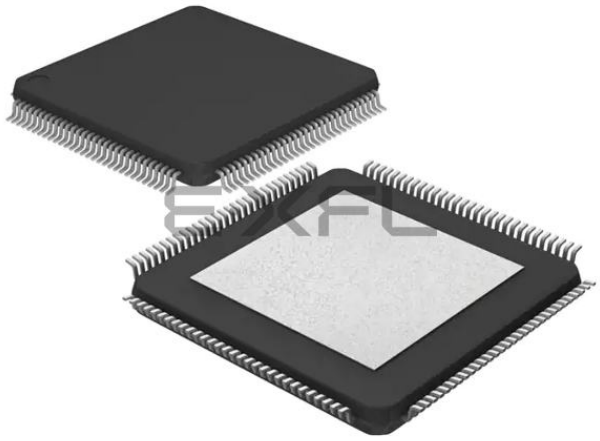


Welcome to E-XFL.COM

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	Obsolete
Type	Fixed Point
Interface	CAN, EBI/EMI, I²C, IrDA, PPI, SPI, SPORT, UART/USART
Clock Rate	400MHz
Non-Volatile Memory	FLASH (16MB)
On-Chip RAM	68kB
Voltage - I/O	3.30V
Voltage - Core	1.29V
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	120-LQFP Exposed Pad
Supplier Device Package	120-LQFP-EP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-bf506bswz-4f

TABLE OF CONTENTS

REVISION HISTORY

04/14—Rev. A to Rev. B

Updated Development Tools 17

Corrected RCKFE bit setting and description in Table 9, The SPORTx Receive Configuration 1 Register (SPORTx_RCR1) 19

Updated footnote 6 in Operating Conditions 26

Updated Table 18 with revised data for Static Current—IDD-DEEPSLEEP (mA) 30

Revised package diagram (Figure 93) to include U-Groove in Outline Dimensions 79

Package thickness changed from 0.75/0.80/0.85 to 0.75/0.85/0.90 in Figure 94 in Outline Dimensions 79

Blackfin processors support a modified Harvard architecture in combination with a hierarchical memory structure. Level 1 (L1) memories are those that typically operate at the full processor speed with little or no latency. At the L1 level, the instruction memory holds instructions only. The data memory holds data, and a dedicated scratchpad data memory stores stack and local variable information.

In addition, multiple L1 memory blocks are provided, offering a configurable mix of SRAM and cache. The memory management unit (MMU) provides memory protection for individual tasks that may be operating on the core and can protect system registers from unintended access.

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

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

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

MEMORY ARCHITECTURE

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

The core-accessible L1 memory system is the highest performance memory available to the Blackfin processor. The interface-accessible memory system, accessed through the external bus interface unit (EBIU), provides access to the internal flash memory and boot ROM.

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

Internal (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, of which 16K bytes can be configured as a four-way set-associative cache. This memory is accessed at full processor speed.

The second core-accessible memory block is the L1 data memory, consisting of 32K bytes of SRAM, of which 16K bytes may be configured as cache. This memory block is accessed at full processor speed.

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

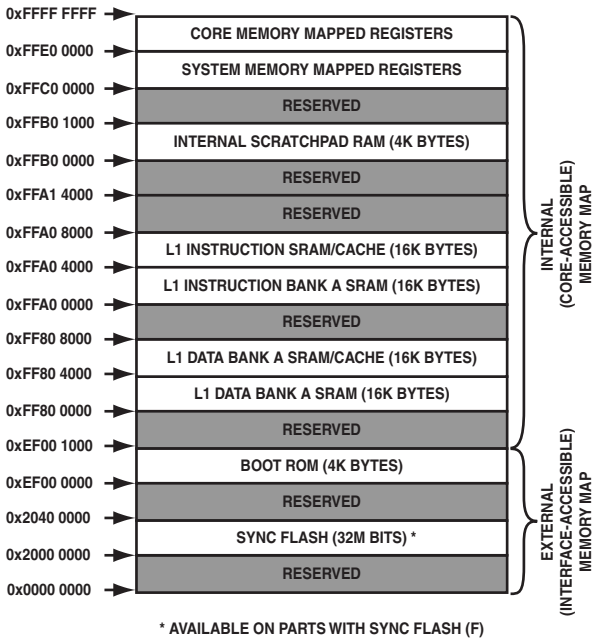


Figure 3. Internal/External Memory Map

External (Interface-Accessible) Memory

External memory is accessed via the EBIU memory port. This 16-bit interface provides a glueless connection to the internal flash memory and boot ROM. Internal flash memory ships from the factory in an erased state except for Block 0 of the parameter bank. Block 0 of the Flash memory parameter bank ships from the factory in an unknown state. An erase operation should be performed prior to programming this block.

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 contains the control MMRs for all core functions, and the other contains the registers needed for setup and control of the on-chip peripherals outside of the core. The MMRs are accessible only in supervisor and emulation modes and appear as reserved space to on-chip peripherals.

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

Peripheral Interrupt Source	General-Purpose Interrupt (at Reset)	Peripheral Interrupt ID	Default Core Interrupt ID	SIC Registers	
MDMA Stream 1	IVG13	43	6	IAR5	IMASK1, ISR1, IWR1
Software Watchdog Timer	IVG13	44	6	IAR5	IMASK1, ISR1, IWR1
Port H Interrupt A	IVG13	45	6	IAR5	IMASK1, ISR1, IWR1
Port H Interrupt B	IVG13	46	6	IAR5	IMASK1, ISR1, IWR1
ACM Status Interrupt	IVG7	47	0	IAR5	IMASK1, ISR1, IWR1
ACM Interrupt	IVG10	48	3	IAR6	IMASK1, ISR1, IWR1
Reserved	—	49	—	IAR6	IMASK1, ISR1, IWR1
Reserved	—	50	—	IAR6	IMASK1, ISR1, IWR1
PWM0 Trip Interrupt	IVG10	51	3	IAR6	IMASK1, ISR1, IWR1
PWM0 Sync Interrupt	IVG10	52	3	IAR6	IMASK1, ISR1, IWR1
PWM1 Trip Interrupt	IVG10	53	3	IAR6	IMASK1, ISR1, IWR1
PWM1 Sync Interrupt	IVG10	54	3	IAR6	IMASK1, ISR1, IWR1
RSI Mask 1 Interrupt	IVG10	55	3	IAR6	IMASK1, ISR1, IWR1
Reserved	—	56 through 63	—	—	IMASK1, ISR1, IWR1

Event Control

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

- CEC interrupt latch register (ILAT)—Indicates when events have been latched. The appropriate bit is set when the processor has latched the event and is cleared when the event has been accepted into the system. This register is updated automatically by the controller, but it may be written only when its corresponding IMASK bit is cleared.
- CEC interrupt mask register (IMASK)—Controls the masking and unmasking of individual events. When a bit is set in the IMASK register, that event is unmasked and is processed by the CEC when asserted. A cleared bit in the IMASK register masks the event, preventing the processor from servicing the event even though the event may be latched in the ILAT register. This register may be read or written while in supervisor mode. (Note that general-purpose interrupts can be globally enabled and disabled with the STI and CLI instructions, respectively.)
- CEC interrupt pending register (IPEND)—The IPEND register keeps track of all nested events. A set bit in the IPEND register indicates the event is currently active or nested at some level. This register is updated automatically by the controller but may be read while in supervisor mode.

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

- SIC interrupt mask registers (SIC_IMASKx)—Control the masking and unmasking of each peripheral interrupt event. When a bit is set in these registers, the corresponding peripheral event is unmasked and is forwarded to the CEC when asserted. A cleared bit in these registers masks the corresponding peripheral event, preventing the event from propagating to the CEC.
- SIC interrupt status registers (SIC_ISRx)—As multiple peripherals can be mapped to a single event, these registers allow the software to determine which peripheral event source triggered the interrupt. A set bit indicates that the peripheral is asserting the interrupt, and a cleared bit indicates that the peripheral is not asserting the event.
- SIC interrupt wakeup enable registers (SIC_IWRx)—By enabling the corresponding bit in these registers, a peripheral can be configured to wake up the processor should the core be idled or in sleep mode when the event is generated. For more information, see Dynamic Power Management on Page 13.

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

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

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

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

The electrical characteristics of each network connection are very stringent. Therefore, the CAN interface is typically divided into two parts: a controller and a transceiver. This allows a single controller to support different drivers and CAN networks. The ADSP-BF50x CAN module represents the controller part of the interface. This module's network I/O is a single transmit output and a single receive input, which connect to a line transceiver.

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

TWI CONTROLLER INTERFACE

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

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

PORTS

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

General-Purpose I/O (GPIO)

The processor has 35 bidirectional, general-purpose I/O (GPIO) pins allocated across three separate GPIO modules—PORTFIO, PORTGIO, and PORTHIO, associated with Port F, Port G, and Port H, respectively. Each GPIO-capable pin shares functionality with other processor peripherals via a multiplexing scheme; however, the GPIO functionality is the default state of the device upon power-up. Neither GPIO output nor input drivers are active by default. Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers:

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

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 volt core supply voltage, the processor enters the hibernate state. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 4 for a summary of the power settings for each mode.

Full-On Operating Mode—Maximum Performance

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

Active Operating Mode—Moderate Dynamic Power Savings

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

In the active mode, it is possible to disable the control input to the PLL by setting the PLL_OFF bit in the PLL control register. This register can be accessed with a user-callable routine in the on-chip ROM called bfrom_SysControl(). If disabled, the PLL control input must be re-enabled before transitioning to the full-on or sleep modes.

Table 4. Power Settings

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

For more information about PLL controls, see the "Dynamic Power Management" chapter in the *ADSP-BF50x 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. When in the sleep mode, asserting a wakeup enabled in the SIC_IWRx registers causes the processor to sense the value of the BYPASS bit in the PLL control register (PLL_CTL). If BYPASS is disabled, the processor transitions to the full on mode. If BYPASS is enabled, the processor transitions to the active mode.

DMA accesses to L1 memory are 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 pin ($\overline{\text{RESET}}$). Assertion of $\overline{\text{RESET}}$ while in deep sleep mode causes the processor to transition to the full on mode.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor core (CCLK) and to all of the peripherals (SCLK). This setting sets the internal power supply voltage (V_{DDINT}) to 0 V to provide the lowest static power dissipation. Any critical information stored internally (for example, memory contents, register contents, and other information) must be written to a non-volatile storage device prior to removing power if the processor state is to be preserved.

Writing 0 to the 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.

The processor can be woken up by asserting the $\overline{\text{RESET}}$ pin. All hibernate wakeup events initiate the hardware reset sequence. Individual sources are enabled by the VR_CTL register. The EXT_WAKE signal indicates the occurrence of a wakeup event.

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 5, the processor supports three 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 5. Power Domains

Power Domain	Power Supply
All internal logic, except Memory	V_{DDINT}
Flash Memory	$V_{DDFLASH}$
All other I/O	V_{DDEXT}
ADC digital supply[1] (Logic, I/O)	DV_{DD}, V_{DRIVE}
ADC analog supply[1]	AV_{DD}

[1] On ADSP-BF506F processor only.

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

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

Power Savings Factor

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

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

- Boot from serial SPI memory, EEPROM or flash (BMODE = 0x3)—8-, 16-, 24-, or 32-bit addressable devices are supported. The processor uses the PF13 GPIO pin to select a single SPI EEPROM/flash device (connected to the SPI0 interface) and submits a read command and successive address bytes (0x00) until a valid 8-, 16-, 24-, or 32-bit addressable device is detected. Pull-up resistors are required on the SPI0_SEL1 and MISO pins. By default, a value of 0x85 is written to the SPI_BAUD register.
- Boot from SPI host device (BMODE = 0x4)—The processor operates in SPI slave mode and is configured to receive the bytes of the LDR file from an SPI host (master) agent. The HWAIT signal must be interrogated by the host before every transmitted byte. A pull-up resistor is required on the SPI0_SS input. A pull-down on the serial clock (SCK) may improve signal quality and booting robustness.
- Boot from PPI host device (BMODE = 0x5)—The processor operates in PPI slave mode and is configured to receive the bytes of the LDR file from a PPI host (master) agent.
- Boot from UART0 host on Port G (BMODE = 0x7)—Using an autobaud handshake sequence, a boot-stream formatted program is downloaded by the host. The host selects a bit rate within the UART clocking capabilities.

  When performing the autobaud detection, the UART expects an "@" (0x40) character (eight bits data, one start bit, one stop bit, no parity bit) on the UA0_RX pin to determine the bit rate. The UART then replies with an acknowledgement composed of 4 bytes (0xBF, the value of UART0_DLL, the value of UART0_DLH, then 0x00). The host can then download the boot stream. The processor deasserts the UA0_RTS output to hold off the host; UA0_CTS functionality is not enabled at boot time.

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

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

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

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

INSTRUCTION SET DESCRIPTION

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

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

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

DEVELOPMENT TOOLS

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.

SIGNAL DESCRIPTIONS

Signal definitions for the ADSP-BF50x processors are listed in Table 11. All pins for the ADC (ADSP-BF506F processor only) are listed in Table 12.

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

During and immediately after reset, all processor signals (not ADC 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 until enabled by user software with the exception of the pins that need pull-ups or pull-downs, as noted in Table 11.

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

Table 11. Processor—Signal Descriptions

Signal Name	Type	Function	Driver Type
Port F: *GPIO and Multiplexed Peripherals*			
PF0/*TSCLK0/UA0_RX/TMR6/CUD0*	I/O	GPIO/*SPORT0 TX Serial CLK/UART0 RX/Timer6/Count Up Dir 0*	C
PF1/*RSCLK0/UA0_TX/TMR5/CDG0*	I/O	GPIO/*SPORT0 RX Serial CLK/UART0 TX/Timer5/Count Down Dir 0*	C
PF2/*DT0PRI/PWM0_BH/PPI_D8/CZM0*	I/O	GPIO/*SPORT0 TX Pri Data/PWM0 Drive B Hi/PPI Data 8/Counter Zero Marker 0*	C
PF3/*TFS0/PWM0_BL/PPI_D9/CDG0*	I/O	GPIO/*SPORT0 TX Frame Sync/PWM0 Drive B Lo/PPI Data 9/Count Down Dir 0*	C
PF4/*RFS0/PWM0_CH/PPI_D10/TACLK0*	I/O	GPIO/*SPORT0 RX Frame Sync/PWM0 Drive C Hi/PPI Data 10/Alt Timer CLK 0*	C
PF5/*DR0PRI/PWM0_CL/PPI_D11/TACLK1*	I/O	GPIO/*SPORT0 Pri RX Data/PWM0 Drive C Lo/PPI Data 11/Alt Timer CLK 1*	C
PF6/*UA1_TX/$\overline{\text{PWM0\_TRIP}}$/PPI_D12*	I/O	GPIO/*UART1 TX/PWM0 TRIP/PPI Data 12*	C
PF7/*UA1_RX/PWM0_SYNC/PPI_D13/TACI3*	I/O	GPIO/*UART1 RX/PWM0 SYNC/PPI Data 13/Alt Capture In 3*	C
PF8/*$\overline{\text{UA1\_RTS}}$/DT0SEC/PPI_D7*	I/O	GPIO/*UART1 RTS/SPORT0 TX Sec Data/PPI Data 7*	C
PF9/*$\overline{\text{UA1\_CTS}}$/DR0SEC/PPI_D6/CZM0*	I/O	GPIO/*UART1 CTS/SPORT0 Sec RX Data/PPI Data 6/Counter Zero Marker 0*	C
PF10/*SPI0_SCK/TMR2/PPI_D5*	I/O	GPIO/*SPI0 SCK/Timer2/PPI Data 5*	C
PF11/*SPI0_MISO/$\overline{\text{PWM0\_TRIP}}$/PPI_D4/TACLK2*	I/O	GPIO/*SPI0 MISO/PWM0 TRIP/PPI Data 4/Alt Timer CLK 2*	C
PF12/*SPI0_MOSI/PWM0_SYNC/PPI_D3*	I/O	GPIO/*SPI0 MOSI/PWM0 SYNC/PPI Data 3*	C
PF13/*$\overline{\text{SPI0\_SEL1}}$/TMR3/PPI_D2/$\overline{\text{SPI0\_SS}}$*	I/O	GPIO/*SPI0 Slave Select 1/Timer3/PPI Data 2/SPI0 Slave Select In*	C
PF14/*$\overline{\text{SPI0\_SEL2}}$/PWM0_AH/PPI_D1*	I/O	GPIO/*SPI0 Slave Select 2/PWM0 AH/PPI Data 1*	C
PF15/*$\overline{\text{SPI0\_SEL3}}$/PWM0_AL/PPI_D0*	I/O	GPIO/*SPI0 Slave Select 3/PWM0 AL/PPI Data 0*	C
Port G: *GPIO and Multiplexed Peripherals*			
PG0/*$\overline{\text{SPI1\_SEL3}}$/TMRCLK/PPI_CLK/UA1_RX/TACI4*	I/O	GPIO/*SPI1 Slave Select 3/Timer CLK/PPI Clock/UART1 RX/Alt Capture In 4*	C
PG1/*$\overline{\text{SPI1\_SEL2}}$/PPI_FS3/CAN_RX/TACI5*	I/O	GPIO/*SPI1 Slave Select 2/PPI FS3/CAN RX/Alt Capture In 5*	C
PG2/*$\overline{\text{SPI1\_SEL1}}$/TMR4/CAN_TX/$\overline{\text{SPI1\_SS}}$*	I/O	GPIO/*SPI1 Slave Select 1/Timer4/CAN TX/SPI1 Slave Select In*	C
PG3/*HWAIT/SPI1_SCK/DT1SEC/UA1_TX*	I/O	GPIO/*HWAIT/SPI1 SCK/SPORT1 TX Sec Data/UART1 TX*	C
PG4/*SPI1_MOSI/DR1SEC/PWM1_SYNC/TACLK6*	I/O	GPIO/*SPI1 MOSI/SPORT1 Sec RX Data/PWM1 SYNC/Alt Timer CLK 6*	C
PG5/*SPI1_MISO/TMR7/$\overline{\text{PWM1\_TRIP}}$*	I/O	GPIO/*SPI1 MISO/Timer7/PWM1 TRIP*	C
PG6/*ACM_SGLDIFF/SD_D3/PWM1_AH*	I/O	GPIO/*ADC CM SGL DIFF/SD Data 3/PWM1 Drive A Hi*	C
PG7/*ACM_RANGE/SD_D2/PWM1_AL*	I/O	GPIO/*ADC CM RANGE/SD Data 2/PWM1 Drive A Lo*	C
PG8/*DR1SEC/SD_D1/PWM1_BH*	I/O	GPIO/*SPORT1 Sec RX Data/SD Data 1/PWM1 Drive B Hi*	C
PG9/*DR1PRI/SD_D0/PWM1_BL*	I/O	GPIO/*SPORT1 Pri RX Data/SD Data 0/PWM1 Drive B Lo*	C
PG10/*RFS1/SD_CMD/PWM1_CH/TACI6*	I/O	GPIO/*SPORT1 RX Frame Sync/SD CMD/PWM1 Drive C Hi/Alt Capture In 6*	C
PG11/*RSCLK1/SD_CLK/PWM1_CL/TACLK7*	I/O	GPIO/*SPORT1 RX Serial CLK/SD CLK/PWM1 Drive C Lo/Alt Timer CLK 7*	C
PG12/*UA0_RX/SD_D4/PPI_D15/TACI2*	I/O	GPIO/*UART0 RX/SD Data 4/PPI Data 15/Alt Capture In 2*	C
PG13/*UA0_TX/SD_D5/PPI_D14/CZM1*	I/O	GPIO/*UART0 TX/SD Data 5/PPI Data 14/Counter Zero Marker 1*	C

Parameter		Test Conditions	Min	Typical	Max	Unit
$I_{DDFLASH2}$	Flash Memory Supply Current 2 — Reset/Powerdown			15	50	µA
$I_{DDFLASH3}$	Flash Memory Supply Current 3 — Standby			15	50	µA
$I_{DDFLASH4}$	Flash Memory Supply Current 4 — Automatic Standby			15	50	µA
$I_{DDFLASH5}$	Flash Memory Supply Current 5 — Program			15	40	mA
	Flash Memory Supply Current 5 — Erase			15	40	mA
$I_{DDFLASH6}$	Flash Memory Supply Current 6 — Dual Operations	Program/Erase in one bank, asynchronous read in another bank		25	60	mA
		Program/Erase in one bank, synchronous read in another bank		43	70	mA
$I_{DDFLASH7}$	Flash Memory Supply Current 7 — Program/Erase Suspended (Standby)			15	50	µA

[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, except SCL and SDA.
[6] Guaranteed, but not tested.
[7] See the *ADSP-BF50x Blackfin Processor Hardware Reference Manual* for definition of sleep, deep sleep, and hibernate operating modes.
[8] Applies to V_{DDEXT} supply only. Clock inputs are tied high or low.
[9] Guaranteed maximum specifications.
[10] Unit for V_{DDINT} is V (Volts). Unit for f_{SCLK} is MHz. Example: 1.4 V, 75 MHz would be $0.16 \times 1.4 \times 75 = 16.8$ mA adder.
[11] See the *ADSP-BF50x Blackfin Processor Hardware Reference Manual* for definition of NORCLK.

Serial Ports

Table 30 through Table 33 on Page 41 and Figure 20 on Page 40 through Figure 22 on Page 41 describe serial port operations.

Table 30. Serial Ports—External Clock

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_{SFSE}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	3.0		3.0		ns
t_{HFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	3.0		3.0		ns
t_{SDRE}	Receive Data Setup Before RSCLKx[1,2]	3.0		3.0		ns
t_{HDRE}	Receive Data Hold After RSCLKx[1,2]	3.5		3.0		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
Switching Characteristics						
t_{DFSE}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[3]		10.0		10.0	ns
t_{HOFSE}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[3]	0.0		0.0		ns
t_{DDTE}	Transmit Data Delay After TSCLKx[3]		11.0		10.0	ns
t_{HDTE}	Transmit Data Hold After TSCLKx[3]	0.0		0.0		ns

[1] Referenced to sample edge.
[2] When SPORT is used in conjunction with the ACM, refer to the timing requirements in Table 41 (ACM Timing).
[3] Referenced to drive edge.

Table 31. Serial Ports—Internal Clock

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_{SFSI}	TFSx/RFSx Setup Before TSCLKx/RSCLKx[1]	11.0		9.6		ns
t_{HFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx[1]	–1.5		–1.5		ns
t_{SDRI}	Receive Data Setup Before RSCLKx[1,2]	11.5		10.0		ns
t_{HDRI}	Receive Data Hold After RSCLKx[1,2]	–1.5		–1.5		ns
Switching Characteristics						
t_{SCLKIW}	TSCLKx/RSCLKx Width	7.0		8.0		ns
t_{DFSI}	TFSx/RFSx Delay After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[3]		4.0		3.0	ns
t_{HOFSI}	TFSx/RFSx Hold After TSCLKx/RSCLKx (Internally Generated TFSx/RFSx)[3]	–2.0		–1.0		ns
t_{DDTI}	Transmit Data Delay After TSCLKx[3]		4.0		3.0	ns
t_{HDTI}	Transmit Data Hold After TSCLKx[3]	–1.8		–1.5		ns

[1] Referenced to sample edge.
[2] When SPORT is used in conjunction with the ACM, refer to the timing requirements in Table 41 (ACM Timing).
[3] Referenced to drive edge.

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

The UART ports receive and transmit operations are described in the *ADSP-BF50x Hardware Reference Manual*.

General-Purpose Port Timing

Table 36 and Figure 25 describe general-purpose port operations.

Table 36. General-Purpose Port 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 Requirement						
t_{WFI}	General-Purpose Port Pin Input Pulse Width	t_{SCLK} + 1		t_{SCLK} + 1		ns
Switching Characteristic						
t_{GPOD}	General-Purpose Port Pin Output Delay from CLKOUT High	0	11.0	0	8.9	ns

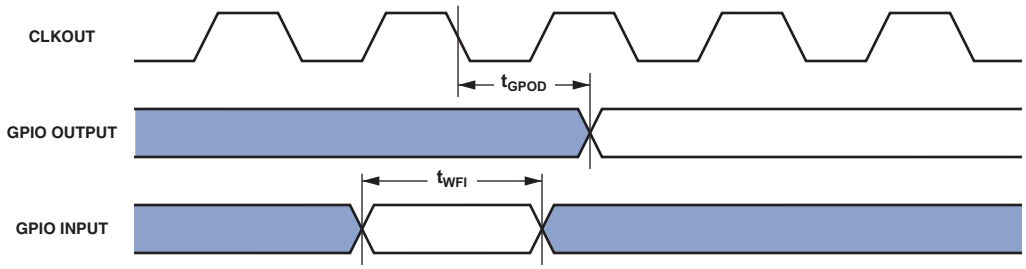


Figure 25. General-Purpose Port Timing

PROCESSOR—OUTPUT DRIVE CURRENTS

Figure 32 through Figure 40 show typical current-voltage characteristics for the output drivers of the ADSP-BF50xF processors.

The curves represent the current drive capability of the output drivers. See Table 11 on Page 22 for information about which driver type corresponds to a particular pin.

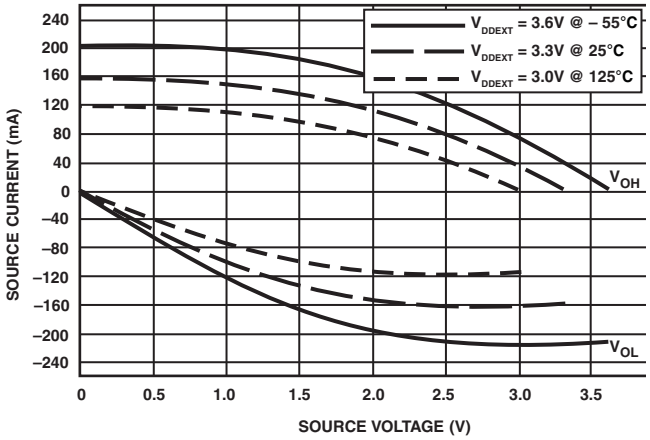


Figure 32. Driver Type B Current (3.3 V V_{DDEXT})

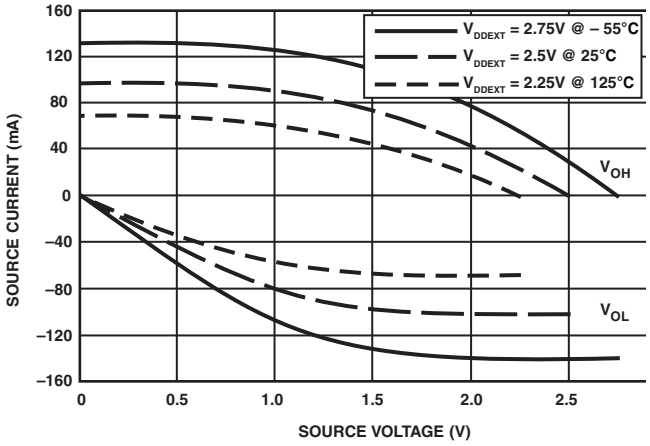


Figure 33. Driver Type B Current (2.5 V V_{DDEXT})

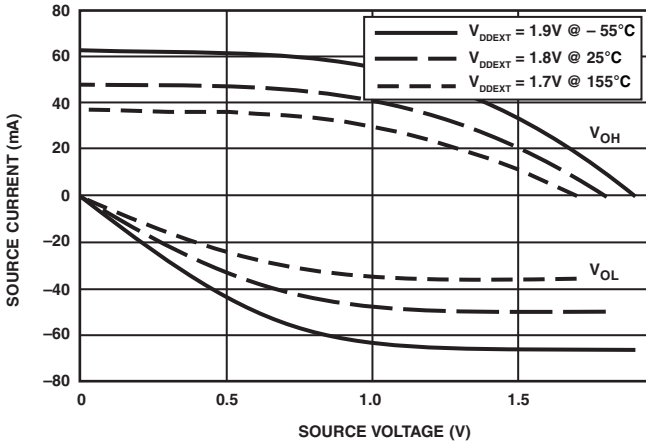


Figure 34. Driver Type B Current (1.8 V V_{DDEXT})

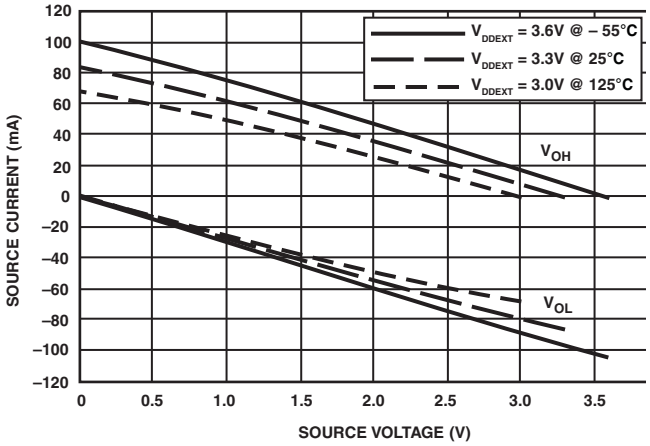


Figure 35. Driver Type C Current (3.3 V V_{DDEXT})

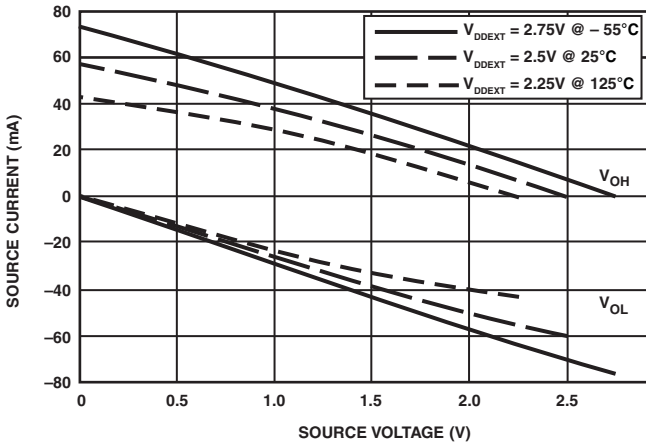


Figure 36. Drive Type C Current (2.5 V V_{DDEXT})

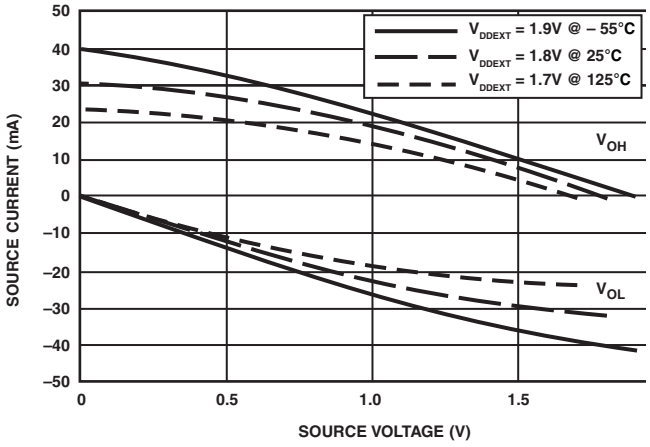


Figure 37. Driver Type C Current (1.8 V V_{DDEXT})

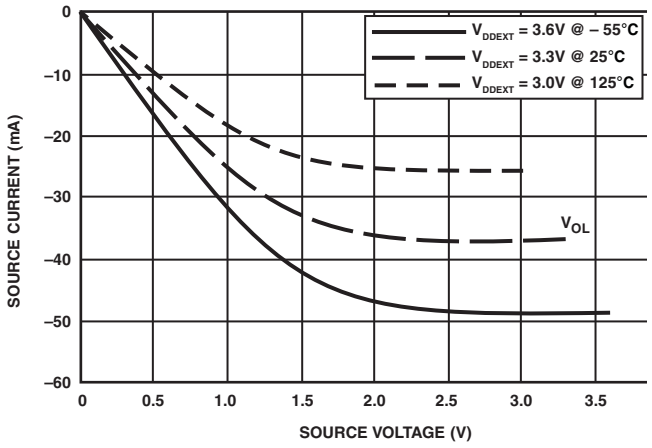


Figure 38. Driver Type D Current (3.3 V V_{DDEXT})

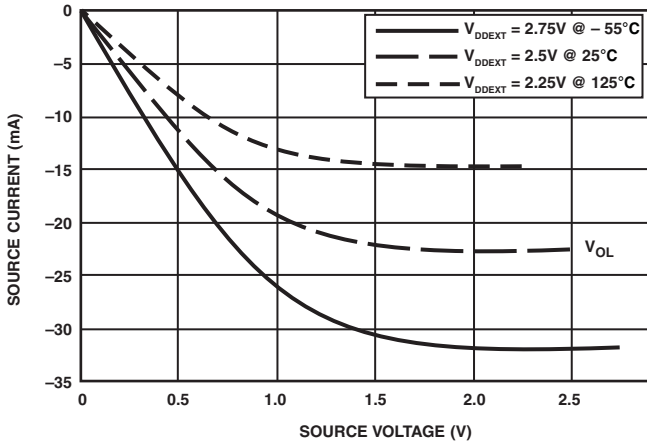


Figure 39. Driver Type D Current (2.5 V V_{DDEXT})

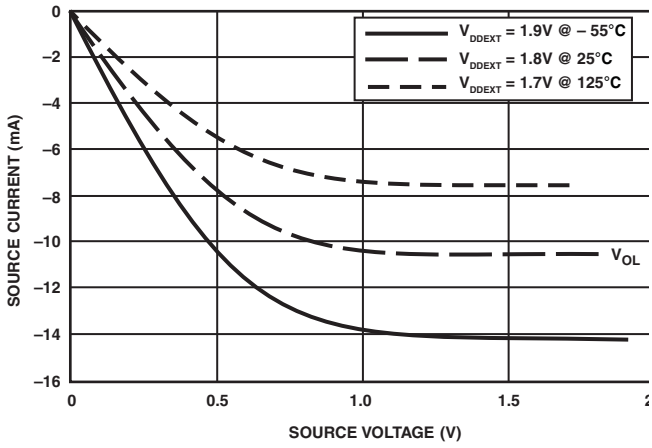


Figure 40. Driver Type D Current (1.8 V V_{DDEXT})

PROCESSOR—TEST CONDITIONS

All timing parameters appearing in this data sheet were measured under the conditions described in this section. Figure 41 shows the measurement point for AC measurements (except output enable/disable). The measurement point V_{MEAS} is $V_{DDEXT}/2$ for V_{DDEXT} (nominal) = 1.8 V/2.5 V/3.3 V.

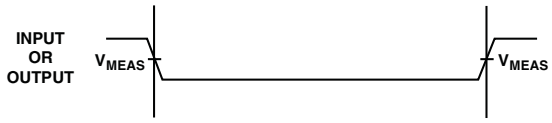


Figure 41. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time Measurement

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

The output enable time t_{ENA} is the interval from the point when a reference signal reaches a high or low voltage level to the point when the output starts driving as shown on the right side of Figure 42.

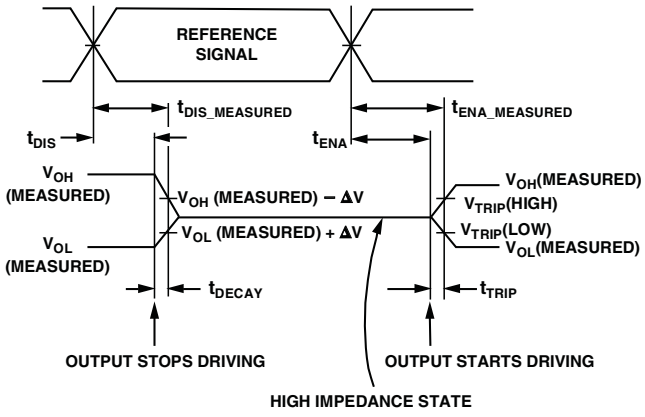


Figure 42. Output Enable/Disable

The time $t_{ENA\_MEASURED}$ is the interval, from when the reference signal switches, to when the output voltage reaches V_{TRIP}(high) or V_{TRIP}(low). For V_{DDEXT} (nominal) = 1.8 V, V_{TRIP} (high) is 1.05 V, and V_{TRIP} (low) is 0.75 V. For V_{DDEXT} (nominal) = 2.5 V, V_{TRIP} (high) is 1.5 V and V_{TRIP} (low) is 1.0 V. For V_{DDEXT} (nominal) = 3.3 V, V_{TRIP} (high) is 1.9 V, and 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 pin to start driving.

Output Disable Time Measurement

Output pins are considered to be disabled when they stop driving, go into a high impedance state, and start to decay from their output high or low voltage. The output disable time t_{DIS} is the difference between $t_{DIS\_MEASURED}$ and t_{DECAY} as shown on the left side of Figure 42.

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

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) = 1.8 V.

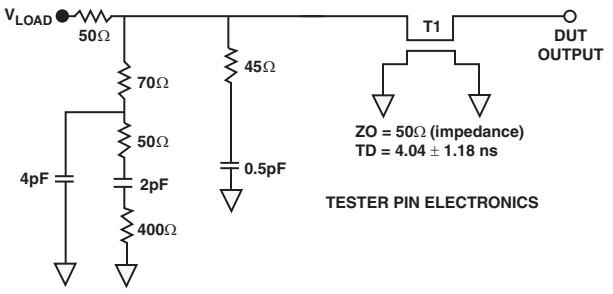
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches, to when the output voltage decays ΔV from the measured output high or output low voltage.

Example System Hold Time Calculation

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

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all pins (see Figure 43). V_{LOAD} is equal to (V_{DDEXT}) /2. The graphs of Figure 44 through Figure 49 show how output rise time varies with capacitance. The delay and hold specifications given should be derated by a factor derived from these figures. The graphs in these figures may not be linear outside the ranges shown.



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

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

Figure 43. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

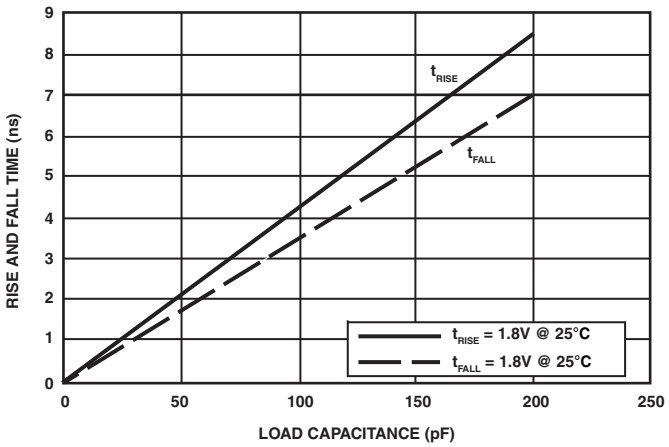


Figure 44. Driver Type B Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (1.8 V V_{DDEXT})

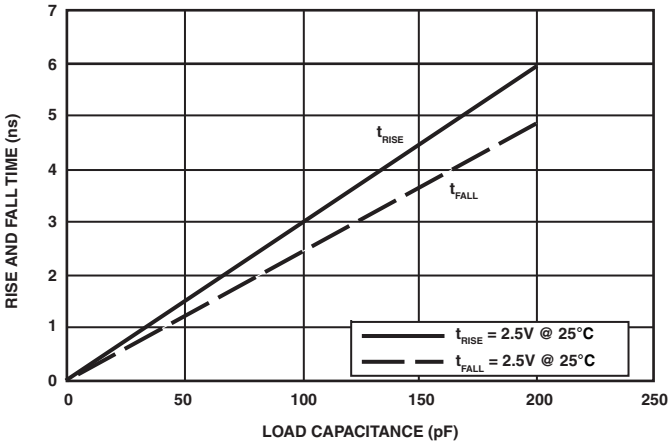


Figure 45. Driver Type B Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (2.5 V V_{DDEXT})

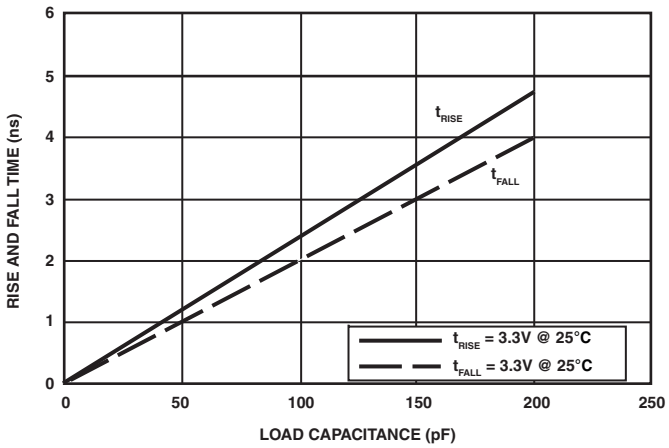


Figure 46. Driver Type B Typical Rise and Fall Times (10%–90%) vs. Load Capacitance (3.3 V V_{DDEXT})

FLASH—SPECIFICATIONS

Specifications subject to change without notice.

FLASH—PROGRAM AND ERASE TIMES AND ENDURANCE CYCLES

The program and erase times and the number of program/ erase cycles per block are shown in Table 45. Exact erase times may change depending on the memory array condition. The best case is when all the bits in the block or bank are at '0' (pre programmed). The worst case is when all the bits in the block or bank are at '1' (not pre programmed). Usually, the system overhead is negligible with respect to the erase time.

Table 45. Program/Erase Times and Endurance Cycles

Parameter	Condition	Typical	Typical After 100k Write/Erase Cycles	Max	Unit
Erase	Parameter Block (4K word)[1]	0.3	1	2.5	s
Erase	Main Block (32K word)—preprogrammed	0.8	3	4	s
Erase	Main Block (32K word)—not preprogrammed	1		4	s
Program[2]	Word	12	12	100	µs
Program[2]	Parameter Block (4K word)	40			ms
Program[2]	Main Block (32K word)	300			ms
Suspend Latency	Program	5		10	µs
Suspend Latency	Erase	5		20	µs
Program/Erase Cycles (per Block)	Main Blocks			100,000	Cycles
Program/Erase Cycles (per Block)	Parameter Blocks			100,000	Cycles

[1] The difference between pre programmed and not pre programmed is not significant (< 30 ms).
[2] Values are liable to change with the external system-level overhead (command sequence and Status Register polling execution).

FLASH—ABSOLUTE MAXIMUM RATINGS

Table 46 shows the ADC absolute maximum ratings.

Table 46. Flash Absolute Maximum Ratings

Parameter	Rating
Junction Temperature While Biased	See Table 20 on Page 31
Storage Temperature Range	See Table 20 on Page 31
Flash Memory Supply Voltage ($V_{DDFLASH}$)	–0.2 V to +2.45 V

When the ADC starts a conversion (see Figure 63 (ADC Conversion Phase)), SW3 opens and SW1 and SW2 move to Position B, causing the comparator to become unbalanced. Both inputs are disconnected once the conversion begins. The control logic and the charge redistribution DACs are used to add and subtract fixed amounts of charge from the sampling capacitor arrays to bring the comparator back into a balanced condition. When the comparator is rebalanced, the conversion is complete. The control logic generates the ADC output code. The output impedances of the sources driving the V_{IN+} and V_{IN-} pins must be matched; otherwise, the two inputs will have different settling times, resulting in errors.

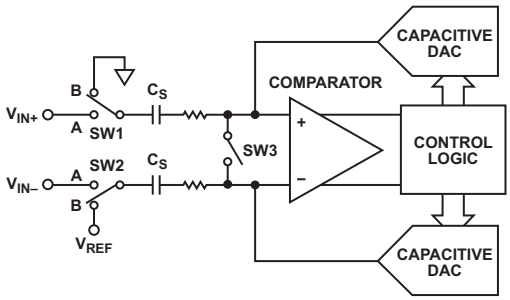


Figure 63. ADC Conversion Phase

Analog Input Structure

Figure 64 (Equivalent Analog Input Circuit, Conversion Phase—Switches Open, Track Phase—Switches Closed) shows the equivalent circuit of the analog input structure of the ADC in differential/pseudo differential mode. In single-ended mode, V_{IN} is internally tied to AGND. The four diodes provide ESD protection for the analog inputs. Care must be taken to ensure that the analog input signals never exceed the supply rails by more than 300 mV. This causes these diodes to become forward-biased and starts conducting into the substrate. These diodes can conduct up to 10 mA without causing irreversible damage to the part.

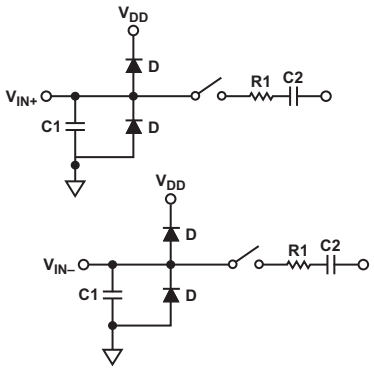


Figure 64. Equivalent Analog Input Circuit, Conversion Phase—Switches Open, Track Phase—Switches Closed

The C1 capacitors in Figure 64 (Equivalent Analog Input Circuit, Conversion Phase—Switches Open, Track Phase—Switches Closed) are typically 4 pF and can primarily be attributed to pin capacitance. The resistors are lumped components made up of the on resistance of the switches. The value of these resistors is typically about 100 Ω. The C2 capacitors are the ADC's sampling capacitors with a capacitance of 45 pF typically.

For ac applications, removing high frequency components from the analog input signal is recommended by the use of an RC low-pass filter on the relevant analog input pins with optimum values of 47 Ω and 10 pF. In applications where harmonic distortion and signal-to-noise ratio are critical, the analog input should be driven from a low impedance source. Large source impedances significantly affect the ac performance of the ADC and may necessitate the use of an input buffer amplifier. The choice of the op amp is a function of the particular application.

When no amplifier is used to drive the analog input, the source impedance should be limited to low values. The maximum source impedance depends on the amount of THD that can be tolerated.

The THD increases as the source impedance increases and performance degrades. Figure 65 (THD vs. Analog Input Frequency for Various Source Impedances, Single-Ended Mode shows a graph of the THD vs. the analog input signal frequency for different source impedances in single-ended mode, while Figure 66 (THD vs. Analog Input Frequency for Various Source Impedances, Differential Mode) shows the THD vs. the analog input signal frequency for different source impedances in differential mode.

Figure 67 (THD vs. Analog Input Frequency for Various Supply Voltages) shows a graph of the THD vs. the analog input frequency for various supplies while sampling at 2 MSPS. In this case, the source impedance is 47 Ω.

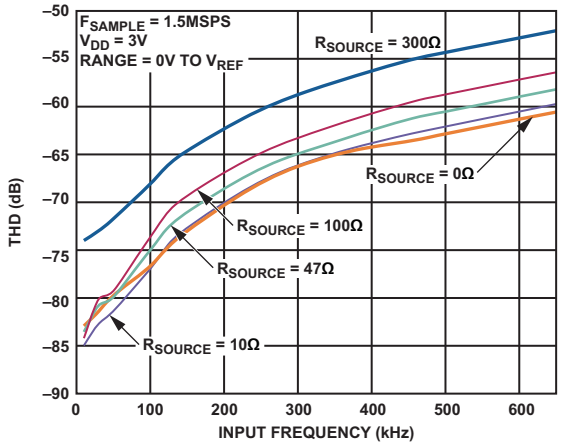


Figure 65. THD vs. Analog Input Frequency for Various Source Impedances, Single-Ended Mode

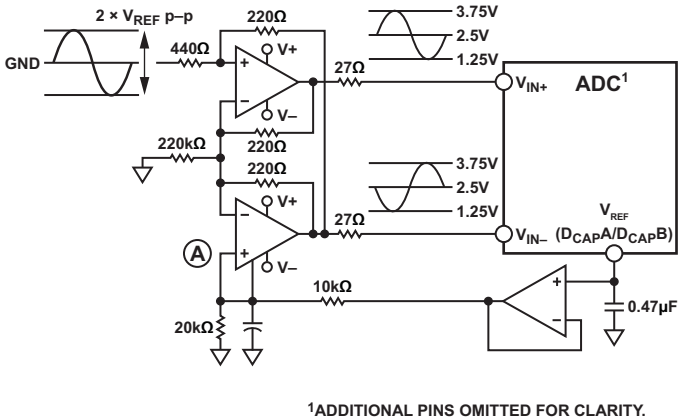


Figure 73. Dual Op Amp Circuit to Convert a Single-Ended Bipolar Signal into a Differential Unipolar Signal

to make use of the full dynamic range of the part. A dc input is applied to the V_{IN-} pin. The voltage applied to this input provides an offset from ground or a pseudo ground for the V_{IN+} input. The benefit of pseudo differential inputs is that they separate the analog input signal ground from the ADC's ground allowing dc common-mode voltages to be cancelled.

The typical voltage range for the V_{IN-} pin, while in pseudo differential mode, is shown in Figure 74 (V_{IN-} Input Voltage Range vs. V_{REF} in Pseudo Differential Mode with V_{DD} = 3 V) and Figure 75 (V_{IN-} Input Voltage Range vs. V_{REF} in Pseudo Differential Mode with V_{DD} = 5 V). Figure 76 (Pseudo Differential Mode Connection Diagram) shows a connection diagram for pseudo differential mode.

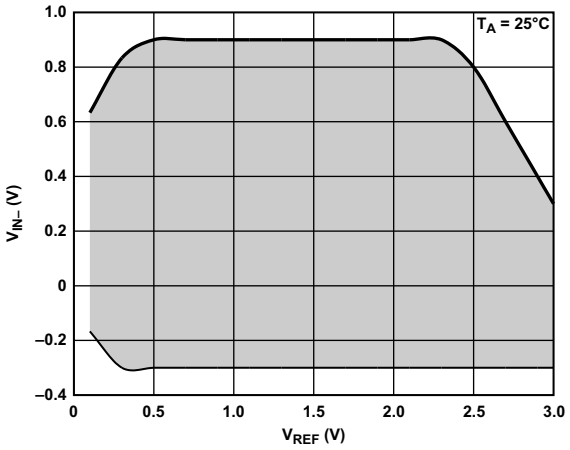


Figure 74. V_{IN-} Input Voltage Range vs. V_{REF} in Pseudo Differential Mode with V_{DD} = 3 V

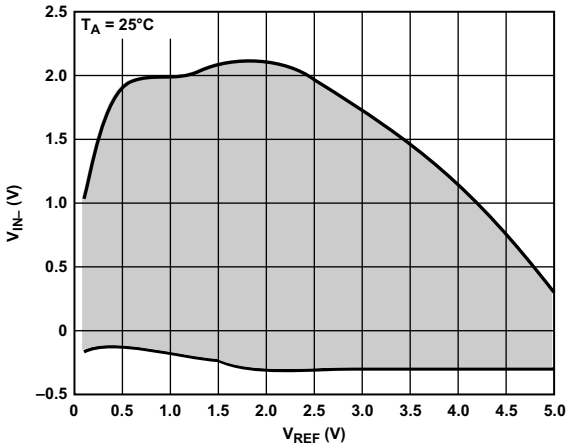


Figure 75. V_{IN-} Input Voltage Range vs. V_{REF} in Pseudo Differential Mode with V_{DD} = 5 V

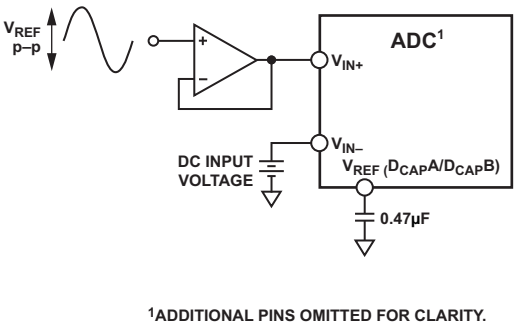


Figure 76. Pseudo Differential Mode Connection Diagram

Analog Input Selection

The analog inputs of the ADC can be configured as single-ended or true differential via the SGL/$\overline{\text{DIFF}}$ logic pin, as shown in Figure 77 (Selecting Differential or Single-Ended Configuration). If this pin is tied to a logic low, the analog input channels to each on-chip ADC are set up as three true differential pairs. If this pin is at logic high, the analog input channels to each on-chip ADC are set up as six single-ended analog inputs. The required logic level on this pin needs to be established prior to the acquisition time and remain unchanged during the conversion time until the track-and-hold has returned to track. The track-and-hold returns to track on the 13th rising edge of ADSCLK after the $\overline{\text{CS}}$ falling edge (see Figure 87 (Serial Interface Timing Diagram)). If the level on this pin is changed, it will be recognized by the ADC; therefore, it is necessary to keep the same logic level during acquisition and conversion to avoid corrupting the conversion in progress.

For example, in Figure 77 (Selecting Differential or Single-Ended Configuration) the SGL/$\overline{\text{DIFF}}$ pin is set at logic high for the duration of both the acquisition and conversion times so the analog inputs are configured as single ended for that conversion (Sampling Point A). The logic level of the SGL/$\overline{\text{DIFF}}$ changed to low after the track-and-hold returned to track and prior to the

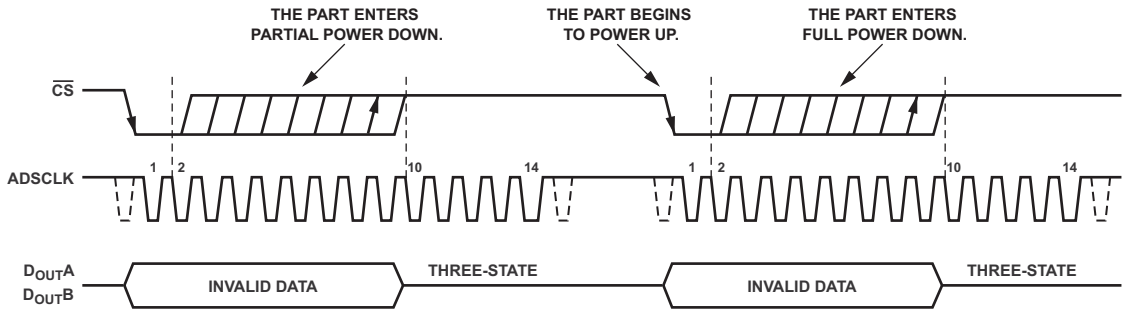


Figure 83. Entering Full Power-Down Mode

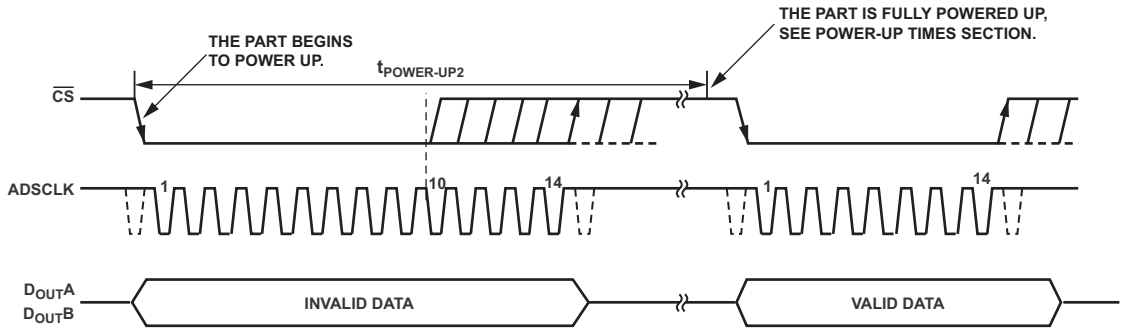


Figure 84. Exiting Full Power-Down Mode

Power-Up Times

As described in detail, the ADC has two power-down modes, partial power-down and full power-down. This section deals with the power-up time required when coming out of either of these modes. It should be noted that the power-up times, as explained in this section, apply with the recommended capacitors in place on the $D_{CAP}A$ and $D_{CAP}B$ pins.

To power up from full power-down, approximately 1.5 ms should be allowed from the falling edge of $\overline{CS}$, shown as $t_{POWER-UP2}$ in Figure 84 (Exiting Full Power-Down Mode). Powering up from partial power-down requires much less time. The power-up time from partial power-down is typically 1 µs; however, if using the internal reference, then the ADC must be in partial power-down for at least 67 µs in order for this power-up time to apply.

When power supplies are first applied to the ADC, the ADC may power up in either of the power-down modes or normal mode. Because of this, it is best to allow a dummy cycle to elapse to ensure the part is fully powered up before attempting a valid conversion. Likewise, if it is intended to keep the part in the partial power-down mode immediately after the supplies are applied, then two dummy cycles must be initiated. The first dummy cycle must hold $\overline{CS}$ low until after the 10th ADSCLK falling edge (see Figure 80 (Normal Mode Operation)); in the second cycle, $\overline{CS}$ must be brought high before the 10th ADSCLK edge but after the second ADSCLK falling edge (see Figure 81 (Entering Partial Power-Down Mode)). Alternatively, if it is intended to place the part in full power-down mode when the supplies are applied, then three dummy cycles must be initiated. The first dummy cycle must hold $\overline{CS}$ low until after the 10th ADSCLK falling edge (see Figure 80 (Normal Mode Operation)); the second and third dummy cycles place the part in full power-down (see Figure 83 (Entering Full Power-Down Mode)).

Once supplies are applied to the ADC, enough time must be allowed for any external reference to power up and charge the various reference buffer decoupling capacitors to their final values.

Power vs. Throughput Rate

The power consumption of the ADC varies with the throughput rate. When using very slow throughput rates and as fast an ADSCLK frequency as possible, the various power-down options can be used to make significant power savings. However, the ADC quiescent current is low enough that even without using the power-down options, there is a noticeable variation in power consumption with sampling rate. This is true whether a fixed ADSCLK value is used or if it is scaled with the sampling rate. Figure 85 (Power vs. Throughput in Normal Mode with VDD = 3 V) and Figure 86 (Power vs. Throughput in Normal Mode with VDD = 5 V) show plots of power vs. the throughput rate when operating in normal mode for a fixed

120-LEAD LQFP LEAD ASSIGNMENT

Table 54 lists the LQFP leads by signal mnemonic. Table 55 on Page 74 lists the LQFP leads by lead number.

Table 54. 120-Lead LQFP Lead Assignment (Alphabetical by Signal)

Signal	Lead No.	Signal	Lead No.	Signal	Lead No.	Signal	Lead No.
A0	100	NC	72	PG11	46	V_{B5}	88
A1	98	$\overline{NMI}$	11	PG12	47	V_{B6}	87
A2	97	PF0	118	PG13	48	V_{DDEXT}	1
AGND	73	PF1	119	PG14	49	V_{DDEXT}	6
AGND	78	PF2	2	PG15	50	V_{DDEXT}	15
AGND	79	PF3	4	PH0	113	V_{DDEXT}	20
AGND	82	PF4	3	PH1	115	V_{DDEXT}	23
AGND	93	PF5	5	PH2	114	V_{DDEXT}	26
AGND	99	PF6	7	RANGE	95	V_{DDEXT}	30
AV_{DD}	76	PF7	8	REF_SELECT	75	V_{DDEXT}	41
BMODE0	58	PF8	9	$\overline{RESET}$	12	V_{DDEXT}	51
BMODE1	57	PF9	10	SCL	55	V_{DDEXT}	59
BMODE2	56	PF10	14	ADSCLK	102	V_{DDEXT}	62
CLKIN	110	PF11	16	SDA	54	V_{DDEXT}	64
$\overline{CS}$	101	PF12	18	SGL/$\overline{DIFF}$	96	V_{DDEXT}	66
$D_{CAP}A$	77	PF13	19	TCK	34	V_{DDEXT}	67
$D_{CAP}B$	94	PF14	21	TDI	33	V_{DDEXT}	112
DGND	74	PF15	22	TDO	36	V_{DDEXT}	116
DGND	104	$\overline{PG}$	71	TMS	35	$V_{DDFLASH}$	25
$D_{OUT}A$	105	PG0	27	$\overline{TRST}$	37	$V_{DDFLASH}$	63
$D_{OUT}B$	103	PG1	28	V_{A1}	80	$V_{DDFLASH}$	69
DV_{DD}	107	PG2	29	V_{A2}	81	V_{DDINT}	24
$\overline{EMU}$	68	PG3	31	V_{A3}	83	V_{DDINT}	42
EXT_WAKE	70	PG4	32	V_{A4}	84	V_{DDINT}	52
EXTCLK	120	PG5	38	V_{A5}	85	V_{DDINT}	53
GND	13	PG6	39	V_{A6}	86	V_{DDINT}	61
GND	17	PG7	40	V_{B1}	92	V_{DDINT}	65
GND	108	PG8	43	V_{B2}	91	V_{DDINT}	117
GND	109	PG9	44	V_{B3}	90	V_{DRIVE}	106
NC	60	PG10	45	V_{B4}	89	XTAL	111
						GND	121*
						AGND	122**

* Pin no. 121 is the GND supply (see Figure 89 and Figure 90) for the processor (4.6mm × 6.17mm); this pad **must** connect to GND.
** Pin no. 122 is the AGND supply (see Figure 89 and Figure 90) for the ADC (2.81mm × 2.81mm); this pad **must** connect to AGND.

Table 57. 88-Lead LFCSP Lead Assignment (Numerical by Lead Number)

Lead No.	Signal	Lead No.	Signal	Lead No.	Signal	Lead No.	Signal
1	$\overline{\text{NMI}}$	23	TDI	45	NC	67	GND
2	$\overline{\text{RESET}}$	24	TCK	46	NC	68	CLKIN
3	GND	25	TMS	47	NC	69	XTAL
4	PF10	26	$\overline{\text{TRST}}$	48	NC	70	V_{DDEXT}
5	V_{DDEXT}	27	TDO	49	BMODE2	71	PH0
6	PF11	28	PG5	50	BMODE1	72	PH1
7	GND	29	PG6	51	BMODE0	73	PH2
8	PF12	30	PG7	52	V_{DDEXT}	74	V_{DDEXT}
9	PF13	31	V_{DDEXT}	53	V_{DDINT}	75	V_{DDINT}
10	V_{DDEXT}	32	V_{DDINT}	54	V_{DDEXT}	76	PF0
11	PF14	33	PG8	55	$V_{DDFLASH}$	77	PF1
12	PF15	34	PG9	56	V_{DDEXT}	78	EXTCLK
13	V_{DDEXT}	35	PG10	57	V_{DDINT}	79	V_{DDEXT}
14	V_{DDINT}	36	PG11	58	V_{DDEXT}	80	PF2
15	$V_{DDFLASH}$	37	PG12	59	V_{DDEXT}	81	PF3
16	V_{DDEXT}	38	PG13	60	$\overline{\text{EMU}}$	82	PF4
17	PG0	39	PG14	61	$V_{DDFLASH}$	83	PF5
18	PG1	40	PG15	62	EXT_WAKE	84	V_{DDEXT}
19	PG2	41	V_{DDEXT}	63	$\overline{\text{PG}}$	85	PF6
20	V_{DDEXT}	42	V_{DDINT}	64	NC	86	PF7
21	PG3	43	SDA	65	NC	87	PF8
22	PG4	44	SCL	66	NC	88	PF9
						89*	GND

* Pin no. 89 is the GND supply (see Figure 92) for the processor; this pad **must** connect to GND.