# THE DATASHEET OF

TMS320C6205GWT200

www.win-source.net 0086-755-83957316

- High-Performance Fixed-Point Digital Signal Processor (DSP) − TMS320C6205
  - 5-ns Instruction Cycle Time
  - 200-MHz Clock Rate
  - Eight 32-Bit Instructions/Cycle
  - 1600 MIPS
- VelociTI™ Advanced-Very-Long-Instruction-Word (VLIW) TMS320C62x™ DSP Core
  - Eight Highly Independent Functional Units:
    - Six ALUs (32-/40-Bit)
    - Two 16-Bit Multipliers (32-Bit Result)
  - Load-Store Architecture With 32 32-Bit General-Purpose Registers
  - Instruction Packing Reduces Code Size
  - All Instructions Conditional
- Instruction Set Features
  - Byte-Addressable (8-, 16-, 32-Bit Data)
  - 8-Bit Overflow Protection
  - Saturation
  - Bit-Field Extract, Set, Clear
  - Bit-Counting
  - Normalization
- 1M-Bit On-Chip SRAM
  - 512K-Bit Internal Program/Cache (16K 32-Bit Instructions)
  - 512K-Bit Dual-Access Internal Data (64K Bytes)
    - Organized as Two 32K-Byte Blocks for Improved Concurrency
- 32-Bit External Memory Interface (EMIF)
  - Glueless Interface to Synchronous Memories: SDRAM or SBSRAM
  - Glueless Interface to Asynchronous Memories: SRAM and EPROM
  - 52M-Byte Addressable External Memory Space
- Four-Channel Bootloading Direct-Memory-Access (DMA) Controller With an Auxiliary Channel
- Flexible Phase-Locked-Loop (PLL) Clock Generator
- 32-Bit/33-MHz Peripheral Component Interconnect (PCI) Master/Slave Interface Conforms to:
  - PCI Specification 2.2
  - Power Management Interface 1.1

  Meets Requirements of PC99
  - PCI Access to All On-Chip RAM, Peripherals, and External Memory (via EMIF)
  - Four 8-Deep x 32-Wide FIFOs for Efficient PCI Bus Data Transfer
  - 3.3/5-V PCI Operation
  - Three PCI Bus Address Registers:
    - Prefetchable Memory
    - Non-Prefetchable Memory I/O
  - Supports 4-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™)
- Two 32-Bit General-Purpose Timers
- IEEE-1149.1 (JTAG†) Boundary-Scan-Compatible
- 288-Pin MicroStar BGA™ Package (GHK and ZHK Suffixes)
- 0.15-μm/5-Level Metal Process
  - CMOS Technology
- 3.3-V I/Os, 1.5-V Internal, 5-V Voltage Tolerance for PCI I/O Pins



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.

VelociTI, TMS320C62x, and MicroStar BGA 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 © 2006, Texas Instruments Incorporated

**Table of Contents**

revision history . . . . . 3
GHK and ZHK BGA packages (bottom view) . . . . . 4
description . . . . . 5
device characteristics . . . . . 6
functional and CPU (DSP core) block diagram . . . . . 7
CPU (DSP core) description . . . . . 8
memory map summary . . . . . 10
signal groups description . . . . . 11
signal descriptions . . . . . 14
development support . . . . . 23
documentation support . . . . . 26
clock PLL . . . . . 27
power-down mode logic . . . . . 29
power-supply sequencing . . . . . 32
absolute maximum ratings over operating case temperature range . . . . . 33
recommended operating conditions . . . . . 33
recommended operating conditions (PCI only) . . . . . 33
electrical characteristics over recommended rangesof supply voltage and operating case temperature . . . . . 34
electrical characteristics over recommended ranges of supply voltage and operating case temperature (PCI only) . . . . . 34
parameter measurement information . . . . . 35
input and output clocks . . . . . 36
asynchronous memory timing . . . . . 38
synchronous-burst memory timing . . . . . 41
synchronous DRAM timing . . . . . 43
$\overline{\text{HOLD}}$/$\overline{\text{HOLDA}}$ timing . . . . . 47
reset timing . . . . . 48
external interrupt timing . . . . . 50
PCI I/O timings . . . . . 51
PCI reset timing . . . . . 52
PCI serial EEPROM interface timing . . . . . 53
multichannel buffered serial port timing . . . . . 54
DMAC, timer, power-down timing . . . . . 64
JTAG test-port timing . . . . . 66
mechanical data . . . . . 67

## REVISION HISTORY

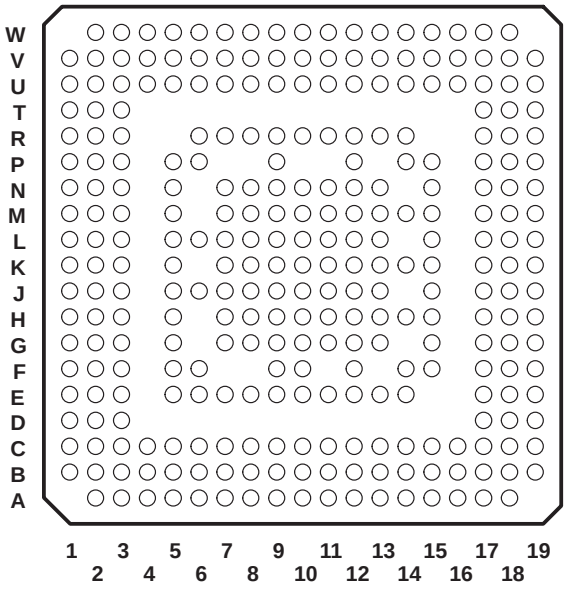
This data sheet revision history highlights the technical changes made to the SPR106E device-specific data sheet to make it an SPRS106F revision. It also highlights technical changes made to SPRS219F to generate SPRS219G . These changes are marked by **[Revision G]** in the Revision History below.

**Scope:** Applicable updates to the C62x device family, specifically relating to the C6205 device, have been incorporated.

| PAGE(S) NO. | ADDITIONS/CHANGES/DELETIONS |
|---|---|
| | Added information for the ZHK Mechanical Package **[Revision G]**<br>Moved Revision History to front of document **[Revision G]** |
| 6 | Device Characteristics, Characteristics of the C6205 Processor table:<br>Hardware Features, Peripherals:<br>Updated description for PCI |
| 24 | device and development-support tool nomenclature section:<br>Updated paragraphs and Figure **[Revision G]** |
| 28 | Table 4, C6205 PLL Component Selection Table, Typical Lock Time (µs) section:<br>Changed “75 MS” to “75 µs” **[Revision G]** |
| 67−68 | Added “Mechanical Data” title and paragraph<br>Added Package Information section **[Revision G]** |

## GHK and ZHK BGA packages (bottom view)

**GHK and ZHK 288-PIN BALL GRID ARRAY (BGA) PACKAGES**
**(BOTTOM VIEW)**



## description

The TMS320C62x™ DSPs (including the TMS320C6205 device) compose the fixed-point DSP generation in the TMS320C6000™ DSP platform. The TMS320C6205 (C6205) device is based on the high-performance, advanced VelociTI™ very-long-instruction-word (VLIW) architecture developed by Texas Instruments (TI), making the C6205 an excellent choice for multichannel and multifunction applications.

With performance of up to 1600 million instructions per second (MIPS) at a clock rate of 200 MHz, the C6205 offers cost-effective solutions to high-performance DSP-programming challenges. The C6205 DSP possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. This processor has 32 general-purpose registers of 32-bit word length and eight highly independent functional units. The eight functional units provide six arithmetic logic units (ALUs) for a high degree of parallelism and two 16-bit multipliers for a 32-bit result. The C6205 can produce two multiply-accumulates (MACs) per cycle for a total of 400 million MACs per second (MMACS). The C6205 DSP also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals.

The C6205 includes a large bank of on-chip memory and has a powerful and diverse set of peripherals. Program memory consists of a 64K-byte block that is user-configurable as cache or memory-mapped program space. Data memory consists of two 32K-byte blocks of RAM. The peripheral set includes two multichannel buffered serial ports (McBSPs), two general-purpose timers, a peripheral component interconnect (PCI) module that supports 33-MHz master/slave interface and 4-wire serial EEPROM interface, and a glueless external memory interface (EMIF) capable of interfacing to SDRAM or SBSRAM and asynchronous peripherals.

The C6205 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 is a trademark of Texas Instruments.
Windows is a registered trademark of Microsoft Corporation.

## device characteristics

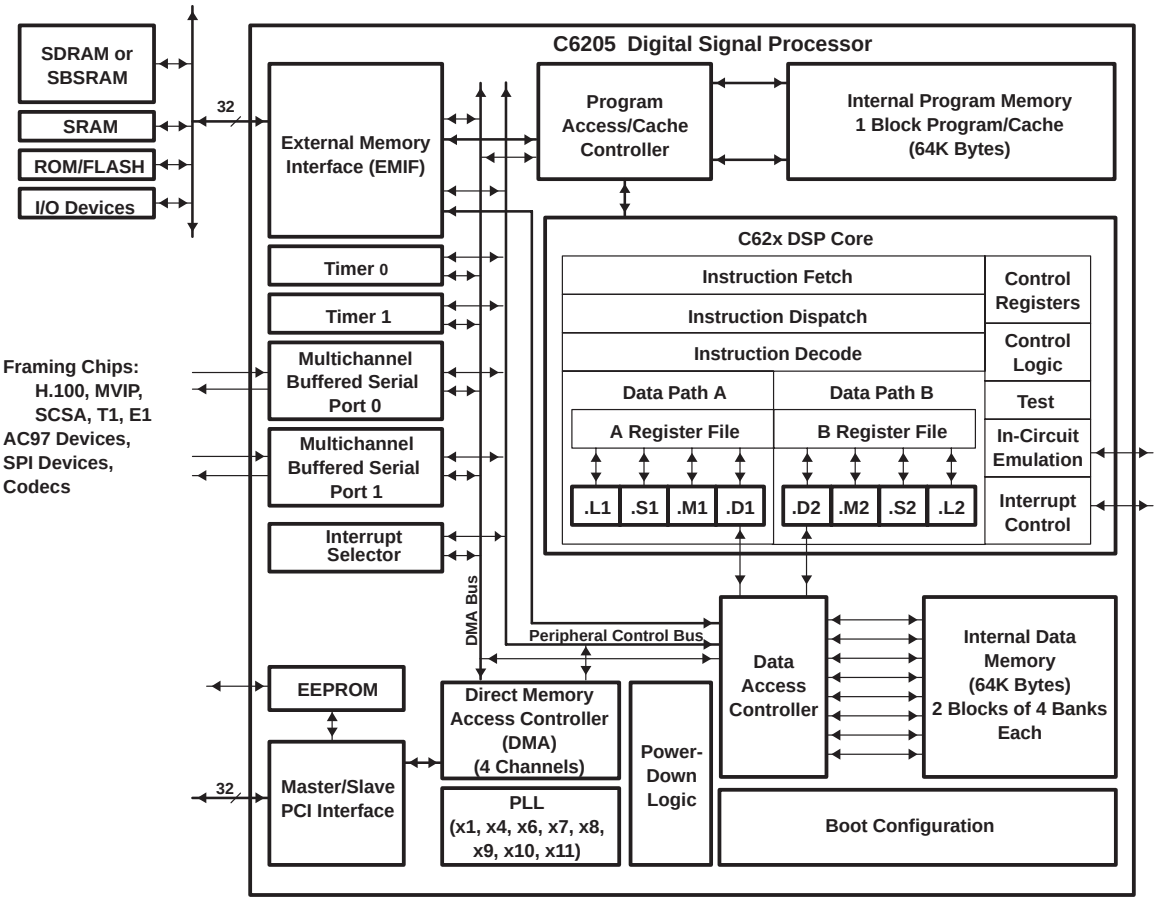
Table 1 provides an overview of the C6205 DSP. The table shows significant features of each device, including the capacity of on-chip RAM, the peripherals, the execution time, and the package type with pin count, etc.

**Table 1. Characteristics of the C6205 Processor**

| HARDWARE FEATURES | | C6205 |
|---|---|---|
| Peripherals | EMIF | 1 |
| | DMA | 4-Channel With Throughput Enhancements |
| | PCI (*Device ID*, bits 15:0, A106h [default value]) | 1 |
| | McBSPs | 2 |
| | 32-Bit Timers | 2 |
| Internal Program Memory | Size (Bytes) | 64K |
| | Organization | 1 Block: 64K Bytes Cache/Mapped Program |
| Internal Data Memory | Size (Bytes) | 64K |
| | Organization | 2 Blocks: Four 16-Bit Banks per Block, 50/50 Split |
| CPU ID+Rev ID | Control Status Register (CSR.[31:16]) | 0x0003 |
| Frequency | MHz | 200 |
| Cycle Time | ns | 5 ns (C6205-200) |
| Voltage | Core (V) | 1.5 |
| | I/O (V) | 3.3 |
| | Voltage Tolerance for PCI I/O Pins (V) | 5.0 |
| PLL Options | CLKIN frequency multiplier | Bypass (x1), x4, x6, x7, x8, x9, x10, and x11 |
| BGA Package | 16 x 16 mm | 288-Pin MicroStar BGA™ (GHK/ZHK) |
| Process Technology | µm | 0.15 µm |
| Product Status | Product Preview (PP)<br>Advance Information (AI)<br>Production Data (PD) | PD |
| Device Part Numbers | (For more details on the C6000™ DSP part numbering, see Figure 4) | TMX320C6205GHK<br>TMX320C6205ZHK |

C6000 is a trademark of Texas Instruments.

## functional and CPU (DSP core) block diagram



## CPU (DSP core) description

The CPU fetches VelociTI™ advanced very-long instruction words (VLIW) (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 C62x CPU from other VLIW architectures.

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 16 32-bit registers for a total of 32 general-purpose registers. The two sets of functional units, along with two register files, compose sides A and B of the CPU [see the Functional and CPU (DSP Core) Block Diagram and Figure 1]. The four functional units on each side of the CPU can freely share the 16 registers belonging to that side. Additionally, each side features 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. While 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, register access using the register file across the CPU supports one read and one write per cycle.

Another key feature of the C62x 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 C62x CPU supports a variety of indirect addressing modes using either linear- or circular-addressing modes with 5- or 15-bit offsets. All instructions are conditional, and most can access any one of the 32 registers. Some registers, however, are singled out to support specific addressing or to hold the condition for conditional instructions (if the condition is not automatically “true”). The two .M functional units are dedicated for multiplies. The two .S and .L functional units perform a general set of arithmetic, logical, and branch functions with results available every clock cycle.

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. If an execute packet crosses the 256-bit wide fetch-packet boundary, the assembler places it in the next fetch packet, while the remainder of the current fetch packet is padded with NOP instructions. 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 or half-words as well. All load and store instructions are byte-, half-word, or word-addressable.

## CPU (DSP core) description (continued)

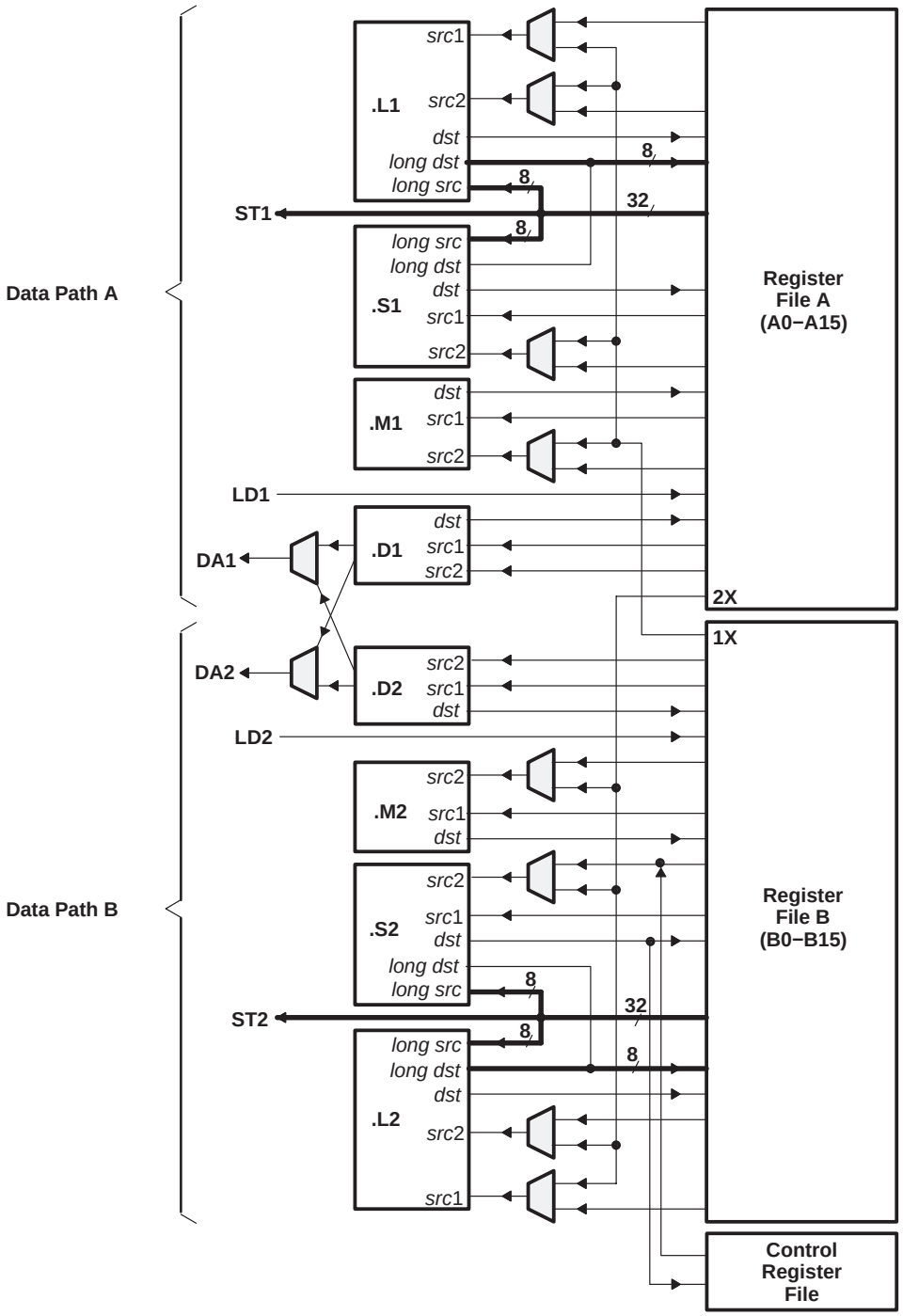



**Figure 1. TMS320C62x CPU (DSP Core) Data Paths**

## memory map summary

Table 2 shows the memory map address ranges of the C6205 device. The C6205 device has the capability of a MAP 0 or MAP 1 memory block configuration. The maps differ in that MAP 0 has *external memory* mapped at address 0x0000 0000 and MAP 1 has *internal memory* mapped at address 0x0000 0000. These memory block configurations are set up at reset by the boot configuration pins (generically called BOOTMODE[4:0]). For the C6205 device, the BOOTMODE configuration is handled, at reset, by the expansion bus module (specifically XD[4:0] pins). For more detailed information on the C6205 device settings, which include the device boot mode configuration at reset and other device-specific configurations, see *TMS320C620x/C670x DSP Boot Modes and Configuration* (literature number SPRU642).

**Table 2. TMS320C6205 Memory Map Summary**

| MEMORY BLOCK DESCRIPTION | | BLOCK SIZE (BYTES) | HEX ADDRESS RANGE |
|---|---|---|---|
| **MAP 0** | **MAP 1** | | |
| External Memory Interface (EMIF) CE0 | Internal Program RAM | 64K | 0000 0000 – 0000 FFFF |
| EMIF CE0 | Reserved | 4M – 64K | 0001 0000 – 003F FFFF |
| EMIF CE0 | EMIF CE0 | 12M | 0040 0000 – 00FF FFFF |
| EMIF CE1 | EMIF CE0 | 4M | 0100 0000 – 013F FFFF |
| Internal Program RAM | EMIF CE1 | 64K | 0140 0000 – 0140 FFFF |
| Reserved | EMIF CE1 | 4M – 64K | 0141 0000 – 017F FFFF |
| EMIF Registers | | 256K | 0180 0000 – 0183 FFFF |
| DMA Controller Registers | | 256K | 0184 0000 – 0187 FFFF |
| Reserved | | 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 |
| Reserved | | 256K | 01A0 0000 – 01A3 FFFF |
| PCI Registers | | 320K | 01A4 0000 – 01A8 FFFF |
| Reserved | | 6M – 576K | 01A9 0000 – 01FF FFFF |
| EMIF CE2 | | 16M | 0200 0000 – 02FF FFFF |
| EMIF CE3 | | 16M | 0300 0000 – 03FF FFFF |
| Reserved | | 2G – 64M | 0400 0000 – 7FFF FFFF |
| Internal Data RAM | | 64K | 8000 0000 – 8000 FFFF |
| Reserved | | 2G – 64K | 8001 0000 – FFFF FFFF |

## signal groups description

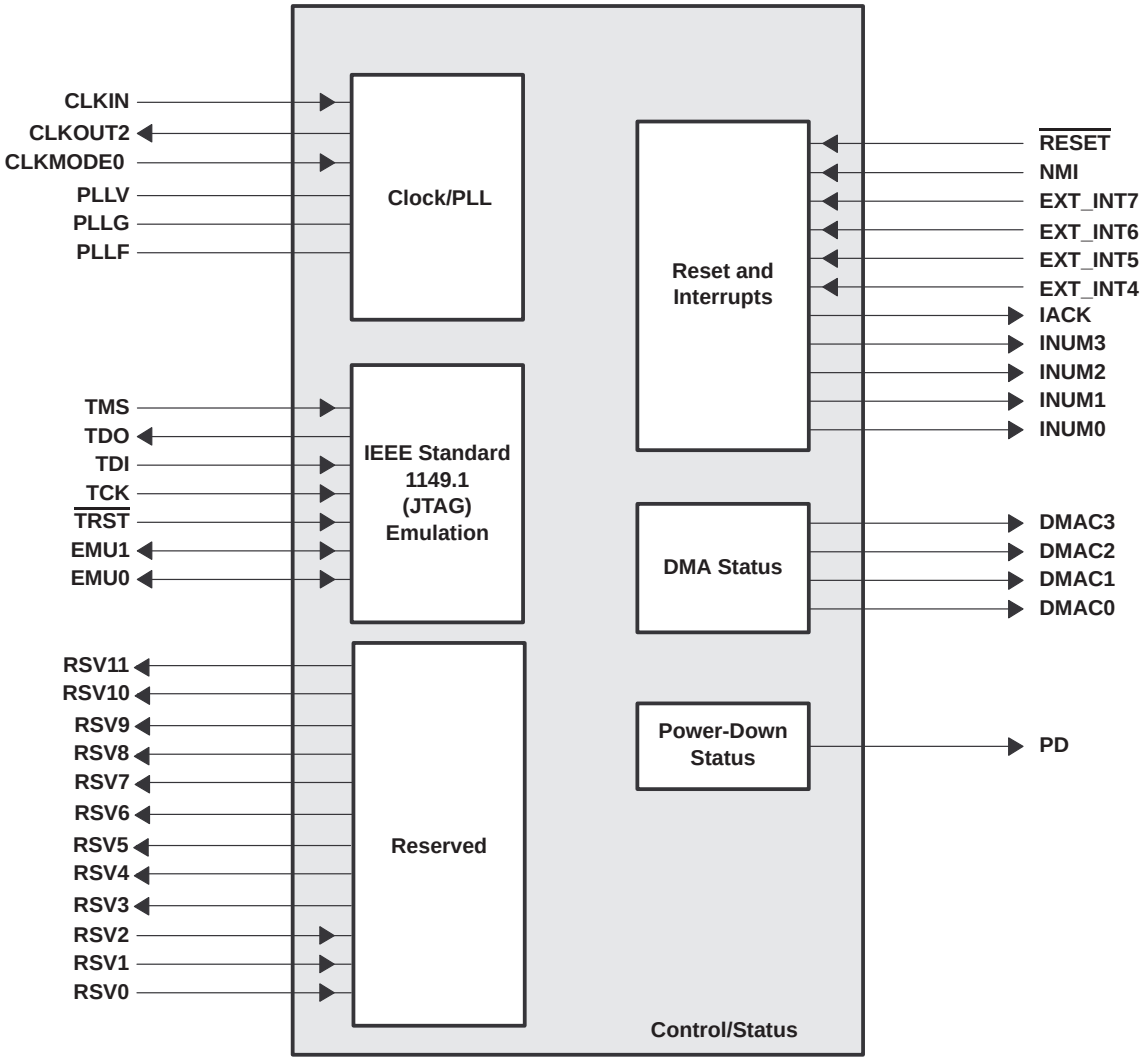



**Figure 2. CPU (DSP Core) Signals**

## signal groups description (continued)

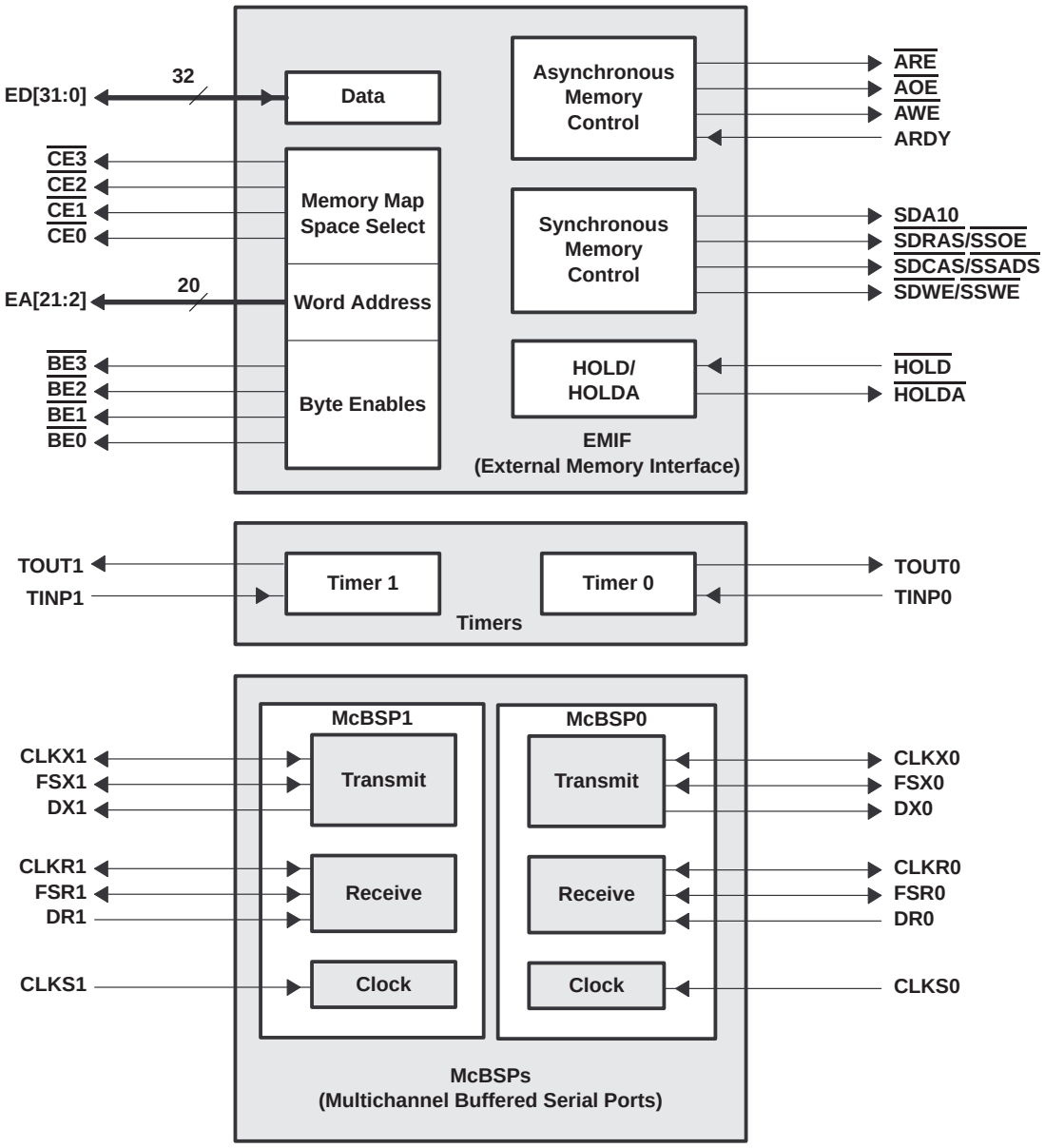



**Figure 3. Peripheral Signals**

## signal groups description (continued)

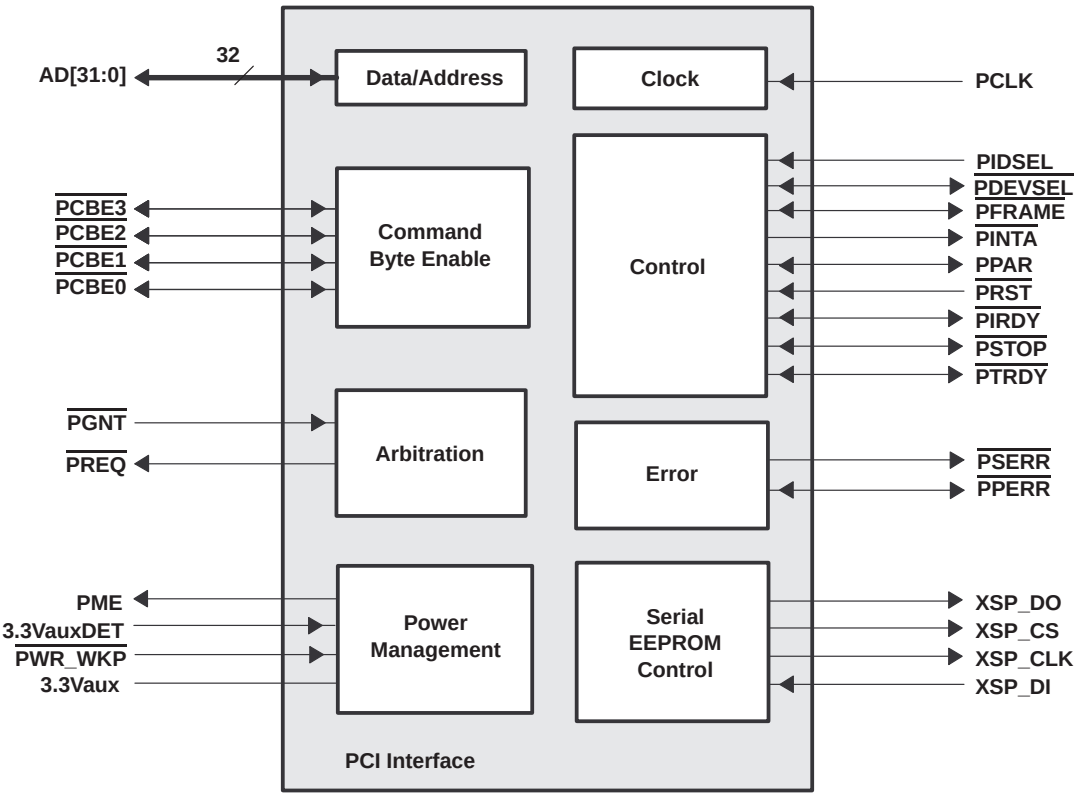



**Figure 3. Peripheral Signals (Continued)**

## Signal Descriptions

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **CLOCK/PLL** | | | |
| CLKIN | J3 | I | Clock Input |
| CLKOUT2 | T19 | O | Clock output at half of device speed<br>•Used for synchronous memory interface |
| CLKMODE0 | L3 | I | Clock mode select 0<br>•Selects whether the on-chip PLL is used or bypassed. For more details, see the *Clock PLL* section.<br>•The PLL Multiply Factor is selected at boot configuration. For more details, see the EMIF − Data pin descriptions and the clock PLL section. |
| PLLV‡ | K5 | A§ | PLL analog $V_{CC}$ connection for the low-pass filter |
| PLLG‡ | L2 | A§ | PLL analog GND connection for the low-pass filter |
| PLLF‡ | L1 | A§ | PLL low-pass filter connection to external components and a bypass capacitor |
| **JTAG EMULATION** | | | |
| TMS | E17 | I | JTAG test-port mode select (features an internal pullup) |
| TDO | D19 | O/Z | JTAG test-port data out |
| TDI | D18 | I | JTAG test-port data in (features an internal pullup) |
| TCK | D17 | I | JTAG test-port clock |
| $\overline{\text{TRST}}$ | C19 | I | JTAG test-port reset (features an internal pulldown) |
| EMU1 | E18 | I/O/Z | Emulation pin 1, pullup with a dedicated 20-kΩ resistor¶ |
| EMU0 | F15 | I/O/Z | Emulation pin 0, pullup with a dedicated 20-kΩ resistor¶ |
| **RESET AND INTERRUPTS** | | | |
| $\overline{\text{RESET}}$ | C3 | I | Device reset |
| NMI | A8 | I | Nonmaskable interrupt<br>•Edge-driven (rising edge) |
| EXT_INT7 | B15 | I | External interrupts<br>•Edge-driven<br>•Polarity independently selected via the External Interrupt Polarity Register bits (EXTPOL.[3:0]) |
| EXT_INT6 | C15 | | |
| EXT_INT5 | A16 | | |
| EXT_INT4 | B16 | | |
| IACK | A15 | O | Interrupt acknowledge for all active interrupts serviced by the CPU |
| INUM3 | F12 | O | Active interrupt identification number<br>•Valid during IACK for all active interrupts (not just external)<br>•Encoding order follows the interrupt-service fetch-packet ordering |
| INUM2 | A14 | | |
| INUM1 | B14 | | |
| INUM0 | C14 | | |
| **POWER-DOWN STATUS** | | | |
| PD | B18 | O | Power-down modes 2 or 3 (active if high) |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground
‡ PLLV, PLLG, and PLLF are not part of external voltage supply or ground. See the *clock PLL* section for information on how to connect these pins.
§ A = Analog Signal (PLL Filter)
¶ For emulation and normal operation, pull up EMU1 and EMU0 with a dedicated 20-kΩ resistor. For boundary scan, pull down EMU1 and EMU0 with a dedicated 20-kΩ resistor.

**Signal Descriptions (Continued)**

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **PCI INTERFACE** | | | |
| PCLK | W5 | I | PCI input clock |
| AD31 | D2 | I/O/Z | PCI Data-Address bus |
| AD30 | E3 | | |
| AD29 | E2 | | |
| AD28 | E1 | | |
| AD27 | F3 | | |
| AD26 | F5 | | |
| AD25 | F1 | | |
| AD24 | G3 | | |
| AD23 | H3 | | |
| AD22 | H2 | | |
| AD21 | J1 | | |
| AD20 | H1 | | |
| AD19 | M2 | | |
| AD18 | M1 | | |
| AD17 | N2 | | |
| AD16 | N1 | | |
| AD15 | T1 | | |
| AD14 | V2 | | |
| AD13 | U2 | | |
| AD12 | U1 | | |
| AD11 | W3 | | |
| AD10 | W2 | | |
| AD9 | V1 | | |
| AD8 | U4 | | |
| AD7 | W4 | | |
| AD6 | U5 | | |
| AD5 | V5 | | |
| AD4 | U6 | | |
| AD3 | V6 | | |
| AD2 | V3 | | |
| AD1 | W6 | | |
| AD0 | U7 | | |
| $\overline{\text{PCBE3}}$ | G2 | I/O/Z | PCI command/byte enable signals |
| $\overline{\text{PCBE2}}$ | M3 | | |
| $\overline{\text{PCBE1}}$ | T2 | | |
| $\overline{\text{PCBE0}}$ | V4 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

## Signal Descriptions (Continued)

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **PCI INTERFACE (CONTINUED)** | | | |
| $\overline{\text{PINTA}}$ | C1 | O/Z | PCI interrupt A |
| $\overline{\text{PREQ}}$ | F2 | O/Z | PCI bus request (bus arbitration) |
| $\overline{\text{PSERR}}$ | P5 | O/Z | PCI system error |
| $\overline{\text{PPERR}}$ | P2 | I/O/Z | PCI parity error |
| $\overline{\text{PRST}}$ | C2 | I | PCI reset |
| $\overline{\text{PDEVSEL}}$ | R2 | I/O/Z | PCI device select |
| $\overline{\text{PGNT}}$ | D1 | I | PCI bus grant (bus arbitration) |
| $\overline{\text{PFRAME}}$ | N5 | I/O/Z | PCI frame |
| $\overline{\text{PIRDY}}$ | P1 | I/O/Z | PCI initiator ready |
| PPAR | T3 | I/O/Z | PCI parity |
| PIDSEL | H5 | I | PCI initialization device select |
| $\overline{\text{PSTOP}}$ | R1 | I/O/Z | PCI stop |
| $\overline{\text{PTRDY}}$ | N3 | I/O/Z | PCI target ready |
| XSP_CLK | C17 | O | Serial EEPROM clock |
| XSP_DI | C18 | I | Serial EEPROM data in, pulldown with a dedicated 20-kΩ resistor |
| XSP_DO | B19 | O | Serial EEPROM data out |
| XSP_CS | C11 | O | Serial EEPROM chip select |
| 3.3VauxDET | B1 | I | 3.3-V auxiliary power supply detect.<br><br>Used to indicate the presence of 3.3Vaux. A weak pulldown must be implemented to this pin. |
| 3.3Vaux | B2 | S | 3.3-V auxiliary power supply voltage |
| PME | D3 | O | Power management event |
| $\overline{\text{PWR_WKP}}$ | A2 | I | Power wakeup signal |
| **EMIF − CONTROL SIGNALS COMMON TO ALL TYPES OF MEMORY** | | | |
| $\overline{\text{CE3}}$ | V18 | O/Z | Memory space enables<br>•Enabled by bits 24 and 25 of the word address<br>•Only one asserted during any external data access |
| $\overline{\text{CE2}}$ | U17 | | |
| $\overline{\text{CE1}}$ | W18 | | |
| $\overline{\text{CE0}}$ | V17 | | |
| $\overline{\text{BE3}}$ | U16 | O/Z | Byte-enable control<br>•Decoded from the two lowest bits of the internal address<br>•Byte-write enables for most types of memory<br>•Can be directly connected to SDRAM read and write mask signal (SDQM) |
| $\overline{\text{BE2}}$ | W17 | | |
| $\overline{\text{BE1}}$ | V16 | | |
| $\overline{\text{BE0}}$ | W16 | | |
| **EMIF − ADDRESS** | | | |
| EA21 | V7 | O/Z | External address (word address) |
| EA20 | W7 | | |
| EA19 | U8 | | |
| EA18 | V8 | | |
| EA17 | W8 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

## Signal Descriptions (Continued)

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **EMIF − ADDRESS (CONTINUED)** | | | |
| EA16 | W9 | O/Z | External address (word address) |
| EA15 | V9 | | |
| EA14 | U9 | | |
| EA13 | W10 | | |
| EA12 | V10 | | |
| EA11 | U10 | | |
| EA10 | W11 | | |
| EA9 | V11 | | |
| EA8 | U11 | | |
| EA7 | R11 | | |
| EA6 | W12 | | |
| EA5 | U12 | | |
| EA4 | R12 | | |
| EA3 | W13 | | |
| EA2 | V13 | | |
| **EMIF − DATA** | | | |
| ED31 | F14 | I/O/Z | External data<br>•Used for transfer of EMIF data<br>•Also controls initialization of DSP modes at reset via pullup/pulldown resistors<br>ED31 - PLL_Conf2<br>ED27 - PLL_Conf1<br>ED23 - PLL_Conf0<br>ED15 - EEPROM autoinitialization<br>ED8 - Endianness<br>ED[7:5] - EEPROM size<br>ED[4:0] - Bootmode |
| ED30 | E19 | | |
| ED29 | F17 | | |
| ED28 | G15 | | |
| ED27 | F18 | | |
| ED26 | F19 | | |
| ED25 | G17 | | |
| ED24 | G18 | | |
| ED23 | G19 | | |
| ED22 | H17 | | |
| ED21 | H18 | | |
| ED20 | H19 | | |
| ED19 | J18 | | |
| ED18 | J19 | | |
| ED17 | K15 | | |
| ED16 | K17 | | |
| ED15 | K18 | | |
| ED14 | K19 | | |
| ED13 | L17 | | |
| ED12 | L18 | | |
| ED11 | L19 | | |
| ED10 | M19 | | |
| ED9 | M18 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **EMIF − DATA (CONTINUED)** | | | |
| ED8 | M17 | I/O/Z | External data |
| ED7 | N19 | | |
| ED6 | P19 | | |
| ED5 | N15 | | |
| ED4 | P18 | | |
| ED3 | P17 | | |
| ED2 | R19 | | |
| ED1 | R18 | | |
| ED0 | R17 | | |
| **EMIF − ASYNCHRONOUS MEMORY CONTROL** | | | |
| $\overline{\text{ARE}}$ | U14 | O/Z | Asynchronous memory read-enable |
| $\overline{\text{AOE}}$ | W14 | O/Z | Asynchronous memory output-enable |
| $\overline{\text{AWE}}$ | V14 | O/Z | Asynchronous memory write-enable |
| ARDY | W15 | I | Asynchronous memory ready input |
| **EMIF − SYNCHRONOUS DRAM (SDRAM)/SYNCHRONOUS BURST SRAM (SBSRAM) CONTROL** | | | |
| SDA10 | U19 | O/Z | SDRAM address 10 (separate for deactivate command) |
| $\overline{\text{SDCAS}}$/$\overline{\text{SSADS}}$ | V19 | O/Z | SDRAM column-address strobe/SBSRAM address strobe |
| $\overline{\text{SDRAS}}$/$\overline{\text{SSOE}}$ | U18 | O/Z | SDRAM row-address strobe/SBSRAM output-enable |
| $\overline{\text{SDWE}}$/$\overline{\text{SSWE}}$ | T17 | O/Z | SDRAM write-enable/SBSRAM write-enable |
| **EMIF − BUS ARBITRATION** | | | |
| $\overline{\text{HOLD}}$ | P14 | I | Hold request from the host |
| $\overline{\text{HOLDA}}$ | V15 | O | Hold-request-acknowledge to the host |
| **TIMER 0** | | | |
| TOUT0 | E5 | O | Timer 0 or general-purpose output |
| TINP0 | C5 | I | Timer 0 or general-purpose input |
| Timer 1 | | | |
| TOUT1 | A5 | O | Timer 1 or general-purpose output |
| TINP1 | B5 | I | Timer 1 or general-purpose input |
| **DMA ACTION COMPLETE STATUS** | | | |
| DMAC3 | A17 | O | DMA action complete |
| DMAC2 | B17 | | |
| DMAC1 | C16 | | |
| DMAC0 | A18 | | |
| **MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP0)** | | | |
| CLKS0 | A12 | I | External clock source (as opposed to internal) |
| CLKR0 | B9 | I/O/Z | Receive clock |
| CLKX0 | C9 | I/O/Z | Transmit clock |
| DR0 | A10 | I | Receive data |
| DX0 | B10 | O/Z | Transmit data |
| FSR0 | E10 | I/O/Z | Receive frame sync |
| FSX0 | A9 | I/O/Z | Transmit frame sync |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

## Signal Descriptions (Continued)

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP1)** | | | |
| CLKS1 | C6 | I | External clock source (as opposed to internal) |
| CLKR1 | B6 | I/O/Z | Receive clock |
| CLKX1 | E6 | I/O/Z | Transmit clock |
| DR1 | A7 | I | Receive data |
| DX1 | B7 | O/Z | Transmit data |
| FSR1 | C7 | I/O/Z | Receive frame sync |
| FSX1 | A6 | I/O/Z | Transmit frame sync |
| **RESERVED FOR TEST** | | | |
| RSV0 | C8 | I | Reserved for testing, pullup with a dedicated 20-kΩ resistor |
| RSV1 | A4 | I | Reserved for testing, pullup with a dedicated 20-kΩ resistor |
| RSV2 | K3 | I | Reserved for testing, pullup with a dedicated 20-kΩ resistor |
| RSV3 | L5 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV4 | T18 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV5 | A3 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV6 | B3 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV7 | B4 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV8 | C4 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV9 | K2 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV10 | J17 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| RSV11 | N18 | O | Reserved (leave unconnected, ***do not*** connect to power or ground) |
| **SUPPLY VOLTAGE PINS** | | | |
| $DV_{DD}$ | B8 | S | 3.3-V I/O supply voltage |
| | E7 | | |
| | E8 | | |
| | E9 | | |
| | E11 | | |
| | E13 | | |
| | H14 | | |
| | K14 | | |
| | L15 | | |
| | M14 | | |
| | P15 | | |
| | R8 | | |
| | R9 | | |
| | R10 | | |
| | R13 | | |
| | R14 | | |
| | U15 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **SUPPLY VOLTAGE PINS (CONTINUED)** | | | |
| $CV_{DD}$ | B12 | S | 1.5-V core supply voltage |
| | E14 | | |
| | F9 | | |
| | F10 | | |
| | G5 | | |
| | H15 | | |
| | J2 | | |
| | J5 | | |
| | J15 | | |
| | M5 | | |
| | M15 | | |
| | N17 | | |
| | P6 | | |
| | P9 | | |
| | P12 | | |
| | U13 | | |
| **PCI SUPPLY VOLTAGE PINS** | | | |
| $V_{IOP}$ | G1 | S | 3.3/5-V PCI clamp pins |
| | P3 | | |
| | U3 | | |
| $V_{DDP}$ | F6 | S | 3.3-V PCI power supply pins |
| | J6 | | |
| | L6 | | |
| | R3 | | |
| | R6 | | |
| | R7 | | |
| **GROUND PINS** | | | |
| $V_{SS}$ | A11 | GND | Ground pins |
| | A13 | | |
| | B11 | | |
| | B13 | | |
| | C10 | | |
| | C12 | | |
| | C13 | | |
| | E12 | | |
| | G7 | | |
| | G8 | | |
| | G9 | | |
| | G10 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **GROUND PINS (CONTINUED)** | | | |
| $V_{SS}$ | G11 | GND | Ground pins |
| | G12 | | |
| | G13 | | |
| | H7 | | |
| | H8 | | |
| | H9 | | |
| | H10 | | |
| | H11 | | |
| | H12 | | |
| | H13 | | |
| | J7 | | |
| | J8 | | |
| | J9 | | |
| | J10 | | |
| | J11 | | |
| | J12 | | |
| | J13 | | |
| | K1 | | |
| | K7 | | |
| | K8 | | |
| | K9 | | |
| | K10 | | |
| | K11 | | |
| | K12 | | |
| | K13 | | |
| | L7 | | |
| | L8 | | |
| | L9 | | |
| | L10 | | |
| | L11 | | |
| | L12 | | |
| | L13 | | |
| | M7 | | |
| | M8 | | |
| | M9 | | |
| | M10 | | |
| | M11 | | |
| | M12 | | |
| | M13 | | |

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

| SIGNAL NAME | SIGNAL NO. | TYPE† | DESCRIPTION |
|---|---|---|---|
| **GROUND PINS (CONTINUED)** | | | |
| $V_{SS}$ | N7 | GND | Ground pins |
| | N8 | | |
| | N9 | | |
| | N10 | | |
| | N11 | | |
| | N12 | | |
| | N13 | | |
| | V12 | | |

† 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) and select “Find Development Tools”. For device-specific tools, under “Semiconductor Products” select “Digital Signal Processors”, choose a product family, and select the particular DSP device. 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 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 (i.e., **TMS**320C6205GHK200). Texas Instruments recommends two of three possible prefix designators for 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."

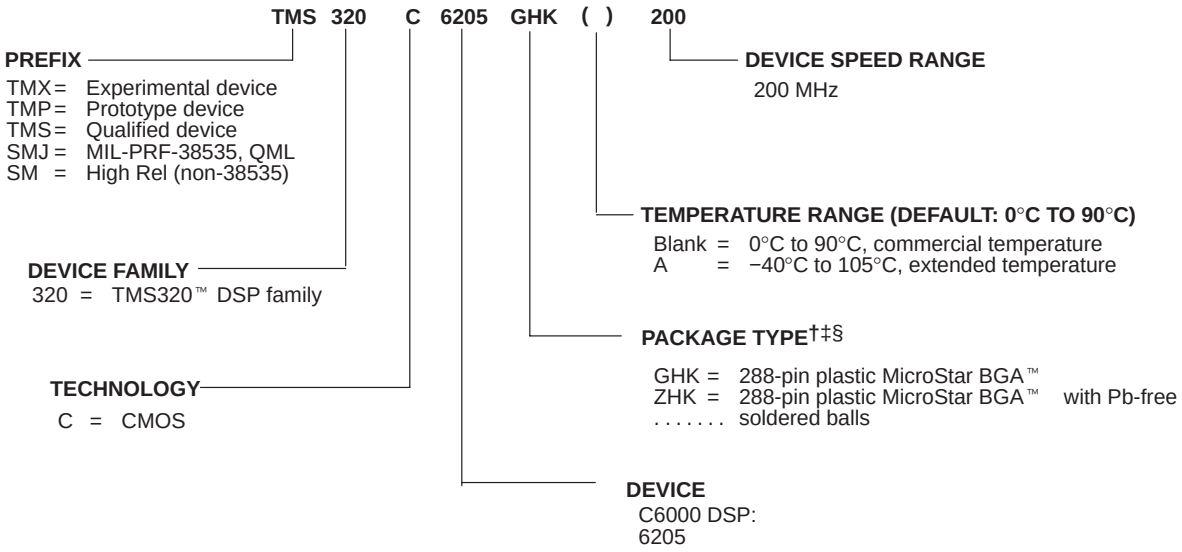
TMS devices and TMDS development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

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, GHK), the temperature range (for example, blank is the default commercial temperature range), and the device speed range in megahertz (for example, -200 is 200 MHz).

The ZHK package, like the GHK package, is a 288-ball plastic BGA *only* with Pb-free balls.For device part numbers and further ordering information for TMS320C6205 in the GHK and ZHK 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 actual device part numbers (P/Ns) and ordering information, see the Mechanical Data section of this document or the TI website (www.ti.com).
§ The ZHK mechanical package designator represents the version of the GHK with Pb−Free soldered balls.

**Figure 4. TMS320C6000™ DSP Platform Device Nomenclature (Including the TMS320C6205 Device)**

## documentation support

Extensive documentation supports all TMS320™ DSP family 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 core (CPU) architecture, instruction set, pipeline, and associated interrupts.

The *TMS320C6000 DSP Peripherals Overview Reference Guide* (literature number SPRU190) briefly describes the functionality of the peripherals available on the C6000™ DSP platform of devices, such as the 64-/32-/16-bit external memory interfaces (EMIFs), 32-/16-bit host-port interfaces (HPIs), multichannel buffered serial ports (McBSPs), direct memory access (DMA), enhanced direct-memory-access (EDMA) controller, expansion bus (XB), peripheral component interconnect (PCI), clocking and phase-locked loop (PLL); and power-down modes.

The *TMS320C6000 Technical Brief* (literature number SPRU197) gives an introduction to the C62x™/C67x™ devices, associated development tools, and third-party support.

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

See the Worldwide Web URL for the new *How to Begin Development with the TMS320C6205 DSP* application report (literature number SPRA596) which describes the functionalities unique to the C6205 device, especially the peripheral component interconnect (PCI) module interface.

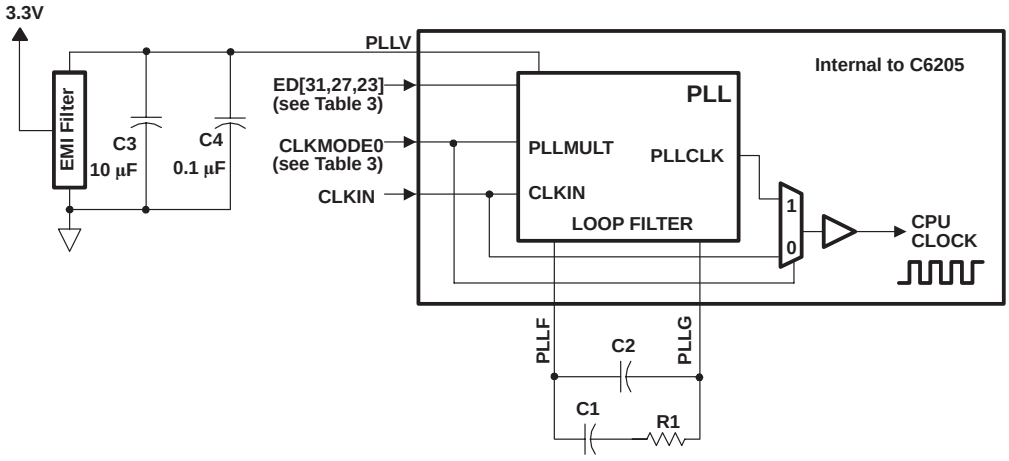
C62x and C67x are trademarks of Texas Instruments.

## clock PLL

Most of the internal C6205 clocks are generated from a single source through the CLKIN pin. This source clock either drives the PLL, which multiplies the source clock in 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 5, Table 3, and Table 4 show the external PLL circuitry for either x1 (PLL bypass) or x4 PLL multiply modes. Figure 6 shows the external PLL circuitry for a system with ONLY x1 (PLL bypass) mode.

To minimize the clock jitter, a single clean power supply should power both the C6205 device and the external clock oscillator circuit. Noise coupling into PLLF directly impacts PLL clock jitter. 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.




NOTES: A. Keep the lead length and the number of vias between pin PLLF, pin PLLG, R1, C1, and C2 to a minimum. In addition, place all PLL components (R1, C1, C2, C3, C4, and EMI Filter) as close to the C6000™ DSP device as possible. Best performance is achieved with the PLL components on a single side of the board without jumpers, switches, or components other than the ones shown.

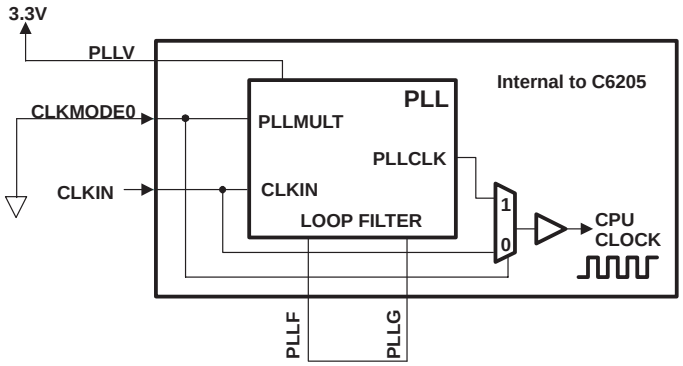
B. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (R1, C1, C2, C3, C4, and the EMI Filter).

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.

E. At power up, the PLL requires a falling edge of RESET to initialize the PLL engine. It may be necessary to toggle reset in order to establish proper PLL operation.

**Figure 5. External PLL Circuitry for Either PLL Multiply Modes or x1 (Bypass) Mode**




NOTES: A. For a system with ONLY PLL x1 (bypass) mode, short the PLLF to PLLG.

B. The 3.3-V supply for PLLV must be from the same 3.3-V power plane supplying the I/O voltage, $DV_{DD}$.

**Figure 6. External PLL Circuitry for x1 (Bypass) PLL Mode Only**

## clock PLL (continued)

### Table 3. C6205 PLL Multiply Modes and x1 (Bypass) Options

| CLKMODE0† | ED[31]‡ | ED[27]‡ | ED[23]‡ | PLL MULTIPLY FACTORS | CPU CLOCK FREQ f(CPU clock) |
|---|---|---|---|---|---|
| 0 | X | X | X | x1 (Bypass) | $1 \times f_{(CLKIN)}$ |
| 1 | 0 | 0 | 0 | x1 (Bypass) | $1 \times f_{(CLKIN)}$ |
| 1 | 0 | 0 | 1 | x4 | $4 \times f_{(CLKIN)}$ |
| 1 | 0 | 1 | 0 | x8 | $8 \times f_{(CLKIN)}$ |
| 1 | 0 | 1 | 1 | x10 | $10 \times f_{(CLKIN)}$ |
| 1 | 1 | 0 | 0 | x6 | $6 \times f_{(CLKIN)}$ |
| 1 | 1 | 0 | 1 | x9 | $9 \times f_{(CLKIN)}$ |
| 1 | 1 | 1 | 0 | x7 | $7 \times f_{(CLKIN)}$ |
| 1 | 1 | 1 | 1 | x11 | $11 \times f_{(CLKIN)}$ |

† CLKMODE0 equal to 0 denotes on-chip PLL bypassed
CLKMODE0 equal to 1 denotes on-chip PLL used, except when configuration bits (ED[31], ED[27], and ED[23]) are 0 at device reset.

‡ ED[31], ED[27], and ED[23] are the on-chip PLL configuration bits that are latched during device reset, along with the other boot configuration bits ED[31:0].

### Table 4. C6205 PLL Component Selection Table§

| CLKMODE | CLKIN RANGE (MHz) | CPU CLOCK FREQUENCY (CLKOUT1) RANGE (MHz) | CLKOUT2 RANGE (MHz) | R1 [±1%] (Ω) | C1 [±10%] (nF) | C2 [±10%] (pF) | TYPICAL LOCK TIME (μs) |
|---|---|---|---|---|---|---|---|
| x4 | 32.5−50 | 130−200 | 65−100 | 60.4 | 27 | 560 | 75 |
| x6 | 21.7−33.3 | | | | | | |
| x7 | 18.6−28.6 | | | | | | |
| x8 | 16.3−25 | | | | | | |
| x9 | 14.4−22.2 | | | | | | |
| x10 | 13−20 | | | | | | |
| x11 | 11.8−18.2 | | | | | | |

§ Under some operating conditions, the maximum PLL lock time may vary 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.

## power-down mode logic

Figure 7 shows the power-down mode logic on the C6205.

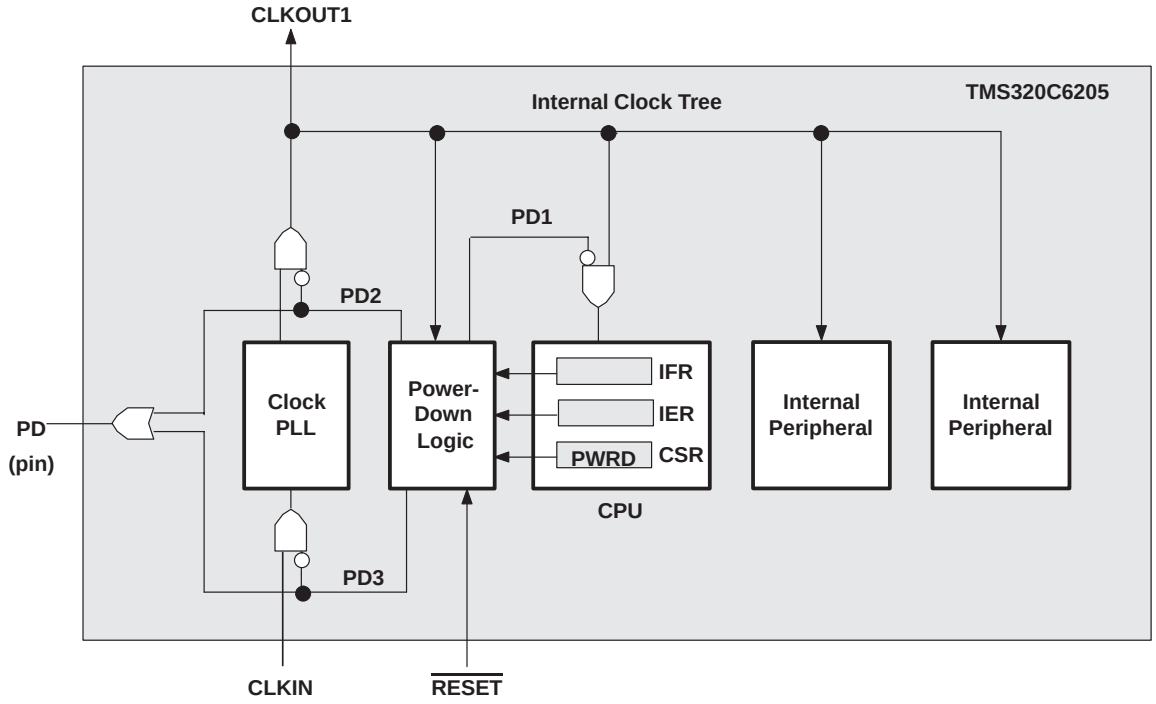
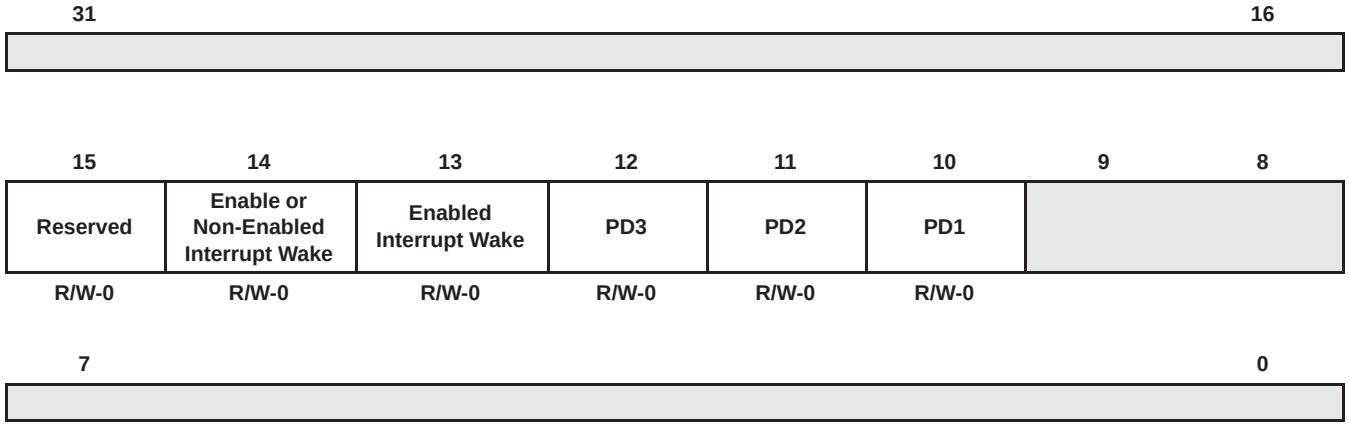



**Figure 7. Power-Down Mode Logic†**

**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 8 and described in Table 5. 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 8. PWRD Field of the CSR Register**

Power-down mode PD1 takes effect eight to nine clock cycles after the instruction that sets the PWRD bits in the CSR.

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. The GIE bit in 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 5 summarizes all the power-down modes.

**Table 5. 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)<br>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, DMA 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. |
| 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.

## 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 if the other supply is below the proper operating voltage.

### system-level design considerations

System-level design considerations, such as bus contention, may require supply sequencing to be implemented. In this case, the core supply should be powered up at the same time as, or prior to (and powered down after), the I/O buffers. This is to ensure that the I/O buffers receive valid inputs from the core before the output buffers are powered up, thus, preventing bus contention with other chips on the board.

### power-supply design considerations

For systems using the C6000™ DSP platform of devices, the core supply may be required to provide in excess of 2 A per DSP until the I/O supply is powered up. This extra current condition is a result of uninitialized logic within the DSP(s) and is corrected once the CPU sees an internal clock pulse. With the PLL enabled, as the I/O supply is powered on, a clock pulse is produced stopping the extra current draw from the supply. With the PLL disabled, as many as five external clock cycle pulses may be required to stop this extra current draw. A normal current state returns once the I/O power supply is turned on and the CPU sees a clock pulse. Decreasing the amount of time between the core supply power up and the I/O supply power up can minimize the effects of this current draw.

A dual-power supply with simultaneous sequencing, such as that available with TPS563xx controllers or PT69xx plug-in power modules, can be used to eliminate the delay between core and I/O power up [see the *Using the TPS56300 to Power DSPs* application report (literature number SLVA088)]. A Schottky diode can also be used to tie the core rail to the I/O rail, effectively pulling up the I/O power supply to a level that can help initialize the logic within the DSP.

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.

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

Supply voltage ranges: $CV_{DD}$ (see Note 1) . . . . . − 0.3 V to 2.3 V
$DV_{DD}$ (see Note 1) . . . . . −0.3 V to 4 V
(PCI), $V_{IOP}$ (see Note 1) . . . . . −0.5 V to 5.5 V
(PCI), $V_{DDP}$ (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 $V_{IOP}$ + 0.5 V
Output voltage ranges: (except PCI), $V_O$ . . . . . −0.3 V to 4 V
(PCI), $V_{OP}$ . . . . . −0.5 V to $V_{IOP}$ + 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 | 1.43 | 1.5 | 1.57 | 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 | 2 | | | V |
| $V_{IL}$ | Low-level input voltage | | | 0.8 | V |
| $I_{OH}$ | High-level output current | | | −8 | mA |
| $I_{OL}$ | Low-level output current | | | 8 | mA |
| $T_C$ | Operating case temperature | 0 | | 90 | °C |

### recommended operating conditions (PCI only)

| | | | OPERATION | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{DDP}$ | 3.3-V PCI power supply voltage‡ | | 3.3 V | 3 | 3.3 | 3.6 | V |
| $V_{IOP}$ | 3.3/5-V PCI Clamp voltage (PCI) | | 3.3 V | 3 | 3.3 | 3.6 | V |
| | | | 5 V | 4.75 | 5 | 5.25 | V |
| $V_{IP}$ | Input voltage (PCI) | | 3.3 V | −0.5 | | $V_{IOP}$ + 0.5 | V |
| | | | 5 V | −0.5 | | $V_{IOP}$ + 0.5 | |
| $V_{IHP}$ | High-level input voltage (PCI) | CMOS-compatible | 3.3 V | $0.5V_{IOP}$ | | $V_{IOP}$ + 0.5 | V |
| | | | 5 V | 2 | | $V_{IOP}$ + 0.5 | |
| $V_{ILP}$ | Low-level input voltage (PCI) | CMOS-compatible | 3.3 V | −0.5 | | $0.3V_{IOP}$ | V |
| | | | 5 V | −0.5 | | 0.8 | |

‡ The 3.3-V PCI power supply voltage should follow similar sequencing as the I/O buffers supply voltage, see the power-supply sequencing section of this data sheet.

## 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 |
| $V_{OL}$ | Low-level output voltage (except PCI) | $DV_{DD}$ = MIN, | $I_{OL}$ = MAX | | | 0.6 | V |
| $I_I$ | Input current† | $V_I = V_{SS}$ to $DV_{DD}$ | | | | ±10 | µA |
| $I_{OZ}$ | Off-state output current | $V_O = DV_{DD}$ or 0 V | | | | ±10 | µA |
| $I_{DD2V}$ | Supply current, CPU + CPU memory access‡ | $CV_{DD}$ = NOM, | CPU clock = 200 MHz | | 290 | | mA |
| $I_{DD2V}$ | Supply current, peripherals‡ | $CV_{DD}$ = NOM, | CPU clock = 200 MHz | | 240 | | mA |
| $I_{DD3V}$ | Supply current, I/O pins‡ | $DV_{DD}$ = NOM, | CPU clock = 200 MHz | | 100 | | mA |
| $C_i$ | Input capacitance | | | | | 10 | pF |
| $C_o$ | Output capacitance | | | | | 10 | pF |

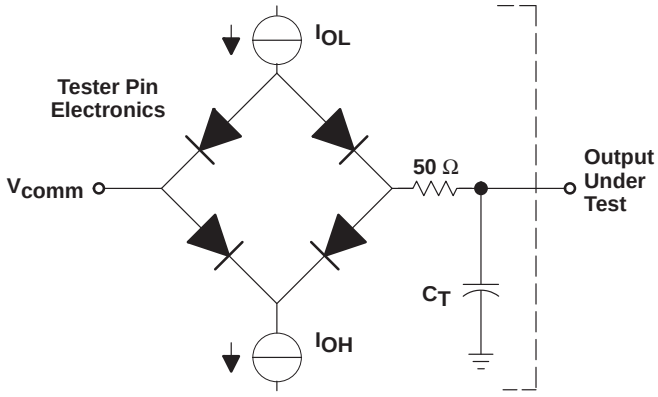
† TMS and TDI are not included due to internal pullups. $\overline{TRST}$ is not included due to internal pulldown.
‡ Measured with average activity (50% high/50% low power). For more details on CPU, peripheral, and I/O activity, see the *TMS320C6000 Power Consumption Summary* application report (literature number SPRA486).

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

| PARAMETER | | PCI SIDE | TEST CONDITIONS AND OPERATION | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{OHP}$ | High-level output voltage (PCI) | All PCI pins | $I_{OHP}$ = −0.5 mA | 3.3 V | $0.9V_{IOP}$§ | | V |
| | | | $I_{OHP}$ = −2 mA | 5 V | 2.4 | | |
| $V_{OLP}$ | Low-level output voltage (PCI) | All PCI pins | $I_{OLP}$ = 1.5 mA | 3.3 V | | $0.1V_{IOP}$§ | V |
| | | | $I_{OLP}$ = 6 mA | 5 V | | 0.55 | |
| $I_{ILP}$ | Low-level input leakage current (PCI) | All PCI pins§ | $0 < V_{IP} < V_{IOP}$ | 3.3 V | | ±10 | µA |
| | | | $V_{IP}$ = 0.5 V | 5 V | | −70 | |
| $I_{IHP}$ | High-level input leakage current (PCI) | All PCI pins§ | $V_{IP}$ = 2.7 V | 5 V | | 70 | µA |

§ Input leakage currents include Hi-Z output leakage for all bidirectional buffers with 3-state outputs.

## PARAMETER MEASUREMENT INFORMATION




Where: $I_{OL}$ = 2 mA
$I_{OH}$ = 2 mA
$V_{comm}$ = 0.8 V
$C_T$ = 15−30-pF typical load-circuit capacitance

**Figure 9. Test Load Circuit for AC Timing Measurements**

### signal transition levels

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

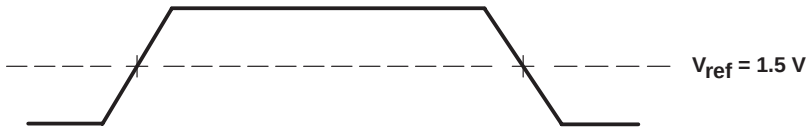



**Figure 10. Input and Output Voltage Reference Levels for ac Timing Measurements**

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.

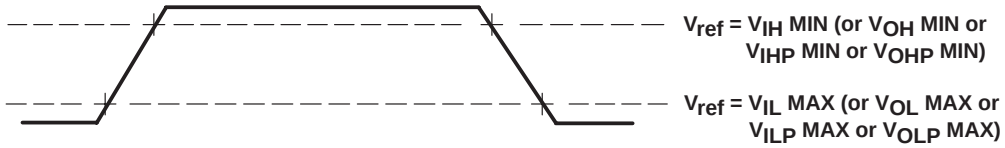



**Figure 11. Rise and Fall Transition Time Voltage Reference Levels**

## INPUT AND OUTPUT CLOCKS

### timing requirements for CLKIN†‡§ (see Figure 12)

| NO. | | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | PLL mode x4, x6, x7, x8, x9, x10, x11 | | PLL mode x1 | | |
| | | | MIN | MAX | MIN | MAX | |
| 1 | $t_{c(CLKIN)}$ | Cycle time, CLKIN | 5 * M | | 5 | | 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 | | 0.6 | ns |

† The reference points for the rise and fall transitions are measured at $V_{IL}$ MAX and $V_{IH}$ MIN.
‡ M = the PLL multiplier factor (x4, x6, x7, x8, x9, x10, or x11). For more details, 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.

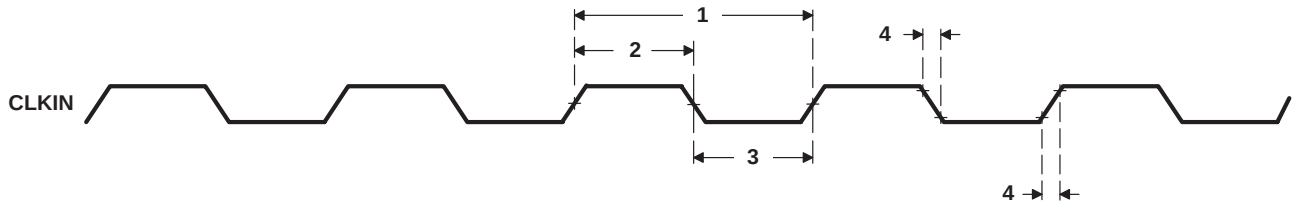



**Figure 12. CLKIN Timings**

### timing requirements for PCLKIN¶ (see Figure 13)

| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(PCLK)}$ | Cycle time, PCLK | 30 | | 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 |

¶ When the 5-V PCI clamp is used, the reference points for the rise and fall transitions are measured $V_{ILP}$ MAX and $V_{IHP}$ MIN for 5 V operation. When the 3.3-V PCI clamp is used, the reference points for the rise and fall transitions are measured at $V_{ILP}$ MAX and $V_{IHP}$ MIN for 3.3 V operation.

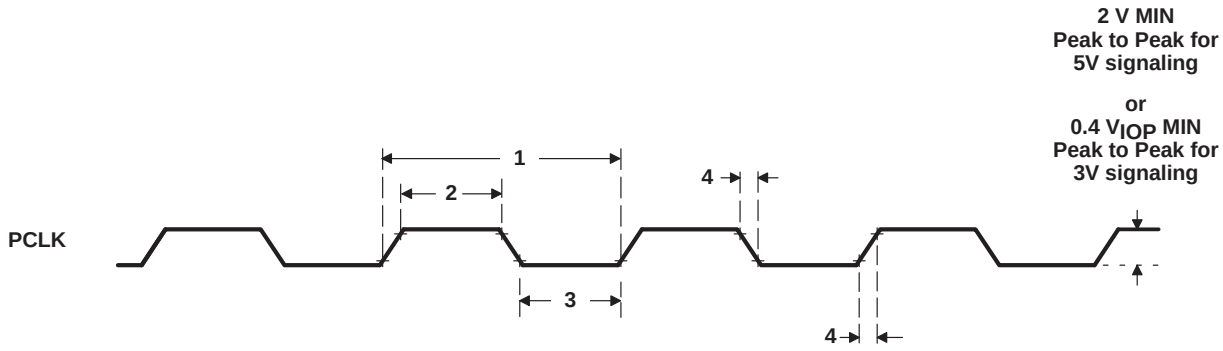



**Figure 13. PCLK Timings**

## INPUT AND OUTPUT CLOCKS (CONTINUED)

### switching characteristics over recommended operating conditions for CLKOUT2†‡ (see Figure 14)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 2 | $t_{w(CKO2H)}$ | Pulse duration, CLKOUT2 high | P − 0.7 | P + 0.7 | ns |
| 3 | $t_{w(CKO2L)}$ | Pulse duration, CLKOUT2 low | P − 0.7 | P + 0.7 | ns |
| 4 | $t_{t(CKO2)}$ | Transition time, CLKOUT2 | | 0.6 | ns |

† The reference points for the rise and fall transitions are measured at $V_{OL}$ MAX and $V_{OH}$ MIN.
‡ P = 1/CPU clock frequency in nanoseconds (ns).

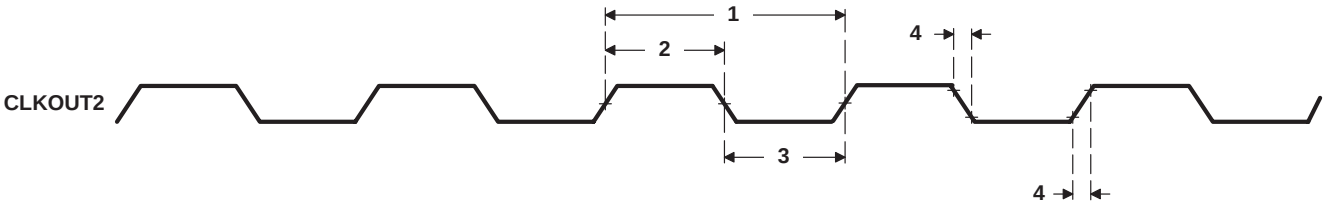



**Figure 14. CLKOUT2 Timings**

## ASYNCHRONOUS MEMORY TIMING

### timing requirements for asynchronous memory cycles[†‡§¶] (see Figure 15 − Figure 18)

| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 3 | $t_{su(EDV-AREH)}$ | Setup time, EDx valid before $\overline{ARE}$ high | 1.5 | | ns |
| 4 | $t_{h(AREH-EDV)}$ | Hold time, EDx valid after $\overline{ARE}$ high | 3.5 | | ns |
| 6 | $t_{su(ARDYH-AREL)}$ | Setup time, ARDY high before $\overline{ARE}$ low | −[(RST − 3) * P − 6] | | ns |
| 7 | $t_{h(AREL-ARDYH)}$ | Hold time, ARDY high after $\overline{ARE}$ low | (RST − 3) * P + 3 | | ns |
| 9 | $t_{su(ARDYL-AREL)}$ | Setup time, ARDY low before $\overline{ARE}$ low | −[(RST − 3) * P − 6] | | ns |
| 10 | $t_{h(AREL-ARDYL)}$ | Hold time, ARDY low after $\overline{ARE}$ low | (RST − 3) * P + 3 | | ns |
| 11 | $t_{w(ARDYH)}$ | Pulse width, ARDY high | 2P | | ns |
| 15 | $t_{su(ARDYH-AWEL)}$ | Setup time, ARDY high before $\overline{AWE}$ low | −[(WST − 3) * P − 6] | | ns |
| 16 | $t_{h(AWEL-ARDYH)}$ | Hold time, ARDY high after $\overline{AWE}$ low | (WST − 3) * P + 3 | | ns |
| 18 | $t_{su(ARDYL-AWEL)}$ | Setup time, ARDY low before $\overline{AWE}$ low | −[(WST − 3) * P − 6] | | ns |
| 19 | $t_{h(AWEL-ARDYL)}$ | Hold time, ARDY low after $\overline{AWE}$ low | (WST − 3) * P + 3 | | ns |

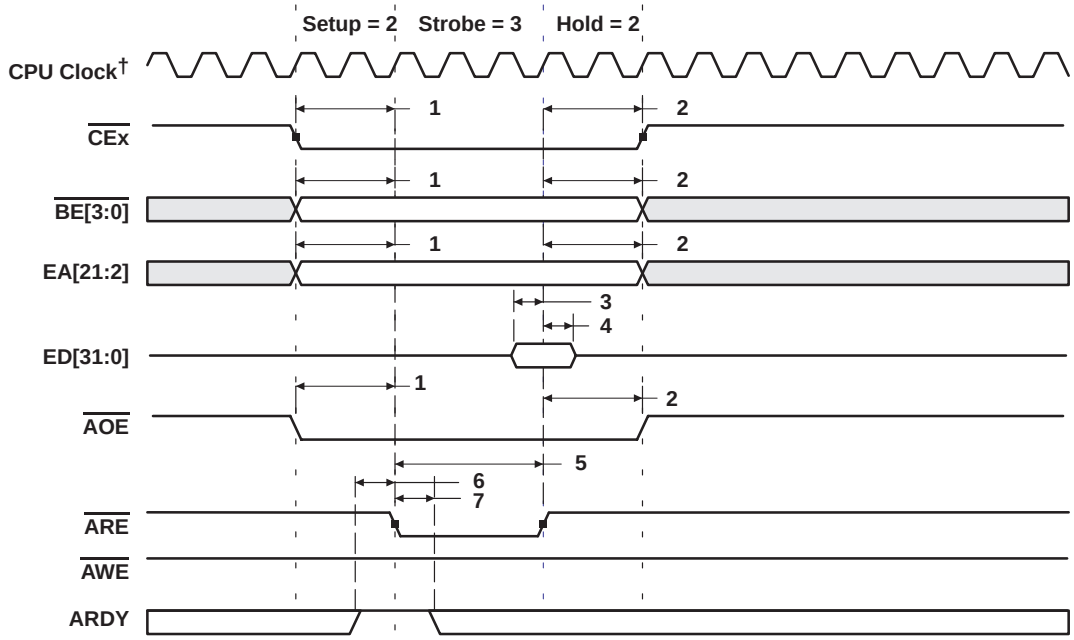
† To ensure data setup time, simply program the strobe width wide enough. ARDY is internally synchronized. If ARDY does meet setup or hold time, it may be recognized in the current cycle or the next cycle. Thus, ARDY can be an asynchronous input.
‡ RS = Read Setup, RST = Read Strobe, RH = Read Hold, WS = Write Setup, WST = Write Strobe, WH = Write Hold. These parameters are programmed via the EMIF CE space control registers.
§ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
¶ The sum of RS and RST (or WS and WST) must be a minimum of 4 in order to use ARDY input to extend strobe width.

### switching characteristics over recommended operating conditions for asynchronous memory cycles[‡§¶#] (see Figure 15 − Figure 18)

| NO. | PARAMETER | | −200 | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP | MAX | |
| 1 | $t_{osu(SELV-AREL)}$ | Output setup time, select signals valid to $\overline{ARE}$ low | RS * P − 2 | | | ns |
| 2 | $t_{oh(AREH-SELIV)}$ | Output hold time, $\overline{ARE}$ high to select signals invalid | RH * P − 2 | | | ns |
| 5 | $t_{w(AREL)}$ | Pulse width, $\overline{ARE}$ low | | RST * P | | ns |
| 8 | $t_{d(ARDYH-AREH)}$ | Delay time, ARDY high to $\overline{ARE}$ high | 3P | | 4P + 5 | ns |
| 12 | $t_{osu(SELV-AWEL)}$ | Output setup time, select signals valid to $\overline{AWE}$ low | WS * P − 2 | | | ns |
| 13 | $t_{oh(AWEH-SELIV)}$ | Output hold time, $\overline{AWE}$ high to select signals invalid | WH * P − 2 | | | ns |
| 14 | $t_{w(AWEL)}$ | Pulse width, $\overline{AWE}$ low | | WST * P | | ns |
| 17 | $t_{d(ARDYH-AWEH)}$ | Delay time, ARDY high to $\overline{AWE}$ high | 3P | | 4P + 5 | ns |

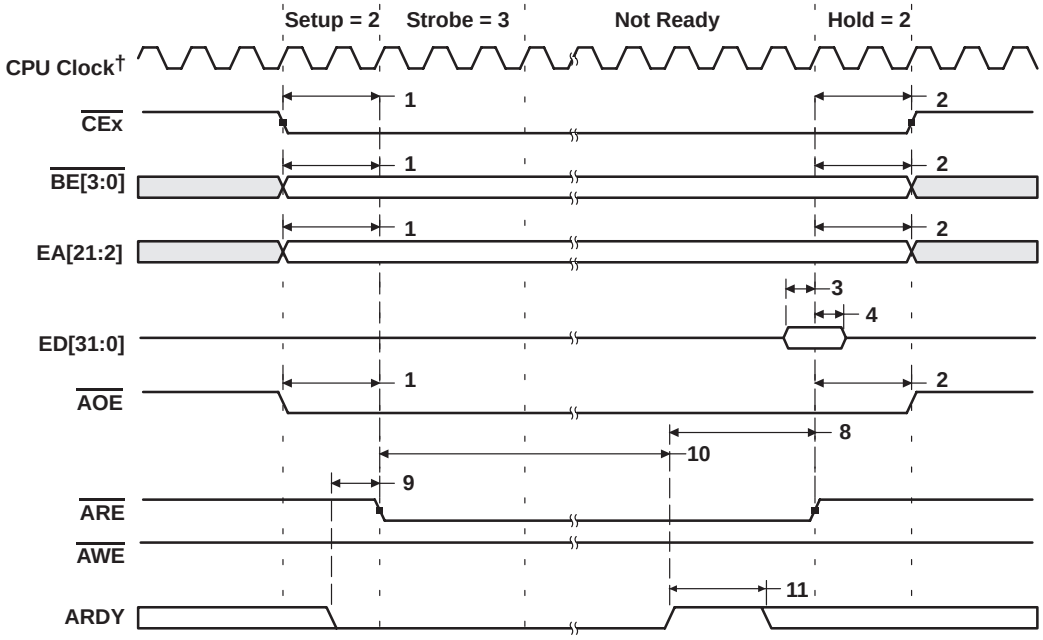
‡ RS = Read Setup, RST = Read Strobe, RH = Read Hold, WS = Write Setup, WST = Write Strobe, WH = Write Hold. These parameters are programmed via the EMIF CE space control registers.
§ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
¶ The sum of RS and RST (or WS and WST) must be a minimum of 4 in order to use ARDY input to extend strobe width.
# Select signals include: $\overline{CEx}$, $\overline{BE[3:0]}$, EA[21:2], $\overline{AOE}$; and for writes, include ED[31:0], with the exception that $\overline{CEx}$ can stay active for an additional 7P ns following the end of the cycle.

## ASYNCHRONOUS MEMORY TIMING (CONTINUED)




† CPU clock is an internal signal.

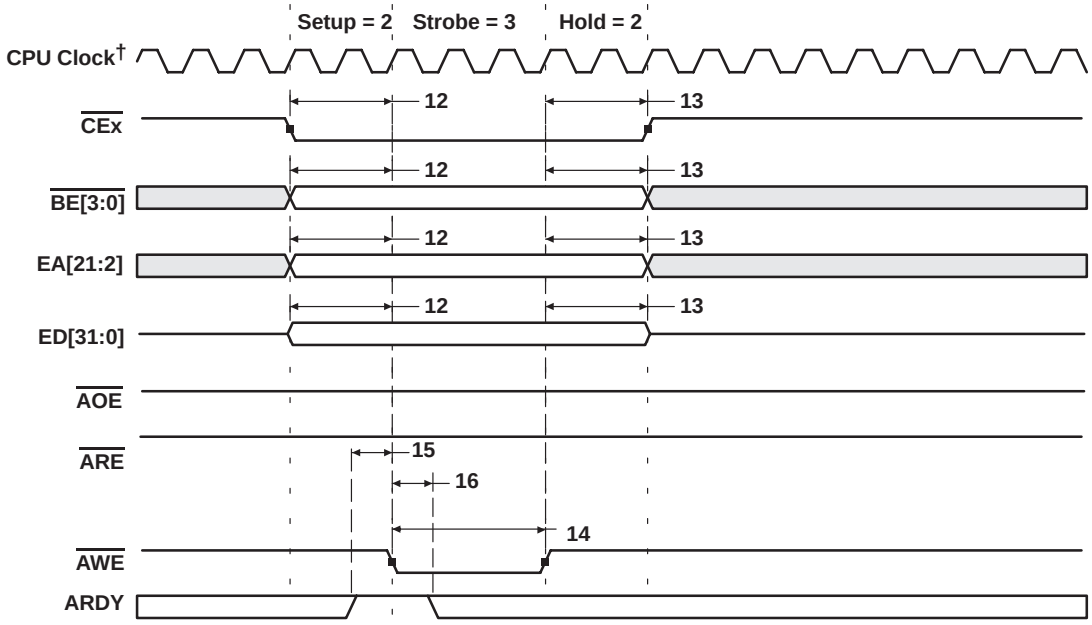
**Figure 15. Asynchronous Memory Read Timing (ARDY Not Used)**




† CPU clock is an internal signal.

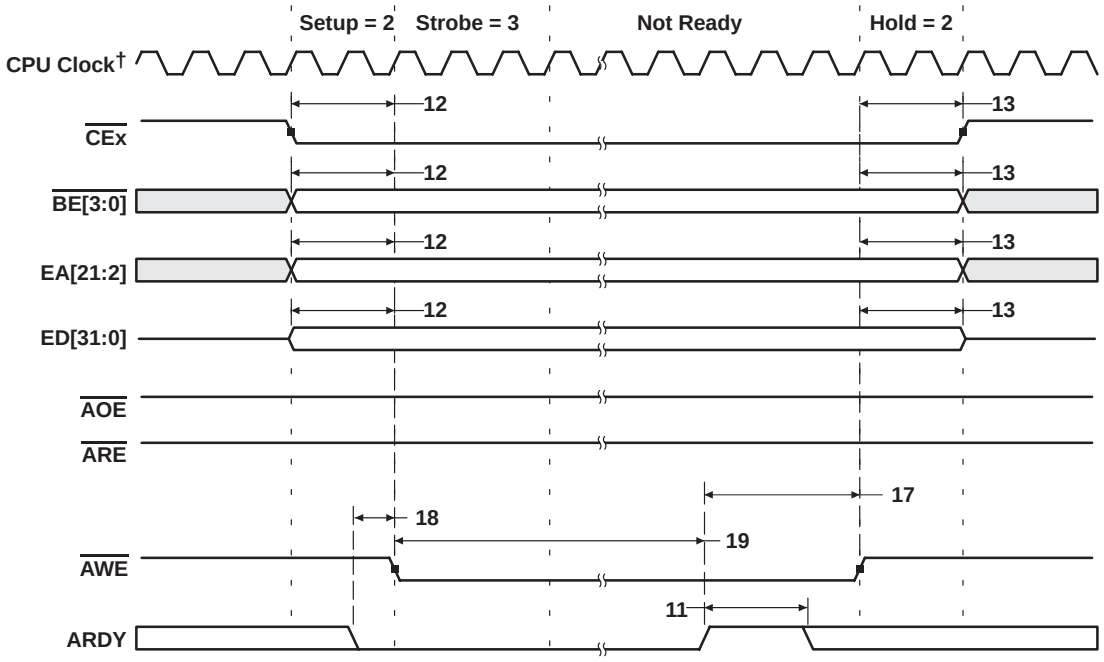
**Figure 16. Asynchronous Memory Read Timing (ARDY Used)**

## ASYNCHRONOUS MEMORY TIMING (CONTINUED)




† CPU clock is an internal signal.

**Figure 17. Asynchronous Memory Write Timing (ARDY Not Used)**




† CPU clock is an internal signal.

**Figure 18. Asynchronous Memory Write Timing (ARDY Used)**

## SYNCHRONOUS-BURST MEMORY TIMING

### timing requirements for synchronous-burst SRAM cycles (see Figure 19)

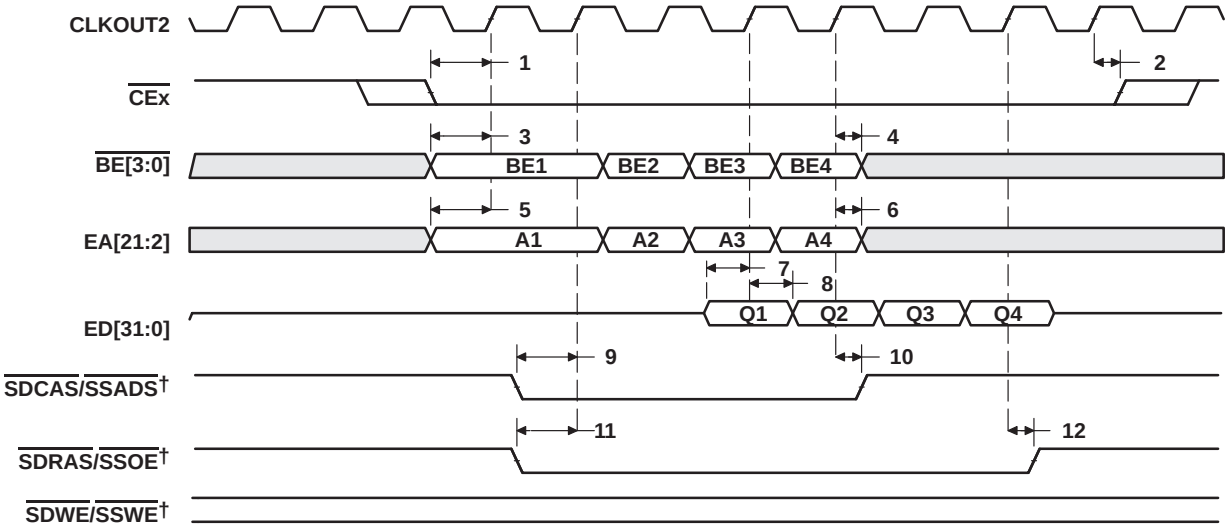
| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 7 | $t_{su(EDV-CKO2H)}$ | Setup time, read EDx valid before CLKOUT2 high | 2.5 | | ns |
| 8 | $t_{h(CKO2H-EDV)}$ | Hold time, read EDx valid after CLKOUT2 high | 1.5 | | ns |

### switching characteristics over recommended operating conditions for synchronous-burst SRAM cycles†‡ (see Figure 19 and Figure 20)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{osu(CEV-CKO2H)}$ | Output setup time, $\overline{CEx}$ valid before CLKOUT2 high | P − 0.8 | | ns |
| 2 | $t_{oh(CKO2H-CEV)}$ | Output hold time, $\overline{CEx}$ valid after CLKOUT2 high | P − 4 | | ns |
| 3 | $t_{osu(BEV-CKO2H)}$ | Output setup time, $\overline{BEx}$ valid before CLKOUT2 high | P − 0.8 | | ns |
| 4 | $t_{oh(CKO2H-BEIV)}$ | Output hold time, $\overline{BEx}$ invalid after CLKOUT2 high | P − 4 | | ns |
| 5 | $t_{osu(EAV-CKO2H)}$ | Output setup time, EAx valid before CLKOUT2 high | P − 0.8 | | ns |
| 6 | $t_{oh(CKO2H-EAIV)}$ | Output hold time, EAx invalid after CLKOUT2 high | P − 4 | | ns |
| 9 | $t_{osu(ADSV-CKO2H)}$ | Output setup time, $\overline{SDCAS}$/$\overline{SSADS}$ valid before CLKOUT2 high | P − 0.8 | | ns |
| 10 | $t_{oh(CKO2H-ADSV)}$ | Output hold time, $\overline{SDCAS}$/$\overline{SSADS}$ valid after CLKOUT2 high | P − 4 | | ns |
| 11 | $t_{osu(OEV-CKO2H)}$ | Output setup time, $\overline{SDRAS}$/$\overline{SSOE}$ valid before CLKOUT2 high | P − 0.8 | | ns |
| 12 | $t_{oh(CKO2H-OEV)}$ | Output hold time, $\overline{SDRAS}$/$\overline{SSOE}$ valid after CLKOUT2 high | P − 4 | | ns |
| 13 | $t_{osu(EDV-CKO2H)}$ | Output setup time, EDx valid before CLKOUT2 high§ | P − 1 | | ns |
| 14 | $t_{oh(CKO2H-EDIV)}$ | Output hold time, EDx invalid after CLKOUT2 high | P − 4 | | ns |
| 15 | $t_{osu(WEV-CKO2H)}$ | Output setup time, $\overline{SDWE}$/$\overline{SSWE}$ valid before CLKOUT2 high | P − 0.8 | | ns |
| 16 | $t_{oh(CKO2H-WEV)}$ | Output hold time, $\overline{SDWE}$/$\overline{SSWE}$ valid after CLKOUT2 high | P − 4 | | ns |

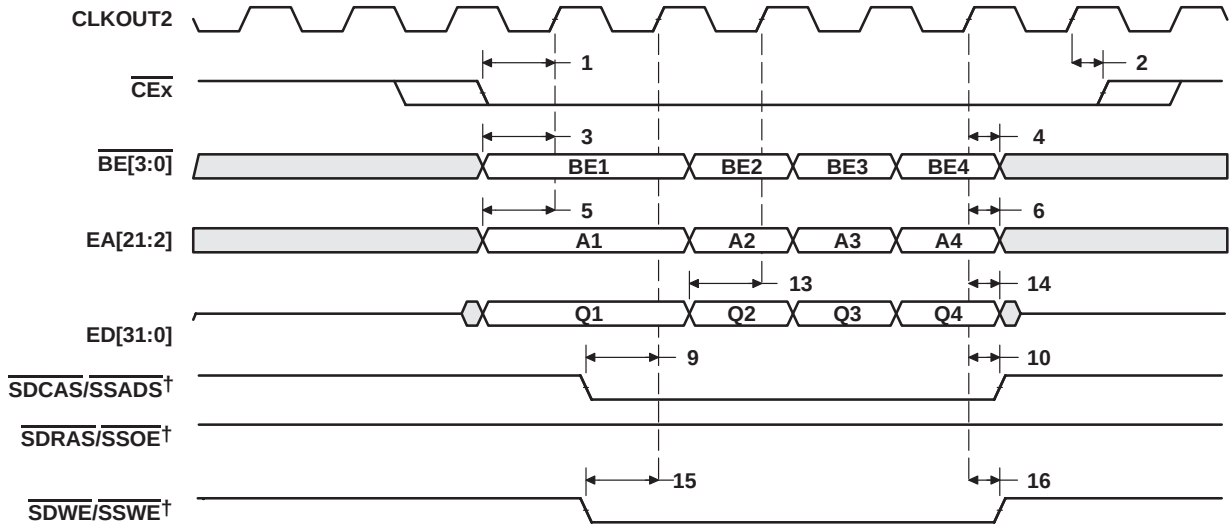
† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ $\overline{SDCAS}$/$\overline{SSADS}$, $\overline{SDRAS}$/$\overline{SSOE}$, and $\overline{SDWE}$/$\overline{SSWE}$ operate as $\overline{SSADS}$, $\overline{SSOE}$, and $\overline{SSWE}$, respectively, during SBSRAM accesses.
§ For the first write in a series of one or more consecutive adjacent writes, the write data is generated one CLKOUT2 cycle early to accommodate the ED enable time.

## SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)




† SDCAS/SSADS, SDRAS/SSOE, and SDWE/SSWE operate as SSADS, SSOE, and SSWE, respectively, during SBSRAM accesses.

**Figure 19. SBSRAM Read Timing**




† SDCAS/SSADS, SDRAS/SSOE, and SDWE/SSWE operate as SSADS, SSOE, and SSWE, respectively, during SBSRAM accesses.

**Figure 20. SBSRAM Write Timing**

## SYNCHRONOUS DRAM TIMING

### timing requirements for synchronous DRAM cycles (see Figure 21)

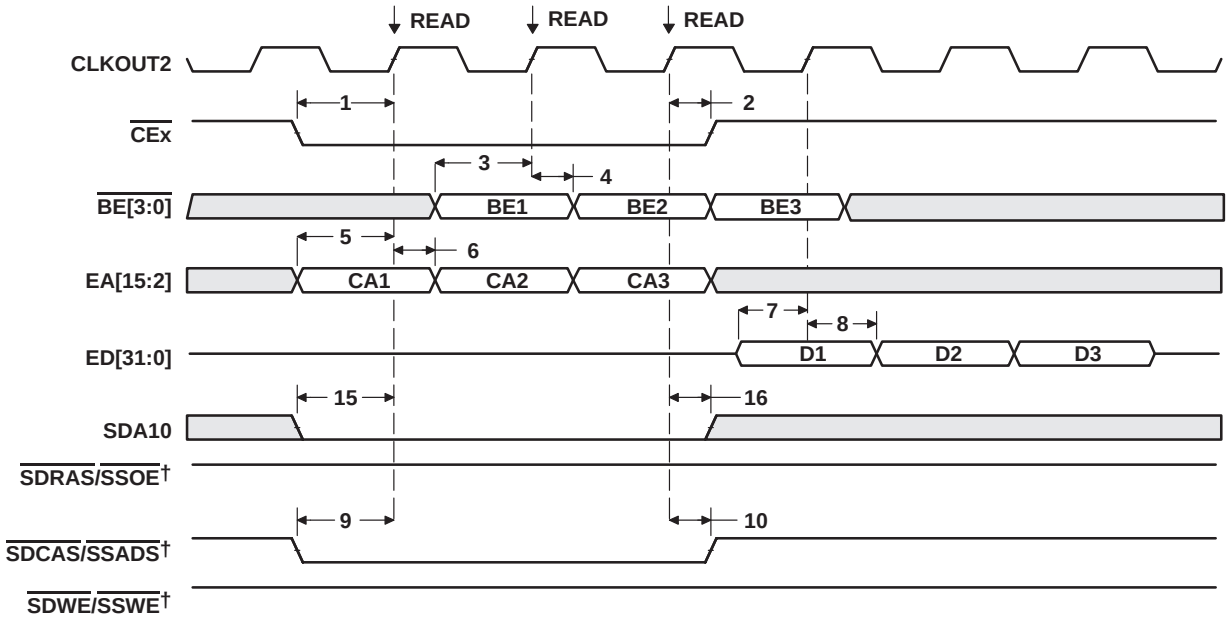
| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 7 | $t_{su(EDV-CKO2H)}$ | Setup time, read EDx valid before CLKOUT2 high | 1.25 | | ns |
| 8 | $t_{h(CKO2H-EDV)}$ | Hold time, read EDx valid after CLKOUT2 high | 3 | | ns |

### switching characteristics over recommended operating conditions for synchronous DRAM cycles†‡ (see Figure 21−Figure 26)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{osu(CEV-CKO2H)}$ | Output setup time, $\overline{CEx}$ valid before CLKOUT2 high | P − 1 | | ns |
| 2 | $t_{oh(CKO2H-CEV)}$ | Output hold time, $\overline{CEx}$ valid after CLKOUT2 high | P − 3.5 | | ns |
| 3 | $t_{osu(BEV-CKO2H)}$ | Output setup time, $\overline{BEx}$ valid before CLKOUT2 high | P − 1 | | ns |
| 4 | $t_{oh(CKO2H-BEIV)}$ | Output hold time, $\overline{BEx}$ invalid after CLKOUT2 high | P − 3.5 | | ns |
| 5 | $t_{osu(EAV-CKO2H)}$ | Output setup time, EAx valid before CLKOUT2 high | P − 1 | | ns |
| 6 | $t_{oh(CKO2H-EAIV)}$ | Output hold time, EAx invalid after CLKOUT2 high | P − 3.5 | | ns |
| 9 | $t_{osu(CASV-CKO2H)}$ | Output setup time, $\overline{SDCAS}$/$\overline{SSADS}$ valid before CLKOUT2 high | P − 1 | | ns |
| 10 | $t_{oh(CKO2H-CASV)}$ | Output hold time, $\overline{SDCAS}$/$\overline{SSADS}$ valid after CLKOUT2 high | P − 3.5 | | ns |
| 11 | $t_{osu(EDV-CKO2H)}$ | Output setup time, EDx valid before CLKOUT2 high§ | P − 3 | | ns |
| 12 | $t_{oh(CKO2H-EDIV)}$ | Output hold time, EDx invalid after CLKOUT2 high | P − 3.5 | | ns |
| 13 | $t_{osu(WEV-CKO2H)}$ | Output setup time, $\overline{SDWE}$/$\overline{SSWE}$ valid before CLKOUT2 high | P − 1 | | ns |
| 14 | $t_{oh(CKO2H-WEV)}$ | Output hold time, $\overline{SDWE}$/$\overline{SSWE}$ valid after CLKOUT2 high | P − 3.5 | | ns |
| 15 | $t_{osu(SDA10V-CKO2H)}$ | Output setup time, SDA10 valid before CLKOUT2 high | P − 1 | | ns |
| 16 | $t_{oh(CKO2H-SDA10IV)}$ | Output hold time, SDA10 invalid after CLKOUT2 high | P − 3.5 | | ns |
| 17 | $t_{osu(RASV-CKO2H)}$ | Output setup time, $\overline{SDRAS}$/$\overline{SSOE}$ valid before CLKOUT2 high | P − 1 | | ns |
| 18 | $t_{oh(CKO2H-RASV)}$ | Output hold time, $\overline{SDRAS}$/$\overline{SSOE}$ valid after CLKOUT2 high | P − 3.5 | | ns |

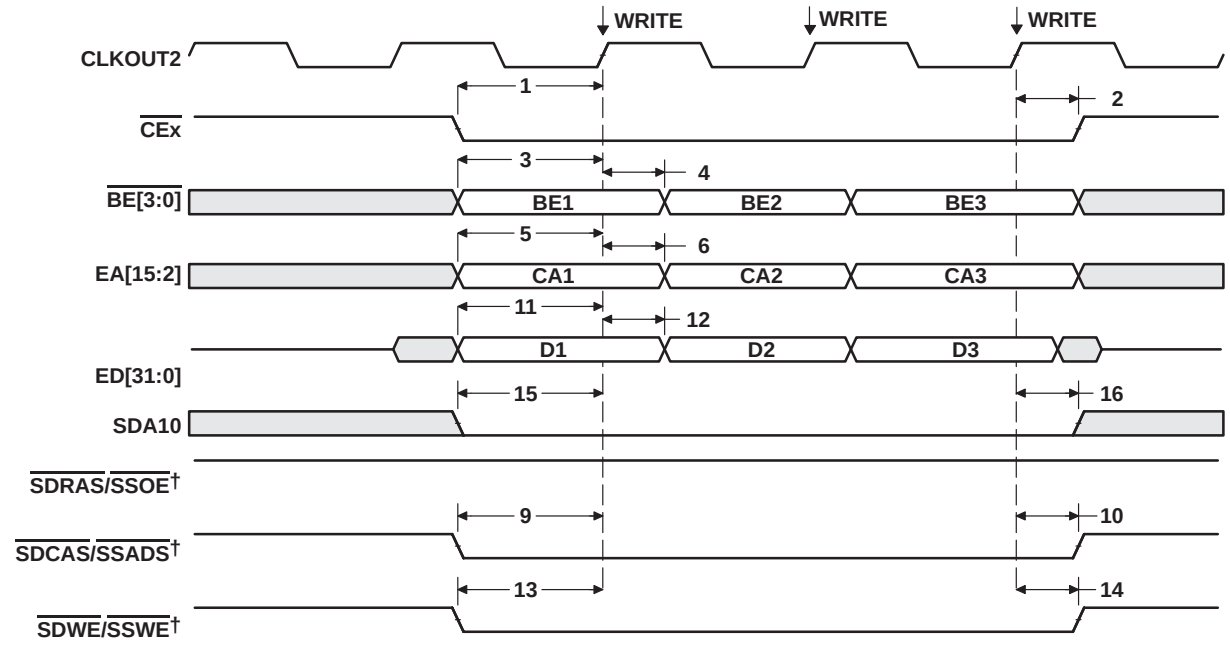
† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ $\overline{SDCAS}$/$\overline{SSADS}$, $\overline{SDRAS}$/$\overline{SSOE}$, and $\overline{SDWE}$/$\overline{SSWE}$ operate as $\overline{SDCAS}$, $\overline{SDRAS}$, and $\overline{SDWE}$, respectively, during SDRAM accesses.
§ For the first write in a series of one or more consecutive adjacent writes, the write data is generated one CLKOUT2 cycle early to accommodate the ED enable time.

## SYNCHRONOUS DRAM TIMING (CONTINUED)




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

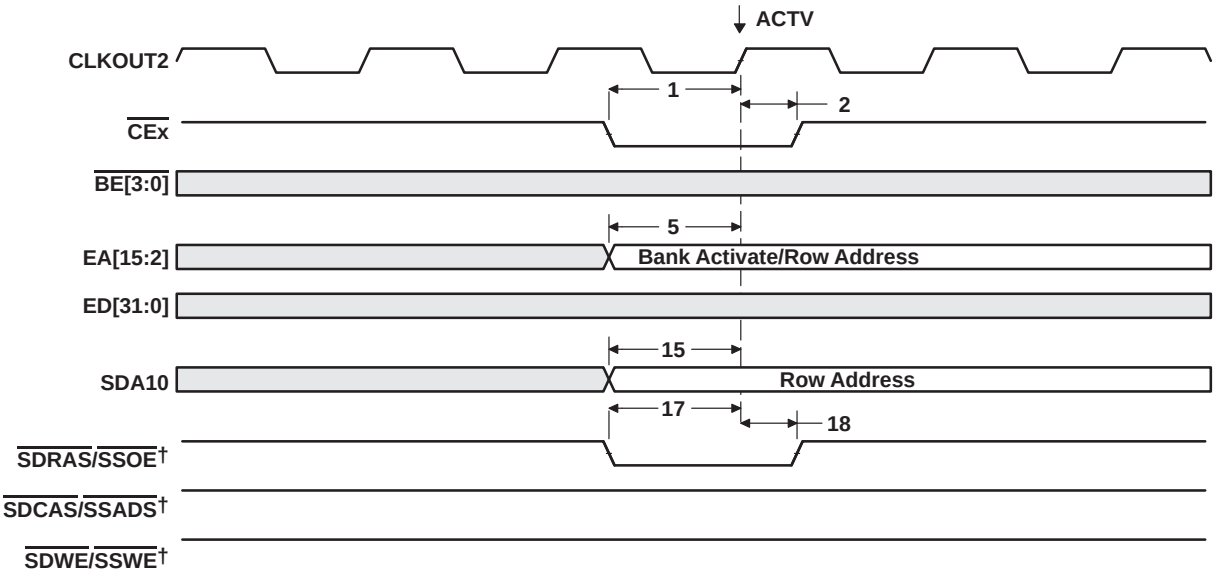
**Figure 21. Three SDRAM READ Commands**




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

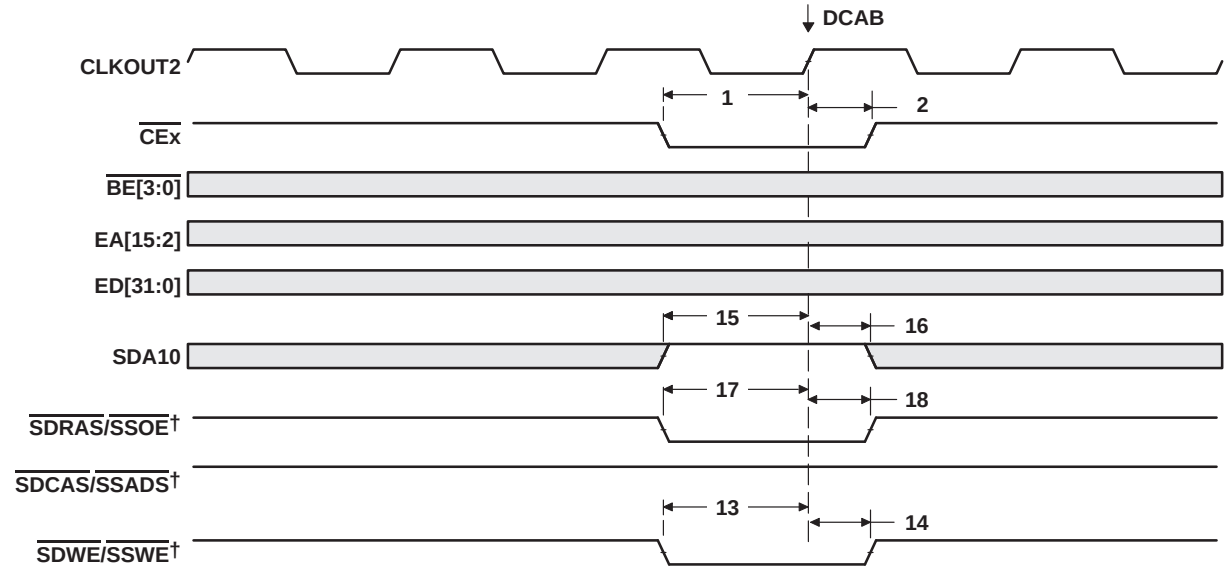
**Figure 22. Three SDRAM WRT Commands**

## SYNCHRONOUS DRAM TIMING (CONTINUED)




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

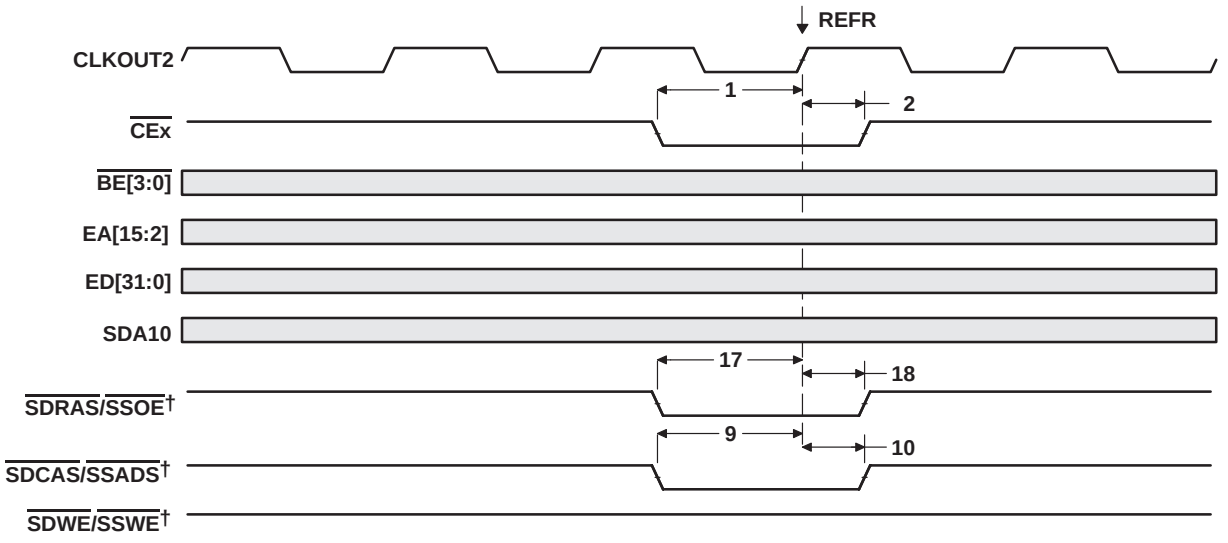
**Figure 23. SDRAM ACTV Command**




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

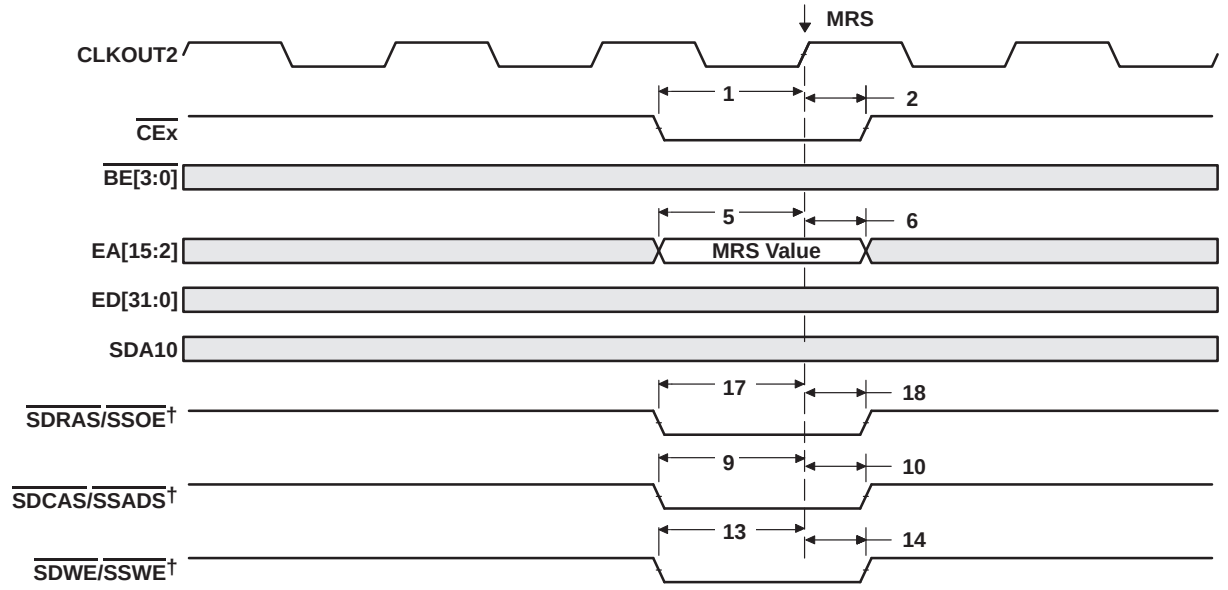
**Figure 24. SDRAM DCAB Command**

## SYNCHRONOUS DRAM TIMING (CONTINUED)




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

**Figure 25. SDRAM REFR Command**




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

**Figure 26. SDRAM MRS Command**

## HOLD/HOLDA TIMING

### timing requirements for the HOLD/HOLDA cycles† (see Figure 27)

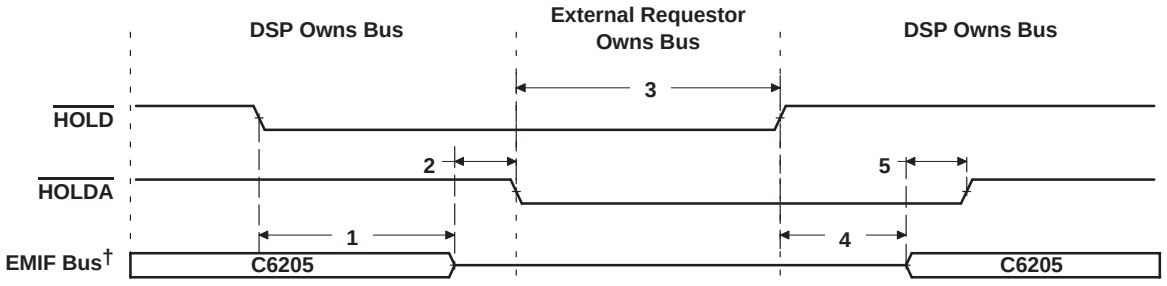
| NO. | | −200 | | UNIT |
|---|---|---|---|---|
| | | MIN | MAX | |
| 3 | $t_{oh(HOLDAL-HOLDL)}$ Output hold time, HOLD low after HOLDA low | P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

### switching characteristics over recommended operating conditions for the HOLD/HOLDA cycles†‡ (see Figure 27)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{d(HOLDL-EMHZ)}$ | Delay time, HOLD low to EMIF Bus high impedance | 4P | § | ns |
| 2 | $t_{d(EMHZ-HOLDAL)}$ | Delay time, EMIF Bus high impedance to HOLDA low | 0 | 2P | ns |
| 4 | $t_{d(HOLDH-EMLZ)}$ | Delay time, HOLD high to EMIF Bus low impedance | 3P | 7P | ns |
| 5 | $t_{d(EMLZ-HOLDAH)}$ | Delay time, EMIF Bus low impedance to HOLDA high | 0 | 2P | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ EMIF Bus consists of CE[3:0], BE[3:0], ED[31:0], EA[21:2], ARE, AOE, AWE, SDCAS/SSADS, SDRAS/SSOE, SDWE/SSWE, and SDA10.
§ All pending EMIF transactions are allowed to complete before HOLDA is asserted. The worst case for this is an asynchronous read or write with external ARDY used or a minimum of eight consecutive SDRAM reads or writes when RBTR8 = 1. 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[21:2], ARE, AOE, AWE, SDCAS/SSADS, SDRAS/SSOE, SDWE/SSWE, and SDA10.

**Figure 27. HOLD/HOLDA Timing**

## RESET TIMING

### timing requirements for reset (see Figure 28)

| NO. | | | −200 | | 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 |
| 10 | $t_{su(ED)}$ | Setup time, ED boot configuration bits valid before RESET high¶ | 5P‡# | | ns |
| 11 | $t_{h(ED)}$ | Hold time, ED boot configuration bits valid after RESET high¶ | 5P‡ | | ns |

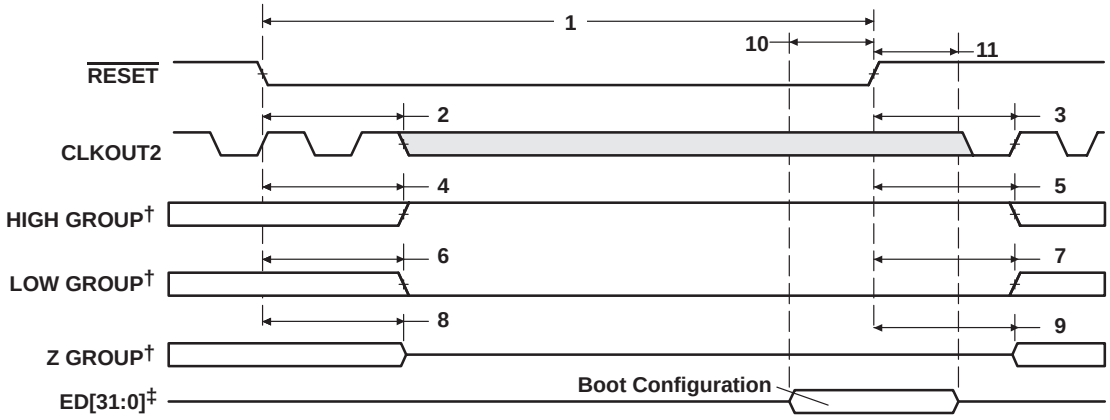
† This parameter applies to CLKMODE x1 when CLKIN is stable, and applies to CLKMODE x4, x6, x7, x8, x9, x10, and x11 when CLKIN and PLL are stable.
‡ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
§ This parameter applies to CLKMODE x4, x6, x7, x8, x9, x10, and x11 only. The RESET signal is not connected internally to the Clock PLL circuit. The PLL requires a minimum of 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 power up (specifically Figure 5, Note E) and for PLL lock times (Table 4).
¶ ED[31:0] are the boot configuration pins during device reset.
# A 250 µs setup time before the rising edge of RESET is required when using CLKMODE x4, x6, x7, x8, x9, x10, or x11.

### switching characteristics over recommended operating conditions during reset‡‖ (see Figure 28)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 2 | $t_{d(RSTL-CKO2IV)}$ | Delay time, RESET low to CLKOUT2 invalid | P | | ns |
| 3 | $t_{d(RSTH-CKO2V)}$ | Delay time, RESET high to CLKOUT2 valid | | 4P | ns |
| 4 | $t_{d(RSTL-HIGHIV)}$ | Delay time, RESET low to high group invalid | P | | ns |
| 5 | $t_{d(RSTH-HIGHV)}$ | Delay time, RESET high to high group valid | | 4P | ns |
| 6 | $t_{d(RSTL-LOWIV)}$ | Delay time, RESET low to low group invalid | P | | ns |
| 7 | $t_{d(RSTH-LOWV)}$ | Delay time, RESET high to low group valid | | 4P | ns |
| 8 | $t_{d(RSTL-ZHZ)}$ | Delay time, RESET low to Z group high impedance | P | | ns |
| 9 | $t_{d(RSTH-ZV)}$ | Delay time, RESET high to Z group valid | | 4P | ns |

‡ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‖ High group consists of: HOLDA
Low group consists of: IACK, INUM[3:0], DMAC[3:0], PD, TOUT0, and TOUT1, XSP_CLK, XSP_DO, and XSP_CS
Z group consists of: EA[21:2], ED[31:0], CE[3:0], BE[3:0], ARE, AWE, AOE, SDCAS/SSADS, SDRAS/SSOE, SDWE/SSWE, SDA10, CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKR0, CLKR1, FSR0, FSR1, AD[31:0], PCBE[3:0], PINTA, PREQ, PSERR, PPERR, PDEVSEL, PFRAME, PIRDY, PPAR, PSTOP, PTRDY, and PME

## RESET TIMING (CONTINUED)




† High group consists of: HOLDA
Low group consists of: IACK, INUM[3:0], DMAC[3:0], PD, TOUT0, and TOUT1, XSP_CLK, XSP_DO, and XSP_CS
Z group consists of: EA[21:2], ED[31:0], CE[3:0], BE[3:0], ARE, AWE, AOE, SDCAS/SSADS, SDRAS/SSOE, SDWE/SSWE, SDA10, CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKR0, CLKR1, FSR0, FSR1, AD[31:0], PCBE[3:0], PINTA, PREQ, PSERR, PPERR, PDEVSEL, PFRAME, PIRDY, PPAR, PSTOP, PTRDY, and PME

‡ ED[31:0] are the boot configuration pins during device reset.

**Figure 28. Reset Timing**

## EXTERNAL INTERRUPT TIMING

### timing requirements for interrupt response cycles† (see Figure 29)

| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 2 | $t_{w(ILOW)}$ | Width of the interrupt pulse low | 2P | | ns |
| 3 | $t_{w(IHIGH)}$ | Width of the interrupt pulse high | 2P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

### switching characteristics over recommended operating conditions during interrupt response cycles† (see Figure 29)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{R(EINTH - IACKH)}$ | Response time, EXT_INTx high to IACK high | 9P | | ns |
| 4 | $t_{d(CKO2L-IACKV)}$ | Delay time, CLKOUT2 low to IACK valid | 0 | 10 | ns |
| 5 | $t_{d(CKO2L-INUMV)}$ | Delay time, CLKOUT2 low to INUMx valid | 0 | 10 | ns |
| 6 | $t_{d(CKO2L-INUMIV)}$ | Delay time, CLKOUT2 low to INUMx invalid | 0 | 10 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

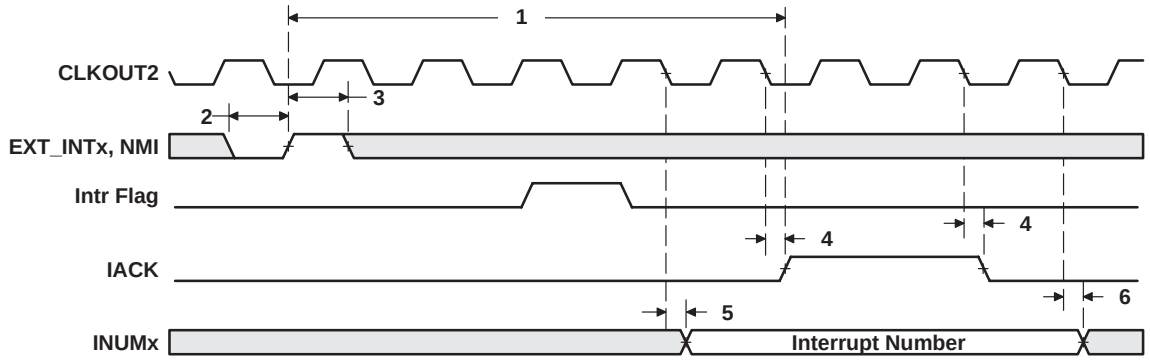



**Figure 29. Interrupt Timing**

## PCI I/O TIMINGS

### timing requirements for PCI inputs (see Figure 30)

| NO. | | | −200 | | 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 30)

| NO. | PARAMETER | | −200 | | 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 |

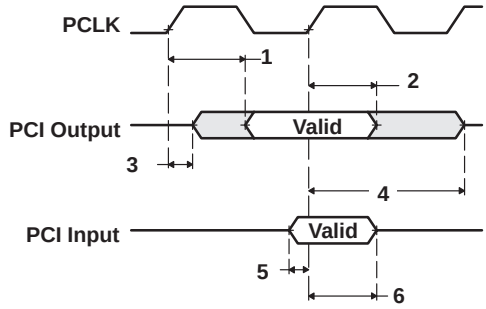



**Figure 30. PCI Intput/Output Timings**

## PCI RESET TIMING

### timing requirements for PCI reset (see Figure 31)

| NO. | | | −200 | | 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 |

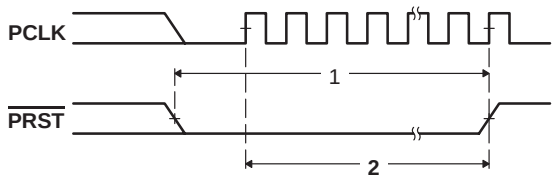



**Figure 31. PCI Reset ($\overline{PRST}$) Timings**

## PCI SERIAL EEPROM INTERFACE TIMING

### timing requirements for serial EEPROM interface (see Figure 32)

| NO. | | | −200 | | 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 32)

| NO. | PARAMETER | | −200 | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | NOM | MAX | |
| 1 | $t_{w(CSL)}$ | Pulse duration, XSP_CS low | | 2046P | | 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 | | 1023P | | ns |
| 4 | $t_{w(CLKH)}$ | Pulse duration, XSP_CLK high | | 1023P | | ns |
| 5 | $t_{w(CLKL)}$ | Pulse duration, XSP_CLK low | | 1023P | | ns |
| 6 | $t_{osu(DOV-CLKH)}$ | Output setup time, XSP_DO valid after XSP_CLK high | | 1023P | | ns |
| 7 | $t_{oh(CLKH-DOV)}$ | Output hold time, XSP_DO valid after XSP_CLK high | | 1023P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

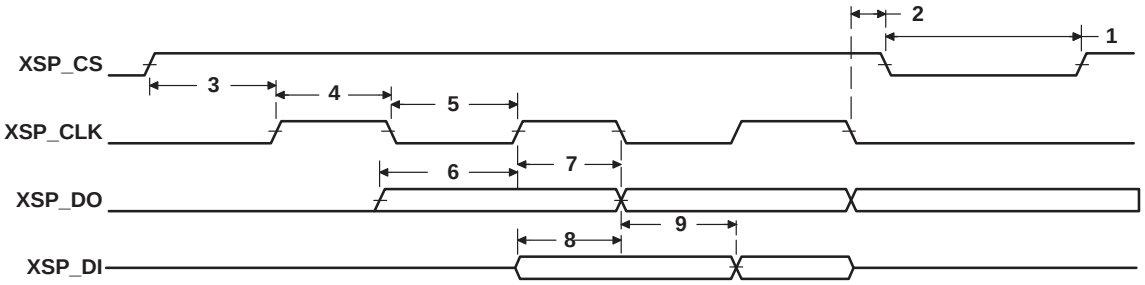



**Figure 32. PCI Serial EEPROM Interface Timing**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING

### timing requirements for McBSP†‡ (see Figure 33)

| NO. | | | | −200 | | UNIT |
|---|---|---|---|---|---|---|
| | | | | MIN | MAX | |
| 2 | $t_{c(CKRX)}$ | Cycle time, CLKR/X | CLKR/X ext | 2P§ | | ns |
| 3 | $t_{w(CKRX)}$ | Pulse duration, CLKR/X high or CLKR/X low | CLKR/X ext | P−1¶ | | ns |
| 5 | $t_{su(FRH-CKRL)}$ | Setup time, external FSR high before CLKR low | CLKR int | 9 | | ns |
| | | | CLKR ext | 2 | | |
| 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 | 0.5 | | |
| 8 | $t_{h(CKRL-DRV)}$ | Hold time, DR valid after CLKR low | CLKR int | 4 | | ns |
| | | | CLKR ext | 3 | | |
| 10 | $t_{su(FXH-CKXL)}$ | Setup time, external FSX high before CLKX low | CLKX int | 9 | | ns |
| | | | CLKX ext | 2 | | |
| 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 the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

‡ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

§ The maximum bit rate for the C6205 devices is 100 Mbps or CPU/2 (the slower of the two). Care must be taken to ensure that the AC timings specified in this data sheet are met. The maximum bit rate for McBSP-to-McBSP communications is 100 MHz; therefore, the minimum CLKR/X clock cycle is either twice the CPU cycle time (2P), or 10 ns (100 MHz), whichever value is larger. For example, when running parts at 200 MHz (P = 5 ns), use 10 ns as the minimum CLKR/X clock cycle (by setting the appropriate CLKGDV ratio or external clock source). When running parts at 100 MHz (P = 10 ns), use 2P = 20 ns (50 MHz) as the minimum CLKR/X clock cycle. The maximum bit rate for McBSP-to-McBSP communications applies when the serial port is a master of the clock and frame syncs (with CLKR connected to CLKX, FSR connected to FSX, CLKXM = FSXM = 1, and CLKRM = FSRM = 0) in data delay 1 or 2 mode (R/XDATDLY = 01b or 10b) and the other device the McBSP communicates to is a slave.

¶ The minimum CLKR/X pulse duration is either (P−1) or 4 ns, whichever is larger. For example, when running parts at 200 MHz (P = 5 ns), use 4 ns as the minimum CLKR/X pulse duration. When running parts at 100 MHz (P = 10 ns), use (P−1) = 9 ns as the minimum CLKR/X pulse duration.

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### switching characteristics over recommended operating conditions for McBSP†‡ (see Figure 33)

| NO. | PARAMETER | | | −200 | | UNIT |
|---|---|---|---|---|---|---|
| | | | | MIN | MAX | |
| 1 | $t_{d(CKSH-CKRXH)}$ | Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input | | 3 | 12 | ns |
| 2 | $t_{c(CKRX)}$ | Cycle time, CLKR/X | CLKR/X int | 2P−2§¶ | | ns |
| 3 | $t_{w(CKRX)}$ | Pulse duration, CLKR/X high or CLKR/X low | CLKR/X int | C − 2# | C + 2# | ns |
| 4 | $t_{d(CKRH-FRV)}$ | Delay time, CLKR high to internal FSR valid | CLKR int | −3 | 3 | ns |
| 9 | $t_{d(CKXH-FXV)}$ | Delay time, CLKX high to internal FSX valid | CLKX int | −3 | 3 | ns |
| | | | CLKX ext | 3 | 9 | |
| 12 | $t_{dis(CKXH-DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX high | CLKX int | −1 | 4 | ns |
| | | | CLKX ext | 3 | 9 | |
| 13 | $t_{d(CKXH-DXV)}$ | Delay time, CLKX high to DX valid | CLKX int | −1 | 4 | ns |
| | | | CLKX ext | 2 | 12 | |
| 14 | $t_{d(FXH-DXV)}$ | Delay time, FSX high to DX valid<br>ONLY applies when in data delay 0 (XDATDLY = 00b) mode. | FSX int | −1 | 5 | ns |
| | | | FSX ext | 2 | 12 | |

† CLKRP = CLKXP = FSRP = FSXP = 0. If the 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.

§ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

¶ The maximum bit rate for the C6205 devices is 100 Mbps or CPU/2 (the slower of the two). Care must be taken to ensure that the AC timings specified in this data sheet are met. The maximum bit rate for McBSP-to-McBSP communications is 100 MHz; therefore, the minimum CLKR/X clock cycle is either twice the CPU cycle time (2P), or 10 ns (100 MHz), whichever value is larger. For example, when running parts at 200 MHz (P = 5 ns), use 10 ns as the minimum CLKR/X clock cycle (by setting the appropriate CLKGDV ratio or external clock source). When running parts at 100 MHz (P = 10 ns), use 2P = 20 ns (50 MHz) as the minimum CLKR/X clock cycle. The maximum bit rate for McBSP-to-McBSP communications applies when the serial port is a master of the clock and frame syncs (with CLKR connected to CLKX, FSR connected to FSX, CLKXM = FSXM = 1, and CLKRM = FSRM = 0) in data delay 1 or 2 mode (R/XDATDLY = 01b or 10b) and the other device the McBSP communicates to is a slave.

# C = H or L
S = sample rate generator input clock = P 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 100-MHz limit.

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

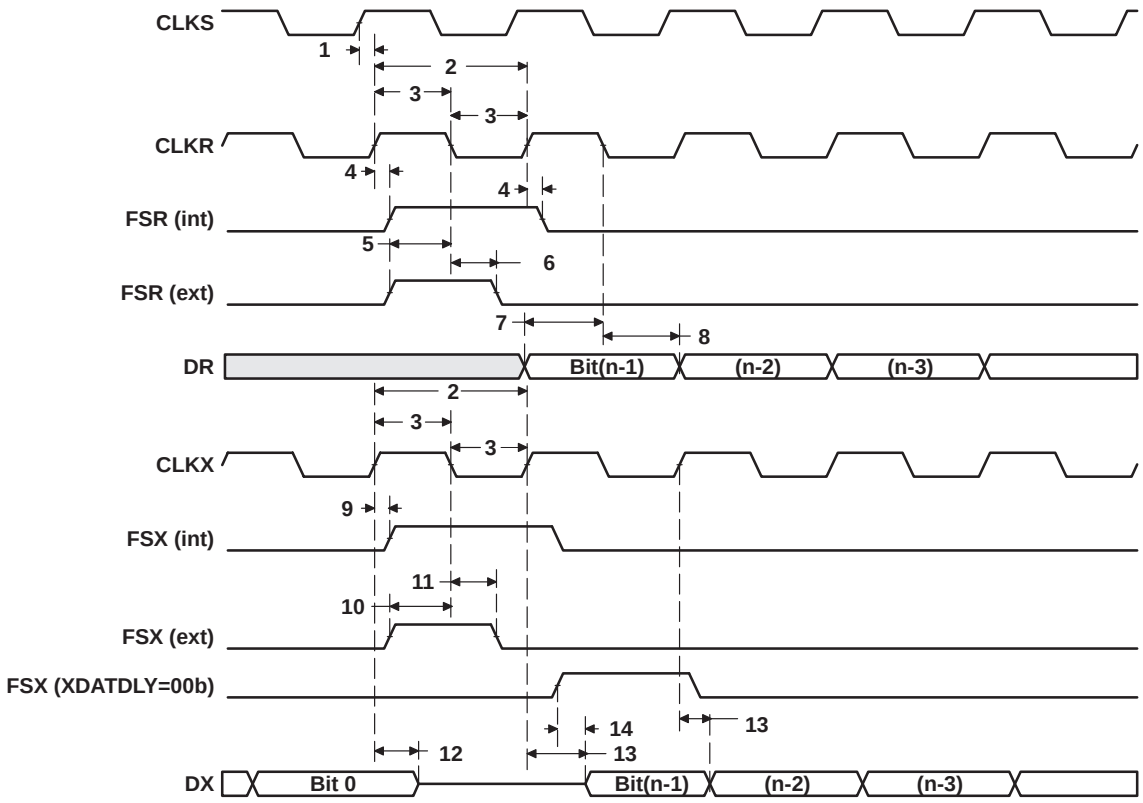



**Figure 33. McBSP Timings**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### timing requirements for FSR when GSYNC = 1 (see Figure 34)

| NO. | | | −200 | | 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 |

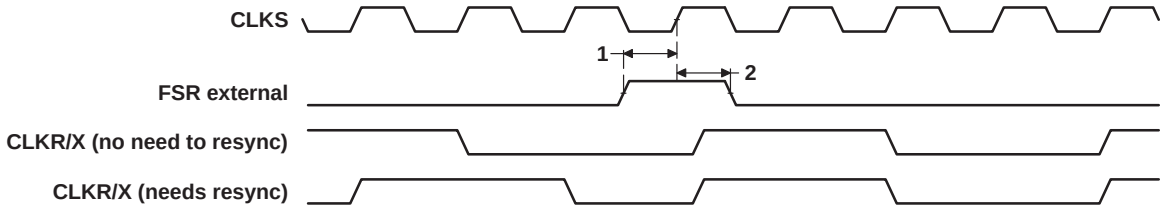



**Figure 34. FSR Timing When GSYNC = 1**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### timing requirements for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 0†‡ (see Figure 35)

| NO. | | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MASTER | | SLAVE | | |
| | | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su(DRV-CKXL)}$ | Setup time, DR valid before CLKX low | 12 | | 2 − 3P | | ns |
| 5 | $t_{h(CKXL-DRV)}$ | Hold time, DR valid after CLKX low | 4 | | 6 + 6P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 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 35)

| NO. | PARAMETER | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MASTER§ | | SLAVE | | |
| | | | MIN | MAX | MIN | MAX | |
| 1 | $t_{h(CKXL-FXL)}$ | Hold time, FSX low after CLKX low¶ | T − 3 | T + 5 | | | ns |
| 2 | $t_{d(FXL-CKXH)}$ | Delay time, FSX low to CLKX high# | L − 4 | L + 5 | | | ns |
| 3 | $t_{d(CKXH-DXV)}$ | Delay time, CLKX high to DX valid | −4 | 5 | 3P + 3 | 5P + 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 | | | P + 3 | 3P + 17 | ns |
| 8 | $t_{d(FXL-DXV)}$ | Delay time, FSX low to DX valid | | | 2P + 2 | 4P + 17 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = sample rate generator input clock = P 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 TIMING (CONTINUED)

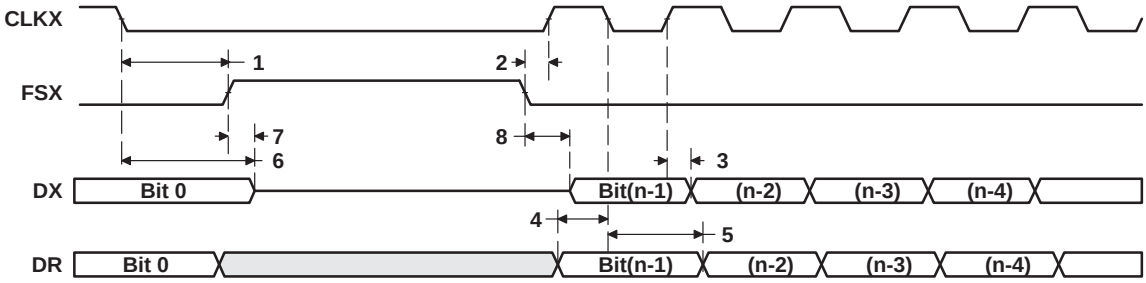



**Figure 35. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### timing requirements for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 0†‡ (see Figure 36)

| NO. | | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MASTER | | SLAVE | | |
| | | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su(DRV-CKXH)}$ | Setup time, DR valid before CLKX high | 12 | | 2 − 3P | | ns |
| 5 | $t_{h(CKXH-DRV)}$ | Hold time, DR valid after CLKX high | 4 | | 5 + 6P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 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 36)

| NO. | PARAMETER | | −200 | | | | 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 | 3P + 4 | 5P + 17 | ns |
| 6 | $t_{dis(CKXL-DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX low | −2 | 4 | 3P + 3 | 5P + 17 | ns |
| 7 | $t_{d(FXL-DXV)}$ | Delay time, FSX low to DX valid | H − 2 | H + 4 | 2P + 2 | 4P + 17 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = sample rate generator input clock = P 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).

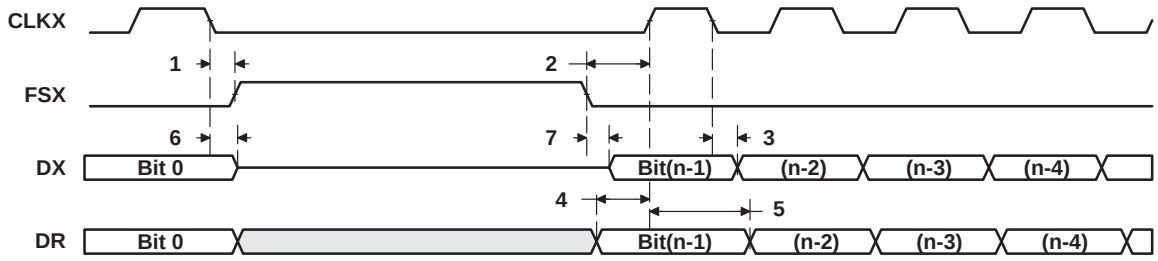



**Figure 36. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### timing requirements for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 1†‡ (see Figure 37)

| NO. | | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MASTER | | SLAVE | | |
| | | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su(DRV-CKXH)}$ | Setup time, DR valid before CLKX high | 12 | | 2 − 3P | | ns |
| 5 | $t_{h(CKXH-DRV)}$ | Hold time, DR valid after CLKX high | 4 | | 5 + 6P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 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 37)

| NO. | PARAMETER | | −200 | | | | 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 | 3P + 4 | 5P + 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 | | | P + 3 | 3P + 17 | ns |
| 8 | $t_{d(FXL-DXV)}$ | Delay time, FSX low to DX valid | | | 2P + 2 | 4P + 17 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = sample rate generator input clock = P 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 TIMING (CONTINUED)

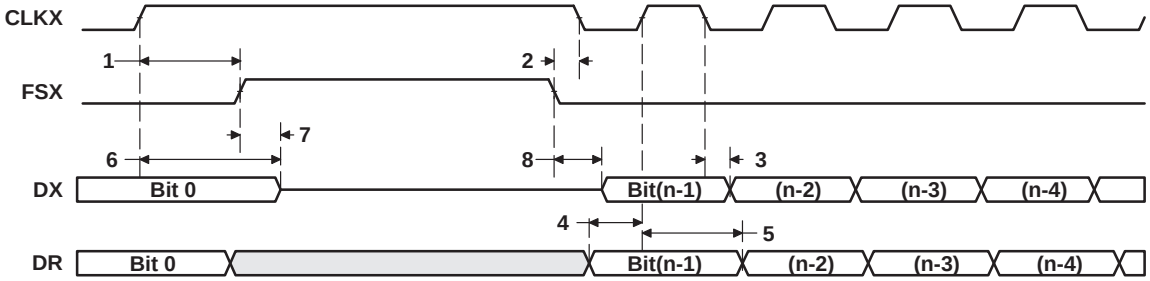



**Figure 37. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

### timing requirements for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 1†‡ (see Figure 38)

| NO. | | | −200 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MASTER | | SLAVE | | |
| | | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su(DRV-CKXL)}$ | Setup time, DR valid before CLKX low | 12 | | 2 − 3P | | ns |
| 5 | $t_{h(CKXL-DRV)}$ | Hold time, DR valid after CLKX low | 4 | | 5 + 6P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 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 38)

| NO. | PARAMETER | | −200 | | | | 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 | 3P + 4 | 5P + 17 | ns |
| 6 | $t_{dis(CKXH-DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX high | −2 | 4 | 3P + 3 | 5P + 17 | ns |
| 7 | $t_{d(FXL-DXV)}$ | Delay time, FSX low to DX valid | L − 2 | L + 4 | 2P + 2 | 4P + 17 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = sample rate generator input clock = P 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).

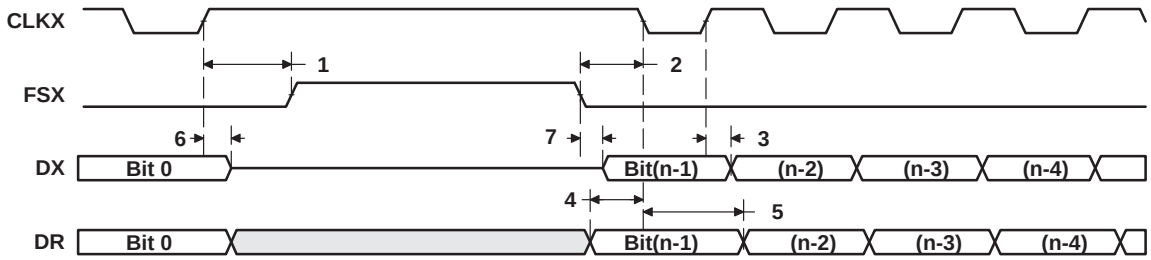



**Figure 38. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1**

## DMAC, TIMER, POWER-DOWN TIMING

### switching characteristics over recommended operating conditions for DMAC outputs† (see Figure 39)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{w(DMACH)}$ | Pulse duration, DMAC high | 2P−3 | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

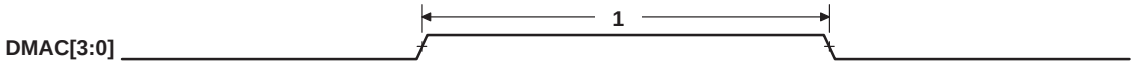



**Figure 39. DMAC Timing**

### timing requirements for timer inputs† (see Figure 40)

| NO. | | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{w(TINPH)}$ | Pulse duration, TINP high | 2P | | ns |
| 2 | $t_{w(TINPL)}$ | Pulse duration, TINP low | 2P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

### switching characteristics over recommended operating conditions for timer outputs† (see Figure 40)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 3 | $t_{w(TOUTH)}$ | Pulse duration, TOUT high | 2P−3 | | ns |
| 4 | $t_{w(TOUTL)}$ | Pulse duration, TOUT low | 2P−3 | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

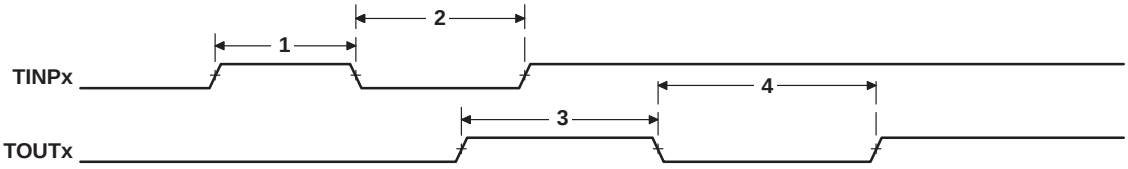



**Figure 40. Timer Timing**

## DMAC, TIMER, POWER-DOWN TIMING (CONTINUED)

### switching characteristics over recommended operating conditions for power-down outputs† (see Figure 41)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{w(PDH)}$ | Pulse duration, PD high | 2P | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

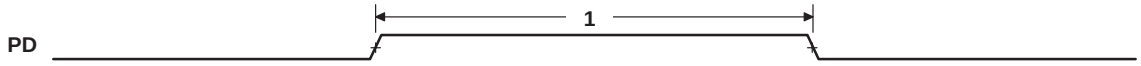



**Figure 41. Power-Down Timing**

## JTAG TEST-PORT TIMING

### timing requirements for JTAG test port (see Figure 42)

| NO. | | | −200 | | 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 | 11 | | 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 42)

| NO. | PARAMETER | | −200 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 2 | $t_{d(TCKL-TDOV)}$ | Delay time, TCK low to TDO valid | −4.5 | 12 | ns |

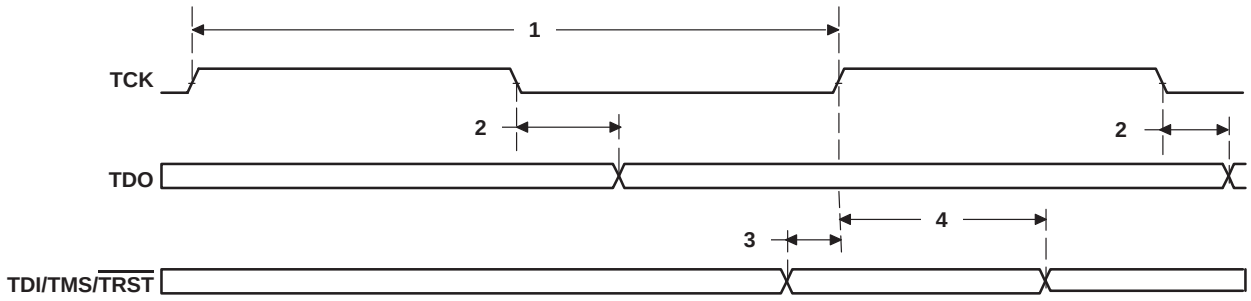



**Figure 42. JTAG Test-Port Timing**

## MECHANICAL DATA FOR TMS320C6205

The following table(s) show the thermal resistance characteristics for the S−PBGA mechanical package.

### thermal resistance characteristics (S-PBGA package) (GHK)

| NO | | | °C/W | Air Flow (m/s†) |
|---|---|---|---|---|
| 1 | $R\Theta_{JC}$ | Junction-to-case | 9.5 | N/A |
| 2 | $R\Theta_{JA}$ | Junction-to-free air | 26.5 | 0.00 |
| 3 | $R\Theta_{JA}$ | Junction-to-free air | 23.9 | 0.50 |
| 4 | $R\Theta_{JA}$ | Junction-to-free air | 22.6 | 1.00 |
| 5 | $R\Theta_{JA}$ | Junction-to-free air | 21.3 | 2.00 |

† m/s = meters per second

### thermal resistance characteristics (S-PBGA package) (ZHK)

| NO | | | °C/W | Air Flow (m/s†) |
|---|---|---|---|---|
| 1 | $R\Theta_{JC}$ | Junction-to-case | 9.5 | N/A |
| 2 | $R\Theta_{JA}$ | Junction-to-free air | 26.5 | 0.00 |
| 3 | $R\Theta_{JA}$ | Junction-to-free air | 23.9 | 0.50 |
| 4 | $R\Theta_{JA}$ | Junction-to-free air | 22.6 | 1.00 |
| 5 | $R\Theta_{JA}$ | Junction-to-free air | 21.3 | 2.00 |

† 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.

## PACKAGING INFORMATION

| Orderable part number | Status (1) | Material type (2) | Package \| Pins | Package qty \| Carrier | RoHS (3) | Lead finish/ Ball material (4) | MSL rating/ Peak reflow (5) | Op temp (°C) | Part marking (6) |
|---|---|---|---|---|---|---|---|---|---|
| TMS320C6205DGWT200 | Obsolete | Production | NFBGA (GWT) \| 288 | - | - | Call TI | Call TI | 0 to 90 | TMS 320C6205DGWT 200 |
| TMS320C6205DZHK200 | Obsolete | Production | BGA MICROSTAR (ZHK) \| 288 | - | - | Call TI | Call TI | 0 to 90 | 320C6205DZHK 200 TMS |
| TMS320C6205DZWT200 | Obsolete | Production | NFBGA (ZWT) \| 288 | - | - | Call TI | Call TI | 0 to 90 | TMS 320C6205DZWT 200 |
| TMS320C6205GHKA200 | Obsolete | Production | BGA MICROSTAR (GHK) \| 288 | - | - | Call TI | Call TI | -40 to 105 | C6205GHK200 A TMS320 |
| TMS320C6205GWT200 | Obsolete | Production | NFBGA (GWT) \| 288 | - | - | Call TI | Call TI | 0 to 90 | TMS 320C6205GWT 200 |
| TMS320C6205GWTA200 | Obsolete | Production | NFBGA (GWT) \| 288 | - | - | Call TI | Call TI | -40 to 105 | TMS320 C6205GWT200 A |
| TMS320C6205ZHK200 | Obsolete | Production | BGA MICROSTAR (ZHK) \| 288 | - | - | Call TI | Call TI | 0 to 90 | 320C6205ZHK 200 TMS |
| TMS320C6205ZHK200.B | Obsolete | Production | BGA MICROSTAR (ZHK) \| 288 | - | - | Call TI | Call TI | -40 to 105 | |
| TMS320C6205ZWT200 | Obsolete | Production | NFBGA (ZWT) \| 288 | - | - | Call TI | Call TI | 0 to 90 | TMS 320C6205ZWT 200 |
| TMS32C6205DGWTA200 | Obsolete | Production | NFBGA (GWT) \| 288 | - | - | Call TI | Call TI | -40 to 105 | TMS320 C6205DGWT200 A |
| TMS32C6205DZWTA200 | Obsolete | Production | NFBGA (ZWT) \| 288 | - | - | Call TI | Call TI | -40 to 105 | TMS320 C6205DZWT200 A |

(1) **Status:** For more details on status, see our product life cycle.

(2) **Material type:** When designated, preproduction parts are prototypes/experimental devices, and are not yet approved or released for full production. Testing and final process, including without limitation quality assurance, reliability performance testing, and/or process qualification, may not yet be complete, and this item is subject to further changes or possible discontinuation. If available for ordering, purchases will be subject to an additional waiver at checkout, and are intended for early internal evaluation purposes only. These items are sold without warranties of any kind.

(3) **RoHS values:** Yes, No, RoHS Exempt. See the TI RoHS Statement for additional information and value definition.

(4) **Lead finish/Ball material:** Parts may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

(5) **MSL rating/Peak reflow:** The moisture sensitivity level ratings and peak solder (reflow) temperatures. In the event that a part has multiple moisture sensitivity ratings, only the lowest level per JEDEC standards is shown. Refer to the shipping label for the actual reflow temperature that will be used to mount the part to the printed circuit board.

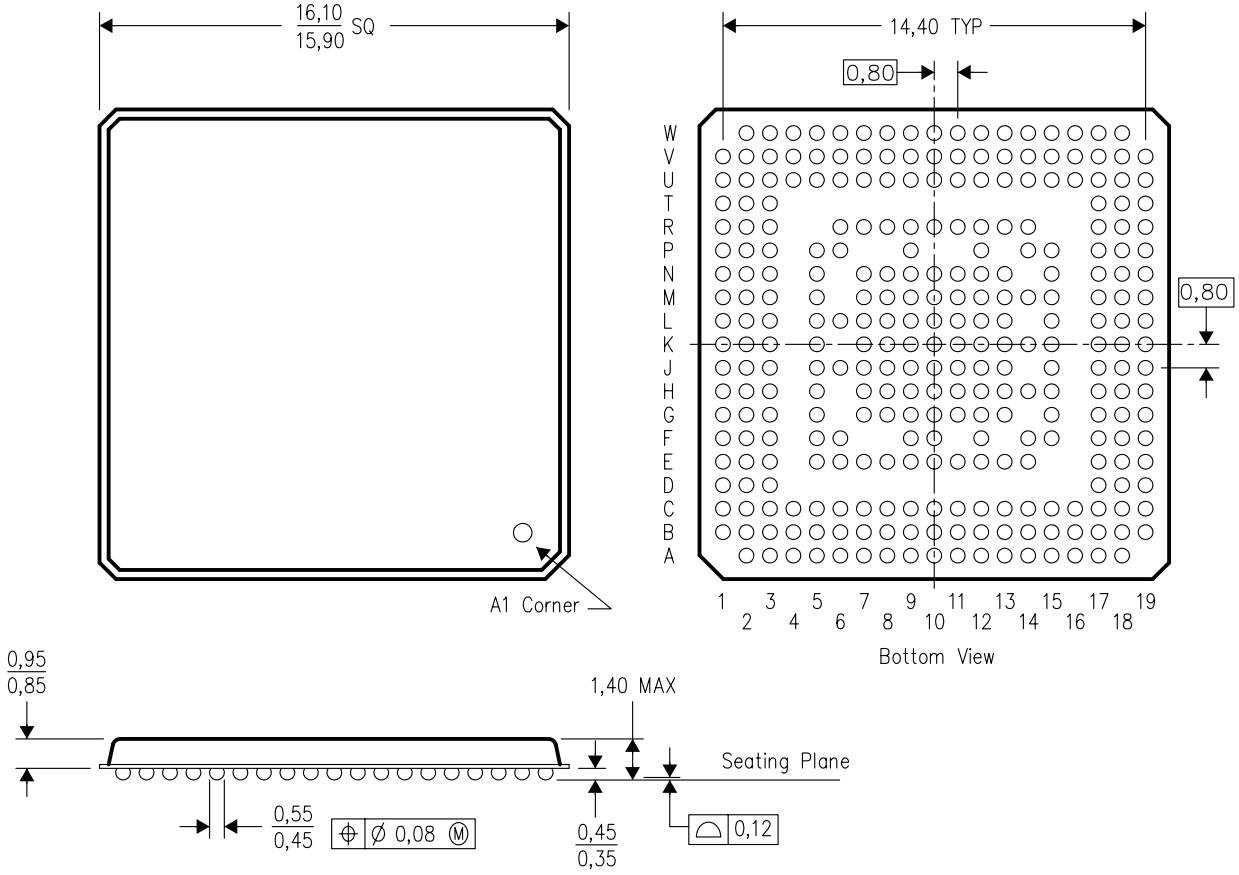
(6) **Part marking:** There may be an additional marking, which relates to the logo, the lot trace code information, or the environmental category of the part.

Multiple part markings will be inside parentheses. Only one part marking contained in parentheses and separated by a "~" will appear on a part. If a line is indented then it is a continuation of the previous line and the two combined represent the entire part marking for that device.

**Important Information and Disclaimer:**The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

## GHK (S–PBGA–N288) PLASTIC BALL GRID ARRAY



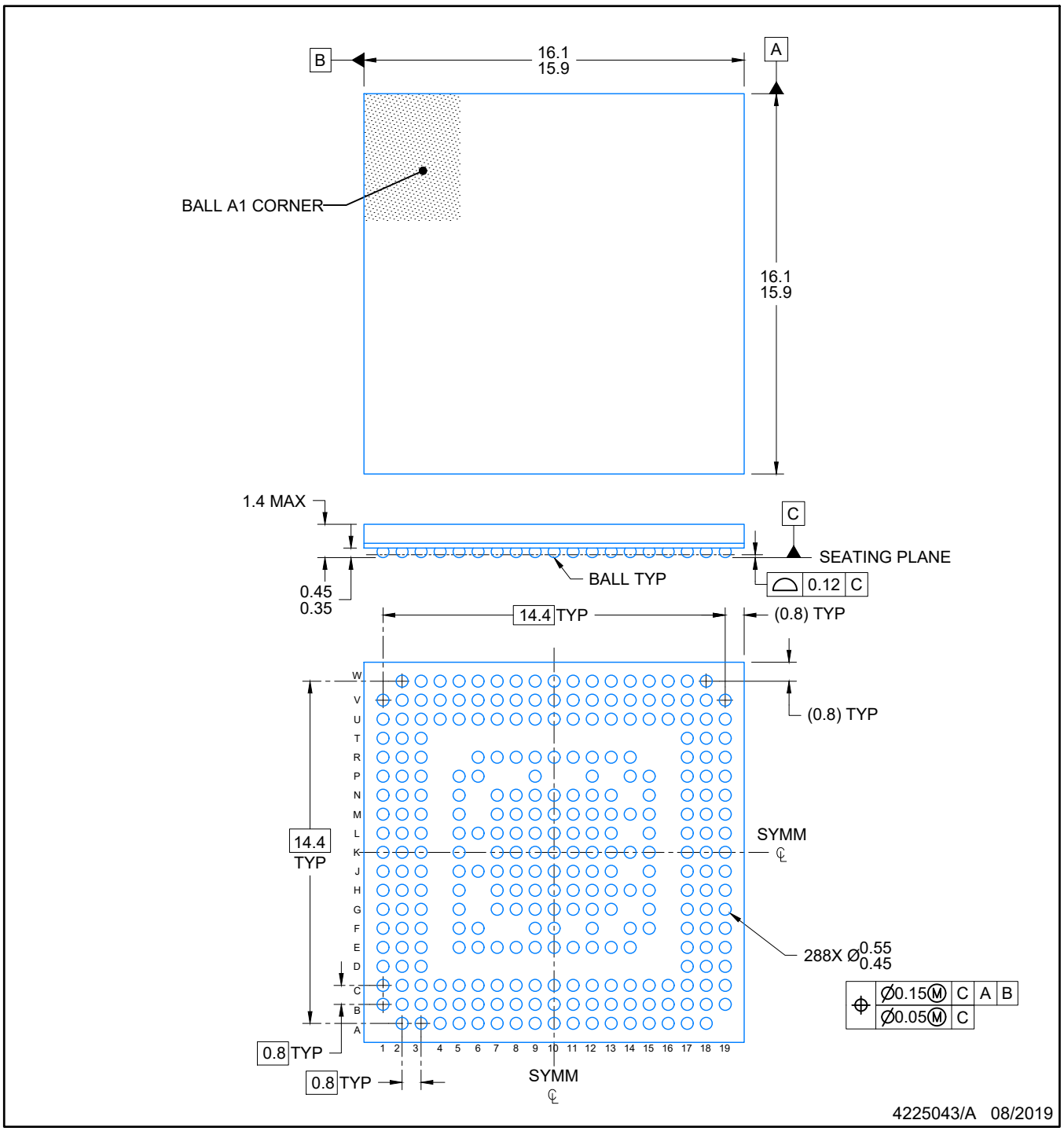

NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.

# PACKAGE OUTLINE

## GWT0288A

## NFBGA - 1.4 mm max height

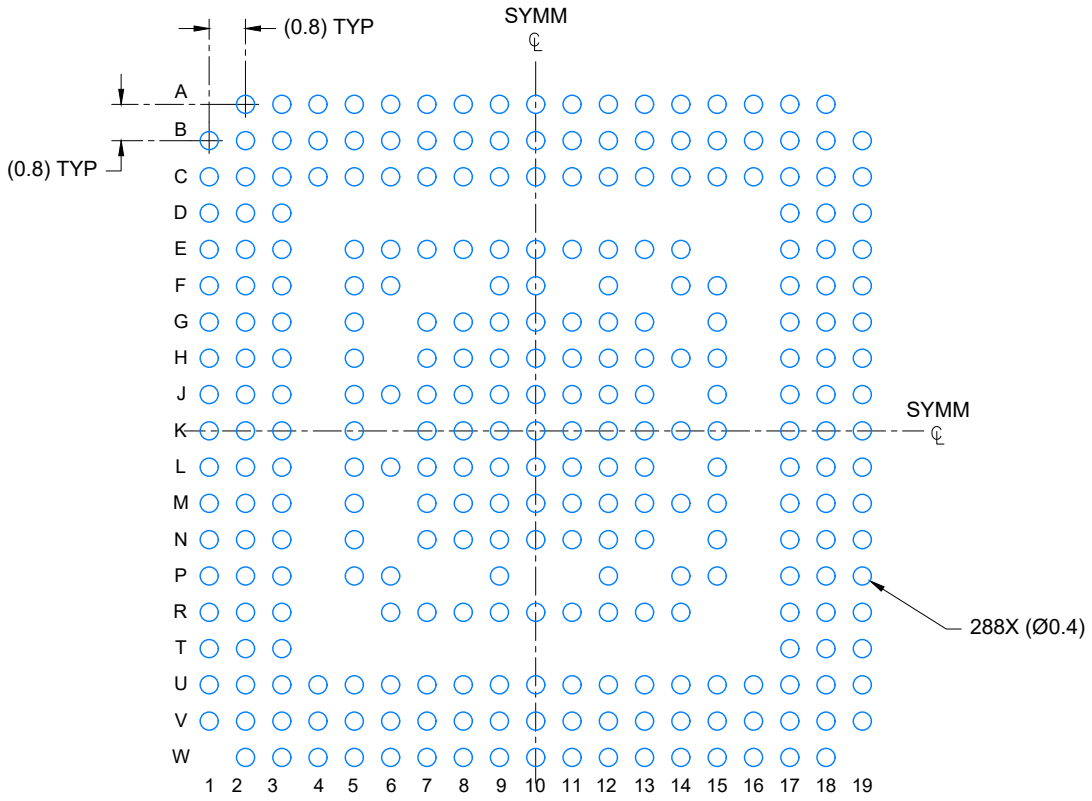
PLASTIC BALL GRID ARRAY




NOTES:

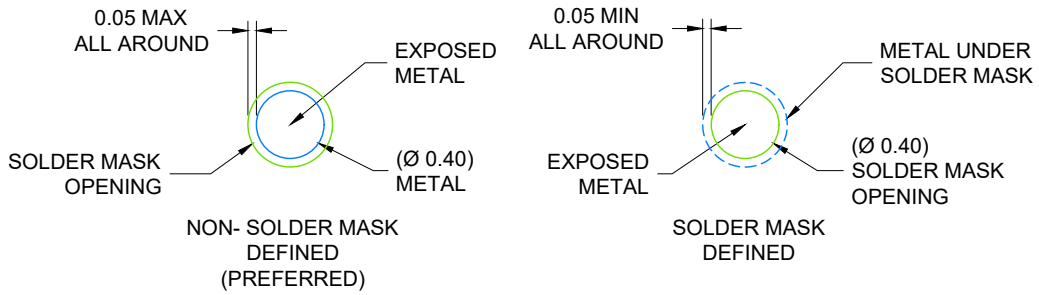
NanoFree is a trademark of Texas Instruments.

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.




LAND PATTERN EXAMPLE

SCALE: 6X



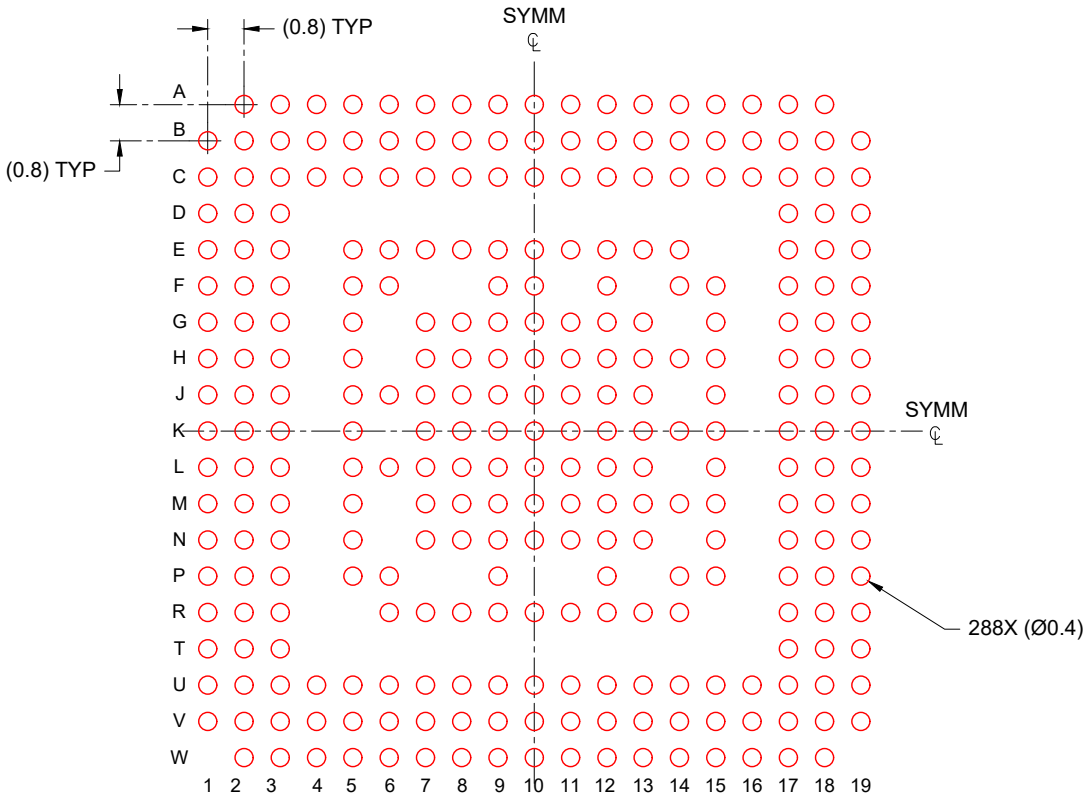

SOLDER MASK DETAILS

NOT TO SCALE

4225043/A 08/2019

NOTES: (continued)

3. Final dimensions may vary due to manufacturing tolerance considerations and also routing constraints. Refer to Texas Instruments Literature number SNVA009 (www.ti.com/lit/snva009).




SOLDER PASTE EXAMPLE
BASED ON 0.150 mm THICK STENCIL
SCALE: 6X

4225043/A 08/2019

NOTES: (continued)

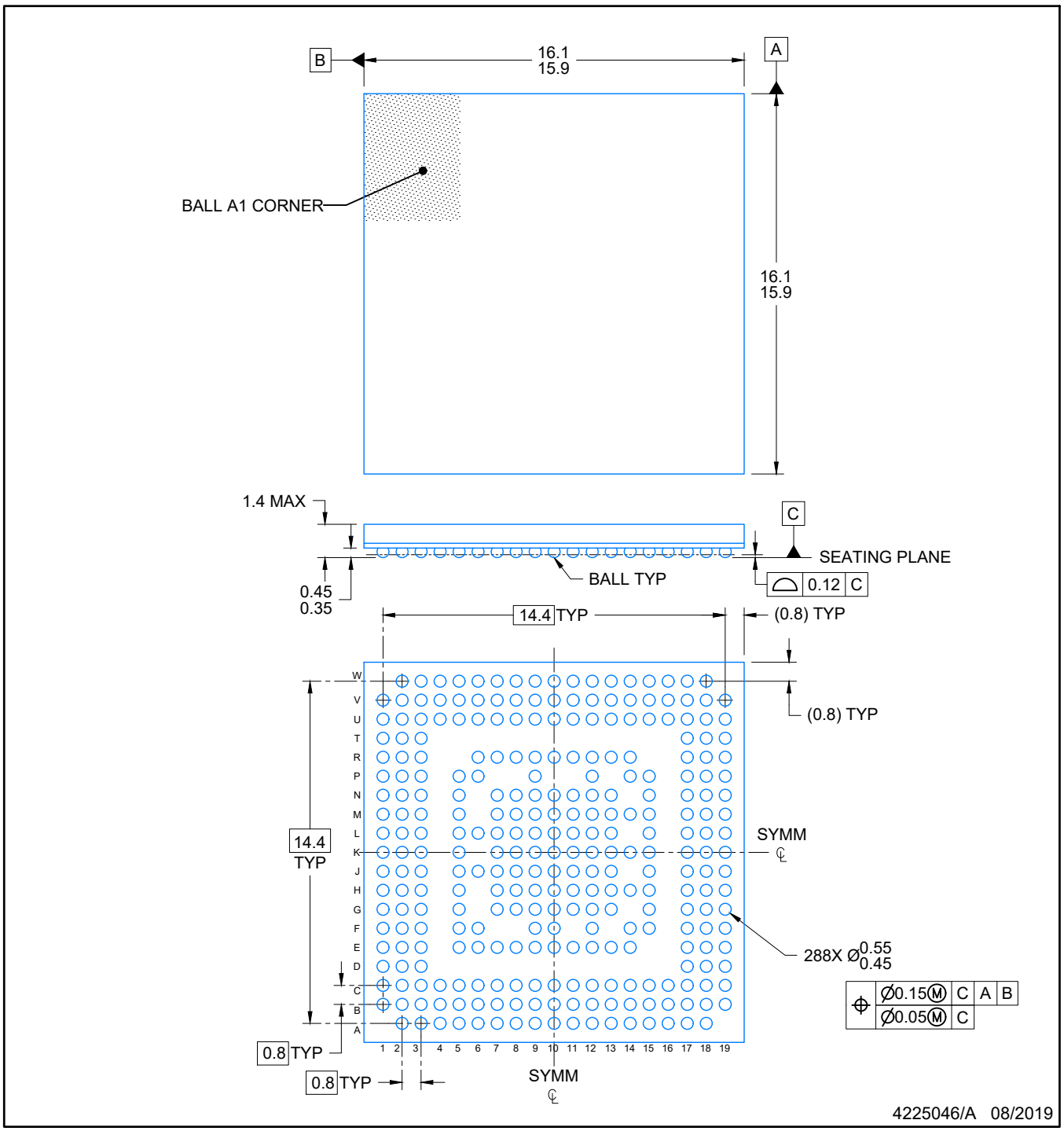
4. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release.

# PACKAGE OUTLINE

## ZWT0288A

## NFBGA - 1.4 mm max height

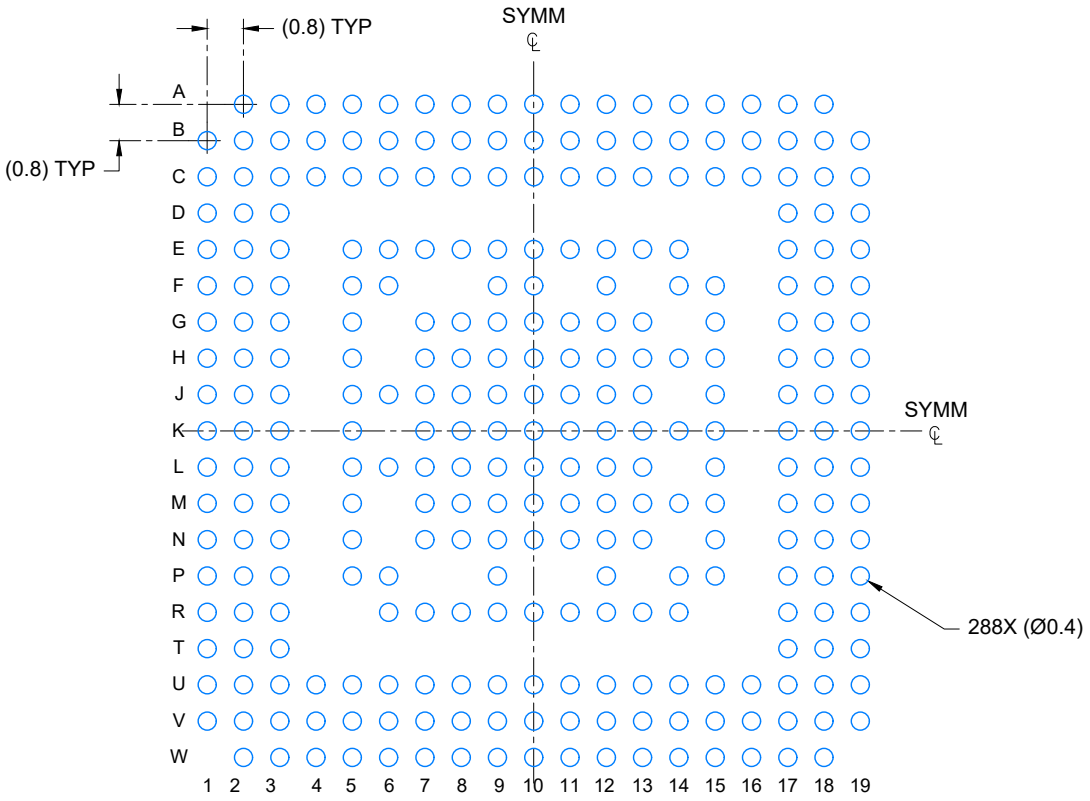
PLASTIC BALL GRID ARRAY




NOTES:

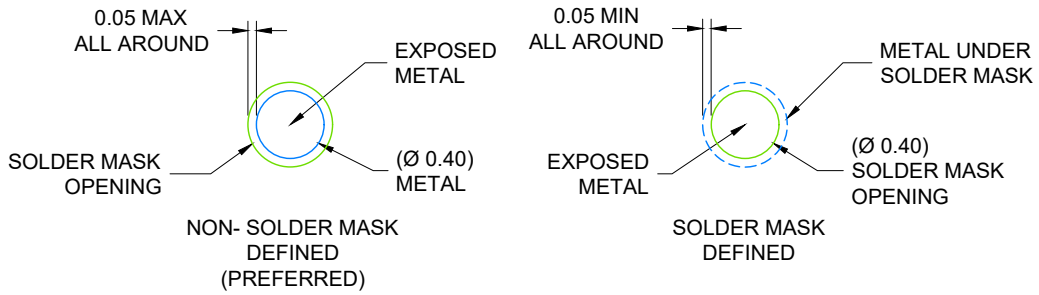
NanoFree is a trademark of Texas Instruments.

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.




LAND PATTERN EXAMPLE

SCALE: 6X




SOLDER MASK DETAILS

NOT TO SCALE

4225046/A 08/2019

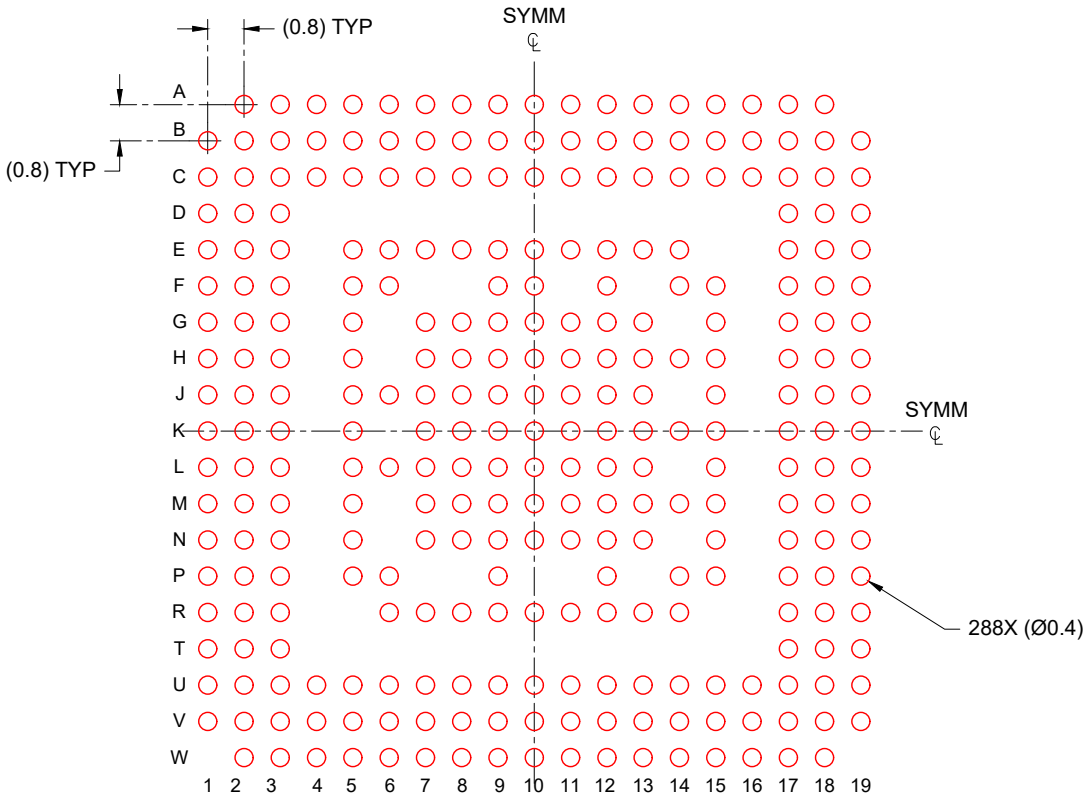
NOTES: (continued)

3. Final dimensions may vary due to manufacturing tolerance considerations and also routing constraints. Refer to Texas Instruments Literature number SNVA009 (www.ti.com/lit/snva009).

# EXAMPLE STENCIL DESIGN

## NFBGA - 1.4 mm max height

PLASTIC BALL GRID ARRAY



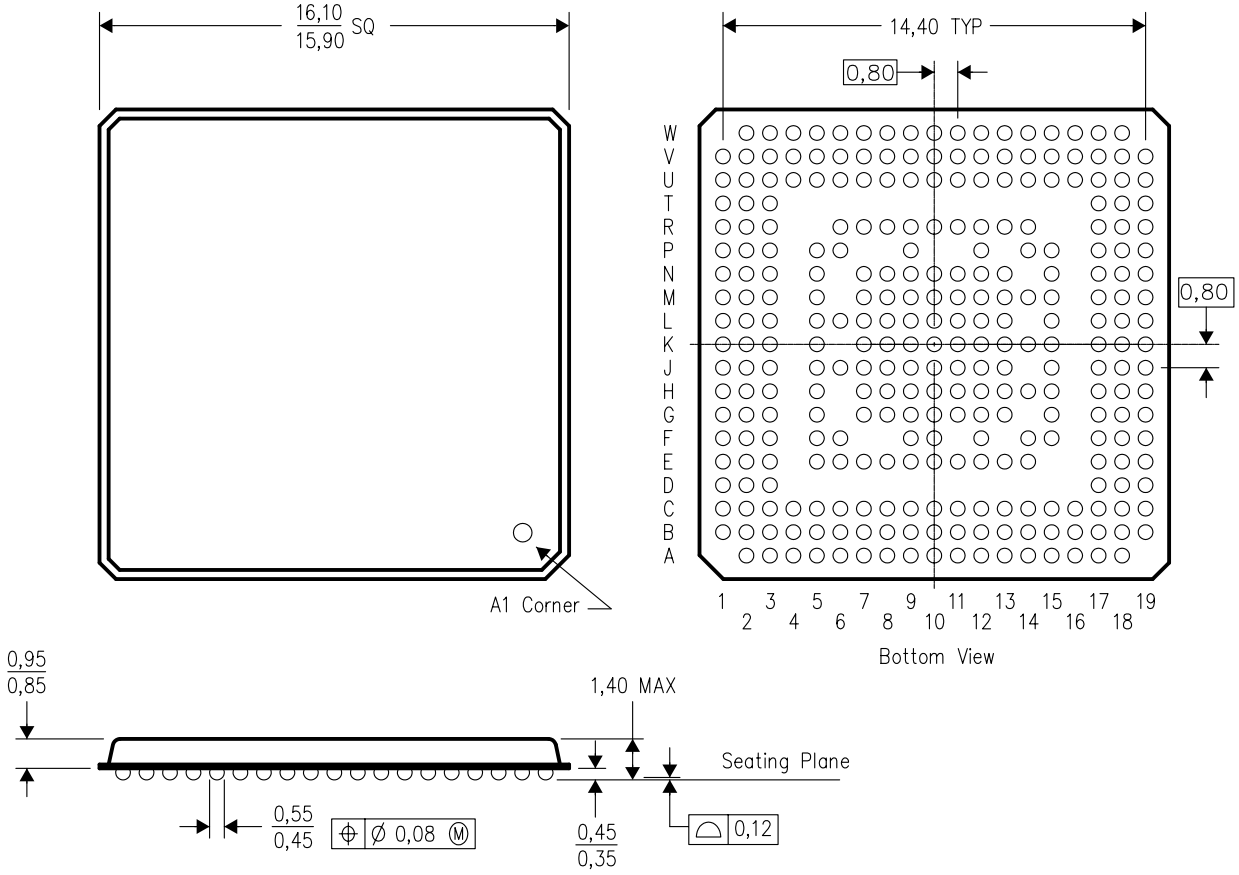

SOLDER PASTE EXAMPLE
BASED ON 0.150 mm THICK STENCIL
SCALE: 6X

4225046/A 08/2019

NOTES: (continued)

4. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release.

## ZHK (S–PBGA–N288) PLASTIC BALL GRID ARRAY




NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. This is a lead–free solder ball design.

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View TMS320C6205GWT200 on WIN SOURCE
- Texas Instruments Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

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