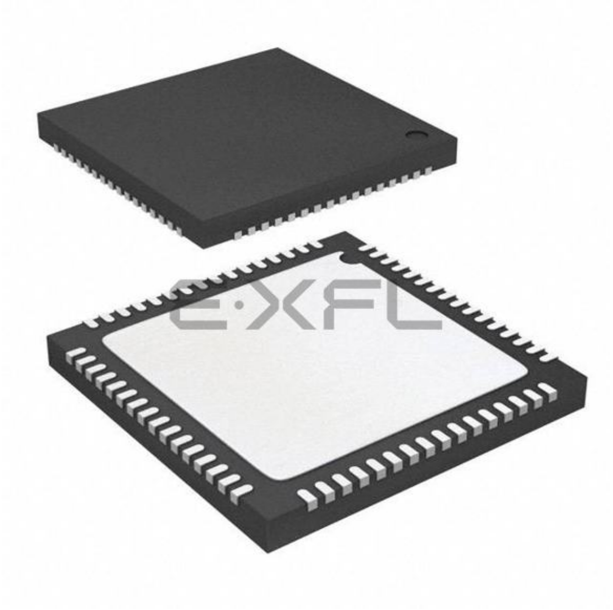
Analog Devices Inc. - ADSP-BF592BCPZ Datasheet



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	400MHz
Non-Volatile Memory	ROM (64kB)
On-Chip RAM	64kB
Voltage - I/O	1.8V, 2.5V, 3.3V
Voltage - Core	1.29V
Operating Temperature	-40°C ~ 85°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-bf592bcpz

ADSP-BF592* PRODUCT PAGE QUICK LINKS

Last Content Update: 02/23/2017

COMPARABLE PARTS

View a parametric search of comparable parts.

EVALUATION KITS

- The ADSP-BF592 EZ-Kit Lite evaluation hardware provides a low-cost hardware solution for evaluating the ADSP-BF59x Blackfin processor family.
- USB-Based Emulator and High Performance USB-Based Emulator

DOCUMENTATION

Application Notes

- EE-104: Setting Up Streams with the VisualDSP Debugger
- EE-112: Class Implementation in Analog C++
- EE-120: Interfacing Assembly Language Programs to C
- EE-126: The ABCs of SDRAMemories
- EE-128: DSP in C++: Calling Assembly Class Member Functions From C++
- EE-149: Tuning C Source Code for the Blackfin® Processor Compiler
- EE-159: Initializing DSP System & Control Registers From C and C++
- EE-175: Emulator and Evaluation Hardware Troubleshooting Guide for VisualDSP++ Users
- EE-183: Rational Sample Rate Conversion with Blackfin® Processors
- EE-185: Fast Floating-Point Arithmetic Emulation on Blackfin® Processors
- EE-192: Using C To Create Interrupt-Driven Systems On Blackfin® Processors
- EE-197: ADSP-BF531/532/533 Blackfin® Processor Multi-cycle Instructions and Latencies
- EE-204: Blackfin® Processor SCCB Software Interface for Configuring I2C® Slave Devices
- EE-210: SDRAM Selection and Configuration Guidelines for ADI Processors
- EE-213: Host Communication via the Asynchronous Memory Interface for Blackfin® Processors
- EE-234: Interfacing T1/E1 Transceivers/Framers to Blackfin® Processors via the Serial Port
- EE-235: An Introduction to Scripting in VisualDSP++®
- EE-236: Real-Time Solutions Using Mixed-Signal Front-End Devices with the Blackfin® Processor
- EE-257: A Boot Compression/Decompression Algorithm for Blackfin® Processors
- EE-261: Understanding Jitter Requirements of PLL-Based Processors
- EE-269: A Beginner’s Guide to Ethernet 802.3
- EE-271: Using Cache Memory on Blackfin® Processors
- EE-273: Using the VisualDSP++ Command-Line Installer
- EE-281: Hardware Design Checklist for the Blackfin® Processors

DESIGN RESOURCES

- ADSP-BF592 Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADSP-BF592 EngineerZone Discussions.

SAMPLE AND BUY

Visit the product page to see pricing options.

TECHNICAL SUPPORT

Submit a technical question or find your regional support number.

DOCUMENT FEEDBACK

Submit feedback for this data sheet.

This page is dynamically generated by Analog Devices, Inc., and inserted into this data sheet. A dynamic change to the content on this page will not trigger a change to either the revision number or the content of the product data sheet. This dynamic page may be frequently modified.

GENERAL DESCRIPTION

The ADSP-BF592 processor is a member of the Blackfin® family of products, incorporating the Analog Devices/Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like microprocessor instruction set, and single-instruction, multiple-data (SIMD) multimedia capabilities into a single instruction-set architecture.

The ADSP-BF592 processor is completely code compatible with other Blackfin processors. The ADSP-BF592 processor offers performance up to 400 MHz and reduced static power consumption. The processor features are shown in Table 1.

Table 1. Processor Features

Feature		ADSP-BF592
Timer/Counters with PWM		3
SPORTs		2
SPIs		2
UART		1
Parallel Peripheral Interface		1
TWI		1
GPIOs		32
Memory (bytes)	L1 Instruction SRAM	32K
	L1 Instruction ROM	64K
	L1 Data SRAM	32K
	L1 Scratchpad SRAM	4K
	L3 Boot ROM	4K
Maximum Instruction Rate[1]		400 MHz
Maximum System Clock Speed		100 MHz
Package Options		64-Lead LFCSP

[1] Maximum instruction rate is not available with every possible SCLK selection.

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

PORTABLE LOW POWER ARCHITECTURE

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

SYSTEM INTEGRATION

The ADSP-BF592 processor is a highly integrated system-on-a-chip solution for the next generation of digital communication and consumer multimedia applications. By combining industry standard interfaces with a high performance signal processing core, cost-effective applications can be developed quickly, without the need for costly external components. The system peripherals include a watchdog timer; three 32-bit timers/counters with PWM support; two dual-channel, full-duplex synchronous serial ports (SPORTs); two serial peripheral interface (SPI) compatible ports; one UART® with IrDA support; a parallel peripheral interface (PPI); and a 2-wire interface (TWI) controller.

BLACKFIN PROCESSOR CORE

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

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

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 includes 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. The compare/select and vector search instructions are also provided.

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

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

The program sequencer controls the flow of instruction execution, including instruction alignment and decoding. For program flow control, the sequencer supports PC relative and indirect conditional jumps (with static branch prediction) and subroutine calls. Hardware is provided to support zero over 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.

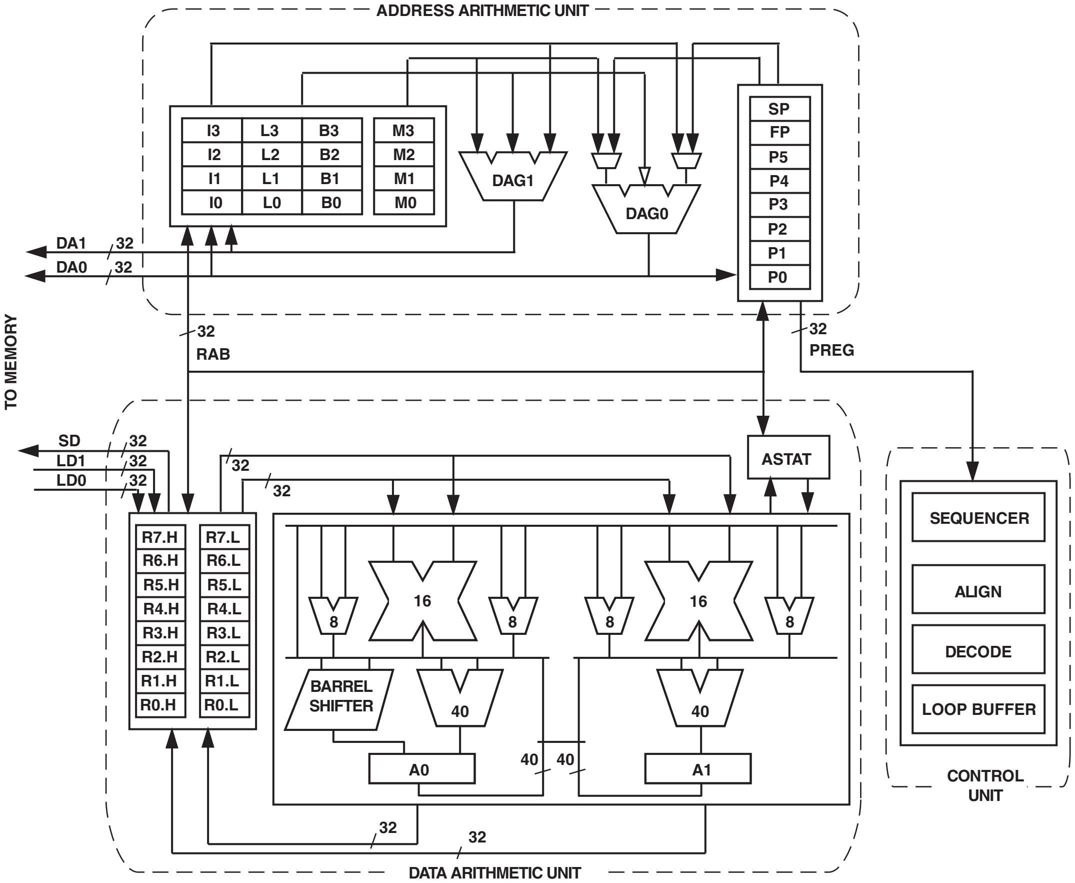


Figure 2. Blackfin Processor Core

- Exceptions – Events that occur synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions such as data alignment violations and undefined instructions cause exceptions.
- Interrupts – Events that occur asynchronously to program flow. They are caused by input signals, timers, and other peripherals, as well as by an explicit software instruction.

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

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

Core Event Controller (CEC)

The CEC supports nine general-purpose interrupts (IVG15–7), in addition to the dedicated interrupt and exception events. Of these general-purpose interrupts, the two lowest priority interrupts (IVG15–14) are recommended to be reserved for software interrupt handlers, leaving seven prioritized interrupt inputs to support the peripherals of the processor. The inputs to the CEC, their names in the event vector table (EVT), and their priorities are described in the *ADSP-BF59x Blackfin Processor Hardware Reference*, “System Interrupts” chapter.

System Interrupt Controller (SIC)

The system interrupt controller provides the mapping and routing of events from the many peripheral interrupt sources to the prioritized general-purpose interrupt inputs of the CEC. Although the processor provides a default mapping, the user can alter the mappings and priorities of interrupt events by writing the appropriate values into the interrupt assignment registers (SIC_IARx). The inputs into the SIC and the default mappings into the CEC are described in the *ADSP-BF59x Blackfin Processor Hardware Reference*, “System Interrupts” chapter.

The SIC allows further control of event processing by providing three pairs of 32-bit interrupt control and status registers. Each register contains a bit, corresponding to each peripheral interrupt event. For more information, see the *ADSP-BF59x Blackfin Processor Hardware Reference*, “System Interrupts” chapter.

DMA CONTROLLERS

The processor has multiple, independent DMA channels that support automated data transfers with minimal overhead for the processor core. DMA transfers can occur between the processor’s internal memories and any of its DMA-capable peripherals. DMA-capable peripherals include the SPORTs, SPI ports, UART, and PPI. Each individual DMA-capable peripheral has at least one dedicated DMA channel.

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

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

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

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

In addition to the dedicated peripheral DMA channels, there are two memory DMA channels, which are provided for transfers between the various memories of the processor system 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.

PROCESSOR PERIPHERALS

The ADSP-BF592 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 Figure 1). The processor also contains dedicated communication modules and high speed serial and parallel ports, an interrupt controller for flexible management of interrupts from the on-chip peripherals or external sources, and power management control functions to tailor the performance and power characteristics of the processor and system to many application scenarios.

The SPORTs, SPIs, UART, and PPI peripherals are supported by a flexible DMA structure. There are also separate memory DMA channels dedicated to data transfers between the processor’s various memory spaces, including boot ROM. Multiple on-chip buses running at up to 100 MHz provide enough bandwidth to keep the processor core running along with activity on all of the on-chip and external peripherals.

The ADSP-BF592 processor includes an interface to an off-chip voltage regulator in support of the processor’s dynamic power management capability.

Watchdog Timer

The processor includes a 32-bit timer that can be used to implement a software watchdog function. A software watchdog can improve system availability by forcing the processor to a known state through generation of a hardware reset, nonmaskable interrupt (NMI), or general-purpose interrupt, if the timer expires before being reset by software. The programmer

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 I^{2}C® 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

Table 7. Signal Descriptions (Continued)

Signal Name	Type	Function	Driver Type
PG10–GPIO/*SPI1_MISO/PPI_D2*	I/O	GPIO/*SPI1 Master In Slave Out/PPI Data 2* (This pin should always be pulled high through a 4.7 kΩ resistor if booting via the SPI port.)	A
PG11–GPIO/*$\overline{\text{SPI1\_SSEL5}}$/PPI_D3*	I/O	GPIO/*SPI1 Slave Select Enable 5/PPI Data 3*	A
PG12–GPIO/*$\overline{\text{SPI1\_SSEL2}}$/PPI_D4/WAKEN2*	I/O	GPIO/*SPI1 Slave Select Enable 2 Output/PPI Data 4/Wake Enable 2*	A
PG13–GPIO/*$\overline{\text{SPI1\_SSEL1}}$/$\overline{\text{SPI1\_SS}}$/PPI_D5*	I/O	GPIO/*SPI1 Slave Select Enable 1 Output/PPI Data 5/SPI1 Slave Select Input*	A
PG14–GPIO/*$\overline{\text{SPI1\_SSEL4}}$/PPI_D6/TACLK1*	I/O	GPIO/*SPI1 Slave Select Enable 4/PPI Data 6/Timer 1 Auxiliary Clock Input*	A
PG15–GPIO/*$\overline{\text{SPI1\_SSEL6}}$/PPI_D7/TACLK2*	I/O	GPIO/*SPI1 Slave Select Enable 6/PPI Data 7/Timer 2 Auxiliary Clock Input*	A
TWI			
SCL	I/O	TWI Serial Clock (This signal is an open-drain output and requires a pull-up resistor. Consult version 2.1 of the I^{2}C specification for the proper resistor value.)	B
SDA	I/O	TWI Serial Data (This signal is an open-drain output and requires a pull-up resistor. Consult version 2.1 of the I^{2}C specification for the proper resistor value.)	B
JTAG Port			
TCK	I	JTAG CLK	
TDO	O	JTAG Serial Data Out	A
TDI	I	JTAG Serial Data In	
TMS	I	JTAG Mode Select	
$\overline{\text{TRST}}$	I	JTAG Reset (This lead should be pulled low if the JTAG port is not used.)	
$\overline{\text{EMU}}$	O	Emulation Output	A
Clock			
CLKIN	I	CLK/Crystal In	
XTAL	O	Crystal Output	
EXTCLK	O	External Clock Output pin/System Clock Output	C
Mode Controls			
$\overline{\text{RESET}}$	I	Reset	
$\overline{\text{NMI}}$	I	Nonmaskable Interrupt (This lead should be pulled high when not used.)	
BMODE2–0	I	Boot Mode Strap 2–0	
PPI_CLK	I	PPI Clock Input	
External Regulator Control			
$\overline{\text{PG}}$	I	Power Good indication	
EXT_WAKE	O	Wake up Indication	A
Power Supplies		**ALL SUPPLIES MUST BE POWERED** See Operating Conditions on Page 16.	
V_{DDEXT}	P	I/O Power Supply	
V_{DDINT}	P	Internal Power Supply	
GND	G	Ground for All Supplies (Back Side of LFCSP Package.)	

ADSP-BF592 Clock Related Operating Conditions

Table 8 describes the core clock timing requirements for the ADSP-BF592 processor. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock and system clock (see Table 10). Table 9 describes phase-locked loop operating conditions.

Table 8. Core Clock (CCLK) Requirements

Parameter		Min V_{DDINT}	Nom V_{DDINT}	Max CCLK Frequency	Unit
f_{CCLK}	Core Clock Frequency (All Models)	1.33 V	1.400 V	400	MHz
	Core Clock Frequency (Industrial/Commercial Models)	1.16 V	1.225 V	300	MHz
	Core Clock Frequency (Industrial/Commercial Models)	1.10 V	1.150 V	250[1]	MHz

[1] See the Ordering Guide on Page 44.

Table 9. Phase-Locked Loop Operating Conditions

Parameter		Min	Max	Unit
f_{VCO}	Voltage Controlled Oscillator (VCO) Frequency (Non-Automotive Models)	72	Instruction Rate[1]	MHz
	Voltage Controlled Oscillator (VCO) Frequency (Automotive Models)	84	Instruction Rate[1]	MHz

[1] See the Ordering Guide on Page 44.

Table 10. Maximum SCLK Conditions

Parameter[1]		V_{DDEXT} 1.8 V/2.5 V/3.3 V Nominal	Unit
f_{SCLK}	CLKOUT/SCLK Frequency ($V_{DDINT} \geq 1.16$ V)	100	MHz
	CLKOUT/SCLK Frequency ($V_{DDINT} < 1.16$ V)	80	MHz

[1] f_{SCLK} must be less than or equal to f_{CCLK}.

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, $\overline{\text{TRST}}$, TMS
10	BMODE2, BMODE1, BMODE0
11	EXT_WAKE, $\overline{\text{PG}}$, $\overline{\text{RESET}}$, $\overline{\text{NMI}}$, 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.

PACKAGE INFORMATION

The information presented in Figure 6 and Table 17 provides details about the package branding for the ADSP-BF592 processor. For a complete listing of product availability, see Ordering Guide on Page 44.

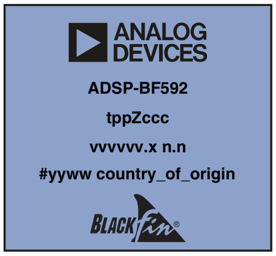


Figure 6. Product Information on Package

Table 17. Package Brand Information

Brand Key	Field Description
ADSP-BF592	Product Name
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Designation
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
#	RoHS Compliance Designator
yyww	Date Code

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	 Max	V_{DDEXT} = 2.5 V/3.3 V Min	 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]	$2 \times t_{SCLK} - 1.5$		$2 \times t_{SCLK} - 1.5$		ns
Timing Requirements—GP Input and Frame Capture Modes						
t_{PSUD}	External Frame Sync Startup Delay[2]	$4 \times t_{PCLK}$		$4 \times t_{PCLK}$		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.

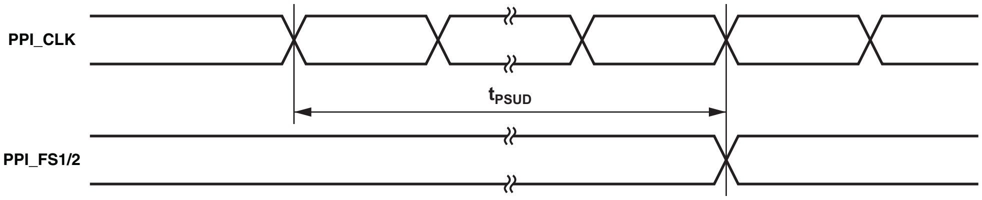


Figure 9. PPI with External Frame Sync Timing

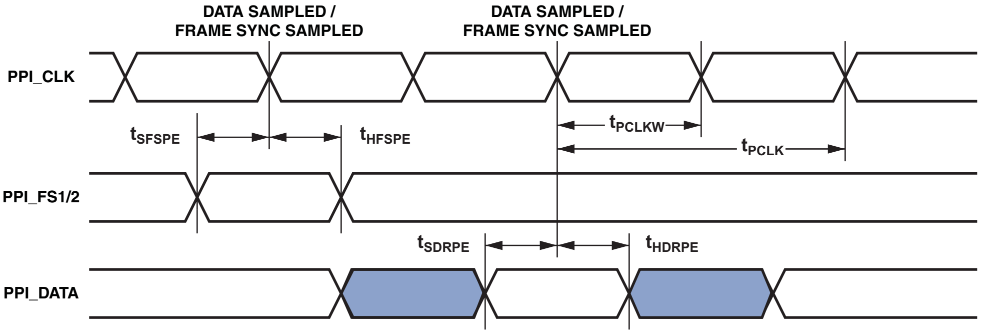


Figure 10. PPI GP Rx Mode with External Frame Sync Timing

Serial Ports

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

Table 21. Serial Ports—External Clock

		V_{DDEXT} 1.8 V Nominal		V_{DDEXT} 2.5 V/3.3 V Nominal		
Parameter		**Min**	**Max**	**Min**	**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

		V_{DDEXT} 1.8 V Nominal		V_{DDEXT} 2.5 V/3.3 V Nominal		
Parameter		**Min**	**Max**	**Min**	**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.

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}	$\overline{\text{SPI\_SELx}}$ 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 $\overline{\text{SPI\_SELx}}$ 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

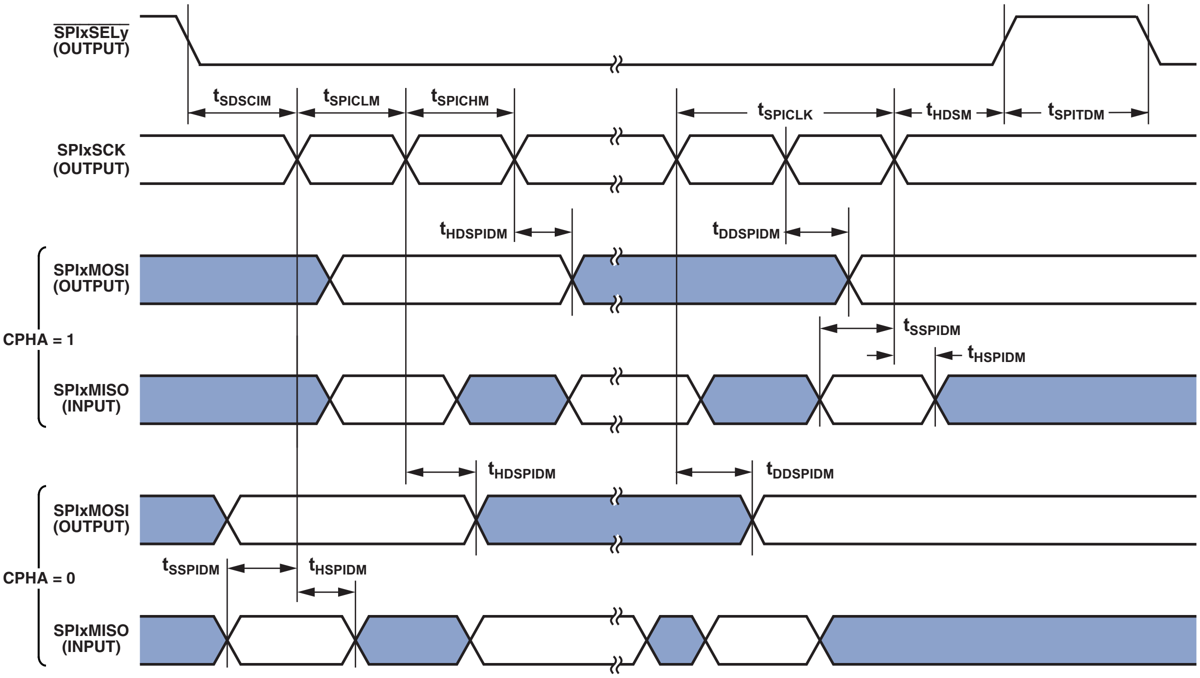


Figure 19. Serial Peripheral Interface (SPI) Port—Master Timing

Timer Cycle Timing

Table 29 and Figure 22 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 29. Timer Cycle Timing

		V_{DDEXT} 1.8 V Nominal		V_{DDEXT} 2.5 V/3.3 V Nominal		
Parameter		**Min**	**Max**	**Min**	**Max**	**Unit**
Timing Requirements						
t_{WL}	Timer Pulse Width Input Low (Measured In SCLK Cycles)[1]	1 × t_{SCLK}		1 × t_{SCLK}		ns
t_{WH}	Timer Pulse Width Input High (Measured In SCLK Cycles)[1]	1 × t_{SCLK}		1 × t_{SCLK}		ns
t_{TIS}	Timer Input Setup Time Before CLKOUT Low[2]	10		8		ns
t_{TIH}	Timer Input Hold Time After CLKOUT Low[2]	–2		–2		ns
Switching Characteristics						
t_{HTO}	Timer Pulse Width Output (Measured In SCLK Cycles)	1 × t_{SCLK} – 2	$(2^{32} – 1)$ × t_{SCLK}	t_{SCLK} – 1.5	$(2^{32} – 1)$ × t_{SCLK}	ns
t_{TOD}	Timer Output Update Delay After CLKOUT High		6		6	ns

[1] The minimum pulse widths apply for TMRx signals in width capture and external clock modes. They also apply to the PG0 or PPI_CLK signals in PWM output mode.
[2] Either a valid setup and hold time or a valid pulse width is sufficient. There is no need to resynchronize programmable flag inputs.

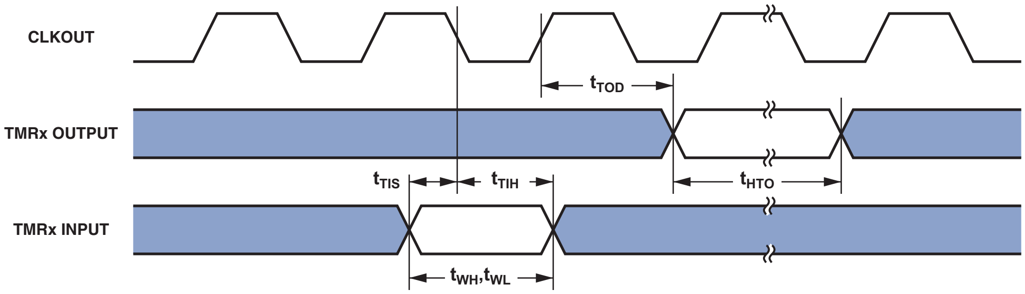


Figure 22. Timer Cycle Timing

Timer Clock Timing

Table 30 and Figure 23 describe timer clock timing.

Table 30. Timer Clock Timing

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

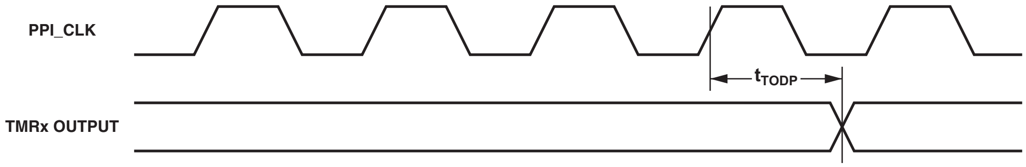


Figure 23. Timer Clock Timing

OUTPUT DRIVE CURRENTS

Figure 25 through Figure 33 show typical current-voltage characteristics for the output drivers of the ADSP-BF592 processor.

The curves represent the current drive capability of the output drivers. See Table 7 on Page 14 for information about which driver type corresponds to a particular pin.

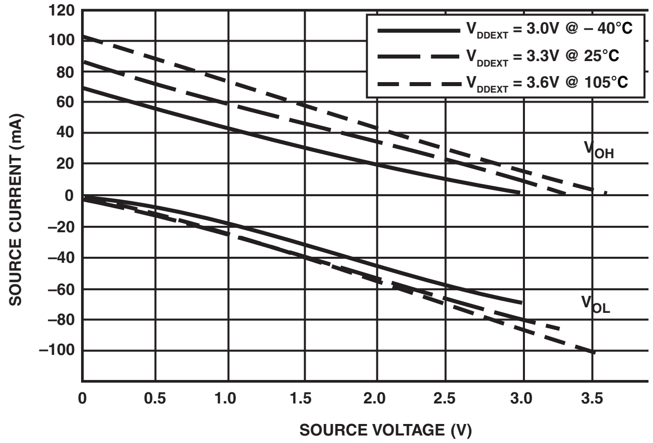


Figure 25. Driver Type A Current (3.3V V_{DDEXT})

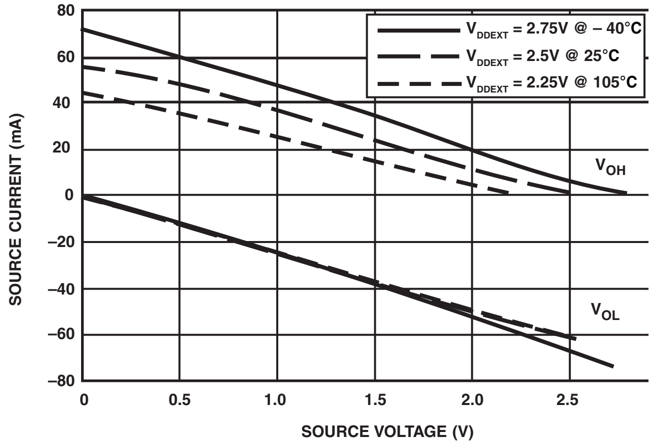


Figure 26. Drive Type A Current (2.5V V_{DDEXT})

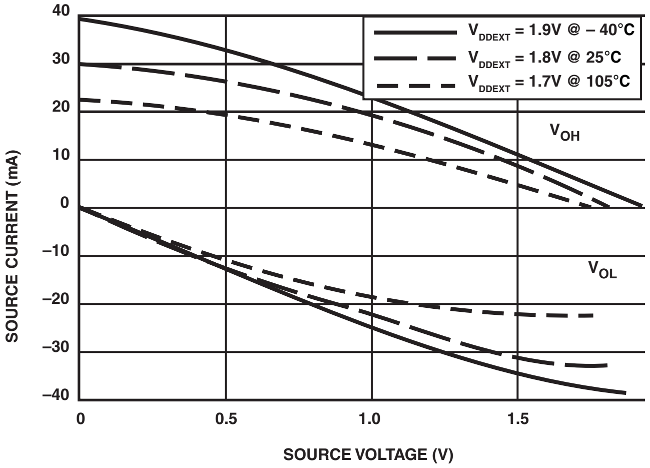


Figure 27. Driver Type A Current (1.8V V_{DDEXT})

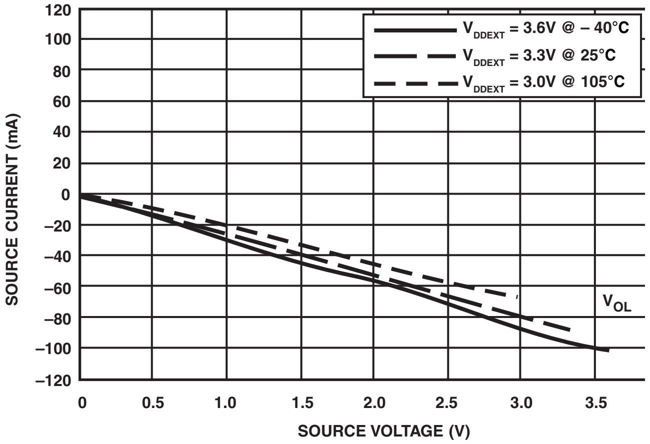


Figure 28. Driver Type B Current (3.3V V_{DDEXT})

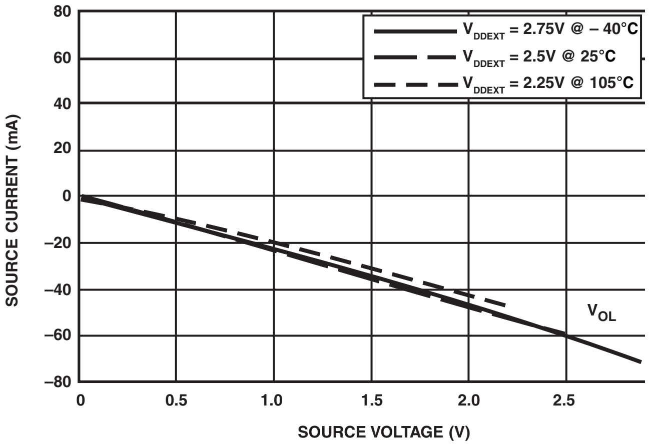


Figure 29. Driver Type B Current (2.5V V_{DDEXT})

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.

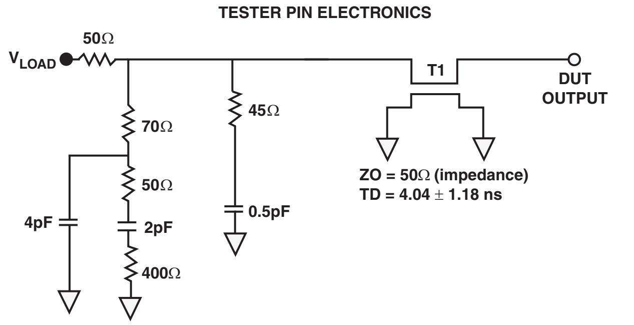
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches to when the output voltage decays ΔV from the measured output high or output low voltage.

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.

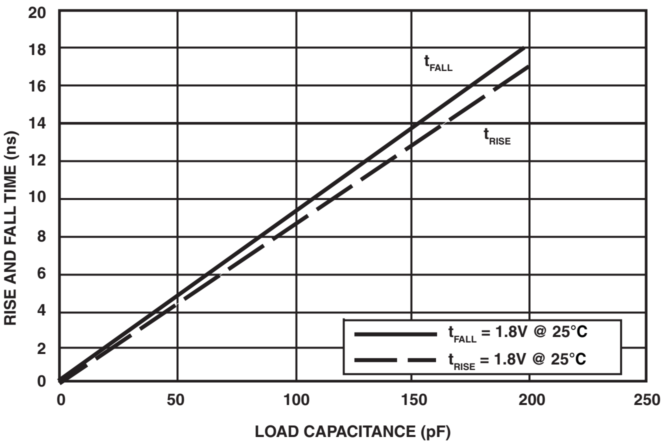


Figure 37. Driver Type A Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (1.8V V_{DDEXT})

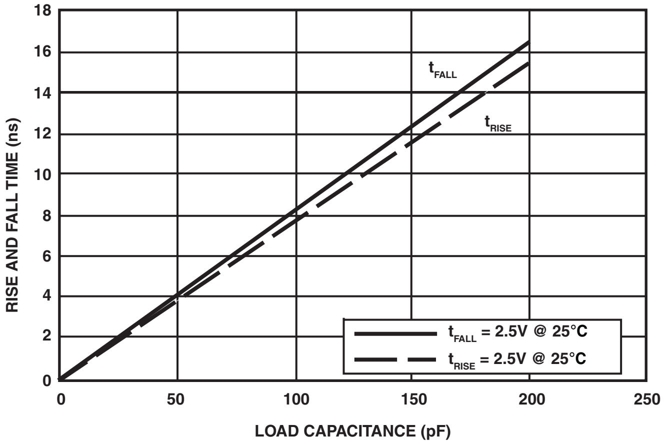


Figure 38. Driver Type A Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (2.5V V_{DDEXT})

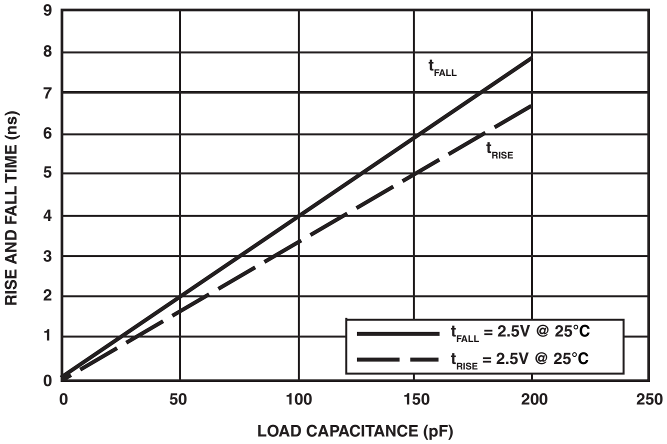


Figure 41. Driver Type C Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (2.5V V_{DDEXT})

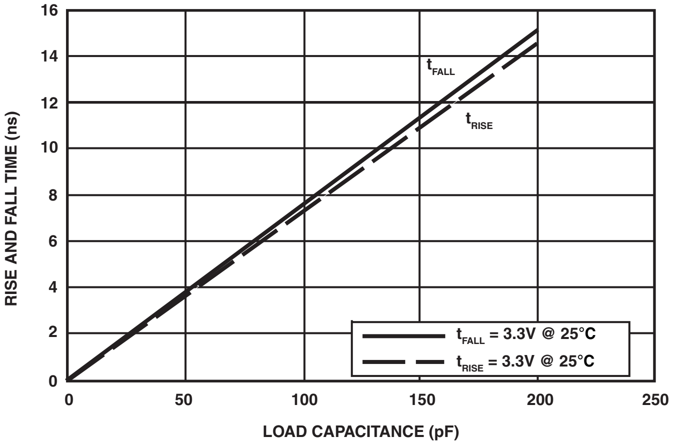


Figure 39. Driver Type A Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (3.3V V_{DDEXT})

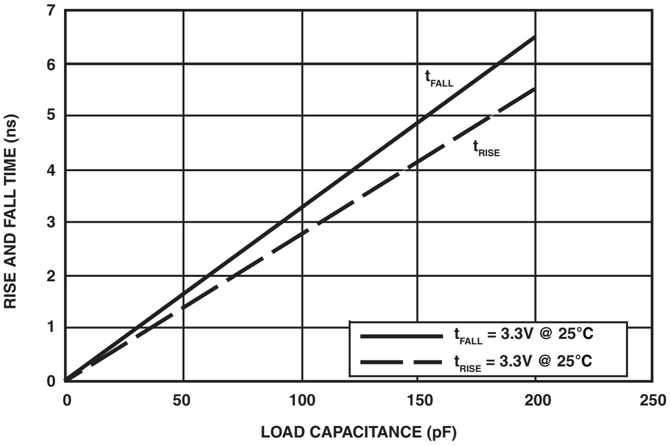


Figure 42. Driver Type C Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (3.3V V_{DDEXT})

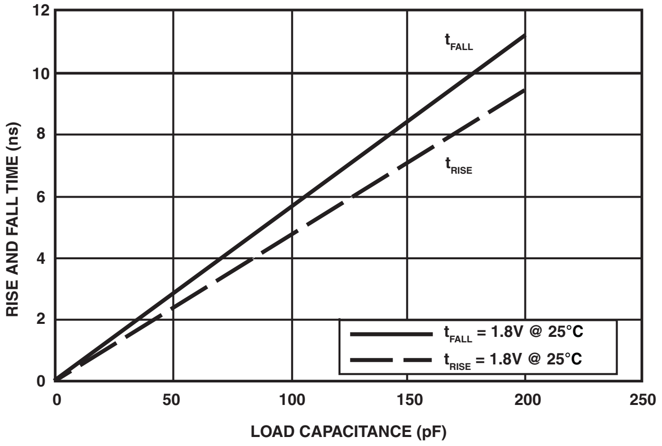


Figure 40. Driver Type C Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (1.8V V_{DDEXT})