Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

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

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	-
Interface	-
Clock Rate	-
Non-Volatile Memory	-
On-Chip RAM	-
Voltage - I/O	-
Voltage - Core	-
Operating Temperature	-
Mounting Type	-
Package / Case	-
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adbf539wbbcz4f805

BLACKFIN PROCESSOR CORE

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

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

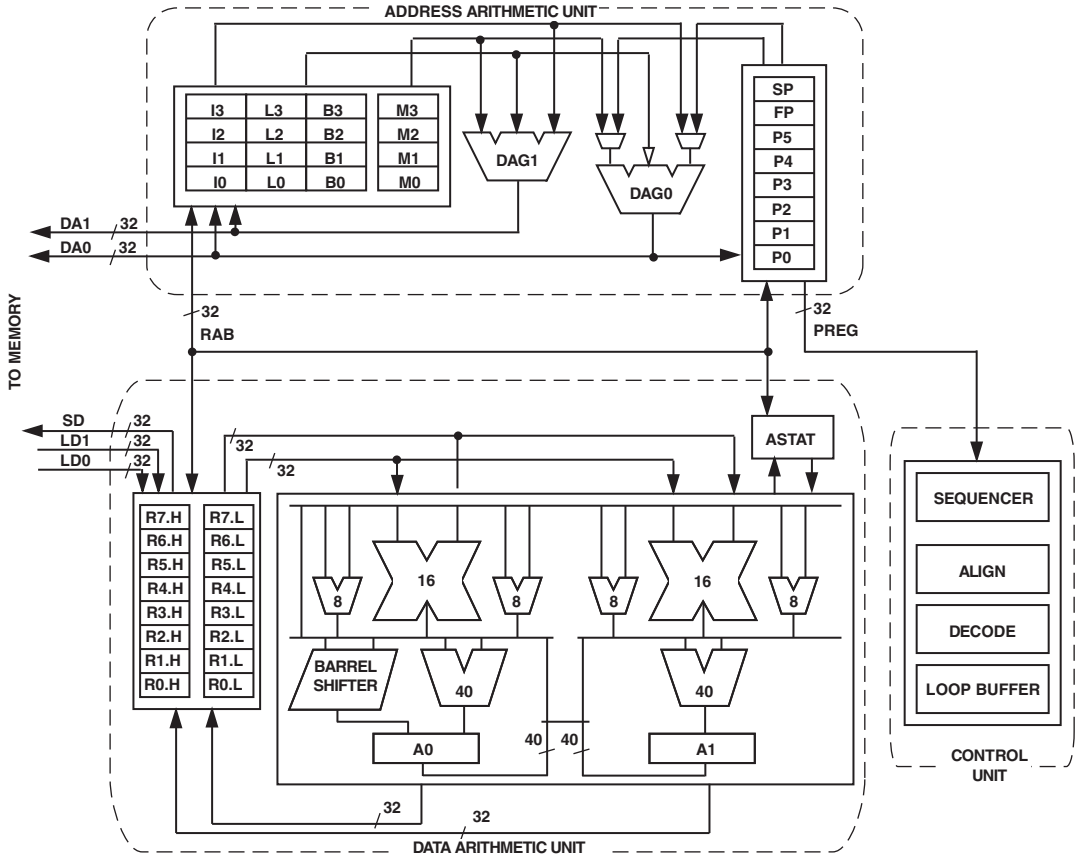


Figure 2. Blackfin Processor Core

Each MAC can perform a 16-bit by 16-bit multiply in each cycle, accumulating the results into the 40-bit accumulators. Signed and unsigned formats, rounding, and saturation are supported.

The ALUs perform a traditional set of arithmetic and logical operations on 16-bit or 32-bit data. In addition, many special instructions are included to accelerate various signal processing tasks. These include bit operations such as field extract and population count, modulo 2^{32} multiply, divide primitives, saturation and rounding, and sign/exponent detection. The set of video instructions include byte alignment and packing operations 16-bit and 8-bit adds with clipping, 8-bit average operations, and 8-bit subtract/absolute value/accumulate (SAA) operations. Also provided are the compare/select and vector search instructions.

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

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

The program sequencer controls the flow of instruction execution, including instruction alignment and decoding. For program flow control, the sequencer supports PC relative and indirect conditional jumps (with static branch prediction), and subroutine calls. Hardware is provided to support zero overhead looping. The architecture is fully interlocked, meaning that the programmer need not manage the pipeline when executing instructions with data dependencies.

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

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

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

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

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

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

MEMORY ARCHITECTURE

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

The L1 memory system is the primary highest performance memory available to the Blackfin processor. The off-chip memory system, accessed through the external bus interface unit (EBIU), provides expansion with SDRAM, flash memory, and SRAM, optionally accessing up to 132M bytes of physical memory.

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

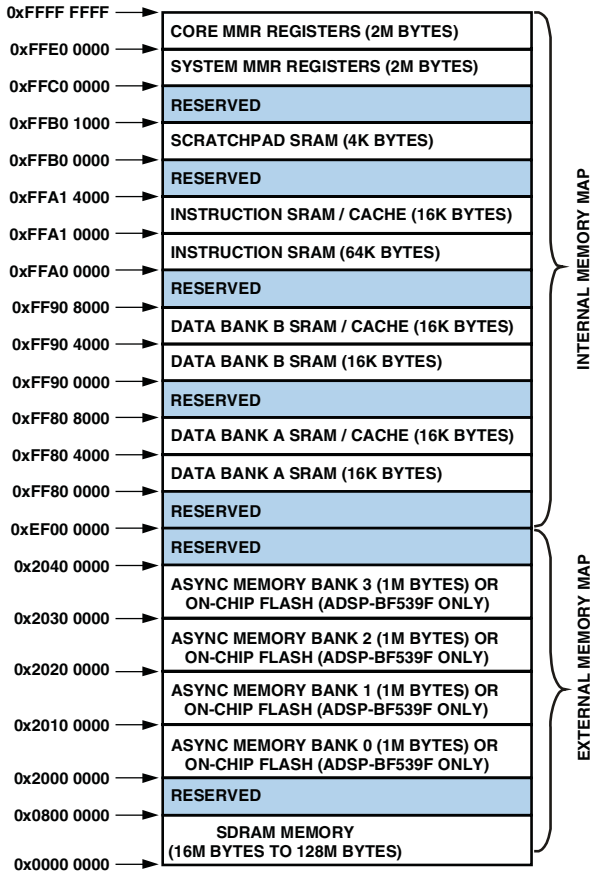


Figure 3. ADSP-BF539/ADSP-BF539F Internal/External Memory Map

Internal (On-Chip) Memory

The ADSP-BF539/ADSP-BF539F processor has three blocks of on-chip memory, providing high bandwidth access to the core.

The first is the L1 instruction memory, consisting of 80K bytes SRAM, of which 16K bytes can be configured as a four-way set-associative cache. This memory is accessed at full processor speed.

The second on-chip memory block is the L1 data memory, consisting of two banks of up to 32K bytes each. Each memory bank is configurable, offering both cache and SRAM functionality. This memory block is accessed at full processor speed.

The third memory block is a 4K byte scratch pad SRAM, which runs at the same speed as the L1 memories, but is only accessible as data SRAM and cannot be configured as cache memory.

External (Off-Chip) Memory

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

general-purpose interrupt to the IPEND output asserted is three core clock cycles; however, the latency can be much higher, depending on the activity within and the state of the processor.

Table 2. Core Event Controller (CEC)

Priority (0 is Highest)	Event Class	EVT Entry
0	Emulation/Test Control	EMU
1	Reset	RST
2	Nonmaskable Interrupt	NMI
3	Exception	EVX
4	Reserved	—
5	Hardware Error	IVHW
6	Core Timer	IVTMR
7	General Interrupt 7	IVG7
8	General Interrupt 8	IVG8
9	General Interrupt 9	IVG9
10	General Interrupt 10	IVG10
11	General Interrupt 11	IVG11
12	General Interrupt 12	IVG12
13	General Interrupt 13	IVG13
14	General Interrupt 14	IVG14
15	General Interrupt 15	IVG15

Table 3. System and Core Event Mapping

Event Source	Core Event Name
PLL Wake-Up Interrupt	IVG7
DMA Controller 0 Error	IVG7
DMA Controller 1 Error	IVG7
PPI Error Interrupt	IVG7
SPORT0 Error Interrupt	IVG7
SPORT1 Error Interrupt	IVG7
SPORT2 Error Interrupt	IVG7
SPORT3 Error Interrupt	IVG7
MXVR Synchronous Data Interrupt	IVG7
SPI0 Error Interrupt	IVG7
SPI1 Error Interrupt	IVG7
SPI2 Error Interrupt	IVG7
UART0 Error Interrupt	IVG7
UART1 Error Interrupt	IVG7
UART2 Error Interrupt	IVG7
CAN Error Interrupt	IVG7
Real-Time Clock Interrupt	IVG8
DMA0 Interrupt (PPI)	IVG8
DMA1 Interrupt (SPORT0 Rx)	IVG9
DMA2 Interrupt (SPORT0 Tx)	IVG9
DMA3 Interrupt (SPORT1 Rx)	IVG9
DMA4 Interrupt (SPORT1 Tx)	IVG9
DMA8 Interrupt (SPORT2 Rx)	IVG9
DMA9 Interrupt (SPORT2 Tx)	IVG9
DMA10 Interrupt (SPORT3 Rx)	IVG9
DMA11 Interrupt (SPORT3 Tx)	IVG9
DMA5 Interrupt (SPI0)	IVG10
DMA14 Interrupt (SPI1)	IVG10
DMA15 Interrupt (SPI2)	IVG10
DMA6 Interrupt (UART0 Rx)	IVG10
DMA7 Interrupt (UART0 Tx)	IVG10
DMA16 Interrupt (UART1 Rx)	IVG10
DMA17 Interrupt (UART1 Tx)	IVG10
DMA18 Interrupt (UART2 Rx)	IVG10
DMA19 Interrupt (UART2 Tx)	IVG10
Timer0, Timer1, Timer2 Interrupts	IVG11
TWI0 Interrupt	IVG11
TWI1 Interrupt	IVG11
CAN Receive Interrupt	IVG11
CAN Transmit Interrupt	IVG11
MXVR Status Interrupt	IVG11
MXVR Control Message Interrupt	IVG11
MXVR Asynchronous Packet Interrupt	IVG11
Programmable Flags Interrupts	IVG12
MDMA0 Stream 0 Interrupt	IVG13
MDMA0 Stream 1 Interrupt	IVG13
MDMA1 Stream 0 Interrupt	IVG13
MDMA1 Stream 1 Interrupt	IVG13
Software Watchdog Timer	IVG13

DMA CONTROLLERS

The processors have multiple, independent DMA controllers that support automated data transfers with minimal overhead for the processor core. DMA transfers can occur between the ADSP-BF539/ADSP-BF539F processor internal memories and any of its DMA capable peripherals. Additionally, DMA transfers can be accomplished between any of the DMA-capable peripherals and external devices connected to the external memory interfaces, including the SDRAM controller and the asynchronous memory controller. DMA capable peripherals include the SPORTs, SPI ports, UARTs, and PPI. Each individual DMA capable peripheral has at least one dedicated DMA channel. In addition, the MXVR peripheral has its own dedicated DMA controller, which supports its own unique set of operating modes.

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

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

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

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

In addition to the dedicated peripheral DMA channels, there are four memory DMA channels provided for transfers between the various memories of the ADSP-BF539/ADSP-BF539F processor system. This enables transfers of blocks of data between any of the memories—including external SDRAM, ROM, SRAM, and flash memory—with minimal processor intervention. Memory DMA transfers can be controlled by a very flexible descriptor-based methodology or by a standard register-based autobuffer mechanism.

REAL-TIME CLOCK

The ADSP-BF539/ADSP-BF539F processor real-time clock (RTC) provides a robust set of digital watch features, including current time, stopwatch, and alarm. The RTC is clocked by a 32.768 kHz crystal external to the Blackfin processors. The RTC peripheral has dedicated power supply pins so that it can remain powered up and clocked even when the rest of the processor is in a low power state. The RTC provides several programmable interrupt options, including interrupt per second, minute, hour, or day clock ticks, interrupt on programmable stopwatch countdown, or interrupt at a programmed alarm time.

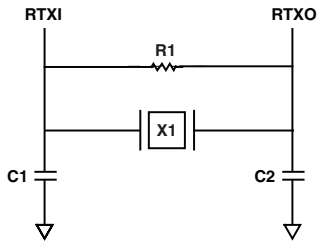
The 32.768 kHz input clock frequency is divided down to a 1 Hz signal by a prescaler. The counter function of the timer consists of four counters: a 60-second counter, a 60-minute counter, a 24-hour counter, and an 32,768-day counter.

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

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

Like the other peripherals, the RTC can wake up the processor from sleep mode upon generation of any RTC wake-up event. Additionally, an RTC wake-up event can wake up the processor from deep sleep mode, and wake up the on-chip internal voltage regulator from a powered down state.

Connect RTC pins RTXI and RTXO with external components as shown in Figure 5.



SUGGESTED COMPONENTS:
ECLIPTEK EC38J (THROUGH-HOLE PACKAGE)
EPSON MC405 12 pF LOAD (SURFACE-MOUNT PACKAGE)
C1 = 22pF
C2 = 22pF
R1 = 10MΩ

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

Figure 5. External Components for RTC

WATCHDOG TIMER

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

If configured to generate a hardware reset, the watchdog timer resets both the core and the processor peripherals. After a reset, software can determine if the watchdog was the source of the hardware reset by interrogating a status bit in the watchdog timer control register.

The timer is clocked by the system clock (SCLK), at a maximum frequency of f_{SCLK}.

TIMERS

There are four general-purpose programmable timer units in the ADSP-BF539/ADSP-BF539F processors. Three timers have an external pin that can be configured either as a pulse-width modulator (PWM) or timer output, as an input to clock the timer, or as a mechanism for measuring pulse widths and periods of external events. These timers can be synchronized to

Sleep Operating Mode—High Dynamic Power Savings

The sleep mode reduces dynamic power dissipation by disabling the clock to the processor core (CCLK). The PLL and system clock (SCLK), however, continue to operate in this mode. Typically, an external event or RTC activity wakes up the processor. When in the sleep mode, assertion of a wake-up event enabled in the SIC_IWRx register causes the processor to sense the value of the BYPASS bit in the PLL control register (PLL_CTL). If BYPASS is disabled, the processor transitions to the full on mode. If BYPASS is enabled, the processor will transition to the active mode. When in the sleep mode, system DMA access to L1 memory is not supported.

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core (CCLK) and to all synchronous peripherals (SCLK). Asynchronous peripherals such as the RTC may still be running but will not be able to access internal resources or external memory. This powered-down mode can only be exited by assertion of the reset interrupt ($\overline{\text{RESET}}$) or by an asynchronous interrupt generated by the RTC. When in deep sleep mode, an RTC asynchronous interrupt causes the processor to transition to the active mode. Assertion of $\overline{\text{RESET}}$ while in deep sleep mode causes the processor to transition to the full-on mode.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor core (CCLK) and to all the synchronous peripherals (SCLK). The internal voltage regulator for the processor can be shut off by writing b#00 to the FREQ bits of the VR_CTL register. This sets the internal power supply voltage (V_{DDINT}) to 0 V to provide the lowest static power dissipation. Any critical information stored internally (memory contents, register contents, etc.) must be written to a nonvolatile storage device prior to removing power if the processor state is to be preserved. Since V_{DDEXT} can still be supplied in this mode, all of the external pins three-state, unless otherwise specified. This allows other devices that may be connected to the processor to still have power applied without drawing unwanted current. The internal supply regulator can be woken up either by a real-time clock wake-up, by CAN bus traffic, by asserting the $\overline{\text{RESET}}$ pin, or by an external source via the $\overline{\text{GPW}}$ pin.

Power Savings

As shown in Table 6, the ADSP-BF539/ADSP-BF539F processors support five different power domains. The use of multiple power domains maximizes flexibility, while maintaining compliance with industry standards and conventions:

- The 3.3 V V_{DDRTC} power domain supplies the RTC I/O and logic so that the RTC can remain functional when the rest of the chip is powered off.
- The 3.3 V MXEVDD power domain supplies the MXVR crystal and is separate to provide noise isolation.
- The 1.25 V MPIVDD power domain supplies the MXVR PLL and is separate to provide noise isolation.
- The 1.25 V V_{DDINT} power domain supplies all internal logic except for the RTC logic and the MXVR PLL.
- The 3.3 V V_{DDEXT} power domain supplies all I/O except for the RTC and MXVR crystals.

There are no sequencing requirements for the various power domains.

Table 6. Power Domains

Power Domain	V_{DD} Range
RTC Crystal I/O and Logic	V_{DDRTC}
MXVR Crystal I/O	MXEVDD
MXVR PLL Analog and Logic	MPIVDD
All Internal Logic Except RTC and MXVR PLL	V_{DDINT}
All I/O Except RTC and MXVR Crystals	V_{DDEXT}

The V_{DDRTC} should either be connected to an isolated supply such as a battery (if the RTC is to operate while the rest of the chip is powered down) or should be connected to the V_{DDEXT} plane on the board. The V_{DDRTC} should remain powered when the processor is in hibernate state and should also remain powered even if the RTC functionality is not being used in an application. The MXEVDD should be connected to the V_{DDEXT} plane on the board at a single location with local bypass capacitors. The MXEVDD should remain powered when the processor is in hibernate state and should also remain powered even when the MXVR functionality is not being used in an application. The MPIVDD should be connected to the V_{DDINT} plane on the board at a single location through a ferrite bead with local bypass capacitors.

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

The dynamic power management feature of the ADSP-BF539/ADSP-BF539F processors allow both the processor input voltage (V_{DDINT}) and clock frequency (f_{CCLK}) to be dynamically controlled.

The savings in power dissipation can be modeled using the power savings factor and % power savings calculations.

The power savings factor is calculated as

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

where:

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

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

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

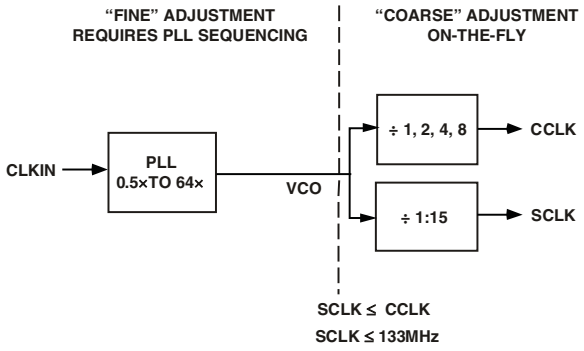


Figure 8. Frequency Modification Methods

All on-chip peripherals are clocked by the system clock (SCLK). The system clock frequency is programmable by means of the SSEL3–0 bits of the PLL_DIV register. The values programmed into the SSEL fields define a divide ratio between the PLL output (VCO) and the system clock. SCLK divider values are 1 through 15. Table 7 illustrates typical system clock ratios.

Table 7. Example System Clock Ratios

Signal Name SSEL3–0	Divider Ratio VCO/SCLK	Example Frequency Ratios (MHz)	
		VCO	SCLK
0001	1:1	100	100
0110	6:1	300	50
1010	10:1	500	50

The maximum frequency of the system clock is f_{SCLK}. Note that the divisor ratio must be chosen to limit the system clock frequency to its maximum of f_{SCLK}. The SSEL value can be changed dynamically without any PLL lock latencies by writing the appropriate values to the PLL divisor register (PLL_DIV).

Note that when the SSEL value is changed, it will affect all the peripherals that derive their clock signals from the SCLK signal.

The core clock (CCLK) frequency can also be dynamically changed by means of the CSEL1–0 bits of the PLL_DIV register. Supported CCLK divider ratios are 1, 2, 4, and 8, as shown in Table 8. This programmable core clock capability is useful for fast core frequency modifications.

Table 8. Core Clock Ratios

Signal Name CSEL1–0	Divider Ratio VCO/CCLK	Example Frequency Ratios	
		VCO	CCLK
00	1:1	300	300
01	2:1	300	150
10	4:1	500	125
11	8:1	200	25

BOOTING MODES

The ADSP-BF539/ADSP-BF539F processors have three mechanisms (listed in Table 9) for automatically loading internal L1 instruction memory after a reset. A fourth mode is provided to execute from external memory, bypassing the boot sequence.

Table 9. Booting Modes

BMODE1–0	Description
00	Execute from 16-bit external memory (bypass boot ROM)
01	Boot from 8-bit or 16-bit flash or boot from on-chip flash (ADSP-BF539F only)
10	Boot from SPI serial master connected to SPI0
11	Boot from SPI serial slave EEPROM/flash (8-,16-, or 24-bit address range, or Atmel AT45DB041, AT45DB081, or AT45DB161serial flash) connected to SPI0

The BMODE pins of the reset configuration register, sampled during power-on resets and software initiated resets, implement the following modes:

- Execute from 16-bit external memory – Execution starts from address 0x2000 0000 with 16-bit packing. The boot ROM is bypassed in this mode. All configuration settings are set for the slowest device possible (3-cycle hold time; 15-cycle R/W access times; 4-cycle setup).
- Boot from 8-bit or 16-bit external flash memory – The 8-bit flash boot routine located in boot ROM memory space is set up using asynchronous memory bank 0. For ADSP-BF539F processors, if $\overline{FCE}$ is connected to $\overline{AMS0}$, then the on-chip flash is booted. All configuration settings are set for the slowest device possible (3-cycle hold time; 15-cycle R/W access times; 4-cycle setup).
- Boot from SPI serial EEPROM/flash (8-, 16-, or 24-bit addressable, or Atmel AT45DB041, AT45DB081, or AT45DB161) connected to SPI0 – The SPI0 port uses the PF2 output pin to select a single SPI EEPROM/flash device, submits a read command and successive address bytes (0x00) until a valid 8-, 16-, or 24-bit, or Atmel addressable device is detected, and begins clocking data into the beginning of the L1 instruction memory.
- Boot from SPI host device connected to SPI0 – The Blackfin processor operates in SPI slave mode and is configured to receive the bytes of the .LDR file from an SPI host (master) agent. To hold off the host device from transmitting while the boot ROM is busy, the Blackfin processor asserts a GPIO pin, called host wait (HWAIT), to signal the host device not to send any more bytes until the flag is deasserted. The flag is chosen by the user and this information is transferred to the Blackfin processor via bits 10:5 of the FLAG header in the .LDR image.

For each of the boot modes, a 10-byte header is first read from an external memory device. The header specifies the number of bytes to be transferred and the memory destination address.

Multiple memory blocks can be loaded by any boot sequence. Once all blocks are loaded, program execution commences from the start of L1 instruction SRAM.

In addition, Bit 4 of the reset configuration register can be set by application code to bypass the normal boot sequence during a software reset. For this case, the processor jumps directly to the beginning of L1 instruction memory.

To augment the boot modes, a secondary software loader is provided that adds additional booting mechanisms. This secondary loader provides the ability to boot from 16-bit flash memory, fast flash, variable baud rate, and other sources. In all boot modes except bypass, program execution starts from on-chip L1 memory address 0xFFA0 0000.

INSTRUCTION SET DESCRIPTION

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

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

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

DEVELOPMENT TOOLS

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

Integrated Development Environments (IDEs)

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

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

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

EZ-KIT Lite Evaluation Board

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

EZ-KIT Lite Evaluation Kits

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

Software Add-Ins for CrossCore Embedded Studio

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

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
Serial Port0			
RSCLK0	I/O	SPORT0 Receive Serial Clock	D
RFS0	I/O	SPORT0 Receive Frame Sync	C
DR0PRI	I	SPORT0 Receive Data Primary	
DR0SEC	I	SPORT0 Receive Data Secondary	
TSCLK0	I/O	SPORT0 Transmit Serial Clock	D
TFS0	I/O	SPORT0 Transmit Frame Sync	C
DT0PRI	O	SPORT0 Transmit Data Primary	C
DT0SEC	O	SPORT0 Transmit Data Secondary	C
Serial Port1			
RSCLK1	I/O	SPORT1 Receive Serial Clock	D
RFS1	I/O	SPORT1 Receive Frame Sync	C
DR1PRI	I	SPORT1 Receive Data Primary	
DR1SEC	I	SPORT1 Receive Data Secondary	
TSCLK1	I/O	SPORT1 Transmit Serial Clock	D
TFS1	I/O	SPORT1 Transmit Frame Sync	C
DT1PRI	O	SPORT1 Transmit Data Primary	C
DT1SEC	O	SPORT1 Transmit Data Secondary	C
Serial Port2			
RSCLK2/*PE0*	I/O	SPORT2 Receive Serial Clock/*GPIO*	D
RFS2/*PE1*	I/O	SPORT2 Receive Frame Sync/*GPIO*	C
DR2PRI/*PE2*	I/O	SPORT2 Receive Data Primary/*GPIO*	C
DR2SEC/*PE3*	I/O	SPORT2 Receive Data Secondary/*GPIO*	C
TSCLK2/*PE4*	I/O	SPORT2 Transmit Serial Clock/*GPIO*	D
TFS2/*PE5*	I/O	SPORT2 Transmit Frame Sync/*GPIO*	C
DT2PRI /*PE6*	I/O	SPORT2 Transmit Data Primary/*GPIO*	C
DT2SEC/*PE7*	I/O	SPORT2 Transmit Data Secondary/*GPIO*	C
Serial Port3			
RSCLK3/*PE8*	I/O	SPORT3 Receive Serial Clock/*GPIO*	D
RFS3/*PE9*	I/O	SPORT3 Receive Frame Sync/*GPIO*	C
DR3PRI/*PE10*	I/O	SPORT3 Receive Data Primary/*GPIO*	C
DR3SEC/*PE11*	I/O	SPORT3 Receive Data Secondary/*GPIO*	C
TSCLK3/*PE12*	I/O	SPORT3 Transmit Serial Clock/*GPIO*	D
TFS3/*PE13*	I/O	SPORT3 Transmit Frame Sync/*GPIO*	C
DT3PRI /*PE14*	I/O	SPORT3 Transmit Data Primary/*GPIO*	C
DT3SEC/*PE15*	I/O	SPORT3 Transmit Data Secondary/*GPIO*	C

Table 10. Pin Descriptions (Continued)

Pin Name	Type	Description	Driver Type[1]
Mode Controls			
$\overline{\text{RESET}}$	I	Reset	
$\overline{\text{NMI}}$	I	Nonmaskable Interrupt (This pin should be pulled high when not used.)	
BMODE1–0	I	Boot Mode Strap (These pins must be pulled to the state required for the desired boot mode.)	
Voltage Regulator			
VROUT1–0	O	External FET Drive 0 (These pins should be left unconnected when not used.)	
Supplies			
V_{DDEXT}	P	I/O Power Supply	
V_{DDINT}	P	Internal Power Supply	
V_{DDRTC}	P	Real-Time Clock Power Supply (This pin should be connected to V_{DDEXT} when not used and should remain powered at all times.)	
MPIVDD	P	MXVR Internal Power Supply	
MXEVDD	P	MXVR External Power Supply	
MXEGND	G	MXVR Ground	
GND	G	Ground	

[1] Refer to Figure 34 on Page 50 to Figure 43 on Page 51.
[2] This pin is 5 V-tolerant when configured as an input and an open-drain when configured as an output; therefore, only the VOL curves in Figure 38 on Page 50 and Figure 39 on Page 51 and the fall time curves in Figure 51 on Page 53 and Figure 52 on Page 53 apply when configured as an output.

ABSOLUTE MAXIMUM RATINGS

Stresses greater than those listed in Table 17 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 17. Absolute Maximum Ratings

Parameter	Rating
Internal (Core) Supply Voltage (V_{DDINT})[1]	–0.3 V to +1.4 V
External (I/O) Supply Voltage (V_{DDEXT})[2]	–0.3 V to +3.8 V
Input Voltage[3, 4]	–0.5 V to +3.8 V
Input Voltage[4, 5]	–0.5 V to +5.5 V
Output Voltage Swing	–0.5 V to V_{DDEXT} + 0.5 V
Junction Temperature While Biased	+125°C
Storage Temperature Range	–65°C to +150°C

[1] Parameter value applies also to MPIVDD.
[2] Parameter value applies also to MXEVDD and V_{DDRTC}.
[3] Applies to 100% transient duty cycle. For other duty cycles, see Table 18.
[4] Applies only when V_{DDEXT} is within specifications. When V_{DDEXT} is outside specifications, the range is V_{DDEXT} ± 0.2 V.
[5] Applies to pins designated as 5 V tolerant only.

Table 18. Maximum Duty Cycle for Input Transient Voltage[1]

V_{IN} Min (V)[2]	V_{IN} Max (V)[2]	Maximum Duty Cycle[3]
–0.50	+3.80	100%
–0.70	+4.00	40%
–0.80	+4.10	25%
–0.90	+4.20	15%
–1.00	+4.30	10%

[1] Applies to all signal pins with the exception of CLKIN, MXI, MXO, MLF, VROUT1–0, XTAL, RTXI, and RTXO.
[2] The individual values cannot be combined for analysis of a single instance of overshoot or undershoot. The worst case observed value must fall within one of the voltages specified and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.
[3] Duty cycle refers to the percentage of time the signal exceeds the value for the 100% case. The is equivalent to the measured duration of a single instance of overshoot or undershoot as a percentage of the period of occurrence.

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.

PACKAGE INFORMATION

The information presented in Figure 10 and Table 19 provides information about how to read the package brand and relate it to specific product features. For a complete listing of product offerings, see the Ordering Guide on Page 60.

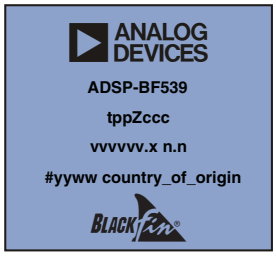


Figure 10. Product Information on Package

Table 19. Package Brand Information[1]

Brand Key	Field Description
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Part
ccc	See Ordering Guide
vvvvvv.xw	Assembly Lot Code
n.n	Silicon Revision
#	RoHS Compliant Designation
yyww	Date Code

[1] Non Automotive only. For branding information specific to Automotive products, contact Analog Devices Inc.

Asynchronous Memory Write Cycle Timing

Table 24 and Table 25 and Figure 15 and Figure 16 describe asynchronous memory write cycle operations for synchronous and for asynchronous ARDY.

Table 24. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

Parameter		Min	Max	Unit
Timing Requirements				
t_{SARDY}	ARDY Setup Before the Falling Edge of CLKOUT	4.0		ns
t_{HARDY}	ARDY Hold After the Falling Edge of CLKOUT	0.0		ns
Switching Characteristics				
t_{DDAT}	DATA15–0 Disable After CLKOUT		6.0	ns
t_{ENDAT}	DATA15–0 Enable After CLKOUT	1.0		ns
t_{DO}	Output Delay After CLKOUT[1]		6.0	ns
t_{HO}	Output Hold After CLKOUT[1]	0.8		ns

[1] Output pins include $\overline{AMS3-0}$, $\overline{ABE1-0}$, ADDR19–1, DATA15–0, $\overline{AOE}$, $\overline{AWE}$.

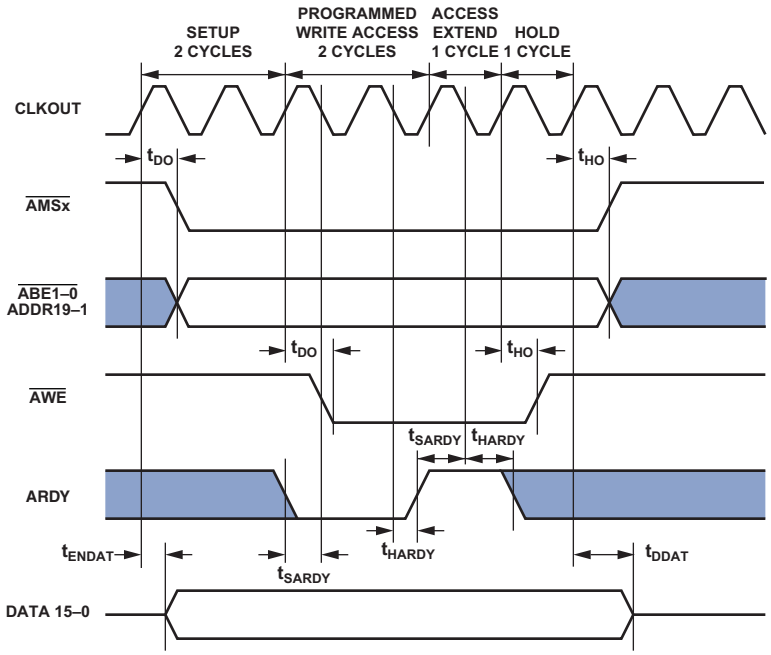


Figure 15. Asynchronous Memory Write Cycle Timing with Synchronous ARDY

Table 28. External Port Bus Request and Grant Cycle Timing with Asynchronous $\overline{BR}$

Parameter		Min	Max	Unit
Timing Requirement				
t_{WBR}	$\overline{BR}$ Pulse Width	$2 \times t_{SCLK}$		ns
Switching Characteristics				
t_{SD}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}/\overline{AWE}$ Disable		4.5	ns
t_{SE}	CLKOUT Low to $\overline{AMSx}$, Address, and $\overline{ARE}/\overline{AWE}$ Enable		4.5	ns
t_{DBG}	CLKOUT High to $\overline{BG}$ High Setup		3.6	ns
t_{EBG}	CLKOUT High to $\overline{BG}$ Deasserted Hold Time		3.6	ns
t_{DBH}	CLKOUT High to $\overline{BGH}$ High Setup		3.6	ns
t_{EBH}	CLKOUT High to $\overline{BGH}$ Deasserted Hold Time		3.6	ns

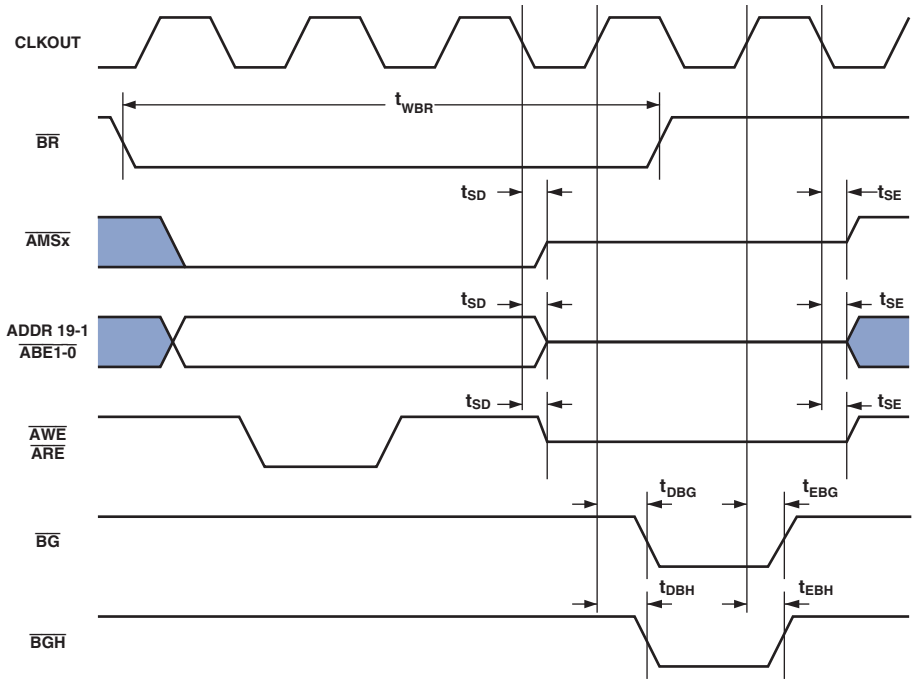


Figure 19. External Port Bus Request and Grant Cycle Timing with Asynchronous $\overline{BR}$

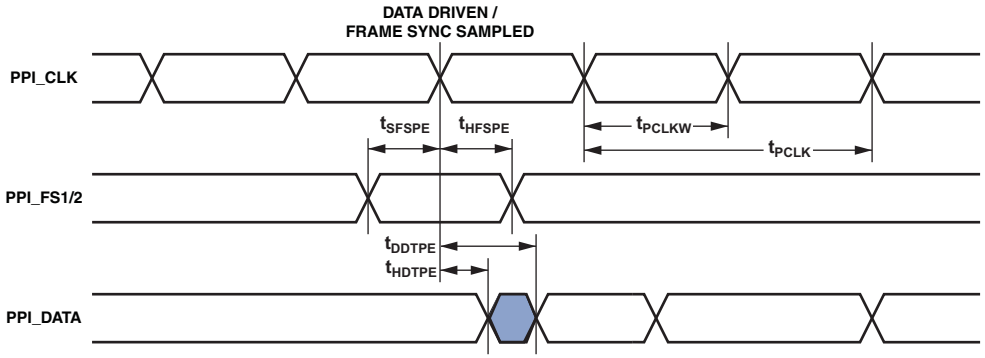


Figure 22. PPI GP Tx Mode with External Frame Sync Timing

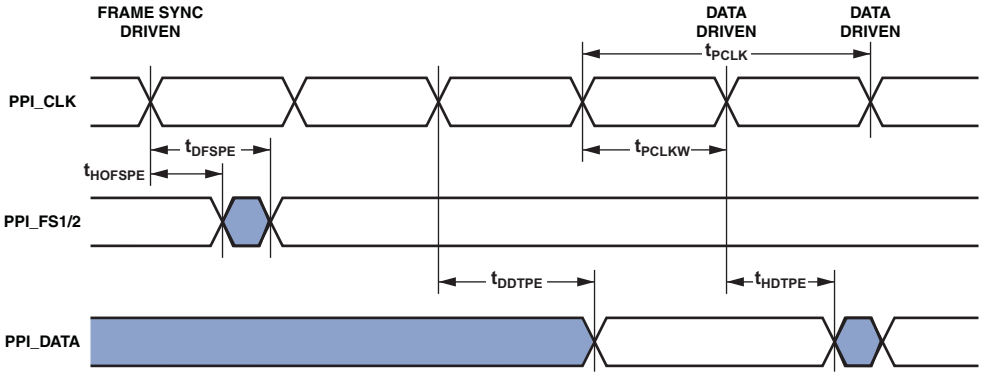


Figure 23. PPI GP Tx Mode with Internal Frame Sync Timing

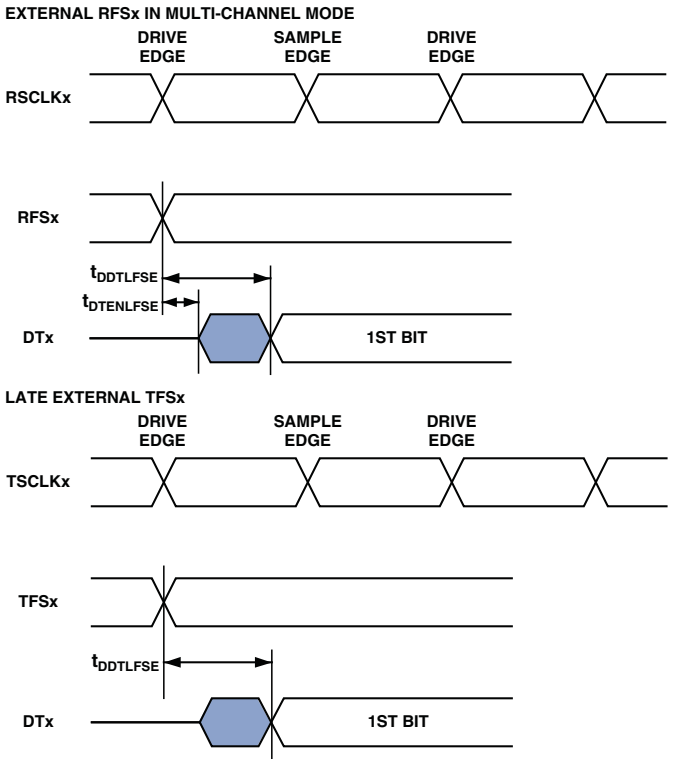


Figure 27. External Late Frame Sync

Timer Clock Timing

Table 39 and Figure 31 describe timer clock timing.

Table 39. Timer Clock Timing

Parameter		Min	Max	Unit
Switching Characteristic				
t_{TODP}	Timer Output Update Delay After PPI_CLK High		12	ns

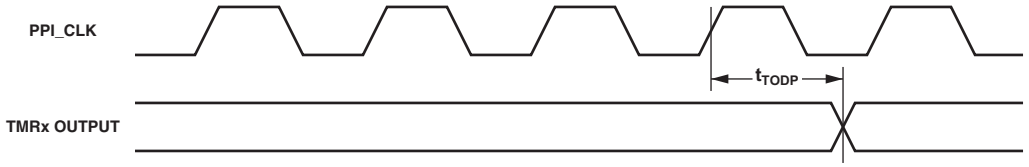


Figure 31. Timer Clock Timing

Timer Cycle Timing

Table 40 and Figure 32 describe timer expired operations. The input signal is asynchronous in "width capture mode" and "external clock mode" and has an absolute maximum input frequency of $f_{SCLK}/2$ MHz.

Table 40. Timer Cycle Timing

Parameter		V_{DDEXT} = 1.8 V Min	V_{DDEXT} = 1.8 V Max	V_{DDEXT} = 2.5 V/3.3 V Min	V_{DDEXT} = 2.5 V/3.3 V Max	Unit
Timing Characteristics						
t_{WL}	Timer Pulse Width Low[1]	$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		ns
t_{WH}	Timer Pulse Width High[1]	$1 \times t_{SCLK}$		$1 \times t_{SCLK}$		ns
t_{TIS}	Timer Input Setup Time Before CLKOUT Low[2]	8.0		6.5		ns
t_{TIH}	Timer Input Hold Time After CLKOUT Low[2]	1.5		1.5		ns
Switching Characteristics						
t_{HTO}	Timer Pulse Width Output	$1 \times t_{SCLK}$	$(2^{32}-1) \times t_{SCLK}$	$1 \times t_{SCLK}$	$(2^{32}-1) \times t_{SCLK}$	ns
t_{TOD}	Timer Output Update Delay After CLKOUT High		7.5		6.5	ns

[1] The minimum pulse widths apply for TMRx input pins in width capture and external clock modes. They also apply to the PF1 or PPI_CLK input pins in PWM output mode.
[2] Either a valid setup and hold time or a valid pulse width is sufficient. There is no need to resynchronize programmable flag inputs.

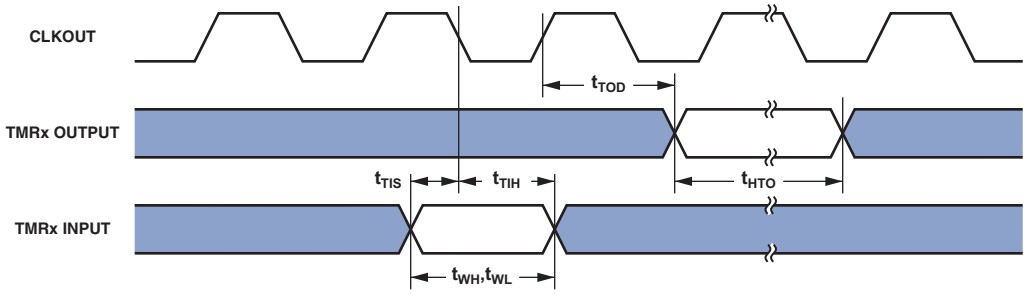


Figure 32. Timer PWM_OUT Cycle Timing

JTAG Test and Emulation Port Timing

Table 41 and Figure 33 describe JTAG port operations.

Table 41. JTAG Port Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{TCK}	TCK Period	20		ns
t_{STAP}	TDI, TMS Setup Before TCK High	4		ns
t_{HTAP}	TDI, TMS Hold After TCK High	4		ns
t_{SSYS}	System Inputs Setup Before TCK High[1]	4		ns
t_{HSYS}	System Inputs Hold After TCK High[1]	6		ns
t_{TRSTW}	$\overline{\text{TRST}}$ Pulse Width[2] (Measured in TCK Cycles)	4		TCK
Switching Characteristics				
t_{DTDO}	TDO Delay from TCK Low		10	ns
t_{DSYS}	System Outputs Delay After TCK Low[3]	0	12	ns

[1] System Inputs = ARDY, BMODE1–0, $\overline{\text{BR}}$, DATA15–0, $\overline{\text{NMI}}$, PF15–0, PPI_CLK, PPI3–0, SCL1–0, SDA1–0, $\overline{\text{MTXON}}$, $\overline{\text{MRXON}}$, MMCLK, MBCLK, MFS, MTX, MRX, $\overline{\text{SPI1SS}}$, $\overline{\text{SPI1SEL1}}$, SCK2–0, MISO2–0, MOSI2–0, $\overline{\text{SPI2SS}}$, $\overline{\text{SPI2SEL1}}$, RX2–0, TX2–1, DR0PRI, DR0SEC, DR1PRI, DR1SEC, DT2PRI, DT2SEC, DR2PRI, DR2SEC, TSCLK3–0, RSCLK3–0, TFS3–0, RFS3–0, DT3PRI, DT3SEC, DR3PRI, DR3SEC, CANTX, CANRX, $\overline{\text{RESET}}$, and TMR2–0.

[2] 50 MHz maximum

[3] System Outputs = $\overline{\text{AMS}}$, $\overline{\text{AOE}}$, $\overline{\text{ARE}}$, $\overline{\text{AWE}}$, $\overline{\text{ABE}}$, $\overline{\text{BG}}$, DATA15–0, PF15–0, PPI3–0, $\overline{\text{MTXON}}$, MMCLK, MBCLK, MFS, MTX, $\overline{\text{SPI1SS}}$, $\overline{\text{SPI1SEL1}}$, SCK2–0, MISO2–0, MOSI2–0, $\overline{\text{SPI2SS}}$, $\overline{\text{SPI2SEL1}}$, RX2–1, TX2–0, DT2PRI, DT2SEC, DR2PRI, DR2SEC, DT3PRI, DT3SEC, DR3PRI, DR3SEC, TSCLK3–0, TFS3–0, RSCLK3–0, RFS3–0, CLKOUT, CANTX, SA10, $\overline{\text{SCAS}}$, SCKE, $\overline{\text{SMS}}$, $\overline{\text{SRAS}}$, $\overline{\text{SWE}}$, and TMR2–0.

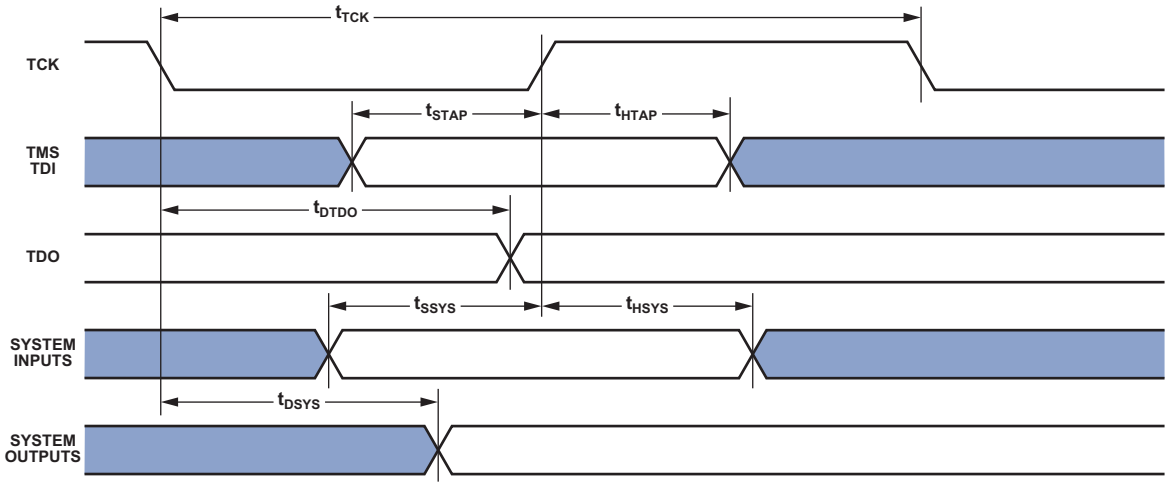


Figure 33. JTAG Port Timing

316-BALL CSP_BGA BALL ASSIGNMENT

Figure 57 lists the top view of the CSP_BGA ball assignment. Figure 58 lists the bottom view of the CSP_BGA ball assignment.

Table 44 on Page 57 lists the CSP_BGA ball assignment by ball number. Table 45 on Page 58 lists the CSP_BGA ball assignment by signal.

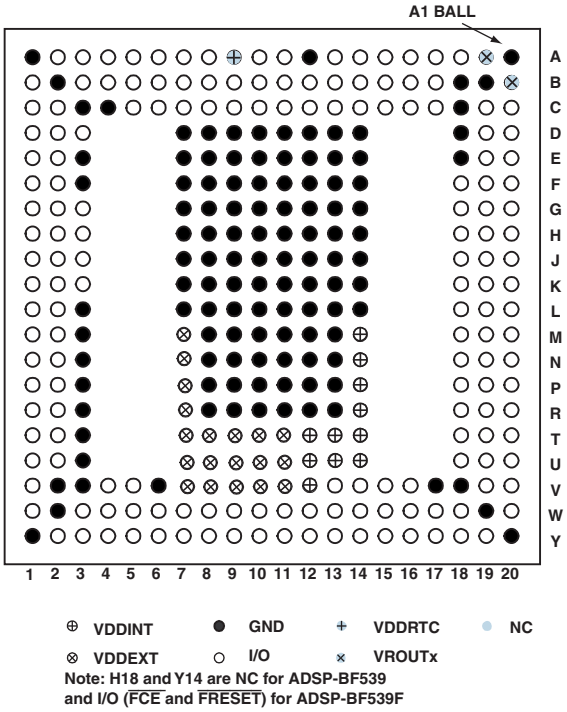


Figure 57. 316-Ball CSP_BGA Ball Assignment (Top View)

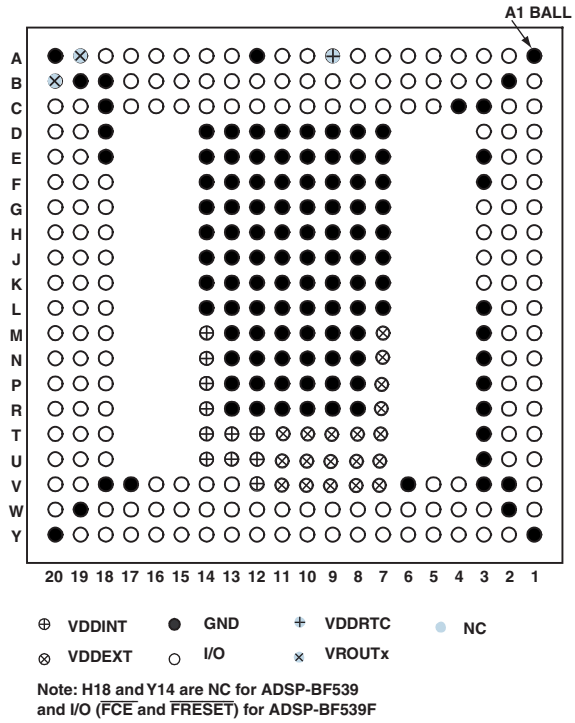


Figure 58. 316-Ball CSP_BGA Ball Assignment (Bottom View)

Table 45. 316-Ball CSP_BGA Ball Assignment (Alphabetically by Signal)

Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.	Signal	Ball No.
$\overline{ABE0}$	M19	DATA8	Y6	GND	E7	GND	K11	GND	V17	PPI2	A6	TSCLK1	H2
$\overline{ABE1}$	M20	DATA9	W6	GND	E8	GND	K12	GND	V18	PPI3	B6	TSCLK2	Y12
ADDR1	N19	DATA10	Y5	GND	E9	GND	K13	GND	W2	$\overline{RESET}$	B14	TSCLK3	L18
ADDR2	N20	DATA11	W5	GND	F8	GND	L13	GND	W19	RFS0	P2	TX0	R1
ADDR3	P19	DATA12	Y4	GND	F9	GND	L14	GND	Y1	RFS1	K1	TX1	C6
ADDR4	P20	DATA13	W4	GND	F10	GND	M3	GND	Y20	RFS2	Y11	TX2	W15
ADDR5	R19	DATA14	Y3	GND	F11	GND	M8	GP	K3	RFS3	T18	V_{DDEXT}	T8
ADDR6	R20	DATA15	W3	GND	F12	GND	M9	MBCLK	D19	RSCLK0	R2	V_{DDEXT}	T9
ADDR7	T19	DR0PRI	N2	GND	F13	GND	M10	MFS	F20	RSCLK1	L1	V_{DDEXT}	T10
ADDR8	T20	DR0SEC	J3	GND	F14	GND	M11	MISO0	F2	RSCLK2	W11	V_{DDEXT}	T11
ADDR9	U19	DR1PRI	J2	GND	G7	GND	M12	MISO1	C14	RSCLK3	U18	V_{DDEXT}	U7
ADDR10	U20	DR1SEC	H3	GND	G8	GND	M13	MISO2	C10	RTXI	A11	V_{DDEXT}	U8
ADDR11	V19	DR2PRI	W12	GND	G9	GND	N3	MLF	A15	RTXO	A10	V_{DDEXT}	U9
ADDR12	V20	DR2SEC	V13	GND	E10	GND	K14	MMCLK	C19	RX0	T1	V_{DDEXT}	U10
ADDR13	W18	DR3PRI	R18	GND	E11	GND	L3	MOSI0	G2	RX1	C5	V_{DDEXT}	U11
ADDR14	W20	DR3SEC	P18	GND	E12	GND	L7	MOSI1	C16	RX2	W14	V_{DDEXT}	V7
ADDR15	W17	DT0PRI	M1	GND	E13	GND	L8	MOSI2	C9	SA10	J20	V_{DDEXT}	M7
ADDR16	Y19	DT0SEC	G3	GND	E14	GND	L9	MPIVDD	C12	$\overline{SCAS}$	H19	V_{DDEXT}	N7
ADDR17	Y18	DT1PRI	H1	GND	E18	GND	L10	$\overline{MRXON}$	A18	SCK0	G1	V_{DDEXT}	P7
ADDR18	W16	DT1SEC	D3	GND	F3	GND	L11	MRX	F19	SCK1	C17	V_{DDEXT}	R7
ADDR19	Y17	DT2PRI	W13	GND	F7	GND	L12	MTX	E19	SCK2	C11	V_{DDEXT}	T7
$\overline{AMS0}$	J18	DT2SEC	V16	GND	G10	GND	N8	$\overline{MTXON}$	B17	SCKE	C20	V_{DDEXT}	V8
$\overline{AMS1}$	K19	DT3PRI	F18	GND	G11	GND	N9	MXEGND	B16	SCL0	B9	V_{DDEXT}	V9
$\overline{AMS2}$	J19	DT3SEC	N18	GND	G12	GND	N10	MXEVDD	B15	SCL1	Y15	V_{DDEXT}	V10
$\overline{AMS3}$	K18	$\overline{EMU}$	T2	GND	G13	GND	N11	MXI	A17	SDA0	B10	V_{DDEXT}	V11
$\overline{AOE}$	K20	$\overline{FCE}$	H18	GND	G14	GND	N12	MXO	A16	SDA1	Y16	V_{DDINT}	M14
ARDY	E20	$\overline{FRESET}$	Y14	GND	H7	GND	N13	$\overline{NMI}$	B13	$\overline{SMS}$	D20	V_{DDINT}	N14
$\overline{ARE}$	L19	GND	A1	GND	H8	GND	P3	PF0	F1	$\overline{SPI1SEL1}$	C13	V_{DDINT}	P14
$\overline{AWE}$	L20	GND	A12	GND	H9	GND	P8	PF1	E1	$\overline{SPI1SS}$	C15	V_{DDINT}	R14
$\overline{BG}$	V14	GND	A20	GND	H10	GND	P9	PF2	E2	$\overline{SPI2SEL1}$	C7	V_{DDINT}	T12
$\overline{BGH}$	V15	GND	B2	GND	H11	GND	P10	PF3	B4	$\overline{SPI2SS}$	C8	V_{DDINT}	T13
BMODE0	V5	GND	B18	GND	H12	GND	P11	PF4	D1	$\overline{SRAS}$	G20	V_{DDINT}	T14
BMODE1	V4	GND	B19	GND	H13	GND	P12	PF5	D2	$\overline{SWE}$	H20	V_{DDINT}	U12
$\overline{BR}$	G18	GND	C3	GND	H14	GND	P13	PF6	C1	TCK	W1	V_{DDINT}	U13
CANRX	B11	GND	C4	GND	J7	GND	R3	PF7	C2	TDI	V1	V_{DDINT}	U14
CANTX	B12	GND	C18	GND	J8	GND	R8	PF8	B1	TDO	Y2	V_{DDINT}	V12
CLKIN	A13	GND	D7	GND	J9	GND	R9	PF9	B3	TFS0	N1	V_{DDRTC}	A9
CLKOUT	G19	GND	D8	GND	J10	GND	R10	PF10	A2	TFS1	J1	VROUT0	B20
DATA0	Y10	GND	D9	GND	J11	GND	R11	PF11	A3	TFS2	Y13	VROUT1	A19
DATA1	W10	GND	D10	GND	J12	GND	R12	PF12	B8	TFS3	M18	XTAL	A14
DATA2	Y9	GND	D11	GND	J13	GND	R13	PF13	A8	TMR0	M2		
DATA3	W9	GND	D12	GND	J14	GND	T3	PF14	B7	TMR1	L2		
DATA4	Y8	GND	D13	GND	K7	GND	U3	PF15	A7	TMR2	K2		
DATA5	W8	GND	D14	GND	K8	GND	V2	PPI_CLK	A4	TMS	U2		
DATA6	Y7	GND	D18	GND	K9	GND	V3	PPI0	A5	$\overline{TRST}$	U1		
DATA7	W7	GND	E3	GND	K10	GND	V6	PPI1	B5	TSCLK0	P1		

OUTLINE DIMENSIONS

Dimensions in the outline dimensions figures are shown in millimeters.

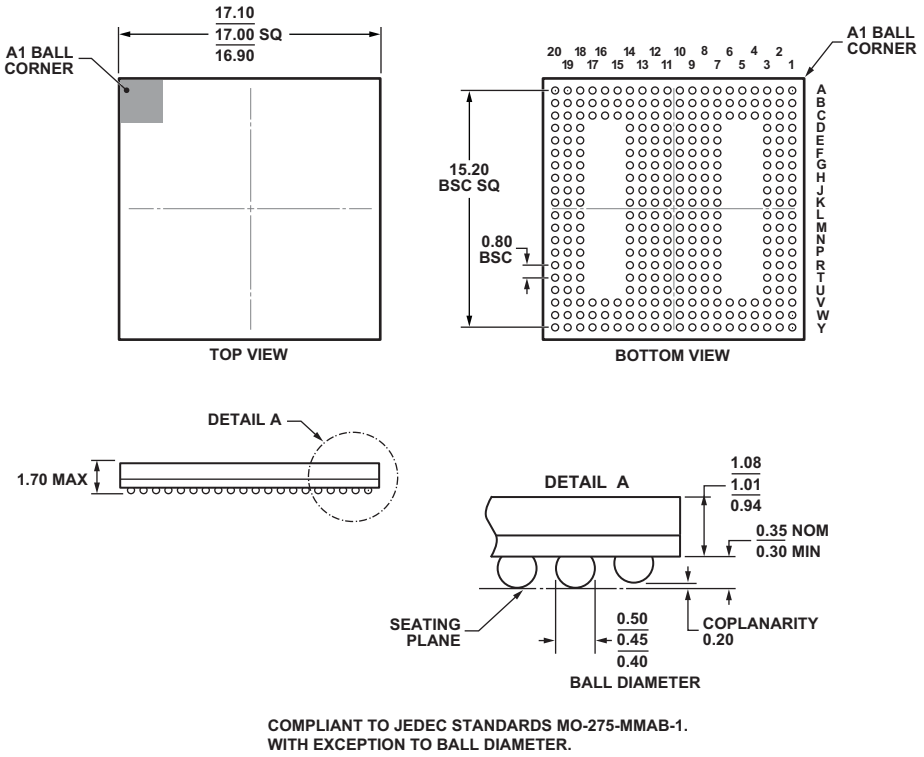


Figure 59. 316-Ball Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-316-2)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 46 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 46. BGA Data for Use with Surface-Mount Design

Package	Package Ball Attach Type	Package Solder Mask Opening	Package Ball Pad Size
316-Ball CSP_BGA (BC-316-2)	Solder Mask Defined	0.40 mm diameter	0.50 mm diameter

ORDERING GUIDE

The models shown 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]	Temperature Range[2]	Instruction Rate (Max)	Flash Memory	Package Description	Package Option
ADBF539WBBCZ4xx	–40°C to +85°C	400 MHz	N/A	316-Ball CSP_BGA	BC-316-2
ADBF539WBBCZ5xx	–40°C to +85°C	533 MHz	N/A	316-Ball CSP_BGA	BC-316-2
ADBF539WBBCZ4F8xx	–40°C to +85°C	400 MHz	8M bit	316-Ball CSP_BGA	BC-316-2
ADBF539WBBCZ5F8xx	–40°C to +85°C	533 MHz	8M bit	316-Ball CSP_BGA	BC-316-2

[1] Z = RoHS compliant part.
[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification. Please see Operating Conditions on Page 26 for junction temperature (T_J) specification which is the only temperature specification.

D06699-0-10/13(F)