Chipsmall

Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Blackfin
Embedded Processor

ADSP-BF534/ADSP-BF536/ADSP-BF537

FEATURES

Up to 600 MHz high performance Blackfin processor
  Two 16-bit MACs, two 40-bit ALUs, four 8-bit video ALUs, 40-bit shifter
  RISC-like register and instruction model for ease of programming and compiler-friendly support
  Advanced debug, trace, and performance monitoring
Wide range of operating voltages (see Operating Conditions on Page 23)
Qualified for Automotive Applications (see Automotive Products on Page 66)
Programmable on-chip voltage regulator
182-ball and 208-ball CSP_BGA packages

MEMORY

Up to 132K bytes of on-chip memory
  Instruction SRAM/cache and instruction SRAM
  Data SRAM/cache plus additional dedicated data SRAM
  Scratchpad SRAM (see Table 1 on Page 3 for available memory configurations)
External memory controller with glueless support for SDRAM and asynchronous 8-bit and 16-bit memories
Flexible booting options from external flash, SPI and TWI memory or from SPI, TWI, and UART host devices
Memory management unit providing memory protection

PERIPHERALS

IEEE 802.3-compliant 10/100 Ethernet MAC (ADSP-BF536 and ADSP-BF537 only)
Controller area network (CAN) 2.0B interface
Parallel peripheral interface (PPI), supporting ITU-R 656 video data formats
2 dual-channel, full-duplex synchronous serial ports (SPORTs), supporting 8 stereo I²S channels
12 peripheral DMAs, 2 mastered by the Ethernet MAC
2 memory-to-memory DMAs with external request lines
Event handler with 32 interrupt inputs
Serial peripheral interface (SPI) compatible
2 UARTs with IrDA support
2-wire interface (TWI) controller
Eight 32-bit timer/counters with PWM support
Real-time clock (RTC) and watchdog timer
32-bit core timer
48 general-purpose I/Os (GPIOs), 8 with high current drivers
On-chip PLL capable of frequency multiplication
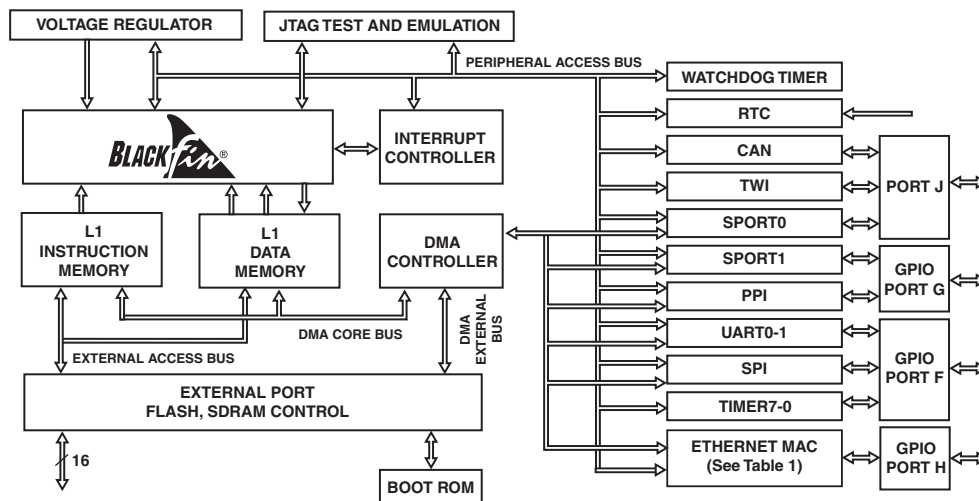**Debug/JTAG interface**

Figure 1. Functional Block Diagram

Blackfin and the Blackfin logo are registered trademarks of Analog Devices, Inc.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106 U.S.A.
Tel: 781.329.4700   ©2014 Analog Devices, Inc. All rights reserved.
Technical Support   www.analog.com

ADSP-BF534/ADSP-BF536/ADSP-BF537

TABLE OF CONTENTS

REVISION HISTORY

2/14—Rev. I to Rev. J

GENERAL DESCRIPTION

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors are members of the Blackfin® family of products, incorporating the Analog Devices, Inc./Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC, state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like microprocessor instruction set, and single-instruction, multiple-data (SIMD) multimedia capabilities into a single instruction-set architecture.

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors are completely code and pin compatible. They differ only with respect to their performance, on-chip memory, and presence of the Ethernet MAC module. Specific performance, memory, and feature configurations are shown in Table 1.

Table 1. Processor Comparison

Features		ADSP-BF534	ADSP-BF536	ADSP-BF537
Ethernet MAC		—	1	1
CAN		1	1	1
TWI		1	1	1
SPORTs		2	2	2
UARTs		2	2	2
SPI		1	1	1
GP Timers		8	8	8
Watchdog Timers		1	1	1
RTC		1	1	1
Parallel Peripheral Interface		1	1	1
GPIOs		48	48	48
Memory Configuration	L1 Instruction SRAM/Cache	16K bytes	16K bytes	16K bytes
	L1 Instruction SRAM	48K bytes	48K bytes	48K bytes
	L1 Data SRAM/Cache	32K bytes	32K bytes	32K bytes
	L1 Data SRAM	32K bytes	—	32K bytes
	L1 Scratchpad	4K bytes	4K bytes	4K bytes
	L3 Boot ROM	2K bytes	2K bytes	2K bytes
Maximum Speed Grade		500 MHz	400 MHz	600 MHz
Package Options: CSP_BGA CSP_BGA		208-Ball 182-Ball	208-Ball 182-Ball	208-Ball 182-Ball

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

PORTABLE LOW POWER ARCHITECTURE

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

SYSTEM INTEGRATION

The Blackfin processor is a highly integrated system-on-a-chip solution for the next generation of embedded network-connected applications. By combining industry-standard interfaces with a high performance signal processing core, cost-effective applications can be developed quickly, without the need for costly external components. The system peripherals include an IEEE-compliant 802.3 10/100 Ethernet MAC (ADSP-BF536 and ADSP-BF537 only), a CAN 2.0B controller, a TWI controller, two UART ports, an SPI port, two serial ports (SPORTs), nine general-purpose 32-bit timers (eight with PWM capability), a real-time clock, a watchdog timer, and a parallel peripheral interface (PPI).

BLACKFIN PROCESSOR PERIPHERALS

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors contain a rich set of peripherals connected to the core via several high bandwidth buses, providing flexibility in system configuration as well as excellent overall system performance (see Figure 1). The processors contain dedicated network communication modules and high speed serial and parallel ports, an interrupt controller for flexible management of interrupts from the on-chip peripherals or external sources, and power management control functions to tailor the performance and power characteristics of the processor and system to many application scenarios.

All of the peripherals, except for the general-purpose I/O, CAN, TWI, real-time clock, and timers, are supported by a flexible DMA structure. There are also separate memory DMA channels dedicated to data transfers between the processor's various memory spaces, including external SDRAM and asynchronous memory. Multiple on-chip buses running at up to 133 MHz provide enough bandwidth to keep the processor core running along with activity on all of the on-chip and external peripherals.

The Blackfin processors include an on-chip voltage regulator in support of the processors' dynamic power management capability. The voltage regulator provides a range of core voltage levels when supplied from V_{DDEXT}. The voltage regulator can be bypassed at the user's discretion.

BLACKFIN PROCESSOR CORE

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

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

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

The ALUs perform a traditional set of arithmetic and logical operations on 16-bit or 32-bit data. In addition, many special instructions are included to accelerate various signal processing tasks. These include bit operations such as field extract and population count, modulo 2^{32} multiply, divide primitives, saturation and rounding, and sign/exponent detection. The set of video instructions 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.

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.
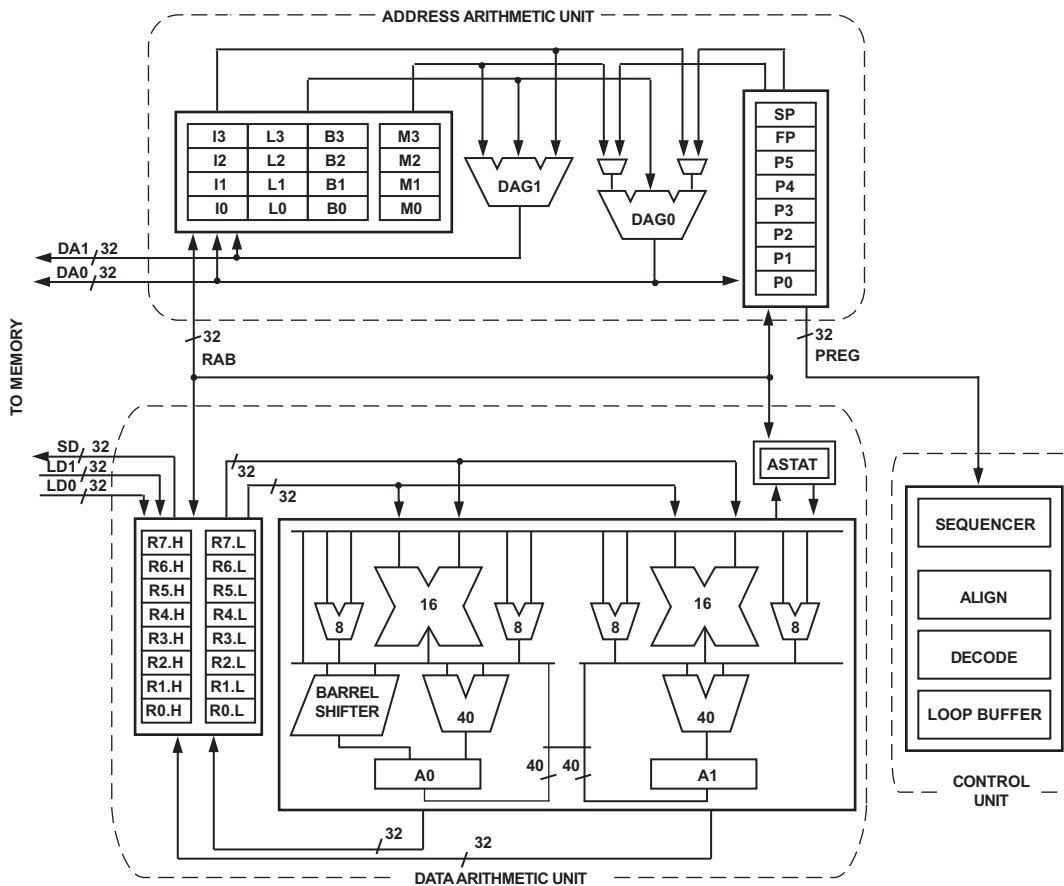

Figure 2. Blackfin Processor Core

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 may be operating on the core and can protect system registers from unintended access.

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

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

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

MEMORY ARCHITECTURE

The ADSP-BF534/ADSP-BF536/ADSP-BF537 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 on-chip L1 memory system is the 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 516M bytes of physical memory.

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

Internal (On-Chip) Memory

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors have three blocks of on-chip memory providing high-bandwidth access to the core.

The first block is the L1 instruction memory, consisting of 64K 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 up to 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 scratchpad 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.

The PC133-compliant SDRAM controller can be programmed to interface to up to 128M bytes of SDRAM. A separate row can be open for each SDRAM internal bank, and the SDRAM controller supports up to 4 internal SDRAM banks, improving overall performance.

The asynchronous memory controller can be programmed to control up to four banks of devices with very flexible timing parameters for a wide variety of devices. Each bank occupies a 1M byte segment regardless of the size of the devices used, so that these banks are only contiguous if each is fully populated with 1M byte of memory.

I/O Memory Space

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors do 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 Blackfin processor contains a small on-chip boot kernel, which configures the appropriate peripheral for booting. If the Blackfin processor is configured to boot from boot ROM

ADSP-BF534/ADSP-BF537 MEMORY MAP

Address	Region
0xFFFF FFFF	CORE MMR REGISTERS (2M BYTES)
0xFFE0 0000	SYSTEM MMR REGISTERS (2M BYTES)
0xFFC0 0000	RESERVED
0xFFB0 1000	SCRATCHPAD SRAM (4K BYTES)
0xFFB0 0000	RESERVED
0xFFA1 4000	INSTRUCTION SRAM/CACHE (16K BYTES)
0xFFA1 0000	RESERVED
0xFFA0 C000	INSTRUCTION BANK B SRAM (16K BYTES)
0xFFA0 8000	INSTRUCTION BANK A SRAM (32K BYTES)
0xFFA0 0000	RESERVED
0xFF90 8000	DATA BANK B SRAM/CACHE (16K BYTES)
0xFF90 4000	DATA BANK B SRAM (16K BYTES)
0xFF90 0000	RESERVED
0xFF80 8000	DATA BANK A SRAM/CACHE (16K BYTES)
0xFF80 4000	DATA BANK A SRAM (16K BYTES)
0xFF80 0000	RESERVED
0xEF00 0800	BOOT ROM (2K BYTES)
0xEF00 0000	RESERVED
0x2040 0000	ASYNC MEMORY BANK 3 (1M BYTES)
0x2030 0000	ASYNC MEMORY BANK 2 (1M BYTES)
0x2020 0000	ASYNC MEMORY BANK 1 (1M BYTES)
0x2010 0000	ASYNC MEMORY BANK 0 (1M BYTES)
0x2000 0000	SDRAM MEMORY (16M BYTES TO 512M BYTES)
0x0000 0000	

(INTERNAL MEMORY MAP; EXTERNAL MEMORY MAP)

ADSP-BF536 MEMORY MAP

Address	Region
0xFFFF FFFF	CORE MMR REGISTERS (2M BYTES)
0xFFE0 0000	SYSTEM MMR REGISTERS (2M BYTES)
0xFFC0 0000	RESERVED
0xFFB0 1000	SCRATCHPAD SRAM (4K BYTES)
0xFFB0 0000	RESERVED
0xFFA1 4000	INSTRUCTION SRAM/CACHE (16K BYTES)
0xFFA1 0000	RESERVED
0xFFA0 C000	INSTRUCTION BANK B SRAM (16K BYTES)
0xFFA0 8000	INSTRUCTION BANK A SRAM (32K BYTES)
0xFFA0 0000	RESERVED
0xFF90 8000	DATA BANK B SRAM/CACHE (16K BYTES)
0xFF90 4000	RESERVED
0xFF90 0000	RESERVED
0xFF80 8000	DATA BANK A SRAM/CACHE (16K BYTES)
0xFF80 4000	RESERVED
0xFF80 0000	RESERVED
0xEF00 0800	BOOT ROM (2K BYTES)
0xEF00 0000	RESERVED
0x2040 0000	ASYNC MEMORY BANK 3 (1M BYTES)
0x2030 0000	ASYNC MEMORY BANK 2 (1M BYTES)
0x2020 0000	ASYNC MEMORY BANK 1 (1M BYTES)
0x2010 0000	ASYNC MEMORY BANK 0 (1M BYTES)
0x2000 0000	SDRAM MEMORY (16M BYTES TO 512M BYTES)
0x0000 0000	

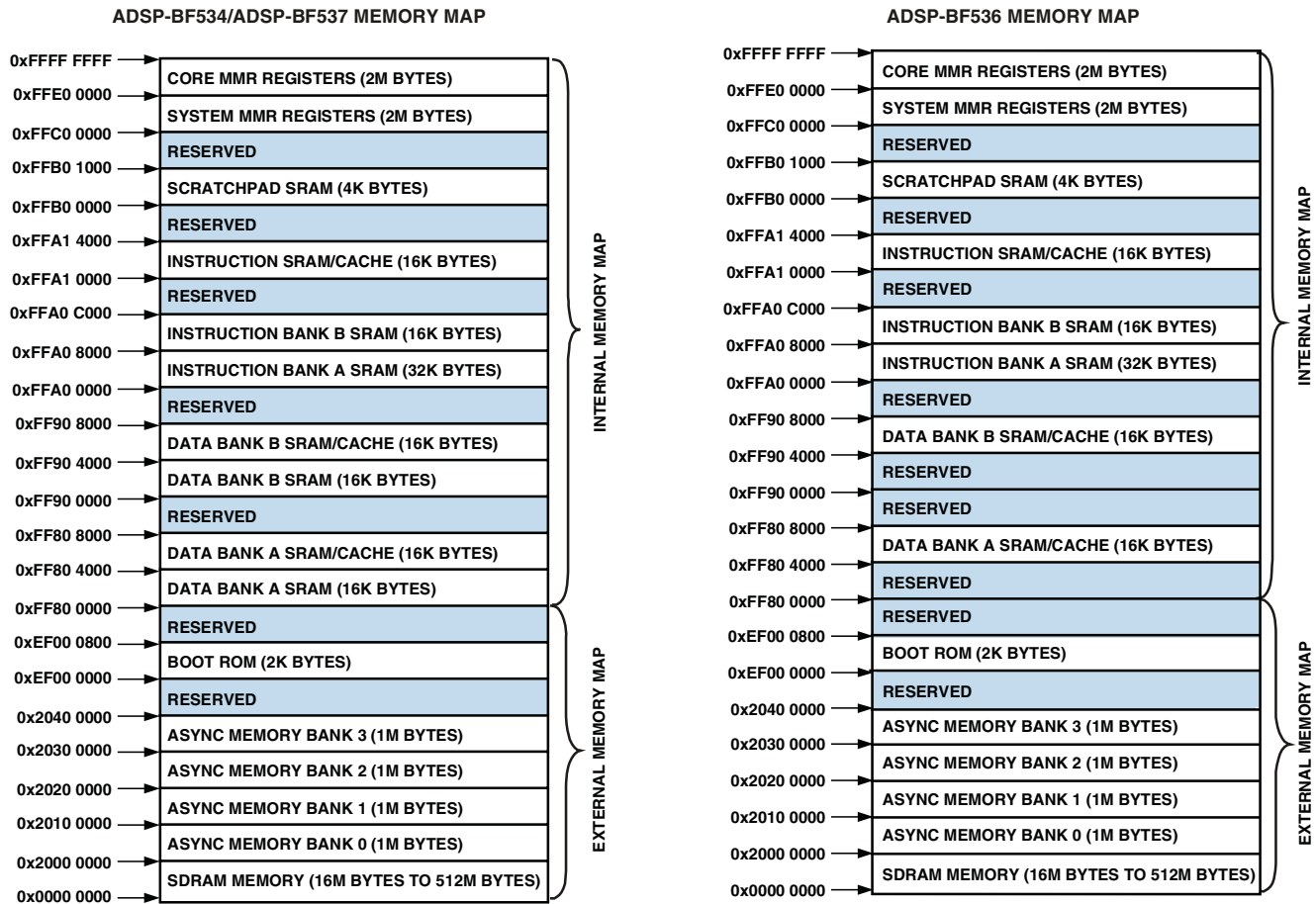(INTERNAL MEMORY MAP; EXTERNAL MEMORY MAP)

Figure 3. ADSP-BF534/ADSP-BF536/ADSP-BF537 Memory Maps

memory space, the processor starts executing from the on-chip boot ROM. For more information, see .

Event Handling

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

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

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

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

Core Event Controller (CEC)

The CEC supports nine general-purpose interrupts (IVG15–7), in addition to the dedicated interrupt and exception events. Of these general-purpose interrupts, the two lowest priority interrupts (IVG15–14) are recommended to be reserved for software interrupt handlers, leaving seven prioritized interrupt inputs to support the peripherals of the Blackfin processor. Table 2 describes the inputs to the CEC, identifies their names in the event vector table (EVT), and lists their priorities.

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-Purpose Interrupt 7	IVG7
8	General-Purpose Interrupt 8	IVG8
9	General-Purpose Interrupt 9	IVG9
10	General-Purpose Interrupt 10	IVG10
11	General-Purpose Interrupt 11	IVG11
12	General-Purpose Interrupt 12	IVG12
13	General-Purpose Interrupt 13	IVG13
14	General-Purpose Interrupt 14	IVG14
15	General-Purpose Interrupt 15	IVG15

System Interrupt Controller (SIC)

The system interrupt controller provides the mapping and routing of events from the many peripheral interrupt sources to the prioritized general-purpose interrupt inputs of the CEC. Although the processor provides a default mapping, the user can alter the mappings and priorities of interrupt events by writing the appropriate values into the interrupt assignment registers (IAR). Table 3 describes the inputs into the SIC and the default mappings into the CEC.

Table 3. System Interrupt Controller (SIC)

Peripheral Interrupt Event	Default Mapping	Peripheral Interrupt ID
PLL Wakeup	IVG7	0
DMA Error (Generic)	IVG7	1
DMAR0 Block Interrupt	IVG7	1
DMAR1 Block Interrupt	IVG7	1
DMAR0 Overflow Error	IVG7	1
DMAR1 Overflow Error	IVG7	1
CAN Error	IVG7	2
Ethernet Error (ADSP-BF536 and ADSP-BF537 only)	IVG7	2
SPORT 0 Error	IVG7	2
SPORT 1 Error	IVG7	2
PPI Error	IVG7	2
SPI Error	IVG7	2
UART0 Error	IVG7	2
UART1 Error	IVG7	2
Real-Time Clock	IVG8	3
DMA Channel 0 (PPI)	IVG8	4
DMA Channel 3 (SPORT 0 Rx)	IVG9	5
DMA Channel 4 (SPORT 0 Tx)	IVG9	6
DMA Channel 5 (SPORT 1 Rx)	IVG9	7
DMA Channel 6 (SPORT 1 Tx)	IVG9	8
TWI	IVG10	9
DMA Channel 7 (SPI)	IVG10	10
DMA Channel 8 (UART0 Rx)	IVG10	11
DMA Channel 9 (UART0 Tx)	IVG10	12
DMA Channel 10 (UART1 Rx)	IVG10	13
DMA Channel 11 (UART1 Tx)	IVG10	14
CAN Rx	IVG11	15
CAN Tx	IVG11	16
DMA Channel 1 (Ethernet Rx, ADSP-BF536 and ADSP-BF537 only)	IVG11	17
Port H Interrupt A	IVG11	17
DMA Channel 2 (Ethernet Tx, ADSP-BF536 and ADSP-BF537 only)	IVG11	18
Port H Interrupt B	IVG11	18
Timer 0	IVG12	19
Timer 1	IVG12	20
Timer 2	IVG12	21
Timer 3	IVG12	22
Timer 4	IVG12	23
Timer 5	IVG12	24
Timer 6	IVG12	25
Timer 7	IVG12	26
Port F, G Interrupt A	IVG12	27
Port G Interrupt B	IVG12	28

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

Peripheral Interrupt Event	Default Mapping	Peripheral Interrupt ID
DMA Channels 12 and 13 (Memory DMA Stream 0)	IVG13	29
DMA Channels 14 and 15 (Memory DMA Stream 1)	IVG13	30
Software Watchdog Timer	IVG13	31
Port F Interrupt B	IVG13	31

Event Control

The Blackfin processor provides a very flexible mechanism to control the processing of events. In the CEC, three registers are used to coordinate and control events. Each register is 32 bits wide:

- CEC interrupt latch register (ILAT) – Indicates when events have been latched. The appropriate bit is set when the processor has latched the event and cleared when the event has been accepted into the system. This register is updated automatically by the controller, but it can be written only when its corresponding IMASK bit is cleared.

- CEC interrupt mask register (IMASK) – Controls the masking and unmasking of individual events. When a bit is set in the IMASK register, that event is unmasked and is processed by the CEC when asserted. A cleared bit in the IMASK register masks the event, preventing the processor from servicing the event even though the event may be latched in the ILAT register. This register can be read or written while in supervisor mode. (Note that general-purpose interrupts can be globally enabled and disabled with the STI and CLI instructions, respectively.)

- CEC interrupt pending register (IPEND) – The IPEND register keeps track of all nested events. A set bit in the IPEND register indicates the event is currently active or nested at some level. This register is updated automatically by the controller but can be read while in supervisor mode.

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

- SIC interrupt mask register (SIC_IMASK) – Controls the masking and unmasking of each peripheral interrupt event. When a bit is set in the register, that peripheral event is unmasked and is processed by the system when asserted. A cleared bit in the register masks the peripheral event, preventing the processor from servicing the event.

- SIC interrupt status register (SIC_ISR) – As multiple peripherals can be mapped to a single event, this register allows the software to determine which peripheral event source triggered the interrupt. A set bit indicates the peripheral is asserting the interrupt, and a cleared bit indicates the peripheral is not asserting the event.

- SIC interrupt wake-up enable register (SIC_IWR) – By enabling the corresponding bit in this register, a peripheral can be configured to wake up the processor, should the core be idled when the event is generated. (For more information, see Dynamic Power Management on Page 13.)

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

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

DMA CONTROLLERS

The Blackfin processors have multiple, independent DMA channels that support automated data transfers with minimal overhead for the processor core. DMA transfers can occur between the processor's internal memories and any of its DMA-capable peripherals. 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 Ethernet MAC (ADSP-BF536 and ADSP-BF537 only), SPORTs, SPI port, UARTs, and PPI. Each individual DMA-capable peripheral has at least one dedicated DMA channel.

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

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

Examples of DMA types supported by the DMA controller include

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

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

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors also have an external DMA controller capability via dual external DMA request pins when used in conjunction with the external bus interface unit (EBIU). This functionality can be used when a high speed interface is required for external FIFOs and high bandwidth communications peripherals such as USB 2.0. It allows control of the number of data transfers for memDMA. The number of transfers per edge is programmable. This feature can be programmed to allow memDMA to have an increased priority on the external bus relative to the core.

REAL-TIME CLOCK

The 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 processor. 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, while 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 the hibernate operating mode.

Connect RTC pins RTXI and RTXO with external components as shown in Figure 4.

WATCHDOG TIMER

The ADSP-BF534/ADSP-BF536/ADSP-BF537 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 system reset, nonmaskable interrupt (NMI), or
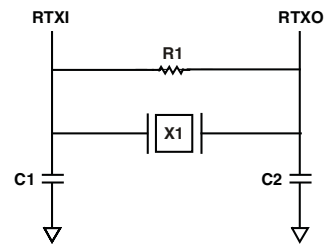
SUGGESTED COMPONENTS:
X1 = ECLIPTEK EC38J (THROUGH-HOLE PACKAGE) OR
    EPSON MC405 12 pF LOAD (SURFACE-MOUNT PACKAGE)
C1 = 22 pF
C2 = 22 pF
R1 = 10 MΩ
NOTE: C1 AND C2 ARE SPECIFIC TO CRYSTAL SPECIFIED FOR X1.
CONTACT CRYSTAL MANUFACTURER FOR DETAILS. C1 AND C2
SPECIFICATIONS ASSUME BOARD TRACE CAPACITANCE OF 3 pF.

Figure 4. External Components for RTC

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.

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 nine general-purpose programmable timer units in the processor. Eight 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 an external clock input to the several other associated PF pins, to an external clock input to the PPI_CLK input pin, or to the internal SCLK.

The timer units can be used in conjunction with the two 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, either to the system clock or to a count of external signals.

In addition to the eight general-purpose programmable timers, a ninth timer is also provided. This extra timer is clocked by the internal processor clock and is typically used as a system tick clock for generating periodic interrupts in an operating system.

SERIAL PORTS (SPORTs)

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors incorporate two dual-channel synchronous serial ports (SPORT0 and SPORT1) for serial and multiprocessor communications. The SPORTs support the following features:

- I^{2}S capable operation.

- Bidirectional operation – Each SPORT has two sets of independent transmit and receive pins, enabling eight channels of I^{2}S stereo audio.

- Buffered (8-deep) transmit and receive ports – Each port has a data register for transferring data words to and from other processor components and shift registers for shifting data in and out of the data registers.

- Clocking – Each transmit and receive port can either use an external serial clock or generate its own, in frequencies ranging from ($f_{SCLK}/131,070$) Hz to ($f_{SCLK}/2$) Hz.

- Word length – Each SPORT supports serial data words from 3 bits to 32 bits in length, transferred most significant bit first or least significant bit first.

- Framing – Each transmit and receive port can run with or without frame sync signals for each data word. Frame sync signals can be generated internally or externally, active high or low, and with either of two pulse widths and early or late frame sync.

- Companding in hardware – Each SPORT can perform A-law or µ-law companding according to ITU recommendation G.711. Companding can be selected on the transmit and/or receive channel of the SPORT without additional latencies.

- DMA operations with single-cycle overhead – Each SPORT can automatically receive and transmit multiple buffers of memory data. The processor can link or chain sequences of DMA transfers between a SPORT and memory.

- Interrupts – Each transmit and receive port generates an interrupt upon completing the transfer of a data word or after transferring an entire data buffer, or buffers, through DMA.

- Multichannel capability – Each SPORT supports 128 channels out of a 1024-channel window and is compatible with the H.100, H.110, MVIP-90, and HMVIP standards.

SERIAL PERIPHERAL INTERFACE (SPI) PORT

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors have an SPI-compatible port that enables the processor to communicate with multiple SPI-compatible devices.

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, SCK). An SPI chip select input pin ($\overline{SPISS}$) lets other SPI devices select the processor, and seven SPI chip select output pins ($\overline{SPISEL7-1}$) let the processor select other SPI devices. The SPI select pins are reconfigured programmable flag pins. Using these pins, the SPI

port provides a full-duplex, synchronous serial interface, which supports both master/slave modes and multimaster environments.

The SPI port's baud rate and clock phase/polarities are programmable, and it has an integrated DMA controller, configurable to support transmit or receive data streams. The SPI's DMA controller can only service unidirectional accesses at any given time.

The SPI port's clock rate is calculated as:

$$SPI\ Clock\ Rate = \frac{f_{SCLK}}{2 \times SPI\_BAUD}$$

where the 16-bit SPI_BAUD register contains a value of 2 to 65,535.

During transfers, the SPI port simultaneously transmits and receives by serially shifting data in and out on its two serial data lines. The serial clock line synchronizes the shifting and sampling of data on the two serial data lines.

UART PORTS

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors provide two full-duplex universal asynchronous receiver and 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, one or two stop bits, and none, even, or odd parity. Each UART port supports two modes of operation:

- PIO (programmed I/O) – The processor sends or receives data by writing or reading I/O mapped UART registers. The data is double-buffered on both transmit and receive.

- DMA (direct memory access) – The DMA controller transfers both transmit and receive data. This reduces the number and frequency of interrupts required to transfer data to and from memory. The UART has two dedicated DMA channels, one for transmit and one for receive. These DMA channels have lower default priority than most DMA channels because of their relatively low service rates.

Each UART port's baud rate, serial data format, error code generation and status, and interrupts are programmable:

- Supporting bit rates ranging from ($f_{SCLK}/1,048,576$) to ($f_{SCLK}/16$) bits per second.

- Supporting data formats from 7 bits to 12 bits per frame.

- Both transmit and receive operations can be configured to generate maskable interrupts to the processor.

The UART port's clock rate is calculated as:

$$UART\ Clock\ Rate = \frac{f_{SCLK}}{16 \times UARTx\_Divisor}$$

where the 16-bit *UARTx_Divisor* comes from the UARTx_DLH register (most significant 8 bits) and UARTx_DLL register (least significant 8 bits).

In conjunction with the general-purpose timer functions, auto-baud detection is supported.

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

CONTROLLER AREA NETWORK (CAN)

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors offer a CAN controller that is a communication controller implementing 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, since the protocol incorporates CRC checking message error tracking, and fault node confinement.

The CAN controller offers the following features:

- 32 mailboxes (eight receive only, eight 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 wake-up from hibernation mode (lowest static power consumption mode).
- Interrupts, including: Tx complete, Rx complete, error, global.

The electrical characteristics of each network connection are very demanding so the CAN interface is typically divided into two parts: a controller and a transceiver. This allows a single controller to support different drivers and CAN networks. The CAN module represents only the controller part of the interface. The controller interface supports connection to 3.3 V high-speed, fault-tolerant, single-wire transceivers.

TWI CONTROLLER INTERFACE

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors include a 2-wire interface (TWI) module for providing a simple exchange method of control data between multiple devices. The TWI is compatible with the widely used I²C® bus standard. The TWI module offers the capabilities of simultaneous master and slave operation, support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (SCL) and data (SDA) and supports the protocol at speeds up to 400 kbps. The TWI interface pins are compatible with 5 V logic levels.

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

10/100 ETHERNET MAC

The ADSP-BF536 and ADSP-BF537 processors offer the capability to directly connect to a network by way of an embedded fast Ethernet Media Access Controller (MAC) that supports both 10-BaseT (10 Mbps) and 100-BaseT (100 Mbps) operation. The 10/100 Ethernet MAC peripheral 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 of MII and RMII protocols for external PHYs.
- Full duplex and half duplex modes.
- Data framing and encapsulation: generation and detection of preamble, length padding, and FCS.
- Media access management (in half-duplex operation): collision and contention handling, including control of retransmission of collision frames and of back-off timing.
- Flow control (in full-duplex operation): generation and detection of PAUSE frames.
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers.
- SCLK operating range down to 25 MHz (active and sleep operating modes).
- Internal loopback from Tx to Rx.

Some advanced features are

- Buffered crystal output to external PHY for support of a single crystal system.
- Automatic checksum computation of IP header and IP payload fields of Rx frames.
- Independent 32-bit descriptor-driven Rx and Tx DMA channels.
- Frame status delivery to memory via 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 support even 32-bit alignment of encapsulated Rx or Tx IP packet data in memory after the 14-byte MAC header.
- Programmable Ethernet event interrupt supports any combination of
  - Any selected Rx or Tx frame status conditions.
  - PHY interrupt condition.
  - Wake-up frame detected.
  - Any selected MAC management counter(s) at half-full.
  - DMA descriptor error.
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value.

- Programmable Rx address filters, including a 64-bit address hash table for multicast and/or unicast frames, and programmable filter modes for broadcast, multicast, unicast, control, and damaged frames.

- Advanced power management supporting unattended transfer of Rx and Tx frames and status to/from external memory via DMA during low power sleep mode.

- System wake-up from sleep operating mode upon magic packet or any of four user-definable wake-up frame filters.

- Support for 802.3Q tagged VLAN frames.

- Programmable MDC clock rate and preamble suppression.

- In RMII operation, 7 unused pins can be configured as GPIO pins for other purposes.

PORTS

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors group the many peripheral signals to four ports—Port F, Port G, Port H, and Port J. Most of the associated pins are shared by multiple signals. The ports function as multiplexer controls. Eight of the pins (Port F7–0) offer high source/high sink current capabilities.

General-Purpose I/O (GPIO)

The processors have 48 bidirectional, general-purpose I/O (GPIO) pins allocated across three separate GPIO modules—PORTFIO, PORTGIO, and PORTHIO, associated with Port F, Port G, and Port H, respectively. Port J does not provide GPIO functionality. Each GPIO-capable pin shares functionality with other processor peripherals via a multiplexing scheme; however, the GPIO functionality is the default state of the device upon power-up. Neither GPIO output or input drivers are active by default. Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers:

- GPIO direction control register – Specifies the direction of each individual GPIO pin as input or output.

- GPIO control and status registers – The processors employ a "write one to modify" mechanism that allows any combination of individual GPIO pins to be modified in a single instruction, without affecting the level of any other GPIO pins. Four control registers are provided. One register is written in order to set pin values, one register is written in order to clear pin values, one register is written in order to toggle pin values, and one register is written in order to specify a pin value. Reading the GPIO status register allows software to interrogate the sense of the pins.

- GPIO interrupt mask registers – The two GPIO interrupt mask registers allow each individual GPIO pin to function as an interrupt to the processor. Similar to the two GPIO control registers that are used to set and clear individual pin values, one GPIO interrupt mask register sets bits to enable interrupt function, and the other GPIO interrupt mask register clears bits to disable interrupt function.

GPIO pins defined as inputs can be configured to generate hardware interrupts, while output pins can be triggered by software interrupts.

- GPIO interrupt sensitivity registers – The two GPIO interrupt sensitivity registers specify whether individual pins are level- or edge-sensitive and specify—if edge-sensitive—whether just the rising edge or both the rising and falling edges of the signal are significant. One register selects the type of sensitivity, and one register selects which edges are significant for edge-sensitivity.

PARALLEL PERIPHERAL INTERFACE (PPI)

The processor provides a parallel peripheral interface (PPI) that can connect directly to parallel ADC and DAC converters, video encoders and decoders, and other general-purpose peripherals. The PPI consists of a dedicated input clock pin, up to three frame synchronization pins, and up to 16 data pins. The input clock supports parallel data rates up to half the system clock rate and the synchronization signals can be configured as either inputs or outputs.

The PPI supports a variety of general-purpose and ITU-R 656 modes of operation. In general-purpose mode, the PPI provides half-duplex, bidirectional data transfer with up to 16 bits of data. Up to three frame synchronization signals are also provided. In ITU-R 656 mode, the PPI provides half-duplex bidirectional transfer of 8- or 10-bit video data. Additionally, on-chip decode of embedded start-of-line (SOL) and start-of-field (SOF) preamble packets is supported.

General-Purpose Mode Descriptions

The general-purpose modes of the PPI are intended to suit a wide variety of data capture and transmission applications. Three distinct submodes are supported:

1. Input mode – Frame syncs and data are inputs into the PPI.

2. Frame capture mode – Frame syncs are outputs from the PPI, but data are inputs.

3. Output mode – Frame syncs and data are outputs from the PPI.

Input Mode

Input mode is intended for ADC applications, as well as video communication with hardware signaling. In its simplest form, PPI_FS1 is an external frame sync input that controls when to read data. The PPI_DELAY MMR allows for a delay (in PPI_CLK cycles) between reception of this frame sync and the initiation of data reads. The number of input data samples is user programmable and defined by the contents of the PPI_COUNT register. The PPI supports 8-bit and 10-bit through 16-bit data, programmable in the PPI_CONTROL register.

Frame Capture Mode

Frame capture mode allows the video source(s) to act as a slave (for frame capture for example). The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors control when to read from the video source(s). PPI_FS1 is an HSYNC output and PPI_FS2 is a VSYNC output.

Output Mode

Output mode is used for transmitting video or other data with up to three output frame syncs. Typically, a single frame sync is appropriate for data converter applications, whereas two or three frame syncs could be used for sending video with hardware signaling.

ITU-R 656 Mode Descriptions

The ITU-R 656 modes of the PPI are intended to suit a wide variety of video capture, processing, and transmission applications. Three distinct submodes are supported:

1. Active video only mode
2. Vertical blanking only mode
3. Entire field mode

Active Video Mode

Active video only mode is used when only the active video portion of a field is of interest and not any of the blanking intervals. The PPI does not read in any data between the end of active video (EAV) and start of active video (SAV) preamble symbols, or any data present during the vertical blanking intervals. In this mode, the control byte sequences are not stored to memory; they are filtered by the PPI. After synchronizing to the start of Field 1, the PPI ignores incoming samples until it sees an SAV code. The user specifies the number of active video lines per frame (in PPI_COUNT register).

Vertical Blanking Interval Mode

In this mode, the PPI only transfers vertical blanking interval (VBI) data.

Entire Field Mode

In this mode, the entire incoming bit stream is read in through the PPI. This includes active video, control preamble sequences, and ancillary data that may be embedded in horizontal and vertical blanking intervals. Data transfer starts immediately after synchronization to Field 1. Data is transferred to or from the synchronous channels through eight DMA engines that work autonomously from the processor core.

DYNAMIC POWER MANAGEMENT

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors provide five operating modes, each with a different performance and power profile. In addition, dynamic power management provides the control functions to dynamically alter the processor core supply voltage, further reducing power dissipation. Control of clocking to each of the peripherals also reduces power consumption. See Table 4 for a summary of the power settings for each mode. Also, see Table 16, Table 15 and Table 17.

Full-On Operating Mode—Maximum Performance

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

Active Operating Mode—Moderate Dynamic Power Savings

In the active mode, the PLL is enabled but bypassed. Because the PLL is bypassed, the processor's core clock (CCLK) and system clock (SCLK) run at the input clock (CLKIN) frequency. In this mode, the CLKIN to CCLK multiplier ratio can be changed, although the changes are not realized until the full-on mode is entered. DMA access is available to appropriately configured L1 memories.

In the active mode, it is possible to disable the PLL through the PLL control register (PLL_CTL). If disabled, the PLL must be re-enabled before transitioning to the full-on or sleep modes.

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, asserting wake-up causes the processor to sense the value of the BYPASS bit in the PLL control register (PLL_CTL). If BYPASS is disabled, the processor transitions to the full on mode. If BYPASS is enabled, the processor transitions to the active mode.

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

Table 4. Power Settings

Mode	PLL	PLL Bypassed	Core Clock (CCLK)	System Clock (SCLK)	Internal Power (V$_{DDINT}$)
Full On	Enabled	No	Enabled	Enabled	On
Active	Enabled/ Disabled	Yes	Enabled	Enabled	On
Sleep	Enabled	—	Disabled	Enabled	On
Deep Sleep	Disabled	—	Disabled	Disabled	On
Hibernate	Disabled	—	Disabled	Disabled	Off

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 cannot access internal resources or external memory. This powered-down mode can only be exited by assertion of the reset interrupt ($\overline{RESET}$) 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 $\overline{RESET}$ while in deep sleep mode causes the processor to transition to the full-on mode.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor core (CCLK) and to all of the 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 disables both CCLK and SCLK. Furthermore, it sets the internal power supply voltage (V_{DDINT}) to 0 V to provide the greatest power savings. To preserve the processor state, prior to removing power, any critical information stored internally (memory contents, register contents, etc.) must be written to a nonvolatile storage device.

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

The Ethernet or CAN modules can wake up the internal supply regulator. If the PH6 pin does not connect as the $\overline{\text{PHYINT}}$ signal to an external PHY device, it can be pulled low by any other device to wake the processor up. The regulator can also be woken up by a real-time clock wake-up event or by asserting the $\overline{\text{RESET}}$ pin. All hibernate wake-up events initiate the hardware reset sequence. Individual sources are enabled by the VR_CTL register.

With the exception of the VR_CTL and the RTC registers, all internal registers and memories lose their content in the hibernate state. State variables can be held in external SRAM or SDRAM. The SCKELOW bit in the VR_CTL register provides a means of waking from hibernate state without disrupting a self-refreshing SDRAM, provided that there is also an external pull-down on the SCKE pin.

Power Savings

As shown in Table 5, the processors support three different power domains which maximizes flexibility, while maintaining compliance with industry standards and conventions. By isolating the internal logic of the processor into its own power domain, separate from the RTC and other I/O, the processor can take advantage of dynamic power management, without affecting the RTC or other I/O devices. There are no sequencing requirements for the various power domains.

Table 5. Power Domains

Power Domain	V_{DD} Range
All internal logic, except RTC	V_{DDINT}
RTC internal logic and crystal I/O	V_{DDRTC}
All other I/O	V_{DDEXT}

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

The power dissipated by a processor is largely a function of its clock frequency and the square of the operating voltage. For example, reducing the clock frequency by 25% results in a 25% reduction in power dissipation, while reducing the voltage by 25% reduces power dissipation by more than 40%. Further,

these power savings are additive, in that if the clock frequency and supply voltage are both reduced, the power savings can be dramatic, as shown in the following equations.

The power savings factor (PSF) is calculated as:

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

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

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

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

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

The percent power savings is calculated as

$$\% \text{ power savings } = (1 - PSF) \times 100\%$$

VOLTAGE REGULATION

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors provide an on-chip voltage regulator that can generate appropriate V_{DDINT} voltage levels from the V_{DDEXT} supply. See Operating Conditions on Page 23 for regulator tolerances and acceptable V_{DDEXT} ranges for specific models.
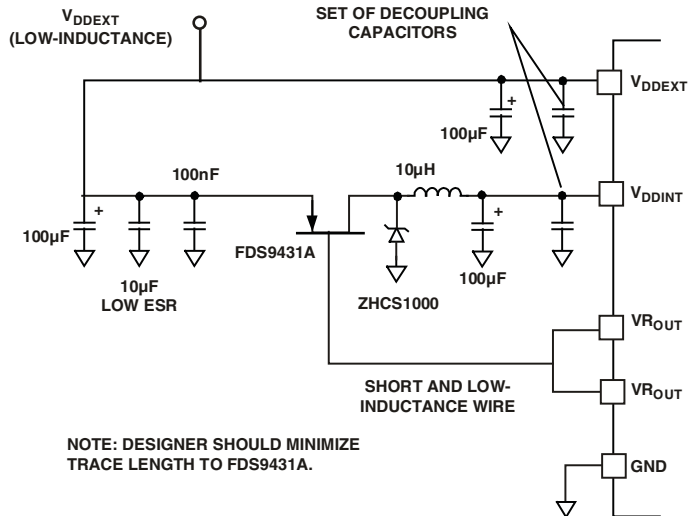
Figure 5. Voltage Regulator Circuit

Figure 5 shows the typical external components required to complete the power management system. The regulator controls the internal logic voltage levels and is programmable with the voltage regulator control register (VR_CTL) in increments of 50 mV. To reduce standby power consumption, the internal voltage regulator can be programmed to remove power to the processor core while keeping I/O power supplied. While in

hibernate state, V_{DDEXT} can still be applied, eliminating the need for external buffers. The voltage regulator can be activated from this power-down state by asserting the $\overline{RESET}$ pin, which then initiates a boot sequence. The regulator can also be disabled and bypassed at the user's discretion. For additional information on voltage regulation, see *Switching Regulator Design Considerations for the ADSP-BF533 Blackfin Processors (EE-228)*.

CLOCK SIGNALS

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors can be clocked by an external crystal, a sine wave input, or a buffered, shaped clock derived from an external clock oscillator.

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

Alternatively, because the processors include an on-chip oscillator circuit, an external crystal can 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 CLKIN and XTAL pins. The on-chip resistance between CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended. 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 of multiple devices over temperature range.
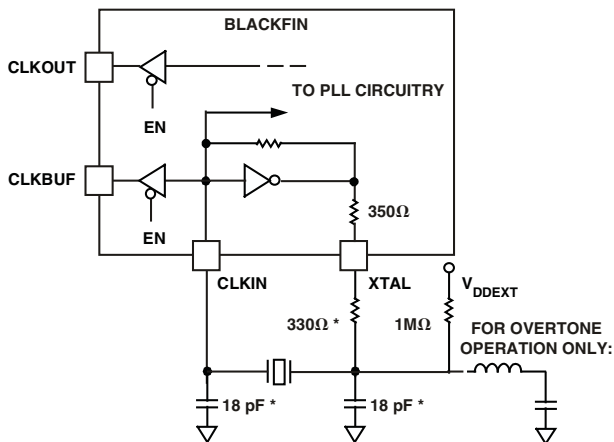
NOTE: VALUES MARKED WITH * MUST BE CUSTOMIZED, DEPENDING ON THE CRYSTAL AND LAYOUT. PLEASE ANALYZE CAREFULLY.

Figure 6. External Crystal Connections

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 the application note *Using Third Overtone Crystals with the ADSP-218x DSP (EE-168)*.

The CLKBUF pin is an output pin, and is a buffer version of the input clock. This pin is particularly useful in Ethernet applications to limit the number of required clock sources in the system. In this type of application, a single 25 MHz or 50 MHz crystal can be applied directly to the processors. The 25 MHz or 50 MHz output of CLKBUF can then be connected to an external Ethernet MII or RMII PHY device.

Because of the default 10× PLL multiplier, providing a 50 MHz CLKIN exceeds the recommended operating conditions of the lower speed grades. Because of this restriction, an RMII PHY requiring a 50 MHz clock input cannot be clocked directly from the CLKBUF pin for the lower speed grades. In this case, either provide a separate 50 MHz clock source, or use an RMII PHY with 25 MHz clock input options. The CLKBUF output is active by default and can be disabled using the VR_CTL register for power savings.

The Blackfin core runs at a different clock rate than the on-chip peripherals. As shown in Figure 7, the core clock (CCLK) and system peripheral clock (SCLK) are derived from the input clock (CLKIN) signal. An on-chip PLL is capable of multiplying the CLKIN signal by a programmable 0.5× to 64× multiplication factor (bounded by specified minimum and maximum VCO frequencies). The default multiplier is 10×, but it can be modified by a software instruction sequence in the PLL_CTL register.
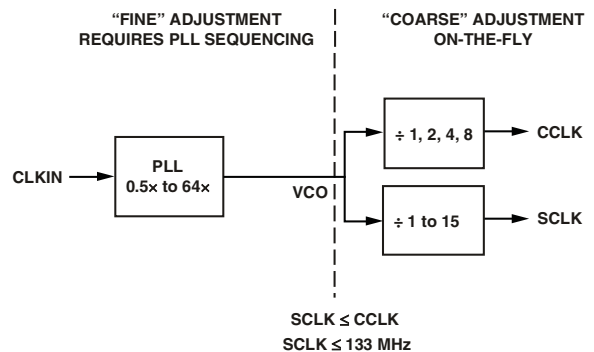
Figure 7. Frequency Modification Methods

On-the-fly CCLK and SCLK frequency changes can be effected by simply writing to the PLL_DIV register. Whereas the maximum allowed CCLK and SCLK rates depend on the applied voltages V_{DDINT} and V_{DDEXT}, the VCO is always permitted to run up to the frequency specified by the part's speed grade. The CLKOUT pin reflects the SCLK frequency to the off-chip world. It belongs to the SDRAM interface, but it functions as a reference signal in other timing specifications as well. While active by default, it can be disabled using the EBIU_SDGCTL and EBIU_AMGCTL registers.

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 6 illustrates typical system clock ratios.

Table 6. 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

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).

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 7. This programmable core clock capability is useful for fast core frequency modifications.

Table 7. Core Clock Ratios

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

The maximum CCLK frequency not only depends on the part's speed grade (see Ordering Guide on Page 67), it also depends on the applied V_{DDINT} voltage (see Table 10, Table 11, and Table 12 on Page 24 for details). The maximal system clock rate (SCLK) depends on the chip package and the applied V_{DDEXT} voltage (see Table 14 on Page 24).

BOOTING MODES

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processor has six mechanisms (listed in Table 8) for automatically loading internal and external memory after a reset. A seventh mode is provided to execute from external memory, bypassing the boot sequence.

Table 8. Booting Modes

BMODE2–0	Description
000	Execute from 16-bit external memory (bypass boot ROM)
001	Boot from 8-bit or 16-bit memory (EPROM/flash)
010	Reserved
011	Boot from serial SPI memory (EEPROM/flash)
100	Boot from SPI host (slave mode)

Table 8. Booting Modes (Continued)

BMODE2–0	Description
101	Boot from serial TWI memory (EEPROM/flash)
110	Boot from TWI host (slave mode)
111	Boot from UART host (slave mode)

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 and 16-bit external flash memory – The 8-bit or 16-bit flash boot routine located in Boot ROM memory space is set up using asynchronous memory bank 0. All configuration settings are set for the slowest device possible (3-cycle hold time; 15-cycle R/W access times; 4-cycle setup). The Boot ROM evaluates the first byte of the boot stream at address 0x2000 0000. If it is 0x40, 8-bit boot is performed. A 0x60 byte assumes a 16-bit memory device and performs 8-bit DMA. A 0x20 byte also assumes 16-bit memory but performs 16-bit DMA.

- Boot from serial SPI memory (EEPROM or flash) – 8-, 16-, or 24-bit addressable devices are supported as well as AT45DB041, AT45DB081, AT45DB161, AT45DB321, AT45DB642, and AT45DB1282 DataFlash® devices from Atmel. The SPI uses the PF10/SPI SSEL1 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 processor.

- Boot from SPI host device – 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.

- Boot from UART – Using an autobaud handshake sequence, a boot-stream-formatted program is downloaded by the host. The host agent selects a baud rate within the UART's clocking capabilities. When performing the autobaud, the UART expects an "@" (boot stream) character (8 bits data, 1 start bit, 1 stop bit, no parity bit) on the RXD pin to determine the bit rate. It then replies with an acknowledgement that is composed of 4 bytes: 0xBF, the value of UART_DLL, the value of UART_DLH, and 0x00. The host can then download the boot stream. When the processor needs to hold off the host, it deasserts CTS. Therefore, the host must monitor this signal.

- Boot from serial TWI memory (EEPROM/flash) – The Blackfin processor operates in master mode and selects the TWI slave with the unique ID 0xA0. It submits successive read commands to the memory device starting at 2-byte internal address 0x0000 and begins clocking data into the processor. The TWI memory device should comply with Philips I^{2}C Bus Specification version 2.1 and have the capability to auto-increment its internal address counter such that the contents of the memory device can be read sequentially.

- Boot from TWI host – The TWI host agent selects the slave with the unique ID 0x5F. The processor replies with an acknowledgement and the host can then download the boot stream. The TWI host agent should comply with Philips I^{2}C Bus Specification version 2.1. An I^{2}C multiplexer can be used to select one processor at a time when booting multiple processors from a single TWI.

For each of the boot modes, a 10-byte header is first brought in from an external 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 can be added to provide additional booting mechanisms. This secondary loader could provide the capability to boot from 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/MCU features are optimized for both 8-bit and 16-bit operations.

- A multi-issue load/store modified-Harvard architecture, which supports two 16-bit MAC or four 8-bit ALU + two load/store + two pointer updates per cycle.

- All registers, I/O, and memory are mapped into a unified 4G byte memory space, providing a simplified programming model.

- Microcontroller features, such as arbitrary bit and bit-field manipulation, insertion, and extraction; integer operations on 8-, 16-, and 32-bit data-types; and separate user and supervisor stack pointers.

- Code density enhancements, which include intermixing of 16-bit and 32-bit instructions (no mode switching, no code segregation). Frequently used instructions are encoded in 16 bits.

DEVELOPMENT TOOLS

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

Integrated Development Environments (IDEs)

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

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

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

EZ-KIT Lite Evaluation Board

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

EZ-KIT Lite Evaluation Kits

For a cost-effective way to learn more about developing with Analog Devices processors, Analog Devices offer a range of EZ-KIT Lite evaluation kits. Each evaluation kit includes an EZ-KIT Lite evaluation board, directions for downloading an evaluation version of the available IDE(s), a USB cable, and a power supply. The USB controller on the EZ-KIT Lite board connects to the USB port of the user's PC, enabling the chosen IDE evaluation

suite to emulate the on-board processor in-circuit. This permits the customer to download, execute, and debug programs for the EZ-KIT Lite system. It also supports in-circuit programming of the on-board Flash device to store user-specific boot code, enabling standalone operation. With the full version of Cross-Core Embedded Studio or VisualDSP++ installed (sold separately), engineers can develop software for supported EZ-KITs or any custom system utilizing supported Analog Devices processors.

Software Add-Ins for CrossCore Embedded Studio

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

Board Support Packages for Evaluation Hardware

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

Middleware Packages

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

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

Algorithmic Modules

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

Designing an Emulator-Compatible DSP Board (Target)

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

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

ADDITIONAL INFORMATION

The following publications that describe the ADSP-BF534/ADSP-BF536/ADSP-BF537 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-BF537 Blackfin Processor Hardware Reference*
- *ADSP-BF53x/ADSP-BF56x Blackfin Processor Programming Reference*
- *ADSP-BF534/ADSP-BF536/ADSP-BF537 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 Wikipedia or 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/signalchains) 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

PIN DESCRIPTIONS

The ADSP-BF534/ADSP-BF536/ADSP-BF537 processors pin definitions are listed in Table 9. In order to maintain maximum functionality and reduce package size and pin count, some pins have dual, multiplexed functions. In cases where pin function is reconfigurable, the default state is shown in plain text, while the alternate function is shown in italics. Pins shown with an asterisk after their name (*) offer high source/high sink current capabilities.

All pins are three-stated during and immediately after reset, with the exception of the external memory interface, asynchronous and synchronous memory control, and the buffered XTAL output pin (CLKBUF). On the external memory interface, the

control and address lines are driven high, with the exception of CLKOUT, which toggles at the system clock rate. If $\overline{BR}$ is active (whether or not $\overline{RESET}$ is asserted), the memory pins are also three-stated. During hibernate, all outputs are three-stated unless otherwise noted in Table 9.

All I/O pins have their input buffers disabled with the exception of the pins noted in the data sheet that need pull-ups or pull-downs if unused.

The SDA (serial data) and SCL (serial clock) pins are open drain and therefore require a pull-up resistor. Consult version 2.1 of the I²C specification for the proper resistor value.

Table 9. Pin Descriptions

Pin Name	Type	Function	Driver Type[1]
Memory Interface			
ADDR19–1	O	Address Bus for Async Access	A
DATA15–0	I/O	Data Bus for Async/Sync Access	A
$\overline{ABE1-0}$/SDQM1–0	O	Byte Enables/Data Masks for Async/Sync Access	A
$\overline{BR}$	I	Bus Request (This pin should be pulled high when not used.)	
$\overline{BG}$	O	Bus Grant	A
$\overline{BGH}$	O	Bus Grant Hang	A
Asynchronous Memory Control			
$\overline{AMS3-0}$	O	Bank Select (Require pull-ups if hibernate is used.)	A
ARDY	I	Hardware Ready Control	
$\overline{AOE}$	O	Output Enable	A
$\overline{ARE}$	O	Read Enable	A
$\overline{AWE}$	O	Write Enable	A
Synchronous Memory Control			
$\overline{SRAS}$	O	Row Address Strobe	A
$\overline{SCAS}$	O	Column Address Strobe	A
$\overline{SWE}$	O	Write Enable	A
SCKE	O	Clock Enable(Requires a pull-down if hibernate with SDRAM self-refresh is used.)	A
CLKOUT	O	Clock Output	B
SA10	O	A10 Pin	A
$\overline{SMS}$	O	Bank Select	A

ADSP-BF534/ADSP-BF536/ADSP-BF537

Table 9. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type[1]
Port F: GPIO/UART1–0/Timer7–0/SPI/ External DMA Request/PPI (= High Source/High Sink Pin)*			
PF0* – GPIO/*UART0 TX/DMAR0*	I/O	GPIO/*UART0 Transmit/DMA Request 0*	C
PF1* – GPIO/*UART0 RX/DMAR1/TACI1*	I/O	GPIO/*UART0 Receive/DMA Request 1/Timer1 Alternate Input Capture*	C
PF2* – GPIO/*UART1 TX/TMR7*	I/O	GPIO/*UART1 Transmit/Timer7*	C
PF3* – GPIO/*UART1 RX/TMR6/TACI6*	I/O	GPIO/*UART1 Receive/Timer6/Timer6 Alternate Input Capture*	C
PF4* – GPIO/*TMR5/SPI SSEL6*	I/O	GPIO/*Timer5/SPI Slave Select Enable 6*	C
PF5* – GPIO/*TMR4/SPI SSEL5*	I/O	GPIO/*Timer4/SPI Slave Select Enable 5*	C
PF6* – GPIO/*TMR3/SPI SSEL4*	I/O	GPIO/*Timer3/SPI Slave Select Enable 4*	C
PF7* – GPIO/*TMR2/PPI FS3*	I/O	GPIO/*Timer2/PPI Frame Sync 3*	C
PF8 – GPIO/*TMR1/PPI FS2*	I/O	GPIO/*Timer1/PPI Frame Sync 2*	C
PF9 – GPIO/*TMR0/PPI FS1*	I/O	GPIO/*Timer0/PPI Frame Sync 1*	C
PF10 – GPIO/*SPI SSEL1*	I/O	GPIO/*SPI Slave Select Enable 1*	C
PF11 – GPIO/*SPI MOSI*	I/O	GPIO/*SPI Master Out Slave In*	C
PF12 – GPIO/*SPI MISO*	I/O	GPIO/*SPI Master In Slave Out* (This pin should be pulled high through a 4.7 kΩ resistor if booting via the SPI port.)	C
PF13 – GPIO/*SPI SCK*	I/O	GPIO/*SPI Clock*	D
PF14 – GPIO/*SPI SS/TACLK0*	I/O	GPIO/*SPI Slave Select/Alternate Timer0 Clock Input*	C
PF15 – GPIO/*PPI CLK/TMRCLK*	I/O	GPIO/*PPI Clock/External Timer Reference*	C
Port G: GPIO/PPI/SPORT1			
PG0 – GPIO/*PPI D0*	I/O	GPIO/*PPI Data 0*	C
PG1 – GPIO/*PPI D1*	I/O	GPIO/*PPI Data 1*	C
PG2 – GPIO/*PPI D2*	I/O	GPIO/*PPI Data 2*	C
PG3 – GPIO/*PPI D3*	I/O	GPIO/*PPI Data 3*	C
PG4 – GPIO/*PPI D4*	I/O	GPIO/*PPI Data 4*	C
PG5 – GPIO/*PPI D5*	I/O	GPIO/*PPI Data 5*	C
PG6 – GPIO/*PPI D6*	I/O	GPIO/*PPI Data 6*	C
PG7 – GPIO/*PPI D7*	I/O	GPIO/*PPI Data 7*	C
PG8 – GPIO/*PPI D8/DR1SEC*	I/O	GPIO/*PPI Data 8/SPORT1 Receive Data Secondary*	C
PG9 – GPIO/*PPI D9/DT1SEC*	I/O	GPIO/*PPI Data 9/SPORT1 Transmit Data Secondary*	C
PG10 – GPIO/*PPI D10/RSCLK1*	I/O	GPIO/*PPI Data 10/SPORT1 Receive Serial Clock*	D
PG11 – GPIO/*PPI D11/RFS1*	I/O	GPIO/*PPI Data 11/SPORT1 Receive Frame Sync*	C
PG12 – GPIO/*PPI D12/DR1PRI*	I/O	GPIO/*PPI Data 12/SPORT1 Receive Data Primary*	C
PG13 – GPIO/*PPI D13/TSCLK1*	I/O	GPIO/*PPI Data 13/SPORT1 Transmit Serial Clock*	D
PG14 – GPIO/*PPI D14/TFS1*	I/O	GPIO/*PPI Data 14/SPORT1 Transmit Frame Sync*	C
PG15 – GPIO/*PPI D15/DT1PRI*	I/O	GPIO/*PPI Data 15/SPORT1 Transmit Data Primary*	C

Table 9. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type[1]
Port H: GPIO/10/100 Ethernet MAC (On ADSP-BF534, these pins are GPIO only)			
PH0 – GPIO/*ETxD0*	I/O	GPIO/*Ethernet MII or RMII Transmit D0*	E
PH1 – GPIO/*ETxD1*	I/O	GPIO/*Ethernet MII or RMII Transmit D1*	E
PH2 – GPIO/*ETxD2*	I/O	GPIO/*Ethernet MII Transmit D2*	E
PH3 – GPIO/*ETxD3*	I/O	GPIO/*Ethernet MII Transmit D3*	E
PH4 – GPIO/*ETxEN*	I/O	GPIO/*Ethernet MII or RMII Transmit Enable*	E
PH5 – GPIO/*MII TxCLK/RMII REF_CLK*	I/O	GPIO/*Ethernet MII Transmit Clock/RMII Reference Clock*	E
PH6 – GPIO/*MII PHYINT/RMII MDINT*	I/O	GPIO/*Ethernet MII PHY Interrupt/RMII Management Data Interrupt* (This pin should be pulled high when used as a hibernate wake-up.)	E
PH7 – GPIO/*COL*	I/O	GPIO/*Ethernet Collision*	E
PH8 – GPIO/*ERxD0*	I/O	GPIO/*Ethernet MII or RMII Receive D0*	E
PH9 – GPIO/*ERxD1*	I/O	GPIO/*Ethernet MII or RMII Receive D1*	E
PH10 – GPIO/*ERxD2*	I/O	GPIO/*Ethernet MII Receive D2*	E
PH11 – GPIO/*ERxD3*	I/O	GPIO/*Ethernet MII Receive D3*	E
PH12 – GPIO/*ERxDV/TACLK5*	I/O	GPIO/*Ethernet MII Receive Data Valid/Alternate Timer5 Input Clock*	E
PH13 – GPIO/*ERxCLK/TACLK6*	I/O	GPIO/*Ethernet MII Receive Clock/Alternate Timer6 Input Clock*	E
PH14 – GPIO/*ERxER/TACLK7*	I/O	GPIO/*Ethernet MII or RMII Receive Error/Alternate Timer7 Input Clock*	E
PH15 – GPIO/*MII CRS/RMII CRS_DV*	I/O	GPIO/*Ethernet MII Carrier Sense/Ethernet RMII Carrier Sense and Receive Data Valid*	E
Port J: SPORT0/TWI/SPI Select/CAN			
PJ0 – MDC	O	Ethernet Management Channel Clock (On ADSP-BF534 processors, do not connect this pin.)	E
PJ1 – MDIO	I/O	Ethernet Management Channel Serial Data (On ADSP-BF534 processors, tie this pin to ground.)	E
PJ2 – SCL	I/O	TWI Serial Clock (This pin is an open-drain output and requires a pull-up resistor.)	F
PJ3 – SDA	I/O	TWI Serial Data (This pin is an open-drain output and requires a pull-up resistor.)	F
PJ4 – DR0SEC/*CANRX/TACI0*	I	SPORT0 Receive Data Secondary/*CAN Receive/Timer0 Alternate Input Capture*	
PJ5 – DT0SEC/*CANTX/SPI SSEL7*	O	SPORT0 Transmit Data Secondary/*CAN Transmit/SPI Slave Select Enable 7*	C
PJ6 – RSCLK0/*TACLK2*	I/O	SPORT0 Receive Serial Clock/*Alternate Timer2 Clock Input*	D
PJ7 – RFS0/*TACLK3*	I/O	SPORT0 Receive Frame Sync/*Alternate Timer3 Clock Input*	C
PJ8 – DR0PRI/*TACLK4*	I	SPORT0 Receive Data Primary/*Alternate Timer4 Clock Input*	
PJ9 – TSCLK0/*TACLK1*	I/O	SPORT0 Transmit Serial Clock/*Alternate Timer1 Clock Input*	D
PJ10 – TFS0/*SPI SSEL3*	I/O	SPORT0 Transmit Frame Sync/*SPI Slave Select Enable 3*	C
PJ11 – DT0PRI/*SPI SSEL2*	O	SPORT0 Transmit Data Primary/*SPI Slave Select Enable 2*	C
Real-Time Clock			
RTXI	I	RTC Crystal Input (This pin should be pulled low when not used.)	
RTXO	O	RTC Crystal Output (Does not three-state in hibernate.)	
JTAG Port			
TCK	I	JTAG Clock	
TDO	O	JTAG Serial Data Out	C
TDI	I	JTAG Serial Data In	
TMS	I	JTAG Mode Select	
TRST	I	JTAG Reset (This pin should be pulled low if the JTAG port is not used.)	
EMU	O	Emulation Output	C

Table 9. Pin Descriptions (Continued)

Pin Name	Type	Function	Driver Type[1]
Clock			
CLKIN	I	Clock/Crystal Input	
XTAL	O	Crystal Output (If CLKBUF is enabled, does not three-state during hibernate.)	
CLKBUF	O	Buffered XTAL Output (If enabled, does not three-state during hibernate.)	E
Mode Controls			
$\overline{RESET}$	I	Reset	
$\overline{NMI}$	I	Nonmaskable Interrupt (This pin should be pulled high when not used.)	
BMODE2–0	I	Boot Mode Strap 2-0 (These pins must be pulled to the state required for the desired boot mode.)	
Voltage Regulator			
VROUT1–0	O	External FET Drive (These pins should be left unconnected when not used and are driven high during hibernate.)	
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.)	
GND	G	External Ground	

[1] See Output Drive Currents on Page 50 for more information about each driver types.

SPECIFICATIONS

Note that component specifications are subject to change without notice.

OPERATING CONDITIONS

Parameter		Conditions	Min	Nominal	Max	Unit
V_{DDINT}	Internal Supply Voltage[1]	Nonautomotive 300 MHz, 400 MHz, and 500 MHz speed grade models[2]	0.8	1.2	1.32	V
V_{DDINT}	Internal Supply Voltage[1]	Nonautomotive 533 MHz speed grade models[2]	0.8	1.25	1.375	V
V_{DDINT}	Internal Supply Voltage[1]	Nonautomotive 600 MHz speed grade models[2]	0.8	1.3	1.43	V
V_{DDINT}	Internal Supply Voltage[1]	Automotive grade models and +105°C nonautomotive grade models[2]	0.95	1.2	1.32	V
V_{DDEXT}	External Supply Voltage	Nonautomotive grade models[2]	2.25	2.5 or 3.3	3.6	V
V_{DDEXT}	External Supply Voltage	Automotive grade models and +105°C nonautomotive grade models[2]	2.7	3.0 or 3.3	3.6	V
V_{DDRTC}	Real-Time Clock Power Supply Voltage		2.25		3.6	V
V_{IH}	High Level Input Voltage[3, 4]	V_{DDEXT} = Maximum	2.0			V
$V_{IHCLKIN}$	High Level Input Voltage[5]	V_{DDEXT} = Maximum	2.2			V
V_{IH5V}	5.0 V Tolerant Pins, High Level Input Voltage[6]		$0.7 \times V_{DDEXT}$			V
V_{IH5V}	5.0 V Tolerant Pins, High Level Input Voltage[7]	V_{DDEXT} = Maximum	2.0			V
V_{IL}	Low Level Input Voltage[3, 8]	V_{DDEXT} = Minimum			+0.6	V
V_{IL5V}	5.0 V Tolerant Pins, Low Level Input Voltage[6]				$0.3 \times V_{DDEXT}$	V
V_{IL5V}	5.0 V Tolerant Pins, Low Level Input Voltage[7]	V_{DDEXT} = Minimum			+0.8	V
T_J	Junction Temperature	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA) @ $T_{AMBIENT}$ = −40°C to +105°C	−40		+120	°C
T_J	Junction Temperature	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA) @ $T_{AMBIENT}$ = −40°C to +85°C	−40		+105	°C
T_J	Junction Temperature	208-Ball Chip Scale Package Ball Grid Array (CSP_BGA) @ $T_{AMBIENT}$ = 0°C to +70°C	0		+95	°C
T_J	Junction Temperature	182-Ball Chip Scale Package Ball Grid Array (CSP_BGA) @ $T_{AMBIENT}$ = −40°C to +85°C	−40		+105	°C
T_J	Junction Temperature	182-Ball Chip Scale Package Ball Grid Array (CSP_BGA) @ $T_{AMBIENT}$ = 0°C to +70°C	0		+100	°C

[1] The regulator can generate V_{DDINT} at levels of 0.85 V to 1.2 V with −5% to +10% tolerance, 1.25 V with −4% to +10% tolerance, and 1.3 V with −0% to +10% tolerance. The required V_{DDINT} is a function of speed grade and operating frequency. See Table 10, Table 11, and Table 12 for details.

[2] See Ordering Guide on Page 67.

[3] Bidirectional pins (DATA15–0, PF15–0, PG15–0, PH15–0, TFS0, TSCLK0, RSCLK0, RFS0, MDIO) and input pins ($\overline{BR}$, ARDY, DR0PRI, DR0SEC, RTXI, TCK, TDI, TMS, $\overline{TRST}$, CLKIN, $\overline{RESET}$, $\overline{NMI}$, and BMODE2–0) of the ADSP-BF534/ADSP-BF536/ADSP-BF537 are 3.3 V-tolerant (always accept up to 3.6 V maximum V_{IH}). Voltage compliance (on outputs, V_{OH}) is limited by the V_{DDEXT} supply voltage.

[4] Parameter value applies to all input and bidirectional pins except CLKIN, SDA, and SCL.

[5] Parameter value applies to CLKIN pin only.

[6] Applies to pins PJ2/SCL and PJ3/SDA which are 5.0 V tolerant (always accept up to 5.5 V maximum V_{IH}). Voltage compliance (on outputs, V_{OH}) is limited by the V_{DDEXT} supply voltage.

[7] Applies to pin PJ4/DR0SEC/CANRX/TACI0 which is 5.0 V tolerant (always accepts up to 5.5 V maximum V_{IH}). Voltage compliance (on outputs, V_{OH}) is limited by the V_{DDEXT} supply voltage.

[8] Parameter value applies to all input and bidirectional pins except SDA and SCL.

ADSP-BF534/ADSP-BF536/ADSP-BF537

Table 10 through Table 12 describe the voltage/frequency requirements for the ADSP-BF534/ADSP-BF536/ADSP-BF537 processor clocks. Take care in selecting MSEL, SSEL, and CSEL ratios so as not to exceed the maximum core clock and system clock. Table 13 describes phase-locked loop operating conditions.

Table 10. Core Clock Requirements—500 MHz, 533 MHz, and 600 MHz Speed Grades[1]

Parameter		Internal Regulator Setting	Max	Unit
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.30 V Minimum)[2]	1.30 V	600	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 1.20 V Minimum)[3]	1.25 V	533	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.14 V Minimum)	1.20 V	500	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.045 V Minimum)	1.10 V	444	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.95 V Minimum)	1.00 V	400	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.85 V Minimum)	0.90 V	333	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.8 V Minimum)	0.85 V	250	MHz

[1] See Ordering Guide on Page 67.
[2] Applies to 600 MHz models only. See Ordering Guide on Page 67.
[3] Applies to 533 MHz and 600 MHz models only. See Ordering Guide on Page 67.

Table 11. Core Clock Requirements—400 MHz Speed Grade[1]

Parameter		Internal Regulator Setting	120°C ≥ T$_J$ > 105°C Max	All[2] Other T$_J$ Max	Unit
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.14 V Minimum)	1.20 V	400	400	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.045 V Minimum)	1.10 V	333	363	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.95 V Minimum)	1.00 V	295	333	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.85 V Minimum)	0.90 V		280	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.8 V Minimum)	0.85 V		250	MHz

[1] See Ordering Guide on Page 67.
[2] See Operating Conditions on Page 23.

Table 12. Core Clock Requirements—300 MHz Speed Grade[1]

Parameter		Internal Regulator Setting	Max	Unit
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.14 V Minimum)	1.20 V	300	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} =1.045 V Minimum)	1.10 V	255	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.95 V Minimum)	1.00 V	210	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.85 V Minimum)	0.90 V	180	MHz
f_{CCLK}	Core Clock Frequency (V_{DDINT} = 0.8 V Minimum)	0.85 V	160	MHz

[1] See Ordering Guide on Page 67.

Table 13. Phase-Locked Loop Operating Conditions

Parameter		Min	Max	Unit
f_{VCO}	Voltage Controlled Oscillator (VCO) Frequency	50	Max f_{CCLK}	MHz

Table 14. System Clock Requirements

Parameter	Condition	Max	Unit
f_{SCLK}[1]	V_{DDEXT} = 3.3 V or 2.5 V, V_{DDINT} ≥ 1.14 V	133[2]	MHz
f_{SCLK}[1]	V_{DDEXT} = 3.3 V or 2.5 V, V_{DDINT} < 1.14 V	100	MHz

[1] f_{SCLK} must be less than or equal to f_{CCLK} and is subject to additional restrictions for SDRAM interface operation. See Table 27 on Page 34.
[2] Rounded number. Actual test specification is SCLK period of 7.5 ns. See Table 27 on Page 34.

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	300 MHz/400 MHz[1]			500 MHz/533 MHz/600 MHz[2]			Unit
			Min	Typ	Max	Min	Typ	Max	
V_{OH}[3]	High Level Output Voltage	V_{DDEXT} = 2.5 V/3.0 V/ 3.3 V ± 10%, I_{OH} = −0.5 mA	V_{DDEXT} − 0.5			V_{DDEXT} − 0.5			V
V_{OH}[4]		V_{DDEXT} = 3.3 V ± 10%, I_{OH} = −8 mA	V_{DDEXT} − 0.5			V_{DDEXT} − 0.5			V
		V_{DDEXT} = 2.5 V/3.0 V ± 10%, I_{OH} = −6 mA	V_{DDEXT} − 0.5			V_{DDEXT} − 0.5			V
V_{OH}[5]		V_{DDEXT} = 2.5 V/3.0 V/ 3.3 V ± 10%, I_{OH} = −2.0 mA	V_{DDEXT} − 0.5			V_{DDEXT} − 0.5			V
I_{OH}[6]	High Level Output Current	V_{OH} = V_{DDEXT} − 0.5 V Min			−64			−64	mA
I_{OH}[7]		V_{OH} = V_{DDEXT} − 0.5 V Min			−144			−144	mA
V_{OL}[3]	Low Level Output Voltage	V_{DDEXT} = 2.5 V/3.0 V/ 3.3 V ± 10%, I_{OL} = 2.0 mA			0.4			0.4	V
V_{OL}[4]		V_{DDEXT} = 3.3 V ± 10%, I_{OL} = 8 mA			0.5			0.5	V
		V_{DDEXT} = 2.5 V/3.0 V ± 10%, I_{OL} = 6 mA			0.5			0.5	V
V_{OL}[5]		V_{DDEXT} = 2.5 V/3.0 V/ 3.3 V ± 10%, I_{OL} = 2.0 mA			0.5			0.5	V
I_{OL}[6]	Low Level Output Current	V_{OL} = 0.5 V Max			64			64	mA
I_{OL}[7]		V_{OL} = 0.5 V Max			144			144	mA
I_{IH}	High Level Input Current[8]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			10			10	µA
I_{IH5V}	High Level Input Current[9]	V_{DDEXT} = 3.6 V, V_{IN} = 5.5 V			10			10	µA
I_{IL}	Low Level Input Current[2]	V_{DDEXT} = 3.6 V, V_{IN} = 0 V			10			10	µA
I_{IHP}	High Level Input Current JTAG[10]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			50			50	µA
I_{OZH}	Three-State Leakage Current[11]	V_{DDEXT} = 3.6 V, V_{IN} = 3.6 V			10			10	µA
I_{OZH5V}	Three-State Leakage Current[12]	V_{DDEXT} = 3.6 V, V_{IN} = 5.5 V			10			10	µA
I_{OZL}	Three-State Leakage Current[5]	V_{DDEXT} = 3.6 V, V_{IN} = 0 V			10			10	µA