TMS320C6413, TMS320C6410 Fixed-Point Digital Signal Processors

Data Manual

Literature Number: SPRS247F
April 2004 − Revised January 2006

TEXAS
INSTRUMENTS

Revision History

This data manual revision history highlights the technical changes made to the SPRS247E device-specific data manual to make it an SPRS247F revision.

Scope: Applicable updates to the C64x device family, specifically relating to the TMS320C6413 and TMS320C6410 devices, have been incorporated.

PAGE(s) NO.	ADDS/CHANGES/DELETES
13	Features, 32−Bit External Memory Interface (EMIF) section: Changed “**1024**M−Byte Total Addressable Memory Space” to “**512**M−Byte Total Addressable Memory Space”
15	Description section: Updated “With performance of up to 3200 million...” paragraph
23	Functional Overview, Memory Map Summary, TMS320C6413/C6410 Memory Map Summary table: Updated Table
47	Device Configurations, Device Configuration at Device Reset, section: Added Note
48	Device Configurations, Device Configuration at Device Reset, C6413/C6410 Device Configuration Pins (TOUT1/LENDIAN, AEA[22:19], TOUT0/$\overline{\text{HPI\_EN}}$, HD5, CLKINSEL, and OSC_DIS) table: Updated AEA(22:21) Configuration Functional Description for 11 to “EMIFA 8−bit ROM boot”
55	Device Configurations, Device Status Register Description, Device Status (DEVSTAT) Register Selection Bit Descriptions table: Updated BOOTMODE0 and BOOTMODE1 Description for 11 to “EMIFA 8−bit ROM boot”
57	Device Configurations, Debugging Considerations section: Deleted paragraphs and added Note
60	Device Configurations, Terminal Functions, Terminal Functions table, CLOCK/PLL CONFIGURATION section: OSCV$_{DD}$ Signal Name: Changed Description from “Power for crystal oscillator (1.2 V), ***Do not*** connect to board power **1.4 V**; “ to “Power for crystal oscillator (1.2 V), ***Do not*** connect to board power **CV$_{DD}$**; “
61	Device Configurations, Terminal Functions, Terminal Functions table, RESETS, INTERRUPTS, AND GENERAL−PURPOSE INPUT/OUTPUTS section, NMI Signal Name: Updated Description
63	Terminal Functions table, EMIFA (32 BIT) − ADDRESS section: Description for AEA22 to AEA3: Added Note Updated AEA(22:3) Description for Boot mode (AEA[21:22]) − 11 to “EMIFA 8−bit ROM boot”
91	Peripherals Detailed Description, IEEE 1149.1 JTAG Compatibility Statement section: Updated paragraphs for clarity
93	Device Electrical Specifications, Recommended Operating Conditions table: Updated V$_{OS,}$ Maximum voltage during overshoot row Added V$_{US,}$ Minimum voltage during undershoot row
96	Parameter Measurement Information, AC transient rise/fall time specifications section: Added AC Transient Specification Rise Time figure Added AC Transient Specification Fall Time figure

PAGE(s) NO.	ADDS/CHANGES/DELETES
101	Peripheral Electrical Specification, Switching Characteristics Over Recommended Operating Conditions for AECLKOUT2 for the EMIFA Module table: Updated Parameter No. 6 from “$t_{d(EKI\mathbf{H}\text{-}EKO2L)}$” to “$t_{d(EKI\mathbf{L}\text{-}EKO2L)}$” Updated Parameter No. 6 from “Delay time, ECLKIN **high** to AECLKOUT2 low” to Delay time ECLKIN **low** to AECLKOUT2 low”
101	Peripheral Electrical Specification, AECLKOUT2 Timing for the EMIFA Module figure: Updated Figure
116	Peripheral Electrical Specifications, Reset Timing section: Added Note
121, 122	MULTICHANNEL AUDIO SERIAL PORT (McASP) TIMING section: Updated McASP Input and Output drawings

Contents

Section *Page*

List of Figures

Figure *Page*

List of Tables

This page intentionally left blank

1 Features

- **High-Performance Fixed-Point Digital Signal Processor (TMS320C6413/C6410)**
  - **TMS320C6413**
    - **2-ns Instruction Cycle Time**
    - **500-MHz Clock Rate**
    - **4000 MIPS**
  - **TMS320C6410**
    - **2.5-ns Instruction Cycle Time**
    - **400-MHz Clock Rate**
    - **3200 MIPS**
  - **Eight 32-Bit Instructions/Cycle**
  - **Fully Software-Compatible With C64x™**
  - **Extended Temperature Devices Available**
- **VelociTI.2™ Extensions to VelociTI™ Advanced Very-Long-Instruction-Word (VLIW) TMS320C64x™ DSP Core**
  - **Eight Highly Independent Functional Units With VelociTI.2™ Extensions:**
    - **Six ALUs (32-/40-Bit), Each Supports Single 32-Bit, Dual 16-Bit, or Quad 8-Bit Arithmetic per Clock Cycle**
    - **Two Multipliers Support Four 16 x 16-Bit Multiplies (32-Bit Results) per Clock Cycle or Eight 8 x 8-Bit Multiplies (16-Bit Results) per Clock Cycle**
  - **Load-Store Architecture With Non-Aligned Support**
  - **64 32-Bit General-Purpose Registers**
  - **Instruction Packing Reduces Code Size**
  - **All Instructions Conditional**
- **Instruction Set Features**
  - **Byte-Addressable (8-/16-/32-/64-Bit Data)**
  - **8-Bit Overflow Protection**
  - **Bit-Field Extract, Set, Clear**
  - **Normalization, Saturation, Bit-Counting**
  - **VelociTI.2™ Increased Orthogonality**
- **VelociTI.2™ Extensions to VelociTI™ Advanced Very-Long-Instruction-Word (VLIW) TMS320C64x™ DSP Core**
- **L1/L2 Memory Architecture**
  - **128K-Bit (16K-Byte) L1P Program Cache (Direct Mapped)**
  - **128K-Bit (16K-Byte) L1D Data Cache (2-Way Set-Associative)**
  - **2M-Bit (256K-Byte) L2 Unified Mapped RAM/Cache [C6413] (Flexible RAM/Cache Allocation)**
  - **1M-Bit (128K-Byte) L2 Unified Mapped RAM/Cache [C6410] (Flexible RAM/Cache Allocation)**
- **Endianess: Little Endian, Big Endian**
- **32-Bit External Memory Interface (EMIF)**
  - **Glueless Interface to Asynchronous Memories (SRAM and EPROM) and Synchronous Memories (SDRAM, SBSRAM, ZBT SRAM, and FIFO)**
  - **512M-Byte Total Addressable External Memory Space**
- **Enhanced Direct-Memory-Access (EDMA) Controller (64 Independent Channels)**
- **Host-Port Interface (HPI) [32-/16-Bit]**
- **Two Multichannel Audio Serial Ports (McASPs) - with Six Serial Data Pins each**
- **Two Inter-Integrated Circuit (I^2C) Buses**
  - **Additional GPIO Capability**
- **Two Multichannel Buffered Serial Ports**
- **Three 32-Bit General-Purpose Timers**
- **Sixteen General-Purpose I/O (GPIO) Pins**
- **Flexible PLL Clock Generator**
- **On-Chip Fundamental Oscillator**
- **IEEE-1149.1 (JTAG†) Boundary-Scan-Compatible**
- **288-Pin Ball Grid Array (BGA) Packages (GTS and ZTS Suffixes), 1.0-mm Ball Pitch**
- **0.13-μm/6-Level Cu Metal Process (CMOS)**
- **3.3-V I/Os, 1.2-V Internal**

VelociTI.2, VelociTI, and TMS320C64x are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.
† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

2 Functional Overview

2.1 GTS and ZTS BGA Packages (Bottom View)

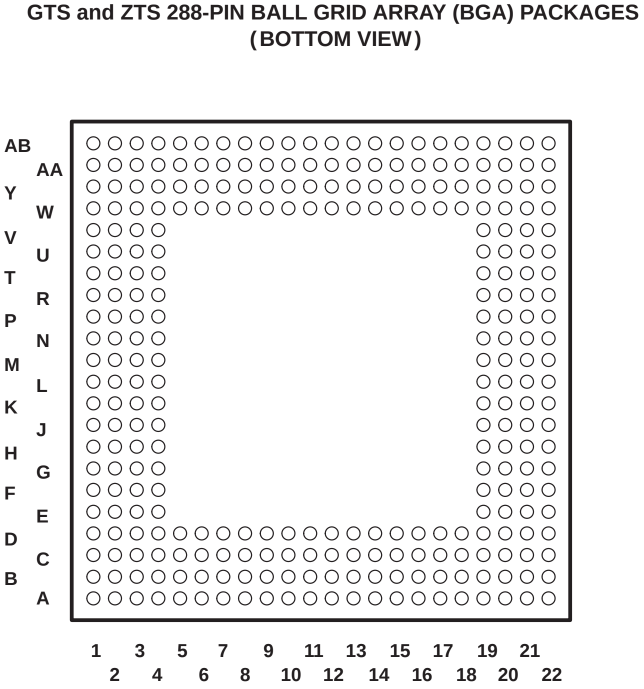


Figure 2−1. GTS and ZTS BGA Packages (Bottom View)

2.2 Description

The TMS320C64x™ DSPs (including the TMS320C6413, TMS320C6410 devices) are the highest-performance fixed-point DSP generation in the TMS320C6000™ DSP platform. The TMS320C6413 and TMS320C6410 (C6413 and C6410) devices are based on the second-generation high-performance, advanced VelociTI™ very-long-instruction-word (VLIW) architecture (VelociTI.2™) developed by Texas Instruments (TI). The high-performance, lower-cost C6413/C6410 DSPs enable customers to reduce system costs for telecom, medical, industrial, office, and photo lab equipment. The C64x™ is a code-compatible member of the C6000™ DSP platform.

With performance of up to 4000 million instructions per second (MIPS) at a clock rate of 500 MHz, the C6413 device offers cost-effective solutions to high-performance DSP programming challenges.

With performance of up to 3200 million instructions per second (MIPS) at a clock rate of 400 MHz, the C6410 device offers cost-effective solutions to high-performance DSP programming challenges. The C6410 device also provides excellent value for packet telephony and for other cost−sensitive applications demanding high performance. The C6410 device also provides excellent value for packet telephony and to other cost−sensitive applications demanding high performance

The C6413/C6410 DSP possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. The C64x™ DSP core processor has 64 general-purpose registers of 32-bit word length and eight highly independent functional units—two multipliers for a 32-bit result and six arithmetic logic units (ALUs)— with VelociTI.2™ extensions. The VelociTI.2™ extensions in the eight functional units include new instructions to accelerate the performance in video and imaging applications and extend the parallelism of the VelociTI™ architecture. The **C6413** can produce four 16-bit multiply-accumulates (MACs) per cycle for a total of 2000 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 4000 MMACS. The **C6410** can produce four 16-bit multiply-accumulates (MACs) per cycle for a total of 1600 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 3200 MMACS. The C6413/C6410 DSP also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000™ DSP platform devices.

The C6413/C6410 uses a two-level cache-based architecture and has a powerful and diverse set of peripherals. The Level 1 program cache (L1P) is a 128-Kbit direct mapped cache and the Level 1 data cache (L1D) is a 128-Kbit 2-way set-associative cache. The Level 2 memory/cache (L2) consists of an 2-Mbit memory space that is shared between program and data space [for **C6413** device] and the Level 2 memory/cache (L2) consists of an 1-Mbit memory space that is shared between program and data space [for **C6410** device]. L2 memory can be configured as mapped memory, cache, or combinations of the two. The peripheral set includes: two multichannel buffered audio serial ports (McASPs); two inter-integrated circuit bus modules (I2Cs) ; two multichannel buffered serial ports (McBSPs); three 32-bit general-purpose timers; a user-configurable 16-bit or 32-bit host-port interface (HPI16/HPI32); a 16-pin general-purpose input/output port (GP0) with programmable interrupt/event generation modes; and a 32-bit glueless external memory interface (EMIFA), which is capable of interfacing to synchronous and asynchronous memories and peripherals.

Each McASP port supports one transmit and one receive clock zone, with six serial data pins which can be individually allocated to any of the two zones. The serial port supports time-division multiplexing on each pin from 2 to 32 time slots. The C6413/C6410 has sufficient bandwidth to support all six serial data pins transmitting a 192-kHz stereo signal. Serial data in each zone may be transmitted and received on multiple serial data pins simultaneously and formatted in a multitude of variations on the Philips Inter-IC Sound (I^2S) format.

In addition, the McASP transmitter may be programmed to output multiple S/PDIF, IEC60958, AES-3, CP-430 encoded data channels simultaneously, with a single RAM containing the full implementation of user data and channel status fields.

TMS320C6000, and C6000 are trademarks of Texas Instruments.

McASP also provides extensive error-checking and recovery features, such as the bad clock detection circuit for each high-frequency master clock which verifies that the master clock is within a programmed frequency range.

The I2C ports on the TMS320C6413/C6410 allows the DSP to easily control peripheral devices and communicate with a host processor. In addition, the standard multichannel buffered serial port (McBSP) may be used to communicate with serial peripheral interface (SPI) mode peripheral devices.

The C6413/C6410 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.

2.3 Device Characteristics

Table 2−1, provides an overview of the C6413 and C6410 DSPs. The tables show significant features of the C6413 and C6410 devices, including the capacity of on-chip RAM, the peripherals, the CPU frequency, and the package type with pin count.

Table 2−1. Characteristics of the C6413 and C6410 Processors

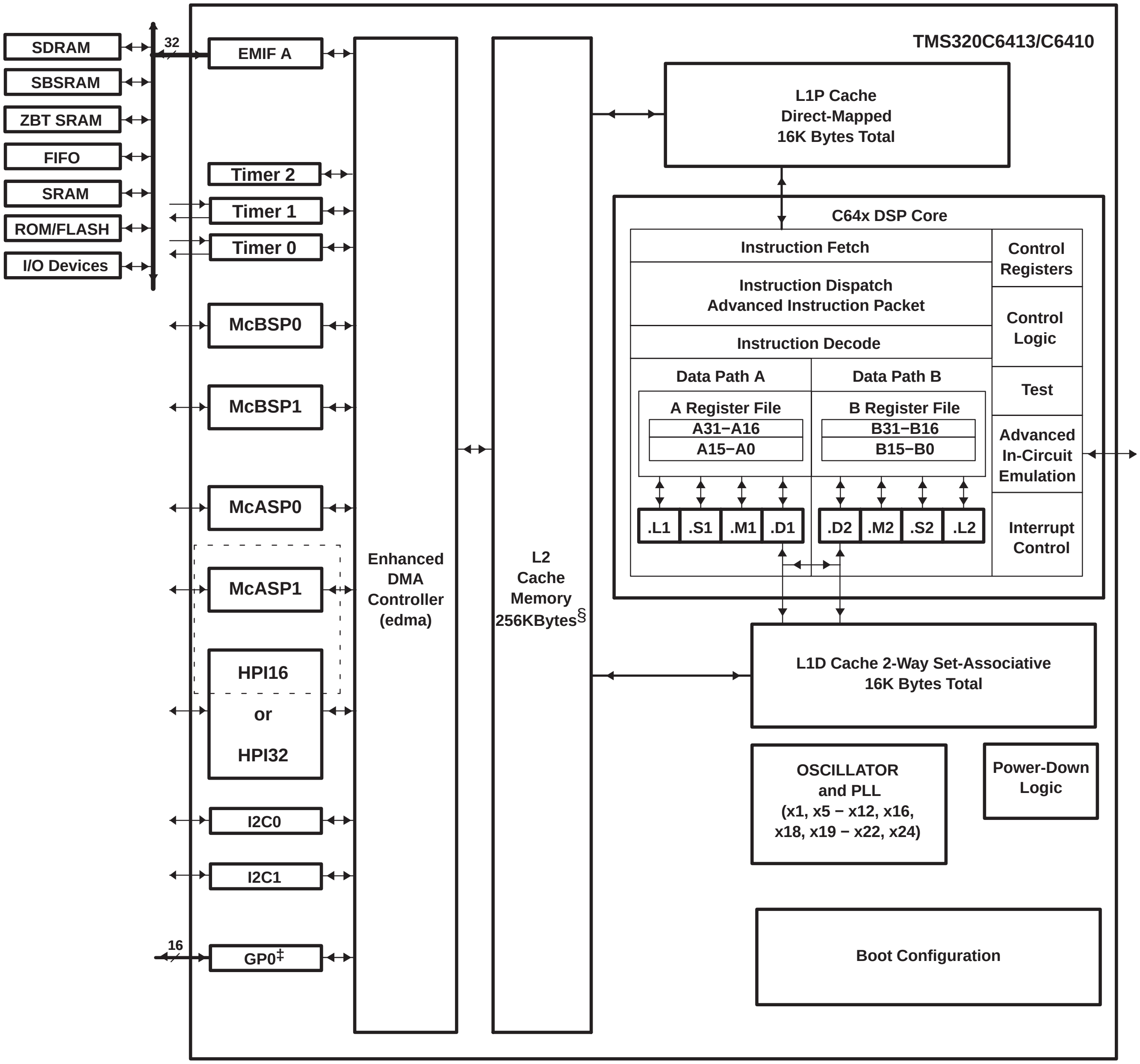
HARDWARE FEATURES		C6413 AND C6410
Peripherals Not all peripherals pins are available at the same time (For more detail, see the Device Configuration section).	EMIFA (32-bit bus width) (clock source = AECLKIN, CLKOUT4, or CLKOUT6)	1
	EDMA (64 independent channels)	1
	McASPs (use Peripheral Clock and AUXCLK)	2
	I2Cs (use Peripheral Clock)	2
	HPI (32- or 16-bit user selectable)	1 (HPI16 or HPI32)
	McBSPs (internal clock source = CPU/4 clock frequency)	2
	32-Bit Timers (internal clock source = CPU/8 clock frequency)	3
	General-Purpose Input/Output Port (GP0)	16
On-Chip Memory	Size (Bytes)	288K (**C6413**) 160K (**C6410**)
	Organization	16K-Byte (16KB) L1 Program (L1P) Cache 16KB L1 Data (L1D) Cache 256KB Unified Mapped RAM/Cache (L2) [**C6413**] 128KB Unified Mapped RAM/Cache (L2) [**C6410**]
CPU ID + CPU Rev ID	Control Status Register (CSR.[31:16])	0x0C01
JTAG BSDL_ID	JTAGID register (address location: 0x01B3F008)	0x0007902F
Frequency	MHz	500 (**C6413**) 400 (**C6410**)
Cycle Time	ns	2 ns (**C6413-500, C6413 A−500**) [500 MHz CPU, 100 MHz EMIF†] 2.5 ns (**C6410-400, C6410 A−400**) [400 MHz CPU, 100 MHz EMIF†]
Voltage	Core (V)	1.2 V
	I/O (V)	3.3 V
PLL Options	CLKIN frequency multiplier	Bypass (x1), x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x22, and x24
BGA Package	23 x 23 mm	288-Pin Flip-Chip Plastic BGA (GTS and ZTS)
Process Technology	µm	0.13 µm
Product Status‡	Product Preview (PP), Advance Information (AI), or Production Data (PD)	PD

† On this C64x™ device, the rated EMIF speed affects only the SDRAM interface on the EMIF. For more detailed information, see the EMIF device speed portion of this data sheet.

‡ 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.

2.3.1 Functional Block Diagram

Figure 2−2 shows the functional block diagram of the C6413/C6410 device.



[†] McBSPs: Framing Chips − H.100, MVIP, SCSA, T1, E1; AC97 Devices; SPI Devices; Codecs

[‡] GP0[15:8] pins are muxed with the HPI HD[15:8] pins and GP0[2:1] pins are muxed with CLKOUT6 and CLKOUT4, respectively.

[§] Note: the C6413 device has 256K-Bytes L2 Cache Memory; the C6410 device has ***only*** 128K-Bytes L2 Cache Memory.

Figure 2−2. Functional Block Diagram

2.4 CPU (DSP Core) Description

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

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

The CPU features two sets of functional units. Each set contains four units and a register file. One set contains functional units .L1, .S1, .M1, and .D1; the other set contains units .D2, .M2, .S2, and .L2. The two register files each contain 32 32-bit registers for a total of 64 general-purpose registers. In addition to supporting the packed 16-bit and 32-/40-bit fixed-point data types found in the C62x™ VelociTI™ VLIW architecture, the C64x™ register files also support packed 8-bit data and 64-bit fixed-point data types. The two sets of functional units, along with two register files, compose sides A and B of the CPU [see the functional block and CPU (DSP core) diagram, and Figure 2−3]. The four functional units on each side of the CPU can freely share the 32 registers belonging to that side. Additionally, each side features a “data cross path”—a single data bus connected to all the registers on the other side, by which the two sets of functional units can access data from the register files on the opposite side. The C64x CPU pipelines data-cross-path accesses over multiple clock cycles. This allows the same register to be used as a data-cross-path operand by multiple functional units in the same execute packet. All functional units in the C64x CPU can access operands via the data cross path. Register access by functional units on the same side of the CPU as the register file can service all the units in a single clock cycle. On the C64x CPU, a delay clock is introduced whenever an instruction attempts to read a register via a data cross path if that register was updated in the previous clock cycle.

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

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

TMS320C62x and C62x are trademarks of Texas Instruments.

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

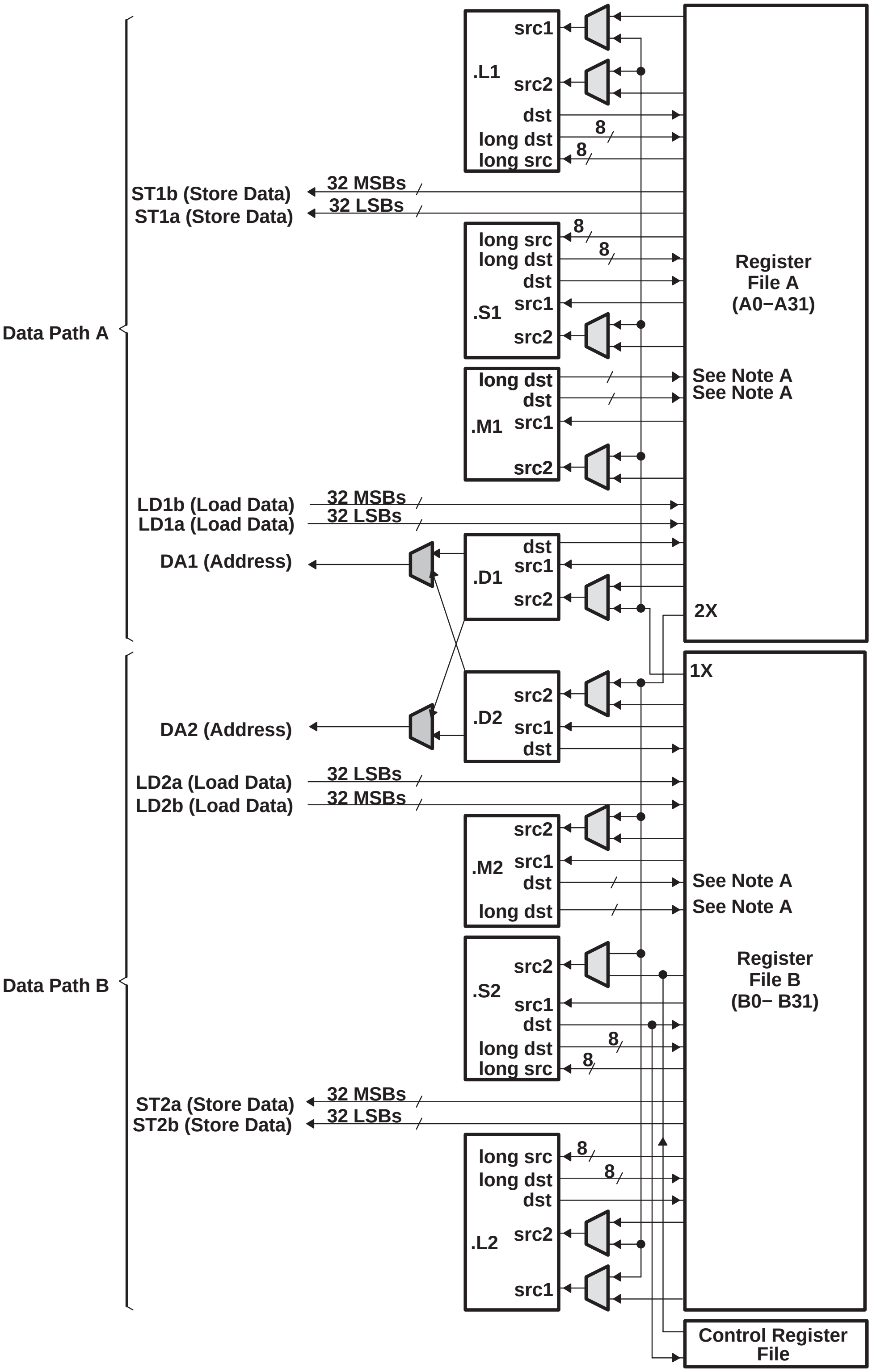
The two .S and .L functional units perform a general set of arithmetic, logical, and branch functions with results available every clock cycle. The arithmetic and logical functions on the C64x CPU include single 32-bit, dual 16-bit, and quad 8-bit operations.

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

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

- *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189)
- *TMS320C64x Technical Overview* (literature number SPRU395)

TMS320C67x is a trademark of Texas Instruments.



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

Figure 2−3. TMS320C64x™ CPU (DSP Core) Data Paths

2.5 Memory Map Summary

Table 2−2 shows the memory map address ranges of the C6413 and C6410 devices. Internal memory is always located at address 0 and can be used as both program and data memory. The external memory address ranges in the C6413/C6410 device begin at the hex address location 0x8000 0000 for EMIFA.

Table 2−2. TMS320C6413/C6410 Memory Map Summary

MEMORY BLOCK DESCRIPTION	BLOCK SIZE (BYTES)	HEX ADDRESS RANGE
Internal RAM (L2) [C6413]	256K	0000 0000 – 0003 FFFF
Reserved [C6413]	1024K minus 256K	0004 0000 – 000F FFFF
Internal RAM (L2) [C6410]	128K	0000 0000 – 0001 FFFF
Reserved [C6410]	1024K minus 128K	0002 0000 – 000F FFFF
Reserved	15M	0010 0000 – 00FF FFFF
Reserved	8M	0100 0000 – 017F FFFF
External Memory Interface A (EMIFA) Registers	256K	0180 0000 – 0183 FFFF
L2 Registers	256K	0184 0000 – 0187 FFFF
HPI Registers	256K	0188 0000 – 018B FFFF
McBSP 0 Registers	256K	018C 0000 – 018F FFFF
McBSP 1 Registers	256K	0190 0000 – 0193 FFFF
Timer 0 Registers	256K	0194 0000 – 0197 FFFF
Timer 1 Registers	256K	0198 0000 – 019B FFFF
Interrupt Selector Registers	256K	019C 0000 – 019F FFFF
EDMA RAM and EDMA Registers	256K	01A0 0000 – 01A3 FFFF
Reserved	512K	01A4 0000 – 01AB FFFF
Timer 2 Registers	256K	01AC 0000 – 01AF FFFF
GP0 Registers	256K minus 4K	01B0 0000 – 01B3 EFFF
Device Configuration Registers	4K	01B3 F000 – 01B3 FFFF
I2C0 Data and Control Registers	16K	01B4 0000 – 01B4 3FFF
I2C1 Data and Control Registers	16K	01B4 4000 – 01B4 7FFF
Reserved	16K	01B4 8000 – 01B4 BFFF
McASP0 Control Registers	16K	01B4 C000 – 01B4 FFFF
McASP1 Control Registers	16K	01B5 0000 – 01B5 3FFF
Reserved	176K	01B5 4000 – 01B7 FFFF
Reserved	128K	01B8 0000 – 01B9 FFFF
Reserved	128K	01BA 0000 – 01BB FFFF
Emulation	256K	01BC 0000 – 01BF FFFF
Reserved	528K	01C0 0000 – 01C8 3FFF
Reserved	3.5M	01C8 4000 – 01FF FFFF
QDMA Registers	52	0200 0000 – 0200 0033
Reserved	928M minus 52	0200 0034 – 2FFF FFFF
McBSP 0 Data	64M	3000 0000 – 33FF FFFF
McBSP 1 Data	64M	3400 0000 – 37FF FFFF
Reserved	64M	3800 0000 – 3BFF FFFF
McASP0 Data	1M	3C00 0000 – 3C0F FFFF

Table 2−2. TMS320C6413/C6410 Memory Map Summary (Continued)

MEMORY BLOCK DESCRIPTION	BLOCK SIZE (BYTES)	HEX ADDRESS RANGE
McASP1 Data	1M	`3C10 0000 – 3C1F FFFF`
Reserved	62M	`3C20 0000 – 3FFF FFFF`
Reserved	1G	`4000 0000 – 7FFF FFFF`
EMIFA CE0	128M	`8000 0000 – 87FF FFFF`
Reserved	128M	`8800 0000 – 8FFF FFFF`
EMIFA CE1	128M	`9000 0000 – 97FF FFFF`
Reserved	128M	`9800 0000 – 9FFF FFFF`
EMIFA CE2	128M	`A000 0000 – A7FF FFFF`
Reserved	128M	`A800 0000 – AFFF FFFF`
EMIFA CE3	128M	`B000 0000 – B7FF FFFF`
Reserved	128M	`B800 0000 – BFFF FFFF`
Reserved	1G	`C000 0000 – FFFF FFFF`

2.5.1 L2 Architecture Expanded

Figure 2−4 and Figure 2−5 show the detail of the L2 architecture on the TMS320C6413 and TMS320C6410 devices, respectively . For more information on the L2MODE bits, see the cache configuration (CCFG) register bit field descriptions in the *TMS320C64x Two-Level Internal Memory Reference Guide* (literature number SPRU610).

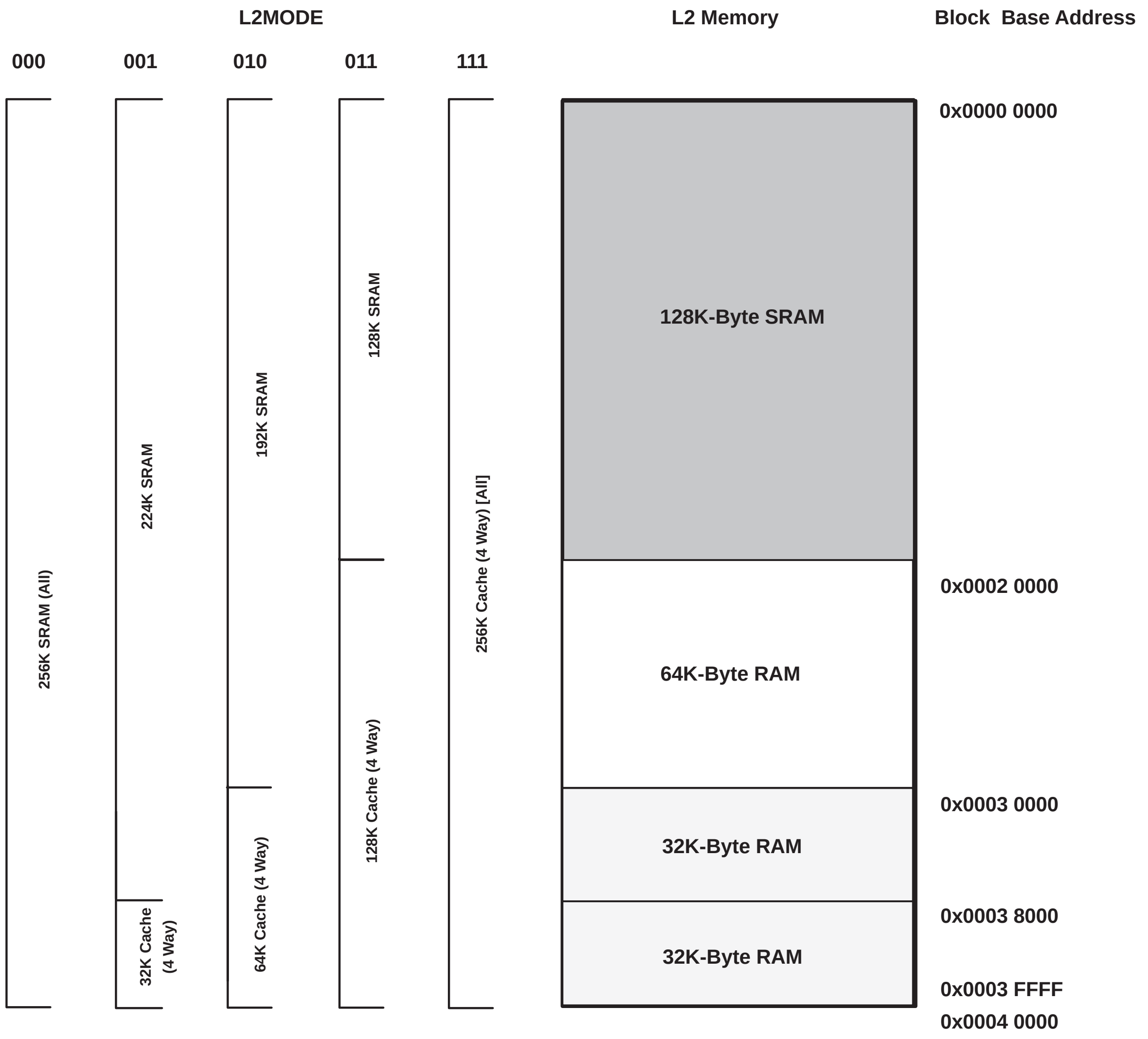
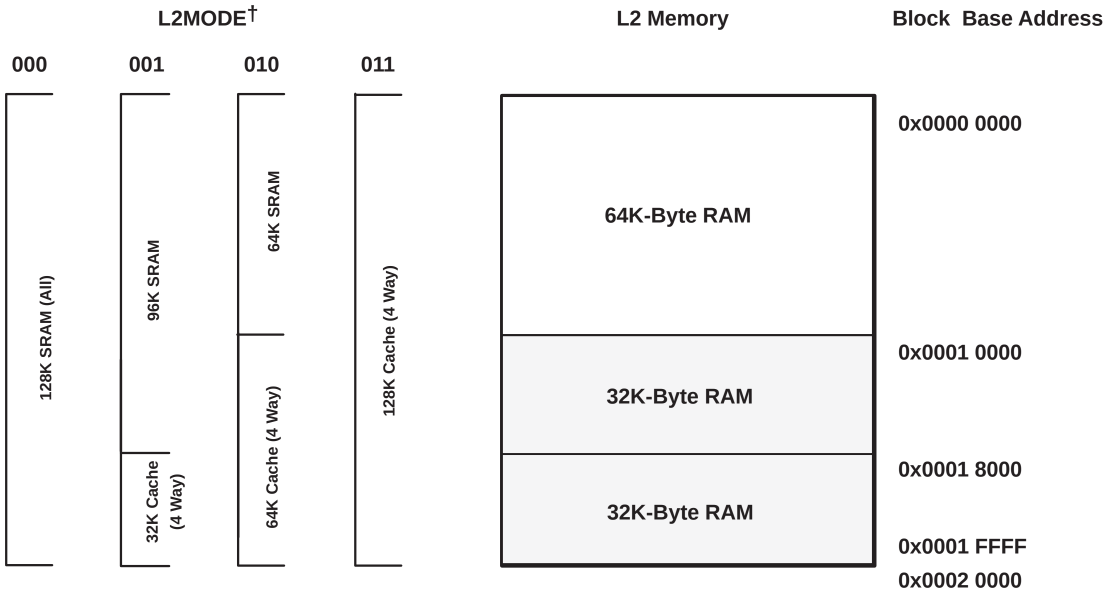


Figure 2−4. TMS320C6413 L2 Architecture Memory Configuration



† The L2MODE = 111b is not supported on the C6410 device.

Figure 2−5. TMS320C6410 L2 Architecture Memory Configuration

2.6 Peripheral Register Descriptions

Table 2−3 through Table 2−20 identify the peripheral registers for the C6413/C6410 device by their register names, acronyms, and hex address or hex address range. For more detailed information on the register contents, bit names and their descriptions, see the specific peripheral reference guide listed in the *TMS320C6000 DSP Peripherals Overview Reference Guide* (literature number SPRU190).

Table 2−3. EMIFA Registers

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

Table 2−4. L2 Cache Registers (C64x)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 0000	CCFG	Cache configuration register	
0184 0004 − 0184 0FFC	−	Reserved	
0184 1000	EDMAWEIGHT	L2 EDMA access control register	
0184 1004 − 0184 1FFC	−	Reserved	
0184 2000	L2ALLOC0	L2 allocation register 0	
0184 2004	L2ALLOC1	L2 allocation register 1	
0184 2008	L2ALLOC2	L2 allocation register 2	
0184 200C	L2ALLOC3	L2 allocation register 3	
0184 2010 − 0184 3FFC	−	Reserved	
0184 4000	L2WBAR	L2 writeback base address register	
0184 4004	L2WWC	L2 writeback word count register	
0184 4010	L2WIBAR	L2 writeback invalidate base address register	
0184 4014	L2WIWC	L2 writeback invalidate word count register	
0184 4018	L2IBAR	L2 invalidate base address register	
0184 401C	L2IWC	L2 invalidate word count register	
0184 4020	L1PIBAR	L1P invalidate base address register	
0184 4024	L1PIWC	L1P invalidate word count register	
0184 4030	L1DWIBAR	L1D writeback invalidate base address register	

Table 2−4. L2 Cache Registers (C64x) (Continued)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 4034	L1DWIWC	L1D writeback invalidate word count register	
0184 4038 − 0184 4044	−	Reserved	
0184 4048	L1DIBAR	L1D invalidate base address register	
0184 404C	L1DIWC	L1D invalidate word count register	
0184 4050 − 0184 4FFC	−	Reserved	
0184 5000	L2WB	L2 writeback all register	
0184 5004	L2WBINV	L2 writeback invalidate all register	
0184 5008 − 0184 7FFC	−	Reserved	
0184 8000 − 0184 81FC	MAR0 to MAR127	Reserved	
0184 8200	MAR128	Controls EMIFA CE0 range 8000 0000 − 80FF FFFF	
0184 8204	MAR129	Controls EMIFA CE0 range 8100 0000 − 81FF FFFF	
0184 8208	MAR130	Controls EMIFA CE0 range 8200 0000 − 82FF FFFF	
0184 820C	MAR131	Controls EMIFA CE0 range 8300 0000 − 83FF FFFF	
0184 8210	MAR132	Controls EMIFA CE0 range 8400 0000 − 84FF FFFF	
0184 8214	MAR133	Controls EMIFA CE0 range 8500 0000 − 85FF FFFF	
0184 8218	MAR134	Controls EMIFA CE0 range 8600 0000 − 86FF FFFF	
0184 821C	MAR135	Controls EMIFA CE0 range 8700 0000 − 87FF FFFF	
0184 8220	MAR136	Controls EMIFA CE0 range 8800 0000 − 88FF FFFF	
0184 8224	MAR137	Controls EMIFA CE0 range 8900 0000 − 89FF FFFF	
0184 8228	MAR138	Controls EMIFA CE0 range 8A00 0000 − 8AFF FFFF	
0184 822C	MAR139	Controls EMIFA CE0 range 8B00 0000 − 8BFF FFFF	
0184 8230	MAR140	Controls EMIFA CE0 range 8C00 0000 − 8CFF FFFF	
0184 8234	MAR141	Controls EMIFA CE0 range 8D00 0000 − 8DFF FFFF	
0184 8238	MAR142	Controls EMIFA CE0 range 8E00 0000 − 8EFF FFFF	
0184 823C	MAR143	Controls EMIFA CE0 range 8F00 0000 − 8FFF FFFF	
0184 8240	MAR144	Controls EMIFA CE1 range 9000 0000 − 90FF FFFF	
0184 8244	MAR145	Controls EMIFA CE1 range 9100 0000 − 91FF FFFF	
0184 8248	MAR146	Controls EMIFA CE1 range 9200 0000 − 92FF FFFF	
0184 824C	MAR147	Controls EMIFA CE1 range 9300 0000 − 93FF FFFF	
0184 8250	MAR148	Controls EMIFA CE1 range 9400 0000 − 94FF FFFF	
0184 8254	MAR149	Controls EMIFA CE1 range 9500 0000 − 95FF FFFF	
0184 8258	MAR150	Controls EMIFA CE1 range 9600 0000 − 96FF FFFF	
0184 825C	MAR151	Controls EMIFA CE1 range 9700 0000 − 97FF FFFF	
0184 8260	MAR152	Controls EMIFA CE1 range 9800 0000 − 98FF FFFF	
0184 8264	MAR153	Controls EMIFA CE1 range 9900 0000 − 99FF FFFF	
0184 8268	MAR154	Controls EMIFA CE1 range 9A00 0000 − 9AFF FFFF	
0184 826C	MAR155	Controls EMIFA CE1 range 9B00 0000 − 9BFF FFFF	
0184 8270	MAR156	Controls EMIFA CE1 range 9C00 0000 − 9CFF FFFF	
0184 8274	MAR157	Controls EMIFA CE1 range 9D00 0000 − 9DFF FFFF	
0184 8278	MAR158	Controls EMIFA CE1 range 9E00 0000 − 9EFF FFFF	
0184 827C	MAR159	Controls EMIFA CE1 range 9F00 0000 − 9FFF FFFF	
0184 8280	MAR160	Controls EMIFA CE2 range A000 0000 − A0FF FFFF	

Table 2−4. L2 Cache Registers (C64x) (Continued)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
0184 8284	MAR161	Controls EMIFA CE2 range A100 0000 − A1FF FFFF	
0184 8288	MAR162	Controls EMIFA CE2 range A200 0000 − A2FF FFFF	
0184 828C	MAR163	Controls EMIFA CE2 range A300 0000 − A3FF FFFF	
0184 8290	MAR164	Controls EMIFA CE2 range A400 0000 − A4FF FFFF	
0184 8294	MAR165	Controls EMIFA CE2 range A500 0000 − A5FF FFFF	
0184 8298	MAR166	Controls EMIFA CE2 range A600 0000 − A6FF FFFF	
0184 829C	MAR167	Controls EMIFA CE2 range A700 0000 − A7FF FFFF	
0184 82A0	MAR168	Controls EMIFA CE2 range A800 0000 − A8FF FFFF	
0184 82A4	MAR169	Controls EMIFA CE2 range A900 0000 − A9FF FFFF	
0184 82A8	MAR170	Controls EMIFA CE2 range AA00 0000 − AAFF FFFF	
0184 82AC	MAR171	Controls EMIFA CE2 range AB00 0000 − ABFF FFFF	
0184 82B0	MAR172	Controls EMIFA CE2 range AC00 0000 − ACFF FFFF	
0184 82B4	MAR173	Controls EMIFA CE2 range AD00 0000 − ADFF FFFF	
0184 82B8	MAR174	Controls EMIFA CE2 range AE00 0000 − AEFF FFFF	
0184 82BC	MAR175	Controls EMIFA CE2 range AF00 0000 − AFFF FFFF	
0184 82C0	MAR176	Controls EMIFA CE3 range B000 0000 − B0FF FFFF	
0184 82C4	MAR177	Controls EMIFA CE3 range B100 0000 − B1FF FFFF	
0184 82C8	MAR178	Controls EMIFA CE3 range B200 0000 − B2FF FFFF	
0184 82CC	MAR179	Controls EMIFA CE3 range B300 0000 − B3FF FFFF	
0184 82D0	MAR180	Controls EMIFA CE3 range B400 0000 − B4FF FFFF	
0184 82D4	MAR181	Controls EMIFA CE3 range B500 0000 − B5FF FFFF	
0184 82D8	MAR182	Controls EMIFA CE3 range B600 0000 − B6FF FFFF	
0184 82DC	MAR183	Controls EMIFA CE3 range B700 0000 − B7FF FFFF	
0184 82E0	MAR184	Controls EMIFA CE3 range B800 0000 − B8FF FFFF	
0184 82E4	MAR185	Controls EMIFA CE3 range B900 0000 − B9FF FFFF	
0184 82E8	MAR186	Controls EMIFA CE3 range BA00 0000 − BAFF FFFF	
0184 82EC	MAR187	Controls EMIFA CE3 range BB00 0000 − BBFF FFFF	
0184 82F0	MAR188	Controls EMIFA CE3 range BC00 0000 − BCFF FFFF	
0184 82F4	MAR189	Controls EMIFA CE3 range BD00 0000 − BDFF FFFF	
0184 82F8	MAR190	Controls EMIFA CE3 range BE00 0000 − BEFF FFFF	
0184 82FC	MAR191	Controls EMIFA CE3 range BF00 0000 − BFFF FFFF	
0184 8300 −0184 83FC	MAR192 to MAR255	Reserved	
0184 8400 −0187 FFFF	−	Reserved	

Table 2−5. Quick DMA (QDMA) and Pseudo Registers

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

Table 2−6. EDMA Registers (C64x)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
01A0 0800 − 01A0 FF98	−	Reserved
01A0 FF9C	EPRH	Event polarity high register
01A0 FFA4	CIPRH	Channel interrupt pending high register
01A0 FFA8	CIERH	Channel interrupt enable high register
01A0 FFAC	CCERH	Channel chain enable high register
01A0 FFB0	ERH	Event high register
01A0 FFB4	EERH	Event enable high register
01A0 FFB8	ECRH	Event clear high register
01A0 FFBC	ESRH	Event set high register
01A0 FFC0	PQAR0	Priority queue allocation register 0
01A0 FFC4	PQAR1	Priority queue allocation register 1
01A0 FFC8	PQAR2	Priority queue allocation register 2
01A0 FFCC	PQAR3	Priority queue allocation register 3
01A0 FFDC	EPRL	Event polarity low register
01A0 FFE0	PQSR	Priority queue status register
01A0 FFE4	CIPRL	Channel interrupt pending low register
01A0 FFE8	CIERL	Channel interrupt enable low register
01A0 FFEC	CCERL	Channel chain enable low register
01A0 FFF0	ERL	Event low register
01A0 FFF4	EERL	Event enable low register
01A0 FFF8	ECRL	Event clear low register
01A0 FFFC	ESRL	Event set low register
01A1 0000 − 01A3 FFFF	–	Reserved

Table 2−7. EDMA Parameter RAM (C64x)[†]

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
01A0 0000 − 01A0 0017	−	Parameters for Event 0 (6 words)	Parameters for Event 0 (6 words) or Reload/Link Parameters for other Event
01A0 0018 − 01A0 002F	−	Parameters for Event 1 (6 words)	
01A0 0030 − 01A0 0047	−	Parameters for Event 2 (6 words)	
01A0 0048 − 01A0 005F	−	Parameters for Event 3 (6 words)	
01A0 0060 − 01A0 0077	−	Parameters for Event 4 (6 words)	
01A0 0078 − 01A0 008F	−	Parameters for Event 5 (6 words)	
01A0 0090 − 01A0 00A7	−	Parameters for Event 6 (6 words)	
01A0 00A8 − 01A0 00BF	−	Parameters for Event 7 (6 words)	
01A0 00C0 − 01A0 00D7	−	Parameters for Event 8 (6 words)	
01A0 00D8 − 01A0 00EF	−	Parameters for Event 9 (6 words)	
01A0 00F0 − 01A0 00107	−	Parameters for Event 10 (6 words)	
01A0 0108 − 01A0 011F	−	Parameters for Event 11 (6 words)	
01A0 0120 − 01A0 0137	−	Parameters for Event 12 (6 words)	
01A0 0138 − 01A0 014F	−	Parameters for Event 13 (6 words)	
01A0 0150 − 01A0 0167	−	Parameters for Event 14 (6 words)	
01A0 0168 − 01A0 017F	−	Parameters for Event 15 (6 words)	
01A0 0150 − 01A0 0197	−	Parameters for Event 16 (6 words)	
01A0 0168 − 01A0 01AF	−	Parameters for Event 17 (6 words)	
...		...	
01A0 05D0 − 01A0 05E7	−	Parameters for Event 62 (6 words)	
01A0 05E8 − 01A0 05FF	−	Parameters for Event 63 (6 words)	
01A0 0600 − 01A0 0617	−	Reload/link parameters for Event 0 (6 words)	Reload/Link Parameters for other Event 0−15
01A0 0618 − 01A0 062F	−	Reload/link parameters for Event 1 (6 words)	
...		...	
01A0 07E0 − 01A0 07F7	−	Reload/link parameters for Event 20 (6 words)	
01A0 07F8 − 01A0 080F	−	Reload/link parameters for Event 21 (6 words)	
01A0 0810 − 01A0 0827	−	Reload/link parameters for Event 22 (6 words)	
...		...	
01A0 13C8 − 01A0 13DF	−	Reload/link parameters for Event 147 (6 words)	
01A0 13E0 − 01A0 13F7	−	Reload/link parameters for Event 148 (6 words)	
01A0 13F8 − 01A0 13FF	−	Scratch pad area (2 words)	
01A0 1400 − 01A3 FFFF	−	Reserved	

[†] The C6413/C6410 device has 213 EDMA parameters total: 64-Event/Reload channels and 149-Reload only parameter sets [six (6) words each] that can be used to reload/link EDMA transfers.

Table 2−8. Interrupt Selector Registers (C64x)

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

Table 2−9. Device Configuration Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
01B3 F000	PERCFG	Peripheral Configuration Register	Enables or disables specific peripherals. This register is also used for power-down of disabled peripherals.
01B3 F004	DEVSTAT	Device Status Register	Read-only. Provides status of the User’s device configuration on reset.
01B3 F008	JTAGID	JTAG Identification Register	Read-only. Provides 32-bit JTAG ID of the device.
01B3 F00C − 01B3 F014	−	Reserved	
01B3 F018	PCFGLOCK	Peripheral Configuration Lock Register	
01B3 F01C − 01B3 FFFF	−	Reserved	

Table 2−10. McBSP 0 Registers

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

Table 2−11. McBSP 1 Registers

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

Table 2−12. Timer 0 Registers

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

Table 2−13. Timer 1 Registers

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

Table 2−14. Timer 2 Registers

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

Table 2−15. HPI Registers

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

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

Table 2−16. GP0 Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
01B0 0000	GPEN	GP0 enable register
01B0 0004	GPDIR	GP0 direction register
01B0 0008	GPVAL	GP0 value register
01B0 000C	−	Reserved
01B0 0010	GPDH	GP0 delta high register
01B0 0014	GPHM	GP0 high mask register
01B0 0018	GPDL	GP0 delta low register
01B0 001C	GPLM	GP0 low mask register
01B0 0020	GPGC	GP0 global control register
01B0 0024	GPPOL	GP0 interrupt polarity register
01B0 0028 − 01B3 EFFF	−	Reserved

Table 2−17. McASP0 and McASP1 Control Registers

HEX ADDRESS RANGE		ACRONYM	REGISTER NAME
McASP0	McASP1		
01B4 C000	01B5 0000	PID	Peripheral Identification register [Register value: 0x0010 0101]
01B4 C004	01B5 0004	PWRDEMU	Power down and emulation management register
01B4 C008	01B5 0008	−	Reserved
01B4 C00C	01B5 000C	−	Reserved
01B4 C010	01B5 0010	PFUNC	Pin function register
01B4 C014	01B5 0014	PDIR	Pin direction register
01B4 C018	01B5 0018	PDOUT	Pin data out register
01B4 C01C	01B5 001C	PDIN/PDSET	Pin data in / data set register Read returns: PDIN Writes affect: PDSET
01B4 C020	01B5 0020	PDCLR	Pin data clear register
01B4 C024 − 01B4 C040	01B5 0024 − 01B5 0040	−	Reserved
01B4 C044	01B5 0044	GBLCTL	Global control register
01B4 C048	01B5 0048	AMUTE	Mute control register
01B4 C04C	01B5 004C	DLBCTL	Digital Loop-back control register
01B4 C050	01B5 0050	DITCTL	DIT mode control register
01B4 C054 − 01B4 C05C	01B5 0054 − 01B5 005C	−	Reserved
01B4 C060	01B5 0060	RGBLCTL	Alias of GBLCTL containing only Receiver Reset bits, allows transmit to be reset independently from receive.
01B4 C064	01B5 0064	RMASK	Receiver format unit bit mask register
01B4 C068	01B5 0068	RFMT	Receive bit stream format register
01B4 C06C	01B5 006C	AFSRCTL	Receive frame sync control register
01B4 C070	01B5 0070	ACLKRCTL	Receive clock control register
01B4 C074	01B5 0074	AHCLKRCTL	High-frequency receive clock control register
01B4 C078	01B5 0078	RTDM	Receive TDM slot 0−31 register
01B4 C07C	01B5 007C	RINTCTL	Receiver interrupt control register
01B4 C080	01B5 0080	RSTAT	Status register − Receiver
01B4 C084	01B5 0084	RSLOT	Current receive TDM slot register
01B4 C088	01B5 0088	RCLKCHK	Receiver clock check control register
01B4 C08C − 01B4 C09C	01B5 008C − 01B5 009C	−	Reserved
01B4 C0A0	01B5 00A0	XGBLCTL	Alias of GBLCTL containing only Transmitter Reset bits, allows transmit to be reset independently from receive.
01B4 C0A4	01B5 00A4	XMASK	Transmit format unit bit mask register
01B4 C0A8	01B5 00A8	XFMT	Transmit bit stream format register
01B4 C0AC	01B5 00AC	AFSXCTL	Transmit frame sync control register
01B4 C0B0	01B5 00B0	ACLKXCTL	Transmit clock control register
01B4 C0B4	01B5 00B4	AHCLKXCTL	High-frequency Transmit clock control register
01B4 C0B8	01B5 00B8	XTDM	Transmit TDM slot 0−31 register
01B4 C0BC	01B5 00BC	XINTCTL	Transmit interrupt control register
01B4 C0C0	01B5 00C0	XSTAT	Status register − Transmitter
01B4 C0C4	01B5 00C4	XSLOT	Current transmit TDM slot
01B4 C0C8	01B5 00C8	XCLKCHK	Transmit clock check control register

Table 2−17. McASP0 and McASP1 Control Registers (Continued)

HEX ADDRESS RANGE		ACRONYM	REGISTER NAME
McASP0	McASP1		
01B4 C0CC − 01B4 C0FC	01B5 00CC − 01B5 00FC	−	Reserved
01B4 C100	01B5 0100	DITCSRA0	Left (even TDM slot) channel status register file
01B4 C104	01B5 0104	DITCSRA1	Left (even TDM slot) channel status register file
01B4 C108	01B5 0108	DITCSRA2	Left (even TDM slot) channel status register file
01B4 C10C	01B5 010C	DITCSRA3	Left (even TDM slot) channel status register file
01B4 C110	01B5 0110	DITCSRA4	Left (even TDM slot) channel status register file
01B4 C114	01B5 0114	DITCSRA5	Left (even TDM slot) channel status register file
01B4 C118	01B5 0118	DITCSRB0	Right (odd TDM slot) channel status register file
01B4 C11C	01B5 011C	DITCSRB1	Right (odd TDM slot) channel status register file
01B4 C120	01B5 0120	DITCSRB2	Right (odd TDM slot) channel status register file
01B4 C124	01B5 0124	DITCSRB3	Right (odd TDM slot) channel status register file
01B4 C128	01B5 0128	DITCSRB4	Right (odd TDM slot) channel status register file
01B4 C12C	01B5 012C	DITCSRB5	Right (odd TDM slot) channel status register file
01B4 C130	01B5 0130	DITUDRA0	Left (even TDM slot) user data register file
01B4 C134	01B5 0134	DITUDRA1	Left (even TDM slot) user data register file
01B4 C138	01B5 0138	DITUDRA2	Left (even TDM slot) user data register file
01B4 C13C	01B5 013C	DITUDRA3	Left (even TDM slot) user data register file
01B4 C140	01B5 0140	DITUDRA4	Left (even TDM slot) user data register file
01B4 C144	01B5 0144	DITUDRA5	Left (even TDM slot) user data register file
01B4 C148	01B5 0148	DITUDRB0	Right (odd TDM slot) user data register file
01B4 C14C	01B5 014C	DITUDRB1	Right (odd TDM slot) user data register file
01B4 C150	01B5 0150	DITUDRB2	Right (odd TDM slot) user data register file
01B4 C154	01B5 0154	DITUDRB3	Right (odd TDM slot) user data register file
01B4 C158	01B5 0158	DITUDRB4	Right (odd TDM slot) user data register file
01B4 C15C	01B5 015C	DITUDRB5	Right (odd TDM slot) user data register file
01B4 C160 − 01B4 C17C	01B5 0160 − 01B5 017C	−	Reserved
01B4 C180	01B5 0180	SRCTL0	Serializer 0 control register
01B4 C184	01B5 0184	SRCTL1	Serializer 1 control register
01B4 C188	01B5 0188	SRCTL2	Serializer 2 control register
01B4 C18C	01B5 018C	SRCTL3	Serializer 3 control register
01B4 C190	01B5 0190	SRCTL4	Serializer 4 control register
01B4 C194	01B5 0194	SRCTL5	Serializer 5 control register
01B4 C198	01B5 0198	−	Reserved
01B4 C19C	01B5 019C	−	Reserved
01B4 C1A0 − 01B4 C1FC	01B5 01A0 − 01B5 01FC	−	Reserved
01B4 C200	01B5 0200	XBUF0	Transmit Buffer for Serializer 0
01B4 C204	01B5 0204	XBUF1	Transmit Buffer for Serializer 1
01B4 C208	01B5 0208	XBUF2	Transmit Buffer for Serializer 2
01B4 C20C	01B5 020C	XBUF3	Transmit Buffer for Serializer 3
01B4 C210	01B5 0210	XBUF4	Transmit Buffer for Serializer 4
01B4 C214	01B5 0214	XBUF5	Transmit Buffer for Serializer 5

Table 2−17. McASP0 and McASP1 Control Registers (Continued)

HEX ADDRESS RANGE		ACRONYM	REGISTER NAME
McASP0	**McASP1**		
01B4 C218	01B5 0218	−	Reserved
01B4 C21C	01B5 021C	−	Reserved
01B4 C220 − 01B4 C27C	01B5 0220 − 01B5 027C	−	Reserved
01B4 C280	01B5 0280	RBUF0	Receive Buffer for Serializer 0
01B4 C284	01B5 0284	RBUF1	Receive Buffer for Serializer 1
01B4 C288	01B5 0288	RBUF2	Receive Buffer for Serializer 2
01B4 C28C	01B5 028C	RBUF3	Receive Buffer for Serializer 3
01B4 C290	01B5 0290	RBUF4	Receive Buffer for Serializer 4
01B4 C294	01B5 0294	RBUF5	Receive Buffer for Serializer 5
01B4 C298	01B5 0298	−	Reserved
01B4 C29C	01B5 029C	−	Reserved
01B4 C2A0 − 01B4 FFFF	01B5 02A0 − 01B5 3FFF	−	Reserved

Table 2−18. McASP0 Data Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
3C00 0000 − 3C0F FFFF	RBUF/XBUFx	McASPx receive buffers or McASPx transmit buffers via the Peripheral Data Bus.	(Used when RSEL or XSEL bits = 0 [these bits are located in the RFMT or XFMT registers, respectively].)

Table 2−19. McASP1 Data Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME	COMMENTS
3C10 0000 − 3C1F FFFF	RBUF/XBUFx	McASPx receive buffers or McASPx transmit buffers via the Peripheral Data Bus.	(Used when RSEL or XSEL bits = 0 [these bits are located in the RFMT or XFMT registers, respectively].)

Table 2−20. I2C0 and I2C1 Registers

HEX ADDRESS RANGE		ACRONYM	REGISTER NAME
I2C0	**I2C1**		
01B4 0000	01B4 4000	I2COARx	I2Cx own address register
01B4 0004	01B4 4004	I2CIMRx	I2Cx interrupt mask/status register
01B4 0008	01B4 4008	I2CSTRx	I2Cx interrupt status register
01B4 000C	01B4 400C	I2CCLKLx	I2Cx clock low-time divider register
01B4 0010	01B4 4010	I2CCLKHx	I2Cx clock high-time divider register
01B4 0014	01B4 4014	I2CCNTx	I2Cx data count register
01B4 0018	01B4 4018	I2CDRRx	I2Cx data receive register
01B4 001C	01B4 401C	I2CSARx	I2Cx slave address register
01B4 0020	01B4 4020	I2CDXRx	I2Cx data transmit register
01B4 0024	01B4 4024	I2CMDRx	I2Cx mode register
01B4 0028	01B4 4028	I2CIVRx	I2Cx interrupt vector register
01B4 002C	01B4 402C	I2CEMDRx	I2Cx Extended mode register
01B4 0030	01B4 4030	I2CPSCx	I2Cx prescaler register
01B4 0034	01B4 4034	I2CPID1x	I2Cx Peripheral Identification register 1 [Value: 0x0000 0105]
01B4 0038	01B4 4038	I2CPID2x	I2Cx Peripheral Identification register 2 [Value: 0x0000 0005]
01B4 003C − 01B4 0044	01B4 403C − 01B4 4044	−	Reserved
01B4 0048	01B4 4048	I2CPFUNCx	I2Cx pin function register
01B4 004C	01B4 404C	I2CPDIRx	I2Cx pin direction register
01B4 0050	01B4 4050	I2CPDINx	I2Cx pin data in register
01B4 0054	01B4 4054	I2CPDOUTx	I2Cx pin data out register
01B4 0058	01B4 4058	I2CPDSETx	I2Cx pin data set register
01B4 005C	01B4 405C	I2CPDCLRx	I2Cx pin data clear register
01B4 0060 − 01B4 3FFF	01B4 4060 − 01B4 7FFF	−	Reserved

2.7 EDMA Channel Synchronization Events

The C64x EDMA supports up to 64 EDMA channels which service peripheral devices and external memory. Table 2−21 lists the source of C64x EDMA synchronization events associated with each of the programmable EDMA channels. For the C6413/C6410 device, the association of an event to a channel is fixed; each of the EDMA channels has one specific event associated with it. These specific events are captured in the EDMA event registers (ERL, ERH) even if the events are disabled by the EDMA event enable registers (EERL, EERH). The priority of each event can be specified independently in the transfer parameters stored in the EDMA parameter RAM. For more detailed information on the EDMA module and how EDMA events are enabled, captured, processed, linked, chained, and cleared, etc., see the *TMS320C6000 DSP Enhanced Direct Memory Access (EDMA) Controller Reference Guide* (literature number SPRU234).

Table 2−21. TMS320C6413/C6410 EDMA Channel Synchronization Events[†]

EDMA CHANNEL	EVENT NAME	EVENT DESCRIPTION
0	DSP_INT	HPI-to-DSP interrupt
1	TINT0	Timer 0 interrupt
2	TINT1	Timer 1 interrupt
3	SD_INTA	EMIFA SDRAM timer interrupt
4	GPINT4/EXT_INT4	GP0 event 4/External interrupt pin 4
5	GPINT5/EXT_INT5	GP0 event 5/External interrupt pin 5
6	GPINT6/EXT_INT6	GP0 event 6/External interrupt pin 6
7	GPINT7/EXT_INT7	GP0 event 7/External interrupt pin 7
8	GPINT0	GP0 event 0
9	GPINT1	GP0 event 1
10	GPINT2	GP0 event 2
11	GPINT3	GP0 event 3
12	XEVT0	McBSP0 transmit event
13	REVT0	McBSP0 receive event
14	XEVT1	McBSP1 transmit event
15	REVT1	McBSP1 receive event
16−18	–	None
19	TINT2	Timer 2 interrupt
20−27	–	None
28	–	None
29	–	None
30−31	–	None
32	AXEVTE0	McASP0 transmit even event
33	AXEVTO0	McASP0 transmit odd event
34	AXEVT0	McASP0 transmit event
35	AREVTE0	McASP0 receive even event
36	AREVTO0	McASP0 receive odd event
37	AREVT0	McASP0 receive event
38	AXEVTE1	McASP1 transmit even event
39	AXEVTO1	McASP1 transmit odd event
40	AXEVT1	McASP1 transmit event

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

Table 2−21. TMS320C6413/C6410 EDMA Channel Synchronization Events[†] (Continued)

EDMA CHANNEL	EVENT NAME	EVENT DESCRIPTION
41	AREVTE1	McASP1 receive even event
42	AREVTO1	McASP1 receive odd event
43	AREVT1	McASP1 receive event
44	ICREVT0	I2C0 receive event
45	ICXEVT0	I2C0 transmit event
46	ICREVT1	I2C1 receive event
47	ICXEVT1	I2C1 transmit event
48	GPINT8	GP0 event 8
49	GPINT9	GP0 event 9
50	GPINT10	GP0 event 10
51	GPINT11	GP0 event 11
52	GPINT12	GP0 event 12
53	GPINT13	GP0 event 13
54	GPINT14	GP0 event 14
55	GPINT15	GP0 event 15
56−63	–	None

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

2.8 Interrupt Sources and Interrupt Selector

The C64x DSP core supports 16 prioritized interrupts, which are listed in Table 2−22. The highest-priority interrupt is INT_00 (dedicated to RESET) while the lowest-priority interrupt is INT_15. The first four interrupts (INT_00−INT_03) are non-maskable and fixed. The remaining interrupts (INT_04−INT_15) are maskable and default to the interrupt source specified in Table 2−22. The interrupt source for interrupts 4−15 can be programmed by modifying the selector value (binary value) in the corresponding fields of the Interrupt Selector Control registers: MUXH (address 0x019C0000) and MUXL (address 0x019C0004).

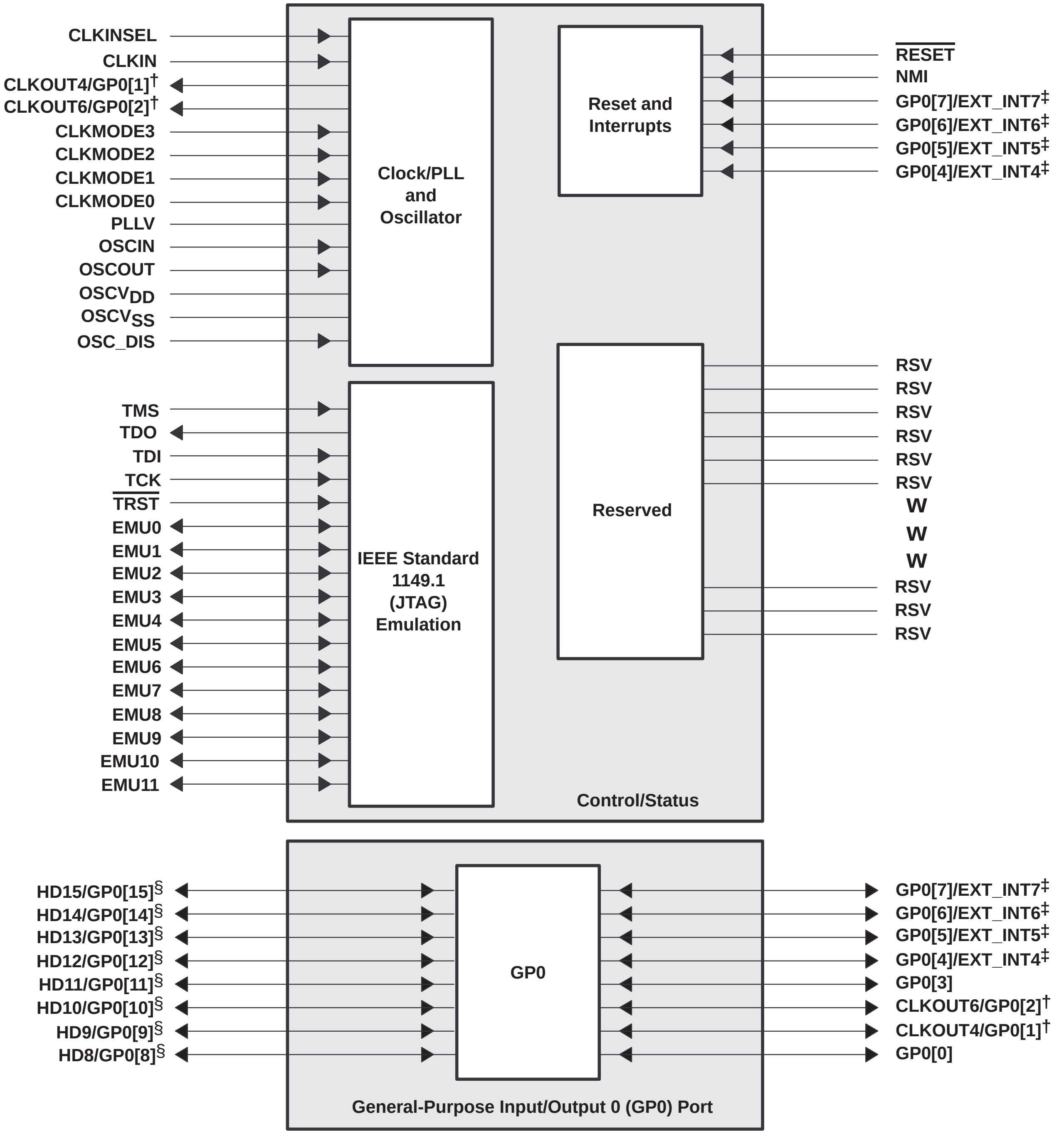
Table 2−22. C6413/C6410 DSP Interrupts

CPU INTERRUPT NUMBER	INTERRUPT SELECTOR CONTROL REGISTER	SELECTOR VALUE (BINARY)	INTERRUPT EVENT	INTERRUPT SOURCE
INT_00[†]	–	–	RESET	
INT_01[†]	–	–	NMI	
INT_02[†]	–	–	Reserved	Reserved. Do not use.
INT_03[†]	–	–	Reserved	Reserved. Do not use.
INT_04[‡]	MUXL[4:0]	00100	GPINT4/EXT_INT4	GP0 interrupt 4/External interrupt pin 4
INT_05[‡]	MUXL[9:5]	00101	GPINT5/EXT_INT5	GP0 interrupt 5/External interrupt pin 5
INT_06[‡]	MUXL[14:10]	00110	GPINT6/EXT_INT6	GP0 interrupt 6/External interrupt pin 6
INT_07[‡]	MUXL[20:16]	00111	GPINT7/EXT_INT7	GP0 interrupt 7/External interrupt pin 7
INT_08[‡]	MUXL[25:21]	01000	EDMA_INT	EDMA channel (0 through 63) interrupt
INT_09[‡]	MUXL[30:26]	01001	EMU_DTDMA	EMU DTDMA
INT_10[‡]	MUXH[4:0]	00011	SD_INTA	EMIFA SDRAM timer interrupt
INT_11[‡]	MUXH[9:5]	01010	EMU_RTDXRX	EMU real-time data exchange (RTDX) receive
INT_12[‡]	MUXH[14:10]	01011	EMU_RTDXTX	EMU RTDX transmit
INT_13[‡]	MUXH[20:16]	00000	DSP_INT	HPI-to-DSP interrupt
INT_14[‡]	MUXH[25:21]	00001	TINT0	Timer 0 interrupt
INT_15[‡]	MUXH[30:26]	00010	TINT1	Timer 1 interrupt
–	–	01100	XINT0	McBSP0 transmit interrupt
–	–	01101	RINT0	McBSP0 receive interrupt
–	–	01110	XINT1	McBSP1 transmit interrupt
–	–	01111	RINT1	McBSP1 receive interrupt
–	–	10000	GPINT0	GP0 interrupt 0
–	–	10001	Reserved	Reserved. Do not use.
–	–	10010	Reserved	Reserved. Do not use.
–	–	10011	TINT2	Timer 2 interrupt
–	–	10100	Reserved	Reserved. Do not use.
–	–	10101	Reserved	Reserved. Do not use.
–	–	10110	ICINT0	I2C0 interrupt
–	–	10111	ICINT1	I2C1 interrupt
–	–	11000	AXINT1	McASP1 transmit interrupt
–	–	11001	ARINT1	McASP1 receive interrupt
–	–	11010	Reserved	Reserved. Do not use.
–	–	11011	Reserved	Reserved. Do not use.
–	–	11100	AXINT0	McASP0 transmit interrupt
–	–	11101	ARINT0	McASP0 receive interrupt
–	–	11110	Reserved	Reserved. Do not use.
–	–	11111	Reserved	Reserved. Do not use.

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

[‡] Interrupts INT_04 through INT_15 are programmable by modifying the binary selector values in the Interrupt Selector Control registers fields. Table 2−22 shows the default interrupt sources for Interrupts INT_04 through INT_15. For more detailed information on interrupt sources and selection, see the *TMS320C6000 DSP Interrupt Selector Reference Guide* (literature number SPRU646).

2.9 Signal Groups Description

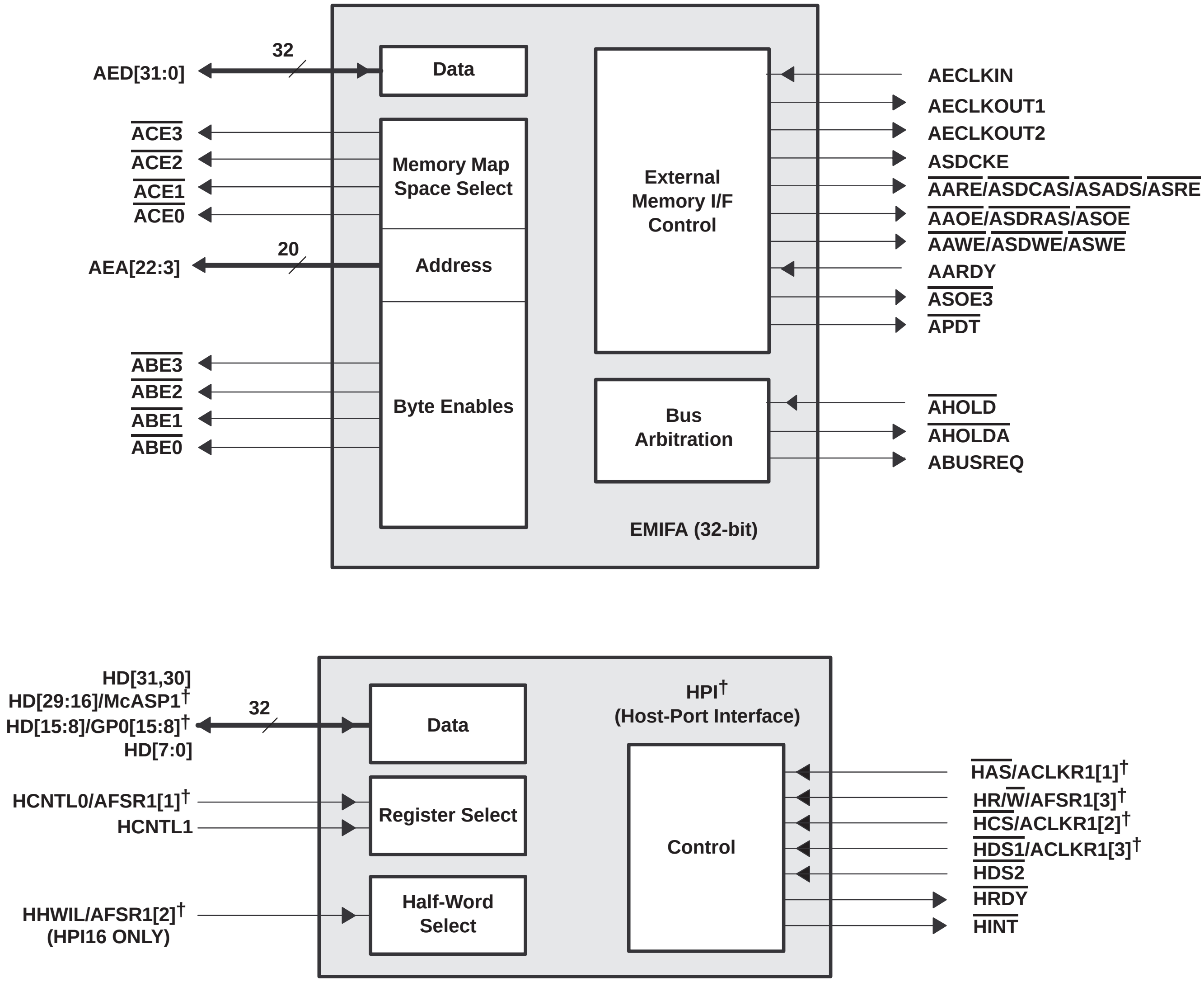


† These pins are muxed with the GP0 pins and by default these signals function as clocks (CLKOUT4 or CLKOUT6). To use these muxed pins as GPIO signals, the appropriate GPIO register bits (GPxEN and GPxDIR) must be properly enabled and configured. For more details, see the Device Configurations section of this data sheet.

‡ These pins are GP0 pins that can also function as external interrupt sources (EXT_INT[7:4]). Default after reset is EXT_INTx or GPIO as input-only.

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

Figure 2−6. CPU and Peripheral Signals



† These HPI pins are muxed with the McASP1 or GP0 peripherals. By default, these signals function as HPI and no function, respectively. For more details on these muxed pins, see the Device Configurations section of this data sheet.

Figure 2−7. Peripheral Signals

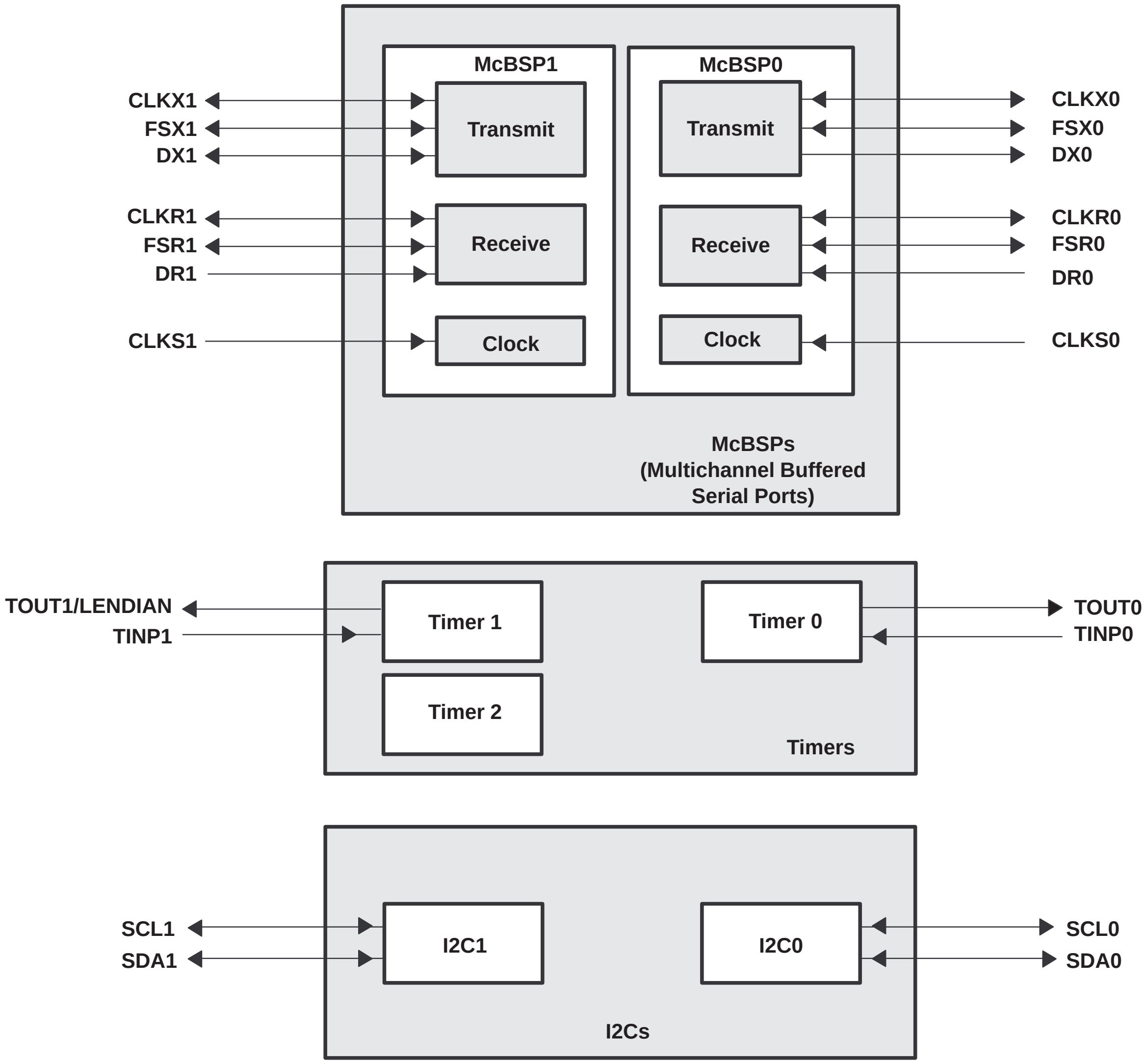
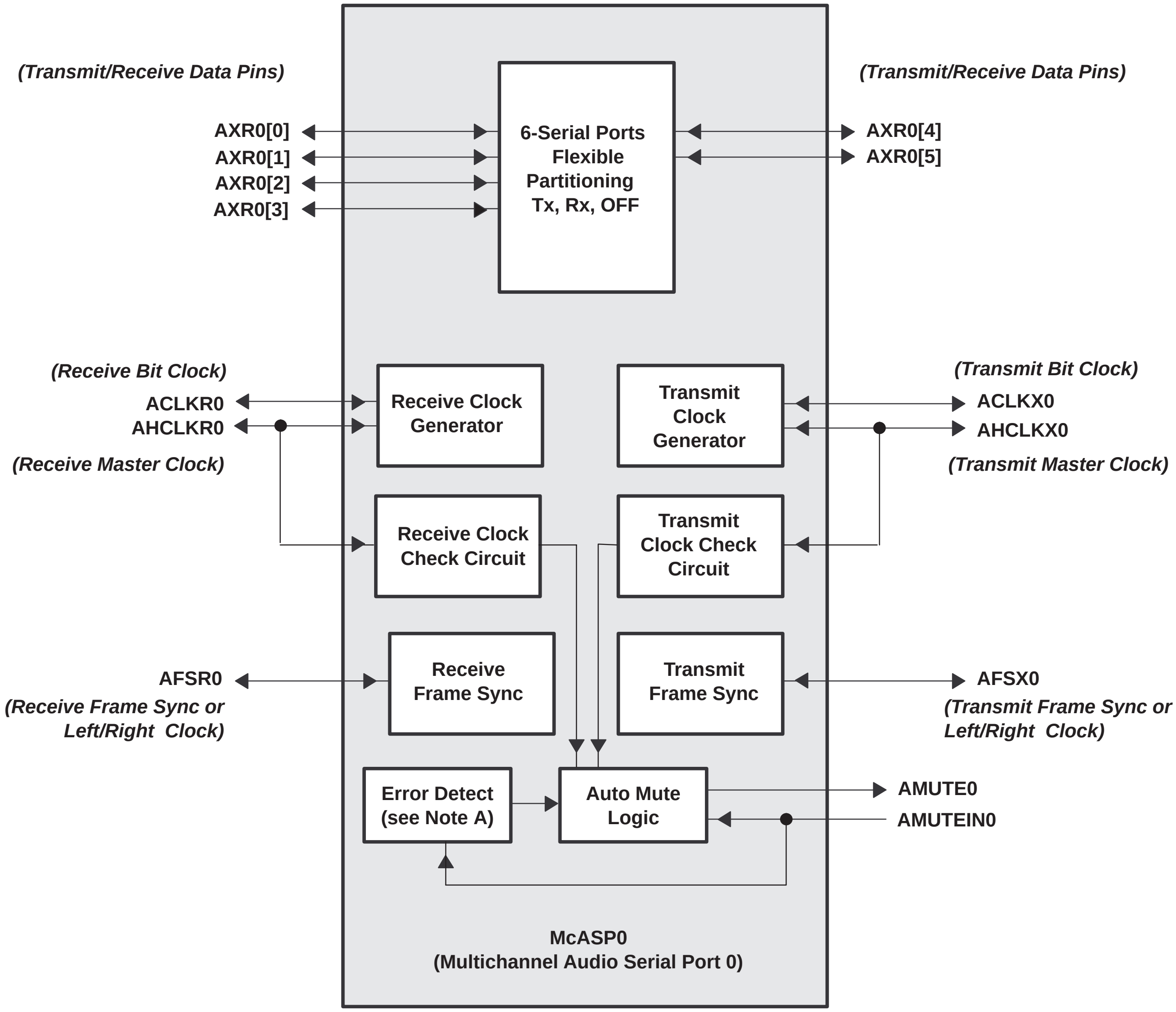
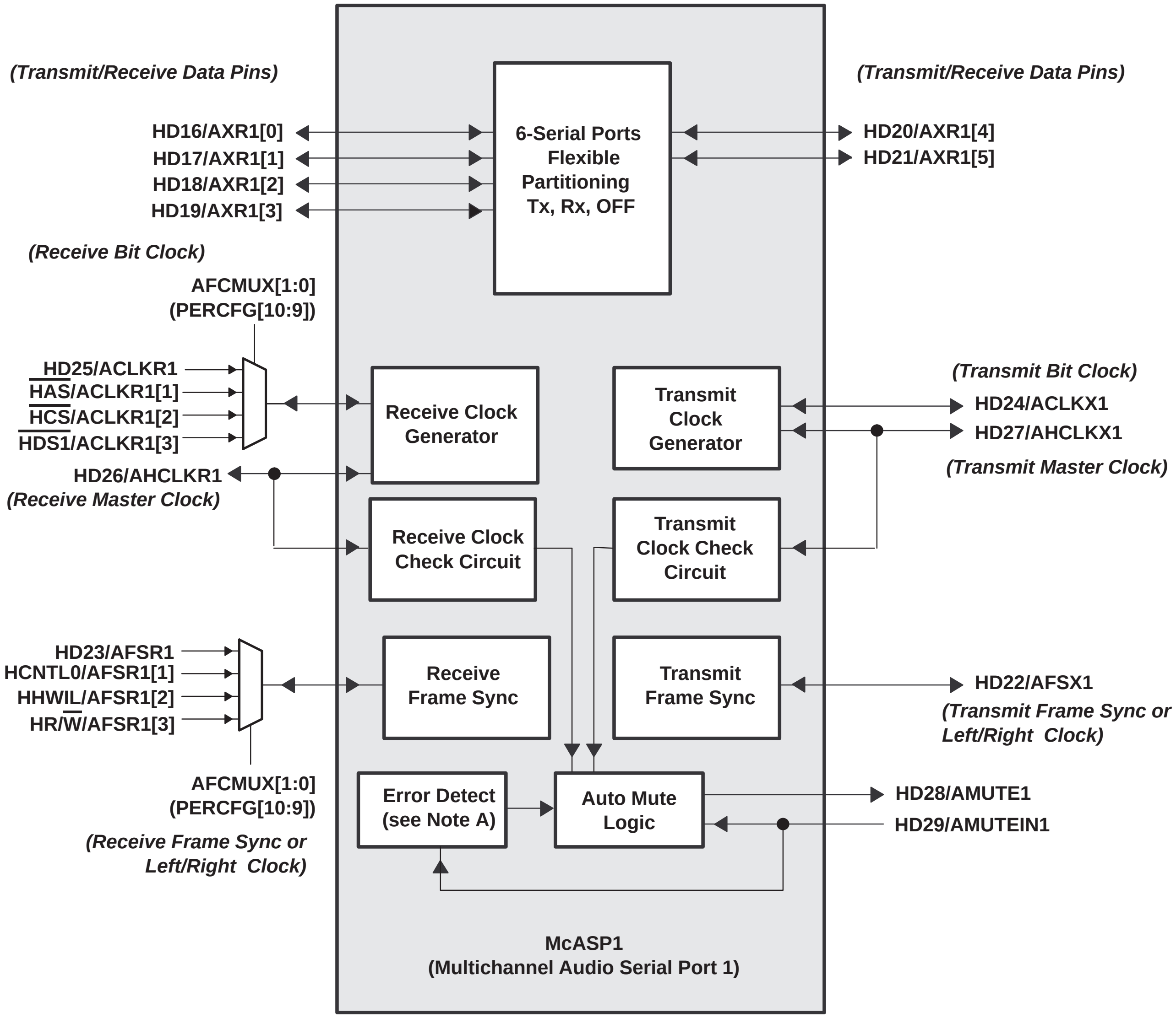


Figure 2−7. Peripheral Signals (Continued)



NOTES: A. The McASPs' Error Detect function detects underruns, overruns, early/late frame syncs, DMA errors, and external mute input.
B. Bolded and italicized text within parentheses denotes the function of the pins in an audio system.

Figure 2−7. Peripheral Signals (Continued)



NOTES: A. The McASPs' Error Detect function detects underruns, overruns, early/late frame syncs, DMA errors, and external mute input.
B. Bolded and italicized text within parentheses denotes the function of the pins in an audio system.

Figure 2−7. Peripheral Signals (Continued)

3 Device Configurations

On the C6413/C6410 device, bootmode and certain device configurations/peripheral selections are determined at device reset, while other device configurations/peripheral selections are software-configurable via the peripheral configurations register (PERCFG) [address location 0x01B3F000] after device reset.

3.1 Device Configuration at Device Reset

Table 3−1 describes the C6413/C6410 device configuration pins. The logic level of the AEA[22:19], TOUT1/LENDIAN, TOUT0/HPI_EN, and HD5 pins is latched at reset to determine the device configuration. The logic level on the device configuration pins can be set by using external pullup/pulldown resistors or by using some control device (e.g., FPGA/CPLD) to intelligently drive these pins. When using a control device, care should be taken to ensure there is no contention on the lines when the device is out of reset. The CLKINSEL and OSC_DIS configuration pins should remain driven to the correct levels during device operation and must only be changed when RESET is low. The device configuration pins are sampled during reset and are driven after the reset is removed. At this time, the control device should ensure it has stopped driving the device configuration pins of the DSP to again avoid contention.

Note: If a configuration pin must be routed out from the device, the internal pullup/pulldown (IPU/IPD) resistor should not be relied upon. TI recommends the use of an external pullup/pulldown resistor.

Table 3−1. C6413/C6410 Device Configuration Pins (TOUT1/**LENDIAN**, AEA[22:19], TOUT0/**HPI_EN**, HD5, CLKINSEL, and OSC_DIS)

CONFIGURATION PIN	NO.	IPD/IPU[†]	FUNCTIONAL DESCRIPTION
TOUT1/**LENDIAN**	AA1	IPU	Device Endian mode (LEND) 0 – System operates in Big Endian mode 1 − System operates in Little Endian mode (default)
AEA[22:21]	[M21, N21]	IPD	Bootmode [1:0] 00 – No boot (default mode) 01 − HPI boot (based on HPI_EN pin) 10 − Reserved 11 − EMIFA 8-bit ROM boot
AEA[20:19]	[P22, N22]	IPD	EMIFA input clock select Clock mode select for EMIFA (AECLKIN_SEL[1:0]) 00 – AECLKIN (default mode) 01 − CPU/4 Clock Rate 10 − CPU/6 Clock Rate 11 − Reserved
TOUT0/**HPI_EN**	AA2	IPD	HPI, McASP1, GP0[15:8] select Selects whether the HPI peripheral or McASP1 peripheral, and GP0[15:8] pins are functionally enabled 0 – HPI is enabled and the McASP1 peripheral and GP0 [15:8] pins are disabled (default mode); [HPI32, if HD5 = 1; HPI16 if HD5 = 0] 1 − HPI I is disabled and the McASP1 peripheral and GP0 [15:8] pins are enabled For more detail on the peripherals (McASP1 and GP0[15:8] pins) muxed with HPI, see the Table 3−2.
HD5	Y13	IPU	HPI peripheral bus width (HPI_WIDTH) select 0 − HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used for HPI and the remaining HD[31:16] muxed pins function as McASP1 peripheral pins or are reserved pins in the Hi-Z state.) 1 − HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.) For more detail on the peripherals (McASP1 and GP0[15:8] pins) muxed with HPI, see the Table 3−2.
CLKINSEL	A11	IPU	PLL input clock source select Selects whether the PLL input clock is CLKIN [pin high] or directly from the crystal oscillator (OSCIN and OSCOUT) [pin low]. For proper device operation, this pin ***must*** be used in conjunction with the OSC_DIS pin. 0 − Oscillator pads (OSCIN, OSCOUT directly from the crystal oscillator) For proper device operation, OSC_DIS ***must*** be 0 1 − CLKIN square wave (default) For proper device operation, OSC_DIS ***must*** be 1 **This pin *must* be pulled to the correct level even after reset.**
OSC_DIS	B7	IPU	Oscillator disable Selects whether the Oscillator is enabled or disabled. For proper device operation, this pin ***must*** follow the CLKINSEL pin operation. 0 − OSC enabled 1 − OSC disabled (default) **This pin *must* be pulled to the correct level even after reset.**

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

3.2 Peripheral Configuration at Device Reset

Some C6413/C6410 peripherals share the same pins (internally muxed) and are mutually exclusive (i.e., HPI, general-purpose input/output 0 pins GP0[15:8], and McASP1).

- HPI, McASP1, and GP0 peripherals

  The TOUT0/**HPI_EN** (AA2 pin) is latched at reset. This pin selects whether the HPI peripheral or McASP1 peripheral, and GP0[15:8] pins are functionally enabled (see Table 3−2).

Table 3−2. TOUT0/**HPI_EN and HD5 Peripheral Selection (HPI or McASP1 and Select GP0 Pins)**[†]

PERIPHERAL SELECTION		PERIPHERALS SELECTED			DESCRIPTION
HPI_EN (AA2)	**HD5 [HPI_WIDTH] (Y13)**	**HPI**	**McASP1**	**GP0 [15:8]**	
0	0	16-bit HPI	Available	N/A[‡]	HPI_EN = 0, HD5 = 0 HPI16 is enabled and McASP1 peripheral is enabled and GP0 [15:8] pins are disabled. All multiplexed HPI/McASP1 pins function as McASP1 pins. All multiplexed HPI/GP0 are reserved pins in the Hi-Z state.
0	1	32-bit HPI	N/A[‡]	N/A[‡]	HPI_EN = 0, HD5 = 1 HPI32 is enabled and McASP1 peripheral and GP0 [15:8] pins are disabled. All multiplexed HPI/McASP1 and HPI/GP0 pins function as HPI pins.
1	x	N/A[‡]	Available	Available	HPI_EN = 1, HD5 = x (don't care) HPI is disabled and the McASP1 peripheral and GP0 [15:8] pins are enabled. All multiplexed HPI/McASP1 and HPI/GP0 pins function as McASP1 and GP0 pins, respectively. To use the GP0 pins, the appropriate bits in the GP0EN and GP0DIR registers need to be set. All standalone HPI pins are reserved pins in the Hi-Z state

[†] The TOUT0/HPI_EN pin has an internal pulldown that enables the HPI by default. The TOUT0/HPI_EN pin ***can*** disable the HPI via an external pullup resistor or be driven high during reset. The TOUT0/HPI_EN pin is ***not*** software-controllable.

[‡] N/A = Not available

3.3 Peripheral Selection After Device Reset

HPI, McBSP1, McBSP0, McASP1, McASP0, I2C1, and I2C0

The C6413/C6410 device has designated registers for peripheral configuration (PERCFG), device status (DEVSTAT), and JTAG identification (JTAGID). These registers are part of the Device Configuration module and are mapped to a 4K block memory starting at 0x01B3F000. The CPU accesses these registers via the CFGBUS.

The peripheral configuration register (PERCFG), allows the user to control the peripheral selection of the McASP1, McASP0, I2C1, and I2C0 peripherals. For more detailed information on the PERCFG register control bits, see Figure 3−1 and Table 3−3.

31 – 28	27 – 24
Reserved†	Reserved†
R-0	R-0

23 – 16
Reserved†
R-0

15 – 11	10 – 9	8
Reserved†	AFCMUX[1:0]	MCASP1EN
R-0	R/W-0	R/W-0

7	6	5	4	3	2	1	0
I2C1EN	Reserved†	Reserved†	Reserved†	I2C0EN	MCBSP1EN	MCBSP0EN	MCASP0EN
R/W-0	R-0	R-0	R-0	R/W-0	**R-1**	**R-1**	R/W-0

Legend: R = Read only; R/W = Read/Write; -n = value after reset

† For proper device operation, all reserved bits have to be written with “0”.

Figure 3−1. Peripheral Configuration Register (PERCFG) [Address Location: 0x01B3F000]

Table 3−3. Peripheral Configuration (PERCFG) Register Selection Bit Descriptions

BIT	NAME	DESCRIPTION
31:11	Reserved	Reserved. Read-only, for proper device operation, all reserved bits have to be written with “0”.
10:9	AFCMUX[1:0]	Clocks and frame syncs select bits. Determines which of the clock and frame sync pairs are input to McASP1. 00 = ACLKR1, AFSR1 pins (default). 01 = ACLKR1[1], AFSR1[1] pins 10 = ACLKR1[2], AFSR1[2] pins 11 = ACLKR1[3], AFSR1[3] pins [designed for multiple non-simultaneous I2S sources with different clock sources].
8	MCASP1EN	McASP1 select bit. Selects whether the McASP1 peripheral is enabled or disabled (default). (This feature allows power savings by disabling the peripheral when not in use.) 0 = McASP1 is disabled and the module is powered down [default]. 1 = McASP1 is enabled.
7	I2C1EN	Inter-integrated circuit 1 (I2C1) enable bit. Selects whether I2C1 peripheral is enabled or disabled (default). (This feature allows power savings by disabling the peripheral when not in use.) 0 = I2C1 is disabled, and the module is powered down (default). 1 = I2C1 is enabled.
6:4	Reserved	Reserved. Read-only, for proper device operation, all reserved bits have to be written with “0”.
3	I2C0EN	Inter-integrated circuit 0 (I2C0) enable bit. Selects whether I2C0 peripheral is enabled or disabled (default). (This feature allows power savings by disabling the peripheral when not in use.) 0 = I2C0 is disabled, and the module is powered down (default). 1 = I2C0 is enabled.
2	MCBSP1EN	McBSP1 enable bit. This bit is read-only as a “1” (McBSP1 always enabled).
1	MCBSP0EN	McBSP0 enable bit . This bit is read-only as a “1” (McBSP0 always enabled).
0	MCASP0EN	McASP0 select bit. Selects whether the McASP0 peripheral is enabled or disabled. (This feature allows power savings by disabling the peripheral when not in use.) 0 = McASP0 is disabled. 1 = McASP0 is enabled.

3.4 Peripheral Configuration Lock

By default, the McASP1, McASP0, I2C1, and I2C0 peripherals are disabled on power up. In order to use these peripherals on the C6413/C6410 device, the peripheral must first be enabled in the Peripheral Configuration register (PERCFG). **Software muxed pins should not be programmed to switch functionalities during run-time. Care should also be taken to ensure that no accesses are being performed before disabling the peripherals.** To help minimize power consumption in the C6413/C6410 device, unused peripherals may be disabled..

Figure 3−2 shows the flow needed to enable (or disable) a given peripheral on the C6413/C6410 device.

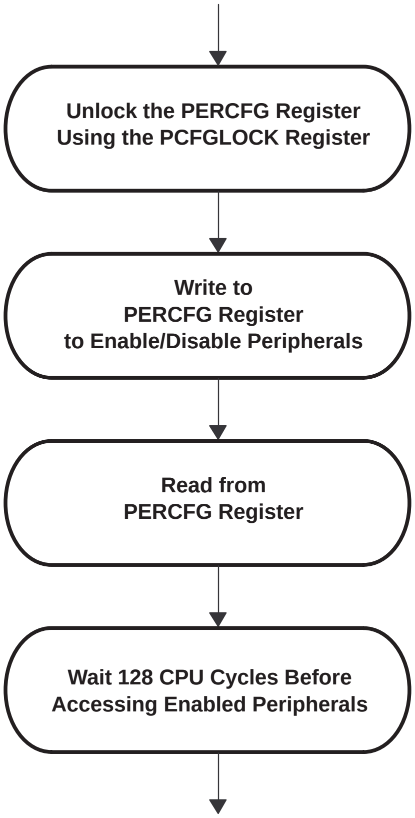


Figure 3−2. Peripheral Enable/Disable Flow Diagram

A 32-bit key (value = 0x10C0010C) must be written to the Peripheral Configuration Lock register (PCFGLOCK) in order to unlock access to the PERCFG register. Reading the PCFGLOCK register determines whether the PERCFG register is currently locked (LOCKSTAT bit = 1) or unlocked (LOCKSTAT bit = 0), see Figure 3−3. A peripheral can only be enabled when the PERCFG register is “unlocked” (LOCKSTAT bit = 0).

Read Accesses

31 … 1	0
Reserved	LOCKSTAT
R-0	R-1

Write Accesses

31 … 0
LOCK
W-0

Legend: R = Read only; R/W = Read/Write; -n = value after reset

Figure 3−3. PCFGLOCK Register Diagram [Address Location: 0x01B3 F018] − Read/Write Accesses

Table 3−4. PCFGLOCK Register Selection Bit Descriptions − Read Accesses

BIT	NAME	DESCRIPTION
31:1	Reserved	Reserved. Read-only, writes have no effect.
0	LOCKSTAT	Lock status bit. Determines whether the PERCFG register is locked or unlocked. 0 = Unlocked, read accesses to the PERCFG register allowed. 1 = Locked, write accesses to the PERCFG register do ***not*** modify the register state [default]. Reads are unaffected by Lock Status.

Table 3−5. PCFGLOCK Register Selection Bit Descriptions − Write Accesses

BIT	NAME	DESCRIPTION
31:0	LOCK	Lock bits. 0x10C0010C = Unlocks PERCFG register accesses.

Any write to the PERCFG register will automatically relock the register. In order to avoid the unnecessary overhead of multiple unlock/enable sequences, all peripherals should be enabled with a single write to the PERCFG register with the necessary enable bits set.

Prior to waiting 128 CPU cycles, the PERCFG register should be read. There is no direct correlation between the CPU issuing a write to the PERCFG register and the write actually occurring. Reading the PERCFG register after the write is issued forces the CPU to wait for the write to the PERCFG register to occur.

Once a peripheral is enabled, the DSP (or other peripherals such as the HPI) must wait a minimum of 128 CPU cycles before accessing the enabled peripheral. The user *must* ensure that no accesses are performed to a peripheral while it is disabled.

In addition to the normal usage, the PCFGLOCK register can be used to override the power saver settings specified in the PERCFG register. When the power saver feature is disabled (PCFGLOCK written with 0xC0100C01), all peripherals controlled by PERCFG are enabled. If the power saver is returned to normal operation (PCFGLOCK written with 0x0C01C010), then the peripherals return to the operating condition specified by PERCFG. Turning off the power saver settings will add a worst-case 50 mW of power to the overall DSP power consumption.

Note: overriding the settings of the PERCFG register will not cause a conflict on the multiplexed pins. For example, with the HPI and McASP1 peripherals, the HPI will still have control over the multiplexed pins provided the TOUT0/$\overline{\text{HPI\_EN}}$ pin was “0” at reset.

3.5 Device Status Register Description

The device status register depicts the status of the device peripheral selection. Once set, these bits will remain set until a device reset; therefore, these bits should be masked when reading the DEVSTAT register since their values can change. For the actual register bit names and their associated bit field descriptions, see Figure 3−4 and Table 3−6.

31	24
Reserved	
R-100x0111	

23	19	18	17	16
PLLM		Reserved	OSC EXT RES	CLKINSEL
R-xxxxx		R-**1**	R-x	R-x

15	14	13	12	11	10	9	8
Reserved			CLKMODE3	Reserved	HPI-WIDTH	Reserved	$\overline{\text{HPI\_EN}}$
R-000			R-x	R-0	R-x	R-0	R-x

7	6	5	4	3	2	1	0
CLKMODE2	CLKMODE1	CLKMODE0	LENDIAN	BOOTMODE1	BOOTMODE0	AECLKINSEL1	AECLKINSEL0
R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x

Legend: R = Read only; R/W = Read/Write; -n = value after reset

Figure 3−4. Device Status Register (DEVSTAT) Description − 0x01B3 F004

Table 3−6. Device Status (DEVSTAT) Register Selection Bit Descriptions

BIT	NAME	DESCRIPTION
31:24	Reserved	Reserved. Read-only, writes have no effect.
23:19	PLLM	PLL multiply factor status bits. Shows the status of the PLL multiply mode selected; whether the CPU clock frequency equals the input clock frequency x1 (Bypass), x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x22, or x24. For more detailed information on the PLL multiply factors, see the *Clock PLL and Oscillator* section of this data sheet.
18	Reserved	Reserved. Read-only, writes have no effect.
17	OSC EXT RES	Oscillator external resistor status bit. Shows the status internal or external of the OSC bias resistor. 0 = Normal functional mode with **interna**l bias resistor. 1 = Normal functional mode with **external** bias resistor [default; internally tied high].
16	CLKINSEL	PLL input clock select status bit. Shows the status of whether the PLL input clock is CLKIN [pin high] or directly from the crystal oscillator (OSCIN and OSCOUT) [pin low] 0 = Crystal oscillator (OSCIN and OSCOUT). 1 = CLKIN (default).
15:13	Reserved	Reserved. Read-only, writes have no effect.
11	Reserved	Reserved. Read-only, writes have no effect.
10	HPI_WIDTH	HPI bus width control bit. Shows the status of whether the HPI bus operates in 32-bit mode or in 16-bit mode (default). 0 = HPI operates in 16-bit mode. (default). 1 = HPI operates in 32-bit mode.
9	Reserved	Reserved. Read-only, writes have no effect.
8	$\overline{\text{HPI\_EN}}$	$\overline{\text{HPI\_EN}}$ pin status bit. Shows the status at device reset of the $\overline{\text{HPI\_EN}}$ pin, which controls the HPI peripheral as enabled [default] or disabled. 0 = $\overline{\text{HPI\_EN}}$ pin is low, meaning the HPI peripheral is enabled (default). 1 = $\overline{\text{HPI\_EN}}$ pin is high, meaning the HPI peripheral is disabled.

Table 3−6. Device Status (DEVSTAT) Register Selection Bit Descriptions (Continued)

BIT	NAME	DESCRIPTION
12	CLKMODE3	Clock mode select status bits Shows the status (”1 or 0”) of the CLKMODE[3:0] select bits: Clock mode select for CPU clock frequency (CLKMODE[3:0]), for example: 0000– Bypass (x1) (default mode) For more details on the CLKMODE pins and the PLL multiply factors, see the *Clock PLL* section of this data sheet.
7	CLKMODE2	
6	CLKMODE1	
5	CLKMODE0	
4	LENDIAN	Device Endian mode (LENDIAN) Shows the status of whether the system is operating in Big Endian mode or Little Endian mode (default). 0 – System is operating in Big Endian mode 1 − System is operating in Little Endian mode (default)
3	BOOTMODE1	Bootmode configuration bits (AEA[22:21] pins) Shows the status of what device bootmode configuration is operational. Bootmode [1:0] 00 – No boot (default mode) 01 − HPI boot (based on $\overline{\text{HPI\_EN}}$ pin) 10 − Reserved 11 − EMIFA 8-bit ROM boot
2	BOOTMODE0	
1	AECLKINSEL1	EMIFA input clock select (AEA[20:19] pins) Shows the status of what clock mode is enabled or disabled for the EMIF. Clock mode select for EMIFA (AECLKIN_SEL[1:0]) 00 – AECLKIN (default mode) 01 − CPU/4 Clock Rate 10 − CPU/6 Clock Rate 11 − Reserved
0	AECLKINSEL0	

3.6 JTAG ID Register Description

The JTAG ID register is a read-only register that identifies to the customer the JTAG/Device ID. For the C6413/C6410 device, the JTAG ID register resides at address location 0x01B3 F008. The register hex value for the C6413/C6410 device is: 0x0007 902F. For the actual register bit names and their associated bit field descriptions, see Figure 3−5 and Table 3−7.

31−28	27−12	11−1	0
VARIANT (4-Bit)	PART NUMBER (16-Bit)	MANUFACTURER (11-Bit)	LSB
R-0000	R-0000 0000 1000 0100	R-0000 0010 111	R-1

Legend: R = Read only; -n = value after reset

Figure 3−5. JTAG ID Register Description − TMS320C6413/C6410 Register Value − 0x0007 902F

Table 3−7. JTAG ID Register Selection Bit Descriptions

BIT	NAME	DESCRIPTION
31:28	VARIANT	Variant (4-Bit) value. C6413/C6410 value: 0000.
27:12	PART NUMBER	Part Number (16-Bit) value. C6413/C6410 value: 0000 0000 1000 0100.
11−1	MANUFACTURER	Manufacturer (11-Bit) value. C6413/C6410 value: 0000 0010 111.
0	LSB	LSB. This bit is read as a “1” for C6413/C6410.

3.7 Multiplexed Pins

Multiplexed pins are pins that are shared by more than one peripheral and are internally multiplexed. Some of these pins are configured by software, and the others are configured by external pullup/pulldown resistors only at reset. Those muxed pins that are configured by software should ***not*** be programmed to switch functionalities during run-time. Those muxed pins that are configured by external pullup/pulldown resistors are mutually exclusive; only one peripheral has primary control of the function of these pins after reset. Table 3−8 identifies the multiplexed pins on the C6413/C6410 device; shows the default (primary) function and the default settings after reset; and describes the pins, registers, etc. necessary to configure specific multiplexed functions.

Table 3−8. C6413/C6410 Device Multiplexed Pins

MULTIPLEXED PINS		IPD/IPU†	DEFAULT FUNCTION	DEFAULT SETTING	DESCRIPTION
NAME	NO.				
CLKOUT4/GP0[1]	A2	IPU	CLKOUT4	GP1EN = 0 (disabled)	These pins are software-configurable. To use these pins as GPIO pins, the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input GPxDIR = 1: GPx pin is an output
CLKOUT6/GP0[2]	B3	IPU	CLKOUT6	GP2EN = 0 (disabled)	
HCNTL0/AFSR1[1]	Y6	IPU	HPI pin function HHWIL pin (HPI16 only)	TOUT0/$\overline{\text{HPI\_EN}}$ = 0, HD5 = 1 (32-Bit HPI enabled) McASP1 pins disabled.	By default, HPI32 is enabled upon reset (McASP1 is disabled). To enable the **McASP1** peripheral, the TOUT0/$\overline{\text{HPI\_EN}}$ pin must be **high** at reset either via an external pullup (PU) resistor (1 kΩ) or driven by a control device (disabling the HPI). **or** the McASP1 peripheral pins can be used if the HPI is used as a 16-bit width [$\overline{\text{HPI\_EN}}$ = 0, HD5 = 0]. The clocks and frame syncs select bits (AFCMUX[1:0]) located in the PERCFG register determine which of the clock and frame sync pairs are input to McASP1. For more detailed information, see the Device Configuration section of this data sheet.
HHWIL/AFSR1[2]	Y7				
HR/$\overline{\text{W}}$/AFSR1[3]	AA5				
$\overline{\text{HAS}}$/ACLKR1[1]	Y5				
$\overline{\text{HCS}}$/ACLKR1[2]	AA11				
$\overline{\text{HDS1}}$/ACLKR1[3]	AB11				
HD29/AMUTEIN1	W11				
HD28/AMUTE1	W10				
HD27/AHCLKX1	Y4				
HD26/AHCLKR1	AB4				
HD25/ACLKR1	AA9				
HD24/ACLKX1	AA4				
HD23/AFSR1	AB9				
HD22/AFSX1	AB5				
HD21/AXR1[5]	Y9	IPU	HPI pin function	TOUT0/$\overline{\text{HPI\_EN}}$ = 0, HD5 = 1 (32-Bit HPI enabled) McASP1 pins disabled.	By default, HPI32 is enabled upon reset (McASP1 is disabled). To enable the **McASP1** peripheral, the TOUT0/$\overline{\text{HPI\_EN}}$ pin must be **high** at reset either via an external pullup (PU) resistor (1 kΩ) or driven by a control device (disabling the HPI). **or** the McASP1 peripheral pins can be used if the HPI is used as a 16-bit width [$\overline{\text{HPI\_EN}}$ = 0, HD5 = 0]. McASP1 pin direction is controlled by the PDIR[x] bits in the McASP1PDIR register. McASP1PDIR = 0 input, = 1 output
HD20/AXR1[4]	AB8				
HD19/AXR1[3]	AA6				
HD18/AXR1[2]	AB7				
HD17/AXR1[1]	AA7				
HD16/AXR1[0]	AB6				

Table 3−8. C6413/C6410 Device Multiplexed Pins (Continued)

MULTIPLEXED PINS		IPD/IPU†	DEFAULT FUNCTION	DEFAULT SETTING	DESCRIPTION
NAME	NO.				
HD15/GP0[15]	Y12	IPU	HPI pin function	$\overline{\text{HPI\_EN}}$ = 0, HD5 = 1 (32-Bit HPI enabled) GPIO pins disabled.	By default, HPI is enabled upon reset (GP0[15:9] pins are disabled). To use GP0[15:9] as GPIO pins, the HPI needs to be disabled ($\overline{\text{HPI\_EN}}$ = 1, HD5 = x (don't care)), the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input GPxDIR = 1: GPx pin is an output
HD14/GP0[14]	AA12				
HD13/GP0[13]	AB13				
HD12/GP0[12]	Y14				
HD11/GP0[11]	AB14				
HD10/GP0[10]	AA15				
HD9/GP0[9]	Y16				
HD8/GP0[8]	AB16				

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

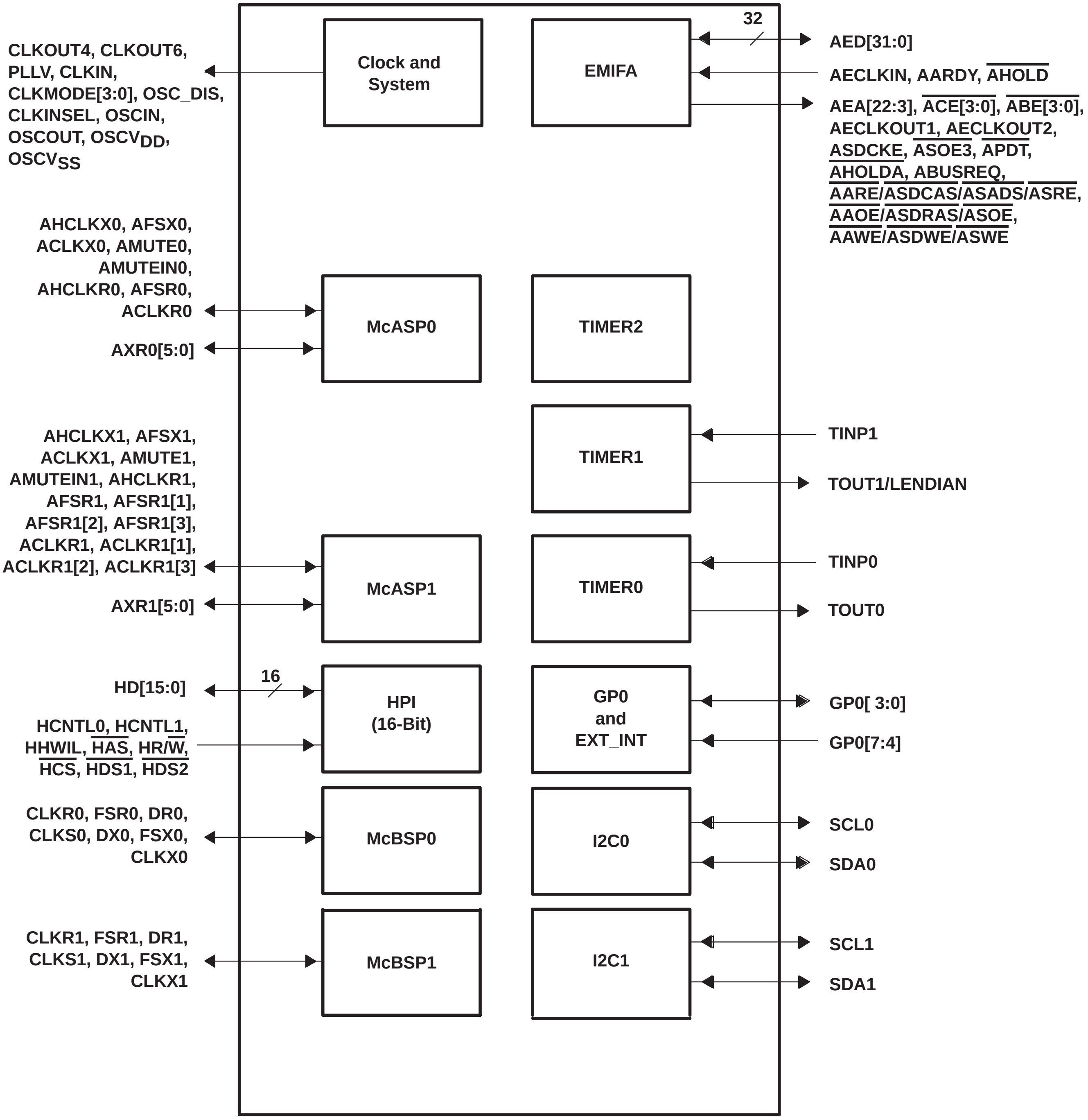
3.8 Debugging Considerations

It is recommended that external connections be provided to device configuration pins, including TOUT1/LENDIAN, AEA[22:19], TOUT0/$\overline{\text{HPI\_EN}}$, CLKINSEL, and OSC_DIS. Although internal pullup/pulldown resistors exist on these pins, providing external connectivity adds convenience to the user in debugging and flexibility in switching operating modes.

Note: If a configuration pin must be routed out from the device, the internal pullup/pulldown (IPU/IPD) resistor should not be relied upon. TI recommends the use of an external pullup/pulldown resistor.

3.9 Configuration Examples

Figure 3−6 illustrates an example of peripheral selections/options that are configurable on the C6413/C610 device.



PERCFG Register Value:	**0x0000_018F [CPU/4 option [default] and AFSR1, ACLKR1 pins selected]**
External Pins:	TOUT0/**HPI_EN = 0; HD5 = 0 (IPU)**

Figure 3−6. Configuration Example A
(HPI16 + 2 McASPs + 2 McBSPs +2 I2Cs + EMIF + 3 Timers + GPIO)

3.10 Terminal Functions

The terminal functions table (Table 3−9) identifies the external signal names, the associated pin (ball) numbers along with the mechanical package designator, the pin type (I, O/Z, or I/O/Z), whether the pin has any internal pullup/pulldown resistors and a functional pin description. For more detailed information on device configuration, peripheral selection, multiplexed/shared pins, and debugging considerations, see the Device Configurations section of this data sheet.

Table 3−9. Terminal Functions

SIGNAL		TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
NAME	NO.			
CLOCK/PLL CONFIGURATION				
CLKIN	A12	I	IPD	Clock Input. This clock is the input to the on-chip PLL.
CLKOUT4/GP0[1][§]	A2	I/O/Z	IPU	Clock output at 1/4 of the device speed (**O/Z**) [default] or this pin can be programmed as a GP0 1 pin (**I/O/Z**).
CLKOUT6/GP0[2][§]	B3	I/O/Z	IPU	Clock output at 1/6 of the device speed (**O/Z**) [default] or this pin can be programmed as a GP0 2 pin (**I/O/Z**).
CLKINSEL	A11	I	IPU	CLKIN select. Selects whether the PLL input clock is CLKIN [pin high] or directly from the crystal oscillator (OSCIN and OSCOUT) [pin low]. For proper device operation, this pin ***must*** be used in conjunction with the OSC_DIS pin.
CLKMODE3	C11	I	IPD	Clock mode selects • Selects whether the CPU clock frequency = input clock frequency x1 (Bypass), x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x22, or x24. For more details on the CLKMODE pins and the PLL multiply factors, see the Clock PLL section of this data sheet.
CLKMODE2	B10	I	IPD	
CLKMODE1	A13	I	IPD	
CLKMODE0	C13	I	IPD	
PLLV[¶]	C12	A		PLL voltage supply
OSCIN	A6	I	—	Crystal oscillator Input (XI)
OSCOUT	A7	O	—	Crystal oscillator output (XO)
OSCV$_{DD}$	B6	S	—	Power for crystal oscillator (1.2 V), ***Do not*** connect to board power CV$_{DD}$; for optimum performance, connected internally. If CLKIN is used instead of the oscillator, then this pin can be left open or connected to CV$_{DD}$.
OSCV$_{SS}$	C6	GND	—	Ground for crystal oscillator, ***Do not*** connect to board ground; for optimum performance, connected internally. If CLKIN is used instead of the oscillator, then this pin can be left open or connected to V$_{SS}$.
OSC_DIS	B7	I	IPU	Oscillator disable select. For proper device operation, this pin ***must*** follow the CLKINSEL pin operation. 0 − OSC enabled; CLKINSEL ***must*** be 0 1 − OSC disabled (default); CLKINSEL ***must*** be 1
JTAG EMULATION				
TMS	U3	I	IPU	JTAG test-port mode select
TDO	T4	O/Z	IPU	JTAG test-port data out
TDI	T1	I	IPU	JTAG test-port data in
TCK	T2	I	IPU	JTAG test-port clock
$\overline{\text{TRST}}$	U1	I	IPD	JTAG test-port reset. For IEEE 1149.1 JTAG compatibility, see the IEEE 1149.1 JTAG compatibility statement portion of this data sheet.
EMU0	R1	I/O/Z	IPU	Emulation pin 0[#]
EMU1	T3	I/O/Z	IPU	Emulation pin 1[#]

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

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

[¶] PLLV is not part of external voltage supply. See the Clock PLL and Oscillator section for information on how to connect this pin.

[#] The EMU0 and EMU1 pins are internally pulled up with 30-kΩ resistors; therefore, for emulation and normal operation, no external pullup/pulldown resistors are necessary. However, for boundary scan operation, pull down the EMU1 and EMU0 pins with a dedicated 1-kΩ resistor.

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
JTAG EMULATION (CONTINUED)				
EMU2	R2	I/O/Z	IPU	Emulation pin 2. Reserved for future use, leave unconnected.
EMU3	U2	I/O/Z	IPU	Emulation pin 3. Reserved for future use, leave unconnected.
EMU4	R3	I/O/Z	IPU	Emulation pin 4. Reserved for future use, leave unconnected.
EMU5	P2	I/O/Z	IPU	Emulation pin 5. Reserved for future use, leave unconnected.
EMU6	R4	I/O/Z	IPU	Emulation pin 6. Reserved for future use, leave unconnected.
EMU7	V2	I/O/Z	IPU	Emulation pin 7. Reserved for future use, leave unconnected.
EMU8	V1	I/O/Z	IPU	Emulation pin 8. Reserved for future use, leave unconnected.
EMU9	V3	I/O/Z	IPU	Emulation pin 9. Reserved for future use, leave unconnected.
EMU10	W3	I/O/Z	IPU	Emulation pin 10. Reserved for future use, leave unconnected.
EMU11	W2	I/O/Z	IPU	Emulation pin 11. Reserved for future use, leave unconnected.
RESETS, INTERRUPTS, AND GENERAL-PURPOSE INPUT/OUTPUTS				
$\overline{\text{RESET}}$	C9	I		Device reset
NMI	B9	I	IPD	Nonmaskable interrupt, edge-driven (rising edge). Any noise on the NMI pin may trigger an NMI interrupt; therefore, if the NMI pin is not used, it is recommended that the NMI pin be grounded versus relying on the IPD.
GP0[7]/EXT_INT7	Y1	I/O/Z	IPU	General-purpose input/output (GPIO) pins (**I/O/Z**) or external interrupts (**input only**). The default after reset setting is GPIO enabled as input-only. • When these pins function as External Interrupts [by selecting the corresponding interrupt enable register bit (IER.[7:4])], they are edge-driven and the polarity can be independently selected via the External Interrupt Polarity Register bits (EXTPOL.[3:0]).
GP0[6]/EXT_INT6	C4	I/O/Z	IPU	
GP0[5]/EXT_INT5	B4	I/O/Z	IPU	
GP0[4]/EXT_INT4	A4	I/O/Z	IPU	
HD15/GP0[15]	Y12	I/O/Z	IPU	Host-port data pins (**I/O/Z**) [default] or General-purpose input/output (GP0) [15:8] pins (**I/O/Z**) GP0 [3:0] pins (**I/O/Z**) Clock output at 1/6 of the device speed (**O/Z**) [default] or this pin can be programmed as a GP0 2 pin (**I/O/Z**). Clock output at 1/4 of the device speed (**O/Z**) [default] or this pin can be programmed as a GP0 1 pin (**I/O/Z**).
HD14/GP0[14]	AA12			
HD13/GP0[13]	AB13			
HD12/GP0[12]	Y14			
HD11/GP0[11]	AB14			
HD10/GP0[10]	AA15			
HD9/GP0[9]	Y16			
HD8/GP0[8]	AB16			
GP0[3]	B13	I/O/Z	IPD	
CLKOUT6/GP0[2][§]	B3	I/O/Z	IPU	
CLKOUT4/GP0[1][§]	A2	I/O/Z	IPU	
GP0[0]	D13	I/O/Z	IPD	

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
EMIFA (32-BIT) − CONTROL SIGNALS COMMON TO ALL TYPES OF MEMORY				
ACE3	H19	O/Z	IPU	EMIFA memory space enables • Enabled by bits 28 through 31 of the word address • Only one pin is asserted during any external data access
ACE2	N20	O/Z	IPU	
ACE1	R20	O/Z	IPU	
ACE0	F20	O/Z	IPU	
ABE3	AB21	O/Z	IPU	EMIFA byte-enable control • Decoded from the low-order address bits. The number of address bits or byte enables used depends on the width of external memory. • Byte-write enables for most types of memory • Can be directly connected to SDRAM read and write mask signal (SDQM)
ABE2	P21	O/Z	IPU	
ABE1	A22	O/Z	IPU	
ABE0	D16	O/Z	IPU	
APDT	T19	O/Z	IPU	EMIFA peripheral data transfer, allows direct transfer between external peripherals
EMIFA (32-BIT) − BUS ARBITRATION				
AHOLDA	J21	O	IPU	EMIFA hold-request-acknowledge to the host
AHOLD	J22	I	IPU	EMIFA hold request from the host
ABUSREQ	R19	O	IPU	EMIFA bus request output
EMIFA (32-BIT) − ASYNCHRONOUS/SYNCHRONOUS MEMORY CONTROL				
AECLKIN	K22	I	IPD	EMIFA external input clock. The EMIFA input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) is selected at reset via the pullup/pulldown resistors on the AEA[20:19] pins. AECLKIN is the default for the EMIFA input clock.
AECLKOUT2	U22	O/Z	IPD	EMIFA output clock 2. Programmable to be EMIFA input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) frequency divided-by-1, -2, or -4.
AECLKOUT1	F22	O/Z	IPD	EMIFA output clock 1 [at EMIFA input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) frequency].
AARE/ ASDCAS/ ASADS/ASRE	D20	O/Z	IPU	EMIFA asynchronous memory read-enable/SDRAM column-address strobe/programmable synchronous interface-address strobe or read-enable • For programmable synchronous interface, the RENEN field in the CE Space Secondary Control Register (CExSEC) selects between ASADS and ASRE: If RENEN = 0, then the ASADS/ASRE signal functions as the ASADS signal. If RENEN = 1, then the ASADS/ASRE signal functions as the ASRE signal.
AAOE/ ASDRAS/ ASOE	E20	O/Z	IPU	EMIFA asynchronous memory output-enable/SDRAM row-address strobe/programmable synchronous interface output-enable
AAWE/ ASDWE/ ASWE	C20	O/Z	IPU	EMIFA asynchronous memory write-enable/SDRAM write-enable/programmable synchronous interface write-enable
ASDCKE	K21	O/Z	IPU	EMIFA SDRAM clock-enable (used for self-refresh mode). • If SDRAM is not in system, ASDCKE can be used as a general-purpose output.
ASOE3	P19	O/Z	IPU	EMIFA synchronous memory output-enable for ACE3 (for glueless FIFO interface)
AARDY	L21	I	IPU	Asynchronous memory ready input

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
EMIFA (32-BIT) − ADDRESS				
AEA22	M21	I/O/Z	IPD	EMIFA external address (word address) Note: EMIF address numbering for the C6413/C6410 devices starts with AEA3 to maintain signal name compatibility with other C64x™ devices (e.g., C6411, C6414, C6415, and C6416) [see the 64-bit EMIF adressing scheme in the *TMS320C6000 DSP External Memory Interface (EMIF) Reference Guide* (literature number SPRU266)]. Note: If a configuration pin must be routed out from the device, the internal pullup/pulldown (IPU/IPD) resistor should not be relied upon. TI recommends the use of an external pullup/pulldown resistor. • Also controls initialization of DSP modes at reset (**I**) via pullup/pulldown resistors − Boot mode (AEA[22:21]): 00 – No boot (default mode) 01 − HPI boot (based on $\overline{\text{HPI\_EN}}$ pin) 10 − Reserved 11 − EMIFA 8-bit ROM boot − EMIF clock select − AEA[20:19]: Clock mode select for EMIFA (AECLKIN_SEL[1:0]) 00 – AECLKIN (default mode) 01 − CPU/4 Clock Rate 10 − CPU/6 Clock Rate 11 − Reserved For more details, see the Device Configurations section of this data sheet.
AEA21	N21			
AEA20	P22			
AEA19	N22			
AEA18	H22	O/Z	IPD	
AEA17	H21			
AEA16	J20			
AEA15	H20			
AEA14	G20			
AEA13	K20			
AEA12	B21			
AEA11	B22			
AEA10	D21			
AEA9	D22			
AEA8	E21			
AEA7	E22			
AEA6	F21			
AEA5	M20			
AEA4	J19			
AEA3	L20			
EMIFA (32-BIT) − DATA				
AED31	W21	I/O/Z	IPU	EMIFA external data
AED30	W22			
AED29	V20			
AED28	W20			
AED27	AA22			
AED26	Y20			
AED25	AA21			
AED24	AB22			
AED23	P20			
AED22	R22			
AED21	R21			
AED20	U21			
AED19	V21			
AED18	T20			
AED17	V22			
AED16	U20			
AED15	A18			
AED14	D17			

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

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
EMIFA (32-BIT) − DATA (CONTINUED)				
AED13	B18	I/O/Z	IPU	EMIFA external data
AED12	C18			
AED11	A19			
AED10	C19			
AED9	B19			
AED8	A21			
AED7	D15			
AED6	A15			
AED5	B15			
AED4	C15			
AED3	A16			
AED2	C16			
AED1	B16			
AED0	C17			
TIMER 2				
	−			No external pins. The timer 2 peripheral pins are ***not*** pinned out as external pins.
TIMER 1				
TOUT1/LENDIAN	AA1	I/O/Z	IPU	Timer 1 output (**O/Z**) or device endian mode (**I**). Also controls initialization of DSP modes at reset via pullup/pulldown resistors − Device Endian mode 0 – Big Endian 1 − Little Endian (default) For more details on LENDIAN, see the Device Configurations section of this data sheet.
TINP1	AB1	I	IPD	Timer 1 or general-purpose input
TIMER 0				
TOUT0/$\overline{\text{HPI\_EN}}$	AA2	I/O/Z	IPD	Timer 0 output pin and HPI enable $\overline{\text{HPI\_EN}}$ pin function The $\overline{\text{HPI\_EN}}$ pin function selects whether the HPI peripheral or McASP1 peripheral, and GP0[15:8] pins are functionally enabled 0 – HPI is enabled and the McASP1 peripheral and GP0 [15:8] pins are disabled (default mode); [HPI32, if HD5 = 1; HPI16 if HD5 = 0] 1 − HPI I is disabled and the McASP1 peripheral and GP0 [15:8] pins are enabled For more details, see the Device Configurations section of this data sheet.
TINP0	AB2	I	IPD	Timer 0 or general-purpose input
INTER-INTEGRATED CIRCUIT 1 (I2C1)				
SCL1	AA18	I/O/Z	—	I2C1 clock. When the I2C module is used, use an external pullup resistor on this pin.
SDA1	AA19	I/O/Z	—	I2C1 data. When I2C is used, ensure there is an external pullup resistors on this pin.
INTER-INTEGRATED CIRCUIT 0 (I2C0)				
SCL0	AB18	I/O/Z	—	I2C0 clock. When I2C is used, ensure there is an external pullup resistors on this pin.
SDA0	AB19	I/O/Z	—	I2C0 data. When I2C is used, ensure there is an external pullup resistors on this pin.

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

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP1)				
CLKR1	G3	I/O/Z	IPD	McBSP1 receive clock
FSR1	G2	I/O/Z	IPD	McBSP1 receive frame sync
DR1	F1	I	IPD	McBSP1 receive data
CLKS1	G1	I	IPD	McBSP1 external clock source (as opposed to internal)
DX1	H2	O/Z	IPD	McBSP1 transmit data
FSX1	H3	I/O/Z	IPD	McBSP1 transmit frame sync
CLKX1	H1	I/O/Z	IPD	McBSP1 transmit clock
MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP0)				
CLKR0	C2	I/O/Z	IPD	McBSP0 receive clock
FSR0	D1	I/O/Z	IPD	McBSP0 receive frame sync
DR0	D2	I	IPD	McBSP0 receive data
CLKS0	D3	I	IPD	McBSP0 external clock source (as opposed to internal)
DX0	E2	O/Z	IPD	McBSP0 transmit data
FSX0	E4	I/O/Z	IPD	McBSP0 transmit frame sync
CLKX0	E3	I/O/Z	IPD	McBSP0 transmit clock
MULTICHANNEL AUDIO SERIAL PORT 0 (McASP0)				
AHCLKX0	N1	I/O/Z	IPD	McASP0 transmit high-frequency master clock.
AFSX0	M2	I/O/Z	IPD	McASP0 transmit frame sync or left/right clock (LRCLK).
ACLKX0	M1	I/O/Z	IPD	McASP0 transmit bit clock.
AMUTE0	K4	I/O/Z	IPD	McASP0 mute output.
AMUTEIN0	J4	I	IPD	McASP0 mute input.
AHCLKR0	L1	I/O/Z	IPD	McASP0 receive high-frequency master clock.
AFSR0	K2	I/O/Z	IPD	McASP0 receive frame sync or left/right clock (LRCLK).
ACLKR0	K1	I/O/Z	IPD	McASP0 receive bit clock.
AXR0[5]	P3	I/O/Z	IPD	McASP0 TX/RX data pin [5].
AXR0[4]	N3			McASP0 TX/RX data pin [4].
AXR0[3]	M3			McASP0 TX/RX data pins [3].
AXR0[2]	L3			McASP0 TX/RX data pin [2].
AXR0[1]	K3			McASP0 TX/RX data pin [1].
AXR0[0]	L2			McASP0 TX/RX data pins[0].

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

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
MCASP1				
HCNTL0/AFSR1[1]	Y6	I	IPU	Host control − selects between control, address, or data registers (**I**) [default] or McASP1 receive frame sync input 1 (**I**).
HHWIL/AFSR1[2]	Y7			Host half-word select − first or second half-word (not necessarily high or low order) [For HPI16 bus width selection only] (**I**) [default] oror McASP1 receive frame sync input 2 (**I**) .
HR/$\overline{\text{W}}$/AFSR1[3]	AA5			Host read or write select (**I**) [default] or McASP1 receive frame sync input 3 (**I**).
$\overline{\text{HAS}}$/ACLKR1[1]	Y5			Host address strobe (**I**) [default] or McASP1 receive clock input 1 (**I**).
$\overline{\text{HCS}}$/ACLKR1[2]	AA11			Host chip select (**I**) [default] or McASP1 receive clock input 2 (**I**).
$\overline{\text{HDS1}}$/ACLKR1[3]	AB11			Host data strobe 1 (**I**) [default] or McASP1 receive clock input 3 (**I**).
HD27/AHCLKX1	Y4	I/O/Z	IPU	Host-port data pin 27 (**I/O/Z**) [default] or McASP1 transmit high-frequency master clock (**I/O/Z**).
HD22/AFSX1	AB5	I/O/Z	IPU	Host-port data pin 22 (**I/O/Z**) [default] or McASP1 transmit frame sync or left/right clock (LRCLK) (**I/O/Z**) .
HD24/ACLKX1	AA4	I/O/Z	IPU	Host-port data pin 24 (**I/O/Z**) [default] or McASP1 transmit bit clock (**I/O/Z**).
HD28/AMUTE1	W10	I/O/Z	IPU	Host-port data pin 28 (**I/O/Z**) [default] or McASP1 mute output (**I/O/Z**).
HD29/AMUTEIN1	W11	I	IPU	Host-port data pin 29 (**I/O/Z**) [default] or McASP1 mute input (**I**).
HD26/AHCLKR1	AB4	I/O/Z	IPU	Host-port data pin 26 (**I/O/Z**) [default] or McASP1 receive high-frequency master clock (**I/O/Z**).
HD23/AFSR1	AB9	I/O/Z	IPU	Host-port data pin 23 (**I/O/Z**) [default] or McASP1 receive frame sync or left/right clock (LRCLK) (**I/O/Z**).
HD25/ACLKR1	AA9	I/O/Z	IPU	Host-port data pin 25 (**I/O/Z**) [default] or McASP1 receive bit clock (**I/O/Z**).
HD21/AXR1[5]	Y9	I/O/Z	IPU	Host-port data pins [21:16] (**I/O/Z**) [default] or McASP1 TX/RX data pins [5:0] (**I/O/Z**).
HD20/AXR1[4]	AB8			
HD19/AXR1[3]	AA6			
HD18/AXR1[2]	AB7			
HD17/AXR1[1]	AA7			
HD16/AXR1[0]	AB6			
HOST-PORT INTERFACE (HPI)				
$\overline{\text{HINT}}$	AA8	O/Z	IPU	Host interrupt from DSP to host (**O**)
HCNTL1	W7	I	IPU	Host control − selects between control, address, or data registers (**I**)
HCNTL0/AFSR1[1]	Y6	I	IPU	Host control − selects between control, address, or data registers (**I**) [default] or McASP1 receive frame sync input 1 (**I**).
HHWIL/AFSR1[2]	Y7			Host half-word select − first or second half-word (not necessarily high or low order) [For HPI16 bus width selection only] (**I**) [default] or McASP1 receive frame sync input 2 (**I**).
HR/$\overline{\text{W}}$/AFSR1[3]	AA5			Host read or write select (**I**) [default] or McASP1 receive frame sync input 3 (**I**).
$\overline{\text{HAS}}$/ACLKR1[1]	Y5			Host address strobe (**I**) [default] or McASP1 receive clock input 1 (**I**).
$\overline{\text{HCS}}$/ACLKR1[2]	AA11			Host chip select (**I**) [default] or McASP1 receive clock input 2 (**I**).
$\overline{\text{HDS1}}$/ACLKR1[3]	AB11			Host data strobe 1 (**I**) [default] or McASP1 receive clock input 3 (**I**).
$\overline{\text{HDS2}}$	AB12	I	IPU	Host data strobe 2 (**I**)
$\overline{\text{HRDY}}$	Y10	O/Z	IPU	Host ready from DSP to host (**O**)

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

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

Table 3−9. Terminal Functions (Continued)

NAME	NO.	TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
HOST-PORT INTERFACE (HPI) (CONTINUED)				
HD31	Y8	I/O/Z	IPU	Host-port data pin 31 (**I/O/Z**)
HD30	Y11	I/O/Z	IPU	Host-port data pin 30 (**I/O/Z**)
HD29/AMUTEIN1	W11	I	IPU	Host-port data pin 29 (**I/O/Z**) [default] or McASP1 mute input (**I**).
HD28/AMUTE1	W10	I/O/Z	IPU	Host-port data pin 28 (**I/O/Z**) [default] or McASP1 mute output (**I/O/Z**).
HD27/AHCLKX1	Y4	I/O/Z	IPU	Host-port data pin 27 (**I/O/Z**) [default] or McASP1 transmit high-frequency master clock (**I/O/Z**).
HD26/AHCLKR1	AB4	I/O/Z	IPU	Host-port data pin 26 (**I/O/Z**) [default] or McASP1 receive high-frequency master clock (**I/O/Z**).
HD25/ACLKR1	AA9	I/O/Z	IPU	Host-port data pin 25 (**I/O/Z**) [default] or McASP1 receive bit clock (**I/O/Z**).
HD24/ACLKX1	AA4	I/O/Z	IPU	Host-port data pin 24 (**I/O/Z**) [default] or McASP1 transmit bit clock (**I/O/Z**).
HD23/AFSR1	AB9	I/O/Z	IPU	Host-port data pin 23 (**I/O/Z**) [default] or McASP1 receive frame sync or left/right clock (LRCLK) (**I/O/Z**).
HD22/AFSX1	AB5	I/O/Z	IPU	Host-port data pin 22 (**I/O/Z**) [default] or McASP1 transmit frame sync or left/right clock (LRCLK) (**I/O/Z**).
HD21/AXR1[5]	Y9	I/O/Z	IPU	Host-port data [21:16] pin (**I/O/Z**) [default] or McASP1 TX/RX data pins [5:0] (**I/O/Z**).
HD20/AXR1[4]	AB8			
HD19/AXR1[3]	AA6			
HD18/AXR1[2]	AB7			
HD17/AXR1[1]	AA7			
HD16/AXR1[0]	AB6			
HD15/GP0[15]	Y12	I/O/Z	IPU	Host-port data [15:8] pins (**I/O/Z**) [default] or General-purpose input/output (GP0) [15:8] pins (**I/O/Z**).
HD14/GP0[14]	AA12			
HD13/GP0[13]	AB13			
HD12/GP0[12]	Y14			
HD11/GP0[11]	AB14			
HD10/GP0[10]	AA15			
HD9/GP0[9]	Y16			
HD8/GP0[8]	AB16			
HD7	W12	I/O/Z	IPU	Host-port data [7:0] pins (**I/O/Z**) Host-Port bus width user-configurable at device reset via a 1-kΩ pullup/ pulldown resistor on the HD5 pin (**I**): HD5 pin = 0: HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the high-impedance state.) HD5 pin = 1: HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.)
HD6	AA13			
HD5	Y13			
HD4	AA14			
HD3	AB15			
HD2	AA16			
HD1	Y15			
HD0	W15			

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

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

Table 3−9. Terminal Functions (Continued)

SIGNAL NAME	NO.	TYPE[†]	IPD/ IPU[‡]	DESCRIPTION
RESERVED FOR TEST				
RSV	U4	A	—	Reserved. This pin must be connected directly to CV_{DD} for proper device operation.
RSV	F3	A	—	Reserved. This pin must be connected directly to DV_{DD} for proper device operation.
RSV	C8	I	IPD	Reserved. This pin must be connected directly to V_{SS} for proper device operation.
RSV	B11	A	—	Reserved (leave unconnected, ***do not*** connect to power or ground). If the signal must be routed from the device, the internal pull−up/down resistance should not be relied upon and an external pull−up/down should be used.
	B12	I	—	
	C10	O	IPU	
	D7	O/Z	—	
	D8	O/Z	—	
SUPPLY VOLTAGE PINS				
DV_{DD}	A3	S		3.3-V supply voltage
	A5			
	A8			
	A9			
	A14			
	A17			
	A20			
	B1			
	C22			
	E1			
	G22			
	J1			
	M22			
	P1			
	T22			
	W1			
	Y2			
	Y17			
	Y19			
	Y22			
	AB3			
	AB10			
	AB17			
	AB20			

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

Table 3−9. Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)				
CV_{DD}	D5 D6 D9 D11 D12 D14 D18 E19 F19 G4 H4 L19 M4 M19 N4 V4 V19 W5 W9 W13 W16 W18	S		1.2-V supply voltage (-400, -500 devices)
GROUND PINS				
V_{SS}	A1 A10 B2 B5 B8 B14 B17 B20 C1 C3 C5 C7	GND		Ground pins

[†] I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog Signal

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

Table 3−9. Terminal Functions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	IPD/ IPU‡	DESCRIPTION
GROUND PINS (CONTINUED)				
V_{SS}	C14	GND		Ground pins
	C21			
	D4			
	D10			
	D19			
	F2			
	F4			
	G19			
	G21			
	J2			
	J3			
	K19			
	L4			
	L22			
	N2			
	N19			
	P4			
	T21			
	U19			
	W4			
	W6			
	W8			
	W14			
	W17			
	W19			
	Y3			
	Y18			
	Y21			
	AA3			
	AA10			
	AA17			
	AA20			

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

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

3.11 Development Support

In case the customer would like to develop their own features and software on the TMS320C6413/C6410 device, 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 tool's support documentation is electronically available within the Code Composer Studio™ Integrated Development Environment (IDE).

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)

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

Code Composer Studio, DSP/BIOS, and XDS are trademarks of Texas Instruments.

3.12 Device Support

3.12.1 Device and Development-Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all DSP devices and support tools. Each DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., **TMS**320C6412GDK600). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX / TMDX) through fully qualified production devices/tools (TMS / TMDS).

Device development evolutionary flow:

TMX	Experimental device that is not necessarily representative of the final device's electrical specifications.
TMP	Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification.
TMS	Fully qualified production device.

Support tool development evolutionary flow:

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

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

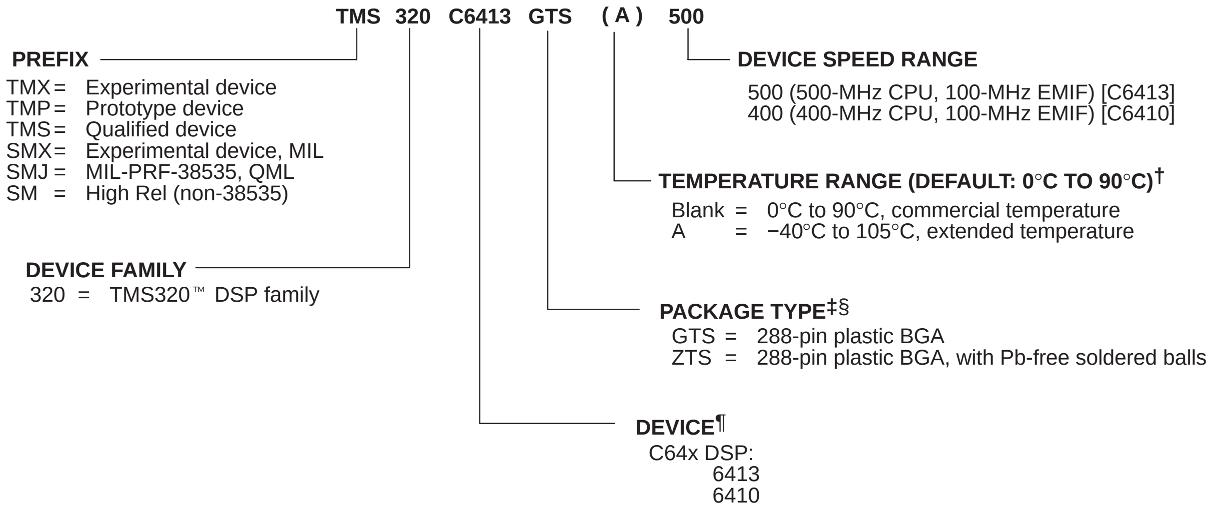
"Developmental product is intended for internal evaluation purposes."

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

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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, GTS), the temperature range (for example, "A" is the extended temperature range), and the device speed range in megahertz (for example, -500 is 500 MHz). Figure 3−7 provides a legend for reading the complete device name for any TMS320C6000™ DSP platform member.

The ZTS package, like the GTS package, is a 288-ball plastic BGA *only* with PB-free balls. For device part numbers and further ordering information for TMS320C6413/C6410 in the GTS and ZTS package types, see the TI website (http://www.ti.com) or contact your TI sales representative.



TMS 320 C6413 GTS (A) 500

PREFIX
TMX = Experimental device
TMP = Prototype device
TMS = Qualified device
SMX = Experimental device, MIL
SMJ = MIL-PRF-38535, QML
SM = High Rel (non-38535)

DEVICE FAMILY
320 = TMS320™ DSP family

DEVICE SPEED RANGE
500 (500-MHz CPU, 100-MHz EMIF) [C6413]
400 (400-MHz CPU, 100-MHz EMIF) [C6410]

TEMPERATURE RANGE (DEFAULT: 0°C TO 90°C)†
Blank = 0°C to 90°C, commercial temperature
A = −40°C to 105°C, extended temperature

PACKAGE TYPE‡§
GTS = 288-pin plastic BGA
ZTS = 288-pin plastic BGA, with Pb-free soldered balls

DEVICE¶
C64x DSP:
6413
6410

† The extended temperature “A version” devices may have different operating conditions than the commercial temperature devices. For more details, see the recommended operating conditions portion of this data sheet.
‡ BGA = Ball Grid Array
§ The ZTS mechanical package designator represents the version of the GTS package with Pb-free balls. For more detailed information, see the *Mechanical Data* section of this document.
¶ For actual device part numbers (P/Ns) and ordering information, see the TI website (www.ti.com).

Figure 3−7. TMS320C6413/C6410 DSP Device Nomenclature

For additional information, see the *TMS320C6413, TMS320C6410 Digital Signal Processors Silicon Errata* (literature number SPRZ219)

3.12.2 Documentation Support

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

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

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

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

The *TMS320C6000 DSP Multichannel Audio Serial Port (McASP) Reference Guide* (literature number SPRU041) describes the functionality of the McASP peripheral.

TMS320C6000 DSP Inter-Integrated Circuit (I2C) Module Reference Guide (literature number SPRU175) describes the functionality of the I2C peripherals available on the C6413/C6410 device except for the additional interrupt and new GPIO capability. For more detailed information on the additional interrupt and GPIO capability, see the *I2C* section of this data manual and the *TMS320C6410/C6413 DSP Inter-Integrated Circuit (I2C) Module Reference Guide* (literature number SPRZ221).

The *TMS320C6413, TMS320C6410 Digital Signal Processors Silicon Errata* (literature number SPRZ219) describes the known exceptions to the functional specifications for particular silicon revisions of the TMS320C6413 and TMS320C6410 devices.

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

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

TMS320 is a trademark of Texas Instruments.

4 Peripherals Detailed Description (Device-Specific)

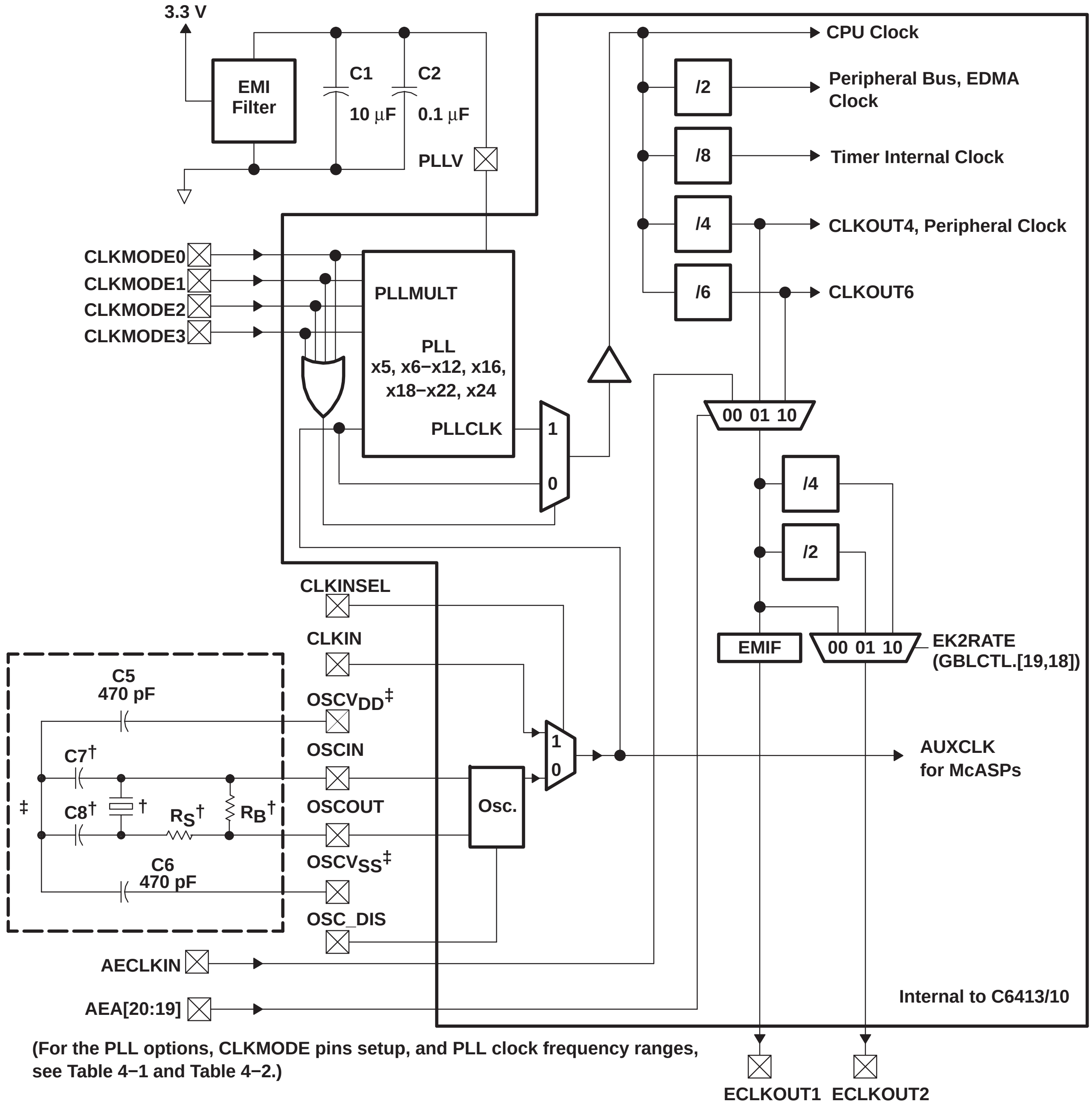
4.1 Clock PLL and Oscillator

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

To use the PLL to generate the CPU clock, the external PLL filter circuit must be properly designed. Figure 4−1 shows the external PLL circuitry for either x1 (PLL bypass) or other PLL multiply modes.

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

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



(For the PLL options, CLKMODE pins setup, and PLL clock frequency ranges, see Table 4−1 and Table 4−2.)

† Exact values for these components depend on choice of crystal. For recommended crystal and component values, see Table 4−3.

‡ Do not connect any of these nodes to board power or ground if the oscillator is used. They are internally connected for proper operation. If CLKIN is being used instead of the oscillator, then $OSCV_{DD}$ and $OSCV_{SS}$ may either be left open, or $OSCV_{DD}$ may be tied to CV_{DD} and $OSCV_{SS}$ may be tied to ground.

NOTES:
A. Place all PLL external components (C1, C2, and the EMI Filter) as close to the C6000™ DSP device as possible. For the best performance, TI recommends that all the PLL external components be on a single side of the board without jumpers, switches, or components other than the ones shown.
B. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (C1, C2, and the EMI Filter).
C. The 3.3-V supply for the EMI filter must be from the same 3.3-V power plane supplying the I/O voltage, DV_{DD}.
D. EMI filter manufacturer TDK part number ACF451832-333, -223, -153, -103. Panasonic part number EXCCET103U.
E. If CLKIN is used instead of OSCIN, tie OSCIN to Ground to minimize noise and current. (Do not leave OSCIN floating.)

Figure 4−1. External PLL Circuitry for Either PLL Multiply Modes or x1 (Bypass) Mode

For proper C6413/C6410 device operation, the CLKINSEL pin ***must*** be used in conjunction with the OSC_DIS pin. The OSC_DIS pin ***must*** follow the CLKINSEL pin operation. For more details on these two configuration pins, see the *Device Configuration at Device Reset* section of this data sheet.

Table 4−1. TMS320C6413 PLL Multiply Factor Options, Clock Frequency Ranges, and Typical Lock Time for −500 Devices[†]

GTS and ZTS PACKAGES − 23 x 23 mm BGA									
CLKMODE[3:0]				**CLKMODE (PLL MULTIPLY FACTORS)**	**CLKIN RANGE (MHz)**	**CPU CLOCK FREQUENCY RANGE (MHz)**	**OSCIN RANGE (MHz)**	**CPU CLOCK FREQUENCY RANGE (MHz)**	**TYPICAL LOCK TIME (μs)[‡]**
0	0	0	0	Bypass (x1)	12−100	12−100	12−30	12−30	N/A
0	0	0	1	x5	28−100	140−500	28−30	140−150	75
0	0	1	0	x6	23−83	140−500	23−30	140−180	
0	0	1	1	x7	20−71	140−500	20−30	140−210	
0	1	0	0	x8	17−63	140−500	17−30	140−240	
0	1	0	1	x9	15−56	140−500	15−30	140−270	
0	1	1	0	x10	14−50	140−500	14−30	140−300	
0	1	1	1	x11	12−45	140−500	12−30	140−330	
1	0	0	0	x12	12−42	144−500	12−30	144−360	
1	0	0	1	x16	12−31	192−500	12−30	192−480	
1	0	1	0	x18	12−28	216−500	12−28	216−500	
1	0	1	1	x19	12−26	228−500	12−26	228−500	
1	1	0	0	x20	12−25	240−500	12−25	240−500	
1	1	0	1	x21	12−24	252−500	12−24	252−500	
1	1	1	0	x22	12−23	264−500	12−23	264−500	
1	1	1	1	x24	12−21	288−500	12−21	288−500	

[†] Use external pullup resistors on the CLKMODE pins (CLKMODE1 and CLKMODE0) to set the C6413/C6410 device to one of the valid PLL multiply clock modes (x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x22, or x24). With internal pulldown resistors on the CLKMODE pins (CLKMODE3, CLKMODE2, CLKMODE1, CLKMODE0), the default clock mode is x1 (bypass).

[‡] Under some operating conditions, the maximum PLL lock time may vary by as much as 150% from the specified typical value. For example, if the typical lock time is specified as 100 μs, the maximum value may be as long as 250 μs.

Table 4−2. TMS320C6410 PLL Multiply Factor Options, Clock Frequency Ranges, and Typical Lock Time for −400 Devices[†]

GTS and ZTS PACKAGES − 23 x 23 mm BGA										
CLKMODE[3:0]				**CLKMODE (PLL MULTIPLY FACTORS)**	**CLKIN RANGE (MHz)**	**CPU CLOCK FREQUENCY RANGE (MHz)**	**OSCIN RANGE (MHz)**	**CPU CLOCK FREQUENCY RANGE (MHz)**	**TYPICAL LOCK TIME (μs)[‡]**
0	0	0	0	Bypass (x1)	12−100	12−100	12−30	12−30	N/A
0	0	0	1	x5	28−80	140−400	28−30	140−150	75
0	0	1	0	x6	23−67	140−400	23−30	140−180	
0	0	1	1	x7	20−57	140−400	20−30	140−210	
0	1	0	0	x8	17−50	140−400	17−30	140−240	
0	1	0	1	x9	15−44	140−400	15−30	140−270	
0	1	1	0	x10	14−40	140−400	14−30	140−300	
0	1	1	1	x11	12−36	140−400	12−30	140−330	
1	0	0	0	x12	12−33	144−400	12−30	144−360	
1	0	0	1	x16	12−25	192−400	12−25	192−400	
1	0	1	0	x18	12−22	216−400	12−22	216−400	
1	0	1	1	x19	12−21	228−400	12−21	228−400	
1	1	0	0	x20	12−20	240−400	12−20	240−400	
1	1	0	1	x21	12−19	252−400	12−19	252−400	
1	1	1	0	x22	12−18	264−400	12−18	264−400	
1	1	1	1	x24	12−17	288−400	12−17	288−400	

[†] Use external pullup resistors on the CLKMODE pins (CLKMODE1 and CLKMODE0) to set the C6413/C6410 device to one of the valid PLL multiply clock modes (x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x22, or x24). With internal pulldown resistors on the CLKMODE pins (CLKMODE3, CLKMODE2, CLKMODE1, CLKMODE0), the default clock mode is x1 (bypass).

[‡] Under some operating conditions, the maximum PLL lock time may vary by as much as 150% from the specified typical value. For example, if the typical lock time is specified as 100 μs, the maximum value may be as long as 250 μs.

For the lowest jitter on the oscillator circuit, it is recommended that a pair of 470-pF capacitors be connected between isolated (not directly connected to the board supply) OSCV_{DD} and OSCV_{SS} pins. This helps to cancel out switching noise from other circuits on the DSP device.

Table 4−3 shows a recommended crystal and tank circuit values for the C6413/C6410 PLL circuitry.

Table 4−3. Crystal and Tank Circuit Recommendations

Components	**RECOMMENDED PART NUMBERS or VALUES**		**MANUFACTURER**
Crystal	24.576 MHz	1AS245766AHA (SMD-49)	KDS™ Diashinku Corp.
		1AF245766AAA (AT-49)	
	22.5792 MHz	1AS225796AG (SMD-49)	
		1AF225796A (AT-49)	
R_B	1 MΩ		—
R_S	0 Ω		—
C7 C8	8 pF		—

4.2 Host-Port Interface (HPI) Peripheral

The TMS320C6413/C6410 device includes a user-configurable 16-bit or 32-bit Host-port interface (HPI16/HPI32). On the C6413/C6410 device the HPI peripheral pins are muxed with the McASP1 and GP0 peripheral pins. By default, the HPI peripheral pin functions are enabled. For more detailed information on the C6413/C6410 device pin muxing, see the Device Configurations section of this data sheet.

The HPI peripheral can be disabled or enabled at reset through the HPI enable function of the TOUT0/$\overline{\text{HPI\_EN}}$ pin. The HPI is enabled when the TOUT0/$\overline{\text{HPI\_EN}}$ pin is sampled low at reset and it is disabled if the pin is sample high at reset. The TOUT0/$\overline{\text{HPI\_EN}}$ pin has an internal pulldown that enables the HPI by default. However, the HPI can be disabled via an external pullup resistor or by having an external device such as an FPGA/CPLD drive that pin high at reset. In the latter case, the external device should ensure it has stopped driving this pin to avoid contention. The HPI enable function can only be set a reset and cannot be changed via software.

The HD5 pin controls the HPI_WIDTH, allowing the user to configure the HPI as a 16-bit or 32-bit peripheral.

For more details on HPI peripheral configuration and the associated pins, see the Device Configurations section of this data sheet.

4.3 Multichannel Audio Serial Port (McASP) Peripheral

The TMS320C6413/C6410 device includes two multichannel audio serial port (McASP) interface peripheral (McASP0 and McASP1). On the C6413/C6410 device the McASP1 peripheral pins are muxed with the HPI peripheral pins. By default, the HPI peripheral pin functions are enabled. For the C6413/C6410 device McASP1 is a standalone peripheral, not muxed. For more detailed information on the C6413/C6410 device pin muxing, see the Device Configurations section of this data sheet.

The McASP is a serial port optimized for the needs of multichannel audio applications.

The McASP consists of a transmit and receive section. These sections can operate completely independently with different data formats, separate master clocks, bit clocks, and frame syncs or alternatively, the transmit and receive sections may be synchronized. The McASP module also includes a pool of 16 shift registers that may be configured to operate as either transmit data, receive data, or general-purpose I/O (GPIO).

The transmit section of the McASP can transmit data in either a time-division-multiplexed (TDM) synchronous serial format or in a digital audio interface (DIT) format where the bit stream is encoded for S/PDIF, AES-3, IEC-60958, CP-430 transmission. The receive section of the McASP supports the TDM synchronous serial format.

The McASP can support one transmit data format (either a TDM format or DIT format) and one receive format at a time. All transmit shift registers use the same format and all receive shift registers use the same format. However, the transmit and receive formats need not be the same.

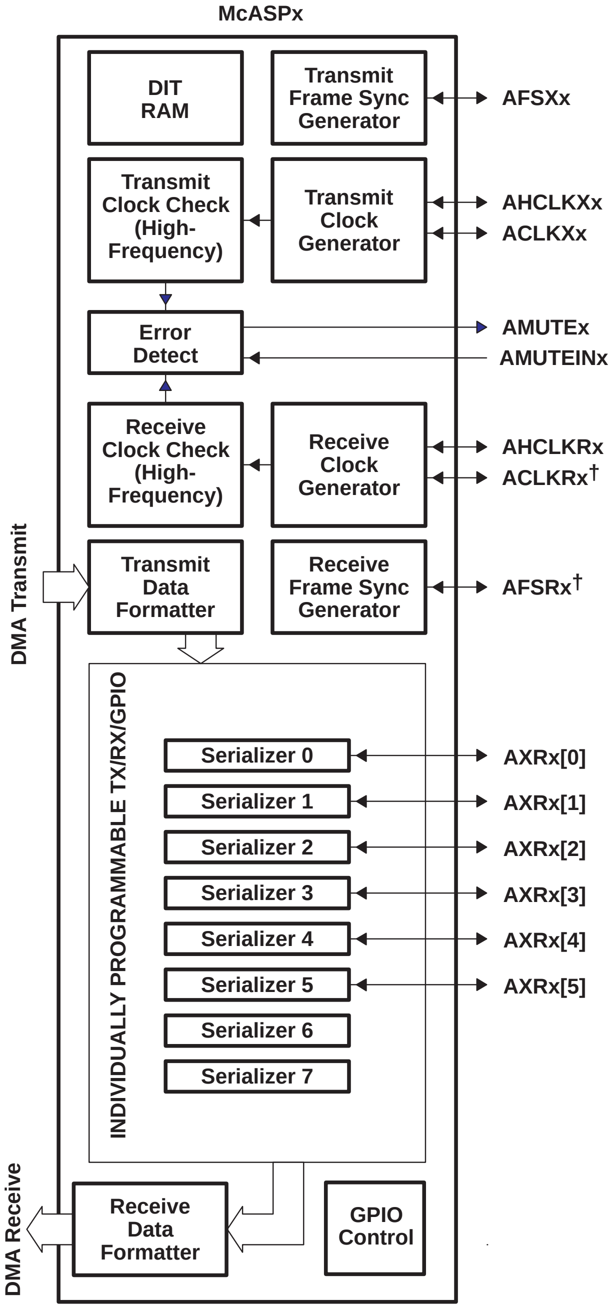
Both the transmit and receive sections of the McASP also support burst mode which is useful for non-audio data (for example, passing control information between two DSPs).

The McASP peripheral has additional capability for flexible clock generation, and error detection/handling, as well as error management.

For more detailed information on and the functionality of the McASP peripheral, see the *TMS320C6000 DSP Multichannel Audio Serial Port (McASP) Reference Guide* (literature number SPRU041).

4.3.1 McASP Block Diagram

Figure 4−2 illustrates the major blocks along with external signals of the TMS320C6413/C6410 McASP peripheral; and shows the 6 serial data [AXRx] pins. The McASP also includes full general-purpose I/O (GPIO) control, so any pins not needed for serial transfers can be used for general-purpose I/O.



† On the C6413/C6410 device, the McASP1 peripheral has some additional pins muxed with AFSR1 and with ACLKR1 pins (i.e., AFSR1[1], AFSR1[2], AFSR1[3] and ACLKR1[1]. ACLKR1[2], ACLKR1[3], respectively).

‡ On the C6413/C6410 device, the McASP0 peripheral is standalone, not muxed and the McASP1 peripheral is muxed with the HPI peripheral. For more detailed information on multiplexed pins, see the Device Configurations section of this data sheet.

Figure 4−2. McASP0 and McASP1‡ Configuration

4.4 I2C

The TMS320C6413/C6410 device includes two I2C peripheral modules (I2C0 and I2C1). NOTE: when using the I2C modules (any mode), ensure there are external pullup resistors on the SDAx and SCLx pins.

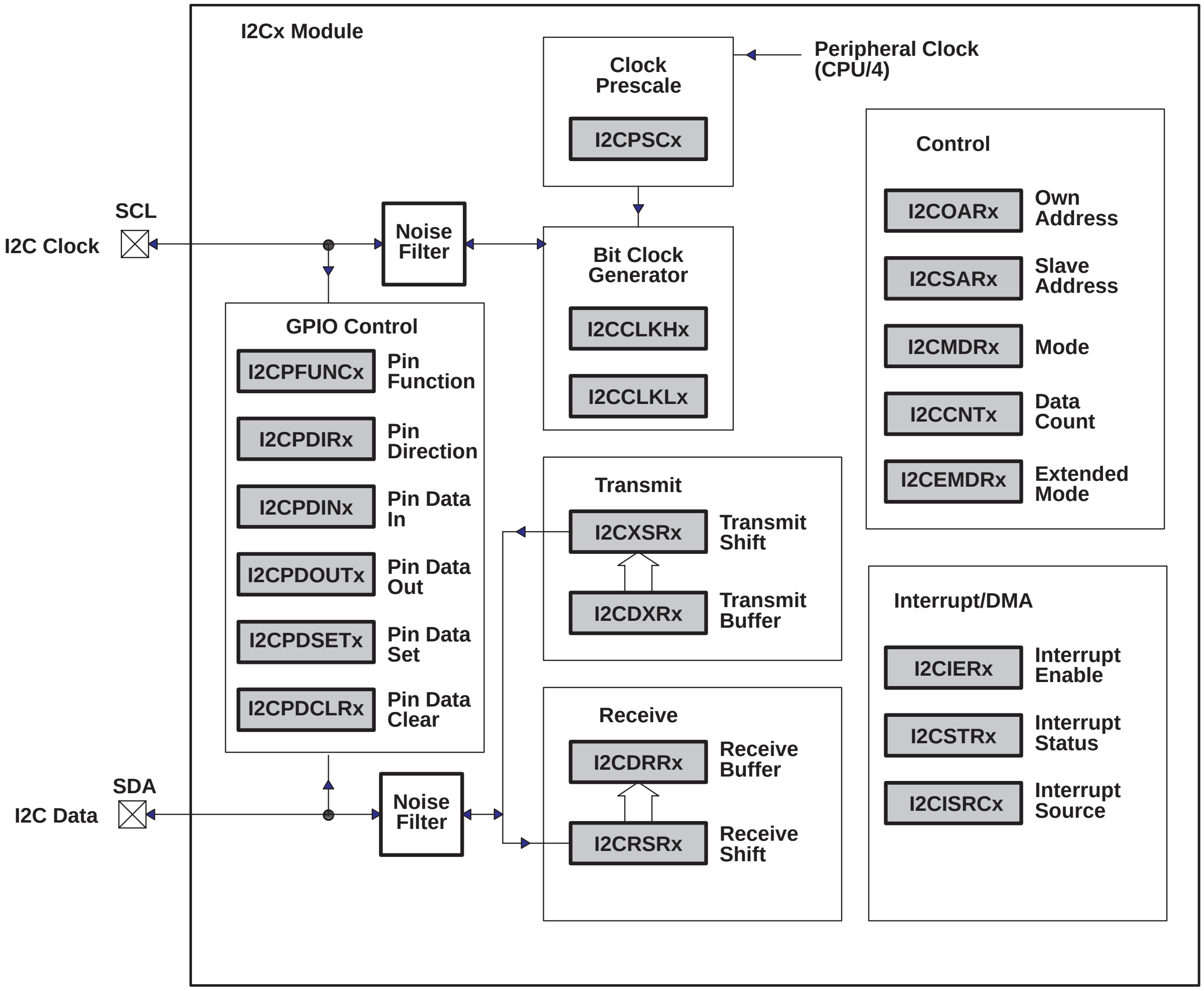
One of the I2C modules on the TMS320C6413/C6410 may be used by the DSP to control local peripherals ICs (DACs, ADCs, etc.) while the other module may be used to communicate with other controllers in a system or to implement a user interface.

The I2Cx port supports:

- Compatible with Philips I2C Specification Revision 2.1 (January 2000)
- Fast Mode up to 400 Kbps (no fail-safe I/O buffers)
- Noise Filter to remove noise 50 ns or less
- 7- and 10-Bit Device Addressing Modes
- Multi-Master (Transmit/Receive) and Slave (Transmit/Receive) Functionality
- Events: DMA, Interrupt, or Polling
- Slew-Rate Limited Open-Drain Output Buffers
- General-purpose input and output (GPIO) functionality for I2C pins

For more detailed information on C6413/6410 I2C additional features, such as GPIO capability, etc., see the TMS320C6000 DSP Inter−Integrated Circuit (I2C) Module Reference Guide (literature number SPRU175) and the TMS320C6410/C6413/C6418 DSP Inter−Integrated Circuit (I2C) Module Reference Guide (literature number SPRZ221) addendum.

Figure 4−3 is a block diagram of the I2C0 and I2C1 modules.



NOTE A: Shading denotes control/status registers.

Figure 4−3. I2Cx Module Block Diagram

4.5 General-Purpose Input/Output (GPIO)

On the C6413/C6410 device the GPIO peripheral pins GP0[15:9] are muxed with the HPI peripheral pins HD[15:9], respectively. By default, the HPI peripheral pin functions are enabled [TOUT0/HPI_EN pin internall pulled low]. For more detailed information on device/peripheral configuration and the C6413/C6410 device pin muxing, see the Device Configurations section of this data sheet.

To use the GP0[15:0] software-configurable GPIO pins, the GPxEN bits in the GP Enable (GPEN) Register and the GPxDIR bits in the GP Direction (GPDIR) Register must be properly configured.

GPxEN = 1 GP[x] pin is enabled

GPxDIR = 0 GP[x] pin is an input

GPxDIR = 1 GP[x] pin is an output

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

Figure 4−4 shows the GPIO enable bits in the GPEN register for the C6413/C6410 device. To use any of the GPx pins as general-purpose input/output functions, the corresponding GPxEN bit must be set to “1” (enabled). Default values are device-specific, so refer to Figure 4−4 for the C6413/C6410 default configuration.

31–24	23–16
Reserved	
R-0	

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

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

Figure 4−4. GPIO Enable Register (GPEN) [Hex Address: 01B0 0000]

Figure 4−5 shows the GPIO direction bits in the GPDIR register. This register determines if a given GPIO pin is an input or an output providing the corresponding GPxEN bit is enabled (set to “1”) in the GPEN register. By default, all the GPIO pins are configured as input pins.

31–24	23–16
Reserved	
R-0	

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

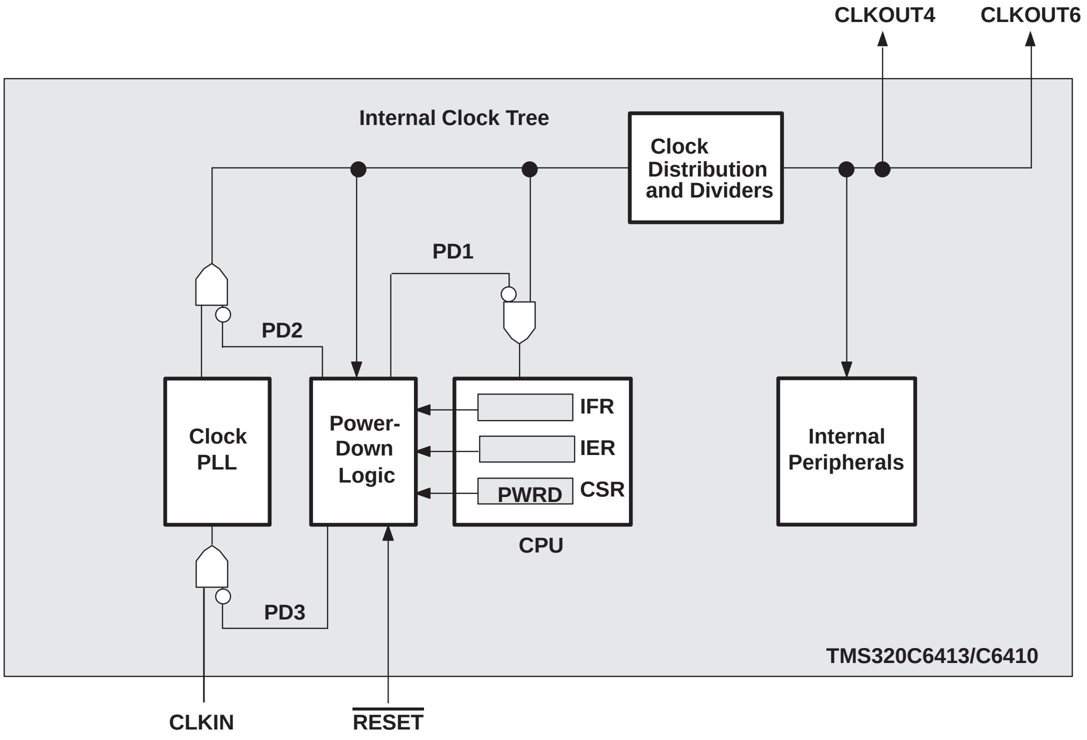
Legend: R/W = Readable/Writeable; -*n* = value after reset, -x = undefined value after reset

Figure 4−5. GPIO Direction Register (GPDIR) [Hex Address: 01B0 0004]

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

4.6 Power-Down Modes Logic

Figure 4−6 shows the power-down mode logic on the C6413/C6410.



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

Figure 4−6. Power-Down Mode Logic†

Note: to further save power, the PERCFG register can be used to disable unused peripherals. For more detailed information on disabling peripherals using the PERCFG register, see the *Device Configurations* section of this data sheet.

4.6.1 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 4−7 and described in Table 4−4. 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).

31	16

15	14	13	12	11	10	9	8
Reserved	**Enable or Non-Enabled Interrupt Wake**	**Enabled Interrupt Wake**	**PD3**	**PD2**	**PD1**		
R/W-0	**R/W-0**	**R/W-0**	**R/W-0**	**R/W-0**	**R/W-0**		

7	0

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 4−7. PWRD Field of the CSR Register

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

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

PD2 and PD3 modes can only be aborted by device reset. Table 4−4 summarizes all the power-down modes.

Table 4−4. Characteristics of the Power-Down Modes

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

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

4.6.2 C64x Power-Down Mode with an Emulator

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

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

4.7 Power-Supply Sequencing

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

4.7.1 Power-Supply Design Considerations

A dual-power supply with simultaneous sequencing can be used to eliminate the delay between core and I/O power up. A Schottky diode can also be used to tie the core rail to the I/O rail (see Figure 4−8).

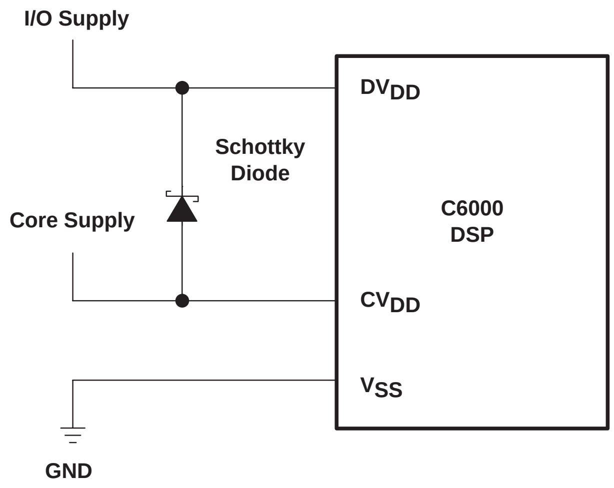


Figure 4−8. Schottky Diode Diagram

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

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.

4.8 Power-Supply Decoupling

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

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

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

4.9 Peripheral Power-Down Operation

The C6413/C6410 device can be powered down in two ways:

- Power-down due to software configuration − relates to the default state of the peripheral configuration bits in the PERCFG register.
- Power-down during run-time via software configuration

On the C6413/C6410 device, the HPI, McASP1, and GP0 peripherals pin muxing is controlled (selected) at the pin level during chip reset (e.g., $\overline{\text{HPI\_EN}}$ and HD5 pins). If McASP1 pin muxing is selected, then the MCASP1EN bit in the peripheral configuration register (PERCFG.8) must be configured properly to enable the McASP1 peripheral.

The McASP1, McASP0, I2C1, and I2C0 peripheral functions are selected via the peripheral configuration (PERCFG) register bits.

For more detailed information on the peripheral configuration pins and the PERCFG register bits, see the *Device Configurations* section of this document.

4.10 IEEE 1149.1 JTAG Compatibility Statement

The TMS320C6413/C6410 DSP requires that both $\overline{\text{TRST}}$ and $\overline{\text{RESET}}$ be asserted upon power up to be properly initialized. While $\overline{\text{RESET}}$ initializes the DSP core, $\overline{\text{TRST}}$ initializes the DSP's emulation logic. Both resets are required for proper operation.

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

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

The TMS320C6413/6410 DSP includes an internal pulldown (IPD) on the $\overline{\text{TRST}}$ pin to ensure that $\overline{\text{TRST}}$ will always be asserted upon power up and the DSP's internal emulation logic will always be properly initialized when this pin is not routed out. JTAG controllers from Texas Instruments actively drive $\overline{\text{TRST}}$ high. However, some third-party JTAG controllers may not drive $\overline{\text{TRST}}$ high but expect the use of an external pullup resistor on $\overline{\text{TRST}}$. When using this type of JTAG controller, assert $\overline{\text{TRST}}$ to initialize the DSP after powerup and externally drive $\overline{\text{TRST}}$ high before attempting any emulation or boundary scan operations.

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

Note: The DESIGN_WARNING section of the TMS320C6413/C6410 BSDL file contains information and constraints regarding proper device operation while in Boundary Scan Mode.

For more detailed information on the C6413/C6410 JTAG emulation, see the *TMS320C6000 DSP Designing for JTAG Emulation Reference Guide* (literature number SPRU641).

4.11 EMIF Device Speed

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

- 1 chip-enable (CE) space (maximum of 2 chips) of SDRAM connected to EMIF
- up to 1 CE space of buffers connected to EMIF
- EMIF trace lengths between 1 and 3 inches
- 143-MHz SDRAM for 100-MHz operation

Other configurations may be possible, but timing analysis must be done to verify all AC timings are met. Verification of AC timings is mandatory when using configurations other than those specified above. TI recommends utilizing I/O buffer information specification (IBIS) to analyze all AC timings.

To properly use IBIS models to attain accurate timing analysis for a given system, see the *Using IBIS Models for Timing Analysis* application report (literature number SPRA839).

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

For more detailed information on the C6413/C6410 EMIF peripheral, see the *TMS320C6000 DSP External Memory Interface (EMIF) Reference Guide* (literature number SPRU266).

4.12 Bootmode

The C6413/C6410 device resets using the active-low signal $\overline{\text{RESET}}$. While $\overline{\text{RESET}}$ is low, the device is held in reset and is initialized to the prescribed reset state. Refer to reset timing for reset timing characteristics and states of device pins during reset. The release of $\overline{\text{RESET}}$ starts the processor running with the prescribed device configuration and boot mode.

The C6413/C6410 has three types of boot modes:

- Host boot

  If host boot is selected, upon release of $\overline{\text{RESET}}$, the CPU is internally “stalled” while the remainder of the device is released. During this period, an external host can initialize the CPU’s memory space as necessary through the host interface, including internal configuration registers, such as those that control the EMIF or other peripherals. For the C6413/C6410 device, the HPI peripheral is used for host boot providing the TOUT0/$\overline{\text{HPI\_EN}}$ pin is low, enabling the HPI peripheral [default]. Once the host is finished with all necessary initialization, it must set the DSPINT bit in the HPIC register to complete the boot process. This transition causes the boot configuration logic to bring the CPU out of the “stalled” state. The CPU then begins execution from address 0. The DSPINT condition is not latched by the CPU, because it occurs while the CPU is still internally “stalled”. Also, DSPINT brings the CPU out of the “stalled” state only if the host boot process is selected. All memory may be written to and read by the host. This allows for the host to verify what it sends to the DSP if required. After the CPU is out of the “stalled” state, the CPU needs to clear the DSPINT, otherwise, no more DSPINTs can be received.

- EMIF boot (using default ROM timings)

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

- No boot

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

4.13 Reset

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

5 Device Electrical Specifications

5.1 Absolute Maximum Ratings Over Operating Case Temperature Range (Unless Otherwise Noted)[†]

Supply voltage ranges:	CV_{DD} (see Note 1)	− 0.3 V to 1.8 V
	DV_{DD} (see Note 1)	−0.3 V to 4 V
Input voltage range:	V_I	−0.3 V to 4 V
Output voltage range:	V_O	−0.3 V to 4 V
Operating case temperature range, T_C:	(default)	0°C to 90°C
	(A version) [GTSA and ZTSA]	−40°C to 105°C
Storage temperature range, T_{stg}		−65°C to 150°C
Package Temperature Cycling:	Temperature Range	−40°C to 125°C
	Number of Cycles (GTS, GTSA)	1000
	Number of Cycles (ZTS, ZTSA)	500

[†] 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}.

5.2 Recommended Operating Conditions

			MIN	NOM	MAX	UNIT
CV_{DD}	Supply voltage, Core (-400, -500 device)[‡]		1.14	1.2	1.26	V
DV_{DD}	Supply voltage, I/O		3.14	3.3	3.46	V
V_{SS}	Supply ground		0	0	0	V
V_{IH}	High-level input voltage		2			V
V_{IL}	Low-level input voltage				0.8	V
V_{OS}	Maximum voltage during overshoot				4.3[§]	V
V_{US}	Minimum voltage during undershoot		−1.0[§]			V
T_C	Operating case temperature	Commercial temperature devices (GTS and ZTS)	0		90	°C
		Extended temperature devices (GTSA and ZTSA)	−40		105	°C

[‡] Future variants of the C64x DSPs may operate at voltages ranging from 0.9 V to 1.4 V to provide a range of system power/performance options. TI highly recommends that users design-in a supply that can handle multiple voltages within this range (i.e., 1.2 V, 1.25 V, 1.3 V, 1.35 V, 1.4 V with ± 3% tolerances) by implementing simple board changes such as reference resistor values or input pin configuration modifications. Examples of such supplies include the PT4660, PT5500, PT5520, PT6440, and PT6930 series from Power Trends, a subsidiary of Texas Instruments. Not incorporating a flexible supply may limit the system’s ability to easily adapt to future versions of C64x devices.

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

5.3 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	DV_{DD} = MIN, I_{OH} = MAX	2.4			V
V_{OL}	Low-level output voltage	DV_{DD} = MIN, I_{OL} = MAX			0.4	V
I_I	Input current	$V_I = V_{SS}$ to DV_{DD} no opposing internal resistor			±10	uA
		$V_I = V_{SS}$ to DV_{DD} opposing internal pullup resistor[‡]	50	100	150	uA
		$V_I = V_{SS}$ to DV_{DD} opposing internal pulldown resistor[‡]	−150	−100	−50	uA
I_{OH}	High-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx			−16	mA
		Timer, TDO, GPIO, McBSP, HPI			−8	mA
I_{OL}	Low-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx			16	mA
		Timer, TDO, GPIO, McBSP, HPI			8	mA
		SCL1, SDA1, SCL0, and SDA0			3	mA
I_{OZ}	Off-state output current	$V_O = DV_{DD}$ or 0 V			±10	uA
I_{CDD}	Core supply current[§]	CV_{DD} = 1.2 V, CPU clock = 500 MHz		568		mA
		CV_{DD} = 1.2 V, CPU clock = 400 MHz		465		mA
I_{DDD}	I/O supply current[§]	DV_{DD} = 3.3 V, CPU clock = 500 MHz		140		mA
		DV_{DD} = 3.3 V, CPU clock = 400 MHz		132		mA
C_i	Input capacitance				10	pF
C_o	Output capacitance				10	pF

[†] For test conditions shown as MIN, MAX, or NOM, use the appropriate value specified in the recommended operating conditions table.

[‡] Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.

[§] Measured with average activity (50% high/50% low power) at 25°C case temperature and 100-MHz EMIF for -500 and -400 speeds. This model represents a device performing high-DSP-activity operations 50% of the time, and the remainder performing low-DSP-activity operations. The high/low-DSP-activity models are defined as follows:

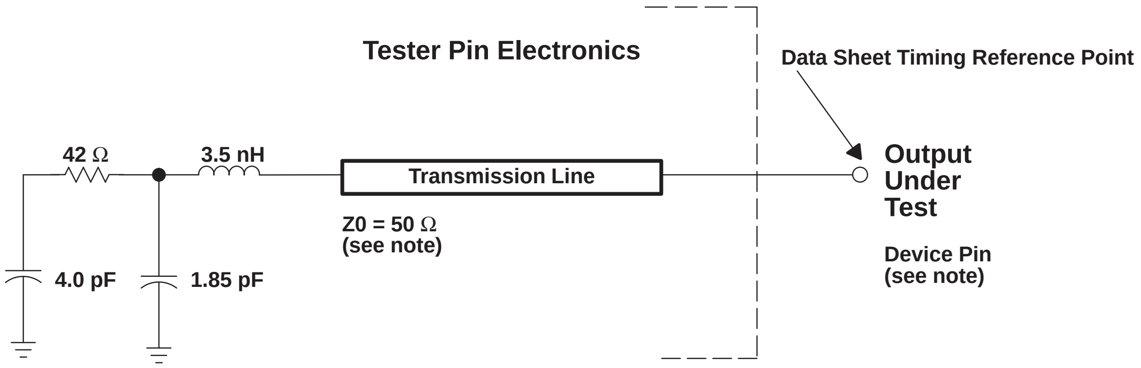
High-DSP-Activity Model:
  CPU: 8 instructions/cycle with 2 LDDW instructions [L1 Data Memory: 128 bits/cycle via LDDW instructions;
    L1 Program Memory: 256 bits/cycle; L2/EMIF EDMA: 50% writes, 50% reads to/from SDRAM (50% bit-switching)]
  McBSP: 2 channels at E1 rate
  Timers: 2 timers at maximum rate
Low-DSP-Activity Model:
  CPU: 2 instructions/cycle with 1 LDH instruction [L1 Data Memory: 16 bits/cycle; L1 Program Memory: 256 bits per 4 cycles;
    L2/EMIF EDMA: None]
  McBSP: 2 channels at E1 rate
  Timers: 2 timers at maximum rate

The actual current draw is highly application-dependent. For more details on core and I/O activity, refer to the *TMS320C6410/13 Power Consumption Summary* application report (literature number SPRAA59).

5.4 Recommended Clock and Control Signal Transition Behavior

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

6 Parameter Information



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

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

Figure 6−1. Test Load Circuit for AC Timing Measurements

The load capacitance value stated is only for characterization and measurement of AC timing signals. This load capacitance value does not indicate the maximum load the device is capable of driving.

6.1 Signal Transition Levels

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

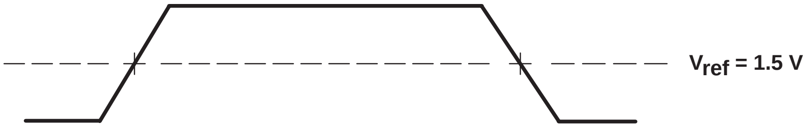


Figure 6−2. Input and Output Voltage Reference Levels for AC Timing Measurements

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.

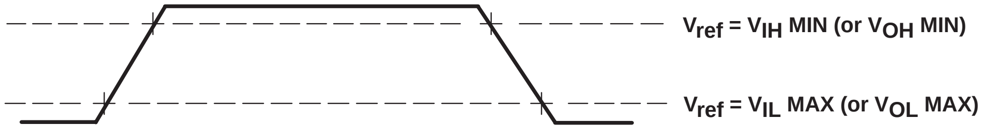


Figure 6−3. Rise and Fall Transition Time Voltage Reference Levels

6.2 Signal Transition Rates

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

PARAMETER MEASUREMENT INFORMATION (CONTINUED)

6.3 AC transient rise/fall time specifications

Figure 6−4 and Figure 6−5 show the AC transient specifications for Rise and Fall Time. For device-specific information on these values, refer to the Recommended Operating Conditions section of this Data Sheet.

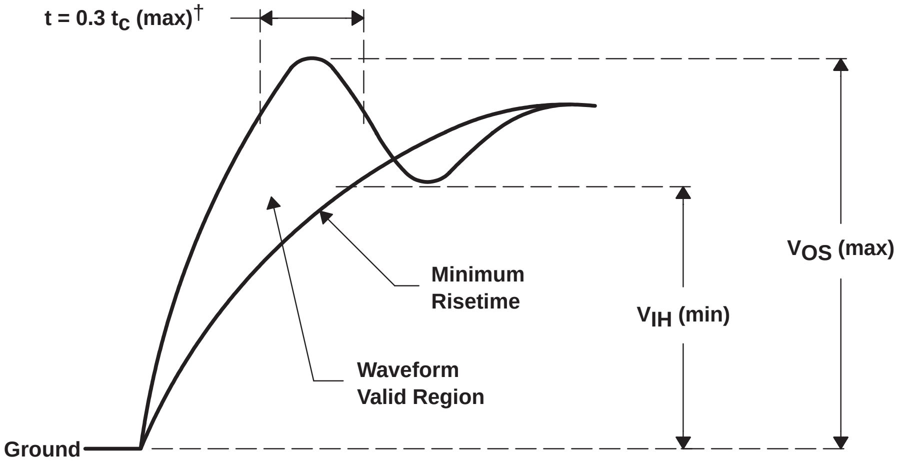
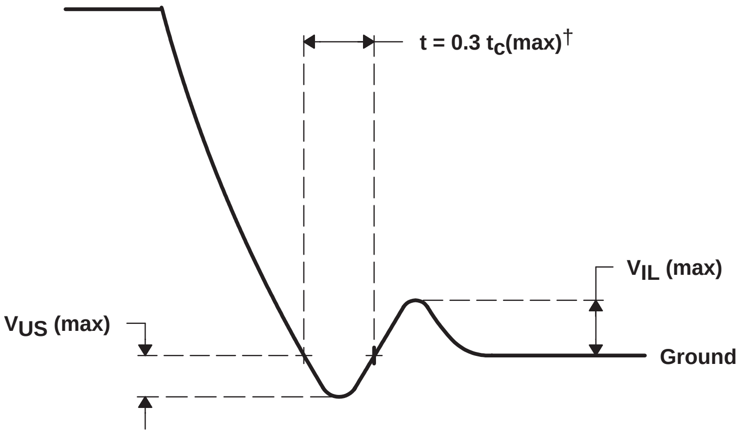


Figure 6−4. AC Transient Specification Rise Time

† t_c = the peripheral cycle time.



† t_c = the peripheral cycle time.

Figure 6−5. AC Transient Specification Fall Time

6.4 Timing Parameters and Board Routing Analysis

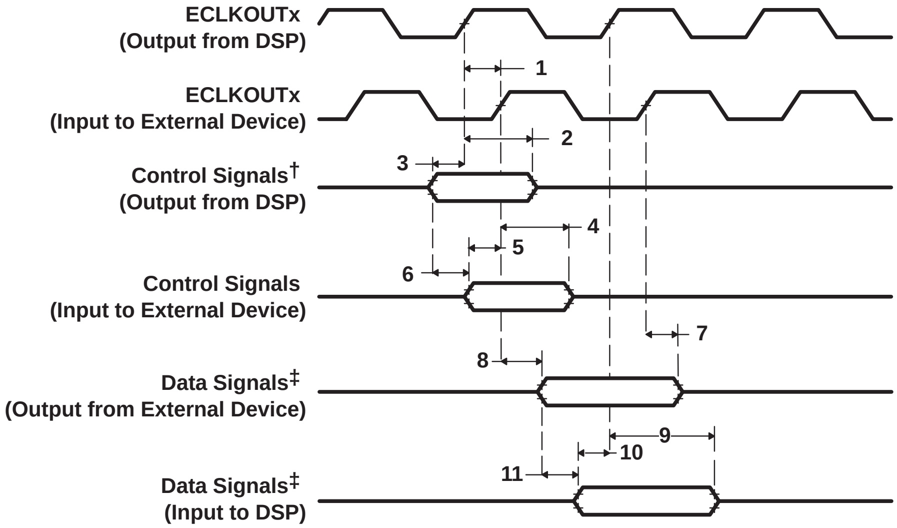
The timing parameter values specified in this data sheet do *not* include delays by board routings. As a good board design practice, such delays must *always* be taken into account. Timing values may be adjusted by increasing/decreasing such delays. TI recommends utilizing the available I/O buffer information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

For inputs, timing is most impacted by the round-trip propagation delay from the DSP to the external device and from the external device to the DSP. This round-trip delay tends to negatively impact the input setup time margin, but also tends to improve the input hold time margins (see Table 6−1 and Figure 6−6).

Figure 6−6 represents a general transfer between the DSP and an external device. The figure also represents board route delays and how they are perceived by the DSP and the external device.

Table 6−1. Board-Level Timing Example (see Figure 6−6)

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



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

Figure 6−6. Board-Level Input/Output Timings

7 Peripheral Electrical Specifications

7.1 Input and Output Clocks

Table 7−1. Timing Requirements for External Crystal Oscillator Input (OSCIN and OSCOUT)

NO.			−400 −500		UNIT
			MIN	MAX	
1	f_{OSC}	Input oscillator frequency	12	30	MHz

[†] The PLL multiplier factors (x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x24) further limit the MIN and MAX values for CLKIN and OSCIN. For more details on these limitations, see Table 4−1 and Table 4−2 of the *Clock PLL and Oscillator* section of this data sheet.

Table 7−2. Timing Requirements for CLKIN[†‡§] (see Figure 7−1)

NO.			−400 −500				UNIT
			PLL MULT MODE		x1 (BYPASS)		
			MIN	MAX	MIN	MAX	
1	$t_{c(CLKIN)}$	Cycle time, CLKIN	10[†]	83.3	10[†]	83.3	ns
2	$t_{w(CLKINH)}$	Pulse duration, CLKIN high	0.45C		0.45C		ns
3	$t_{w(CLKINL)}$	Pulse duration, CLKIN low	0.45C		0.45C		ns
4	$t_{t(CLKIN)}$	Transition time, CLKIN		5		1	ns
5	$t_{J(CLKIN)}$	Period jitter, CLKIN		0.02C		0.02C	ns

[†] The PLL multiplier factors (x5, x6, x7, x8, x9, x10, x11, x12, x16, x18, x19, x20, x21, x24) further limit the MIN and MAX values for CLKIN and OSCIN. For more details on these limitations, see Table 4−1 and Table 4−2 of the *Clock PLL and Oscillator* section of this data sheet.

[‡] The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.

[§] C = CLKIN cycle time in ns. For example, when CLKIN frequency is 50 MHz, use C = 20 ns.

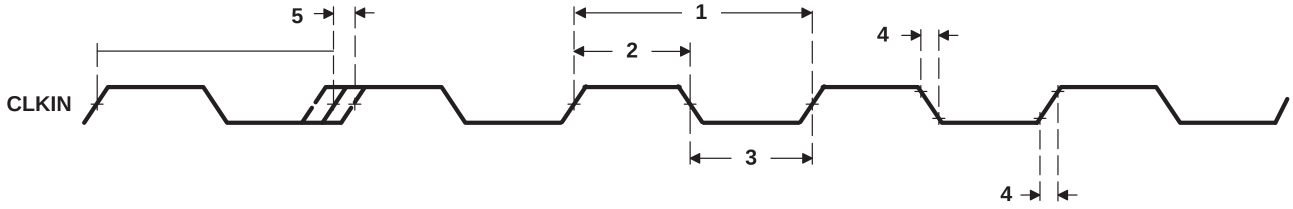


Figure 7−1. CLKIN Timing

Table 7−3. Switching Characteristics Over Recommended Operating Conditions for CLKOUT4[†‡§] (see Figure 7−2)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{c(CKO4)}$	Cycle time, CLKOUT4	4P − 0.7	4P + 0.7	ns
2	$t_{w(CKO4H)}$	Pulse duration, CLKOUT4 high	2P − 0.7	2P + 0.7	ns
3	$t_{w(CKO4L)}$	Pulse duration, CLKOUT4 low	2P − 0.7	2P + 0.7	ns
4	$t_{t(CKO4)}$	Transition time, CLKOUT4		1	ns

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

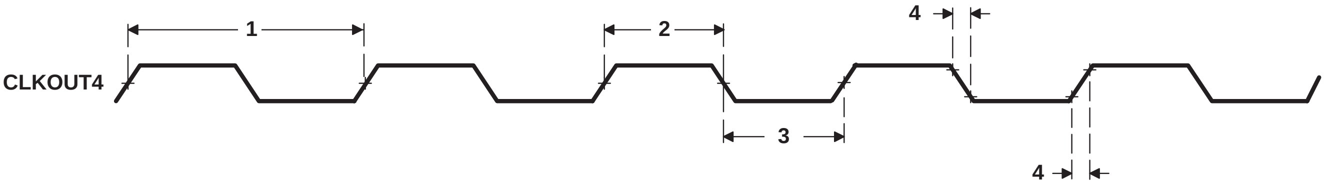


Figure 7−2. CLKOUT4 Timing

Table 7−4. Switching Characteristics Over Recommended Operating Conditions for CLKOUT6[†‡§] (see Figure 7−3)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{c(CKO6)}$	Cycle time, CLKOUT6	6P − 0.7	6P + 0.7	ns
2	$t_{w(CKO6H)}$	Pulse duration, CLKOUT6 high	3P − 0.7	3P + 0.7	ns
3	$t_{w(CKO6L)}$	Pulse duration, CLKOUT6 low	3P − 0.7	3P + 0.7	ns
4	$t_{t(CKO6)}$	Transition time, CLKOUT6		1	ns

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

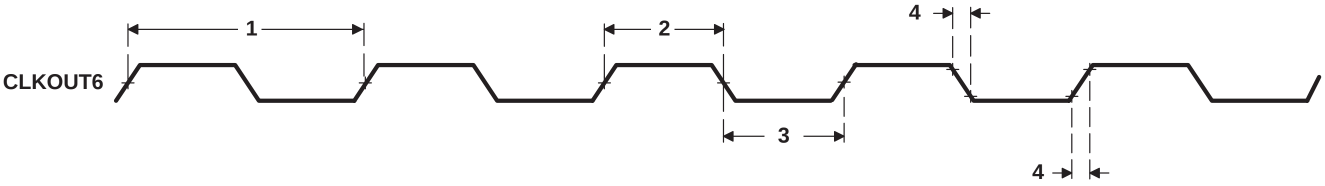


Figure 7−3. CLKOUT6 Timing

Table 7−5. Timing Requirements for AECLKIN for EMIFA[†‡§] (see Figure 7−4)

NO.			−400 −500		UNIT
			MIN	MAX	
1	$t_{c(EKI)}$	Cycle time, AECLKIN	6[¶]	16P	ns
2	$t_{w(EKIH)}$	Pulse duration, AECLKIN high	2.7		ns
3	$t_{w(EKIL)}$	Pulse duration, AECLKIN low	2.7		ns
4	$t_{t(EKI)}$	Transition time, AECLKIN		3	ns
5	$t_{J(EKI)}$	Period jitter, AECLKIN		0.02E	ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
[‡] The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
[§] E = the EMIF input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) period in ns for EMIFA.
[¶] Minimum AECLKIN cycle times *must* be met, even when AECLKIN is generated by an internal clock source. Minimum AECLKIN times are based on internal logic speed; the maximum useable speed of the EMIF may be lower due to AC timing requirements. 100-MHz operation is achievable if the requirements of the EMIF Device Speed section are met.

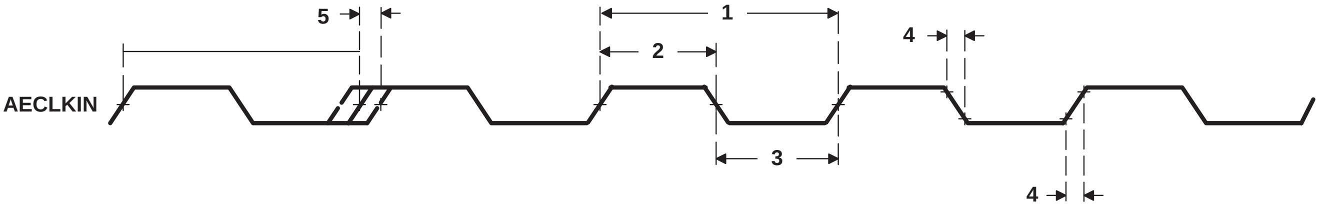


Figure 7−4. AECLKIN Timing for EMIFA

Table 7−6. Switching Characteristics Over Recommended Operating Conditions for AECLKOUT1 for the EMIFA Module[§#||] (see Figure 7−5)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{c(EKO1)}$	Cycle time, AECLKOUT1	E − 0.7	E + 0.7	ns
2	$t_{w(EKO1H)}$	Pulse duration, AECLKOUT1 high	EH − 0.7	EH + 0.7	ns
3	$t_{w(EKO1L)}$	Pulse duration, AECLKOUT1 low	EL − 0.7	EL + 0.7	ns
4	$t_{t(EKO1)}$	Transition time, AECLKOUT1		1	ns
5	$t_{d(EKIH\text{-}EKO1H)}$	Delay time, AECLKIN high to AECLKOUT1 high	1	8	ns
6	$t_{d(EKIL\text{-}EKO1L)}$	Delay time, AECLKIN low to AECLKOUT1 low	1	8	ns

[§] E = the EMIF input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) period in ns.
[#] The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
[||] EH is the high period of E (EMIF input clock period) in ns and EL is the low period of E (EMIF input clock period) in ns for EMIFA.

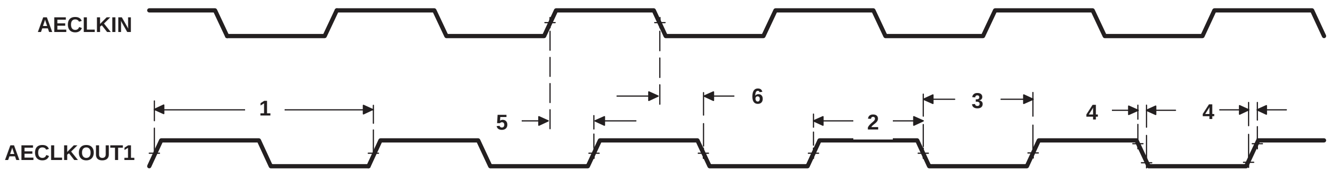


Figure 7−5. AECLKOUT1 Timing for the EMIFA Module

Table 7−7. Switching Characteristics Over Recommended Operating Conditions for AECLKOUT2 for the EMIFA Module†‡ (see Figure 7−6)

NO.	PARAMETER		−400 −500 MIN	MAX	UNIT
1	$t_{c(EKO2)}$	Cycle time, AECLKOUT2	NE − 0.7	NE + 0.7	ns
2	$t_{w(EKO2H)}$	Pulse duration, AECLKOUT2 high	0.5NE − 0.7	0.5NE + 0.7	ns
3	$t_{w(EKO2L)}$	Pulse duration, AECLKOUT2 low	0.5NE − 0.7	0.5NE + 0.7	ns
4	$t_{t(EKO2)}$	Transition time, AECLKOUT2		1	ns
5	$t_{d(EKIH-EKO2H)}$	Delay time, ECLKIN high to AECLKOUT2 high	1	8	ns
6	$t_{d(EKIL-EKO2L)}$	Delay time, ECLKIN low to AECLKOUT2 low	1	8	ns

† The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.
‡ E = the EMIF input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) period in ns for EMIFA.
N = the EMIF input clock divider; N = 1, 2, or 4.

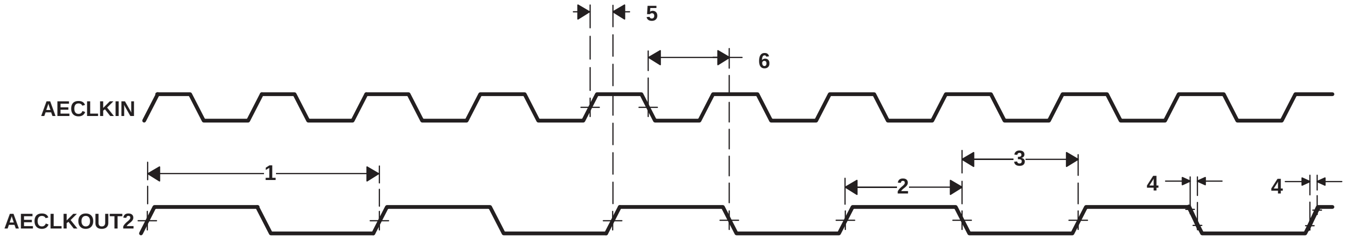


Figure 7−6. AECLKOUT2 Timing for the EMIFA Module

7.2 Asynchronous Memory Timing

Table 7−8. Timing Requirements for Asynchronous Memory Cycles for EMIFA Module[†][‡] (see Figure 7−7 and Figure 7−8)

NO.			−400 −500		UNIT
			MIN	MAX	
3	$t_{su(EDV\text{-}AREH)}$	Setup time, AEDx valid before $\overline{\text{AARE}}$ high	6.5		ns
4	$t_{h(AREH\text{-}EDV)}$	Hold time, AEDx valid after $\overline{\text{AARE}}$ high	1		ns
6	$t_{su(ARDY\text{-}EKO1H)}$	Setup time, AARDY valid before AECLKOUTx high	3		ns
7	$t_{h(EKO1H\text{-}ARDY)}$	Hold time, AARDY valid after AECLKOUTx high	3		ns

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

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

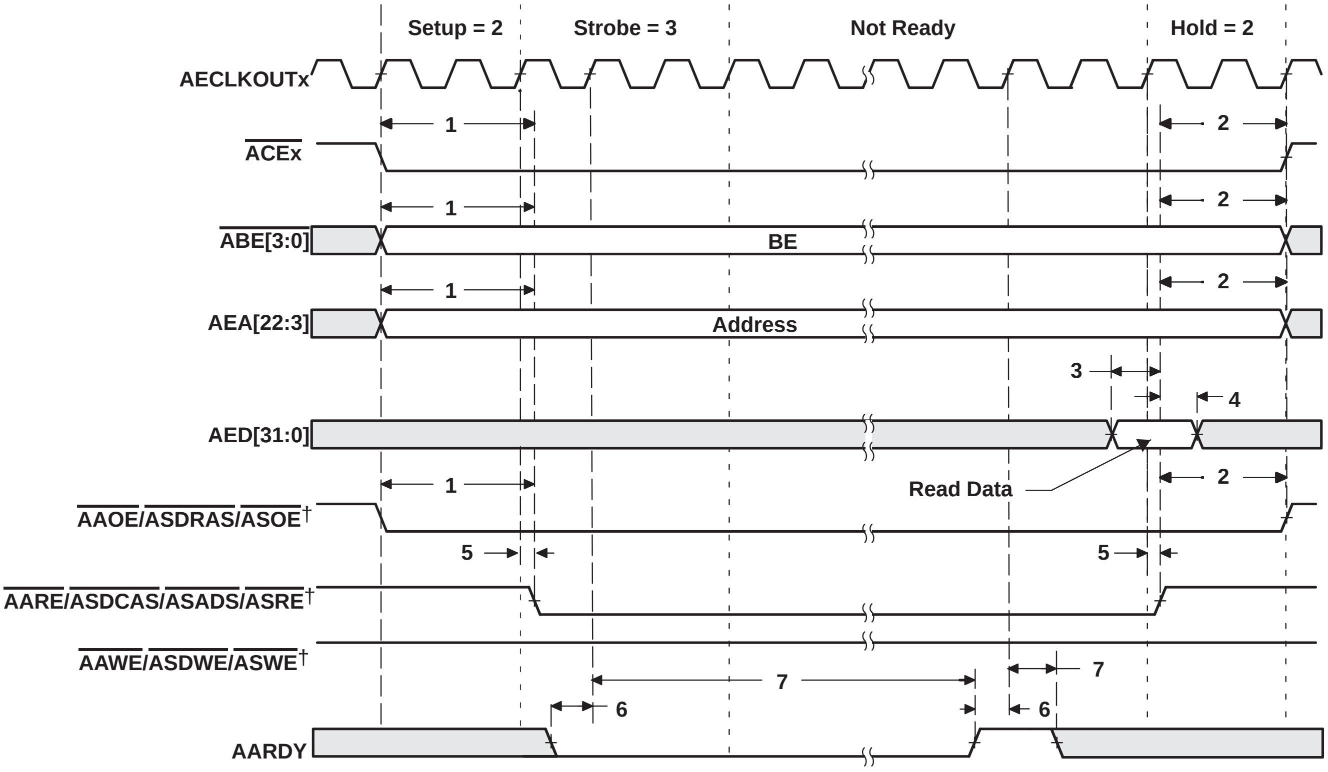
Table 7−9. Switching Characteristics Over Recommended Operating Conditions for Asynchronous Memory Cycles for EMIFA Module[‡][§][¶] (see Figure 7−7 and Figure 7−8)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{osu(SELV\text{-}AREL)}$	Output setup time, select signals valid to $\overline{\text{AARE}}$ low	RS * E − 1.5		ns
2	$t_{oh(AREH\text{-}SELIV)}$	Output hold time, $\overline{\text{AARE}}$ high to select signals invalid	RH * E − 1.9		ns
5	$t_{d(EKO1H\text{-}AREV)}$	Delay time, AECLKOUTx high to $\overline{\text{AARE}}$ valid	1	7	ns
8	$t_{osu(SELV\text{-}AWEL)}$	Output setup time, select signals valid to $\overline{\text{AAWE}}$ low	WS * E − 1.7		ns
9	$t_{oh(AWEH\text{-}SELIV)}$	Output hold time, $\overline{\text{AAWE}}$ high to select signals invalid	WH * E − 1.8		ns
10	$t_{d(EKO1H\text{-}AWEV)}$	Delay time, AECLKOUTx high to $\overline{\text{AAWE}}$ valid	1.3	7.1	ns

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

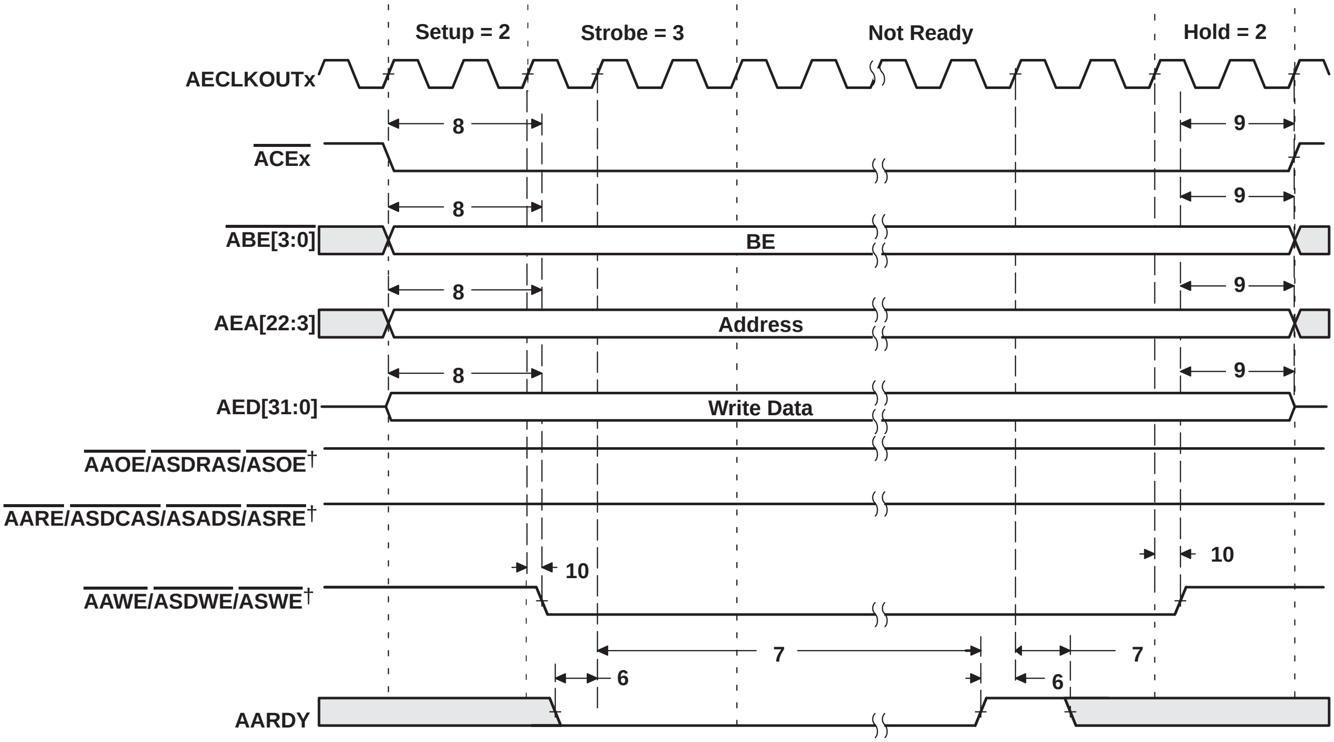
[§] E = ECLKOUT1 period in ns for EMIFA

[¶] Select signals for EMIFA include: $\overline{\text{ACEx}}$, $\overline{\text{ABE[3.:0]}}$, AEA[22:3], $\overline{\text{AAOE}}$; and for EMIFA writes, include AED[31:0].



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

Figure 7−7. Asynchronous Memory Read Timing for EMIFA



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

Figure 7−8. Asynchronous Memory Write Timing for EMIFA

7.3 Programmable Synchronous Interface Timing

Table 7−10. Timing Requirements for Programmable Synchronous Interface Cycles for EMIFA Module (see Figure 7−9)

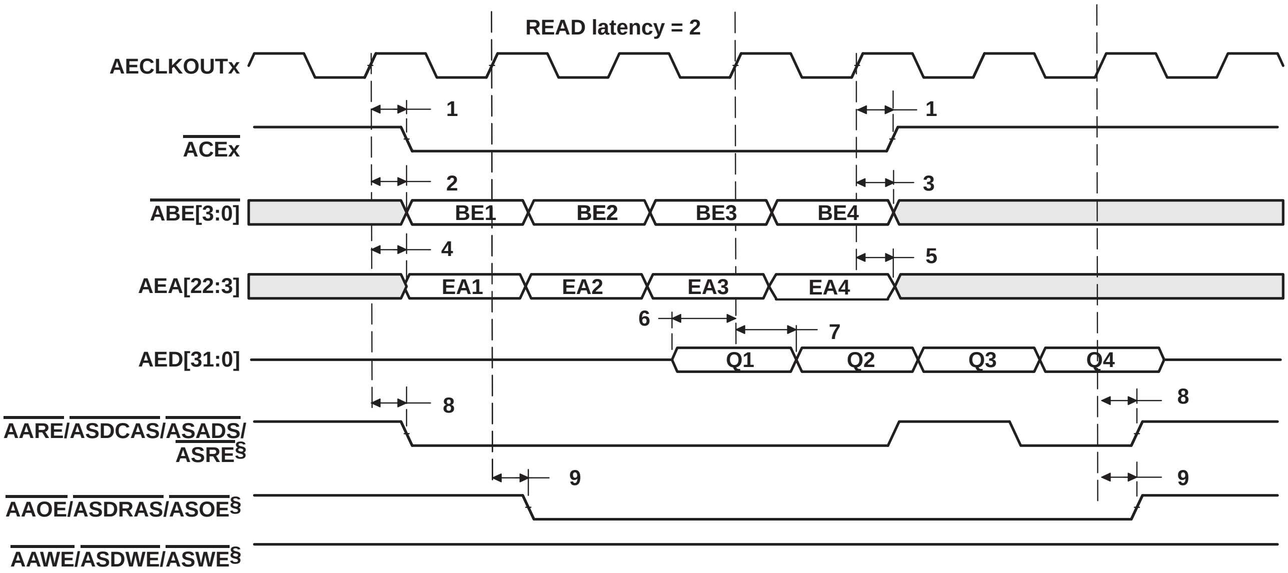
NO.			−400 −500		UNIT
			MIN	MAX	
6	$t_{su(EDV\text{-}EKOxH)}$	Setup time, read AEDx valid before AECLKOUTx high	3.1		ns
7	$t_{h(EKOxH\text{-}EDV)}$	Hold time, read AEDx valid after AECLKOUTx high	1.5		ns

Table 7−11. Switching Characteristics Over Recommended Operating Conditions for Programmable Synchronous Interface Cycles for EMIFA Module† (see Figure 7−9−Figure 7−11)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{d(EKOxH\text{-}CEV)}$	Delay time, AECLKOUTx high to ACEx valid	1.3	6.4	ns
2	$t_{d(EKOxH\text{-}BEV)}$	Delay time, AECLKOUTx high to ABEx valid		6.4	ns
3	$t_{d(EKOxH\text{-}BEIV)}$	Delay time, AECLKOUTx high to ABEx invalid	1.3		ns
4	$t_{d(EKOxH\text{-}EAV)}$	Delay time, AECLKOUTx high to AEAx valid		6.4	ns
5	$t_{d(EKOxH\text{-}EAIV)}$	Delay time, AECLKOUTx high to AEAx invalid	1.3		ns
8	$t_{d(EKOxH\text{-}ADSV)}$	Delay time, AECLKOUTx high to ASADS/ASRE valid	1.3	6.4	ns
9	$t_{d(EKOxH\text{-}OEV)}$	Delay time, AECLKOUTx high to, ASOE valid	1.3	6.4	ns
10	$t_{d(EKOxH\text{-}EDV)}$	Delay time, AECLKOUTx high to AEDx valid		6.4	ns
11	$t_{d(EKOxH\text{-}EDIV)}$	Delay time, AECLKOUTx high to AEDx invalid	1.3		ns
12	$t_{d(EKOxH\text{-}WEV)}$	Delay time, AECLKOUTx high to ASWE valid	1.3	6.4	ns

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

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- ACEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, ACEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, ACEx is active when ASOE is active (CEEXT = 1).
- Function of ASADS/ASRE (RENEN): For standard SBSRAM or ZBT SRAM interface, ASADS/ASRE acts as ASADS with deselect cycles (RENEN = 0). For FIFO interface, ASADS/ASRE acts as ASRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to AECLKOUT1 or AECLKOUT2



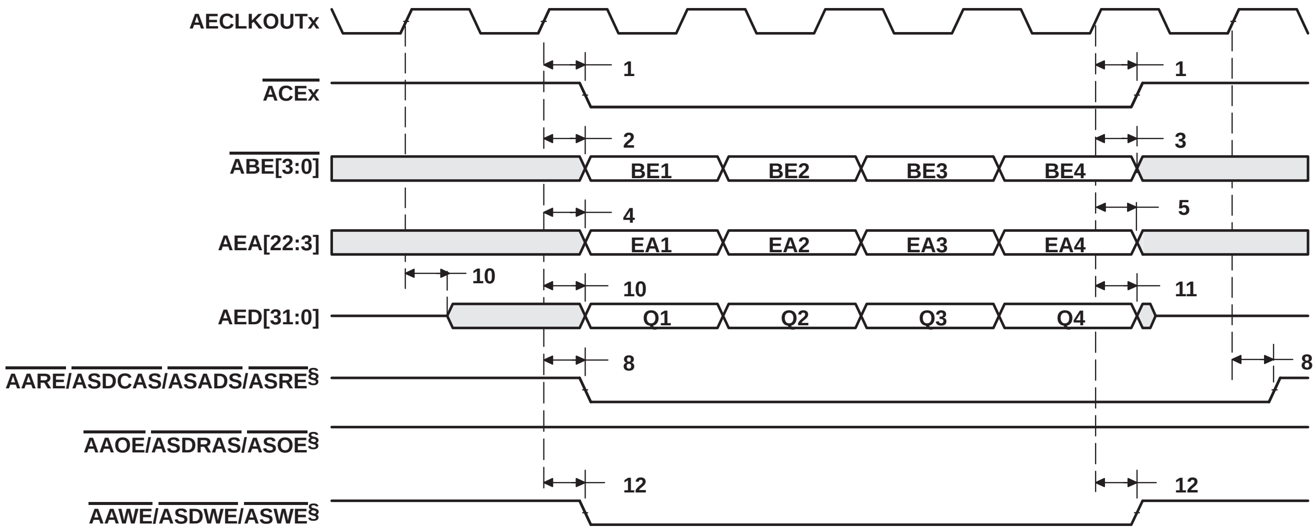
† The read latency and the length of CEx assertion are programmable via the SYNCRL and CEEXT fields, respectively, in the EMIFA CE Space Secondary Control register (CExSEC). In this figure, SYNCRL = 2 and CEEXT = 0.

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

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- ACEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, ACEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, ACEx is active when ASOE is active (CEEXT = 1).
- Function of ASADS/ASRE (RENEN): For standard SBSRAM or ZBT SRAM interface, ASADS/ASRE acts as ASADS with deselect cycles (RENEN = 0). For FIFO interface, ASADS/ASRE acts as ASRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to AECLKOUT1 or AECLKOUT2

§ AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE operate as ASADS/ASRE, ASOE, and ASWE, respectively, during programmable synchronous interface accesses.

Figure 7−9. Programmable Synchronous Interface Read Timing for EMIFA (With Read Latency = 2)†‡



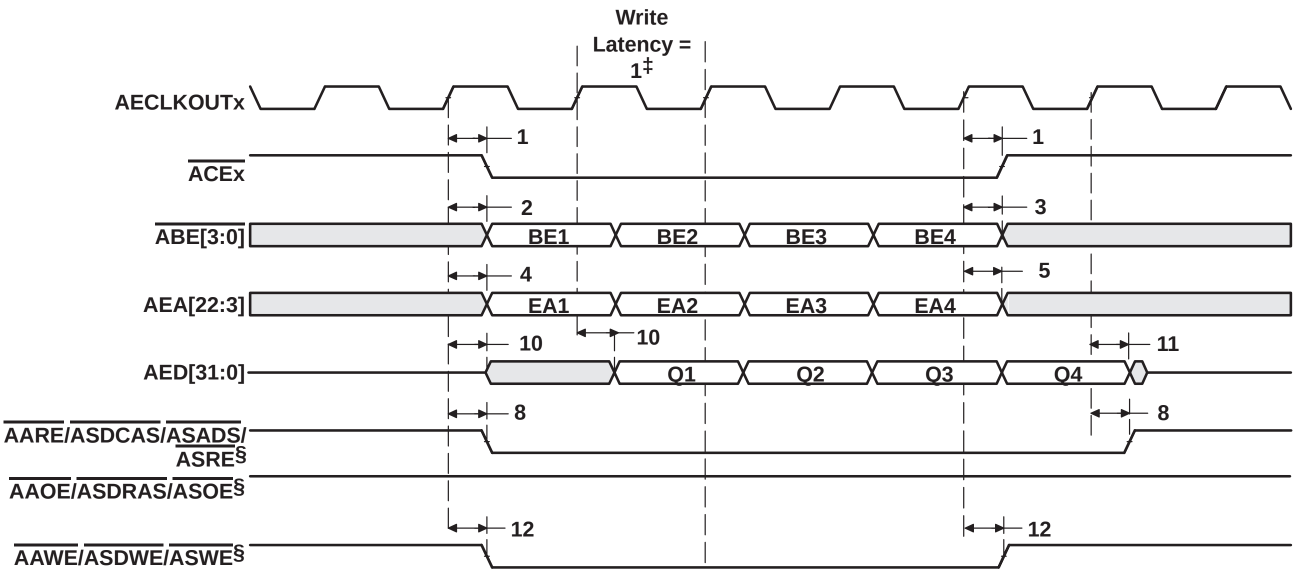
† The write latency and the length of ACEx assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIFA CE Space Secondary Control register (CExSEC). In this figure, SYNCWL = 0 and CEEXT = 0.

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

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- ACEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, ACEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, ACEx is active when ASOE is active (CEEXT = 1).
- Function of ASADS/ASRE (RENEN): For standard SBSRAM or ZBT SRAM interface, ASADS/ASRE acts as ASADS with deselect cycles (RENEN = 0). For FIFO interface, ASADS/ASRE acts as ASRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to AECLKOUT1 or AECLKOUT2

§ AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE operate as ASADS/ASRE, ASOE, and ASWE, respectively, during programmable synchronous interface accesses.

Figure 7−10. Programmable Synchronous Interface Write Timing for EMIFA (With Write Latency = 0)†‡§



† The write latency and the length of ACEx assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIFA CE Space Secondary Control register (CExSEC). In this figure, SYNCWL = 1 and CEEXT = 0.

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

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

§ AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE operate as ASADS/ASRE, ASOE, and ASWE, respectively, during programmable synchronous interface accesses.

Figure 7−11. Programmable Synchronous Interface Write Timing for EMIFA (With Write Latency = 1)†‡

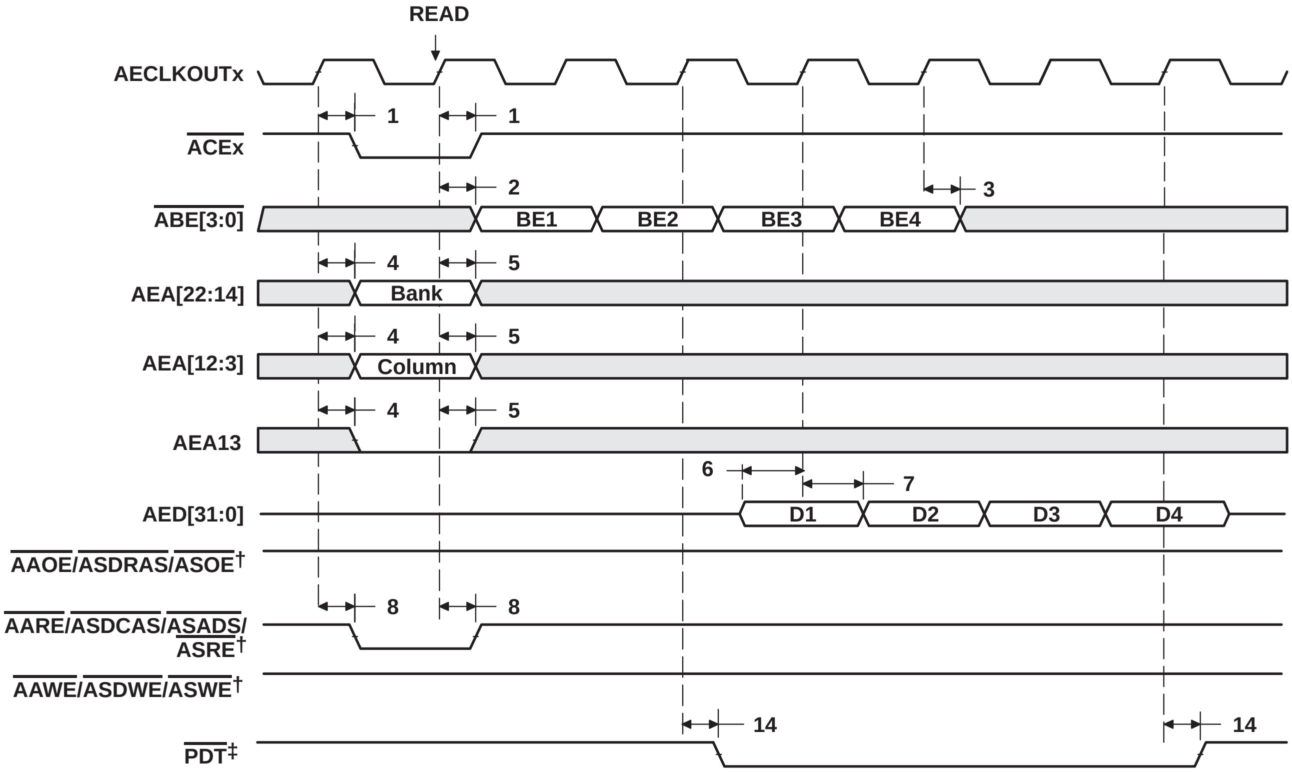
7.4 Synchronous DRAM Timing

Table 7−12. Timing Requirements for Synchronous DRAM Cycles for EMIFA Module (see Figure 7−12)

NO.			−400 −500		UNIT
			MIN	MAX	
6	$t_{su(EDV\text{-}EKO1H)}$	Setup time, read AEDx valid before AECLKOUTx high	2.1		ns
7	$t_{h(EKO1H\text{-}EDV)}$	Hold time, read AEDx valid after AECLKOUTx high	2.5		ns

Table 7−13. Switching Characteristics Over Recommended Operating Conditions for Synchronous DRAM Cycles for EMIFA Module (see Figure 7−12−Figure 7−19)

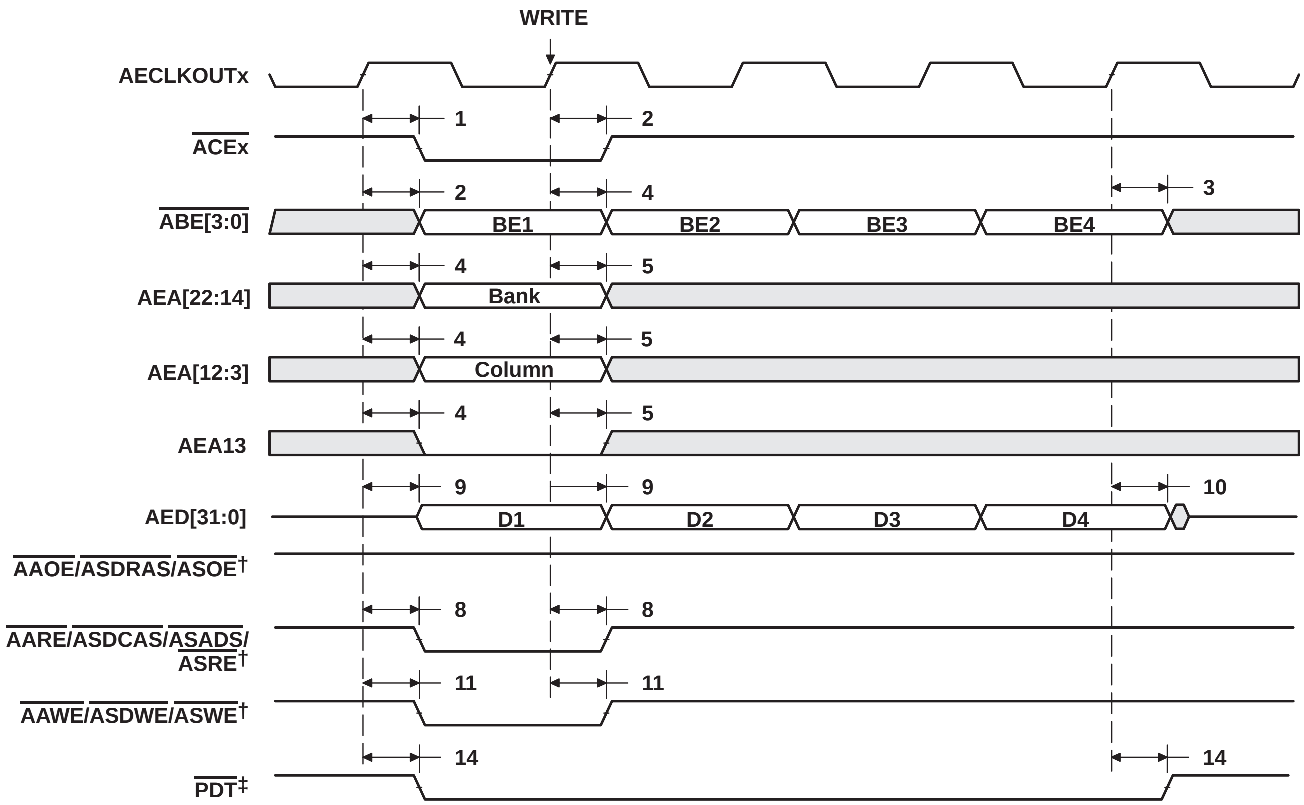
NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{d(EKO1H\text{-}CEV)}$	Delay time, AECLKOUTx high to $\overline{\text{ACEx}}$ valid	1.3	6.4	ns
2	$t_{d(EKO1H\text{-}BEV)}$	Delay time, AECLKOUTx high to $\overline{\text{ABEx}}$ valid		6.4	ns
3	$t_{d(EKO1H\text{-}BEIV)}$	Delay time, AECLKOUTx high to $\overline{\text{ABEx}}$ invalid	1.3		ns
4	$t_{d(EKO1H\text{-}EAV)}$	Delay time, AECLKOUTx high to AEAx valid		6.4	ns
5	$t_{d(EKO1H\text{-}EAIV)}$	Delay time, AECLKOUTx high to AEAx invalid	1.3		ns
8	$t_{d(EKO1H\text{-}CASV)}$	Delay time, AECLKOUTx high to $\overline{\text{ASDCAS}}$ valid	1.3	6.4	ns
9	$t_{d(EKO1H\text{-}EDV)}$	Delay time, AECLKOUTx high to AEDx valid		6.4	ns
10	$t_{d(EKO1H\text{-}EDIV)}$	Delay time, AECLKOUTx high to AEDx invalid	1.3		ns
11	$t_{d(EKO1H\text{-}WEV)}$	Delay time, AECLKOUTx high to $\overline{\text{ASDWE}}$ valid	1.3	6.4	ns
12	$t_{d(EKO1H\text{-}RAS)}$	Delay time, AECLKOUTx high to $\overline{\text{ASDRAS}}$ valid	1.3	6.4	ns
13	$t_{d(EKO1H\text{-}ACKEV)}$	Delay time, AECLKOUTx high to ASDCKE valid	1.3	6.4	ns
14	$t_{d(EKO1H\text{-}PDTV)}$	Delay time, AECLKOUTx high to $\overline{\text{PDT}}$ valid	1.3	6.4	ns



† AARE/ASDCAS/ASADS/ASRE, AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE operate as ASDCAS, ASDWE, and ASDRAS, respectively, during SDRAM accesses.

‡ PDT signal is only asserted when the EDMA is in PDT mode (set the PDTS bit to 1 in the EDMA options parameter RAM). For PDT read, data is not latched into EMIF. The PDTRL field in the PDT control register (PDTCTL) configures the latency of the PDT signal with respect to the data phase of a read transaction. The latency of the PDT signal for a read can be programmed to 0, 1, 2, or 3 by setting PDTRL to 00, 01, 10, or 11, respectively. PDTRL equals 00 (zero latency) in Figure 7−12.

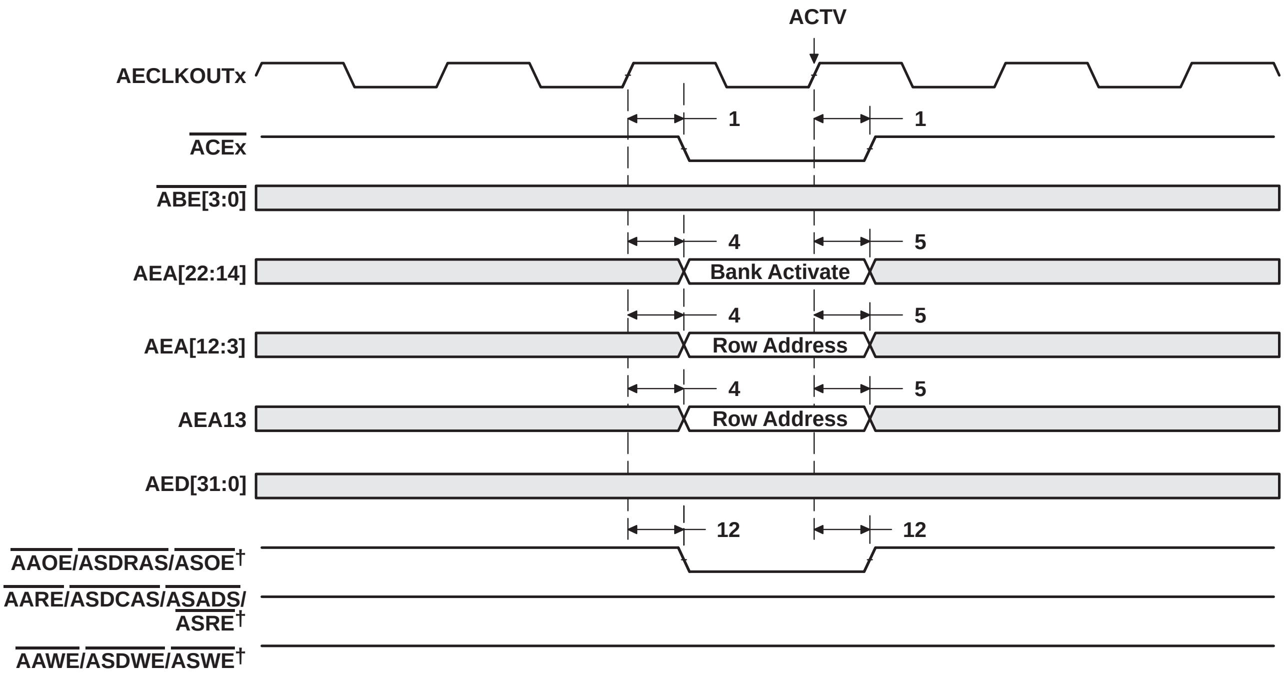
Figure 7−12. SDRAM Read Command (CAS Latency 3) for EMIFA



† $\overline{\text{AARE}}$/$\overline{\text{ASDCAS}}$/$\overline{\text{ASADS}}$/$\overline{\text{ASRE}}$, $\overline{\text{AAWE}}$/$\overline{\text{ASDWE}}$/$\overline{\text{ASWE}}$, and $\overline{\text{AAOE}}$/$\overline{\text{ASDRAS}}$/$\overline{\text{ASOE}}$ operate as $\overline{\text{AsDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$, respectively, during SDRAM accesses.

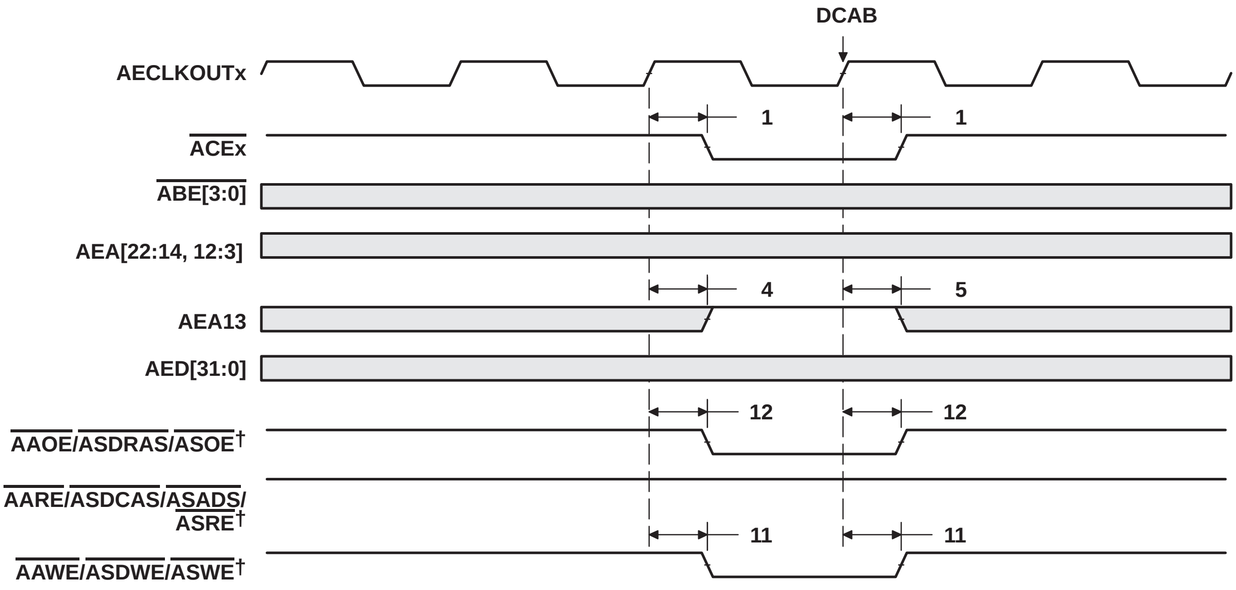
‡ $\overline{\text{PDT}}$ signal is only asserted when the EDMA is in PDT mode (set the PDTD bit to 1 in the EDMA options parameter RAM). For $\overline{\text{PDT}}$ write, data is not driven (in High-Z). The PDTWL field in the $\overline{\text{PDT}}$ control register (PDTCTL) configures the latency of the $\overline{\text{PDT}}$ signal with respect to the data phase of a write transaction. The latency of the $\overline{\text{PDT}}$ signal for a write transaction can be programmed to 0, 1, 2, or 3 by setting PDTWL to 00, 01, 10, or 11, respectively. PDTWL equals 00 (zero latency) in Figure 7−13.

Figure 7−13. SDRAM Write Command for EMIFA



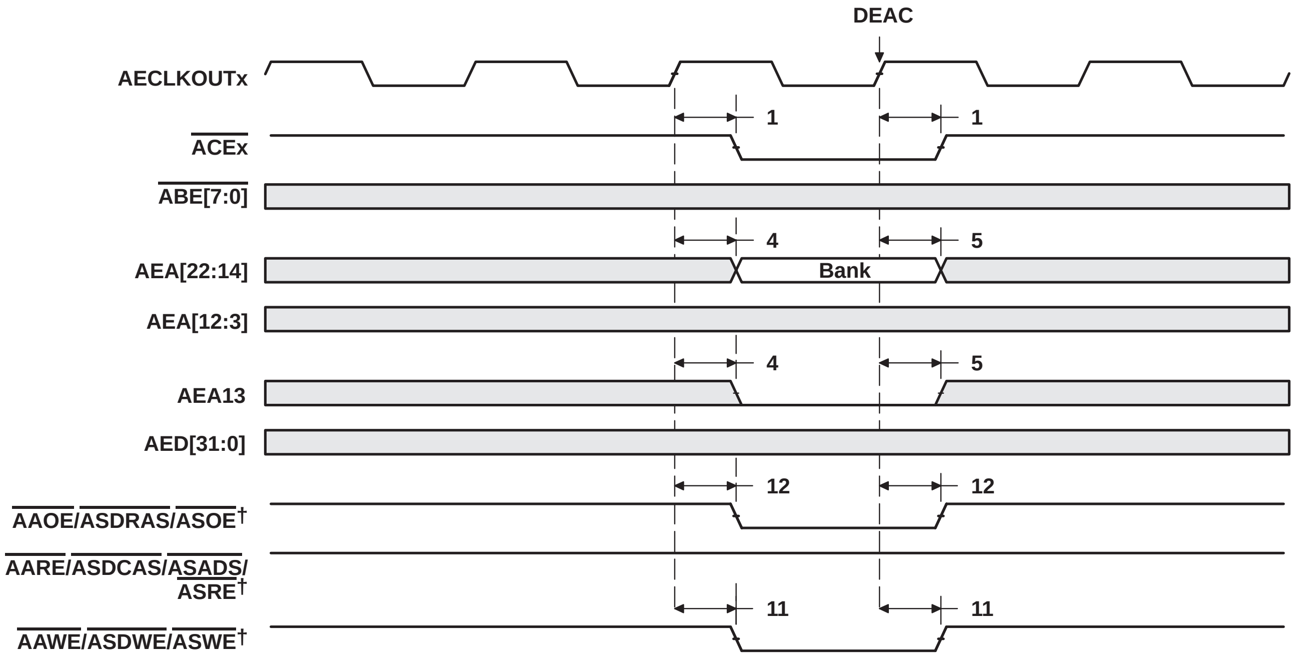
† AARE/ASDCAS/ASADS/ASRE, AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE operate as ASDCAS, ASDWE, and ASDRAS, respectively, during SDRAM accesses.

Figure 7−14. SDRAM ACTV Command for EMIFA



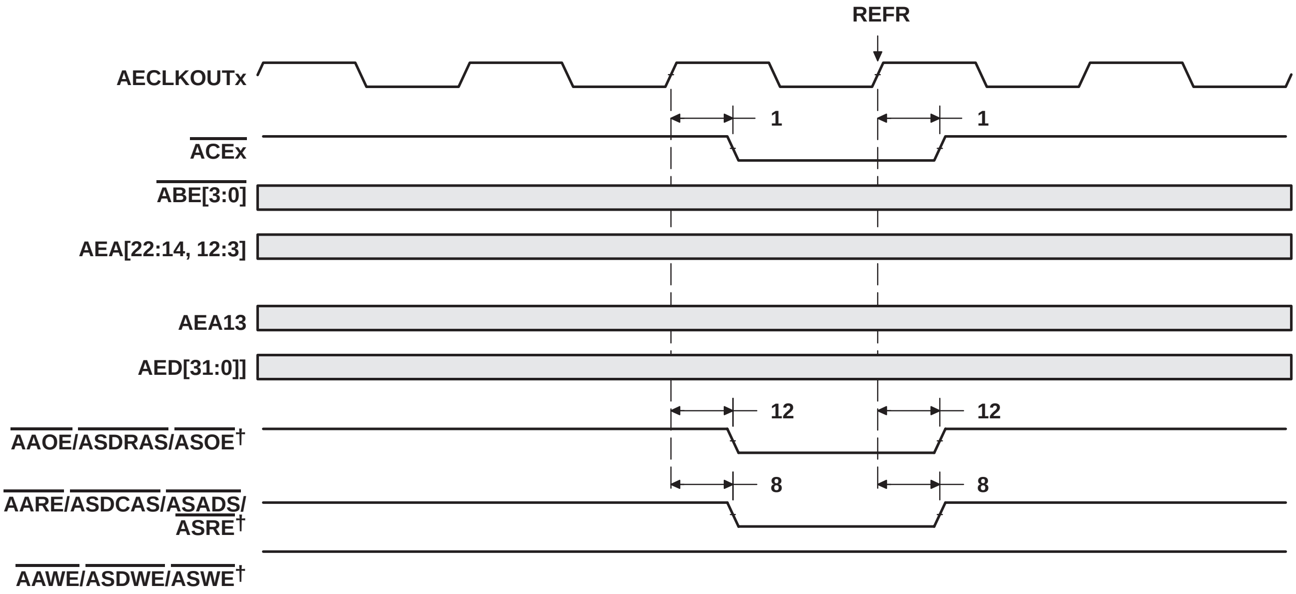
† AARE/ASDCAS/ASADS/ASRE, AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE operate as ASDCAS, ASDWE, and ASDRAS, respectively, during SDRAM accesses.

Figure 7−15. SDRAM DCAB Command for EMIFA



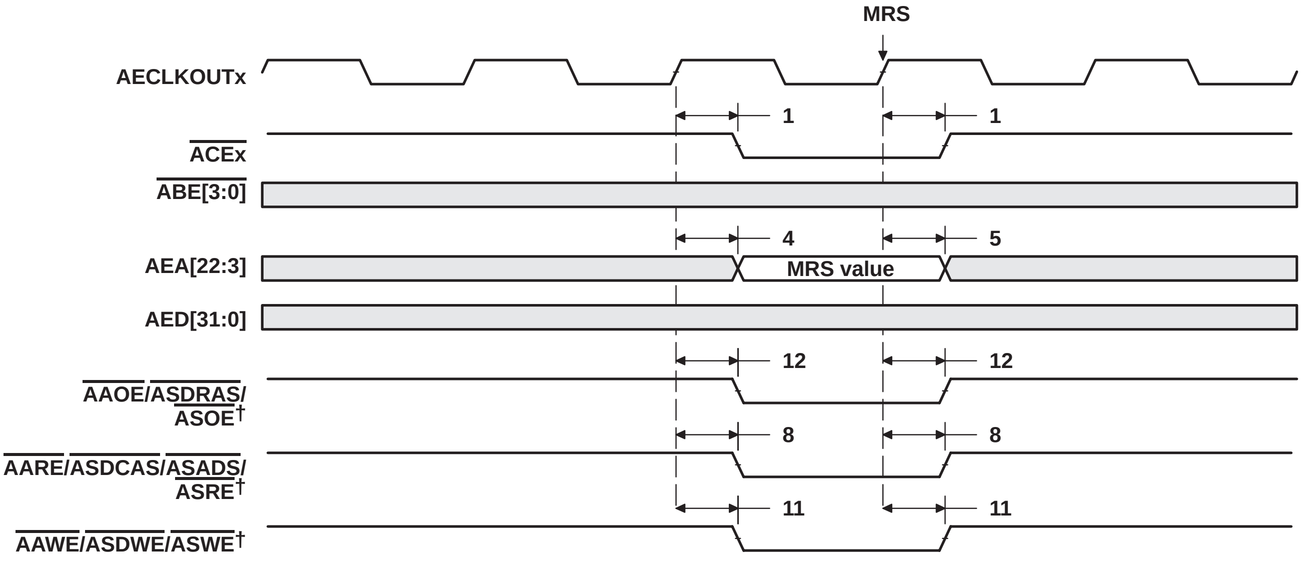
† $\overline{\text{AARE}}/\overline{\text{ASDCAS}}/\overline{\text{ASADS}}/\overline{\text{ASRE}}$, $\overline{\text{AAWE}}/\overline{\text{ASDWE}}/\overline{\text{ASWE}}$, and $\overline{\text{AAOE}}/\overline{\text{ASDRAS}}/\overline{\text{ASOE}}$ operate as $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$, respectively, during SDRAM accesses.

Figure 7−16. SDRAM DEAC Command for EMIFA



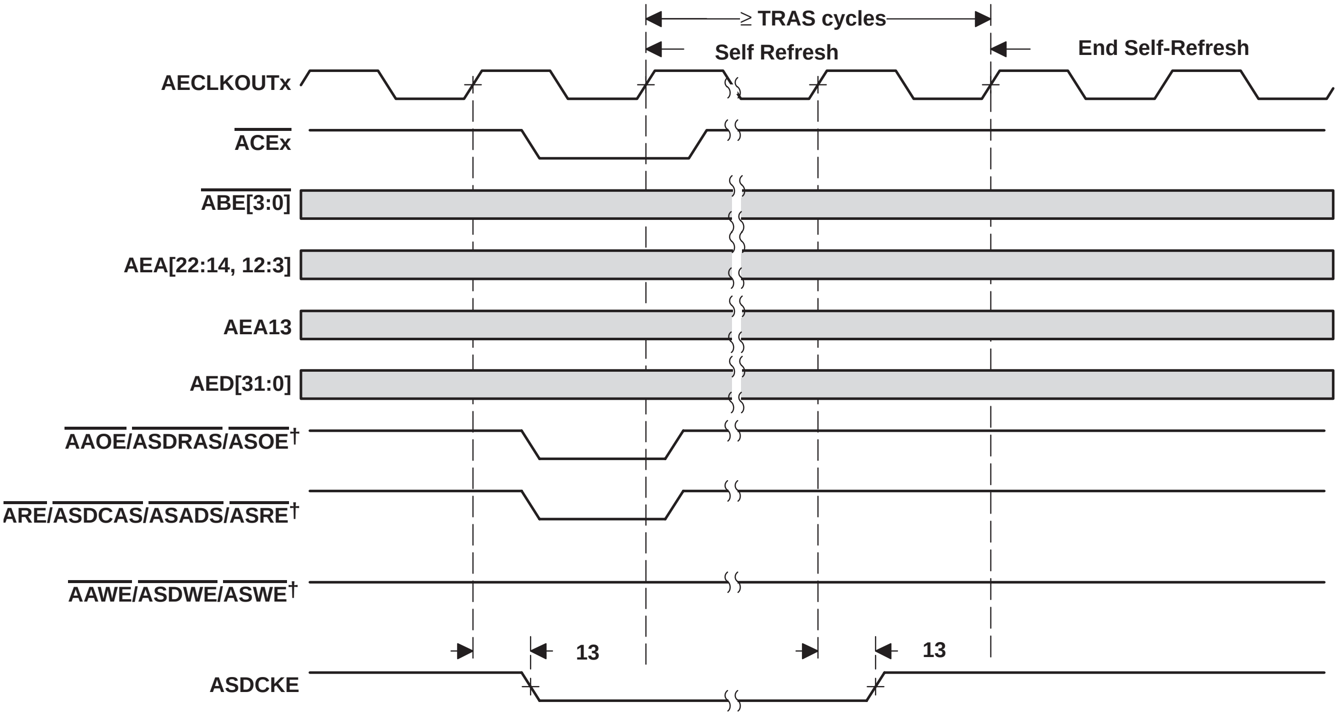
† $\overline{\text{AARE}}/\overline{\text{ASDCAS}}/\overline{\text{ASADS}}/\overline{\text{ASRE}}$, $\overline{\text{AAWE}}/\overline{\text{ASDWE}}/\overline{\text{ASWE}}$, and $\overline{\text{AAOE}}/\overline{\text{ASDRAS}}/\overline{\text{ASOE}}$ operate as $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$, respectively, during SDRAM accesses.

Figure 7−17. SDRAM REFR Command for EMIFA



† AARE/ASDCAS/ASADS/ASRE, AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE operate as ASDCAS, ASDWE, and ASDRAS, respectively, during SDRAM accesses.

Figure 7−18. SDRAM MRS Command for EMIFA



† AARE/ASDCAS/ASADS/ASRE, AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE operate as ASDCAS, ASDWE, and ASDRAS, respectively, during SDRAM accesses.

Figure 7−19. SDRAM Self-Refresh Timing for EMIFA

7.5 HOLD/HOLDA Timing

Table 7−14. Timing Requirements for the HOLD/HOLDA Cycles for EMIFA Module[†] (see Figure 7−20)

NO.			−400 −500		UNIT
			MIN	MAX	
3	$t_{h(HOLDAL-HOLDL)}$	Hold time, HOLD low after HOLDA low	E		ns

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

Table 7−15. Switching Characteristics Over Recommended Operating Conditions for the HOLD/HOLDA Cycles for EMIFA Module[†‡§] (see Figure 7−20)

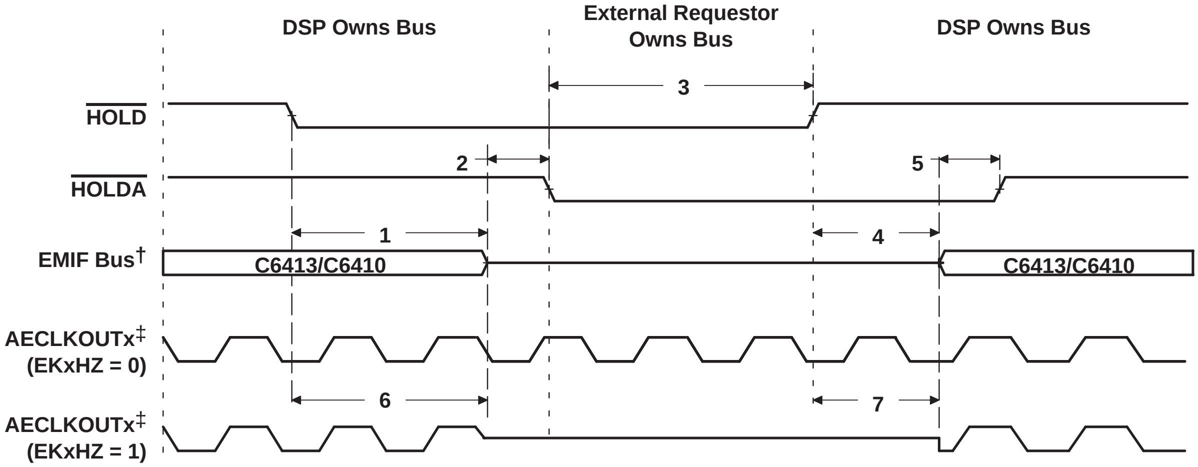
NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
1	$t_{d(HOLDL-EMHZ)}$	Delay time, HOLD low to EMIFA Bus high impedance	2E	[¶]	ns
2	$t_{d(EMHZ-HOLDAL)}$	Delay time, EMIF Bus high impedance to HOLDA low	0	2E	ns
4	$t_{d(HOLDH-EMLZ)}$	Delay time, HOLD high to EMIF Bus low impedance	2E	7E	ns
5	$t_{d(EMLZ-HOLDAH)}$	Delay time, EMIFA Bus low impedance to HOLDA high	0	2E	ns
6	$t_{d(HOLDL-EKOHZ)}$	Delay time, HOLD low to AECLKOUTx high impedance	2E	[¶]	ns
7	$t_{d(HOLDH-EKOLZ)}$	Delay time, HOLD high to AECLKOUTx low impedance	2E	7E	ns

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

[‡] EMIFA Bus consists of: ACE[3:0], ABE[3:0], AED[31:0], AEA[22:3], AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE , ASDCKE, ASOE3, and APDT.

[§] The EKxHZ bits in the EMIF Global Control register (GBLCTL) determine the state of the ECLKOUTx signals during HOLDA. If EKxHZ = 0, ECLKOUTx continues clocking during Hold mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during Hold mode, as shown in Figure 7−20.

[¶] All pending EMIF transactions are allowed to complete before HOLDA is asserted. If no bus transactions are occurring, then the minimum delay time can be achieved. Also, bus hold can be indefinitely delayed by setting NOHOLD = 1.



[†] EMIFA Bus consists of: ACE[3:0], ABE[3:0], AED[31:0], AEA[22:3], AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE, ASDCKE, ASOE3, and APDT.

[‡] The EKxHZ bits in the EMIF Global Control register (GBLCTL) determine the state of the ECLKOUTx signals during HOLDA. If EKxHZ = 0, ECLKOUTx continues clocking during Hold mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during Hold mode, as shown in Figure 7−20.

Figure 7−20. HOLD/HOLDA Timing for EMIFA

7.6 BUSREQ Timing

Table 7−16. Switching Characteristics Over Recommended Operating Conditions for the BUSREQ Cycles for EMIFA Module (see Figure 7−21)

NO.	PARAMETER		−400 −500 MIN	−400 −500 MAX	UNIT
1	$t_{d(AEKO1H\text{-}ABUSRV)}$	Delay time, AECLKOUTx high to ABUSREQ valid	0.6	7.1	ns

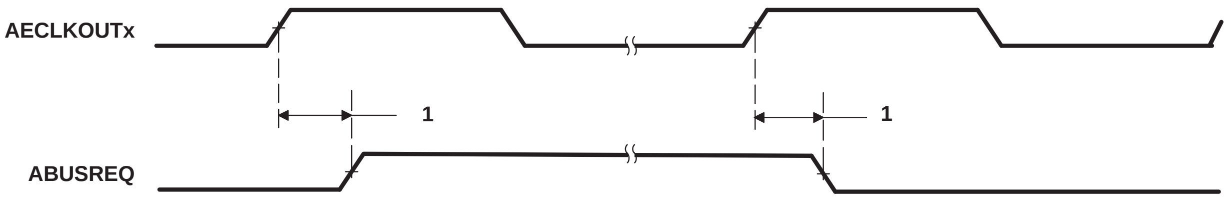


Figure 7−21. BUSREQ Timing for EMIFA

7.7 Reset Timing

Note: If a configuration pin must be routed out from the device, the internal pullup/pulldown (IPU/IPD) resistor should not be relied upon. TI recommends the use of an external pullup/pulldown resistor.

Table 7−17. Timing Requirements for Reset (see Figure 7−22)

NO.			−400 −500 MIN	−400 −500 MAX	UNIT
1	$t_{w(RST)}$	Width of the $\overline{\text{RESET}}$ pulse¶	250		µs
16	$t_{su(boot)}$	Setup time, boot configuration bits valid before $\overline{\text{RESET}}$ high†	4E or 4C‡		ns
17	$t_{h(boot)}$	Hold time, boot configuration bits valid after $\overline{\text{RESET}}$ high†	4P§		ns

† AEA[22:19], LENDIAN, BOOTMODE[1:0], and AECLKIN_SEL[1:0] are the boot configuration pins during device reset.

‡ E = 1/ECLKIN clock frequency in ns. C = 1/CLKIN clock frequency in ns.
Select the MIN parameter value, whichever value is larger.

§ P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

¶ The device must be reset after the oscillator has stabilized. If RESETz is low during power-up, it can be kept low until the oscillator has stabilized.
Note: a device reset does not affect the state of the oscillator.

Table 7−18. Switching Characteristics Over Recommended Operating Conditions During Reset[§#☆] (see Figure 7−22)

NO.	PARAMETER			−400, −500		UNIT
				MIN	MAX	
2	$t_{d(RSTL\text{-}ECKI)}$	Delay time, $\overline{\text{RESET}}$ low to ECLKIN synchronized internally		2E	3P + 20E	ns
3	$t_{d(RSTH\text{-}ECKI)}$	Delay time, $\overline{\text{RESET}}$ high to ECLKIN synchronized internally		2E	8P + 20E	ns
4	$t_{d(RSTL\text{-}ECKO1HZ)}$	Delay time, $\overline{\text{RESET}}$ low to ECLKOUT1 high impedance		2E		ns
5	$t_{d(RSTH\text{-}ECKO1V)}$	Delay time, $\overline{\text{RESET}}$ high to ECLKOUT1 valid			8P + 20E	ns
6	$t_{d(RSTL\text{-}EMIFZHZ)}$	Delay time, $\overline{\text{RESET}}$ low to EMIF Z high impedance		2E	3P + 4E	ns
7	$t_{d(RSTH\text{-}EMIFZV)}$	Delay time, $\overline{\text{RESET}}$ high to EMIF Z valid		16E	8P + 20E	ns
8	$t_{d(RSTL\text{-}EMIFHIV)}$	Delay time, $\overline{\text{RESET}}$ low to EMIF high group invalid		2E		ns
9	$t_{d(RSTH\text{-}EMIFHV)}$	Delay time, $\overline{\text{RESET}}$ high to EMIF high group valid			8P + 20E	ns
10	$t_{d(RSTL\text{-}EMIFLIV)}$	Delay time, $\overline{\text{RESET}}$ low to EMIF low group invalid		2E		ns
11	$t_{d(RSTH\text{-}EMIFLV)}$	Delay time, $\overline{\text{RESET}}$ high to EMIF low group valid			8P + 20E	ns
12	$t_{d(RSTL\text{-}HIGHIV)}$	Delay time, $\overline{\text{RESET}}$ low to high group invalid		0		ns
13	$t_{d(RSTH\text{-}HIGHV)}$	Delay time, $\overline{\text{RESET}}$ high to high group valid			11P	ns
14	$t_{d(RSTL\text{-}ZHZ)}$	Delay time, $\overline{\text{RESET}}$ low to Z group high impedance		0		ns
15	$t_{d(RSTH\text{-}ZV)}$	Delay time, $\overline{\text{RESET}}$ high to Z group valid		2P	8P	ns
18	$t_{d(OSCSTART)}$	Delay time, Internal oscillator startup time[‖¶]	CLKINSEL = 0		41032 x OSCIN	ns

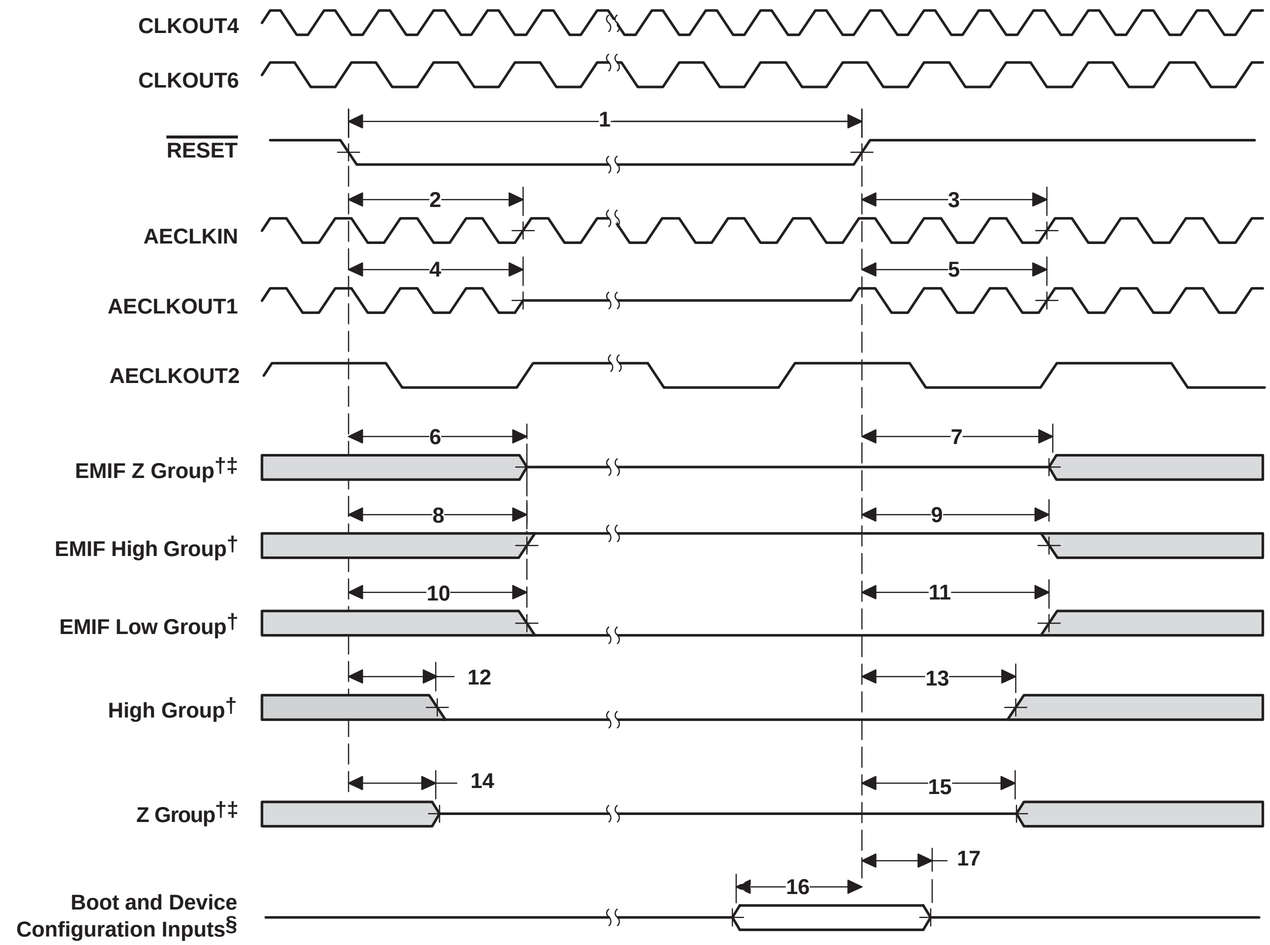
[§] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[¶] The device must be reset after the oscillator has stabilized. If RESETz is low during power-up, it can be kept low until the oscillator has stabilized. Note: a device reset does not affect the state of the oscillator.

[#] E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns for EMIFA.

[‖] Assuming core power supply has stabilized at recommended operating conditions.

[☆] EMIF Z group consists of:	AEA[22:3], AED[31:0], ACE[3:0], ABE[3:0], AARE/ASDCAS/ASADS/ASRE,AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE, ASOE3, ASDCKE, APDT., and AECLKOUT1
EMIF high group consists of:	AHOLDA (when the corresponding HOLD input is high)
EMIF low group consists of:	ABUSREQ; AHOLDA (when the corresponding HOLD input is low)
High group consists of:	HRDY (when HPI is enabled, otherwise in Z group)
Z group consists of:	CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKS0, CLKS1, DR0, DR1, CLKR0, CLKR1, FSR0, FSR1, TOUT0/HPI_EN, TOUT1/LENDIAN, GP0[7:0], HD[7:0], HD[15:8]/GP0[15:8], HD[21:16]/AXR1[5:0], HD22/AFSX1, HD23/AFSR1, HD24/ACLKX1, HD25/ACLKR1, HD26/AHCLKR1, HD27/AHCLKX1, HD28/AMUTE1, HD29/AMUTEIN1, HD30, HD31, HRDY, HDS2, HDS1/ACLKR1[3], HCS/ACLKR1[2], HAS/ACLKR1[1], HR/W/AFSR1[3], HHWIL/AFSR1[2] (16-bit HPI mode only), HCNTL0/AFSR1[1], HCNTL1, HINT,, ACLKR0, AFSR0, AHCLKR0, AMUTEIN0, AMUTE0, AXR0[5:0], SDA1, SCL1, SDA0, SCL0, TDO, and EMU[11:0]



[†] EMIF Z group consists of: AEA[22:3], AED[31:0], ACE[3:0], ABE[3:0], AARE/ASDCAS/ASADS/ASRE,AAWE/ASDWE/ASWE, and AAOE/ASDRAS/ASOE, ASOE3, ASDCKE, APDT., and AECLKOUT1

EMIF high group consists of: AHOLDA (when the corresponding HOLD input is high)

EMIF low group consists of: ABUSREQ; AHOLDA (when the corresponding HOLD input is low)

High group consists of: HRDY (when HPI is enabled, otherwise in Z group)

Z group consists of: CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKS0, CLKS1, DR0, DR1, CLKR0, CLKR1, FSR0, FSR1, TOUT0/HPI_EN, TOUT1/LENDIAN, GP0[7:0], HD[7:0], HD[15:8]/GP0[15:8], HD[21:16]/AXR1[5:0], HD22/AFSX1, HD23/AFSR1, HD24/ACLKX1, HD25/ACLKR1, HD26/AHCLKR1, HD27/AHCLKX1, HD28/AMUTE1, HD29/AMUTEIN1, HD30, HD31, HRDY, HDS2, HDS1/ACLKR1[3], HCS/ACLKR1[2], HAS/ACLKR1[1], HR/W/AFSR1[3], HHWIL/AFSR1[2] (16-bit HPI mode only), HCNTL0/AFSR1[1], HCNTL1, HINT,, ACLKR0, AFSR0, AHCLKR0, AMUTEIN0, AMUTE0, AXR0[5:0], SDA1, SCL1, SDA0, SCL0, TDO, and EMU[11:0]

[‡] If AEA[22:19], LENDIAN, BOOTMODE[1:0], and AECLKIN_SEL[1:0] pins are actively driven, care must be taken to ensure no timing contention between parameters 6, 7, 14, 15, 16, and 17.

[§] Boot and Device Configurations Inputs (during reset) include: AEA[22:19], LENDIAN, BOOTMODE[1:0], and AECLKIN_SEL[1:0].

Figure 7−22. Reset Timing[†]

7.8 External Interrupt Timing

Table 7−19. Timing Requirements for External Interrupts[†] (see Figure 7−23)

NO.			−400 −500 MIN	MAX	UNIT
1	$t_{w(ILOW)}$	Width of the NMI interrupt pulse low	4P		ns
		Width of the EXT_INT interrupt pulse low	8P		ns
2	$t_{w(IHIGH)}$	Width of the NMI interrupt pulse high	4P		ns
		Width of the EXT_INT interrupt pulse high	8P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

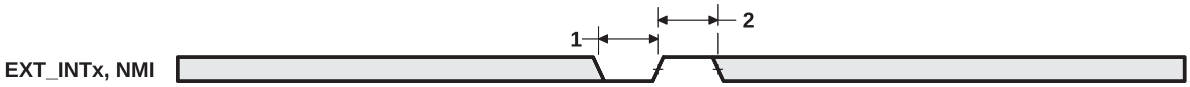


Figure 7−23. External/NMI Interrupt Timing

7.9 Multichannel Audio Serial Port (McASP) Timing

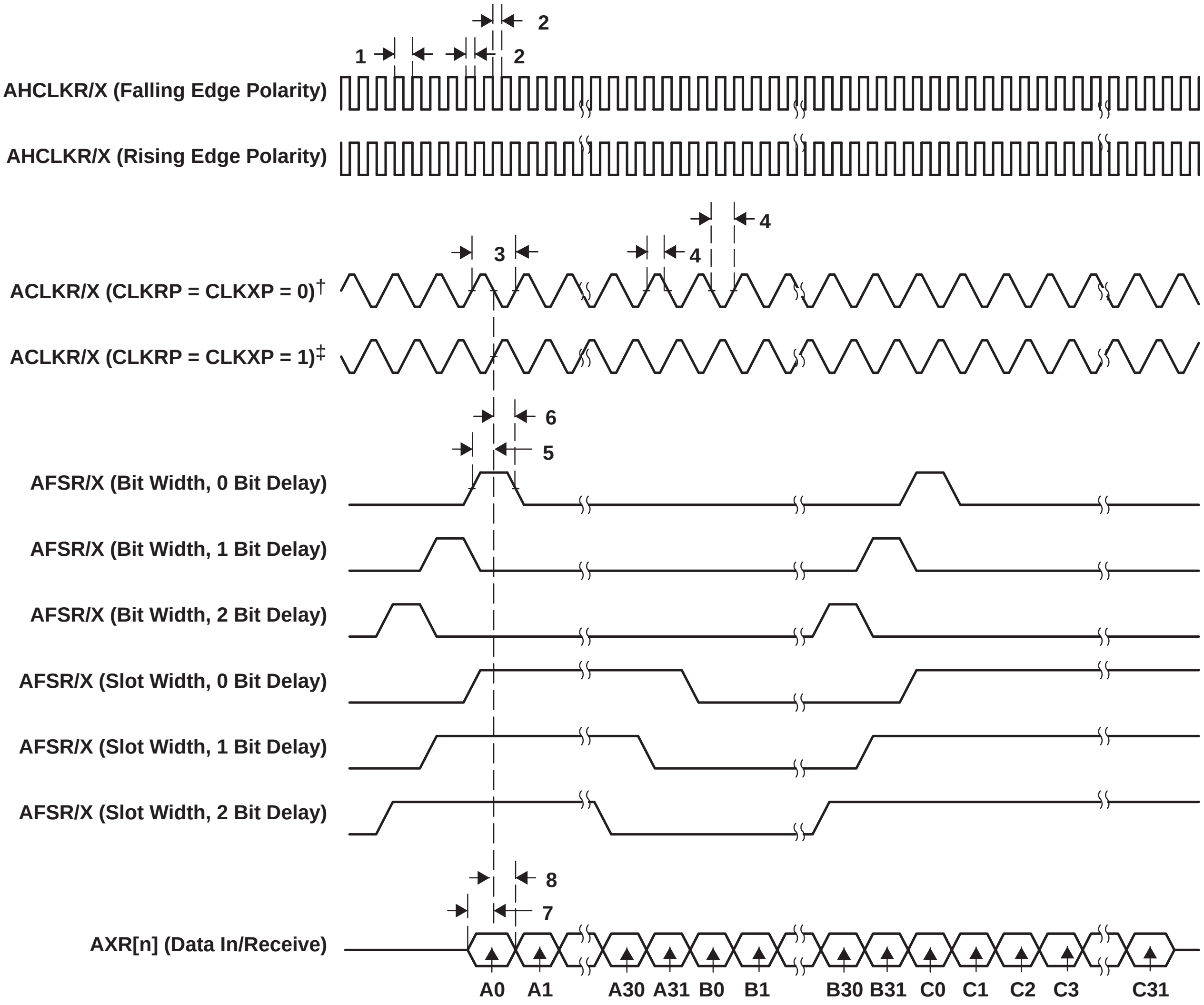
Table 7−20. Timing Requirements for McASP (see Figure 7−24 and Figure 7−25)

NO.				−400 −500 MIN	MAX	UNIT
1	$t_{c(AHCKRX)}$	Cycle time, AHCLKR/X		20		ns
2	$t_{w(AHCKRX)}$	Pulse duration, AHCLKR/X high or low		10		ns
3	$t_{c(CKRX)}$	Cycle time, ACLKR/X	ACLKR/X ext	25		ns
4	$t_{w(CKRX)}$	Pulse duration, ACLKR/X high or low	ACLKR/X ext	12.25		ns
5	$t_{su(FRXC-KRX)}$	Setup time, AFSR/X input valid before ACLKR/X latches data	ACLKR/X int	5		ns
			ACLKR/X ext	5		ns
6	$t_{h(CKRX-FRX)}$	Hold time, AFSR/X input valid after ACLKR/X latches data	ACLKR/X int	5		ns
			ACLKR/X ext	5		ns
7	$t_{su(AXR-CKRX)}$	Setup time, AXR input valid before ACLKR/X latches data	ACLKR/X int	5		ns
			ACLKR/X ext	5		ns
8	$t_{h(CKRX-AXR)}$	Hold time, AXR input valid after ACLKR/X latches data	ACLKR/X int	5		ns
			ACLKR/X ext	5		ns

Table 7−21. Switching Characteristics Over Recommended Operating Conditions for McASP (see Figure 7−26 and Figure 7−27)

NO.	PARAMETER			−400 −500 MIN	−400 −500 MAX	UNIT
9	$t_{c(AHCKRX)}$	Cycle time, AHCLKR/X		20		ns
10	$t_{w(AHCKRX)}$	Pulse duration, AHCLKR/X high or low		10		ns
11	$t_{c(CKRX)}$	Cycle time, ACLKR/X	ACLKR/X int	33		ns
12	$t_{w(CKRX)}$	Pulse duration, ACLKR/X high or low	ACLKR/X int	16.5		ns
13	$t_{d(CKRX\text{-}FRX)}$	Delay time, ACLKR/X transmit edge to AFSX/R output valid	ACLKR/X int	−1	5	ns
			ACLKR/X ext	2	12	ns
14	$t_{d(CKRX\text{-}AXRV)}$	Delay time, ACLKR/X transmit edge to AXR output valid	ACLKR/X int	−1	5	ns
			ACLKR/X ext	2	12	ns
15	$t_{dis(CKRX-AXRHZ)}$	Disable time, AXR high impedance following last data bit from ACLKR/X transmit edge	ACLKR/X int	0	10	ns
			ACLKR/X ext	0	10	ns

MULTICHANNEL AUDIO SERIAL PORT (McASP) TIMING (CONTINUED)

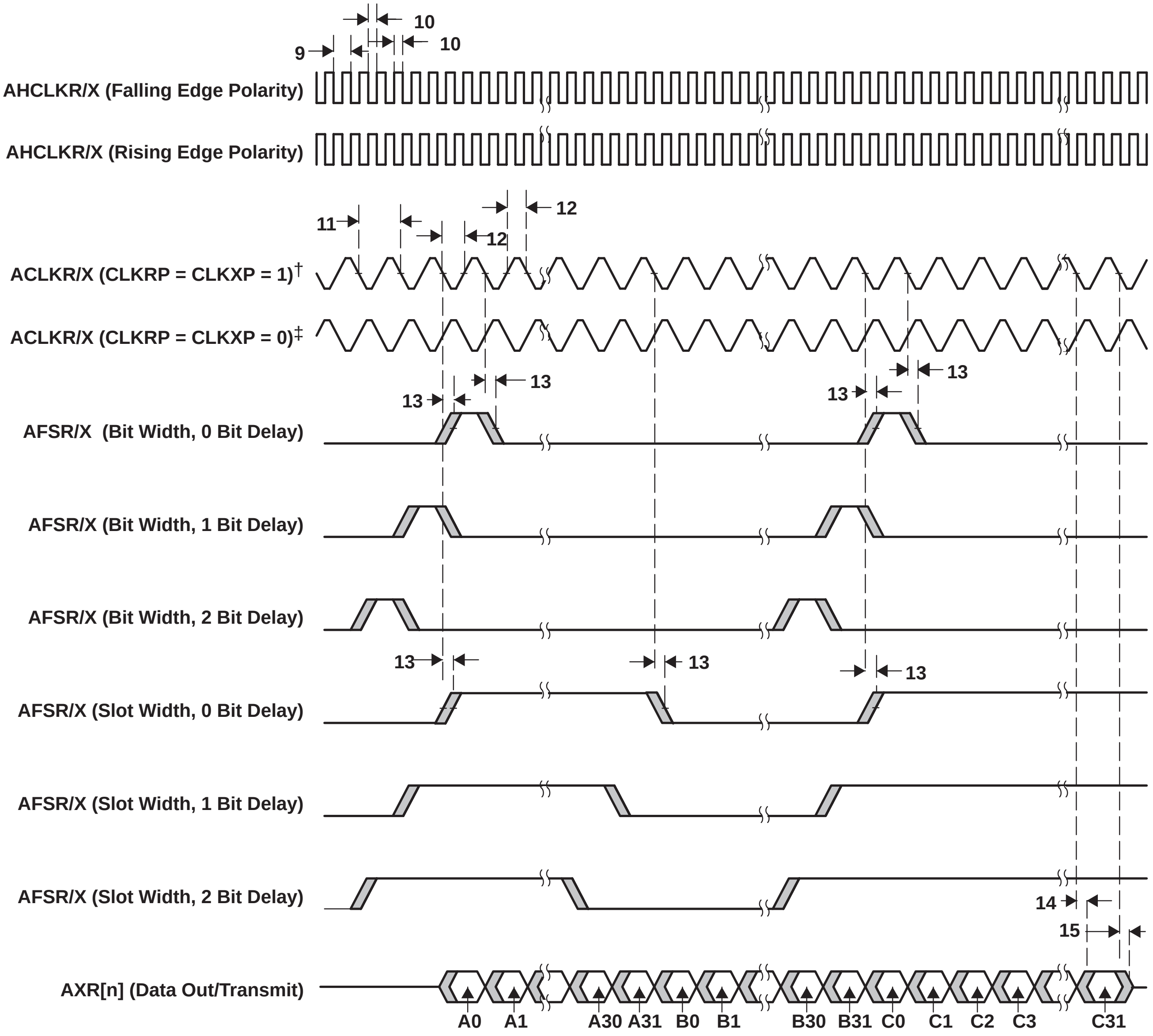


[†] For CLKRP = CLKXP = 0, the McASP transmitter is configured for rising edge (to shift data out) and the McASP receiver is configured for falling edge (to shift data in).

[‡] For CLKRP = CLKXP = 1, the McASP transmitter is configured for falling edge (to shift data out) and the McASP receiver is configured for rising edge (to shift data in).

Figure 7−28. McASP Input Timings

MULTICHANNEL AUDIO SERIAL PORT (McASP) TIMING (CONTINUED)



† For CLKRP = CLKXP = 1, the McASP transmitter is configured for falling edge (to shift data out) and the McASP receiver is configured for rising edge (to shift data in).

‡ For CLKRP = CLKXP = 0, the McASP transmitter is configured for rising edge (to shift data out) and the McASP receiver is configured for falling edge (to shift data in).

Figure 7−29. McASP Output Timings

7.10 Inter-Integrated Circuits (I2C) Timing

Table 7−22. Timing Requirements for I2C Timings[†] (see Figure 7−30)

NO.			−400 −500				UNIT
			STANDARD MODE		FAST MODE		
			MIN	MAX	MIN	MAX	
1	$t_{c(SCL)}$	Cycle time, SCL	10		2.5		µs
2	$t_{su(SCLH\text{-}SDAL)}$	Setup time, SCL high before SDA low (for a repeated START condition)	4.7		0.6		µs
3	$t_{h(SCLL\text{-}SDAL)}$	Hold time, SCL low after SDA low (for a START and a repeated START condition)	4		0.6		µs
4	$t_{w(SCLL)}$	Pulse duration, SCL low	4.7		1.3		µs
5	$t_{w(SCLH)}$	Pulse duration, SCL high	4		0.6		µs
6	$t_{su(SDAV\text{-}SDLH)}$	Setup time, SDA valid before SCL high	250		100[‡]		ns
7	$t_{h(SDA\text{-}SDLL)}$	Hold time, SDA valid after SCL low (For I^{2}C bus™ devices)	0[§]		0[§]	0.9[¶]	µs
8	$t_{w(SDAH)}$	Pulse duration, SDA high between STOP and START conditions	4.7		1.3		µs
9	$t_{r(SDA)}$	Rise time, SDA		1000	20 + 0.1C_b[#]	300	ns
10	$t_{r(SCL)}$	Rise time, SCL		1000	20 + 0.1C_b[#]	300	ns
11	$t_{f(SDA)}$	Fall time, SDA		300	20 + 0.1C_b[#]	300	ns
12	$t_{f(SCL)}$	Fall time, SCL		300	20 + 0.1C_b[#]	300	ns
13	$t_{su(SCLH\text{-}SDAH)}$	Setup time, SCL high before SDA high (for STOP condition)	4		0.6		µs
14	$t_{w(SP)}$	Pulse duration, spike (must be suppressed)			0	50	ns
15	C_b[#]	Capacitive load for each bus line		400		400	pF

[†] The I2C pins SDA and SCL do not feature fail-safe I/O buffers. These pins could potentially draw current when the device is powered down.

[‡] A Fast-mode I^{2}C-bus™ device can be used in a Standard-mode I^{2}C-bus™ system, but the requirement $t_{su(SDA-SCLH)} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line t_r max + $t_{su(SDA-SCLH)}$ = 1000 + 250 = 1250 ns (according to the Standard-mode I^{2}C-Bus Specification) before the SCL line is released.

[§] A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL.

[¶] The maximum $t_{h(SDA-SCLL)}$ has only to be met if the device does not stretch the low period [$t_{w(SCLL)}$] of the SCL signal.

[#] C_b = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times are allowed.

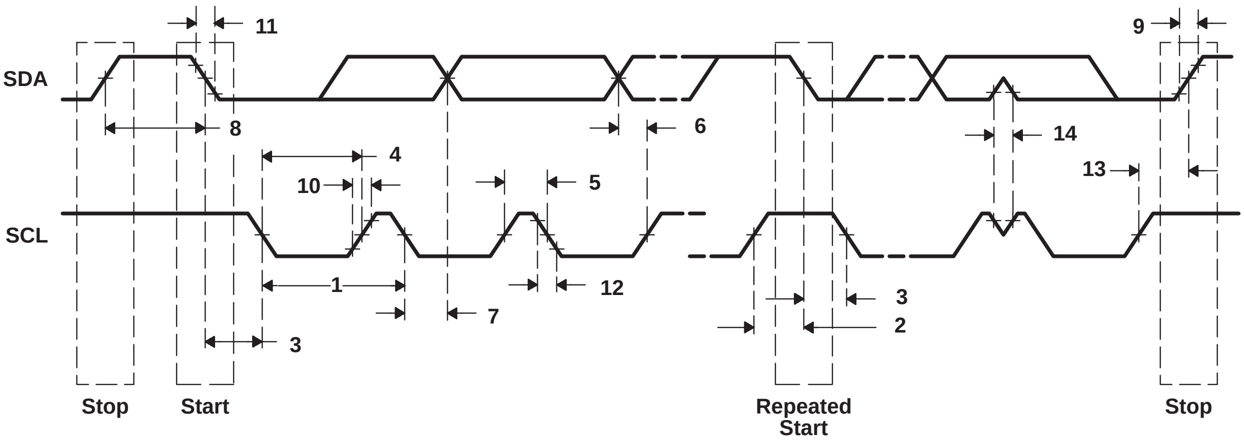


Figure 7−30. I2C Receive Timings

Table 7−23. Switching Characteristics for I2C Timings† (see Figure 7−31)

NO.	PARAMETER	−400 −500				UNIT
		STANDARD MODE		FAST MODE		
		MIN	MAX	MIN	MAX	
16	$t_{c(SCL)}$ Cycle time, SCL	10		2.5		µs
17	$t_{d(SCLH-SDAL)}$ Delay time, SCL high to SDA low (for a repeated START condition)	4.7		0.6		µs
18	$t_{d(SDAL-SCLL)}$ Delay time, SDA low to SCL low (for a START and a repeated START condition)	4		0.6		µs
19	$t_{w(SCLL)}$ Pulse duration, SCL low	4.7		1.3		µs
20	$t_{w(SCLH)}$ Pulse duration, SCL high	4		0.6		µs
21	$t_{d(SDAV-SDLH)}$ Delay time, SDA valid to SCL high	250		100		ns
22	$t_{v(SDLL-SDAV)}$ Valid time, SDA valid after SCL low (For I^{2}C bus™ devices)	0		0	0.9	µs
23	$t_{w(SDAH)}$ Pulse duration, SDA high between STOP and START conditions	4.7		1.3		µs
24	$t_{r(SDA)}$ Rise time, SDA		1000	20 + 0.1C_b†	300	ns
25	$t_{r(SCL)}$ Rise time, SCL		1000	20 + 0.1C_b†	300	ns
26	$t_{f(SDA)}$ Fall time, SDA		300	20 + 0.1C_b†	300	ns
27	$t_{f(SCL)}$ Fall time, SCL		300	20 + 0.1C_b†	300	ns
28	$t_{d(SCLH-SDAH)}$ Delay time, SCL high to SDA high (for STOP condition)	4		0.6		µs
29	C_p Capacitance for each I2C pin		10		10	pF

† C_b = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times are allowed.

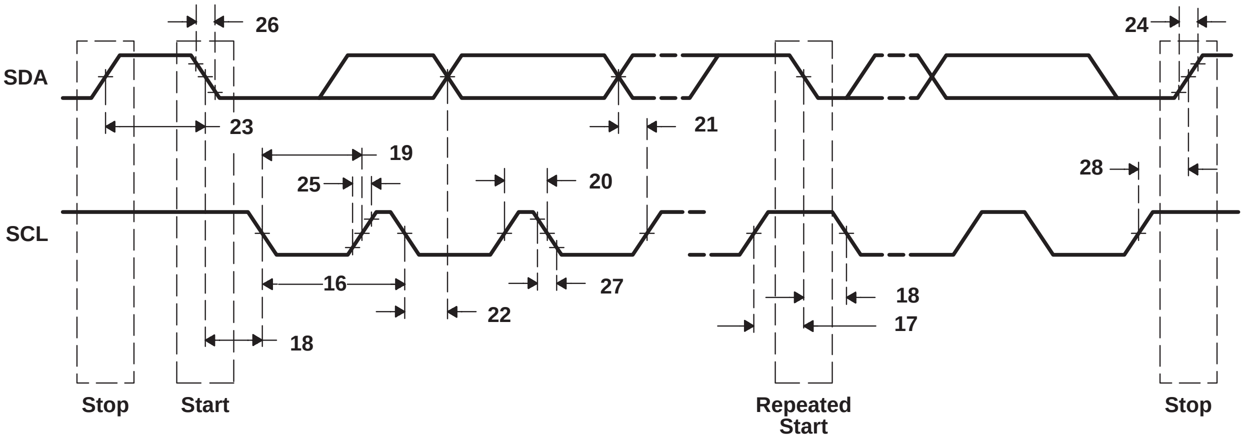


Figure 7−31. I2C Transmit Timings

7.11 Host-Port Interface (HPI) Timing

Table 7−24. Timing Requirements for Host-Port Interface Cycles[†‡] (see Figure 7−32 through Figure 7−39)

NO.			−400 −500		UNIT
			MIN	**MAX**	
1	$t_{\text{su(SELV-HSTBL)}}$	Setup time, select signals[§] valid before $\overline{\text{HSTROBE}}$ low	5		ns
2	$t_{\text{h(HSTBL-SELV)}}$	Hold time, select signals[§] valid after $\overline{\text{HSTROBE}}$ low	2.4		ns
3	$t_{\text{w(HSTBL)}}$	Pulse duration, $\overline{\text{HSTROBE}}$ low	4P[¶]		ns
4	$t_{\text{w(HSTBH)}}$	Pulse duration, $\overline{\text{HSTROBE}}$ high between consecutive accesses	4P		ns
10	$t_{\text{su(SELV-HASL)}}$	Setup time, select signals[§] valid before $\overline{\text{HAS}}$ low	5		ns
11	$t_{\text{h(HASL-SELV)}}$	Hold time, select signals[§] valid after $\overline{\text{HAS}}$ low	2		ns
12	$t_{\text{su(HDV-HSTBH)}}$	Setup time, host data valid before $\overline{\text{HSTROBE}}$ high	5		ns
13	$t_{\text{h(HSTBH-HDV)}}$	Hold time, host data valid after $\overline{\text{HSTROBE}}$ high	2.8		ns
14	$t_{\text{h(HRDYL-HSTBL)}}$	Hold time, $\overline{\text{HSTROBE}}$ low after $\overline{\text{HRDY}}$ low. $\overline{\text{HSTROBE}}$ should not be inactivated until $\overline{\text{HRDY}}$ is active (low); otherwise, HPI writes will not complete properly.	2		ns
18	$t_{\text{su(HASL-HSTBL)}}$	Setup time, $\overline{\text{HAS}}$ low before $\overline{\text{HSTROBE}}$ low	2		ns
19	$t_{\text{h(HSTBL-HASL)}}$	Hold time, $\overline{\text{HAS}}$ low after $\overline{\text{HSTROBE}}$ low	2.1		ns

[†] $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.

[‡] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[§] Select signals include: HCNTL[1:0] and HR/$\overline{\text{W}}$. For HPI16 mode only, select signals also include HHWIL.

[¶] Select the parameter value of 4P or 12.5 ns, whichever is larger.

Table 7−25. Switching Characteristics Over Recommended Operating Conditions During Host-Port Interface Cycles[†‡] (see Figure 7−32 through Figure 7−39)

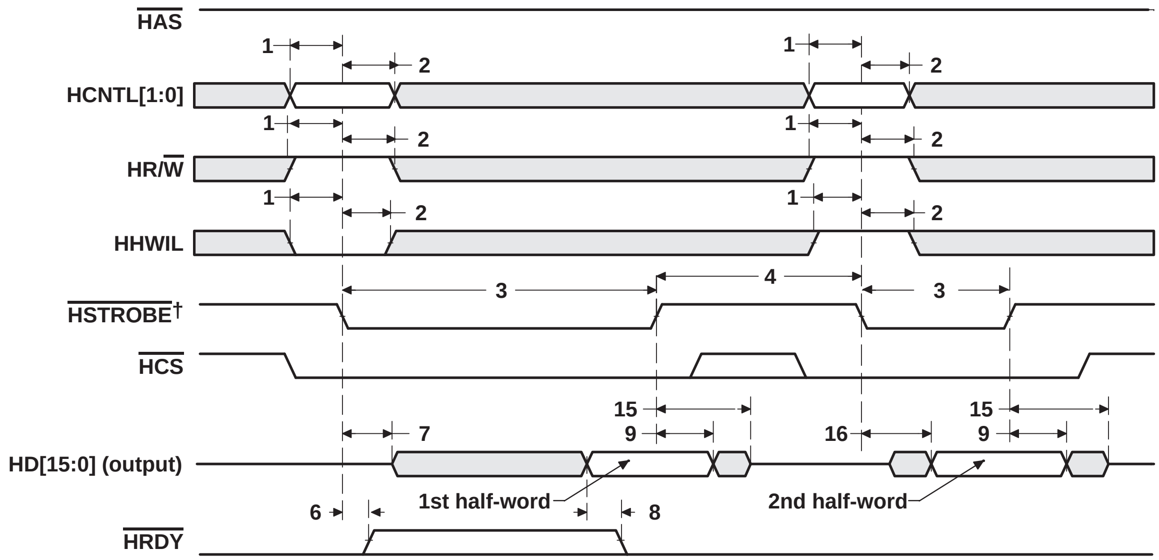
NO.	PARAMETER		−400 −500		UNIT
			MIN	**MAX**	
6	$t_{\text{d(HSTBL-HRDYH)}}$	Delay time, $\overline{\text{HSTROBE}}$ low to $\overline{\text{HRDY}}$ high[#]	1.3	4P + 8	ns
7	$t_{\text{d(HSTBL-HDLZ)}}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD low impedance for an HPI read	2		ns
8	$t_{\text{d(HDV-HRDYL)}}$	Delay time, HD valid to $\overline{\text{HRDY}}$ low	−3		ns
9	$t_{\text{oh(HSTBH-HDV)}}$	Output hold time, HD valid after $\overline{\text{HSTROBE}}$ high	1.5		ns
15	$t_{\text{d(HSTBH-HDHZ)}}$	Delay time, $\overline{\text{HSTROBE}}$ high to HD high impedance		12	ns
16	$t_{\text{d(HSTBL-HDV)}}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD valid (HPI16 mode, 2nd half-word only)		2P + 8 or 0P + 8[‖]	ns

[†] $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.

[‡] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

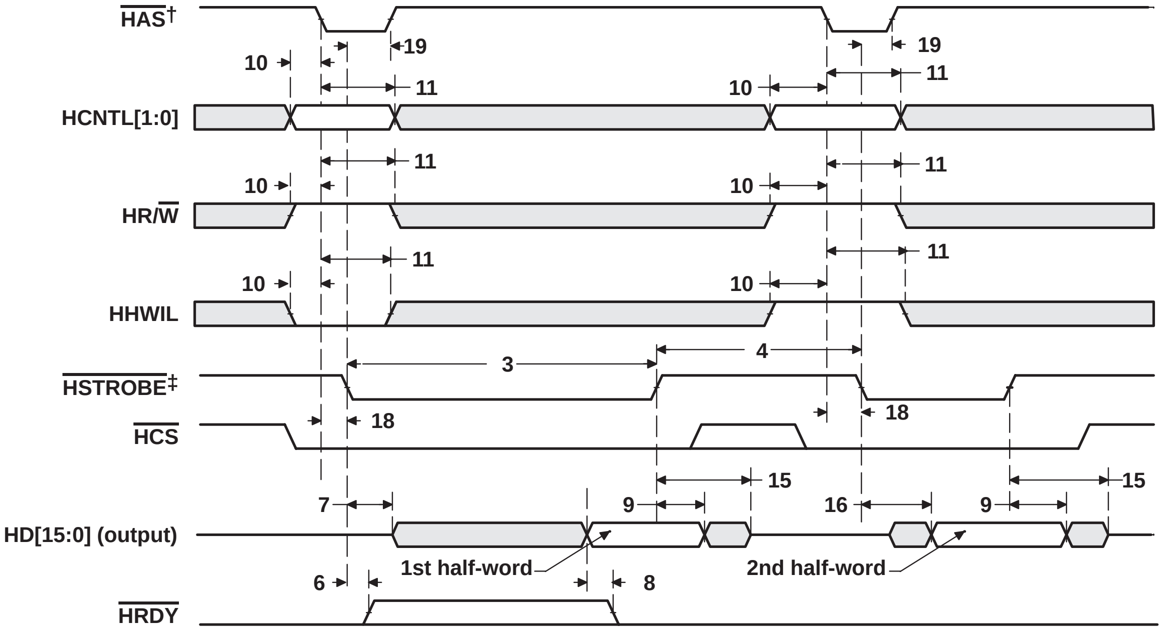
[#] This parameter is used during HPID reads and writes. For reads, at the beginning of a word transfer (HPI32) or the first half-word transfer (HPI16) on the falling edge of $\overline{\text{HSTROBE}}$, the HPI sends the request to the EDMA internal address generation hardware, and $\overline{\text{HRDY}}$ remains high until the EDMA internal address generation hardware loads the requested data into HPID. For writes, $\overline{\text{HRDY}}$ goes high if the internal write buffer is full.

[‖] If preceeding $\overline{\text{HSTROBE}}$ high pulse width > 6P, then this parameter value can be 0P + 8 ns.



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

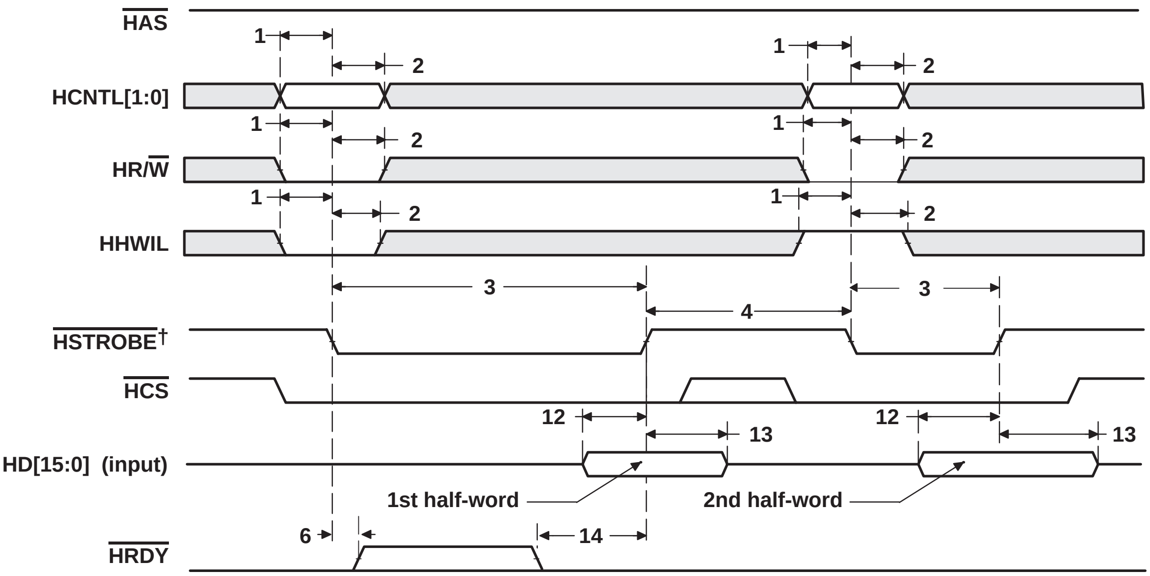
Figure 7−32. HPI16 Read Timing (HAS Not Used, Tied High)



† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.

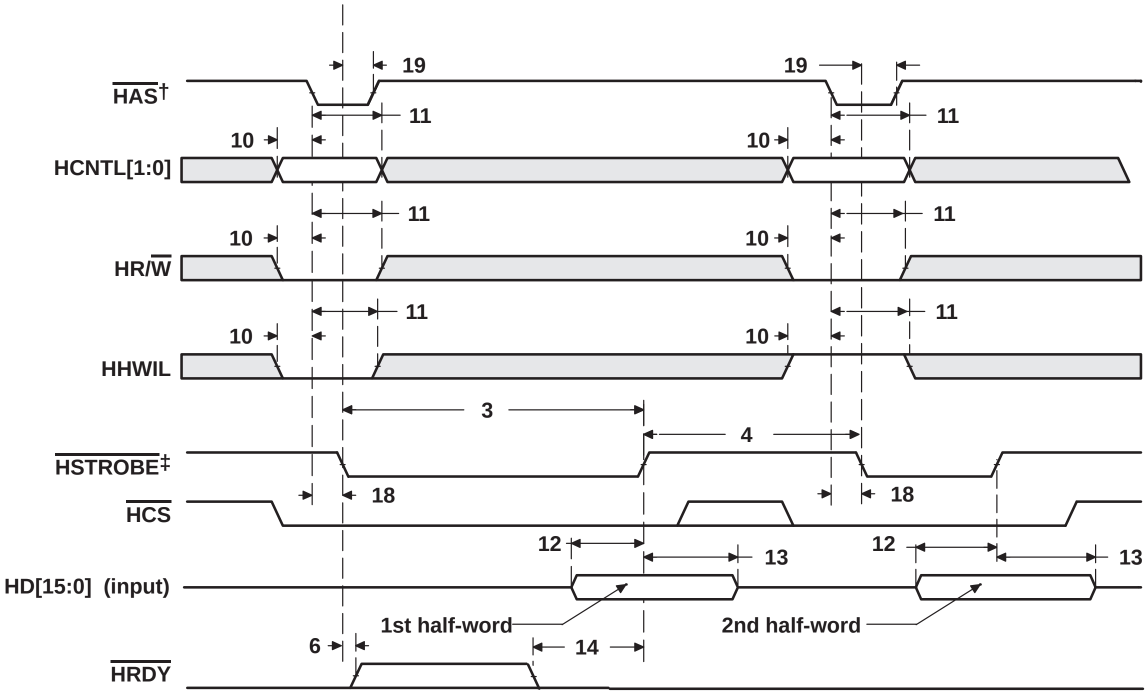
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 7−33. HPI16 Read Timing (HAS Used)



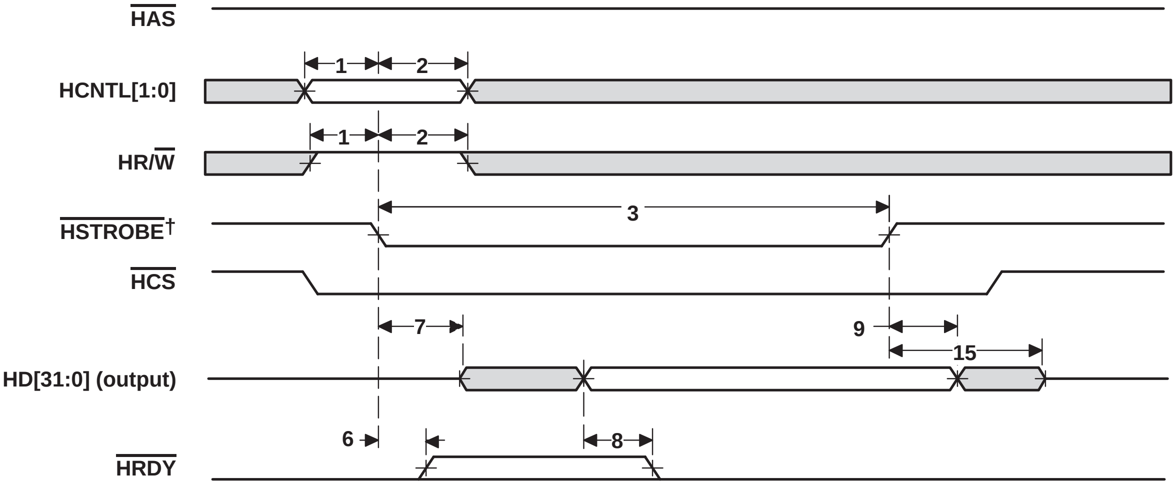
† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 7−34. HPI16 Write Timing (HAS Not Used, Tied High)



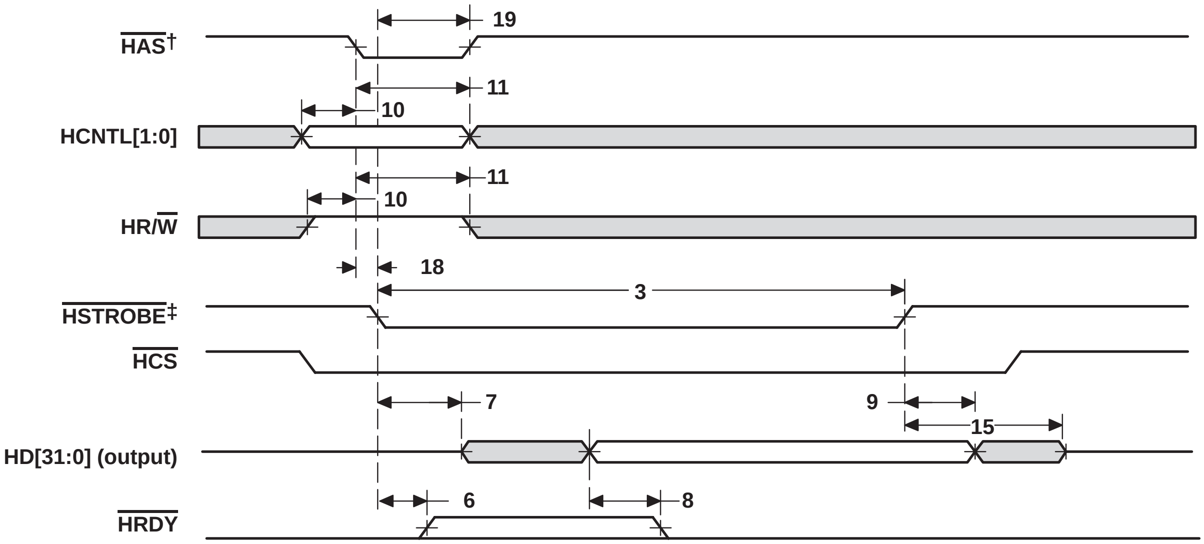
† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 7−35. HPI16 Write Timing (HAS Used)



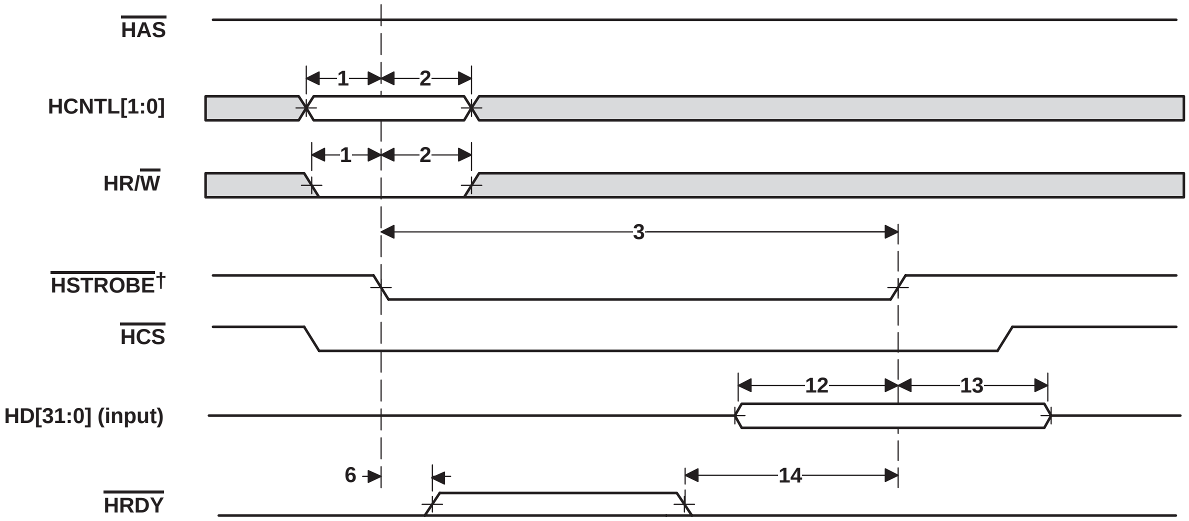
† HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 7−36. HPI32 Read Timing (HAS Not Used, Tied High)



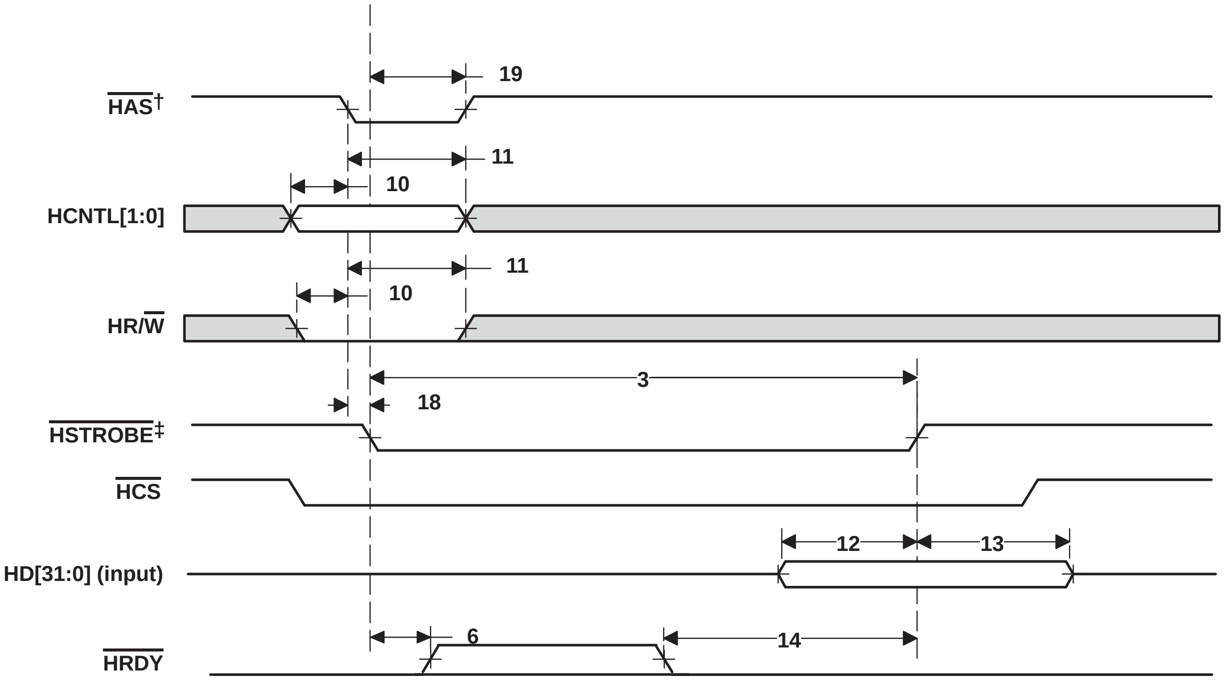
† For correct operation, strobe the HAS signal only once per HSTROBE active cycle.
‡ HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 7−37. HPI32 Read Timing (HAS Used)



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

Figure 7−38. HPI32 Write Timing (HAS Not Used, Tied High)



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

Figure 7−39. HPI32 Write Timing (HAS Used)

7.12 Multichannel Buffered Serial Port (McBSP) Timing

Table 7−26. Timing Requirements for McBSP[†][‡] (see Figure 7−40)

NO.				−400 −500		UNIT
				MIN	MAX	
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X ext	4P or 6.67[‡][§]		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	$0.5t_{c(CKRX)} - 1$[¶]		ns
5	$t_{su(FRH-CKRL)}$	Setup time, external FSR high before CLKR low	CLKR int	9		ns
			CLKR ext	1.3		
6	$t_{h(CKRL-FRH)}$	Hold time, external FSR high after CLKR low	CLKR int	6		ns
			CLKR ext	3		
7	$t_{su(DRV-CKRL)}$	Setup time, DR valid before CLKR low	CLKR int	8		ns
			CLKR ext	0.9		
8	$t_{h(CKRL-DRV)}$	Hold time, DR valid after CLKR low	CLKR int	3		ns
			CLKR ext	3.1		
10	$t_{su(FXH-CKXL)}$	Setup time, external FSX high before CLKX low	CLKX int	9		ns
			CLKX ext	1.3		
11	$t_{h(CKXL-FXH)}$	Hold time, external FSX high after CLKX low	CLKX int	6		ns
			CLKX ext	3		

[†] CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

[‡] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[§] Use whichever value is greater. Minimum CLKR/X cycle times *must* be met, even when CLKR/X is generated by an internal clock source. The minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.

[¶] This parameter applies to the maximum McBSP frequency. Operate serial clocks (CLKR/X) in the reasonable range of 40/60 duty cycle.

Table 7−27. Switching Characteristics Over Recommended Operating Conditions for McBSP[†‡] (see Figure 7−40)

NO.	PARAMETER			−400 −500		UNIT
				MIN	MAX	
1	$t_{d(CKSH\text{-}CKRXH)}$	Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input		1.4	10	ns
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X int	4P or 6.67[§¶#]		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X int	C − 1[||]	C + 1[||]	ns
4	$t_{d(CKRH\text{-}FRV)}$	Delay time, CLKR high to internal FSR valid	CLKR int	−2.1	3	ns
9	$t_{d(CKXH\text{-}FXV)}$	Delay time, CLKX high to internal FSX valid	CLKX int	−1.7	3	ns
			CLKX ext	1.7	9	
12	$t_{dis(CKXH\text{-}DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	CLKX int	−3.9	4	ns
			CLKX ext	2	9	
13	$t_{d(CKXH\text{-}DXV)}$	Delay time, CLKX high to DX valid	CLKX int	−3.9 + D1☆	4 + D2☆	ns
			CLKX ext	2.0 + D1☆	9 + D2☆	
14	$t_{d(FXH\text{-}DXV)}$	Delay time, FSX high to DX valid ONLY applies when in data delay 0 (XDATDLY = 00b) mode	FSX int	−2.3 + D1[□]	5.6 + D2[□]	ns
			FSX ext	1.9 + D1[□]	9 + D2[□]	

[†] CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

[‡] Minimum delay times also represent minimum output hold times.

[§] Minimum CLKR/X cycle times must be met, even when CLKR/X is generated by an internal clock source. Minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.

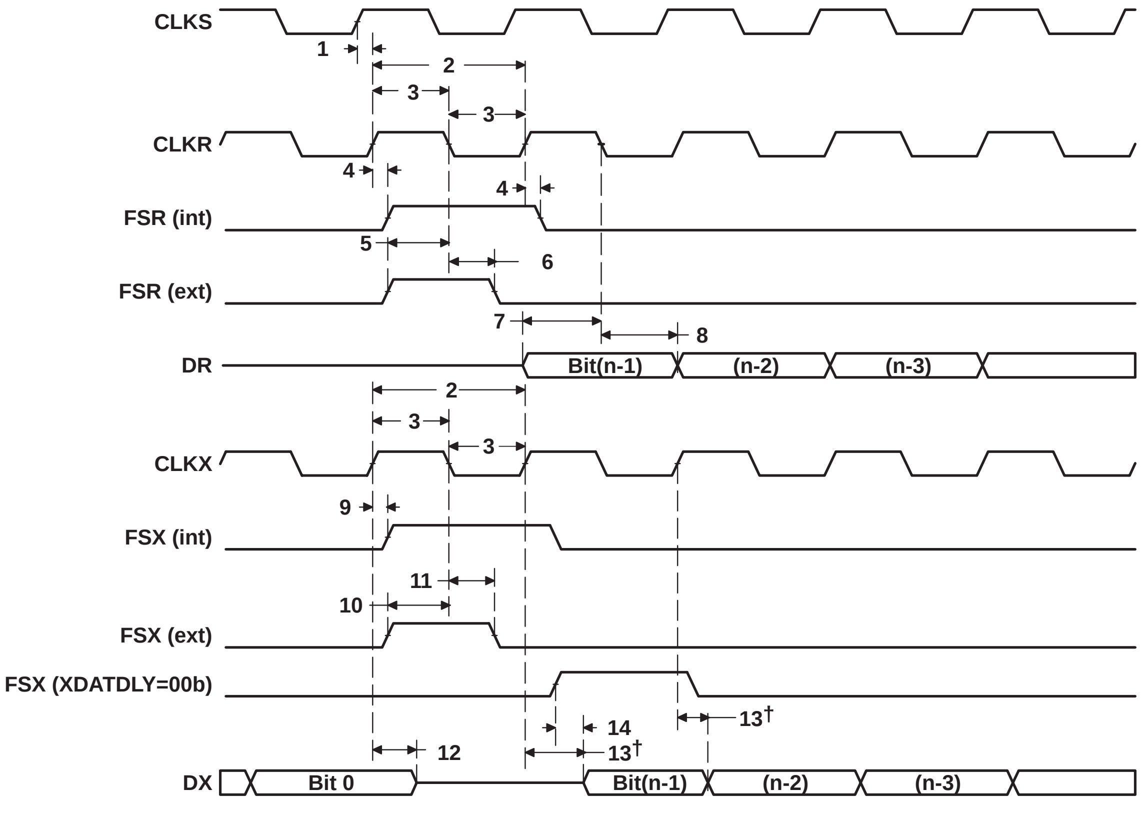
[¶] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[#] Use whichever value is greater.

[||] C = H or L
S = sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
CLKGDV should be set appropriately to ensure the McBSP bit rate does not exceed the maximum limit (see ¶ footnote above).

☆Extra delay from CLKX high to DX valid **applies *only* to the first data bit of a device**, if and only if DXENA = 1 in SPCR.
if DXENA = 0, then D1 = D2 = 0
if DXENA = 1, then D1 = 4P, D2 = 8P

□Extra delay from FSX high to DX valid **applies *only* to the first data bit of a device**, if and only if DXENA = 1 in SPCR.
if DXENA = 0, then D1 = D2 = 0
if DXENA = 1, then D1 = 4P, D2 = 8P



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

Figure 7−40. McBSP Timing

Table 7−28. Timing Requirements for FSR When GSYNC = 1 (see Figure 7−41)

NO.			−400 −500 MIN	−400 −500 MAX	UNIT
1	$t_{su(FRH\text{-}CKSH)}$	Setup time, FSR high before CLKS high	4		ns
2	$t_{h(CKSH\text{-}FRH)}$	Hold time, FSR high after CLKS high	4		ns

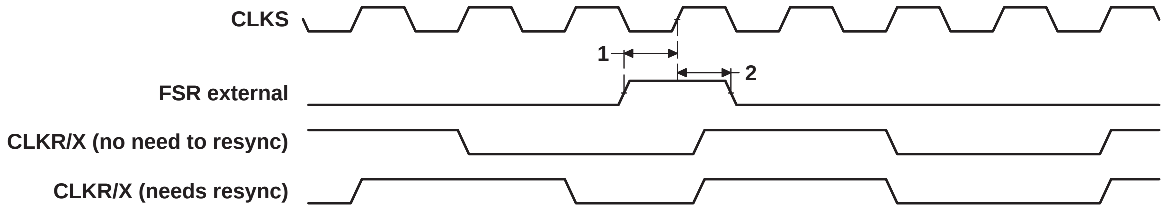


Figure 7−41. FSR Timing When GSYNC = 1

Table 7−29. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0†‡ (see Figure 7−42)

NO.			−400 −500				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXL)}$	Setup time, DR valid before CLKX low	12		2 − 12P		ns
5	$t_{h(CKXL-DRV)}$	Hold time, DR valid after CLKX low	4		5 + 24P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

Table 7−30. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0†‡ (see Figure 7−42)

NO.	PARAMETER		−400 −500				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low¶	T − 2	T + 3			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high#	L − 2	L + 3			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	−2	4	12P + 2.8	20P + 17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	L − 2	L + 3			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high			4P + 3	12P + 17	ns
8	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid			8P + 1.8	16P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
‡ For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.
§ S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
¶ FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
# FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

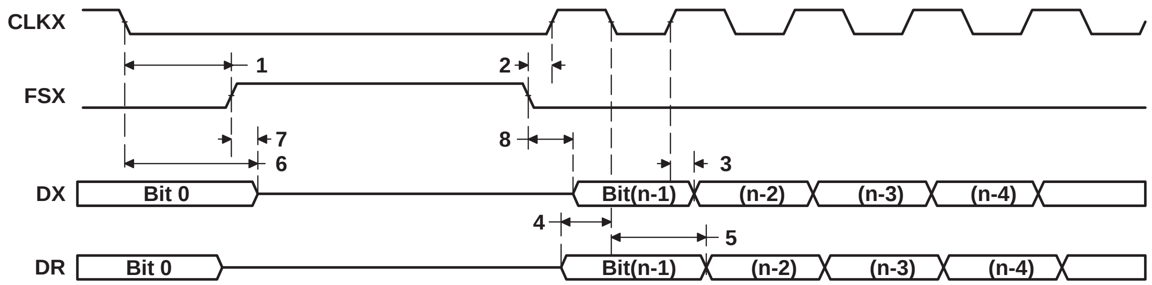


Figure 7−42. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

Table 7−31. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0[†][‡] (see Figure 7−43)

NO.			−400 −500 MASTER MIN	MASTER MAX	SLAVE MIN	SLAVE MAX	UNIT
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

Table 7−32. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0[†][‡] (see Figure 7−43)

NO.	PARAMETER		−400 −500 MASTER[§] MIN	MASTER[§] MAX	SLAVE MIN	SLAVE MAX	UNIT
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low[¶]	L − 2	L + 3			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high[#]	T − 2	T + 3			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	−2	4	12P + 2.8	20P + 17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	−2	4	12P + 3	20P + 17	ns
7	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	H − 2	H + 4	8P + 2	16P + 17	ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

[§] S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero

[¶] FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP

[#] FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

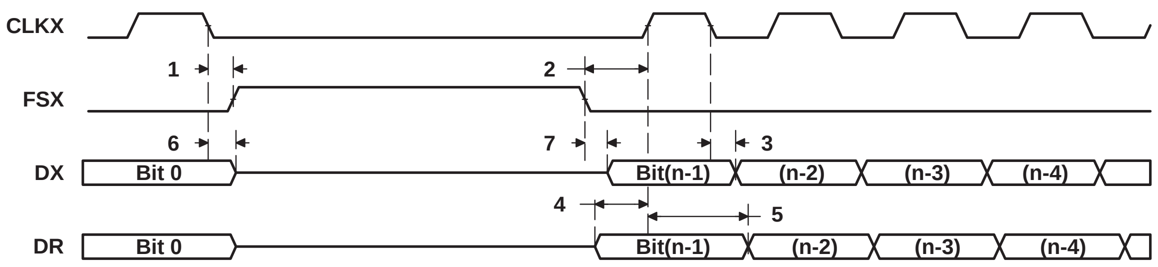


Figure 7−43. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0

Table 7−33. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1[†][‡] (see Figure 7−44)

NO.			−400 −500				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

Table 7−34. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1[†][‡] (see Figure 7−44)

NO.	PARAMETER		−400 −500				UNIT
			MASTER[§]		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high[¶]	T − 2	T + 3			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low[#]	H − 2	H + 3			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	−2	4	12P + 2.8	20P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	H − 2	H + 3			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high			4P + 3	12P + 17	ns
8	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid			8P + 2	16P + 17	ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

[§] S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero

[¶] FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP

[#] FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

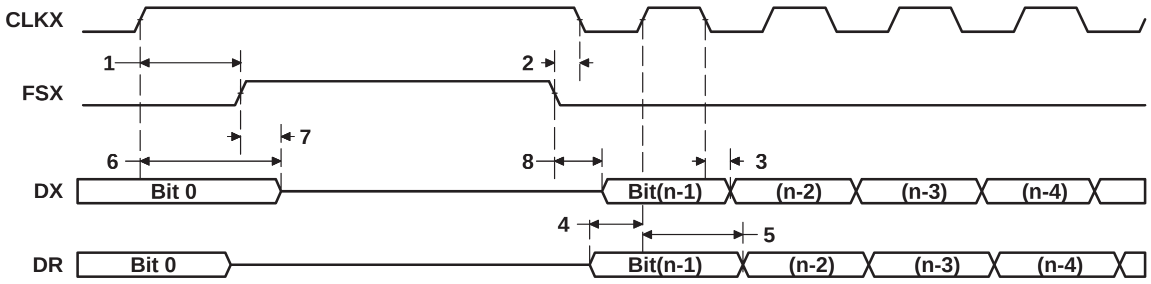


Figure 7−44. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

Table 7−35. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1[†][‡] (see Figure 7−45)

NO.			−400 −500				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	12		2 − 12P		ns
5	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	4		5 + 24P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

Table 7−36. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1[†][‡] (see Figure 7−45)

NO.	PARAMETER		−400 −500				UNIT
			MASTER[§]		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high[¶]	H − 2	H + 3			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low[#]	T − 2	T + 1			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	−2	4	12P + 2.8	20P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	−2	4	12P + 3	20P + 17	ns
7	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	L − 2	L + 4	8P + 2	16P + 17	ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] For all SPI Slave modes, CLKG is programmed as 1/4 of the CPU clock by setting CLKSM = CLKGDV = 1.

[§] S = Sample rate generator input clock = 4P if CLKSM = 1 (P = 1/CPU clock frequency)
= Sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
= (CLKGDV + 1)/2 * S if CLKGDV is odd or zero

[¶] FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP

[#] FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

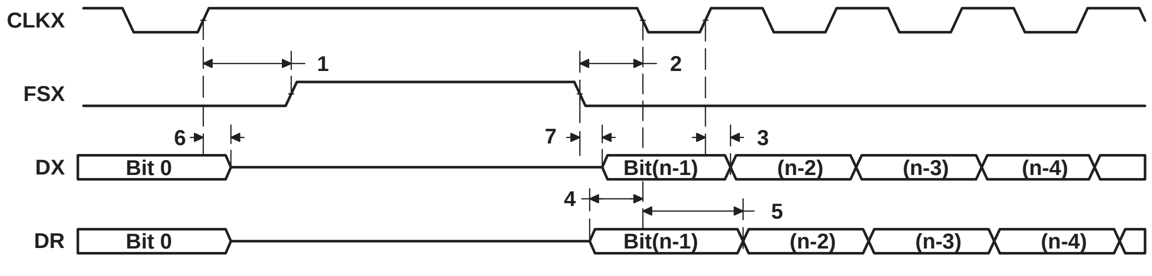


Figure 7−45. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

7.13 Timer Timing

Table 7−37. Timing Requirements for Timer Inputs[†] (see Figure 7−46)

NO.			−400 −500		UNIT
			MIN	MAX	
1	$t_{w(TINPH)}$	Pulse duration, TINP high	8P		ns
2	$t_{w(TINPL)}$	Pulse duration, TINP low	8P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

Table 7−38. Switching Characteristics Over Recommended Operating Conditions for Timer Outputs[†] (see Figure 7−46)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
3	$t_{w(TOUTH)}$	Pulse duration, TOUT high	8P − 3		ns
4	$t_{w(TOUTL)}$	Pulse duration, TOUT low	8P − 3		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

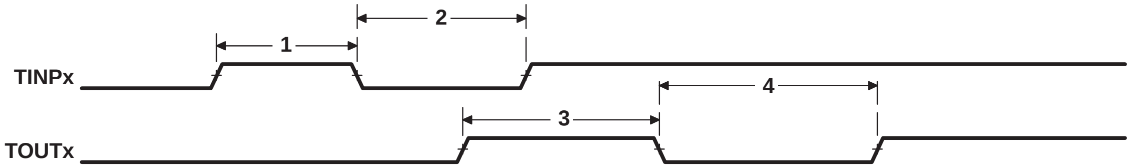


Figure 7−46. Timer Timing

7.14 General-Purpose Input/Output (GPIO) Port Timing

Table 7−39. Timing Requirements for GPIO Inputs[†][‡] (see Figure 7−47)

NO.			−400 −500		UNIT
			MIN	MAX	
1	$t_{w(GPIH)}$	Pulse duration, GPIx high	8P		ns
2	$t_{w(GPIL)}$	Pulse duration, GPIx low	8P		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[‡] The pulse width given is sufficient to generate a CPU interrupt or an EDMA event. However, if a user wants to have the DSP recognize the GPIx changes through software polling of the GPIO register, the GPIx duration must be extended to at least 12P to allow the DSP enough time to access the GPIO register through the CFGBUS.

Table 7−40. Switching Characteristics Over Recommended Operating Conditions for GPIO Outputs[†] (see Figure 7−47)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
3	$t_{w(GPOH)}$	Pulse duration, GPOx high	24P − 8[§]		ns
4	$t_{w(GPOL)}$	Pulse duration, GPOx low	24P − 8[§]		ns

[†] P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.

[§] This parameter value should not be used as a maximum performance specification. Actual performance of back-to-back accesses of the GPIO is dependent upon internal bus activity.

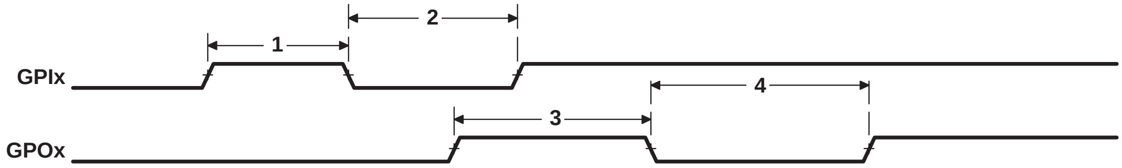


Figure 7−47. GPIO Port Timing

7.15 JTAG Test-Port Timing

Table 7−41. Timing Requirements for JTAG Test Port (see Figure 7−48)

NO.			−400 −500		UNIT
			MIN	MAX	
1	$t_{c(TCK)}$	Cycle time, TCK	35		ns
3	$t_{su(TDIV\text{-}TCKH)}$	Setup time, TDI/TMS/$\overline{\text{TRST}}$ valid before TCK high	10		ns
4	$t_{h(TCKH\text{-}TDIV)}$	Hold time, TDI/TMS/$\overline{\text{TRST}}$ valid after TCK high	9		ns

Table 7−42. Switching Characteristics Over Recommended Operating Conditions for JTAG Test Port (see Figure 7−48)

NO.	PARAMETER		−400 −500		UNIT
			MIN	MAX	
2	$t_{d(TCKL\text{-}TDOV)}$	Delay time, TCK low to TDO valid	0	18	ns

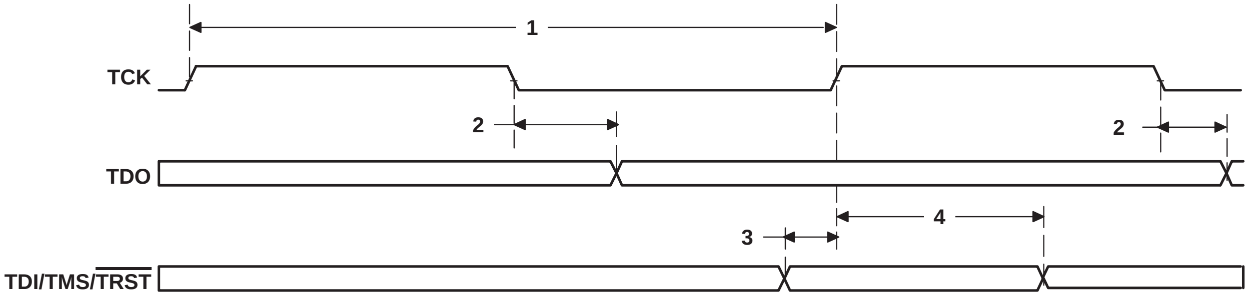


Figure 7−48. JTAG Test-Port Timing

8 Mechanical Data

8.1 Thermal Data

The following tables show the thermal resistance characteristics for the PBGA − GTS and ZTS mechanical packages.

Table 8−1. Thermal Resistance Characteristics (S-PBGA Package) [GTS]

NO.			°C/W	Board Type[†]	Air Flow (m/s[‡])
1	$R\Theta_{JC}$	Junction-to-case	5.60	JEDEC Low-K Test Card	N/A
2	$R\Theta_{JB}$	Junction-to-board	9.37	JEDEC High-K Test Card	N/A
3	$R\Theta_{JA}$	Junction-to-free air	20.8	JEDEC High-K Test Card	0.00
4			16.8	JEDEC High-K Test Card	0.5
5			15.4	JEDEC High-K Test Card	1.0
6			14.1	JEDEC High-K Test Card	2.00
7	Psi_{JT}	Junction-to-package top	1.87	JEDEC High-K Test Card	0.00
			1.98	JEDEC High-K Test Card	0.5
			2.03	JEDEC High-K Test Card	1.0
			2.12	JEDEC High-K Test Card	2.00
8	Psi_{JB}	Junction-to-board	11.1	JEDEC High-K Test Card	0.00
			10.4	JEDEC High-K Test Card	0.5
			10.3	JEDEC High-K Test Card	1.0
			10.1	JEDEC High-K Test Card	2.00

[†] Board types are as defined by JEDEC. Reference JEDEC Standard JESD51−9. Test Boards for Area Array Surface Mount Package Thermal Measurements.

[‡] m/s = meters per second

Table 8−2. Thermal Resistance Characteristics (S-PBGA Package) [ZTS]

NO.			°C/W	Board Type[†]	Air Flow (m/s[‡])
1	$R\Theta_{JC}$	Junction-to-case	5.60	JEDEC Low-K Test Card	N/A
2	$R\Theta_{JB}$	Junction-to-board	9.37	JEDEC High-K Test Card	N/A
3	$R\Theta_{JA}$	Junction-to-free air	20.8	JEDEC High-K Test Card	0.00
4			16.8	JEDEC High-K Test Card	0.5
5			15.4	JEDEC High-K Test Card	1.0
6			14.1	JEDEC High-K Test Card	2.00
7	Psi_{JT}	Junction-to-package top	1.87	JEDEC High-K Test Card	0.00
			1.98	JEDEC High-K Test Card	0.5
			2.03	JEDEC High-K Test Card	1.0
			2.12	JEDEC High-K Test Card	2.00
8	Psi_{JB}	Junction-to-board	11.1	JEDEC High-K Test Card	0.00
			10.4	JEDEC High-K Test Card	0.5
			10.3	JEDEC High-K Test Card	1.0
			10.1	JEDEC High-K Test Card	2.00

[†] Board types are as defined by JEDEC. Reference JEDEC Standard JESD51−9. Test Boards for Area Array Surface Mount Package Thermal Measurements.

[‡] m/s = meters per second

8.2 Packaging Information

The following packaging information reflects 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.

PACKAGE OPTION ADDENDUM

PACKAGING INFORMATION

Orderable Device	Status [1]	Package Type	Package Drawing	Pins	Package Qty	Eco Plan [2]	Lead/ Ball Finish	MSL Peak Temp [3]	Samples (Requires Login)
TMS320C6410GTS400	ACTIVE	FCBGA	GTS	288	60	TBD	SNPB	Level-4-220C-72 HR	
TMS320C6410ZTS400	ACTIVE	FCBGA	ZTS	288	60	Pb-Free (RoHS Exempt)	SNAGCU	Level-4-260C-72HR	
TMS320C6410ZTSA400	ACTIVE	FCBGA	ZTS	288	60	Pb-Free (RoHS Exempt)	SNAGCU	Level-4-260C-72HR	
TMS320C6413GTS500	ACTIVE	FCBGA	GTS	288	60	TBD	SNPB	Level-4-220C-72 HR	
TMS320C6413GTSA500	ACTIVE	FCBGA	GTS	288	60	TBD	SNPB	Level-4-220C-72 HR	
TMS320C6413ZTS500	ACTIVE	FCBGA	ZTS	288	60	Pb-Free (RoHS Exempt)	SNAGCU	Level-4-260C-72HR	
TMS320C6413ZTSA500	ACTIVE	FCBGA	ZTS	288	60	Pb-Free (RoHS Exempt)	SNAGCU	Level-4-260C-72HR	
TMX320C6410GTS	OBSOLETE	FCBGA	GTS	288		TBD	Call TI	Call TI	

[1] The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

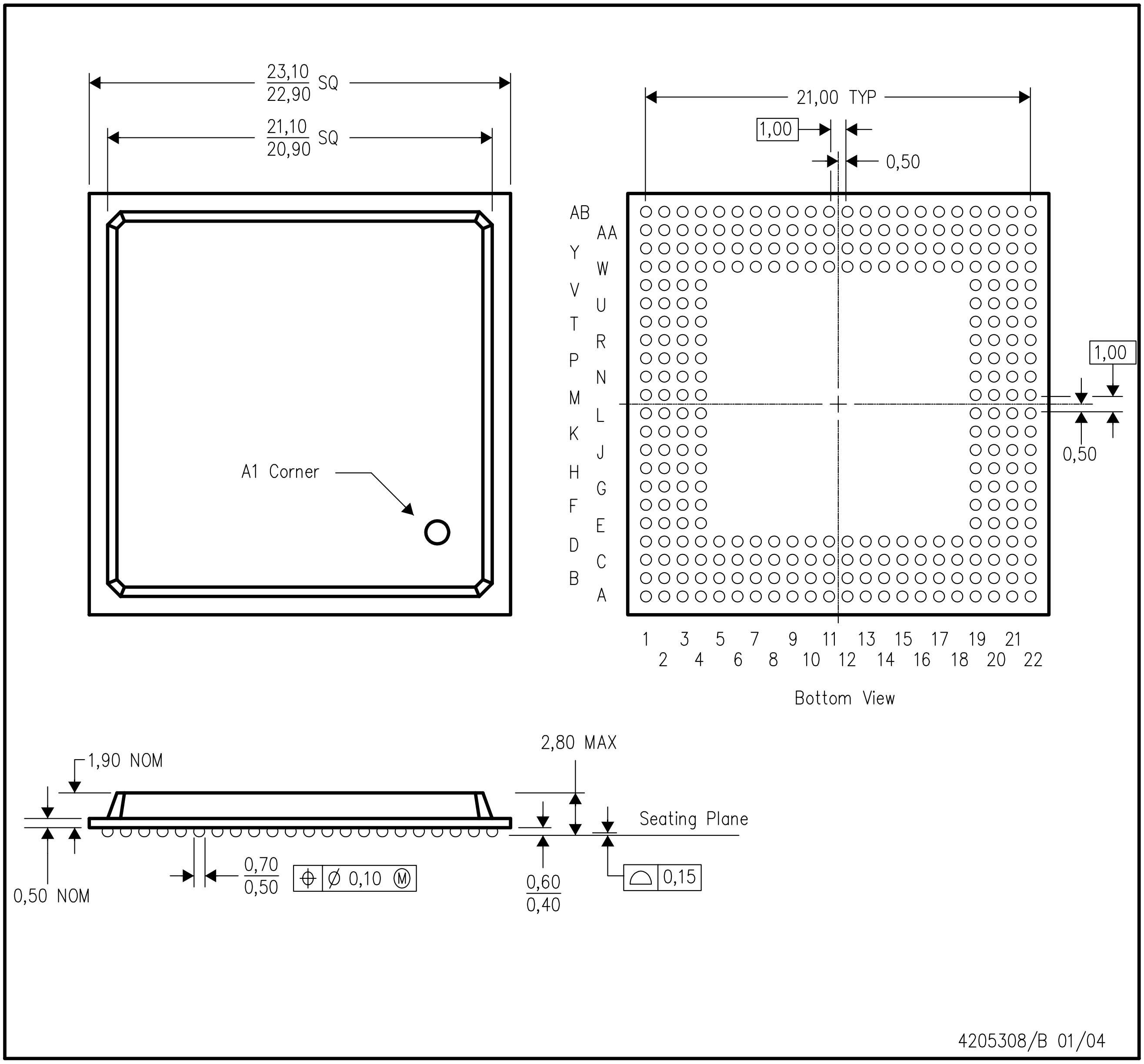
[2] Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
TBD: The Pb-Free/Green conversion plan has not been defined.
Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

[3] MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

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.

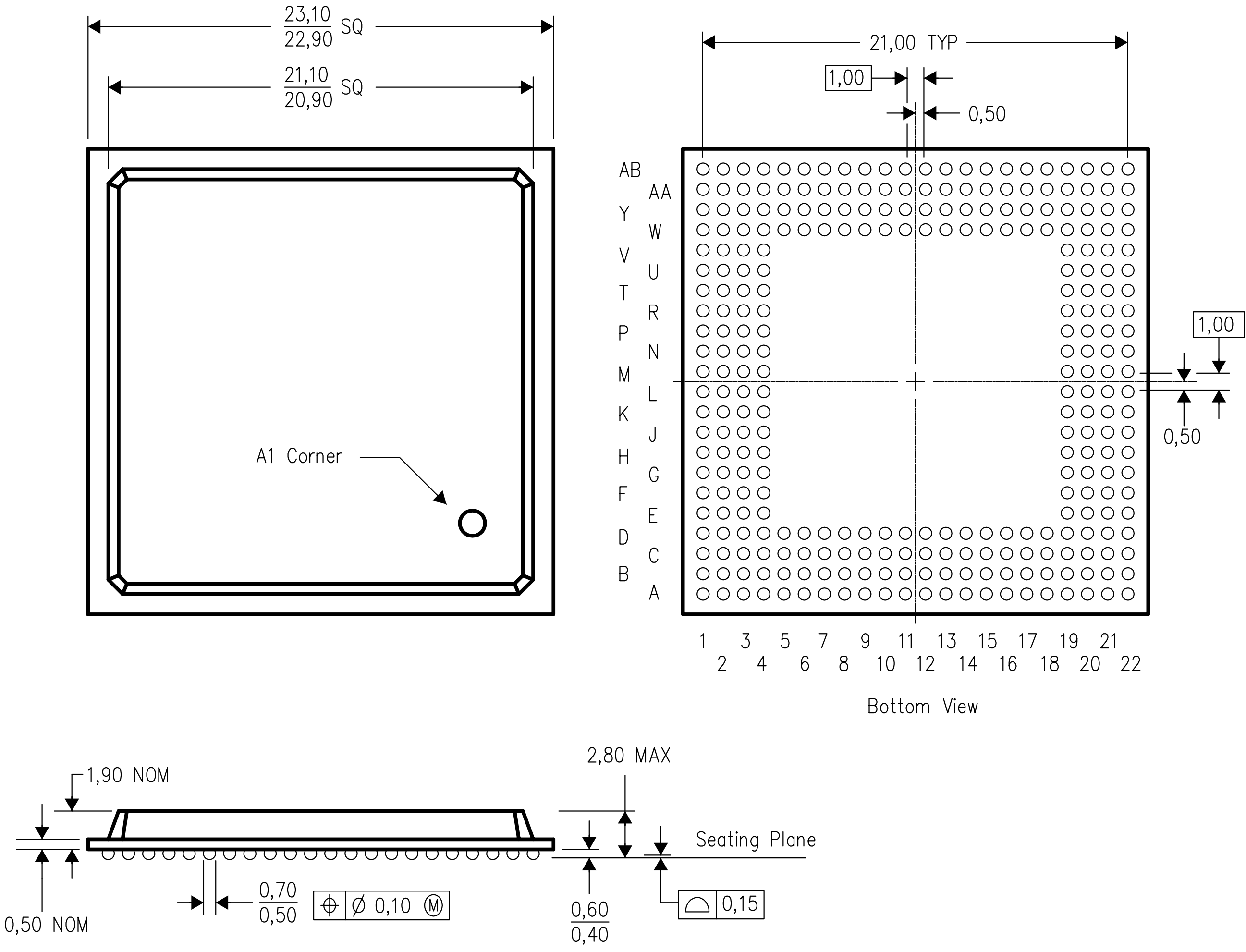
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

GTS (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. Flip chip application only.
- D. Falls within JEDEC MS–034B.

ZTS (S–PBGA–N288) PLASTIC BALL GRID ARRAY



4206098/A 05/04

NOTES:
- A. All linear dimensions are in millimeters.
- B. This drawing is subject to change without notice.
- C. Flip chip application only.
- D. Falls within JEDEC MS–034B.
- E. This package is lead–free.

IMPORTANT NOTICE

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

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

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

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

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

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

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

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

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

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

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

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265