	0	12	0	10.3	ns
t_{DSDHI}	$\overline{SPI\_SS}$ Deassertion to Data High Impedance	0	11	0	9	ns
t_{DDSPID}	SCK Edge to Data Out Valid (Data Out Delay)		10		10	ns
t_{HDSPID}	SCK Edge to Data Out Invalid (Data Out Hold)	0		0		ns

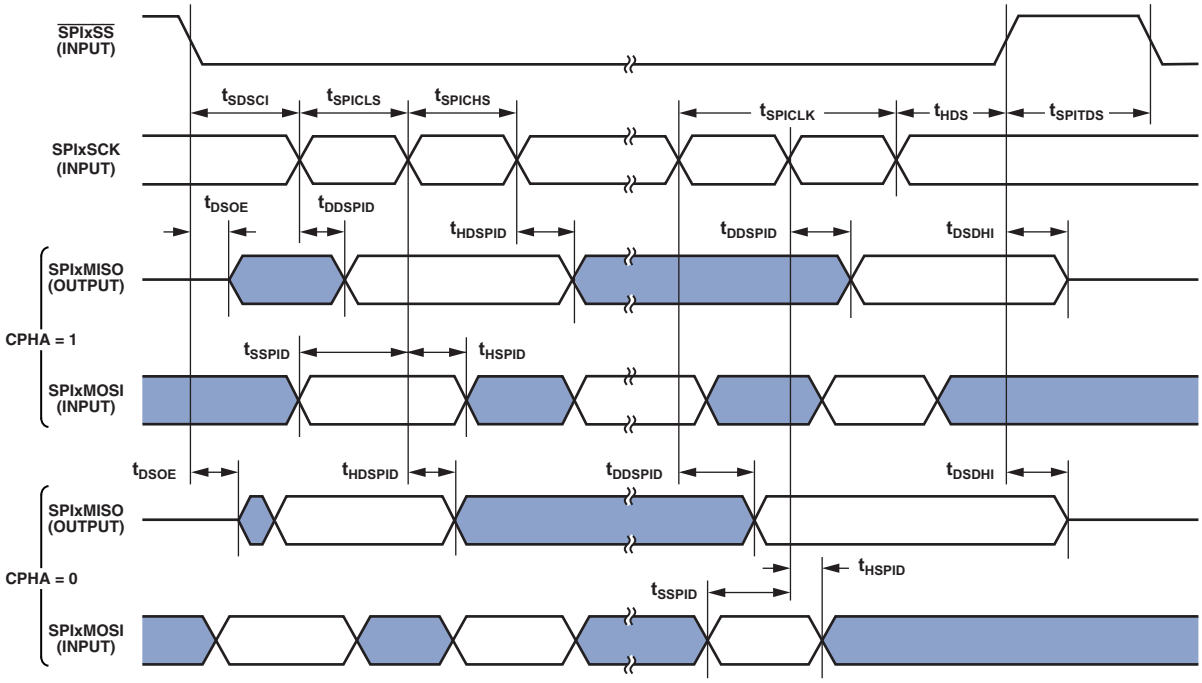


Figure 20. Serial Peripheral Interface (SPI) Port—Slave Timing

JTAG Test And Emulation Port Timing

Table 31 and Figure 24 describe JTAG port operations.

Table 31. JTAG Port Timing

Parameter		V_{DDEXT} 1.8 V Nominal Min	V_{DDEXT} 1.8 V Nominal Max	V_{DDEXT} 2.5 V/3.3 V Nominal Min	V_{DDEXT} 2.5 V/3.3 V Nominal Max	Unit
Timing Requirements						
t_{TCK}	TCK Period	20		20		ns
t_{STAP}	TDI, TMS Setup Before TCK High	4		4		ns
t_{HTAP}	TDI, TMS Hold After TCK High	4		4		ns
t_{SSYS}	System Inputs Setup Before TCK High[1]	4		5		ns
t_{HSYS}	System Inputs Hold After TCK High[1]	5		5		ns
t_{TRSTW}	$\overline{TRST}$ Pulse Width[2] (measured in TCK cycles)	4		4		TCK
Switching Characteristics						
t_{DTDO}	TDO Delay from TCK Low		10		10	ns
t_{DSYS}	System Outputs Delay After TCK Low[3]		13		13	ns

[1] System inputs = SCL, SDA, PF15–0, PG15–0, PH2–0, TCK, $\overline{NMI}$, BMODE3–0, $\overline{PG}$.
[2] 50 MHz maximum.
[3] System outputs = CLKOUT, SCL, SDA, PF15–0, PG15–0, PH2–0, TDO, $\overline{EMU}$, EXT_WAKE.

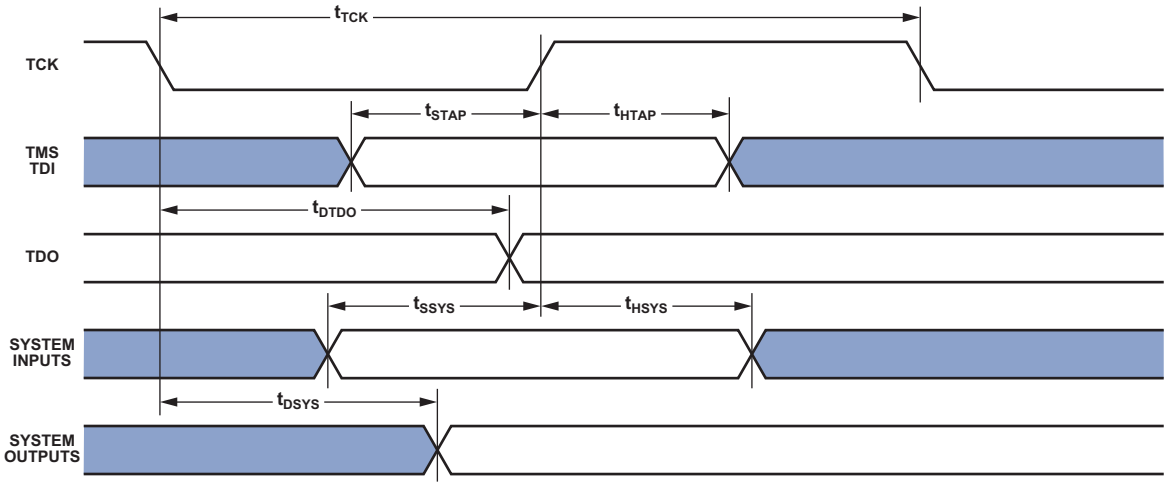


Figure 24. JTAG Port Timing

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) and is shown below.

- V_{DDEXT} (nominal) = 1.8 V, V_{TRIP} (high) is 1.05 V, V_{TRIP} (low) is 0.75 V
- V_{DDEXT} (nominal) = 2.5 V, V_{TRIP} (high) is 1.5 V, V_{TRIP} (low) is 1.0 V
- V_{DDEXT} (nominal) = 3.3 V, V_{TRIP} (high) is 1.9 V, V_{TRIP} (low) is 1.4 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 are enabled, the measurement value is that of the first lead 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 35.

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

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 0.25 V for V_{DDEXT} (nominal) = 2.5 V/3.3 V and 0.15 V for V_{DDEXT} (nominal) = 1.8V.

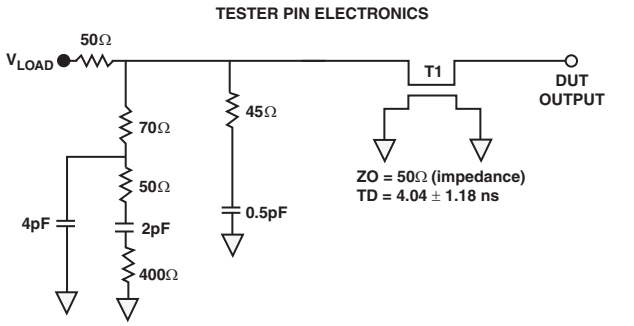
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches to when the output voltage decays ΔV from the measured output high or output low voltage.

Example System Hold Time Calculation

To determine the data output hold time in a particular system, first calculate t_{DECAY} using the equation given above. Choose ΔV to be the difference between the processor's output voltage and the input threshold for the device requiring the hold time. C_L is the total bus capacitance (per data line), and I_L is the total leakage or three-state current (per data line). The hold time will be t_{DECAY} plus the various output disable times as specified in the Timing Specifications on Page 22.

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all pins (see Figure 36). V_{LOAD} is equal to (V_{DDEXT})/2.



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

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

Figure 36. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

The graphs of Figure 37 through Figure 42 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.

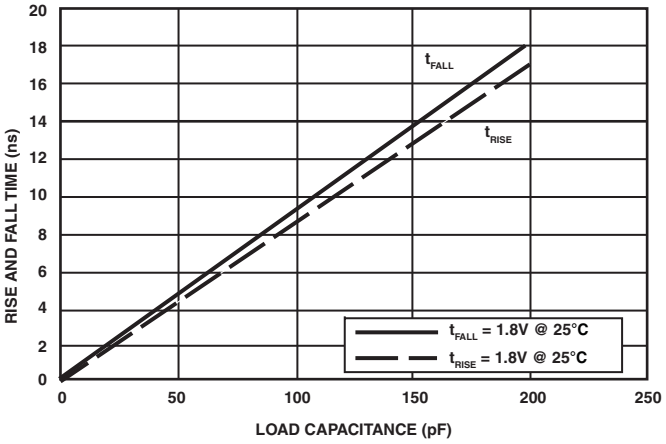


Figure 37. Driver Type A Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (1.8V V_{DDEXT})

OUTLINE DIMENSIONS

Dimensions in Figure 45 are shown in millimeters.

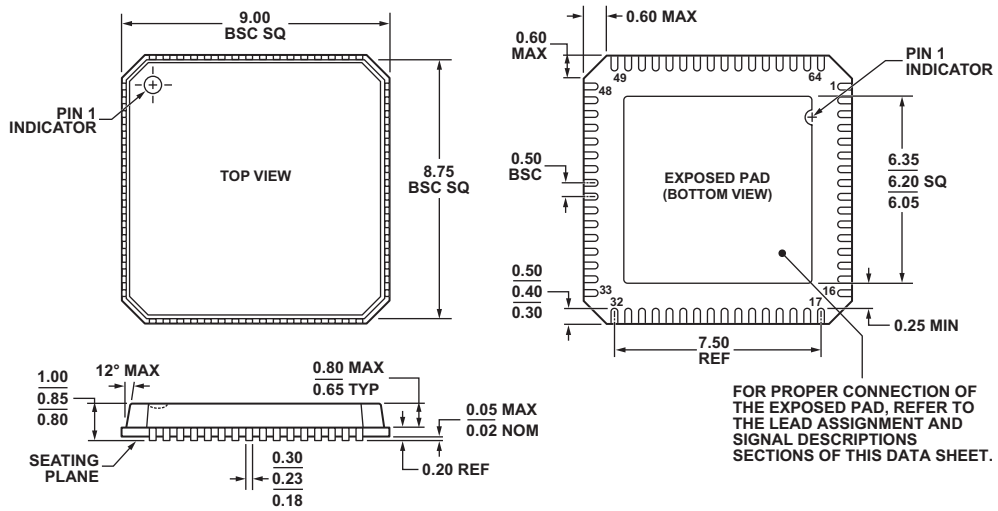


Figure 45. 64-Lead Lead Frame Chip Scale Package [LFCSP_VQ[1]]
Very Thin Quad (CP-64-4)
Dimensions shown in millimeters

[1] For information relating to the CP-64-4 package's exposed pad, see the table endnotes on Page 41.