THE DATASHEET OF

SM32C6414DGLZ50AEP

1 Introduction

1.1 Features

- **Highest-Performance Fixed-Point Digital Signal Processors (DSPs)**
  - **2-ns Instruction Cycle Time**
  - **500-MHz Clock Rate**
  - **Eight 32-Bit Instructions/Cycle**
  - **28 Operations/Cycle**
  - **4000 MIPS**
  - **Fully Software Compatible With C62x™**
  - **C6414/15/16 Devices Pin Compatible**
- **VelociTI.2™ Extensions to VelociTI™ Advanced Very Long Instruction Word (VLIW) TMS320C64x™ DSP Core**
  - **Eight Highly Independent Functional Units With VelociTI.2 Extensions With Six ALUs and Two Multipliers**
  - **Nonaligned Load-Store Architecture**
  - **64 32-Bit General-Purpose Registers**
  - **Instruction Packing Reduces Code Size**
  - **All Instructions Conditional**
- **Instruction Set Features**
  - **Byte-Addressable (8-/16-/32-/64-Bit Data)**
  - **8-Bit Overflow Protection**
  - **Bit-Field Extract, Set, Clear**
  - **Normalization, Saturation, Bit Counting**
  - **VelociTI.2 Increased Orthogonality**
- **Viterbi Decoder Coprocessor (VCP) (C6416)**
  - **Supports Over 500 7.95-Kbps Adaptive Multi-Rate (AMR)**
  - **Programmable Code Parameters**
- **Turbo Decoder Coprocessor (TCP) (C6416)**
  - **Supports up to Six 2-Mbps 3GPP (Six Iterations)**
  - **Programmable Turbo Code and Decoding Parameters**
- **L1/L2 Memory Architecture**
  - **128K-Bit (16K-Byte) L1P Program Cache**
  - **128K-Bit (16K-Byte) L1D Data Cache**
  - **8M-Bit (1024K-Byte) L2 Unified Mapped RAM/Cache**
- **Two External Memory Interfaces (EMIFs) for 1280M-Byte Addressable External Memory**
- **Enhanced Direct Memory Access (EDMA) Controller (64 Independent Channels)**
- **Host-Port Interface (HPI)**
  - **User-Configurable Bus Width (32/16 Bit)**
- **32-Bit/33-MHz, 3.3-V PCI Master/Slave Interface Conforms to PCI Specification 2.2 (C6415/C6416)**
  - **Three PCI Bus Address Registers**
  - **Four-Wire Serial EEPROM Interface**
  - **PCI Interrupt Request Under DSP Program Control**
  - **DSP Interrupt Via PCI I/O Cycle**
- **Three Multichannel Buffered Serial Ports (McBSPs)**
  - **Direct Interface to T1/E1, MVIP, and SCSA Framers**
  - **Up to 256 Channels Each**
  - **ST Bus Switching, AC97 Compatible**
  - **Serial Peripheral Interface (SPI) Compatible (Motorola)**
- **Three 32-Bit General-Purpose Timers**
- **Universal Test and Operations Physical Layer (PHY) Interface for ATM (UTOPIA) (C6415/C6416)**
  - **UTOPIA Level-2 Slave ATM Controller**
  - **8-Bit Transmit and Receive Operations up to 50 MHz per Direction**
  - **User-Defined Cell Format up to 64 Bytes**
- **16 General-Purpose I/O (GPIO) Pins**
- **Flexible Phase-Locked Loop (PLL) Clock Generator**
- **IEEE Std 1149.1 (JTAG(1)) Boundary Scan Compatible**
- **532-Pin Ball Grid Array (BGA) Package (GLZ Suffix), 0.8-mm Ball Pitch**
- **0.13-μm/6-Level Metal Process (CMOS)**
- **3.3-V I/Os, 1.25-V Internal (500 MHz)**

(1) IEEE Std 1149.1-1990 Standard Test-Access Port and Boundary Scan Architecture

1.2 SUPPORTS DEFENSE, AEROSPACE, AND MEDICAL APPLICATIONS

- **Controlled Baseline**
- **One Assembly/Test Site**
- **One Fabrication Site**
- **Available in A-Version (–40°C/105°C) and S-Version (–55°C/105°C) Temperature Ranges**(2)
- **Extended Product Life Cycle**
- **Extended Product-Change Notification**
- **Product Traceability**

(2) S-Version currently available for C6415 only. Additional custom temperature ranges available upon request.

1.3 Description

The TMS320C64x™ DSPs (including the SM320C6414-EP, SM320C6415-EP, and SM320C6416-EP devices) are the highest-performance fixed-point DSP generation in the TMS320C6000™ DSP platform. The SM320C64x™ (C64x™) device is based on the second-generation, high-performance, advanced VelociTI™ very-long-instruction word (VLIW) architecture (VelociTI.2™) developed by Texas Instruments (TI), making these DSPs an excellent choice for multichannel and multifunctional applications. 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 C64x devices offer cost-effective solutions to high-performance DSP programming challenges. The C64x DSPs possess 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 – 2 multipliers for a 32-bit result and 6 arithmetic logic units (ALUs) – with VelociTI.2 extensions. The VelociTI.2 extensions in the eight functional units include new instructions to accelerate the performance in key applications and extend the parallelism of the VelociTI architecture. The C64x can produce four 32-bit multiply-accumulates (MACs) per cycle for a total of 2400 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 4800 MMACS. The C64x DSP also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000 DSP platform devices.

The C6416 device has two high-performance embedded coprocessors [Viterbi decoder coprocessor (VCP) and turbo decoder coprocessor (TCP)] that significantly speed up channel-decoding operations on chip. The VCP operating at CPU clock divided-by-4 can decode over 500 7.95-Kbps adaptive multi-rate (AMR) (K = 9, R = 1/3) voice channels. The VCP supports constraint lengths K = 5, 6, 7, 8, and 9, rates R = 1/2, 1/3, and 1/4, and flexible polynomials, while generating hard decisions or soft decisions. The TCP operating at CPU clock divided-by-2 can decode up to 36 384-Kbps or 6 2-Mbps turbo encoded channels (assuming iterations). The TCP implements the max*log-map algorithm and is designed to support all polynomials and rates required by Third-Generation Partnership Projects (3GPP and 3GPP2), with fully programmable frame length and turbo interleaver. Decoding parameters, such as the number of iterations and stopping criteria, are also programmable. Communications between the VCP/TCP and the CPU are carried out through the EDMA controller.

The C64x uses a two-level cache-based architecture and has a powerful and diverse set of peripherals. The level 1 program (L1P) cache is a 128K-bit direct-mapped cache and the level 1 data (L1D) cache is a 128K-bit 2-way set-associative cache. The level 2 memory/cache (L2) consists of an 8M-bit memory space that is shared between program and data space. L2 memory can be configured as mapped memory or combinations of cache (up to 256K bytes) and mapped memory. The peripheral set includes 3 multichannel buffered serial ports (McBSPs), an 8-bit universal test and operations PHY interface for asynchronous transfer mode (ATM) slave (UTOPIA slave) port (C6415/C6416 only), 3 32-bit general-purpose timers, a user-configurable 16-bit or 32-bit host-port interface (HPI16/HPI32), a peripheral component interconnect (PCI) (C6415/C6416 only), a general-purpose input/output port (GPIO) with 16 GPIO pins, and two glueless external memory interfaces (64-bit EMIFA and 16-bit EMIFB), both of which are capable of interfacing to synchronous and asynchronous memories and peripherals.

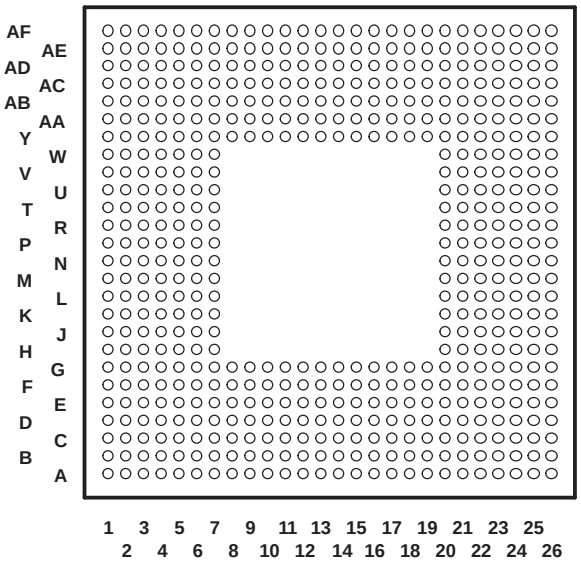
The C64x has a complete set of development tools that includes an advanced C compiler with C64x-specific enhancements, an assembly optimizer to simplify programming and scheduling, and a Windows™ debugger interface for visibility into source code execution.(3)(4)

(3) Throughout the remainder of this document, the SM320C6414-EP, SM320C6415-EP, and SM320C6416-EP are referred to as SM320C64x or C64x where generic and, where specific, their individual full device part numbers are used or abbreviated as C6414, C6415, or C6416, respectively.

(4) These C64x devices have two EMIFs (64-bit EMIFA and 16-bit EMIFB). The prefix "A" in front of a signal name indicates it is an EMIFA signal whereas a prefix "B" in front of a signal name indicates it is an EMIFB signal. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix "A" or "B" may be omitted from the signal name.

1.4 Ball-Grid Array (BGA) Package

GLZ 532-PIN BALL GRID ARRAY (BGA) PACKAGE
(BOTTOM VIEW)



1.4.1 Device Characteristics

Table 1-1 provides an overview of the C6414, C6415, and C6416 DSPs. Table 1-1 shows significant features of the C64x devices, including the capacity of on-chip RAM, the peripherals, the CPU frequency, and the package type with pin count.

Table 1-1. Characteristics of the C6414, C6415, and C6416 Processors

HARDWARE FEATURES		C6414, C6415, AND C6416
Peripherals Not all peripherals pins are available at the same time. (For more details, see the Device Configuration section.) Peripheral performance is dependent on chip-level configuration.	EMIFA (64-bit bus width) (default clock source = AECLKIN)	1
	EMIFB (16-bit bus width) (default clock source = BECLKIN)	1
	EDMA (64 independent channels)	1
	HPI (32- or 16-bit user selectable)	1 (HPI16 or HPI32)
	PCI (32-bit) [DeviceID Register value 0xA106]	1 (**C6415/C6416 only**)
	McBSPs (default internal clock source = CPU/4 clock frequency)	3
	UTOPIA (8-bit mode)	1 (**C6415/C6416 only**)
	32-bit timers (default internal clock source = CPU/8 clock frequency)	3
	General-purpose input/output 0 (GP0)	16
Decoder coprocessors	VCP	1 (**C6416 only**)
	TCP	1 (**C6416 only**)
On-chip memory	Size (bytes)	1056K
	Organization	16K-byte (16KB) L1 program (L1P) cache 16KB L1 data (L1D) cache 1024KB unified mapped RAM/cache (L2)
CPU ID + CPU Rev ID	Control Status Register (CSR[31:16])	0x0C01
Device_ID	Silicon Revision Identification Register (DEVICE_REV[19:16]) Address: 0x01B0 0200	DEVICE_REV[19:16] Silicon revision 1111 1.03 or earlier 0001 1.03 0010 1.1
Frequency	MHz	500
Cycle time	ns	2 ns (C6414-50A, C6415-50A, C6416-50A) (500-MHz CPU, 100-MHz EMIF)(1)
Voltage	Core (V)	1.25 V (-50A)
	I/O (V)	3.3 V
PLL options	CLKIN frequency multiplier	Bypass (x1), x6, x12
BGA package	23 mm × 23 mm	532-pin BGA (GLZ)
Process technology	CMOS	0.3 μm
Product status	Product Preview (PP) Advance Information (AI) Production Data (PD)	PD

(1) On these C64x devices, the rated EMIF speed affects only the SDRAM interface on EMIFA. For more detailed information, see the EMIF Device Speed section of this data manual.

1.4.2 Device Compatiblity

The C64x generation of devices has a diverse and powerful set of peripherals. The common peripheral set and pin compatibility that the C6414, C6415, and C6416 devices offer lead to easier system designs and faster time to market. Table 1-2 identifies the peripherals and coprocessors that are available on the C6414, C6415, and C6416 devices.

The C6414, C6415, and C6416 devices are pin-for-pin compatible, provided the following conditions are met:

- All devices use the same peripherals.
  - The C6414 is pin-for-pin compatible with the C6415/C6416 when the PCI and UTOPIA peripherals on the C6415/C6416 are disabled.
  - The C6415 is pin-for-pin compatible with the C6416 when they are in the same peripheral selection mode. For more information on peripheral selection, see the Device Configurations section of this data manual.
- The BEA[9:7] pins are properly pulled up/down.
  - For more details on the device-specific BEA[9:7] pin configurations, see the Terminal Functions table of this data manual.

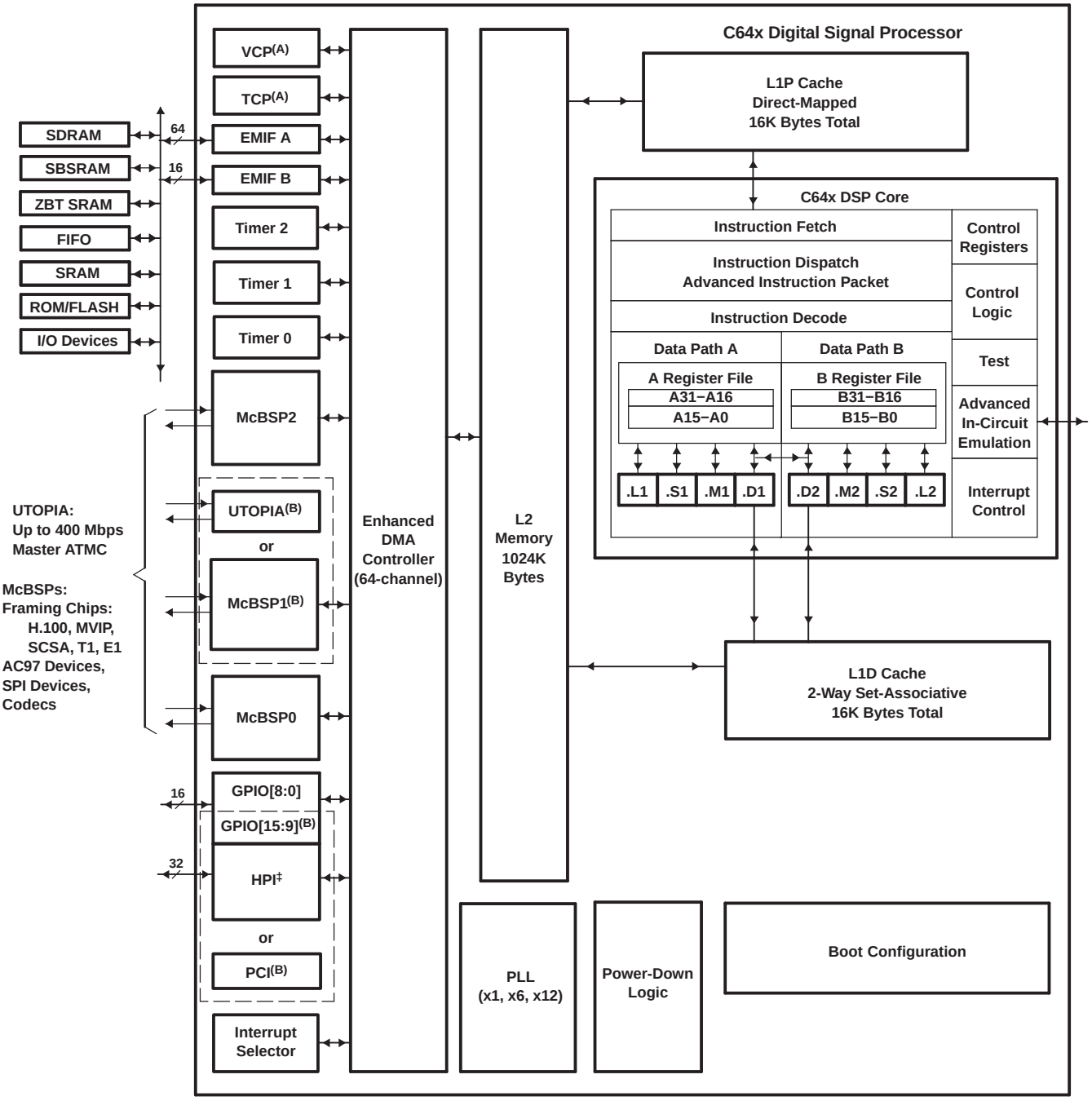
Table 1-2. Peripherals and Coprocessors Available on C6414, C6415, and C6416 Devices(1) (2)

PERIPHERALS/COPROCESSORS	C6414	C6415	C6416
EMIFA (64-bit bus width)	√	√	√
EMIFB (16-bit bus width)	√	√	√
EDMA (64 independent channels)	√	√	√
HPI (32- or 16-bit user selectable)	√	√	√
PCI (32 bit) (specification v2.2)	—	√	√
McBSPs (McBSP0, McBSP1, McBSP2)	√	√	√
UTOPIA (8-bit mode) (specification v1.0)	—	√	√
Timers (32 bit) (TIMER0, TIMER1, TIMER2)	√	√	√
GPIOs (GP[15:0])	√	√	√
VCP/TCP coprocessors	—	—	√

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

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

1.4.3 Functional Block and CPU (DSP Core) Diagram



A. VCP and TCP decoder coprocessors are applicable to the C6416 device only.

B. For the C6415 and C6416 devices, the UTOPIA peripheral is multiplexed with McBSP1, and the PCI peripheral is multiplexed with the HPI peripheral and the GPIO[15:9] port. For more details on the multiplexed pins of these peripherals, see the Device Configurations section of this data manual.

1.4.4 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 1-1]. The four functional units on each side of the CPU can freely share the 32 registers belonging to that side. Additionally, each side features a data cross path – a single data bus connected to all the registers on the other side, by which the two sets of functional units can access data from the register files on the opposite side. The C64x CPU pipelines data-cross-path accesses over multiple clock cycles. This allows the same register to be used as a data-cross-path operand by multiple functional units in the same execute packet. All functional units in the C64x CPU can access operands via the data cross path. Register access by functional units on the same side of the CPU as the register file can service all the units in a single clock cycle. On the C64x CPU, a delay clock is introduced when 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 nonaligned 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).

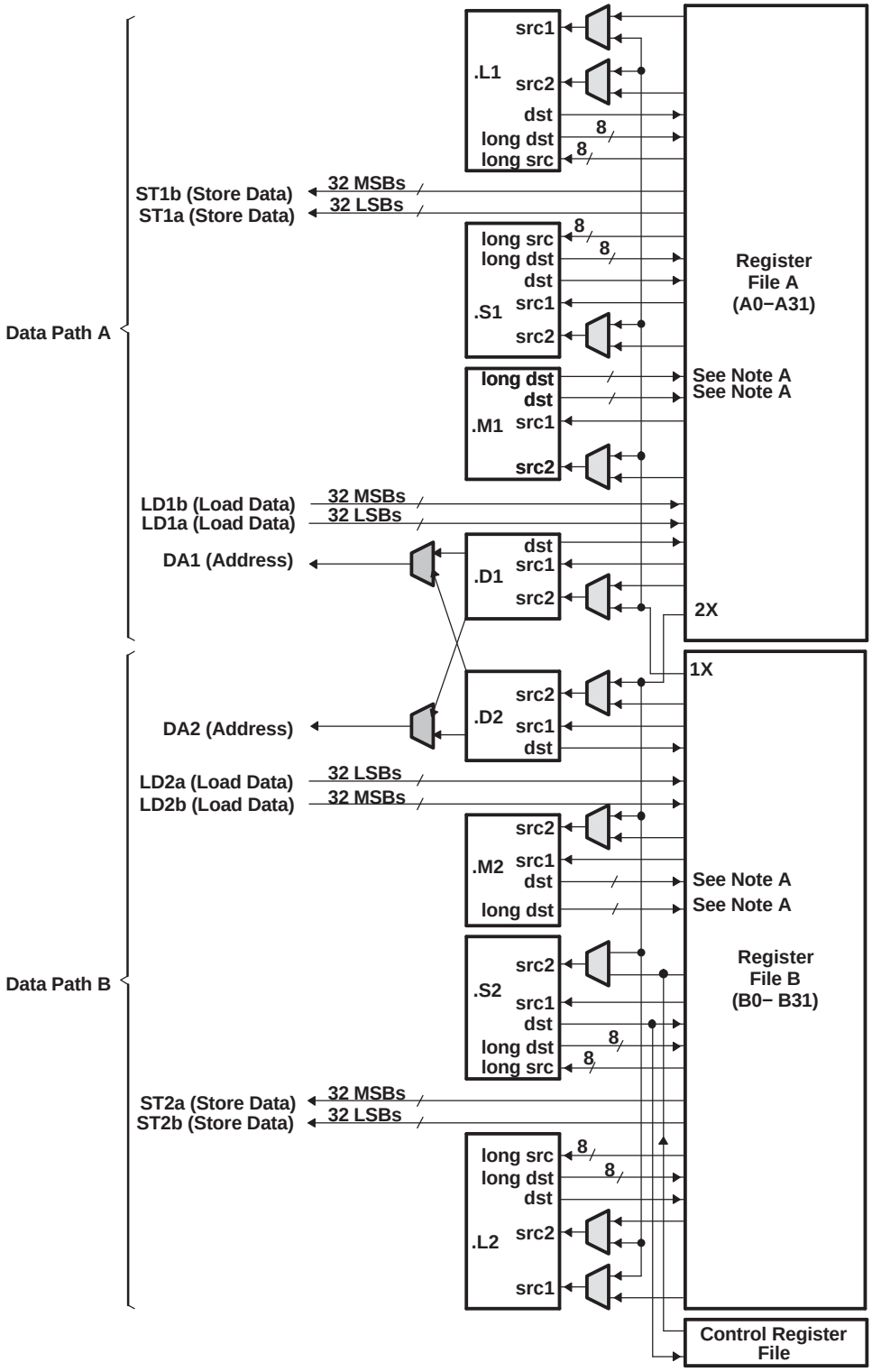
The two .M functional units perform all multiplication operations. Each of the C64x .M units can perform two 16-bit × 16-bit multiplies or four 8-bit × 8-bit multiplies per clock cycle. The .M unit can also perform 16-bit × 32-bit multiply operations, dual 16-bit × 16-bit multiplies with add/subtract operations, and quad 8-bit × 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 no-operation (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, halfwords, words, or doublewords. All load and store instructions are byte, halfword, 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)
- *How to Begin Development Today With the TMS320C6414, TMS320C6415, and TMS320C6416 DSPs* application report (literature number SPRA718)



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

Figure 1-1. SM320C64x™ CPU (DSP Core) Data Paths

1.4.5 Memory Map Summary

Table 1-3 shows the memory map address ranges of the SM320C64x device. 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 C64x device begin at the hex address locations 0x6000 0000 for EMIFB and 0x8000 0000 for EMIFA.

Table 1-3. SM320C64x Memory Map Summary

MEMORY BLOCK DESCRIPTION	BLOCK SIZE (BYTES)	HEX ADDRESS RANGE
Internal RAM (L2)	1M	0000 0000–000F FFFF
Reserved	23M	0010 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
McBSP 2 Registers	256K	01A4 0000–01A7 FFFF
EMIFB Registers	256K	01A8 0000–01AB FFFF
Timer 2 Registers	256K	01AC 0000–01AF FFFF
GPIO Registers	256K	01B0 0000–01B3 FFFF
UTOPIA Registers (C6415 and C6416 only)(1)	256K	01B4 0000–01B7 FFFF
TCP/VCP Registers (C6416 only)(2)	256K	01B8 0000–01BB FFFF
Reserved	256K	01BC 0000–01BF FFFF
PCI Registers (C6415 and C6416 only)(1)	256K	01C0 0000–01C3 FFFF
Reserved	4M–256K	01C4 0000–01FF FFFF
QDMA Registers	52	0200 0000–0200 0033
Reserved	736M–52	0200 0034–2FFF FFFF
McBSP 0 Data	64M	3000 0000–33FF FFFF
McBSP 1 Data	64M	3400 0000–37FF FFFF
McBSP 2 Data	64M	3800 0000–3BFF FFFF
UTOPIA Queues (C6415 and C6416 only)(1)	64M	3C00 0000–3FFF FFFF
Reserved	256K	4000 0000–4FFF FFFF
TCP/VCP (C6416 only)(2)	256K	5000 0000–5FFF FFFF
EMIFB CE0	64M	6000 0000–63FF FFFF
EMIFB CE1	64M	6400 0000–67FF FFFF
EMIFB CE2	64M	6800 0000–6BFF FFFF
EMIFB CE3	64M	6C00 0000–6FFF FFFF
Reserved	256K	7000 0000–7FFF FFFF
EMIFA CE0	256K	8000 0000–8FFF FFFF
EMIFA CE1	256K	9000 0000–9FFF FFFF
EMIFA CE2	256K	A000 0000–AFFF FFFF
EMIFA CE3	256K	B000 0000–BFFF FFFF
Reserved	1G	C000 0000–FFFF FFFF

(1) For the C6414 device, these memory address locations are reserved. The C6414 device does not support the UTOPIA and PCI peripherals.
(2) Only the C6416 device supports the VCP/TCP coprocessors. For the C6414 and C6415 devices, these memory address locations are reserved.

1.4.6 Peripheral Register Descriptions

Table 1-4 through Table 1-24 identify the peripheral registers for the C6414, C6415, and C6416 devices by their register names, acronyms, and hex address or hex address range. For more detailed information on the register contents and bit names and their descriptions, see the *TMS320C6000 Peripherals Reference Guide* (literature number SPRU190).

Table 1-4. EMIFA Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
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 1-5. EMIFB Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
01A8 0000	GBLCTL	EMIFB Global Control
01A8 0004	CECTL1	EMIFB CE1 Space Control
01A8 0008	CECTL0	EMIFB CE0 Space Control
01A8 000C	–	Reserved
01A8 0010	CECTL2	EMIFB CE2 Space Control
01A8 0014	CECTL3	EMIFB CE3 Space Control
01A8 0018	SDCTL	EMIFB SDRAM Control
01A8 001C	SDTIM	EMIFB SDRAM Refresh Control
01A8 0020	SDEXT	EMIFB SDRAM Extension
01A8 0024–01A8 003C	–	Reserved
01A8 0040	PDTCTL	Peripheral Device Transfer (PDT) Control
01A8 0044	CESEC1	EMIFB CE1 Space Secondary Control
01A8 0048	CESEC0	EMIFB CE0 Space Secondary Control
01A8 004C	–	Reserved
01A8 0050	CESEC2	EMIFB CE2 Space Secondary Control
01A8 0054	CESEC3	EMIFB CE3 Space Secondary Control
01A8 0058–01AB FFFF	–	Reserved

Table 1-6. L2 Cache Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
0184 0000	CCFG	Cache Configuration
0184 0004–0184 0FFC	–	Reserved
0184 1000	EDMAWEIGHT	L2 EDMA Access Control
0184 1004–0184 1FFC	–	Reserved
0184 2000	L2ALLOC0	L2 Allocation 0
0184 2004	L2ALLOC1	L2 Allocation 1
0184 2008	L2ALLOC2	L2 Allocation 2
0184 200C	L2ALLOC3	L2 Allocation 3
0184 2010–0184 3FFC	–	Reserved
0184 4000	L2FBAR	L2 Flush Base Address Register
0184 4004	L2FWC	L2 Flush Word Count
0184 4010	L2CBAR	L2 Clean Base Address Register
0184 4014	L2CWC	L2 Clean Word Count
0184 4020	L1PFBAR	L1P Flush Base Address Register
0184 4024	L1PFWC	L1P Flush Word Count
0184 4030	L1DFBAR	L1D Flush Base Address Register
0184 4034	L1DFWC	L1D Flush Word Count
0184 4038–0184 4FFC	–	Reserved
0184 5000	L2FLUSH	L2 Flush
0184 5004	L2CLEAN	L2 Clean
0184 5008–0184 7FFC	–	Reserved
0184 8000–0184 817C	MAR0 to MAR95	Reserved
0184 8180	MAR96	Controls EMIFB CE0 range 6000 0000–60FF FFFF
0184 8184	MAR97	Controls EMIFB CE0 range 6100 0000–61FF FFFF
0184 8188	MAR98	Controls EMIFB CE0 range 6200 0000–62FF FFFF
0184 818C	MAR99	Controls EMIFB CE0 range 6300 0000–63FF FFFF
0184 8190	MAR100	Controls EMIFB CE1 range 6400 0000–64FF FFFF
0184 8194	MAR101	Controls EMIFB CE1 range 6500 0000–65FF FFFF
0184 8198	MAR102	Controls EMIFB CE1 range 6600 0000–66FF FFFF
0184 819C	MAR103	Controls EMIFB CE1 range 6700 0000–67FF FFFF
0184 81A0	MAR104	Controls EMIFB CE2 range 6800 0000–68FF FFFF
0184 81A4	MAR105	Controls EMIFB CE2 range 6900 0000–69FF FFFF
0184 81A8	MAR106	Controls EMIFB CE2 range 6A00 0000–6AFF FFFF
0184 81AC	MAR107	Controls EMIFB CE2 range 6B00 0000–6BFF FFFF
0184 81B0	MAR108	Controls EMIFB CE3 range 6C00 0000–6CFF FFFF
0184 81B4	MAR109	Controls EMIFB CE3 range 6D00 0000–6DFF FFFF
0184 81B8	MAR110	Controls EMIFB CE3 range 6E00 0000–6EFF FFFF
0184 81BC	MAR111	Controls EMIFB CE3 range 6F00 0000–6FFF FFFF
0184 81C0–0184 81FC	MAR112 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

Table 1-6. L2 Cache Registers (continued)

HEX ADDRESS	ACRONYM	REGISTER NAME
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
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 B000–AFFF FFFF
0184 82A0	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

Table 1-6. L2 Cache Registers (continued)

HEX ADDRESS	ACRONYM	REGISTER NAME
0184 82D8	MAR182	Controls EMIFA CE3 range B600 0000–B6FF FFFF
0184 82DC	MAR183	Controls EMIFA CE3 range B700 B000–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 B000–BDFF FFFF
0184 8300–0184 83FC	MAR192 to MAR255	Reserved
0184 8400–0187 FFFF	–	Reserved

Table 1-7. EDMA Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
01A0 FF9C	EPRH	Event Polarity High Register
01A0 FFA4	CIPRH	Channel Interrupt Pending High Register
01A0 FFA8	CIERH	Channel Interrupt Enable High Register
01A0 FFAC	CCERH	Channel Chain Enable High Register
01A0 FFB0	ERH	Event High Register
01A0 FFB4	EERH	Event Enable High Register
01A0 FFB8	ECRH	Event Clear High Register
01A0 FFBC	ESRH	Event Set High Register
01A0 FFC0	PQAR0	Priority Queue Allocation Register 0
01A0 FFC4	PQAR1	Priority Queue Allocation Register 1
01A0 FFC8	PQAR2	Priority Queue Allocation Register 2
01A0 FFCC	PQAR3	Priority Queue Allocation Register 3
01A0 FFDC	EPRL	Event Polarity Low Register
01A0 FFE0	PQSR	Priority Queue Status Register
01A0 FFE4	CIPRL	Channel Interrupt Pending Low Register
01A0 FFE8	CIERL	Channel Interrupt Enable Low Register
01A0 FFEC	CCERL	Channel Chain Enable Low Register
01A0 FFF0	ERL	Event Low Register
01A0 FFF4	EERL	Event Enable Low Register
01A0 FFF8	ECRL	Event Clear Low Register
01A0 FFFC	ESRL	Event Set Low Register
01A1 0000–01A3 FFFF	–	Reserved

Table 1-8. EDMA Parameter RAM(1)

HEX ADDRESS	REGISTER NAME
01A0 0000–01A0 0017	Parameters for Event 0 (6 words)
01A0 0018–01A0 002F	Parameters for Event 1 (6 words)
01A0 0030–01A0 0047	Parameters for Event 2 (6 words)
01A0 0048–01A0 005F	Parameters for Event 3 (6 words)
01A0 0060—01A0 0077	Parameters for Event 4 (6 words)
01A0 0078–01A0 008F	Parameters for Event 5 (6 words)

(1) The C64x device has 21 parameter sets (6 words each) that can be used to reload/link EDMA transfers.

Table 1-8. EDMA Parameter RAM (continued)

HEX ADDRESS	REGISTER NAME
01A0 0090–01A0 00A7	Parameters for Event 6 (6 words)
01A0 00A8–01A0 00BF	Parameters for Event 7 (6 words)
01A0 00C0–01A0 00D7	Parameters for Event 8 (6 words)
01A0 00D8–01A0 00EF	Parameters for Event 9 (6 words)
01A0 00F0–01A0 00107	Parameters for Event 10 (6 words)
01A0 0108–01A0 011F	Parameters for Event 11 (6 words)
01A0 0120–01A0 0137	Parameters for Event 12 (6 words)
01A0 0138–01A0 014F	Parameters for Event 13 (6 words)
01A0 0150–01A0 0167	Parameters for Event 14 (6 words)
01A0 0168–01A0 017F	Parameters for Event 15 (6 words)
01A0 0150–01A0 0167	Parameters for Event 16 (6 words)
01A0 0168–01A0 017F	Parameters for Event 17 (6 words)
...	...
...	...
01A0 05D0–01A0 05E7	Parameters for Event 62 (6 words)
01A0 05E8–01A0 05FF	Parameters for Event 63 (6 words)
01A0 0600–01A0 0617	Reload/link parameters for Event M (6 words)
01A0 0618–01A0 062F	Reload/link parameters for Event N (6 words)
...	...
01A0 07E0–01A0 07F7	Reload/link parameters for Event Z (6 words)
01A0 07F8–01A0 07FF	Scratch pad area (2 words)

Table 1-9. Quick DMA (QDMA) and Pseudo Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
0200 0000	QOPT	QDMA Options Parameter
0200 0004	QSRC	QDMA Source Address
0200 0008	QCNT	QDMA Frame Count
0200 000C	QDST	QDMA Destination Address
0200 0010	QIDX	QDMA Index
0200 0014–0200 001C		Reserved
0200 0020	QSOPT	QDMA Pseudo Options
0200 0024	QSSRC	QDMA Pseudo Source Address
0200 0028	QSCNT	QDMA Pseudo Frame Count
0200 002C	QSDST	QDMA Pseudo Destination Address
0200 0030	QSIDX	QDMA Pseudo Index

Table 1-10. Interrupt Selector Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
019C 0000	MUXH	Interrupt Multiplexer High	Selects which interrupts drive CPU interrupts 10–15 (INT10–INT15)
019C 0004	MUXL	Interrupt Multiplexer Low	Selects which interrupts drive CPU interrupts 4–9 (INT04–INT09)
019C 0008	EXTPOL	External Interrupt Polarity	Sets the polarity of the external interrupts (EXT_INT4–EXT_INT7)
019C 000C–019C 01FF	–	Reserved	

Table 1-11. Peripheral Power-Down Control Register

HEX ADDRESS	ACRONYM	REGISTER NAME
019C 0200	PDCTL	Peripheral Power-Down Control
019C 0204–019F FFFF		Reserved

Table 1-12. McBSP 0 Registers

HEX ADDRESS	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 1-13. McBSP 1 Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
018C 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	
019C 0004	DXR1	McBSP1 Data Transmit Register via configuration bus	

Table 1-13. McBSP 1 Registers (continued)

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
0x3400 0000–0x37FF FFFF	DXR1	McBSP1 Data Transmit Register via peripheral bus	
019C 0008	SPCR1	McBSP1 Serial Port Control Register	
019C 000C	RCR1	McBSP1 Receive Control Register	
019C 0010	XCR1	McBSP1 Transmit Control Register	
019C 0014	SRGR1	McBSP1 Sample Rate Generator Register	
019C 0018	MCR1	McBSP1 Multichannel Control Register	
019C 001C	RCERE01	McBSP1 Enhanced Receive Channel Enable Register 0	
019C 0020	XCERE01	McBSP1 Enhanced Transmit Channel Enable Register 0	
019C 0024	PCR1	McBSP1 Pin Control Register	
019C 0028	RCERE11	McBSP1 Enhanced Receive Channel Enable Register 1	
019C 002C	XCERE11	McBSP1 Enhanced Transmit Channel Enable Register 1	
019C 0030	RCERE21	McBSP1 Enhanced Receive Channel Enable Register 2	
019C 0034	XCERE21	McBSP1 Enhanced Transmit Channel Enable Register 2	
019C 0038	RCERE31	McBSP1 Enhanced Receive Channel Enable Register 3	
019C 003C	XCERE31	McBSP1 Enhanced Transmit Channel Enable Register 3	
019C 0040–0193 FFFF	–	Reserved	

Table 1-14. McBSP 2 Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
01A4 0000	DRR2	McBSP2 Data Receive Register via configuration bus	The CPU and EDMA controller can only read this register; they cannot write to it.
0x3800 0000–0x3BFF FFFF	DRR2	McBSP2 Data Receive Register via peripheral bus	
01A4 0004	DXR2	McBSP2 Data Transmit Register via configuration bus	
0x3800 0000–0x3BFF FFFF	DXR2	McBSP2 Data Transmit Register via peripheral bus	
01A4 0008	SPCR2	McBSP2 Serial Port Control Register	
01A4 000C	RCR2	McBSP2 Receive Control Register	
01A4 0010	XCR2	McBSP2 Transmit Control Register	
01A4 0014	SRGR2	McBSP2 Sample Rate Generator Register	
01A4 0018	MCR2	McBSP2 Multichannel Control Register	
01A4 001C	RCERE02	McBSP2 Enhanced Receive Channel Enable Register 0	
01A4 0020	XCERE02	McBSP2 Enhanced Transmit Channel Enable Register 0	
01A4 0024	PCR2	McBSP2 Pin Control Register	
01A4 0028	RCERE12	McBSP2 Enhanced Receive Channel Enable Register 1	
01A4 002C	XCERE12	McBSP2 Enhanced Transmit Channel Enable Register 1	
01A4 0030	RCERE22	McBSP2 Enhanced Receive Channel Enable Register 2	
01A4 0034	XCERE22	McBSP2 Enhanced Transmit Channel Enable Register 2	
01A4 0038	RCERE32	McBSP2 Enhanced Receive Channel Enable Register 3	
01A4 003C	XCERE32	McBSP2 Enhanced Transmit Channel Enable Register 3	
01A4 0040–01A7 FFFF	–	Reserved	

Table 1-15. Timer 0 Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
0194 0000	CTL0	Timer 0 Control	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	Contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency.
0194 0008	CNT0	Timer 0 Counter	Contains the current value of the incrementing counter
0194 000C–0197 FFFF	–	Reserved	

Table 1-16. Timer 1 Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
0198 0000	CTL1	Timer 1 Control	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	Contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency.
0198 0008	CNT1	Timer 1 Counter	Contains the current value of the incrementing counter
0198 000C–019B FFFF	–	Reserved	

Table 1-17. Timer 2 Registers

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

Table 1-18. HPI Registers

HEX ADDRESS	ACRONYM	REGISTER NAME	COMMENTS
–	HPID	HPI Data	Host read/write access only
0188 0000	HPIC	HPI Control	HPIC has both host/CPU read/write access.
0188 0004	HPIA (HPIAW)(1)	HPI Address (Write)	HPIA has both host/CPU read/write access.
0188 0008	HPIA (HPIAR)(1)	HPI Address (Read)	
0188 000C–0189 FFFF	–	Reserved	
018A 0000	TRCTL	HPI Transfer Request Control	
018A 0004–018B FFFF	–	Reserved	

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

Table 1-19. GPIO Registers

HEX ADDRESS	ACRONYM	REGISTER NAME
01B0 0000	GPEN	GPIO Enable
01B0 0004	GPDIR	GPIO Direction
01B0 0008	GPVAL	GPIO Value
01B0 000C	–	Reserved
01B0 0010	GPDH	GPIO Delta High
01B0 0014	GPHM	GPIO High Mask
01B0 0018	GPDL	GPIO Delta Low
01B0 001C	GPLM	GPIO Low Mask

Table 1-19. GPIO Registers (continued)

HEX ADDRESS	ACRONYM	REGISTER NAME
01B0 0020	GPGC	GPIO Global Control
01B0 0024	GPPOL	GPIO Interrupt Polarity
01B0 0028–01B0 01FF	–	Reserved
01B0 0200	DEVICE_REV	Silicon revision identification (For more details, see the device characteristics listed in Table 1-3.)
01B0 0204–01B3 FFFF	–	Reserved

Table 1-20. PCI Peripheral Registers (C6415 and C6416 Only)(1)

HEX ADDRESS	ACRONYM	REGISTER NAME
01C0 0000	RSTSRC	DSP Reset Source/Status
01C0 0004	–	Reserved
01C0 0008	PCIIS	PCI Interrupt Source
01C0 000C	PCIIEN	PCI Interrupt Enable
01C0 0010	DSPMA	DSP Master Address
01C0 0014	PCIMA	PCI Master Address
01C0 0018	PCIMC	PCI Master Control
01C0 001C	CDSPA	Current DSP Address
01C0 0020	CPCIA	Current PCI Address
01C0 0024	CCNT	Current Byte Count
01C0 0028	–	Reserved
01C0 02C–01C1 FFEF	–	Reserved
0x01C1 FFF0	HSR	Host Status Register
0x01C1 FFF4	HDCR	Host-to-DSP Control Register
0x01C1 FFF8	DSPP	DSP Page
0x01C1 FFFC	–	Reserved
01C2 0000	EEADD	EEPROM Address
01C2 0004	EEDAT	EEPROM Data
01C2 0008	EECTL	EEPROM Control
01C2 000C–01C2 FFFF	–	Reserved
01C3 0000	TRCTL	PCI Transfer Request Control
01C3 0004–01C3 FFFF	–	Reserved

(1) These PCI registers are not supported on the C6414 device.

Table 1-21. UTOPIA Registers (C6415 and C6416 Only)(1)

HEX ADDRESS	ACRONYM	REGISTER NAME
01B4 0000	UCR	UTOPIA Control Register
01B4 0004	–	Reserved
01B4 0008	–	Reserved
01B4 000C	UIER	UTOPIA Interrupt Enable Register
01B4 0010	UIPR	UTOPIA Interrupt Pending Register
01B4 0014	CDR	Clock Detect Register
01B4 0018	EIER	Error Interrupt Enable Register
01B4 001C	EIPR	Error Interrupt Pending Register
01B4 0020–01B7 FFFF	–	Reserved

(1) These UTOPIA registers are not supported on the C6414 device.

Table 1-22. UTOPIA Queue Registers (C6415 and C6416 Only)(1)

HEX ADDRESS	ACRONYM	REGISTER NAME
3C00 0000	URQ	UTOPIA Receive Queue
3D00 0000	UXQ	UTOPIA Transmit Queue
3D00 0004–3FFF FFFF	–	Reserved

(1) These UTOPIA registers are not supported on the C6414 device.

Table 1-23. VCP Registers (C6414 Only)

HEX ADDRESS		ACRONYM	REGISTER
EDMA BUS	PERIPHERAL BUS		
5000 0000	01B8 0000	VCPIC0	VCP Input Configuration 0
5000 0004	01B8 0004	VCPIC1	VCP Input Configuration 1
5000 0008	01B8 0008	VCPIC2	VCP Input Configuration 2
5000 000C	01B8 000C	VCPIC3	VCP Input Configuration 3
5000 0010	01B8 0010	VCPIC4	VCP Input Configuration 4
5000 0014	01B8 0014	VCPIC5	VCP Input Configuration 5
5000 0040	01B8 0024	VCPOUT0	VCP Output 0
5000 0044	01B8 0028	VCPOUT1	VCP Output 1
5000 0080	–	VCPWBM	VCP Write Branch Metrics
5000 0088	–	VCPRDECS	VCP Read Decisions
–	01B8 0018	VCPEXE	VCP Execution
–	01B8 0020	VCPEND	VCP Endian
–	01B8 0040	VCPSTAT0	VCP Status Register 0
–	01B8 0044	VCPSTAT1	VCP Status Register 1
–	01B8 0050	VCPERR	VCP Error

Table 1-24. TCP Registers (C6414 Only)

HEX ADDRESS		ACRONYM	REGISTER
EDMA BUS	PERIPHERAL BUS		
5800 0000	01BA 0000	TCPIC0	TCP Input Configuration 0
5800 0004	01BA 0004	TCPIC1	TCP Input Configuration 1
5800 0008	01BA 0008	TCPIC2	TCP Input Configuration 2
5800 000C	01BA 000C	TCPIC3	TCP Input Configuration 3
5800 0010	01BA 0010	TCPIC4	TCP Input Configuration 4
5800 0014	01BA 0014	TCPIC5	TCP Input Configuration 5
5800 0018	01BA 0018	TCPIC6	TCP Input Configuration 6
5800 001C	01BA 001C	TCPIC7	TCP Input Configuration 7
5800 0020	01BA 0020	TCPIC8	TCP Input Configuration 8
5800 0024	01BA 0024	TCPIC9	TCP Input Configuration 9
5800 0028	01BA 0028	TCPIC10	TCP Input Configuration 10
5800 002C	01BA 002C	TCPIC11	TCP Input Configuration11
5800 0030	01BA 0030	TCPOUT	TCP Output Parameters
5802 0000	–	TCPSP	TCP Systematics and Parities Memory
5804 0000	–	TCPEXT	TCP Extrinsic Memory
5806 0000	–	TCPAP	TCP Apriori Memory
5808 0000	–	TCPINTER	TCP Interleaver Memory
580A 0000	–	TCPHD	TCP Hard Decisions Memory
–	01BA 0038	TCPEXE	TCP Execution

Table 1-24. TCP Registers (C6414 Only) (continued)

HEX ADDRESS		ACRONYM	REGISTER
EDMA BUS	PERIPHERAL BUS		
–	01BA 0040	TCPEND	TCP Endian
–	01BA 0050	TCPERR	TCP Error
–	01BA 0058	TCPSTAT	TCP Status

1.4.7 EDMA Channel Synchronization Events

The C64x EDMA supports up to 64 EDMA channels that service peripheral devices and external memory. Table 1-25 lists the source of C64x EDMA synchronization events associated with each of the programmable EDMA channels. For the C64x 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 EDMA Controller chapter of the *TMS320C6000 Peripherals Reference Guide* (literature number SPRU190).

Table 1-25. SM320C64x EDMA Channel Synchronization Events(1)

EDMA CHANNEL	EVENT NAME	EVENT DESCRIPTION
0	DSP_INT	HPI/PCI-to-DSP interrupt (PCI peripheral supported on C6415 and C6416 only)(2)
1	TINT0	Timer 0 interrupt
2	TINT1	Timer 1 interrupt
3	SD_INTA	EMIFA SDRAM timer interrupt
4	GPINT4/EXT_INT4	GPIO event 4/External interrupt pin 4
5	GPINT5/EXT_INT5	GPIO event 5/External interrupt pin 5
6	GPINT6/EXT_INT6	GPIO event 6/External interrupt pin 6
7	GPINT7/ EXT_INT7	GPIO event 7/External interrupt pin 7
8	GPINT0	GPIO event 0
9	GPINT1	GPIO event 1
10	GPINT2	GPIO event 2
11	GPINT3	GPIO event 3
12	XEVT0	McBSP0 transmit event
13	REVT0	McBSP0 receive event
14	XEVT1	McBSP1 transmit event
15	REVT1	McBSP1 receive event
16	–	None
17	XEVT2	McBSP2 transmit event
18	REVT2	McBSP2 receive event
19	TINT2	Timer 2 interrupt
20	SD_INTB	EMIFB SDRAM timer interrupt
21	–	Reserved, for future expansion
22–27	–	None
28	VCPREVT	VCP receive event (C6416 only)(3)

(1) 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 EDMA Controller chapter of the *TMS320C6000 Peripherals Reference Guide* (SPRU190).
(2) The PCI and UTOPIA peripherals are not supported on the C6414 device; therefore, these EDMA synchronization events are reserved.
(3) The VCP/TCP EDMA synchronization events are supported on the C6416 only. For the C6414 and C6415 devices, these events are reserved.

Table 1-25. SM320C64x EDMA Channel Synchronization Events (continued)

EDMA CHANNEL	EVENT NAME	EVENT DESCRIPTION
29	VCPXEVT	VCP transmit event (C6416 only)(3)
30	TCPREVT	TCP receive event (C6416 only)(3)
31	TCPXEVT	TCP transmit event (C6416 only)(3)
32	UREVT	UTOPIA receive event (C6415 and C6416 only)(2)
33–39	–	None
40	UXEVT	UTOPIA transmit event (C6415 and C6416 only)(2)
41–47	–	None
48	GPINT8	GPIO event 8
49	GPINT9	GPIO event 9
50	GPINT10	GPIO event 10
51	GPINT11	GPIO event 11
52	GPINT12	GPIO event 12
53	GPINT13	GPIO event 13
54	GPINT14	GPIO event 14
55	GPINT15	GPIO event 15
56–63	–	None

1.4.8 Interrupt Sources and Interrupt Selector

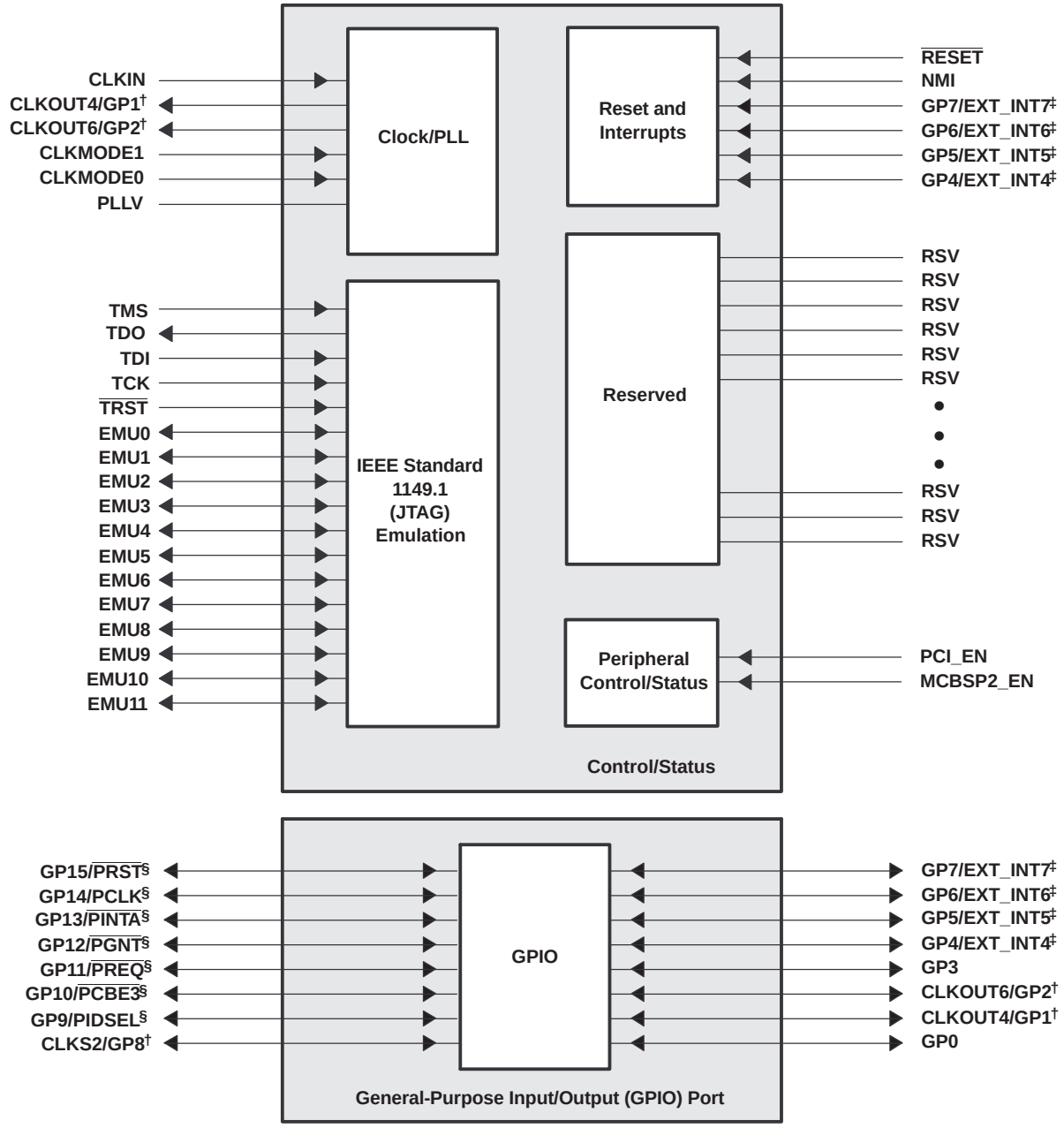
The C64x DSP core supports 16 prioritized interrupts, which are listed in Table 1-26. The highest-priority interrupt is INT_00 (dedicated to $\overline{\text{RESET}}$), while the lowest-priority interrupt is INT_15. The first four interrupts (INT_00–INT_03) are nonmaskable and fixed. The remaining interrupts (INT_04–INT_15) are maskable and default to the interrupt source specified in Table 1-26. 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 1-26. C64x DSP Interrupts

CPU INTERRUPT NUMBER	INTERRUPT SELECTOR CONTROL REGISTER	SELECTOR VALUE (BINARY)	INTERRUPT EVENT	INTERRUPT SOURCE
INT_00(1)	–	–	RESET	
INT_01(1)	–	–	NMI	
INT_02(1)	–	–	Reserved	Reserved. Do not use.
INT_03(1)	–		Reserved	GPIO interrupt 4/external interrupt 4
INT_04(2)	MUXL[4:0]	00100	GPINT4/EXT_INT4	GPIO interrupt 5/external interrupt 5
INT_05(2)	MUXL[9:5]	00101	GPINT5/EXT_INT5	TCP interrupt (C6416 only)
INT_06(2)	MUXL[14:10]	00110	GPINT6/EXT_INT6	GPIO interrupt 6/external interrupt 6
INT_07(2)	MUXL[20:16]	00111	GPINT7/EXT_INT7	GPIO interrupt 7/external interrupt 7
INT_08(2)	MUXL[25:21]	01000	EDMA_INT	EDMA channel (0–63) interrupt
INT_09(2)	MUXL[30:26]	01001	EMU_DTDMA	EMU DTDMA
INT_10(2)	MUXH[4:0]	00011	SD_INTA	EMIFA SDRAM timer interrupt
INT_11(2)	MUXH[9:5]	01010	EMU_RTDXRX	EMU real-time data exchange (RTDX) receive
INT_12(2)	MUXH[14:10]	01011	EMU_RTDXTX	EMU RTDX transmit
INT_13(2)	MUXH[20:16]	00000	DSP_INT	HPI/PCI-to-DSP interrupt (PCI supported on C6415 and C6416 only)
INT_14(2)	MUXH[25:21]	00001	TINT0	Timer 0 interrupt
INT_15(2)	MUXH[30:26]	00010	TINT1	Timer 1 interrupt
–	–	01100	XINT0	McBSP0 transmit interrupt
–	–	01101	RINT0	McBSP0 receive interrupt
–	–	01110	XINT1	McBSP1 transmit interrupt
–	–	01111	RINT1	McBSP1 receive interrupt
–	–	10000	GPINT0	GPIO interrupt 0
–	–	10001	XINT2	McBSP2 transmit interrupt
–	–	10010	RINT2	McBSP2 receive interrupt
–	–	10011	TINT2	Timer 2 interrupt
–	–	10100	SD_INTB	EMIFB SDRAM timer interrupt
–	–	10101	Reserved	Reserved. Do not use.
–	–	10110	Reserved	Reserved. Do not use.
–	–	10111	UINT	UTOPIA interrupt (C6415/C6416 only)
–	–	11000–11101	Reserved	Reserved. Do not use.
–	–	11110	VCPINT	VCP interrupt (C6416 only)
–	–	11111	TCPINT	TCP interrupt (C6416 only)

(1) Interrupts INT_00–INT_03 are nonmaskable and fixed.
(2) Interrupts INT_04–INT_15 are programmable by modifying the binary selector values in the Interrupt Selector Control registers fields. Table 1-26 shows the default interrupt sources for interrupts INT_04–INT_15. For more detailed information on interrupt sources and selection, see the Interrupt Selector and External Interrupts chapter of the *TMS320C6000 Peripherals Reference Guide* (SPRU190).

1.4.9 Signal Groups Description

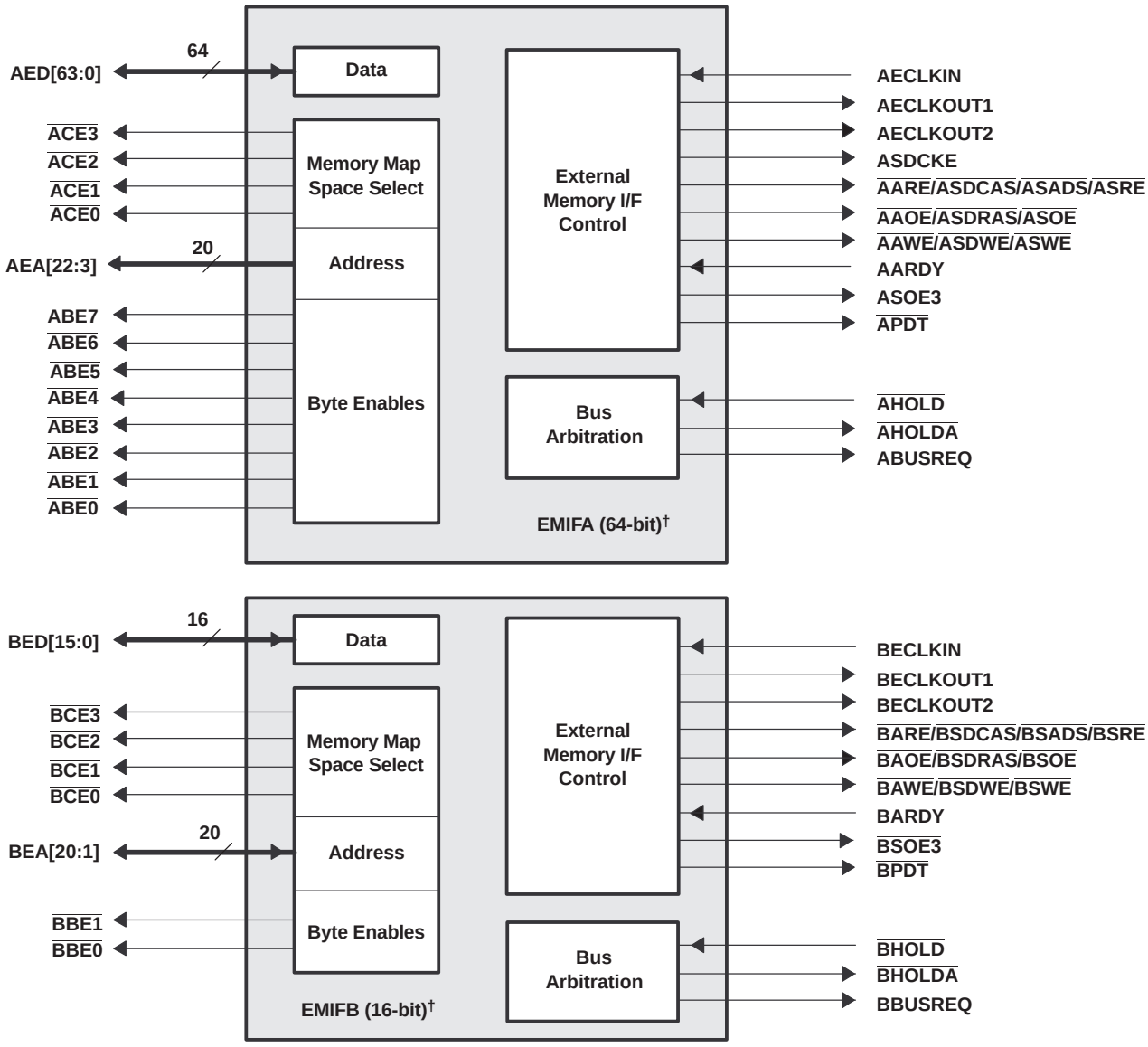


† These pins are MUXed with the GPIO port pins and by default these signals function as clocks (CLKOUT4 or CLKOUT6) or McBSP2 clock source (CLKS2). To use these MUXed pins as GPIO signals, the appropriate GPIO register bits (GPxEN and GPxDIR) must be properly enabled and configured. For more details, see the Device Configurations section of this data sheet.

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

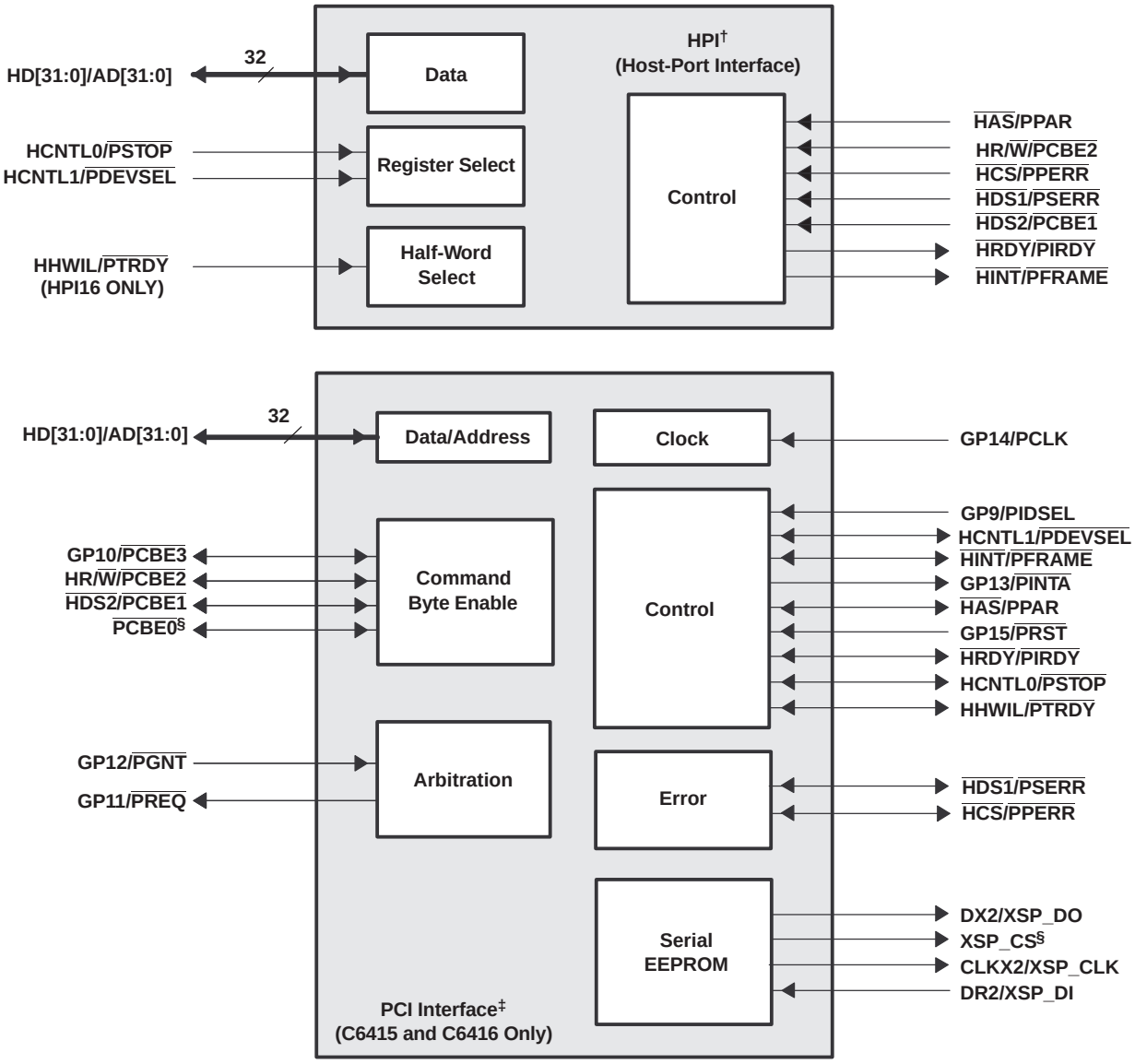
§ For the C6415 and C6416 devices, these GPIO pins are MUXed with the PCI peripheral pins. By default, these signals are set up to no function with both the GPIO and PCI pin functions *disabled*. For more details on these MUXed pins, see the Device Configurations section of this data sheet. For the C6414 device, the GPIO peripheral pins are *not* MUXed; the C6414 device does *not* support the PCI peripheral.

Figure 1-2. CPU and Peripheral Signals



† These C64x™ devices have two EMIFs (64-bit EMIFA and 16-bit EMIFB). The prefix “A” in front of a signal name indicates it is an EMIFA signal whereas a prefix “B” in front of a signal name indicates it is an EMIFB signal. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix “A” or “B” may be omitted from the signal name.

Figure 1-3. Peripheral Signals (1)

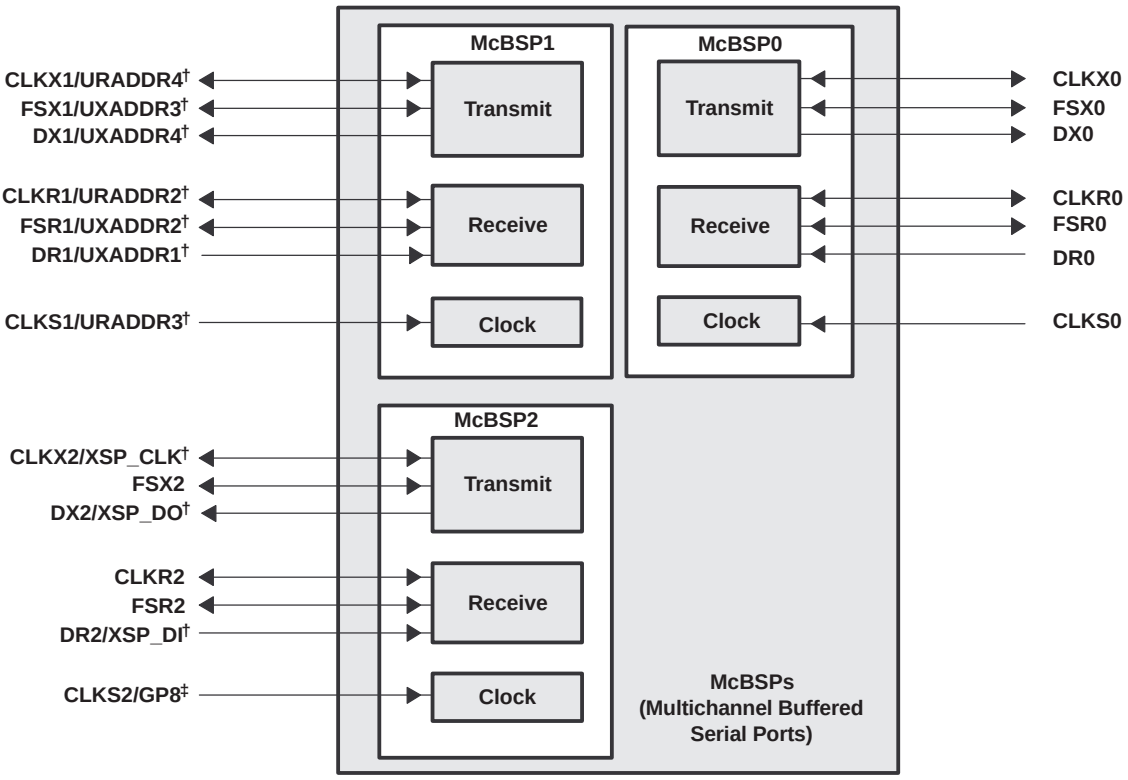


† For the C6415 and C6416 devices, these HPI pins are MUXed with the PCI peripheral. By default, these signals function as HPI. For more details on these MUXed pins, see the Device Configurations section of this data sheet. For the C6414 device, these HPI pins are *not* MUXed; the C6414 device does *not* support the PCI peripheral.

‡ For the C6415 and C6416 devices, these PCI pins (excluding PCBE0 and XSP_CS) are MUXed with the HPI, McBSP2, or GPIO peripherals. By default, these signals function as HPI, McBSP2, and no function, respectively. For more details on these MUXed pins, see the Device Configurations section of this data sheet. For the C6414 device, the HPI, McBSP2, and GPIO peripheral pins are *not* MUXed; the C6414 device does *not* support the PCI peripheral.

§ For the C6414 device, these pins are "Reserved (leave unconnected, ***do not*** connect to power or ground)."

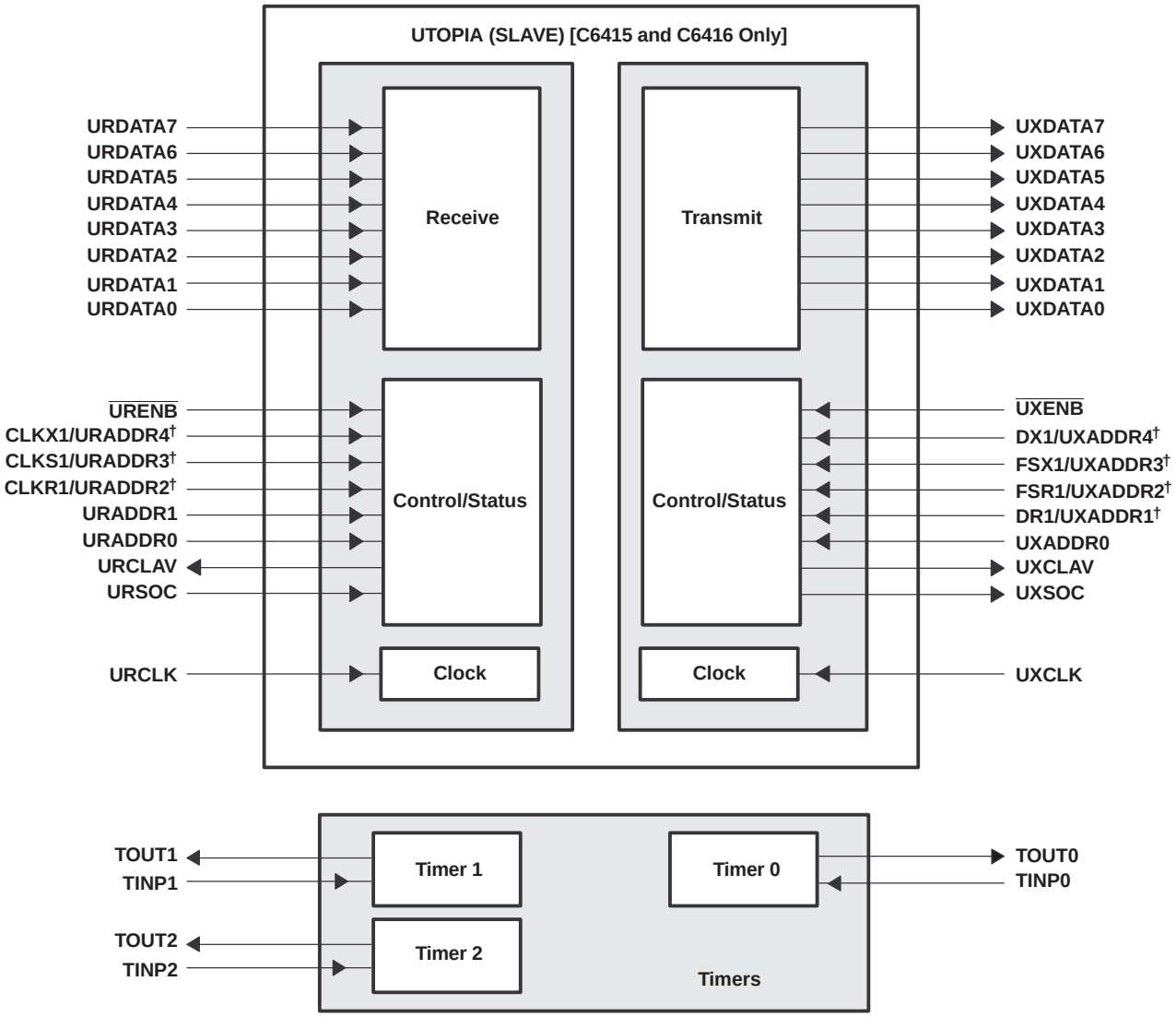
Figure 1-4. Peripheral Signals (2)



† For the C6415 and C6416 devices, these McBSP2 and McBSP1 pins are MUXed with the PCI and UTOPIA peripherals, respectively. By default, these signals function as McBSP2 and McBSP1, respectively. For more details on these MUXed pins, see the Device Configurations section of this data sheet.
For the C6414 device, these McBSP2 and McBSP1 peripheral pins are *not* MUXed; the C6414 device does *not* support PCI and UTOPIA peripherals.

‡ The McBSP2 clock source pin (CLKS2, default) is MUXed with the GP8 pin. To use this MUXed pin as the GP8 signal, the appropriate GPIO register bits (GP8EN and GP8DIR) must be properly enabled and configured. For more details, see the Device Configurations section of this data sheet.

Figure 1-5. Peripheral Signals (3)



† For the C6415 and C6416 devices, these UTOPIA pins are MUXed with the McBSP1 peripheral. By default, these signals function as McBSP1. For more details on these MUXed pins, see the Device Configurations section of this data sheet.
For the C6414 device, these McBSP1 peripheral pins are *not* MUXed; the C6414 does *not* support the UTOPIA peripheral.

Figure 1-6. Peripheral Signals (4)

Contents

2 Device Configurations

The C6414, C6415, and C6416 peripheral selections and other device configurations are determined by external pullup/pulldown resistors on the following pins (all of which are latched during device reset):

- Peripherals selection (C6415 and C6416 devices)
  - BEA11 (UTOPIA_EN)
  - PCI_EN (for C6415 or C6416, see Table 28 footnotes)
  - MCBSP2_EN (for C6414 or C6416, see Table 28 footnotes)

    The C6414 device does not support the PCI and UTOPIA peripherals; for proper operation of the C6414 device, do not oppose the internal pulldowns (IPDs) on the BEA11, PCI_EN, and MCBSP2_EN pins. (For IPUs/IPDs on pins, see the Terminal Functions table of this data manual.)
- Other device configurations (C64x)
  - BEA[20:13, 7]
  - HD5

2.1 Peripherals Selection

Some C6415/C6416 peripherals share the same pins (internally multiplexed) and are mutually exclusive (i.e., HPI, GPIO pins GP[15:9], PCI and its internal EEPROM, McBSP1, McBSP2, and UTOPIA). The VCP/TCP coprocessors (C6416 only) and other C64x peripherals (i.e., the timers, McBSP0, and the GP[8:0] pins) are always available.

- UTOPIA and McBSP1 peripherals

  The UTOPIA_EN pin (BEA11) is latched at reset. For C6415 and C6416 devices, this pin selects whether the UTOPIA peripheral or McBSP1 peripheral is functionally enabled (see Table 2-1).

  The C6414 device does not support the UTOPIA peripheral; for proper device operation, do not oppose the internal pulldown (IPD) on the BEA11 pin.

Table 2-1. UTOPIA_EN Peripheral Selection (McBSP1 and UTOPIA) (C6415/C6416 Only)

PERIPHERAL SELECTION UTOPIA_EN (BEA11) PIN [D16]	PERIPHERALS SELECTED		DESCRIPTION
	UTOPIA	McBSP1	
0		√	McBSP1 is enabled and UTOPIA is disabled (default). This means all multiplexed McBSP1/UTOPIA pins function as McBSP1 and all other stand alone UTOPIA pins are tied off (Hi-Z).
1	√		UTOPIA is enabled and McBSP1 is disabled. This means all multiplexed McBSP1/UTOPIA pins now function as UTOPIA and all other stand alone McBSP1 pins are tied off (Hi-Z).

- HPI, GP[15:9], PCI, EEPROM (internal to PCI), and McBSP2 peripherals

  The PCI_EN and MCBSP2_EN pins are latched at reset. They determine specific peripheral selection for the C6415 and C6416 devices, summarized in Table 2-2.

  The C6414 device does not support the PCI peripheral; for proper device operation, do not oppose the internal pulldowns (IPDs) on the PCI_EN and MCBSP2_EN pins.

Table 2-2. PCI_EN and MCBSP2_EN Peripheral Selection (HPI, GP[15:9], PCI, and McBSP2)

PERIPHERAL SELECTION(1)		PERIPHERALS SELECTED				
PCI_EN PIN [AA4]	MCBSP2_EN PIN [AF3]	HPI	GP[15:9]	PCI	EEPROM (INTERNAL TO PCI)	McBSP2
0	0	√	√			√
0	1	√	√			√

(1) The PCI_EN pin must be driven valid at all times and the user must not switch values throughout device operation. The MCBSP2_EN pin must be driven valid at all times and the user can switch values throughout device operation.

Table 2-2. PCI_EN and MCBSP2_EN Peripheral Selection (HPI, GP[15:9], PCI, and McBSP2) (continued)

PERIPHERAL SELECTION(1)		PERIPHERALS SELECTED				
PCI_EN PIN [AA4]	MCBSP2_EN PIN [AF3]	HPI	GP[15:9]	PCI	EEPROM (INTERNAL TO PCI)	McBSP2
1	0			√	√	
(2)1	1			√		√

(2) The only time McBSP2 is disabled is when both PCI_EN = 1 and MCBSP2_EN = 0. This configuration enables, at reset, the auto-initialization of the PCI peripheral through the PCI internal EEPROM [provided the PCI EEPROM auto-initialization pin (BEA13) is pulled up (EEAI = 1)]. The user then can enable the McBSP2 peripheral (disabling EEPROM) by dynamically changing MCBSP2_EN to a 1 after the device is initialized (out of reset).

- If the PCI is disabled (PCI_EN = 0), the HPI peripheral is enabled and GP[15:9] pins can be programmed as GPIO, provided the GPxEN and GPxDIR bits are properly configured.

  This means all multiplexed HPI/PCI pins function as HPI and all stand-alone PCI pins (PCBE0 and XSP_CS) are tied off (Hi-Z). Also, the multiplexed GPIO/PCI pins can be used as GPIO with the proper software configuration of the GPIO Enable and Direction registers (for more details, see Table 2-4).
- If the PCI is enabled (PCI_EN = 1), the HPI peripheral is disabled.

  This means all multiplexed HPI/PCI pins function as PCI. Also, the multiplexed GPIO/PCI pins function as PCI pins (for more details, see Table 2-4).
- The MCBSP2_EN pin, in combination with the PCI_EN pin, controls the selection of the McBSP2 peripheral and the PCI internal EEPROM (for more details, see Table 2-2 and its footnotes).

2.2 Other Device Configurations

Table 2-3 describes the C6414, C6415, and C6416 devices configuration pins, which are set up via external pullup/pulldown resistors through the specified EMIFB address bus pins (BEA[20:13, 11, 9:7]) and the HD5 pin. For more details on these device configuration pins, see the Terminal Functions table and the Debugging Considerations section.

Table 2-3. Device Configuration Pins (BEA[20:13, 9:7], HD5, and BEA11)

CONFIGURATION PIN	NO.	FUNCTIONAL DESCRIPTION
BEA20	E16	Device endian mode (LEND) 0 – System operates in big endian mode. 1 – System operates in little endian mode (default).
BEA19 BEA18	D18 C18	Bootmode [1:0] 00 – No boot 01 – HPI boot 10 – EMIFB 8-bit ROM boot with default timings (default mode) 11 – Reserved
BEA17 BEA16	B18 A18	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
BEA15 BEA14	D17 C17	EMIFA input clock select. Clock-mode select for EMIFB (BECLKIN_SEL[1:0]). 00 – BECLKIN (default mode) 01 – CPU/4 clock rate 10 – CPU/6 clock rate 11 – Reserved
BEA13	B17	PCI EEPROM auto-initialization (EEAI) (C6415 and C6416 devices only) [The C6414 device does not support the PCI peripheral; for proper device operation, do not oppose the internal pulldown (IPD) on the BEA13 pin.] PCI auto-initialization via external EEPROM 0 – PCI auto-initialization through EEPROM is disabled; the PCI peripheral uses the specified PCI default values (default). 1 – PCI auto-initialization through EEPROM is enabled; the PCI peripheral is configured through EEPROM provided the PCI peripheral pin is enabled (PCI_EN = 1) and the McBSP2 peripheral pin is disabled (MCBSP2_EN = 0). Note: If the PCI peripheral is disabled (PCI_EN pin = 0), this pin must not be pulled up. For more information on the PCI EEPROM default values, see the PCI chapter of the *TMS320C6000 Peripherals Reference Guide* (literature number SPRU190).
BEA11	D16	UTOPIA enable (UTOPIA_EN) (C6415 and C6416 devices only) [The C6414 device does not support the UTOPIA peripheral. For proper device operation, do not oppose the internal pulldown (IPD) on the BEA11 pin.] UTOPIA peripheral enable (functional) 0 – UTOPIA peripheral disabled (McBSP1 functions are enabled) (default). This means all multiplexed McBSP1/UTOPIA pins function as McBSP1 and all other standalone UTOPIA pins are tied off (Hi-Z). 1 – UTOPIA peripheral enabled (McBSP1 functions are disabled). This means all multiplexed McBSP1/UTOPIA pins now function as UTOPIA and all other stand-alone McBSP1 pins are tied off (Hi-Z).
		C6414 Devices / **C6415 Devices** / **C6416 Devices**
BEA7 BEA8 BEA9	D15 A16 B16	Do not oppose internal pulldown (IPD) / Pullup(1) / Do not oppose IPD Do not oppose IPD / Do not oppose IPD / Pullup(1) Do not oppose IPD / Do not oppose IPD / Pullup(1)
HD5	Y1	HPI peripheral bus width (HPI_WIDTH) 0 – HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the Hi-Z state.) 1 – HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.)

(1) For proper device operation, this pin must be externally pulled up with a 1-kΩ resistor.

2.3 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. The multiplexed pins that are configured by software can be programmed to switch functionalities at any time. The multiplexed 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 2-4 identifies the multiplexed pins on the C6414, C6415, and C6416 devices, shows the default (primary) function and the default settings after reset, and describes the pins, registers, etc., necessary to configure specific multiplexed functions.

2.4 Debugging Considerations

It is recommended that external connections be provided to device configuration pins, including CLKMODE[1:0], BEA[20:13, 11, 9:7], HD5/AD5, PCI_EN, and MCBSP2_EN. Although internal pullup/pulldown resistors exist on these pins (except for HD5/AD5), providing external connectivity adds convenience to the user in debugging and flexibility in switching operating modes.

Internal pullup/pulldown resistors also exist on the nonconfiguration pins on the BEA bus (BEA[12, 10, 6:1]). Do not oppose the internal pullup/pulldown resistors on these nonconfiguration pins with external pullup/pulldown resistors. If an external controller provides signals to these nonconfiguration pins, these signals must be driven to the default state of the pins at reset or not be driven at all.

For the internal pullup/pulldown resistors on the C6414, C6415, and C6416 device pins, see the Terminal Functions table.

Table 2-4. C6414, C6415, and C6416 Device Multiplexed Pins(1)

MULTIPLEXED PINS		DEFAULT FUNCTION	DEFAULT SETTING	DESCRIPTION
NAME	NO.			
CLKOUT4/GP1(2)	AE6	CLKOUT4	GP1EN = 0 (disabled)	These pins are software configurable. To use these pins as GPIO pins, the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input. GPxDIR = 1: GPx pin is an output.
CLKOUT6/GP2(2)	AD6	CLKOUT6	GP2EN = 0 (disabled)	
CLKS2/GP8(2)	AE4	CLKS2	GP8EN = 0 (disabled)	
GP9/PIDSEL	M3	None	GPxEN = 0 (disabled) PCI_EN = 0 (disabled)(1)	To use GP[15:9] as GPIO pins, the PCI must be disabled (PCI_EN = 0), the GPxEN bits in the GPIO Enable Register and the GPxDIR bits in the GPIO Direction. Register must be properly configured. GPxEN = 1: GPx pin enabled GPxDIR = 0: GPx pin is an input. GPxDIR = 1: GPx pin is an output.
GP10/$\overline{\text{PCBE3}}$	L2			
GP11/$\overline{\text{PREQ}}$	F1			
GP12/$\overline{\text{PGNT}}$	J3			
GP14/PCLK	G4			
GP13/$\overline{\text{PINTA}}$	F2			
GP15/$\overline{\text{PRST}}$	G3			
DX1/UXADDR4	AB11	DX1	UTOPIA_EN (BEA11) = 0 (disabled)(1)	By default, McBSP1 is enabled upon reset (UTOPIA is disabled). To enable the UTOPIA peripheral, an external pullup resistor (1 kΩ) must be provided on the BEA11 pin (setting UTOPIA_EN = 1 at reset).
FSX1/UXADDR3	AB13	FSX1		
FSR1/UXADDR2	AC9	FSR1		
DR1/UXADDR1	AF11	DR1		
CLKX1/URADDR4	AB12	CLKX1		
CLKS1/URADDR3	AC8	CLKS1		
CLKR1/URADDR2	AC10	CLKR1		

(1) For the C6415 and C6416 devices, all other stand-alone UTOPIA and PCI pins are tied off internally (pins in Hi-Z) when the peripheral is disabled [UTOPIA_EN (BEA11) = 0 or PCI_EN = 0].
(2) The C6414 device does not support the PCI and UTOPIA peripherals. These are the only multiplexed pins on the C6414 device, all other pins are stand-alone peripheral functions and are not multiplexed.

Table 2-4. C6414, C6415, and C6416 Device Multiplexed Pins (continued)

MULTIPLEXED PINS		DEFAULT FUNCTION	DEFAULT SETTING	DESCRIPTION
NAME	NO.			
CLKX2/XSP_CLK	AC2	CLKX2	PCI_EN = 0 (disabled)(1)	By default, HPI is enabled upon reset (PCI is disabled). To enable the PCI peripheral an external pullup resistor (1 kΩ) must be provided on the PCI_EN pin (setting PCI_EN = 1 at reset).
DR2/XSP_DI	AB3	DR2		
DX2/XSP_DO	AA2	DX2		
HD[31:0]/AD[31:0]		(3)HD[31:0]		
$\overline{\text{HAS}}$/PPAR	T3	$\overline{\text{HAS}}$		
HCNTL1/$\overline{\text{PDEVSEL}}$	R1	HCNTL1		
HCNTL0/$\overline{\text{PSTOP}}$	T4	HCNTL0		
$\overline{\text{HDS1}}$/$\overline{\text{PSERR}}$	T1	$\overline{\text{HDS1}}$		
$\overline{\text{HDS2}}$/$\overline{\text{PCBE1}}$	T2	$\overline{\text{HDS2}}$		
HR/$\overline{\text{W}}$/$\overline{\text{PCBE2}}$	P1	HR/$\overline{\text{W}}$		
HWWIL/$\overline{\text{PTRDY}}$	R3	HHWIL (HPI16 only)		
$\overline{\text{HINT}}$/$\overline{\text{PFRAME}}$	R4	$\overline{\text{HINT}}$		
$\overline{\text{HCS}}$/$\overline{\text{PPERR}}$	R2	$\overline{\text{HCS}}$		
$\overline{\text{HRDY}}$/$\overline{\text{PIRDY}}$	P4	$\overline{\text{HRDY}}$		

(3) For the pin numbers of the HD[31:0]/AD[31:0] multiplexed pins, see the Terminal Functions table.

2.5 Terminal Functions

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
CLOCK/PLL Configuration				
CLKIN	H4	I	IPD	Clock input. This clock is the input to the on-chip PLL.
CLKOUT4/GP1(3)	AE6	I/O/Z	IPD	Clock output at 1/4 of the device speed (O/Z) (default) or this pin can be programmed as a GPIO1 pin (I/O/Z).
CLKOUT6/GP2(3)	AD6	I/O/Z	IPD	Clock output at 1/6 of the device speed (O/Z) (default) or this pin can be programmed as a GPIO2 pin (I/O/Z).
CLKMODE1	G1	I	IPD	Clock-mode select. Selects whether the CPU clock frequency = input clock frequency ×1 (bypass), ×6, or ×12. For more details on the CLKMODE pins and the PLL multiply factors, see the Clock PLL section of this data manual.
CLKMODE0	H2	I	IPD	
PLLV(4)	J6	A(5)		PLL voltage supply
JTAG Emulation				
TMS	AB16	I	IPU	JTAG test-port mode select
TDO	AE19	O/Z	IPU	JTAG test-port data out
TDI	AF18	I	IPU	JTAG test-port data in
TCK	AF16	I	IPU	JTAG test-port clock
$\overline{\text{TRST}}$	AB15	I	IPD	JTAG test-port reset. For IEEE Std 1149.1 JTAG compatibility, see the IEEE Std 1149.1 JTAG Compatibility Statement section of this data manual.
EMU11	AC18	I/O/Z	IPU	Emulation pin [11:2]. Reserved for future use, leave unconnected.
EMU10	AD18			
EMU9	AE18			
EMU8	AC17			
EMU7	AF17			
EMU6	AD17			
EMU5	AE17			
EMU4	AC16			
EMU3	AD16			
EMU2	AE16			
EMU1 EMU0	AC15 AF15	I/O/Z	IPU	Emulation pins [1:0] Select the device functional mode of operation: **EMU[1:0] — Operation** 00 — Boundary Scan/Normal mode (see **Note**) 01 — Reserved 10 — Reserved 11 — Emulation/Normal mode (default) (see the IEEE 1149.1 JTAG Compatibility Statement section of this data manual) Normal mode refers to the DSPs normal operational mode, when the DSP is free running. The DSP can be placed in normal operational mode when the EMU[1:0] pins are configured for either Boundary Scan or Emulation. **Note:** When the EMU[1:0] pins are configured for Boundary Scan mode, the internal pulldown (IPD) on the $\overline{\text{TRST}}$ signal must not be opposed in order to operate in Normal mode. For the Boundary Scan mode pulldown EMU[1:0] pins with a dedicated 1-kΩ resistor.
Resets, Interrupts, and General-Purpose Input/Outputs (GPIOs)				
$\overline{\text{RESET}}$	AC7	I		Device reset
NMI	B4	I	IPD	Nonmaskable interrupt, edge-driven (rising edge)

(1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground
(2) 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) These pins are multiplexed pins. For more details, see the Device Configurations section of this data manual.
(4) PLLV is not part of external voltage supply. See the Clock PLL section for information on how to connect this pin.
(5) A = Analog signal (PLL filter)

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
GP7/EXT_INT7	AF4	I/O/Z	IPU	General-purpose input/outputs (GPIO) (I/O/Z) or external interrupts (input only). The default after reset setting is GPIO enabled as input only. When these pins function as external interrupts [by selecting the corresponding interrupt enable register bit (IER.[7:4])], they are edge driven and the polarity can be independently selected via the External Interrupt Polarity Register bits (EXTPOL[3:0]).
GP6/EXT_INT6	AD5			
GP5/EXT_INT5	AE5			
GP4/EXT_INT4	AF5			
GP15/$\overline{\text{PRST}}$(6)	G3	I/O/Z		General-purpose input/output (GPIO) 15 (I/O/Z) or PCI reset (I). No function at default.
GP14/PCLK(6)	F2			GPIO 14 (I/O/Z) or PCI clock (I). No function at default.
GP13/$\overline{\text{PINTA}}$(6)	G4			GPIO 13 (I/O/Z) or PCI interrupt A (O/Z). No function at default.
GP12/$\overline{\text{PGNT}}$(6)	J3			GPIO 12 (I/O/Z) or PCI bus grant (I). No function at default.
GP11/$\overline{\text{PREQ}}$(6)	F1			GPIO 11 (I/O/Z) or PCI bus request (O/Z). No function at default.
GP10/$\overline{\text{PCBE3}}$(6)	L2			GPIO 10 (I/O/Z) or PCI command/byte enable 3 (I/O/Z). No function at default.
GP9/PIDSEL(6)	M3			GPIO 9 (I/O/Z) or PCI initialization device select (I). No function at default.
GP3	AC6	I/O/Z		GPIO 3 (I/O/Z). The default after reset setting is GPIO 3 enabled as input only.
GP0	AF6	I/O/Z	IPD	GPIO 0 (I/O/Z). Can be programmed as GPIO 0 (input only) (default) or as GPIO 0 (output only) pin or output as a general-purpose interrupt (GP0INT) signal (output only).
CLKS2/GP8(6) (7)	AE4	I/O/Z	IPD	McBSP2 external clock source (CLKS2) [input only] (default) or this pin can be programmed as a GPIO 8 pin (I/O/Z).
CLKOUT6/GP2(6) (7)	AD6	I/O/Z	IPD	Clock output at 1/6 of the device speed (O/Z) (default) or this pin can be programmed as a GPIO 2 pin (I/O/Z).
CLKOUT4/GP1 (6) (7)	AE6	I/O/Z	IPD	Clock output at 1/4 of the device speed (O/Z) (default) or this pin can be programmed as a GPIO 1 pin (I/O/Z).
Host-Port Interface (HPI) (C64x) or Peripheral Component Interconnect (PCI) (C6415 or C6416 Devices Only)				
PCI_EN	AA4	I	IPD	PCI enable. This pin controls the selection (enable/disable) of the HPI and GP[15:9], or PCI peripherals (for the C6415 and C6416 devices). This pin works in conjunction with the MCBSP2_EN pin to enable/disable other peripherals (for more details, see the Device Configurations section of this data manual). The C6414 device does not support the PCI peripheral; for proper device operation, do not oppose the internal pulldown (IPD) on this pin.
$\overline{\text{HINT}}$/$\overline{\text{PFRAME}}$(6)	R4	I/O/Z		Host interrupt from DSP to host (O) (default) or PCI frame (I/O/Z)
HCNTL1/$\overline{\text{PDEVSEL}}$(6)	R1	I/O/Z		Host control 1. Selects between control, address, or data registers (I) (default) or PCI device select (I/O/Z).
HCNTL0/$\overline{\text{PSTOP}}$(6)	T4	I/O/Z		Host control 0. Selects between control, address, or data registers (I) (default) or PCI stop (I/O/Z).
HHWIL/$\overline{\text{PTRDY}}$(6)	R3	I/O/Z		Host half-word select-first or second half word (not necessarily high or low order) (For HPI16 bus width selection only) (I) (default) or PCI target ready (I/O/Z).
HR/$\overline{\text{W}}$/$\overline{\text{PCBE2}}$(6)	P1	I/O/Z		Host read or write select (I) (default) or PCI command/byte enable 2 (I/O/Z)
$\overline{\text{HAS}}$/PPAR(6)	T3	I/O/Z		Host address strobe (I) (default) or PCI parity (I/O/Z)
$\overline{\text{HCS}}$/$\overline{\text{PPERR}}$(6)	R2	I/O/Z		Host chip select (I) (default) or PCI parity error (I/O/Z)
$\overline{\text{HDS1}}$/$\overline{\text{PSERR}}$(6)	T1	I/O/Z		Host data strobe 1 (I) (default) or PCI system error (I/O/Z)
$\overline{\text{HDS2}}$/$\overline{\text{PCBE1}}$(6)	T2	I/O/Z		Host data strobe 2 (I) (default) or PCI command/byte enable 1 (I/O/Z)
$\overline{\text{HRDY}}$/$\overline{\text{PIRDY}}$(6)	P4	I/O/Z		Host ready from DSP to host (O) (default) or PCI initiator ready (I/O/Z)

(6) For the C6415 and C6416 devices, these pins are multiplexed pins. For more details, see the Device Configurations section of this data manual. The C6414 device does not support the PCI or UTOPIA peripherals; therefore, these MUXed peripheral pins are stand alone peripheral functions for this device.
(7) For the C6414 device, only these pins are multiplexed pins.

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
HD31/AD31(6)	J2	I/O/Z		Host-port data (I/O/Z) (default) (C64x) or PCI data-address bus (I/O/Z) (C6415 and C6416) AS HPI data bus (PCIPEN pin = 0): Used for transfer of data, address, and control . Host-port bus width user configurable at device reset via a 10-kΩ. resistor pullup/pulldown resistor on the HD5 pin: HD5 pin = 0: HPI operates as an HPI16. (HPI bus is 16 bits wide. HD[15:0] pins are used and the remaining HD[31:16] pins are reserved pins in the high-impedance state.) HD5 pin = 1: HPI operates as an HPI32. (HPI bus is 32 bits wide. All HD[31:0] pins are used for host-port operations.) As PCI data-address bus (PCI_EN pin = 1) (C6415 and C6416 devices only): Used for transfer of data and address The C6414 device does not support the PCI peripheral; therefore, the HPI peripheral pins are stand alone peripheral functions, not MUXed.
HD30/AD30(6)	K3			
HD29/AD29(6)	J1			
HD28/AD28(6)	K4			
HD27/AD27(6)	K2			
HD26/AD26(6)	L3			
HD25/AD25(6)	K1			
HD24/AD24(6)	L4			
HD23/AD23(6)	L1			
HD22/AD22(6)	M4			
HD21/AD21(6)	M2			
HD20/AD20(6)	N4			
HD19/AD19(6)	M1			
HD18/AD18(6)	N5			
HD17/AD17(6)	N1			
HD15/AD15(6)	U4			
HD16/AD16(6)	P5			
HD15/AD15(6)	U4			
HD14/AD14(6)	U1			
HD13/AD13(6)	U3			
HD12/AD12(6)	U2			
HD11/AD11(6)	V4			
HD10/AD10(6)	V1			
HD9/AD9(6)	V3			
HD8/AD8(6)	V2			
HD7/AD7(6)	W2			
HD6/AD6(6)	W4			
HD5/AD5(6)	Y1			
HD4/AD4(6)	Y3			
HD3/AD3(6)	Y2			
HD2/AD2(6)	Y4			
HD1/AD1(6)	AA1			
HD0/AD0(6)	AA3			
$\overline{\text{PCBE0}}$	W3	I/O/Z		PCI command/byte enable 0 (I/O/Z). When PCI is disabled (PCI_EN = 0), this pin is tied–off. For the C6414 device this pin is “Reserved (leave unconnected, do not connect to power or ground).”
XSP_CS	AD1	O	IPD	PCI serial interface chip select (O). When PCI is disabled (PCI_EN = 0), this pin is tied–off. For the C6414 device this pin is “Reserved (leave unconnected, do not connect to power or ground).”
CLKX2/ XSP_CLK(6)	AC2	I/O/Z	IPD	McBSP2 transmit clock (I/O/Z) [default] or PCI serial interface clock (O).
DR2/XSP_DI(6)	AB3	I	IPU	McBSP2 receive data (I) [default] or PCI serial interface data in (I). In PCI mode, this pin is connected to the output data pin of the serial PROM.
DX2/XSP_DO(6)	AA2	O/Z	IPU	McBSP2 transmit data (O/Z) [default] or PCI serial interface data out (O). In PCI mode, this pin is connected to the input data pin of the serial PROM.

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
GP15/$\overline{\text{PRST}}$(6)	G3	I/O/Z		General–purpose input/output (GPIO) 15 pin (I/O/Z) or PCI reset (I). No function at default.
GP14/PCLK(6)	F2			GPIO 14 pin (I/O/Z) or PCI clock (I). No function at default.
GP13/$\overline{\text{PINTA}}$(6)	G4			GPIO 13 pin (I/O/Z) or PCI interrupt A (O/Z). No function at default.
GP12/$\overline{\text{PGNT}}$(6)	J3			GPIO 12 pin (I/O/Z) or PCI bus grant (I). No function at default.
GP11/$\overline{\text{PREQ}}$(6)	F1			GPIO 11 pin (I/O/Z) or PCI bus request (O/Z). No function at default.
GP10/$\overline{\text{PCBE3}}$(6)	L2			GPIO 10 pin (I/O/Z) or PCI command/byte enable 3 (I/O/Z). No function at default.
GP9/PIDSEL(6)	M3			GPIO 9 pin (I/O/Z) or PCI initialization device select (I). No function at default.
EMIFA (64-bit) Control Signals Common to All Types of Memory(8)(9)				
$\overline{\text{ACE3}}$	L26	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
$\overline{\text{ACE2}}$	K23	O/Z	IPU	
$\overline{\text{ACE1}}$	K24	O/Z	IPU	
$\overline{\text{ACE0}}$	K25	O/Z	IPU	
$\overline{\text{ABE7}}$	T23	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)
$\overline{\text{ABE6}}$	T24	O/Z	IPU	
$\overline{\text{ABE5}}$	R25	O/Z	IPU	
$\overline{\text{ABE4}}$	R26	O/Z	IPU	
$\overline{\text{ABE3}}$	M25	O/Z	IPU	
$\overline{\text{ABE2}}$	M26	O/Z	IPU	
$\overline{\text{ABE1}}$	L23	O/Z	IPU	
$\overline{\text{ABE0}}$	L24	O/Z	IPU	
$\overline{\text{APDT}}$	M22	O/Z	IPU	EMIFA peripheral data transfer, allows direct transfer between external peripherals
EMIFA (64-Bit) — Bus Arbitration				
AHOLDA	N22	O	IPU	EMIFA hold-request-acknowledge to the host
AHOLD	V23	I	IPU	EMIFA hold request from the host
ABUSREQ	P22	O	IPU	EMIFA bus request output
EMIFA (64-Bit) — Asynchronous/Synchronous Memory Control				
AECLKIN	H25	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 BEA[17:16] pins. AECLKIN is the default for the EMIFA input clock.
AECLKOUT2	J23	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	J26	O/Z	IPD	EMIFA output clock 1 [at EMIFA input clock (AECLKIN, CPU/4 clock, or CPU/6 clock) frequency].
AARE/ ASDCAS/ ASADS/ASRE	J25	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	J24	O/Z	IPU	EMIFA asynchronous memory output–enable/SDRAM row-address strobe/programmable synchronous interface output–enable
AAWE/ ASDWE/ ASWE	K26	O/Z	IPU	EMIFA asynchronous memory write-enable/SDRAM write-enable/programmable synchronous interface write–enable
ASDCKE	L25	O/Z	IPU	EMIFA SDRAM clock-enable (used for self–refresh mode). (EMIFA module only.) • If SDRAM is not in system, ASDCKE can be used as a general–purpose output.
ASOE3	R22	O/Z	IPU	EMIFA synchronous memory output–enable for ACE3 (for glueless FIFO interface)
AARDY	L22	I	IPU	Asynchronous memory ready input
EMIFA (64–Bit) — Address				

(8) These C64x devices have two EMIFs (64-bit EMIFA and 16-bit EMIFB). The prefix "A" in front of a signal name indicates it is an EMIFA signal whereas a prefix "B" in front of a signal name indicates it is an EMIFB signal. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix "A" or "B" may be omitted from the signal name.
(9) To maintain signal integrity for the EMIF signals, serial termination resistors should be inserted into all EMIF output signal lines.

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU (2)	DESCRIPTION
AEA22	T22	O/Z	IPD	EMIFA external address (doubleword address)
AEA21	V24			
AEA20	V25			
AEA19	V26			
AEA18	U23			
AEA17	U24			
AEA16	U25			
AEA15	U26			
AEA14	T25			
AEA13	T26			
AEA12	R23			
AEA11	R24			
EMIFA (64-Bit) — Address				
AEA10	P23	O/Z	IPD	EMIFA external address (doubleword address)
AEA9	P24			
AEA8	P26			
AEA7	N23			
AEA6	N24			
AEA5	N26			
AEA4	M23			
AEA3	M24			
EMIFA (64-Bit) — Data				
AED63	AF24	I/O/Z	IPU	EMIFA external data
AED62	AF23			
AED61	AE23			
AED60	AE22			
AED59	AD22			
AED58	AF22			
AED57	AD21			
AED56	AE21			
AED55	AC21			
AED54	AF21			
AED53	AD20			
AED52	AE20			
AED51	AC20			
AED50	AF20			
AED49	AC19			
AED48	AD19			
AED47	W24			
AED46	W23			
AED45	Y26			
AED44	Y23			
AED43	Y25			
AED42	Y24			
AED41	AA26			
AED40	AA23			

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
AED39	AA25	I/O/Z	IPU	EMIFA external data
AED38	AA24			
AED37	AB26			
AED36	AB24			
AED35	AB25			
AED34	AC25			
AED33	AC26			
AED32	AD26			
AED31	C26			
AED30	D26			
AED29	D25			
AED28	E25			
AED27	E24			
AED26	E26			
AED25	F24			
AED24	F25			
AED23	F23			
AED22	F26			
AED21	G24			
AED20	G25			
AED19	G23			
AED18	G26			
AED17	H23			
AED16	H24			
AED15	C19			
AED14	D19			
AED13	A20			
AED12	D20			
AED11	B20			
AED10	C20			
AED9	A21			
AED8	D21			
AED7	B21			
AED6	C21			
AED5	A22			
AED4	C22			
AED3	B22			
AED2	B23			
AED1	A23			
AED0	A24			
EMIFB (16-bit) — Control Signals Common to All Types of Memory				
BCE3	A13	O/Z	IPU	EMIFB memory space enables • Enabled by bits 26 through 31 of the word address • Only one pin is asserted during any external data access
BCE2	C12	O/Z	IPU	
BCE1	B12	O/Z	IPU	
BCE0	A12	O/Z	IPU	

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
BBE1	D13	O/Z	IPU	EMIFB 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)
BBE0	C13	O/Z	IPU	
BPDT	E12	O/Z	IPU	EMIFB peripheral data transfer, allows direct transfer between external peripherals
EMIFB (16-Bit) — Bus Arbitration				
BHOLDA	E13	O	IPU	EMIFB hold–request–acknowledge to the host
BHOLD	B19	I	IPU	EMIFB hold request from the host
BBUSREQ	E14	O	IPU	EMIFB bus request output
EMIFB (16-Bit) — Asynchronous/Synchronous Memory Control				
BECLKIN	A11	I	IPD	EMIFB external input clock. The EMIFB input clock (BECLKIN, CPU/4 clock, or CPU/6 clock) is selected at reset via the pullup/pulldown resistors on the BEA[15:14] pins. BECLKIN is the default for the EMIFB input clock.
BECLKOUT2	D11	O/Z	IPD	EMIFB output clock 2. Programmable to be EMIFB input clock (BECLKIN, CPU/4 clock, or CPU/6 clock) frequency divided by 1, 2, or 4.
BECLKOUT1	D12	O/Z	IPD	EMIFB output clock 1 [at EMIFB input clock (BECLKIN, CPU/4 clock, or CPU/6 clock) frequency].
BARE/BSDCAS/ BSADS/BSRE	A10	O/Z	IPU	EMIFB 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 BSADS and BSRE: If RENEN = 0, then the BSADS/BSRE signal functions as the BSADS signal. If RENEN = 1, then the BSADS/BSRE signal functions as the BSRE signal.
BAOE/BSDRAS/ BSOE	B11	O/Z	IPU	EMIFB asynchronous memory output–enable/SDRAM row–address strobe/programmable synchronous interface output–enable
BAWE/BSDWE/ BSWE	C11	O/Z	IPU	EMIFB asynchronous memory write–enable/SDRAM write–enable/programmable synchronous interface write–enable
BSOE3	E15	O/Z	IPU	EMIFB synchronous memory output enable for BCE3 (for glueless FIFO interface)
BARDY	E11	I	IPU	EMIFB asynchronous memory ready input

TERMINAL NAME	TERMINAL NO.	TYPE(1)	IPD/IPU(2)	DESCRIPTION
EMIFB (16-Bit) Address				
BEA20	E16	I/O/Z	IPU	EMIFB external address (half–word address) (O/Z) • Also controls initialization of DSP modes at reset (I) via pullup/pulldown resistors - – Device Endian mode
BEA19	D18		IPU	BEA20: 0 – Big Endian 1 - Little Endian (default mode)
BEA18	C18		IPD	
BEA17	B18			
BEA16	A18			
BEA15	D17			
BEA14	C17			
BEA13	B17			
BEA12	A17			
BEA11	D16			
BEA10	C16			
BEA9	B16			
BEA8	A16			
BEA7	D15			
BEA6	C15			
BEA5	B15			
BEA4	A15			
BEA3	D14			
BEA2	C14			
BEA1	A14			

3 Development Support

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

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

- Software Development Tools:
  - Code Composer Studio. Integrated Development Environment (IDE) including Editor
  - C/C++/Assembly Code Generation, and Debug plus additional development tools
  - Scalable, Real-Time Foundation Software (DSP/BIOS.), which provides the basic run-time target software needed to support any DSP application.
- Hardware Development Tools:
  - Extended Development System (XDS.) Emulator (supports C6000. DSP multiprocessor system debug) EVM (Evaluation Module)

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.

3.1 Device and Development Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all TMS320. DSP devices and support tools. Each TMS320. DSP family member has one of three prefixes; TMX, TMP, or TMS. TI recommends two of three possible prefix designators for support tools; TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

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

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

SM Fully qualified production device

Support tool development evolutionary flow:

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

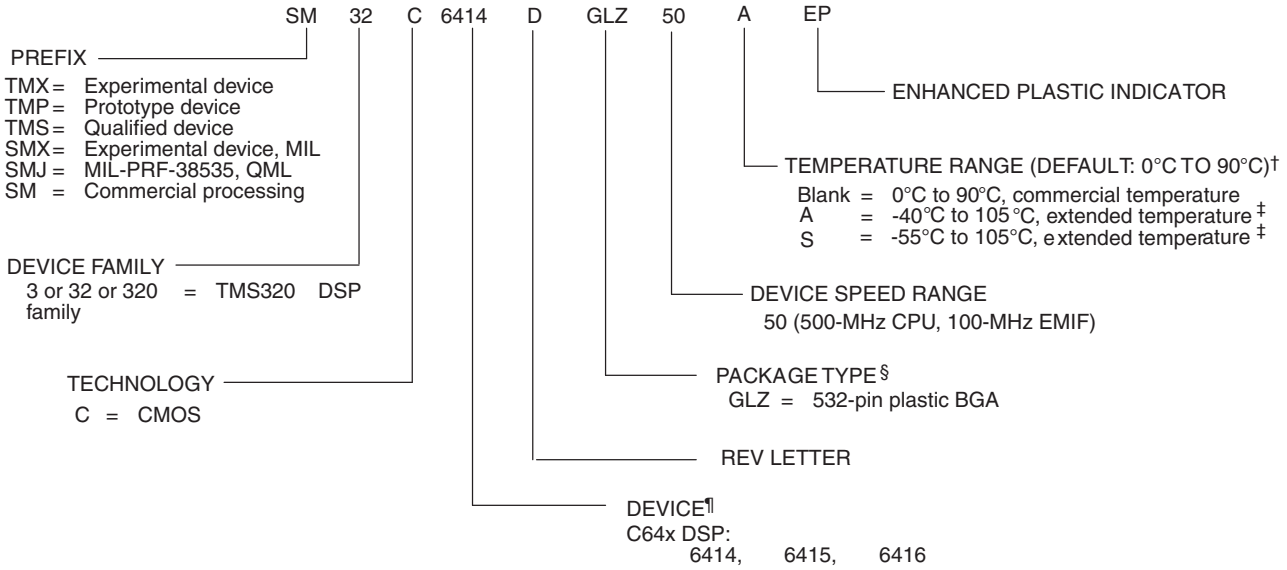
TMDS Fully qualified development-support product

TMX and TMP devices and TMDX development-support tools are shipped with appropriate disclaimers describing their limitations and intended uses. Experimental devices (TMX) may not be representative of a final product and Texas Instruments reserves the right to change or discontinue these products without notice.

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

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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, GLZ), the temperature range (for example, blank is the default commercial temperature range), and the device speed range in megahertz. Figure 3-1 provides a legend for reading the complete device name for any SM320C64x. DSP generation member.



† See the Recommended Operating Conditions section of this data sheet for more details.
‡ The extended temperature "A or S version" devices may have different operating conditions than the commercial temperature devices. See the Recommended Operating Conditions section of this data sheet for more details.
§ BGA = Ball Grid Array
¶ For the actual device part numbers (P/Ns) and ordering information, see the orderable information at the end of this data sheet.

Figure 3-1. TMS320C64xE DSP Device Nomenclature (Including C6414, C6415, and C6416 Devices)

3.1.1 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* (SPRU189) describes the C6000E DSP CPU (core) architecture, instruction set, pipeline, and associated interrupts.

The *TMS320C6000 Peripherals Reference Guide* (SPRU190) describes the functionality of the peripherals available on the C6000E DSP platform of devices, such as the 64-/32-/16-bit external memory interfaces (EMIFs), direct-memory-access (DMA), enhanced direct-memory-access (EDMA) controller, multichannel buffered serial ports (McBSPs), an 8-bit Universal Test and Operations PHY Interface for ATM Slave (UTOPIA Slave) port, 32-/16-bit host-port interfaces (HPIs), a peripheral component interconnect (PCI), expansion bus (XB), peripheral component interconnect (PCI), clocking and phase-locked loop (PLL); general-purpose timers, general-purpose input/output (GPIO) port, and power-down modes. This guide also includes information on internal data and program memories.

The *TMS320C6000 Technical Brief* (SPRU197) gives an introduction to the TMS320C62xE/ TMS320C67xE devices, associated development tools, and third-party support.

The *TMS320C64x Technical Overview* (SPRU395) gives an introduction to the C64xE digital signal processor, and discusses the application areas that are enhanced by the C64xE DSP VelociTI.2E VLIW architecture.

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

The tools support documentation is electronically available within the Code Composer StudioE Integrated Development Environment (IDE). For a complete listing of C6000E DSP latest documentation, visit the Texas Instruments web site on the Worldwide Web at http://www.ti.com uniform resource locator (URL).

See the worldwide web URL for the *How to Begin Development Today With the TMS320C6414, TMS320C6415, and TMS320C6416 DSPs* application report (SPRA718), which describes in more details the compatibility and similarities/differences among the C6414, C6415, C6416, and C6211 devices.

3.1.2 Clock PLL

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

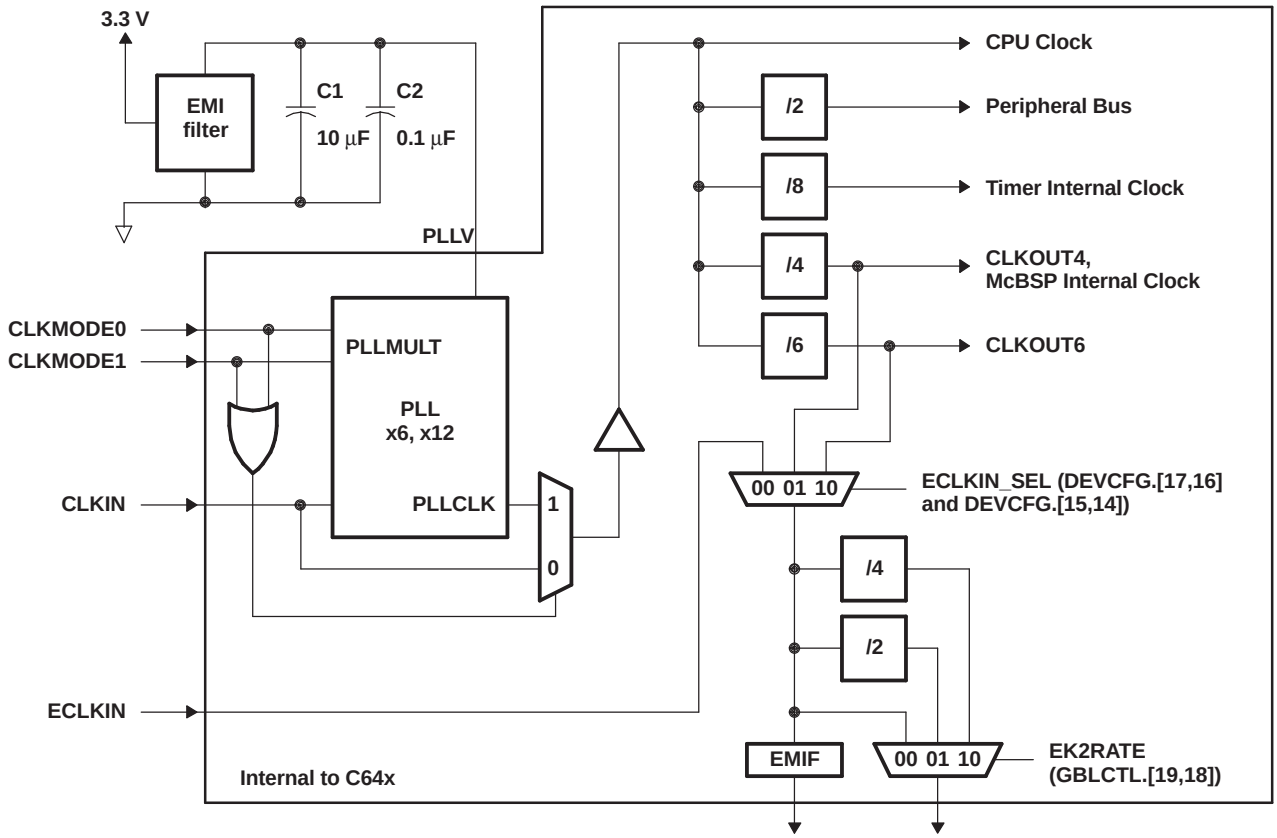
To use the PLL to generate the CPU clock, the external PLL filter circuit must be properly designed. Figure 5 shows the external PLL circuitry for either x1 (PLL bypass) or other PLL multiply modes.

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 electrical 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 electrical section). Table 3-1 lists some examples of compatible CLKIN external clock sources.

Table 3-1. Compatible CLKIN External Clock Sources

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



(For the PLL Options, CLKMODE Pins Setup, and PLL Clock Frequency Ranges, see NO TAG.)

A. Place all PLL external components (C1, C2, and the EMI filter) as close to the C6000 DSP device as possible. For the best performance, TI recommends that all the PLL external components be on a single side of the board without jumpers, switches, or components other than the ones shown.

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

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

D. EMI filter manufacturer TDK part number ACF451832-333, -223, -153, -103. Panasonic part number EXCCET103U.

Figure 3-2. External PLL Circuitry for Either PLL Multiply Modes or x1 (Bypass) Mode

Table 3-2. SM320C64x PLL Multiply Factor Options, Clock Frequency Ranges, and Typical Lock Time(1) (2)

GLZ PACKAGE-23 × 23-mm BGA							
CLKMODE1	CLKMODE0	CLKMODE (PLL MULTIPLY FACTORS)	CLKIN RANGE (MHz)	CPU CLOCK FREQUENCY RANGE (MHz)	CLKOUT4 RANGE (MHz)	CLKOUT6 RANGE (MHz)	TYPICAL LOCK TIME (ms)(3)
0	0	Bypass (x1)	30–75	30–75	7.5–18.8	5–12.5	N/A
0	1	x6	30–75	180–450	45–112.5	30–75	75
1	0	x12	30–60	360–720	90–180	60–120	
1	1	Reserved	–	–	–	–	–

(1) These clock frequency range values are applicable to a 600-MHz CPU, 133-MHz EMIFA speed device. For a 500-MHz CPU, 100-MHz EMIF, and 700-MHz CPU, 100 MHz EMIF device speed values, see the CLKIN timing requirements table for the specific device speed.
(2) Use external pullup resistors on the CLKMODE pins (CLKMODE1 and CLKMODE0) to set the C64x device to one of the valid PLL multiply clock
(3) 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.

3.1.3 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.

3.1.4 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 3-3).

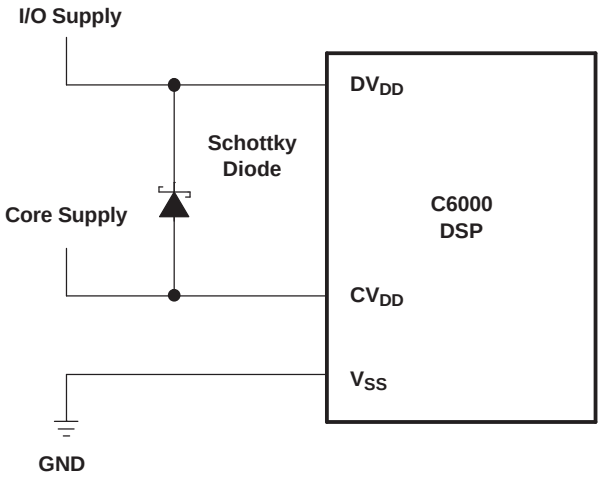


Figure 3-3. Schottky Diode Diagram

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.

3.1.5 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, no more than 1.25-cm maximum distance to be effective. Physically smaller caps are better, such as 0402, but need to be evaluated from a yield/manufacturing point of view. Parasitic inductance limits the effectiveness of the decoupling capacitors, therefore physically smaller capacitors should be used while maintaining the largest available capacitance value. As with the selection of any component, verification of capacitor availability over the product's production lifetime should be considered.

3.1.6 IEEE 1149.1 JTAG Compatibility Statement

The TMS320C6414/15/16 DSP requires that both $\overline{\text{TRST}}$ and $\overline{\text{RESET}}$ be asserted upon power up to be properly initialized. While RESET initializes the DSP core, TRST initializes the DSP emulation logic. Both resets are required for proper operation.

While both $\overline{\text{TRST}}$ and $\overline{\text{RESET}}$ should be asserted upon power up, only $\overline{\text{RESET}}$ should 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 emulation logic in the reset state.

$\overline{\text{TRST}}$ only should be released when it is necessary to use a JTAG controller to debug the DSP or exercise the DSP boundary scan functionality.

For maximum reliability, the TMS320C6414/15/16 DSP includes an internal pulldown (IPD) on the $\overline{\text{TRST}}$ pin to ensure that $\overline{\text{TRST}}$ is always be asserted upon power up and the DSP internal emulation logic is always properly initialized.

JTAG controllers from TI actively drive $\overline{\text{TRST}}$ high. However, some third-party JTAG controllers may not drive $\overline{\text{TRST}}$ high, but expect the use of a pullup resistor on $\overline{\text{TRST}}$.

When using this type of JTAG controller, assert $\overline{\text{TRST}}$ to initialize the DSP after power up 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 be *seen* to latch the state of EMU1 and EMU0. The EMU[1:0] pins configure the device for either boundary scan mode or emulation mode. For more detailed information, see the terminal functions section of this data sheet.

3.1.7 EMIF Device Speed

The rated EMIF speed (referring to both EMIFA and EMIFB) of these devices only applies to the SDRAM interface when in a system that meets the following requirements:

- One bank (maximum of 2 chips) of SDRAM connected to EMIF
- Up to one bank of buffers connected to EMIF
- EMIF trace lengths between 1 inch and 3 inches
- 183-MHz SDRAM for 133-MHz operation (applies only to EMIFA)
- 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. 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 (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).

3.1.8 Boot Mode

The C6414/15/16 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 C6414/C6415/C6416 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 memory space as necessary through the host interface, including internal configuration registers, such as those that control the EMIF or other peripherals. For the C6414 device, the HPI peripheral is used for host boot. For the C6415/C6416 device, the HPI peripheral is used for host boot if PCI_EN = 0, and the PCI peripheral is used for host boot if PCI_EN = 1. Once the host is finished with all necessary initialization, it must set the DSPINT bit in the HPIC register to complete the boot process. This transition causes the boot configuration logic to bring the CPU out of the "stalled" state. The CPU then begins execution from address 0. The DSPINT condition is not latched by the CPU, because it occurs while the CPU is still internally "stalled". Also, DSPINT brings the CPU out of the "stalled" state only if the host boot process is selected. All memory may be written to and read by the host. This allows for the host to verify what it sends to the DSP if required. After the CPU is out of the "stalled" state, the CPU needs to clear the DSPINT, otherwise, no more DSPINTs can be received.
- EMIF boot (using default ROM timings)

  Upon the release of $\overline{\text{RESET}}$, the 1K-Byte ROM code located in the beginning of $\overline{\text{CE1}}$ is copied to

address 0 by the EDMA using the default ROM timings, while the CPU is internally "stalled". The data should be stored in the endian format that the system is using. In this case, the EMIF automatically assembles consecutive 8-bit bytes to form the 32-bit instruction words to be copied. The transfer is automatically done by the EDMA as a single-frame block transfer from the ROM to address 0. After completion of the block transfer, the CPU is released from the "stalled" state and starts running from address 0.

- No boot

  With no boot, the CPU begins direct execution from the memory located at address 0. If SDRAM is used in the system, the CPU is internally "stalled" until the SDRAM initialization is complete. Note: operation is undefined if invalid code is located at address 0.

4 Electrical Specifications

4.1 ABSOLUTE MAXIMUM RATINGS

over operating free-air temperature range (unless otherwise noted) (1)

			VALUE	UNIT
Supply voltage ranges		CV_{DD} (2)	–0.3 to 1.8	V
		DV_{DD} (2)	–0.3 to 4	V
V_I	Input voltage ranges	Except PCI	–0.3 to 4	V
V_{IP}		PCI (C6415 and C6416 only)	–0.5 to DV_{DD} + 0.5	V
V_O	Output voltage ranges	Except PCI	–0.3 V to 4	V
V_{OP}		PCI (C6415 and C6416 only)	–0.5 to DV_{DD} + 0.5	V
T_C	Operating case temperature ranges (A version) [C6414/15/16-EP]		–40 to 105	°C
	Operating case temperature ranges (S version) [C6415-EP]		–55 to 105	
T_{stg}	Storage temperature range		–65 to 150	°C

(1) 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.
(2) All voltage values are with respect to V_{SS}.

4.2 RECOMMENDED OPERATING CONDITIONS

		MIN	NOM	MAX	UNIT
CV_{DD}	Supply voltage, Core (-50xEP device) (1)	1.19	1.25	1.31	V
DV_{DD}	Supply voltage, I/O	3.14	3.3	3.46	V
V_{SS}	Supply ground	0	0	0	V
V_{IH}	High-level input voltage (except PCI)	2			V
V_{IL}	Low-level input voltage (except PCI)			0.8	V
V_{IP}	Input voltage (PCI) (C6415 and C6416 only)	–0.5		DV_{DD} + 0.5	V
V_{IHP}	High-level input voltage (PCI) (C6415 and C6416 only)	$0.5DV_{DD}$		DV_{DD} + 0.5	V
V_{ILP}	Low-level input voltage (PCI) (C6415 and C6416 only)	–0.5		$0.3DV_{DD}$	V
T_C	Operating case temperature (A version)	–40		105	°C
	Operating case temperature (S version)	–55		105	°C

(1) Future variants of the C641x 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 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 TI. Not incorporating a flexible supply may limit the system's ability to easily adapt to future versions of C641x devices.

4.3 ELECTRICAL CHARACTERISTICS

over operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage (except PCI)	DV_{DD} = MIN, I_{OH} = MAX	2.4			V
V_{OHP}	High-level output voltage (PCI) (C6415/C6416 only)	I_{OHP} = -0.5 mA, DV_{DD} = 3.3 V	$0.9DV_{DD}$ (1)			V
V_{OL}	Low-level output voltage (except PCI)	DV_{DD} = MIN, I_{OL} = MAX			0.4	V
V_{OLP}	Low-level output voltage (PCI) (C6415/C6416 only)	I_{OLP} = 1.5 mA, DV_{DD} = 3.3 V			$0.9DV_{DD}$ (1)	V

(1) These rated numbers are from the PCI specification version 2.3. The dc specification and ac specification are defined in Tables 4-3 and 4-4, respectively.

ELECTRICAL CHARACTERISTICS *(continued)*

over operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
I_I	Input current (except PCI)	$V_I = V_{SS}$ to DV_{DD} no opposing internal resistor			±10	µA
		$V_I = V_{SS}$ to DV_{DD} opposing internal pullup resistor (A-Temp)(2)	50	100	150	
		$V_I = V_{SS}$ to DV_{DD} opposing internal pullup resistor (S-Temp)(2)	50	100	160	
		$V_I = V_{SS}$ to DV_{DD} opposing internal pulldown resistor (A-Temp)(2)	–150	–100	–50	
		$V_I = V_{SS}$ to DV_{DD} opposing internal pulldown resistor (S-Temp)(2)	–160	–100	–50	
I_{IP}	Input leakage current (PCI) (C6415/C6416 only)(3)	$0 < V_{IP} < DV_{DD} = 3.3$ V			±50	µA
I_{OH}	High-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx			–16	mA
		Timer, UTOPIA, TDO, GPIO (excluding GP[15:9, 2, 1]), McBSP			–8	
		PCI/HPI			–0.5(1)	
I_{OL}	Low-level output current	EMIF, CLKOUT4, CLKOUT6, EMUx			16	mA
		Timer, UTOPIA, TDO, GPIO (excluding GP[15:9, 2, 1]), McBSP			8	
		PCI/HPI			1.5(1)	
I_{OZ}	Off-state output current	$V_O = DV_{DD}$ or 0 V			±10	µA
I_{CDD}	Core supply current(4)	CV_{DD} = 1.4 V, CPU clock = 720 MHz		900(5)		mA
		CV_{DD} = 1.4 V, CPU clock = 600 MHz		750(5)		
		CV_{DD} = 1.2 V, CPU clock = 500 MHz		550		
I_{DDD}	I/O supply current(4)	DV_{DD} = 3.3 V, CPU clock = 600 MHz		125(5)		mA
C_i	Input capacitance				10	pF
C_o	Output capacitance				10	pF

(2) Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.
(3) PCI input leakage currents include Hi-Z output leakage for all bidirectional buffers with 3-state outputs.
(4) Measured with average activity (50% high/50% low power). The actual current draw is highly application-dependent. For more details on core and I/O activity, refer to the TMS320C6414/15/16 Power Consumption Summary application report (SPRA811).
(5) Advance information

5 PARAMETER MEASUREMENT INFORMATION

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.

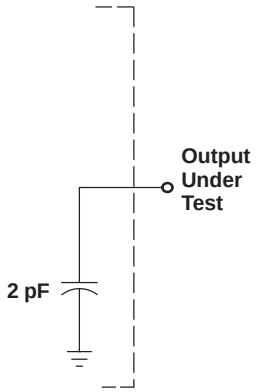


Figure 5-1. AC Timing Reference Circuit for AC Timing Measurements

5.1 Signal Transition Levels

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

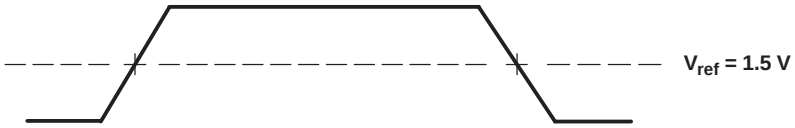


Figure 5-2. Input and Output Voltage Reference Levels for AC Timing Measurements

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, V_{ILP} MAX and V_{IHP} MIN for PCI input clocks, and V_{OLP} MAX and V_{OHP} MIN for PCI output clocks.

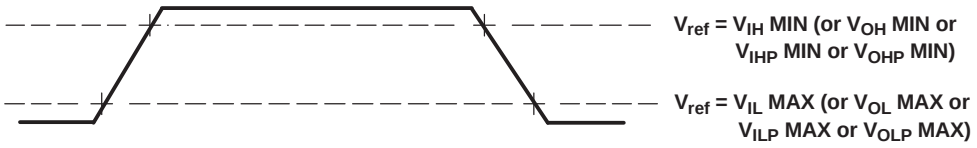


Figure 5-3. Rise and Fall Transition Time Voltage Reference Levels

5.2 Signal Transition Rates

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

5.3 Timing Parameters and Board Routing Analysis

The timing parameter values specified in this data sheet do *not* include delays by board routing. 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. 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 5-1 and Figure 5-4).

Figure 5-4 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 5-1. Board-Level Timing (see Figure 5-4)

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

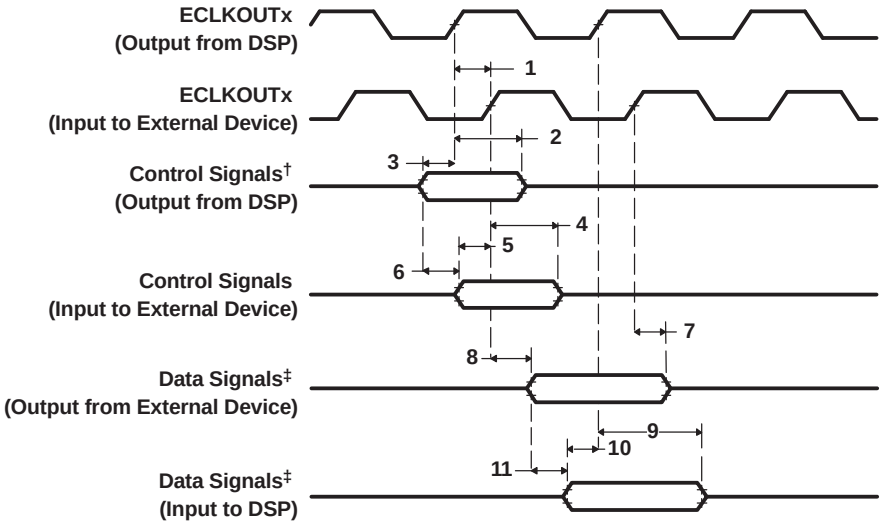


Figure 5-4. Board-Level Input/Output Timing

5.1 INPUT AND OUTPUT CLOCKS

5.1.1 Timing Requirements for CLKIN for –50xEP Devices[1] [2] [3] (see Figure 5-5)

NO.			–50xEP						UNIT
			PLL MODE x12		PLL MODE x6		x1 (BYPASS)		
			MIN	MAX	MIN	MAX	MIN	MAX	
1	$t_{c(CLKIN)}$	Cycle time, CLKIN	24	33.3	13.3	33.3	13.3	33.3	ns

(1) The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
(2) For more details on the PLL multiplier factors (x6, x12), see the *Clock PLL* section of this data manual.
(3) C = CLKIN cycle time in ns. For example, when CLKIN frequency is 50 MHz, use C = 20 ns.

NO.			–50xEP						UNIT
			PLL MODE x12		PLL MODE x6		x1 (BYPASS)		
			MIN	MAX	MIN	MAX	MIN	MAX	
2	$t_{w(CLKINH)}$	Pulse duration, CLKIN high	0.4C		0.4C		0.45C		ns
3	$t_{w(CLKINL)}$	Pulse duration, CLKIN low	0.4C		0.4C		0.45C		ns
4	$t_{t(CLKIN)}$	Transition time, CLKIN		5		5		1	ns
5	$t_{J(CLKIN)}$	Peak-to-peak jitter, CLKIN		0.02C		0.02C		0.02C	ns

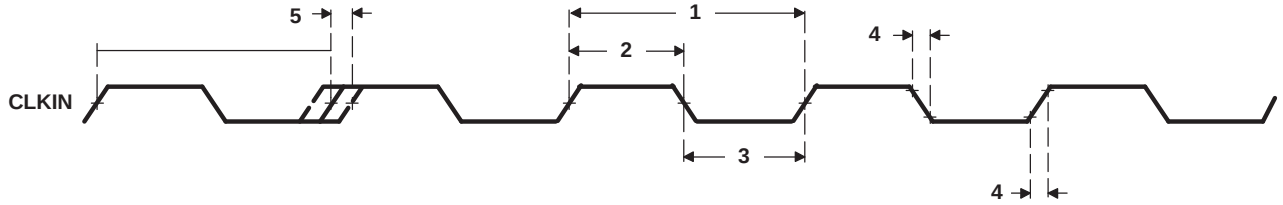


Figure 5-5. CLKIN Timing

NO.	PARAMETER		–50xEP		UNIT
			CLKMODE = x1, x6, x12		
			MIN	MAX	
1	$t_{J(CKO4)}$	Cycle-to-cycle jitter, CLKOUT4	0	±175	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

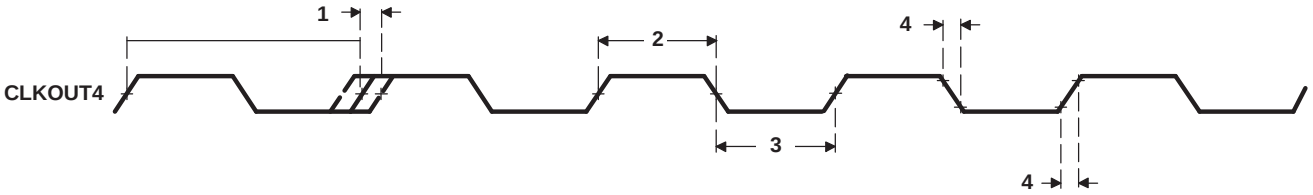


Figure 5-6. CLKOUT4 Timing

NO.	PARAMETER		–50xEP		UNIT
			CLKMODE = x1, x6, x12		
			MIN	MAX	
1	$t_{J(CKO6)}$	Cycle-to-cycle jitter, CLKOUT6	0	±175	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

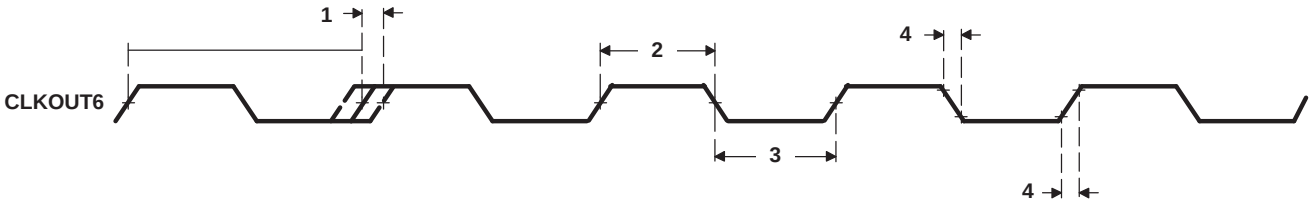


Figure 5-7. CLKOUT6 Timing

5.1.4 Timing Requirements ECLKIN for EMIFA and EMIFB (1) (2) (3) (see Figure 5-8)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{c(EKI)}$	Cycle time, ECLKIN	6(4)	16P	ns
2	$t_{w(EKIH)}$	Pulse duration, ECLKIN high	2.7		ns
3	$t_{w(EKIL)}$	Pulse duration, ECLKIN low	2.7		ns
4	$t_{t(EKI)}$	Transition time, ECLKIN		2	ns
5	$t_{J(EKI)}$	Peak-to-Peak jitter, ECLKIN		0.02E	ns

(1) P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
(2) The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
(3) These C64xE devices have two EMIFs (64-bit EMIFA and 16-bit EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted.
(4) Minimum ECLKIN times are based on internal logic speed; the maximum useable speed of the EMIF may be lower due to ac timing requirements. On the 7E3 and 6E3 devices, 133-MHz operation is achievable if the requirements of the EMIF Device Speed section are met. On the 5E0 devices, 100-MHz operation is achievable if the requirements of the EMIF Device Speed section are met.

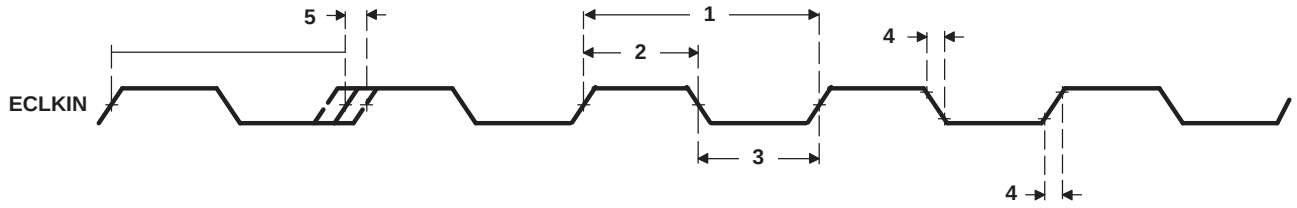


Figure 5-8. ECLKIN Timing for EMIFA and EMIFB

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{J(EKO1)}$	Cycle-to-cycle jitter, ECLKOUT1	0	±175(1)	ns
2	$t_{w(EKO1H)}$	Pulse duration, ECLKOUT1 high	ER – 0.7	EH + 0.7	ns
3	$t_{w(EKO1L)}$	Pulse duration, ECLKOUT1 low	EL – 0.7	EL + 0.7	ns
4	$t_{t(EKO1)}$	Transition time, ECLKOUT1		1	ns
5	$t_{d(EKIH-EKO1H)}$	Delay time, ECLKIN high to ECLKOUT1 high	1	8	ns
6	$t_{d(EKIL-EKO1L)}$	Delay time, ECLKIN low to ECLKOUT1 low	1	8	ns

(1) This cycle-to-cycle jitter specification was measured with CPU/4 or CPU/6 as the source of the EMIF input clock.

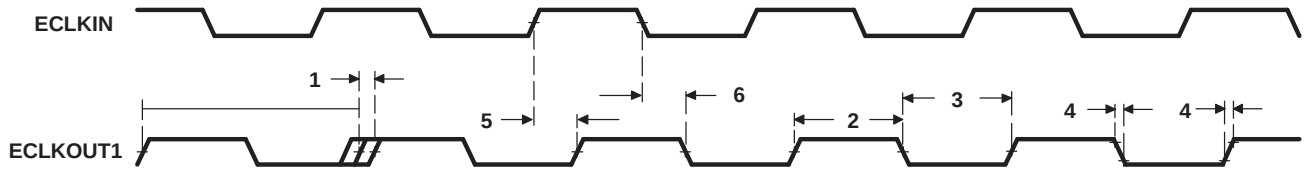


Figure 5-9. ECLKOUT1 Timing for EMIFA and EMIFB Modules

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{J(EKO2)}$	Cycle-to-cycle jitter, ECLKOUT2	0	±175(1)	ns
2	$t_{w(EKO2H)}$	Pulse duration, ECLKOUT2 high	05NE – 0.7	05NE + 0.7	ns
3	$t_{w(EKO2L)}$	Pulse duration, ECLKOUT2 low	05NE – 0.7	05NE + 0.7	ns
4	$t_{t(EKO2)}$	Transition time, ECLKOUT2		1	ns
5	$t_{d(EKIH-EKO2H)}$	Delay time, ECLKIN high to ECLKOUT2 high	1	8	ns
6	$t_{d(EKIL-EKO2L)}$	Delay time, ECLKIN low to ECLKOUT2 low	1	8	ns

(1) This cycle-to-cycle jitter specification was measured with CPU/4 or CPU/6 as the source of the EMIF input clock.

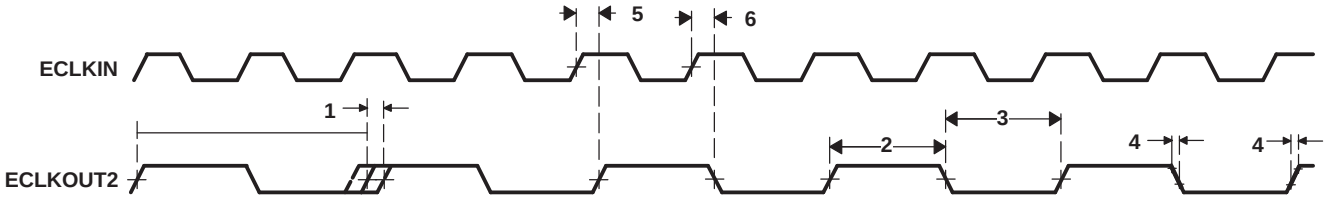


Figure 5-10. ECLKOUT2 Timing for EMIFA and EMIFB Modules

5.2 ASYNCHRONOUS MEMORY TIMING

5.2.1 Timing Requirements for Asynchronous Memory Cycles for EMIFA Module (1) (2) (3) (see Figure 5-11 and Figure 5-12)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
3	$t_{su(EDV-AREH)}$	Setup time, EDx valid before $\overline{ARE}$ high	6.5		ns
4	$t_{h(AREH-EDV)}$	Hold time, EDx valid after $\overline{ARE}$ high	1		ns
6	$t_{su(ARDY-EKO1H)}$	Setup time, ARDY valid before ECLKOUT1 high	3		ns
7	$t_{h(EKO1H-ARDY)}$	Hold time, ARDY valid after ECLKOUT1 high	1.5		ns

(1) To ensure data setup time, program the strobe width wide enough. ARDY is internally synchronized. The ARDY is recognized in the cycle for which the setup and hold time is met. To use ARDY as an asynchronous input, the pulse width of ARDY signal should be wide enough (e.g., pulse width = 2E) to ensure setup and hold time is met.o
(2) 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.
(3) These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the asynchronous memory access signals are shown as generic ($\overline{AOE}$, $\overline{ARE}$, and $\overline{AWE}$) instead of $\overline{AAOE}$, $\overline{AARE}$, and $\overline{AAWE}$ (for EMIFA) and $\overline{BAOE}$, $\overline{BARE}$, and $\overline{BAWE}$ (for EMIFB)].

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{osu(SELV-AREL)}$	Output setup time, select signals valid to $\overline{ARE}$ low	RS × E-1.5		ns
2	$t_{oh(AREH-SELIV)}$	Output hold time, $\overline{ARE}$ high to select signals invalid	RH × E-1.9		ns
5	$t_{d(EKO1H-AREV)}$	Delay time, ECLKOUT1 high to $\overline{ARE}$ valid	1	7	ns
8	$t_{osu(SELV-AWEL)}$	Output setup time, select signals valid to $\overline{AWE}$ low	WS × E-1.7		ns
9	$t_{oh(AWEH-SELIV)}$	Output hold time, $\overline{AWE}$ high to select signals invalid	WH × E-1.8		ns
10	$t_{d(EKO1H-AWEV)}$	Delay time, ECLKOUT1 high to $\overline{AWE}$ valid	1.3	7.1	ns

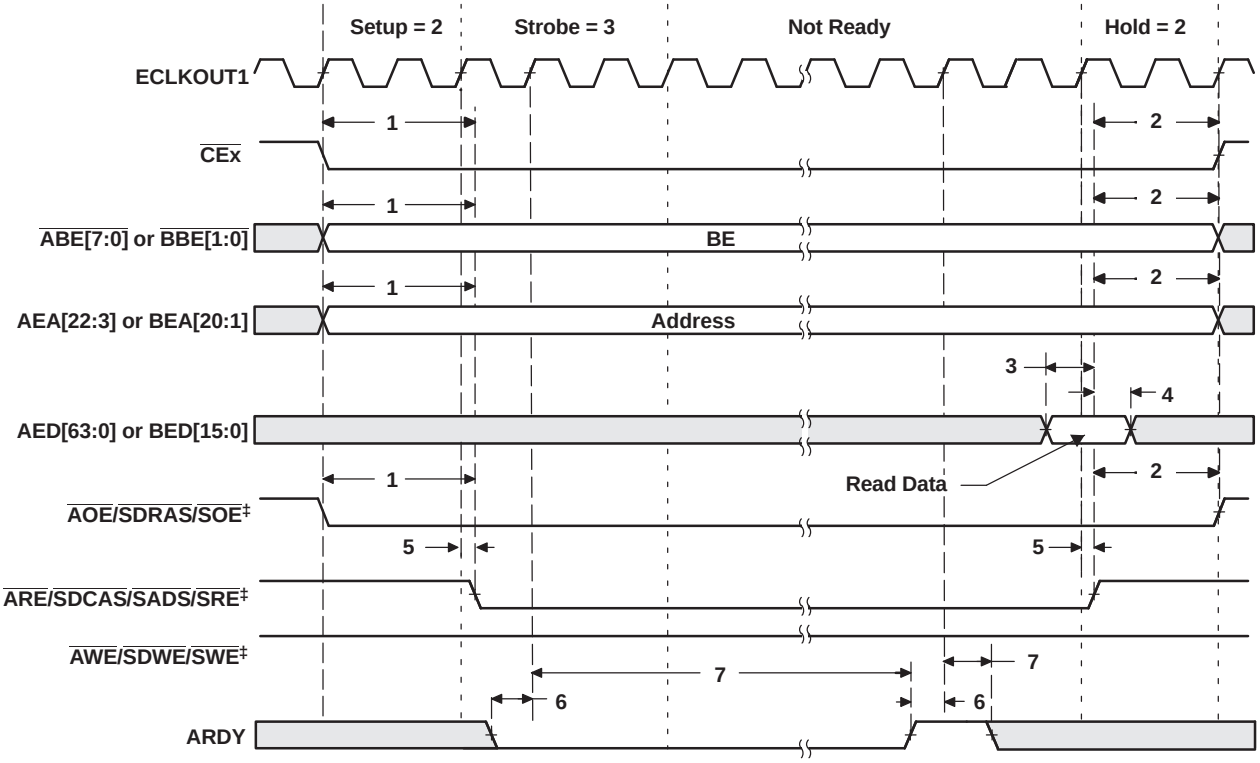
5.2.3 Timing Requirements for Asynchronous Memory Cycles for EMIFB Module (1) (2) (3) (see Figure 5-11 and Figure 5-12)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
3	$t_{su(EDV-AREH)}$	Setup time, EDx valid before $\overline{ARE}$ high	6.2		ns
4	$t_{h(AREH-EDV)}$	Hold time, EDx valid after $\overline{ARE}$ high	1		ns

(1) To ensure data setup time, program the strobe width wide enough. ARDY is internally synchronized. The ARDY is recognized in the cycle for which the setup and hold time is met. To use ARDY as an asynchronous input, the pulse width of ARDY signal should be wide enough (e.g., pulse width = 2E) to ensure setup and hold time is met.o
(2) 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.
(3) These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the asynchronous memory access signals are shown as generic ($\overline{AOE}$, $\overline{ARE}$, and $\overline{AWE}$) instead of $\overline{AAOE}$, $\overline{AARE}$, and $\overline{AAWE}$ (for EMIFA) and $\overline{BAOE}$, $\overline{BARE}$, and $\overline{BAWE}$ (for EMIFB)].

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{su(ARDY-EKO1H)}$	Setup time, ARDY valid before ECLKOUT1 high	3		ns
7	$t_{h(EKO1H-ARDY)}$	Hold time, ARDY valid after ECLKOUT1 high	1.7		ns

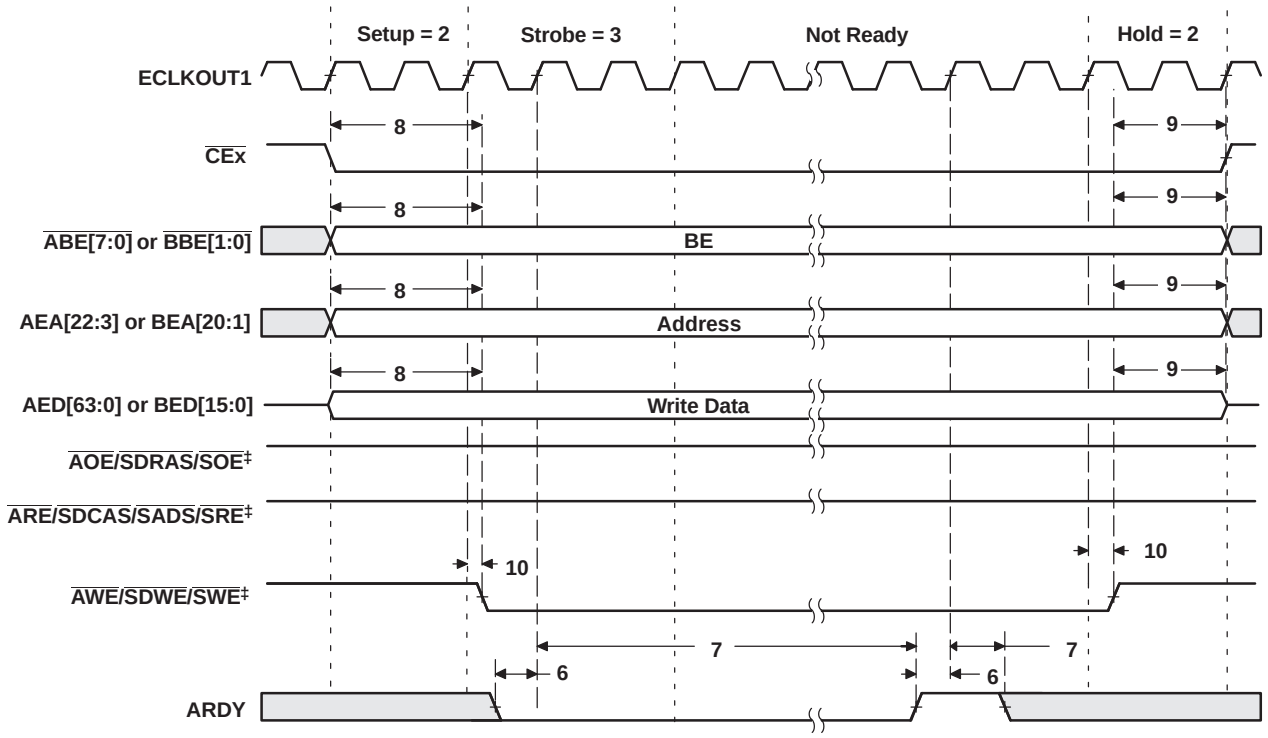
NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{osu(SELV-AREL)}$	Output setup time, select signals valid to $\overline{ARE}$ low	RS × E-1.6		ns
2	$t_{oh(AREH-SELIV)}$	Output hold time, $\overline{ARE}$ high to select signals invalid	RH × E-1.7		ns
5	$t_{d(EKO1H-AREV)}$	Delay time, ECLKOUT1 high to $\overline{ARE}$ valid	0.8	6.6	ns
8	$t_{osu(SELV-AWEL)}$	Output setup time, select signals valid to $\overline{AWE}$ low	WS × E-1.9		ns
9	$t_{oh(AWEH-SELIV)}$	Output hold time, $\overline{AWE}$ high to select signals invalid	WH × E-1.7		ns
10	$t_{d(EKO1H-AWEV)}$	Delay time, ECLKOUT1 high to $\overline{AWE}$ valid	0.9	6.7	ns



NOTE: These C64xE devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the asynchronous memory access signals are shown as generic ($\overline{AOE}$, $\overline{ARE}$, and $\overline{AWE}$) instead of $\overline{AAOE}$, $\overline{AARE}$, and $\overline{AAWE}$ (for EMIFA) and $\overline{BAOE}$, $\overline{BARE}$, and $\overline{BAWE}$ (for EMIFB)].

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

Figure 5-11. Asynchronous Memory Read Timing for EMIFA and EMIFB (see NOTE NoLabel)



NOTE: These C64xE devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the asynchronous memory access signals are shown as generic ($\overline{AOE}$, $\overline{ARE}$, and $\overline{AWE}$) instead of $\overline{AAOE}$, $\overline{AARE}$, and $\overline{AAWE}$ (for EMIFA) and $\overline{BAOE}$, $\overline{BARE}$, and $\overline{BAWE}$ (for EMIFB)].

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

Figure 5-12. Asynchronous Memory Write Timing for EMIFA and EMIFB (see NOTE [(A)])

5.3 PROGRAMMABLE SYNCHRONOUS INTERFACE TIMING

5.3.1 Timing Requirements for Programmable Synchronous Interface Cycles for EMIFA Module [(1)] (see Figure 5-13)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKOxH)}$	Setup time, read EDx valid before ECLKOUTx high	3.1		ns
7	$t_{h(EKOxH-EDV)}$	Hold time, read EDx valid after ECLKOUTx high	1.5		ns

(1) These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the programmable synchronous interface access signals are shown as generic ($\overline{SADS}$/$\overline{SRE}$, $\overline{SOE}$, and $\overline{SWE}$) instead of $\overline{ASADS}$/$\overline{ASRE}$, $\overline{ASOE}$, and $\overline{ASWE}$ (for EMIFA) and $\overline{BSADS}$/$\overline{BSRE}$, $\overline{BSOE}$, and $\overline{BSWE}$ (for EMIFB)].

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(EKOxH-CEV)}$	Delay time, ECLKOUTx high to $\overline{CEx}$ valid	1.3	6.4	ns
2	$t_{d(EKOxH-BEV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ valid		6.4	ns
3	$t_{d(EKOxH-BEIV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ invalidE	1.3		ns
4	$t_{d(EKOxH-EAV)}$	Delay time, ECLKOUTx high to EAx valid		6.4	ns
5	$t_{d(EKOxH-EAIV)}$	Delay time, ECLKOUTx high to EAx invalid	1.3		ns

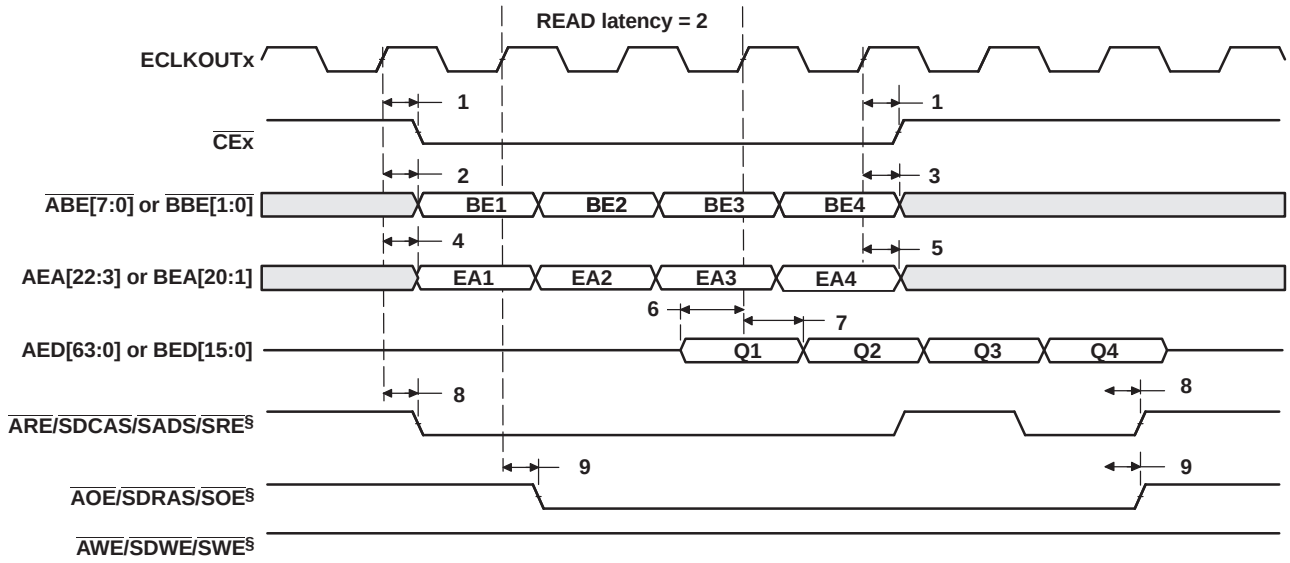
NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
8	$t_{d(EKOxH-ADSV)}$	Delay time, ECLKOUTx high to $\overline{SADS}$/$\overline{SRE}$ valid	1.3	6.4	ns
9	$t_{d(EKOxH-OEV)}$	Delay time, ECLKOUTx high to, $\overline{SOE}$ validE	1.3	6.4	ns
10	$t_{d(EKOxH-EDV)}$	Delay time, ECLKOUTx high to $\overline{EDx}$ valid		6.4	ns
11	$t_{d(EKOxH-EDIV)}$	Delay time, ECLKOUTx high to $\overline{EDx}$ invalidE	1.3		ns
12	$t_{d(EKOxH-WEV)}$	Delay time, ECLKOUTx high to $\overline{SWE}$ valid	1.3	6.4	ns

5.3.3 Timing Requirements for Programmable Synchronous Interface Cycles for EMIFB Module [1] (see Figure 5-13)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKOxH)}$	Setup time, read EDx valid before ECLKOUTx high	3.1		ns
7	$t_{h(EKOxH-EDV)}$	Hold time, read EDx valid after ECLKOUTx high	1.5		ns

(1) These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the programmable synchronous interface access signals are shown as generic ($\overline{SADS}$/$\overline{SRE}$, $\overline{SOE}$, and $\overline{SWE}$) instead of $\overline{ASADS}$/$\overline{ASRE}$, $\overline{ASOE}$, and $\overline{ASWE}$ (for EMIFA) and $\overline{BSADS}$/$\overline{BSRE}$, $\overline{BSOE}$, and $\overline{BSWE}$ (for EMIFB)].

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(EKOxH-CEV)}$	Delay time, ECLKOUTx high to $\overline{CEx}$ valid	1.3	6.4	ns
2	$t_{d(EKOxH-BEV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ valid		6.4	ns
3	$t_{d(EKOxH-BEIV)}$	Delay time, ECLKOUTx high to $\overline{BEx}$ invalidE	1.3		ns
4	$t_{d(EKOxH-EAV)}$	Delay time, ECLKOUTx high to EAx valid		6.4	ns
5	$t_{d(EKOxH-EAIV)}$	Delay time, ECLKOUTx high to EAx invalid	1.3		ns
8	$t_{d(EKOxH-ADSV)}$	Delay time, ECLKOUTx high to $\overline{SADS}$/$\overline{SRE}$ valid	1.3	6.4	ns
9	$t_{d(EKOxH-OEV)}$	Delay time, ECLKOUTx high to, $\overline{SOE}$ validE	1.3	6.4	ns
10	$t_{d(EKOxH-EDV)}$	Delay time, ECLKOUTx high to $\overline{EDx}$ valid		6.4	ns
11	$t_{d(EKOxH-EDIV)}$	Delay time, ECLKOUTx high to $\overline{EDx}$ invalidE	1.3		ns
12	$t_{d(EKOxH-WEV)}$	Delay time, ECLKOUTx high to $\overline{SWE}$ valid	1.3	6.4	ns



A. These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the programmable synchronous interface access signals are shown as generic ($\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{SOE}}$, and $\overline{\text{SWE}}$) instead of $\overline{\text{ASADS}}$/$\overline{\text{ASRE}}$, $\overline{\text{ASOE}}$, and $\overline{\text{ASWE}}$ (for EMIFA) and $\overline{\text{BSADS}}$/$\overline{\text{BSRE}}$, $\overline{\text{BSOE}}$, and $\overline{\text{BSWE}}$ (for EMIFB)].

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

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

   a. Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency.

   b. Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency

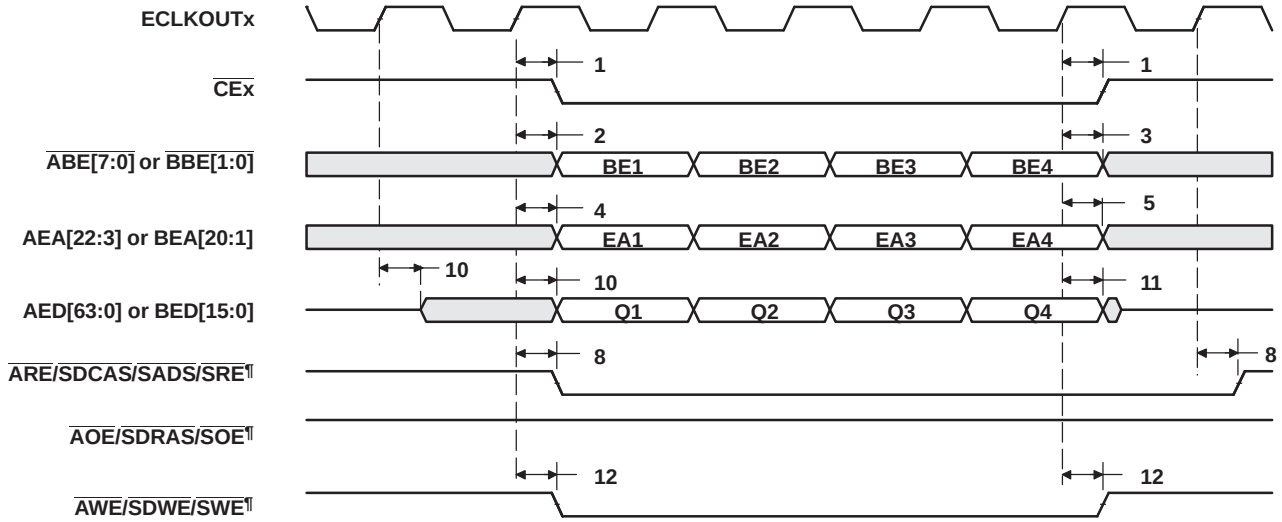
   c. $\overline{\text{CEx}}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{CEx}}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{\text{CEx}}$ is active when $\overline{\text{SOE}}$ is active (CEEXT = 1).

   d. Function of $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SADS}}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SRE}}$ with NO deselect cycles (RENEN = 1).

   e. Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

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

Figure 5-13. Programmable Synchronous Interface Read Timing for EMIFA and EMIFB (With Read Latency = 2) (See Notes A, B, C)



A. These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the programmable synchronous interface access signals are shown as generic ($\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{SOE}}$, and $\overline{\text{SWE}}$) instead of $\overline{\text{ASADS}}$/$\overline{\text{ASRE}}$, $\overline{\text{ASOE}}$, and $\overline{\text{ASWE}}$ (for EMIFA) and $\overline{\text{BSADS}}$/$\overline{\text{BSRE}}$, $\overline{\text{BSOE}}$, and $\overline{\text{BSWE}}$ (for EMIFB)].

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

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

   a. Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency.

   b. Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency

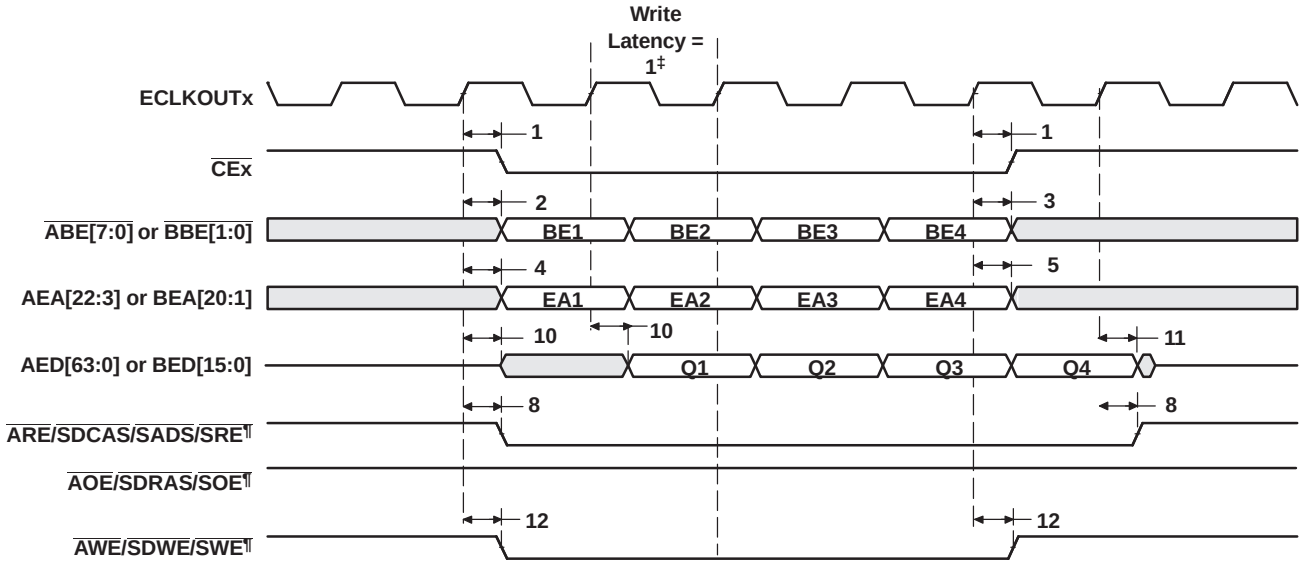
   c. $\overline{\text{CEx}}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{CEx}}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{\text{CEx}}$ is active when $\overline{\text{SOE}}$ is active (CEEXT = 1).

   d. Function of $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SADS}}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SRE}}$ with NO deselect cycles (RENEN = 1).

   e. Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

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

Figure 5-14. Programmable Synchronous Interface Write Timing for EMIFA and EMIFB (With Write Latency = 0) (See Notes A, B, C)



A. These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the programmable synchronous interface access signals are shown as generic ($\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{SOE}}$, and $\overline{\text{SWE}}$) instead of $\overline{\text{ASADS}}$/$\overline{\text{ASRE}}$, $\overline{\text{ASOE}}$, and $\overline{\text{ASWE}}$ (for EMIFA) and $\overline{\text{BSADS}}$/$\overline{\text{BSRE}}$, $\overline{\text{BSOE}}$, and $\overline{\text{BSWE}}$ (for EMIFB)].

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

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

   a. Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency.

   b. Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency

   c. $\overline{\text{CEx}}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{CEx}}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, $\overline{\text{CEx}}$ is active when $\overline{\text{SOE}}$ is active (CEEXT = 1).

   d. Function of $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SADS}}$ with deselect cycles (RENEN = 0). For FIFO interface, $\overline{\text{SADS}}$/$\overline{\text{SRE}}$ acts as $\overline{\text{SRE}}$ with NO deselect cycles (RENEN = 1).

   e. Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

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

Figure 5-15. Programmable Synchronous Interface Write Timing for EMIFA and EMIFB (With Write Latency = 1) See Notes A, B, C)

5.4 SYNCHRONOUS DRAM TIMING

5.4.1 Timing Requirements for Synchronous DRAM Cycles for EMIFA Module (see Figure 5-16

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKO1H)}$	Setup time, read EDx valid before ECLKOUT1 high	2.1		ns
7	$t_{h(EKO1H-EDV)}$	Hold time, read EDx valid after ECLKOUT1 high	2.5		ns

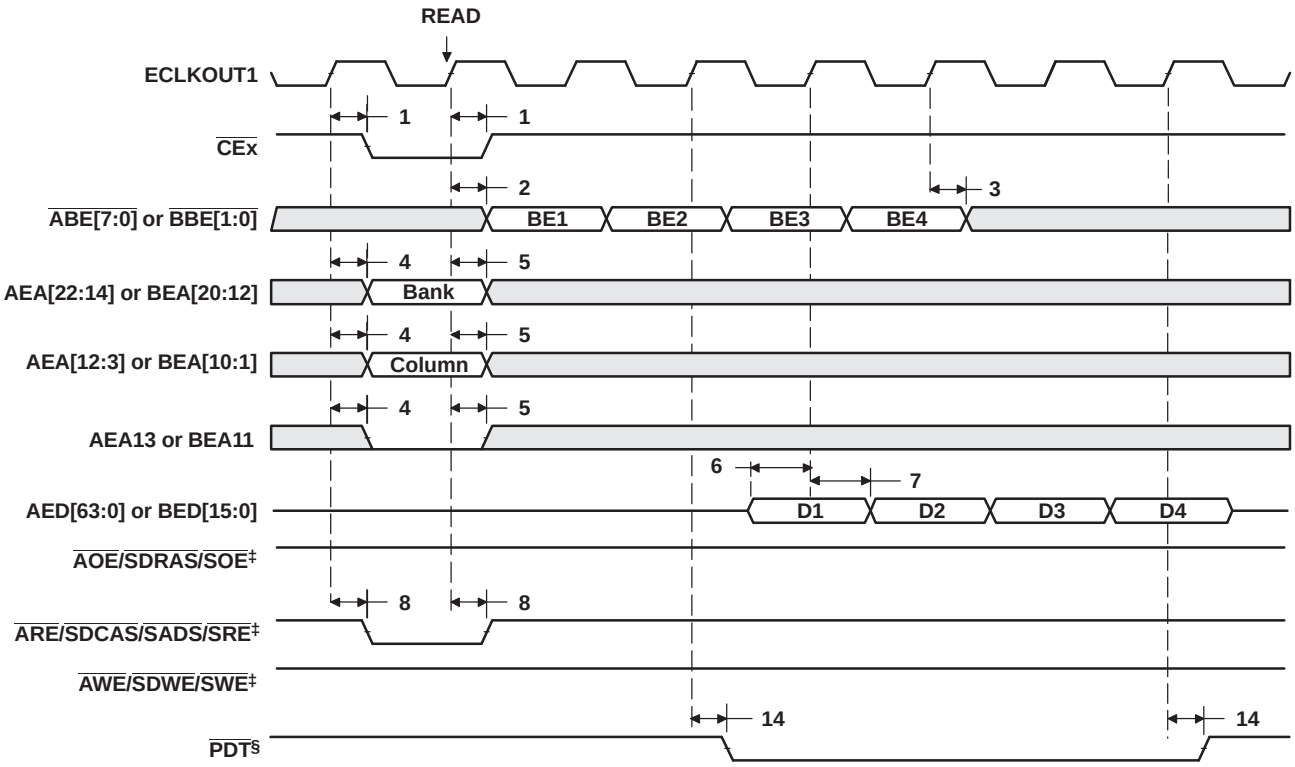
NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(EKO1H-CEV)}$	Delay time, ECLKOUT1 high to $\overline{CEx}$ valid	1.3	6.4	ns
2	$t_{d(EKO1H-BEV)}$	Delay time, ECLKOUT1 high to $\overline{BEx}$ valid		6.4	ns
3	$t_{d(EKO1H-BEIV)}$	Delay time, ECLKOUT1 high to $\overline{BEx}$ invalid	1.3		ns
4	$t_{d(EKO1H-EAV)}$	Delay time, ECLKOUT1 high to EAx valid		6.4	ns
5	$t_{d(EKO1H-EAIV)}$	Delay time, ECLKOUT1 high to EAx invalid	1.3		ns
8	$t_{d(EKO1H-CASV)}$	Delay time, ECLKOUT1 high to $\overline{SDCAS}$ valid	1.3	6.4	ns
9	$t_{d(EKO1H-EDV)}$	Delay time, ECLKOUT1 high to $\overline{EDx}$ valid		6.4	ns
10	$t_{d(EKO1H-EDIV)}$	Delay time, ECLKOUT1 high to $\overline{EDx}$ invalid	1.3		ns
11	$t_{d(EKO1H-WEV)}$	Delay time, ECLKOUT1 high to $\overline{SDWE}$ valid	1.3	6.4	ns
12	$t_{d(EKO1H-RAS)}$	Delay time, ECLKOUT1 high to $\overline{SDRAS}$ valid	1.3	6.4	ns
13	$t_{d(EKO1H-ACKEV)}$	Delay time, ECLKOUT1 high to ASDCKE valid (EMIFA only)	1.3	6.4	ns
14	$t_{d(EKO1H-PDTV)}$	Delay time, ECLKOUT1 high to $\overline{PDT}$ valid	1.3	6.4	ns

5.4.3 Timing Requirements for Synchronous DRAM Cycles for EMIFB Module[1](see Figure 5-16)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{su(EDV-EKO1H)}$	Setup time, read EDx valid before ECLKOUT1 high	2.1		ns
7	$t_{h(EKO1H-EDV)}$	Hold time, read EDx valid after ECLKOUT1 high	2.5		ns

(1) These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{SDCAS}$, $\overline{SDWE}$, and $\overline{SDRAS}$) instead of $\overline{ASDCAS}$, $\overline{ASDWE}$, and $\overline{ASDRAS}$ (for EMIFA) and $\overline{BSDCAS}$, $\overline{BSDWE}$, and $\overline{BSDRAS}$ (for EMIFB)].

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(EKO1H-CEV)}$	Delay time, ECLKOUT1 high to $\overline{CEx}$ valid	1.3	6.4	ns
2	$t_{d(EKO1H-BEV)}$	Delay time, ECLKOUT1 high to $\overline{BEx}$ valid		6.4	ns
3	$t_{d(EKO1H-BEIV)}$	Delay time, ECLKOUT1 high to $\overline{BEx}$ invalid	1.3		ns
4	$t_{d(EKO1H-EAV)}$	Delay time, ECLKOUT1 high to EAx valid		6.4	ns
5	$t_{d(EKO1H-EAIV)}$	Delay time, ECLKOUT1 high to EAx invalid	1.3		ns
8	$t_{d(EKO1H-CASV)}$	Delay time, ECLKOUT1 high to $\overline{SDCAS}$ valid	1.3	6.4	ns
9	$t_{d(EKO1H-EDV)}$	Delay time, ECLKOUT1 high to $\overline{EDx}$ valid		6.4	ns
10	$t_{d(EKO1H-EDIV)}$	Delay time, ECLKOUT1 high to $\overline{EDx}$ invalidE	1.3		ns
11	$t_{d(EKO1H-WEV)}$	Delay time, ECLKOUT1 high to $\overline{SDWE}$ valid	1.3	6.4	ns
12	$t_{d(EKO1H-RAS)}$	Delay time, ECLKOUT1 high to $\overline{SDRAS}$ valid	1.3	6.4	ns
13	$t_{d(EKO1H-ACKEV)}$	Delay time, ECLKOUT1 high to ASDCKE valid (EMIFA only)	1.3	6.4	ns
14	$t_{d(EKO1H-PDTV)}$	Delay time, ECLKOUT1 high to $\overline{PDT}$ valid	1.3	6.4	ns

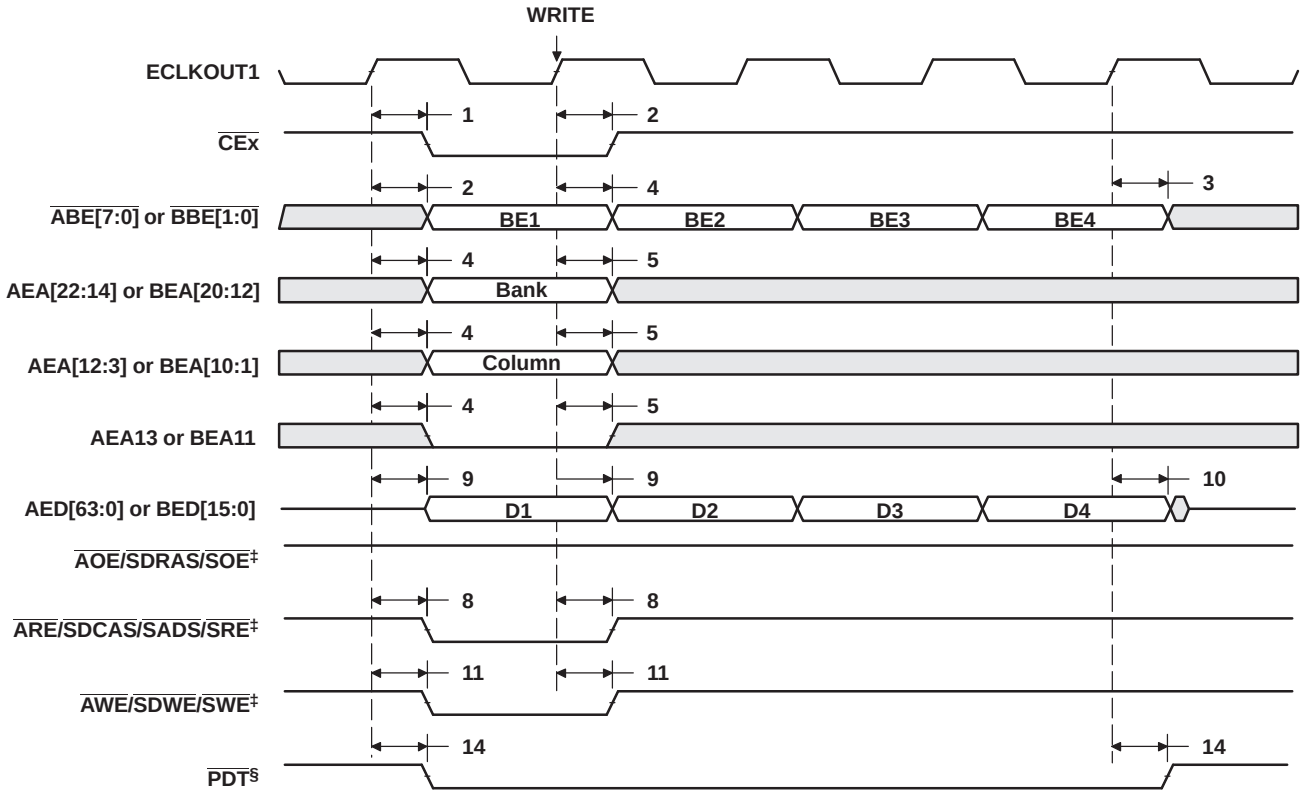


NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

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

B. $\overline{\text{PDT}}$ signal is only asserted when the EDMA is in PDT mode (set the PDTS bit to 1 in the EDMA options parameter RAM). For $\overline{\text{PDT}}$ read, data is not latched into EMIF. The PDTRAL field in the PDT cotnrol register (PDTCTL) configures the latency of the $\overline{\text{PDT}}$ signal with respect ot the data phase of read transaction. The latency of the $\overline{\text{PDT}}$ 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 22.

Figure 5-16. SDRAM Read Command (CAS Latency 3) for EMIFA and EMIFB (see Note)

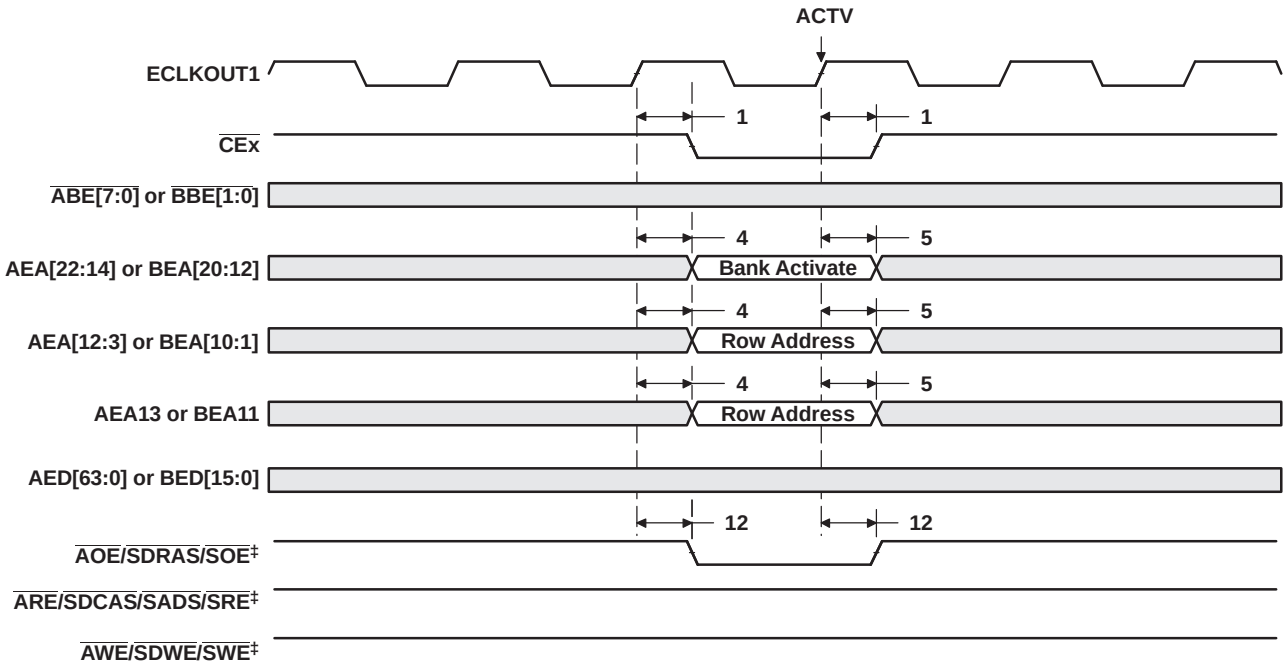


NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

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

B. $\overline{\text{PDT}}$ 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 PDT control register (PDTCTL) configures the latency of the $\overline{\text{PDT}}$ signal with respect ot the data phase of write transaction. The latency of the $\overline{\text{PDT}}$ signal for a write 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 23.

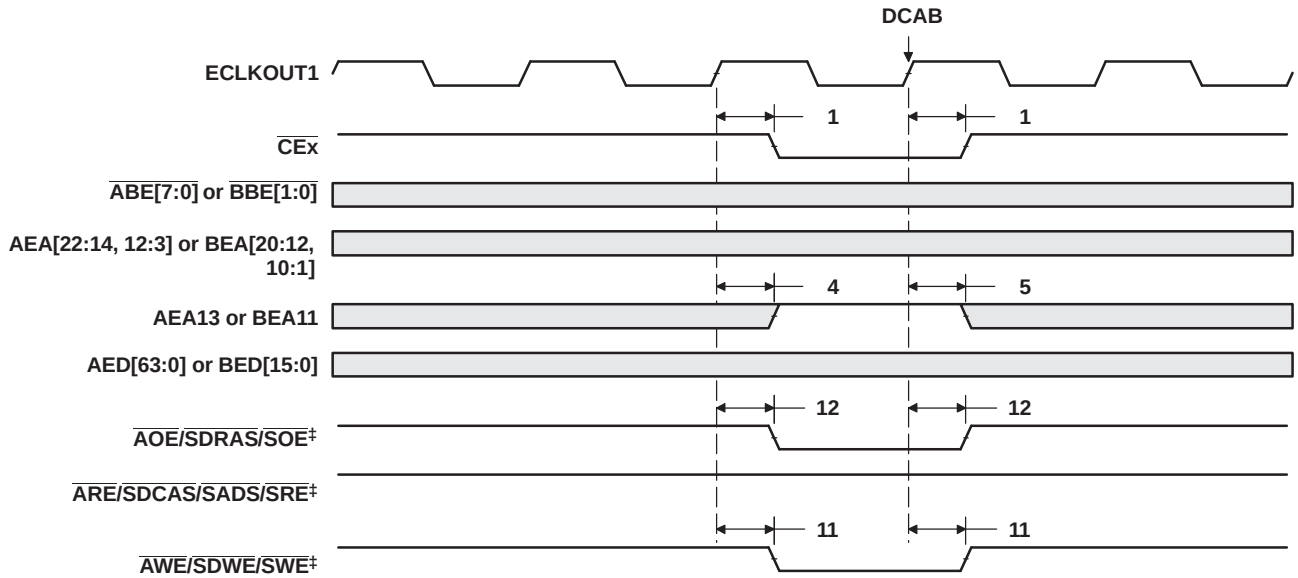
Figure 5-17. SDRAM Write Command for EMIFA and EMIFB (see Note)



NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

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

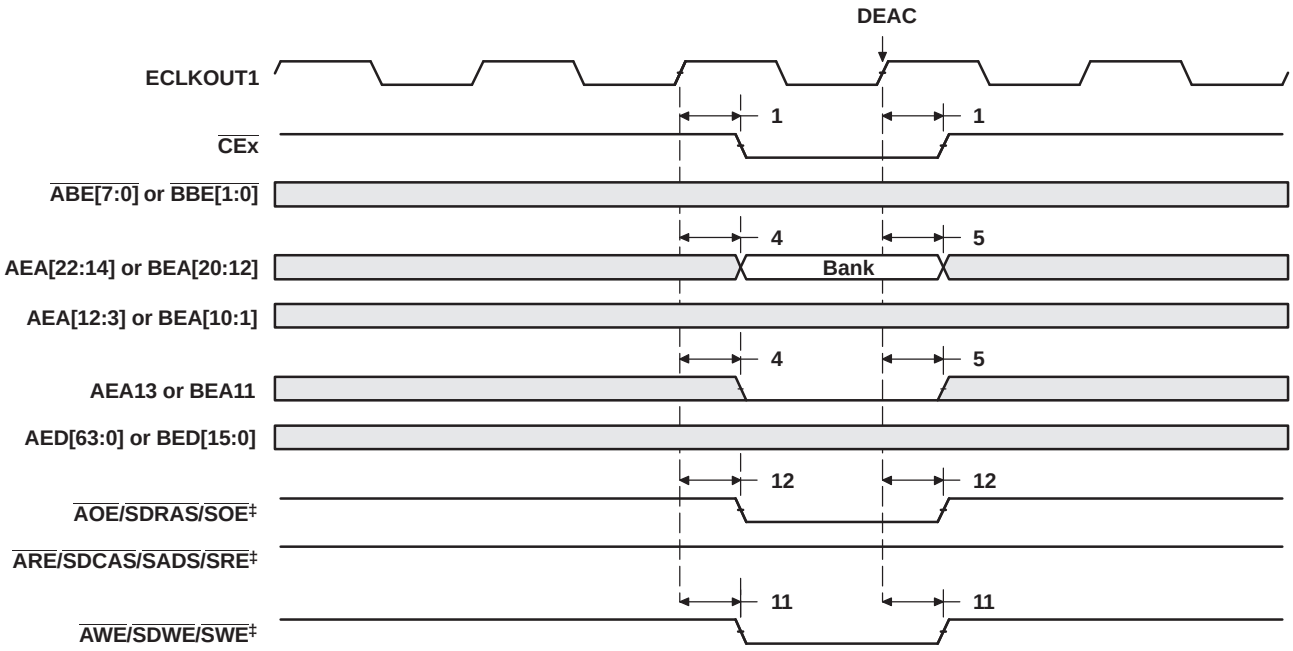
Figure 5-18. SDRAM ACTV Command for EMIFA and EMFB (see Note)



NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

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

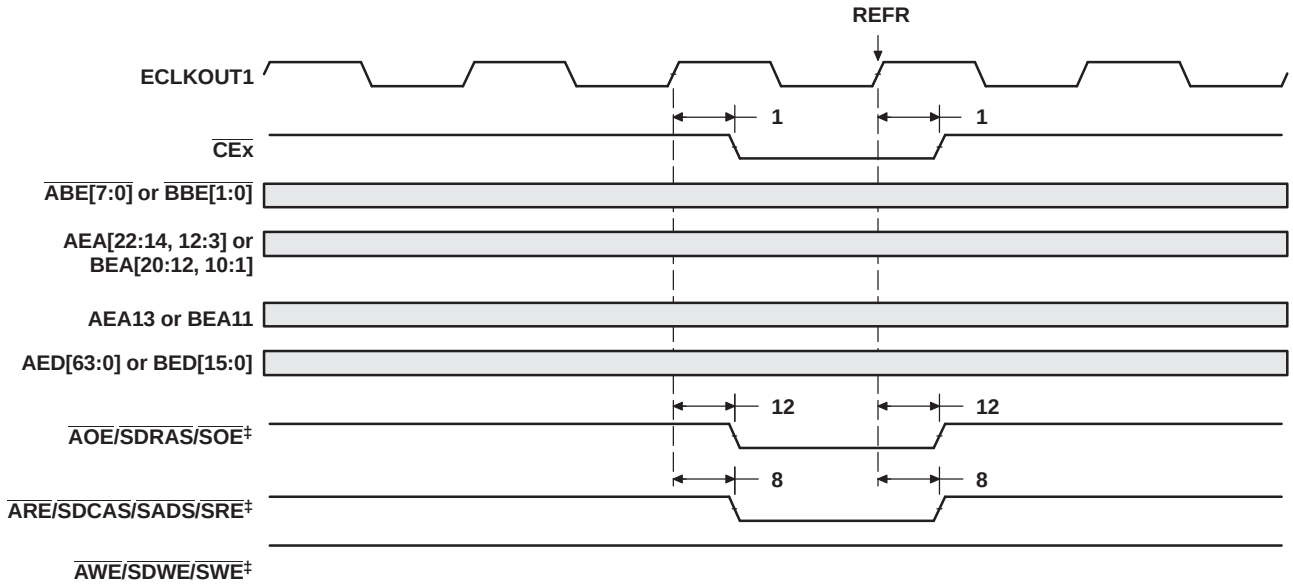
Figure 5-19. SDRAM DCAB Command for EMIFA and EMIFB (see Note)



NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

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

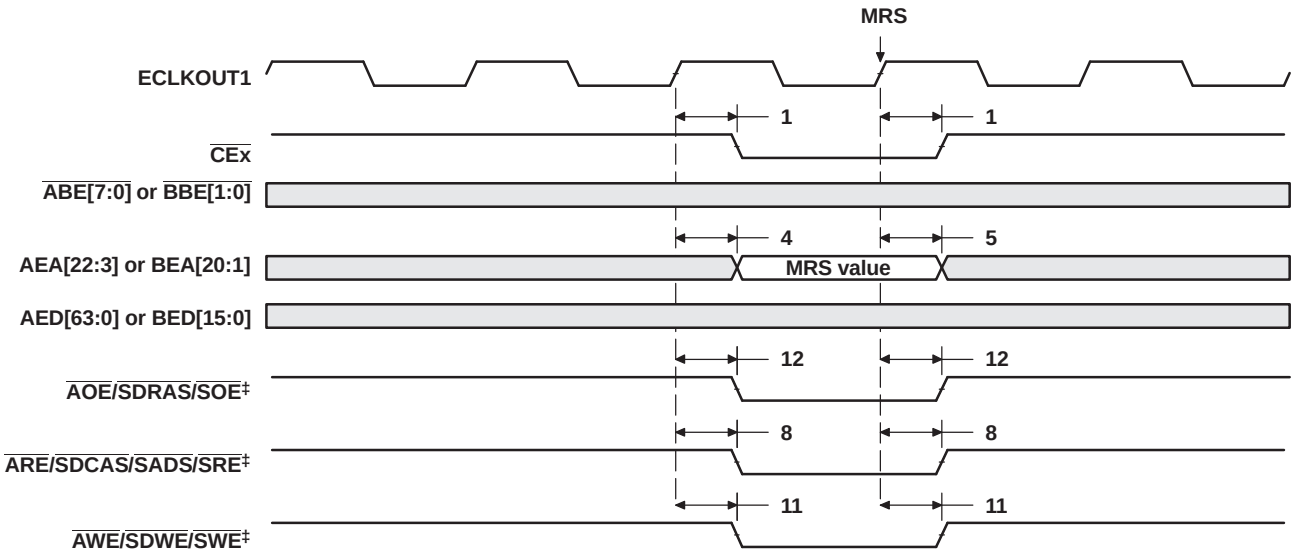
Figure 5-20. SDRAM DEAC Command for EMIFA and EMIFB (see Note)



NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic (SDCAS, SDWE, and SDRAS) instead of ASDCAS, ASDWE, and ASDRAS (for EMIFA) and BSDCAS, BSDWE, and BSDRAS (for EMIFB)].

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

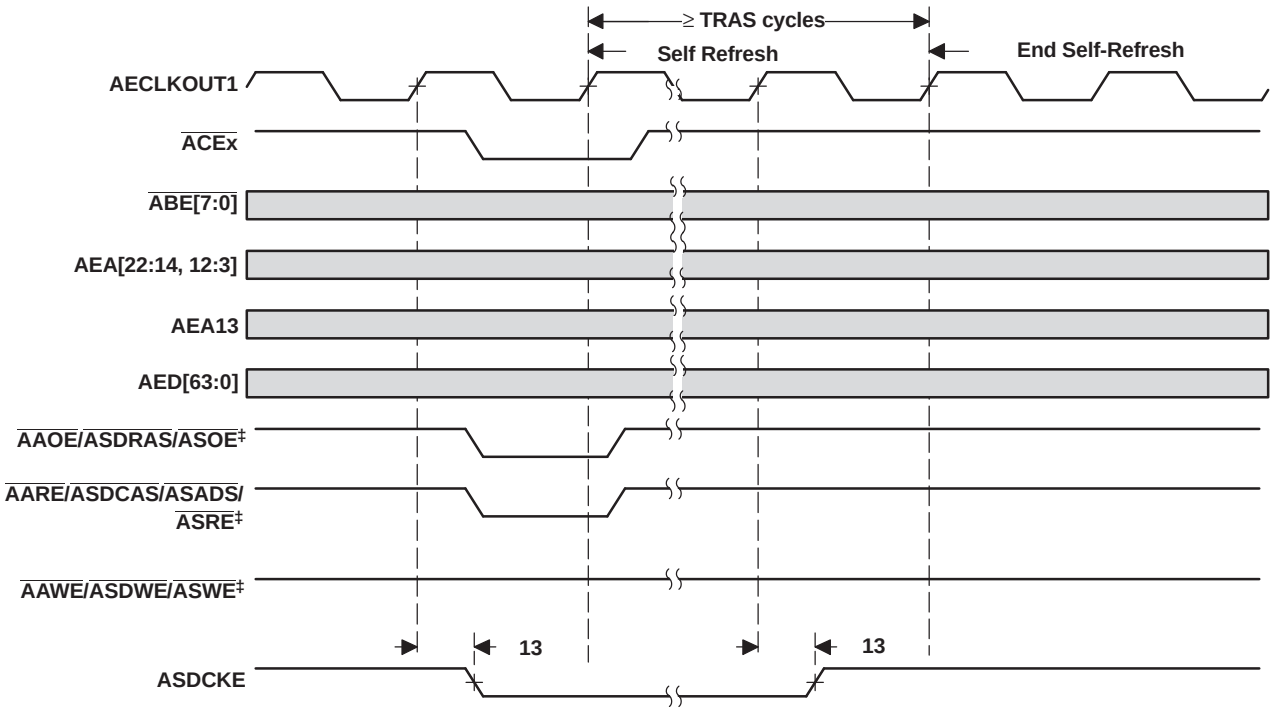
Figure 5-21. SDRAM REFR Command for EMIFA and EMIFB (see Note)



NOTE: These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic (SDCAS, SDWE, and SDRAS) instead of ASDCAS, ASDWE, and ASDRAS (for EMIFA) and BSDCAS, BSDWE, and BSDRAS (for EMIFB)].

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

Figure 5-22. SDRAM MRS Command for EMIFA and EMIFB (see Note)



A. These C64x devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., the synchronous DRAM memory access signals are shown as generic ($\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, and $\overline{\text{SDRAS}}$) instead of $\overline{\text{ASDCAS}}$, $\overline{\text{ASDWE}}$, and $\overline{\text{ASDRAS}}$ (for EMIFA) and $\overline{\text{BSDCAS}}$, $\overline{\text{BSDWE}}$, and $\overline{\text{BSDRAS}}$ (for EMIFB)].

B. $\overline{\text{AARE}}/\overline{\text{ASDCAS}}/\overline{\text{ASADS}}/\overline{\text{ASRE}}$, $\overline{\text{AAWE}}/\overline{\text{ASDWE}}/\overline{\text{ASWE}}$, 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 5-23. SDRAM Self-Refresh Timing for EMIFA Only

5.5 $\overline{\text{HOLD}}/\overline{\text{HOLDA}}$ TIMING

5.5.1 Timing Requirements for the $\overline{\text{HOLD}}/\overline{\text{HOLDA}}$ cycles for EMIFA and EMIFB Modules [(1)](see Figure 5-24)

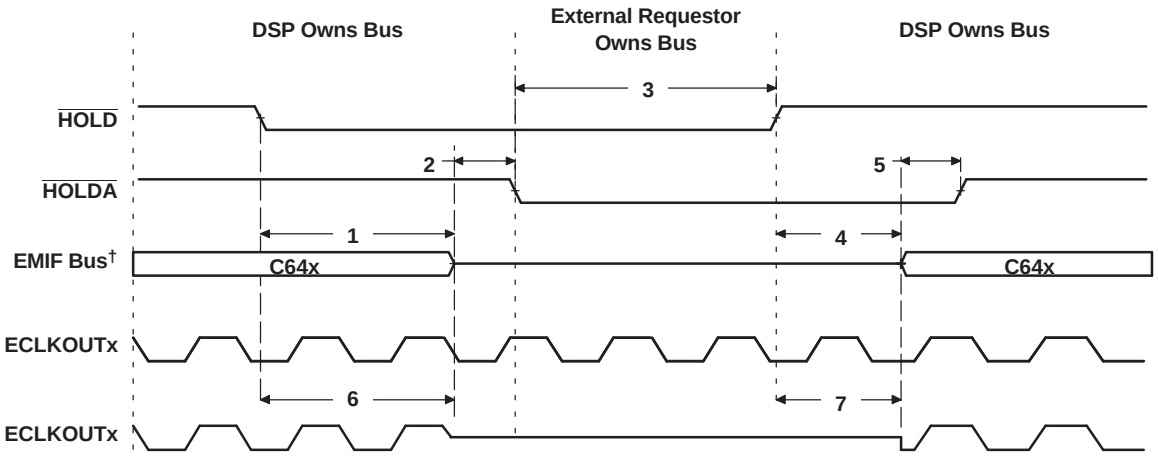
NO.	PARAMETER	–50xEP		UNIT
		MIN	MAX	
3	$t_{oh(HOLDAL-HOLDL)}$ Hold time, $\overline{\text{HOLD}}$ low after $\overline{\text{HOLDA}}$ low	E		ns

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

5.6 Switching Characteristics Over Recommended Operating Conditions for the HOLD/HOLDA Cycles for EMIFA and EMIFB Modules (1) (2) (3) (see Figure 5-24)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(HOLDL-EMHZ)}$	Delay time, $\overline{HOLD}$ low to EMIF Bus high impedance	2E	See (4)	ns
2	$t_{d(EMHZ-HOLDAL)}$	Delay time, EMIF Bus high impedance to $\overline{HOLDA}$ low	0	2E	ns
4	$t_{d(HOLDH-EMLZ)}$	Delay time, $\overline{HOLD}$ high to EMIF Bus low impedance	2E	7E	ns
5	$t_{d(EMLZ-HOLDAH)}$	Delay time, EMIF Bus low impedance to $\overline{HOLDA}$ high	0	2E	ns
6	$t_{d(HOLDL-EKOHZ)}$	Delay time, $\overline{HOLD}$ low to ECLKOUTx high impedance	2E	See (4)	ns
7	$t_{d(HOLDH-EKOLZ)}$	Delay time, $\overline{HOLD}$ high to ECLKOUTx low impedance	2E	7E	ns

(1) E = the EMIF input clock (ECLKIN, CPU/4 clock, or CPU/6 clock) period in ns for EMIFA or EMIFB
(2) For EMIFA, EMIF Bus consists of: $\overline{ACE}[3:0]$, $\overline{ABE}[7:0]$, $\overline{AED}[63:0]$, AEA[22:3], $\overline{AARE}/\overline{ASDCAS}/\overline{ASADS}/\overline{ASRE}$, $\overline{AAOE}/\overline{ASDRAS}/\overline{ASOE}$, and $\overline{AAWE}/\overline{ASDWE}/\overline{ASWE}$, ASDCKE, $\overline{ASOE3}$, and $\overline{APDT}$.
For EMIFB, EMIF Bus consists of: $\overline{BCE}[3:0]$, $\overline{BBE}[1:0]$, BED[15:0], BEA[20:1], $\overline{BARE}/\overline{BSDCAS}/\overline{BSADS}/\overline{BSRE}$, $\overline{BAOE}/\overline{BSDRAS}/\overline{BSOE}$, and $\overline{BAWE}/\overline{BSDWE}/\overline{BSWE}$, $\overline{BSOE3}$, and $\overline{BPDT}$.
(3) The EKxHZ bits in the EMIF Global Control register (GBLCTL) determine the state of the ECLKOUTx signals during $\overline{HOLDA}$. If EKxHZ = 0, ECLKOUTx continues clocking during Hold mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during Hold mode, as shown in Figure 5-24.
(4) All pending EMIF transactions are allowed to complete before $\overline{HOLDA}$ 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.



A. For EMIFA, EMIF Bus consists of : ACE[3:0], ABE[7:0], AED[63:0], AEA[22:3], AARE/ASDCAS/ASADS/ASRE, AAOE/ASDRAS/ASOE, and AAWE/ASDWE/ASWE, ASDCKE, ASOE3, and APDT.
For EMIFB, EMIF Bus consists of: $\overline{BCE}[3:0]$, $\overline{BBE}[1:0]$, BED[15:0], BEA[20:1], $\overline{BARE}/\overline{BSDCAS}/\overline{BSADS}/\overline{BSRE}$, $\overline{BAOE}/\overline{BSDRAS}/\overline{BSOE}$, and $\overline{BAWE}/\overline{BSDWE}/\overline{BSWE}$, $\overline{BSOE3}$, and $\overline{BPDT}$.

Figure 5-24. $\overline{HOLD}/\overline{HOLDA}$ Timing for EMIFA and EMIFB

5.7 BUSREQ TIMING

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(AEKO1H-ABUSRV)}$	Delay time, AECLKOUT1 high to ABUSREQ validA	0.6	7.1	ns
2	$t_{d(BEKO1H-BBUSRV)}$	Delay time, BECLKOUT1 high to BBUSREQ validB	0.5	6.9	ns

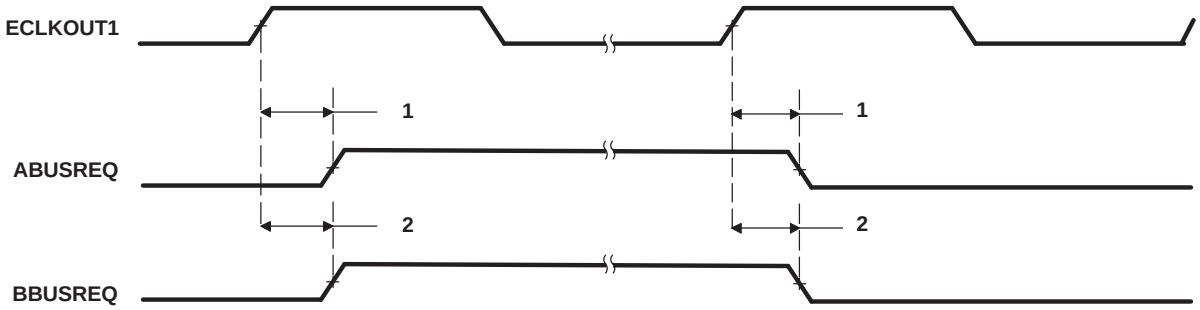


Figure 5-25. BUSREQ Timing for EMIFA and EMIFB

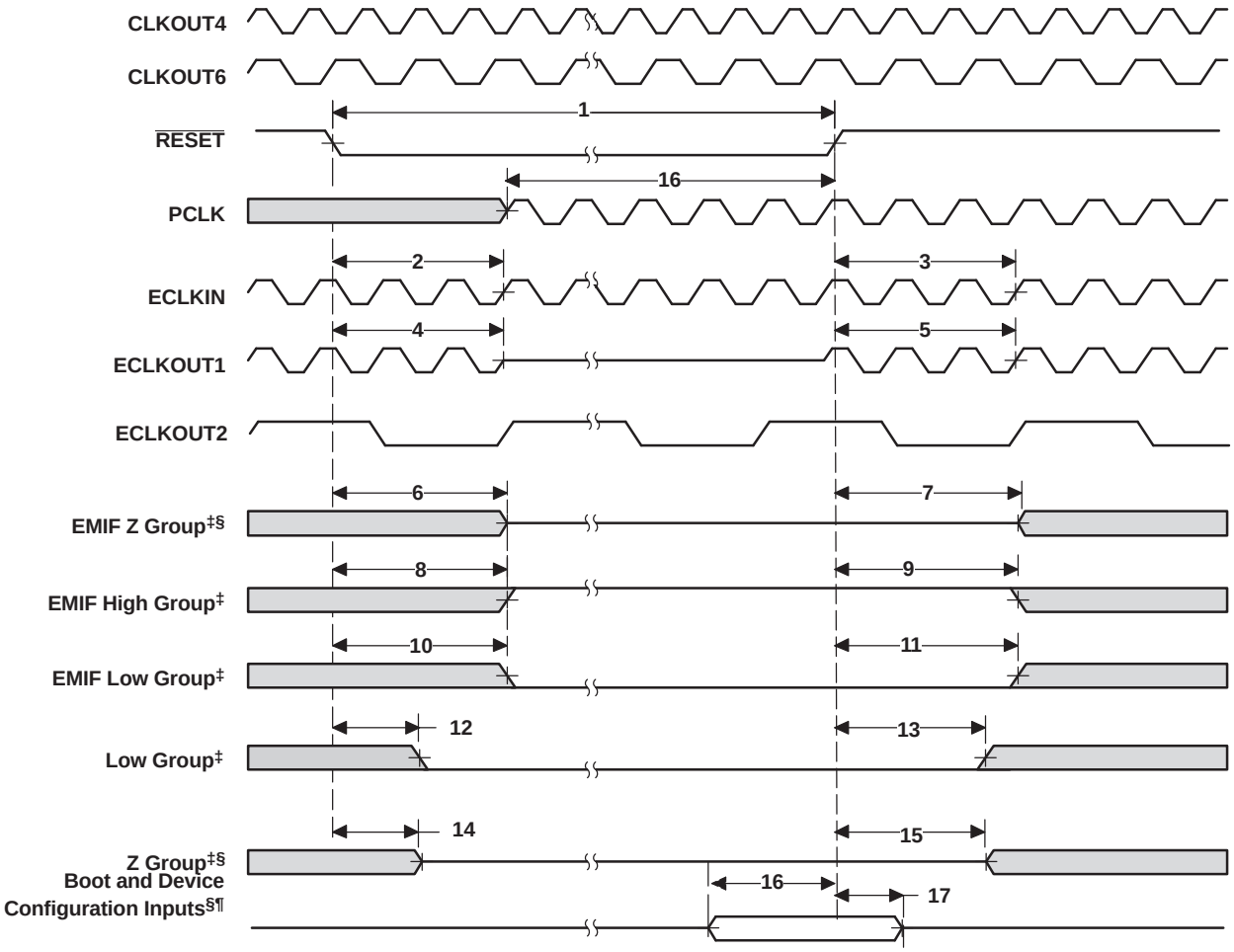
5.8 RESET TIMING

5.8.1 Timing Requirements for Reset(1) (see Figure 5-26)

NO.	PARAMETER		–50xEP MIN	–50xEP MAX	UNIT
1	$t_{w(RST)}$ Width of the RESET pulse	PLL stable(2)	10P		ns
		PLL needs to sync up(3)	250		µs
16	$t_{su(boot)}$	Setup time, boot configuration bits valid before RESET high(4)	4P		ns
17	$t_{h(boot)}$	Hold time, boot configuration bits valid after RESET high(4)	4P		ns

(1) P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
(2) This parameter applies to CLKMODE x1 when CLKIN is stable, and applies to CLKMODE x6, x12 when CLKIN and PLL are stable.
(3) This parameter applies to CLKMODE x6, x12 only (it does not apply to CLKMODE x1). The RESET signal is not connected internally to the clock PLL circuit. The PLL, however, may need up to 250 = s to stabilize following device power up or after PLL configuration has been changed. During that time, RESET must be asserted to ensure proper device operation. See the clock PLL section for PLL lock times.
(4) EMIFB address pins BEA[20:13, 11, 7] are the boot-configuration pins during device reset.

NO.	PARAMETER		–50xEP MIN	–50xEP MAX	UNIT
2	$t_{d(RSTL-ECKI)}$	Delay time, RESET low to ECLKIN synchronized internally	2E	3P + 20E	ns
3	$t_{d(RSTH-ECKI)}$	Delay time, RESET high to ECLKIN synchronized internally	2E	8P + 20E	ns
4	$t_{d(RSTL-ECKO1HZ)}$	Delay time, RESET low to ECLKOUT1 high impedance	2E		ns
5	$t_{d(RSTH-ECKO1V)}$	Delay time, RESET high to ECLKOUT1 valid		8P + 20E	ns
6	$t_{d(RSTL-EMIFZHZ)}$	Delay time, RESET low to EMIF Z high impedance	2E	3P + 4E	ns
7	$t_{d(RSTH-EMIFZV)}$	Delay time, RESET high to EMIF Z valid	16E	8P + 20E	ns
8	$t_{d(RSTL-EMIFHIV)}$	Delay time, RESET low to EMIF high group invalid	2E		ns
9	$t_{d(RSTH-EMIFHV)}$	Delay time, RESET high to EMIF high group valid		8P + 20E	ns
10	$t_{d(RSTL-EMIFLIV)}$	Delay time, RESET low to EMIF low group invalid	2E		ns
11	$t_{d(RSTH-EMIFLV)}$	Delay time, RESET high to EMIF low group valid		8P + 20E	ns
12	$t_{d(RSTL-LOWIV)}$	Delay time, RESET low to low group invalid	0		ns
13	$t_{d(RSTH-LOWV)}$	Delay time, RESET high to low group valid		11P	ns
14	$t_{d(RSTL-ZHZ)}$	Delay time, RESET low to Z group high impedance	0		ns
15	$t_{d(RSTH-ZV)}$	Delay time, RESET high to Z group valid	2P	8P	ns
16	$t_{d(PCLK-RSTH)}$	Delay time, PCLK active to RESET high (3)	32N		ns



A. These C64x. devices have two EMIFs (EMIFA and EMIFB). All EMIFA signals are prefixed by an A and all EMIFB signals are prefixed by a B. Throughout the rest of this document, in generic EMIF areas of discussion, the prefix A or B may be omitted [e.g., ECLKIN, ECLKOUT1, and ECLKOUT2].

B. The following groups consist of:

   a. **EMIF Z group consists of:** AEA[22:3], BEA[20:1], AED[63:0], BED[15:0], $\overline{\text{CE}}$[3:0], $\overline{\text{ABE}}$[7:0], $\overline{\text{BBE}}$[1:0], $\overline{\text{ARE}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SADS}}$/$\overline{\text{SRE}}$, $\overline{\text{AWE}}$/$\overline{\text{SDWE}}$/$\overline{\text{SWE}}$, and $\overline{\text{AOE}}$/$\overline{\text{SDRAS}}$/$\overline{\text{SOE}}$, $\overline{\text{SOE3}}$, ASDCKE, and $\overline{\text{PDT}}$.

   b. **EMIF high group consists of:** AHOLDA and BHOLDA (when the corresponding HOLD input is high)

   c. **EMIF low group consists of:** ABUSREQ and BBUSREQ; $\overline{\text{AHOLDA}}$ and $\overline{\text{BHOLDA}}$ (when the corresponding $\overline{\text{HOLD}}$ input is low)

   d. **Low group consists of:** XSP_CS, CLKX2/XSP_CLK, and DX2/XSP_DO; all of which apply only when PCI EEPROM (BEA13) is enabled (with PCI_EN = 1 and MCBSP2_EN = 0). Otherwise, the CLKX2/XSP_CLK and DX2/XSP_DO pins are in the Z group. For more details on the PCI configuration pins, see the Device Configurations section of this data sheet.

   e. **Z group consists of :** HD[310]/AD[31:0], CLKX0, CLKX1/URADDR4, CLKX2/XSP_CLK, FSX0, FSX1/UXADDR3, FSX2, DX0, DX1/UXADDR4, DX2/XSP_DO, CLKR0, CLKR1/URADDR2, CLKR2, FSR0, FSR1/UXADDR2, FSR2, TOUT0, TOUT1, TOUT2, GP[8:0], GP10/$\overline{\text{PCBE3}}$, HR/$\overline{\text{W}}$/$\overline{\text{PCBE2}}$, $\overline{\text{HDS2}}$/$\overline{\text{PCBE1}}$, $\overline{\text{PCBE0}}$, GP13/$\overline{\text{PINTA}}$, GP11/$\overline{\text{PREQ}}$, $\overline{\text{HDS1}}$/$\overline{\text{PSERR}}$, $\overline{\text{HCS}}$/$\overline{\text{PPERR}}$, HCNTL1/$\overline{\text{PDEVSEL}}$, $\overline{\text{HAS}}$/PPAR, HCNTL0/$\overline{\text{PSTOP}}$, HHWIL/$\overline{\text{PTRDY}}$ (16-bit HPI mode only), $\overline{\text{HRDY}}$/$\overline{\text{PIRDY}}$, $\overline{\text{HINT}}$/$\overline{\text{PFRAME}}$, UXDATA[7:0], UXSOC, UXCLAV, and URCLAV.

C. If BEA[20:13, 11, 7] and HD5/AD5 pins are actively driven, car must be taken to ensure no timing contention between parameters 6, 7, 14, 15, 16, and 17.

D. Boot and Device Configurations Inputs (during reset) includ: EMIFB address pins BEA[20:13, 11, 7] and HD5/AD5. The CI_EN pin must be driven valid at all times and the user must not switch values throughout device operation. The MCBSP2_EN pin must be driven valid at all times and the user can switch values throughout device operation.

Figure 5-26. Reset Timing(A)

5.9 EXTERNAL INTERRUPT TIMING

5.9.1 Timing Requirements for External Interrupts[1] (see Figure 5-27)

NO.	PARAMETER		–50xEP MIN	–50xEP 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	$t_{w(IHIGH)H}$	Width of the NMI interrupt pulse high	4P		ns
		Width of the EXT_INT interrupt pulse high	8P		ns

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

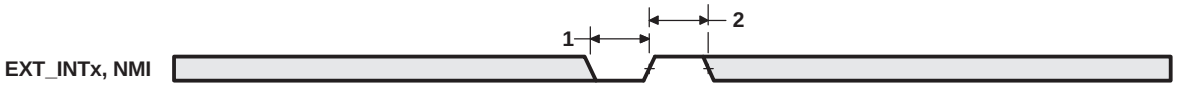


Figure 5-27. External/NMI Interrupt Timing

5.10 HOST-PORT INTERFACE (HPI)

5.10.1 Timing Requirements for Host-Port Interface Cycles[1] [2] (see Figure 5-28 through Figure 5-35)

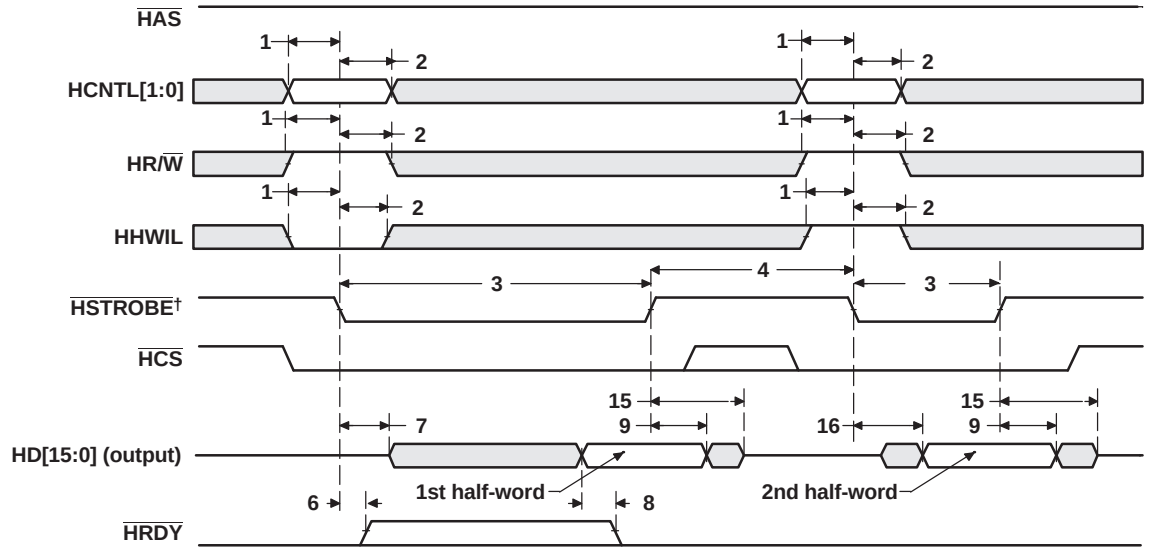
NO.	PARAMETER		–50xEP MIN	–50xEP MAX	UNIT
1	$t_{su(SELV-HSTBL)}$	Setup time, select signals[3] valid before $\overline{\text{HSTROBE}}$ low	5		ns
2	$t_{h(HSTBL-SELV)}$	Hold time, select signals[3] valid after $\overline{\text{HSTROBE}}$ low	2.4		ns
3	$t_{w(HSTBL)}$	Pulse duration, $\overline{\text{HSTROBE}}$ low	4P [4]		ns
4	$t_{w(HSTBH)}$	Pulse duration, $\overline{\text{HSTROBE}}$ high between consecutive accesses	4P		ns
10	$t_{su(SELV-HASL)}$	Setup time, select signals[3] valid before $\overline{\text{HAS}}$ low	5		ns
11	$t_{h(HASL-SELV)}$	Hold time, select signals[3] valid after $\overline{\text{HAS}}$ low	2		ns
12	$t_{su(HDV-HSTBH)}$	Setup time, host data valid before $\overline{\text{HSTROBE}}$ high	5		ns
13	$t_{h(HSTBH-HDV)}$	Hold time, host data valid after $\overline{\text{HSTROBE}}$ high	2.8		ns
14	$t_{h(HRDYL-HSTBL)}$	Hold time, $\overline{\text{HSTROBE}}$ low after $\overline{\text{HRDY}}$ low. $\overline{\text{HSTROBE}}$ should not be inactivated until $\overline{\text{HRDY}}$ is active (low); otherwise, HPI writes will not complete properly.	2		ns
18	$t_{su(HASL-HSTBL)}$	Setup time, $\overline{\text{HAS}}$ low before $\overline{\text{HSTROBE}}$ low	2		ns
19	$t_{h(HSTBL-HASL)}$	Hold time, $\overline{\text{HAS}}$ low after $\overline{\text{HSTROBE}}$ low	2.1		ns

(1) $\overline{\text{HSTROBE}}$ 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 $\overline{\text{HCS}}$.
(2) P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
(3) Select signals include: HCNTL[1:0] and HR/$\overline{\text{W}}$. For HPI16 mode only, select signals also include HHWIL.
(4) Select the parameter value of 4P or 12.5 ns, whichever is greater.

NO.	PARAMETER		–50xEP MIN	–50xEP MAX	UNIT
6	$t_{d(HSTBL-HRDYH)}$	Delay time, $\overline{\text{HSTROBE}}$ low to $\overline{\text{HRDY}}$ high[1]	1.3	4P + 9	ns
7	$t_{d(HSTBL-HDLZ)}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD low impedance for an HPI read	2		ns
8	$t_{d(HDV-HRDYL)}$	Delay time, HD valid to $\overline{\text{HRDY}}$ low	–3		ns
9	$t_{oh(HSTBH-HDV)}$	Output hold time, HD valid after $\overline{\text{HSTROBE}}$ high	1.5		ns

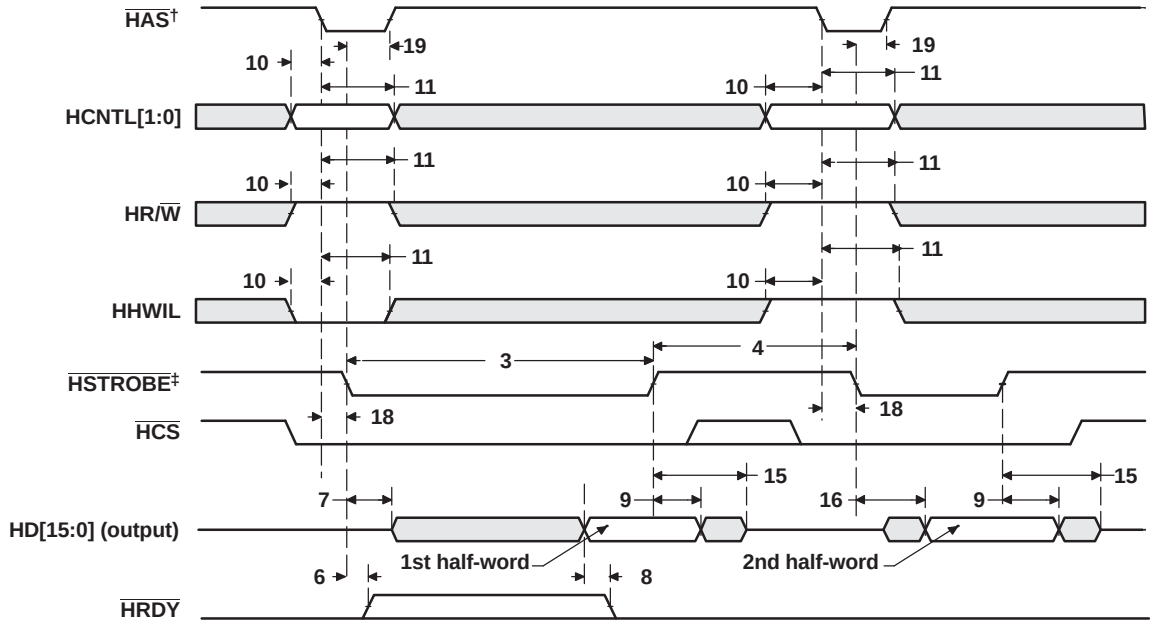
(1) This parameter is used during HPID reads and writes. For reads, at the beginning of a word transfer (HPI32) or the first half-word transfer (HPI16) on the falling edge of $\overline{\text{HSTROBE}}$, the HPI sends the request to the EDMA internal address generation hardware, and $\overline{\text{HRDY}}$ 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.

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
15	$t_{d(HSTBH-HDHZ)}$	Delay time, HSTROBE high to HD high impedance		12	ns
16	$t_{d(HSTBL-HDV)}$	Delay time, HSTROBE low to HD valid (HPI16 only)		4P + 8	ns



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

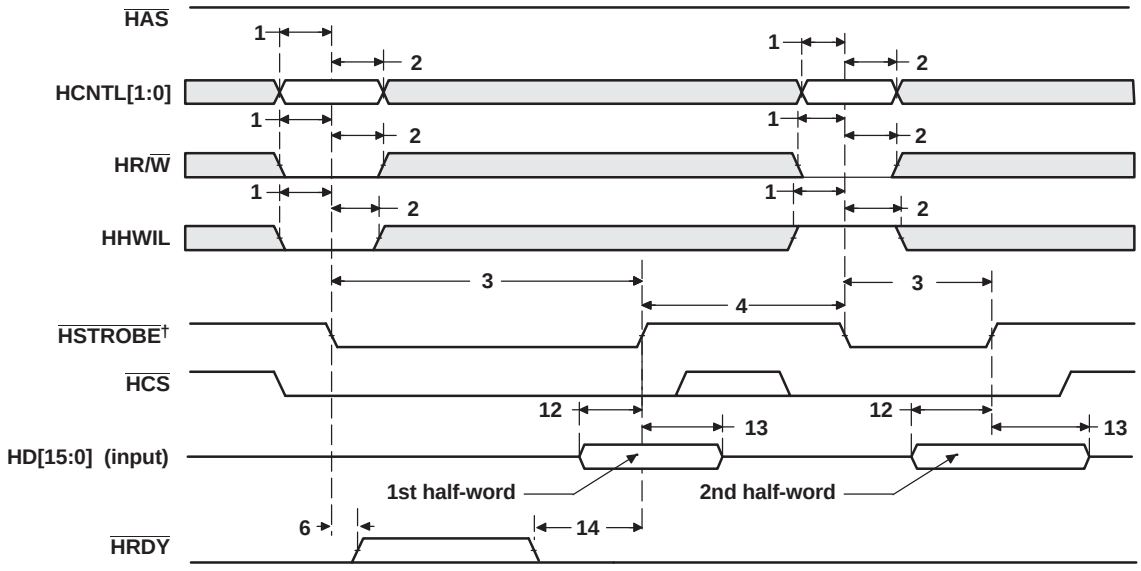
Figure 5-28. HPI16 Read Timing (HAS Not Used, Tied High)



(1) For correct operation, strobe the HAS signal only once per HSTROBE active cycle.

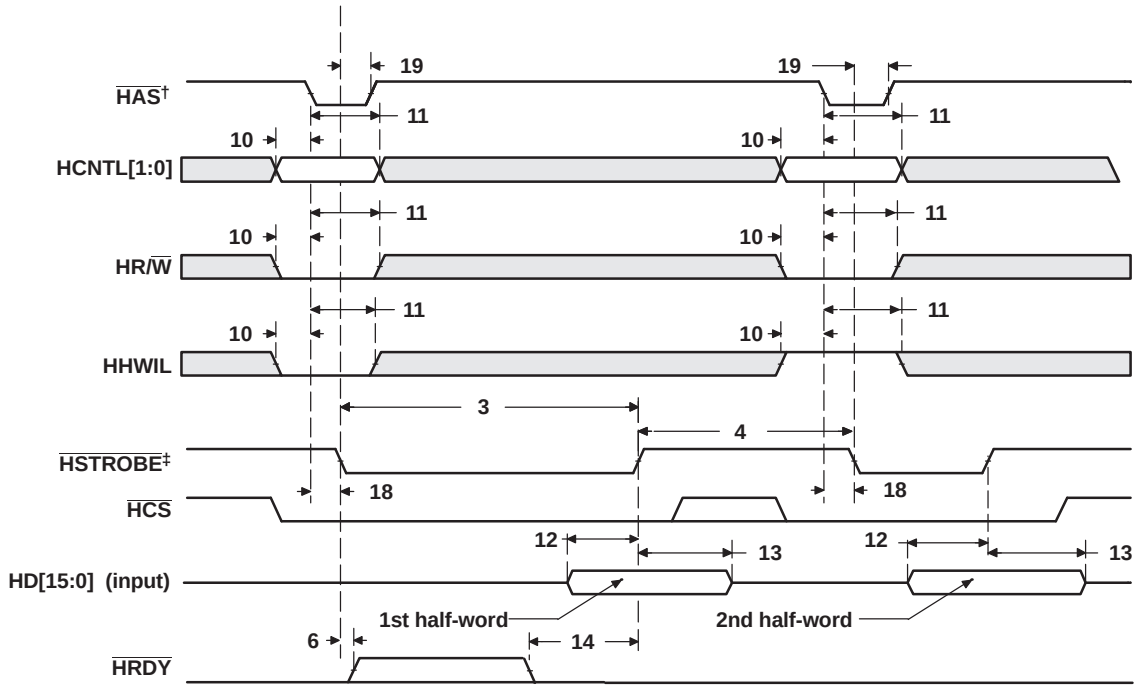
(2) HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 5-29. HPI16 Read Timing (HAS Used)



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

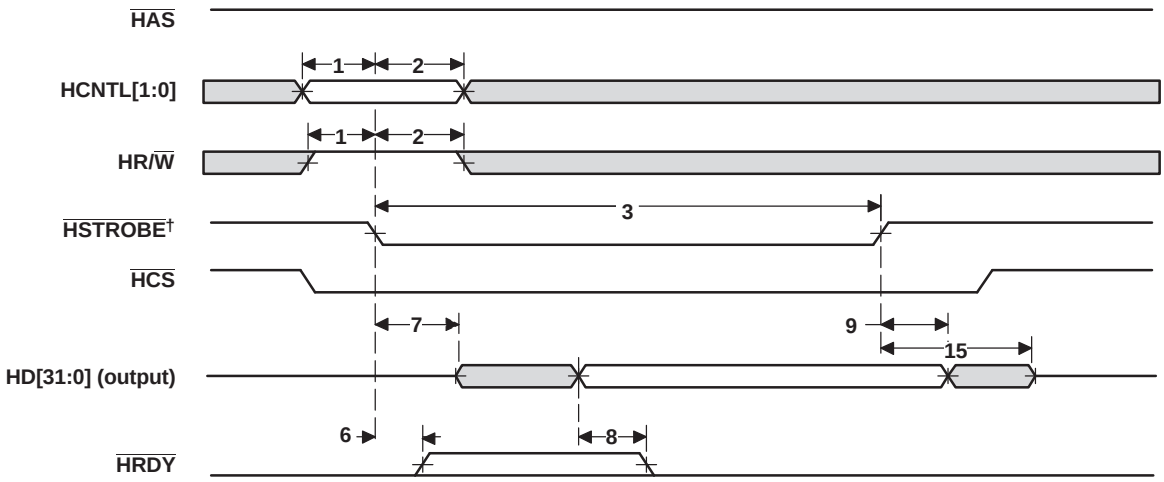
Figure 5-30. HPI16 Write Timing (HAS Not Used, Tied High)



(1) For correct operation, strobe the HAS signal only once per HSTROBE active cycle.

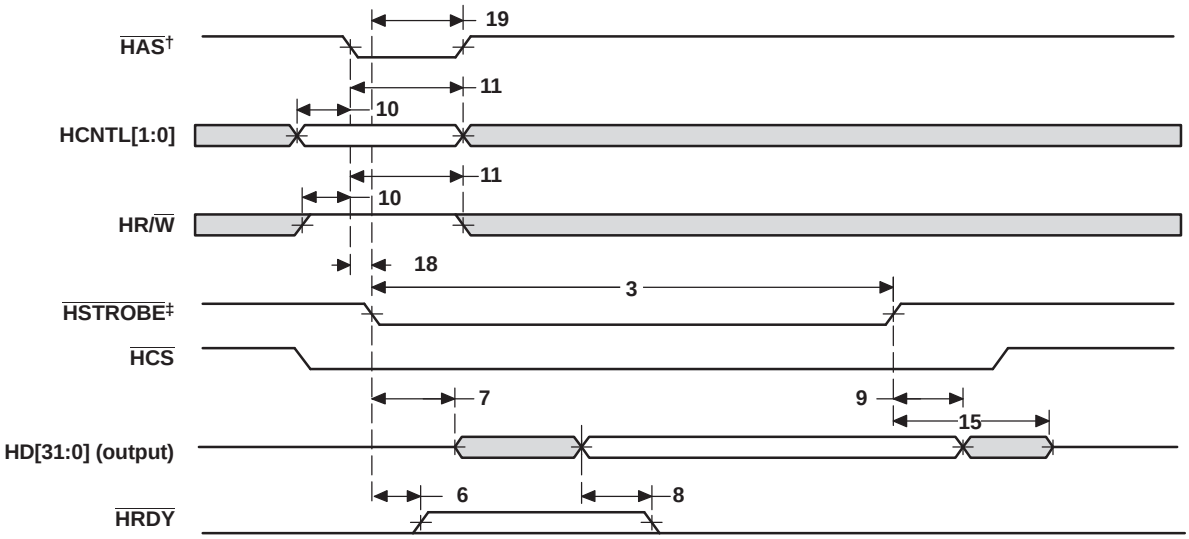
(2) HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 5-31. HPI16 Write Timing (HAS Used)



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

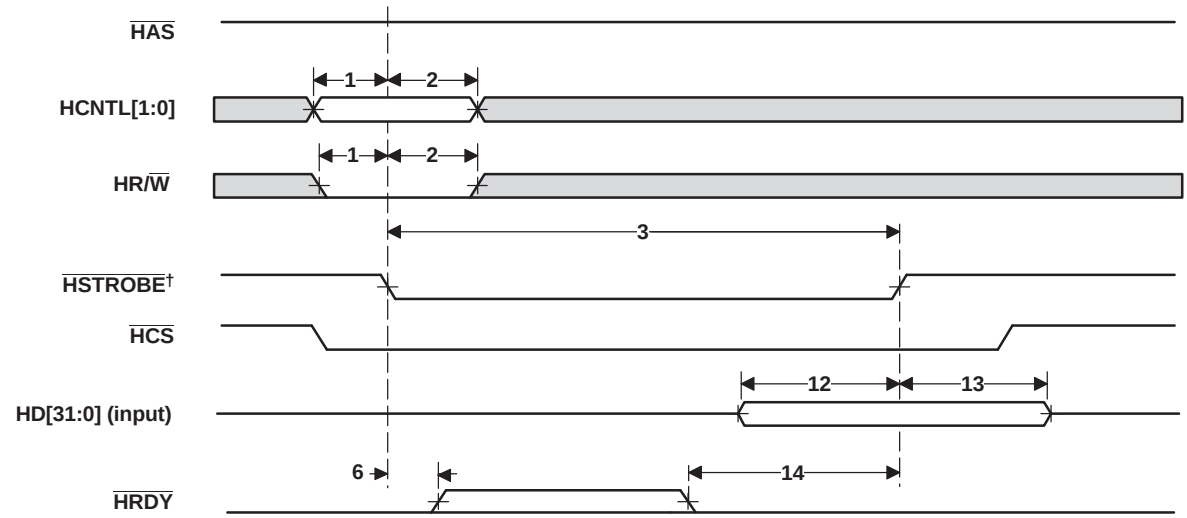
Figure 5-32. HPI32 Read Timing (HAS Not Used, Tied High)



(1) For correct operation, strobe the HAS signal only once per HSTROBE active cycle.

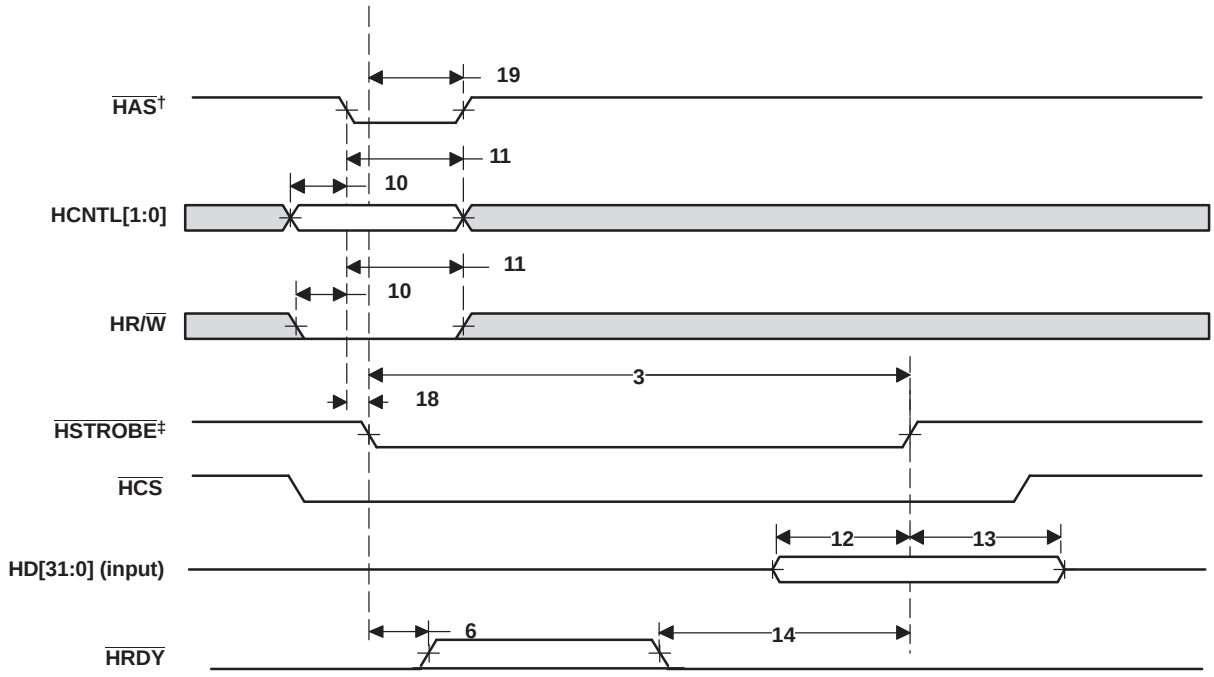
(2) HSTROBE refers to the following logical operation on HCS, HDS1, and HDS2: [NOT(HDS1 XOR HDS2)] OR HCS.

Figure 5-33. HPI32 Read Timing (HAS Used)



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

Figure 5-34. HPI32 Write Timing (HAS Not Used, Tied High)



(1) For correct operation, strobe the HAS signal only once per HSTROBE active cycle.

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

Figure 5-35. HPI32 Write Timing (HAS Used)

5.11 PERIPHERAL COMPONENT INTERCONNECT (PCI) TIMING (C6415 AND C6416 ONLY)

5.11.1 Timing Requirements for PCLK (1) (2)(see Figure 5-36

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{c(PCLK)}$	Cycle time, PCLK	30 (or 8P(3))		ns
2	$t_{w(PCLKH)}$	Pulse duration, PCLK high	11		ns
3	$t_{w(PCLKL)}$	Pulse duration, PCLK low	11		ns
4	$t_{sr(PCLK)}$	Δv/Δt slew rate, PCLK	1	4	V/ns

(1) For 3.3-V operation, the reference points for the rise and fall transitions are measured at V_{ILP} MAX and V_{IHP} MIN.
(2) P = 1/CPU clock frequency in ns. For example when running parts at 500 MHz, use P = 2 ns.
(3) Select the parameter value of 30 ns or 8P, whichever is greater.

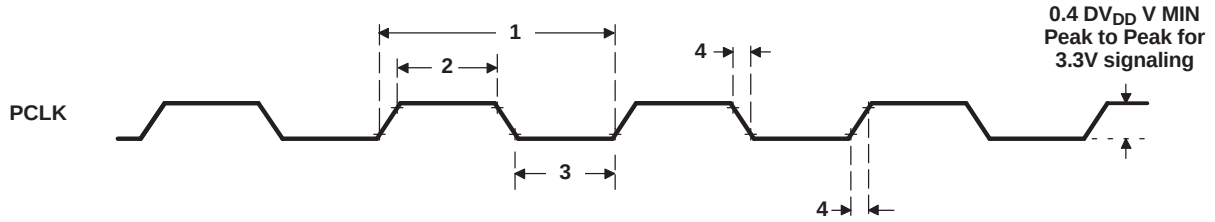


Figure 5-36. PCLK Timing

5.11.2 Timing Requirements for PCI Reset (see Figure 5-37)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{w(PRST)}$	Pulse duration, $\overline{PRST}$	1		ms
2	$t_{su(PCLKA-PRSTH)}$	Setup time, PCLK active before $\overline{PRST}$ high	100		µs

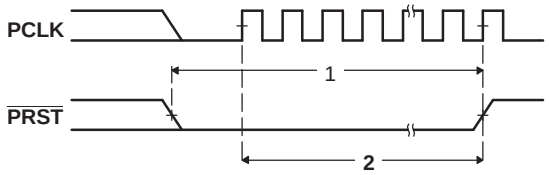


Figure 5-37. PCI Reset ($\overline{PRST}$) Timing

5.11.3 Timing Requirements for PCI Inputs (see Figure 5-38)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{su(IV-PCLKH)}$	Setup time, input valid before PCLK high	7		ns
2	$t_{h(IV-PCLKH)}$	Hold time, input valid after PCLK high	0		ns

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(PCLKH-OV)}$	Delay time, PCLK high to output valid		11	ns
2	$t_{d(PCLKH-OIV)}$	Delay time, PCLK high to output invalid	2		ns
3	$t_{d(PCLKH-OLZ)}$	Delay time, PCLK high to output low impedance	2		ns
4	$t_{d(PCLKH-OHZ)}$	Delay time, PCLK high to output high impedance		28	ns

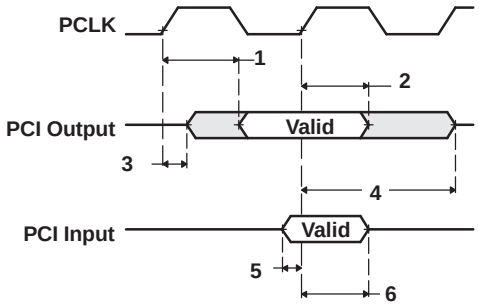


Figure 5-38. PCI Input/Output Timing

5.11.5 Timing Requirements for Serial EEPROM Interface (see Figure 5-39)

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

NO.			–50xEP			UNIT
			MIN	TYP	MAX	
1	$t_{w(CSL)}$	Pulse duration, XSP_CS low		4092P		ns
2	$t_{d(CLKL-CSL)}$	Delay time, XSP_CLK low to XSP_CS low		0		ns
3	$t_{d(CSH-CLKH)}$	Delay time, XSP_CS high to XSP_CLK high		2046P		ns
4	$t_{w(CLKH)}$	Pulse duration, XSP_CLK high		2046P		ns
5	$t_{w(CLKL)}$	Pulse duration, XSP_CLK low		2046P		ns
6	$t_{osu(DOV-CLKH)}$	Output setup time, XSP_DO valid after XSP_CLK high		2046P		
7	$t_{oh(CLKH-DOV)}$	Output hold time, XSP_DO valid after XSP_CLK high		2046P		

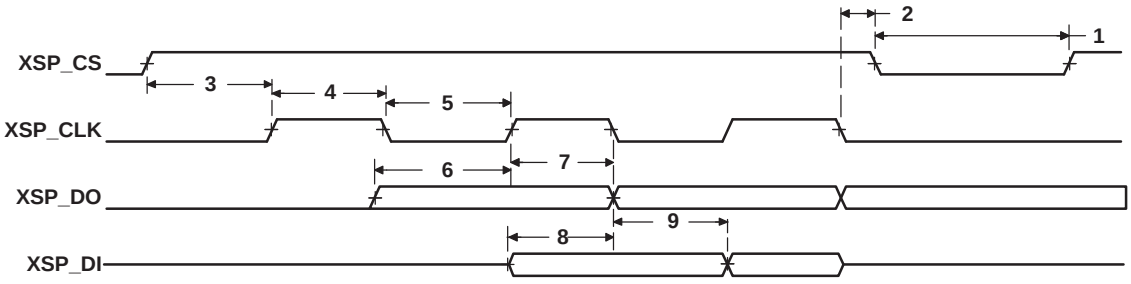


Figure 5-39. PCI Serial EEPROM Interface Timing

5.12 MULTICHANNEL BUFFERED SERIAL PORT (McBSP) TIMING

5.12.1 Timing Requirements for McBSP (1) (2) (see Figure 5-40

NO.				–50xEP		UNIT
				MIN	MAX	
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X ext	6.67(3)		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	$0.5t_{c(CKRX)}$ - 1(3)		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	CLKR int	9		ns
			CLKR ext	1.3		
11	$t_{h(CKXL-FXH)}$	Hold time, external FSX high after CLKX low	CLKR int	6		ns
			CLKR ext	3		

(1) CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
(2) P = 1/CPU clock frequency in ns. For example, when running parts at 500 MHz, use P = 2 ns.
(3) 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.

NO.	PARAMETER			–50xEP		UNIT
				MIN	MAX	
1	$t_{d(CKSH-CKRXH)}$	Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input		1.4	10	ns
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X int	6.67(1)		ns
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X int	C – 1(4)	C + 1(4)	ns
4	$t_{d(CKRH-FRV)}$	Delay time, CLKR high to internal FSR valid	CLKR int	–2.1	3	ns
9	$t_{d(CKXH-FXV)}$	Delay time, CLKX high to internal FSX valid	CLKR int	–1.7	3	ns
			CLKR ext	1.7	9	
12	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	CLKR int	–3.9	4	ns
			CLKR ext	–2.1	9	
13	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	CLKR int	–3.9 + D1(5)	4 + D2(5)	ns
			CLKR ext	–2.1 + D1(5)	9 + D2(5)	
14	$t_{d(FXH-DXV)H}$	Delay time, FSX high to DX valid ONLY applies when in data delay 0 (XDATDLY = 00b) moded	FSX int	–2.3	5.6	ns
			FSX ext	1.9	9	

(1) 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.

(4) 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

(5) 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

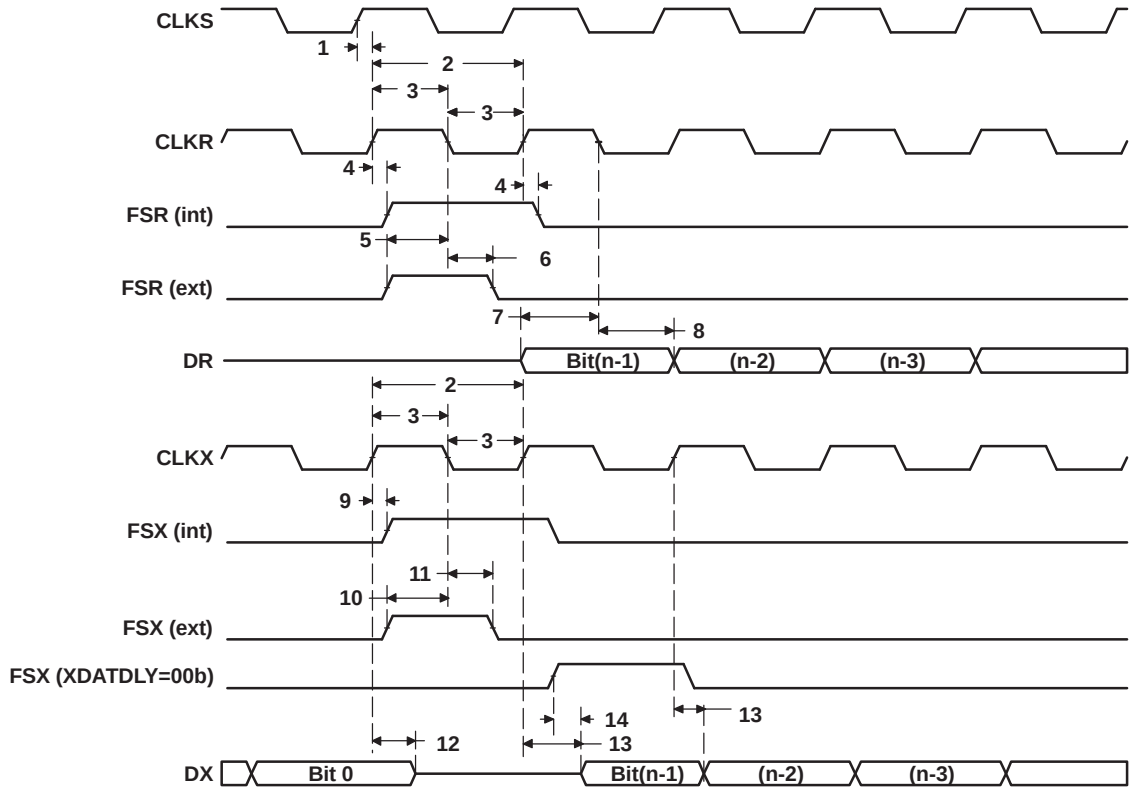


Figure 5-40. McBSP Timing

5.12.3 Timing Requirements for FSR When GSYNC = 1 (see Figure 5-41)

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{su(FRH-CKSH)}$	Setup time, FSR high before CLKS high	4		ns
2	$t_{h(CKSH-FRH)}$	Hold time, FSR high after CLKS high	4		ns

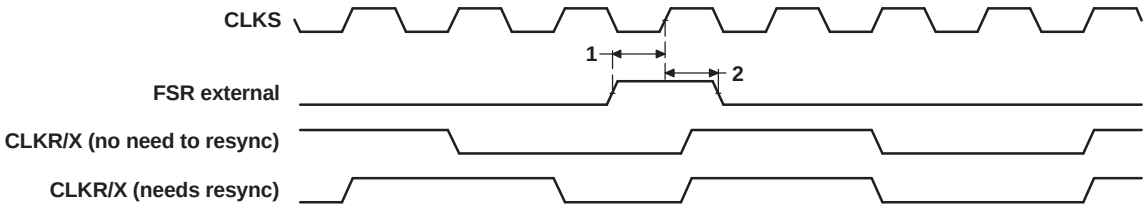


Figure 5-41. FSR Timing When GSYNC = 1

5.12.4 Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0 (1) (2) (see)

NO.			–50xEP				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
4	$t_{su(DRV-CKXL)}$	Setup time, DR valid before CLKX low	12		2 – 12P		ns

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

NO.			–50xEP				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
5	$t_{h(CKXL-DRV)}$	Hold time, DR valid after CLKX low	4		5 + 24P		ns

NO.	PARAMETER		–50xEP				UNIT
			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low(4)	T – 2	T + 3			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high(5)	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

(3) 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

(4) 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 McBSPF

(5) 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).

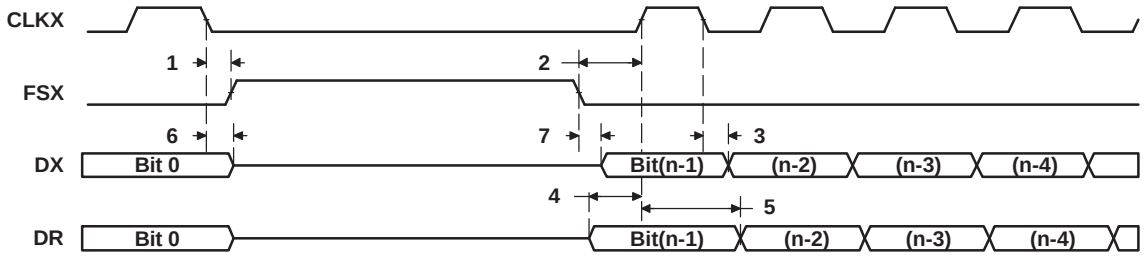


Figure 5-42. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

5.12.6 Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1[1] [2](see Figure 5-43)

NO.			–50xEP				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

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

NO.	PARAMETER		–50xEP				UNIT
			MASTER[3]		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX low[4]	H – 2	H + 3			ns
2	$t_{d(FXL-CKXXL)}$	Delay time, FSX low to CLKX high[5]	T – 2	T + 1			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	–2	4	12P + 4	20P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	–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

(3) 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

(4) 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 McBSPF

(5) 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).

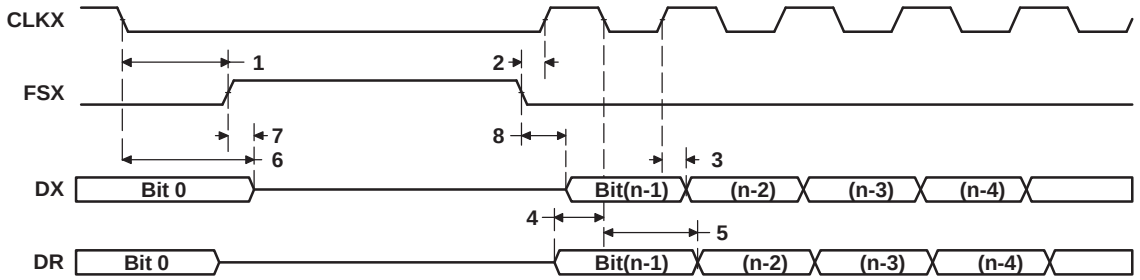


Figure 5-43. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

5.13 UTOPIA SLAVE TIMING (C6415 AND C6416 ONLY)

5.13.1 Timing Requirements for UXCLK(1) (see Figure 5-44)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{c(UXCK)}$	Cycle time, UXCLK	20		ns
2	$t_{w(UXCKH)}$	Pulse duration, UXCLK high	$0.4t_{c(UXCK)}$	$0.6t_{c(UXCK)}$	ns
3	$t_{w(UXCKL)}$	Pulse duration, UXCLK low	$0.4t_{c(UXCK)}$	$0.6t_{c(UXCK)}$	ns
4	$t_{t(UXCK)}$	Transition time, UXCLK		2	ns

(1) The reference points for the rise and fall transitions are measured at VIL MAX and VIH MIN.

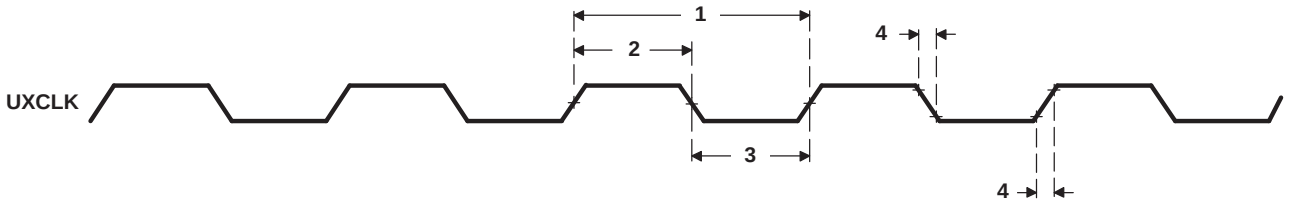


Figure 5-44. UXCLK Timing

5.13.2 Timing Requirements for URCLK(1) (see Figure 5-45

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{c(URCK)}$	Cycle time, URCLK	20		ns
2	$t_{w(URCKH)}$	Pulse duration, URCLK high	$0.4t_{c(URCK)}$	$0.6t_{c(URCK)}$	ns
3	$t_{w(URCKL)}$	Pulse duration, URCLK low	$0.4t_{c(URCK)}$	$0.6t_{c(URCK)}$	ns
4	$t_{t(URCK)}$	Transition time, URCLK		2	ns

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

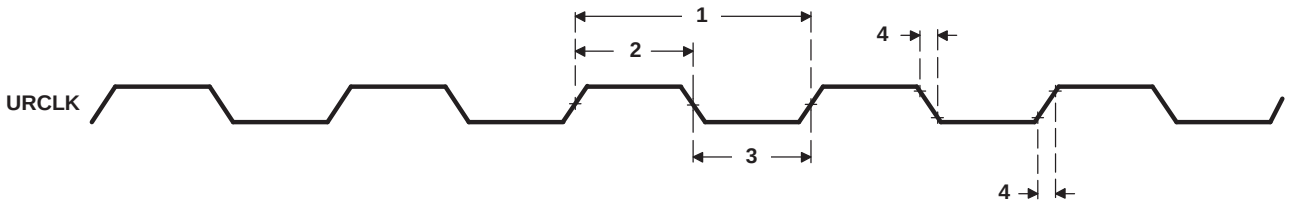


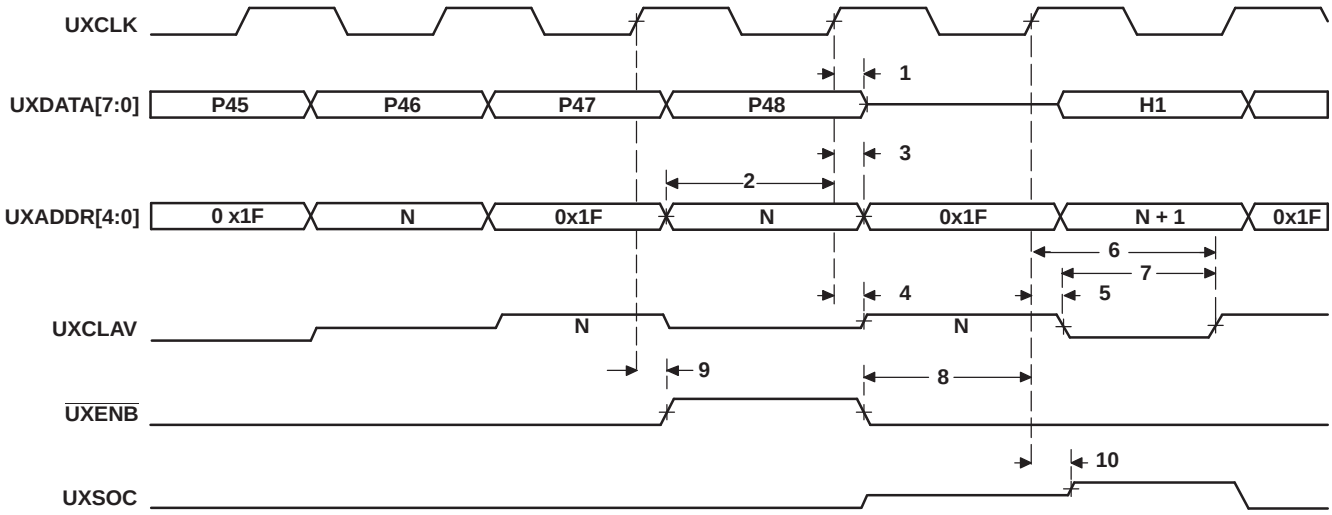
Figure 5-45. URCLK Timing

5.13.3 Timing Requirements for UTOPIA Slave Transmit (see Figure 5-46)

NO.			–50xEP		UNIT
			MIN	MAX	
2	$t_{su(UXAV-UXCH)}$	Setup time, UXADDR valid before UXCLK high	4		ns
3	$t_{h(UXCH-UXAV)}$	Hold time, UXADDR valid after UXCLK high	1		ns
8	$t_{su(UXENBL-UXCH)}$	Setup time, UXENB low before UXCLK high	4		ns
9	$t_{h(UXCH-UXENBL)}$	Hold time, UXENB low after UXCLK high	1		ns

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
1	$t_{d(UXCH-UXDV)}$	Delay time, UXCLK high to UXDATA valid	3	12	ns
4	$t_{d(UXCH-UXCLAV)}$	Delay time, UXCLK high to UXCLAV driven active value	3	12	ns
5	$t_{d(UXCH-UXCLAVL)}$	Delay time, UXCLK high to UXCLAV driven inactive low	3	12	ns

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
6	$t_{d(UXCH-UXCLAVHZ)}$	Delay time, UXCLK high to UXCLAV Hi-Z	9	18.5	ns
7	$t_{w(UXCLAVL-UXCLAVHZ)}$	Pulse duration (low), UXCLAV low to UXCLAV Hi-Z	3		ns
10	$t_{d(UXCH-UXSV)}$	Delay time, UXCLK high to UXSOC valid	3	12	ns



† The UTOPIA Slave module has signals that are middle-level signals indicating a high-impedance state (i.e., the UXCLAV and UXSOC signals).

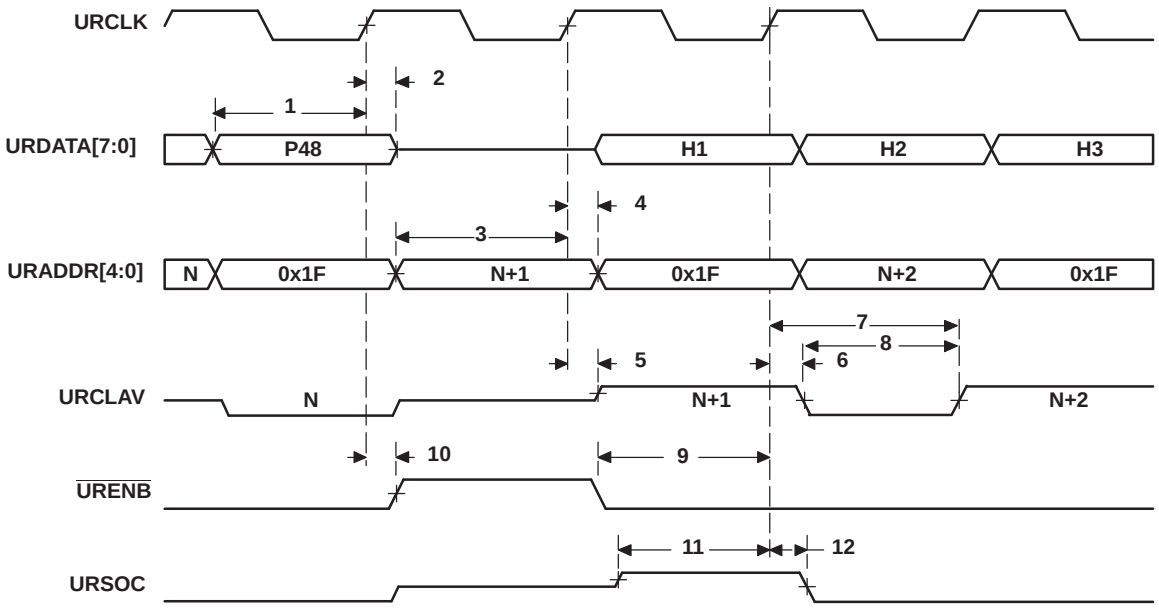
(1) The UTOPIA Slave module has signals that are middle-level signals indicating a high-impedance state (i.e., the UXCLAV and UXSOC signals).

Figure 5-46. UTOPIA Slave Transmit Timing[(1)]

5.13.5 Timing Requirements for UTOPIA Slave Receive (see Figure 5-47)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{su(URDV-URCH)}$	Setup time, URDATA valid before URCLK highR	4		ns
2	$t_{h(URCH-URDV)}$	Hold time, URDATA valid after URCLK high	1		ns
3	$t_{su(URAV-URCH)}$	Setup time, URADDR valid before URCLK highR	4		ns
4	$t_{h(URCH-URAV)}$	Hold time, URADDR valid after URCLK high	1		ns
9	$t_{su(URENBL-URCH)}$	Setup time, $\overline{\text{URENB}}$ low before URCLK high	4		ns
10	$t_{h(URCH-URENBL)}$	Hold time, $\overline{\text{URENB}}$ low after URCLK high	1		ns
11	$t_{su(URSH-URCH)}$	Setup time, URSOC high before URCLK highU	4		ns
12	$t_{h(URCH-URSH)}$	Hold time, URSOC high after URCLK high	1		ns

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
5	$t_{d(URCH-URCLAV)}$	Delay time, URCLK high to URCLAV driven active value	3	12	ns
6	$t_{d(URCH-URCLAVL)}$	Delay time, URCLK high to URCLAV driven inactive low	3	12	ns
7	$t_{d(URCH-URCLAVHZ)}$	Delay time, URCLK high to URCLAV Hi-Z	9	18.5	ns
8	$t_{w(URCLAVL-URCLAVHZ)}$	Pulse duration (low), URCLAV low to URCLAV Hi-Z	3		ns



† The UTOPIA Slave module has signals that are middle-level signals indicating a high-impedance state (i.e., the URCLAV and URSOC signals).

(1) The UTOPIA Slave module has signals that are middle-level signals indicating a high-impedance state (i.e., the URCLAV and URSOC signals).

Figure 5-47. UTOPIA Slave Receive Timing (1)

5.14 TIMER TIMING

5.14.1 Timing Requirements for Timer Inputs(1) (see Figure 5-48)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{w(TINPH)}$	Pulse duration, TINP high	8P		ns
2	$t_{w(TINPL)}$	Pulse duration, TINP low	8P		ns

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

NO.	PARAMETER		–50xEP		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

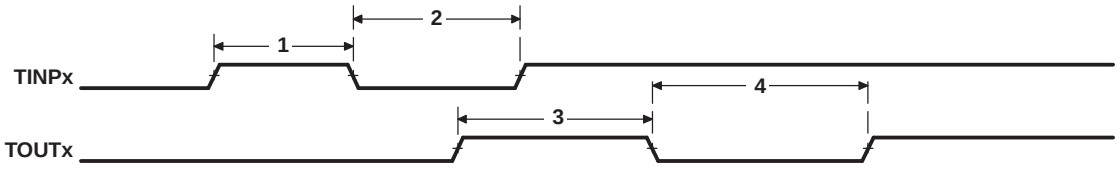


Figure 5-48. Timer Timing

5.15 GENERAL-PURPOSE INPUT/OUTPUT (GPIO) PORT TIMING

5.15.1 Timing Requirements for GPIO Inputs[1] [2] (see Figure 5-48)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{w(GPIH)}$	Pulse duration, GPIx high 8P	8P		ns
2	$t_{w(GPIL)}$	Pulse duration, GPIx low 8P	8P		ns

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

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
3	$t_{w(GPOH)}$	Pulse duration, GPOx high	32P		ns
4	$t_{w(GPOL)}$	Pulse duration, GPOx low	32P		ns

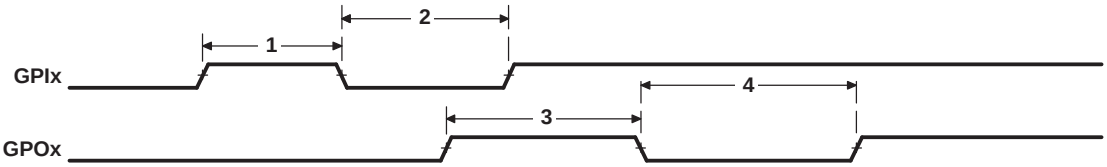


Figure 5-49. GPIO Port Timing

5.16 JTAG TEST PORT TIMING

5.16.1 Timing Requirements for JTAG Test Port (see Figure 5-50)

NO.			–50xEP		UNIT
			MIN	MAX	
1	$t_{c(TCK)}$	Cycle time, TCK	35		ns
3	$t_{su(TDIV-TCKH)}$	Setup time, TDI/TMS/$\overline{TRST}$ valid before TCK high	10		ns
4	$t_{h(TCKH-TDIV)}$	Hold time, TDI/TMS/$\overline{TRST}$ valid after TCK high	9		ns

NO.	PARAMETER		–50xEP		UNIT
			MIN	MAX	
2	$t_{d(TCKL-TDOV)}$	Delay time, TCK low to TDO valid	–3	18	ns

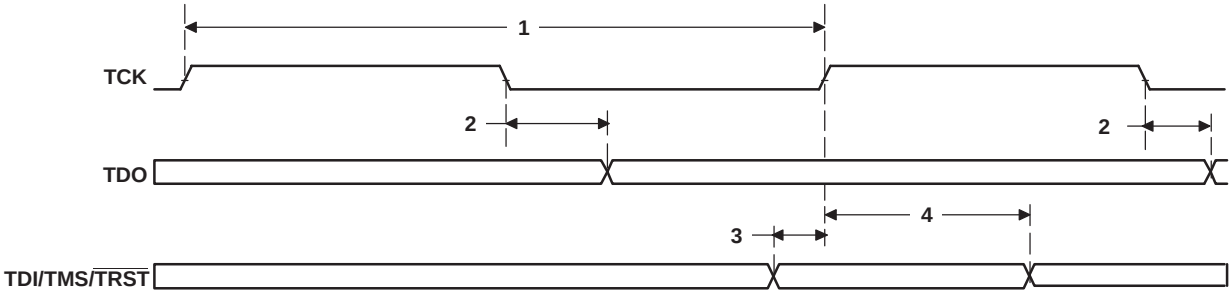


Figure 5-50. JTAG Test Port Timing

5.16.3 Thermal Resistance Characteristics (S-PBGA Package)

NO.			AIR FLOW (m/s[1])	°C/W	°C/W (WITH HEAT SINK[2])
1	$R_{\theta JC}$	Junction to case	N/A	1.55	1.0
2	$R_{\theta JB}$	Junction to board	N/A	9.1	9.0
3	$R_{\theta JA}$	Junction to free air	0.00	17.9	13.8
4	$R_{\theta JA}$	Junction to free air	0.5	15.02	8.95
5	$R_{\theta JA}$	Junction to free air	1.0	13.4	7.35
6	$R_{\theta JA}$	Junction to free air	2.00	11.89	6.46
7	Psi_{JT}	Junction to package top	N/A	0.5	0.5
8	Psi_{JB}	Junction to board	N/A	7.4	7.4

(1) m/s = meters per seconds
(2) These thermal resistance numbers were modeled using a heatsink, part number 374024B00035, manufactured by AAVID Thermalloy. AAVID Thermalloy also manufactures a similar epoxy-mounted heatsink, part number 374024B00000. TI recommends a passive, laminar heatsink, similar to the part numbers mentioned above.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead/Ball Finish	MSL Peak Temp (3)
SM32C6414DGLZ50AEP	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
SM32C6414EGLZ50AEP	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
SM32C6415DGLZ50AEP	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
SM32C6415EGLZ50AEP	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
SM32C6415EGLZ50SEP	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
SM32C6416DGLZ50AEP	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
SM32C6416EGLZ50AEP	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
V62/04609-01XA	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
V62/04609-02XA	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
V62/04609-03XA	OBSOLETE	FCBGA	GLZ	532		TBD	Call TI	Call TI
V62/04609-04XA	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
V62/04609-05XA	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
V62/04609-06XA	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR
V62/04609-08XA	ACTIVE	FCBGA	GLZ	532	60	TBD	SNPB	Level-4-220C-72 HR

(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)

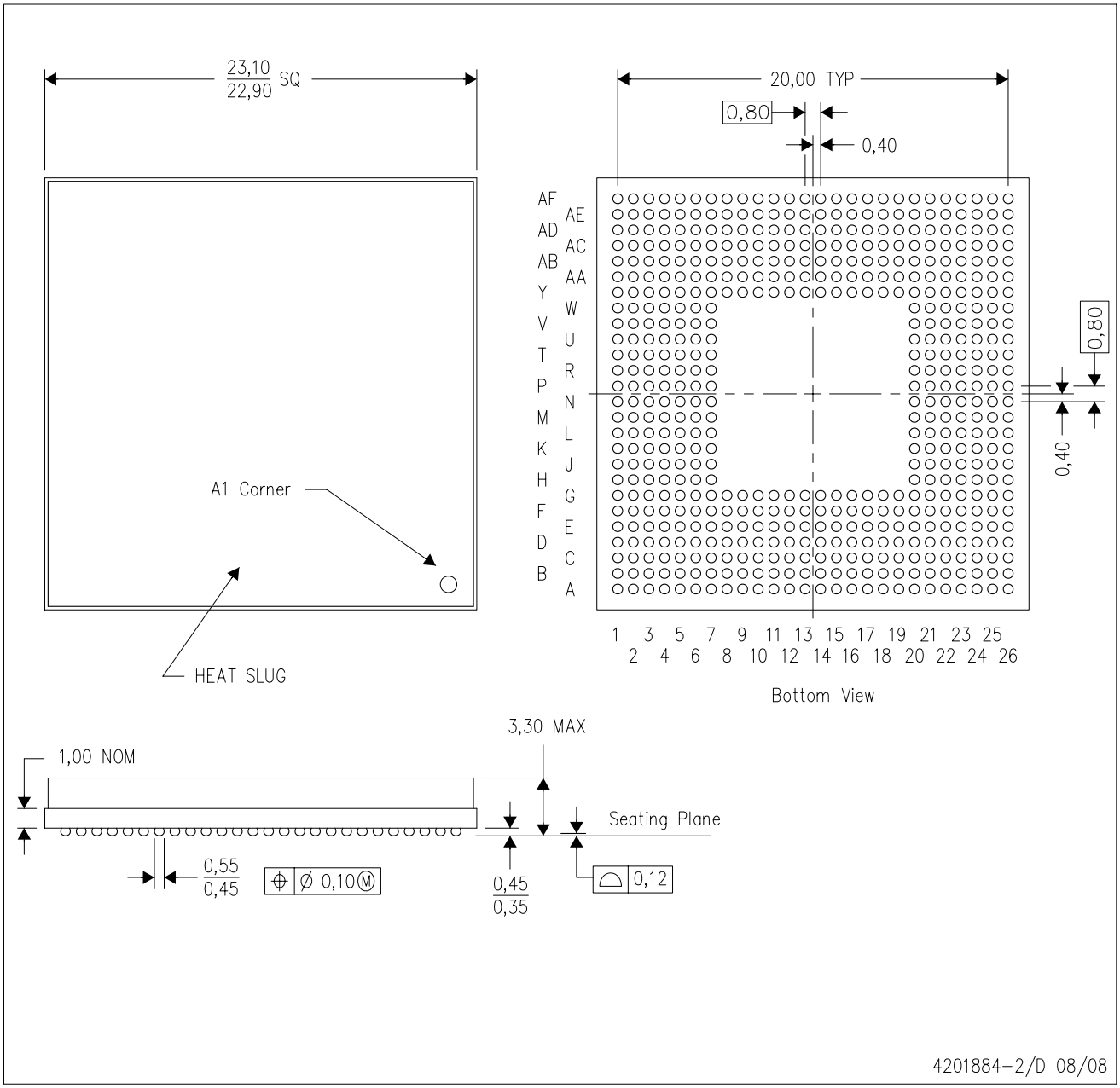
(3) MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

OTHER QUALIFIED VERSIONS OF SM320C6415-EP :
- Catalog: SM320C6415

NOTE: Qualified Version Definitions:

- Catalog - TI's standard catalog product

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



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

IMPORTANT NOTICE

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

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

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

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

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

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

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

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

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

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

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Broadband	www.ti.com/broadband
DSP	dsp.ti.com	Digital Control	www.ti.com/digitalcontrol
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Military	www.ti.com/military
Logic	logic.ti.com	Optical Networking	www.ti.com/opticalnetwork
Power Mgmt	power.ti.com	Security	www.ti.com/security
Microcontrollers	microcontroller.ti.com	Telephony	www.ti.com/telephony
RFID	www.ti-rfid.com	Video & Imaging	www.ti.com/video
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Wireless	www.ti.com/wireless

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

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View SM32C6414DGLZ50AEP on WIN SOURCE
- Texas Instruments Information

Optimize Your Supply Chain with WIN SOURCE Solutions

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