THE DATASHEET OF

ADBF609WCBCZ502

Blackfin Dual Core Embedded Processor

ADSP-BF606/ADSP-BF607/ADSP-BF608/ADSP-BF609

FEATURES

Dual-core symmetric high-performance Blackfin processor, up to 500 MHz per core
- **Each core contains two 16-bit MACs, two 40-bit ALUs, and a 40-bit barrel shifter**
- **RISC-like register and instruction model for ease of programming and compiler-friendly support**
- **Advanced debug, trace, and performance monitoring**

Pipelined Vision Processor provides hardware to process signal and image algorithms used for pre- and co-processing of video frames in ADAS or other video processing applications

Accepts a range of supply voltages for I/O operation. See **Operating Conditions on Page 52**

Off-chip voltage regulator interface

349-ball BGA package (19 mm × 19 mm), RoHS compliant

MEMORY

Each core contains 148K bytes of L1 SRAM memory (processor core-accessible) with multi-parity bit protection

Up to 256K bytes of L2 SRAM memory with ECC protection

Dynamic memory controller provides 16-bit interface to a single bank of DDR2 or LPDDR DRAM devices

Static memory controller with asynchronous memory interface that supports 8-bit and 16-bit memories

4 Memory-to-memory DMA streams, 2 of which feature CRC protection

Flexible booting options from flash, SD EMMC, and SPI memories and from SPI, link port and UART hosts

Memory management unit provides memory protection

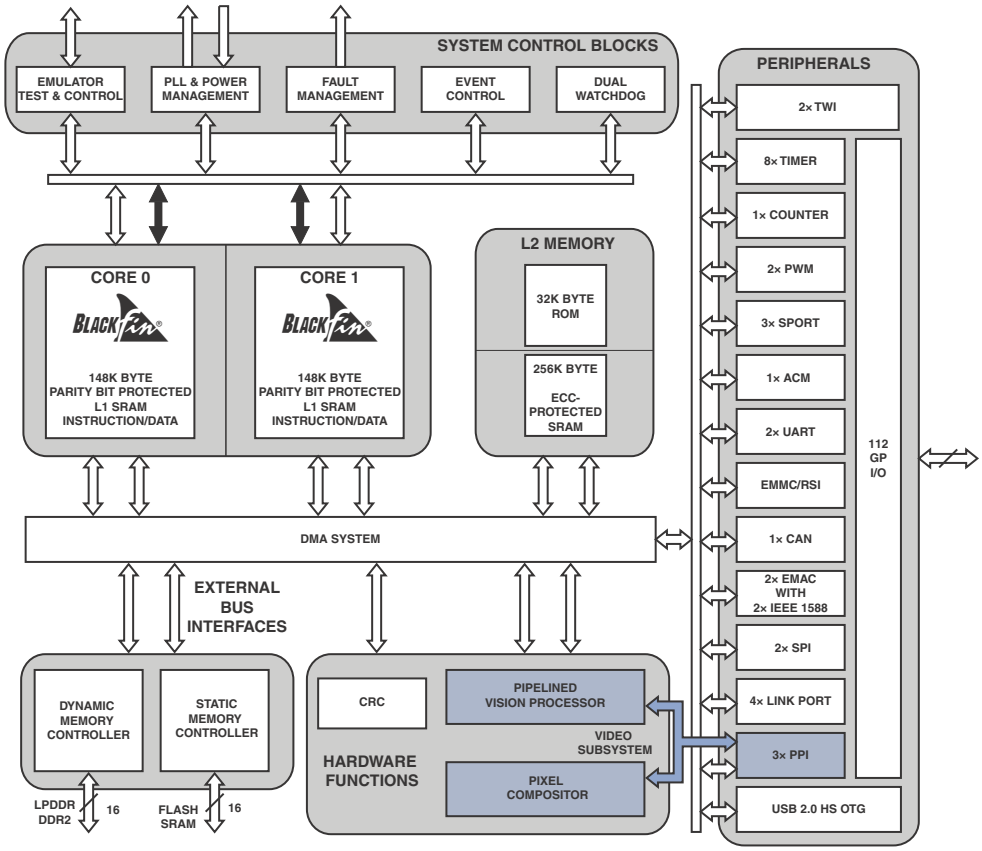


Figure 1. Processor Block Diagram

Rev. A **Document Feedback**

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106 U.S.A.
Tel: 781.329.4700
Technical Support **www.analog.com**

TABLE OF CONTENTS

REVISION HISTORY

2/14—Rev. 0 to Rev. A

Added the system clock output specification and additional peripheral external clocks in Clock Related Operating Conditions on Page 53. These changes affect the following peripheral timing sections.

Enhanced Parallel Peripheral Interface Timing 74
Link Ports 78
Serial Ports—External Clock 80
Serial Peripheral Interface (SPI) Port—Master Timing 86
Serial Peripheral Interface (SPI) Port—Slave Timing 88
ADC Controller Module (ACM) Timing 96

Additional revisions include the following.

Corrected S0SEL and S1SEL in Figure 8 Clock Relationships and Divider Values 54

Revised the dynamic and static current tables CCLK Dynamic Current per core (mA, with ASF = 1) 57
Static Current—IDD_DEEPSLEEP (mA) 58

Corrected the t_{WARE} parameter in Asynchronous Page Mode Read 64

Corrected the timing diagram in Bus Request/Bus Grant . 69

Corrected the signal names in the following figures:
DDR2 SDRAM Clock and Control Cycle Timing 69
DDR2 SDRAM Controller Input AC Timing 70
Mobile DDR SDRAM Clock and Control Cycle Timing ... 72

Added Figure 29 and updated Table 42 in Enhanced Parallel Peripheral Interface Timing 74

Corrected the t_{HSPIDM}, t_{SDSCIM}, t_{SPICLK}, t_{HDSM}, and t_{SPITDM} specifications in Serial Peripheral Interface (SPI) Port—Master Timing 86

Corrected the t_{HDSPID} specification in Serial Peripheral Interface (SPI) Port—Slave Timing 88

Corrected $t_{SRDYSCKM1}$ in Serial Peripheral Interface (SPI) Port—SPI_RDY Timing 92

Revised all parameters in Timer Cycle Timing 94

Corrected the timing diagram in ADC Controller Module (ACM) Timing 96

Removed TWI signals in footnote 3 in JTAG Test And Emulation Port Timing 101

Added models to Automotive Products 112

GENERAL DESCRIPTION

The ADSP-BF60x processors are members of the Blackfin family of products, incorporating the Analog Devices/Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like microprocessor instruction set, and single-instruction, multiple-data (SIMD) multimedia capabilities into a single instruction-set architecture.

The processors offer performance up to 500 MHz, as well as low static power consumption. Produced with a low-power and low-voltage design methodology, they provide world-class power management and performance.

By integrating a rich set of industry-leading system peripherals and memory (shown in Table 1), 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. These applications span a wide array of markets, from automotive systems to embedded industrial, instrumentation and power/motor control applications.

Table 1. Processor Comparison

Processor Feature	ADSP-BF606	ADSP-BF607	ADSP-BF608	ADSP-BF609
Up/Down/Rotary Counters	1			
Timer/Counters with PWM	8			
3-Phase PWM Units (4-pair)	2			
SPORTs	3			
SPIs	2			
USB OTG	1			
Parallel Peripheral Interface	3			
Removable Storage Interface	1			
CAN	1			
TWI	2			
UART	2			
ADC Control Module (ACM)	1			
Link Ports	4			
Ethernet MAC (IEEE 1588)	2			
Pixel Compositor (PIXC)	No		1	1
Pipelined Vision Processor (PVP) Video Resolution[1]	No		VGA	HD
Maximum PVP Line Buffer Size	N/A		640	1280
GPIOs	112			

Table 1. Processor Comparison (Continued)

Processor Feature		ADSP-BF606	ADSP-BF607	ADSP-BF608	ADSP-BF609
Memory (bytes, per core)	L1 Instruction SRAM	64K			
	L1 Instruction SRAM/Cache	16K			
	L1 Data SRAM	32K			
	L1 Data SRAM/Cache	32K			
	L1 Scratchpad	4K			
	L2 Data SRAM	128K	256K		
	L2 Boot ROM	32K			
Maximum Speed Grade (MHz)[2]		400	500		
Maximum SYSCLK (MHz)		250			
Package Options		349-Ball CSP_BGA			

[1] VGA is 640 × 480 pixels per frame. HD is 1280 × 960 pixels per frame.
[2] Maximum speed grade is not available with every possible SYSCLK selection.

BLACKFIN PROCESSOR CORE

As shown in Figure 1, the processor integrates two Blackfin processor cores. Each core, shown in Figure 2, 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 include byte alignment and packing operations, 16-bit and 8-bit adds with clipping, 8-bit average operations, and 8-bit subtract/absolute value/accumulate (SAA) operations. Also provided are the compare/select and vector search instructions.

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

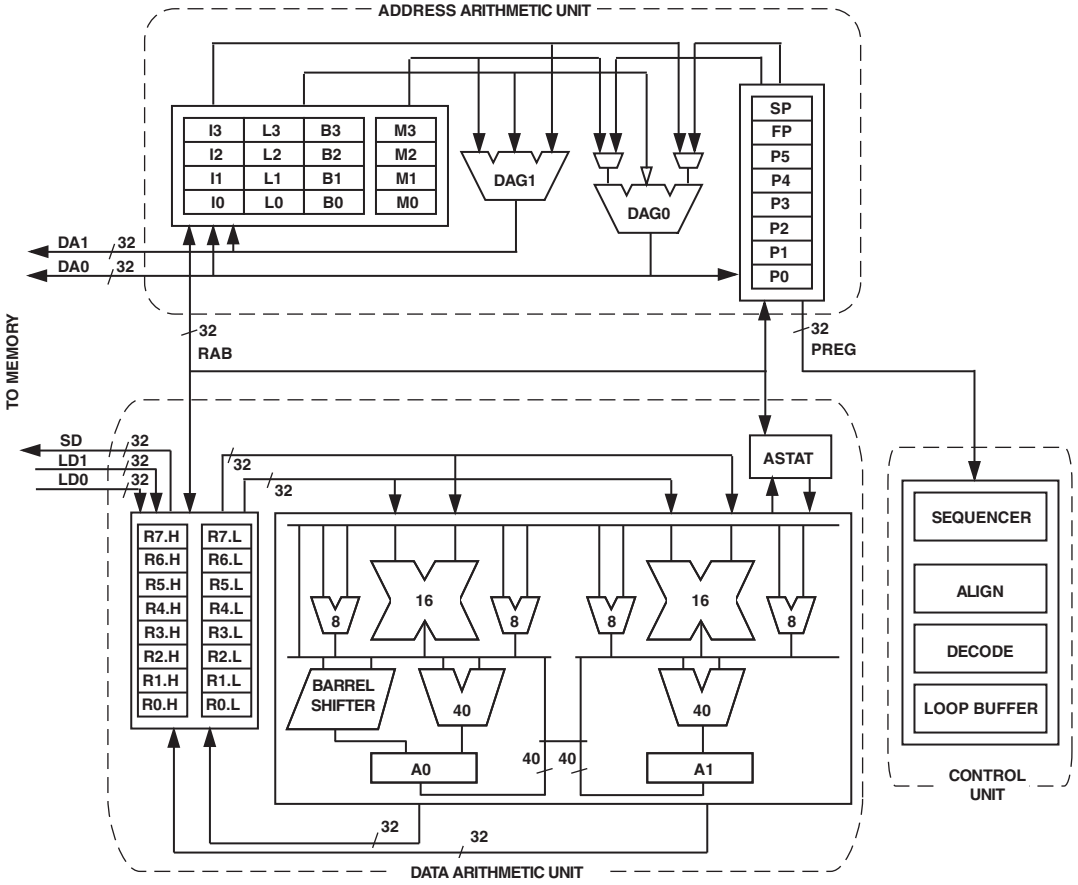


Figure 2. Blackfin Processor Core

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 supports zero-overhead looping. The architecture is fully interlocked, meaning that the programmer need not manage the pipeline when executing instructions with data dependencies.

The address arithmetic unit provides two addresses for simultaneous dual fetches from memory. It contains a multiported register file consisting of four sets of 32-bit index, modify, length, and base registers (for circular buffering), and eight additional 32-bit pointer registers (for C-style indexed stack manipulation).

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

INSTRUCTION SET DESCRIPTION

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 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.
- Control of all asynchronous and synchronous events to the processor is handled by two subsystems: the Core Event Controller (CEC) and the System Event Controller (SEC).
- 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.

PROCESSOR INFRASTRUCTURE

The following sections provide information on the primary infrastructure components of the ADSP-BF609 processor.

DMA Controllers

The processor uses Direct Memory Access (DMA) to transfer data within memory spaces or between a memory space and a peripheral. The processor can specify data transfer operations and return to normal processing while the fully integrated DMA controller carries out the data transfers independent of processor activity.

DMA transfers can occur between memory and a peripheral or between one memory and another memory. Each Memory-to-memory DMA stream uses two channels, where one channel is the source channel, and the second is the destination channel.

All DMAs can transport data to and from all on-chip and off-chip memories. Programs can use two types of DMA transfers, descriptor-based or register-based. Register-based DMA allows the processor to directly program DMA control registers to initiate a DMA transfer. On completion, the control registers may be automatically updated with their original setup values for continuous transfer. Descriptor-based DMA transfers require a set of parameters stored within memory to initiate a DMA sequence. Descriptor-based DMA transfers allow multiple DMA sequences to be chained together and a DMA channel can be programmed to automatically set up and start another DMA transfer after the current sequence completes.

The DMA controller supports the following DMA operations.

- A single linear buffer that stops on completion.
- A linear buffer with negative, positive or zero stride length.
- A circular, auto-refreshing buffer that interrupts when each buffer becomes full.
- A similar buffer that interrupts on fractional buffers (for example, 1/2, 1/4).
- 1D DMA – uses a set of identical ping-pong buffers defined by a linked ring of two-word descriptor sets, each containing a link pointer and an address.
- 1D DMA – uses a linked list of 4 word descriptor sets containing a link pointer, an address, a length, and a configuration.
- 2D DMA – uses an array of one-word descriptor sets, specifying only the base DMA address.
- 2D DMA – uses a linked list of multi-word descriptor sets, specifying everything.

CRC Protection

The two CRC protection modules allow system software to periodically calculate the signature of code and/or data in memory, the content of memory-mapped registers, or communication message objects. Dedicated hardware circuitry compares the signature with pre calculated values and triggers appropriate fault events.

For example, every 100 ms the system software might initiate the signature calculation of the entire memory contents and compare these contents with expected, pre calculated values. If a mismatch occurs, a fault condition can be generated (via the processor core or the trigger routing unit).

The CRC is a hardware module based on a CRC32 engine that computes the CRC value of the 32-bit data words presented to it. Data is provided by the source channel of the memory-to-memory DMA (in memory scan mode) and is optionally forwarded to the destination channel (memory transfer mode). The main features of the CRC peripheral are:

- Memory scan mode
- Memory transfer mode
- Data verify mode
- Data fill mode
- User-programmable CRC32 polynomial
- Bit/byte mirroring option (endianness)
- Fault/error interrupt mechanisms
- 1D and 2D fill block to initialize array with constants.
- 32-bit CRC signature of a block of a memory or MMR block.

Event Handling

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

- Emulation – An emulation event causes the processor to enter emulation mode, allowing command and control of the processor via the JTAG interface.
- Reset – This event resets the processor.
- Nonmaskable Interrupt (NMI) – The NMI event can be generated either by the software watchdog timer, by the NMI input signal to the processor, or by software. The NMI event is frequently used as a power-down indicator to initiate an orderly shutdown of the system.
- Exceptions – Events that occur synchronously to program flow (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.

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. For more information, see the *ADSP-BF60x Processor Programmer's Reference*.

System Event Controller (SEC)

The SEC manages the enabling, prioritization, and routing of events from each system interrupt or fault source. Additionally, it provides notification and identification of the highest priority active system interrupt request to each core and routes system fault sources to its integrated fault management unit.

Trigger Routing Unit (TRU)

The TRU provides system-level sequence control without core intervention. The TRU maps trigger masters (generators of triggers) to trigger slaves (receivers of triggers). Slave endpoints can be configured to respond to triggers in various ways. Common applications enabled by the TRU include:

- Automatically triggering the start of a DMA sequence after a sequence from another DMA channel completes
- Software triggering
- Synchronization of concurrent activities

Pin Interrupts

Every port pin on the processor can request interrupts in either an edge-sensitive or a level-sensitive manner with programmable polarity. Interrupt functionality is decoupled from GPIO operation. Six system-level interrupt channels (PINT0–5) are reserved for this purpose. Each of these interrupt channels can manage up to 32 interrupt pins. The assignment from pin to interrupt is not performed on a pin-by-pin basis. Rather, groups of eight pins (half ports) can be flexibly assigned to interrupt channels.

Every pin interrupt channel features a special set of 32-bit memory-mapped registers that enable half-port assignment and interrupt management. This includes masking, identification, and clearing of requests. These registers also enable access to the respective pin states and use of the interrupt latches, regardless of whether the interrupt is masked or not. Most control registers feature multiple MMR address entries to write-one-to-set or write-one-to-clear them individually.

General-Purpose I/O (GPIO)

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 – A "write one to modify" mechanism allows any combination of individual GPIO pins to be modified in a single instruction, without affecting the level of any other GPIO pins.
- GPIO interrupt mask registers – Allow each individual GPIO pin to function as an interrupt to the processor. 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 – 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.

Pin Multiplexing

The processor supports a flexible multiplexing scheme that multiplexes the GPIO pins with various peripherals. A maximum of 4 peripherals plus GPIO functionality is shared by each GPIO pin. All GPIO pins have a bypass path feature – that is, when the output enable and the input enable of a GPIO pin are both active, the data signal before the pad driver is looped back to the receive path for the same GPIO pin. For more information, see GP I/O Multiplexing for 349-Ball CSP_BGA on Page 33.

MEMORY ARCHITECTURE

The 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 larger, lower-cost and performance interface-accessible memory systems. See Figure 3 and Figure 4.

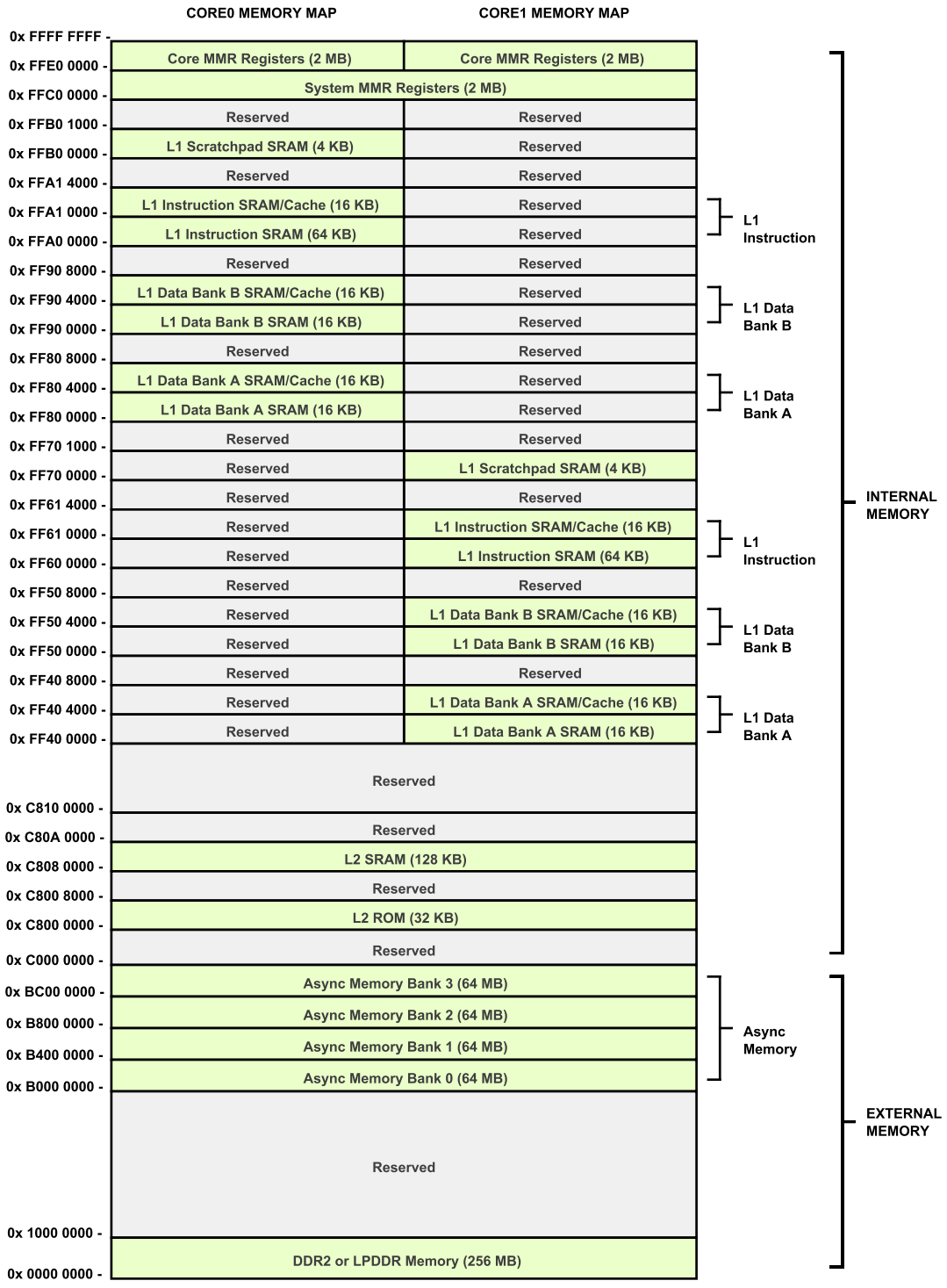


Figure 3. ADSP-BF606 Internal/External Memory Map

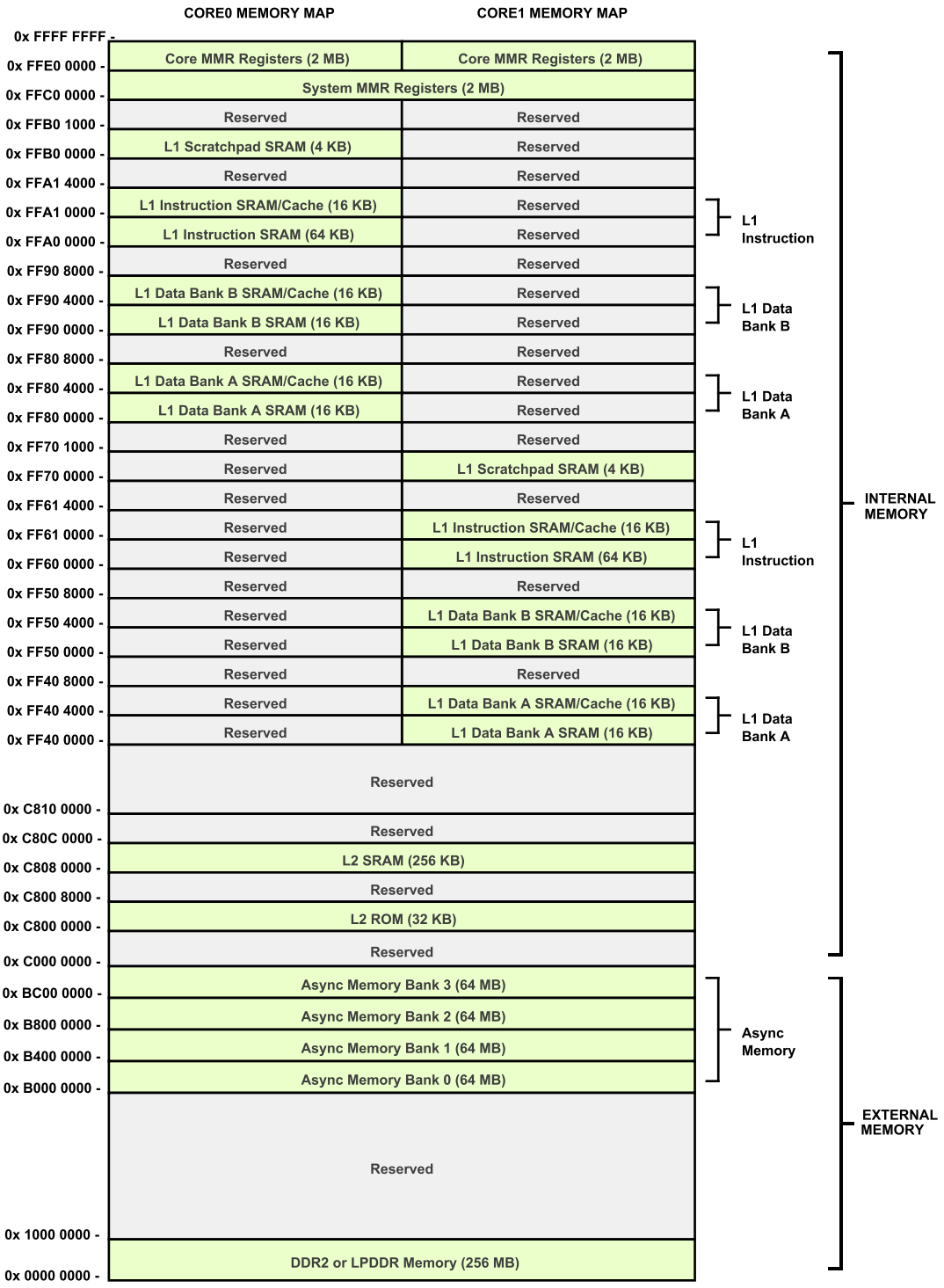


Figure 4. ADSP-BF607/ADSP-BF608/ADSP-BF609 Internal/External Memory Map

Internal (Core-Accessible) Memory

The L1 memory system is the highest-performance memory available to the Blackfin processor cores.

Each core has its own private L1 memory. The modified Harvard architecture supports two concurrent 32-bit data accesses along with an instruction fetch at full processor speed which provides high bandwidth processor performance. In each core a 64K-byte block of data memory partners with an 80K-byte memory block for instruction storage. Each data block is multi-banked for efficient data exchange through DMA and can be configured as SRAM. Alternatively, 16K bytes of each block can be configured in L1 cache mode. The four-way set-associative instruction cache and the 2 two-way set-associative data caches greatly accelerate memory access performance, especially when accessing external memories.

The L1 memory domain also features a 4K-byte scratchpad SRAM block which is ideal for storing local variables and the software stack. All L1 memory is protected by a multi-parity bit concept, regardless of whether the memory is operating in SRAM or cache mode.

Outside of the L1 domain, L2 and L3 memories are arranged using a Von Neumann topology. The L2 memory domain is a unified instruction and data memory and can hold any mixture of code and data required by the system design. The L2 memory domain is accessible by both Blackfin cores through a dedicated 64-bit interface. It operates at SYSCLK frequency.

The processor features up to 256K bytes of L2 SRAM which is ECC-protected and organized in eight banks. Individual banks can be made private to any of the cores or the DMA subsystem. There is also a 32K-byte single-bank ROM in the L2 domain. It contains boot code and safety functions.

Static Memory Controller (SMC)

The SMC can be programmed to control up to four banks of external memories or memory-mapped devices, with very flexible timing parameters. Each bank occupies a 64M byte segment regardless of the size of the device used, so that these banks are only contiguous if each is fully populated with 64M bytes of memory.

Dynamic Memory Controller (DMC)

The DMC includes a controller that supports JESD79-2E compatible double data rate (DDR2) SDRAM and JESD209A low power DDR (LPDDR) SDRAM devices.

I/O Memory Space

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

Booting

The processor has several mechanisms for automatically loading internal and external memory after a reset. The boot mode is defined by the SYS_BMODE input pins dedicated for this purpose. There are two categories of boot modes. In master boot modes, the processor actively loads data from parallel or serial memories. In slave boot modes, the processor receives data from external host devices.

The boot modes are shown in Table 2. These modes are implemented by the SYS_BMODE bits of the reset configuration register and are sampled during power-on resets and software-initiated resets.

Table 2. Boot Modes

SYS_BMODE Setting	Boot Mode
000	No boot/Idle
001	Memory
010	RSI0 Master
011	SPI0 Master
100	SPI0 Slave
101	Reserved
110	LP0 Slave
111	UART0 Slave

VIDEO SUBSYSTEM

The following sections describe the components of the processor's video subsystem. These blocks are shown with blue shading in Figure 1 on Page 1.

Video Interconnect (VID)

The Video Interconnect provides a connectivity matrix that interconnects the Video Subsystem: three PPIs, the PIXC, and the PVP. The interconnect uses a protocol to manage data transfer among these video peripherals.

Pipelined Vision Processor (PVP)

The PVP engine provides hardware implementation of signal and image processing algorithms that are required for co-processing and pre-processing of monochrome video frames in ADAS applications, robotic systems, and other machine applications.

The PVP works in conjunction with the Blackfin cores. It is optimized for convolution and wavelet based object detection and classification, and tracking and verification algorithms. The PVP has the following processing blocks.

- Four 5 × 5 16-bit convolution blocks optionally followed by down scaling
- A 16-bit cartesian-to-polar coordinate conversion block
- A pixel edge classifier that supports 1st and 2nd derivative modes
- An arithmetic unit with 32-bit addition, multiply and divide

- A 32-bit threshold block with 16 thresholds, a histogram, and run-length encoding
- Two 32-bit integral blocks that support regular and diagonal integrals
- An up- and down-scaling unit with independent scaling ratios for horizontal and vertical components
- Input and output formatters for compatibility with many data formats, including Bayer input format

The PVP can form a pipe of all the constituent algorithmic modules and is dynamically reconfigurable to form different pipeline structures.

The PVP supports the simultaneous processing of up to four data streams. The memory pipe stream operates on data received by DMA from any L1, L2, or L3 memory. The three camera pipe streams operate on a common input received directly from any of the three PPI inputs. Optionally, the PIXC can convert color data received by the PPI and forward luma values to the PVP's monochrome engine. Each stream has a dedicated DMA output. This preprocessing concept ensures careful use of available power and bandwidth budgets and frees up the processor cores for other tasks.

The PVP provides for direct core MMR access to all control/status registers. Two hardware interrupts interface to the system event controller. For optimal performance, the PVP allows register programming through its control DMA interface, as well as outputting selected status registers through the status DMA interface. This mechanism enables the PVP to automatically process job lists completely independent of the Blackfin cores.

Pixel Compositor (PIXC)

The pixel compositor (PIXC) provides image overlays with transparent-color support, alpha blending, and color space conversion capabilities for output to TFT LCDs and NTSC/PAL video encoders. It provides all of the control to allow two data streams from two separate data buffers to be combined, blended, and converted into appropriate forms for both LCD panels and digital video outputs. The main image buffer provides the basic background image, which is presented in the data stream. The overlay image buffer allows the user to add multiple foreground text, graphics, or video objects on top of the main image or video data stream.

Parallel Peripheral Interface (PPI)

The processor provides up to three parallel peripheral interfaces (PPIs), supporting data widths up to 24 bits. The PPI supports direct connection to TFT LCD panels, parallel analog-to-digital and digital-to-analog converters, video encoders and decoders, image sensor modules and other general-purpose peripherals.

The following features are supported in the PPI module:

- Programmable data length: 8 bits, 10 bits, 12 bits, 14 bits, 16 bits, 18 bits, and 24 bits per clock.
- Various framed, non-framed, and general-purpose operating modes. Frame syncs can be generated internally or can be supplied by an external device.
- ITU-656 status word error detection and correction for ITU-656 receive modes and ITU-656 preamble and status word decode.
- Optional packing and unpacking of data to/from 32 bits from/to 8 bits, 16 bits and 24 bits. If packing/unpacking is enabled, endianness can be configured to change the order of packing/unpacking of bytes/words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various de-interleaving/interleaving modes for receiving/transmitting 4:2:2 YCrCb data.
- Configurable LCD data enable (DEN) output available on Frame Sync 3.

PROCESSOR SAFETY FEATURES

The ADSP-BF60x processor has been designed for functional safety applications. While the level of safety is mainly dominated by the system concept, the following primitives are provided by the devices to build a robust safety concept.

Dual Core Supervision

The processor has been implemented as dual-core devices to separate critical tasks to large independency. Software models support mutual supervision of the cores in symmetrical fashion.

Multi-Parity-Bit-Protected L1 Memories

In the processor's L1 memory space, whether SRAM or cache, each word is protected by multiple parity bits to detect the single event upsets that occur in all RAMs. This applies both to L1 instruction and data memory spaces.

ECC-Protected L2 Memories

Error correcting codes (ECC) are used to correct single event upsets. The L2 memory is protected with a Single Error Correct-Double Error Detect (SEC-DED) code. By default ECC is enabled, but it can be disabled on a per-bank basis. Single-bit errors are transparently corrected. Dual-bit errors can issue a system event or fault if enabled. ECC protection is fully transparent to the user, even if L2 memory is read or written by 8-bit or 16-bit entities.

CRC-Protected Memories

While parity bit and ECC protection mainly protect against random soft errors in L1 and L2 memory cells, the CRC engines can be used to protect against systematic errors (pointer errors) and static content (instruction code) of L1, L2 and even L3 memories (DDR2, LPDDR). The processors feature two CRC engines which are embedded in the memory-to-memory DMA controllers. CRC check sums can be calculated or compared on the fly during memory transfers, or one or multiple memory regions can be continuously scrubbed by single DMA work unit as per DMA descriptor chain instructions. The CRC engine also protects data loaded during the boot process.

Memory Protection

The Blackfin cores feature a memory protection concept, which grants data and/or instruction accesses from enabled memory regions only. A supervisor mode vs. user mode programming model supports dynamically varying access rights. Increased flexibility in memory page size options supports a simple method of static memory partitioning.

System Protection

All system resources and L2 memory banks can be controlled by either the processor cores, memory-to-memory DMA, or the system debug unit (SDU). A system protection unit (SPU) enables write accesses to specific resources that are locked to any of four masters: Core 0, Core 1, Memory DMA, and the System Debug Unit. System protection is enabled in greater granularity for some modules (L2, SEC and GPIO controllers) through a *global lock* concept.

Watchpoint Protection

The primary purpose of watchpoints and hardware breakpoints is to serve emulator needs. When enabled, they signal an emulator event whenever user-defined system resources are accessed or a core executes from user-defined addresses. Watchpoint events can be configured such that they signal the events to the other Blackfin core or to the fault management unit.

Dual Watchdog

The two on-chip watchdog timers each may supervise one Blackfin core.

Bandwidth Monitor

All DMA channels that operate in memory-to-memory mode (Memory DMA, PVP Memory Pipe DMA, PIXC DMA) are equipped with a bandwidth monitor mechanism. They can signal a system event or fault when transactions tend to starve because system buses are fully loaded with higher-priority traffic.

Signal Watchdogs

The eight general-purpose timers feature two new modes to monitor off-chip signals. The Watchdog Period mode monitors whether external signals toggle with a period within an expected range. The Watchdog Width mode monitors whether the pulse widths of external signals are in an expected range. Both modes help to detect incorrect undesired toggling (or lack thereof) of system-level signals.

Up/Down Count Mismatch Detection

The up/down counter can monitor external signal pairs, such as request/grant strobes. If the edge count mismatch exceeds the expected range, the up/down counter can flag this to the processor or to the fault management unit.

Fault Management

The fault management unit is part of the system event controller (SEC). Any system event, whether a dual-bit uncorrectable ECC error, or any peripheral status interrupt, can be defined as being a "fault". Additionally, the system events can be defined as an interrupt to the cores. If defined as such, the SEC forwards the event to the fault management unit which may automatically reset the entire device for reboot, or simply toggle the SYS_FAULT output pins to signal off-chip hardware. Optionally, the fault management unit can delay the action taken via a keyed sequence, to provide a final chance for the Blackfin cores to resolve the crisis and to prevent the fault action from being taken.

ADDITIONAL PROCESSOR PERIPHERALS

The processor contains a rich set of peripherals connected to the core via several high-bandwidth buses, providing flexibility in system configuration as well as excellent overall system performance (see the block diagram on Page 1). The processors contain 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 following sections describe additional peripherals that were not described in the previous sections.

Timers

The processor includes several timers which are described in the following sections.

General-Purpose Timers

There is one GP timer unit and it provides eight general-purpose programmable timers. Each timer has an external pin that can be configured either as a pulse width modulator (PWM) or timer output, as an input to clock the timer, or as a mechanism for measuring pulse widths and periods of external events. These timers can be synchronized to an external clock input on the TMRx pins, an external clock TMRCLK input pin, or to the internal SCLK0.

The timer units can be used in conjunction with the UARTs and the CAN controller to measure the width of the pulses in the data stream to provide a software auto-baud detect function for the respective serial channels.

The timers can generate interrupts to the processor core, providing periodic events for synchronization to either the system clock or to external signals. Timer events can also trigger other peripherals via the TRU (for instance, to signal a fault).

Core Timers

Each processor core also has its own dedicated timer. This extra timer is clocked by the internal processor clock and is typically used as a system tick clock for generating periodic operating system interrupts.

Watchdog Timers

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

being reset by software. The programmer initializes the count value of the timer, enables the appropriate interrupt, then enables the timer. Thereafter, the software must reload the counter before it counts to zero from the programmed value. This protects the system from remaining in an unknown state where software, which would normally reset the timer, has stopped running due to an external noise condition or software error.

After a reset, software can determine if the watchdog was the source of the hardware reset by interrogating a status bit in the timer control register, which is set only upon a watchdog generated reset.

3-Phase PWM Units

The Pulse Width Modulator (PWM) module is a flexible and programmable waveform generator. With minimal CPU intervention the PWM peripheral is capable of generating complex waveforms for motor control, Pulse Coded Modulation (PCM), Digital to Analog Conversion (DAC), power switching and power conversion. The PWM module has 4 PWM pairs capable of 3-phase PWM generation for source inverters for AC induction and DC brush less motors.

The two 3-phase PWM generation units each feature:

- 16-bit center-based PWM generation unit
- Programmable PWM pulse width
- Single update mode with option for asymmetric duty
- Programmable dead time and switching frequency
- Twos-complement implementation which permits smooth transition to full ON and full OFF states
- Dedicated asynchronous PWM shutdown signal

Link Ports

Four DMA-enabled, 8-bit-wide link ports can connect to the link ports of other DSPs or processors. Link ports are bidirectional ports having eight data lines, an acknowledge line and a clock line.

Serial Ports (SPORTs)

Three synchronous serial ports that provide an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices such as Analog Devices' AD183x family of audio codecs, ADCs, and DACs. The serial ports are made up of two data lines, a clock, and frame sync. The data lines can be programmed to either transmit or receive and each data line has a dedicated DMA channel.

Serial port data can be automatically transferred to and from on-chip memory/external memory via dedicated DMA channels. Each of the serial ports can work in conjunction with another serial port to provide TDM support. In this configuration, one SPORT provides two transmit signals while the other SPORT provides the two receive signals. The frame sync and clock are shared.

Serial ports operate in five modes:

- Standard DSP serial mode
- Multichannel (TDM) mode
- I^2S mode
- Packed I^2S mode
- Left-justified mode

ACM Interface

The ADC control module (ACM) provides an interface that synchronizes the controls between the processor and an analog-to-digital converter (ADC). The analog-to-digital conversions are initiated by the processor, based on external or internal events.

The ACM allows for flexible scheduling of sampling instants and provides precise sampling signals to the ADC.

Figure 5 shows how to connect an external ADC to the ACM and one of the SPORTs.

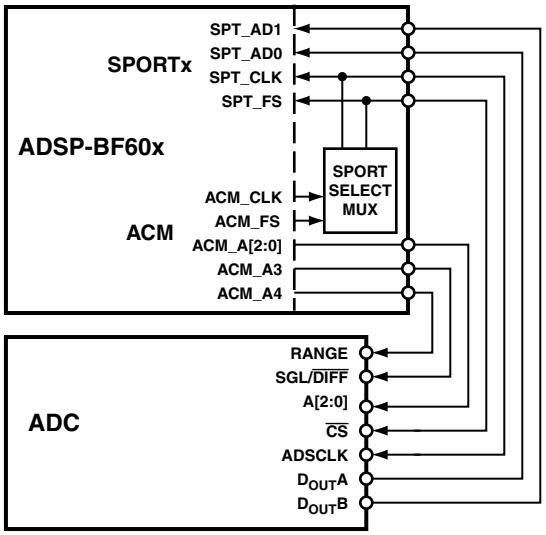


Figure 5. ADC, ACM, and SPORT Connections

The ACM synchronizes the ADC conversion process, generating the ADC controls, the ADC conversion start signal, and other signals. The actual data acquisition from the ADC is done by a peripheral such as a SPORT or a SPI.

The processor interfaces directly to many ADCs without any glue logic required.

General-Purpose Counters

A 32-bit counter is provided that can operate in general-purpose up/down count modes and can sense 2-bit quadrature or binary codes as typically emitted by industrial drives or manual thumbwheels. Count direction is either controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can alternatively be used to input the push-button signal of thumb wheels. All three pins have a programmable debouncing circuit.

Internal signals forwarded to each general-purpose timer enable these timers to measure the intervals between count events. Boundary registers enable auto-zero operation or simple system warning by interrupts when programmable count values are exceeded.

Serial Peripheral Interface (SPI) Ports

The processors have two SPI-compatible ports that allow the processor to communicate with multiple SPI-compatible devices.

In its simplest mode, the SPI interface uses three pins for transferring data: two data pins (Master Output-Slave Input, MOSI, and Master Input-Slave Output, MISO) and a clock pin (serial clock, SPI_CLK). A SPI chip select input pin ($\overline{\text{SPI\_SS}}$) lets other SPI devices select the processor, and seven SPI chip select output pins ($\overline{\text{SPI\_SEL7–1}}$) let the processor select other SPI devices. The SPI select pins are reconfigured general-purpose I/O pins. Using these pins, the SPI port provides a full-duplex, synchronous serial interface, which supports both master/slave modes and multimaster environments.

In a multi-master or multi-slave SPI system, the MOSI and MISO data output pins can be configured to behave as open drain outputs (using the ODM bit) to prevent contention and possible damage to pin drivers. An external pull-up resistor is required on both the MOSI and MISO pins when this option is selected.

When ODM is set and the SPI is configured as a master, the MOSI pin is three-stated when the data driven out on MOSI is a logic-high. The MOSI pin is not three-stated when the driven data is a logic-low. Similarly, when ODM is set and the SPI is configured as a slave, the MISO pin is three-stated if the data driven out on MISO is a logic-high.

The SPI port's baud rate and clock phase/polarities are programmable, and it has integrated DMA channels for both transmit and receive data streams.

UART Ports

The processors provide two full-duplex universal asynchronous receiver/transmitter (UART) ports, which are fully compatible with PC-standard UARTs. Each UART port provides a simplified UART interface to other peripherals or hosts, supporting full-duplex, DMA-supported, asynchronous transfers of serial data. A UART port includes support for five to eight data bits, and none, even, or odd parity. Optionally, an additional address bit can be transferred to interrupt only addressed nodes in multi-drop bus (MDB) systems. A frame is terminates by one, one and a half, two or two and a half stop bits.

The UART ports support automatic hardware flow control through the Clear To Send (CTS) input and Request To Send (RTS) output with programmable assertion FIFO levels.

To help support the Local Interconnect Network (LIN) protocols, a special command causes the transmitter to queue a break command of programmable bit length into the transmit buffer. Similarly, the number of stop bits can be extended by a programmable inter-frame space.

The capabilities of the UARTs are further extended with support for the Infrared Data Association (IrDA®) serial infrared physical layer link specification (SIR) protocol.

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 module is compatible with the widely used I^2C 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 (TWI_SCL) and data (TWI_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.

Removable Storage Interface (RSI)

The removable storage interface (RSI) controller acts as the host interface for multimedia cards (MMC), secure digital memory cards (SD), secure digital input/output cards (SDIO). The following list describes the main features of the RSI controller.

- Support for a single MMC, SD memory, SDIO card
- Support for 1-bit and 4-bit SD modes
- Support for 1-bit, 4-bit, and 8-bit MMC modes
- Support for eMMC 4.3 embedded NAND flash devices
- A ten-signal external interface with clock, command, and up to eight data lines
- Card interface clock generation from SCLK0
- SDIO interrupt and read wait features

Controller Area Network (CAN)

A CAN controller implements the CAN 2.0B (active) protocol. This protocol is an asynchronous communications protocol used in both industrial and automotive control systems. The CAN protocol is well suited for control applications due to its capability to communicate reliably over a network. This is because the protocol incorporates CRC checking, message error tracking, and fault node confinement.

The CAN controller offers the following features:

- 32 mailboxes (8 receive only, 8 transmit only, 16 configurable for receive or transmit).
- Dedicated acceptance masks for each mailbox.
- Additional data filtering on first two bytes.
- Support for both the standard (11-bit) and extended (29-bit) identifier (ID) message formats.

- Support for remote frames.
- Active or passive network support.
- CAN wakeup from hibernation mode (lowest static power consumption mode).
- Interrupts, including: TX complete, RX complete, error and global.

An additional crystal is not required to supply the CAN clock, as the CAN clock is derived from a system clock through a programmable divider.

10/100 Ethernet MAC

The processor can directly connect to a network by way of an embedded fast Ethernet media access controller (MAC) that supports both 10-BaseT (10M bits/sec) and 100-BaseT (100M bits/sec) operation. The 10/100 Ethernet MAC peripheral on the processor is fully compliant to the IEEE 802.3-2002 standard and it provides programmable features designed to minimize supervision, bus use, or message processing by the rest of the processor system.

Some standard features are:

- Support and RMII protocols for external PHYs
- Full duplex and half duplex modes
- Media access management (in half-duplex operation)
- Flow control
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers

Some advanced features are:

- Automatic checksum computation of IP header and IP payload fields of RX frames
- Independent 32-bit descriptor-driven receive and transmit DMA channels
- Frame status delivery to memory through DMA, including frame completion semaphores for efficient buffer queue management in software
- TX DMA support for separate descriptors for MAC header and payload to eliminate buffer copy operations
- Convenient frame alignment modes
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value
- Advanced power management
- Magic packet detection and wakeup frame filtering
- Support for 802.3Q tagged VLAN frames
- Programmable MDC clock rate and preamble suppression

IEEE 1588 Support

The IEEE 1588 standard is a precision clock synchronization protocol for networked measurement and control systems. The processor includes hardware support for IEEE 1588 with an integrated precision time protocol synchronization engine (PTP_TSYNC). This engine provides hardware assisted time stamping to improve the accuracy of clock synchronization between PTP nodes. The main features of the engine are:

- Support for both IEEE 1588-2002 and IEEE 1588-2008 protocol standards
- Hardware assisted time stamping capable of up to 12.5 ns resolution
- Lock adjustment
- Automatic detection of IPv4 and IPv6 packets, as well as PTP messages
- Multiple input clock sources (SCLK0, RMII clock, external clock)
- Programmable pulse per second (PPS) output
- Auxiliary snapshot to time stamp external events

USB 2.0 On-the-Go Dual-Role Device Controller

The USB 2.0 OTG dual-role device controller provides a low-cost connectivity solution for the growing adoption of this bus standard in industrial applications, as well as consumer mobile devices such as cell phones, digital still cameras, and MP3 players. The USB 2.0 controller allows these devices to transfer data using a point-to-point USB connection without the need for a PC host. The module can operate in a traditional USB peripheral-only mode as well as the host mode presented in the On-the-Go (OTG) supplement to the USB 2.0 specification.

The USB clock (USB_CLKIN) is provided through a dedicated external crystal or crystal oscillator.

The USB On-the-Go dual-role device controller includes a Phase Locked Loop with programmable multipliers to generate the necessary internal clocking frequency for USB.

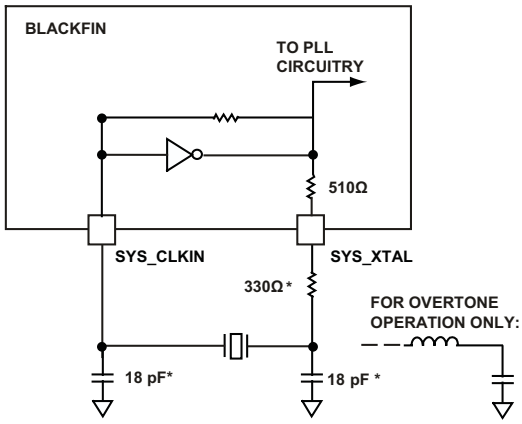
POWER AND CLOCK MANAGEMENT

The processor provides four operating modes, each with a different performance/power profile. When configured for a 0 V internal supply voltage (V_{DD_INT}), the processor enters the hibernate state. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 5 for a summary of the power settings for each mode.

Crystal Oscillator (SYS_XTAL)

The processor can be clocked by an external crystal (Figure 6), 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 SYS_CLKIN pin. When an external clock is used, the SYS_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 6. A parallel-resonant, fundamental frequency, microprocessor grade crystal is connected across the SYS_CLKIN and XTAL pins. The on-chip resistance between SYS_CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended.



NOTE: VALUES MARKED WITH * MUST BE CUSTOMIZED, DEPENDING ON THE CRYSTAL AND LAYOUT. PLEASE ANALYZE CAREFULLY. FOR FREQUENCIES ABOVE 33 MHz, THE SUGGESTED CAPACITOR VALUE OF 18pF SHOULD BE TREATED AS A MAXIMUM, AND THE SUGGESTED RESISTOR VALUE SHOULD BE REDUCED TO 0 Ω.

Figure 6. External Crystal Connection

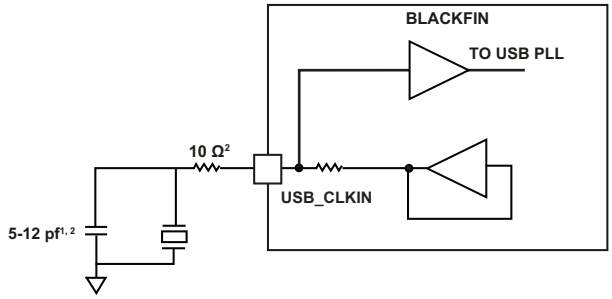
The two capacitors and the series resistor shown in Figure 6 fine tune phase and amplitude of the sine frequency. The capacitor and resistor values shown in Figure 6 are typical values only. The capacitor values are dependent upon the crystal manufacturers' load capacitance recommendations and the PCB physical layout. The resistor value depends on the drive level specified by the crystal manufacturer. The user should verify the customized values based on careful investigations on multiple devices over temperature range.

A third-overtone crystal can be used for frequencies above 25 MHz. The circuit is then modified to ensure crystal operation only at the third overtone by adding a tuned inductor circuit as shown in Figure 6. A design procedure for third-overtone operation is discussed in detail in application note (EE-168) Using Third Overtone Crystals with the ADSP-218x DSP on the Analog Devices website (www.analog.com)—use site search on "EE-168."

USB Crystal Oscillator

The USB 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 USB_CLKIN pin. 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 7. A parallel-resonant, fundamental frequency, microprocessor grade crystal is connected between the USB_CLKIN pin and ground. A load capacitor is placed in parallel with the crystal. The combined capacitive value of the board trace parasitic, the case capacitance of the crystal (from crystal manufacturer) and the parallel capacitor in the diagram should be in the range of 8 pF to 15 pF.



NOTES:
1. CAPACITANCE VALUE SHOWN INCLUDES BOARD PARASITICS
2. VALUES ARE A PRELIMINARY ESTIMATE.

Figure 7. External USB Crystal Connection

The crystal should be chosen so that its rated load capacitance matches the nominal total capacitance on this node. A series resistor may be added between the USB_CLKIN pin and the parallel crystal and capacitor combination, in order to further reduce the drive level of the crystal.

The parallel capacitor and the series resistor shown in Figure 7 fine tune phase and amplitude of the sine frequency. The capacitor and resistor values shown in Figure 7 are typical values only. The capacitor values are dependent upon the crystal manufacturers' load capacitance recommendations and the PCB physical layout. The resistor value depends on the drive level specified by the crystal manufacturer. The user should verify the customized values based on careful investigations on multiple devices over temperature range.

Clock Generation

The clock generation unit (CGU) generates all on-chip clocks and synchronization signals. Multiplication factors are programmed to the PLL to define the PLLCLK frequency. Programmable values divide the PLLCLK frequency to generate the core clock (CCLK), the system clocks (SYSCLK, SCLK0 and SCLK1), the LPDDR or DDR2 clock (DCLK) and the output clock (OCLK). This is illustrated in Figure 8 on Page 54.

Writing to the CGU control registers does not affect the behavior of the PLL immediately. Registers are first programmed with a new value, and the PLL logic executes the changes so that it transitions smoothly from the current conditions to the new ones.

SYS_CLKIN oscillations start when power is applied to the V_{DD_EXT} pins. The rising edge of $\overline{\text{SYS\_HWRST}}$ can be applied after all voltage supplies are within specifications (see Operating Conditions on Page 52), and SYS_CLKIN oscillations are stable.

Clock Out/External Clock

The SYS_CLKOUT output pin has programmable options to output divided-down versions of the on-chip clocks. By default, the SYS_CLKOUT pin drives a buffered version of the SYS_CLKIN input. Clock generation faults (for example PLL unlock) may trigger a reset by hardware. The clocks shown in Table 3 can be outputs from SYS_CLKOUT.

Table 3. Clock Dividers

Clock Source	Divider
CCLK (core clock)	By 4
SYSCLK (System clock)	By 2
SCLK0 (system clock for PVP, all peripherals not covered by SCLK1)	None
SCLK1 (system clock for SPORTS, SPI, ACM)	None
DCLK (LPDDR/DDR2 clock)	By 2
OCLK (output clock)	Programmable
CLKBUF	None, direct from SYS_CLKIN

Power Management

As shown in Table 4, the processor supports five different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions. 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 4. Power Domains

Power Domain	V_{DD} Range
All internal logic	V_{DD_INT}
DDR2/LPDDR	V_{DD_DMC}
USB	V_{DD_USB}
Thermal diode	V_{DD_TD}
All other I/O (includes SYS, JTAG, and Ports pins)	V_{DD_EXT}

The dynamic power management feature of the processor allows the processor's core 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.

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 cores 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 clocks and system clocks run at the input clock (SYS_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-BF60x Blackfin Processor Hardware Reference.*

See Table 5 for a summary of the power settings for each mode.

Table 5. Power Settings

Mode/State	PLL	PLL Bypassed	f_{CCLK}	f_{SYSCLK}, f_{DCLK}, f_{SCLK0}, f_{SCLK1}	Core Power
Full On	Enabled	No	Enabled	Enabled	On
Active	Enabled/ Disabled	Yes	Enabled	Enabled	On
Deep Sleep	Disabled	—	Disabled	Disabled	On
Hibernate	Disabled	—	Disabled	Disabled	Off

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core and to all synchronous peripherals. Asynchronous peripherals may still be running but cannot access internal resources or external memory.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor cores and to all of the peripherals. This setting signals the external voltage regulator supplying the V_{DD_INT} pins to shut off using the SYS_EXTWAKE signal, which provides 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.

Since the V_{DD_EXT} pins 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.

Reset Control Unit

Reset is the initial state of the whole processor or one of the cores and is the result of a hardware or software triggered event. In this state, all control registers are set to their default values and functional units are idle. Exiting a full system reset starts with Core-0 only being ready to boot. Exiting a Core-n only reset starts with this Core-n being ready to boot.

The Reset Control Unit (RCU) controls how all the functional units enter and exit reset. Differences in functional requirements and clocking constraints define how reset signals are generated. Programs must guarantee that none of the reset functions puts the system into an undefined state or causes resources to stall. This is particularly important when only one of the cores is reset (programs must ensure that there is no pending system activity involving the core that is being reset).

From a system perspective reset is defined by both the reset target and the reset source as described below.

Target defined:

- Hardware Reset – All functional units are set to their default states without exception. History is lost.
- System Reset – All functional units except the RCU are set to their default states.
- Core-n only Reset – Affects Core-n only. The system software should guarantee that the core in reset state is not accessed by any bus master.

Source defined:

- Hardware Reset – The $\overline{\text{SYS\_HWRST}}$ input signal is asserted active (pulled down).
- System Reset – May be triggered by software (writing to the RCU_CTL register) or by another functional unit such as the dynamic power management (DPM) unit (Hibernate) or any of the system event controller (SEC), trigger routing unit (TRU), or emulator inputs.
- Core-n-only reset – Triggered by software.
- Trigger request (peripheral).

Voltage Regulation

The processor requires an external voltage regulator to power the V_{DD_INT} pins. To reduce standby power consumption, the external voltage regulator can be signaled through SYS_EXTWAKE 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, all external supply pins (V_{DD_EXT}, V_{DD_USB}, V_{DD_DMC}) can still be powered, eliminating the need for external buffers. The external voltage regulator can be activated from this power down state by asserting the $\overline{\text{SYS\_HWRST}}$ pin, which then initiates a boot sequence. SYS_EXTWAKE indicates a wakeup to the external voltage regulator.

SYSTEM DEBUG

The processor includes various features that allow for easy system debug. These are described in the following sections.

System Watchpoint Unit

The System Watchpoint Unit (SWU) is a single module which connects to a single system bus and provides for transaction monitoring. One SWU is attached to the bus going to each system slave. The SWU provides ports for all system bus address channel signals. Each SWU contains four match groups of registers with associated hardware. These four SWU match groups operate independently, but share common event (interrupt, trigger and others) outputs.

System Debug Unit

The System Debug Unit (SDU) provides IEEE-1149.1 support through its JTAG interface. In addition to traditional JTAG features, present in legacy Blackfin products, the SDU adds more features for debugging the chip without halting the core processors.

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".

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 *EE-68: Analog Devices JTAG Emulation Technical Reference* 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-BF606/ADSP-BF607/ADSP-BF608/ADSP-BF609 processors (and related processors) can be ordered from any Analog Devices sales office or accessed electronically on our website:

- *Getting Started With Blackfin Processors*
- *ADSP-BF60x Blackfin Processor Hardware Reference*
- *Blackfin Processor Programming Reference*
- *ADSP-BF60x 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 Application Signal Chains page in the Circuits from the Lab™ site (http:\\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

ADSP-BF60x DETAILED SIGNAL DESCRIPTIONS

Table 6 provides a detailed description of each signal.

Table 6. Detailed Signal Descriptions

Signal Name	Direction	Description
ACM_An	Output	**ADC Control Signals** Function varies by mode.
ACM_CLK	Output	**Clock** SCLK derived clock for connecting to an ADC.
ACM_FS	Output	**Frame Sync** Typically used as an ADC chip select.
ACM_Tn	Input	**External Trigger n** Input for external trigger events.
CAN_RX	Input	**Receive** Typically an external CAN transceiver's RX output.
CAN_TX	Output	**Transmit** Typically an external CAN transceiver's TX input.
CNT_DG	Input	**Count Down and Gate** Depending on the mode of operation this input acts either as a count down signal or a gate signal. Count Down: This input causes the GP counter to decrement. Gate: Stops the GP counter from incrementing or decrementing.
CNT_UD	Input	**Count Up and Direction** Depending on the mode of operation this input acts either as a count up signal or a direction signal. Count Up: This input causes the GP counter to increment. Direction: Selects whether the GP counter is incrementing or decrementing.
CNT_ZM	Input	**Count Zero Marker** Input that connects to the zero marker output of a rotary device or detects the pressing of a push button.
DMC_Ann	Output	**Address n** Address bus.
DMC_BAn	Output	**Bank Address Input n** Defines which internal bank an ACTIVATE, READ, WRITE, or PRECHARGE command is being applied to on the dynamic memory. Also defines which mode registers (MR, EMR, EMR2, and/or EMR3) are loaded during the LOAD MODE REGISTER command.
$\overline{\text{DMC\_CAS}}$	Output	**Column Address Strobe** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the CAS input of dynamic memory.
$\overline{\text{DMC\_CK}}$	Output	**Clock (complement)** Complement of DMC_CK.
DMC_CK	Output	**Clock** Outputs DCLK to external dynamic memory.
DMC_CKE	Output	**Clock enable** Active high clock enables. Connects to the dynamic memory's CKE input.
$\overline{\text{DMC\_CSn}}$	Output	**Chip Select n** Commands are recognized by the memory only when this signal is asserted.
DMC_DQnn	I/O	**Data n** Bidirectional data bus.
DMC_LDM	Output	**Data Mask for Lower Byte** Mask for DMC_DQ07:DMC_DQ00 write data when driven high. Sampled on both edges of the data strobe by the dynamic memory.
$\overline{\text{DMC\_LDQS}}$	I/O	**Data Strobe for Lower Byte (complement)** Complement of LDQS. Not used in single-ended mode.
DMC_LDQS	I/O	**Data Strobe for Lower Byte** DMC_DQ07:DMC_DQ00 data strobe. Output with Write Data. Input with Read Data. May be single-ended or differential depending on register settings.
DMC_ODT	Output	**On-die Termination** Enables dynamic memory termination resistances when driven high (assuming the memory is properly configured). ODT is enabled/disabled regardless of read or write commands.
$\overline{\text{DMC\_RAS}}$	Output	**Row Address Strobe** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the RAS input of dynamic memory.
DMC_UDM	Output	**Data Mask for Upper Byte** Mask for DMC_DQ15:DMC_DQ08 write data when driven high. Sampled on both edges of the data strobe by the dynamic memory.
$\overline{\text{DMC\_UDQS}}$	I/O	**Data Strobe for Upper Byte (complement)** Complement of UDQS. Not used in single-ended mode.
DMC_UDQS	I/O	**Data Strobe for Upper Byte** DMC_DQ15:DMC_DQ08 data strobe. Output with Write Data. Input with Read Data. May be single-ended or differential depending on register settings.
$\overline{\text{DMC\_WE}}$	Output	**Write Enable** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the $\overline{\text{WE}}$ input of dynamic memory.

Table 6. Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
ETH_CRS	Input	**Carrier Sense/RMII Receive Data Valid** Multiplexed on alternate clock cycles. CRS: Asserted by the PHY when either the transmit or receive medium is not idle. De-asserted when both are idle. RXDV: Asserted by the PHY when the data on RXDn is valid.
ETH_MDC	Output	**Management Channel Clock** Clocks the MDC input of the PHY.
ETH_MDIO	I/O	**Management Channel Serial Data** Bidirectional data bus for PHY control.
ETH_PTPAUXIN	Input	**PTP Auxiliary Trigger Input** Assert this signal to take an auxiliary snapshot of the time and store it in the auxiliary time stamp FIFO.
ETH_PTPCLKIN	Input	**PTP Clock Input** Optional external PTP clock input.
ETH_PTPPPS	Output	**PTP Pulse-Per-Second Output** When the Advanced Time Stamp feature is enabled, this signal is asserted based on the PPS mode selected. Otherwise, PTPPPS is asserted every time the seconds counter is incremented.
ETH_REFCLK	Input	**Reference Clock** Externally supplied Ethernet clock.
ETH_RXDn	Input	**Receive Data n** Receive data bus.
ETH_TXDn	Output	**Transmit Data n** Transmit data bus.
ETH_TXEN	I/O	**Transmit Enable** When asserted indicates that the data on TXDn is valid.
$\overline{\text{JTG\_EMU}}$	Output	**Emulation Output** JTAG emulation flag.
JTG_TCK	Input	**Clock** JTAG test access port clock.
JTG_TDI	Input	**Serial Data In** JTAG test access port data input.
JTG_TDO	Output	**Serial Data Out** JTAG test access port data output.
JTG_TMS	Input	**Mode Select** JTAG test access port mode select.
$\overline{\text{JTG\_TRST}}$	Input	**Reset** JTAG test access port reset.
LP_ACK	I/O	**Acknowledge** Provides handshaking. When the link port is configured as a receiver, ACK is an output. When the link port is configured as a transmitter, ACK is an input.
LP_CLK	I/O	**Clock** When the link port is configured as a receiver, CLK is an input. When the link port is configured as a transmitter, CLK is an output.
LP_Dn	I/O	**Data n** Data bus. Input when receiving, output when transmitting.
PPI_CLK	I/O	**Clock** Input in external clock mode, output in internal clock mode.
PPI_Dnn	I/O	**Data n** Bidirectional data bus.
PPI_FS1	I/O	**Frame Sync 1 (HSYNC)** Behavior depends on PPI mode. See the PPI chapter in the processor hardware reference for more details.
PPI_FS2	I/O	**Frame Sync 2 (VSYNC)** Behavior depends on PPI mode. See the PPI chapter in the processor hardware reference for more details.
PPI_FS3	I/O	**Frame Sync 3 (FIELD)** Behavior depends on PPI mode. See the PPI chapter in the processor hardware reference for more details.
PWM_AH	Output	**Channel A High Side** High side drive signal.
PWM_AL	Output	**Channel A Low Side** Low side drive signal.
PWM_BH	Output	**Channel B High Side** High side drive signal.
PWM_BL	Output	**Channel B Low Side** Low side drive signal.
PWM_CH	Output	**Channel C High Side** High side drive signal.
PWM_CL	Output	**Channel C Low Side** Low side drive signal.
PWM_DH	Output	**Channel D High Side** High side drive signal.
PWM_DL	Output	**Channel D Low Side** Low side drive signal.
PWM_SYNC	Input	**PWM External Sync** This input is for an externally generated sync signal. If the sync signal is internally generated no connection is necessary.
$\overline{\text{PWM\_TRIPn}}$	Input	**Shutdown Input n** When asserted the selected PWM channel outputs are shut down immediately.
Px_nn	I/O	**Position n** General purpose input/output. See the GP Ports chapter in the processor hardware reference for programming information.

Table 6. Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
RSI_CLK	Output	**Clock** The clock signal applied to the connected device from the RSI.
RSI_CMD	I/O	**Command** Used to send commands to and receive responses from the connected device.
RSI_Dn	I/O	**Data n** Bidirectional data bus.
SMC_ABEn	Output	**Byte Enable n** Indicate whether the lower or upper byte of a memory is being accessed. When an asynchronous write is made to the upper byte of a 16-bit memory, SMC_ABE1 =0 and SMC_ABE0 =1. When an asynchronous write is made to the lower byte of a 16-bit memory, SMC_ABE1 =1 and SMC_ABE0 =0.
SMC_AMSn	Output	**Memory Select n** Typically connects to the chip select of a memory device.
SMC_Ann	Output	**Address n** Address bus.
SMC_AOE	Output	**Output Enable** Asserts at the beginning of the setup period of a read access.
SMC_ARDY	Input	**Asynchronous Ready** Flow control signal used by memory devices to indicate to the SMC when further transactions may proceed.
SMC_ARE	Output	**Read Enable** Asserts at the beginning of a read access.
SMC_AWE	Output	**Write Enable** Asserts for the duration of a write access period.
SMC_BG	Output	**Bus Grant** Output used to indicate to an external device that it has been granted control of the SMC buses.
SMC_BGH	Output	**Bus Grant Hang** Output used to indicate that the SMC has a pending transaction which requires control of the bus to be restored before it can be completed.
SMC_BR	Input	**Bus Request** Input used by an external device to indicate that it is requesting control of the SMC buses.
SMC_Dnn	I/O	**Data n** Bidirectional data bus.
SMC_NORCLK	Output	**NOR Clock** Clock for synchronous burst mode.
SMC_NORDV	Output	**NOR Data Valid** Asserts for the duration of a synchronous burst mode read setup period.
SMC_NORWT	Input	**NOR Wait** Flow control signal used by memory devices in synchronous burst mode to indicate to the SMC when further transactions may proceed.
SPI_CLK	I/O	**Clock** Input in slave mode, output in master mode.
SPI_D2	I/O	**Data 2** Used to transfer serial data in quad mode. Open drain in ODM mode.
SPI_D3	I/O	**Data 3** Used to transfer serial data in quad mode. Open drain in ODM mode.
SPI_MISO	I/O	**Master In, Slave Out** Used to transfer serial data. Operates in the same direction as SPI_MOSI in dual and quad modes. Open drain in ODM mode.
SPI_MOSI	I/O	**Master Out, Slave In** Used to transfer serial data. Operates in the same direction as SPI_MISO in dual and quad modes. Open drain in ODM mode.
SPI_RDY	I/O	**Ready** Optional flow signal. Output in slave mode, input in master mode.
SPI_SELn	Output	**Slave Select Output n** Used in master mode to enable the desired slave.
SPI_SS	Input	**Slave Select Input** Slave mode: acts as the slave select input. Master mode: optionally serves as an error detection input for the SPI when there are multiple masters.
SPT_ACLK	I/O	**Channel A Clock** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_AD0	I/O	**Channel A Data 0** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_AD1	I/O	**Channel A Data 1** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_AFS	I/O	**Channel A Frame Sync** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_ATDV	Output	**Channel A Transmit Data Valid** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SPT_BCLK	I/O	**Channel B Clock** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_BD0	I/O	**Channel B Data 0** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.

Table 6. Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
SPT_BD1	I/O	**Channel B Data 1** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_BFS	I/O	**Channel B Frame Sync** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_BTDV	Output	**Channel B Transmit Data Valid** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SYS_BMODEn	Input	**Boot Mode Control n** Selects the boot mode of the processor.
SYS_CLKIN	Input	**Clock/Crystal Input** Connect to an external clock source or crystal.
SYS_CLKOUT	Output	**Processor Clock Output** Outputs internal clocks. Clocks may be divided down. See the CGU chapter in the processor hardware reference for more details.
SYS_EXTWAKE	Output	**External Wake Control** Drives low during hibernate and high all other times. Typically connected to the enable input of the voltage regulator controlling the V_{DD_INT} supply.
$\overline{\text{SYS\_FAULT}}$	I/O	**Complementary Fault** Complement of SYS_FAULT.
SYS_FAULT	I/O	**Fault** Indicates internal faults or senses external faults depending on the operating mode.
$\overline{\text{SYS\_HWRST}}$	Input	**Processor Hardware Reset Control** Resets the device when asserted.
$\overline{\text{SYS\_IDLEn}}$	Output	**Core n Idle Indicator** When low indicates that core n is in idle mode or being held in reset.
$\overline{\text{SYS\_NMI}}$	Input	**Non-maskable Interrupt** Priority depends on the core that receives the interrupt. See the processor hardware and programming references for more details.
SYS_PWRGD	Input	**Power Good Indicator** When high it indicates to the processor that the V_{DD_INT} level is within specifications such that it is safe to begin booting upon return from hibernate.
$\overline{\text{SYS\_RESOUT}}$	Output	**Reset Output** Indicates that the device is in the reset state.
$\overline{\text{SYS\_SLEEP}}$	Output	**Processor Sleep Indicator** When low indicates that the processor is in the deep sleep power saving mode.
SYS_TDA	Input	**Thermal Diode Anode** May be used by an external temperature sensor to measure the die temperature.
SYS_TDK	Input	**Thermal Diode Cathode** May be used by an external temperature sensor to measure the die temperature.
SYS_XTAL	Output	**Crystal Output** Drives an external crystal. Must be left unconnected if an external clock is driving CLKIN.
TMR_ACIn	Input	**Alternate Capture Input n** Provides an additional input for WIDCAP, WATCHDOG, and PININT modes.
TMR_ACLKn	Input	**Alternate Clock n** Provides an additional time base for use by an individual timer.
TMR_CLK	Input	**Clock** Provides an additional global time base for use by all the GP timers.
TMR_TMRn	I/O	**Timer n** The main input/output signal for each timer.
TWI_SCL	I/O	**Serial Clock** Clock output when master, clock input when slave.
TWI_SDA	I/O	**Serial Data** Receives or transmits data.
$\overline{\text{UART\_CTS}}$	Input	**Clear to Send** Flow control signal.
$\overline{\text{UART\_RTS}}$	Output	**Request to Send** Flow control signal.
$\overline{\text{UART\_RX}}$	Input	**Receive** Receive input. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
$\overline{\text{UART\_TX}}$	Output	**Transmit** Transmit output. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
USB_CLKIN	Input	**Clock/Crystal Input** This clock input is multiplied by a PLL to form the USB clock. See Universal Serial Bus (USB) On-The-Go—Receive and Transmit Timing for frequency/tolerance information.
USB_DM	I/O	**Data –** Bidirectional differential data line.
USB_DP	I/O	**Data +** Bidirectional differential data line.
USB_ID	Input	**OTG ID** Senses whether the controller is a host or device. This signal is pulled low when an A-type plug is sensed (signifying that the USB controller is the A device), but the input is high when a B-type plug is sensed (signifying that the USB controller is the B device).
USB_VBC	Output	**VBUS Control** Controls an external voltage source to supply VBUS when in host mode. May be configured as open drain. Polarity is configurable as well.
USB_VBUS	I/O	**Bus Voltage** Connects to bus voltage in host and device modes.

349-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processors' pin definitions are shown in the table. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the Signal Name for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- Port: The General-Purpose I/O Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power-on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
ACM0_A0	ACM0 Address 0	F	PF_14
ACM0_A1	ACM0 Address 1	F	PF_15
ACM0_A2	ACM0 Address 2	F	PF_12
ACM0_A3	ACM0 Address 3	F	PF_13
ACM0_A4	ACM0 Address 4	F	PF_10
ACM0_CLK	ACM0 Clock	E	PE_04
ACM0_FS	ACM0 Frame Sync	E	PE_03
ACM0_T0	ACM0 External Trigger 0	E	PE_08
ACM0_T1	ACM0 External Trigger 1	G	PG_05
CAN0_RX	CAN0 Receive	G	PG_04
CAN0_TX	CAN0 Transmit	G	PG_01
CNT0_DG	CNT0 Count Down and Gate	G	PG_12
CNT0_UD	CNT0 Count Up and Direction	G	PG_11
CNT0_ZM	CNT0 Count Zero Marker	G	PG_07
DMC0_A00	DMC Address 0	Not Muxed	DMC0_A00
DMC0_A01	DMC Address 1	Not Muxed	DMC0_A01
DMC0_A02	DMC Address 2	Not Muxed	DMC0_A02
DMC0_A03	DMC Address 3	Not Muxed	DMC0_A03
DMC0_A04	DMC Address 4	Not Muxed	DMC0_A04
DMC0_A05	DMC Address 5	Not Muxed	DMC0_A05
DMC0_A06	DMC Address 6	Not Muxed	DMC0_A06
DMC0_A07	DMC Address 7	Not Muxed	DMC0_A07
DMC0_A08	DMC Address 8	Not Muxed	DMC0_A08
DMC0_A09	DMC Address 9	Not Muxed	DMC0_A09
DMC0_A10	DMC Address 10	Not Muxed	DMC0_A10
DMC0_A11	DMC Address 11	Not Muxed	DMC0_A11
DMC0_A12	DMC Address 12	Not Muxed	DMC0_A12
DMC0_A13	DMC Address 13	Not Muxed	DMC0_A13
DMC0_BA0	DMC Bank Address Input 0	Not Muxed	DMC0_BA0
DMC0_BA1	DMC Bank Address Input 1	Not Muxed	DMC0_BA1
DMC0_BA2	DMC Bank Address Input 2	Not Muxed	DMC0_BA2
$\overline{\text{DMC0\_CAS}}$	DMC Column Address Strobe	Not Muxed	$\overline{\text{DMC0\_CAS}}$
DMC0_CK	DMC Clock	Not Muxed	DMC0_CK
DMC0_CKE	DMC Clock Enable	Not Muxed	DMC0_CKE
$\overline{\text{DMC0\_CK}}$	DMC Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
$\overline{\text{DMC0\_CS0}}$	DMC Chip Select 0	Not Muxed	$\overline{\text{DMC0\_CS0}}$

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC0_DQ00	DMC Data 0	Not Muxed	DMC0_DQ00
DMC0_DQ01	DMC Data 1	Not Muxed	DMC0_DQ01
DMC0_DQ02	DMC Data 2	Not Muxed	DMC0_DQ02
DMC0_DQ03	DMC Data 3	Not Muxed	DMC0_DQ03
DMC0_DQ04	DMC Data 4	Not Muxed	DMC0_DQ04
DMC0_DQ05	DMC Data 5	Not Muxed	DMC0_DQ05
DMC0_DQ06	DMC Data 6	Not Muxed	DMC0_DQ06
DMC0_DQ07	DMC Data 7	Not Muxed	DMC0_DQ07
DMC0_DQ08	DMC Data 8	Not Muxed	DMC0_DQ08
DMC0_DQ09	DMC Data 9	Not Muxed	DMC0_DQ09
DMC0_DQ10	DMC Data 10	Not Muxed	DMC0_DQ10
DMC0_DQ11	DMC Data 11	Not Muxed	DMC0_DQ11
DMC0_DQ12	DMC Data 12	Not Muxed	DMC0_DQ12
DMC0_DQ13	DMC Data 13	Not Muxed	DMC0_DQ13
DMC0_DQ14	DMC Data 14	Not Muxed	DMC0_DQ14
DMC0_DQ15	DMC Data 15	Not Muxed	DMC0_DQ15
DMC0_LDM	DMC Data Mask for Lower Byte	Not Muxed	DMC0_LDM
DMC0_LDQS	DMC Data Strobe for Lower Byte	Not Muxed	DMC0_LDQS
$\overline{\text{DMC0\_LDQS}}$	DMC Data Strobe for Lower Byte (complement)	Not Muxed	$\overline{\text{DMC0\_LDQS}}$
DMC0_ODT	DMC On-die Termination	Not Muxed	DMC0_ODT
$\overline{\text{DMC0\_RAS}}$	DMC Row Address Strobe	Not Muxed	$\overline{\text{DMC0\_RAS}}$
DMC0_UDM	DMC Data Mask for Upper Byte	Not Muxed	DMC0_UDM
DMC0_UDQS	DMC Data Strobe for Upper Byte	Not Muxed	DMC0_UDQS
$\overline{\text{DMC0\_UDQS}}$	DMC Data Strobe for Upper Byte (complement)	Not Muxed	$\overline{\text{DMC0\_UDQS}}$
$\overline{\text{DMC0\_WE}}$	DMC Write Enable	Not Muxed	$\overline{\text{DMC0\_WE}}$
ETH0_CRS	EMAC0 Carrier Sense/RMII Receive Data Valid	C	PC_05
ETH0_MDC	EMAC0 Management Channel Clock	C	PC_06
ETH0_MDIO	EMAC0 Management Channel Serial Data	C	PC_07
ETH0_PTPPPS	EMAC0 PTP Pulse-Per-Second Output	B	PB_15
ETH0_REFCLK	EMAC0 Reference Clock	B	PB_14
ETH0_RXD0	EMAC0 Receive Data 0	C	PC_00
ETH0_RXD1	EMAC0 Receive Data 1	C	PC_01
ETH0_TXD0	EMAC0 Transmit Data 0	C	PC_02
ETH0_TXD1	EMAC0 Transmit Data 1	C	PC_03
ETH0_TXEN	EMAC0 Transmit Enable	B	PB_13
ETH1_CRS	EMAC1 Carrier Sense/RMII Receive Data Valid	E	PE_13
ETH1_MDC	EMAC1 Management Channel Clock	E	PE_10
ETH1_MDIO	EMAC1 Management Channel Serial Data	E	PE_11
ETH1_PTPPPS	EMAC1 PTP Pulse-Per-Second Output	C	PC_09
ETH1_REFCLK	EMAC1 Reference Clock	G	PG_06
ETH1_RXD0	EMAC1 Receive Data 0	G	PG_00
ETH1_RXD1	EMAC1 Receive Data 1	E	PE_15
ETH1_TXD0	EMAC1 Transmit Data 0	G	PG_03
ETH1_TXD1	EMAC1 Transmit Data 1	G	PG_02
ETH1_TXEN	EMAC1 Transmit Enable	G	PG_05
ETH_PTPAUXIN	EMAC0/EMAC1 PTP Auxiliary Trigger Input	C	PC_11

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
ETH_PTPCLKIN	EMAC0/EMAC1 PTP Clock Input	C	PC_13
GND	Ground	Not Muxed	GND
$\overline{\text{JTG\_EMU}}$	Emulation Output	Not Muxed	$\overline{\text{JTG\_EMU}}$
JTG_TCK	JTAG Clock	Not Muxed	JTG_TCK
JTG_TDI	JTAG Serial Data Input	Not Muxed	JTG_TDI
JTG_TDO	JTAG Serial Data Output	Not Muxed	JTG_TDO
JTG_TMS	JTAG Mode Select	Not Muxed	JTG_TMS
$\overline{\text{JTG\_TRST}}$	JTAG Reset	Not Muxed	$\overline{\text{JTG\_TRST}}$
LP0_ACK	LP0 Acknowledge	B	PB_01
LP0_CLK	LP0 Clock	B	PB_00
LP0_D0	LP0 Data 0	A	PA_00
LP0_D1	LP0 Data 1	A	PA_01
LP0_D2	LP0 Data 2	A	PA_02
LP0_D3	LP0 Data 3	A	PA_03
LP0_D4	LP0 Data 4	A	PA_04
LP0_D5	LP0 Data 5	A	PA_05
LP0_D6	LP0 Data 6	A	PA_06
LP0_D7	LP0 Data 7	A	PA_07
LP1_ACK	LP1 Acknowledge	B	PB_02
LP1_CLK	LP1 Clock	B	PB_03
LP1_D0	LP1 Data 0	A	PA_08
LP1_D1	LP1 Data 1	A	PA_09
LP1_D2	LP1 Data 2	A	PA_10
LP1_D3	LP1 Data 3	A	PA_11
LP1_D4	LP1 Data 4	A	PA_12
LP1_D5	LP1 Data 5	A	PA_13
LP1_D6	LP1 Data 6	A	PA_14
LP1_D7	LP1 Data 7	A	PA_15
LP2_ACK	LP2 Acknowledge	E	PE_08
LP2_CLK	LP2 Clock	E	PE_09
LP2_D0	LP2 Data 0	F	PF_00
LP2_D1	LP2 Data 1	F	PF_01
LP2_D2	LP2 Data 2	F	PF_02
LP2_D3	LP2 Data 3	F	PF_03
LP2_D4	LP2 Data 4	F	PF_04
LP2_D5	LP2 Data 5	F	PF_05
LP2_D6	LP2 Data 6	F	PF_06
LP2_D7	LP2 Data 7	F	PF_07
LP3_ACK	LP3 Acknowledge	E	PE_07
LP3_CLK	LP3 Clock	E	PE_06
LP3_D0	LP3 Data 0	F	PF_08
LP3_D1	LP3 Data 1	F	PF_09
LP3_D2	LP3 Data 2	F	PF_10
LP3_D3	LP3 Data 3	F	PF_11
LP3_D4	LP3 Data 4	F	PF_12
LP3_D5	LP3 Data 5	F	PF_13

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
LP3_D6	LP3 Data 6	F	PF_14
LP3_D7	LP3 Data 7	F	PF_15
PA_00 – PA_15	PORTA Position 00 through PORTA Position 15	A	PA_00 – PA_15
PB_00 – PB_15	PORTB Position 00 through PORTB Position 15	B	PB_00 – PB_15
PC_00 – PC_15	PORTC Position 00 through PORTC Position 15	C	PC_00 – PC_15
PD_00 – PD_15	PORTD Position 00 through PORTD Position 15	D	PD_00 – PD_15
PE_00 – PE_15	PORTE Position 00 through PORTE Position 15	E	PE_00 – PE_15
PF_00 – PF_15	PORTF Position 00 through PORTF Position 15	F	PF_00 – PF_15
PG_00 – PG_15	PORTG Position 00 through PORTG Position 15	G	PG_00 – PG_15
PPI0_CLK	EPPI0 Clock	E	PE_09
PPI0_D00	EPPI0 Data 0	F	PF_00
PPI0_D01	EPPI0 Data 1	F	PF_01
PPI0_D02	EPPI0 Data 2	F	PF_02
PPI0_D03	EPPI0 Data 3	F	PF_03
PPI0_D04	EPPI0 Data 4	F	PF_04
PPI0_D05	EPPI0 Data 5	F	PF_05
PPI0_D06	EPPI0 Data 6	F	PF_06
PPI0_D07	EPPI0 Data 7	F	PF_07
PPI0_D08	EPPI0 Data 8	F	PF_08
PPI0_D09	EPPI0 Data 9	F	PF_09
PPI0_D10	EPPI0 Data 10	F	PF_10
PPI0_D11	EPPI0 Data 11	F	PF_11
PPI0_D12	EPPI0 Data 12	F	PF_12
PPI0_D13	EPPI0 Data 13	F	PF_13
PPI0_D14	EPPI0 Data 14	F	PF_14
PPI0_D15	EPPI0 Data 15	F	PF_15
PPI0_D16	EPPI0 Data 16	E	PE_03
PPI0_D17	EPPI0 Data 17	E	PE_04
PPI0_D18	EPPI0 Data 18	E	PE_00
PPI0_D19	EPPI0 Data 19	E	PE_01
PPI0_D20	EPPI0 Data 20	D	PD_12
PPI0_D21	EPPI0 Data 21	D	PD_15
PPI0_D22	EPPI0 Data 22	E	PE_02
PPI0_D23	EPPI0 Data 23	E	PE_05
PPI0_FS1	EPPI0 Frame Sync 1 (HSYNC)	E	PE_08
PPI0_FS2	EPPI0 Frame Sync 2 (VSYNC)	E	PE_07
PPI0_FS3	EPPI0 Frame Sync 3 (FIELD)	E	PE_06
PPI1_CLK	EPPI1 Clock	B	PB_14
PPI1_D00	EPPI1 Data 0	C	PC_00
PPI1_D01	EPPI1 Data 1	C	PC_01
PPI1_D02	EPPI1 Data 2	C	PC_02
PPI1_D03	EPPI1 Data 3	C	PC_03
PPI1_D04	EPPI1 Data 4	C	PC_04
PPI1_D05	EPPI1 Data 5	C	PC_05
PPI1_D06	EPPI1 Data 6	C	PC_06
PPI1_D07	EPPI1 Data 7	C	PC_07

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
PPI1_D08	EPPI1 Data 8	C	PC_08
PPI1_D09	EPPI1 Data 9	C	PC_09
PPI1_D10	EPPI1 Data 10	C	PC_10
PPI1_D11	EPPI1 Data 11	C	PC_11
PPI1_D12	EPPI1 Data 12	C	PC_12
PPI1_D13	EPPI1 Data 13	C	PC_13
PPI1_D14	EPPI1 Data 14	C	PC_14
PPI1_D15	EPPI1 Data 15	C	PC_15
PPI1_D16	EPPI1 Data 16	D	PD_00
PPI1_D17	EPPI1 Data 17	D	PD_01
PPI1_FS1	EPPI1 Frame Sync 1 (HSYNC)	B	PB_13
PPI1_FS2	EPPI1 Frame Sync 2 (VSYNC)	D	PD_06
PPI1_FS3	EPPI1 Frame Sync 3 (FIELD)	B	PB_15
PPI2_CLK	EPPI2 Clock	B	PB_00
PPI2_D00	EPPI2 Data 0	A	PA_00
PPI2_D01	EPPI2 Data 1	A	PA_01
PPI2_D02	EPPI2 Data 2	A	PA_02
PPI2_D03	EPPI2 Data 3	A	PA_03
PPI2_D04	EPPI2 Data 4	A	PA_04
PPI2_D05	EPPI2 Data 5	A	PA_05
PPI2_D06	EPPI2 Data 6	A	PA_06
PPI2_D07	EPPI2 Data 7	A	PA_07
PPI2_D08	EPPI2 Data 8	A	PA_08
PPI2_D09	EPPI2 Data 9	A	PA_09
PPI2_D10	EPPI2 Data 10	A	PA_10
PPI2_D11	EPPI2 Data 11	A	PA_11
PPI2_D12	EPPI2 Data 12	A	PA_12
PPI2_D13	EPPI2 Data 13	A	PA_13
PPI2_D14	EPPI2 Data 14	A	PA_14
PPI2_D15	EPPI2 Data 15	A	PA_15
PPI2_D16	EPPI2 Data 16	B	PB_07
PPI2_D17	EPPI2 Data 17	B	PB_08
PPI2_FS1	EPPI2 Frame Sync 1 (HSYNC)	B	PB_01
PPI2_FS2	EPPI2 Frame Sync 2 (VSYNC)	B	PB_02
PPI2_FS3	EPPI2 Frame Sync 3 (FIELD)	B	PB_03
PWM0_AH	PWM0 Channel A High Side	F	PF_01
PWM0_AL	PWM0 Channel A Low Side	F	PF_00
PWM0_BH	PWM0 Channel B High Side	F	PF_03
PWM0_BL	PWM0 Channel B Low Side	F	PF_02
PWM0_CH	PWM0 Channel C High Side	F	PF_05
PWM0_CL	PWM0 Channel C Low Side	F	PF_04
PWM0_DH	PWM0 Channel D High Side	F	PF_07
PWM0_DL	PWM0 Channel D Low Side	F	PF_06
PWM0_SYNC	PWM0 Sync	E	PE_08
$\overline{\text{PWM0\_TRIP0}}$	PWM0 Shutdown Input 0	E	PE_09
$\overline{\text{PWM0\_TRIP1}}$	PWM0 Shutdown Input 1	F	PF_11

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
PWM1_AH	PWM1 Channel A High Side	G	PG_03
PWM1_AL	PWM1 Channel A Low Side	G	PG_02
PWM1_BH	PWM1 Channel B High Side	G	PG_00
PWM1_BL	PWM1 Channel B Low Side	E	PE_15
PWM1_CH	PWM1 Channel C High Side	E	PE_13
PWM1_CL	PWM1 Channel C Low Side	E	PE_12
PWM1_DH	PWM1 Channel D High Side	E	PE_11
PWM1_DL	PWM1 Channel D Low Side	E	PE_10
PWM1_SYNC	PWM1 Sync	G	PG_05
$\overline{\text{PWM1\_TRIP0}}$	PWM1 Shutdown Input 0	G	PG_06
$\overline{\text{PWM1\_TRIP1}}$	PWM1 Shutdown Input 1	G	PG_08
RSI0_CLK	RSI0 Clock	G	PG_06
RSI0_CMD	RSI0 Command	G	PG_05
RSI0_D0	RSI0 Data 0	G	PG_03
RSI0_D1	RSI0 Data 1	G	PG_02
RSI0_D2	RSI0 Data 2	G	PG_00
RSI0_D3	RSI0 Data 3	E	PE_15
RSI0_D4	RSI0 Data 4	E	PE_13
RSI0_D5	RSI0 Data 5	E	PE_12
RSI0_D6	RSI0 Data 6	E	PE_10
RSI0_D7	RSI0 Data 7	E	PE_11
SMC0_A01	SMC0 Address 1	Not Muxed	SMC0_A01
SMC0_A02	SMC0 Address 2	Not Muxed	SMC0_A02
SMC0_A03	SMC0 Address 3	A	PA_00
SMC0_A04	SMC0 Address 4	A	PA_01
SMC0_A05	SMC0 Address 5	A	PA_02
SMC0_A06	SMC0 Address 6	A	PA_03
SMC0_A07	SMC0 Address 7	A	PA_04
SMC0_A08	SMC0 Address 8	A	PA_05
SMC0_A09	SMC0 Address 9	A	PA_06
SMC0_A10	SMC0 Address 10	A	PA_07
SMC0_A11	SMC0 Address 11	A	PA_08
SMC0_A12	SMC0 Address 12	A	PA_09
SMC0_A13	SMC0 Address 13	B	PB_02
SMC0_A14	SMC0 Address 14	A	PA_10
SMC0_A15	SMC0 Address 15	A	PA_11
SMC0_A16	SMC0 Address 16	B	PB_03
SMC0_A17	SMC0 Address 17	A	PA_12
SMC0_A18	SMC0 Address 18	A	PA_13
SMC0_A19	SMC0 Address 19	A	PA_14
SMC0_A20	SMC0 Address 20	A	PA_15
SMC0_A21	SMC0 Address 21	B	PB_06
SMC0_A22	SMC0 Address 22	B	PB_07
SMC0_A23	SMC0 Address 23	B	PB_08
SMC0_A24	SMC0 Address 24	B	PB_10
SMC0_A25	SMC0 Address 25	B	PB_11

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SMC0_ABE0	SMC0 Byte Enable 0	B	PB_04
SMC0_ABE1	SMC0 Byte Enable 1	B	PB_05
SMC0_AMS0	SMC0 Memory Select 0	Not Muxed	SMC0_AMS0
SMC0_AMS1	SMC0 Memory Select 1	B	PB_01
SMC0_AMS2	SMC0 Memory Select 2	B	PB_04
SMC0_AMS3	SMC0 Memory Select 3	B	PB_05
SMC0_AOE	SMC0 Output Enable	Not Muxed	SMC0_AOE_NORDV
SMC0_ARDY	SMC0 Asynchronous Ready	Not Muxed	SMC0_ARDY_NORWT
SMC0_ARE	SMC0 Read Enable	Not Muxed	SMC0_ARE
SMC0_AWE	SMC0 Write Enable	Not Muxed	SMC0_AWE
SMC0_BGH	SMC0 Bus Grant Hang	B	PB_09
SMC0_BG	SMC0 Bus Grant	B	PB_12
SMC0_BR	SMC0 Bus Request	Not Muxed	SMC0_BR
SMC0_D00	SMC0 Data 0	Not Muxed	SMC0_D00
SMC0_D01	SMC0 Data 1	Not Muxed	SMC0_D01
SMC0_D02	SMC0 Data 2	Not Muxed	SMC0_D02
SMC0_D03	SMC0 Data 3	Not Muxed	SMC0_D03
SMC0_D04	SMC0 Data 4	Not Muxed	SMC0_D04
SMC0_D05	SMC0 Data 5	Not Muxed	SMC0_D05
SMC0_D06	SMC0 Data 6	Not Muxed	SMC0_D06
SMC0_D07	SMC0 Data 7	Not Muxed	SMC0_D07
SMC0_D08	SMC0 Data 8	Not Muxed	SMC0_D08
SMC0_D09	SMC0 Data 9	Not Muxed	SMC0_D09
SMC0_D10	SMC0 Data 10	Not Muxed	SMC0_D10
SMC0_D11	SMC0 Data 11	Not Muxed	SMC0_D11
SMC0_D12	SMC0 Data 12	Not Muxed	SMC0_D12
SMC0_D13	SMC0 Data 13	Not Muxed	SMC0_D13
SMC0_D14	SMC0 Data 14	Not Muxed	SMC0_D14
SMC0_D15	SMC0 Data 15	Not Muxed	SMC0_D15
SMC0_NORCLK	SMC0 NOR Clock	B	PB_00
SMC0_NORDV	SMC0 NOR Data Valid	Not Muxed	SMC0_AOE_NORDV
SMC0_NORWT	SMC0 NOR Wait	Not Muxed	SMC0_ARDY_NORWT
SPI0_CLK	SPI0 Clock	D	PD_04
SPI0_D2	SPI0 Data 2	D	PD_00
SPI0_D3	SPI0 Data 3	D	PD_01
SPI0_MISO	SPI0 Master In, Slave Out	D	PD_02
SPI0_MOSI	SPI0 Master Out, Slave In	D	PD_03
SPI0_RDY	SPI0 Ready	D	PD_10
SPI0_SEL1	SPI0 Slave Select Output 1	D	PD_11
SPI0_SEL2	SPI0 Slave Select Output 2	D	PD_01
SPI0_SEL3	SPI0 Slave Select Output 3	D	PD_00
SPI0_SEL4	SPI0 Slave Select Output 4	C	PC_15
SPI0_SEL5	SPI0 Slave Select Output 5	D	PD_09
SPI0_SEL6	SPI0 Slave Select Output 6	C	PC_13
SPI0_SEL7	SPI0 Slave Select Output 7	C	PC_12
SPI0_SS	SPI0 Slave Select Input	D	PD_11

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SPI1_CLK	SPI1 Clock	D	PD_05
SPI1_D2	SPI1 Data 2	E	PE_01
SPI1_D3	SPI1 Data 3	E	PE_00
SPI1_MISO	SPI1 Master In, Slave Out	D	PD_14
SPI1_MOSI	SPI1 Master Out, Slave In	D	PD_13
SPI1_RDY	SPI1 Ready	E	PE_02
$\overline{\text{SPI1\_SEL1}}$	SPI1 Slave Select Output 1	D	PD_12
$\overline{\text{SPI1\_SEL2}}$	SPI1 Slave Select Output 2	D	PD_15
$\overline{\text{SPI1\_SEL3}}$	SPI1 Slave Select Output 3	D	PD_10
$\overline{\text{SPI1\_SEL4}}$	SPI1 Slave Select Output 4	D	PD_09
$\overline{\text{SPI1\_SEL5}}$	SPI1 Slave Select Output 5	F	PF_08
$\overline{\text{SPI1\_SEL6}}$	SPI1 Slave Select Output 6	F	PF_09
$\overline{\text{SPI1\_SEL7}}$	SPI1 Slave Select Output 7	C	PC_14
$\overline{\text{SPI1\_SS}}$	SPI1 Slave Select Input	D	PD_12
SPT0_ACLK	SPORT0 Channel A Clock	B	PB_05
SPT0_AD0	SPORT0 Channel A Data 0	B	PB_09
SPT0_AD1	SPORT0 Channel A Data 1	B	PB_12
SPT0_AFS	SPORT0 Channel A Frame Sync	B	PB_04
SPT0_ATDV	SPORT0 Channel A Transmit Data Valid	B	PB_06
SPT0_BCLK	SPORT0 Channel B Clock	B	PB_08
SPT0_BD0	SPORT0 Channel B Data 0	B	PB_11
SPT0_BD1	SPORT0 Channel B Data 1	B	PB_10
SPT0_BFS	SPORT0 Channel B Frame Sync	B	PB_07
SPT0_BTDV	SPORT0 Channel B Transmit Data Valid	B	PB_12
SPT1_ACLK	SPORT1 Channel A Clock	E	PE_02
SPT1_AD0	SPORT1 Channel A Data 0	D	PD_15
SPT1_AD1	SPORT1 Channel A Data 1	D	PD_12
SPT1_AFS	SPORT1 Channel A Frame Sync	E	PE_05
SPT1_ATDV	SPORT1 Channel A Transmit Data Valid	E	PE_06
SPT1_BCLK	SPORT1 Channel B Clock	E	PE_04
SPT1_BD0	SPORT1 Channel B Data 0	E	PE_01
SPT1_BD1	SPORT1 Channel B Data 1	E	PE_00
SPT1_BFS	SPORT1 Channel B Frame Sync	E	PE_03
SPT1_BTDV	SPORT1 Channel B Transmit Data Valid	E	PE_07
SPT2_ACLK	SPORT2 Channel A Clock	G	PG_04
SPT2_AD0	SPORT2 Channel A Data 0	G	PG_09
SPT2_AD1	SPORT2 Channel A Data 1	G	PG_08
SPT2_AFS	SPORT2 Channel A Frame Sync	G	PG_01
SPT2_ATDV	SPORT2 Channel A Transmit Data Valid	E	PE_14
SPT2_BCLK	SPORT2 Channel B Clock	G	PG_10
SPT2_BD0	SPORT2 Channel B Data 0	G	PG_12
SPT2_BD1	SPORT2 Channel B Data 1	G	PG_11
SPT2_BFS	SPORT2 Channel B Frame Sync	G	PG_07
SPT2_BTDV	SPORT2 Channel B Transmit Data Valid	G	PG_06
SYS_BMODE0	Boot Mode Control 0	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control 1	Not Muxed	SYS_BMODE1

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SYS_BMODE2	Boot Mode Control 2	Not Muxed	SYS_BMODE2
SYS_CLKIN	Clock/Crystal Input	Not Muxed	SYS_CLKIN
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_EXTWAKE	External Wake Control	Not Muxed	SYS_EXTWAKE
SYS_FAULT	Fault Output	Not Muxed	SYS_FAULT
$\overline{\text{SYS\_FAULT}}$	Complementary Fault Output	Not Muxed	$\overline{\text{SYS\_FAULT}}$
$\overline{\text{SYS\_HWRST}}$	Processor Hardware Reset Control	Not Muxed	$\overline{\text{SYS\_HWRST}}$
$\overline{\text{SYS\_IDLE0}}$	Core 0 Idle Indicator	G	PG_15
$\overline{\text{SYS\_IDLE1}}$	Core 1 Idle Indicator	G	PG_14
$\overline{\text{SYS\_NMI}}$	Non-maskable Interrupt	Not Muxed	$\overline{\text{SYS\_NMI\_RESOUT}}$
SYS_PWRGD	Power Good Indicator	Not Muxed	SYS_PWRGD
$\overline{\text{SYS\_RESOUT}}$	Reset Output	Not Muxed	$\overline{\text{SYS\_NMI\_RESOUT}}$
$\overline{\text{SYS\_SLEEP}}$	Processor Sleep Indicator	G	PG_15
SYS_TDA	Thermal Diode Anode	Not Muxed	SYS_TDA
SYS_TDK	Thermal Diode Cathode	Not Muxed	SYS_TDK
SYS_XTAL	Crystal Output	Not Muxed	SYS_XTAL
TM0_ACI0	TIMER0 Alternate Capture Input 0	D	PD_08
TM0_ACI1	TIMER0 Alternate Capture Input 1	G	PG_14
TM0_ACI2	TIMER0 Alternate Capture Input 2	G	PG_04
TM0_ACI3	TIMER0 Alternate Capture Input 3	D	PD_07
TM0_ACI4	TIMER0 Alternate Capture Input 4	G	PG_15
TM0_ACI5	TIMER0 Alternate Capture Input 5	D	PD_06
TM0_ACI6	TIMER0 Alternate Capture Input 6	B	PB_13
TM0_ACLK0	TIMER0 Alternate Clock 0	B	PB_10
TM0_ACLK1	TIMER0 Alternate Clock 1	B	PB_12
TM0_ACLK2	TIMER0 Alternate Clock 2	B	PB_09
TM0_ACLK3	TIMER0 Alternate Clock 3	B	PB_11
TM0_ACLK4	TIMER0 Alternate Clock 4	B	PB_06
TM0_ACLK5	TIMER0 Alternate Clock 5	D	PD_13
TM0_ACLK6	TIMER0 Alternate Clock 6	D	PD_14
TM0_ACLK7	TIMER0 Alternate Clock 7	D	PD_05
TM0_CLK	TIMER0 Clock	G	PG_13
TM0_TMR0	TIMER0 Timer 0	E	PE_14
TM0_TMR1	TIMER0 Timer 1	G	PG_04
TM0_TMR2	TIMER0 Timer 2	G	PG_01
TM0_TMR3	TIMER0 Timer 3	G	PG_08
TM0_TMR4	TIMER0 Timer 4	G	PG_09
TM0_TMR5	TIMER0 Timer 5	G	PG_07
TM0_TMR6	TIMER0 Timer 6	G	PG_11
TM0_TMR7	TIMER0 Timer 7	G	PG_12
TWI0_SCL	TWI0 Serial Clock	Not Muxed	TWI0_SCL
TWI0_SDA	TWI0 Serial Data	Not Muxed	TWI0_SDA
TWI1_SCL	TWI1 Serial Clock	Not Muxed	TWI1_SCL
TWI1_SDA	TWI1 Serial Data	Not Muxed	TWI1_SDA
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	D	PD_10
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	D	PD_09

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
UART0_RX	UART0 Receive	D	PD_08
UART0_TX	UART0 Transmit	D	PD_07
UART1_CTS	UART1 Clear to Send	G	PG_13
UART1_RTS	UART1 Request to Send	G	PG_10
UART1_RX	UART1 Receive	G	PG_14
UART1_TX	UART1 Transmit	G	PG_15
USB0_CLKIN	USB0 Clock/Crystal Input	Not Muxed	USB0_CLKIN
USB0_DM	USB0 Data –	Not Muxed	USB0_DM
USB0_DP	USB0 Data +	Not Muxed	USB0_DP
USB0_ID	USB0 OTG ID	Not Muxed	USB0_ID
USB0_VBC	USB0 VBUS Control	Not Muxed	USB0_VBC
USB0_VBUS	USB0 Bus Voltage	Not Muxed	USB0_VBUS
VDD_DMC	VDD for DMC	Not Muxed	VDD_DMC
VDD_EXT	External VDD	Not Muxed	VDD_EXT
VDD_INT	Internal VDD	Not Muxed	VDD_INT
VDD_TD	VDD for Thermal Diode	Not Muxed	VDD_TD
VDD_USB	VDD for USB	Not Muxed	VDD_USB
VREF_DMC	VREF for DMC	Not Muxed	VREF_DMC

GP I/O MULTIPLEXING FOR 349-BALL CSP_BGA

Table 8 through Table 14 identifies the pin functions that are multiplexed on the general-purpose I/O pins of the 349-ball CSP_BGA package.

Table 8. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PA_00	SMC0_A03	PPI2_D00	LP0_D0	
PA_01	SMC0_A04	PPI2_D01	LP0_D1	
PA_02	SMC0_A05	PPI2_D02	LP0_D2	
PA_03	SMC0_A06	PPI2_D03	LP0_D3	
PA_04	SMC0_A07	PPI2_D04	LP0_D4	
PA_05	SMC0_A08	PPI2_D05	LP0_D5	
PA_06	SMC0_A09	PPI2_D06	LP0_D6	
PA_07	SMC0_A10	PPI2_D07	LP0_D7	
PA_08	SMC0_A11	PPI2_D08	LP1_D0	
PA_09	SMC0_A12	PPI2_D09	LP1_D1	
PA_10	SMC0_A14	PPI2_D10	LP1_D2	
PA_11	SMC0_A15	PPI2_D11	LP1_D3	
PA_12	SMC0_A17	PPI2_D12	LP1_D4	
PA_13	SMC0_A18	PPI2_D13	LP1_D5	
PA_14	SMC0_A19	PPI2_D14	LP1_D6	
PA_15	SMC0_A20	PPI2_D15	LP1_D7	

Table 9. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PB_00	SMC0_NORCLK	PPI2_CLK	LP0_CLK	
PB_01	$\overline{\text{SMC0\_AMS1}}$	PPI2_FS1	LP0_ACK	
PB_02	SMC0_A13	PPI2_FS2	LP1_ACK	
PB_03	SMC0_A16	PPI2_FS3	LP1_CLK	
PB_04	$\overline{\text{SMC0\_AMS2}}$	$\overline{\text{SMC0\_ABE0}}$	SPT0_AFS	
PB_05	$\overline{\text{SMC0\_AMS3}}$	$\overline{\text{SMC0\_ABE1}}$	SPT0_ACLK	
PB_06	SMC0_A21	SPT0_ATDV		TM0_ACLK4
PB_07	SMC0_A22	PPI2_D16	SPT0_BFS	
PB_08	SMC0_A23	PPI2_D17	SPT0_BCLK	
PB_09	$\overline{\text{SMC0\_BGH}}$		SPT0_AD0	TM0_ACLK2
PB_10	SMC0_A24		SPT0_BD1	TM0_ACLK0
PB_11	SMC0_A25		SPT0_BD0	TM0_ACLK3
PB_12	$\overline{\text{SMC0\_BG}}$	SPT0_BTDV	SPT0_AD1	TM0_ACLK1
PB_13	ETH0_TXEN	PPI1_FS1		TM0_ACI6
PB_14	ETH0_REFCLK	PPI1_CLK		
PB_15	ETH0_PTPPPS	PPI1_FS3		

Table 10. Signal Multiplexing for Port C

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PC_00	ETH0_RXD0	PPI1_D00		
PC_01	ETH0_RXD1	PPI1_D01		
PC_02	ETH0_TXD0	PPI1_D02		
PC_03	ETH0_TXD1	PPI1_D03		
PC_04		PPI1_D04		
PC_05	ETH0_CRS	PPI1_D05		
PC_06	ETH0_MDC	PPI1_D06		
PC_07	ETH0_MDIO	PPI1_D07		
PC_08		PPI1_D08		
PC_09	ETH1_PTPPPS	PPI1_D09		
PC_10		PPI1_D10		
PC_11		PPI1_D11	ETH_PTPAUXIN	
PC_12	$\overline{\text{SPI0\_SEL7}}$	PPI1_D12		
PC_13	$\overline{\text{SPI0\_SEL6}}$	PPI1_D13	ETH_PTPCLKIN	
PC_14	$\overline{\text{SPI1\_SEL7}}$	PPI1_D14		
PC_15	$\overline{\text{SPI0\_SEL4}}$	PPI1_D15		

Table 11. Signal Multiplexing for Port D

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PD_00	SPI0_D2	PPI1_D16	$\overline{\text{SPI0\_SEL3}}$	
PD_01	SPI0_D3	PPI1_D17	$\overline{\text{SPI0\_SEL2}}$	
PD_02	SPI0_MISO			
PD_03	SPI0_MOSI			
PD_04	SPI0_CLK			
PD_05	SPI1_CLK			TM0_ACLK7
PD_06		PPI1_FS2		TM0_ACI5
PD_07		$\overline{\text{UART0\_TX}}$		TM0_ACI3
PD_08		$\overline{\text{UART0\_RX}}$		TM0_ACI0
PD_09	$\overline{\text{SPI0\_SEL5}}$	$\overline{\text{UART0\_RTS}}$	$\overline{\text{SPI1\_SEL4}}$	
PD_10	SPI0_RDY	$\overline{\text{UART0\_CTS}}$	$\overline{\text{SPI1\_SEL3}}$	
PD_11	$\overline{\text{SPI0\_SEL1}}$			$\overline{\text{SPI0\_SS}}$
PD_12	$\overline{\text{SPI1\_SEL1}}$	PPI0_D20	SPT1_AD1	$\overline{\text{SPI1\_SS}}$
PD_13	SPI1_MOSI			TM0_ACLK5
PD_14	SPI1_MISO			TM0_ACLK6
PD_15	$\overline{\text{SPI1\_SEL2}}$	PPI0_D21	SPT1_AD0	

Table 12. Signal Multiplexing for Port E

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PE_00	SPI1_D3	PPI0_D18	SPT1_BD1	
PE_01	SPI1_D2	PPI0_D19	SPT1_BD0	
PE_02	SPI1_RDY	PPI0_D22	SPT1_ACLK	
PE_03		PPI0_D16	ACM0_FS/SPT1_BFS	
PE_04		PPI0_D17	ACM0_CLK/SPT1_BCLK	
PE_05		PPI0_D23	SPT1_AFS	
PE_06	SPT1_ATDV	PPI0_FS3	LP3_CLK	
PE_07	SPT1_BTDV	PPI0_FS2	LP3_ACK	
PE_08	PWM0_SYNC	PPI0_FS1	LP2_ACK	ACM0_T0
PE_09		PPI0_CLK	LP2_CLK	$\overline{\text{PWM0\_TRIP0}}$
PE_10	ETH1_MDC	PWM1_DL	RSI0_D6	
PE_11	ETH1_MDIO	PWM1_DH	RSI0_D7	
PE_12		PWM1_CL	RSI0_D5	
PE_13	ETH1_CRS	PWM1_CH	RSI0_D4	
PE_14		SPT2_ATDV	TM0_TMR0	
PE_15	ETH1_RXD1	PWM1_BL	RSI0_D3	

Table 13. Signal Multiplexing for Port F

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PF_00	PWM0_AL	PPI0_D00	LP2_D0	
PF_01	PWM0_AH	PPI0_D01	LP2_D1	
PF_02	PWM0_BL	PPI0_D02	LP2_D2	
PF_03	PWM0_BH	PPI0_D03	LP2_D3	
PF_04	PWM0_CL	PPI0_D04	LP2_D4	
PF_05	PWM0_CH	PPI0_D05	LP2_D5	
PF_06	PWM0_DL	PPI0_D06	LP2_D6	
PF_07	PWM0_DH	PPI0_D07	LP2_D7	
PF_08	$\overline{\text{SPI1\_SEL5}}$	PPI0_D08	LP3_D0	
PF_09	$\overline{\text{SPI1\_SEL6}}$	PPI0_D09	LP3_D1	
PF_10	ACM0_A4	PPI0_D10	LP3_D2	
PF_11		PPI0_D11	LP3_D3	$\overline{\text{PWM0\_TRIP1}}$
PF_12	ACM0_A2	PPI0_D12	LP3_D4	
PF_13	ACM0_A3	PPI0_D13	LP3_D5	
PF_14	ACM0_A0	PPI0_D14	LP3_D6	
PF_15	ACM0_A1	PPI0_D15	LP3_D7	

Table 14. Signal Multiplexing for Port G

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PG_00	ETH1_RXD0	PWM1_BH	RSI0_D2	
PG_01	SPT2_AFS	TM0_TMR2	CAN0_TX	
PG_02	ETH1_TXD1	PWM1_AL	RSI0_D1	
PG_03	ETH1_TXD0	PWM1_AH	RSI0_D0	
PG_04	SPT2_ACLK	TM0_TMR1	CAN0_RX	TM0_ACI2
PG_05	ETH1_TXEN	RSI0_CMD	PWM1_SYNC	ACM0_T1
PG_06	ETH1_REFCLK	RSI0_CLK	SPT2_BTDV	$\overline{\text{PWM1\_TRIP0}}$
PG_07	SPT2_BFS	TM0_TMR5		CNT0_ZM
PG_08	SPT2_AD1	TM0_TMR3		$\overline{\text{PWM1\_TRIP1}}$
PG_09	SPT2_AD0	TM0_TMR4		
PG_10	$\overline{\text{UART1\_RTS}}$	SPT2_BCLK		
PG_11	SPT2_BD1	TM0_TMR6		CNT0_UD
PG_12	SPT2_BD0	TM0_TMR7		CNT0_DG
PG_13	$\overline{\text{UART1\_CTS}}$			TM0_CLK
PG_14	$\overline{\text{UART1\_RX}}$	$\overline{\text{SYS\_IDLE1}}$		TM0_ACI1
PG_15	$\overline{\text{UART1\_TX}}$	$\overline{\text{SYS\_IDLE0}}$	$\overline{\text{SYS\_SLEEP}}$	TM0_ACI4

ADSP-BF60x DESIGNER QUICK REFERENCE

The table provides a quick reference summary of pin related information for circuit board design. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the Signal Name for every pin.
- Type: The Pin Type column in the table identifies the I/O type or supply type of the pin. The abbreviations used in this column are na (None), I/O (Digital input and/or output), a (Analog), s (Supply), and g (Ground).
- Driver Type: The Driver Type column in the table identifies the driver type used by the pin. The driver types are defined in Output Drive Currents on Page 102.
- Int Term: The Internal Termination column in the table specifies the termination present when the processor is not in the reset or hibernate state. The abbreviations used in this column are wk (Weak Keeper, weakly retains previous value driven on the pin), pu (Pull-up resistor), or pd (Pull-down resistor).
- Reset Term: The Reset Termination column in the table specifies the termination present when the processor is in the reset state. The abbreviations used in this column are wk (Weak Keeper, weakly retains previous value driven on the pin), pu (Pull-up resistor), or pd (Pull-down resistor).
- Reset Drive: The Reset Drive column in the table specifies the active drive on the signal when the processor is in the reset state.
- Hiber Term: The Hibernate Termination column in the table specifies the termination present when the processor is in the hibernate state. The abbreviations used in this column are wk (Weak Keeper, weakly retains previous value driven on the pin), pu (Pull-up resistor), or pd (Pull-down resistor).
- Hiber Drive: The Hibernate Drive column in the table specifies the active drive on the signal when the processor is in the hibernate state.
- Power Domain: The Power Domain column in the table specifies the power supply domain in which the signal resides.
- Description and Notes: The Description and Notes column in the table identifies any special requirements or characteristics for the signal. If no special requirements are listed the signal may be left unconnected if it is not used. Also, for multiplexed general-purpose I/O pins, this column identifies the functions available on the pin.

Table 15. ADSP-BF60x Designer Quick Reference

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
DMC0_A00	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 0. Notes: No notes.
DMC0_A01	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 1. Notes: No notes.
DMC0_A02	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 2. Notes: No notes.
DMC0_A03	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 3. Notes: No notes.
DMC0_A04	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 4. Notes: No notes.
DMC0_A05	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 5. Notes: No notes.
DMC0_A06	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 6. Notes: No notes.
DMC0_A07	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 7. Notes: No notes.
DMC0_A08	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 8. Notes: No notes.
DMC0_A09	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 9. Notes: No notes.
DMC0_A10	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 10. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
DMC0_A11	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 11. Notes: No notes.
DMC0_A12	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 12. Notes: No notes.
DMC0_A13	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 13. Notes: No notes.
DMC0_BA0	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 0. Notes: No notes.
DMC0_BA1	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 1. Notes: No notes.
DMC0_BA2	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 2. Notes: For LPDDR, leave unconnected.
$\overline{\text{DMC0\_CAS}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Column Address Strobe. Notes: No notes.
DMC0_CK	I/O	C	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock. Notes: No notes.
$\overline{\text{DMC0\_CK}}$	I/O	C	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock (complement). Notes: No notes.
DMC0_CKE	I/O	B	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock enable. Notes: No notes.
$\overline{\text{DMC0\_CS0}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Chip Select 0. Notes: No notes.
DMC0_DQ00	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 0. Notes: No notes.
DMC0_DQ01	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 1. Notes: No notes.
DMC0_DQ02	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 2. Notes: No notes.
DMC0_DQ03	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 3. Notes: No notes.
DMC0_DQ04	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 4. Notes: No notes.
DMC0_DQ05	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 5. Notes: No notes.
DMC0_DQ06	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 6. Notes: No notes.
DMC0_DQ07	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 7. Notes: No notes.
DMC0_DQ08	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 8. Notes: No notes.
DMC0_DQ09	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 9. Notes: No notes.
DMC0_DQ10	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 10. Notes: No notes.
DMC0_DQ11	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 11. Notes: No notes.
DMC0_DQ12	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 12. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
DMC0_DQ13	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 13. Notes: No notes.
DMC0_DQ14	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 14. Notes: No notes.
DMC0_DQ15	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 15. Notes: No notes.
DMC0_LDM	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Lower Byte. Notes: No notes.
DMC0_LDQS	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte. Notes: For LPDDR, a 100k ohm pull-down resistor is required.
$\overline{\text{DMC0\_LDQS}}$	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte (complement). Notes: For single ended DDR2, connect to VREF_DMC. For LPDDR, leave unconnected.
DMC0_ODT	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 On-die termination. Notes: For LPDDR, leave unconnected.
$\overline{\text{DMC0\_RAS}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Row Address Strobe. Notes: No notes.
DMC0_UDM	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Upper Byte. Notes: No notes.
DMC0_UDQS	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte. Notes: For LPDDR, a 100k ohm pull-down resistor is required.
$\overline{\text{DMC0\_UDQS}}$	I/O	C	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte (complement). Notes: For single ended DDR2, connect to VREF_DMC. For LPDDR, leave unconnected.
$\overline{\text{DMC0\_WE}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Write Enable. Notes: No notes.
GND	g	na	none	none	none	none	none	na	Desc: Ground. Notes: No notes.
$\overline{\text{JTG\_EMU}}$	I/O	A	none	none	none	none	none	VDD_EXT	Desc: Emulation Output. Notes: No notes.
JTG_TCK	I/O	na	pd	none	none	none	none	VDD_EXT	Desc: JTG Clock. Notes: Functional during reset.
JTG_TDI	I/O	na	pu	none	none	none	none	VDD_EXT	Desc: JTG Serial Data Input. Notes: Functional during reset.
JTG_TDO	I/O	A	none	none	none	none	none	VDD_EXT	Desc: JTG Serial Data Output. Notes: Functional during reset, three-state when $\overline{\text{JTG\_TRST}}$ is asserted.
JTG_TMS	I/O	na	pu	none	none	none	none	VDD_EXT	Desc: JTG Mode Select. Notes: Functional during reset.
$\overline{\text{JTG\_TRST}}$	I/O	na	pd	none	none	none	none	VDD_EXT	Desc: JTG Reset. Notes: Functional during reset.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PA_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 0 | SMC0 Address 3 | EPPI2 Data 0 | LP0 Data 0. Notes: No notes.
PA_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 1 | SMC0 Address 4 | EPPI2 Data 1 | LP0 Data 1. Notes: No notes.
PA_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 2 | SMC0 Address 5 | EPPI2 Data 2 | LP0 Data 2. Notes: No notes.
PA_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 3 | SMC0 Address 6 | EPPI2 Data 3 | LP0 Data 3. Notes: No notes.
PA_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 4 | SMC0 Address 7 | EPPI2 Data 4 | LP0 Data 4. Notes: No notes.
PA_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 5 | SMC0 Address 8 | EPPI2 Data 5 | LP0 Data 5. Notes: No notes.
PA_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 6 | SMC0 Address 9 | EPPI2 Data 6 | LP0 Data 6. Notes: No notes.
PA_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 7 | SMC0 Address 10 | EPPI2 Data 7 | LP0 Data 7. Notes: No notes.
PA_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 8 | SMC0 Address 11 | EPPI2 Data 8 | LP1 Data 0. Notes: No notes.
PA_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 9 | SMC0 Address 12 | EPPI2 Data 9 | LP1 Data 1. Notes: No notes.
PA_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 10 | SMC0 Address 14 | EPPI2 Data 10 | LP1 Data 2. Notes: No notes.
PA_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 11 | SMC0 Address 15 | EPPI2 Data 11 | LP1 Data 3. Notes: No notes.
PA_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 12 | SMC0 Address 17 | EPPI2 Data 12 | LP1 Data 4. Notes: No notes.
PA_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 13 | SMC0 Address 18 | EPPI2 Data 13 | LP1 Data 5. Notes: No notes.
PA_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 14 | SMC0 Address 19 | EPPI2 Data 14 | LP1 Data 6. Notes: No notes.
PA_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PA Position 15 | SMC0 Address 20 | EPPI2 Data 15 | LP1 Data 7. Notes: May be used to wake the processor from hibernate or deep sleep mode.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PB_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 0 | SMC0 NOR Clock | EPPI2 Clock | LP0 Clock. Notes: No notes.
PB_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 1 | SMC0 Memory Select 1 | EPPI2 Frame Sync 1 (HSYNC) | LP0 Acknowledge. Notes: No notes.
PB_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 2 | SMC0 Address 13 | EPPI2 Frame Sync 2 (VSYNC) | LP1 Acknowledge. Notes: No notes.
PB_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 3 | SMC0 Address 16 | EPPI2 Frame Sync 3 (FIELD) | LP1 Clock. Notes: No notes.
PB_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 4 | SMC0 Memory Select 2 | SMC0 Byte Enable 0 | SPORT0 Channel A Frame Sync. Notes: No notes.
PB_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 5 | SMC0 Memory Select 3 | SMC0 Byte Enable 1 | SPORT0 Channel A Clock. Notes: No notes.
PB_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 6 | SMC0 Address 21 | SPORT0 Channel A Transmit Data Valid | TIMER0 Alternate Clock 4. Notes: No notes.
PB_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 7 | SMC0 Address 22 | EPPI2 Data 16 | SPORT0 Channel B Frame Sync. Notes: No notes.
PB_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 8 | SMC0 Address 23 | EPPI2 Data 17 | SPORT0 Channel B Clock. Notes: No notes.
PB_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 9 | SMC0 Bus Grant Hang | SPORT0 Channel A Data 0 | TIMER0 Alternate Clock 2. Notes: No notes.
PB_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 10 | SMC0 Address 24 | SPORT0 Channel B Data 1 | TIMER0 Alternate Clock 0. Notes: No notes.
PB_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 11 | SMC0 Address 25 | SPORT0 Channel B Data 0 | TIMER0 Alternate Clock 3. Notes: No notes.
PB_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 12 | SMC0 Bus Grant | SPORT0 Channel B Transmit Data Valid | SPORT0 Channel A Data 1 | TIMER0 Alternate Clock 1. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PB_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 13 | EPPI1 Frame Sync 1 (HSYNC) | ETH0 Transmit Enable | TIMER0 Alternate Capture Input 6. Notes: No notes.
PB_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 14 | EPPI1 Clock | ETH0 Reference Clock. Notes: No notes.
PB_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PB Position 15 | EPPI1 Frame Sync 3 (FIELD) | ETH0 PTP Pulse-Per-Second Output. Notes: May be used to wake the processor from hibernate or deep sleep mode.
PC_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 0 | EPPI1 Data 0 | ETH0 Receive Data 0. Notes: No notes.
PC_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 1 | EPPI1 Data 1 | ETH0 Receive Data 1. Notes: No notes.
PC_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 2 | EPPI1 Data 2 | ETH0 Transmit Data 0. Notes: No notes.
PC_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 3 | EPPI1 Data 3 | ETH0 Transmit Data 1. Notes: No notes.
PC_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 4 | EPPI1 Data 4 | ETH0 Receive Error. Notes: No notes.
PC_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 5 | EPPI1 Data 5 | ETH0 Carrier Sense/RMII Receive Data Valid. Notes: No notes.
PC_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 6 | EPPI1 Data 6 | ETH0 Management Channel Clock. Notes: No notes.
PC_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 7 | EPPI1 Data 7 | ETH0 Management Channel Serial Data. Notes: No notes.
PC_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 8 | EPPI1 Data 8. Notes: No notes.
PC_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 9 | EPPI1 Data 9 | ETH1 PTP Pulse-Per-Second Output. Notes: No notes.
PC_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 10 | EPPI1 Data 10. Notes: No notes.
PC_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 11 | EPPI1 Data 11 | ETH PTP Auxiliary Trigger Input. Notes: No notes.
PC_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 12 | SPI0 Slave Select Output b | EPPI1 Data 12. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PC_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 13 | SPI0 Slave Select Output b | EPPI1 Data 13 | ETH PTP Clock Input. Notes: No notes.
PC_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 14 | SPI1 Slave Select Output b | EPPI1 Data 14. Notes: No notes.
PC_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PC Position 15 | SPI0 Slave Select Output b | EPPI1 Data 15. Notes: May be used to wake the processor from hibernate or deep sleep mode.
PD_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 0 | SPI0 Data 2 | EPPI1 Data 16 | SPI0 Slave Select Output b. Notes: No notes.
PD_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 1 | SPI0 Data 3 | EPPI1 Data 17 | SPI0 Slave Select Output b. Notes: No notes.
PD_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 2 | SPI0 Master In, Slave Out. Notes: No notes.
PD_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 3 | SPI0 Master Out, Slave In. Notes: No notes.
PD_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 4 | SPI0 Clock. Notes: No notes.
PD_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 5 | SPI1 Clock | TIMER0 Alternate Clock 7. Notes: No notes.
PD_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 6 | EPPI1 Frame Sync 2 (VSYNC) | ETH0 RMII Management Data Interrupt | TIMER0 Alternate Capture Input 5. Notes: May be used to wake the processor from hibernate or deep sleep mode.
PD_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 7 | UART0 Transmit | TIMER0 Alternate Capture Input 3. Notes: No notes.
PD_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 8 | UART0 Receive | TIMER0 Alternate Capture Input 0. Notes: No notes.
PD_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 9 | SPI1 Slave Select Output b | UART0 Request to Send | SPI0 Slave Select Output b. Notes: No notes.
PD_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 10 | SPI0 Ready | UART0 Clear to Send | SPI1 Slave Select Output b. Notes: No notes.
PD_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 11 | SPI0 Slave Select Output b | SPI0 Slave Select Input. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PD_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 12 | SPI1 Slave Select Output b | EPPI0 Data 20 | SPORT1 Channel A Data 1 | SPI1 Slave Select Input. Notes: No notes.
PD_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 13 | SPI1 Master Out, Slave In | TIMER0 Alternate Clock 5. Notes: No notes.
PD_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 14 | SPI1 Master In, Slave Out | TIMER0 Alternate Clock 6. Notes: No notes.
PD_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PD Position 15 | SPI1 Slave Select Output b | EPPI0 Data 21 | SPORT1 Channel A Data 0. Notes: No notes.
PE_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 0 | SPI1 Data 3 | EPPI0 Data 18 | SPORT1 Channel B Data 1. Notes: No notes.
PE_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 1 | SPI1 Data 2 | EPPI0 Data 19 | SPORT1 Channel B Data 0. Notes: No notes.
PE_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 2 | SPI1 Ready | EPPI0 Data 22 | SPORT1 Channel A Clock. Notes: No notes.
PE_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 3 | EPPI0 Data 16 | SPORT1 Channel B Frame Sync | ACM0 Frame Sync. Notes: No notes.
PE_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 4 | EPPI0 Data 17 | SPORT1 Channel B Clock | ACM0 Clock. Notes: No notes.
PE_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 5 | EPPI0 Data 23 | SPORT1 Channel A Frame Sync. Notes: No notes.
PE_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 6 | SPORT1 Channel A Transmit Data Valid | EPPI0 Frame Sync 3 (FIELD) | LP3 Clock. Notes: No notes.
PE_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 7 | SPORT1 Channel B Transmit Data Valid | EPPI0 Frame Sync 2 (VSYNC) | LP3 Acknowledge. Notes: No notes.
PE_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 8 | PWM0 Sync | EPPI0 Frame Sync 1 (HSYNC) | LP2 Acknowledge | ACM0 External Trigger 0. Notes: No notes.
PE_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 9 | EPPI0 Clock | LP2 Clock | PWM0 Shutdown Input. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PE_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 10 | PWM1 Channel D Low Side | RSI0 Data 6 | ETH1 Management Channel Clock. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PE_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 11 | PWM1 Channel D High Side | ETH1 Management Channel Serial Data | RSI0 Data 7. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PE_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 12 | PWM1 Channel C Low Side | RSI0 Data 5 | ETH1 RMII Management Data Interrupt. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details. May be used to wake the processor from hibernate or deep sleep mode.
PE_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 13 | PWM1 Channel C High Side | RSI0 Data 4 | ETH1 Carrier Sense/RMII Receive Data Valid. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PE_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 14 | SPORT2 Channel A Transmit Data Valid | TIMER0 Timer 0 | ETH1 Receive Error. Notes: No notes.
PE_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PE Position 15 | PWM1 Channel B Low Side | RSI0 Data 3 | ETH1 Receive Data 1. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PF_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 0 | PWM0 Channel A Low Side | EPPI0 Data 0 | LP2 Data 0. Notes: No notes.
PF_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 1 | PWM0 Channel A High Side | EPPI0 Data 1 | LP2 Data 1. Notes: No notes.
PF_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 2 | PWM0 Channel B Low Side | EPPI0 Data 2 | LP2 Data 2. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PF_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 3 | PWM0 Channel B High Side | EPPI0 Data 3 | LP2 Data 3. Notes: No notes.
PF_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 4 | PWM0 Channel C Low Side | EPPI0 Data 4 | LP2 Data 4. Notes: No notes.
PF_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 5 | PWM0 Channel C High Side | EPPI0 Data 5 | LP2 Data 5. Notes: No notes.
PF_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 6 | PWM0 Channel D Low Side | EPPI0 Data 6 | LP2 Data 6. Notes: No notes.
PF_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 7 | PWM0 Channel D High Side | EPPI0 Data 7 | LP2 Data 7. Notes: No notes.
PF_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 8 | SPI1 Slave Select Output b | EPPI0 Data 8 | LP3 Data 0. Notes: No notes.
PF_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 9 | SPI1 Slave Select Output b | EPPI0 Data 9 | LP3 Data 1. Notes: No notes.
PF_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 10 | ACM0 Address 4 | EPPI0 Data 10 | LP3 Data 2. Notes: No notes.
PF_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 11 | EPPI0 Data 11 | LP3 Data 3 | PWM0 Shutdown Input. Notes: No notes.
PF_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 12 | ACM0 Address 2 | EPPI0 Data 12 | LP3 Data 4. Notes: No notes.
PF_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 13 | ACM0 Address 3 | EPPI0 Data 13 | LP3 Data 5. Notes: No notes.
PF_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 14 | EPPI0 Data 14 | ACM0 Address 0 | LP3 Data 6. Notes: No notes.
PF_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PF Position 15 | ACM0 Address 1 | EPPI0 Data 15 | LP3 Data 7. Notes: No notes.
PG_00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 0 | PWM1 Channel B High Side | RSI0 Data 2 | ETH1 Receive Data 0. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PG_01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 1 | SPORT2 Channel A Frame Sync | TIMER0 Timer 2 | CAN0 Transmit. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PG_02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 2 | PWM1 Channel A Low Side | RSI0 Data 1 | ETH1 Transmit Data 1. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PG_03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 3 | PWM1 Channel A High Side | RSI0 Data 0 | ETH1 Transmit Data 0. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PG_04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 4 | SPORT2 Channel A Clock | TIMER0 Timer 1 | CAN0 Receive | TIMER0 Alternate Capture Input 2. Notes: May be used to wake the processor from hibernate or deep sleep mode.
PG_05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 5 | RSI0 Command | ETH1 Transmit Enable | PWM1 Sync | ACM0 External Trigger 1. Notes: Has an optional internal pull-up resistor for use with RSI. See the RSI chapter in the processor hardware reference for more details.
PG_06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 6 | RSI0 Clock | SPORT2 Channel B Transmit Data Valid | ETH1 Reference Clock | PWM1 Shutdown Input. Notes: No notes.
PG_07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 7 | SPORT2 Channel B Frame Sync | TIMER0 Timer 5 | CNT0 Count Zero Marker. Notes: No notes.
PG_08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 8 | SPORT2 Channel A Data 1 | TIMER0 Timer 3 | PWM1 Shutdown Input. Notes: No notes.
PG_09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 9 | SPORT2 Channel A Data 0 | TIMER0 Timer 4. Notes: No notes.
PG_10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 10 | UART1 Request to Send | SPORT2 Channel B Clock. Notes: No notes.
PG_11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 11 | SPORT2 Channel B Data 1 | TIMER0 Timer 6 | CNT0 Count Up and Direction. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
PG_12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 12 | SPORT2 Channel B Data 0 | TIMER0 Timer 7 | CNT0 Count Down and Gate. Notes: No notes.
PG_13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 13 | UART1 Clear to Send | TIMER0 Clock. Notes: No notes.
PG_14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 14 | UART1 Receive | SYS Core 1 Idle Indicator | TIMER0 Alternate Capture Input 1. Notes: No notes.
PG_15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: PG Position 15 | UART1 Transmit | SYS Core 0 Idle Indicator | SYS Processor Sleep Indicator | TIMER0 Alternate Capture Input 4. Notes: No notes.
SMC0_A01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Address 1. Notes: No notes.
SMC0_A02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Address 2. Notes: No notes.
$\overline{\text{SMC0\_AMS0}}$	I/O	A	pu	pu	none	pu	none	VDD_EXT	Desc: SMC0 Memory Select 0. Notes: No notes.
$\overline{\text{SMC0\_AOE\_}}$ NORDV	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 NOR Data Valid | SMC0 Output Enable. Notes: No notes.
SMC0_ARDY_ NORWT	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SMC0 NOR Wait | SMC0 Asynchronous Ready. Notes: Requires an external pull-up resistor.
$\overline{\text{SMC0\_ARE}}$	I/O	A	pu	pu	none	pu	none	VDD_EXT	Desc: SMC0 Read Enable. Notes: No notes.
$\overline{\text{SMC0\_AWE}}$	I/O	A	pu	pu	none	pu	none	VDD_EXT	Desc: SMC0 Write Enable. Notes: No notes.
$\overline{\text{SMC0\_BR}}$	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SMC0 Bus Request. Notes: Requires an external pull-up resistor.
SMC0_D00	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 0. Notes: No notes.
SMC0_D01	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 1. Notes: No notes.
SMC0_D02	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 2. Notes: No notes.
SMC0_D03	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 3. Notes: No notes.
SMC0_D04	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 4. Notes: No notes.
SMC0_D05	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 5. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
SMC0_D06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 6. Notes: No notes.
SMC0_D07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 7. Notes: No notes.
SMC0_D08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 8. Notes: No notes.
SMC0_D09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 9. Notes: No notes.
SMC0_D10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 10. Notes: No notes.
SMC0_D11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 11. Notes: No notes.
SMC0_D12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 12. Notes: No notes.
SMC0_D13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 13. Notes: No notes.
SMC0_D14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 14. Notes: No notes.
SMC0_D15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 15. Notes: No notes.
SYS_BMODE0	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 0. Notes: No notes.
SYS_BMODE1	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 1. Notes: No notes.
SYS_BMODE2	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 2. Notes: No notes.
SYS_CLKIN	a	na	none	none	none	none	none	VDD_EXT	Desc: SYS Clock Input/Crystal Input. Notes: Active during reset.
SYS_CLKOUT	I/O	A	none	none	L	none	none	VDD_EXT	Desc: SYS Processor Clock Output. Notes: No notes.
SYS_EXTWAKE	I/O	A	none	none	H	none	L	VDD_EXT	Desc: SYS External Wake Control. Notes: Drives low during hibernate and high all other times.
SYS_FAULT	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SYS Fault. Notes: Open source, requires an external pull-down resistor.
$\overline{\text{SYS\_FAULT}}$	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SYS Complementary Fault. Notes: Open drain, requires an external pull-up resistor.
$\overline{\text{SYS\_HWRST}}$	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Processor Hardware Reset Control. Notes: Active during reset.
$\overline{\text{SYS\_NMI\_RESOUT}}$	I/O	A	none	none	L	none	none	VDD_EXT	Desc: SYS Reset Output | SYS Non-maskable Interrupt. Notes: Requires an external pull-up resistor.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
SYS_PWRGD	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Power Good Indicator. Notes: If hibernate is not used or the internal Power Good Counter is used, connect to VDD_EXT.
SYS_TDA	a	na	none	none	none	none	none	VDD_TD	Desc: SYS Thermal Diode Anode. Notes: Active during reset and hibernate. If the thermal diode is not used, connect to ground.
SYS_TDK	a	na	none	none	none	none	none	VDD_TD	Desc: SYS Thermal Diode Cathode. Notes: Active during reset and hibernate. If the thermal diode is not used, connect to ground.
SYS_XTAL	a	na	none	none	none	none	none	VDD_EXT	Desc: SYS Crystal Output. Notes: Leave unconnected if an oscillator is used to provide SYS_CLKIN. Active during reset. State during hibernate is controlled by DPM_HIB_DIS.
TWI0_SCL	I/O	D	none	none	none	none	none	VDD_EXT	Desc: TWI0 Serial Clock. Notes: Open drain, requires external pull-up resistor. Consult Version 2.1 of the I2C specification for the proper resistor value. If TWI is not used, connect to ground.
TWI0_SDA	I/O	D	none	none	none	none	none	VDD_EXT	Desc: TWI0 Serial Data. Notes: Open drain, requires external pull-up resistor. Consult Version 2.1 of the I2C specification for the proper resistor value. If TWI is not used, connect to ground.
TWI1_SCL	I/O	D	none	none	none	none	none	VDD_EXT	Desc: TWI1 Serial Clock. Notes: Open drain, requires external pull-up resistor. Consult Version 2.1 of the I2C specification for the proper resistor value. If TWI is not used, connect to ground.
TWI1_SDA	I/O	D	none	none	none	none	none	VDD_EXT	Desc: TWI1 Serial Data. Notes: Open drain, requires external pull-up resistor. See the I2C-Bus Specification, Version 2.1,January 2000 for the proper resistor value. If TWI is not used, connect to ground.
USB0_CLKIN	a	na	none	none	none	none	none	VDD_USB	Desc: USB0 Clock/Crystal Input. Notes: If USB is not used, connect to ground. Active during reset.
USB0_DM	I/O	F	none	none	none	none	none	VDD_USB	Desc: USB0 Data –. Notes: Pull low if not using USB. For complete documentation of hibernate behavior when USB is used, see the USB chapter in the processor hardware reference.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
USB0_DP	I/O	F	none	none	none	none	none	VDD_USB	Desc: USB0 Data +. Notes: Pull low if not using USB. For complete documentation of hibernate behavior when USB is used, see the USB chapter in the processor hardware reference.
USB0_ID	I/O	na	none	none	none	pu	none	VDD_USB	Desc: USB0 OTG ID. Notes: If USB is not used, connect to ground. When USB is being used, the internal pull-up resistor that is present during hibernate is programmable. See the USB chapter in the processor hardware reference. Active during reset.
USB0_VBC	I/O	E	none	none	none	wk	none	VDD_USB	Desc: USB0 VBUS Control. Notes: If USB is not used, pull low.
USB0_VBUS	I/O	G	none	none	none	none	none	VDD_USB	Desc: USB0 Bus Voltage. Notes: If USB is not used, connect to ground.
VDD_DMC	s	na	none	none	none	none	none	na	Desc: VDD for DMC. Notes: If the DMC is not used, connect to VDD_INT.
VDD_EXT	s	na	none	none	none	none	none	na	Desc: External VDD. Notes: Must be powered.
VDD_INT	s	na	none	none	none	none	none	na	Desc: Internal VDD. Notes: Must be powered.
VDD_TD	s	na	none	none	none	none	none	na	Desc: VDD for Thermal Diode. Notes: If the thermal diode is not used, connect to ground.
VDD_USB	s	na	none	none	none	none	none	na	Desc: VDD for USB. Notes: If USB is not used, connect to VDD_EXT.
VREF_DMC	s	na	none	none	none	none	none	na	Desc: VREF for DMC. Notes: If the DMC is not used, connect to VDD_INT.

SPECIFICATIONS

For information about product specifications please contact your ADI representative.

OPERATING CONDITIONS

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DD_INT}	Internal Supply Voltage	CCLK ≤ 500 MHz	1.19	1.25	1.32	V
V_{DD_EXT}[1]	External Supply Voltage	1.8 V I/O	1.7	1.8	1.9	V
V_{DD_EXT}[1]	External Supply Voltage	3.3 V I/O	3.13	3.3	3.47	V
V_{DD_DMC}	DDR2/LPDDR Supply Voltage		1.7	1.8	1.9	V
V_{DD_USB}[2]	USB Supply Voltage		3.13	3.3	3.47	V
V_{DD_TD}	Thermal Diode Supply Voltage		3.13	3.3	3.47	V
V_{IH}[3]	High Level Input Voltage	V_{DD_EXT} = 3.47 V	2.1			V
V_{IH}[3]	High Level Input Voltage	V_{DD_EXT} = 1.9 V	0.7 × V_{DD_EXT}			V
V_{IHTWI}[4, 5]	High Level Input Voltage	V_{DD_EXT} = Maximum	0.7 × $V_{VBUSTWI}$		$V_{VBUSTWI}$	V
V_{IH_DDR2}[6, 7]		V_{DD_DMC} = 1.9 V	V_{DDR_REF} + 0.25			V
V_{IH_LPDDR}[8]		V_{DD_DMC} = 1.9 V	0.8 × V_{DD_DMC}			V
V_{ID_DDR2}[9]	Differential Input Voltage	V_{IX} = 1.075 V	0.50			V
V_{ID_DDR2}[9]	Differential Input Voltage	V_{IX} = 0.725 V	0.55			V
V_{IL}[3]	Low Level Input Voltage	V_{DD_EXT} = 3.13 V			0.8	V
V_{IL}[3]	Low Level Input Voltage	V_{DD_EXT} = 1.7 V			0.3 × V_{DD_EXT}	V
V_{ILTWI}[4, 5]	Low Level Input Voltage	V_{DD_EXT} = Minimum			0.3 × $V_{VBUSTWI}$	V
V_{IL_DDR2}[6, 7]		V_{DD_DMC} = 1.7 V			V_{DDR_REF} – 0.25	V
V_{IL_LPDDR}[8]		V_{DD_DMC} = 1.7 V			0.2 × V_{DD_DMC}	V
T_J	Junction Temperature	$T_{AMBIENT}$ = 0°C to +70°C	0		+105	°C
T_J	Junction Temperature	$T_{AMBIENT}$ = –40°C to +85°C	–40		+105	°C
T_J	Junction Temperature	$T_{AMBIENT}$ = –40°C to +105°C	–40		+125	°C

[1] Must remain powered (even if the associated function is not used).
[2] If not used, connect to 1.8 V or 3.3 V.
[3] Parameter value applies to all input and bidirectional signals except TWI signals, DMC0 signals and USB0 signals.
[4] Parameter applies to TWI signals.
[5] TWI signals are pulled up to V_{BUSTWI}. See Table 16.
[6] Parameter applies to DMC0 signals in DDR2 mode.
[7] V_{DDR_REF} is the voltage applied to pin V_{REF_DMC}, nominally V_{DD_DMC}/2.
[8] Parameter applies to DMC0 signals in LPDDR mode.
[9] Parameter applies to signals DMC0_CK, $\overline{\text{DMC0\_CK}}$, DMC0_LDQS, $\overline{\text{DMC0\_LDQS}}$, DMC0_UDQS, $\overline{\text{DMC0\_UDQS}}$ when used in DDR2 differential input mode.

Table 16. TWI_VSEL Selections and V_{DD_EXT}/V_{BUSTWI}

	V_{DD_EXT} Nominal	V_{BUSTWI} Min	V_{BUSTWI} Nom	V_{BUSTWI} Max	Unit
TWI000[1]	3.30	3.13	3.30	3.47	V
TWI001	1.80	1.70	1.80	1.90	V
TWI011	1.80	3.13	3.30	3.47	V
TWI100	3.30	4.75	5.00	5.25	V

[1] Designs must comply with the V_{DD_EXT} and V_{BUSTWI} voltages specified for the default TWI_DT setting for correct JTAG boundary scan operation during reset.

Clock Related Operating Conditions

Table 17 describes the core clock, system clock, and peripheral clock timing requirements. The data presented in the tables applies to all speed grades (found in Automotive Products on Page 112 and Ordering Guide on Page 112) except where expressly noted. Figure 8 provides a graphical representation of the various clocks and their available divider values.

Table 17. Clock Operating Conditions

Parameter		Restriction	Min	Typ	Max	Unit
f_{CCLK}	Core Clock Frequency	$f_{CCLK} \geq f_{SYSCLK}$			500	MHz
f_{SYSCLK}	SYSCLK Frequency				250	MHz
f_{SCLK0}	SCLK0 Frequency[1]	$f_{SYSCLK} \geq f_{SCLK0}$	30		125	MHz
f_{SCLK1}	SCLK1 Frequency	$f_{SYSCLK} \geq f_{SCLK1}$			125	MHz
f_{DCLK}	DDR2/LPDDR Clock Frequency	$f_{SYSCLK} \geq f_{DCLK}$			250	MHz
f_{OCLK}	Output Clock Frequency				125	MHz
$f_{SYS\_CLKOUTJ}$	SYS_CLKOUT Period Jitter[2, 3]			±1		%
f_{PVPCLK}	PVP Clock Frequency				83.3	MHz
$f_{NRCLKPROG}$	Programmed NOR Burst Clock				66.67	MHz
$f_{PCLKPROG}$	Programmed PPI Clock When Transmitting Data and Frame Sync				83.3	MHz
$f_{PCLKPROG}$	Programmed PPI Clock When Receiving Data or Frame Sync				62.5	MHz
$f_{PCLKEXT}$	External PPI Clock When Receiving Data and Frame Sync[4, 5]	$f_{PCLKEXT} \leq f_{SCLK0}$			83.3	MHz
$f_{PCLKEXT}$	External PPI Clock Transmitting Data or Frame Sync[4, 5]	$f_{PCLKEXT} \leq f_{SCLK0}$			58.8	MHz
$f_{LCLKTPROG}$	Programmed Link Port Transmit Clock				83.3	MHz
$f_{LCLKREXT}$	External Link Port Receive Clock[4, 5]	$f_{LCLKEXT} \leq f_{SCLK0}$			83.3	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Transmitting Data and Frame Sync				83.3	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Receiving Data or Frame Sync				62.5	MHz
$f_{SPTCLKEXT}$	External SPT Clock When Receiving Data and Frame Sync[4, 5]	$f_{SPTCLKEXT} \leq f_{SCLK1}$			83.3	MHz
$f_{SPTCLKEXT}$	External SPT Clock Transmitting Data or Frame Sync[4, 5]	$f_{SPTCLKEXT} \leq f_{SCLK1}$			58.8	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Transmitting Data				83.3	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data				75	MHz
$f_{SPICLKEXT}$	External SPI Clock When Receiving Data[4, 5]	$f_{SPICLKEXT} \leq f_{SCLK1}$			83.3	MHz
$f_{SPICLKEXT}$	External SPI Clock When Transmitting Data[4, 5]	$f_{SPICLKEXT} \leq f_{SCLK1}$			58.8	MHz
$f_{ACLKPROG}$	Programmed ACM Clock				62.5	MHz

[1] The minimum frequency for SCLK0 applies only when the USB is used.
[2] SYS_CLKOUT jitter is dependent on the application system design including pin switching activity, board layout, and the jitter characteristics of the SYS_CLKIN source. Due to the dependency on these factors the measured jitter may be higher or lower than this typical specification for each end application.
[3] The value in the Typ field is the percentage of the SYS_CLKOUT period.
[4] The maximum achievable frequency for any peripheral in external clock mode is dependent on being able to meet the setup and hold times in the AC timing specifications section for that peripheral. Pay particular attention to setup and hold times for VDD_EXT = 1.8 V which may preclude the maximum frequency listed here.
[5] The peripheral external clock frequency must also be less than or equal to the f_{SCLK} (f_{SCLK0} or f_{SCLK1}) that clocks the peripheral.

Table 18. Phase-Locked Loop Operating Conditions

Parameter		Min	Max	Unit
f_{PLLCLK}	PLL Clock Frequency	250	1000	MHz

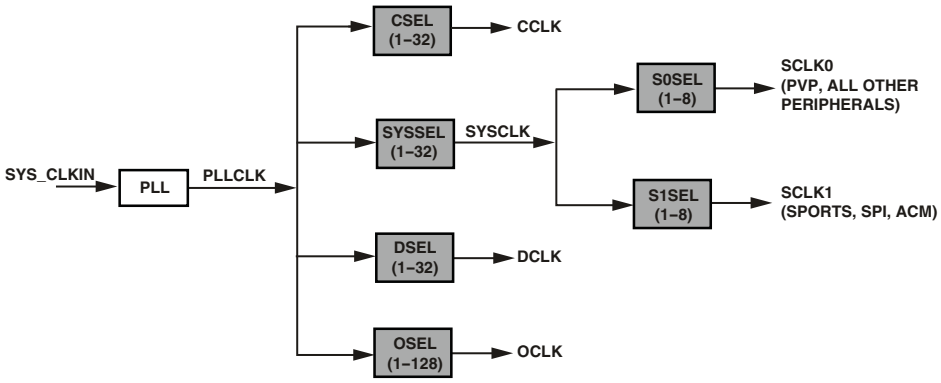


Figure 8. Clock Relationships and Divider Values

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	Min	Typical	Max	Unit
V_{OH}[1]	High Level Output Voltage	V_{DD_EXT} = 1.7 V, I_{OH} = –0.5 mA	V_{DD_EXT} – 0.40			V
V_{OH}[1]	High Level Output Voltage	V_{DD_EXT} = 3.13 V, I_{OH} = –0.5 mA	V_{DD_EXT} – 0.40			V
V_{OH_DDR2}[2]	High Level Output Voltage, ds = 00	V_{DD_DMC} = 1.70 V, I_{OH} = –13.4 mA	1.388			V
V_{OH_DDR2}[3]	High Level Output Voltage, ds = 10	V_{DD_DMC} = 1.70 V, I_{OH} = –6.70 mA	1.311			V
V_{OH_LPDDR}[4]	High Level Output Voltage, ds = 00	V_{DD_DMC} = 1.70 V, I_{OH} = –11.2 mA	1.300			V
V_{OH_LPDDR}[5]	High Level Output Voltage, ds = 01	V_{DD_DMC} = 1.70 V, I_{OH} = –7.85 mA	1.300			V
V_{OH_LPDDR}[6]	High Level Output Voltage, ds = 10	V_{DD_DMC} = 1.70 V, I_{OH} = –5.10 mA	1.300			V
V_{OH_LPDDR}[7]	High Level Output Voltage, ds = 11	V_{DD_DMC} = 1.70 V, I_{OH} = –2.55 mA	1.300			V
V_{OL}[8]	Low Level Output Voltage	V_{DD_EXT} = 1.7 V, I_{OL} = 2.0 mA			0.400	V
V_{OL}[8]	Low Level Output Voltage	V_{DD_EXT} = 3.13 V, I_{OL} = 2.0 mA			0.400	V
V_{OL_DDR2}[2]	Low Level Output Voltage, ds = 00	V_{DD_DMC} = 1.70 V, I_{OL}13.4 mA			0.312	V
V_{OL_DDR2}[3]	Low Level Output Voltage, ds = 10	V_{DD_DMC} = 1.70 V, I_{OL} = 6.70 mA			0.390	V
V_{OL_LPDDR}[4]	Low Level Output Voltage, ds = 00	V_{DD_DMC} = 1.70 V, I_{OL} = 11.2 mA			0.400	V
V_{OL_LPDDR}[5]	Low Level Output Voltage, ds = 01	V_{DD_DMC} = 1.70 V, I_{OL} = 7.85 mA			0.400	V
V_{OL_LPDDR}[6]	Low Level Output Voltage, ds = 10	V_{DD_DMC} = 1.70 V, I_{OL} = 5.10 mA			0.400	V
V_{OL_LPDDR}[7]	Low Level Output Voltage, ds = 11	V_{DD_DMC} = 1.70 V, I_{OL} = 2.55 mA			0.400	V
I_{IH}[9]	High Level Input Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 3.47 V			10	µA
I_{IH_PD}[10]	High Level Input Current with Pull-down Resistor	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 3.47 V			110	µA
I_{IL}[11]	Low Level Input Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 0 V			10	µA
I_{IL_PU}[12]	Low Level Input Current with Pull-up Resistor	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 0 V			100	µA
I_{IH_USB0}[13]	High Level Input Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 3.47 V			240	µA
I_{IL_USB0}[13]	Low Level Input Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 0 V			100	µA
I_{OZH}[14]	Three-State Leakage Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 3.47 V			10	µA
I_{OZH}[15]	Three-State Leakage Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 1.9 V			10	µA
I_{OZL}[16]	Three-State Leakage Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 0 V			10	µA
I_{OZL_PU}[17]	Three-State Leakage Current with Pull-up Resistor	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 0 V			100	µA
I_{OZH_TWI}[18]	Three-State Leakage Current	V_{DD_EXT} = 3.47 V, V_{DD_DMC} = 1.9 V, V_{DD_USB} = 3.47 V, V_{IN} = 5.5 V			10	µA
C_{IN}[19, 20]	Input Capacitance	$T_{AMBIENT}$ = 25°C		4.9	6.7	pF
C_{IN_TWI}[18, 20]	Input Capacitance	$T_{AMBIENT}$ = 25°C		8.9	9.9	pF
C_{IN_DDR}[20, 21]	Input Capacitance	$T_{AMBIENT}$ = 25°C		5.8	6.6	pF
I_{DD_TD}	V_{DD_TD} Current	V_{DD_TD} = 3.3 V			1	µA
$I_{DD\_DEEPSLEEP}$[22, 23]	V_{DD_INT} Current in Deep Sleep Mode	f_{CCLK} = 0 MHz $f_{SCLK0/1}$ = 0 MHz		Table 21 on Page 58		mA

Parameter		Test Conditions	Min	Typical	Max	Unit
I_{DD_IDLE}[23]	V_{DD_INT} Current in Idle	f_{CCLK} = 500 MHz ASFC0 = 0.14 (Idle) ASFC1 = 0 (Disabled) f_{SYSCLK} = 250 MHz, $f_{SCLK0/1}$ = 125 MHz f_{DCLK} = 0 MHz (DDR Disabled) f_{USBCLK} = 0 MHz (USB Disabled) No PVP or DMA activity T_J = 25°C		137		mA
I_{DD_TYP}[23]	V_{DD_INT} Current	f_{CCLK} = 500 MHz ASFC0 = 1.0 (Full-on Typical) ASFC1 = 0.86 (App) f_{SYSCLK} = 250 MHz, $f_{SCLK0/1}$ = 125 MHz f_{DCLK} = 250 MHz f_{USBCLK} = 0 MHz (USB Disabled) DMA Data Rate = 124 MB/s Medium PVP Activity T_J = 25°C		357		mA
$I_{DD\_HIBERNATE}$[22, 24]	Hibernate State Current	V_{DD_INT} = 0 V, V_{DD_EXT} = V_{DD_TD} = V_{DD_USB} = 3.3 V, V_{DD_DMC} = 1.8 V, V_{REF_DMC} = 0.9 V, T_J = 25°C, f_{CLKIN} = 0 MHz		40		µA
$I_{DD\_HIBERNATE}$[22, 24]	Hibernate State Current Without USB	V_{DD_INT} = 0 V, V_{DD_EXT} = V_{DD_TD} = V_{DD_USB} = 3.3 V, V_{DD_DMC} = 1.8 V, V_{REF_DMC} = 0.9 V, T_J = 25°C, f_{CLKIN} = 0 MHz, USB protection disabled (USB0_PHY_CTL.DIS=1)		10		µA
I_{DD_INT}[23]	V_{DD_INT} Current	f_{CCLK} > 0 MHz $f_{SCLK0/1}$ ≥ 0 MHz			See I_{DDINT_TOT} equation on Page 57	mA

[1] Applies to all output and bidirectional signals except DMC0 signals, TWI signals and USB0 signals.
[2] Applies to all DMC0 output and bidirectional signals in DDR2 full drive strength mode.
[3] Applies to all DMC0 output and bidirectional signals in DDR2 half drive strength mode.
[4] Applies to all DMC0 output and bidirectional signals in LPDDR full drive strength mode.
[5] Applies to all DMC0 output and bidirectional signals in LPDDR three-quarter drive strength mode.
[6] Applies to all DMC0 output and bidirectional signals in LPDDR half drive strength mode.
[7] Applies to all DMC0 output and bidirectional signals in LPDDR one-quarter drive strength mode.
[8] Applies to all output and bidirectional signals except DMC0 signals and USB0 signals.
[9] Applies to signals SMC0_ARDY, $\overline{\text{SMC0\_BR}}$, SYS_BMODE0–2, SYS_CLKIN, SYS_HWRST, SYS_PWRGD, JTG_TDI, and $\overline{\text{JTG\_TMS}}$.
[10] Applies to signals JTG_TCK and $\overline{\text{JTG\_TRST}}$.
[11] Applies to signals SMC0_ARDY, $\overline{\text{SMC0\_BR}}$, SYS_BMODE0–2, SYS_CLKIN, SYS_HWRST, SYS_PWRGD, JTG_TCK, and $\overline{\text{JTG\_TRST}}$.
[12] Applies to signals JTG_TDI, JTG_TMS.
[13] Applies to signal USB0_CLKIN.
[14] Applies to signals PA0–15, PB0–15, PC0–15, PD0–15, PE0–15, PF0–15, PG0–15, $\overline{\text{SMC0\_AMS0}}$, $\overline{\text{SMC0\_ARE}}$, $\overline{\text{SMC0\_AWE}}$, $\overline{\text{SMC0\_A0E}}$, SMC0_A01–02, SMC0_D00–15, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{JTG\_EMU}}$, JTG_TDO, USB0_DM, USB0_DP, USB0_ID, USB0_VBC, USB0_VBUS.
[15] Applies to DMC0_A[00:13], $\overline{\text{DMC0\_BA}}$[0:2], $\overline{\text{DMC0\_CAS}}$, $\overline{\text{DMC0\_CS0}}$, DMC0_DQ[00:15], DMC0_LQDS, $\overline{\text{DMC0\_LDQS}}$, DMC0_UDQS, $\overline{\text{DMC0\_UDQS}}$, DMC0_LDM, DMC0_UDM, DMC0_ODT, $\overline{\text{DMC0\_RAS}}$, and $\overline{\text{DMC0\_WE}}$.
[16] Applies to signals PA0–15, PB0–15, PC0–15, PD0–15, PE0–15, PF0–15, PG0–15, $\overline{\text{SMC0\_A0E}}$, SMC0_A01–02, SMC0_D00–15, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{JTG\_EMU}}$, JTG_TDO, USB0_DM, USB0_DP, USB0_ID, USB0_VBC, USB0_VBUS, DMC0_A00–13, DMC0_BA0–2, $\overline{\text{DMC0\_CAS}}$, $\overline{\text{DMC0\_CS0}}$, DMC0_DQ00–15, DMC0_LQDS, $\overline{\text{DMC0\_LDQS}}$, DMC0_UDQS, $\overline{\text{DMC0\_UDQS}}$, DMC0_LDM, DMC0_UDM, DMC0_ODT, $\overline{\text{DMC0\_RAS}}$, $\overline{\text{DMC0\_WE}}$, and TWI signals.
[17] Applies to signals $\overline{\text{SMC0\_AMS0}}$, $\overline{\text{SMC0\_ARE}}$, $\overline{\text{SMC0\_AWE}}$, and when RSI pull-up resistors are enabled, PE10–13, 15 and PG00, 02, 03, 05.
[18] Applies to all TWI signals.
[19] Applies to all signals, except DMC0 and TWI signals.
[20] Guaranteed, but not tested.
[21] Applies to all DMC0 signals.
[22] See the *ADSP-BF60x Blackfin Processor Hardware Reference Manual* for definition of deep sleep and hibernate operating modes.
[23] Additional information can be found at Total Internal Power Dissipation on Page 57.
[24] Applies to V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB} and V_{DD_TD} supply signals only. Clock inputs are tied high or low.

Total Internal Power Dissipation

Total power dissipation has two components:

1. Static, including leakage current (deep sleep)
2. Dynamic, due to transistor switching characteristics for each clock domain

Many operating conditions can also affect power dissipation, including temperature, voltage, operating frequency, and processor activity. The following equation describes the internal current consumption.

$$I_{DDINT\_TOT} = I_{DDINT\_CCLK\_DYN} + I_{DDINT\_SYSCLK\_DYN} + I_{DDINT\_SCLK0\_DYN} + I_{DDINT\_SCLK1\_DYN} + I_{DDINT\_DCLK\_DYN} + I_{DDINT\_USBCLK\_DYN} + I_{DDINT\_DMA\_DR\_DYN} + I_{DDINT\_DEEPSLEEP} + I_{DDINT\_PVP\_DYN}$$

$I_{DDINT\_DEEPSLEEP}$ is the only item present that is part of the static power dissipation component. $I_{DDINT\_DEEPSLEEP}$ is specified as a function of voltage (V_{DD_INT}) and temperature (see Table 21).

There are eight different items that contribute to the dynamic power dissipation. These components fall into three broad categories: application-dependent currents, clock currents and data transmission currents.

Application-Dependent Current

The application-dependent currents include the dynamic current in the core clock domain and the dynamic current of the PVP.

Core clock (CCLK) use is subject to an activity scaling factor (ASF) that represents application code running on the processor cores and L1/L2 memories (Table 20). The ASF is combined with the CCLK frequency and V_{DD_INT} dependent data in Table 19 to calculate this portion.

$I_{DDINT\_CCLK\_DYN}$ (mA) = Table 19 × (ASFC0 + ASFC1)

The dynamic current of the PVP is determined by selecting the appropriate use case from Table 22.

$I_{DDINT\_PVP\_DYN}$ (mA) = Table 22

Clock Current

The dynamic clock currents provide the total power dissipated by all transistors switching in the clock paths. The power dissipated by each clock domain is dependent on voltage (V_{DD_INT}), operating frequency and a unique scaling factor.

$I_{DDINT\_SYSCLK\_DYN}$ (mA) = 0.187 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK0\_DYN}$ (mA) = 0.217 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK1\_DYN}$ (mA) = 0.042 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_DCLK\_DYN}$ (mA) = 0.024 × f_{DCLK} (MHz) × V_{DD_INT} (V)

The dynamic component of the USB clock is a unique case. The USB clock contributes a near constant current value when used.

$I_{DDINT\_USBCLK\_DYN}$ (mA) = 5 mA (if USB enabled)

Data Transmission Current

The data transmission current represents the power dissipated when transmitting data. This current is expressed in terms of data rate. The calculation is performed by adding the data rate (MB/s) of each DMA and core driven access to peripherals and L2/external memory. This number is then multiplied by a coefficient and V_{DD_INT}. The following equation provides an estimate of all data transmission current.

$I_{DDINT\_DMA\_DR\_DYN}$(mA) = 0.0578 × data rate (MB/s) × V_{DD_INT} (V)

For details on using this equation see the related Engineer Zone material.

Table 19. CCLK Dynamic Current per core (mA, with ASF = 1)

	Voltage (V_{DD_INT})						
f_{CCLK} (MHz)	**1.190**	**1.200**	**1.225**	**1.250**	**1.275**	**1.300**	**1.320**
500	97.9	98.8	101.5	103.9	106.7	109.3	110.8
450	88.6	89.5	91.9	94.1	96.7	98.9	100.6
400	79.3	80.1	82.2	84.3	86.5	88.6	90.1
350	70.0	70.7	72.5	74.4	76.3	78.3	79.4
300	60.6	61.2	63.0	64.6	66.3	68.0	69.1
250	51.3	51.8	53.2	54.7	56.3	57.6	58.5
200	42.0	42.4	43.6	44.8	46.0	47.2	48.2
150	32.5	32.9	34.0	34.8	35.9	37.0	37.4
100	23.2	23.5	24.2	25.0	25.7	26.5	26.9

Table 20. Activity Scaling Factors (ASF)

I_{DDINT} Power Vector	ASF
$I_{DD-PEAK}$	1.34
$I_{DD-HIGH}$	1.25
$I_{DD-FULL-ON-TYP}$	1.00
I_{DD-APP}	0.86
I_{DD-NOP}	0.72
$I_{DD-IDLE}$	0.14

Table 21. Static Current—$I_{DD\_DEEPSLEEP}$ (mA)

	Voltage (V_{DD_INT})						
T_J (°C)	1.190	1.200	1.225	1.250	1.275	1.300	1.320
–40	1.7	1.8	2.2	2.5	2.7	3.1	3.4
–20	4.0	4.2	4.6	5.1	5.6	6.2	6.8
0	8.4	9.0	9.6	10.6	11.5	12.5	13.4
25	19.0	19.8	21.5	23.2	25.3	27.2	29.0
40	29.9	31.7	34.4	36.8	40.0	42.8	45.4
55	46.6	48.9	52.4	56.4	60.6	65.0	68.1
70	66.4	70.4	75.5	80.6	86.2	92.4	97.9
85	93.9	99.3	105.9	113.0	120.7	128.9	136.4
100	137.2	144.2	153.6	163.4	173.9	185.1	194.1
105	153.8	162.4	172.5	183.4	195.2	207.5	217.5
115	193.3	203.7	216.2	229.5	243.9	258.6	271.1
125	236.1	247.2	261.8	277.3	294.0	311.9	326.4

Table 22. $I_{DDINT\_PVP\_DYN}$ (mA)

PVP Activity Level	PVPSF (PVP Scaling Factor)
High	42.4
Medium	20
Low	0

PROCESSOR — ABSOLUTE MAXIMUM RATINGS

Stresses greater than those listed in Table 23 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 23. Absolute Maximum Ratings

Parameter	Rating
Internal Supply Voltage (V_{DD_INT})	–0.33 V to 1.32 V
External (I/O) Supply Voltage (V_{DD_EXT})	–0.33 V to 3.63 V
Thermal Diode Supply Voltage (V_{DD_TD})	–0.33 V to 3.63 V
DDR2 Controller Supply Voltage (V_{DD_DMC})	–0.33 V to 1.90 V
USB PHY Supply Voltage (V_{DD_USB})	–0.33 V to 3.63 V
Input Voltage[1, 2, 3]	–0.33 V to 3.63 V
TWI Input Voltage[2, 4]	–0.33 V to 5.50 V
USB0_Dx Input Voltage[5]	–0.33 V to 5.25 V
USB0_VBUS Input Voltage[5]	–0.33 V to 6.00 V
DDR2 Input Voltage[6]	–0.33 V to 1.90 V
Output Voltage Swing	–0.33 V to V_{DD_EXT} + 0.5 V
I_{OH}/I_{OL} Current per Signal[1]	12.5 mA (max)
Storage Temperature Range	–65°C to +150°C
Junction Temperature Under Bias	+125°C

[1] Applies to 100% transient duty cycle.
[2] Applies only when V_{DD_EXT} is within specifications. When V_{DD_EXT} is outside specifications, the range is V_{DD_EXT} ± 0.2 V.
[3] For other duty cycles see Table 24.
[4] Applies to balls TWI_SCL and TWI_SDA.
[5] If the USB is not used, connect USB0_Dx and USB0_VBUS according to Table 15 on Page 37.
[6] Applies only when V_{DD_DMC} is within specifications. When V_{DD_DMC} is outside specifications, the range is V_{DD_DMC} ± 0.2 V.

Table 24. Maximum Duty Cycle for Input Transient Voltage[1, 2]

Maximum Duty Cycle (%)[2]	V_{IN} Min (V)[3]	V_{IN} Max (V)[3]
100	–0.33	3.63
50	–0.50	3.80
40	–0.56	3.86
25	–0.67	3.97
20	–0.73	4.03
15	–0.80	4.10
10	–0.90	4.20

[1] Applies to all signal balls with the exception of SYS_CLKIN, SYS_XTAL, SYS_EXT_WAKE, USB0_DP, USB0_DM, USB0_VBUS, TWI signals, and DMC0 signals.
[2] Applies only when V_{DD_EXT} is within specifications. When V_{DD_EXT} is outside specifications, the range is V_{DD_EXT} ± 0.2 V.
[3] 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 specified voltages, and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.

ESD SENSITIVITY

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

PROCESSOR — PACKAGE INFORMATION

The information presented in Figure 9 and Table 25 provides details about package branding. For a complete listing of product availability, see Automotive Products on Page 112.

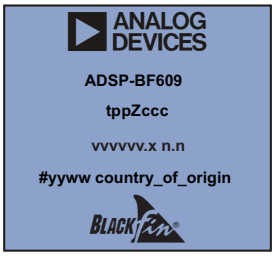


Figure 9. Product Information on Package

Table 25. Package Brand Information

Brand Key	Field Description
ADSP-BF609	Product Model
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Designation
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
yyww	Date Code

TIMING SPECIFICATIONS

Specifications are subject to change without notice.

Clock and Reset Timing

Table 26 and Figure 10 describe clock and reset operations. Per the CCLK, SYSCLK, SCLK0, SCLK1, DCLK, and OCLK timing specifications in Table 17 on Page 53, combinations of SYS_CLKIN and clock multipliers must not select clock rates in excess of the processor's maximum instruction rate.

Table 26. Clock and Reset Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
f_{CKIN}	SYS_CLKIN Frequency (using a crystal)[1, 2, 3]	20	50	MHz
f_{CKIN}	SYS_CLKIN Frequency (using a crystal oscillator)[1, 2, 3]	20	60	MHz
t_{CKINL}	SYS_CLKIN Low Pulse[1]	6.67		ns
t_{CKINH}	SYS_CLKIN High Pulse[1]	6.67		ns
t_{WRST}	$\overline{\text{SYS\_HWRST}}$ Asserted Pulse Width Low[4]	11 × t_{CKIN}		ns

[1] Applies to PLL bypass mode and PLL non bypass mode.
[2] The t_{CKIN} period (see Figure 10) equals $1/f_{CKIN}$.
[3] If the CGU_CTL.DF bit is set, the minimum f_{CKIN} specification is 40 MHz.
[4] Applies after power-up sequence is complete. See Table 27 and Figure 11 for power-up reset timing.

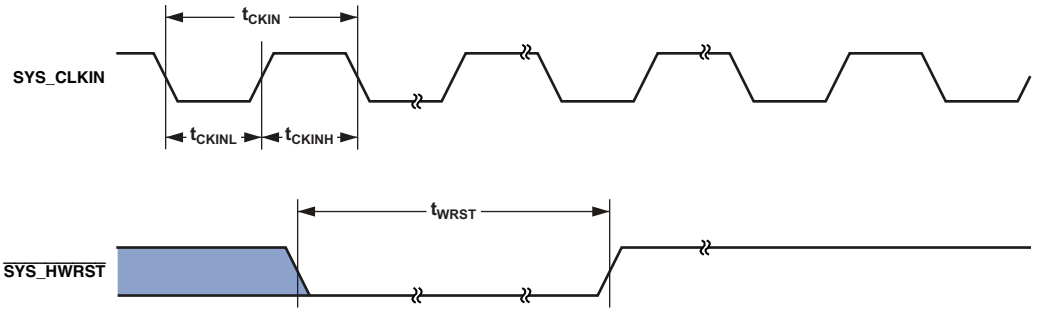


Figure 10. Clock and Reset Timing

Power-Up Reset Timing

In Figure 11, $V_{DD\_SUPPLIES}$ are V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB}, and V_{DD_TD}.

Table 27. Power-Up Reset Timing

Parameter		Min	Max	Unit
Timing Requirement				
$t_{RST\_IN\_PWR}$	$\overline{SYS\_HWRST}$ Deasserted after V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB}, V_{DD_TD}, and SYS_CLKIN are Stable and Within Specification	$11 \times t_{CKIN}$		ns

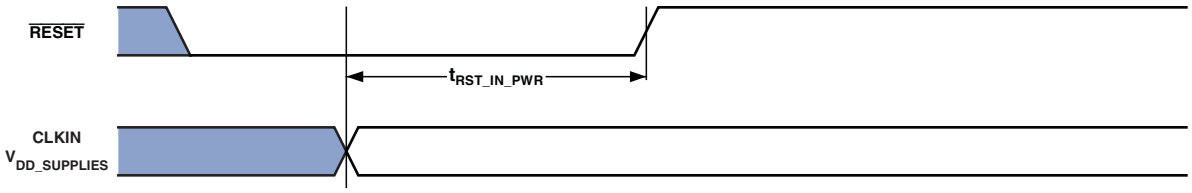


Figure 11. Power-Up Reset Timing

Asynchronous Read

Table 28. Asynchronous Memory Read (BxMODE = b#00)

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
$t_{SDATARE}$	DATA in Setup Before $\overline{SMC0\_ARE}$ High	8.2		ns
$t_{HDATARE}$	DATA in Hold After $\overline{SMC0\_ARE}$ High	0		ns
$t_{DARDYARE}$	SMC0_ARDY Valid After $\overline{SMC0\_ARE}$ Low[1, 2]		(RAT – 2.5) × t_{SCLK0} – 17.5	ns
Switching Characteristics				
$t_{ADDRARE}$	SMC0_Ax/$\overline{SMC0\_AMSx}$ Assertion Before $\overline{SMC0\_ARE}$ Low[3]	(PREST + RST + PREAT) × t_{SCLK0} – 2		ns
t_{AOEARE}	$\overline{SMC0\_AOE}$ Assertion Before $\overline{SMC0\_ARE}$ Low	(RST + PREAT) × t_{SCLK0} – 2		ns
t_{HARE}	Output[4] Hold After $\overline{SMC0\_ARE}$ High[5]	RHT × t_{SCLK0} –2		ns
t_{WARE}	$\overline{SMC0\_ARE}$ Active Low Width[6]	RAT × t_{SCLK0} – 2		ns
$t_{DAREARDY}$	$\overline{SMC0\_ARE}$ High Delay After SMC0_ARDY Assertion[1]	2.5 × t_{SCLK0}	3.5 × t_{SCLK0} + 17.5	ns

[1] SMC0_BxCTL.ARDYEN bit = 1.
[2] RAT value set using the SMC_BxTIM.RAT bits.
[3] PREST, RST, and PREAT values set using the SMC_BxETIM.PREST bits, SMC_BxTIM.RST bits, and the SMC_BxETIM.PREAT bits.
[4] Output signals are SMC0_Ax, $\overline{SMC0\_AMS}$, $\overline{SMC0\_AOE}$, $\overline{SMC0\_ABEx}$.
[5] RHT value set using the SMC_BxTIM.RHT bits.
[6] SMC0_BxCTL.ARDYEN bit = 0.

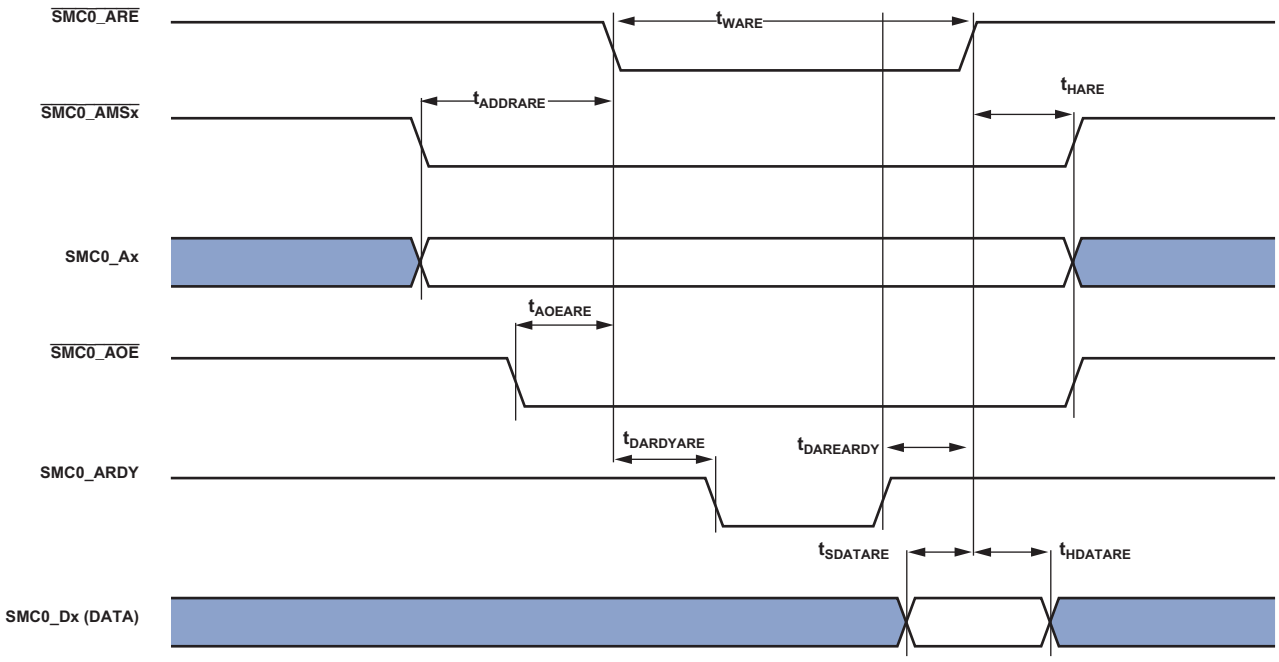


Figure 12. Asynchronous Read

Asynchronous Flash Read

Table 29. Asynchronous Flash Read

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
t_{AMSADV}	SMC0_Ax (Address)/$\overline{\text{SMC0\_AMSx}}$ Assertion Before SMC0_NORDV Low[1]	PREST × t_{SCLK0} – 2		ns
t_{WADV}	SMC0_NORDV Active Low Width[2]	RST × t_{SCLK0} – 2		ns
$t_{DADVARE}$	$\overline{\text{SMC0\_ARE}}$ Low Delay From SMC0_NORDV High[3]	PREAT × t_{SCLK0} – 2		ns
t_{HARE}	Output[4] Hold After $\overline{\text{SMC0\_ARE}}$ High[5]	RHT × t_{SCLK0} – 2		ns
t_{WARE}[6]	$\overline{\text{SMC0\_ARE}}$ Active Low Width[7]	RAT × t_{SCLK0} – 2		ns

[1] PREST value set using the SMC_BxETIM.PREST bits.
[2] RST value set using the SMC_BxTIM.RST bits.
[3] PREAT value set using the SMC_BxETIM.PREAT bits.
[4] Output signals are SMC0_Ax, $\overline{\text{SMC0\_AMS}}$, $\overline{\text{SMC0\_AOE}}$.
[5] RHT value set using the SMC_BxTIM.RHT bits.
[6] SMC0_BxCTL.ARDYEN bit = 0.
[7] RAT value set using the SMC_BxTIM.RAT bits.

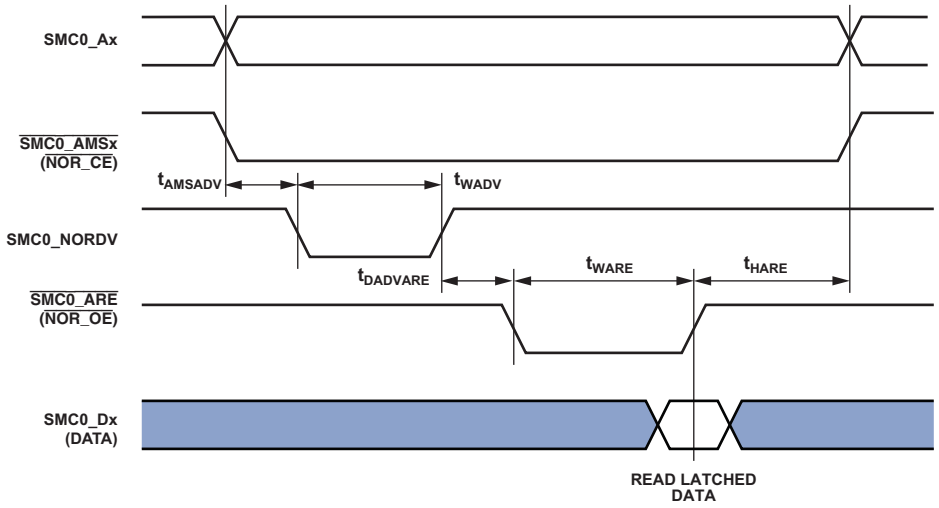


Figure 13. Asynchronous Flash Read

Asynchronous Page Mode Read

Table 30. Asynchronous Page Mode Read

Parameter		V_{DD_EXT} 1.8 V /3.3 V Nominal Min	Max	Unit
Switching Characteristics				
t_{AV}	SMC0_Ax (Address) Valid for First Address Min Width[1]	(PREST + RST + PREAT + RAT) × t_{SCLK0} – 2		ns
t_{AV1}	SMC0_Ax (Address) Valid for Subsequent SMC0_Ax (Address) Min Width	PGWS × t_{SCLK0} – 2		ns
t_{WADV}	SMC0_NORDV Active Low Width[2]	RST × t_{SCLK0} – 2		ns
t_{HARE}	Output[3] Hold After $\overline{\text{SMC0\_ARE}}$ High[4]	RHT × t_{SCLK0} – 2		ns
t_{WARE}[5]	$\overline{\text{SMC0\_ARE}}$ Active Low Width[6, 7]	(RAT + (Nw – 1) × PGWS) × t_{SCLK0} – 2		ns

[1] PREST, RST, PREAT and RAT values set using the SMC_BxETIM.PREST bits, SMC_BxTIM.RST bits, SMC_BxETIM.PREAT bits, and the SMC_BxTIM.RAT bits.
[2] RST value set using the SMC_BxTIM.RST bits.
[3] Output signals are SMC0_Ax, $\overline{\text{SMC0\_AMSx}}$, $\overline{\text{SMC0\_AOE}}$.
[4] RHT value set using the SMC_BxTIM.RHT bits.
[5] SMC_BxCTL.ARDYEN bit = 0.
[6] RAT value set using the SMC_BxTIM.RAT bits.
[7] Nw = Number of 16-bit data words read.

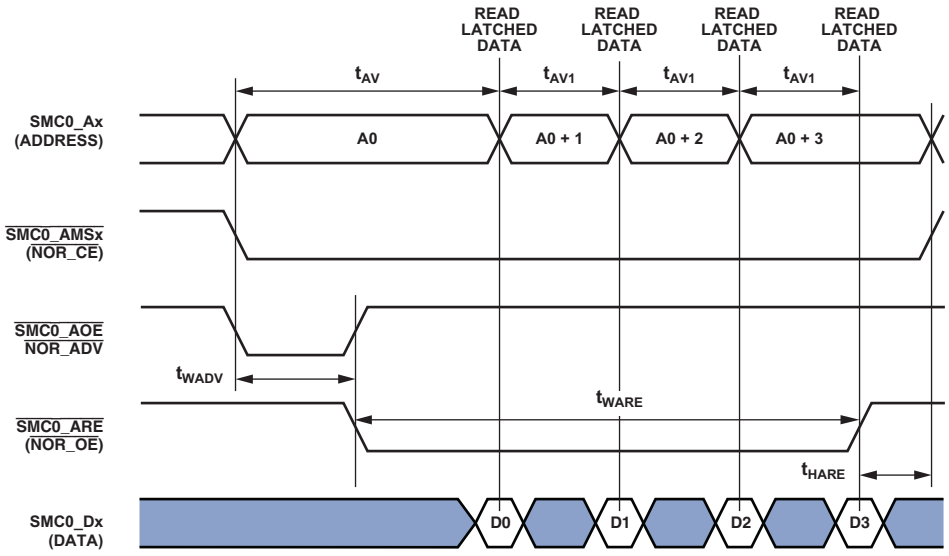


Figure 14. Asynchronous Page Mode Read

Synchronous Burst Flash Read

In synchronous burst mode the programmed NOR burst clock ($f_{NRCLKPROG}$) frequency in MHz is set by the following equation where BCLK is a field in the SMC_BxCTL register that can be set from 0 to 3:

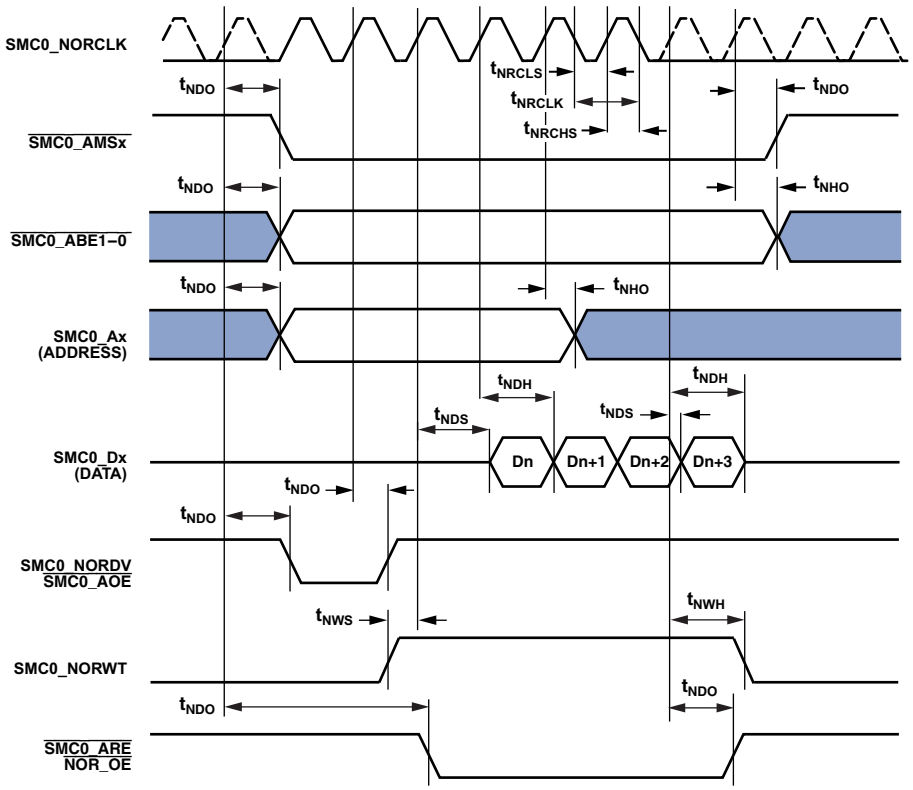
$$f_{NRCLKPROG} = \frac{f_{SCLK0}}{(BCLK + 1)}$$

$$t_{NRCLKPROG} = \frac{1}{f_{NRPCLKPROG}}$$

Table 31. Synchronous Burst AC Timing (BxMODE = b#11)

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
t_{NDS}	DATA-In Setup Before SMC0_NORCLK High	3		ns
t_{NDH}	DATA-In Hold After SMC0_NORCLK High	1.5		ns
t_{NWS}	WAIT-In Setup Before SMC0_NORCLK High	3		ns
t_{NWH}	WAIT-In Hold After SMC0_NORCLK High	1.5		ns
Switching Characteristics				
t_{NRCLS}	NOR_CLK Low Period[1]	$0.5 \times t_{NRCLKPROG} - 1$		ns
t_{NRCHS}	NOR_CLK High Period[1]	$0.5 \times t_{NRCLKPROG} - 1$		ns
t_{NRCLK}	NOR_CLK Period[1]	$t_{NRCLKPROG} - 1$		ns
t_{NDO}	Output Delay After SMC0_NORCLK High[2]		6	ns
t_{NHO}	Output Hold After SMC0_NORCLK High[2]	0.8		ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{NRCLKPROG}$.
[2] Output = SMC0_Ax (address), SMC0_NORDV, $\overline{\text{SMC0\_ARE}}$, $\overline{\text{SMC0\_AMSx}}$ (N0R_CE).



NOTE: SMC0_NORCLK dotted line represents a free running version of SMC0_NORCLK that is not visible on the SMC0_NORCLK pin.

Figure 15. Synchronous Burst AC Interface Timing

Asynchronous Write

Table 32. Asynchronous Memory Write (BxMODE = b#00)

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirement				
$t_{DARDYAWE}$[1]	SMC0_ARDY Valid After $\overline{SMC0\_AWE}$ Low[2]		(WAT – 2.5) × t_{SCLK0} – 17.5	ns
Switching Characteristics				
t_{ENDAT}	DATA Enable After $\overline{SMC0\_AMSx}$ Assertion	–3		ns
t_{DDAT}	DATA Disable After $\overline{SMC0\_AMSx}$ Deassertion		3	ns
t_{AMSAWE}	SMC0_Ax/$\overline{SMC0\_AMSx}$ Assertion Before $\overline{SMC0\_AWE}$ Low[3]	(PREST + WST + PREAT) × t_{SCLK0} – 2		ns
t_{HAWE}	Output[4] Hold After $\overline{SMC0\_AWE}$ High[5]	WHT × t_{SCLK0} – 2		ns
t_{WAWE}[6]	$\overline{SMC0\_AWE}$ Active Low Width[2]	WAT × t_{SCLK0} – 2		ns
$t_{DAWEARDY}$[1]	$\overline{SMC0\_AWE}$ High Delay After SMC0_ARDY Assertion	2.5 × t_{SCLK0}	3.5 × t_{SCLK0} + 17.5	ns

[1] SMC_BxCTL.ARDYEN bit = 1.
[2] WAT value set using the SMC_BxTIM.WAT bits.
[3] PREST, WST, PREAT values set using the SMC_BxETIM.PREST bits, SMC_BxTIM.WST bits, SMC_BxETIM.PREAT bits, and the SMC_BxTIM.RAT bits.
[4] Output signals are DATA, SMC0_Ax, $\overline{SMC0\_AMSx}$, $\overline{SMC0\_ABEx}$.
[5] WHT value set using the SMC_BxTIM.WHT bits.
[6] SMC_BxCTL.ARDYEN bit = 0.

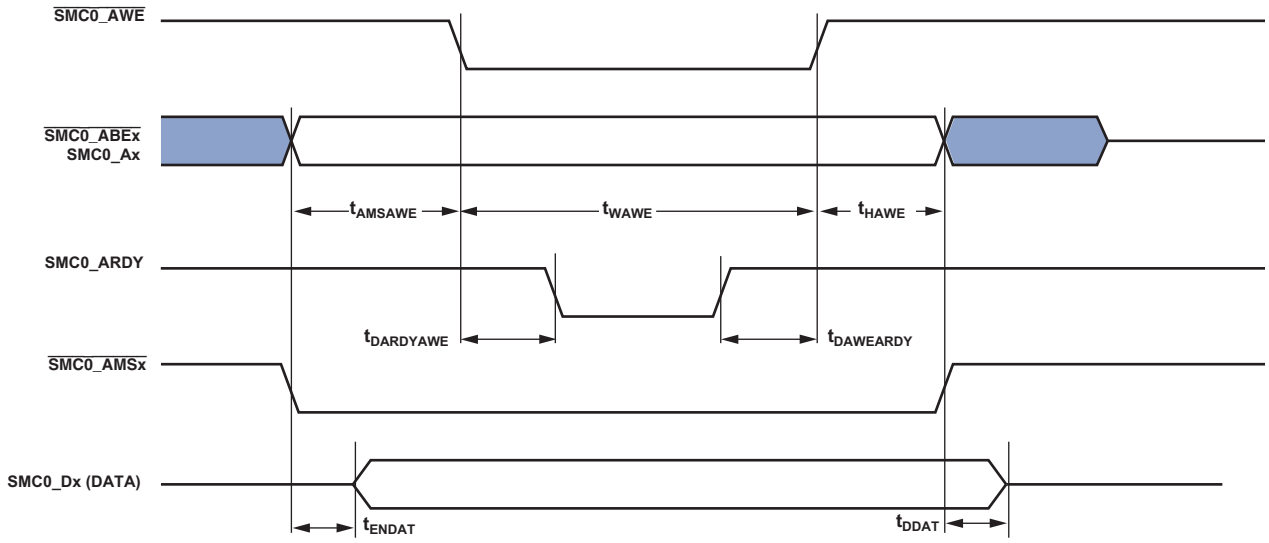


Figure 16. Asynchronous Write

Asynchronous Flash Write

Table 33. Asynchronous Flash Write

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
t_{AMSADV}	SMC0_Ax/$\overline{\text{SMC0\_AMSx}}$ Assertion Before ADV Low[1]	PREST × t_{SCLK0} – 2		ns
$t_{DADVAWE}$	$\overline{\text{SMC0\_AWE}}$ Low Delay From ADV High[2]	PREAT × t_{SCLK0} – 2		ns
t_{WADV}	NR_ADV Active Low Width[3]	WST × t_{SCLK0} – 2		ns
t_{HAWE}	Output[4] Hold After $\overline{\text{SMC0\_AWE}}$ High[5]	WHT × t_{SCLK0} – 2		ns
t_{WAWE}[6]	$\overline{\text{SMC0\_AWE}}$ Active Low Width[7]	WAT × t_{SCLK0} – 2		ns

[1] PREST value set using the SMC_BxETIM.PREST bits.
[2] PREAT value set using the SMC_BxETIM.PREAT bits.
[3] WST value set using the SMC_BxTIM.WST bits.
[4] Output signals are DATA, SMC0_Ax, $\overline{\text{SMC0\_AMSx}}$, $\overline{\text{SMC0\_ABEx}}$.
[5] WHT value set using the SMC_BxTIM.WHT bits.
[6] SMC_BxCTL.ARDYEN bit = 0.
[7] WAT value set using the SMC_BxTIM.WAT bits.

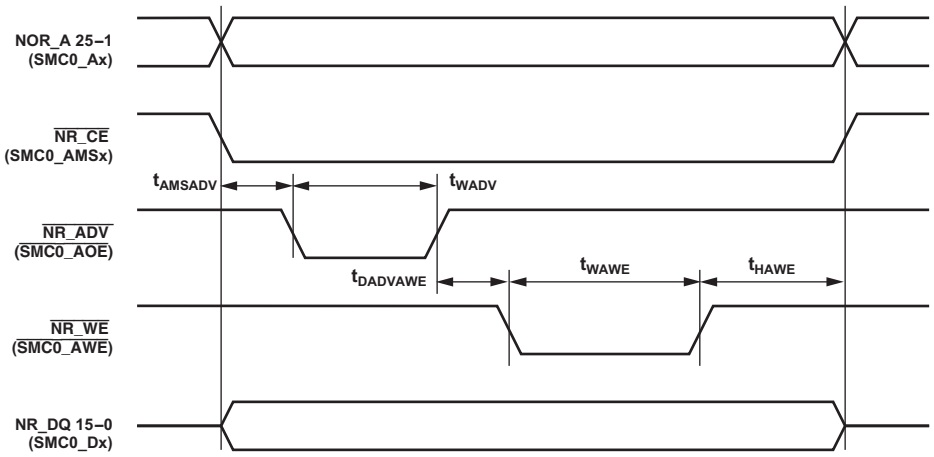


Figure 17. Asynchronous Flash Write

All Accesses

Table 34. All Accesses

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristic				
t_{TURN}	$\overline{\text{SMC0\_AMSx}}$ Inactive Width	(IT + TT) × t_{SCLK0} – 2		ns

Bus Request/Bus Grant

Table 35. Bus Request/Bus Grant

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
t_{DBGBR}	$\overline{SMC0\_BG}$ Delay After $\overline{SMC0\_BR}$	2.5 × t_{SCLK0}	3.5 × t_{SCLK0} + 17.5	ns
t_{ENGDAT}	DATA Enable After $\overline{SMC0\_BG}$ Deassertion	–3		ns
t_{DBGDAT}	DATA Disable After $\overline{SMC0\_BG}$ Assertion		3	ns

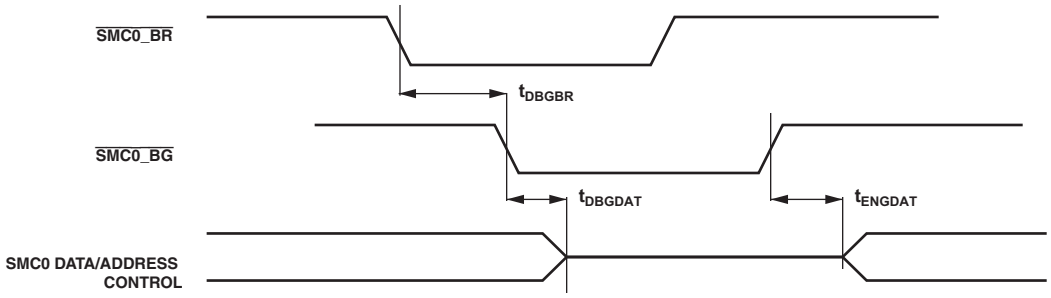
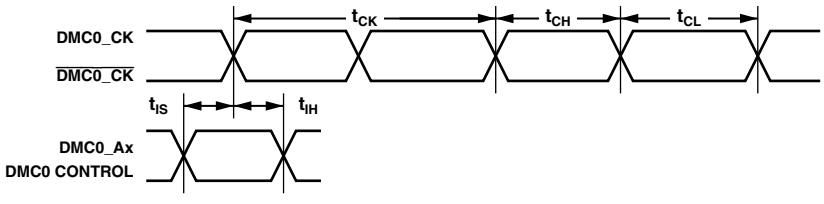


Figure 18. Bus Request/Bus Grant

DDR2 SDRAM Clock and Control Cycle Timing

Table 36. DDR2 SDRAM Read Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		250 MHz Min	Max	Unit
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	4		ns
t_{CH}	Minimum Clock Pulse Width	0.45	0.55	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.45	0.55	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	350		ps
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	475		ps



NOTE: CONTROL = $\overline{DMC0\_CS0}$, DMC0_CKE, $\overline{DMC0\_RAS}$, $\overline{DMC0\_CAS}$, AND $\overline{DMC0\_WE}$.
ADDRESS = DMC0_A00–13, AND DMC0_BA0–1.

Figure 19. DDR2 SDRAM Clock and Control Cycle Timing

DDR2 SDRAM Read Cycle Timing

Table 37. DDR2 SDRAM Read Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		250 MHz[1] Min	Max	Unit
Timing Requirements				
t_{DQSQ}	DMC0_DQS-DMC0_DQ Skew for DMC0_DQS and Associated DMC0_DQ Signals		0.35	ns
t_{QH}	DMC0_DQ, DMC0_DQS Output Hold Time From DMC0_DQS	1.6		ns
t_{RPRE}	Read Preamble	0.9		t_{CK}
t_{RPST}	Read Postamble	0.4		t_{CK}

[1] In order to ensure proper operation of the DDR2, all the DDR2 guidelines have to be strictly followed.

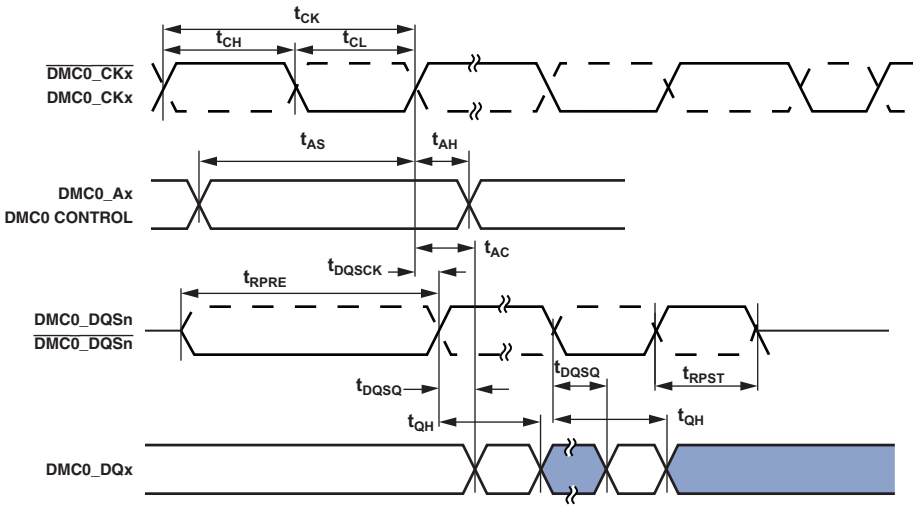


Figure 20. DDR2 SDRAM Controller Input AC Timing

DDR2 SDRAM Write Cycle Timing

Table 38. DDR2 SDRAM Write Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		250 MHz[1] Min	250 MHz[1] Max	Unit
Switching Characteristics				
t_{DQSS}[2]	DMC0_DQS Latching Rising Transitions to Associated Clock Edges	–0.15	0.15	t_{CK}
t_{DS}	Last Data Valid to DMC0_DQS Delay	0.15		ns
t_{DH}	DMC0_DQS to First Data Invalid Delay	0.3		ns
t_{DSS}	DMC0_DQS Falling Edge to Clock Setup Time	0.25		t_{CK}
t_{DSH}	DMC0_DQS Falling Edge Hold Time From DMC0_CK	0.25		t_{CK}
t_{DQSH}	DMC0_DQS Input High Pulse Width	0.35		t_{CK}
t_{DQSL}	DMC0_DQS Input Low Pulse Width	0.35		t_{CK}
t_{WPRE}	Write Preamble	0.35		t_{CK}
t_{WPST}	Write Postamble	0.4		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	0.6		t_{CK}
t_{DIPW}	DMC0_DQ and DMC0_DM Output Pulse Width	0.35		t_{CK}

[1] In order to ensure proper operation of the DDR2, all the DDR2 guidelines have to be strictly followed.
[2] Write command to first DMC0_DQS delay = WL × t_{CK} + t_{DQSS}.

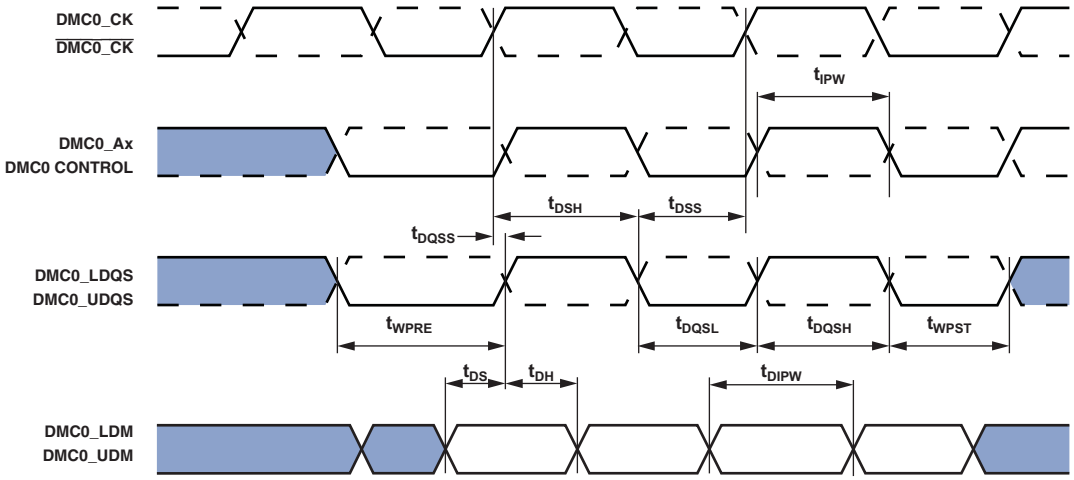
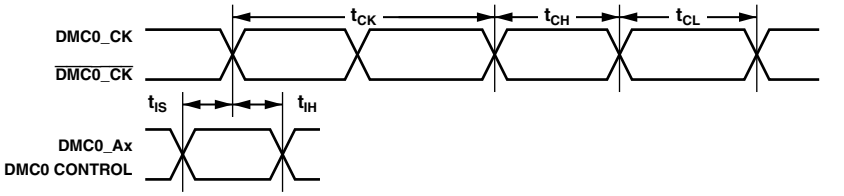


Figure 21. DDR2 SDRAM Controller Output AC Timing

Mobile DDR SDRAM Clock and Control Cycle Timing

Table 39. Mobile DDR SDRAM Read Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		200 MHz Min	200 MHz Max	Unit
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	5		ns
t_{CH}	Minimum Clock Pulse Width	0.45	0.55	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.45	0.55	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	1		ns
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	1		ns



NOTE: CONTROL = DMC0_CS0, DMC0_CKE, DMC0_RAS, DMC0_CAS, AND DMC0_WE.
ADDRESS = DMC0_A00–13, AND DMC0_BA0–1.

Figure 22. Mobile DDR SDRAM Clock and Control Cycle Timing

Mobile DDR SDRAM Read Cycle Timing

Table 40. Mobile DDR SDRAM Read Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		200 MHz Min	200 MHz Max	Unit
Timing Requirements				
t_{QH}	DMC0_DQ, DMC0_DQS Output Hold Time From DMC0_DQS	1.75		ns
t_{DQSQ}	DMC0_DQS-DMC0_DQ Skew for DMC0_DQS and Associated DMC0_DQ Signals		0.4	ns
t_{RPRE}	Read Preamble	0.9	1.1	t_{CK}
t_{RPST}	Read Postamble	0.4	0.6	t_{CK}

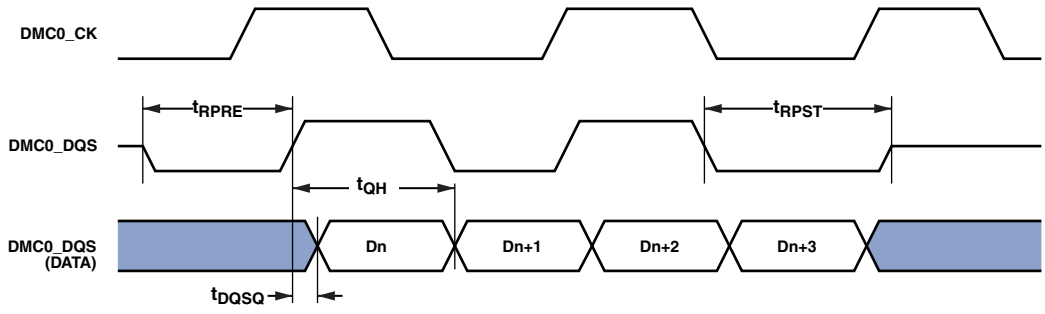


Figure 23. Mobile DDR SDRAM Controller Input AC Timing

Mobile DDR SDRAM Write Cycle Timing

Table 41. Mobile DDR SDRAM Write Cycle Timing, V_{DD_DMC} Nominal 1.8 V

Parameter		200 MHz Min	200 MHz Max	Unit
Switching Characteristics				
t_{DQSS}[1]	DMC0_DQS Latching Rising Transitions to Associated Clock Edges	0.75	1.25	t_{CK}
t_{DS}	Last Data Valid to DMC0_DQS Delay (Slew > 1 V/ns)	0.48		ns
t_{DH}	DMC0_DQS to First Data Invalid Delay (Slew > 1 V/ns)	0.48		ns
t_{DSS}	DMC0_DQS Falling Edge to Clock Setup Time	0.2		t_{CK}
t_{DSH}	DMC0_DQS Falling Edge Hold Time From DMC0_CK	0.2		t_{CK}
t_{DQSH}	DMC0_DQS Input High Pulse Width	0.4		t_{CK}
t_{DQSL}	DMC0_DQS Input Low Pulse Width	0.4		t_{CK}
t_{WPRE}	Write Preamble	0.25		t_{CK}
t_{WPST}	Write Postamble	0.4		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	2.3		ns
t_{DIPW}	DMC0_DQ and DMC0_DM Output Pulse Width	1.8		ns

[1] Write command to first DMC0_DQS delay = WL × t_{CK} + t_{DQSS}.

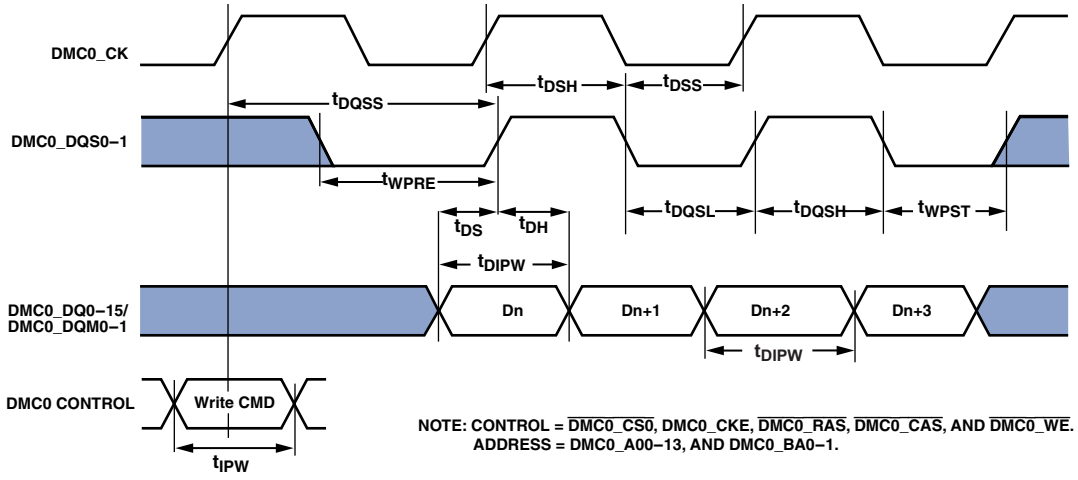


Figure 24. Mobile DDR SDRAM Controller Output AC Timing

Enhanced Parallel Peripheral Interface Timing

The following tables and figures describe enhanced parallel peripheral interface timing operations. The POLC bits in the EPPI_CTL register may be used to set the sampling/driving edges of the EPPI clock.

When internally generated, the programmed PPI clock ($f_{PCLKPROG}$) frequency in MHz is set by the following equation where VALUE is a field in the EPPI_CLKDIV register that can be set from 0 to 65535:

$$f_{PCLKPROG} = \frac{f_{SCLK0}}{(VALUE + 1)}$$

$$t_{PCLKPROG} = \frac{1}{f_{PCLKPROG}}$$

When externally generated the EPPI_CLK is called $f_{PCLKEXT}$:

$$t_{PCLKEXT} = \frac{1}{f_{PCLKEXT}}$$

Table 42. Enhanced Parallel Peripheral Interface—Internal Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSPI}	External FS Setup Before EPPI_CLK	7.9		6.5		ns
t_{HFSPI}	External FS Hold After EPPI_CLK	0		0		ns
t_{SDRPI}	Receive Data Setup Before EPPI_CLK	7.9		6.5		ns
t_{HDRPI}	Receive Data Hold After EPPI_CLK	0		0		ns
t_{SFS3GI}	External FS3 Input Setup Before EPPI_CLK Fall Edge in Clock Gating Mode	15.4		14		ns
t_{HFS3GI}	External FS3 Input Hold Before EPPI_CLK Fall Edge in Clock Gating Mode	0		0		ns
Switching Characteristics						
t_{PCLKW}	EPPI_CLK Width[1]	0.5 × $t_{PCLKPROG}$ – 1		0.5 × $t_{PCLKPROG}$ – 1		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKPROG}$ – 1		$t_{PCLKPROG}$ – 1		ns
t_{DFSPI}	Internal FS Delay After EPPI_CLK		3.5		3.5	ns
t_{HOFSPI}	Internal FS Hold After EPPI_CLK	–0.5		–0.5		ns
t_{DDTPI}	Transmit Data Delay After EPPI_CLK		3.5		3.5	ns
t_{HDTPI}	Transmit Data Hold After EPPI_CLK	–0.5		–0.5		ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{PCLKPROG}$.

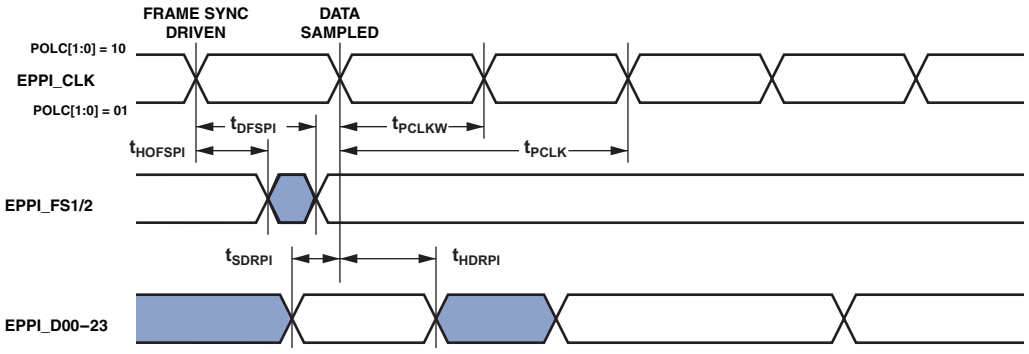


Figure 25. PPI Internal Clock GP Receive Mode with Internal Frame Sync Timing

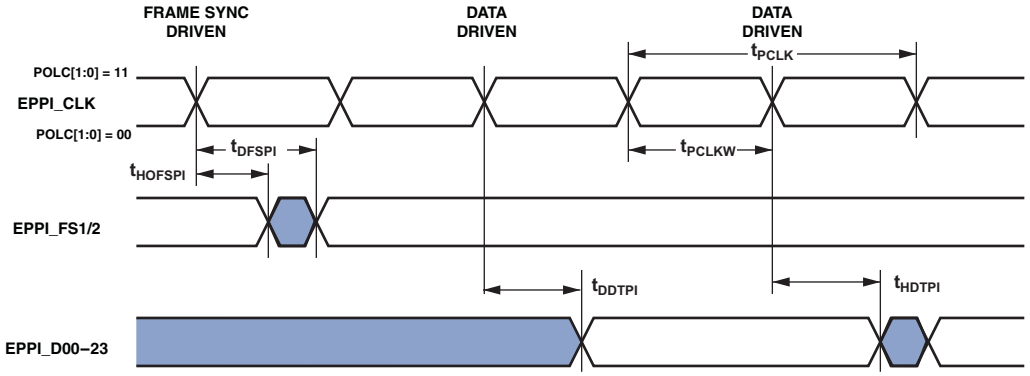


Figure 26. PPI Internal Clock GP Transmit Mode with Internal Frame Sync Timing

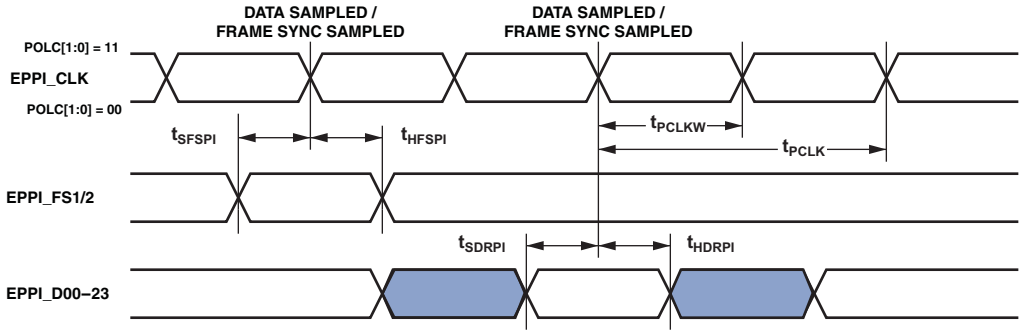


Figure 27. PPI Internal Clock GP Receive Mode with External Frame Sync Timing

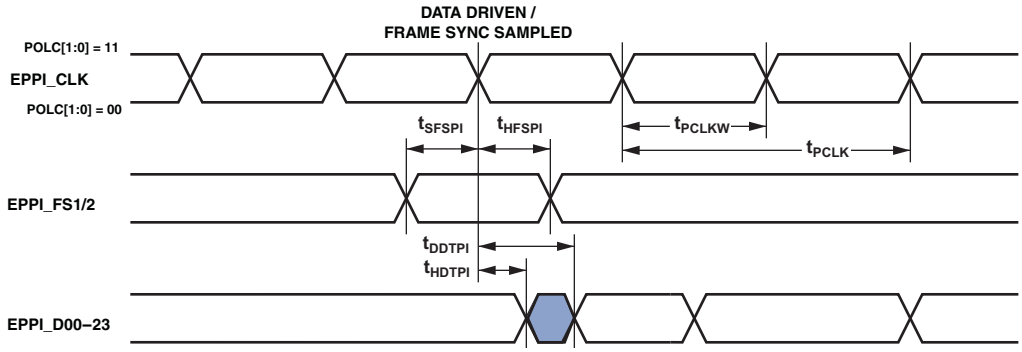


Figure 28. PPI Internal Clock GP Transmit Mode with External Frame Sync Timing

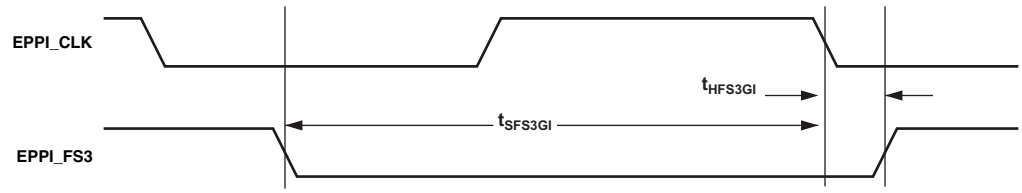


Figure 29. Clock Gating Mode with Internal Clock and External Frame Sync Timing

Table 43. Enhanced Parallel Peripheral Interface—External Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{PCLKW}	EPPI_CLK Width[1]	$(0.5 \times t_{PCLKEXT}) - 1.25$		$(0.5 \times t_{PCLKEXT}) - 1.25$		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKEXT} - 1.25$		$t_{PCLKEXT} - 1.25$		ns
t_{SFSPE}	External FS Setup Before EPPI_CLK	2		2		ns
t_{HFSPE}	External FS Hold After EPPI_CLK	3.7		3.7		ns
t_{SDRPE}	Receive Data Setup Before EPPI_CLK	2		2		ns
t_{HDRPE}	Receive Data Hold After EPPI_CLK	3.7		3.7		ns
Switching Characteristics						
t_{DFSPE}	Internal FS Delay After EPPI_CLK		20.1		15.3	ns
t_{HOFSPE}	Internal FS Hold After EPPI_CLK	2.4		2.4		ns
t_{DDTPE}	Transmit Data Delay After EPPI_CLK		20.1		15.3	ns
t_{HDTPE}	Transmit Data Hold After EPPI_CLK	2.4		2.4		ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external EPPI_CLK. For the external EPPI_CLK ideal maximum frequency see the $f_{PCLKEXT}$ specification in Table 17 on Page 53.

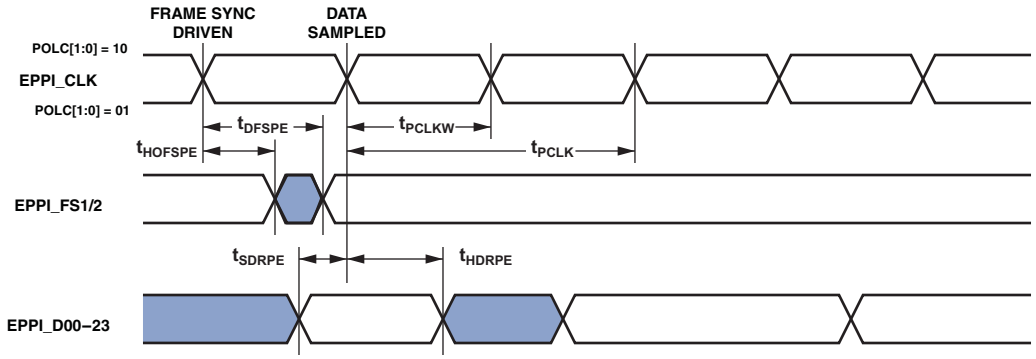


Figure 30. PPI External Clock GP Receive Mode with Internal Frame Sync Timing

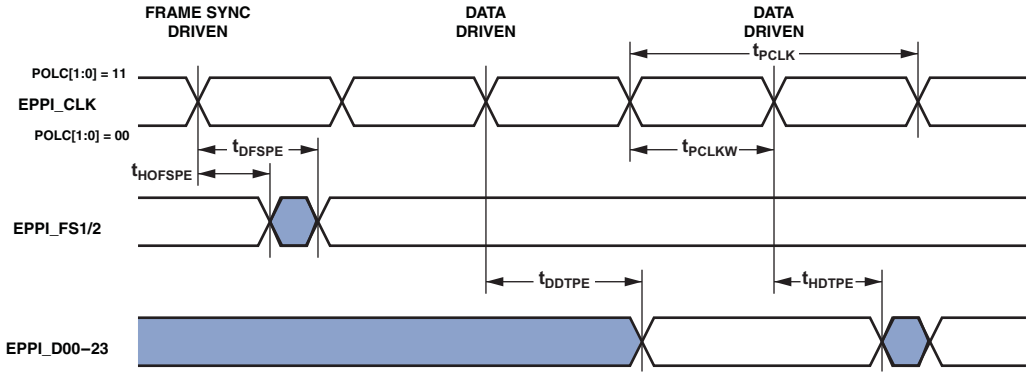


Figure 31. PPI External Clock GP Transmit Mode with Internal Frame Sync Timing

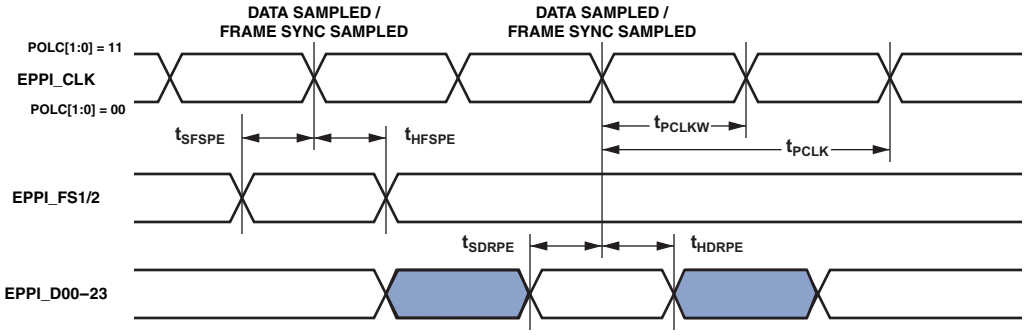


Figure 32. PPI External Clock GP Receive Mode with External Frame Sync Timing

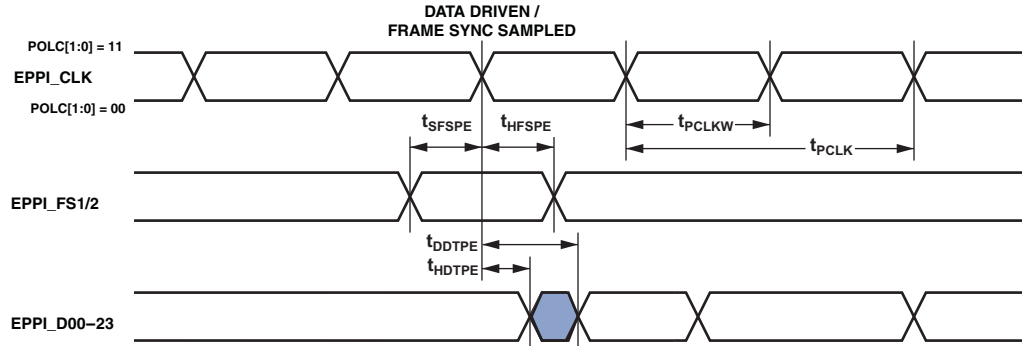


Figure 33. PPI External Clock GP Transmit Mode with External Frame Sync Timing

Link Ports

In link port receive mode the link port clock is supplied externally and is called $f_{LCLKREXT}$:

$$t_{LCLKREXT} = \frac{1}{f_{LCLKREXT}}$$

In link port transmit mode the programmed link port clock ($f_{LCLKTPROG}$) frequency in MHz is set by the following equation where VALUE is a field in the LP_DIV register that can be set from 1 to 255:

$$f_{LCLKTPROG} = \frac{f_{SCLK0}}{(VALUE \times 2)}$$

In the case where VALUE = 0, $f_{LCLKTPROG} = f_{SCLK0}$. For all settings of VALUE the following equation also holds:

$$t_{LCLKTPROG} = \frac{1}{f_{LCLKTPROG}}$$

Calculation of link receiver data setup and hold relative to link clock is required to determine the maximum allowable skew that can be introduced in the transmission path length difference between LP_Dx (data) and LP_CLK. Setup skew is the maximum delay that can be introduced in LP_Dx relative to LP_CLK:
(setup skew = $t_{LCLKTWH}$ min – t_{DLDCH} – t_{SLDCL}). Hold skew is the maximum delay that can be introduced in LP_CLK relative to LP_Dx: (hold skew = $t_{LCLKTWL}$ min – t_{HLDCH} – t_{HLDCL}).

Table 44. Link Ports—Receive

Parameter		V_{DD_EXT} 1.8 V Nominal/3.3 V Nominal Min	Max	Unit
Timing Requirements				
t_{SLDCL}	Data Setup Before LP_CLK Low	2		ns
t_{HLDCL}	Data Hold After LP_CLK Low	3		ns
t_{LCLKIW}	LP_CLK Period[1]	$t_{LCLKREXT}$ – 1.5		ns
$t_{LCLKRWL}$	LP_CLK Width Low[1]	(0.5 × $t_{LCLKREXT}$) – 1.5		ns
$t_{LCLKRWH}$	LP_CLK Width High[1]	(0.5 × $t_{LCLKREXT}$) – 1.5		ns
Switching Characteristic				
t_{DLALC}	LP_ACK Low Delay After LP_CLK Low[2]	1.5 × t_{SCLK0} + 4	2.5 × t_{SCLK0} + 12	ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external LP_CLK. For the external LP_CLK ideal maximum frequency see the $f_{LCLKTEXT}$ specification in Table 17 on Page 53 in Clock Related Operating Conditions.
[2] LP_ACK goes low with t_{DLALC} relative to rise of LP_CLK after first byte, but does not go low if the receiver's link buffer is not about to fill.

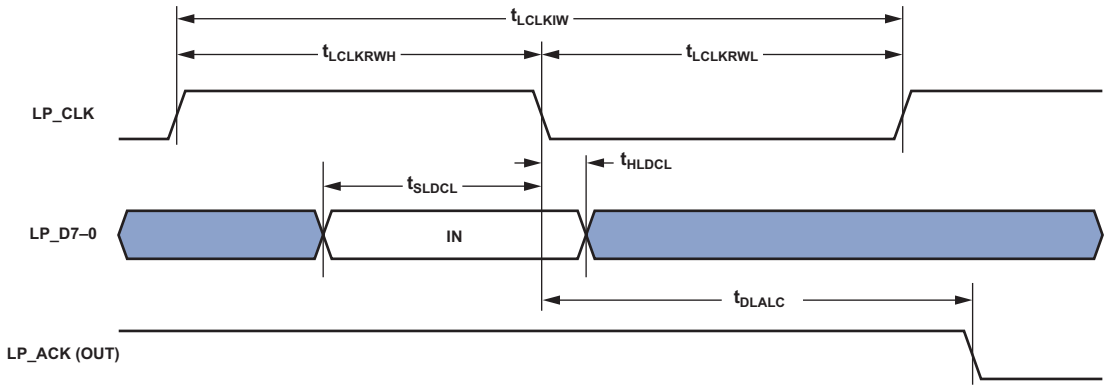
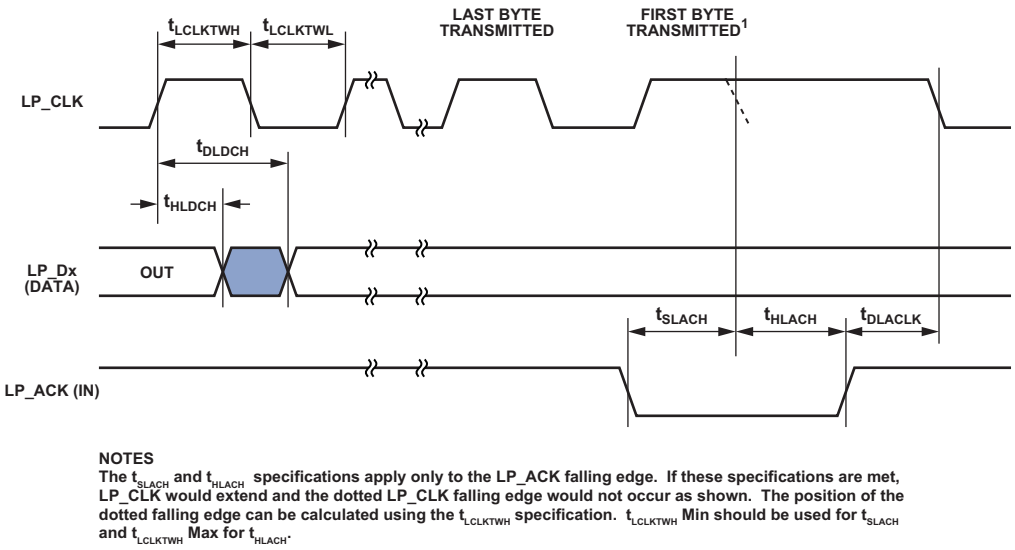


Figure 34. Link Ports—Receive

Table 45. Link Ports—Transmit

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SLACH}	LP_ACK Setup Before LP_CLK Low	$2 \times t_{SCLK0} + 17.5$		$2 \times t_{SCLK0} + 13.5$		ns
t_{HLACH}	LP_ACK Hold After LP_CLK Low	0		0		ns
Switching Characteristics						
t_{DLDCH}	Data Delay After LP_CLK High		2.5		2.5	ns
t_{HLDCH}	Data Hold After LP_CLK High	–1.5		–1.5		ns
$t_{LCLKTWL}$[1]	LP_CLK Width Low	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
$t_{LCLKTWH}$[1]	LP_CLK Width High	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
t_{LCLKTW}[1]	LP_CLK Period	$t_{LCLKTPROG} - 1.2$		$t_{LCLKTPROG} - 1.2$		ns
t_{DLACLK}	LP_CLK Low Delay After LP_ACK High	$t_{SCLK0} + 4$	$(2 \times t_{SCLK0}) + t_{LCLK} + 10$	$t_{SCLK0} + 4$	$(2 \times t_{SCLK0}) + t_{LCLK} + 10$	ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{LCLKTPROG}$.



NOTES
The t_{SLACH} and t_{HLACH} specifications apply only to the LP_ACK falling edge. If these specifications are met, LP_CLK would extend and the dotted LP_CLK falling edge would not occur as shown. The position of the dotted falling edge can be calculated using the $t_{LCLKTWH}$ specification. $t_{LCLKTWH}$ Min should be used for t_{SLACH} and $t_{LCLKTWH}$ Max for t_{HLACH}.

Figure 35. Link Ports—Transmit

Serial Ports

To determine whether communication is possible between two devices at clock speed n, the following specifications must be confirmed: 1) frame sync delay and frame sync setup and hold, 2) data delay and data setup and hold, and 3) serial clock (SPT_CLK) width. In Figure 36 either the rising edge or the falling edge of SPT_CLK (external or internal) can be used as the active sampling edge.

When externally generated the SPORT clock is called $f_{SPTCLKEXT}$:

$$t_{SPTCLKEXT} = \frac{1}{f_{SPTCLKEXT}}$$

When internally generated, the programmed SPORT clock ($f_{SPTCLKPROG}$) frequency in MHz is set by the following equation where CLKDIV is a field in the SPORT_DIV register that can be set from 0 to 65535:

$$f_{SPTCLKPROG} = \frac{f_{SCLK1}}{(CLKDIV + 1)}$$

$$t_{SPTCLKPROG} = \frac{1}{f_{SPTCLKPROG}}$$

Table 46. Serial Ports—External Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSE}	Frame Sync Setup Before SPT_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[1]	2		2		ns
t_{HFSE}	Frame Sync Hold After SPT_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[1]	2.7		2.7		ns
t_{SDRE}	Receive Data Setup Before Receive SPT_CLK[1]	2		2		ns
t_{HDRE}	Receive Data Hold After SPT_CLK[1]	2.7		2.7		ns
t_{SCLKW}	SPT_CLK Width[2]	$(0.5 \times t_{SPTCLKEXT}) - 1.5$		$(0.5 \times t_{SPTCLKEXT}) - 1.5$		ns
t_{SPTCLK}	SPT_CLK Period[2]	$t_{SPTCLKEXT} - 1.5$		$t_{SPTCLKEXT} - 1.5$		ns
Switching Characteristics						
t_{DFSE}	Frame Sync Delay After SPT_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[3]		19.3		14.5	ns
t_{HOFSE}	Frame Sync Hold After SPT_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[3]	2		2		ns
t_{DDTE}	Transmit Data Delay After Transmit SPT_CLK[3]		18.8		14	ns
t_{HDTE}	Transmit Data Hold After Transmit SPT_CLK[3]	2		2		ns

[1] Referenced to sample edge.

[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPT_CLK. For the external SPT_CLK ideal maximum frequency see the $f_{SPTCLKEXT}$ specification in Table 17 on Page 53 in Clock Related Operating Conditions.

[3] Referenced to drive edge.

Table 47. Serial Ports—Internal Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SFSI}	Frame Sync Setup Before SPT_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[1]	16.8		12		ns
t_{HFSI}	Frame Sync Hold After SPT_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[1]	0		–0.5		ns
t_{SDRI}	Receive Data Setup Before SPT_CLK[1]	4.8		3.4		ns
t_{HDRI}	Receive Data Hold After SPT_CLK[1]	1.5		1.5		ns
Switching Characteristics						
t_{DFSI}	Frame Sync Delay After SPT_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[2]		3.5		3.5	ns
t_{HOFSI}	Frame Sync Hold After SPT_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[2]	–1		–1		ns
t_{DDTI}	Transmit Data Delay After SPT_CLK[2]		3.5		3.5	ns
t_{HDTI}	Transmit Data Hold After SPT_CLK[2]	–1		–1		ns
t_{SCLKIW}	SPT_CLK Width[3]	$0.5 \times t_{SPTCLKPROG} - 1.5$		$0.5 \times t_{SPTCLKPROG} - 1.5$		ns
t_{SPTCLK}	SPT_CLK Period[3]	$t_{SPTCLKPROG} - 1.5$		$t_{SPTCLKPROG} - 1.5$		ns

[1] Referenced to the sample edge.
[2] Referenced to drive edge.
[3] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{SPTCLKPROG}$.

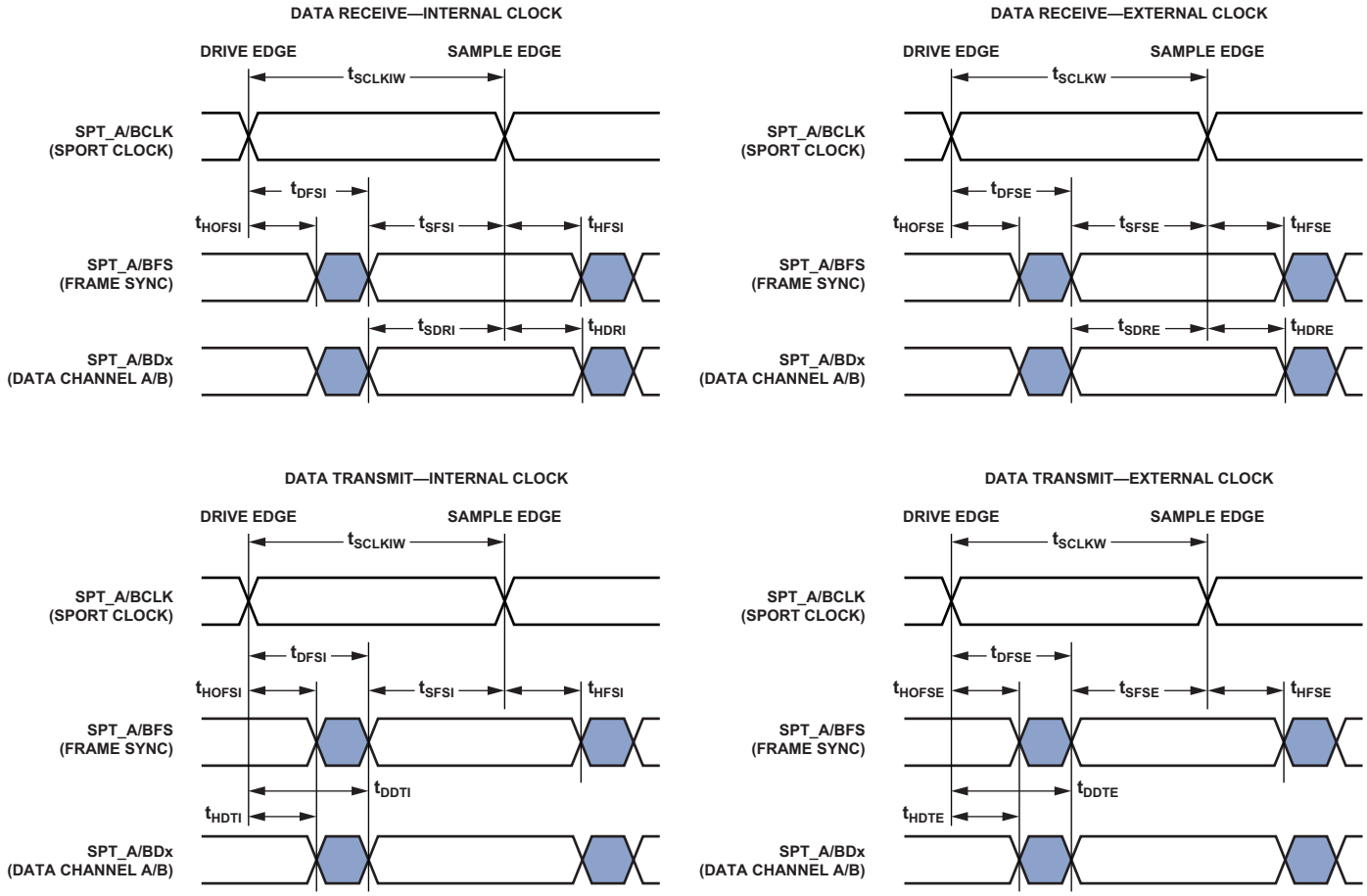


Figure 36. Serial Ports

Table 48. Serial Ports—Enable and Three-State

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Switching Characteristics						
t_{DDTEN}	Data Enable from External Transmit SPT_CLK[1]	1		1		ns
t_{DDTTE}	Data Disable from External Transmit SPT_CLK[1]		18.8		14	ns
t_{DDTIN}	Data Enable from Internal Transmit SPT_CLK[1]	–1		–1		ns
t_{DDTTI}	Data Disable from Internal Transmit SPT_CLK[1]		2.8		2.8	ns

[1] Referenced to drive edge.

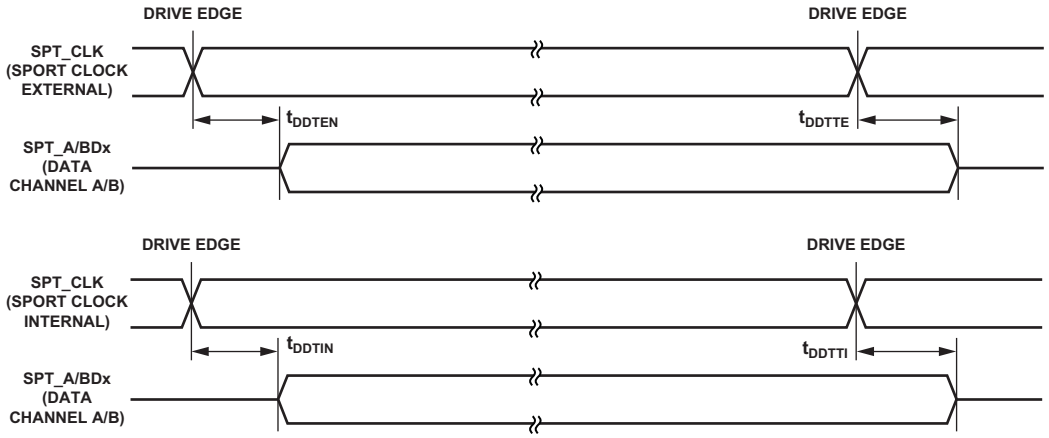


Figure 37. Serial Ports—Enable and Three-State

The SPT_TDV output signal becomes active in SPORT multi-channel mode. During transmit slots (enabled with active channel selection registers) the SPT_TDV is asserted for communication with external devices.

Table 49. Serial Ports—TDV (Transmit Data Valid)

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Switching Characteristics						
t_{DRDVEN}	Data-Valid Enable Delay from Drive Edge of External Clock[1]	2		2		ns
t_{DFDVEN}	Data-Valid Disable Delay from Drive Edge of External Clock[1]		18.8		14	ns
t_{DRDVIN}	Data-Valid Enable Delay from Drive Edge of Internal Clock[1]	–1		–1		ns
t_{DFDVIN}	Data-Valid Disable Delay from Drive Edge of Internal Clock[1]		3.5		3.5	ns

[1] Referenced to drive edge.

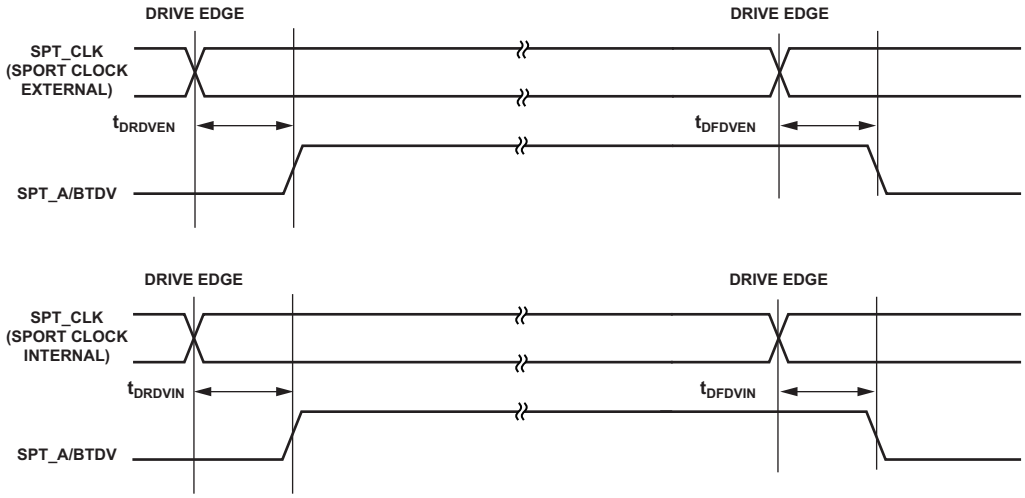


Figure 38. Serial Ports—Transmit Data Valid Internal and External Clock

Table 50. Serial Ports—External Late Frame Sync

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Switching Characteristics						
$t_{DDTLFSE}$	Data Delay from Late External Transmit Frame Sync or External Receive Frame Sync with MCE = 1, MFD = 0[1]		18.8		14	ns
$t_{DDTENFS}$	Data Enable for MCE = 1, MFD = 0[1]	0.5		0.5		ns

[1] The $t_{DDTLFSE}$ and $t_{DDTENFS}$ parameters apply to left-justified as well as standard serial mode, and MCE = 1, MFD = 0.

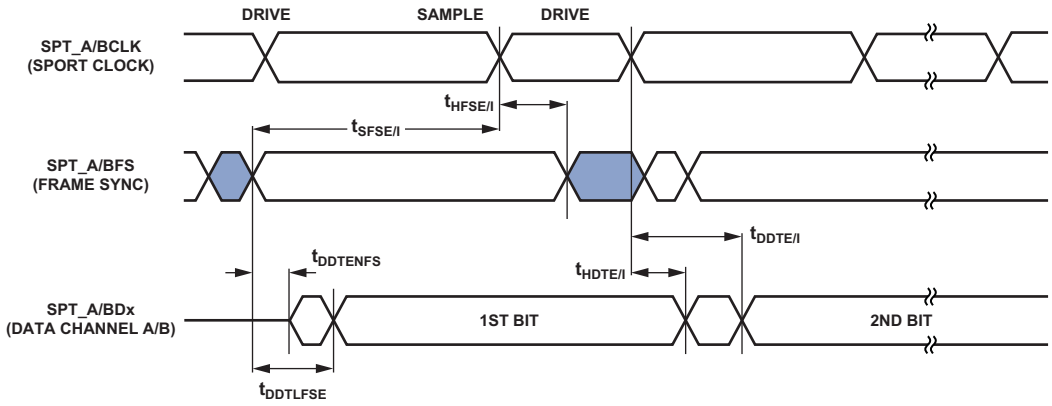


Figure 39. External Late Frame Sync

Serial Peripheral Interface (SPI) Port—Master Timing

Table 51 and Figure 40 describe SPI port master operations.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in MHz is set by the following equation where BAUD is a field in the SPI_CLK register that can be set from 0 to 65535:

$$f_{SPICLKPROG} = \frac{f_{SCLK1}}{(BAUD + 1)}$$

$$t_{SPICLKPROG} = \frac{1}{f_{SPICLKPROG}}$$

Note that:

- In dual mode data transmit the SPI_MISO signal is also an output.
- In quad mode data transmit the SPI_MISO, SPI_D2, and SPI_D3 signals are also outputs.
- In dual mode data receive the SPI_MOSI signal is also an input.
- In quad mode data receive the SPI_MOSI, SPI_D2, and SPI_D3 signals are also inputs.
- To add additional frame delays see the documentation for the SPI_DLY register in the hardware reference manual.

Table 51. Serial Peripheral Interface (SPI) Port—Master Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SSPIDM}	Data Input Valid to SPI_CLK Edge (Data Input Setup)	4.6		3.2		ns
t_{HSPIDM}	SPI_CLK Sampling Edge to Data Input Invalid	1.3		1.3		ns
Switching Characteristics						
t_{SDSCIM}	$\overline{SPI\_SEL}$ low to First SPI_CLK Edge	$0.5 \times t_{SCLK1} - 2$		$0.5 \times t_{SCLK1} - 2$		ns
t_{SPICHM}	SPI_CLK High Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLM}	SPI_CLK Low Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLK}	SPI_CLK Period[1]	$t_{SPICLKPROG} - 1.5$		$t_{SPICLKPROG} - 1.5$		ns
t_{HDSM}	Last SPI_CLK Edge to $\overline{SPI\_SEL}$ High	$(0.5 \times t_{SCLK1}) - 1.5$		$(0.5 \times t_{SCLK1}) - 1.5$		ns
t_{SPITDM}	Sequential Transfer Delay	$t_{SCLK1} - 1.5$		$t_{SCLK1} - 1.5$		ns
$t_{DDSPIDM}$	SPI_CLK Edge to Data Out Valid (Data Out Delay)		2.6		2.6	ns
$t_{HDSPIDM}$	SPI_CLK Edge to Data Out Invalid (Data Out Hold)	–1		–1		ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.

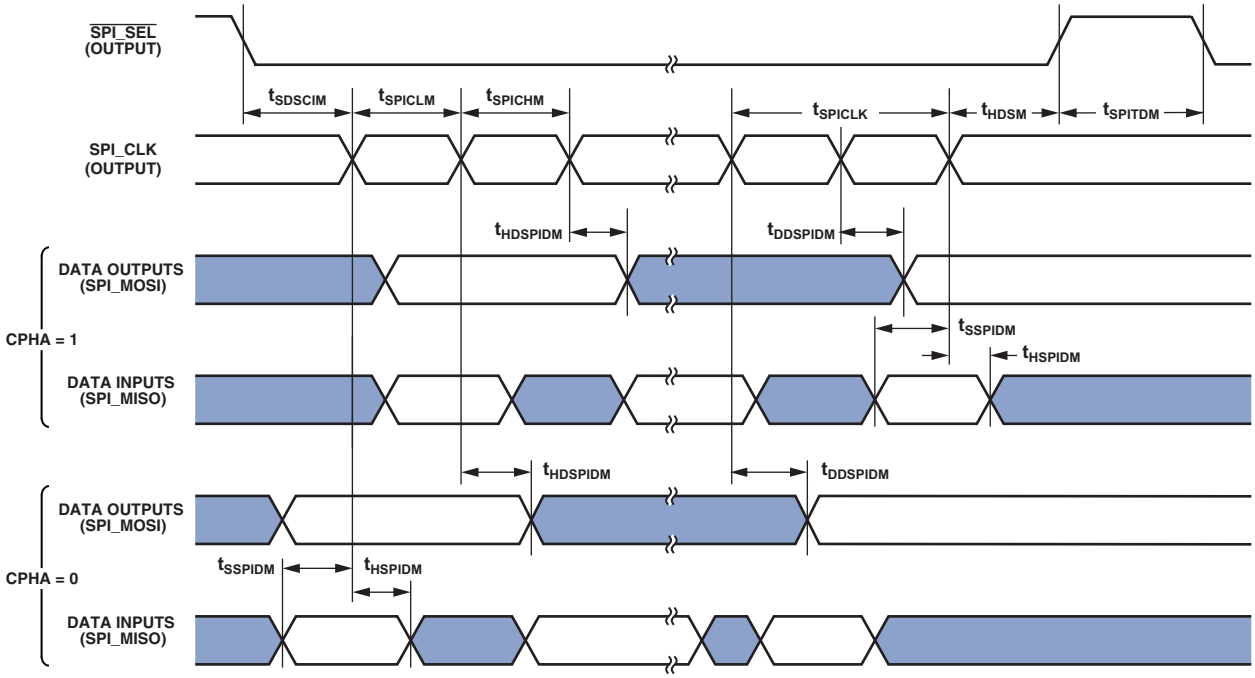


Figure 40. Serial Peripheral Interface (SPI) Port—Master Timing

Serial Peripheral Interface (SPI) Port—Slave Timing

Table 52 and Figure 41 describe SPI port slave operations. Note that:

- In dual mode data transmit the SPI_MOSI signal is also an output.
- In quad mode data transmit the SPI_MOSI, SPI_D2, and SPI_D3 signals are also outputs.
- In dual mode data receive the SPI_MISO signal is also an input.
- In quad mode data receive the SPI_MISO, SPI_D2, and SPI_D3 signals are also inputs.
- In SPI slave mode the SPI clock is supplied externally and is called $f_{SPICLKEXT}$:

$$t_{SPICLKEXT} = \frac{1}{f_{SPICLKEXT}}$$

Table 52. Serial Peripheral Interface (SPI) Port—Slave Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SPICHS}	SPI_CLK High Period[1]	$(0.5 \times t_{SPICLKEXT}) - 1.5$		$(0.5 \times t_{SPICLKEXT}) - 1.5$		ns
t_{SPICLS}	SPI_CLK Low Period[1]	$(0.5 \times t_{SPICLKEXT}) - 1.5$		$(0.5 \times t_{SPICLKEXT}) - 1.5$		ns
t_{SPICLK}	SPI_CLK Period[1]	$t_{SPICLKEXT} - 1.5$		$t_{SPICLKEXT} - 1.5$		ns
t_{HDS}	Last SPI_CLK Edge to $\overline{\text{SPI\_SS}}$ Not Asserted	5		5		ns
t_{SPITDS}	Sequential Transfer Delay	$0.5 \times t_{SPICLK} - 1.5$		$0.5 \times t_{SPICLK} - 1.5$		ns
t_{SDSCI}	$\overline{\text{SPI\_SS}}$ Assertion to First SPI_CLK Edge	11.9		10.5		ns
t_{SSPID}	Data Input Valid to SPI_CLK Edge (Data Input Setup)	2.0		2.0		ns
t_{HSPID}	SPI_CLK Sampling Edge to Data Input Invalid	1.6		1.6		ns
Switching Characteristics						
t_{DSOE}	$\overline{\text{SPI\_SS}}$ Assertion to Data Out Active	0	18.8	0	14	ns
t_{DSDHI}	$\overline{\text{SPI\_SS}}$ Deassertion to Data High Impedance	0	16.3	0	12.5	ns
t_{DDSPID}	SPI_CLK Edge to Data Out Valid (Data Out Delay)		18.8		14	ns
t_{HDSPID}	SPI_CLK Edge to Data Out Invalid (Data Out Hold)	1.5		1.5		ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPI_CLK. For the external SPI_CLK ideal maximum frequency see the $f_{SPICLKTEXT}$ specification in the Clock Related Operating Conditions table on Page 53.

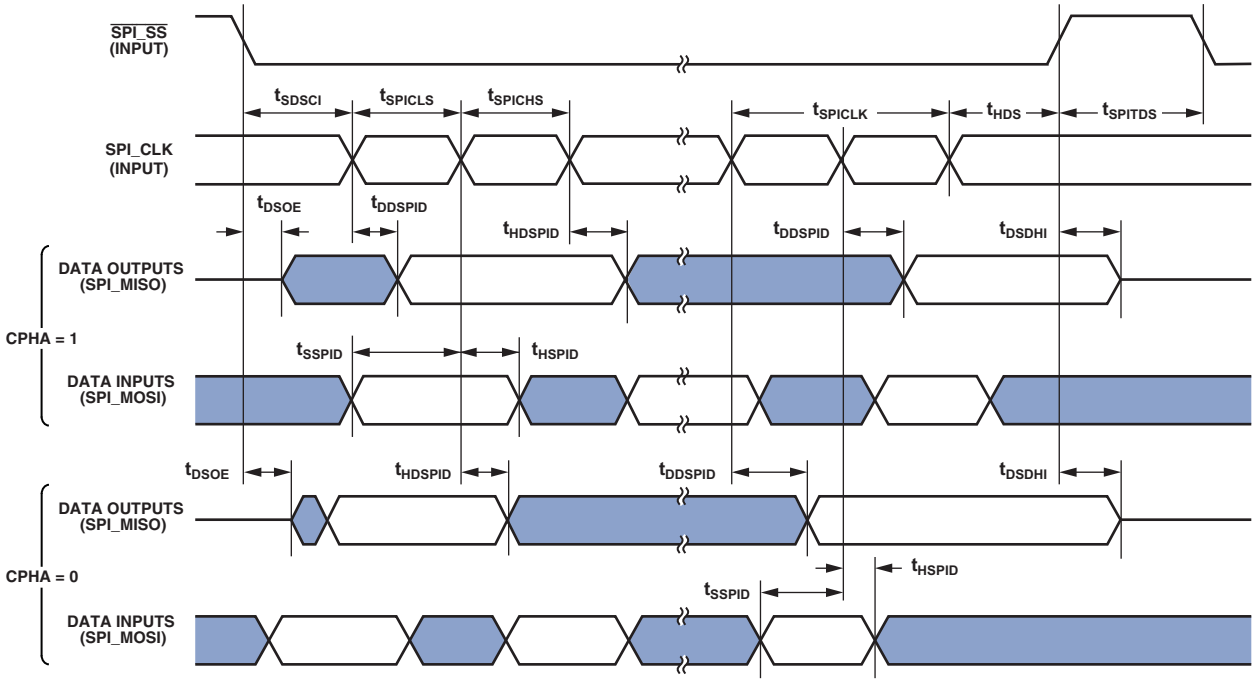


Figure 41. Serial Peripheral Interface (SPI) Port—Slave Timing

Serial Peripheral Interface (SPI) Port—SPI_RDY Slave Timing

Table 53. SPI Port—SPI_RDY Slave Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
$t_{DSPISCKRDYSR}$	SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Receive	$2.5 \times t_{SCLK1}$	$3.5 \times t_{SCLK1} + 17.5$	ns
$t_{DSPISCKRDYST}$	SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Transmit	$3.5 \times t_{SCLK1}$	$4.5 \times t_{SCLK1} + 17.5$	ns

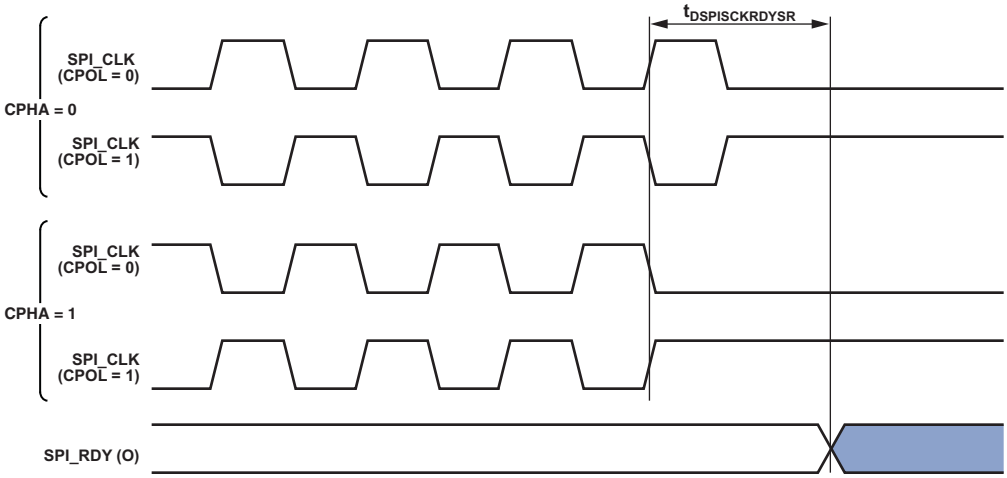


Figure 42. SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Receive (FCCH = 0)

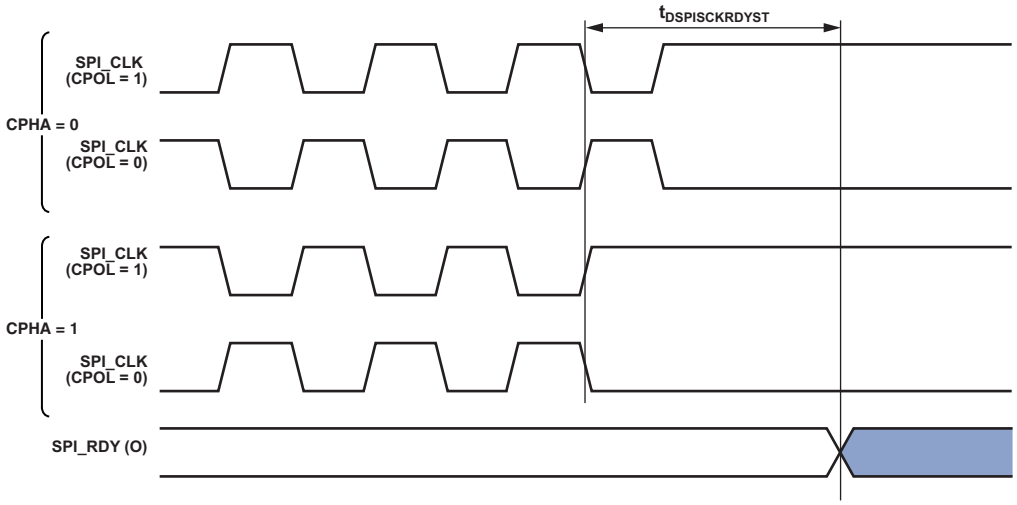


Figure 43. SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Transmit (FCCH = 1)

Serial Peripheral Interface (SPI) Port—Open Drain Mode Timing

In Figure 44 and Figure 45, the outputs can be SPI_MOSI SPI_MISO, SPI_D2, and/or SPI_D3 depending on the mode of operation.

Table 54. SPI Port ODM Master Mode Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
$t_{HDSPIODMM}$	SPI_CLK Edge to High Impedance from Data Out Valid	–1		ns
$t_{DDSPIODMM}$	SPI_CLK Edge to Data Out Valid from High Impedance	0	6	ns

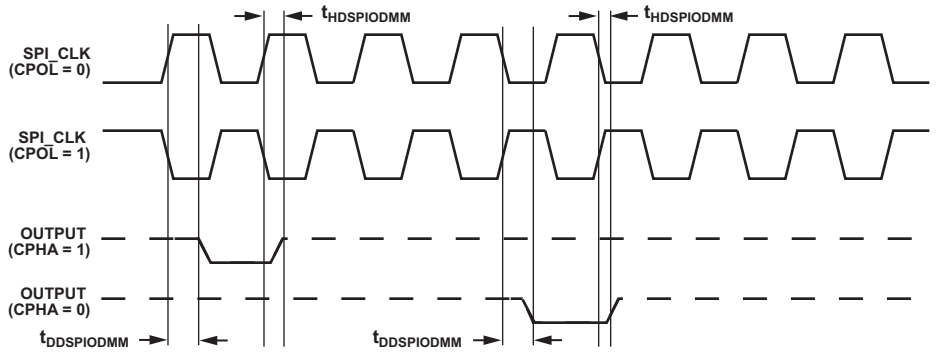


Figure 44. ODM Master

Table 55. SPI Port—ODM Slave Mode

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
$t_{HDSPIODMS}$	SPI_CLK Edge to High Impedance from Data Out Valid	0		ns
$t_{DDSPIODMS}$	SPI_CLK Edge to Data Out Valid from High Impedance		11.5	ns

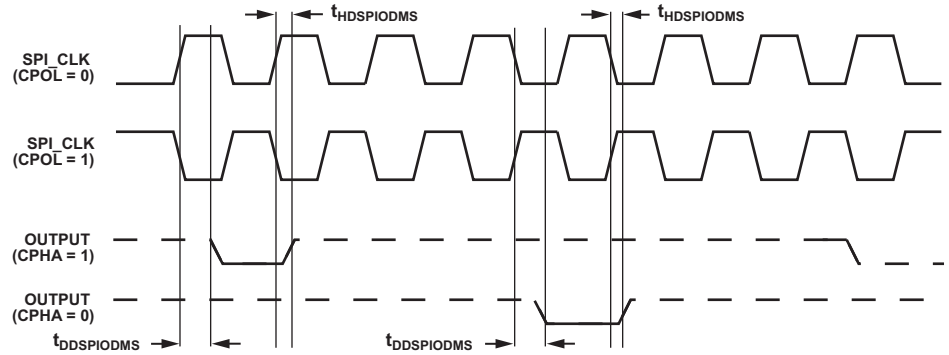


Figure 45. ODM Slave

Serial Peripheral Interface (SPI) Port—SPI_RDY Timing

SPI_RDY is used to provide flow control. The CPOL and CPHA bits are set in SPI_CTL, while LEADX, LAGX, and STOP are in SPI_DLY.

Table 56. SPI Port—SPI_RDY Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
$t_{SRDYSCKM0}$	Minimum Setup Time for SPI_RDY De-assertion in Master Mode Before Last SPI_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 0	$(2.5 + 1.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$		ns
$t_{SRDYSCKM1}$	Minimum Setup Time for SPI_RDY De-assertion in Master Mode Before Last SPI_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 1	$(1.5 + BAUD^{1}) \times t_{SCLK1} + 17.5$		ns
Switching Characteristic				
$t_{SRDYSCKM}$	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 0 and BAUD = 0 (STOP, LEADX, LAGX = 0)	$3 \times t_{SCLK1}$	$4 \times t_{SCLK1} + 17.5$	ns
	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 0 and BAUD ≥ 1 (STOP, LEADX, LAGX = 0)	$(4 + 1.5 \times BAUD^{1}) \times t_{SCLK1}$	$(5 + 1.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$	ns
	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 1 (STOP, LEADX, LAGX = 0)	$(3 + 0.5 \times BAUD^{1}) \times t_{SCLK1}$	$(4 + 0.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$	ns

[1] BAUD value set using the SPI_CLK.BAUD bits.

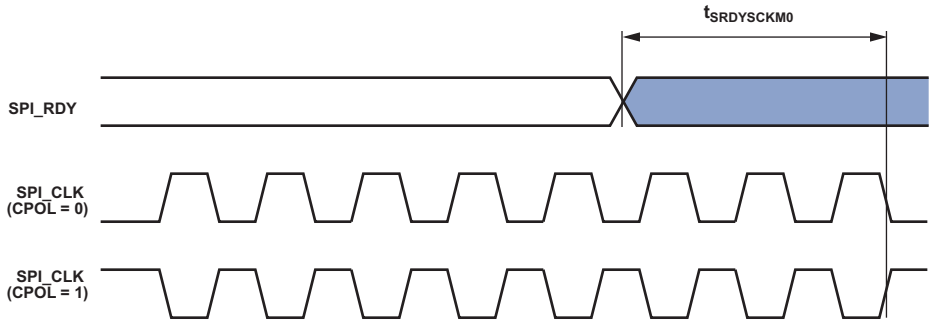


Figure 46. SPI_RDY Setup Before SPI_CLK with CPHA = 0

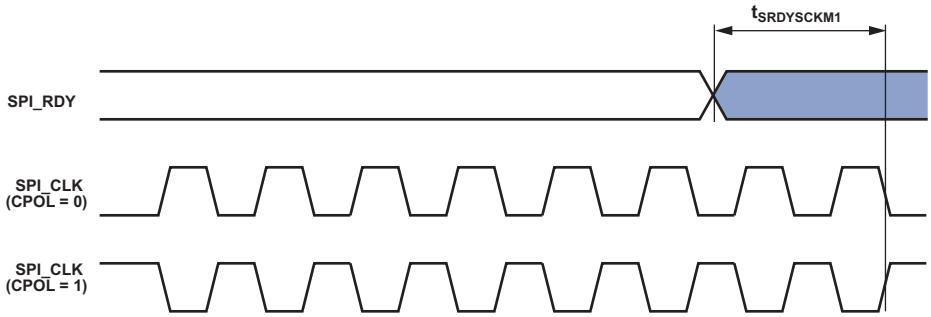


Figure 47. SPI_RDY Setup Before SPI_CLK with CPHA = 1

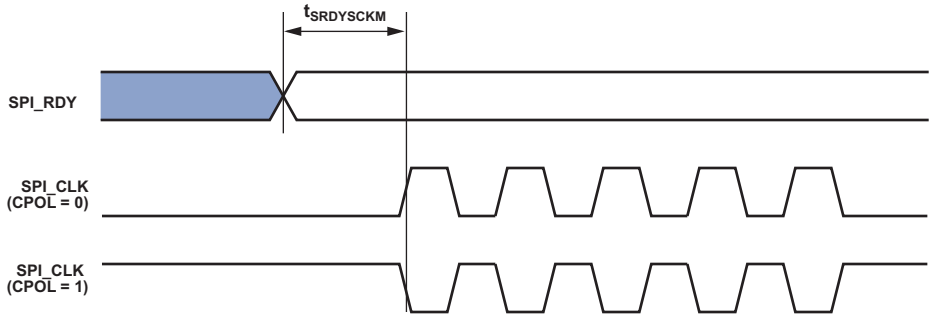


Figure 48. SPI_CLK Switching Diagram after SPI_RDY Assertion, CPHA = x

General-Purpose Port Timing

Table 57 and Figure 49 describe general-purpose port operations.

Table 57. General-Purpose Port Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirement				
t_{WFI}	General-Purpose Port Pin Input Pulse Width	$2 \times t_{SCLK0} - 1.5$		ns

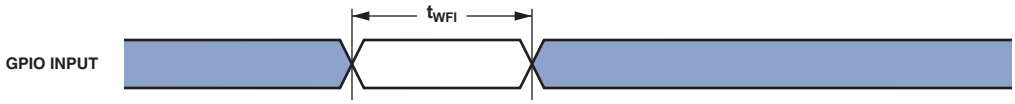


Figure 49. General-Purpose Port Timing

Timer Cycle Timing

Table 58 and Figure 50 describe timer expired operations. The input signal is asynchronous in "width capture mode" and "external clock mode" and has an ideal maximum input frequency of (f_{SCLK0}/4) MHz. The Period Value (VALUE) is the timer period assigned in the TMx_TMRn_PER register and can range from 2 to $2^{32} - 1$.

Table 58. Timer Cycle Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	Max	V_{DD_EXT} 3.3 V Nominal Min	Max	Unit
Timing Requirements						
t_{WL}	Timer Pulse Width Input Low[1]	$2 \times t_{SCLK0} - 1.5$		$2 \times t_{SCLK0} - 1.5$		ns
t_{WH}	Timer Pulse Width Input High[1]	$2 \times t_{SCLK0} - 1.5$		$2 \times t_{SCLK0} - 1.5$		ns
Switching Characteristics						
t_{HTO}	Timer Pulse Width Output	$t_{SCLK0} \times VALUE - 1.5$		$t_{SCLK0} \times VALUE - 1.5$		ns

[1] This specification indicates the minimum instantaneous width that can be tolerated due to duty cycle variation or jitter for TMx signals in width capture and external clock modes. The ideal maximum frequency for TMx signals is listed in Timer Cycle Timing on this page.

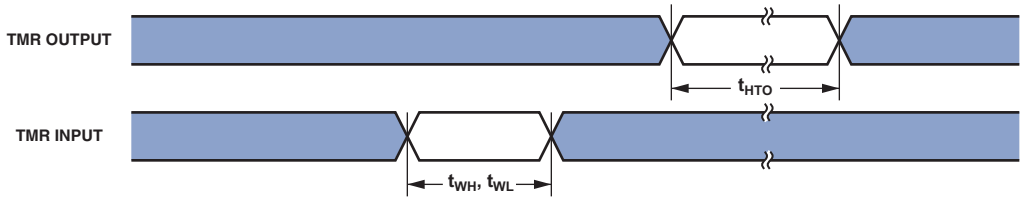


Figure 50. Timer Cycle Timing

Up/Down Counter/Rotary Encoder Timing

Table 59. Up/Down Counter/Rotary Encoder Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirement						
t_{WCOUNT}	Up/Down Counter/Rotary Encoder Input Pulse Width	$2 \times t_{SCLK0}$		$2 \times t_{SCLK0}$		ns

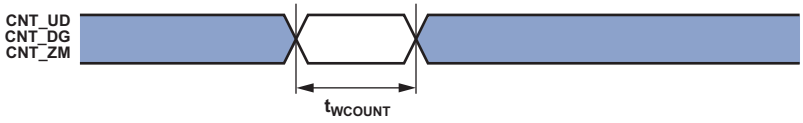


Figure 51. Up/Down Counter/Rotary Encoder Timing

Pulse Width Modulator (PWM) Timing

Table 60 and Figure 52 describe PWM operations.

Table 60. PWM Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirement						
t_{ES}	External Sync Pulse Width			$2 \times t_{SCLK0}$		ns
Switching Characteristics						
t_{DODIS}	Output Inactive (OFF) After Trip Input[1]				15	ns
t_{DOE}	Output Delay After External Sync[1, 2]			$2 \times t_{SCLK0} + 5.5$	$5 \times t_{SCLK0} + 14$	ns

[1] PWM outputs are: PWMx_AH, PWMx_AL, PWMx_BH, PWMx_BL, PWMx_CH, and PWMx_CL.

[2] When the external sync signal is synchronous to the peripheral clock, it takes fewer clock cycles for the output to appear compared to when the external sync signal is asynchronous to the peripheral clock. For more information, see the *ADSP-BF60x Blackfin Processor Hardware Reference*.

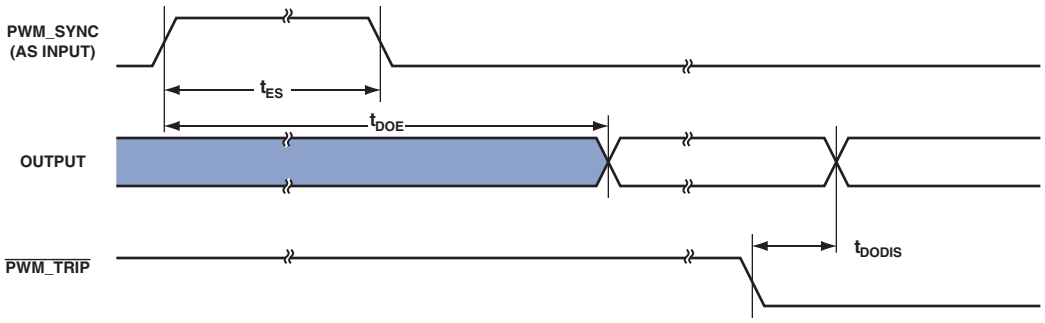


Figure 52. PWM Timing

ADC Controller Module (ACM) Timing

Table 61 and Figure 53 describe ACM operations.

When internally generated, the programmed ACM clock ($f_{ACLKPROG}$) frequency in MHz is set by the following equation where CKDIV is a field in the ACM_TC0 register and ranges from 1 to 255:

$$f_{ACLKPROG} = \frac{f_{SCLK1}}{CKDIV + 1}$$

$$t_{ACLKPROG} = \frac{1}{f_{ACLKPROG}}$$

Setup cycles (SC) in Table 61 is also a field in the ACM_TC0 register and ranges from 0 to 4095. Hold Cycles (HC) is a field in the ACM_TC1 register that ranges from 0 to 15.

Table 61. ACM Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
t_{SDR}	SPORT DRxPRI/DRxSEC Setup Before ACMx_CLK	3		ns
t_{HDR}	SPORT DRxPRI/DRxSEC Hold After ACMx_CLK	1.5		ns
Switching Characteristics				
t_{SCTLCS}	ACM Controls (ACMx_A[4:0]) Setup Before Assertion of $\overline{CS}$	(SC + 1) × t_{SCLK1} – 3		ns
t_{HCTLCS}	ACM Control (ACMx_A[4:0]) Hold After De-assertion of $\overline{CS}$	HC × t_{ACLK} + 0.1		ns
t_{ACLKW}	ACM Clock Pulse Width[1]	(0.5 × $t_{ACLKPROG}$) – 1.5		ns
t_{ACLK}	ACM Clock Period[1]	$t_{ACLKPROG}$ – 1.5		ns
$t_{HCSACLK}$	$\overline{CS}$ Hold to ACMx_CLK Edge	–0.1		ns
$t_{SCSACLK}$	$\overline{CS}$ Setup to ACMx_CLK Edge	t_{ACLK} – 3.5		ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{ACLKPROG}$.

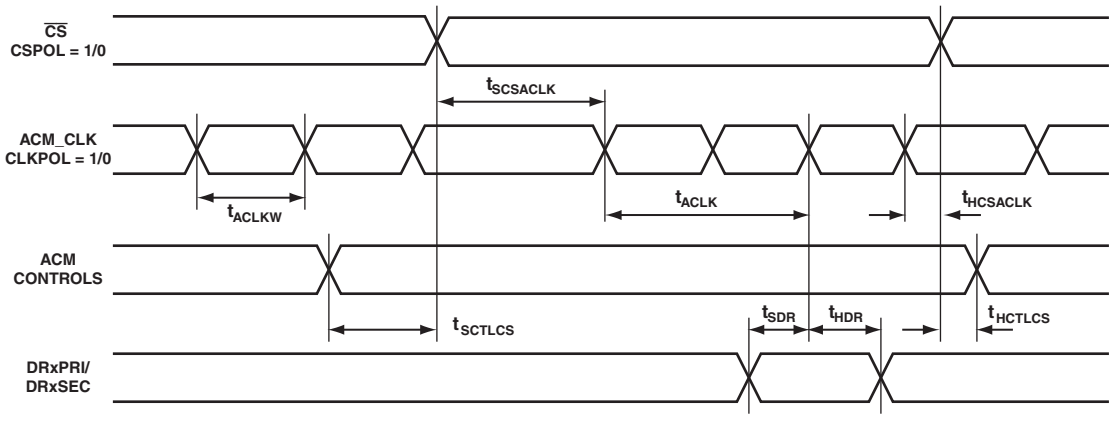


Figure 53. ACM Timing

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

The UART ports receive and transmit operations are described in the *ADSP-BF60x Blackfin Processor Hardware Reference Manual*.

CAN Interface

The CAN interface timing is described in the *ADSP-BF60x Blackfin Processor Hardware Reference Manual*.

Universal Serial Bus (USB) On-The-Go—Receive and Transmit Timing

Table 62 describes the USB On-The-Go receive and transmit operations.

Table 62. USB On-The-Go—Receive and Transmit Timing

Parameter		V_{DD_USB} 3.3 V Nominal Min	Max	Unit
Timing Requirements				
f_{USBS}	USB_XI Frequency	48	48	MHz
fs_{USB}	USB_XI Clock Frequency Stability	–50	+50	ppm

RSI Controller Timing

Table 63 and Figure 54 describe RSI controller timing.

Table 63. RSI Controller Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{ISU}	Input Setup Time	11		9.6		ns
t_{IH}	Input Hold Time	2		2		ns
Switching Characteristics						
f_{PP}	Clock Frequency Data Transfer Mode[1]		41.67		41.67	MHz
t_{WL}	Clock Low Time	8		8		ns
t_{WH}	Clock High Time	8		8		ns
t_{TLH}	Clock Rise Time		3		3	ns
t_{THL}	Clock Fall Time		3		3	ns
t_{ODLY}	Output Delay Time During Data Transfer Mode		2.5		2.5	ns
t_{OH}	Output Hold Time	–1		–1		ns

[1] $t_{PP} = 1/f_{PP}$

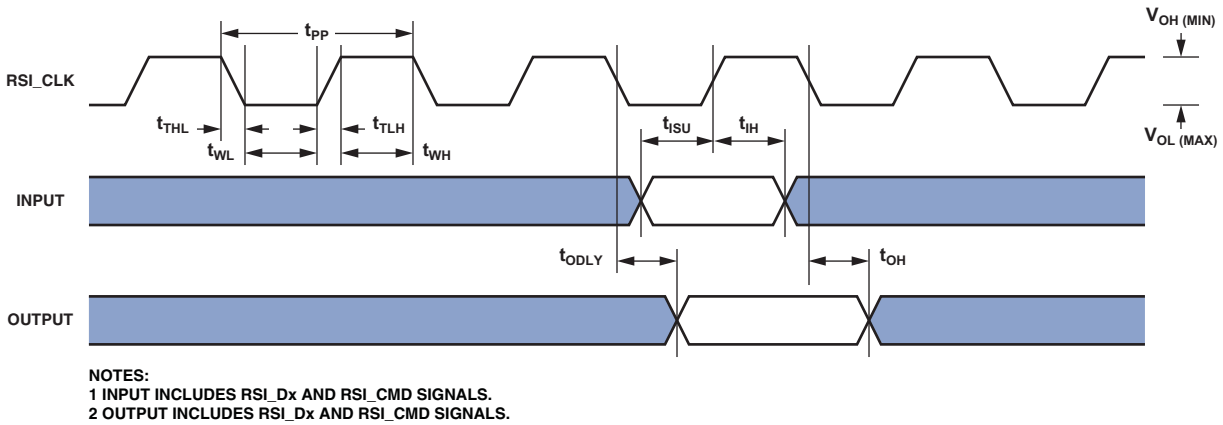


Figure 54. RSI Controller Timing

10/100 Ethernet MAC Controller Timing

Table 64 through Table 66 and Figure 55 through Figure 57 describe the 10/100 Ethernet MAC Controller operations.

Table 64. 10/100 Ethernet MAC Controller Timing: RMII Receive Signal

Parameter[1]		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
$t_{REFCLKF}$	ETHx_REFCLK Frequency (f_{SCLK0} = SCLK0 Frequency)	None	50 + 1%	MHz
$t_{REFCLKW}$	ETHx_REFCLK Width (t_{REFCLK} = ETHx_REFCLK Period)	$t_{REFCLK} \times 35\%$	$t_{REFCLK} \times 65\%$	ns
$t_{REFCLKIS}$	Rx Input Valid to RMII ETHx_REFCLK Rising Edge (Data In Setup)	4		ns
$t_{REFCLKIH}$	RMII ETHx_REFCLK Rising Edge to Rx Input Invalid (Data In Hold)	2.2		ns

[1] RMII inputs synchronous to RMII REF_CLK are ERxD1–0, RMII CRS_DV, and ERxER.

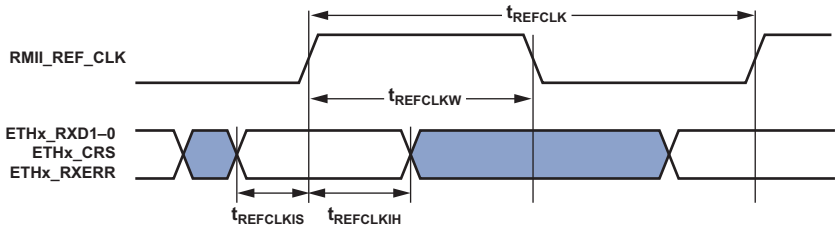


Figure 55. 10/100 Ethernet MAC Controller Timing: RMII Receive Signal

Table 65. 10/100 Ethernet MAC Controller Timing: RMII Transmit Signal

Parameter[1]		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
$t_{REFCLKOV}$	RMII ETHx_REFCLK Rising Edge to Transmit Output Valid (Data Out Valid)		14	ns
$t_{REFCLKOH}$	RMII ETHx_REFCLK Rising Edge to Transmit Output Invalid (Data Out Hold)	2		ns

[1] RMII outputs synchronous to RMII REF_CLK are ETxD1–0.

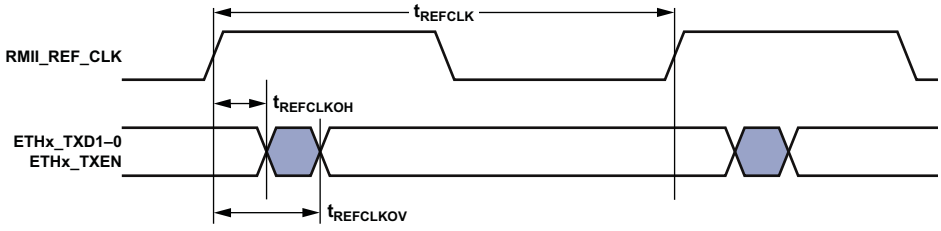


Figure 56. 10/100 Ethernet MAC Controller Timing: RMII Transmit Signal

Table 66. 10/100 Ethernet MAC Controller Timing: RMII Station Management

Parameter[1]		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
t_{MDIOS}	ETHx_MDIO Input Valid to ETHx_MDC Rising Edge (Setup)	14		ns
t_{MDCIH}	ETHx_MDC Rising Edge to ETHx_MDIO Input Invalid (Hold)	0		ns
Switching Characteristics				
t_{MDCOV}	ETHx_MDC Falling Edge to ETHx_MDIO Output Valid		t_{SCLK0} + 5	ns
t_{MDCOH}	ETHx_MDC Falling Edge to ETHx_MDIO Output Invalid (Hold)	t_{SCLK0} −1		ns

[1] ETHx_MDC/ETHx_MDIO is a 2-wire serial bidirectional port for controlling one or more external PHYs. ETHx_MDC is an output clock whose minimum period is programmable as a multiple of the system clock SCLK0. ETHx_MDIO is a bidirectional data line.

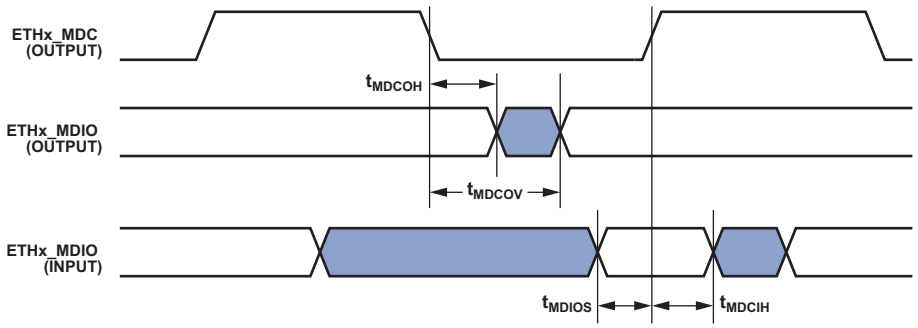


Figure 57. 10/100 Ethernet MAC Controller Timing: RMII Station Management

JTAG Test And Emulation Port Timing

Table 67 and Figure 58 describe JTAG port operations.

Table 67. JTAG Port Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{TCK}	JTG_TCK Period	20		20		ns
t_{STAP}	JTG_TDI, JTG_TMS Setup Before JTG_TCK High	4		4		ns
t_{HTAP}	JTG_TDI, JTG_TMS Hold After JTG_TCK High	4		4		ns
t_{SSYS}	System Inputs Setup Before JTG_TCK High[1]	12		12		ns
t_{HSYS}	System Inputs Hold After JTG_TCK High[1]	5		5		ns
t_{TRSTW}	$\overline{\text{JTG\_TRST}}$ Pulse Width (measured in JTG_TCK cycles)[2]	4		4		T_{CK}
Switching Characteristics						
t_{DTDO}	JTG_TDO Delay from JTG_TCK Low		18		13.5	ns
t_{DSYS}	System Outputs Delay After JTG_TCK Low[3]		22		17	ns

[1] System Inputs = DMC0_DQ00–15, DMC0_LDQS, $\overline{\text{DMC0\_LDQS}}$, DMC0_UDQS, $\overline{\text{DMC0\_UDQS}}$, PA_15–0, PB_15–0, PC_15–0, PD_15–0, PE_15–0, PF_15–0, PG_15–0, SMC0_ARDY_NORWT, $\overline{\text{SMC0\_BR}}$, SMC0_D15–0, SYS_BMODE0–2, $\overline{\text{SYS\_HWRST}}$, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_NMI\_RESOUT}}$, SYS_PWRGD, TWI0_SCL, TWI0_SDA, TWI1_SCL, TWI1_SDA.

[2] 50 MHz Maximum.

[3] System Outputs = DMC0_A00–13, DMC0_BA0–2, $\overline{\text{DMC0\_CAS}}$, DMC0_CK, $\overline{\text{DMC0\_CK}}$, DMC0_CKE, $\overline{\text{DMC0\_CS0}}$, DMC0_DQ00–15, DMC0_LDM, DMC0_LDQS, $\overline{\text{DMC0\_LDQS}}$, DMC0_ODT, $\overline{\text{DMC0\_RAS}}$, DMC0_UDM, DMC0_UDQS, $\overline{\text{DMC0\_UDQS}}$, $\overline{\text{DMC0\_WE}}$, $\overline{\text{JTG\_EMU}}$, PA_15–0, PB_15–0, PC_15–0, PD_15–0, PE_15–0, PF_15–0, PG_15–0, $\overline{\text{SMC0\_AMS0}}$, $\overline{\text{SMC0\_AOE\_NORDV}}$, $\overline{\text{SMC0\_ARE}}$, $\overline{\text{SMC0\_AWE}}$, SMC0_A01, SMC0_A02, SMC0_D15–0, SYS_CLKOUT, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_NMI\_RESOUT}}$.

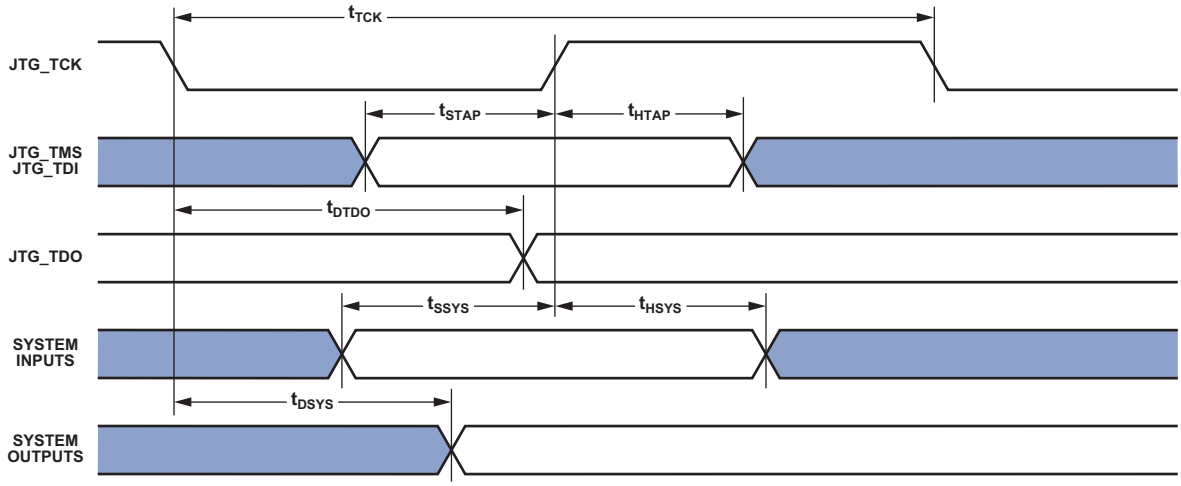


Figure 58. JTAG Port Timing

OUTPUT DRIVE CURRENTS

Figure 59 through Figure 64 show typical current-voltage characteristics for the output drivers of the ADSP-BF60x Blackfin processors. The curves represent the current drive capability of the output drivers as a function of output voltage.

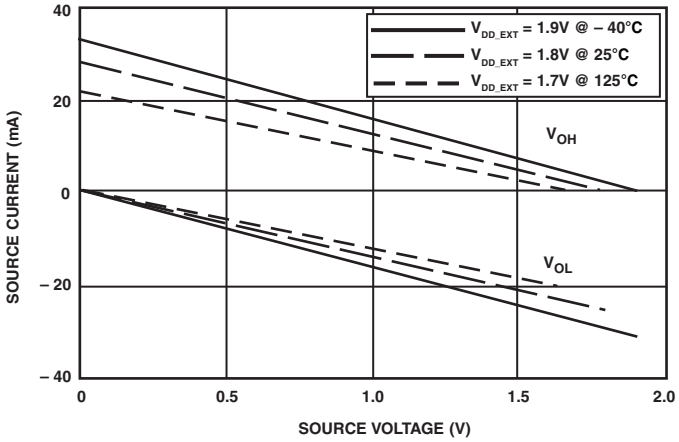


Figure 59. Driver Type A Current (1.8 V V_{DD_EXT})

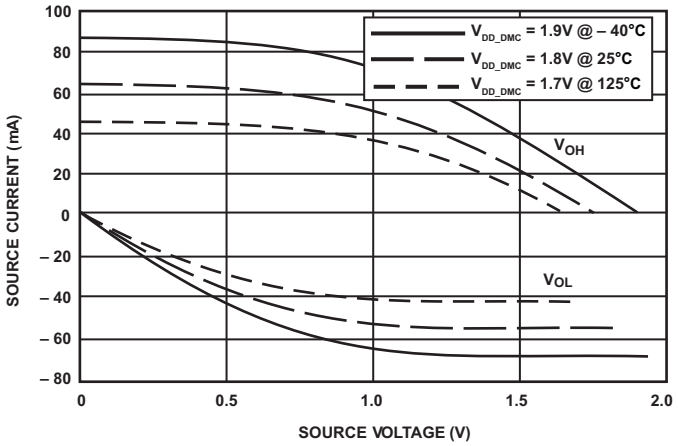


Figure 61. Driver Type B Current (1.8 V V_{DD_DMC})

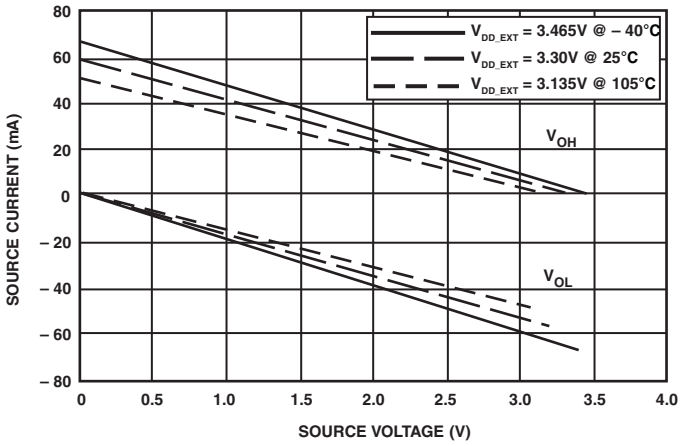


Figure 60. Driver Type A Current (3.3 V V_{DD_EXT})

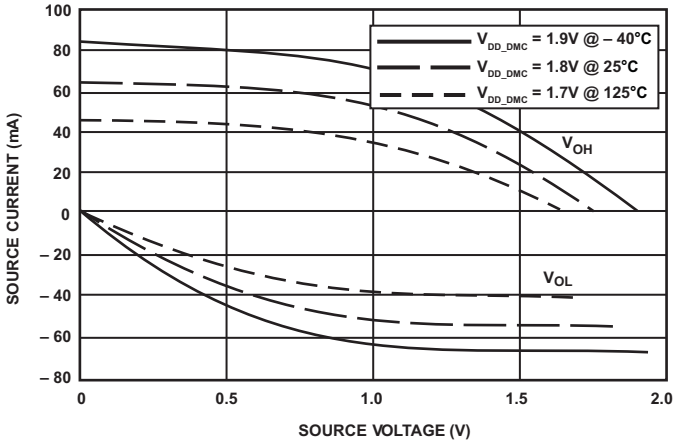


Figure 62. Driver Type C Current (1.8 V V_{DD_DMC})

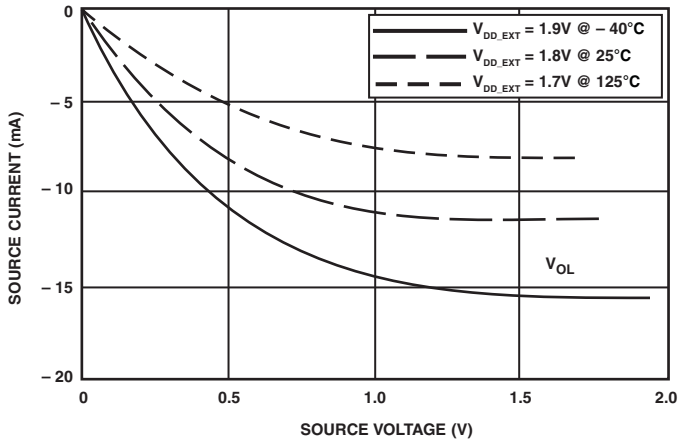


Figure 63. Driver Type D Current (1.8 V V_{DD_EXT})

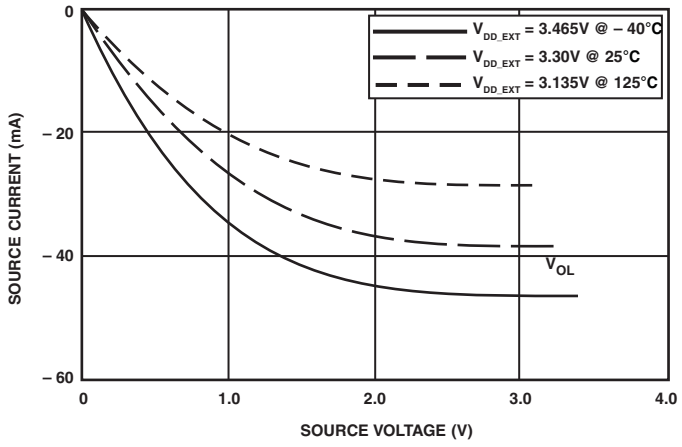


Figure 64. Driver Type D Current (3.3 V V_{DD_EXT})

TEST CONDITIONS

All Timing Requirements appearing in this data sheet were measured under the conditions described in this section. Figure 65 shows the measurement point for AC measurements (except output enable/disable). The measurement point V_{MEAS} is $V_{DDEXT}/2$ or $V_{DDMEM}/2$ for V_{DDEXT}/V_{DDMEM} (nominal) = 1.8 V/ 2.5 V/3.3 V.

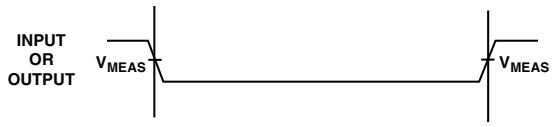


Figure 65. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time Measurement

Output balls 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 66.

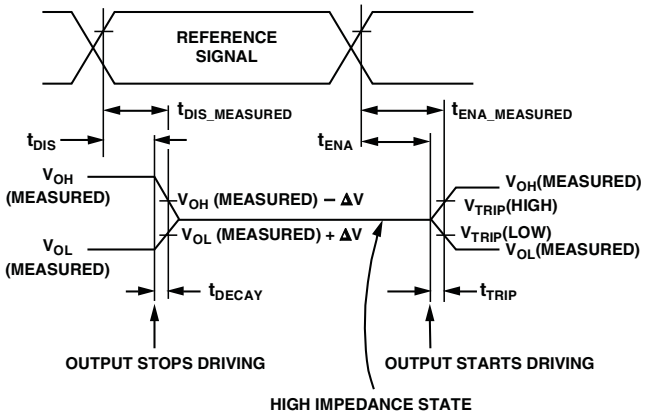


Figure 66. Output Enable/Disable

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}/V_{DDMEM} (nominal) = 1.8 V, V_{TRIP} (high) is 1.05 V, and V_{TRIP} (low) is 0.75 V. For V_{DDEXT}/V_{DDMEM} (nominal) = 2.5 V, V_{TRIP} (high) is 1.5 V and V_{TRIP} (low) is 1.0 V. For V_{DDEXT}/V_{DDMEM} (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 balls (such as the data bus) are enabled, the measurement value is that of the first ball to start driving.

Output Disable Time Measurement

Output balls 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 66.

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

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

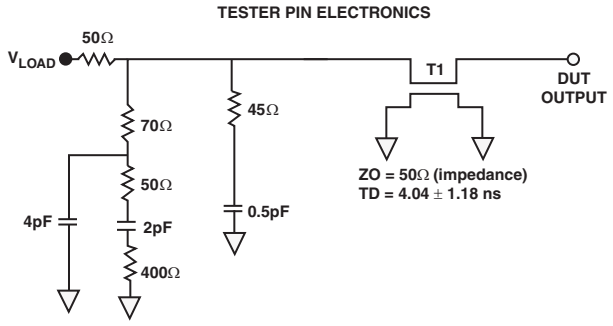
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches, to when the output voltage decays ΔV from the measured output high or output low voltage.

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 is t_{DECAY} plus the various output disable times as specified in the Timing Specifications on Page 60.

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all pins (see Figure 67). V_{LOAD} is equal to $(V_{DD\_EXT})/2$.



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

The graphs of Figure 68 through Figure 70 show how output rise and fall times vary with capacitance. The delay and hold specifications given should be derated by a factor derived from these figures. The graphs in these figures may not be linear outside the ranges shown.

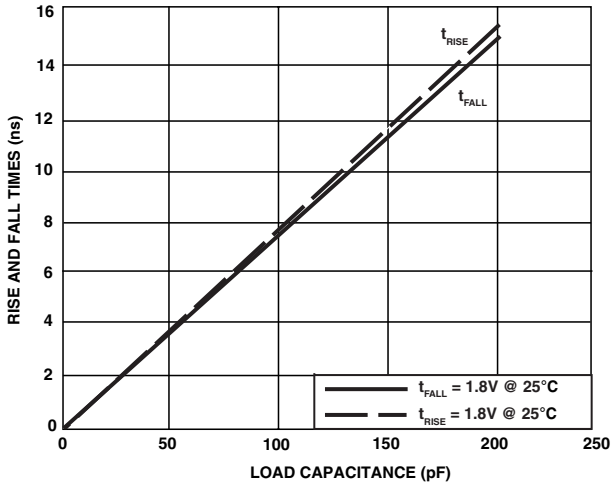


Figure 68. Driver Type A Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_EXT} = 1.8 V)

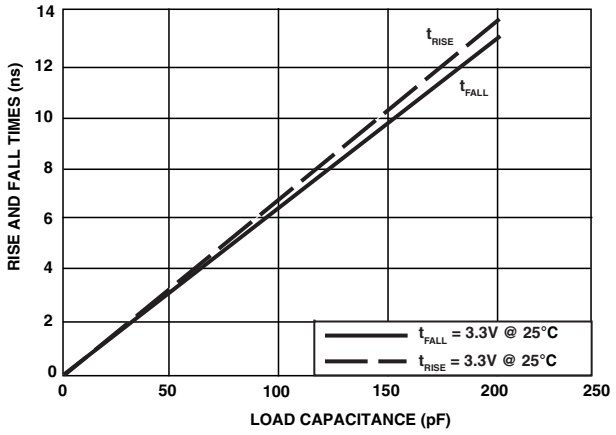


Figure 69. Driver Type A Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_EXT} = 3.3 V)

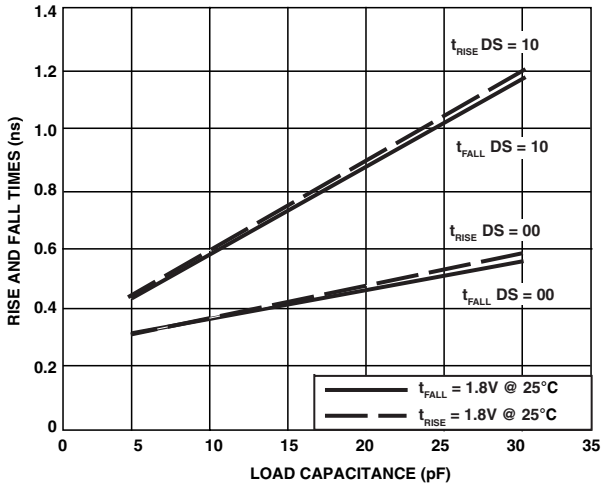


Figure 70. Driver Type B & C Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_DMC} = 1.8 V)

ENVIRONMENTAL CONDITIONS

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

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

where:

T_J = Junction temperature (°C)

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

Ψ_{JT} = From Table 68

P_D = Power dissipation (see Total Internal Power Dissipation on Page 57 for the method to calculate P_D)

Table 68. Thermal Characteristics

Parameter	Condition	Typical	Unit
θ_{JA}	0 linear m/s air flow	16.7	°C/W
θ_{JMA}	1 linear m/s air flow	14.6	°C/W
θ_{JMA}	2 linear m/s air flow	13.9	°C/W
θ_{JC}		4.41	°C/W
Ψ_{JT}	0 linear m/s air flow	0.11	°C/W
Ψ_{JT}	1 linear m/s air flow	0.24	°C/W
Ψ_{JT}	2 linear m/s air flow	0.25	°C/W

Values of θ_{JA} are provided for package comparison and printed circuit board design considerations. θ_{JA} can be used for a first order approximation of T_J by the equation:

$$T_J = T_A + (\theta_{JA} \times P_D)$$

where:

T_A = Ambient temperature (°C)

Values of θ_{JC} are provided for package comparison and printed circuit board design considerations when an external heat sink is required.

In Table 68, airflow measurements comply with JEDEC standards JESD51-2 and JESD51-6. The junction-to-case measurement complies with MIL-STD-883 (Method 1012.1). All measurements use a 2S2P JEDEC test board.

Thermal Diode

The processor incorporates a thermal diode to monitor the die temperature. The thermal diode is a grounded collector, PNP Bipolar Junction Transistor (BJT). The SYS_TDA ball is connected to the emitter and the SYS_TDK ball is connected to the base of the transistor. These balls can be used by an external temperature sensor (such as the ADM 1021A or the LM86 or others) to read the die temperature of the chip.

The technique used by the external temperature sensor is to measure the change in V_{BE} when the thermal diode is operated at two different currents. This is shown in the following equation:

$$\Delta V_{BE} = n_Q \times \frac{kT}{q} \times ln(N)$$

where:

n_Q = multiplication factor close to 1, depending on process variations

k = Boltzmann's constant

T = temperature (°Kelvin)

q = charge of the electron

N = ratio of the two currents

The two currents are usually in the range of 10 micro Amperes to 300 micro Amperes for the common temperature sensor chips available.

Table 69 contains the thermal diode specifications using the transistor model. Note that Measured Ideality Factor already takes into effect variations in beta (B).

Table 69. Thermal Diode Parameters—Transistor Model

Symbol	Parameter	Min	Typ	Max	Unit
I_{FW}[1]	Forward Bias Current	10		300	µA
I_E	Emitter Current	10		300	µA
n_Q[2, 3]	Transistor Ideality		1.006		
R_T[2, 4]	Series Resistance		2.8		Ω

[1] Analog Devices does not recommend operation of the thermal diode under reverse bias.

[2] Not 100% tested. Specified by design characterization.

[3] The ideality factor, n_Q, represents the deviation from ideal diode behavior as exemplified by the diode equation: $I_C = I_S \times (\exp(qV_{BE}/n_QkT - 1)$, where I_S = saturation current, q = electrical charge, V_{BE} = voltage across the diode, k = Boltzmann Constant, and T = absolute temperature (Kelvin).

[4] The series resistance (R_T) can be used for more accurate readings as needed.

ADSP-BF60x 349-BALL CSP_BGA BALL ASSIGNMENTS

The 349-Ball CSP_BGA Ball Assignment (Numerical by Ball Number) table lists the CSP_BGA package by ball number for the ADSP-BF609.

The 349-Ball CSP_BGA Ball Assignment (Alphabetical by Pin Name) table lists the CSP_BGA package by signal.

349-BALL CSP_BGA BALL ASSIGNMENT (NUMERICAL BY BALL NUMBER)

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
A01	GND	B21	GND	F06	V_{DD_EXT}	H22	DMC0_DQ14
A02	USB0_DM	B22	$\overline{\text{SMC0_AOE}}$_NORDV	F07	V_{DD_INT}	J01	GND
A03	USB0_DP	C01	USB0_CLKIN	F08	V_{DD_INT}	J02	SYS_PWRGD
A04	PB_10	C02	USB0_VBC	F09	V_{DD_INT}	J03	SYS_BMODE0
A05	PB_07	C03	GND	F10	V_{DD_INT}	J06	V_{DD_EXT}
A06	PA_14	C04	PB_12	F11	V_{DD_EXT}	J09	GND
A07	PA_12	C05	PB_09	F12	V_{DD_EXT}	J10	GND
A08	PA_10	C06	PB_06	F13	V_{DD_INT}	J11	GND
A09	PA_08	C07	PB_05	F14	V_{DD_INT}	J12	GND
A10	PA_06	C08	PB_04	F15	V_{DD_INT}	J13	GND
A11	PA_04	C09	PB_03	F16	V_{DD_INT}	J14	GND
A12	PA_02	C10	PB_02	F17	V_{DD_DMC}	J17	V_{DD_DMC}
A13	PA_00	C11	PB_01	F20	$\overline{\text{DMC0\_CS0}}$	J20	DMC0_ODT
A14	SMC0_A01	C12	PB_00	F21	DMC0_DQ15	J21	DMC0_DQ12
A15	SMC0_D00	C13	$\overline{\text{SMC0\_BR}}$	F22	DMC0_DQ08	J22	DMC0_DQ11
A16	$\overline{\text{SMC0\_AMS0}}$	C14	SMC0_D06	G01	GND	K01	PC_00
A17	SMC0_D03	C15	SMC0_D12	G02	$\overline{\text{SYS\_HWRST}}$	K02	SYS_EXTWAKE
A18	SMC0_D04	C16	$\overline{\text{SMC0\_ARE}}$	G03	SYS_BMODE2	K03	PB_13
A19	SMC0_D07	C17	SMC0_D08	G06	V_{DD_EXT}	K06	V_{DD_EXT}
A20	SMC0_D10	C18	SMC0_D11	G07	V_{DD_EXT}	K08	GND
A21	$\overline{\text{SMC0\_AWE}}$	C19	SMC0_D14	G08	V_{DD_INT}	K09	GND
A22	GND	C20	GND	G09	V_{DD_INT}	K10	GND
B01	USB0_VBUS	C21	TWI1_SCL	G10	V_{DD_EXT}	K11	GND
B02	GND	C22	TWI0_SCL	G11	V_{DD_EXT}	K12	GND
B03	USB0_ID	D01	JTG_TDI	G12	V_{DD_EXT}	K13	GND
B04	PB_11	D02	JTG_TDO	G13	V_{DD_EXT}	K14	GND
B05	PB_08	D03	JTG_TCK	G14	V_{DD_INT}	K15	GND
B06	PA_15	D11	V_{DD_EXT}	G15	V_{DD_INT}	K17	V_{DD_DMC}
B07	PA_13	D12	GND	G16	V_{DD_DMC}	K20	DMC0_LDM
B08	PA_11	D20	SMC0_ARDY_NORWT	G17	V_{DD_DMC}	K21	$\overline{\text{DMC0\_LDQS}}$
B09	PA_09	D21	TWI1_SDA	G20	DMC0_UDM	K22	DMC0_LDQS
B10	PA_07	D22	TWI0_SDA	G21	DMC0_UDQS	L01	PC_02
B11	PA_05	E01	$\overline{\text{JTG\_TRST}}$	G22	$\overline{\text{DMC0\_UDQS}}$	L02	PC_01
B12	PA_03	E02	$\overline{\text{JTG\_EMU}}$	H01	SYS_CLKIN	L03	PB_14
B13	PA_01	E03	JTG_TMS	H02	SYS_XTAL	L04	V_{DD_EXT}
B14	SMC0_A02	E05	V_{DD_USB}	H03	SYS_BMODE1	L06	V_{DD_EXT}
B15	SMC0_D01	E20	$\overline{\text{DMC0\_CAS}}$	H06	V_{DD_EXT}	L08	GND
B16	SMC0_D15	E21	DMC0_DQ10	H07	V_{DD_EXT}	L09	GND
B17	SMC0_D09	E22	DMC0_DQ13	H16	V_{DD_DMC}	L10	GND
B18	SMC0_D02	F01	$\overline{\text{SYS\_FAULT}}$	H17	V_{DD_DMC}	L11	GND
B19	SMC0_D13	F02	SYS_FAULT	H20	$\overline{\text{DMC0\_RAS}}$	L12	GND
B20	SMC0_D05	F03	$\overline{\text{SYS\_NMI\_RESOUT}}$	H21	DMC0_DQ09	L13	GND

Ball No.	Pin Name
L14	GND
L15	GND
L17	V_{DD_DMC}
L19	VREF_DMC
L20	$\overline{DMC0\_CK}$
L21	DMC0_DQ06
L22	DMC0_DQ07
M01	PC_04
M02	PC_03
M03	PB_15
M04	GND
M06	V_{DD_EXT}
M08	GND
M09	GND
M10	GND
M11	GND
M12	GND
M13	GND
M14	GND
M15	GND
M17	V_{DD_DMC}
M19	GND
M20	DMC0_CK
M21	DMC0_DQ00
M22	DMC0_DQ01
N01	PC_06
N02	PC_05
N03	SYS_CLKOUT
N06	V_{DD_EXT}
N08	GND
N09	GND
N10	GND
N11	GND
N12	GND
N13	GND
N14	GND
N15	GND
N17	V_{DD_DMC}
N20	$\overline{DMC0\_WE}$
N21	DMC0_DQ04
N22	DMC0_DQ03
P01	PC_08
P02	PC_07
P03	PD_06
P06	V_{DD_EXT}
P09	GND
P10	GND

Ball No.	Pin Name
P11	GND
P12	GND
P13	GND
P14	GND
P17	V_{DD_DMC}
P20	DMC0_CKE
P21	DMC0_DQ02
P22	DMC0_DQ05
R01	PC_10
R02	PC_09
R03	PD_07
R06	V_{DD_EXT}
R07	V_{DD_EXT}
R16	V_{DD_DMC}
R17	V_{DD_DMC}
R20	DMC0_BA2
R21	DMC0_BA0
R22	DMC0_A10
T01	PC_12
T02	PC_11
T03	PD_08
T06	V_{DD_EXT}
T07	V_{DD_EXT}
T08	V_{DD_INT}
T09	V_{DD_INT}
T10	V_{DD_EXT}
T11	V_{DD_EXT}
T12	V_{DD_EXT}
T13	V_{DD_EXT}
T14	V_{DD_INT}
T15	V_{DD_INT}
T16	V_{DD_DMC}
T17	V_{DD_DMC}
T20	DMC0_A03
T21	DMC0_A07
T22	DMC0_A12
U01	PC_14
U02	PC_13
U03	PD_09
U06	V_{DD_EXT}
U07	V_{DD_INT}
U08	V_{DD_INT}
U09	V_{DD_INT}
U10	V_{DD_INT}
U11	V_{DD_EXT}
U12	V_{DD_EXT}
U13	V_{DD_INT}

Ball No.	Pin Name
U14	V_{DD_INT}
U15	V_{DD_INT}
U16	V_{DD_INT}
U17	V_{DD_DMC}
U20	DMC0_A09
U21	DMC0_A05
U22	DMC0_A01
V01	PD_00
V02	PC_15
V03	PD_10
V20	DMC0_BA1
V21	DMC0_A13
V22	DMC0_A11
W01	PD_04
W02	PD_01
W03	PD_12
W11	GND
W12	V_{DD_TD}
W20	DMC0_A04
W21	DMC0_A06
W22	DMC0_A08
Y01	PD_03
Y02	PD_02
Y03	GND
Y04	PD_15
Y05	PE_02
Y06	PE_05
Y07	PE_06
Y08	PE_07
Y09	PE_08
Y10	PE_09
Y11	SYS_TDK
Y12	SYS_TDA
Y13	PE_12
Y14	PE_10
Y15	PE_11
Y16	PG_09
Y17	PG_01
Y18	PG_04
Y19	PG_11
Y20	GND
Y21	DMC0_A00
Y22	DMC0_A02
AA01	PD_11
AA02	GND
AA03	PD_13
AA04	PE_00

Ball No.	Pin Name
AA05	PE_03
AA06	PF_14
AA07	PF_12
AA08	PF_10
AA09	PF_08
AA10	PF_06
AA11	PF_04
AA12	PF_02
AA13	PF_00
AA14	PG_00
AA15	PE_15
AA16	PE_14
AA17	PG_05
AA18	PG_08
AA19	PG_07
AA20	PG_13
AA21	GND
AA22	GND
AB01	GND
AB02	PD_05
AB03	PD_14
AB04	PE_01
AB05	PE_04
AB06	PF_15
AB07	PF_13
AB08	PF_11
AB09	PF_09
AB10	PF_07
AB11	PF_05
AB12	PF_03
AB13	PF_01
AB14	PE_13
AB15	PG_03
AB16	PG_06
AB17	PG_02
AB18	PG_12
AB19	PG_14
AB20	PG_15
AB21	PG_10
AB22	GND

349-BALL CSP_BGA BALL ASSIGNMENT (ALPHABETICAL BY PIN NAME)

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
DMC0_A00	Y21	GND	A01	GND	N11	PB_09	C05
DMC0_A01	U22	GND	A22	GND	N12	PB_10	A04
DMC0_A02	Y22	GND	AA02	GND	N13	PB_11	B04
DMC0_A03	T20	GND	AA21	GND	N14	PB_12	C04
DMC0_A04	W20	GND	AA22	GND	N15	PB_13	K03
DMC0_A05	U21	GND	AB01	GND	P09	PB_14	L03
DMC0_A06	W21	GND	AB22	GND	P10	PB_15	M03
DMC0_A07	T21	GND	B21	GND	P11	PC_00	K01
DMC0_A08	W22	GND	C20	GND	P12	PC_01	L02
DMC0_A09	U20	GND	D12	GND	P13	PC_02	L01
DMC0_A10	R22	GND	G01	GND	P14	PC_03	M02
DMC0_A11	V22	GND	J01	GND	W11	PC_04	M01
DMC0_A12	T22	GND	J09	GND	Y03	PC_05	N02
DMC0_A13	V21	GND	J10	GND	Y20	PC_06	N01
DMC0_BA0	R21	GND	J11	GND	C03	PC_07	P02
DMC0_BA1	V20	GND	J12	GND	B02	PC_08	P01
DMC0_BA2	R20	GND	J13	$\overline{\text{JTG\_EMU}}$	E02	PC_09	R02
$\overline{\text{DMC0\_CAS}}$	E20	GND	J14	JTG_TCK	D03	PC_10	R01
DMC0_CK	M20	GND	K08	JTG_TDI	D01	PC_11	T02
DMC0_CKE	P20	GND	K09	JTG_TDO	D02	PC_12	T01
$\overline{\text{DMC0\_CK}}$	L20	GND	K10	JTG_TMS	E03	PC_13	U02
$\overline{\text{DMC0\_CS0}}$	F20	GND	K11	$\overline{\text{JTG\_TRST}}$	E01	PC_14	U01
DMC0_DQ00	M21	GND	K12	PA_00	A13	PC_15	V02
DMC0_DQ01	M22	GND	K13	PA_01	B13	PD_00	V01
DMC0_DQ02	P21	GND	K14	PA_02	A12	PD_01	W02
DMC0_DQ03	N22	GND	K15	PA_03	B12	PD_02	Y02
DMC0_DQ04	N21	GND	L08	PA_04	A11	PD_03	Y01
DMC0_DQ05	P22	GND	L09	PA_05	B11	PD_04	W01
DMC0_DQ06	L21	GND	L10	PA_06	A10	PD_05	AB02
DMC0_DQ07	L22	GND	L11	PA_07	B10	PD_06	P03
DMC0_DQ08	F22	GND	L12	PA_08	A09	PD_07	R03
DMC0_DQ09	H21	GND	L13	PA_09	B09	PD_08	T03
DMC0_DQ10	E21	GND	L14	PA_10	A08	PD_09	U03
DMC0_DQ11	J22	GND	L15	PA_11	B08	PD_10	V03
DMC0_DQ12	J21	GND	M04	PA_12	A07	PD_11	AA01
DMC0_DQ13	E22	GND	M08	PA_13	B07	PD_12	W03
DMC0_DQ14	H22	GND	M09	PA_14	A06	PD_13	AA03
DMC0_DQ15	F21	GND	M10	PA_15	B06	PD_14	AB03
DMC0_LDM	K20	GND	M11	PB_00	C12	PD_15	Y04
DMC0_LDQS	K22	GND	M12	PB_01	C11	PE_00	AA04
$\overline{\text{DMC0\_LDQS}}$	K21	GND	M13	PB_02	C10	PE_01	AB04
DMC0_ODT	J20	GND	M14	PB_03	C09	PE_02	Y05
$\overline{\text{DMC0\_RAS}}$	H20	GND	M15	PB_04	C08	PE_03	AA05
DMC0_UDM	G20	GND	M19	PB_05	C07	PE_04	AB05
DMC0_UDQS	G21	GND	N08	PB_06	C06	PE_05	Y06
$\overline{\text{DMC0\_UDQS}}$	G22	GND	N09	PB_07	A05	PE_06	Y07
$\overline{\text{DMC0\_WE}}$	N20	GND	N10	PB_08	B05	PE_07	Y08

Pin Name	Ball No.
PE_08	Y09
PE_09	Y10
PE_10	Y14
PE_11	Y15
PE_12	Y13
PE_13	AB14
PE_14	AA16
PE_15	AA15
PF_00	AA13
PF_01	AB13
PF_02	AA12
PF_03	AB12
PF_04	AA11
PF_05	AB11
PF_06	AA10
PF_07	AB10
PF_08	AA09
PF_09	AB09
PF_10	AA08
PF_11	AB08
PF_12	AA07
PF_13	AB07
PF_14	AA06
PF_15	AB06
PG_00	AA14
PG_01	Y17
PG_02	AB17
PG_03	AB15
PG_04	Y18
PG_05	AA17
PG_06	AB16
PG_07	AA19
PG_08	AA18
PG_09	Y16
PG_10	AB21
PG_11	Y19
PG_12	AB18
PG_13	AA20
PG_14	AB19
PG_15	AB20
SMC0_A01	A14
SMC0_A02	B14
$\overline{\text{SMC0\_AMS0}}$	A16
$\overline{\text{SMC0_AOE}}$_NORDV	B22
SMC0_ARDY_NORWT	D20
$\overline{\text{SMC0\_ARE}}$	C16
$\overline{\text{SMC0\_AWE}}$	A21

Pin Name	Ball No.
$\overline{\text{SMC0\_BR}}$	C13
SMC0_D00	A15
SMC0_D01	B15
SMC0_D02	B18
SMC0_D03	A17
SMC0_D04	A18
SMC0_D05	B20
SMC0_D06	C14
SMC0_D07	A19
SMC0_D08	C17
SMC0_D09	B17
SMC0_D10	A20
SMC0_D11	C18
SMC0_D12	C15
SMC0_D13	B19
SMC0_D14	C19
SMC0_D15	B16
SYS_BMODE0	J03
SYS_BMODE1	H03
SYS_BMODE2	G03
SYS_CLKIN	H01
SYS_CLKOUT	N03
SYS_EXTWAKE	K02
SYS_FAULT	F02
$\overline{\text{SYS\_FAULT}}$	F01
$\overline{\text{SYS\_NMI\_RESOUT}}$	F03
SYS_PWRGD	J02
$\overline{\text{SYS\_HWRST}}$	G02
SYS_TDA	Y12
SYS_TDK	Y11
SYS_XTAL	H02
TWI0_SCL	C22
TWI0_SDA	D22
TWI1_SCL	C21
TWI1_SDA	D21
USB0_CLKIN	C01
USB0_DM	A02
USB0_DP	A03
USB0_ID	B03
USB0_VBC	C02
USB0_VBUS	B01
V_{DD_DMC}	F17
V_{DD_DMC}	G16
V_{DD_DMC}	G17
V_{DD_DMC}	H16
V_{DD_DMC}	H17
V_{DD_DMC}	J17

Pin Name	Ball No.
V_{DD_DMC}	K17
V_{DD_DMC}	L17
V_{DD_DMC}	M17
V_{DD_DMC}	N17
V_{DD_DMC}	P17
V_{DD_DMC}	R16
V_{DD_DMC}	R17
V_{DD_DMC}	T16
V_{DD_DMC}	T17
V_{DD_DMC}	U17
V_{DD_EXT}	D11
V_{DD_EXT}	F06
V_{DD_EXT}	F11
V_{DD_EXT}	F12
V_{DD_EXT}	G06
V_{DD_EXT}	G07
V_{DD_EXT}	G10
V_{DD_EXT}	G11
V_{DD_EXT}	G12
V_{DD_EXT}	G13
V_{DD_EXT}	H06
V_{DD_EXT}	H07
V_{DD_EXT}	J06
V_{DD_EXT}	K06
V_{DD_EXT}	L04
V_{DD_EXT}	L06
V_{DD_EXT}	M06
V_{DD_EXT}	N06
V_{DD_EXT}	P06
V_{DD_EXT}	R06
V_{DD_EXT}	R07
V_{DD_EXT}	T06
V_{DD_EXT}	T07
V_{DD_EXT}	T10
V_{DD_EXT}	T11
V_{DD_EXT}	T12
V_{DD_EXT}	T13
V_{DD_EXT}	U06
V_{DD_EXT}	U11
V_{DD_EXT}	U12
V_{DD_INT}	F07
V_{DD_INT}	F08
V_{DD_INT}	F09
V_{DD_INT}	F10
V_{DD_INT}	F13
V_{DD_INT}	F14
V_{DD_INT}	F15

Pin Name	Ball No.
V_{DD_INT}	F16
V_{DD_INT}	G08
V_{DD_INT}	G09
V_{DD_INT}	G14
V_{DD_INT}	G15
V_{DD_INT}	T08
V_{DD_INT}	T09
V_{DD_INT}	T14
V_{DD_INT}	T15
V_{DD_INT}	U07
V_{DD_INT}	U08
V_{DD_INT}	U09
V_{DD_INT}	U10
V_{DD_INT}	U13
V_{DD_INT}	U14
V_{DD_INT}	U15
V_{DD_INT}	U16
V_{DD_TD}	W12
V_{DD_USB}	E05
VREF_DMC	L19

349-BALL CSP_BGA BALL CONFIGURATION

Figure 71 shows an overview of signal placement on the 349-ball CSP_BGA package.

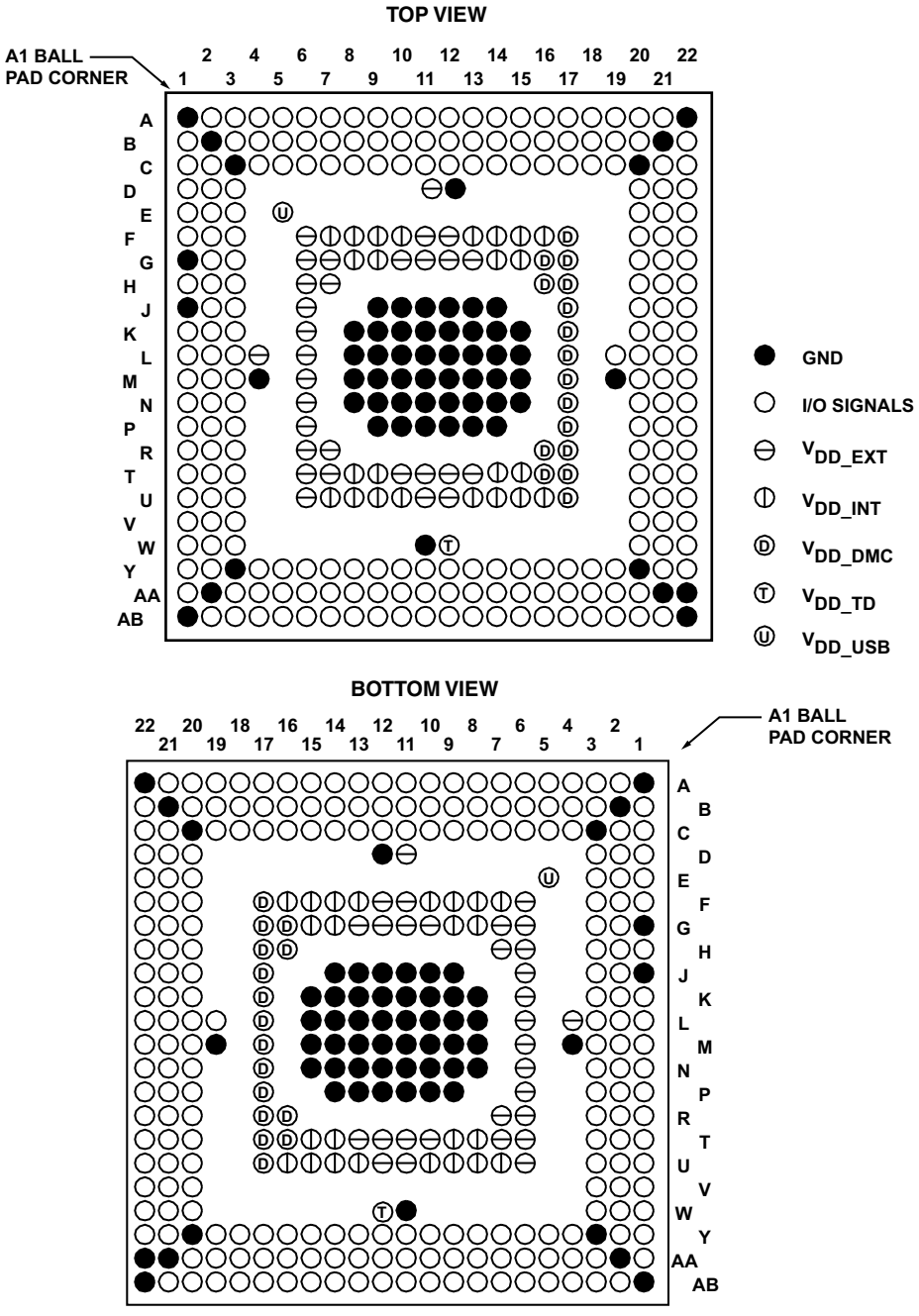


Figure 71. 349-Ball CSP_BGA Ball Configuration

OUTLINE DIMENSIONS

Dimensions for the 19 mm × 19 mm CSP_BGA package in Figure 72 are shown in millimeters.

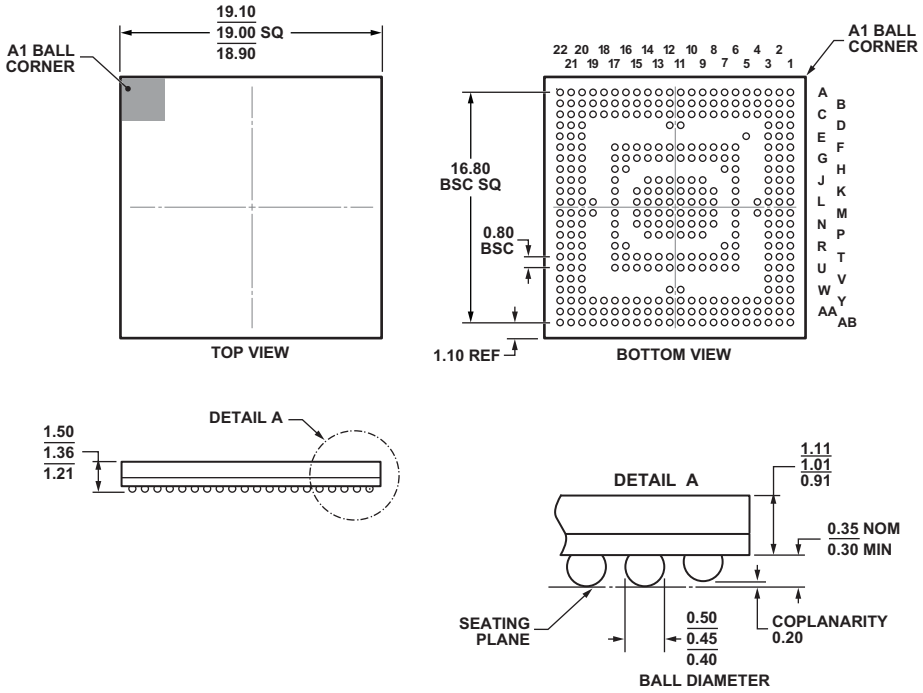


Figure 72. 349-Ball Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-349-1)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 70 is provided as an aid to PCB design. For industry-standard design recommendations, refer to IPC-7351, *Generic Requirements for Surface-Mount Design and Land Pattern Standard.*

Table 70. BGA Data for Use with Surface-Mount Design

Package	Package Ball Attach Type	Package Solder Mask Opening	Package Ball Pad Size
BC-349-1	Solder Mask Defined	0.4 mm Diameter	0.5 mm Diameter

AUTOMOTIVE PRODUCTS

The models in the following table are available with controlled manufacturing to support the quality and reliability requirements of automotive applications. Note that these automotive models may have specifications that differ from the commercial models and designers should review the product specifications section of this data sheet carefully. Contact your local ADI account representative for specific product ordering information and to obtain the specific Automotive Reliability reports for these models.

Model[1]	Max. Core Clock	Temperature Range[2]	Package Description	Package Option
ADBF606WCBCZ4xx	400 MHz	–40°C to +105°C	349-Ball CSP_BGA	BC-349-1
ADBF607WCBCZ5xx	500 MHz	–40°C to +105°C	349-Ball CSP_BGA	BC-349-1
ADBF608WCBCZ5xx	500 MHz	–40°C to +105°C	349-Ball CSP_BGA	BC-349-1
ADBF609WCBCZ5xx	500 MHz	–40°C to +105°C	349-Ball CSP_BGA	BC-349-1

[1] Z =RoHS compliant part.

[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification. Please see Operating Conditions on Page 52 for the junction temperature (T_J) specification which is the only temperature specification.

ORDERING GUIDE

Model[1]	Max. Core Clock	Temperature Range[2]	Package Description	Package Option
ADSP-BF606KBCZ-4	400 MHz	0°C to +70°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF606BBCZ-4	400 MHz	–40°C to +85°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF607KBCZ-5	500 MHz	0°C to +70°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF607BBCZ-5	500 MHz	–40°C to +85°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF608KBCZ-5	500 MHz	0°C to +70°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF608BBCZ-5	500 MHz	–40°C to +85°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF609KBCZ-5	500 MHz	0°C to +70°C	349-Ball CSP_BGA	BC-349-1
ADSP-BF609BBCZ-5	500 MHz	–40°C to +85°C	349-Ball CSP_BGA	BC-349-1

[1] Z =RoHS compliant part.

[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification. Please see Operating Conditions on Page 52 for the junction temperature (T_J) specification which is the only temperature specification.

D10659-0-02/14(A)

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View ADBF609WCBCZ502 on WIN SOURCE
- Analog Devices Inc. Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management