SPRS196I − MARCH 2002 − REVISED JUNE 2005

- **Low-Cost, High-Performance Fixed-Point DSP − TMS320C6411**
  - **3.33-ns Instruction Cycle Time**
  - **300-MHz Clock Rate**
  - **Eight 32-Bit Instructions/Cycle**
  - **Twenty-Eight Operations/Cycle**
  - **2400 MIPS**
  - **Fully Software-Compatible With TMS320C62x™**
- **VelociTI.2™ Extensions to VelociTI™ Advanced Very-Long-Instruction-Word (VLIW) TMS320C64x™ DSP Core**
  - **Eight Highly Independent Functional Units With VelociTI.2™ Extensions:**
    - **Six ALUs (32-/40-Bit), Each Supports Single 32-Bit, Dual 16-Bit, or Quad 8-Bit Arithmetic per Clock Cycle**
    - **Two Multipliers Support Four 16 x 16-Bit Multiplies (32-Bit Results) per Clock Cycle or Eight 8 x 8-Bit Multiplies (16-Bit Results) per Clock Cycle**
  - **Non-Aligned Load-Store Architecture**
  - **64 32-Bit General-Purpose Registers**
  - **Instruction Packing Reduces Code Size**
  - **All Instructions Conditional**
- **Instruction Set Features**
  - **Byte-Addressable (8-/16-/32-/64-Bit Data)**
  - **8-Bit Overflow Protection**
  - **Bit-Field Extract, Set, Clear**
  - **Normalization, Saturation, Bit-Counting**
  - **VelociTI.2™ Increased Orthogonality**
- **L1/L2 Memory Architecture**
  - **128K-Bit (16K-Byte) L1P Program Cache (Direct Mapped)**
  - **128K-Bit (16K-Byte) L1D Data Cache (2-Way Set-Associative)**
  - **2M-Bit (256K-Byte) L2 Unified Mapped RAM/Cache (Flexible RAM/Cache Allocation)**
- **32-Bit External Memory Interface (EMIF)**
  - **Glueless Interface to Asynchronous Memories (SRAM and EPROM) and Synchronous Memories (SDRAM, SBSRAM, ZBT SRAM, and FIFO)**
  - **512M-Byte Total Addressable External Memory Space**
- **Enhanced Direct-Memory-Access (EDMA) Controller (64 Independent Channels)**
- **Host-Port Interface (HPI)**
  - **User-Configurable Bus Width (32-/16-Bit)**
  - **Access to Entire Memory Map**
- **32-Bit/33-MHz, 3.3-V Peripheral Component Interconnect (PCI) Master/Slave Interface Conforms to PCI Specification 2.2**
  - **Access to Entire Memory Map**
  - **Three PCI Bus Address Registers: Prefetchable Memory Non-Prefetchable Memory I/O**
  - **Four-Wire Serial EEPROM Interface**
  - **PCI Interrupt Request Under DSP Program Control**
  - **DSP Interrupt Via PCI I/O Cycle**
- **Two Multichannel Buffered Serial Ports (McBSPs)**
  - **Direct Interface to T1/E1, MVIP, SCSA Framers**
  - **ST-Bus-Switching Compatible**
  - **Up to 256 Channels Each**
  - **AC97-Compatible**
  - **Serial Peripheral Interface (SPI) Compatible (Motorola™)**
- **Three 32-Bit General-Purpose Timers**
- **Sixteen General-Purpose I/O (GPIO) Pins**
  - **Programmable Interrupt/Event Generation Modes**
- **Flexible PLL Clock Generator**
- **IEEE-1149.1 (JTAG†) Boundary-Scan-Compatible**
- **532-Pin Ball Grid Array (BGA) Package (GLZ, ZLZ and CLZ Suffixes), 0.8-mm Ball Pitch**
- **0.13-μm/6-Level Copper Metal Process**
  - **CMOS Technology**
- **3.3-V I/Os, 1.2-V Internal**

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TMS320C62x, VelociTI.2, VelociTI, and TMS320C64x are trademarks of Texas Instruments.
Motorola is a trademark of Motorola, Inc.
† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 2004, Texas Instruments Incorporated

Table of Contents

revision history 3
GLZ, ZLZ and CLZ BGA packages (bottom view) 4
description 5
device characteristics 6
device compatibility 7
functional block and CPU (DSP core) diagram 8
CPU (DSP core) description 9
memory map summary 12
peripheral register descriptions 14
EDMA channel synchronization event 24
interrupt sources and interrupt selector 25
signal groups description 26
device configurations 30
multiplexed pins 33
debugging considerations 33
terminal functions 35
development support 52
device support 53
clock PLL 56
general-purpose input/output (GPIO) 58
power-down mode logic 59
power-supply sequencing 61
power-supply decoupling 62
IEEE 1149.1 JTAG compatibility statement 63
EMIF device speed 63
bootmode 64
reset 64
absolute maximum ratings over operating case temperature range 65
recommended operating conditions 65
electrical characteristics over recommended ranges of supply voltage and operating case temperature . 66
recommended clock and control signal transition behavior 66
parameter measurement information 67
input and output clocks 69
asynchronous memory timing 73
programmable synchronous interface timing 76
synchronous DRAM timing 80
HOLD/HOLDA timing 87
BUSREQ timing 88
reset timing 89
external interrupt timing 91
host-port interface (HPI) timing 92
peripheral component interconnect (PCI) timing 97
multichannel buffered serial port (McBSP) timing 100
timer timing 111
general-purpose input/output (GPIO) port timing 112
JTAG test-port timing 113
mechanical data 114

REVISION HISTORY

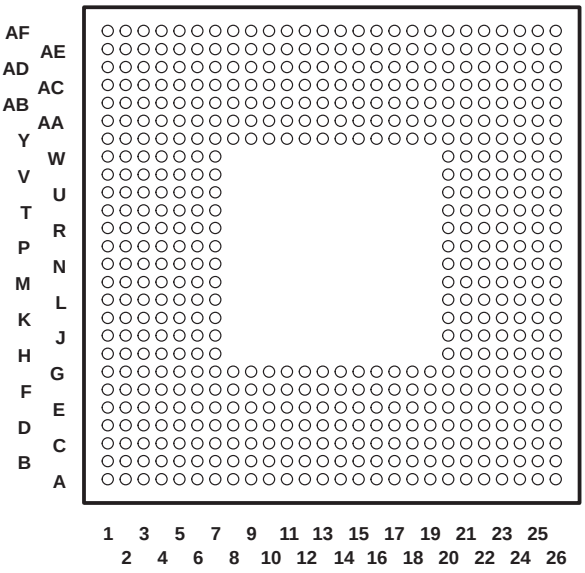
This data sheet revision history highlights the technical changes made to the SPRS196H device-specific data sheet to make it an SPRS196I revision.

Scope: Applicable updates to the C64x device family, specifically relating to the C6411 devices, have been incorporated.

PAGE(S) NO.	ADDITIONS/CHANGES/DELETIONS
	Global: Added “CLZ” (532-pin plastic BGA, Pb−free bump and Pb−free soldered balls) mechanical package information
41	Terminal Functions table: Added “Following RESET, TOUTx will be configured as a general-purpose output (GPO) due to the default value of the Timerx Control Register (CTLx); therefore, an external resistor may *not* be used to pull the signal to the opposite supply rail” footnote
53	Device Support: Device and Development-Support Tool Nomenclature section: Updated the “To designate the stages ...” paragraph
54	Device Support: Figure 5, TMS320C6411 DSP Device Nomenclature: Added “The CLZ mechanical package designator represents ...” footnote
114	Mechanical Data: Updated title of section to include the devices Added new “Packaging Information” title Added lead-in sentence for Packaging Information title

GLZ, ZLZ and CLZ BGA packages (bottom view)

GLZ, ZLZ and CLZ 532-PIN BALL GRID ARRAY (BGA) PACKAGE
(BOTTOM VIEW)†‡



† The ZLZ mechanical package designator represents the version of the GLZ package with lead-free soldered balls. For more detailed information, see the *Mechanical Data* section of this document.

‡ The CLZ mechanical package designator represents the version of the GLZ package with lead-free bump and lead−free soldered balls. For more detailed information, see the *Mechanical Data* section of this document.

description

The TMS320C64x™ DSPs (including the TMS320C6411 device) are the highest-performance fixed-point DSP generation in the TMS320C6000™ DSP platform. The TMS320C6411 (C6411) device is based on the second-generation high-performance, advanced VelociTI™ very-long-instruction-word (VLIW) architecture (VelocTI.2™) developed by Texas Instruments (TI), making these DSPs an excellent choice for multichannel and multifunction applications. The C64x™ is a code-compatible member of the C6000™ DSP platform.

With performance of up to 2400 million instructions per second (MIPS) at a clock rate of 300 MHz, the C6411 device offers cost-effective solutions to high-performance DSP programming challenges. The C6411 DSP possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. The C64x™ DSP core processor has 64 general-purpose registers of 32-bit word length and eight highly independent functional units—two multipliers for a 32-bit result and six arithmetic logic units (ALUs)— with VelociTI.2™ extensions. The VelociTI.2™ extensions in the eight functional units include new instructions to accelerate the performance in key applications and extend the parallelism of the VelociTI™ architecture. The C6411 can produce four 16-bit multiply-accumulates (MACs) per cycle for a total of 600 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 2400 MMACS. The C6411 DSP also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000™ DSP platform devices.

The C6411 uses a two-level cache-based architecture and has a powerful and diverse set of peripherals. The Level 1 program cache (L1P) is a 128-Kbit direct mapped cache and the Level 1 data cache (L1D) is a 128-Kbit 2-way set-associative cache. The Level 2 memory/cache (L2) consists of a 2-Mbit memory space that is shared between program and data space. L2 memory can be configured as mapped memory or combinations of cache (up to 256K bytes) and mapped memory. The peripheral set includes two multichannel buffered serial ports (McBSPs); three 32-bit general-purpose timers; a user-configurable 16-bit or 32-bit host-port interface (HPI16/HPI32); a peripheral component interconnect (PCI); a general-purpose input/output port (GPIO) with 16 GPIO pins; and a glueless external memory interface (32-bit EMIF), which is capable of interfacing to synchronous and asynchronous memories and peripherals.

The C6411 has a complete set of development tools which includes: a new C compiler, an assembly optimizer to simplify programming and scheduling, and a Windows™ debugger interface for visibility into source code execution.

TMS320C6000, C64x, and C6000 are trademarks of Texas Instruments.
Windows is a registered trademark of the Microsoft Corporation.
Other trademarks are the property of their respective owners.

device characteristics

Table 1 provides an overview of the C6411 DSP. The table shows significant features of the C6411 device, including the capacity of on-chip RAM, the peripherals, the CPU frequency, and the package type with pin count.

Table 1. Characteristics of the C6411 Processor

HARDWARE FEATURES		INTERNAL CLOCK SOURCE	C6411
Peripherals Not all peripherals pins are available at the same time. (For more details, see the Device Configuration section.) Peripheral performance is dependent on chip-level configuration.	EMIF (32-bit bus width)	ECLKIN	1
	EDMA (64 independent channels)	CPU/2 Clock Frequency	1
	HPI (32- or 16-bit user selectable)	CPU/4 Clock Frequency	1 (HPI16 or HPI32)
	PCI (32-bit)	PCLK	1
	McBSPs (McBSP0 and McBSP1)	CPU/4 Clock Frequency	2
	32-Bit Timers	CPU/8 Clock Frequency	3
	General-Purpose Input/Outputs (GPIOs)	CPU/8 Clock Frequency	16
On-Chip Memory	Size (Bytes)		288K
	Organization		16K-Byte (16KB) L1 Program (L1P) Cache 16KB L1 Data (L1D) Cache 256KB Unified Mapped RAM/Cache (L2)
CPU ID + CPU Rev ID	Control Status Register (CSR.[31:16])		0x0C01
Device_ID	Silicon Revision Identification Register (DEVICE_REV [19:16]) Address: 0x01B0 0200		DEVICE_REV[19:16] Silicon Revision 0010 or 0000 1.1 0011 2.0
Frequency	MHz		300
Cycle Time	ns		3.33 ns
Voltage	Core (V)		1.2 V
	I/O (V)		3.3 V
PLL Options	CLKIN frequency multiplier		Bypass (x1), x6
BGA Package	23 x 23 mm		532-Pin BGA (GLZ, ZLZ and CLZ)
Process Technology	µm		0.13 µm
Product Status†	Product Preview (PP) Advance Information (AI) Production Data (PD)		PD (1.1, 2.0)
Device Part Numbers	(For more details on the C64x™ DSP part numbering, see Figure 5)		TMS320C6411GLZ TMS320C6411AGLZ

† PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.
ADVANCE INFORMATION concerns new products in the sampling or preproduction phase of development. Characteristic data and other specifications are subject to change without notice.
PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

device compatibility

The C64x™ family of devices has a diverse and powerful set of peripherals. The common peripheral set that the C6411 and C6415 devices offer lead to easier system designs and faster time to market.

The C6411 device is a low-cost C64x device which features significant enhancements from the C6211/C6211B devices and can be considered a subset of the C6415 device. Table 2 identifies the C6411 features in comparison with the C6211 and C6415 devices.

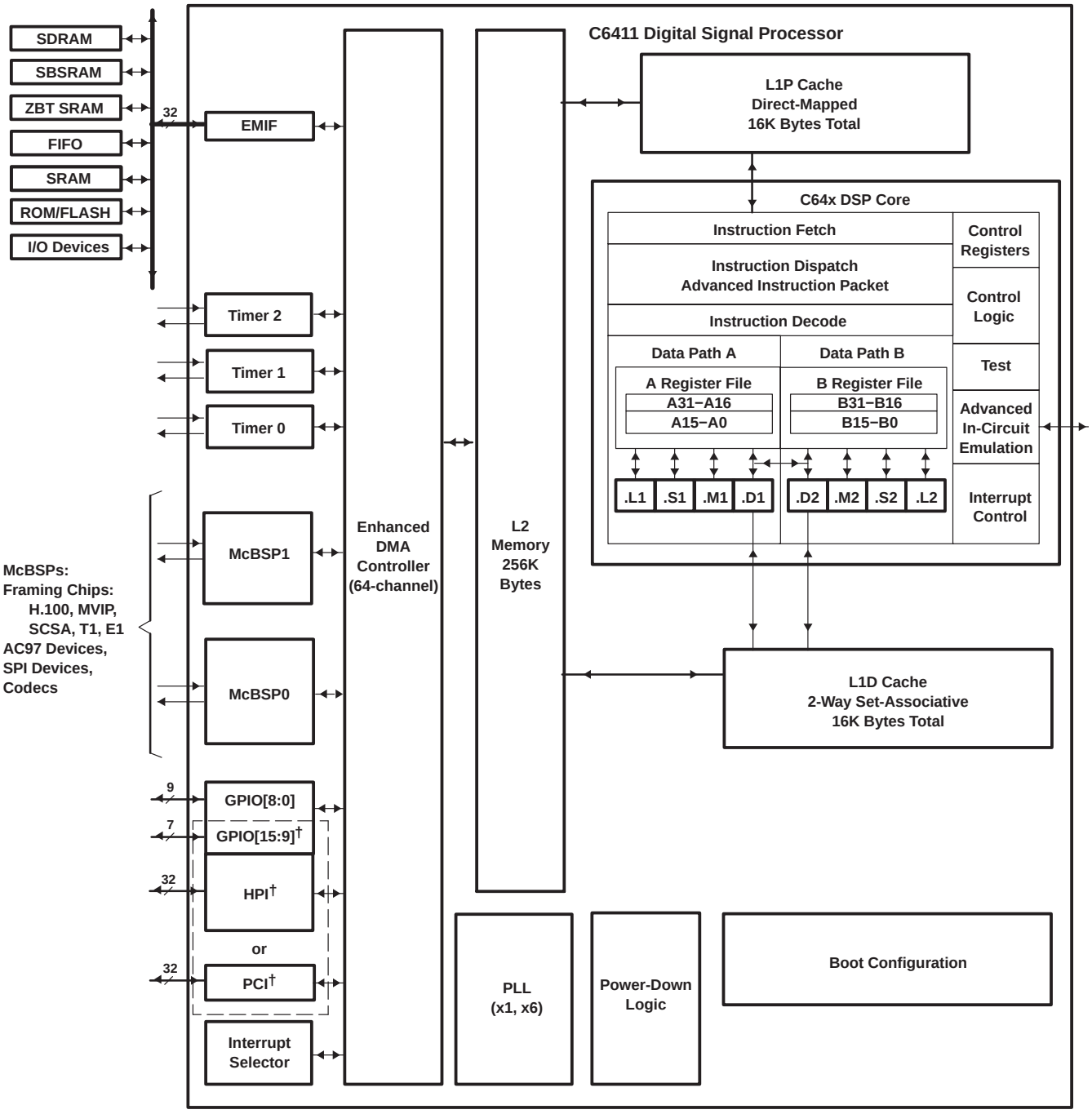
Table 2. C6211, C6411, and C6415 Device Comparison†‡

CPU/PERIPHERALS	C6211/C6211B	C6411	C6415
DSP Core	C62x	C64x	C64x
L1P (Program Cache)	4 KB	16 KB	16 KB
L1D (Data Cache)	4 KB	16 KB	16 KB
L2 (Unified Mapped RAM/Cache)	64 KB	256 KB	1024 KB
EMIF (64-, 32-, 16-bit bus width)	(1) 32-Bit	(1) 32-Bit EMIF programmable synchronous mode	(1) 64-Bit(1) [EMIFA] (1) 16-Bit [EMIFB] programmable synchronous mode
EDMA (# of independent channels)	16	64	64
HPI (32- or 16-bit user selectable)	16-Bit	32-/16-Bit	32-/16-Bit
PCI (32-bit)	—	32-Bit, 33 MHz	32-Bit, 33 MHz
McBSPs (McBSP0, McBSP1, and McBSP2)	2	2 Enhanced	3 Enhanced
UTOPIA	—	—	(1) Transmit (1) Receive
Timers (32-bit) [TIMER0, TIMER1, TIMER2]	2	3	3
GPIOs (GP[15:0])	—	16	16
Core Frequency (MHz)	150-, 167-MHz	300-MHz	500-, 600-MHz
Core Voltage (V)	1.8 V	1.2 V	1.2 V to 1.4 V
PLL Modes (x1 [Bypass], x4, x6, x12)	x1, x4	x1, x6	x1, x6, x12
Package	256-pin BGA 27 x 27 mm GFN suffix	532-pin BGA 23 x 23 mm GLZ suffix	532-pin BGA 23 x 23 mm GLZ suffix
Process Technology	0.18 μm	0.13 μm	0.13 μm

† — denotes peripheral/coprocessor is *not* available on this device.
‡ Not all peripherals pins are available at the same time. (For more details, see the Device Configuration section.)

For more detailed information on the device compatibility and similarities/differences among the C6211, C6411, and C6415 devices, see the *How To Begin Development Today With the TMS320C6414, TMS320C6415, and TMS320C6416 DSPs* application report (literature number SPRA718) and *How To Begin Development Today With the TMS320C6411 DSP* application report (literature number SPRA374).

functional block and CPU (DSP core) diagram



† The PCI peripheral is muxed with the HPI peripheral and the GPIO[15:9] port. For more details on the multiplexed pins of these peripherals, see the Device Configurations section of this data sheet.

CPU (DSP core) description

The CPU fetches VelociTI™ advanced very-long instruction words (VLIWs) (256 bits wide) to supply up to eight 32-bit instructions to the eight functional units during every clock cycle. The VelociTI™ VLIW architecture features controls by which all eight units do not have to be supplied with instructions if they are not ready to execute. The first bit of every 32-bit instruction determines if the next instruction belongs to the same execute packet as the previous instruction, or whether it should be executed in the following clock as a part of the next execute packet. Fetch packets are always 256 bits wide; however, the execute packets can vary in size. The variable-length execute packets are a key memory-saving feature, distinguishing the C64x CPUs from other VLIW architectures. The C64x™ VelociTI.2™ extensions add enhancements to the TMS320C62x™ DSP VelociTI™ architecture. These enhancements include:

- Register file enhancements
- Data path extensions
- Quad 8-bit and dual 16-bit extensions with data flow enhancements
- Additional functional unit hardware
- Increased orthogonality of the instruction set
- Additional instructions that reduce code size and increase register flexibility

The CPU features two sets of functional units. Each set contains four units and a register file. One set contains functional units .L1, .S1, .M1, and .D1; the other set contains units .D2, .M2, .S2, and .L2. The two register files each contain 32 32-bit registers for a total of 64 general-purpose registers. In addition to supporting the packed 16-bit and 32-/40-bit fixed-point data types found in the C62x™ VelociTI™ VLIW architecture, the C64x™ register files also support packed 8-bit data and 64-bit fixed-point data types. The two sets of functional units, along with two register files, compose sides A and B of the CPU [see the functional block and CPU (DSP core) diagram, and Figure 1]. The four functional units on each side of the CPU can freely share the 32 registers belonging to that side. Additionally, each side features a “data cross path”—a single data bus connected to all the registers on the other side, by which the two sets of functional units can access data from the register files on the opposite side. The C64x CPU pipelines data-cross-path accesses over multiple clock cycles. This allows the same register to be used as a data-cross-path operand by multiple functional units in the same execute packet. All functional units in the C64x CPU can access operands via the data cross path. Register access by functional units on the same side of the CPU as the register file can service all the units in a single clock cycle. On the C64x CPU, a delay clock is introduced whenever an instruction attempts to read a register via a data cross path if that register was updated in the previous clock cycle.

In addition to the C62x™ DSP fixed-point instructions, the C64x™ DSP includes a comprehensive collection of quad 8-bit and dual 16-bit instruction set extensions. These VelociTI.2™ extensions allow the C64x CPU to operate directly on packed data to streamline data flow and increase instruction set efficiency.

Another key feature of the C64x CPU is the load/store architecture, where all instructions operate on registers (as opposed to data in memory). Two sets of data-addressing units (.D1 and .D2) are responsible for all data transfers between the register files and the memory. The data address driven by the .D units allows data addresses generated from one register file to be used to load or store data to or from the other register file. The C64x .D units can load and store bytes (8 bits), half-words (16 bits), and words (32 bits) with a single instruction. And with the new data path extensions, the C64x .D unit can load and store doublewords (64 bits) with a single instruction. Furthermore, the non-aligned load and store instructions allow the .D units to access words and doublewords on any byte boundary. The C64x CPU supports a variety of indirect addressing modes using either linear- or circular-addressing with 5- or 15-bit offsets. All instructions are conditional, and most can access any one of the 64 registers. Some registers, however, are singled out to support specific addressing modes or to hold the condition for conditional instructions (if the condition is not automatically “true”).

TMS320C62x is a trademark of Texas Instruments.

CPU (DSP core) description (continued)

The two .M functional units perform all multiplication operations. Each of the C64x .M units can perform two 16 × 16-bit multiplies or four 8 × 8-bit multiplies per clock cycle. The .M unit can also perform 16 × 32-bit multiply operations, dual 16 × 16-bit multiplies with add/subtract operations, and quad 8 × 8-bit multiplies with add operations. In addition to standard multiplies, the C64x .M units include bit-count, rotate, Galois field multiplies, and bidirectional variable shift hardware.

The two .S and .L functional units perform a general set of arithmetic, logical, and branch functions with results available every clock cycle. The arithmetic and logical functions on the C64x CPU include single 32-bit, dual 16-bit, and quad 8-bit operations.

The processing flow begins when a 256-bit-wide instruction fetch packet is fetched from a program memory. The 32-bit instructions destined for the individual functional units are “linked” together by “1” bits in the least significant bit (LSB) position of the instructions. The instructions that are “chained” together for simultaneous execution (up to eight in total) compose an execute packet. A “0” in the LSB of an instruction breaks the chain, effectively placing the instructions that follow it in the next execute packet. A C64x™ DSP device enhancement now allows execute packets to cross fetch-packet boundaries. In the TMS320C62x™/TMS320C67x™ DSP devices, if an execute packet crosses the fetch-packet boundary (256 bits wide), the assembler places it in the next fetch packet, while the remainder of the current fetch packet is padded with NOP instructions. In the C64x™ DSP device, the execute boundary restrictions have been removed, thereby, eliminating all of the NOPs added to pad the fetch packet, and thus, decreasing the overall code size. The number of execute packets within a fetch packet can vary from one to eight. Execute packets are dispatched to their respective functional units at the rate of one per clock cycle and the next 256-bit fetch packet is not fetched until all the execute packets from the current fetch packet have been dispatched. After decoding, the instructions simultaneously drive all active functional units for a maximum execution rate of eight instructions every clock cycle. While most results are stored in 32-bit registers, they can be subsequently moved to memory as bytes, half-words, words, or doublewords. All load and store instructions are byte-, half-word-, word-, or doubleword-addressable.

For more details on the C64x CPU functional units enhancements, see the following documents:

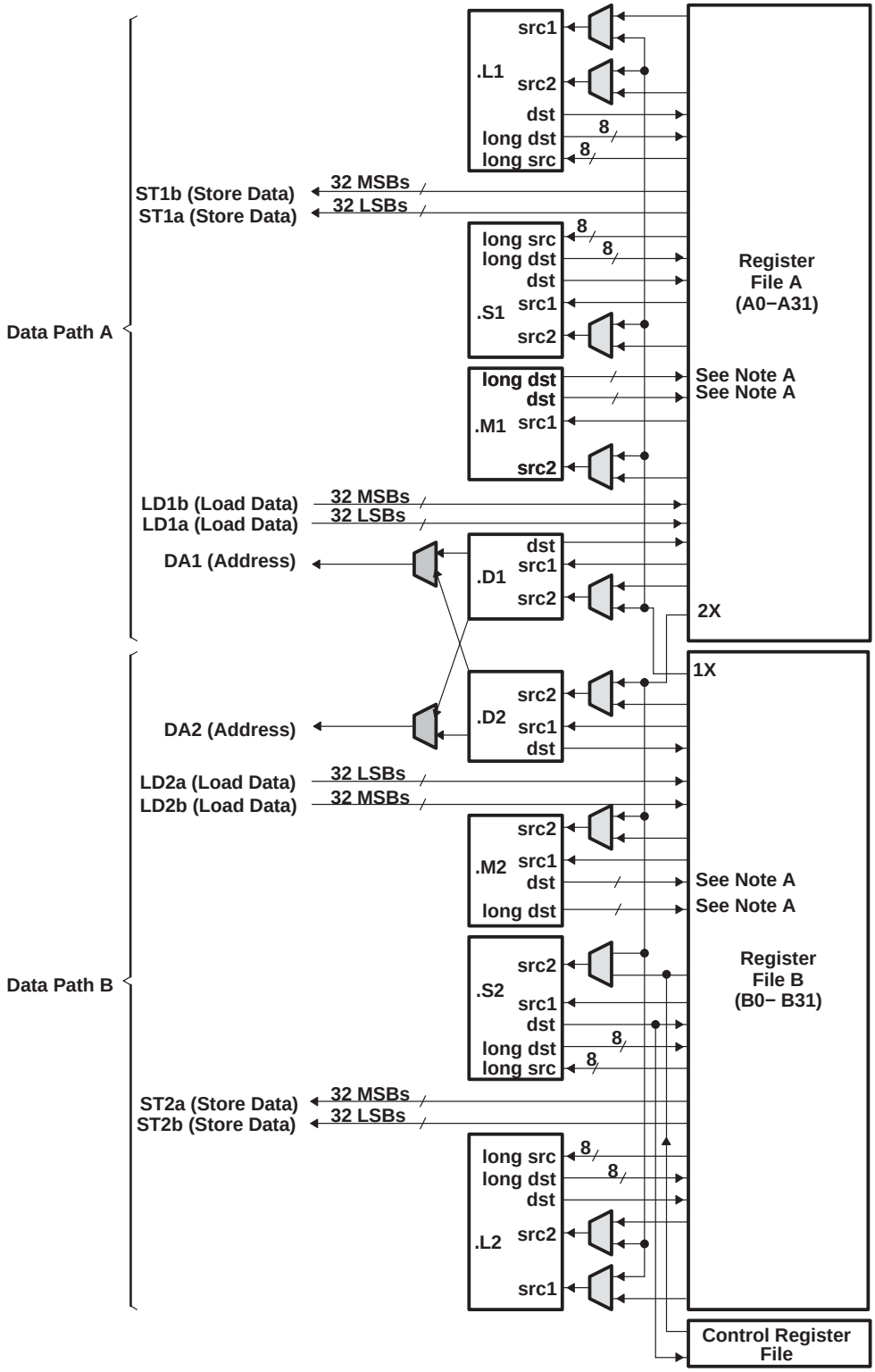
The *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189)

TMS320C64x Technical Overview (literature number SPRU395)

How To Begin Development Today With the TMS320C6411 DSP application report (literature number SPRA374)

TMS320C67x is a trademark of Texas Instruments.

CPU (DSP core) description (continued)



NOTE A: For the .M functional units, the long dst is 32 MSBs and the dst is 32 LSBs.

Figure 1. TMS320C64x™ CPU (DSP Core) Data Paths

memory map summary

Table 3 shows the memory map address ranges of the C6411 device. Internal memory is always located at address 0 and can be used as both program and data memory. The external memory address range in the C6411 device begins at the hex address location 0x8000 0000 for the EMIF.

Table 3. TMS320C6411 Memory Map Summary

MEMORY BLOCK DESCRIPTION	BLOCK SIZE (BYTES)	HEX ADDRESS RANGE
Internal RAM (L2)	256K	0000 0000 – 0003 FFFF
Reserved	24M − 256K	0004 0000 – 017F FFFF
External Memory Interface (EMIF) Registers	256K	0180 0000 – 0183 FFFF
L2 Registers	256K	0184 0000 – 0187 FFFF
HPI Registers	256K	0188 0000 – 018B FFFF
McBSP 0 Registers	256K	018C 0000 – 018F FFFF
McBSP 1 Registers	256K	0190 0000 – 0193 FFFF
Timer 0 Registers	256K	0194 0000 – 0197 FFFF
Timer 1 Registers	256K	0198 0000 – 019B FFFF
Interrupt Selector Registers	256K	019C 0000 – 019F FFFF
EDMA RAM and EDMA Registers	256K	01A0 0000 – 01A3 FFFF
Reserved	512K	01A4 0000 – 01AB FFFF
Timer 2 Registers	256K	01AC 0000 – 01AF FFFF
GPIO Registers	256K	01B0 0000 – 01B3 FFFF
Reserved	768K	01B4 0000 – 01BF FFFF
PCI Registers	256K	01C0 0000 – 01C3 FFFF
Reserved	4M – 256K	01C4 0000 – 01FF FFFF
QDMA Registers	52	0200 0000 – 0200 0033
Reserved	736M – 52	0200 0034 – 2FFF FFFF
McBSP 0 Data	64M	3000 0000 – 33FF FFFF
McBSP 1 Data	64M	3400 0000 – 37FF FFFF
Reserved	1G + 128M	3800 0000 – 7FFF FFFF
EMIF CE0†	256M	8000 0000 – 8FFF FFFF
EMIF CE1†	256M	9000 0000 – 9FFF FFFF
EMIF CE2†	256M	A000 0000 – AFFF FFFF
EMIF CE3†	256M	B000 0000 – BFFF FFFF
Reserved	1G	C000 0000 – FFFF FFFF

† The number of EMIF address pins (EA[22:3]) limits the maximum addressable memory (SDRAM) to 128MB per CE space. To get 256MB of addressable memory, an additional general-purpose output pin or external logic is required.

L2 architecture expanded

Figure 2 shows the detail of the L2 architecture on the TMS320C6411 device. For more information on the L2MODE bits, see the cache configuration (CCFG) register bit field descriptions in the *TMS320C64x Two-Level Internal Memory Reference Guide* (literature number SPRU610).

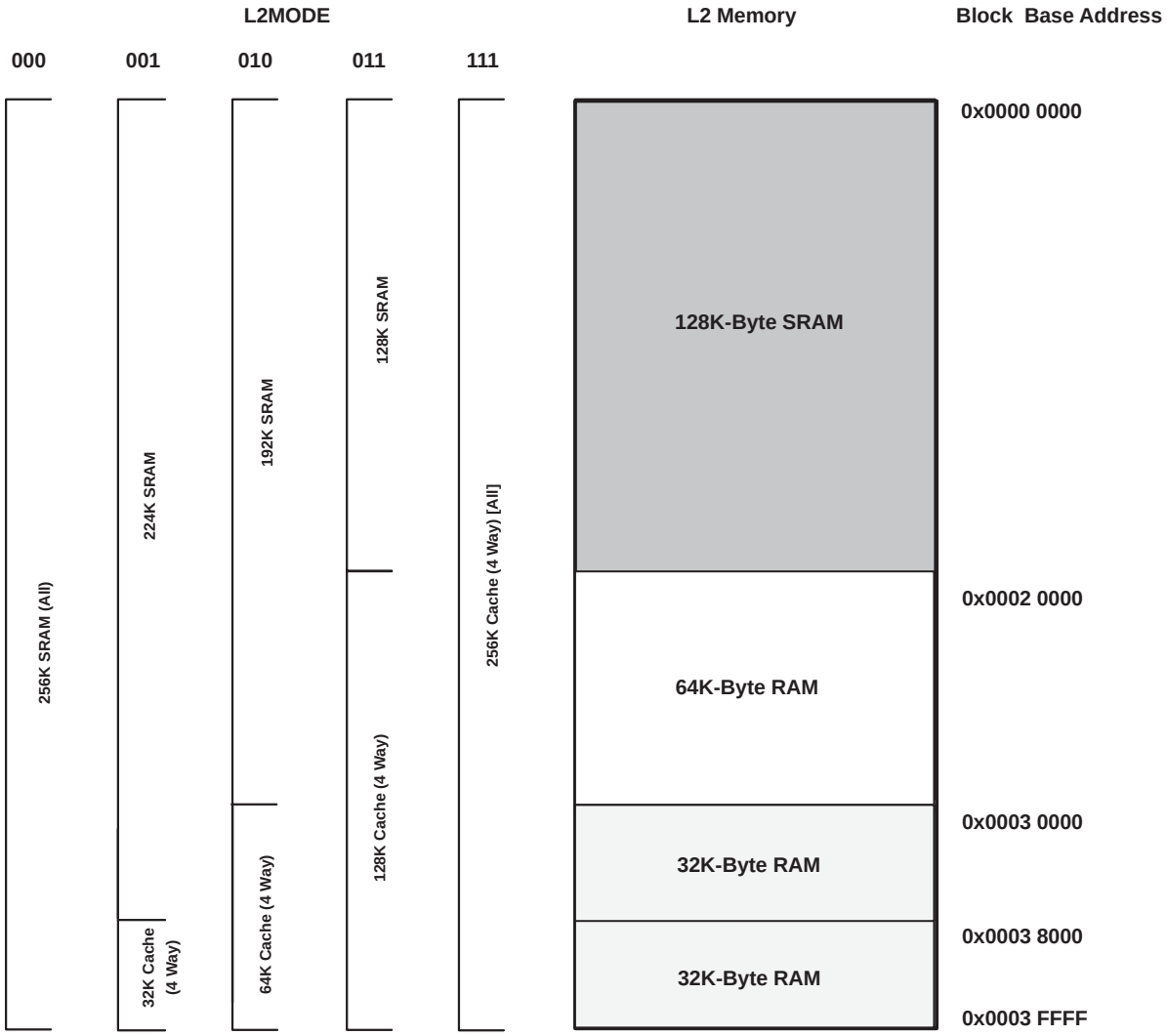


Figure 2. TMS320C6411 L2 Architecture Memory Configuration

peripheral register descriptions

Table 4 through Table 17 identify the peripheral registers for the C6411 device by their register names, acronyms, and hex address or hex address range. For more detailed information on the register contents, bit names and their descriptions, see the specific peripheral reference guide listed in the *TMS320C6000 DSP Peripherals Overview Reference Guide* (literature number SPRU190).

Table 4. EMIF Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
0180 0000	GBLCTL	EMIF global control
0180 0004	CECTL1	EMIF CE1 space control
0180 0008	CECTL0	EMIF CE0 space control
0180 000C	−	Reserved
0180 0010	CECTL2	EMIF CE2 space control
0180 0014	CECTL3	EMIF CE3 space control
0180 0018	SDCTL	EMIF SDRAM control
0180 001C	SDTIM	EMIF SDRAM refresh control
0180 0020	SDEXT	EMIF SDRAM extension
0180 0024 − 0180 003C	−	Reserved
0180 0040	PDTCTL	Peripheral device transfer (PDT) control
0180 0044	CESEC1	EMIF CE1 space secondary control
0180 0048	CESEC0	EMIF CE0 space secondary control
0180 004C	−	Reserved
0180 0050	CESEC2	EMIF CE2 space secondary control
0180 0054	CESEC3	EMIF CE3 space secondary control
0180 0058 − 0183 FFFF	–	Reserved

peripheral register descriptions (continued)

Table 5. L2 Cache Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 0000	CCFG	Cache configuration register	
0184 0004 −0184 0FFC	−	Reserved	
0184 1000	EDMAWEIGHT	L2 EDMA access control register	
0184 1004 − 0184 1FFC	−	Reserved	
0184 2000	L2ALLOC0	L2 allocation register 0	
0184 2004	L2ALLOC1	L2 allocation register 1	
0184 2008	L2ALLOC2	L2 allocation register 2	
0184 200C	L2ALLOC3	L2 allocation register 3	
0184 2010 −0184 3FFF	−	Reserved	
0184 4000	L2WBAR	L2 writeback base address register	
0184 4004	L2WWC	L2 writeback word count register	
0184 4010	L2WIBAR	L2 writeback-invalidate base address register	
0184 4014	L2WIWC	L2 writeback-invalidate word count register	
0184 4018	L2IBAR	L2 invalidate base address register	
0184 401C	L2IWC	L2 invalidate word count register	
0184 4020	L1PIBAR	L1P invalidate base address register	
0184 4024	L1PIWC	L1P invalidate word count register	
0184 4030	L1DWIBAR	L1D writeback-invalidate base address register	
0184 4034	L1DWIWC	L1D writeback-invalidate word count register	
0184 4038 −0184 4044	−	Reserved	
0184 4048	L1DIBAR	L1D invalidate base address register	
0184 404C	L1DIWC	L1D invalidate word count register	
0184 4050 −0184 4FFC	−	Reserved	
0184 5000	L2WB	L2 writeback all register	
0184 5004	L2WBINV	L2 writeback-invalidate all register	
0184 5008 −0184 7FFF	−	Reserved	
0184 8000 −0184 81FC	MAR0 to MAR127	Reserved	
0184 8200	MAR128	Controls EMIF CE0 range 8000 0000 − 80FF FFFF	
0184 8204	MAR129	Controls EMIF CE0 range 8100 0000 − 81FF FFFF	
0184 8208	MAR130	Controls EMIF CE0 range 8200 0000 − 82FF FFFF	
0184 820C	MAR131	Controls EMIF CE0 range 8300 0000 − 83FF FFFF	
0184 8210	MAR132	Controls EMIF CE0 range 8400 0000 − 84FF FFFF	
0184 8214	MAR133	Controls EMIF CE0 range 8500 0000 − 85FF FFFF	
0184 8218	MAR134	Controls EMIF CE0 range 8600 0000 − 86FF FFFF	
0184 821C	MAR135	Controls EMIF CE0 range 8700 0000 − 87FF FFFF	
0184 8220	MAR136	Controls EMIF CE0 range 8800 0000 − 88FF FFFF	
0184 8224	MAR137	Controls EMIF CE0 range 8900 0000 − 89FF FFFF	
0184 8228	MAR138	Controls EMIF CE0 range 8A00 0000 − 8AFF FFFF	
0184 822C	MAR139	Controls EMIF CE0 range 8B00 0000 − 8BFF FFFF	

peripheral register descriptions (continued)

Table 5. L2 Cache Registers (Continued)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 8230	MAR140	Controls EMIF CE0 range 8C00 0000 − 8CFF FFFF	
0184 8234	MAR141	Controls EMIF CE0 range 8D00 0000 − 8DFF FFFF	
0184 8238	MAR142	Controls EMIF CE0 range 8E00 0000 − 8EFF FFFF	
0184 823C	MAR143	Controls EMIF CE0 range 8F00 0000 − 8FFF FFFF	
0184 8240	MAR144	Controls EMIF CE1 range 9000 0000 − 90FF FFFF	
0184 8244	MAR145	Controls EMIF CE1 range 9100 0000 − 91FF FFFF	
0184 8248	MAR146	Controls EMIF CE1 range 9200 0000 − 92FF FFFF	
0184 824C	MAR147	Controls EMIF CE1 range 9300 0000 − 93FF FFFF	
0184 8250	MAR148	Controls EMIF CE1 range 9400 0000 − 94FF FFFF	
0184 8254	MAR149	Controls EMIF CE1 range 9500 0000 − 95FF FFFF	
0184 8258	MAR150	Controls EMIF CE1 range 9600 0000 − 96FF FFFF	
0184 825C	MAR151	Controls EMIF CE1 range 9700 0000 − 97FF FFFF	
0184 8260	MAR152	Controls EMIF CE1 range 9800 0000 − 98FF FFFF	
0184 8264	MAR153	Controls EMIF CE1 range 9900 0000 − 99FF FFFF	
0184 8268	MAR154	Controls EMIF CE1 range 9A00 0000 − 9AFF FFFF	
0184 826C	MAR155	Controls EMIF CE1 range 9B00 0000 − 9BFF FFFF	
0184 8270	MAR156	Controls EMIF CE1 range 9C00 0000 − 9CFF FFFF	
0184 8274	MAR157	Controls EMIF CE1 range 9D00 0000 − 9DFF FFFF	
0184 8278	MAR158	Controls EMIF CE1 range 9E00 0000 − 9EFF FFFF	
0184 827C	MAR159	Controls EMIF CE1 range 9F00 0000 − 9FFF FFFF	
0184 8280	MAR160	Controls EMIF CE2 range A000 0000 − A0FF FFFF	
0184 8284	MAR161	Controls EMIF CE2 range A100 0000 − A1FF FFFF	
0184 8288	MAR162	Controls EMIF CE2 range A200 0000 − A2FF FFFF	
0184 828C	MAR163	Controls EMIF CE2 range A300 0000 − A3FF FFFF	
0184 8290	MAR164	Controls EMIF CE2 range A400 0000 − A4FF FFFF	
0184 8294	MAR165	Controls EMIF CE2 range A500 0000 − A5FF FFFF	
0184 8298	MAR166	Controls EMIF CE2 range A600 0000 − A6FF FFFF	
0184 829C	MAR167	Controls EMIF CE2 range A700 0000 − A7FF FFFF	
0184 82A0	MAR168	Controls EMIF CE2 range A800 0000 − A8FF FFFF	
0184 82A4	MAR169	Controls EMIF CE2 range A900 0000 − A9FF FFFF	
0184 82A8	MAR170	Controls EMIF CE2 range AA00 0000 − AAFF FFFF	
0184 82AC	MAR171	Controls EMIF CE2 range AB00 0000 − ABFF FFFF	
0184 82B0	MAR172	Controls EMIF CE2 range AC00 0000 − ACFF FFFF	
0184 82B4	MAR173	Controls EMIF CE2 range AD00 0000 − ADFF FFFF	
0184 82B8	MAR174	Controls EMIF CE2 range AE00 0000 − AEFF FFFF	
0184 82BC	MAR175	Controls EMIF CE2 range AF00 0000 − AFFF FFFF	
0184 82C0	MAR176	Controls EMIF CE3 range B000 0000 − B0FF FFFF	
0184 82C4	MAR177	Controls EMIF CE3 range B100 0000 − B1FF FFFF	
0184 82C8	MAR178	Controls EMIF CE3 range B200 0000 − B2FF FFFF	
0184 82CC	MAR179	Controls EMIF CE3 range B300 0000 − B3FF FFFF	
0184 82D0	MAR180	Controls EMIF CE3 range B400 0000 − B4FF FFFF	
0184 82D4	MAR181	Controls EMIF CE3 range B500 0000 − B5FF FFFF	

peripheral register descriptions (continued)

Table 5. L2 Cache Registers (Continued)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 82D8	MAR182	Controls EMIF CE3 range B600 0000 − B6FF FFFF	
0184 82DC	MAR183	Controls EMIF CE3 range B700 0000 − B7FF FFFF	
0184 82E0	MAR184	Controls EMIF CE3 range B800 0000 − B8FF FFFF	
0184 82E4	MAR185	Controls EMIF CE3 range B900 0000 − B9FF FFFF	
0184 82E8	MAR186	Controls EMIF CE3 range BA00 0000 − BAFF FFFF	
0184 82EC	MAR187	Controls EMIF CE3 range BB00 0000 − BBFF FFFF	
0184 82F0	MAR188	Controls EMIF CE3 range BC00 0000 − BCFF FFFF	
0184 82F4	MAR189	Controls EMIF CE3 range BD00 0000 − BDFF FFFF	
0184 82F8	MAR190	Controls EMIF CE3 range BE00 0000 − BEFF FFFF	
0184 82FC	MAR191	Controls EMIF CE3 range BF00 0000 − BFFF FFFF	
0184 8300 −0184 83FC	MAR192 to MAR255	Reserved	
0184 8400 −0187 FFFF	−	Reserved	

Table 6. EDMA Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
01A0 FF9C	EPRH	Event polarity high register
01A0 FFA4	CIPRH	Channel interrupt pending high register
01A0 FFA8	CIERH	Channel interrupt enable high register
01A0 FFAC	CCERH	Channel chain enable high register
01A0 FFB0	ERH	Event high register
01A0 FFB4	EERH	Event enable high register
01A0 FFB8	ECRH	Event clear high register
01A0 FFBC	ESRH	Event set high register
01A0 FFC0	PQAR0	Priority queue allocation register 0
01A0 FFC4	PQAR1	Priority queue allocation register 1
01A0 FFC8	PQAR2	Priority queue allocation register 2
01A0 FFCC	PQAR3	Priority queue allocation register 3
01A0 FFDC	EPRL	Event polarity low register
01A0 FFE0	PQSR	Priority queue status register
01A0 FFE4	CIPRL	Channel interrupt pending low register
01A0 FFE8	CIERL	Channel interrupt enable low register
01A0 FFEC	CCERL	Channel chain enable low register
01A0 FFF0	ERL	Event low register
01A0 FFF4	EERL	Event enable low register
01A0 FFF8	ECRL	Event clear low register
01A0 FFFC	ESRL	Event set low register
01A1 0000 − 01A3 FFFF	–	Reserved

peripheral register descriptions (continued)

Table 7. EDMA Parameter RAM†

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
01A0 0000 − 01A0 0017	−	Parameters for Event 0 (6 words)	
01A0 0018 − 01A0 002F	−	Parameters for Event 1 (6 words)	
01A0 0030 − 01A0 0047	−	Parameters for Event 2 (6 words)	
01A0 0048 − 01A0 005F	−	Parameters for Event 3 (6 words)	
01A0 0060 − 01A0 0077	−	Parameters for Event 4 (6 words)	
01A0 0078 − 01A0 008F	−	Parameters for Event 5 (6 words)	
01A0 0090 − 01A0 00A7	−	Parameters for Event 6 (6 words)	
01A0 00A8 − 01A0 00BF	−	Parameters for Event 7 (6 words)	
01A0 00C0 − 01A0 00D7	−	Parameters for Event 8 (6 words)	
01A0 00D8 − 01A0 00EF	−	Parameters for Event 9 (6 words)	
01A0 00F0 − 01A0 00107	−	Parameters for Event 10 (6 words)	
01A0 0108 − 01A0 011F	−	Parameters for Event 11 (6 words)	
01A0 0120 − 01A0 0137	−	Parameters for Event 12 (6 words)	
01A0 0138 − 01A0 014F	−	Parameters for Event 13 (6 words)	
01A0 0150 − 01A0 0167	−	Parameters for Event 14 (6 words)	
01A0 0168 − 01A0 017F	−	Parameters for Event 15 (6 words)	
01A0 0150 − 01A0 0167	−	Parameters for Event 16 (6 words)	
01A0 0168 − 01A0 017F	−	Parameters for Event 17 (6 words)	
...		...	
...		...	
01A0 05D0 − 01A0 05E7	−	Parameters for Event 62 (6 words)	
01A0 05E8 − 01A0 05FF	−	Parameters for Event 63 (6 words)	
01A0 0600 − 01A0 0617	−	Reload/link parameters for Event M (6 words)	
01A0 0618 − 01A0 062F	−	Reload/link parameters for Event N (6 words)	
...		...	
01A0 07E0 − 01A0 07F7	−	Reload/link parameters for Event Z (6 words)	
01A0 07F8 − 01A0 07FF	−	Scratch pad area (2 words)	

† The C6411 device has twenty-one parameter sets [six (6) words each] that can be used to reload/link EDMA transfers.

Table 8. Quick DMA (QDMA) and Pseudo Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
0200 0000	QOPT	QDMA options parameter register
0200 0004	QSRC	QDMA source address register
0200 0008	QCNT	QDMA frame count register
0200 000C	QDST	QDMA destination address register
0200 0010	QIDX	QDMA index register
0200 0014 − 0200 001C		Reserved
0200 0020	QSOPT	QDMA pseudo options register
0200 0024	QSSRC	QDMA psuedo source address register
0200 0028	QSCNT	QDMA psuedo frame count register
0200 002C	QSDST	QDMA destination address register
0200 0030	QSIDX	QDMA psuedo index register

peripheral register descriptions (continued)

Table 9. Interrupt Selector Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
019C 0000	MUXH	Interrupt multiplexer high	Selects which interrupts drive CPU interrupts 10−15 (INT10−INT15)
019C 0004	MUXL	Interrupt multiplexer low	Selects which interrupts drive CPU interrupts 4−9 (INT04−INT09)
019C 0008	EXTPOL	External interrupt polarity	Sets the polarity of the external interrupts (EXT_INT4−EXT_INT7)
019C 000C − 019C 01FF	−	Reserved	

Table 10. McBSP 0 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
018C 0000	DRR0	McBSP0 data receive register via Configuration Bus	The CPU and EDMA controller can only read this register; they cannot write to it.
0x3000 0000 − 0x33FF FFFF	DRR0	McBSP0 data receive register via Peripheral Data Bus	
018C 0004	DXR0	McBSP0 data transmit register via Configuration Bus	
0x3000 0000 − 0x33FF FFFF	DXR0	McBSP0 data transmit register via Peripheral Data Bus	
018C 0008	SPCR0	McBSP0 serial port control register	
018C 000C	RCR0	McBSP0 receive control register	
018C 0010	XCR0	McBSP0 transmit control register	
018C 0014	SRGR0	McBSP0 sample rate generator register	
018C 0018	MCR0	McBSP0 multichannel control register	
018C 001C	RCERE00	McBSP0 enhanced receive channel enable register 0	
018C 0020	XCERE00	McBSP0 enhanced transmit channel enable register 0	
018C 0024	PCR0	McBSP0 pin control register	
018C 0028	RCERE10	McBSP0 enhanced receive channel enable register 1	
018C 002C	XCERE10	McBSP0 enhanced transmit channel enable register 1	
018C 0030	RCERE20	McBSP0 enhanced receive channel enable register 2	
018C 0034	XCERE20	McBSP0 enhanced transmit channel enable register 2	
018C 0038	RCERE30	McBSP0 enhanced receive channel enable register 3	
018C 003C	XCERE30	McBSP0 enhanced transmit channel enable register 3	
018C 0040 − 018F FFFF	–	Reserved	

peripheral register descriptions (continued)

Table 11. McBSP 1 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0190 0000	DRR1	Data receive register via Configuration Bus	The CPU and EDMA controller can only read this register; they cannot write to it.
0x3400 0000 − 0x37FF FFFF	DRR1	McBSP1 data receive register via Peripheral Data Bus	
0190 0004	DXR1	McBSP1 data transmit register via Configuration Bus	
0x3400 0000 − 0x37FF FFFF	DXR1	McBSP1 data transmit register via Peripheral Data Bus	
0190 0008	SPCR1	McBSP1 serial port control register	
0190 000C	RCR1	McBSP1 receive control register	
0190 0010	XCR1	McBSP1 transmit control register	
0190 0014	SRGR1	McBSP1 sample rate generator register	
0190 0018	MCR1	McBSP1 multichannel control register	
0190 001C	RCERE01	McBSP1 enhanced receive channel enable register 0	
0190 0020	XCERE01	McBSP1 enhanced transmit channel enable register 0	
0190 0024	PCR1	McBSP1 pin control register	
0190 0028	RCERE11	McBSP1 enhanced receive channel enable register 1	
0190 002C	XCERE11	McBSP1 enhanced transmit channel enable register 1	
0190 0030	RCERE21	McBSP1 enhanced receive channel enable register 2	
0190 0034	XCERE21	McBSP1 enhanced transmit channel enable register 2	
0190 0038	RCERE31	McBSP1 enhanced receive channel enable register 3	
0190 003C	XCERE31	McBSP1 enhanced transmit channel enable register 3	
0190 0040 − 0193 FFFF	–	Reserved	

peripheral register descriptions (continued)

Table 12. Timer 0 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0194 0000	CTL0	Timer 0 control register	Determines the operating mode of the timer, monitors the timer status, and controls the function of the TOUT pin.
0194 0004	PRD0	Timer 0 period register	Contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency.
0194 0008	CNT0	Timer 0 counter register	Contains the current value of the incrementing counter.
0194 000C − 0197 FFFF	−	Reserved	

Table 13. Timer 1 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0198 0000	CTL1	Timer 1 control register	Determines the operating mode of the timer, monitors the timer status, and controls the function of the TOUT pin.
0198 0004	PRD1	Timer 1 period register	Contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency.
0198 0008	CNT1	Timer 1 counter register	Contains the current value of the incrementing counter.
0198 000C − 019B FFFF	−	Reserved	

Table 14. Timer 2 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
01AC 0000	CTL2	Timer 2 control register	Determines the operating mode of the timer, monitors the timer status, and controls the function of the TOUT pin.
01AC 0004	PRD2	Timer 2 period register	Contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency.
01AC 0008	CNT2	Timer 2 counter register	Contains the current value of the incrementing counter.
01AC 000C − 01AF FFFF	−	Reserved	

peripheral register descriptions (continued)

Table 15. HPI Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
−	HPID	HPI data register	Host read/write access only
0188 0000	HPIC	HPI control register	HPIC has both Host/CPU read/write access
0188 0004	HPIA (HPIAW)†	HPI address register (Write)	HPIA has both Host/CPU read/write access
0188 0008	HPIA (HPIAR)†	HPI address register (Read)	
0188 000C − 0189 FFFF	−	Reserved	
018A 0000	TRCTL	HPI transfer request control register	
018A 0004 − 018B FFFF	−	Reserved	

† Host access to the HPIA register updates both the HPIAW and HPIAR registers. The CPU can access HPIAW and HPIAR independently.

Table 16. GPIO Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
01B0 0000	GPEN	GPIO enable register
01B0 0004	GPDIR	GPIO direction register
01B0 0008	GPVAL	GPIO value register
01B0 000C	−	Reserved
01B0 0010	GPDH	GPIO delta high register
01B0 0014	GPHM	GPIO high mask register
01B0 0018	GPDL	GPIO delta low register
01B0 001C	GPLM	GPIO low mask register
01B0 0020	GPGC	GPIO global control register
01B0 0024	GPPOL	GPIO interrupt polarity register
01B0 0028 − 01B0 01FF	−	Reserved
01B0 0200	DEVICE_REV	Silicon Revision Identification Register (For more details, see the device characteristics listed in Table 1.)
01B0 0204 − 01B3 FFFF	−	Reserved

peripheral register descriptions (continued)

Table 17. PCI Peripheral Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
01C0 0000	RSTSRC	DSP Reset source/status register
01C0 0004	–	Reserved
01C0 0008	PCIIS	PCI interrupt source register
01C0 000C	PCIIEN	PCI interrupt enable register
01C0 0010	DSPMA	DSP master address register
01C0 0014	PCIMA	PCI master address register
01C0 0018	PCIMC	PCI master control register
01C0 001C	CDSPA	Current DSP address register
01C0 0020	CPCIA	Current PCI address register
01C0 0024	CCNT	Current byte count register
01C0 0028	−	Reserved
01C0 002C − 01C1 FFEF	–	Reserved
0x01C1 FFF0	HSR	Host status register
0x01C1 FFF4	HDCR	Host-to-DSP control register
0x01C1 FFF8	DSPP	DSP page register
0x01C1 FFFC	−	Reserved
01C2 0000	EEADD	EEPROM address register
01C2 0004	EEDAT	EEPROM data register
01C2 0008	EECTL	EEPROM control register
01C2 000C − 01C2 FFFF	–	Reserved
01C3 0000	TRCTL	PCI transfer request control register
01C3 0004 − 01C3 FFFF	–	Reserved

EDMA channel synchronization events

The C64x EDMA supports up to 64 EDMA channels which service peripheral devices and external memory. Table 18 lists the source of C64x EDMA synchronization events associated with each of the programmable EDMA channels. For the C6411 device, the association of an event to a channel is fixed; each of the EDMA channels has one specific event associated with it. These specific events are captured in the EDMA event registers (ERL, ERH) even if the events are disabled by the EDMA event enable registers (EERL, EERH). The priority of each event can be specified independently in the transfer parameters stored in the EDMA parameter RAM. For more detailed information on the EDMA module and how EDMA events are enabled, captured, processed, linked, chained, and cleared, etc., see the *TMS320C6000 DSP Enhanced Direct Memory Access (EDMA) Controller Reference Guide* (literature number SPRU234).

Table 18. TMS320C6411 EDMA Channel Synchronization Events†

EDMA CHANNEL	EVENT NAME	EVENT DESCRIPTION
0	DSP_INT	HPI/PCI-to-DSP interrupt
1	TINT0	Timer 0 interrupt
2	TINT1	Timer 1 interrupt
3	SD_INT	EMIF SDRAM timer interrupt
4	GPINT4/EXT_INT4	GPIO event 4/External interrupt pin 4
5	GPINT5/EXT_INT5	GPIO event 5/External interrupt pin 5
6	GPINT6/EXT_INT6	GPIO event 6/External interrupt pin 6
7	GPINT7/EXT_INT7	GPIO event 7/External interrupt pin 7
8	GPINT0	GPIO event 0
9	GPINT1	GPIO event 1
10	GPINT2	GPIO event 2
11	GPINT3	GPIO event 3
12	XEVT0	McBSP0 transmit event
13	REVT0	McBSP0 receive event
14	XEVT1	McBSP1 transmit event
15	REVT1	McBSP1 receive event
16−18	–	None
19	TINT2	Timer 2 interrupt
20−47	–	None
48	GPINT8	GPIO event 8
49	GPINT9	GPIO event 9
50	GPINT10	GPIO event 10
51	GPINT11	GPIO event 11
52	GPINT12	GPIO event 12
53	GPINT13	GPIO event 13
54	GPINT14	GPIO event 14
55	GPINT15	GPIO event 15
56−63	–	None

† In addition to the events shown in this table, each of the 64 channels can also be synchronized with the transfer completion or alternate transfer completion events. For more detailed information on EDMA event-transfer chaining, see the *TMS320C6000 DSP Enhanced Direct Memory Access (EDMA) Controller Reference Guide* (literature number SPRU234).

interrupt sources and interrupt selector

The C64x DSP core supports 16 prioritized interrupts, which are listed in Table 19. The highest-priority interrupt is INT_00 (dedicated to RESET) while the lowest-priority interrupt is INT_15. The first four interrupts (INT_00−INT_03) are non-maskable and fixed. The remaining interrupts (INT_04−INT_15) are maskable and default to the interrupt source specified in Table 19. The interrupt source for interrupts 4−15 can be programmed by modifying the selector value (binary value) in the corresponding fields of the Interrupt Selector Control registers: MUXH (address 0x019C0000) and MUXL (address 0x019C0004).

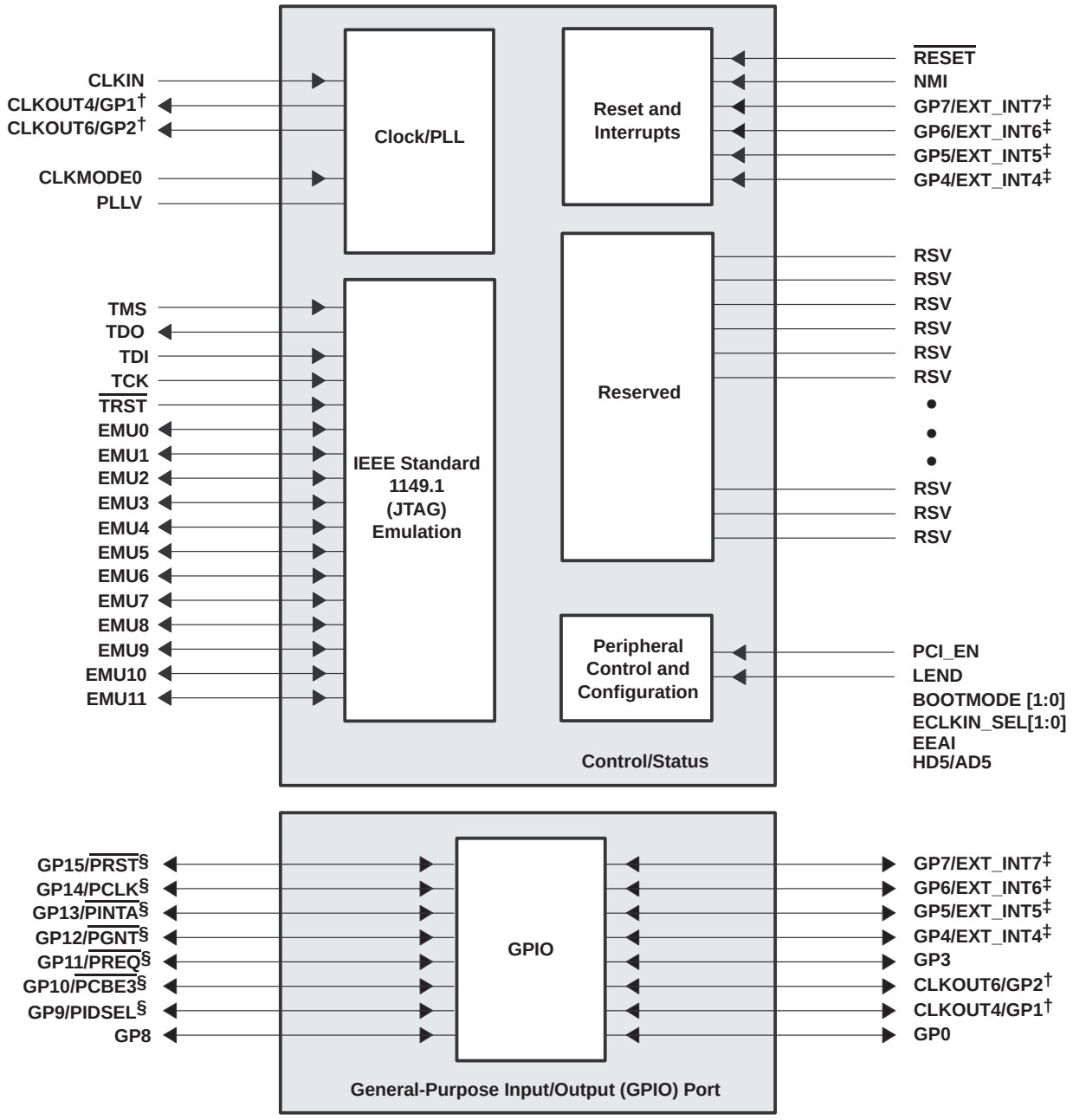
Table 19. C6411 DSP Interrupts

CPU INTERRUPT NUMBER	INTERRUPT SELECTOR CONTROL REGISTER	SELECTOR VALUE (BINARY)	INTERRUPT EVENT	INTERRUPT SOURCE
INT_00†	−	−	RESET	
INT_01†	−	−	NMI	
INT_02†	−	−	Reserved	Reserved. Do not use.
INT_03†	−	−	Reserved	Reserved. Do not use.
INT_04‡	MUXL[4:0]	00100	GPINT4/EXT_INT4	GPIO interrupt 4/External interrupt pin 4
INT_05‡	MUXL[9:5]	00101	GPINT5/EXT_INT5	GPIO interrupt 5/External interrupt pin 5
INT_06‡	MUXL[14:10]	00110	GPINT6/EXT_INT6	GPIO interrupt 6/External interrupt pin 6
INT_07‡	MUXL[20:16]	00111	GPINT7/EXT_INT7	GPIO interrupt 7/External interrupt pin 7
INT_08‡	MUXL[25:21]	01000	EDMA_INT	EDMA channel (0 through 63) interrupt
INT_09‡	MUXL[30:26]	01001	EMU_DTDMA	EMU DTDMA
INT_10‡	MUXH[4:0]	00011	SD_INT	EMIF SDRAM timer interrupt
INT_11‡	MUXH[9:5]	01010	EMU_RTDXRX	EMU real-time data exchange (RTDX) receive
INT_12‡	MUXH[14:10]	01011	EMU_RTDXTX	EMU RTDX transmit
INT_13‡	MUXH[20:16]	00000	DSP_INT	HPI/PCI-to-DSP interrupt
INT_14‡	MUXH[25:21]	00001	TINT0	Timer 0 interrupt
INT_15‡	MUXH[30:26]	00010	TINT1	Timer 1 interrupt
−	−	01100	XINT0	McBSP0 transmit interrupt
−	−	01101	RINT0	McBSP0 receive interrupt
−	−	01110	XINT1	McBSP1 transmit interrupt
−	−	01111	RINT1	McBSP1 receive interrupt
−	−	10000	GPINT0	GPIO interrupt 0
−	−	10001	Reserved	Reserved. Do not use.
−	−	10010	Reserved	Reserved. Do not use.
−	−	10011	TINT2	Timer 2 interrupt
−	−	10100 − 11111	Reserved	Reserved. Do not use.

† Interrupts INT_00 through INT_03 are non-maskable and fixed.

‡ Interrupts INT_04 through INT_15 are programmable by modifying the binary selector values in the Interrupt Selector Control registers fields. Table 19 shows the default interrupt sources for Interrupts INT_04 through INT_15. For more detailed information on interrupt sources and selection, see the *TMS320C6000 DSP Interrupt Selector Reference Guide* (literature number SPRU646).

signal groups description



† These pins are muxed with the GPIO port pins and by default these signals function as clocks (CLKOUT4 or CLKOUT6). To use these muxed pins as GPIO signals, the appropriate GPIO register bits (GPxEN and GPxDIR) must be properly enabled and configured. For more details, see the Device Configurations section of this data sheet.

‡ These pins are GPIO pins that can also function as external interrupt sources (EXT_INT[7:4]). Default after reset is EXT_INTx or GPIO as input-only.

§ These GPIO pins are muxed with the PCI peripheral pins and by default these signals are set up to no function with both the GPIO and PCI pin functions *disabled*. For more details on these muxed pins, see the Device Configurations section of this data sheet.

Figure 3. CPU and Peripheral Signals

signal groups description (continued)

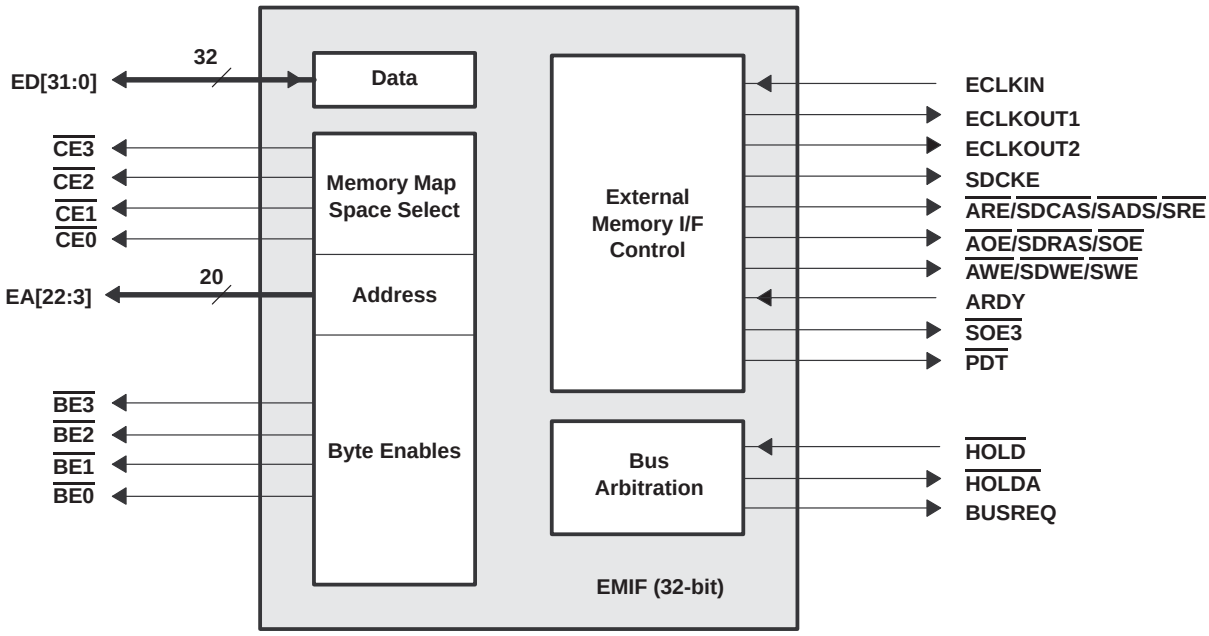
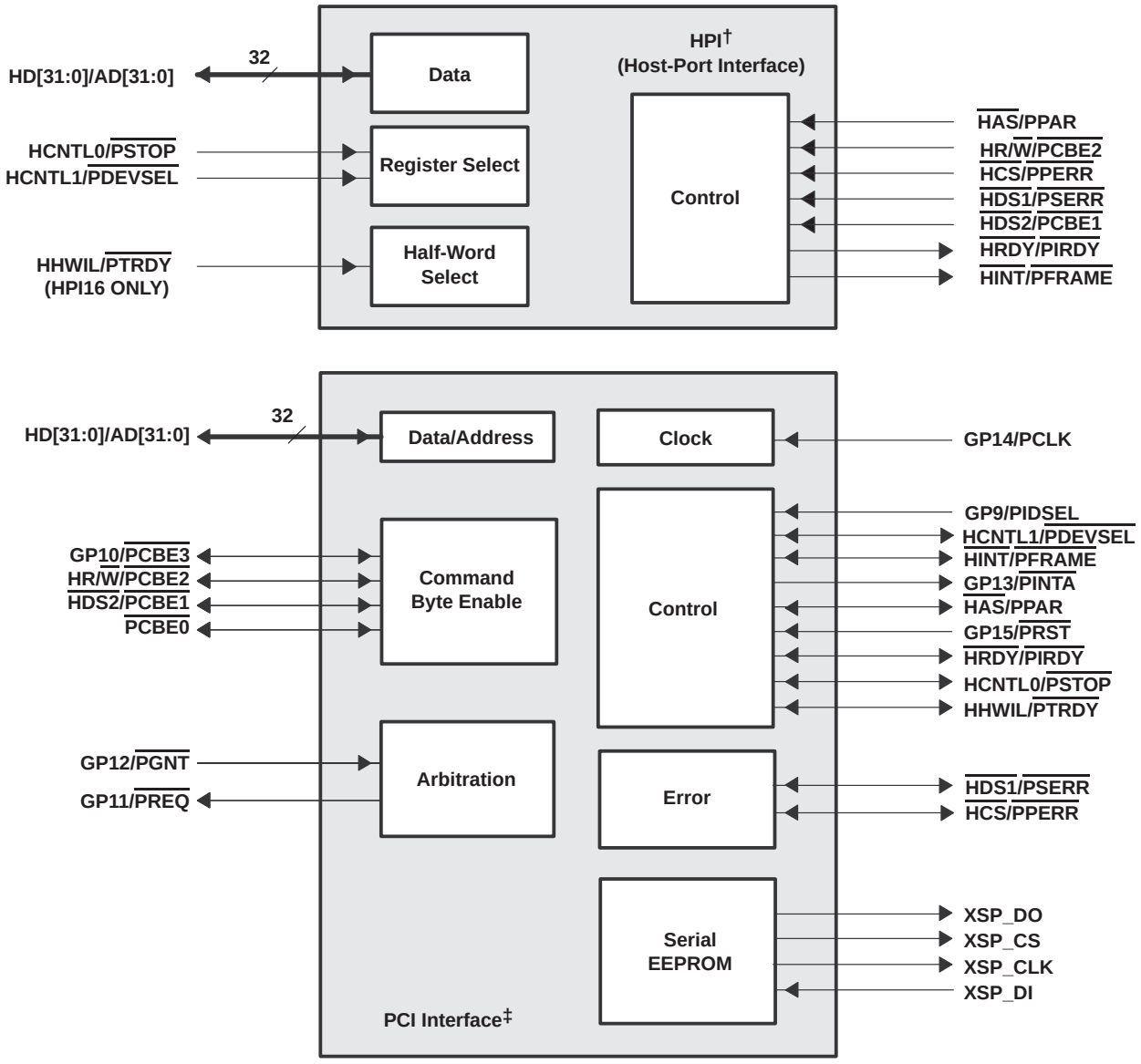


Figure 4. Peripheral Signals

signal groups description (continued)



† These HPI pins are muxed with the PCI peripheral. By default, these signals function as HPI. For more details on these muxed pins, see the Device Configurations section of this data sheet.

‡ These PCI pins (excluding PCBE0 , XSP_DO, XSP_CLK, XSP_DI, and XSP_CS) are muxed with the HPI or GPIO peripherals. By default, these signals function as HPI and no function, respectively. For more details on these muxed pins, see the Device Configurations section of this data sheet.

Figure 4. Peripheral Signals (Continued)

signal groups description (continued)

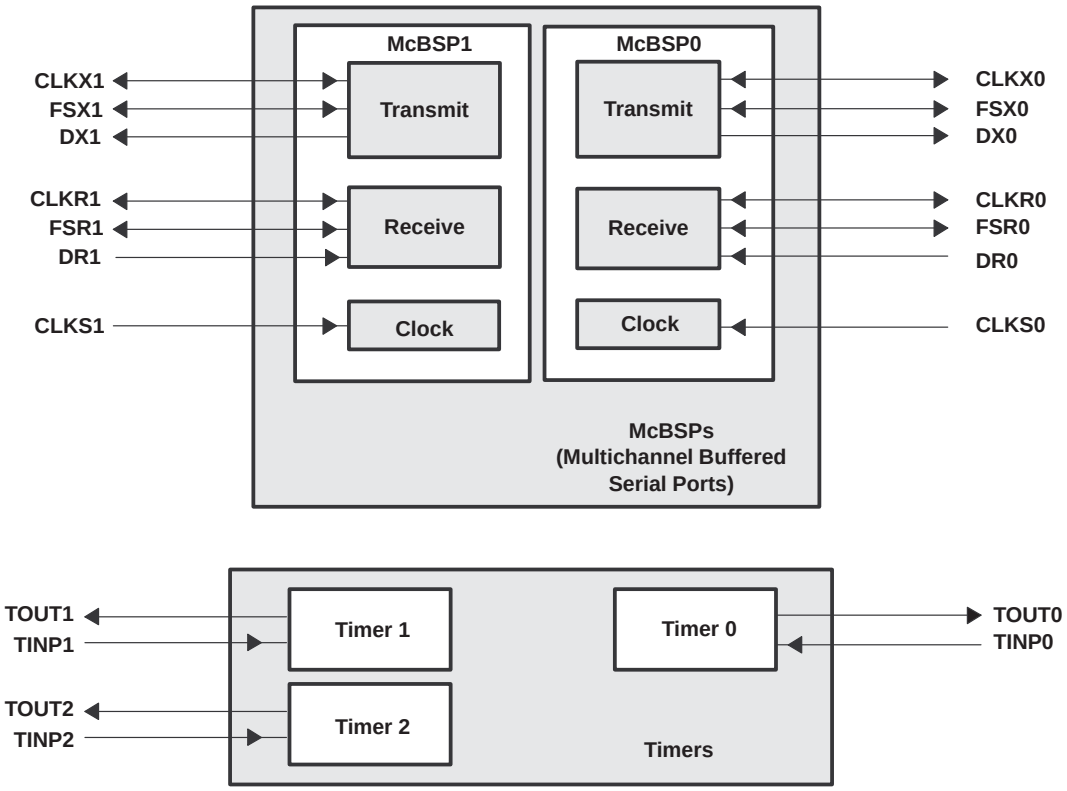


Figure 4. Peripheral Signals (Continued)

DEVICE CONFIGURATIONS

The C6411 peripheral selections and other device configurations are determined by external pullup/pulldown resistors on the following pins (all of which are latched during device reset):

- peripherals selection
  - PCI_EN
- other device configurations
  - LEND
  - BOOTMODE [1:0]
  - ECLKIN_SEL[1:0]
  - EEAI
  - HD5/AD5

peripherals selection

Some C6411 peripherals share the same pins (internally muxed) and are mutually exclusive (i.e., HPI, general-purpose input/output pins GP[15:9], and PCI). Other C6411 peripherals (i.e., EMIF, three Timers, two McBSPs, and the GP[8:0] pins), are always available.

- HPI/GP[15:9] versus PCI

  The PCI_EN pin is latched at reset. This pin determines the HPI/GP[15:9] versus the PCI peripheral selection, summarized in Table 20.

DEVICE CONFIGURATIONS (CONTINUED)

Table 20. PCI_EN Peripheral Selection (HPI/GP[15:9] or PCI)

PCI_EN Pin [AA4]†	PERIPHERALS SELECTED			
	HPI	GP[15:9]	PCI	DESCRIPTION
0	√	√		[default] HPI is enabled, GP[15:9] pins can be programmed as GPIO, PCI is disabled. This means all multiplexed HPI/PCI pins function as HPI and all standalone PCI pins ($\overline{\text{PCBE0}}$, XSP_DO, XSP_DI, XSP_CLK, and XSP_CS) are tied-off (Hi-Z). Also, the multiplexed GPIO/PCI pins can be used as GPIO with the proper software configuration of the GPIO enable (GPxEN) and direction (GPxDIR) registers (for more details, see Table 22).
1			√	PCI is enabled, HPI/GP[15:9] are disabled. This means all multiplexed HPI/PCI pins function as PCI. Also, the multiplexed GPIO/PCI pins function as PCI pins (for more details, see Table 22). Auto-initialization through PCI EEPROM or initialization via specified PCI default values is controlled by the EEAI pin, see Table 21.

† The PCI_EN pin is latched at reset and *must* be driven valid at all times and the user *must not* switch values throughout device operation.

other device configurations

Table 21 describes the C6411 device configuration pins, which are set up via external pullup/pulldown resistors through the specified pins. For more details on these device configuration pins, see the Terminal Functions table and the Debugging Considerations section.

DEVICE CONFIGURATIONS (CONTINUED)

Table 21. Device Configuration Pins (LEND, BOOTMODE[1:0], ECLKIN_SEL[1:0], EEAI, and HD5/AD5)

CONFIGURATION PIN	NO.	FUNCTIONAL DESCRIPTION
LEND	E16	Device Endian mode (LEND) 0 – System operates in Big Endian mode 1 – System operates in Little Endian mode (default)
BOOTMODE[1:0]	[D18, C18]	Bootmode [1:0]. Default is reserved. External pullup and/or pulldown resistors must be used to select a valid bootmode configuration. 00 – No boot 01 – HPI boot 10 – Reserved 11 – EMIF 8-bit ROM boot with default timings (default mode)
ECLKIN_SEL[1:0]	[B18, A18]	EMIF input clock select Clock mode select for EMIF (ECLKIN_SEL[1:0]) 00 – ECLKIN (default mode) 01 – CPU/4 Clock Rate 10 – CPU/6 Clock Rate 11 – Reserved
EEAI	B17	PCI EEPROM Auto-Initialization (EEAI) PCI auto-initialization via external EEPROM 0 – PCI auto-initialization through EEPROM is disabled; the PCI peripheral uses the specified PCI default values (default). 1 – PCI auto-initialization through EEPROM is enabled; the PCI peripheral is configured through EEPROM provided the PCI peripheral pin is enabled (PCI_EN = 1). Note: ***If*** the PCI peripheral is disabled (PCI_EN pin = 0), this pin must ***not*** be pulled up. For more information on the PCI EEPROM default values, see the *TMS320C6000 DSP Peripheral Component Interconnect (PCI) Reference Guide* (literature number SPRU581).
HD5/AD5	Y1	HPI configuration bus width (HPI_WIDTH) 0 – HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the Hi-Z state.) 1 – HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.)

DEVICE CONFIGURATIONS (CONTINUED)

multiplexed pins

Multiplexed pins are pins that are shared by more than one peripheral and are internally multiplexed. Some of these pins are configured by software, and the others are configured by external pullup/pulldown resistors only at reset. Those muxed pins that are configured by software can be programmed to switch functionalities at any time. Those muxed pins that are configured by external pullup/pulldown resistors are mutually exclusive; only one peripheral has primary control of the function of these pins after reset. Table 22 identifies the multiplexed pins on the C6411 device; shows the default (primary) function and the default settings after reset; and describes the pins, registers, etc. necessary to configure specific multiplexed functions.

debugging considerations

It is recommended that external connections be provided to device configuration pins, including CLKMODE0, LEND, BOOTMODE[1:0], ECLKIN_SEL[1:0]. EEAI, HD5/AD5, and PCI_EN. Although internal pullup/pulldown resistors exist on these pins, providing external connectivity adds convenience to the user in debugging and flexibility in switching operating modes.

Internal pullup/pulldown resistors also exist on specified reserved (RSV) pins. Do not oppose the internal pullup/pulldown resistors, unless otherwised noted, on these RSV pins.

For the internal pullup/pulldown resistors for all device pins, see the terminal functions table.

DEVICE CONFIGURATIONS (CONTINUED)

Table 22. C6411 Device Multiplexed Pins†

MULTIPLEXED PINS NAME	MULTIPLEXED PINS NO.	DEFAULT FUNCTION	DEFAULT SETTING	DESCRIPTION
CLKOUT4/GP1	AE6	CLKOUT4	GP1EN = 0 (disabled)	These pins are software-configurable. To use these pins as GPIO pins, the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input GPxDIR = 1: GPx pin is an output
CLKOUT6/GP2	AD6	CLKOUT6	GP2EN = 0 (disabled)	
GP9/PIDSEL	M3	None	GPxEN = 0 (disabled) PCI_EN = 0 (disabled)†	To use GP[15:9] as GPIO pins, the PCI needs to be disabled (PCI_EN = 0), the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input GPxDIR = 1: GPx pin is an output
GP10/$\overline{\text{PCBE3}}$	L2			
GP11/$\overline{\text{PREQ}}$	F1			
GP12/$\overline{\text{PGNT}}$	J3			
GP13/$\overline{\text{PINTA}}$	G4			
GP14/PCLK	F2			
GP15/$\overline{\text{PRST}}$	G3			
HD[31:0]/AD[31:0]	‡	HD[31:0]	PCI_EN = 0 (disabled)†	By default, HPI is enabled upon reset (PCI is disabled). To enable the PCI peripheral an external pullup resistor (1 kΩ) must be provided on the PCI_EN pin (setting PCI_EN = 1 at reset and keeping valid “1” after reset).
$\overline{\text{HAS}}$/PPAR	T3	$\overline{\text{HAS}}$		
HCNTL1/$\overline{\text{PDEVSEL}}$	R1	HCNTL1		
HCNTL0/$\overline{\text{PSTOP}}$	T4	HCNTL0		
$\overline{\text{HDS1}}$/$\overline{\text{PSERR}}$	T1	$\overline{\text{HDS1}}$		
$\overline{\text{HDS2}}$/$\overline{\text{PCBE1}}$	T2	$\overline{\text{HDS2}}$		
HR/$\overline{\text{W}}$/$\overline{\text{PCBE2}}$	P1	HR/$\overline{\text{W}}$		
HHWIL/$\overline{\text{PTRDY}}$	R3	HHWIL (HPI16 only)		
$\overline{\text{HINT}}$/$\overline{\text{PFRAME}}$	R4	$\overline{\text{HINT}}$		
$\overline{\text{HCS}}$/$\overline{\text{PPERR}}$	R2	$\overline{\text{HCS}}$		
$\overline{\text{HRDY}}$/$\overline{\text{PIRDY}}$	P4	$\overline{\text{HRDY}}$		
CLOCK/PLL CONFIGURATION				
CLKIN	H4	I	IPD	Clock Input. This clock is the input to the on-chip PLL.
CLKOUT4/GP1§	AE6	I/O/Z	IPD	Clock output at 1/4 of the device speed (**O/Z**) [default] or this pin can be programmed as a GPIO 1 pin (**I/O/Z**).
CLKOUT6/GP2§	AD6	I/O/Z	IPD	Clock output at 1/6 of the device speed (**O/Z**) [default] or this pin can be programmed as a GPIO 2 pin (**I/O/Z**).

† All other standalone PCI pins are tied-off internally (pins in Hi-Z) when the peripheral is disabled [PCI_EN = 0].
‡ For the HD[31:0]/AD[31:0] multiplexed pins pin numbers, see the *Terminal Functions* table.

Terminal Functions

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
CLOCK/PLL CONFIGURATION (CONTINUED)				
CLKMODE0	H2	I	IPD	Clock mode select • Selects whether the CPU clock frequency = input clock frequency x1 (Bypass) [default] or x6. For more details on the CLKMODE0 pin and the PLL multiply factors, see the Clock PLL section of this data sheet.
PLLV¶	J6	A#		PLL voltage supply
JTAG EMULATION				
TMS	AB16	I	IPU	JTAG test-port mode select
TDO	AE19	O/Z	IPU	JTAG test-port data out
TDI	AF18	I	IPU	JTAG test-port data in
TCK	AF16	I	IPU	JTAG test-port clock
$\overline{\text{TRST}}$	AB15	I	IPD	JTAG test-port reset. For IEEE 1149.1 JTAG compatibility, see the IEEE 1149.1 JTAG Compatibility Statement section of this data sheet.
EMU11	AC18	I/O/Z	IPU	Emulation pin 11. Reserved for future use, leave unconnected.
EMU10	AD18	I/O/Z	IPU	Emulation pin 10. Reserved for future use, leave unconnected.
EMU9	AE18	I/O/Z	IPU	Emulation pin 9. Reserved for future use, leave unconnected.
EMU8	AC17	I/O/Z	IPU	Emulation pin 8. Reserved for future use, leave unconnected.
EMU7	AF17	I/O/Z	IPU	Emulation pin 7. Reserved for future use, leave unconnected.
EMU6	AD17	I/O/Z	IPU	Emulation pin 6. Reserved for future use, leave unconnected.
EMU5	AE17	I/O/Z	IPU	Emulation pin 5. Reserved for future use, leave unconnected.
EMU4	AC16	I/O/Z	IPU	Emulation pin 4. Reserved for future use, leave unconnected.
EMU3	AD16	I/O/Z	IPU	Emulation pin 3. Reserved for future use, leave unconnected.
EMU2	AE16	I/O/Z	IPU	Emulation pin 2. Reserved for future use, leave unconnected.
EMU1 EMU0	AC15 AF15	I/O/Z	IPU	Emulation [1:0] pins • Select the device functional mode of operation EMU[1:0] Operation 00 Boundary Scan/Normal Mode (see Note) 01 Reserved 10 Reserved 11 Emulation/Normal Mode [default] (see the IEEE 1149.1 JTAG Compatibility Statement section of this data sheet) Normal mode refers to the DSPs normal operational mode, when the DSP is free running. The DSP can be placed in normal operational mode when the EMU[1:0] pins are configured for either Boundary Scan or Emulation. Note: When the EMU[1:0] pins are configured for Boundary Scan mode, the internal pulldown (IPD) on the $\overline{\text{TRST}}$ signal must ***not*** be opposed in order to operate in Normal mode. For the Boundary Scan mode pulldown EMU[1:0] pins with a dedicated 1-kΩ resister.

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground
‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)
§ These pins are multiplexed pins. For more details, see the Device Configurations section of this data sheet.
¶ PLLV is not part of external voltage supply. See the Clock PLL section for information on how to connect this pin.
# A = Analog signal (PLL Filter)

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
DEVICE CONFIGURATION				
LEND	E16	I/O/Z	IPU	Device Endian mode LEND: 0 – Big Endian 1 – Little Endian (default mode)
HD5/AD5§	Y1	I/O/Z		Host-Port bus width (HPI_WIDTH) user-configurable at device reset via a 10-kΩ resistor pullup/pulldown resistor on the HD5 pin: HD5 pin = 0: HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the high-impedance state.) HD5 pin = 1: HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.)
PCI_EN	AA4	I	IPD	PCI enable pin. This pin controls the selection (enable/disable) of the HPI and GP[15:9], or PCI peripherals. For more details, see the Device Configurations section of this data sheet.
EEAI	B17	I/O/Z	IPD	PCI EEPROM Auto-Initialization (EEAI) via external EEPROM ***If*** the PCI peripheral is disabled (PCI_EN pin = 0), this pin must ***not*** be pulled up. EEAI: 0 – PCI auto-initialization through EEPROM is disabled (default). 1 – PCI auto-initialization through EEPROM is enabled.
BOOTMODE1	D18	I/O/Z	IPU	Boot mode. Default is reserved. External pullup and/or pulldown resistors must be used to select a valid bootmode configuration. BOOTMODE[1:0]: 00 – No boot 01 – HPI boot 10 – Reserved 11 – EMIF 8-bit ROM boot with default timings (default mode)
BOOTMODE0	C18		IPD	
ECLKIN_SEL1	B18	I/O/Z	IPD	EMIF clock mode select ECLKIN_SEL[1:0]: 00 – ECLKIN (default mode) 01 – CPU/4 Clock Rate 10 – CPU/6 Clock Rate 11 – Reserved
ECLKIN_SEL0	A18			
RESETS, INTERRUPTS, AND GENERAL-PURPOSE INPUT/OUTPUTS				
$\overline{\text{RESET}}$	AC7	I		Device reset
NMI	B4	I	IPD	Nonmaskable interrupt, edge-driven (rising edge)
GP7/EXT_INT7	AF4	I/O/Z	IPU	General-purpose input/output (GPIO) pins (**I/O/Z**) or external interrupts (**input only**). The default after reset setting is GPIO enabled as input-only. • When these pins function as External Interrupts [by selecting the corresponding interrupt enable register bit (IER.[7:4])], they are edge-driven and the polarity can be independently selected via the External Interrupt Polarity Register bits (EXTPOL.[3:0]).
GP6/EXT_INT6	AD5			
GP5/EXT_INT5	AE5			
GP4/EXT_INT4	AF5			
GP15/$\overline{\text{PRST}}$§	G3	I/O/Z		General-purpose input/output (GPIO) 15 pin (**I/O/Z**) or PCI reset (**I**). No function at default.
GP14/PCLK§	F2			GPIO 14 pin (**I/O/Z**) or PCI clock (**I**). No function at default.
GP13/$\overline{\text{PINTA}}$§	G4			GPIO 13 pin (**I/O/Z**) or PCI interrupt A (**O/Z**). No function at default.
GP12/$\overline{\text{PGNT}}$§	J3			GPIO 12 pin (**I/O/Z**) or PCI bus grant (**I**). No function at default.
GP11/$\overline{\text{PREQ}}$§	F1			GPIO 11 pin (**I/O/Z**) or PCI bus request (**O/Z**). No function at default.
GP10/$\overline{\text{PCBE3}}$§	L2			GPIO 10 pin (**I/O/Z**) or PCI command/byte enable 3 (**I/O/Z**). No function at default.
GP9/PIDSEL§	M3			GPIO 9 pin (**I/O/Z**) or PCI initialization device select (**I**). No function at default.
GP3	AC6		IPD	GPIO 3 pin (**I/O/Z**).

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ These pins are multiplexed pins. For more details, see the Device Configurations section of this data sheet.

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
RESETS, INTERRUPTS, AND GENERAL-PURPOSE INPUT/OUTPUTS (CONTINUED)				
GP0	AF6	I/O/Z	IPD	GPIO 0 pin. The general-purpose I/O 0 pin (GPIO 0) (**I/O/Z**) can be programmed as GPIO 0 (**input only**) [default] or as GPIO 0 (**output only**) pin or output as a general-purpose interrupt (GP0INT) signal (**output only**).
GP8	AE4	I/O/Z	IPD	This pin has no function at default [default] or this pin can be programmed as a GPIO 8 pin (**I/O/Z**).
CLKOUT6/GP2§	AD6	I/O/Z	IPD	Clock output at 1/6 of the device speed (**O/Z**) [default] or this pin can be programmed as a GPIO 2 pin (**I/O/Z**).
CLKOUT4/GP1§	AE6	I/O/Z	IPD	Clock output at 1/4 of the device speed (**O/Z**) [default] or this pin can be programmed as a GPIO 1 pin (**I/O/Z**).
HOST-PORT INTERFACE (HPI) or PERIPHERAL COMPONENT INTERCONNECT (PCI)				
$\overline{\text{HINT}}$/ $\overline{\text{PFRAME}}$§	R4	I/O/Z		Host interrupt from DSP to host (**O**) [default] or PCI frame (**I/O/Z**)
HCNTL1/ $\overline{\text{PDEVSEL}}$§	R1	I/O/Z		Host control − selects between control, address, or data registers (**I**) [default] or PCI device select (**I/O/Z**).
HCNTL0/ $\overline{\text{PSTOP}}$§	T4	I/O/Z		Host control − selects between control, address, or data registers (**I**) [default] or PCI stop (**I/O/Z**)
HHWIL/$\overline{\text{PTRDY}}$§	R3	I/O/Z		Host half-word select − first or second half-word (not necessarily high or low order) [For HPI16 bus width selection only] (**I**) [default] or PCI target ready (**I/O/Z**)
HR/$\overline{\text{W}}$/$\overline{\text{PCBE2}}$§	P1	I/O/Z		Host read or write select (**I**) [default] or PCI command/byte enable 2 (**I/O/Z**)
$\overline{\text{HAS}}$/PPAR§	T3	I/O/Z		Host address strobe (**I**) [default] or PCI parity (**I/O/Z**)
$\overline{\text{HCS}}$/$\overline{\text{PPERR}}$§	R2	I/O/Z		Host chip select (**I**) [default] or PCI parity error (**I/O/Z**)
$\overline{\text{HDS1}}$/$\overline{\text{PSERR}}$§	T1	I/O/Z		Host data strobe 1 (**I**) [default] or PCI system error (**I/O/Z**)
$\overline{\text{HDS2}}$/$\overline{\text{PCBE1}}$§	T2	I/O/Z		Host data strobe 2 (**I**) [default] or PCI command/byte enable 1 (**I/O/Z**)
$\overline{\text{HRDY}}$/$\overline{\text{PIRDY}}$§	P4	I/O/Z		Host ready from DSP to host (**O**) [default] or PCI initiator ready (**I/O/Z**).
HD31/AD31§	J2	I/O/Z		Host-port data (**I/O/Z**) [default] or PCI data-address bus (**I/O/Z**) As HPI data bus (PCI_EN pin = 0) • Used for transfer of data, address, and control • Host-Port bus width (HPI_WIDTH) user-configurable at device reset via a 10-kΩ resistor pullup/pulldown resistor on the HD5 pin: HD5 pin = 0: HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the high-impedance state.) HD5 pin = 1: HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.) As PCI data-address bus (PCI_EN pin = 1) • Used for transfer of data and address
HD30/AD30§	K3			
HD29/AD29§	J1			
HD28/AD28§	K4			
HD27/AD27§	K2			
HD26/AD26§	L3			
HD25/AD25§	K1			
HD24/AD24§	L4			
HD23/AD23§	L1			
HD22/AD22§	M4			
HD21/AD21§	M2			
HD20/AD20§	N4			
HD19/AD19§	M1			
HD18/AD18§	N5			
HD17/AD17§	N1			
HD16/AD16§	P5			
HD15/AD15§	U4			

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ These pins are multiplexed pins. For more details, see the Device Configurations section of this data sheet.

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
HOST-PORT INTERFACE (HPI) or PERIPHERAL COMPONENT INTERCONNECT (PCI) (CONTINUED)				
HD14/AD14§	U1	I/O/Z		Host-port data (**I/O/Z**) [default] or PCI data-address bus (**I/O/Z**) As HPI data bus (PCI_EN pin = 0) • Used for transfer of data, address, and control • Host-Port bus width (HPI_WIDTH) user-configurable at device reset via a 10-kΩ resistor pullup/pulldown resistor on the HD5 pin: HD5 pin = 0: HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the high-impedance state.) HD5 pin = 1: HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.) As PCI data-address bus (PCI_EN pin = 1) • Used for transfer of data and address
HD13/AD13§	U3			
HD12/AD12§	U2			
HD11/AD11§	V4			
HD10/AD10§	V1			
HD9/AD9§	V3			
HD8/AD8§	V2			
HD7/AD7§	W2			
HD6/AD6§	W4			
HD5/AD5§	Y1			
HD4/AD4§	Y3			
HD3/AD3§	Y2			
HD2/AD2§	Y4			
HD1/AD1§	AA1			
HD0/AD0§	AA3			
$\overline{\text{PCBE0}}$§	W3	I/O/Z		PCI command/byte enable 0 (**I/O/Z**). When PCI is disabled (PCI_EN = 0), this pin is tied-off.
XSP_CS	AD1	O	IPD	PCI serial interface chip select (**O**). When PCI is disabled (PCI_EN = 0), this pin is tied-off.
XSP_CLK	AC2	I/O/Z	IPD	This pin has no function at default [default] or when PCI is enabled (PCI_EN = 1), this pin is the PCI serial interface clock (**O**).
XSP_DI	AB3	I	IPU	This pin has no function at default [default] or when PCI is enabled (PCI_EN = 1), this pin is the PCI serial interface data in (**I**). In PCI mode, this pin is connected to the output data pin of the serial PROM.
XSP_DO	AA2	O/Z	IPU	This pin has no function at default [default] or when PCI is enabled (PCI_EN = 1), this pin is the PCI serial interface data out (**O**). In PCI mode, this pin is connected to the input data pin of the serial PROM.
GP15/$\overline{\text{PRST}}$§	G3	I/O/Z		General-purpose input/output (GPIO) 15 pin (**I/O/Z**) or PCI reset (**I**). No function at default.
GP14/PCLK§	F2			GPIO 14 pin (**I/O/Z**) or PCI clock (**I**). No function at default.
GP13/$\overline{\text{PINTA}}$§	G4			GPIO 13 pin (**I/O/Z**) or PCI interrupt A (**O/Z**). No function at default.
GP12/$\overline{\text{PGNT}}$§	J3			GPIO 12 pin (**I/O/Z**) or PCI bus grant (**I**). No function at default.
GP11/$\overline{\text{PREQ}}$§	F1			GPIO 11 pin (**I/O/Z**) or PCI bus request (**O/Z**). No function at default.
GP10/$\overline{\text{PCBE3}}$§	L2			GPIO 10 pin (**I/O/Z**) or PCI command/byte enable 3 (**I/O/Z**). No function at default.
GP9/PIDSEL§	M3			GPIO 9 pin (**I/O/Z**) or PCI initialization device select (**I**). No function at default.

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ These pins are multiplexed pins. For more details, see the Device Configurations section of this data sheet.

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
EMIF (32-bit) − CONTROL SIGNALS COMMON TO ALL TYPES OF MEMORY§				
$\overline{CE3}$	L26	O/Z	IPU	EMIF memory space enables • Enabled by bits 28 through 31 of the word address • Only one pin is asserted during any external data access
$\overline{CE2}$	K23	O/Z	IPU	
$\overline{CE1}$	K24	O/Z	IPU	
$\overline{CE0}$	K25	O/Z	IPU	
$\overline{BE3}$	M25	O/Z	IPU	EMIF byte-enable control • Decoded from the low-order address bits. The number of address bits or byte enables used depends on the width of external memory. • Byte-write enables for most types of memory • Can be directly connected to SDRAM read and write mask signal (SDQM)
$\overline{BE2}$	M26	O/Z	IPU	
$\overline{BE1}$	L23	O/Z	IPU	
$\overline{BE0}$	L24	O/Z	IPU	
$\overline{PDT}$	M22	O/Z	IPU	EMIF peripheral data transfer, allows direct transfer between external peripherals
EMIF (32-BIT) − BUS ARBITRATION§				
$\overline{HOLDA}$	N22	O	IPU	EMIF hold-request-acknowledge to the host
$\overline{HOLD}$	V23	I	IPU	EMIF hold request from the host
BUSREQ	P22	O	IPU	EMIF bus request output
EMIF (32-BIT) − ASYNCHRONOUS/SYNCHRONOUS MEMORY CONTROL§				
ECLKIN	H25	I	IPD	EMIF external input clock. The EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) is selected at reset via the pullup/pulldown resistors on the ECLKIN_SEL[1:0] pins. AECLKIN is the default for the EMIF input clock.
ECLKOUT2	J23	O/Z	IPD	EMIF output clock 2. Programmable to be EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) frequency divided-by-1, -2, or -4.
ECLKOUT1	J26	O/Z	IPD	EMIF output clock 1 at EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) frequency.
$\overline{ARE}$/ $\overline{SDCAS}$/ $\overline{SADS}$/$\overline{SRE}$	J25	O/Z	IPU	EMIF asynchronous memory read-enable/SDRAM column-address strobe/programmable synchronous interface-address strobe or read-enable • For programmable synchronous interface, the RENEN field in the CE Space Secondary Control Register (CExSEC) selects between $\overline{SADS}$ and $\overline{SRE}$: If RENEN = 0, then the $\overline{SADS}$/$\overline{SRE}$ signal functions as the $\overline{SADS}$ signal. If RENEN = 1, then the $\overline{SADS}$/$\overline{SRE}$ signal functions as the $\overline{SRE}$ signal.
$\overline{AOE}$/ $\overline{SDRAS}$/ $\overline{SOE}$	J24	O/Z	IPU	EMIF asynchronous memory output-enable/SDRAM row-address strobe/programmable synchronous interface output-enable
$\overline{AWE}$/ $\overline{SDWE}$/ $\overline{SWE}$	K26	O/Z	IPU	EMIF asynchronous memory write-enable/SDRAM write-enable/programmable synchronous interface write-enable
SDCKE	L25	O/Z	IPU	EMIF SDRAM clock-enable (used for self-refresh mode). • If SDRAM is not in system, SDCKE can be used as a general-purpose output.
$\overline{SOE3}$	R22	O/Z	IPU	EMIF synchronous memory output-enable for $\overline{CE3}$ (for glueless FIFO interface)
ARDY	L22	I	IPU	Asynchronous memory ready input

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ To maintain signal integrity for the EMIF signals, serial termination resistors should be inserted into all EMIF output signal lines.

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
EMIF (32-BIT) − ADDRESS§				
EA22	T22	O/Z	IPD	EMIF external address (word address) Note: EMIF address numbering for the C6411 device starts with EA3 to maintain signal name compatibility with other C64x™ devices (e.g., C6414, C6415, and C6416) [see the 64-bit EMIF addressing scheme in the *TMS320C6000 DSP External Memory Interface (EMIF) Reference Guide* (literature number SPRU266)].
EA21	V24			
EA20	V25			
EA19	V26			
EA18	U23			
EA17	U24			
EA16	U25			
EA15	U26			
EA14	T25			
EA13	T26			
EA12	R23			
EA11	R24			
EA10	P23			
EA9	P24			
EA8	P26			
EA7	N23			
EA6	N24			
EA5	N26			
EA4	M23			
EA3	M24			
EMIF (32-bit) − DATA§				
ED31	C26	I/O/Z	IPU	EMIF external data
ED30	D26			
ED29	D25			
ED28	E25			
ED27	E24			
ED26	E26			
ED25	F24			
ED24	F25			
ED23	F23			
ED22	F26			
ED21	G24			
ED20	G25			
ED19	G23			
ED18	G26			
ED17	H23			
ED16	H24			
ED15	C19			
ED14	D19			

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ To maintain signal integrity for the EMIF signals, serial termination resistors should be inserted into all EMIF output signal lines.

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
EMIF (32-bit) − DATA§ (CONTINUED)				
ED13	A20	I/O/Z	IPU	EMIF external data
ED12	D20			
ED11	B20			
ED10	C20			
ED9	A21			
ED8	D21			
ED7	B21			
ED6	C21			
ED5	A22			
ED4	C22			
ED3	B22			
ED2	B23			
ED1	A23			
ED0	A24			
TIMER 2				
TOUT2¶	A4	O/Z	IPD	Timer 2 or general-purpose output
TINP2	C5	I	IPD	Timer 2 or general-purpose input
TIMER 1				
TOUT1¶	B5	O/Z	IPD	Timer 1 or general-purpose output
TINP1	A5	I	IPD	Timer 1 or general-purpose input
TIMER 0				
TOUT0¶	D6	O/Z	IPD	Timer 0 or general-purpose output
TINP0	C6	I	IPD	Timer 0 or general-purpose input
MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP1)				
CLKS1	AC8	I		McBSP1 external clock source (as opposed to internal)
CLKR1	AC10	I/O/Z		McBSP1 receive clock
CLKX1	AB12	I/O/Z		McBSP1 transmit clock
DR1	AF11	I		McBSP1 receive data
DX1	AB11	O/Z		McBSP1 transmit data
FSR1	AC9	I/O/Z		McBSP1 receive frame sync
FSX1	AB13	I/O/Z		McBSP1 transmit frame sync
MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP0)				
CLKS0	F4	I	IPD	McBSP0 external clock source (as opposed to internal)
CLKR0	D1	I/O/Z	IPD	McBSP0 receive clock
CLKX0	E1	I/O/Z	IPD	McBSP0 transmit clock
DR0	D2	I	IPU	McBSP0 receive data
DX0	E2	O/Z	IPU	McBSP0 transmit data

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

§ To maintain signal integrity for the EMIF signals, serial termination resistors should be inserted into all EMIF output signal lines.

¶ Following RESET, TOUTx will be configured as a general-purpose output (GPO) due to the default value of the Timerx Control Register (CTLx); therefore, an external resistor may *not* be used to pull the signal to the opposite supply rail.

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
RESERVED FOR TEST				
FSR0	C1	I/O/Z	IPD	McBSP0 receive frame sync
FSX0	E3	I/O/Z	IPD	McBSP0 transmit frame sync
RSV	AD11			**For proper device operation, these RSV pins must be externally pulled down to ground with a 10-kΩ resistor.**
	AD12			
RSV	A16		IPD	**For proper device operation, these RSV pins must be externally pulled up to DV_{DD} with a 1-kΩ resistor.**
	D15			
RSV	G14			Reserved. These pins must be connected directly to CV_{DD} for proper device operation.
	H7			
	N20			
	P7			
	Y13			
RSV	R6			Reserved. This pin must be connected directly to DV_{DD} for proper device operation.
RSV	A3			Reserved (leave unconnected, ***do not*** connect to power or ground)
RSV	A6		IPU	Reserved (leave unconnected, ***do not*** connect to power or ground)
	A7		IPU	
	A9		IPU	
	A10		IPU	
	A11		IPD	
	A12		IPU	
	A13		IPU	
	A14		IPD	
	A15		IPD	
	A17		IPD	
	B6		IPU	
	B7		IPU	
	B8		IPU	
	B9		IPU	
	B10		IPU	
	B11		IPU	
	B12		IPU	
	B15		IPD	
	B16		IPD	
	B19		IPU	
	C7		IPU	
	C8		IPU	
	C9		IPU	
	C10		IPU	
	C11		IPU	
	C12		IPU	

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
RESERVED FOR TEST (CONTINUED)				
	C13		IPU	
	C14		IPD	
	C15		IPD	
	C16		IPD	
	C17		IPD	
	D7		IPU	
	D8		IPU	
	D9		IPU	
	D10		IPU	
	D11		IPD	
	D12		IPD	
	D13		IPU	
	D14		IPD	
	D16		IPD	
	D17		IPD	
	E11		IPU	
	E12		IPU	
	E13		IPU	
	E14		IPU	
RSV	E15		IPU	Reserved (leave unconnected, ***do not*** connect to power or ground)
	G1		IPD	
	G2			
	H3			
	J4			
	K6			
	N3			
	P3			
	R25		IPU	
	R26		IPU	
	T23		IPU	
	T24		IPU	
	W23		IPU	
	W24		IPU	
	W25		IPD	
	Y23		IPU	
	Y24		IPU	
	Y25		IPU	
	Y26		IPU	
	AA23		IPU	

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
RESERVED FOR TEST (CONTINUED)				
RSV	AA24		IPU	Reserved (leave unconnected, ***do not*** connect to power or ground)
	AA25		IPU	
	AA26		IPU	
	AB1		IPD	
	AB2		IPD	
	AB14			
	AB24		IPU	
	AB25		IPU	
	AB26		IPU	
	AC1		IPD	
	AC11			
	AC12			
	AC13			
	AC14			
	AC19		IPU	
	AC20		IPU	
	AC21		IPU	
	AC25		IPU	
	AC26		IPU	
	AD7			
	AD8			
	AD9			
	AD10			
	AD13			
	AD14			
	AD15			
	AD19		IPU	
	AD20		IPU	
	AD21		IPU	
	AD22		IPU	
	AD26		IPU	
	AE7			
	AE8			
	AE9			
	AE10			
	AE11			
	AE12			
	AE15			
	AE20		IPU	
	AE21		IPU	

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
RESERVED FOR TEST (CONTINUED)				
RSV	AE22		IPU	Reserved (leave unconnected, ***do not*** connect to power or ground)
	AE23		IPU	
	AF3		IPD	
	AF7			
	AF9			
	AF10			
	AF12			
	AF13			
	AF14			
	AF20		IPU	
	AF21		IPU	
	AF22		IPU	
	AF23		IPU	
	AF24		IPU	

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

‡ IPD = Internal pulldown, IPU = Internal pullup. (These IPD/IPU signal pins feature a 30-kΩ IPD or IPU resistor. To pull up a signal to the opposite supply rail, a 1-kΩ resistor should be used, unless otherwise noted.)

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS			
DV_{DD}	A2	S	3.3-V supply voltage (see the Power-Supply Decoupling section of this data sheet)
	A25		
	B1		
	B14		
	B26		
	E7		
	E8		
	E10		
	E17		
	E19		
	E20		
	F3		
	F9		
	F12		
	F15		
	F18		
	G5		
	G22		
	H5		
	H22		
	J21		
	K5		
	K22		
	L5		
	M5		
	M6		
	M21		
	N2		
	P25		
	R5		
	R21		
	T5		
	U5		
	U22		
	V6		
	V21		
	W5		
	W22		
	Y5		
	Y22		

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
DV_{DD}	AA9	S	3.3-V supply voltage (see the Power-Supply Decoupling section of this data sheet)
	AA12		
	AA15		
	AA18		
	AB7		
	AB8		
	AB10		
	AB17		
	AB19		
	AB20		
	AE1		
	AE13		
	AE26		
	AF2		
	AF25		
CV_{DD}	A1		1.2-V supply voltage (see the Power-Supply Decoupling section of this data sheet)
	A26		
	B2		
	B25		
	C3		
	C24		
	D4		
	D23		
	E5		
	E22		
	F6		
	F7		
	F20		
	F21		
	G6		
	G7		
	G8		
	G10		
	G11		
	G13		
	G16		
	G17		
	G19		
	G20		

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
CV_{DD}	G21, H20, K7, K20, L7, L20, N7, P20, T7, T20, U7, U20, W7, W20, Y6, Y7, Y8, Y10, Y11, Y14, Y16, Y17, Y19, Y20, Y21, AA6, AA7, AA20, AA21, AB5, AB22, AC4, AC23, AD3, AD24, AE2, AE25, AF1, AF26	S	1.2-V supply voltage (see the Power-Supply Decoupling section of this data sheet)

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
GROUND PINS			
V_{SS}	A8	GND	Ground pins
	A19		
	B3		
	B13		
	B24		
	C2		
	C4		
	C23		
	C25		
	D3		
	D5		
	D22		
	D24		
	E4		
	E6		
	E9		
	E18		
	E21		
	E23		
	F5		
	F8		
	F10		
	F11		
	F13		
	F14		
	F16		
	F17		
	F19		
	F22		
	G9		
	G12		
	G15		
	G18		
	H1		
	H6		
	H21		
	H26		
	J5		
	J7		

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
GROUND PINS (CONTINUED)			
V_{SS}	J20	GND	Ground pins
	J22		
	K21		
	L6		
	L21		
	M7		
	M20		
	N6		
	N21		
	N25		
	P2		
	P6		
	P21		
	R7		
	R20		
	T6		
	T21		
	U6		
	U21		
	V5		
	V7		
	V20		
	V22		
	W1		
	W6		
	W21		
	W26		
	Y9		
	Y12		
	Y15		
	Y18		
	AA5		
	AA8		
	AA10		
	AA11		
	AA13		
	AA14		
	AA16		
	AA17		

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
GROUND PINS (CONTINUED)			
V_{SS}	AA19	GND	Ground pins
	AA22		
	AB4		
	AB6		
	AB9		
	AB18		
	AB21		
	AB23		
	AC3		
	AC5		
	AC22		
	AC24		
	AD2		
	AD4		
	AD23		
	AD25		
	AE3		
	AE14		
	AE24		
	AF8		
	AF19		

† I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground

development support

TI offers an extensive line of development tools for the TMS320C6000™ DSP platform, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of C6000™ DSP-based applications:

Software Development Tools:
Code Composer Studio™ Integrated Development Environment (IDE): including Editor
C/C++/Assembly Code Generation, and Debug plus additional development tools
Scalable, Real-Time Foundation Software (DSP BIOS), which provides the basic run-time target software needed to support any DSP application.

Hardware Development Tools:
Extended Development System (XDS™) Emulator (supports C6000™ DSP multiprocessor system debug)
EVM (Evaluation Module)

The *TMS320 DSP Development Support Reference Guide* (SPRU011) contains information about development-support products for all TMS320™ DSP family member devices, including documentation. See this document for further information on TMS320™ DSP documentation or any TMS320™ DSP support products from Texas Instruments. An additional document, the *TMS320 Third-Party Support Reference Guide* (SPRU052), contains information about TMS320™ DSP-related products from other companies in the industry. To receive TMS320™ DSP literature, contact the Literature Response Center at 800/477-8924.

For a complete listing of development-support tools for the TMS320C6000™ DSP platform, visit the Texas Instruments web site on the Worldwide Web at http://www.ti.com uniform resource locator (URL). For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

Code Composer Studio, XDS, and TMS320 are trademarks of Texas Instruments.

device support

device and development-support tool nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all DSP devices and support tools. Each DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., **TMS**320C6411GLZ). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

TMX Experimental device that is not necessarily representative of the final device's electrical specifications

TMP Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification

TMS Fully qualified production device

Support tool development evolutionary flow:

TMDX Development-support product that has not yet completed Texas Instruments internal qualification testing.

TMDS Fully qualified development-support product

TMX and TMP devices and TMDX development-support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

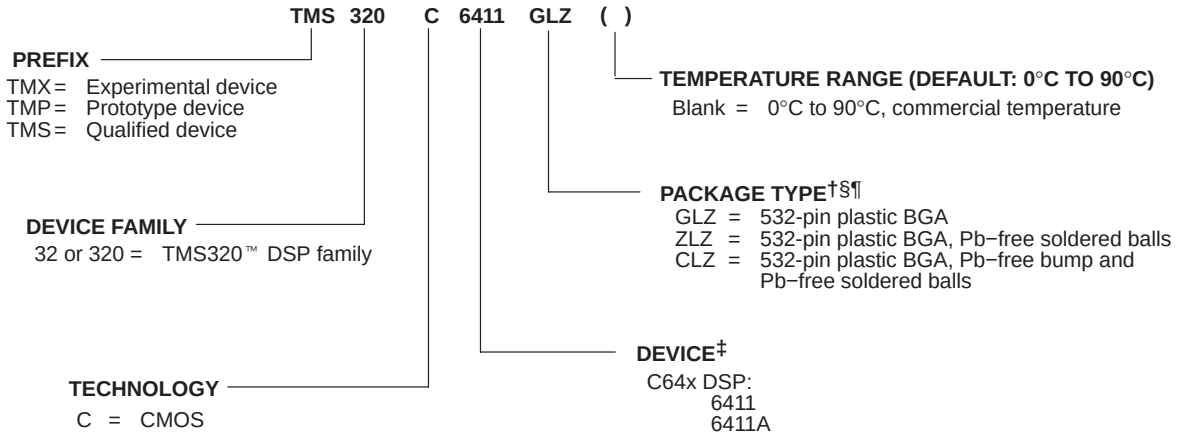
TMS devices and TMDS development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices (TMX or TMP) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, GLZ) and the temperature range (for example, blank is the default commercial temperature range). Figure 5 provides a legend for reading the complete device name for any TMS320C6000™ DSP platform member.

The ZLZ package, like the GLZ package, is a 532-pin plastic BGA *only* with lead-free soldered balls. The CLZ package, like the GLZ package, is a 532-pin plastic BGA *only* with lead-free bump and lead−free soldered balls. The ZLZ and CLZ package types are available upon request. For device part numbers and further ordering information for TMS320C6414/C6415/C6416 in the GLZ, ZLZ and CLZ package types, see the TI website (http://www.ti.com) or contact your TI sales representative.

device and development-support tool nomenclature (continued)



† BGA = Ball Grid Array
‡ For the actual device part number (P/N) and ordering information, see the TI website (www.ti.com).
§ The ZLZ mechanical package designator represents the version of the GLZ package with Pb-free soldered balls. For more detailed information, see the *Mechanical Data* section of this document.
¶ The CLZ mechanical package designator represents the version of the GLZ package with Pb-free bump and soldered balls. For more detailed information, see the *Mechanical Data* section of this document.

Figure 5. TMS320C6411 DSP Device Nomenclature

For additional information, see the *TMS320C6411 Digital Signal Processor Silicon Errata* (literature number SPRZ194)

documentation support

Extensive documentation supports all TMS320™ DSP family generations of devices from product announcement through applications development. The types of documentation available include: data sheets, such as this document, with design specifications; complete user's reference guides for all devices and tools; technical briefs; development-support tools; on-line help; and hardware and software applications. The following is a brief, descriptive list of support documentation specific to the C6000™ DSP devices:

The *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189) describes the C6000™ DSP CPU (core) architecture, instruction set, pipeline, and associated interrupts.

The *TMS320C6000 DSP Peripherals Overview Reference Guide* (literature number SPRU190) provides an overview and briefly describes the functionality of the peripherals available on the C6000™ DSP platform of devices. This document also includes a table listing the peripherals available on the C6000 devices along with literature numbers and hyperlinks to the associated peripheral documents.

The *TMS320C64x Technical Overview* (literature number SPRU395) gives an introduction to the C64x™ digital signal processor, and discusses the application areas that are enhanced by the C64x™ DSP VelociTI.2™ VLIW architecture.

The *TMS320C6414, TMS320C6415, and TMS320C6416 Fixed-Point Digital Signal Processors* data sheet (literature number SPRS146) describes the features of the TMS320C6414, TMS320C6415, and TMS320C6416 DSP devices and provides pinouts, electrical specifications, and timings.

The *TMS320C6414, TMS320C6415, and TMS320C6416 Digital Signal Processors Silicon Errata* (literature number SPRZ011) describes the known exceptions to the functional specifications for the TMS320C6414, TMS320C6415, and TMS320C6416 DSP devices.

The *TMS320C6411 Digital Signal Processor Silicon Errata* (literature number SPRZ194) describes the known exceptions to the functional specifications for particular silicon revisions of the TMS320C6411 device.

The *Using IBIS Models for Timing Analysis* application report (literature number SPRA839) describes how to properly use IBIS models to attain accurate timing analysis for a given system.

For more detailed information on the device compatibility and similarities/differences among the C6211, C6411, C6414, C6415, and C6416 devices, see the *How To Begin Development Today With the TMS320C6414, TMS320C6415, and TMS320C6416 DSPs* application report (literature number SPRA718) and *How To Begin Development Today With the TMS320C6411 DSP* application report (literature number SPRA374).

The tools support documentation is electronically available within the Code Composer Studio™ Integrated Development Environment (IDE). For a complete listing of C6000™ DSP latest documentation, visit the Texas Instruments web site on the Worldwide Web at http://www.ti.com uniform resource locator (URL).

clock PLL

Most of the internal C64x™ DSP clocks are generated from a single source through the CLKIN pin. This source clock either drives the PLL, which multiplies the source clock frequency to generate the internal CPU clock, or bypasses the PLL to become the internal CPU clock.

To use the PLL to generate the CPU clock, the external PLL filter circuit must be properly designed. Figure 6 shows the external PLL circuitry for either x1 (PLL bypass) or other PLL multiply modes.

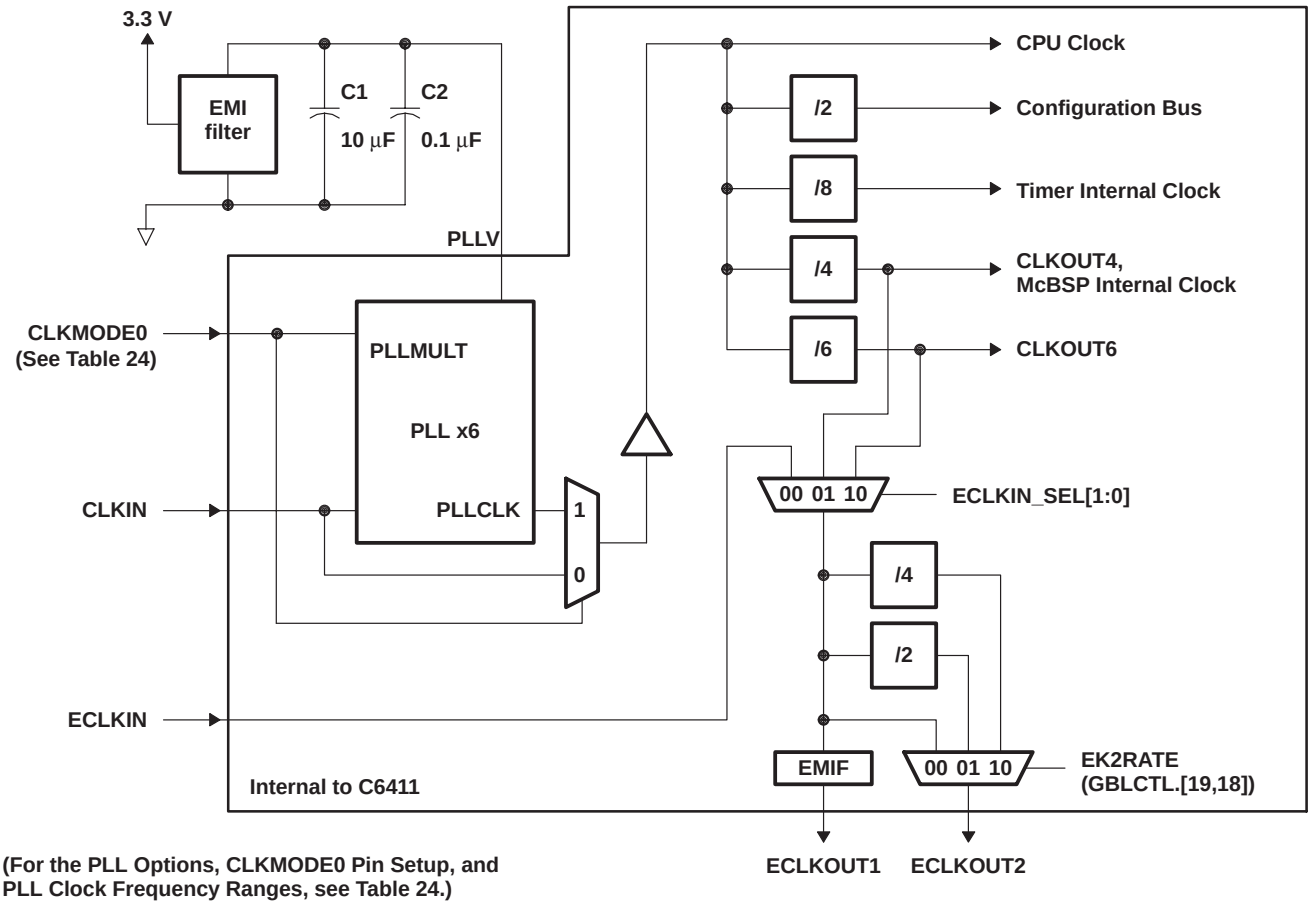
To minimize the clock jitter, a single clean power supply should power both the C64x™ DSP device and the external clock oscillator circuit. The minimum CLKIN rise and fall times should also be observed. For the input clock timing requirements, see the *input and output clocks* electricals section.

Rise/fall times, duty cycles (high/low pulse durations), and the load capacitance of the external clock source must meet the DSP requirements in this data sheet (see the *electrical characteristics over recommended ranges of suppy voltage and operating case temperature* table and the *input and output clocks* electricals section). Table 23 lists some examples of compatible CLKIN external clock sources:

Table 23. Compatible CLKIN External Clock Sources

COMPATIBLE PARTS FOR EXTERNAL CLOCK SOURCES (CLKIN)	PART NUMBER	MANUFACTURER
Oscillators	JITO-2	Fox Electronix
	STA series, ST4100 series	SaRonix Corporation
	SG-636	Epson America
	342	Corning Frequency Control
PLL	ICS525-02	Integrated Circuit Systems
Spread Spectrum Clock Generator	MK1714	

clock PLL (continued)



(For the PLL Options, CLKMODE0 Pin Setup, and PLL Clock Frequency Ranges, see Table 24.)

NOTES: A. Place all PLL external components (C1, C2, and the EMI Filter) as close to the C6000™ DSP device as possible. For the best performance, TI recommends that all the PLL external components be on a single side of the board without jumpers, switches, or components other than the ones shown.
B. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (C1, C2, and the EMI Filter).
C. The 3.3-V supply for the EMI filter must be from the same 3.3-V power plane supplying the I/O voltage, DV_{DD}.
D. EMI filter manufacturer TDK part number ACF451832-333, -223, -153, -103. Panasonic part number EXCCET103U.

Figure 6. External PLL Circuitry for Either PLL Multiply Modes or x1 (Bypass) Mode

Table 24. TMS320C6411 PLL Multiply Factor Options, Clock Frequency Ranges, and Typical Lock Time†‡

GLZ, ZLZ and CLZ PACKAGES − 23 x 23 mm BGA						
CLKMODE0	CLKMODE (PLL MULTIPLY FACTORS)	CLKIN RANGE (MHz)	CPU CLOCK FREQUENCY RANGE (MHz)	CLKOUT4 RANGE (MHz)	CLKOUT6 RANGE (MHz)	TYPICAL LOCK TIME (μs)§
0	Bypass (x1) [default]	30−75.75	30−75.75	7.5−18.93	5−12.62	N/A
1	x6	30−50.5	180−303	45−75.75	30−50.5	75

† These clock frequency range values are applicable to a C6411−300 speed device.
‡ Use an external pullup resistor on the CLKMODE0 pin to set the C6411 device to the valid PLL multiply clock mode (x6). With an internal pulldown resistor on the CLKMODE0 pin, the default clock mode is x1 (bypass).
§ Under some operating conditions, the maximum PLL lock time may vary by as much as 150% from the specified typical value. For example, if the typical lock time is specified as 100 μs, the maximum value may be as long as 250 μs.

general-purpose input/output (GPIO)

To use the GP[15:0] software-configurable GPIO pins, the GPxEN bits in the GP Enable (GPEN) Register and the GPxDIR bits in the GP Direction (GPDIR) Register must be properly configured.

GPxEN = 1 GP[x] pin is enabled

GPxDIR = 0 GP[x] pin is an input

GPxDIR = 1 GP[x] pin is an output

where “x” represents one of the 15 through 0 GPIO pins

Figure 7 shows the GPIO enable bits in the GPEN register for the C6411 device. To use any of the GPx pins as general-purpose input/output functions, the corresponding GPxEN bit must be set to “1” (enabled). Default values are device-specific, so refer to Figure 7 for the C6411 default configuration.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GP15 EN	GP14 EN	GP13 EN	GP12 EN	GP11 EN	GP10 EN	GP9 EN	GP8 EN	GP7 EN	GP6 EN	GP5 EN	GP4 EN	GP3 EN	GP2 EN	GP1 EN	GP0 EN
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-1

Legend: R/W = Readable/Writeable; -*n* = value after reset, -x = undefined value after reset

Figure 7. GPIO Enable Register (GPEN) [Hex Address: 01B0 0000]

Figure 8 shows the GPIO direction bits in the GPDIR register. This register determines if a given GPIO pin is an input or an output providing the corresponding GPxEN bit is enabled (set to “1”) in the GPEN register. By default, all the GPIO pins are configured as input pins.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GP15 DIR	GP14 DIR	GP13 DIR	GP12 DIR	GP11 DIR	GP10 DIR	GP9 DIR	GP8 DIR	GP7 DIR	GP6 DIR	GP5 DIR	GP4 DIR	GP3 DIR	GP2 DIR	GP1 DIR	GP0 DIR
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

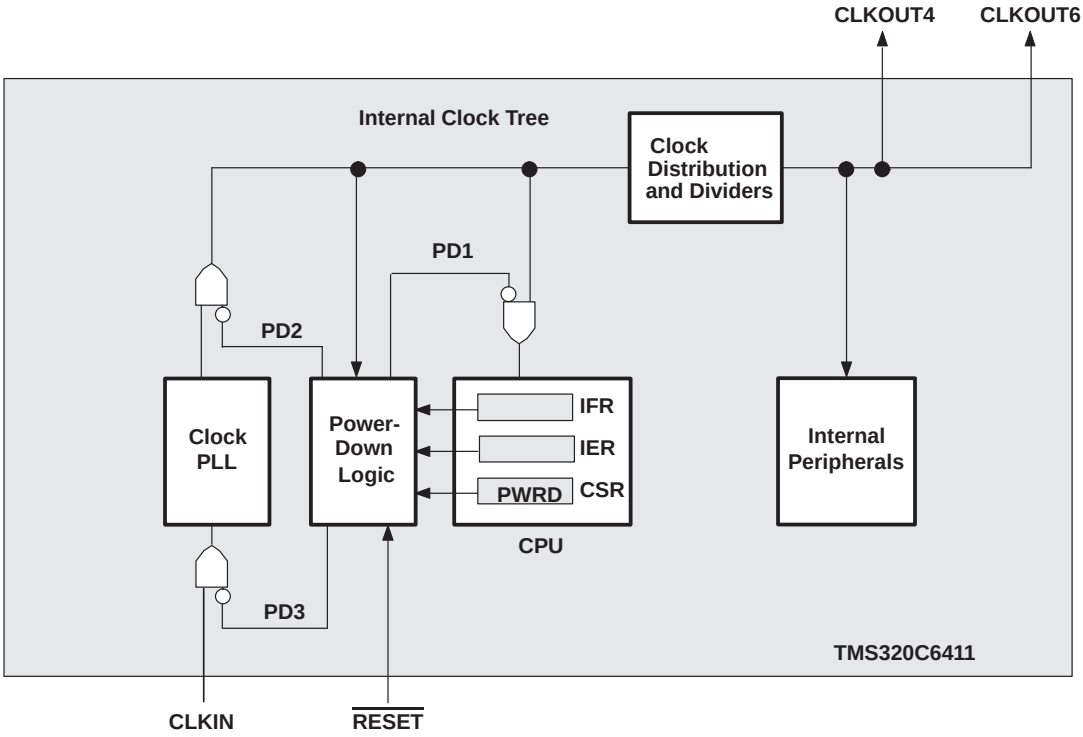
Legend: R/W = Readable/Writeable; -*n* = value after reset, -x = undefined value after reset

Figure 8. GPIO Direction Register (GPDIR) [Hex Address: 01B0 0004]

For more detailed information on general-purpose inputs/outputs (GPIOs), see the *TMS320C6000 DSP General-Purpose Input/Output (GPIO) Reference Guide* (literature number SPRU584).

power-down mode logic

Figure 9 shows the power-down mode logic on the C6411.

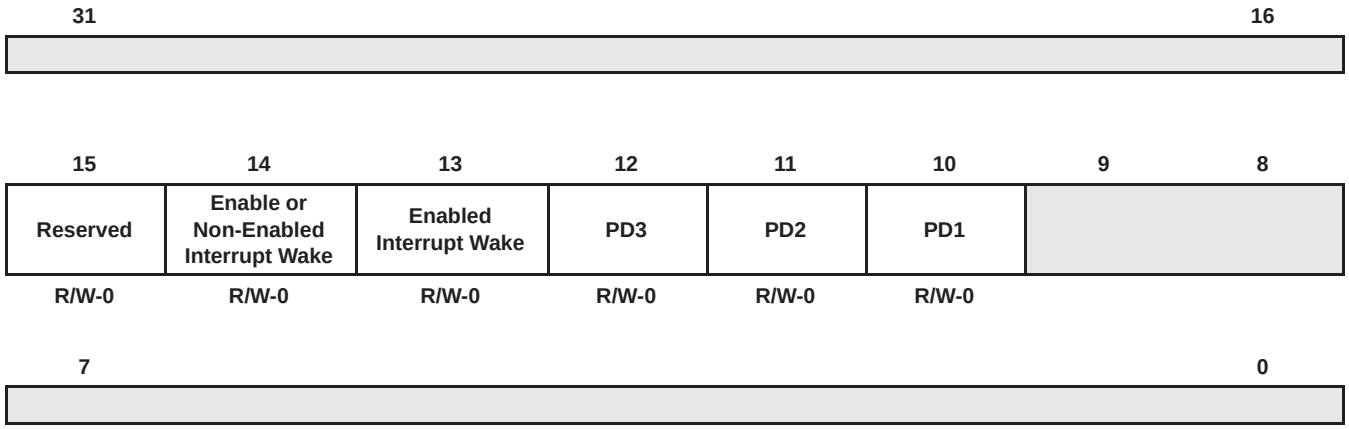


† External input clocks, with the exception of CLKIN, are *not* gated by the power-down mode logic.

Figure 9. Power-Down Mode Logic†

triggering, wake-up, and effects

The power-down modes and their wake-up methods are programmed by setting the PWRD field (bits 15−10) of the control status register (CSR). The PWRD field of the CSR is shown in Figure 10 and described in Table 25. When writing to the CSR, all bits of the PWRD field should be set at the same time. Logic 0 should be used when writing to the reserved bit (bit 15) of the PWRD field. The CSR is discussed in detail in the *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189).



Legend: R/W−x = Read/write reset value

NOTE: The shadowed bits are not part of the power-down logic discussion and therefore are not covered here. For information on these other bit fields in the CSR register, see the *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189).

Figure 10. PWRD Field of the CSR Register

A delay of up to nine clock cycles may occur after the instruction that sets the PWRD bits in the CSR before the PD mode takes effect. As best practice, NOPs should be padded after the PWRD bits are set in the CSR to account for this delay.

If PD1 mode is terminated by a non-enabled interrupt, the program execution returns to the instruction where PD1 took effect. If PD1 mode is terminated by an enabled interrupt, the interrupt service routine will be executed first, then the program execution returns to the instruction where PD1 took effect. In the case with an enabled interrupt, the GIE bit in the CSR and the NMIE bit in the interrupt enable register (IER) must also be set in order for the interrupt service routine to execute; otherwise, execution returns to the instruction where PD1 took effect upon PD1 mode termination by an enabled interrupt.

PD2 and PD3 modes can only be aborted by device reset. Table 25 summarizes all the power-down modes.

Table 25. Characteristics of the Power-Down Modes

PRWD FIELD (BITS 15–10)	POWER-DOWN MODE	WAKE-UP METHOD	EFFECT ON CHIP'S OPERATION
000000	No power-down	—	—
001001	PD1	Wake by an enabled interrupt	CPU halted (except for the interrupt logic) Power-down mode blocks the internal clock inputs at the boundary of the CPU, preventing most of the CPU's logic from switching. During PD1, EDMA transactions can proceed between peripherals and internal memory.
010001	PD1	Wake by an enabled or non-enabled interrupt	
011010	PD2†	Wake by a device reset	Output clock from PLL is halted, stopping the internal clock structure from switching and resulting in the entire chip being halted. All register and internal RAM contents are preserved. All functional I/O "freeze" in the last state when the PLL clock is turned off.
011100	PD3†	Wake by a device reset	Input clock to the PLL stops generating clocks. All register and internal RAM contents are preserved. All functional I/O "freeze" in the last state when the PLL clock is turned off. Following reset, the PLL needs time to re-lock, just as it does following power-up. Wake-up from PD3 takes longer than wake-up from PD2 because the PLL needs to be re-locked, just as it does following power-up.
All others	Reserved	—	—

† When entering PD2 and PD3, all functional I/O remains in the previous state. However, for peripherals which are asynchronous in nature or peripherals with an external clock source, output signals may transition in response to stimulus on the inputs. Under these conditions, peripherals will not operate according to specifications.

C64x power-down mode with an emulator

If user power-down modes are programmed, and an emulator is attached, the modes will be masked to allow the emulator access to the system. This condition prevails until the emulator is reset or the cable is removed from the header. If power measurements are to be performed when in a power-down mode, the emulator cable should be removed.

When the DSP is in power-down mode PD2 or PD3, emulation logic will force any emulation execution command (such as Step or Run) to spin in IDLE. For this reason, PC writes (such as loading code) will fail. A DSP reset will be required to get the DSP out of PD2/PD3.

power-supply sequencing

TI DSPs do not require specific power sequencing between the core supply and the I/O supply. However, systems should be designed to ensure that neither supply is powered up for extended periods of time (>1 second) if the other supply is below the proper operating voltage.

power-supply design considerations

A dual-power supply with simultaneous sequencing can be used to eliminate the delay between core and I/O power up. A Schottky diode can also be used to tie the core rail to the I/O rail (see Figure 11).

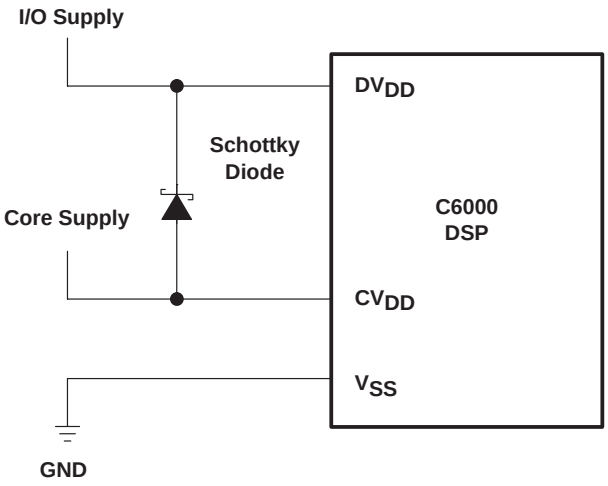


Figure 11. Schottky Diode Diagram

Core and I/O supply voltage regulators should be located close to the DSP (or DSP array) to minimize inductance and resistance in the power delivery path. Additionally, when designing for high-performance applications utilizing the C6000™ platform of DSPs, the PC board should include separate power planes for core, I/O, and ground, all bypassed with high-quality low-ESL/ESR capacitors.

power-supply decoupling

In order to properly decouple the supply planes from system noise, place as many capacitors (caps) as possible close to the DSP. Assuming 0603 caps, the user should be able to fit a total of 60 caps, 30 for the core supply and 30 for the I/O supply. These caps need to be close to the DSP power pins, no more than 1.25 cm maximum distance to be effective. Physically smaller caps, such as 0402, are better because of their lower parasitic inductance. Proper capacitance values are also important. Small bypass caps (near 560 pF) should be closest to the power pins. Medium bypass caps (220 pF or as large as can be obtained in a small package) should be next closest. TI recommends no less than 8 small and 8 medium caps per supply (32 total) be placed immediately next to the BGA vias, using the “interior” BGA space and at least the corners of the “exterior”.

Eight larger caps (4 for each supply) can be placed further away for bulk decoupling. Large bulk caps (on the order of 100 μF) should be furthest away (but still as close as possible). No less than 4 large caps per supply (8 total) should be placed outside of the BGA.

Any cap selection needs to be evaluated from a yield/manufacturing point-of-view. As with the selection of any component, verification of capacitor availability over the product’s production lifetime should be considered.

IEEE 1149.1 JTAG compatibility statement

The TMS320C6411 DSP requires that both $\overline{\text{TRST}}$ and $\overline{\text{RESET}}$ be asserted upon power up to be properly initialized. While $\overline{\text{RESET}}$ initializes the DSP core, $\overline{\text{TRST}}$ initializes the DSP's emulation logic. Both resets are required for proper operation.

Note: $\overline{\text{TRST}}$ is synchronous and ***must*** be clocked by TCLK; otherwise, BSCAN may not respond as expected after $\overline{\text{TRST}}$ is asserted.

While both $\overline{\text{TRST}}$ and $\overline{\text{RESET}}$ need to be asserted upon power up, only $\overline{\text{RESET}}$ needs to be released for the DSP to boot properly. $\overline{\text{TRST}}$ may be asserted indefinitely for normal operation, keeping the JTAG port interface and DSP's emulation logic in the reset state. $\overline{\text{TRST}}$ only needs to be released when it is necessary to use a JTAG controller to debug the DSP or exercise the DSP's boundary scan functionality. $\overline{\text{RESET}}$ must be released in order for boundary-scan JTAG to read the variant field of IDCODE correctly. Other boundary-scan instructions work correctly independent of current state of $\overline{\text{RESET}}$.

For maximum reliability, the TMS320C6411 DSP includes an internal pulldown (IPD) on the $\overline{\text{TRST}}$ pin to ensure that $\overline{\text{TRST}}$ will always be asserted upon power up and the DSP's internal emulation logic will always be properly initialized. JTAG controllers from Texas Instruments actively drive $\overline{\text{TRST}}$ high. However, some third-party JTAG controllers may not drive $\overline{\text{TRST}}$ high but expect the use of a pullup resistor on $\overline{\text{TRST}}$. When using this type of JTAG controller, assert $\overline{\text{TRST}}$ to intialize the DSP after powerup and externally drive $\overline{\text{TRST}}$ high before attempting any emulation or boundary scan operations.

Following the release of $\overline{\text{RESET}}$, the low-to-high transition of $\overline{\text{TRST}}$ must occur to latch the state of EMU1 and EMU0. The EMU[1:0] pins configure the device for either Boundary Scan mode or Normal/Emulation mode. For more detailed information, see the terminal functions section of this data sheet.

Note: The DESIGN_WARNING section of the TMS320C6411 BSDL file contains information and constraints regarding proper device operation while in Boundary Scan Mode.

EMIF device speed

The rated EMIF speed of this device only applies to the SDRAM interface when in a system that meets the following requirements:

- 1 chip-enable (CE) space (maximum of 2 chips) of SDRAM connected to EMIF
- up to 1 CE space of buffers connected to EMIF
- EMIF trace lengths between 1 and 3 inches
- 143-MHz SDRAM for 75-MHz operation

Other configurations may be possible, but timing analysis must be done to verify all AC timings are met. Verification of AC timings is mandatory when using configurations other than those specified above.

TI recommends utilizing the input/output buffer information specification (IBIS) models to analyze all AC timings. To properly use IBIS models to attain accurate timing analysis for a given system, see the *Using IBIS Models for Timing Analysis* application report (literature number SPRA839).

To maintain signal integrity, serial termination resistors should be inserted into all EMIF output signal lines (see the Terminal Functions table for the EMIF output signals).

bootmode

The C6411 device resets using the active-low signal $\overline{\text{RESET}}$. While $\overline{\text{RESET}}$ is low, the device is held in reset and is initialized to the prescribed reset state. Refer to reset timing for reset timing characteristics and states of device pins during reset. The release of $\overline{\text{RESET}}$ starts the processor running with the prescribed device configuration and boot mode.

The C6411 has three types of boot modes:

- Host boot

  If host boot is selected, upon release of $\overline{\text{RESET}}$, the CPU is internally “stalled” while the remainder of the device is released. During this period, an external host can initialize the CPU’s memory space as necessary through the host interface, including internal configuration registers, such as those that control the EMIF or other peripherals. For the C6411 device, the HPI peripheral is used for host boot if PCI_EN = 0, and the PCI peripheral is used if PCI_EN = 1. Once the host is finished with all necessary initialization, it must set the DSPINT bit in the HPIC register to complete the boot process. This transition causes the boot configuration logic to bring the CPU out of the “stalled” state. The CPU then begins execution from address 0. The DSPINT condition is not latched by the CPU, because it occurs while the CPU is still internally “stalled”. Also, DSPINT brings the CPU out of the “stalled” state only if the host boot process is selected. All memory may be written to and read by the host. This allows for the host to verify what it sends to the DSP if required. After the CPU is out of the “stalled” state, the CPU needs to clear the DSPINT, otherwise, no more DSPINTs can be received.

- EMIF boot (using default ROM timings)

  Upon the release of $\overline{\text{RESET}}$, the 1K-Byte ROM code located in the beginning of $\overline{\text{CE1}}$ is copied to address 0 by the EDMA using the default ROM timings, while the CPU is internally “stalled”. The data should be stored in the endian format that the system is using. In this case, the EMIF automatically assembles consecutive 8-bit bytes to form the 32-bit instruction words to be copied. The transfer is automatically done by the EDMA as a single-frame block transfer from the ROM to address 0. After completion of the block transfer, the CPU is released from the “stalled” state and starts running from address 0.

- No boot

  With no boot, the CPU begins direct execution from the memory located at address 0. Note: operation is undefined if invalid code is located at address 0.

reset

A hardware reset ($\overline{\text{RESET}}$) is required to place the DSP into a known good state out of power-up. The $\overline{\text{RESET}}$ signal can be asserted (pulled low) prior to ramping the core and I/O voltages or after the core and I/O voltages have reached their proper operating conditions. As a best practice, reset should be held low during power-up. Prior to deasserting $\overline{\text{RESET}}$ (low-to-high transition), the core and I/O voltages should be at their proper operating conditions and CLKIN should also be running at the correct frequency.

absolute maximum ratings over operating case temperature range (unless otherwise noted)†

Supply voltage ranges: CV_{DD} (see Note 1) . . . . . − 0.3 V to 1.8 V
DV_{DD} (see Note 1) . . . . . −0.3 V to 4 V
Input voltage ranges: (except PCI), V_I . . . . . −0.3 V to 4 V
(PCI), V_{IP} . . . . . −0.5 V to DV_{DD} + 0.5 V
Output voltage ranges: (except PCI), V_O . . . . . −0.3 V to 4 V
(PCI), V_{OP} . . . . . −0.5 V to DV_{DD} + 0.5 V
Operating case temperature range, T_C . . . . . 0°C to 90°C
Storage temperature range, T_{stg} . . . . . −65°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to V_{SS}.

recommended operating conditions

		MIN	NOM	MAX	UNIT
CV_{DD}	Supply voltage, Core‡	1.14	1.2	1.26	V
DV_{DD}	Supply voltage, I/O	3.14	3.3	3.46	V
V_{SS}	Supply ground	0	0	0	V
V_{IH}	High-level input voltage (except PCI)	2			V
V_{IL}	Low-level input voltage (except PCI)			0.8	V
V_{IP}	Input voltage (PCI)	−0.5		DV_{DD} + 0.5	V
V_{IHP}	High-level input voltage (PCI)	$0.5DV_{DD}$		DV_{DD} + 0.5	V
V_{ILP}	Low-level input voltage (PCI)	−0.5		$0.3DV_{DD}$	V
V_{OS}	Maximum voltage during overshoot/undershoot	−1.0§		4.3§	V
T_C	Operating case temperature	0		90	°C

‡ Future variants of the C6411 DSP may operate at voltages ranging from 0.9 V to 1.4 V to provide a range of system power/performance options. TI highly recommends that users design-in a supply that can handle multiple voltages within this range (i.e., 1.2 V, 1.25 V, 1.3 V, 1.35 V, 1.4 V with ± 3% tolerances) by implementing simple board changes such as reference resistor values or input pin configuration modifications. Examples of such supplies include the PT4660, PT5500, PT5520, PT6440, and PT6930 series from Power Trends, a subsidiary of Texas Instruments. Not incorporating a flexible supply may limit the system’s ability to easily adapt to future versions of C641x devices.

§ The absolute maximum ratings should *not* be exceeded for more than 30% of the cycle period.

electrical characteristics over recommended ranges of supply voltage and operating case temperature (unless otherwise noted)

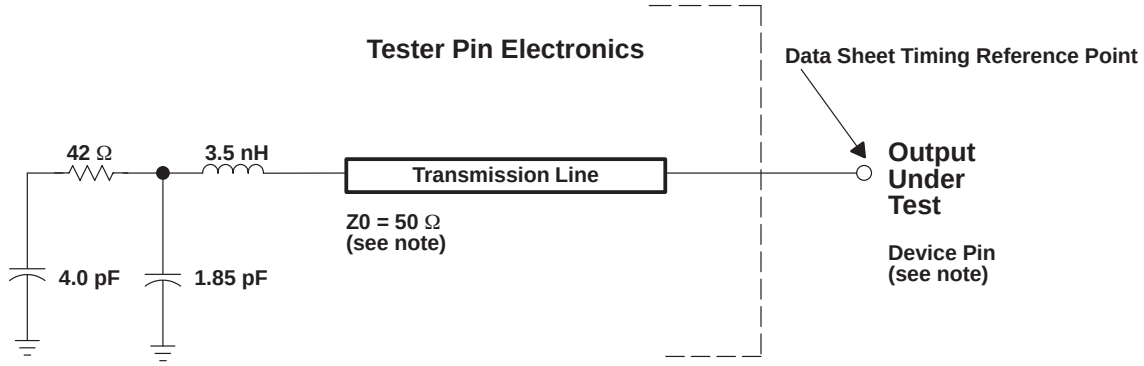
PARAMETER		TEST CONDITIONS†		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage (except PCI)	DV_{DD} = MIN,	I_{OH} = MAX	2.4			V
V_{OHP}	High-level output voltage (PCI)	I_{OHP} = −0.5 mA,	DV_{DD} = 3.3 V	$0.9DV_{DD}$¶			V
V_{OL}	Low-level output voltage (except PCI)	DV_{DD} = MIN,	I_{OL} = MAX			0.4	V
V_{OLP}	Low-level output voltage (PCI)	I_{OLP} = 1.5 mA,	DV_{DD} = 3.3 V			$0.1DV_{DD}$¶	V
I_I	Input current (except PCI)	V_I = V_{SS} to DV_{DD} no opposing internal resistor				±10	uA
		V_I = V_{SS} to DV_{DD} opposing internal pullup resistor‡		50	100	150	uA
		V_I = V_{SS} to DV_{DD} opposing internal pulldown resistor‡		−150	−100	−50	uA
I_{IP}	Input leakage current (PCI)§	0 < V_{IP} < DV_{DD},	DV_{DD} = 3.3 V			±10	uA
I_{OH}	High-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx				−16	mA
		Timer, TDO, GPIO (Excluding GP[15:9, 2, 1]), McBSP				−8	mA
		PCI/HPI				−0.5§¶	mA
I_{OL}	Low-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx				16	mA
		Timer, TDO, GPIO (Excluding GP[15:9, 2, 1]), McBSP				8	mA
		PCI/HPI				1.5¶	mA
I_{OZ}	Off-state output current	V_O = DV_{DD} or 0 V				±10	uA
I_{CDD}	Core supply current#	CV_{DD} = 1.2 V, CPU clock = 300 MHz			550		mA
I_{DDD}	I/O supply current#	DV_{DD} = 3.3 V, CPU clock = 300 MHz			125		mA
C_i	Input capacitance					10	pF
C_o	Output capacitance					10	pF

† For test conditions shown as MIN, MAX, or NOM, use the appropriate value specified in the recommended operating conditions table.
‡ Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.
§ PCI input leakage currents include Hi-Z output leakage for all bidirectional buffers with 3-state outputs.
¶ These rated numbers are from the PCI specification version 2.3. The DC specification and AC specification are defined in Tables 4-3 and 4-4, respectively.
# Measured with average activity (50% high/50% low power). The actual current draw is highly application-dependent. For more details on core and I/O activity, refer to the *TMS320C6411 Power Consumption Summary* application report (literature number SPRA373).

recommended clock and control signal transition behavior

All clocks and control signals ***must*** transition between V_{IH} and V_{IL} (or between V_{IL} and V_{IH}) in a monotonic manner.

PARAMETER MEASUREMENT INFORMATION



NOTE: The data sheet provides timing at the device pin. For output timing analysis, the tester pin electronics and its transmission line effects must be taken into account. A transmission line with a delay of 2 ns or longer can be used to produce the desired transmission line effect. The transmission line is intended as a load only. It is not necessary to add or subtract the transmission line delay (2 ns or longer) from the data sheet timings.

Input requirements in this data sheet are tested with an input slew rate of < 4 Volts per nanosecond (4 V/ns) at the device pin.

Figure 12. Test Load Circuit for AC Timing Measurements

The tester load circuit is for characterization and measurement of AC timing signals. This load does not indicate the maximum load the device is capable of driving.

signal transition levels

All input and output timing parameters are referenced to 1.5 V for both “0” and “1” logic levels.

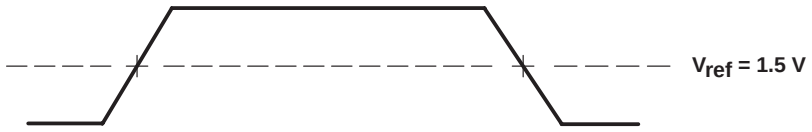


Figure 13. Input and Output Voltage Reference Levels for AC Timing Measurements

All rise and fall transition timing parameters are referenced to V_{IL} MAX and V_{IH} MIN for input clocks, V_{OL} MAX and V_{OH} MIN for output clocks, V_{ILP} MAX and V_{IHP} MIN for PCI input clocks, and V_{OLP} MAX and V_{OHP} MIN for PCI output clocks.

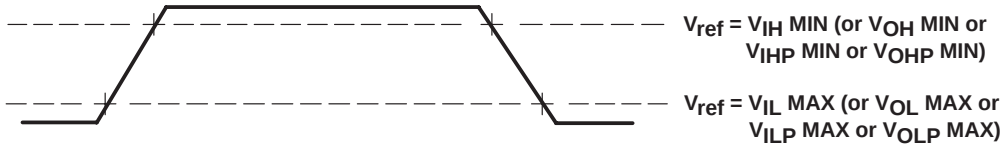


Figure 14. Rise and Fall Transition Time Voltage Reference Levels

signal transition rates

All timings are tested with an input edge rate of 4 Volts per nanosecond (4 V/ns).

PARAMETER MEASUREMENT INFORMATION (CONTINUED)

timing parameters and board routing analysis

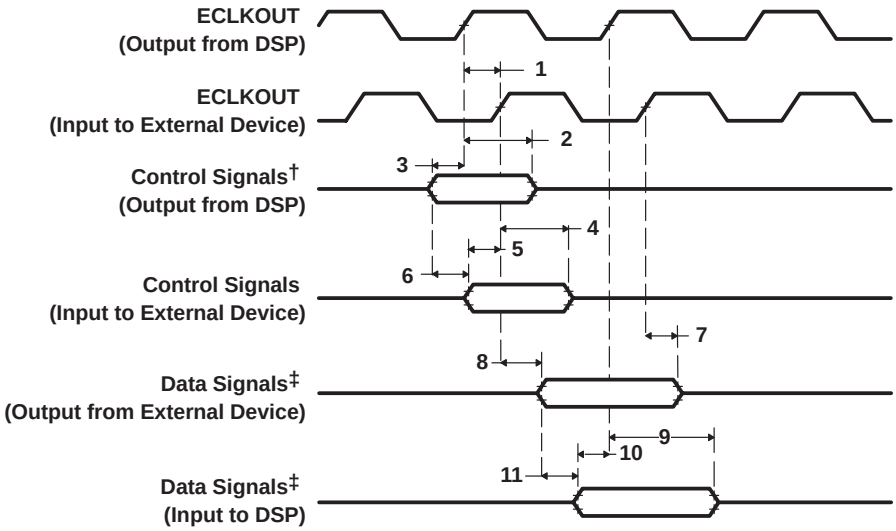
The timing parameter values specified in this data sheet do *not* include delays by board routings. As a good board design practice, such delays must *always* be taken into account. Timing values may be adjusted by increasing/decreasing such delays. TI recommends utilizing the available I/O buffer information specification (IBIS) models to analyze the timing characteristics correctly. To properly use IBIS models to attain accurate timing analysis for a given system, see the *Using IBIS Models for Timing Analysis* application report (literature number SPRA839). If needed, external logic hardware such as buffers may be used to compensate any timing differences.

For inputs, timing is most impacted by the round-trip propagation delay from the DSP to the external device and from the external device to the DSP. This round-trip delay tends to negatively impact the input setup time margin, but also tends to improve the input hold time margins (see Table 26 and Figure 15).

Figure 15 represents a general transfer between the DSP and an external device. The figure also represents board route delays and how they are perceived by the DSP and the external device.

Table 26. Board-Level Timings Example (see Figure 15)

NO.	DESCRIPTION
1	Clock route delay
2	Minimum DSP hold time
3	Minimum DSP setup time
4	External device hold time requirement
5	External device setup time requirement
6	Control signal route delay
7	External device hold time
8	External device access time
9	DSP hold time requirement
10	DSP setup time requirement
11	Data route delay



† Control signals include data for Writes.
‡ Data signals are generated during Reads from an external device.

Figure 15. Board-Level Input/Output Timings

INPUT AND OUTPUT CLOCKS

timing requirements for CLKIN†‡§ (see Figure 16)

NO.			−300				UNIT
			PLL MODE x6		x1 (BYPASS)		
			MIN	MAX	MIN	MAX	
1	$t_{c(CLKIN)}$	Cycle time, CLKIN	20	33.3	13.3	33.3	ns
2	$t_{w(CLKINH)}$	Pulse duration, CLKIN high	0.4C		0.45C		ns
3	$t_{w(CLKINL)}$	Pulse duration, CLKIN low	0.4C		0.45C		ns
4	$t_{t(CLKIN)}$	Transition time, CLKIN		5		1	ns
5	$t_{J(CLKIN)}$	Period jitter, CLKIN		0.02C		0.02C	ns

† The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
‡ For more details on the PLL multiplier factor (x6), see the *Clock PLL* section of this data sheet.
§ C = CLKIN cycle time in ns. For example, when CLKIN frequency is 50 MHz, use C = 20 ns.

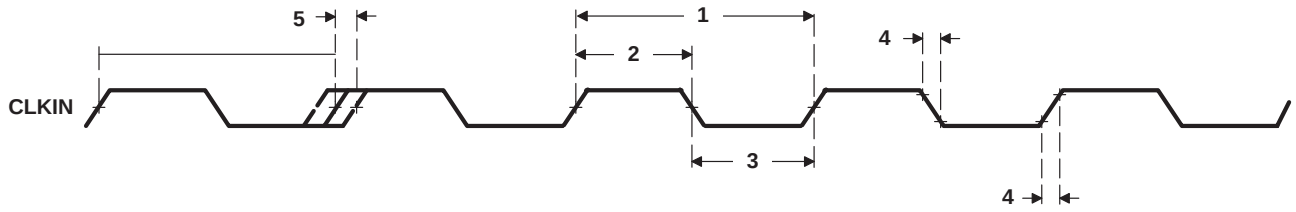


Figure 16. CLKIN Timing

INPUT AND OUTPUT CLOCKS (CONTINUED)

switching characteristics over recommended operating conditions for CLKOUT4†‡§ (see Figure 17)

NO.	PARAMETER		−300 CLKMODE = x1, x6 MIN	−300 CLKMODE = x1, x6 MAX	UNIT
1	$t_{J(CKO4)}$	Period jitter, CLKOUT4	0	±175	ps
2	$t_{w(CKO4H)}$	Pulse duration, CLKOUT4 high	2P − 0.7	2P + 0.7	ns
3	$t_{w(CKO4L)}$	Pulse duration, CLKOUT4 low	2P − 0.7	2P + 0.7	ns
4	$t_{t(CKO4)}$	Transition time, CLKOUT4		1	ns

† The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
‡ PH is the high period of CLKIN in ns and PL is the low period of CLKIN in ns.
§ P = 1/CPU clock frequency in nanoseconds (ns)

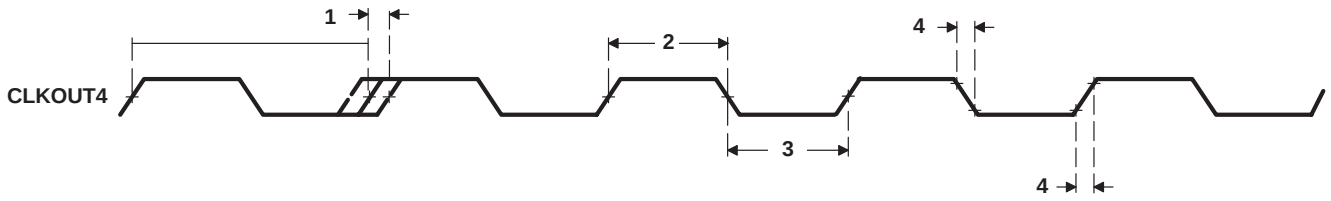


Figure 17. CLKOUT4 Timing

switching characteristics over recommended operating conditions for CLKOUT6†‡§ (see Figure 18)

NO.	PARAMETER		−300 CLKMODE = x1, x6 MIN	−300 CLKMODE = x1, x6 MAX	UNIT
1	$t_{J(CKO6)}$	Period jitter, CLKOUT6	0	±175	ps
2	$t_{w(CKO6H)}$	Pulse duration, CLKOUT6 high	3P − 0.7	3P + 0.7	ns
3	$t_{w(CKO6L)}$	Pulse duration, CLKOUT6 low	3P − 0.7	3P + 0.7	ns
4	$t_{t(CKO6)}$	Transition time, CLKOUT6		1	ns

† The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
‡ PH is the high period of CLKIN in ns and PL is the low period of CLKIN in ns.
§ P = 1/CPU clock frequency in nanoseconds (ns)

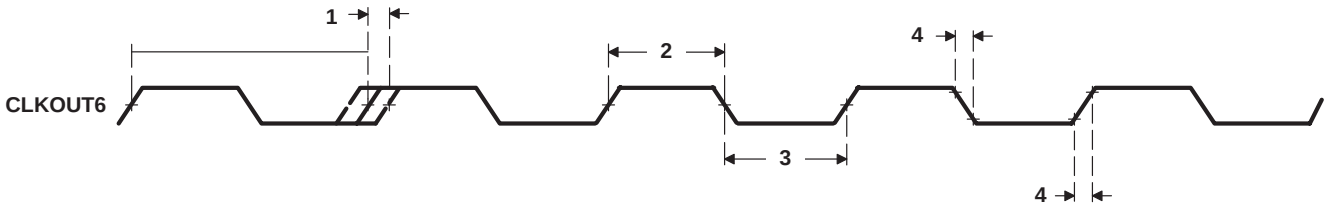


Figure 18. CLKOUT6 Timing

INPUT AND OUTPUT CLOCKS (CONTINUED)

timing requirements for ECLKIN†‡§ (see Figure 19)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{c(EKI)}$	Cycle time, ECLKIN	7.5¶	16P	ns
2	$t_{w(EKIH)}$	Pulse duration, ECLKIN high	3.38		ns
3	$t_{w(EKIL)}$	Pulse duration, ECLKIN low	3.38		ns
4	$t_{t(EKI)}$	Transition time, ECLKIN		2	ns
5	$t_{J(EKI)}$	Period jitter, ECLKIN		0.02E	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
§ E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns for EMIFA or EMIFB.
¶ Minimum ECLKIN times are based on internal logic speed; the maximum useable speed of the EMIF may be lower due to AC timing requirements. On the -300 device, 75-MHz operation is achievable if the requirements of the EMIF Device Speed section are met. Minimum ECLKIN cycle times *must* be met, even when ECLKIN is generated by an internal clock source.

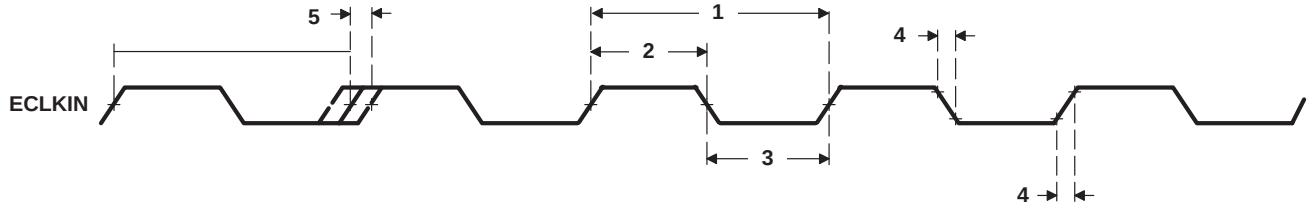


Figure 19. ECLKIN Timing

switching characteristics over recommended operating conditions for ECLKOUT1§#‖ (see Figure 20)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{J(EKO1)}$	Period jitter, ECLKOUT1	0	±175☆	ps
2	$t_{w(EKO1H)}$	Pulse duration, ECLKOUT1 high	EH − 0.7	EH + 0.7	ns
3	$t_{w(EKO1L)}$	Pulse duration, ECLKOUT1 low	EL − 0.7	EL + 0.7	ns
4	$t_{t(EKO1)}$	Transition time, ECLKOUT1		1	ns
5	$t_{d(EKIH-EKO1H)}$	Delay time, ECLKIN high to ECLKOUT1 high	1	8	ns
6	$t_{d(EKIL-EKO1L)}$	Delay time, ECLKIN low to ECLKOUT1 low	1	8	ns

§ E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.
# The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
‖ EH is the high period of E (EMIF input clock period) in ns and EL is the low period of E (EMIF input clock period) in ns.
☆This cycle-to-cycle jitter specification was measured with CPU/4 or CPU/6 as the source of the EMIF input clock.

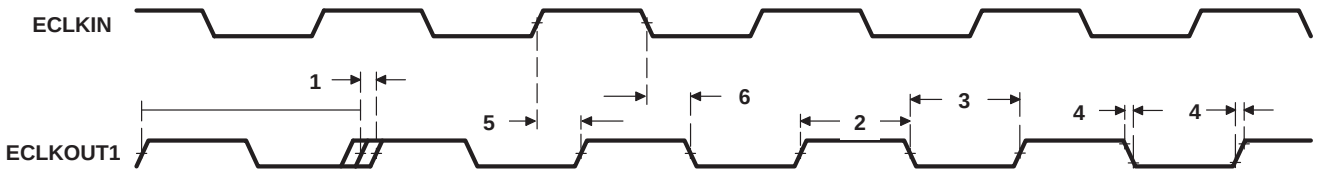


Figure 20. ECLKOUT1 Timing

INPUT AND OUTPUT CLOCKS (CONTINUED)

switching characteristics over recommended operating conditions for ECLKOUT2†‡ (see Figure 21)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{J(EKO2)}$	Period jitter, ECLKOUT2	0	±175§	ps
2	$t_{w(EKO2H)}$	Pulse duration, ECLKOUT2 high	0.5NE − 0.7	0.5NE + 0.7	ns
3	$t_{w(EKO2L)}$	Pulse duration, ECLKOUT2 low	0.5NE − 0.7	0.5NE + 0.7	ns
4	$t_{t(EKO2)}$	Transition time, ECLKOUT2		1	ns
5	$t_{d(EKIH-EKO2H)}$	Delay time, ECLKIN high to ECLKOUT2 high	1	8	ns
6	$t_{d(EKIH-EKO2L)}$	Delay time, ECLKIN high to ECLKOUT2 low	1	8	ns

† The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
‡ E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.
N = the EMIF input clock divider; N = 1, 2, or 4.
§ This cycle-to-cycle jitter specification was measured with CPU/4 or CPU/6 as the source of the EMIF input clock.

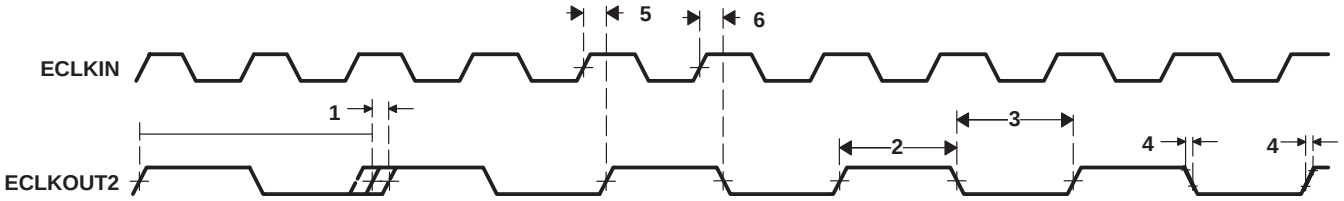


Figure 21. ECLKOUT2 Timing

ASYNCHRONOUS MEMORY TIMING

timing requirements for asynchronous memory cycles†‡ (see Figure 22 and Figure 23)

NO.				−300		UNIT
				MIN	MAX	
3	$t_{su(EDV-AREH)}$	Setup time, EDx valid before $\overline{ARE}$ high		6.5		ns
4	$t_{h(AREH-EDV)}$	Hold time, EDx valid after $\overline{ARE}$ high		1		ns
6	$t_{su(ARDY-EKO1H)}$	Setup time, ARDY valid before ECLKOUTx high		3		ns
7	$t_{h(EKO1H-ARDY)}$	Hold time, ARDY valid after ECLKOUTx high	Rev 1.1	1		ns
			Rev 2.0	1.3		ns

† To ensure data setup time, simply program the strobe width wide enough. ARDY is internally synchronized. The ARDY signal is *only* recognized two cycles before the end of the programmed strobe time and while ARDY is low, the strobe time is extended cycle-by-cycle. When ARDY is recognized low, the end of the strobe time is two cycles after ARDY is recognized high. To use ARDY as an asynchronous input, the pulse width of the ARDY signal should be wide enough (e.g., pulse width = 2E) to ensure setup and hold time is met.

‡ RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold. These parameters are programmed via the EMIF CE space control registers.

switching characteristics over recommended operating conditions for asynchronous memory cycles‡§¶ (see Figure 22 and Figure 23)

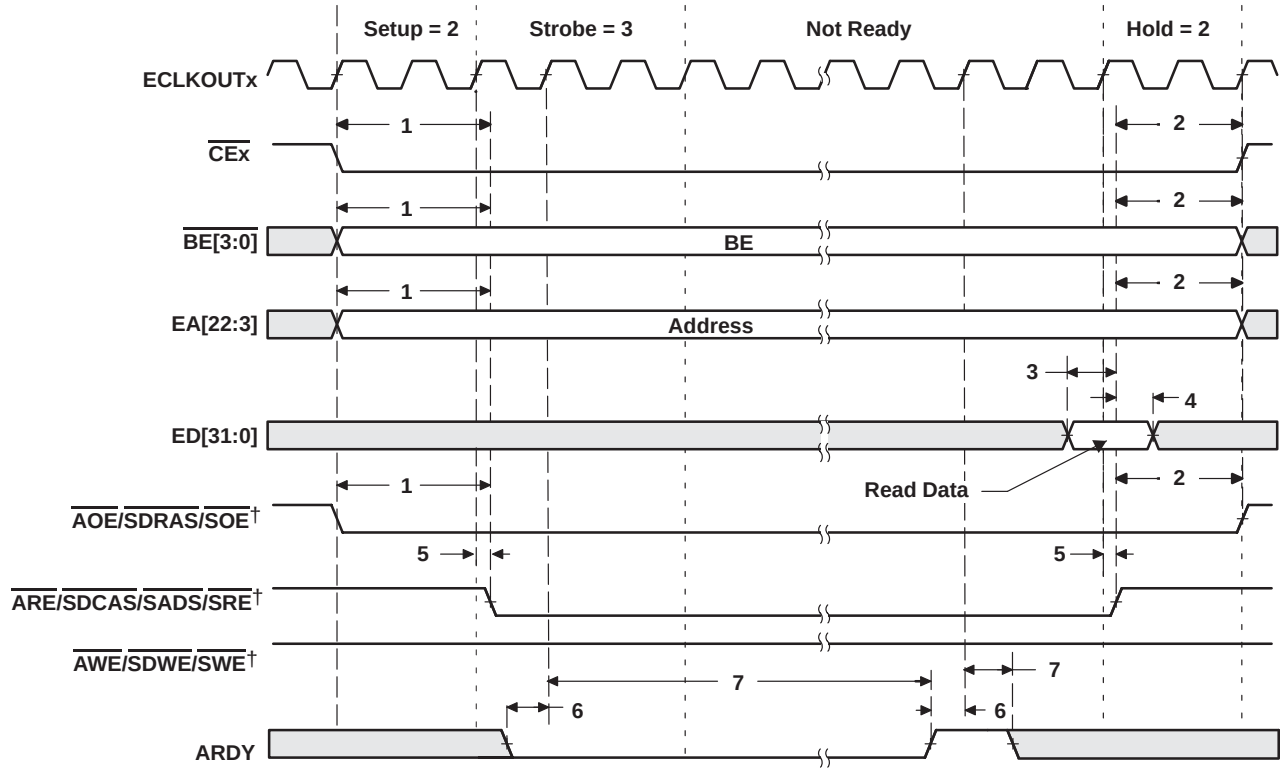
NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{osu(SELV-AREL)}$	Output setup time, select signals valid to $\overline{ARE}$ low	RS * E − 1.5		ns
2	$t_{oh(AREH-SELIV)}$	Output hold time, $\overline{ARE}$ high to select signals invalid	RS * E − 1.9		ns
5	$t_{d(EKO1H-AREV)}$	Delay time, ECLKOUTx high to $\overline{ARE}$ vaild	1	7	ns
8	$t_{osu(SELV-AWEL)}$	Output setup time, select signals valid to $\overline{AWE}$ low	WS * E − 1.7		ns
9	$t_{oh(AWEH-SELIV)}$	Output hold time, $\overline{AWE}$ high to select signals invalid	WH * E − 1.8		ns
10	$t_{d(EKO1H-AWEV)}$	Delay time, ECLKOUTx high to $\overline{AWE}$ vaild	1.3	7.1	ns

‡ RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold. These parameters are programmed via the EMIF CE space control registers.

§ E = ECLKOUT1 period in ns

¶ Select signals for EMIF include: $\overline{CEx}$, $\overline{BE[3:0]}$, EA[22:3], $\overline{AOE}$; and for EMIF writes, include ED[31:0].

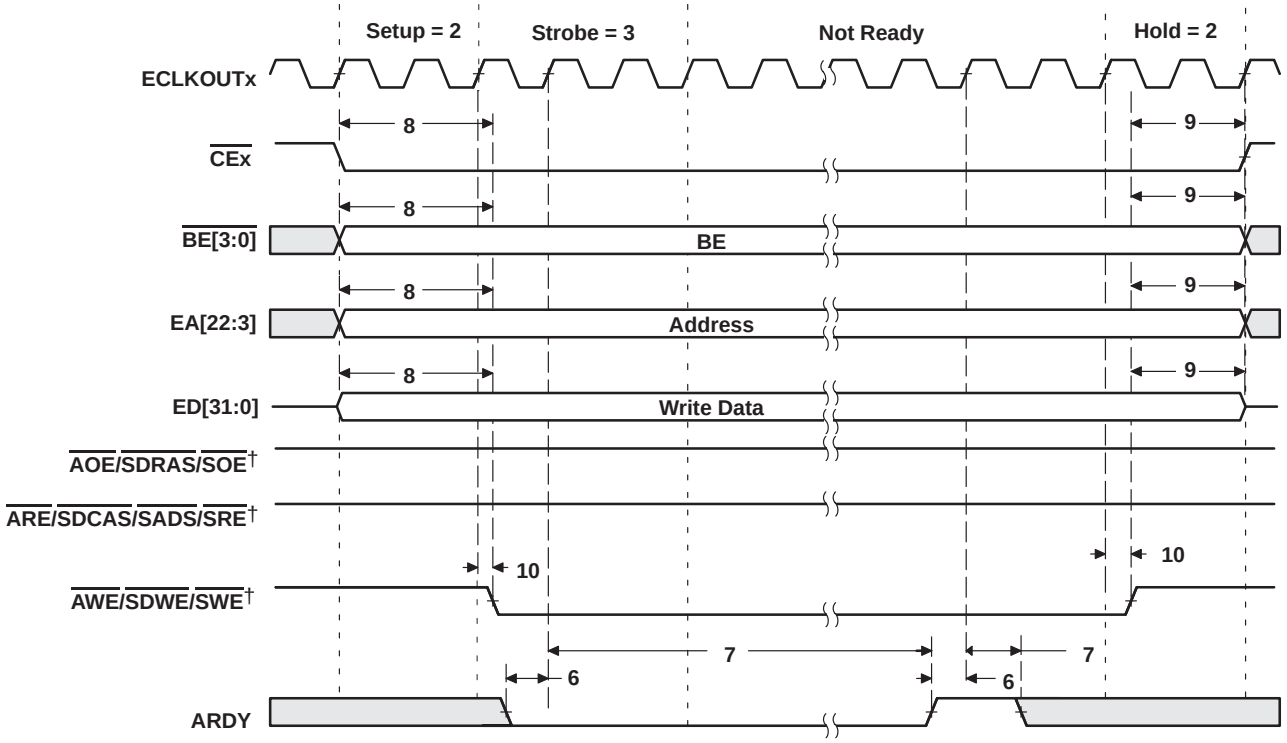
ASYNCHRONOUS MEMORY TIMING (CONTINUED)



† AOE/SDRAS/SOE, ARE/SDCAS/SADS/SRE, and AWE/SDWE/SWE operate as AOE (identified under select signals), ARE, and AWE, respectively, during asynchronous memory accesses.

Figure 22. Asynchronous Memory Read Timing

ASYNCHRONOUS MEMORY TIMING (CONTINUED)



† AOE/SDRAS/SOE, ARE/SDCAS/SADS/SRE, and AWE/SDWE/SWE operate as AOE (identified under select signals), ARE, and AWE, respectively, during asynchronous memory accesses.

Figure 23. Asynchronous Memory Write Timing

PROGRAMMABLE SYNCHRONOUS INTERFACE TIMING

timing requirements for programmable synchronous interface cycles (see Figure 24)

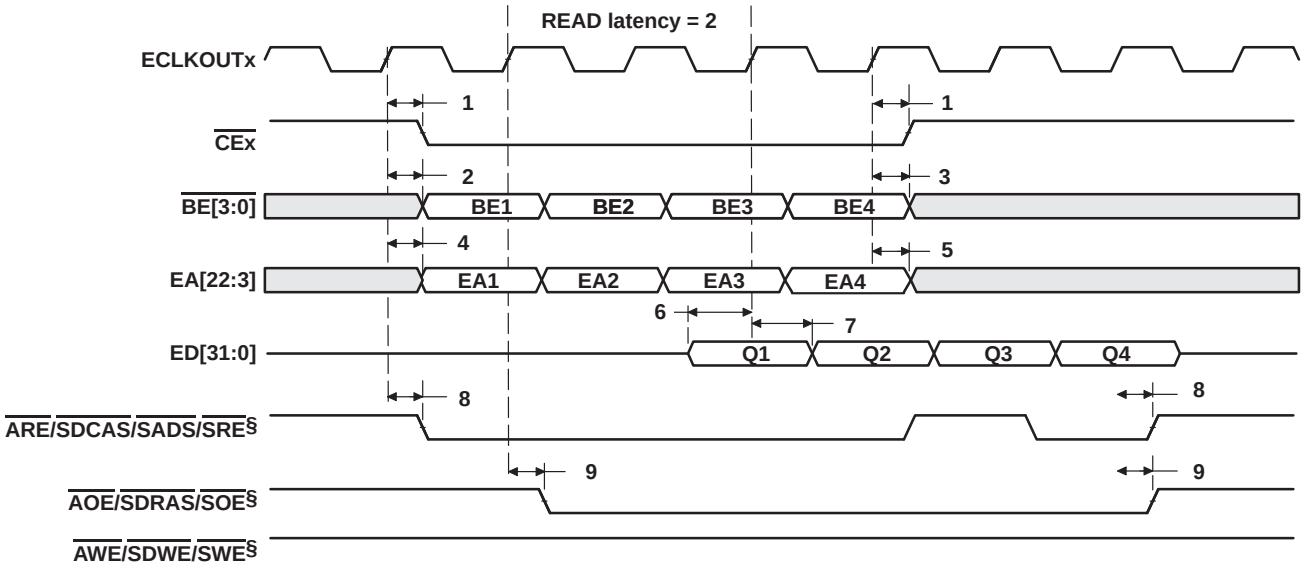
NO.			−300		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKOxH)}$	Setup time, read EDx valid before ECLKOUTx high	6.4		ns
7	$t_{h(EKOxH-EDV)}$	Hold time, read EDx valid after ECLKOUTx high	1.5		ns

switching characteristics over recommended operating conditions for programmable synchronous interface cycles† (see Figure 24−Figure 26)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{d(EKOxH-CEV)}$	Delay time, ECLKOUTx high to $\overline{CEx}$ valid	1.3	9.7	ns
2	$t_{d(EKOxH-BEV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ valid		9.7	ns
3	$t_{d(EKOxH-BEIV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ invalid	1.3		ns
4	$t_{d(EKOxH-EAV)}$	Delay time, ECLKOUTx high to EAx valid		9.7	ns
5	$t_{d(EKOxH-EAIV)}$	Delay time, ECLKOUTx high to EAx invalid	1.3		ns
8	$t_{d(EKOxH-ADSV)}$	Delay time, ECLKOUTx high to $\overline{SADS}/\overline{SRE}$ valid	1.3	9.7	ns
9	$t_{d(EKOxH-OEV)}$	Delay time, ECLKOUTx high to, $\overline{SOE}$ valid	1.3	9.7	ns
10	$t_{d(EKOxH-EDV)}$	Delay time, ECLKOUTx high to EDx valid		9.7	ns
11	$t_{d(EKOxH-EDIV)}$	Delay time, ECLKOUTx high to EDx invalid	1.3		ns
12	$t_{d(EKOxH-WEV)}$	Delay time, ECLKOUTx high to $\overline{SWE}$ valid	1.3	9.7	ns

† The following parameters are programmable via the EMIF CE Space Secondary Control register (CExSEC):
- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- CEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, CEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, CEx is active when SOE is active (CEEXT = 1).
- Function of SADS/SRE (RENEN): For standard SBSRAM or ZBT SRAM interface, SADS/SRE acts as SADS with deselect cycles (RENEN = 0). For FIFO interface, SADS/SRE acts as SRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

PROGRAMMABLE SYNCHRONOUS INTERFACE TIMING (CONTINUED)



† The read latency and the length of $\overline{\text{CEx}}$ assertion are programmable via the SYNCRL and CEEXT fields, respectively, in the EMIF CE Space Secondary Control register (CExSEC). In this figure, SYNCRL = 2 and CEEXT = 0.

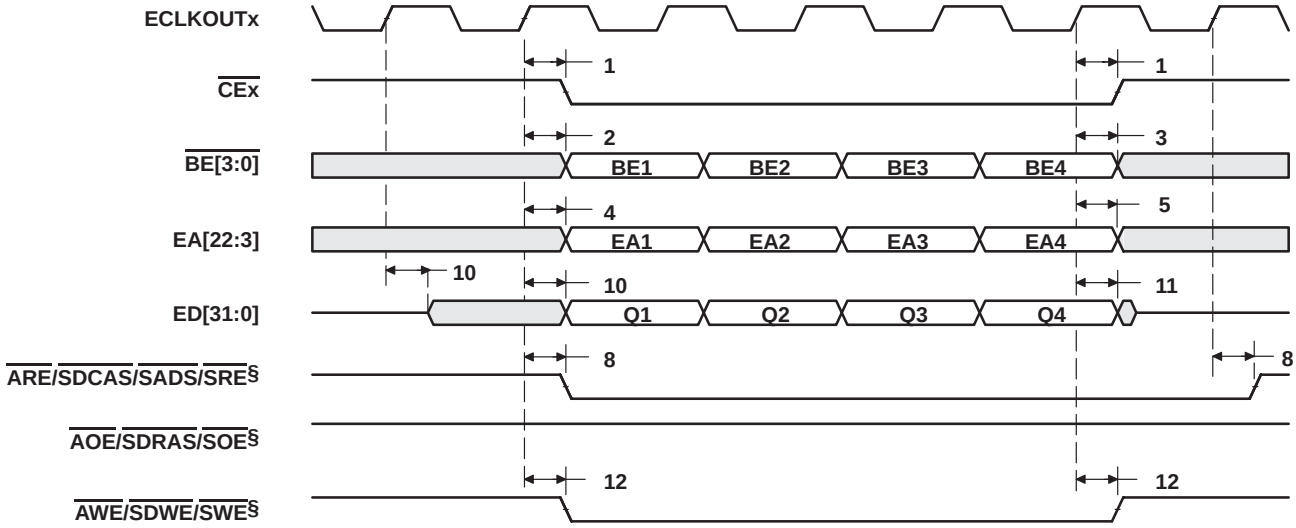
‡ The following parameters are programmable via the EMIF CE Space Secondary Control register (CExSEC):

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- $\overline{\text{CEx}}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{CEx}}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{\text{CEx}}$ is active when $\overline{\text{SOE}}$ is active (CEEXT = 1).
- Function of $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SADS}}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SRE}}$ with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

§ $\overline{\text{ARE}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{AOE}}$/$\overline{\text{SDRAS}}$/$\overline{\text{SOE}}$, and $\overline{\text{AWE}}$/$\overline{\text{SDWE}}$/$\overline{\text{SWE}}$ operate as $\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{SOE}}$, and $\overline{\text{SWE}}$, respectively, during programmable synchronous interface accesses.

Figure 24. Programmable Synchronous Interface Read Timing (With Read Latency = 2)†‡

PROGRAMMABLE SYNCHRONOUS INTERFACE TIMING (CONTINUED)



† The write latency and the length of $\overline{\text{CEx}}$ assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIF CE Space Secondary Control register (CExSEC). In this figure, SYNCWL = 0 and CEEXT = 0.

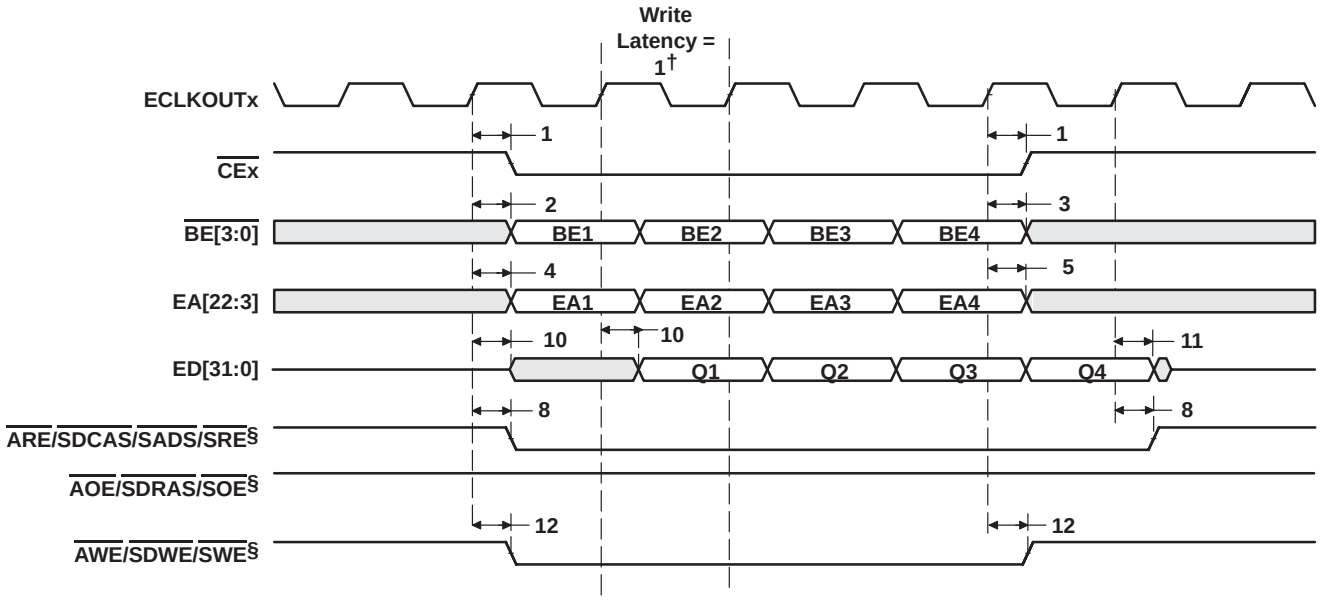
‡ The following parameters are programmable via the EMIF CE Space Secondary Control register (CExSEC):

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- $\overline{\text{CEx}}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{CEx}}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{\text{CEx}}$ is active when $\overline{\text{SOE}}$ is active (CEEXT = 1).
- Function of $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SADS}}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SRE}}$ with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

§ $\overline{\text{ARE}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{AOE}}$/$\overline{\text{SDRAS}}$/$\overline{\text{SOE}}$, and $\overline{\text{AWE}}$/$\overline{\text{SDWE}}$/$\overline{\text{SWE}}$ operate as $\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{SOE}}$, and $\overline{\text{SWE}}$, respectively, during programmable synchronous interface accesses.

Figure 25. Programmable Synchronous Interface Write Timing (With Write Latency = 0)†‡

PROGRAMMABLE SYNCHRONOUS INTERFACE TIMING (CONTINUED)



† The write latency and the length of $\overline{CEx}$ assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIF CE Space Secondary Control register (CExSEC). In this figure, SYNCWL = 1 and CEEXT = 0.

‡ The following parameters are programmable via the EMIF CE Space Secondary Control register (CExSEC):

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- $\overline{CEx}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{CEx}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{CEx}$ is active when $\overline{SOE}$ is active (CEEXT = 1).
- Function of $\overline{SADS}$/$\overline{SRE}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{SADS}$/$\overline{SRE}$ acts as $\overline{SADS}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{SADS}$/$\overline{SRE}$ acts as $\overline{SRE}$ with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

§ $\overline{ARE}$/$\overline{SDCAS}$/$\overline{SADS}$/$\overline{SRE}$, $\overline{AOE}$/$\overline{SDRAS}$/$\overline{SOE}$, and $\overline{AWE}$/$\overline{SDWE}$/$\overline{SWE}$ operate as $\overline{SADS}$/$\overline{SRE}$, $\overline{SOE}$, and $\overline{SWE}$, respectively, during programmable synchronous interface accesses.

Figure 26. Programmable Synchronous Interface Write Timing (With Write Latency = 1)†‡

SYNCHRONOUS DRAM TIMING

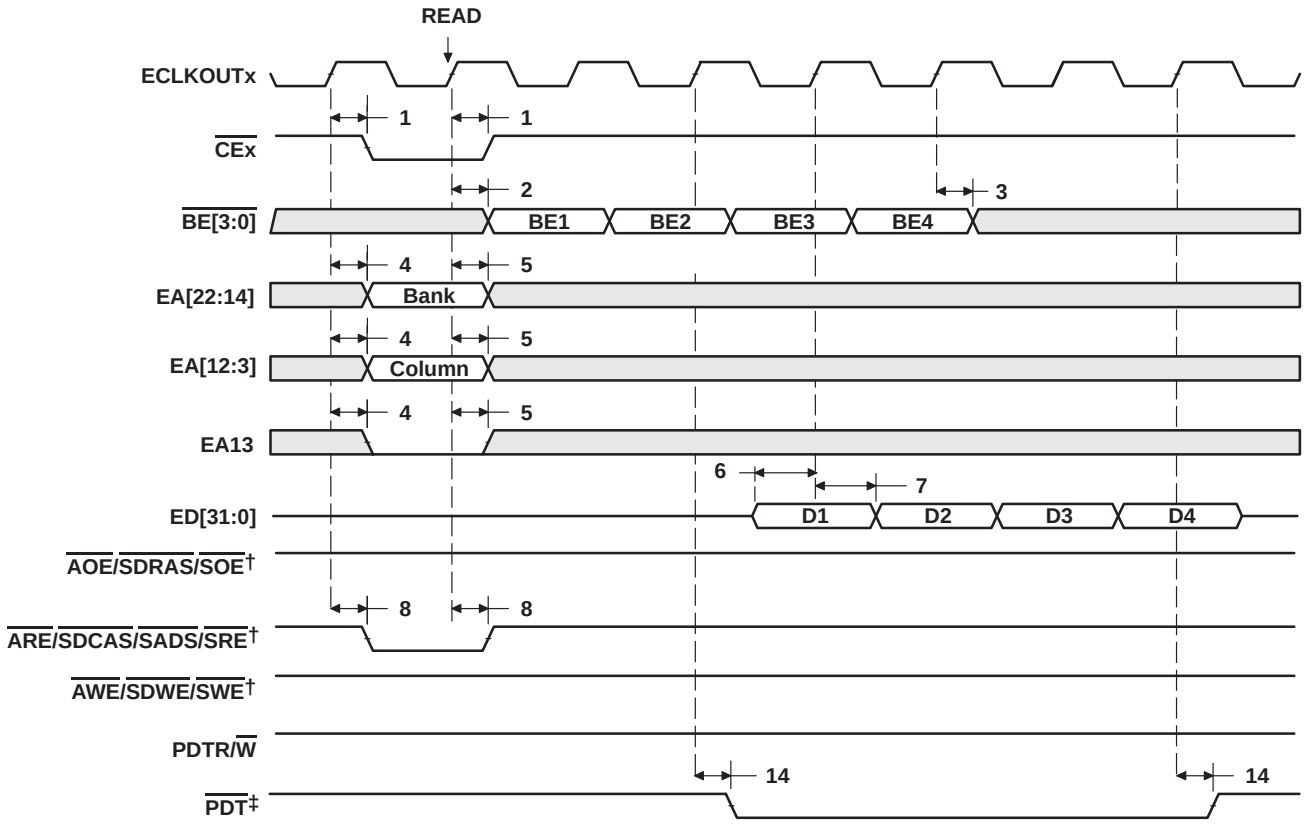
timing requirements for synchronous DRAM cycles (see Figure 27)

NO.			−300		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKO1H)}$	Setup time, read EDx valid before ECLKOUTx high	5.4		ns
7	$t_{h(EKO1H-EDV)}$	Hold time, read EDx valid after ECLKOUTx high	2.5		ns

switching characteristics over recommended operating conditions for synchronous DRAM cycles (see Figure 27−Figure 34)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{d(EKO1H-CEV)}$	Delay time, ECLKOUTx high to $\overline{CE}$x valid	1.3	9.7	ns
2	$t_{d(EKO1H-BEV)}$	Delay time, ECLKOUTx high to $\overline{BE}$x valid		9.7	ns
3	$t_{d(EKO1H-BEIV)}$	Delay time, ECLKOUTx high to $\overline{BE}$x invalid	1.3		ns
4	$t_{d(EKO1H-EAV)}$	Delay time, ECLKOUTx high to EAx valid		9.7	ns
5	$t_{d(EKO1H-EAIV)}$	Delay time, ECLKOUTx high to EAx invalid	1.3		ns
8	$t_{d(EKO1H-CASV)}$	Delay time, ECLKOUTx high to $\overline{SDCAS}$ valid	1.3	9.7	ns
9	$t_{d(EKO1H-EDV)}$	Delay time, ECLKOUTx high to EDx valid		9.7	ns
10	$t_{d(EKO1H-EDIV)}$	Delay time, ECLKOUTx high to EDx invalid	1.3		ns
11	$t_{d(EKO1H-WEV)}$	Delay time, ECLKOUTx high to $\overline{SDWE}$ valid	1.3	9.7	ns
12	$t_{d(EKO1H-RAS)}$	Delay time, ECLKOUTx high to $\overline{SDRAS}$ valid	1.3	9.7	ns
13	$t_{d(EKO1H-SDCKEV)}$	Delay time, ECLKOUTx high to SDCKE valid	1.3	9.7	ns
14	$t_{d(EKO1H-PDTV)}$	Delay time, ECLKOUTx high to $\overline{PDT}$ valid	1.3	9.7	ns

SYNCHRONOUS DRAM TIMING (CONTINUED)

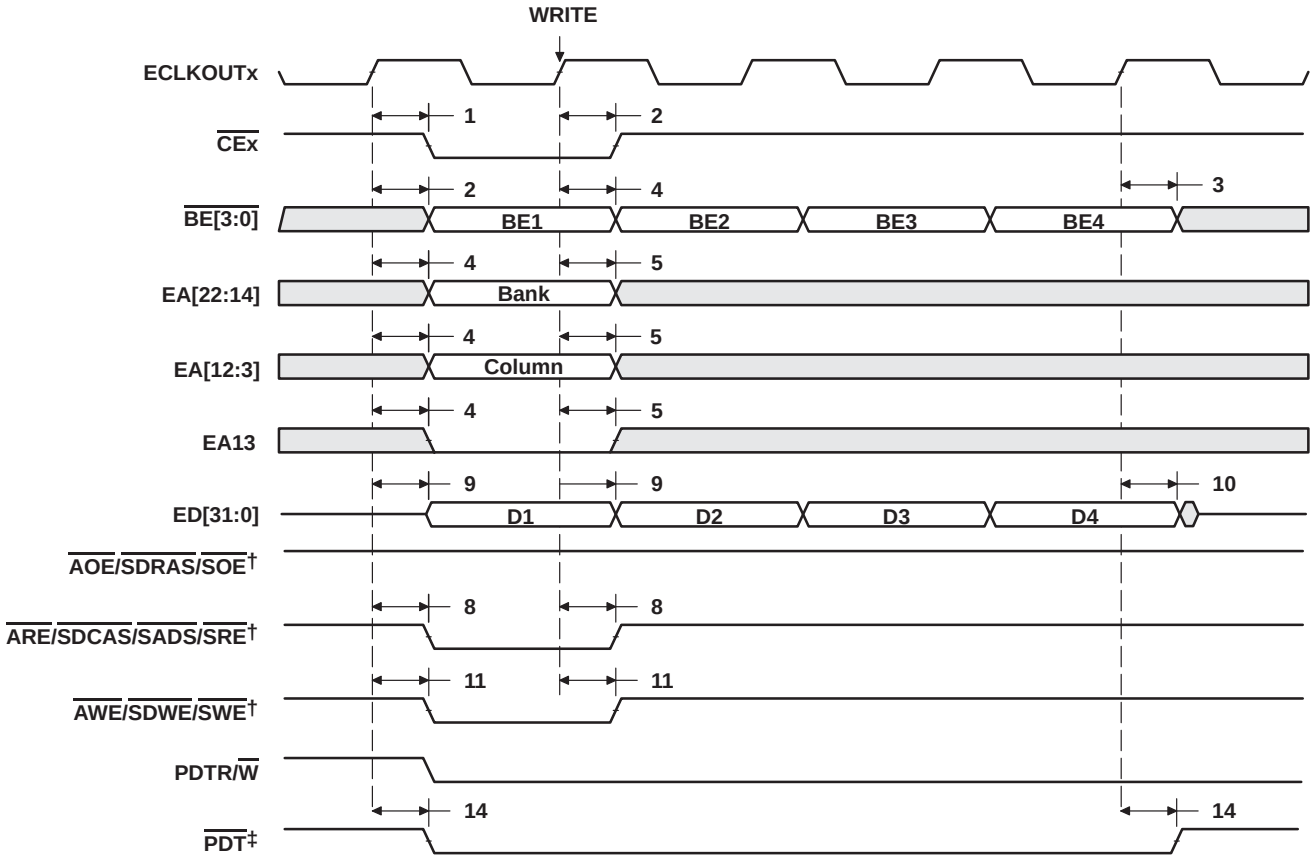


† ARE/SDCAS/SADS/SRE, AWE/SDWE/SWE, and AOE/SDRAS/SOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

‡ PDT signal is only asserted when the EDMA is in PDT mode (set the PDTS bit to 1 in the EDMA options parameter RAM). For PDT read, data is not latched into EMIF. The PDTRL field in the PDT control register (PDTCTL) configures the latency of the PDT signal with respect to the data phase of a read transaction. The latency of the PDT signal for a read can be programmed to 0, 1, 2, or 3 by setting PDTRL to 00, 01, 10, or 11, respectively. PDTRL equals 00 (zero latency) in Figure 27.

Figure 27. SDRAM Read Command (CAS Latency 3)

SYNCHRONOUS DRAM TIMING (CONTINUED)

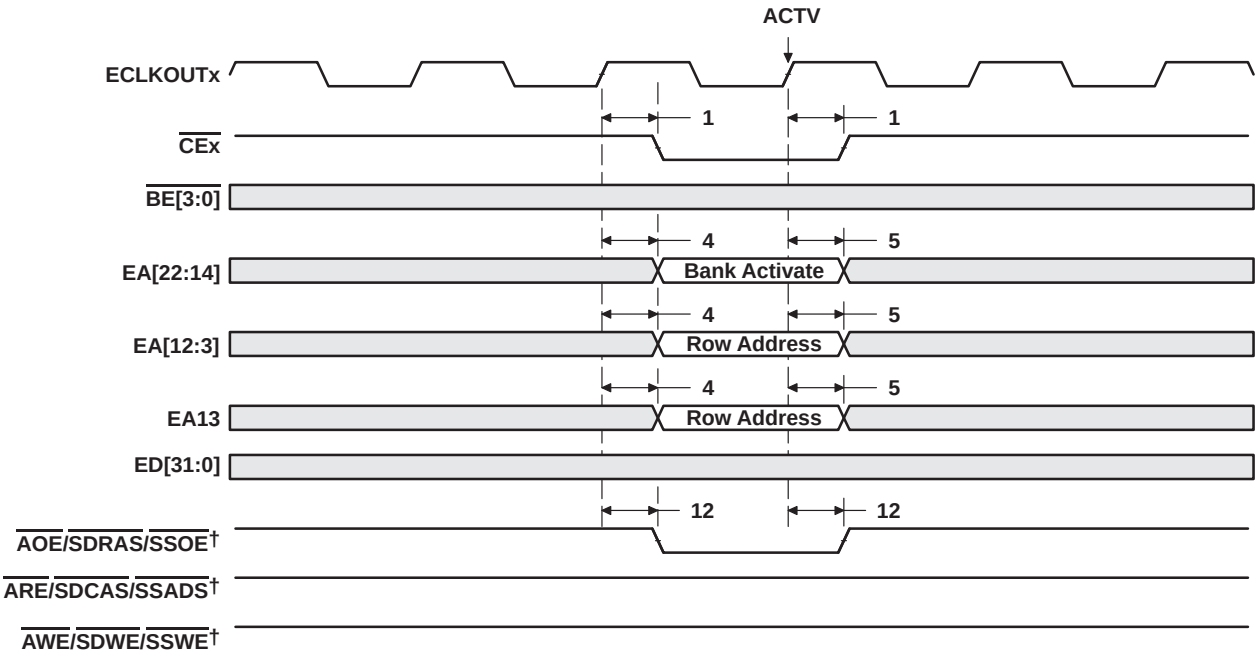


† ARE/SDCAS/SADS/SRE, AWE/SDWE/SWE, and AOE/SDRAS/SOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

‡ PDT signal is only asserted when the EDMA is in PDT mode (set the PDTD bit to 1 in the EDMA options parameter RAM). For PDT write, data is not driven (in High-Z). The PDTWL field in the PDT control register (PDTCTL) configures the latency of the PDT signal with respect to the data phase of a write transaction. The latency of the PDT signal for a write transaction can be programmed to 0, 1, 2, or 3 by setting PDTWL to 00, 01, 10, or 11, respectively. PDTWL equals 00 (zero latency) in Figure 28.

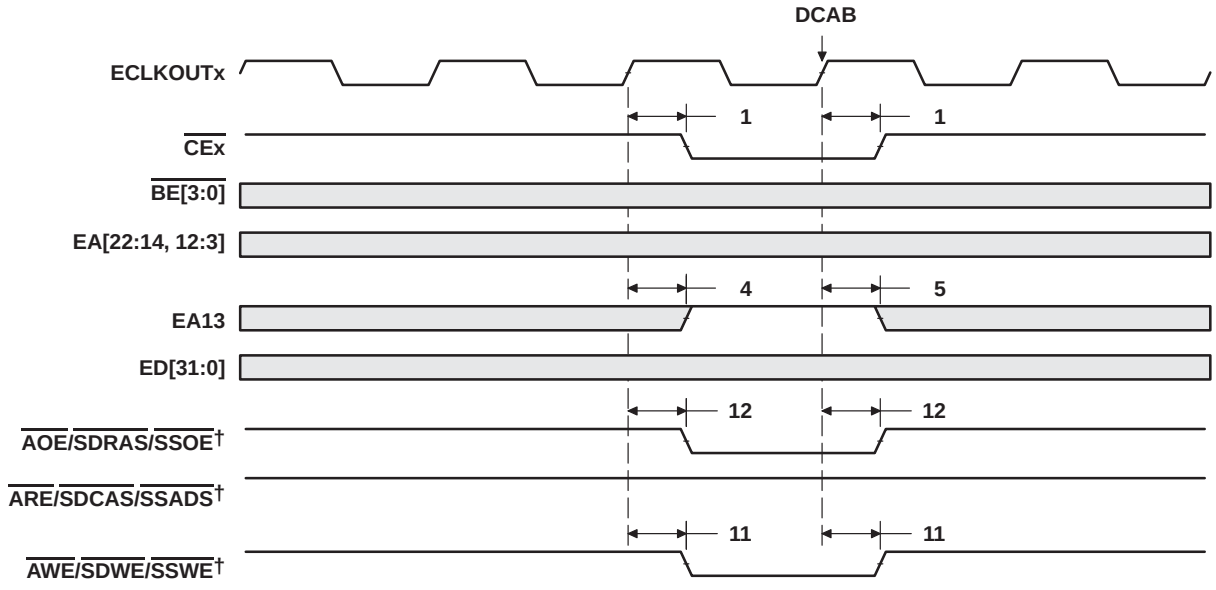
Figure 28. SDRAM Write Command

SYNCHRONOUS DRAM TIMING (CONTINUED)



† ARE/SDCAS/SSADS, AWE/SDWE/SSWE, and AOE/SDRAS/SSOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

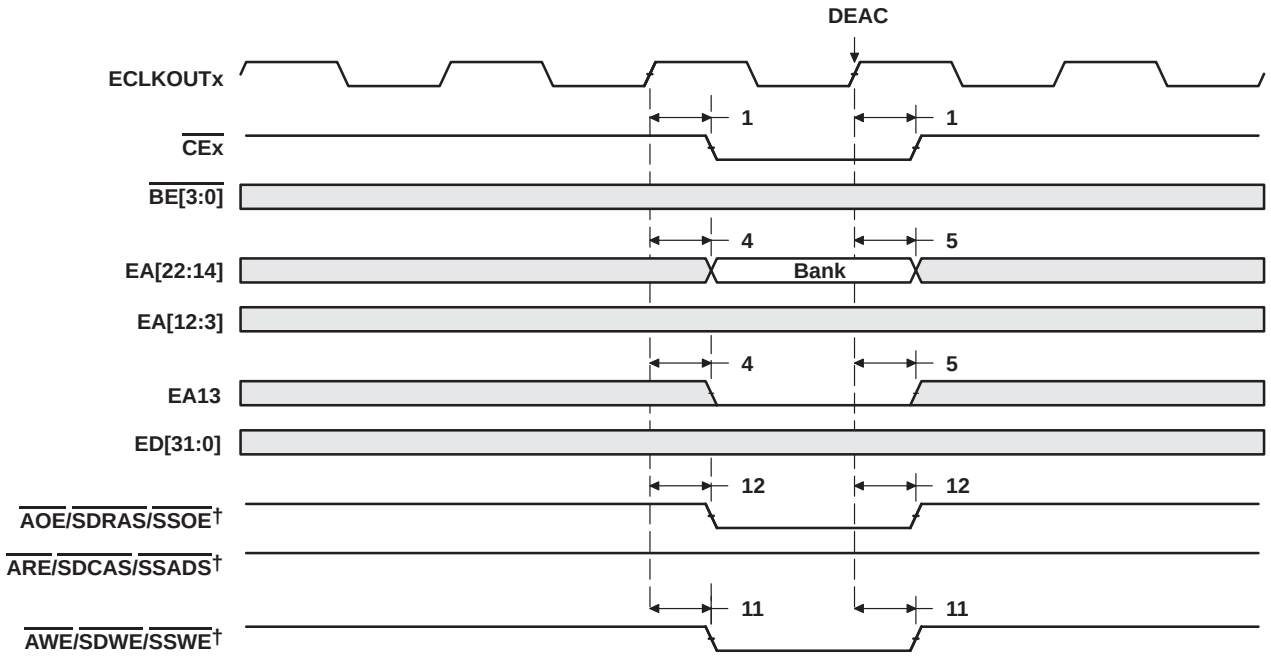
Figure 29. SDRAM ACTV Command



† ARE/SDCAS/SSADS, AWE/SDWE/SSWE, and AOE/SDRAS/SSOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

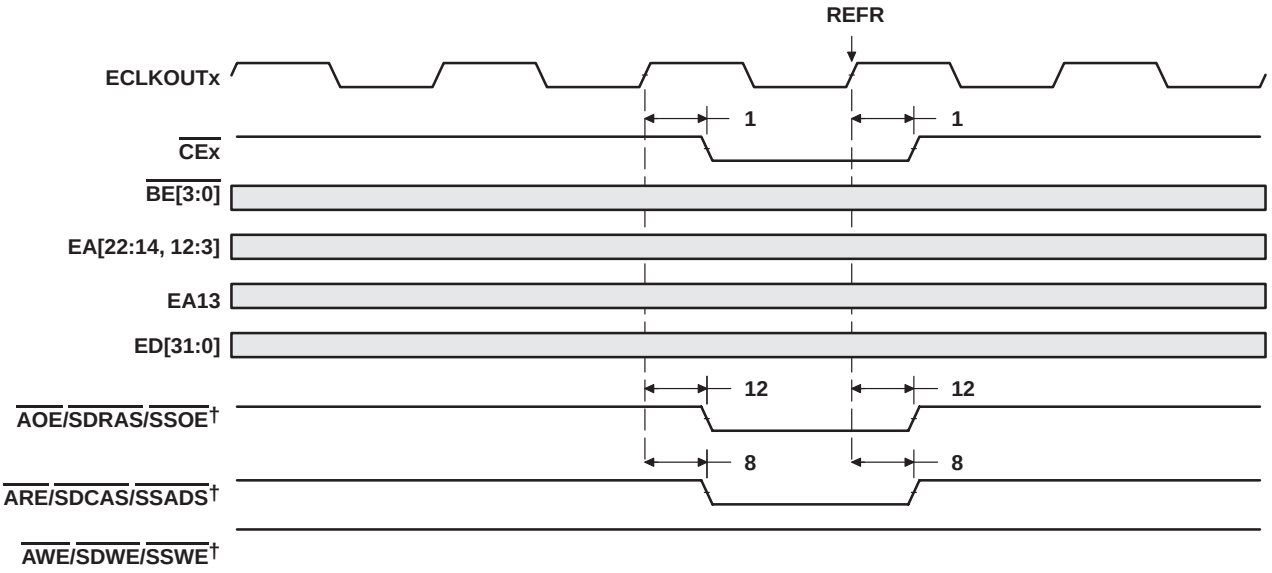
Figure 30. SDRAM DCAB Command

SYNCHRONOUS DRAM TIMING (CONTINUED)



† ARE/SDCAS/SSADS, AWE/SDWE/SSWE, and AOE/SDRAS/SSOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

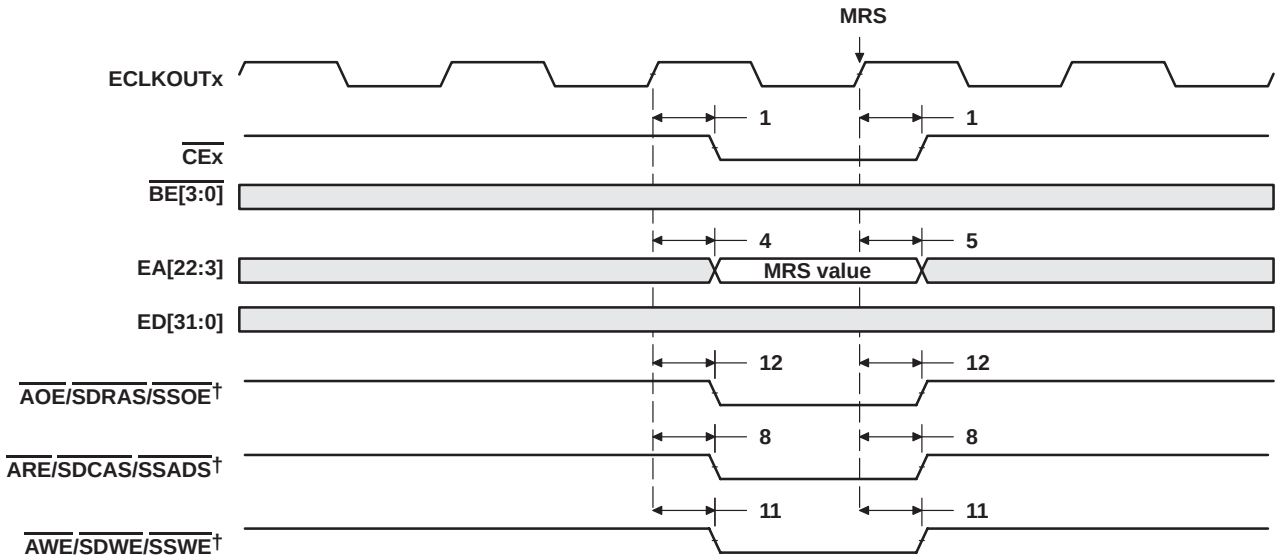
Figure 31. SDRAM DEAC Command



† ARE/SDCAS/SSADS, AWE/SDWE/SSWE, and AOE/SDRAS/SSOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

Figure 32. SDRAM REFR Command

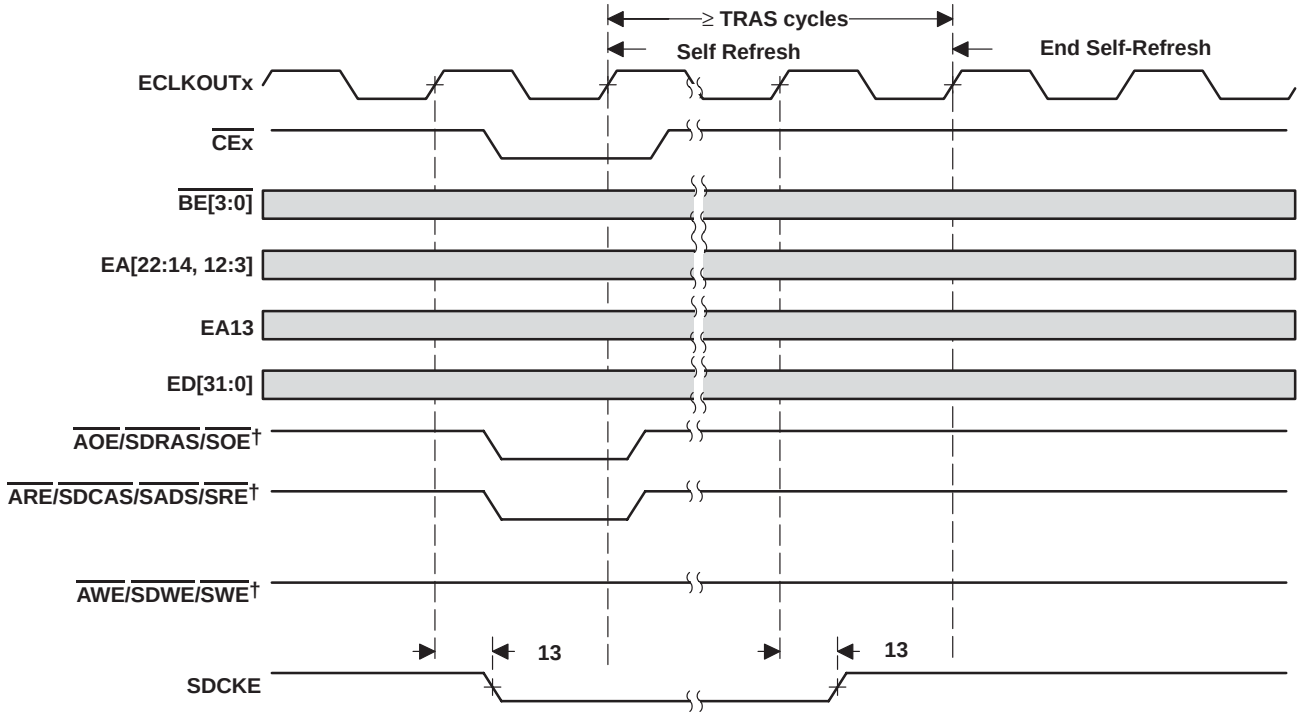
SYNCHRONOUS DRAM TIMING (CONTINUED)



† ARE/SDCAS/SSADS, AWE/SDWE/SSWE, and AOE/SDRAS/SSOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

Figure 33. SDRAM MRS Command

SYNCHRONOUS DRAM TIMING (CONTINUED)



† ARE/SDCAS/SADS/SRE, AWE/SDWE/SWE, and AOE/SDRAS/SOE operate as SDCAS, SDWE, and SDRAS, respectively, during SDRAM accesses.

Figure 34. SDRAM Self-Refresh Timing

HOLD/HOLDA TIMING

timing requirements for the HOLD/HOLDA cycles† (see Figure 35)

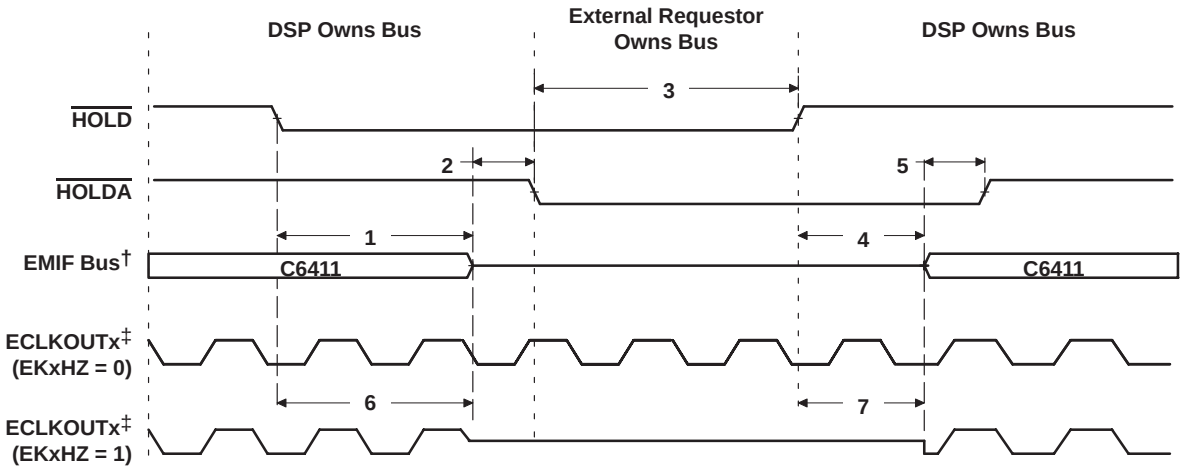
NO.			−300		UNIT
			MIN	MAX	
3	$t_{h(HOLDAL-HOLDL)}$	Hold time, HOLD low after HOLDA low	E		ns

† E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.

switching characteristics over recommended operating conditions for the HOLD/HOLDA cycles†‡§ (see Figure 35)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{d(HOLDL-EMHZ)}$	Delay time, HOLD low to EMIF Bus high impedance	2E	¶	ns
2	$t_{d(EMHZ-HOLDAL)}$	Delay time, EMIF Bus high impedance to HOLDA low	0	2E	ns
4	$t_{d(HOLDH-EMLZ)}$	Delay time, HOLD high to EMIF Bus low impedance	2E	7E	ns
5	$t_{d(EMLZ-HOLDAH)}$	Delay time, EMIF Bus low impedance to HOLDA high	0	2E	ns
6	$t_{d(HOLDL-EKOHZ)}$	Delay time, HOLD low to ECLKOUTx high impedance	2E	¶	ns
7	$t_{d(HOLDH-EKOLZ)}$	Delay time, HOLD high to ECLKOUTx low impedance	2E	7E	ns

† E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.
‡ EMIF Bus consists of: CE[3:0], BE[3:0], ED[31:0], EA[22:3], ARE/SDCAS/SADS/SRE, AOE/SDRAS/SOE, and AWE/SDWE/SWE , SDCKE, SOE3, and PDT.
§ The EKxHZ bits in the EMIF Global Control register (GBLCTL) determine the state of the ECLKOUTx signals during HOLDA. If EKxHZ = 0, ECLKOUTx continues clocking during Hold mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during Hold mode, as shown in Figure 35.
¶ All pending EMIF transactions are allowed to complete before HOLDA is asserted. If no bus transactions are occurring, then the minimum delay time can be achieved. Also, bus hold can be indefinitely delayed by setting NOHOLD = 1.



† EMIF Bus consists of: CE[3:0], BE[3:0], ED[31:0], EA[22:3], ARE/SDCAS/SADS/SRE, AOE/SDRAS/SOE, and AWE/SDWE/SWE, SDCKE, SOE3, and PDT.
‡ The EKxHZ bits in the EMIF Global Control register (GBLCTL) determine the state of the ECLKOUTx signals during HOLDA. If EKxHZ = 0, ECLKOUTx continues clocking during Hold mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during Hold mode, as shown in Figure 35.

Figure 35. HOLD/HOLDA Timing

BUSREQ TIMING

switching characteristics over recommended operating conditions for the BUSREQ cycles (see Figure 36)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{d(EKO1H-BUSRV)}$	Delay time, ECLKOUTx high to BUSREQ valid	0.6	7.1	ns

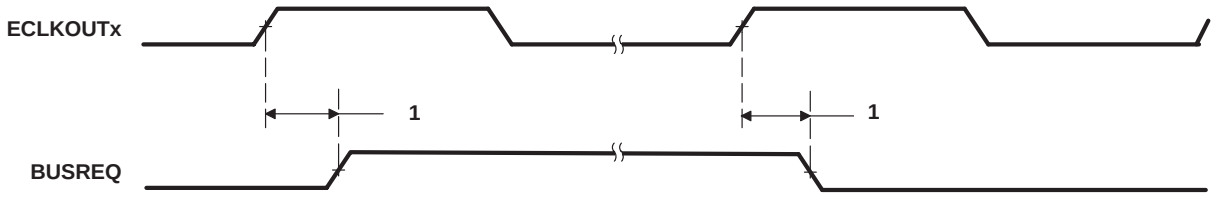


Figure 36. BUSREQ Timing

RESET TIMING

timing requirements for reset† (see Figure 37)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{w(RST)}$	Width of the RESET pulse (PLL stable)‡	10P		ns
		Width of the RESET pulse (PLL needs to sync up)§	250		µs
16	$t_{su(boot)}$	Setup time, boot configuration bits valid before RESET high¶	4E or 4C#		ns
17	$t_{h(boot)}$	Hold time, boot configuration bits valid after RESET high¶	4P		ns
18	$t_{su(PCLK-RSTH)}$	Setup time, PCLK active before RESET high‖	32N		ns

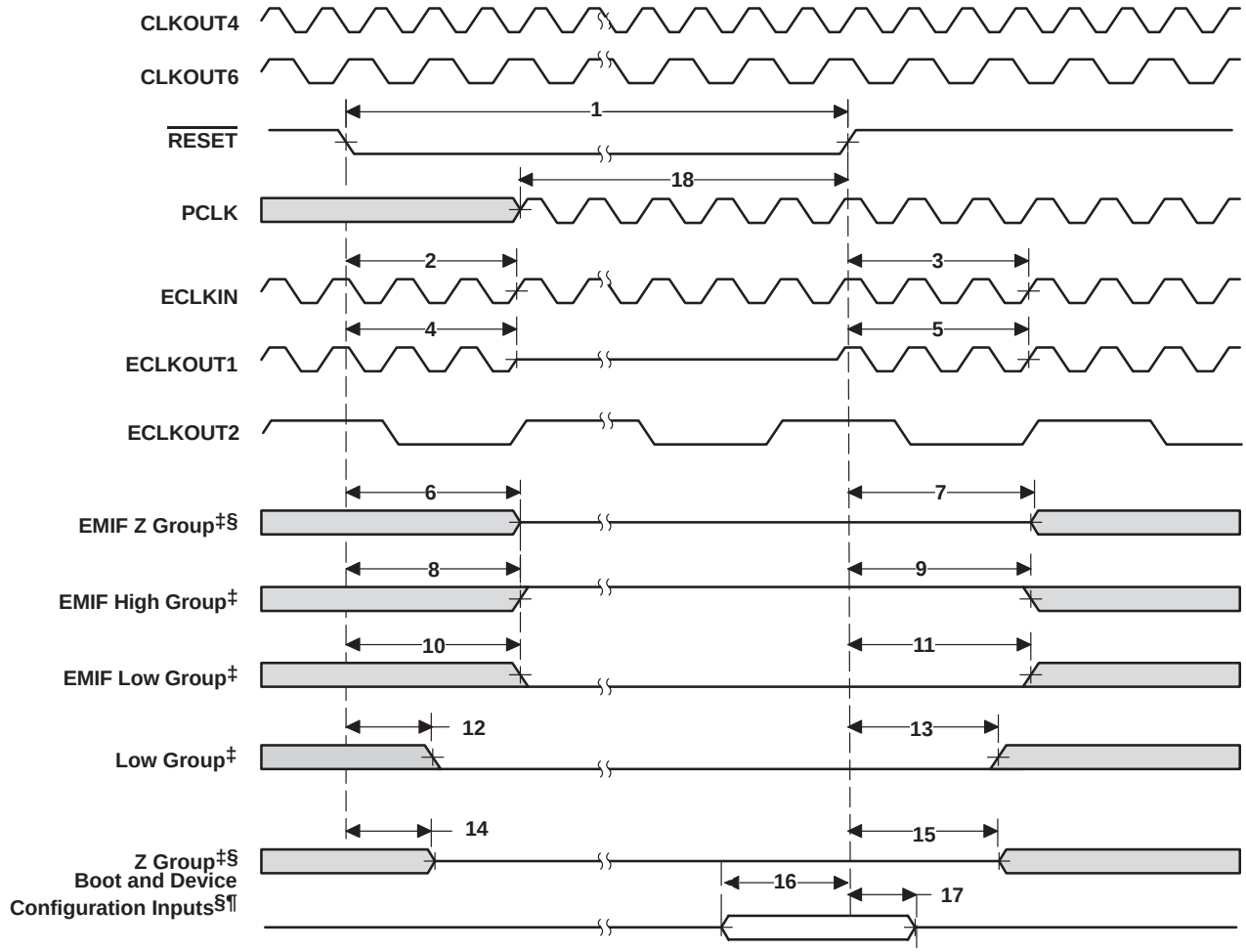
† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ This parameter applies to CLKMODE x1 when CLKIN is stable, and applies to CLKMODE x6 when CLKIN and PLL are stable.
§ This parameter applies to CLKMODE x6 only (it does not apply to CLKMODE x1). The RESET signal is not connected internally to the clock PLL circuit. The PLL, however, may need up to 250 µs to stabilize following device power up or after PLL configuration has been changed. During that time, RESET must be asserted to ensure proper device operation. See the *clock PLL* section for PLL lock times.
¶ LEND, BOOTMODE[1:0], ECLKIN_SEL[1:0], EEAI, and HD5/AD5 are the boot configuration pins during device reset.
# E = 1/AECLKIN clock frequency in ns. C = 1/CLKIN clock frequency in ns. Select whichever value is larger for the **MIN** parameter.
‖ N = the PCI input clock (PCLK) period in ns. When PCI is enabled (PCI_EN = 1), this parameter ***must*** be met.

switching characteristics over recommended operating conditions during reset†☆□ (see Figure 37)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
2	$t_{d(RSTL-ECKI)}$	Delay time, RESET low to ECLKIN synchronized internally	2E	3P + 20E	ns
3	$t_{d(RSTH-ECKI)}$	Delay time, RESET high to ECLKIN synchronized internally	2E	8P + 20E	ns
4	$t_{d(RSTL-ECKO1HZ)}$	Delay time, RESET low to ECLKOUT1 high impedance	2E		ns
5	$t_{d(RSTH-ECKO1V)}$	Delay time, RESET high to ECLKOUT1 valid		8P + 20E	ns
6	$t_{d(RSTL-EMIFZHZ)}$	Delay time, RESET low to EMIF Z high impedance	2E	3P + 4E	ns
7	$t_{d(RSTH-EMIFZV)}$	Delay time, RESET high to EMIF Z valid	16E	8P + 20E	ns
8	$t_{d(RSTL-EMIFHIV)}$	Delay time, RESET low to EMIF high group invalid	2E		ns
9	$t_{d(RSTH-EMIFHV)}$	Delay time, RESET high to EMIF high group valid		8P + 20E	ns
10	$t_{d(RSTL-EMIFLIV)}$	Delay time, RESET low to EMIF low group invalid	2E		ns
11	$t_{d(RSTH-EMIFLV)}$	Delay time, RESET high to EMIF low group valid		8P + 20E	ns
12	$t_{d(RSTL-LOWIV)}$	Delay time, RESET low to low group invalid	0		ns
13	$t_{d(RSTH-LOWV)}$	Delay time, RESET high to low group valid		11P	ns
14	$t_{d(RSTL-ZHZ)}$	Delay time, RESET low to Z group high impedance	0		ns
15	$t_{d(RSTH-ZV)}$	Delay time, RESET high to Z group valid	2P	8P	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
☆E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.
□EMIF Z group consists of: EA[22:3], ED[31:0], CE[3:0], BE[3:0], ARE/SDCAS/SADS/SRE, AWE/SDWE/SWE, and AOE/SDRAS/SOE, SOE3, SDCKE, and PDT.
EMIF high group consists of: HOLDA (when the corresponding HOLD input is high)
EMIF low group consists of: BUSREQ; HOLDA (when the corresponding HOLD input is low)
Low group consists of: XSP_CS, XSP_CLK, and XSP_DO; all of which apply only when PCI EEPROM (EEAI) is enabled (with PCI_EN = 1). Otherwise, the XSP_CLK and XSP_DO pins are in the Z group. For more details on the PCI configuration pins, see the Device Configurations section of this data sheet.
Z group consists of: HD[31:0]/AD[31:0], CLKX0, CLKX1, XSP_CLK, FSX0, FSX1, DX0, DX1, XSP_DO, CLKR0, CLKR1, FSR0, FSR1, TOUT0, TOUT1, GP[8:0], GP10/PCBE3, HR/W/PCBE2, HDS2/PCBE1, PCBE0, GP13/PINTA, GP11/PREQ, HDS1/PSERR, HCS/PPERR, HCNTL1/PDEVSEL, HAS/PPAR, HCNTL0/PSTOP, HHWIL/PTRDY (16-bit HPI mode only), HRDY/PIRDY, and HINT/PFRAME.

RESET TIMING (CONTINUED)



† EMIF Z group consists of: EA[22:3], ED[31:0], CE[3:0], BE[3:0], ARE/SDCAS/SADS/SRE, AWE/SDWE/SWE, and AOE/SDRAS/SOE, SOE3, SDCKE, and PDT.
EMIF high group consists of: HOLDA (when the corresponding HOLD input is high)
EMIF low group consists of: BUSREQ; HOLDA (when the corresponding HOLD input is low)
Low group consists of: XSP_CS, XSP_CLK, and XSP_DO; all of which apply only when PCI EEPROM (EEAI) is enabled (with PCI_EN = 1). Otherwise, the XSP_CLK and XSP_DO pins are in the Z group. For more details on the PCI configuration pins, see the Device Configurations section of this data sheet.
Z group consists of: HD[31:0]/AD[31:0], CLKX0, CLKX1, XSP_CLK, FSX0, FSX1, DX0, DX1, XSP_DO, CLKR0, CLKR1, FSR0, FSR1, TOUT0, TOUT1, GP[8:0], GP10/PCBE3, HR/W/PCBE2, HDS2/PCBE1, PCBE0, GP13/PINTA, GP11/PREQ, HDS1/PSERR, HCS/PPERR, HCNTL1/PDEVSEL, HAS/PPAR, HCNTL0/PSTOP, HHWIL/PTRDY (16-bit HPI mode only), HRDY/PIRDY, and HINT/PFRAME.

‡ If LEND, BOOTMODE[1:0], ECLKIN_SEL[1:0], EEAI, and HD5/AD5 pins are actively driven, care must be taken to ensure no timing contention between parameters 6, 7, 14, 15, 16, and 17.

§ Boot and Device Configurations Inputs (during reset) include: LEND, BOOTMODE[1:0], ECLKIN_SEL[1:0], EEAI, and HD5/AD5. The PCI_EN pin *must* be driven valid at all times and the user *must not* switch values throughout device operation.

Figure 37. Reset Timing†

EXTERNAL INTERRUPT TIMING

timing requirements for external interrupts† (see Figure 38)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{w(ILOW)}$	Width of the NMI interrupt pulse low	4P		ns
		Width of the EXT_INT interrupt pulse low	8P		ns
2	$t_{w(IHIGH)}$	Width of the NMI interrupt pulse high	4P		ns
		Width of the EXT_INT interrupt pulse high	8P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.

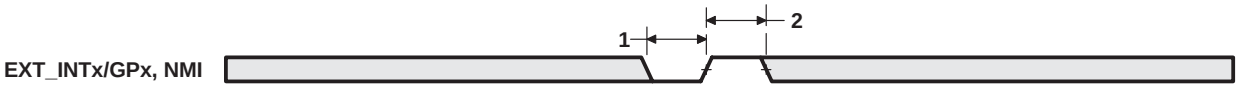


Figure 38. External/NMI Interrupt Timing

HOST-PORT INTERFACE (HPI) TIMING

timing requirements for host-port interface cycles†‡ (see Figure 39 through Figure 46)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{su(SELV-HSTBL)}$	Setup time, select signals§ valid before $\overline{\text{HSTROBE}}$ low	5		ns
2	$t_{h(HSTBL-SELV)}$	Hold time, select signals§ valid after $\overline{\text{HSTROBE}}$ low	2.4		ns
3	$t_{w(HSTBL)}$	Pulse duration, $\overline{\text{HSTROBE}}$ low	4P¶		ns
4	$t_{w(HSTBH)}$	Pulse duration, $\overline{\text{HSTROBE}}$ high between consecutive accesses	4P		ns
10	$t_{su(SELV-HASL)}$	Setup time, select signals§ valid before $\overline{\text{HAS}}$ low	5		ns
11	$t_{h(HASL-SELV)}$	Hold time, select signals§ valid after $\overline{\text{HAS}}$ low	2		ns
12	$t_{su(HDV-HSTBH)}$	Setup time, host data valid before $\overline{\text{HSTROBE}}$ high	5		ns
13	$t_{h(HSTBH-HDV)}$	Hold time, host data valid after $\overline{\text{HSTROBE}}$ high	2.8		ns
14	$t_{h(HRDYL-HSTBL)}$	Hold time, $\overline{\text{HSTROBE}}$ low after $\overline{\text{HRDY}}$ low. $\overline{\text{HSTROBE}}$ should not be inactivated until $\overline{\text{HRDY}}$ is active (low); otherwise, HPI writes will not complete properly.	2		ns
18	$t_{su(HASL-HSTBL)}$	Setup time, $\overline{\text{HAS}}$ low before $\overline{\text{HSTROBE}}$ low	2		ns
19	$t_{h(HSTBL-HASL)}$	Hold time, $\overline{\text{HAS}}$ low after $\overline{\text{HSTROBE}}$ low	2.1		ns

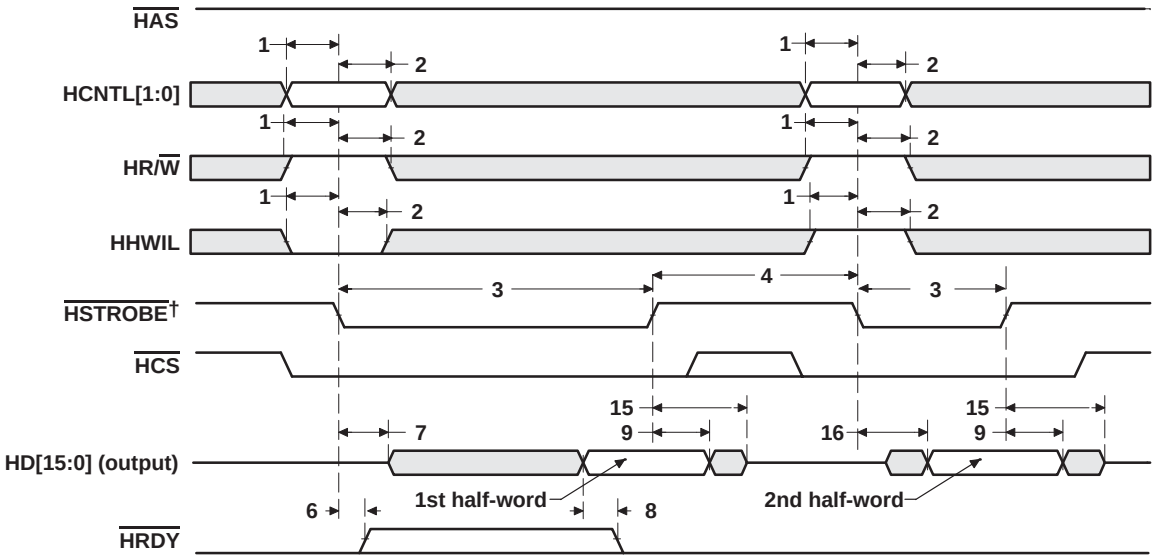
† $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.
‡ P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
§ Select signals include: HCNTL[1:0] and HR/$\overline{\text{W}}$. For HPI16 mode only, select signals also include HHWIL.
¶ Select the parameter value of 4P or 12.5 ns, whichever is greater.

switching characteristics over recommended operating conditions during host-port interface cycles†‡ (see Figure 39 through Figure 46)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
6	$t_{d(HSTBL-HRDYH)}$	Delay time, $\overline{\text{HSTROBE}}$ low to $\overline{\text{HRDY}}$ high#	1.3	4P+8	ns
7	$t_{d(HSTBL-HDLZ)}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD low impedance for an HPI read	2		ns
8	$t_{d(HDV-HRDYL)}$	Delay time, HD valid to $\overline{\text{HRDY}}$ low	−3		ns
9	$t_{oh(HSTBH-HDV)}$	Output hold time, HD valid after $\overline{\text{HSTROBE}}$ high	1.5		ns
15	$t_{d(HSTBH-HDHZ)}$	Delay time, $\overline{\text{HSTROBE}}$ high to HD high impedance		12	ns
16	$t_{d(HSTBL-HDV)}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD valid (HPI16 mode, 2nd half-word only)		4P+8	ns

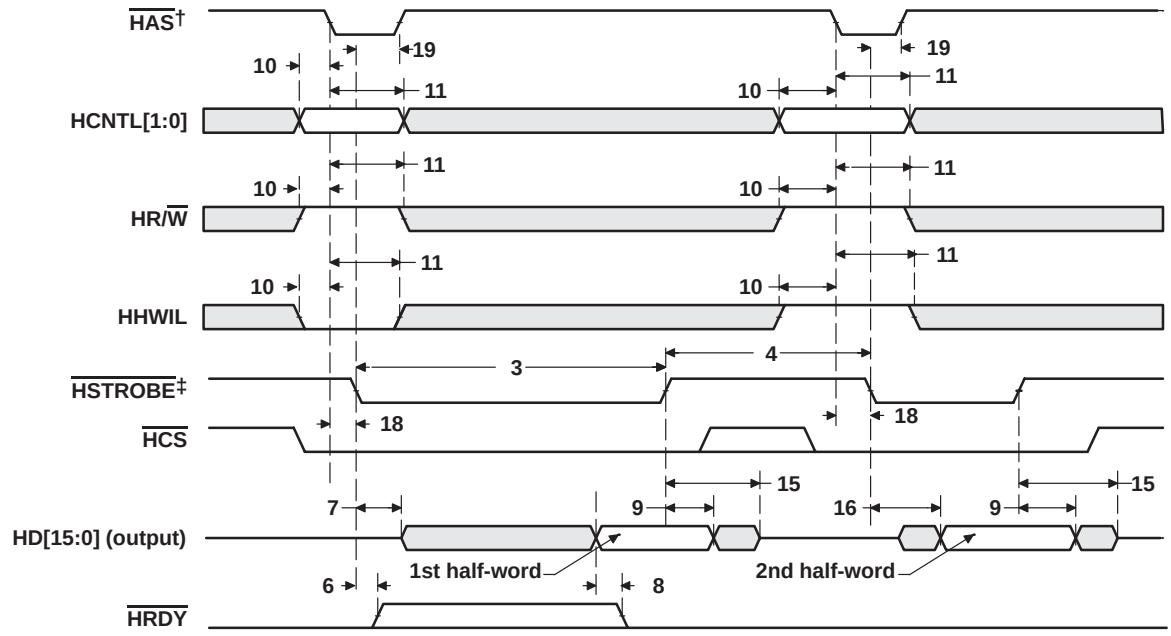
† $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.
‡ P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
# This parameter is used during HPID reads and writes. For reads, at the beginning of a word transfer (HPI32) or the first half-word transfer (HPI16) on the falling edge of $\overline{\text{HSTROBE}}$, the HPI sends the request to the EDMA internal address generation hardware, and $\overline{\text{HRDY}}$ remains high until the EDMA internal address generation hardware loads the requested data into HPID. For writes, $\overline{\text{HRDY}}$ goes high if the internal write buffer is full.

HOST-PORT INTERFACE (HPI) TIMING (CONTINUED)



† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

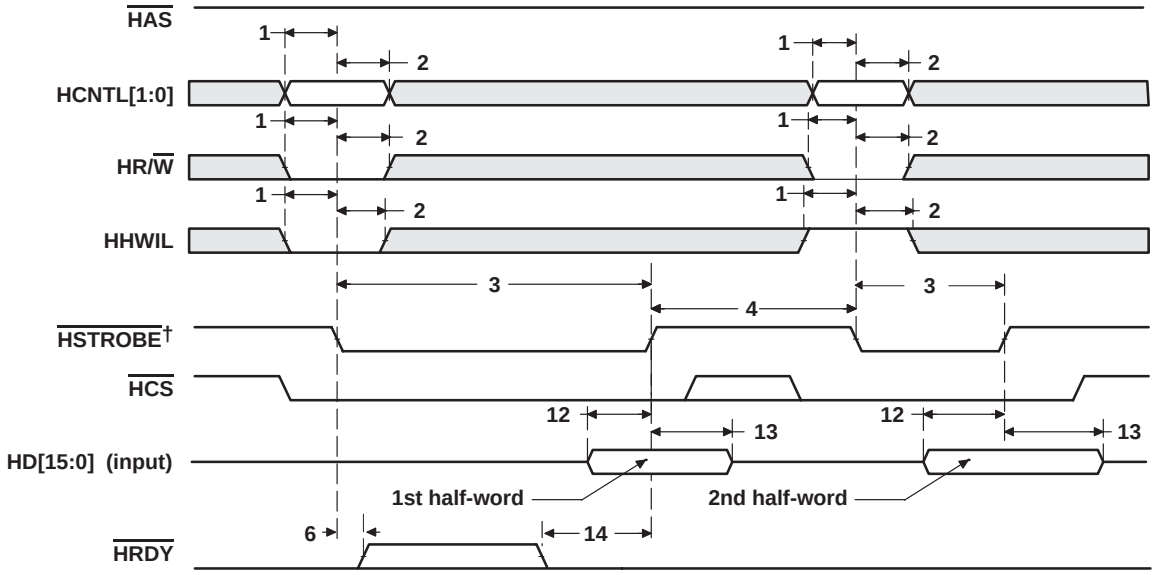
Figure 39. HPI16 Read Timing (HAS Not Used, Tied High)



† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

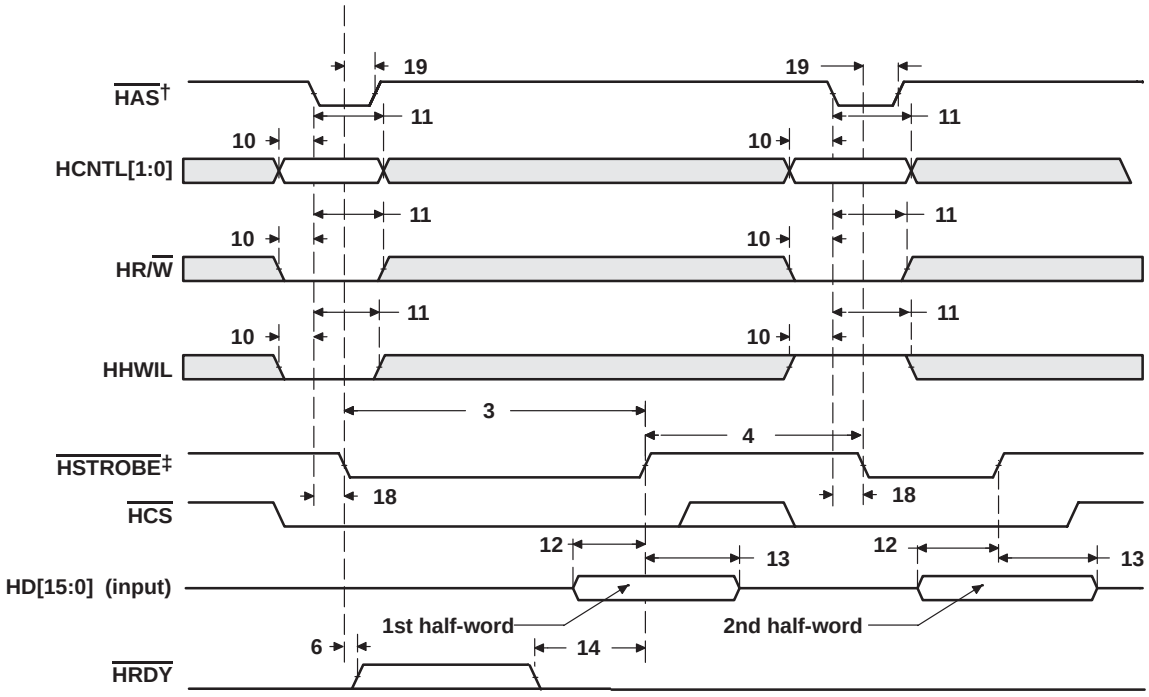
Figure 40. HPI16 Read Timing (HAS Used)

HOST-PORT INTERFACE (HPI) TIMING (CONTINUED)



† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

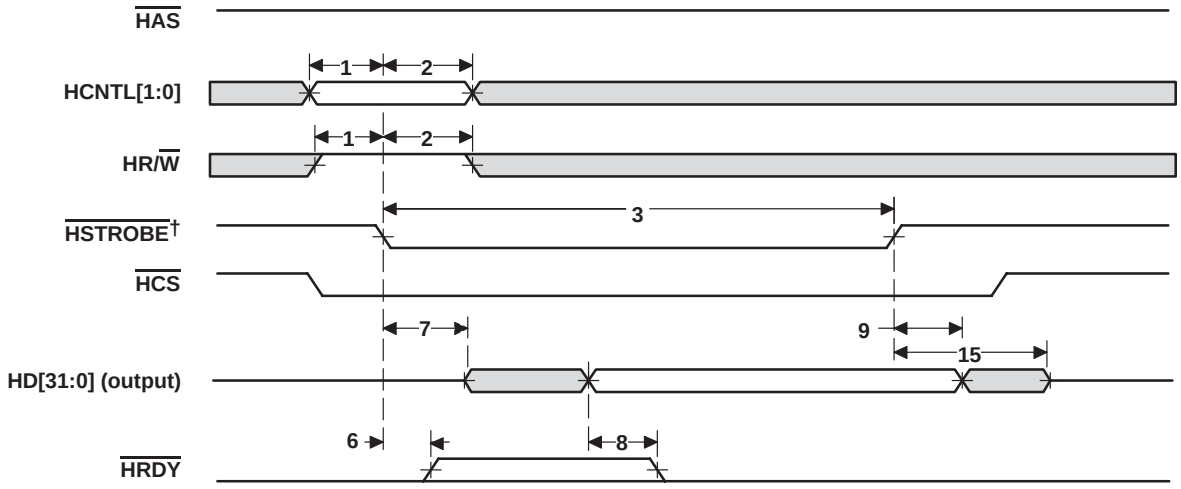
Figure 41. HPI16 Write Timing (HAS Not Used, Tied High)



† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

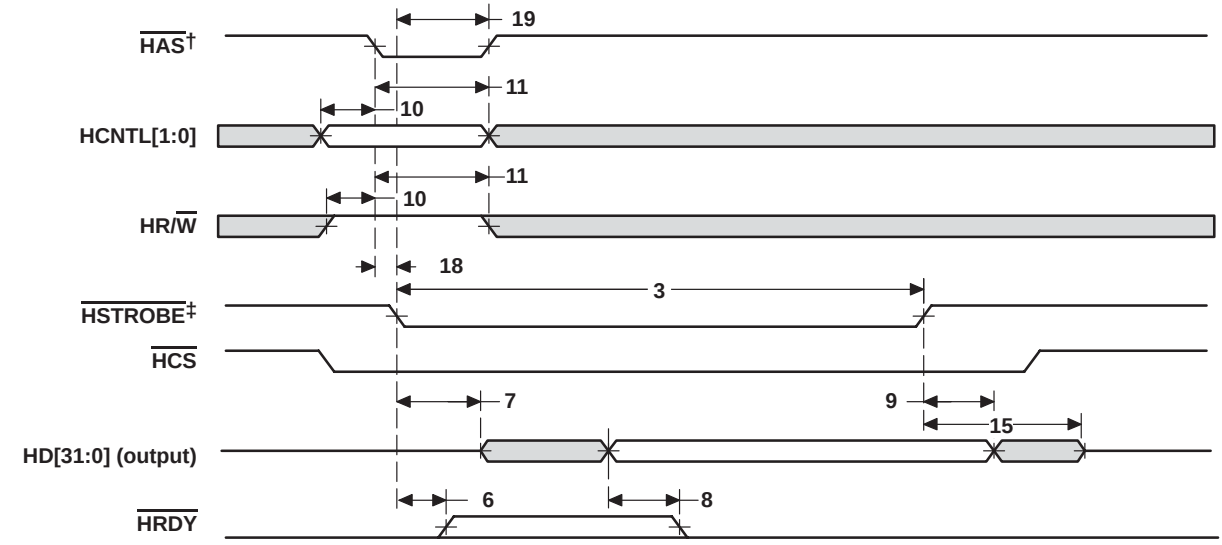
Figure 42. HPI16 Write Timing (HAS Used)

HOST-PORT INTERFACE (HPI) TIMING (CONTINUED)



† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

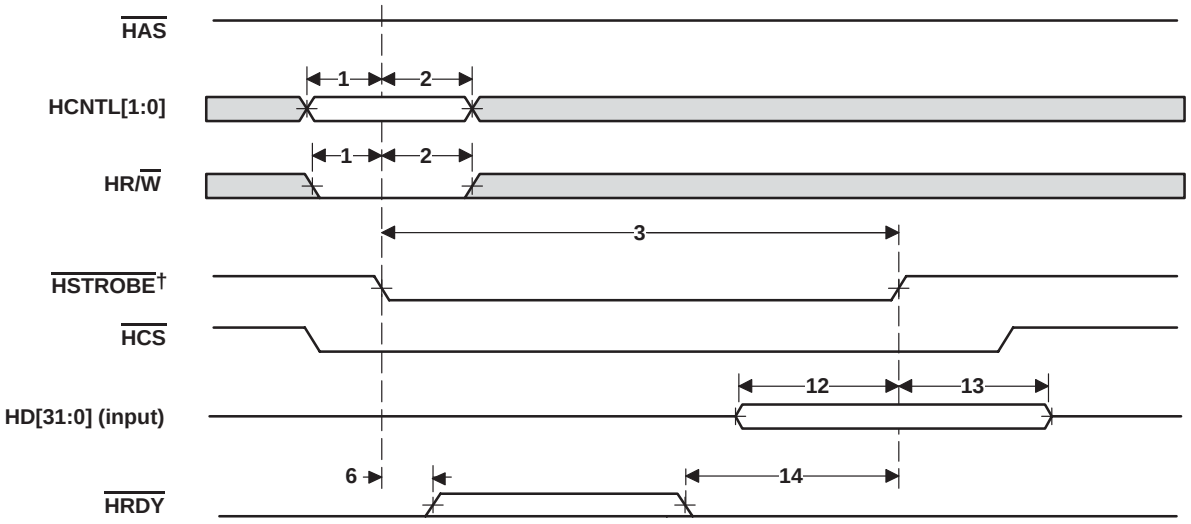
Figure 43. HPI32 Read Timing (HAS Not Used, Tied High)



† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

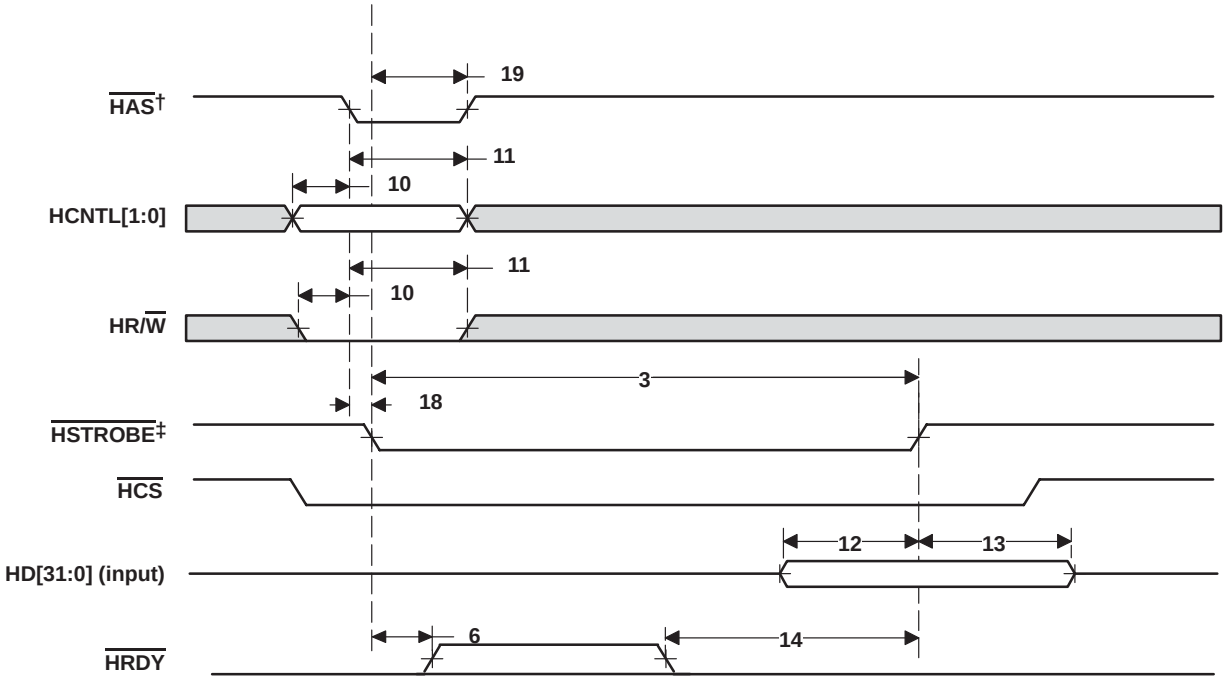
Figure 44. HPI32 Read Timing (HAS Used)

HOST-PORT INTERFACE (HPI) TIMING (CONTINUED)



† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 45. HPI32 Write Timing (HAS Not Used, Tied High)



† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 46. HPI32 Write Timing (HAS Used)

PERIPHERAL COMPONENT INTERCONNECT (PCI) TIMING

timing requirements for PCLK†‡ (see Figure 47)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{c(PCLK)}$	Cycle time, PCLK	30(or 8P§)		ns
2	$t_{w(PCLKH)}$	Pulse duration, PCLK high	11		ns
3	$t_{w(PCLKL)}$	Pulse duration, PCLK low	11		ns
4	$t_{sr(PCLK)}$	Δv/Δt slew rate, PCLK	1	4	V/ns

† For 3.3 V operation, the reference points for the rise and fall transitions are measured at V_{ILP} MAX and V_{IHP} MIN.
‡ P = 1/CPU clock frequency in ns. For example when running parts at 300 MHz, use P = 3.33 ns.
§ Select the parameter value of 30 ns or 8P, whichever is greater.

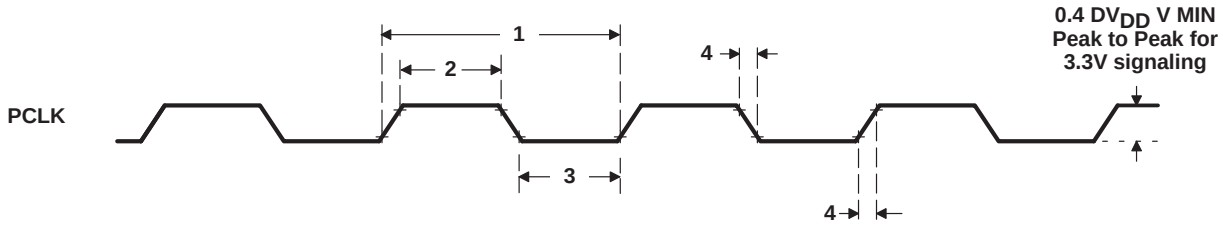


Figure 47. PCLK Timing

timing requirements for PCI reset (see Figure 48)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{w(PRST)}$	Pulse duration, $\overline{PRST}$	1		ms
2	$t_{su(PCLKA-PRSTH)}$	Setup time, PCLK active before $\overline{PRST}$ high	100		µs

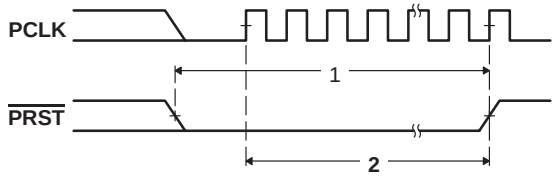


Figure 48. PCI Reset ($\overline{PRST}$) Timing

PERIPHERAL COMPONENT INTERCONNECT (PCI) TIMING (CONTINUED)

timing requirements for PCI inputs (see Figure 49)

NO.			−300		UNIT
			MIN	MAX	
5	$t_{su(IV-PCLKH)}$	Setup time, input valid before PCLK high	7		ns
6	$t_{h(IV-PCLKH)}$	Hold time, input valid after PCLK high	0		ns

switching characteristics over recommended operating conditions for PCI outputs (see Figure 49)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
1	$t_{d(PCLKH-OV)}$	Delay time, PCLK high to output valid		11	ns
2	$t_{d(PCLKH-OIV)}$	Delay time, PCLK high to output invalid	2		ns
3	$t_{d(PCLKH-OLZ)}$	Delay time, PCLK high to output low impedance	2		ns
4	$t_{d(PCLKH-OHZ)}$	Delay time, PCLK high to output high impedance		28	ns

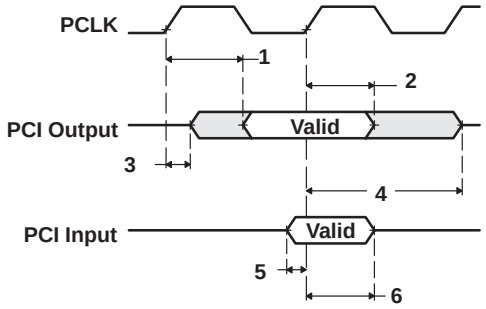


Figure 49. PCI Intput/Output Timing

timing requirements for serial EEPROM interface (see Figure 50)

NO.			−300		UNIT
			MIN	MAX	
8	$t_{su(DIV-CLKH)}$	Setup time, XSP_DI valid before XSP_CLK high	50		ns
9	$t_{h(CLKH-DIV)}$	Hold time, XSP_DI valid after XSP_CLK high	0		ns

switching characteristics over recommended operating conditions for serial EEPROM interface† (see Figure 50)

NO.	PARAMETER		−300			UNIT
			MIN	NOM	MAX	
1	$t_{w(CSL)}$	Pulse duration, XSP_CS low		4092P		ns
2	$t_{d(CLKL-CSL)}$	Delay time, XSP_CLK low to XSP_CS low		0		ns
3	$t_{d(CSH-CLKH)}$	Delay time, XSP_CS high to XSP_CLK high		2046P		ns
4	$t_{w(CLKH)}$	Pulse duration, XSP_CLK high		2046P		ns
5	$t_{w(CLKL)}$	Pulse duration, XSP_CLK low		2046P		ns
6	$t_{osu(DOV-CLKH)}$	Output setup time, XSP_DO valid before XSP_CLK high		2046P		ns
7	$t_{oh(CLKH-DOV)}$	Output hold time, XSP_DO valid after XSP_CLK high		2046P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.

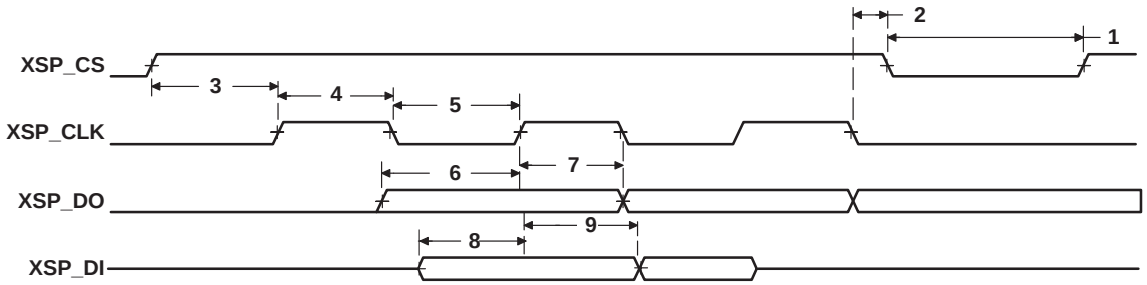


Figure 50. PCI Serial EEPROM Interface Timing

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING

timing requirements for McBSP† (see Figure 51)

NO.				−300		UNIT
				MIN	MAX	
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X ext	4P or 6.67‡§¶		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	$0.5t_{c(CKRX)}$ − 1#		ns
5	$t_{su(FRH-CKRL)}$	Setup time, external FSR high before CLKR low	CLKR int	9		ns
			CLKR ext	1.3		
6	$t_{h(CKRL-FRH)}$	Hold time, external FSR high after CLKR low	CLKR int	6		ns
			CLKR ext	3		
7	$t_{su(DRV-CKRL)}$	Setup time, DR valid before CLKR low	CLKR int	8		ns
			CLKR ext	0.9		
8	$t_{h(CKRL-DRV)}$	Hold time, DR valid after CLKR low	CLKR int	3		ns
			CLKR ext	3.1		
10	$t_{su(FXH-CKXL)}$	Setup time, external FSX high before CLKX low	CLKX int	9		ns
			CLKX ext	1.3		
11	$t_{h(CKXL-FXH)}$	Hold time, external FSX high after CLKX low	CLKX int	6		ns
			CLKX ext	3		

† CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
‡ Minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.
§ P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.3 ns.
¶ Use whichever value is greater.
# This parameter applies to the maximum McBSP frequency. Operate serial clocks (CLKR/X) in the resonable range of 40/60 duty cycle.

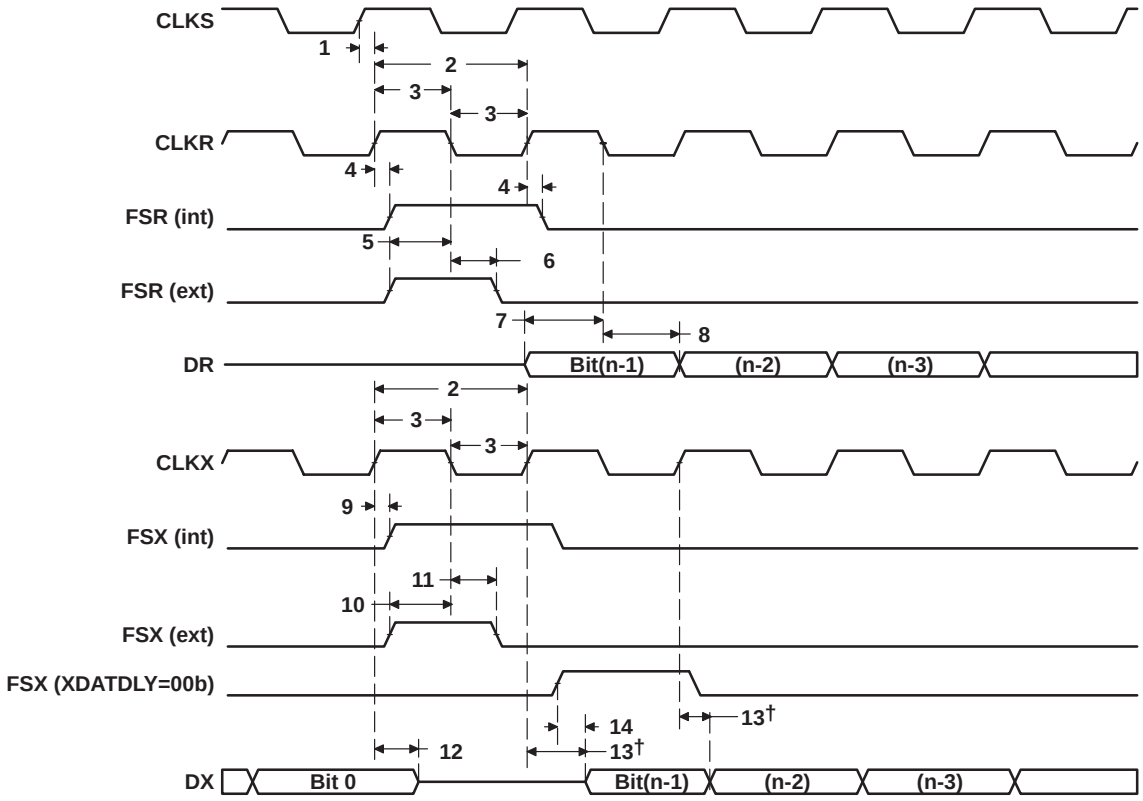
MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

switching characteristics over recommended operating conditions for McBSP†‡ (see Figure 51)

NO.	PARAMETER			−300 MIN	−300 MAX	UNIT
1	$t_{d(CKSH-CKRXH)}$	Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input		1.4	10	ns
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X int	4P or 6.67§¶#		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X int	C − 1‖	C + 1‖	ns
4	$t_{d(CKRH-FRV)}$	Delay time, CLKR high to internal FSR valid	CLKR int	−2.1	3	ns
9	$t_{d(CKXH-FXV)}$	Delay time, CLKX high to internal FSX valid	CLKX int	−1.7	3	ns
			CLKX ext	1.7	9	
12	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	CLKX int	−3.9	4	ns
			CLKX ext	2.0	9	
13	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	CLKX int	−3.9 + D1☆	4 + D2☆	ns
			CLKX ext	2.0 + D1☆	9 + D2☆	
14	$t_{d(FXH-DXV)}$	Delay time, FSX high to DX valid ONLY applies when in data delay 0 (XDATDLY = 00b) mode	FSX int	−2.3 + D1□	5.6 + D2□	ns
			FSX ext	1.9 + D1□	9 + D2□	

† CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
‡ Minimum delay times also represent minimum output hold times.
§ Minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.
¶ P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.3 ns.
# Use whichever value is greater.
‖ C = H or L
S = sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
CLKGDV should be set appropriately to ensure the McBSP bit rate does not exceed the maximum limit (see ¶ footnote above).
☆ Extra delay from CLKX high to DX valid applies *only* to the first data bit of a device, if and only if DXENA = 1 in SPCR.
if DXENA = 0, then D1 = D2 = 0
if DXENA = 1, then D1 = 4P, D2 = 8P
□ Extra delay from FSX high to DX valid **applies *only* to the first data bit of a device**, if and only if DXENA = 1 in SPCR.
if DXENA = 0, then D1 = D2 = 0
if DXENA = 1, then D1 = 4P, D2 = 8P

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)



† Parameter No. 13 applies to the first data bit *only* when XDATDLY ≠ 0

Figure 51. McBSP Timing

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

timing requirements for FSR when GSYNC = 1 (see Figure 52)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{su(FRH-CKSH)}$	Setup time, FSR high before CLKS high	4		ns
2	$t_{h(CKSH-FRH)}$	Hold time, FSR high after CLKS high	4		ns

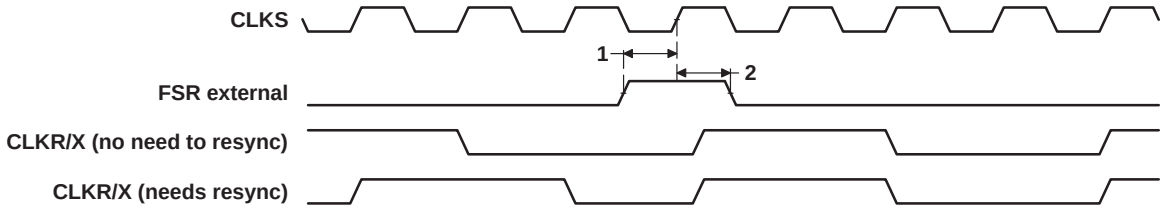


Figure 52. FSR Timing When GSYNC = 1

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

timing requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0†‡ (see Figure 53)

NO.			−300				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXL)}$	Setup time, DR valid before CLKX low	12		2 − 12P		ns
5	$t_{h(CKXL-DRV)}$	Hold time, DR valid after CLKX low	4		5 + 24P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics over recommended operating conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0†‡ (see Figure 53)

NO.	PARAMETER		−300				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low¶	T − 2	T + 3			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high#	L − 2	L + 3			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	−2	4	12P+2.8	20P+17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	L − 2	L + 3			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high			4P+3	12P+17	ns
8	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid			8P+1.8	16P+17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
¶ FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
# FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

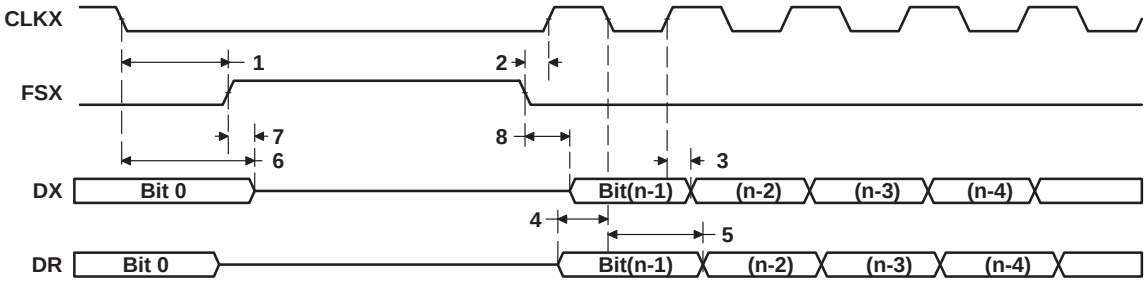


Figure 53. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

timing requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0†‡ (see Figure 54)

NO.			−300				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics over recommended operating conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0†‡ (see Figure 54)

NO.	PARAMETER		−300				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low¶	L − 2	L + 3			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high#	T − 2	T + 3			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	−2	4	12P+4	20P+17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	−2	4	12P+3	20P+17	ns
7	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	H − 2	H + 4	8P+2	16P+17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
¶ FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
# FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

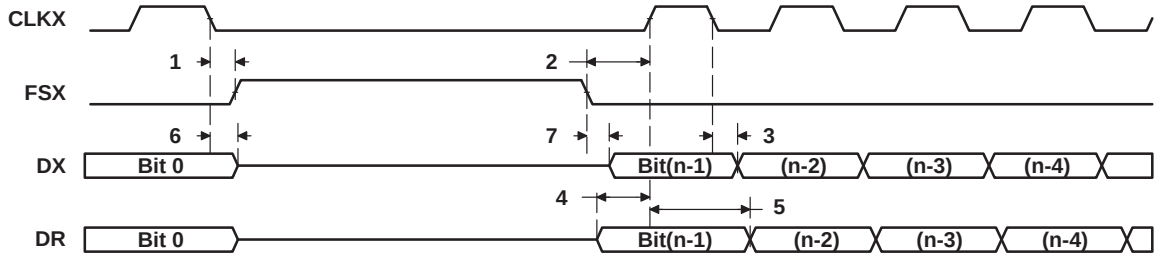


Figure 54. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

timing requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1†‡ (see Figure 55)

NO.			−300				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics over recommended operating conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1†‡ (see Figure 55)

NO.	PARAMETER		−300				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high¶	T − 2	T + 3			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low#	H − 2	H + 3			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	−2	4	12P + 4	20P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	H − 2	H + 3			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high			4P + 3	12P + 17	ns
8	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid			8P + 2	16P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
¶ FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
# FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

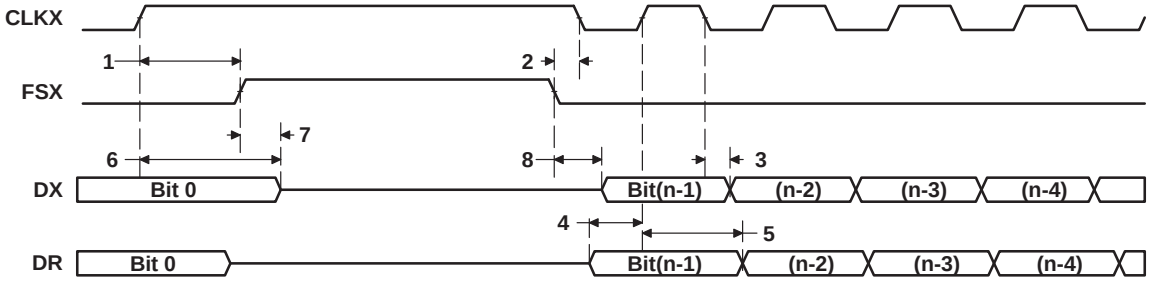


Figure 55. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

timing requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1†‡ (see Figure 56)

NO.			−300				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics over recommended operating conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1†‡ (see Figure 56)

NO.	PARAMETER		−300				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high¶	H − 2	H + 3			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low#	T − 2	T + 1			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	−2	4	12P + 4	20P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	−2	4	12P + 3	20P + 17	ns
7	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	L − 2	L + 4	8P + 2	16P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
¶ FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
# FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING (CONTINUED)

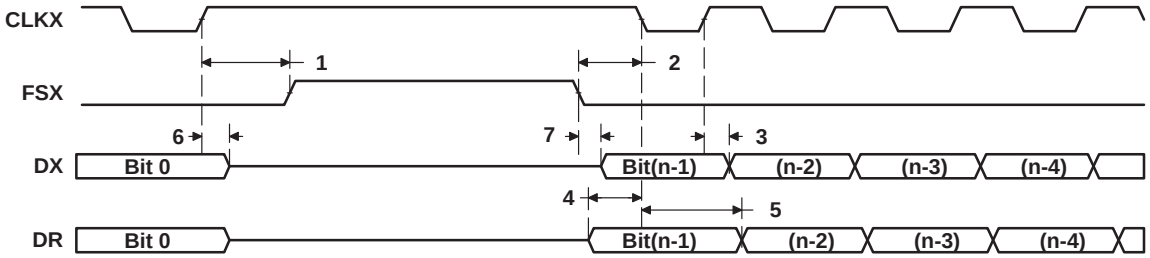


Figure 56. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

TIMER TIMING

timing requirements for timer inputs† (see Figure 57)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{w(TINPH)}$	Pulse duration, TINP high	8P		ns
2	$t_{w(TINPL)}$	Pulse duration, TINP low	8P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.

switching characteristics over recommended operating conditions for timer outputs† (see Figure 57)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
3	$t_{w(TOUTH)}$	Pulse duration, TOUT high	8P−3		ns
4	$t_{w(TOUTL)}$	Pulse duration, TOUT low	8P−3		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.

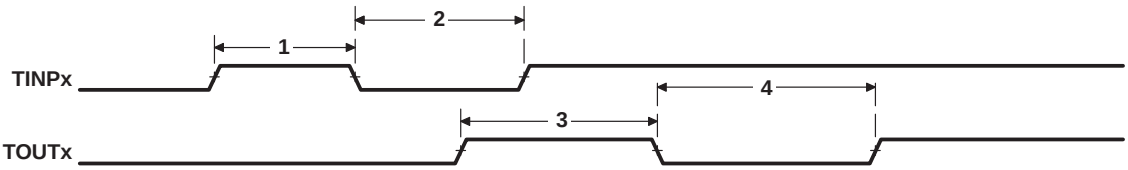


Figure 57. Timer Timing

GENERAL-PURPOSE INPUT/OUTPUT (GPIO) PORT TIMING

timing requirements for GPIO inputs†‡ (see Figure 58)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{w(GPIH)}$	Pulse duration, GPIx high	8P		ns
2	$t_{w(GPIL)}$	Pulse duration, GPIx low	8P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ The pulse width given is sufficient to generate a CPU interrupt or an EDMA event. However, if a user wants to have the DSP recognize the GPIx changes through software polling of the GPIO register, the GPIx duration must be extended to at least 12P to allow the DSP enough time to access the GPIO register through the CFGBUS.

switching characteristics over recommended operating conditions for GPIO outputs† (see Figure 58)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
3	$t_{w(GPOH)}$	Pulse duration, GPOx high	24P − 8‡		ns
4	$t_{w(GPOL)}$	Pulse duration, GPOx low	24P − 8‡		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ This parameter value should not be used as a maximum performance specification. Actual performance of back-to-back accesses of the GPIO is dependent upon internal bus activity.

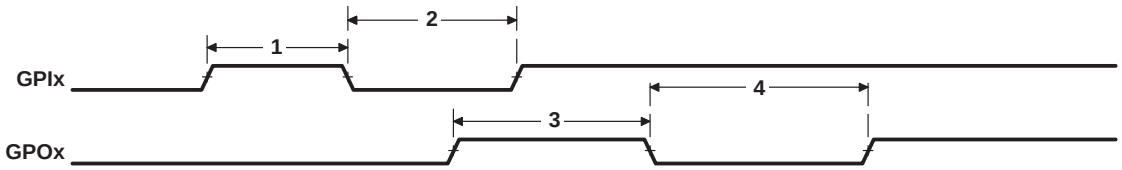


Figure 58. GPIO Port Timing

JTAG TEST-PORT TIMING

timing requirements for JTAG test port (see Figure 59)

NO.			−300		UNIT
			MIN	MAX	
1	$t_{c(TCK)}$	Cycle time, TCK	35		ns
3	$t_{su(TDIV-TCKH)}$	Setup time, TDI/TMS/$\overline{TRST}$ valid before TCK high	10		ns
4	$t_{h(TCKH-TDIV)}$	Hold time, TDI/TMS/$\overline{TRST}$ valid after TCK high	9		ns

switching characteristics over recommended operating conditions for JTAG test port (see Figure 59)

NO.	PARAMETER		−300		UNIT
			MIN	MAX	
2	$t_{d(TCKL-TDOV)}$	Delay time, TCK low to TDO valid	0	18	ns

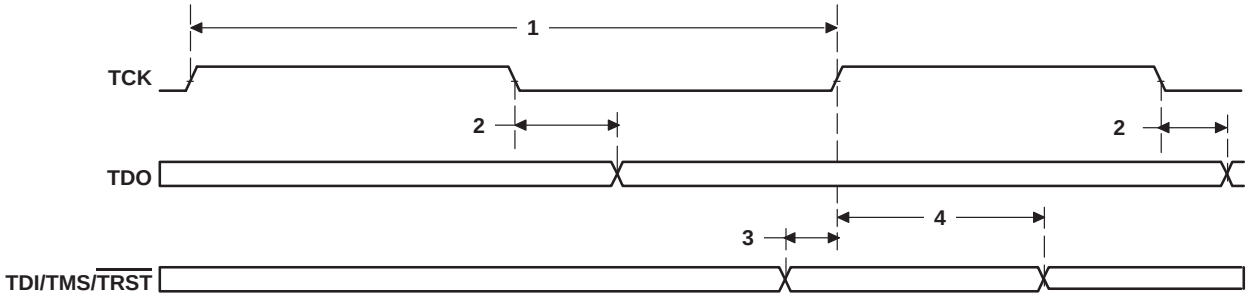


Figure 59. JTAG Test-Port Timing

MECHANICAL DATA

The following table(s) show the thermal resistance characteristics for the PBGA — GLZ, ZLZ and CLZ mechanical packages.

thermal resistance characteristics (S-PBGA package) [GLZ]

NO.			°C/W	Air Flow (m/s†)
1	$R\Theta_{JC}$	Junction-to-case	1.55	N/A
2	$R\Theta_{JB}$	Junction-to-board	9.1	N/A
3	$R\Theta_{JA}$	Junction-to-free air	17.9	0.00
4	$R\Theta_{JA}$	Junction-to-free air	15.02	0.5
5	$R\Theta_{JA}$	Junction-to-free air	13.4	1.0
6	$R\Theta_{JA}$	Junction-to-free air	11.89	2.00
7	Psi_{JT}	Junction-to-package top	0.5	N/A
8	Psi_{JB}	Junction-to-board	7.4	N/A

† m/s = meters per second

thermal resistance characteristics (S-PBGA package) [ZLZ]

NO.			°C/W	Air Flow (m/s†)
1	$R\Theta_{JC}$	Junction-to-case	1.55	N/A
2	$R\Theta_{JB}$	Junction-to-board	9.1	N/A
3	$R\Theta_{JA}$	Junction-to-free air	17.9	0.00
4	$R\Theta_{JA}$	Junction-to-free air	15.02	0.5
5	$R\Theta_{JA}$	Junction-to-free air	13.4	1.0
6	$R\Theta_{JA}$	Junction-to-free air	11.89	2.00
7	Psi_{JT}	Junction-to-package top	0.5	N/A
8	Psi_{JB}	Junction-to-board	7.4	N/A

† m/s = meters per second

The following mechanical package diagram(s) reflect the most up-to-date mechanical data released for these designated device(s).

thermal resistance characteristics (S-PBGA package) [CLZ]

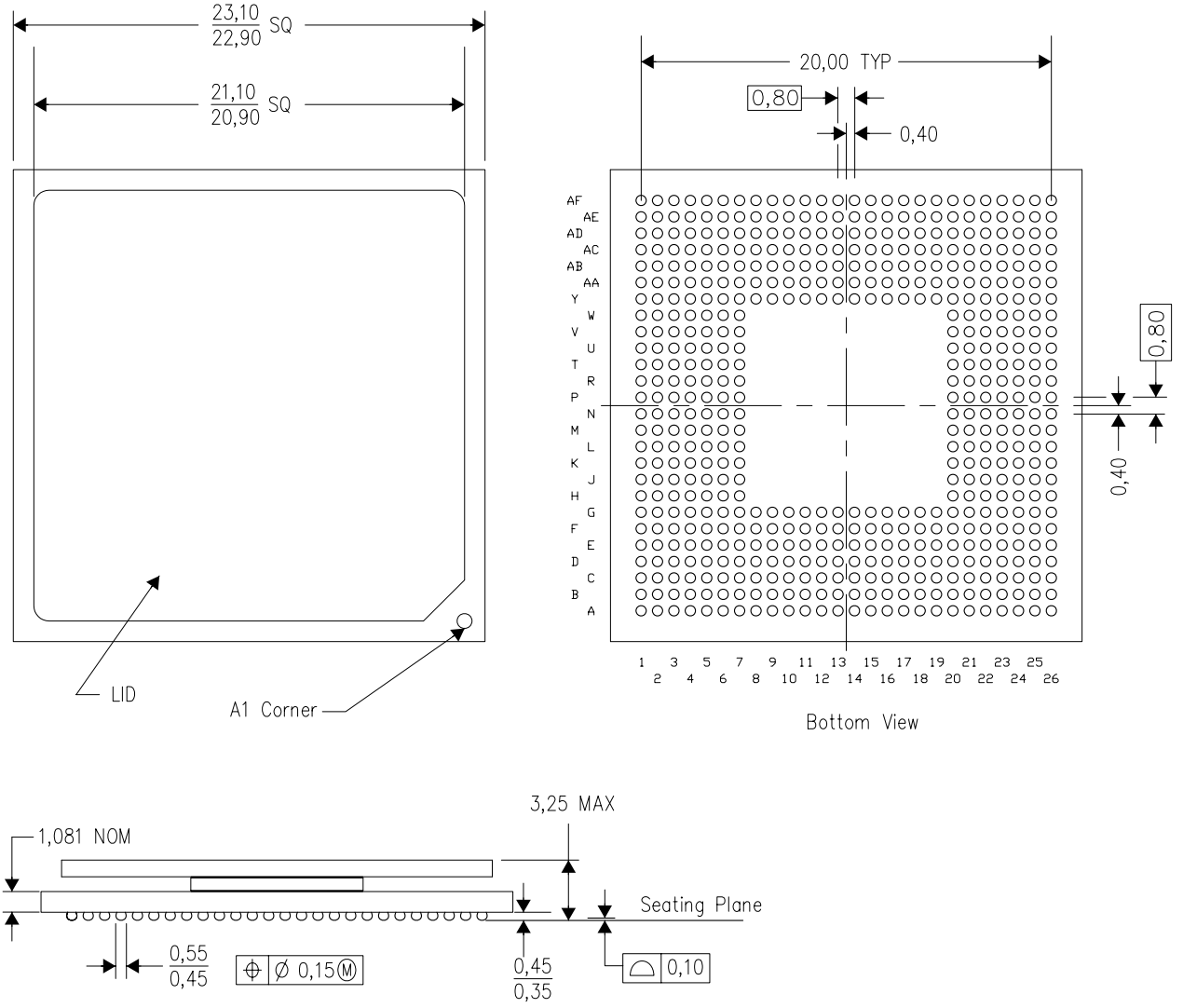
NO.			°C/W	Air Flow (m/s†)
1	$R\Theta_{JC}$	Junction-to-case	1.55	N/A
2	$R\Theta_{JB}$	Junction-to-board	9.1	N/A
3	$R\Theta_{JA}$	Junction-to-free air	17.9	0.00
4	$R\Theta_{JA}$	Junction-to-free air	15.02	0.5
5	$R\Theta_{JA}$	Junction-to-free air	13.4	1.0
6	$R\Theta_{JA}$	Junction-to-free air	11.89	2.00
7	Psi_{JT}	Junction-to-package top	0.5	N/A
8	Psi_{JB}	Junction-to-board	7.4	N/A

† m/s = meters per second

packaging information

The following packaging information and addendum reflect the most current released data available for the designated device(s). This data is subject to change without notice and without revision of this document.

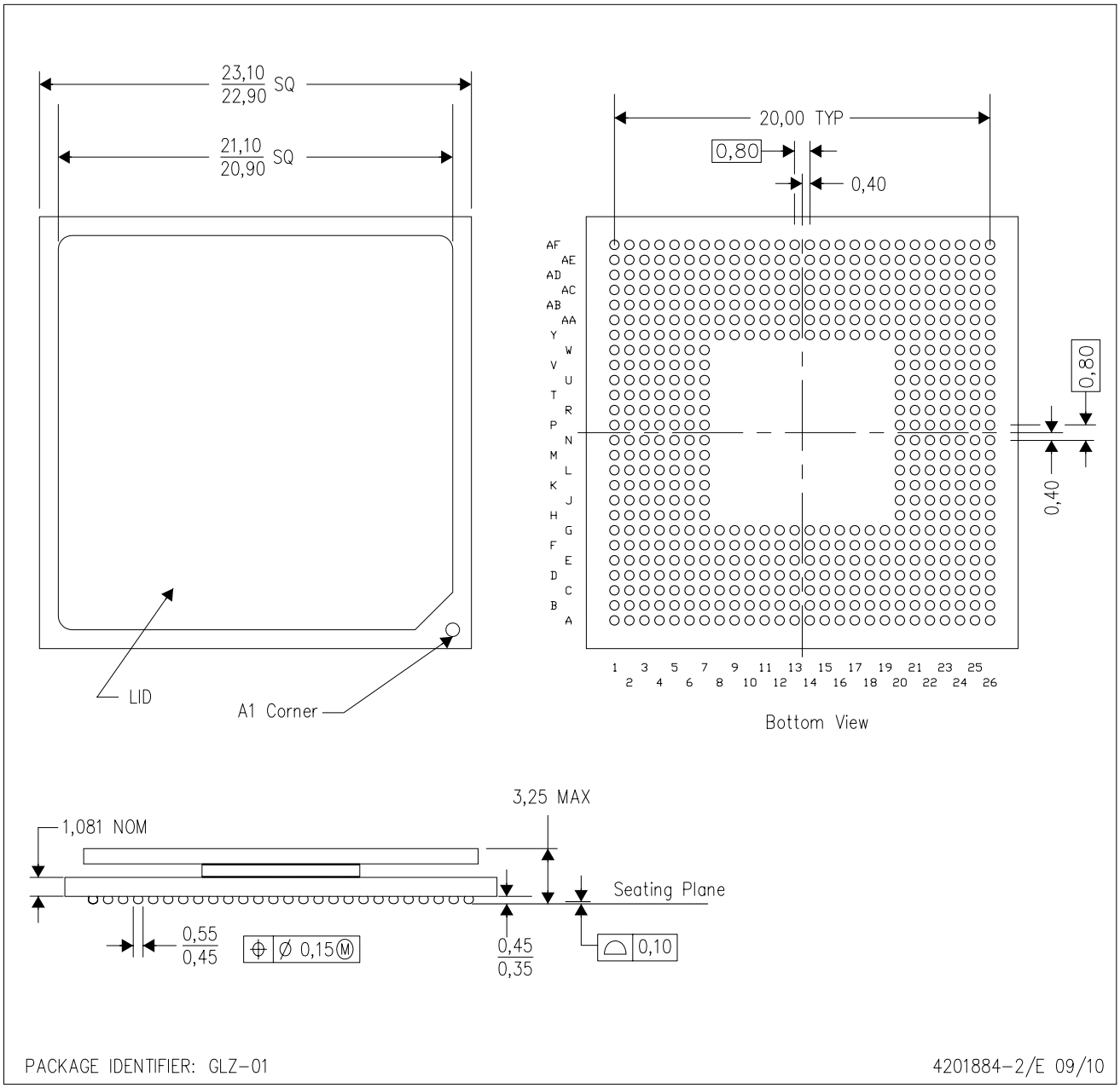
CLZ (S–PBGA–N532) PLASTIC BALL GRID ARRAY



PACKAGE IDENTIFIER: CLZ–01 4206995–2/B 09/10

NOTES:
A. All linear dimensions are in millimeters. Dimensioning and tolerancing per ASME Y14.5M–1994.
B. This drawing is subject to change without notice.
C. Thermally enhanced plastic package with a lid.
D. Flip chip application only.
E. Pb–free die bump and solder ball.

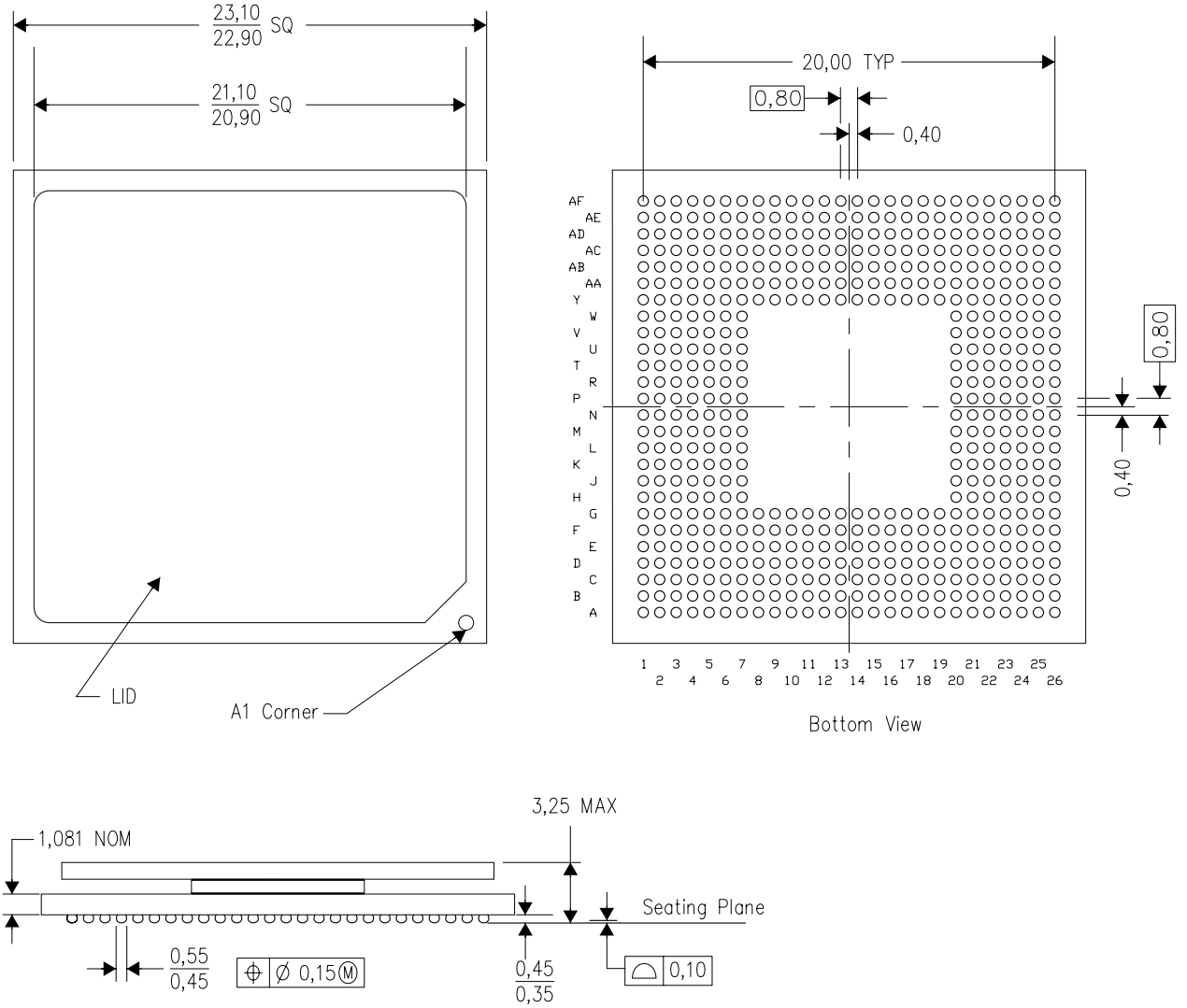
GLZ (S–PBGA–N532) PLASTIC BALL GRID ARRAY



PACKAGE IDENTIFIER: GLZ–01 4201884–2/E 09/10

NOTES:
A. All linear dimensions are in millimeters. Dimensioning and tolerancing per ASME Y14.5M–1994.
B. This drawing is subject to change without notice.
C. Thermally enhanced plastic package with a lid.
D. Flip chip application only.

ZLZ (S–PBGA–N532) PLASTIC BALL GRID ARRAY



PACKAGE IDENTIFIER: ZLZ–01 4205952–2/B 10/10

NOTES:
A. All linear dimensions are in millimeters. Dimensioning and tolerancing per ASME Y14.5M–1994.
B. This drawing is subject to change without notice.
C. Thermally enhanced plastic package with a lid.
D. Flip chip application only.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Audio	www.ti.com/audio	Communications and Telecom	www.ti.com/communications
Amplifiers	amplifier.ti.com	Computers and Peripherals	www.ti.com/computers
Data Converters	dataconverter.ti.com	Consumer Electronics	www.ti.com/consumer-apps
DLP® Products	www.dlp.com	Energy and Lighting	www.ti.com/energy
DSP	dsp.ti.com	Industrial	www.ti.com/industrial
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Security	www.ti.com/security
Logic	logic.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Power Mgmt	power.ti.com	Transportation and Automotive	www.ti.com/automotive
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com	Wireless	www.ti.com/wireless-apps
RF/IF and ZigBee® Solutions	www.ti.com/lprf		
	TI E2E Community Home Page		e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2011, Texas Instruments Incorporated