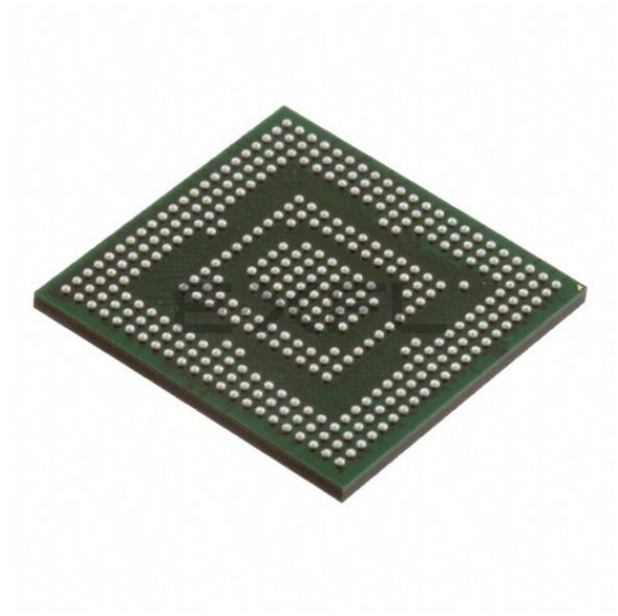

Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

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

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Dual Core
Interface	CAN, EBI/EMI, Ethernet, I²C, SPI, SPORT, UART/USART, USB OTG
Clock Rate	500MHz
Non-Volatile Memory	ROM (64kB)
On-Chip RAM	808K x 8
Voltage - I/O	1.8V, 3.3V
Voltage - Core	1.25V
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	349-LFBGA, CSPBGA
Supplier Device Package	349-CSPBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adbf608wcbcz502rl

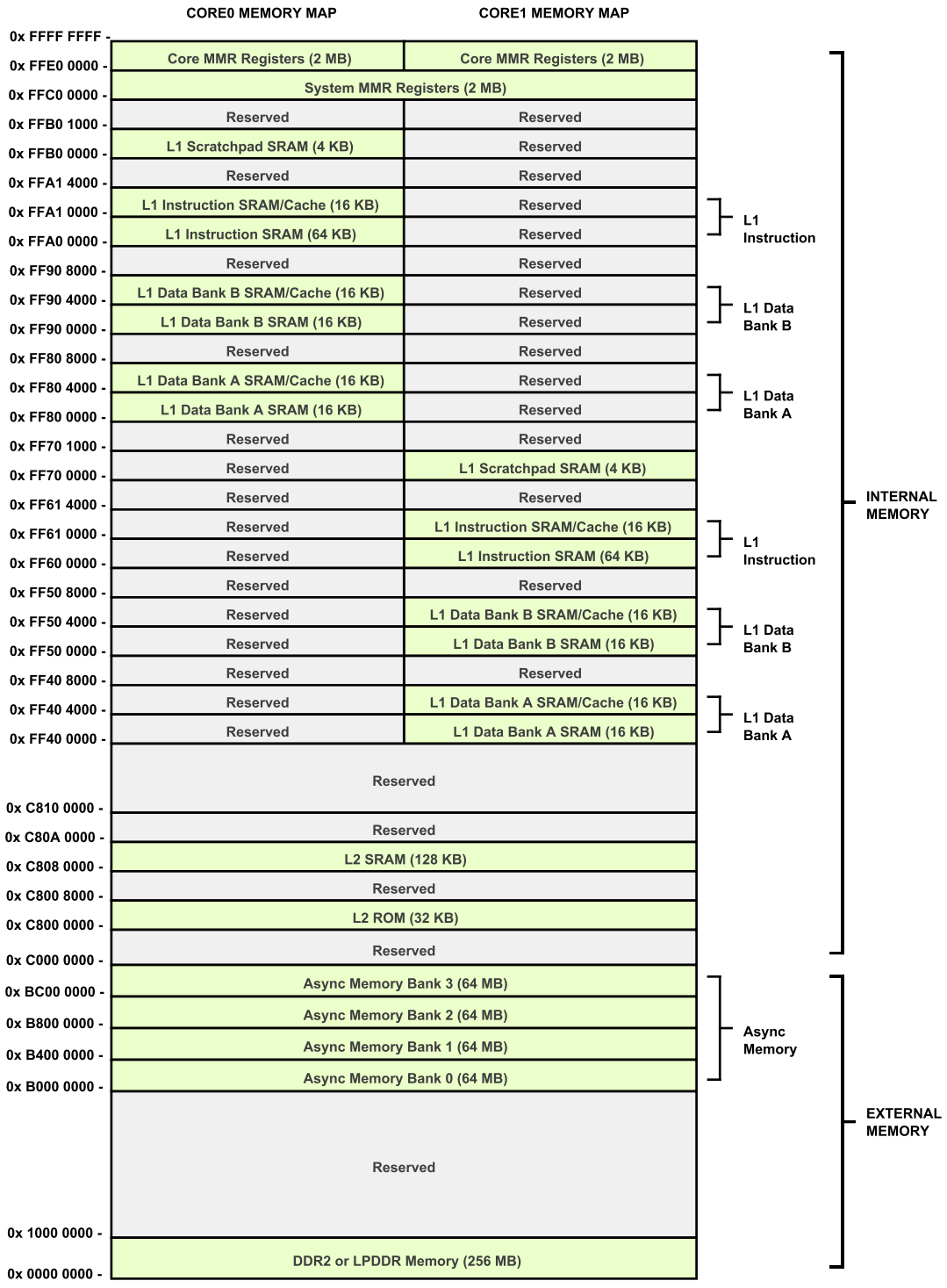

Figure 3. ADSP-BF606 Internal/External Memory Map

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

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

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

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

Pixel Compositor (PIXC)

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

Parallel Peripheral Interface (PPI)

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

The following features are supported in the PPI module:

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

PROCESSOR SAFETY FEATURES

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

Dual Core Supervision

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

Multi-Parity-Bit-Protected L1 Memories

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

ECC-Protected L2 Memories

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

CRC-Protected Memories

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

Memory Protection

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

System Protection

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

Watchpoint Protection

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

Dual Watchdog

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

Bandwidth Monitor

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

Signal Watchdogs

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

Up/Down Count Mismatch Detection

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

Fault Management

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

ADDITIONAL PROCESSOR PERIPHERALS

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

The following sections describe additional peripherals that were not described in the previous sections.

Timers

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

General-Purpose Timers

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

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

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

Core Timers

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

Watchdog Timers

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

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

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

Serial Peripheral Interface (SPI) Ports

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

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

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

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

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

UART Ports

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

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

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

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

TWI Controller Interface

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

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

Removable Storage Interface (RSI)

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

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

Controller Area Network (CAN)

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

The CAN controller offers the following features:

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

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

Target defined:

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

Source defined:

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

Voltage Regulation

The processor requires an external voltage regulator to power the V_{DD_INT} pins. To reduce standby power consumption, the external voltage regulator can be signaled through SYS_EXTWAKE to remove power from the processor core. This signal is high-true for power-up and may be connected directly to the low-true shut-down input of many common regulators.

While in the hibernate state, all external supply pins (V_{DD_EXT}, V_{DD_USB}, V_{DD_DMC}) can still be powered, eliminating the need for external buffers. The external voltage regulator can be activated from this power down state by asserting the $\overline{\text{SYS\_HWRST}}$ pin, which then initiates a boot sequence. SYS_EXTWAKE indicates a wakeup to the external voltage regulator.

SYSTEM DEBUG

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

System Watchpoint Unit

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

System Debug Unit

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

DEVELOPMENT TOOLS

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

Integrated Development Environments (IDEs)

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

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

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

EZ-KIT Lite Evaluation Board

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

EZ-KIT Lite Evaluation Kits

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

Table 6. Detailed Signal Descriptions (Continued)

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

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

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

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

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

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

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

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

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

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

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

SPECIFICATIONS

For information about product specifications please contact your ADI representative.

OPERATING CONDITIONS

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

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

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

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

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

ELECTRICAL CHARACTERISTICS

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

Asynchronous Flash Read

Table 29. Asynchronous Flash Read

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

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

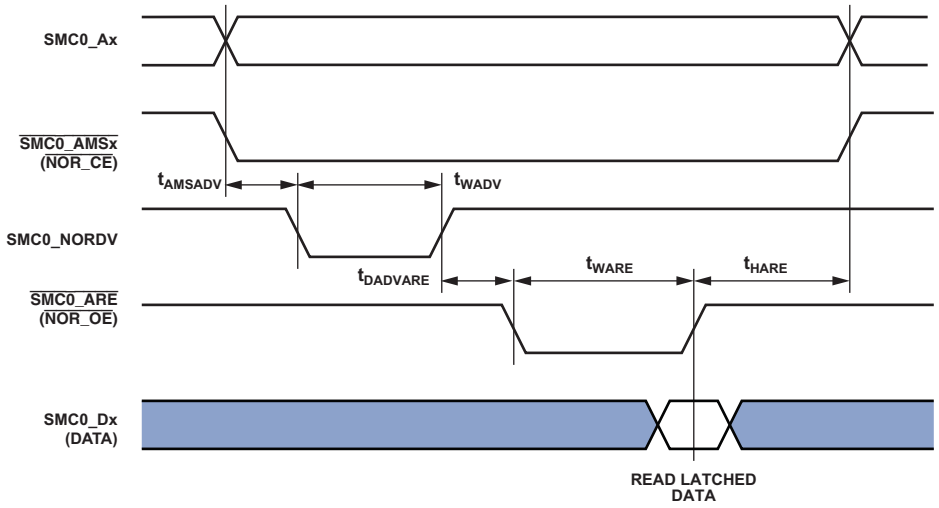

Figure 13. Asynchronous Flash Read

Asynchronous Flash Write

Table 33. Asynchronous Flash Write

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

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

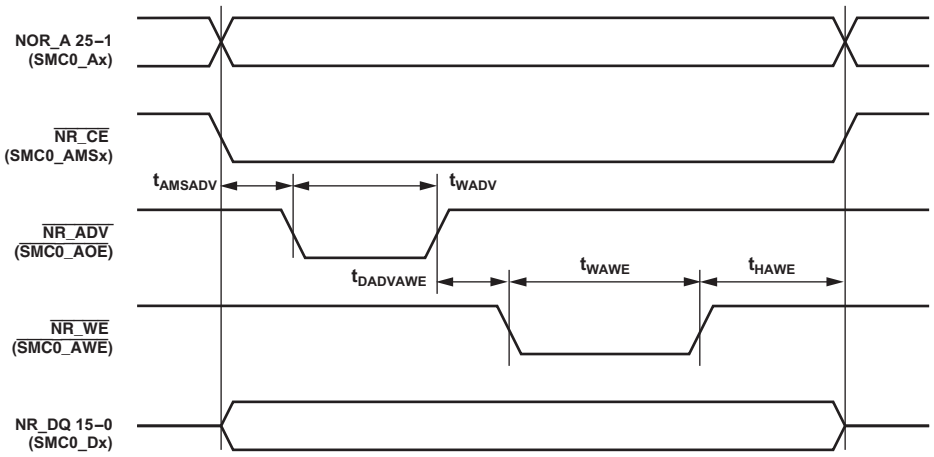

Figure 17. Asynchronous Flash Write

All Accesses

Table 34. All Accesses

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

DDR2 SDRAM Write Cycle Timing

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

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

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

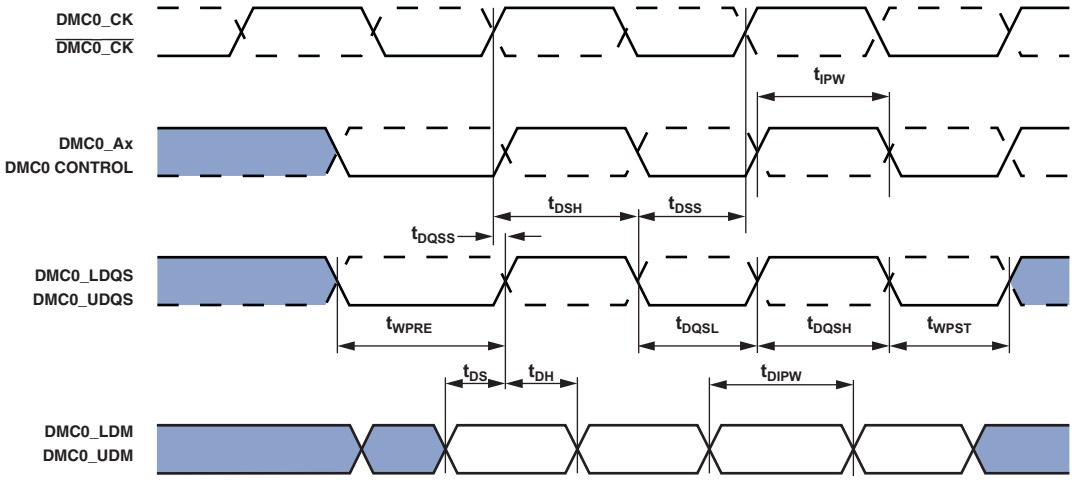

Figure 21. DDR2 SDRAM Controller Output AC Timing

Serial Peripheral Interface (SPI) Port—SPI_RDY Slave Timing

Table 53. SPI Port—SPI_RDY Slave Timing

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

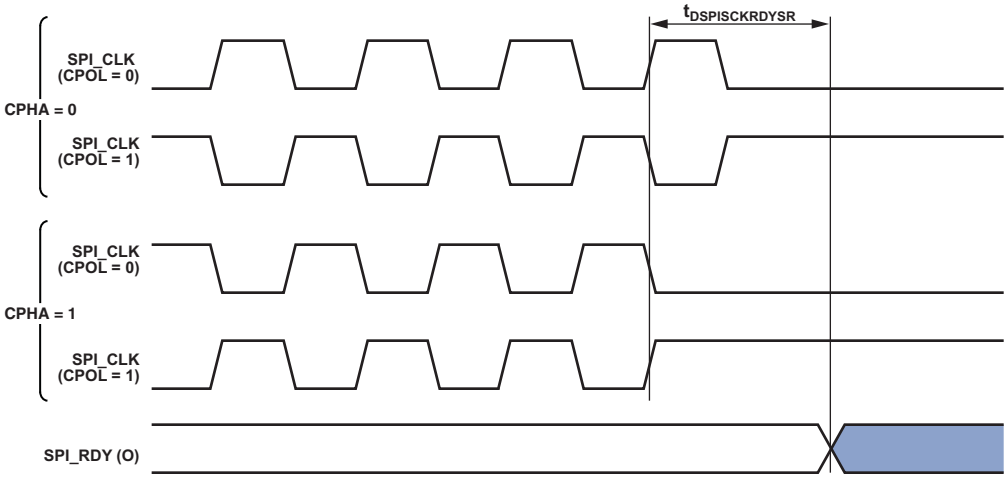

Figure 42. SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Receive (FCCH = 0)

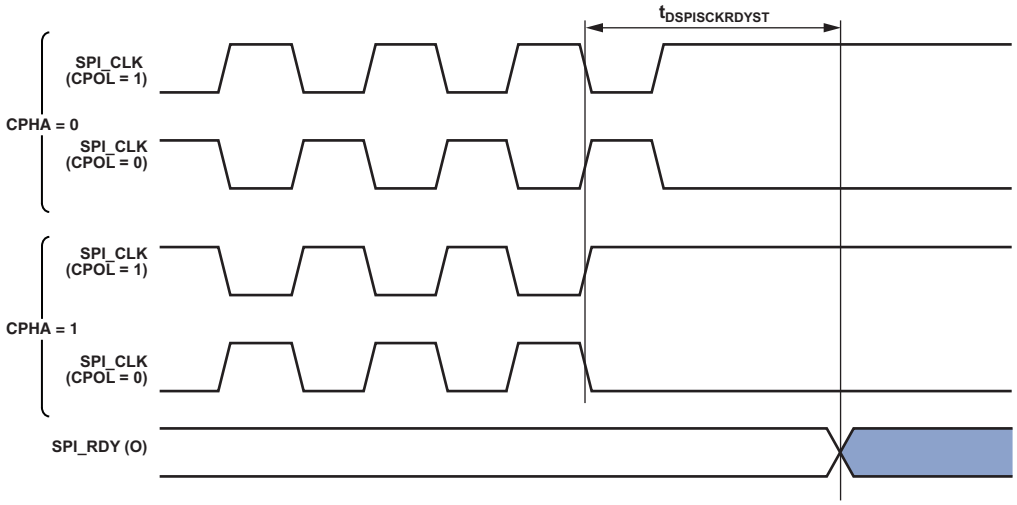

Figure 43. SPI_RDY De-assertion from Valid Input SPI_CLK Edge in Slave Mode Transmit (FCCH = 1)

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

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

Table 54. SPI Port ODM Master Mode Timing

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

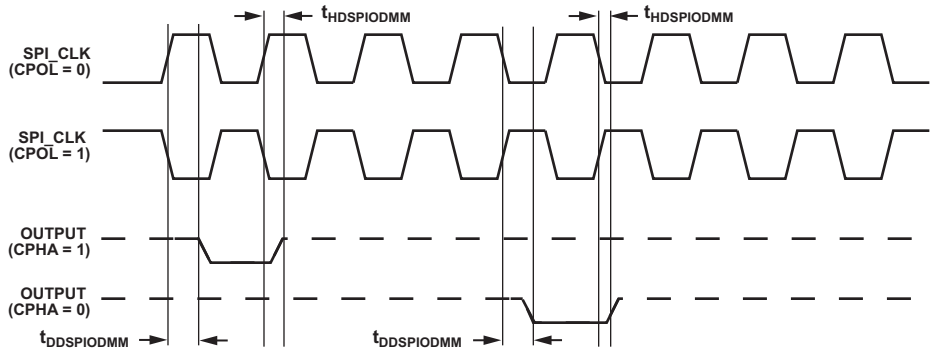

Figure 44. ODM Master

Table 55. SPI Port—ODM Slave Mode

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

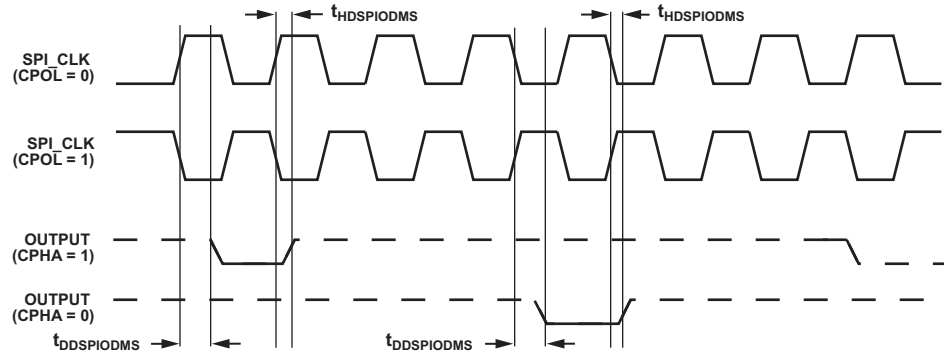

Figure 45. ODM Slave

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

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

CAN Interface

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

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

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

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

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

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

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

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

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