# TMS320C6452 Digital Signal Processor

**Check for Samples: TMS320C6452**

## 1 Features

- High-Performance Digital Media Processor
  - 720-MHz, 900-MHz C64x+™ Clock Rates
  - 1.39 ns (-720), 1.11 ns (-900) Instruction Cycle Time
  - 5760, 7200 MIPS
  - Eight 32-Bit C64x+ Instructions/Cycle
  - Fully Software-Compatible With C64x/Debug
  - Commercial Temperature Ranges (-720, -900 only)
  - Industrial Temperature Ranges (-720, -900 only)
- VelociTI.2™ Extensions to VelociTI™ Advanced Very-Long-Instruction-Word (VLIW) TMS320C64x+™ DSP Core
  - Eight Highly Independent Functional Units With VelociTI.2 Extensions:
    - Six ALUs (32-/40-Bit), Each Supports Single 32-bit, Dual 16-bit, or Quad 8-bit Arithmetic per Clock Cycle
    - Two Multipliers Support Four 16 x 16-bit Multiplies (32-bit Results) per Clock Cycle or Eight 8 x 8-bit Multiplies (16-Bit Results) per Clock Cycle
  - Load-Store Architecture With Non-Aligned Support
  - 64 32-bit General-Purpose Registers
  - Instruction Packing Reduces Code Size
  - All Instructions Conditional
  - Additional C64x+™ Enhancements
    - Protected Mode Operation
    - Exceptions Support for Error Detection and Program Redirection
    - Hardware Support for Modulo Loop Auto-Focus Module Operation
- C64x+ 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
  - C64x+ Extensions
    - Compact 16-bit Instructions
    - Additional Instructions to Support Complex Multiplies
- C64x+ L1/L2 Memory Architecture
  - 256K-bit (32K-byte) L1P Program RAM/Cache [Direct Mapped]
  - 256K-bit (32K-byte) L1D Data RAM/Cache [2-Way Set-Associative]
  - 1408KB L2 Unified Mapped RAM/Cache [Flexible Allocation]
- Supports Little Endian Mode Only
- External Memory Interfaces (EMIFs)
  - 32-Bit DDR2 SDRAM Memory Controller With 512M-Byte Address Space (1.8-V I/O)
  - Asynchronous 16-Bit Wide EMIF (EMIFA)
    - Up to 128M-Byte Total Address Reach
    - 64M-Byte Address Reach per CE Space
  - Glueless Interface to Asynchronous Memories (SRAM, Flash, and EEPROM)
  - Synchronous Memories (SBSRAM and ZBT SRAM)
  - Supports Interface to Standard Sync Devices and Custom Logic (FPGA, CPLD, ASICs, etc.)
- Enhanced Direct-Memory-Access (EDMA) Controller (64 Independent Channels)
- 3-Port Gigabit Ethernet Switch Subsystem
- Four 64-Bit General-Purpose Timers (Each Configurable as Two 32-Bit Timers)
- One UART (With RTS and CTS Flow Control)
- One 4-wire Serial Port Interface (SPI) With Two Chip-Selects
- Master/Slave Inter-Integrated Circuit (I2C Bus™)
- Two Telecom Serial Interface Ports (TSIP0/1)
- Multichannel Audio Serial Port (McASP)
  - Ten Serializers and SPDIF (DIT) Mode
- 16/32-Bit Host-Port Interface (HPI)
- Advanced Event Triggering (AET) Compatible
- 32-Bit 33-/66-MHz, 3.3-V Peripheral Component Interconnect (PCI) Master/Slave Interface Conforms to PCI Specification 2.3
- VLYNQ™ Interface (FPGA Interface)
- On-Chip ROM Bootloader
- Individual Power-Saving Modes
- Flexible PLL Clock Generators
- IEEE-1149.1 (JTAG™) Boundary-Scan-Compatible
- 32 General-Purpose I/O (GPIO) Pins (Multiplexed With Other Device Functions)

- **Package:**
  - **529-pin nFBGA (ZUT suffix)**
  - **19x19 mm 0.8 mm pitch BGA**
  - **0.09-µm/6-Level Cu Metal Process (CMOS)**
- **3.3-V and 1.8-V I/O, 1.2-V Internal (-720,-900)**

## 1.1 Applications

- **Medical Diagnostics**
- **Machine Vision/Inspection**
- **Radar and Sonar**
- **Military/Aerospace**
- **Communications**

## 1.2 Trademarks

TMS320C64x+, C64x, C64x+, VelociTI, VelociTI.2, VLYNQ, TMS320C6000, C6000, TI, and TMS320 are trademarks of Texas Instruments.

I2C Bus is a registered trademark of Koninklijke Philips Electronics N.V.

Windows is a registered trademark of Microsoft Corporation in the United States and/or other countries.

All trademarks are the property of their respective owners.

## 1.3 Description

The TMS320C64x+™ DSPs (including the TMS320C6452 device is the highest-performance fixed-point DSP generation in the TMS320C6000™ DSP platform. The C6452 device is based on the third-generation high-performance, advanced VelociTI™ very-long-instruction-word (VLIW) architecture developed by Texas Instruments (TI), making these DSPs an excellent choice for digital media applications. The C64x+™ devices are upward code-compatible from previous devices that are part of the C6000™ DSP platform. The C64x™ DSPs support added functionality and have an expanded instruction set from previous devices.

Any reference to the C64x DSP or C64x CPU also applies, unless otherwise noted, to the C64x+ DSP and C64x+ CPU, respectively.

With performance of up to 7200 million instructions per second (MIPS) at a clock rate of 900MHz, the C64x+ core offers solutions to high-performance DSP programming challenges. The DSP core possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. The C64x+ DSP core processor has 64 general-purpose registers of 32-bit word length and eight highly independent functional units—two multipliers for a 32-bit result and six arithmetic logic units (ALUs). The eight functional units include instructions to accelerate the performance in video and imaging applications. The DSP core can produce four 16-bit multiply-accumulates (MACs) per cycle for up to 3600 million MACs per second (MMACS), or eight 8-bit MACs per cycle for up tp 8800 MMACS. For more details on the C64x+ DSP, see the *TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide* (literature number SPRU732).

The devices also have application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000 DSP platform devices. The core uses a two-level cache-based architecture. The Level 1 program cache (L1P) is a 256K-bit direct mapped cache and the Level 1 data cache (L1D) is a 256K-bit 2-way set-associative cache. The Level 2 memory/cache (L2) consists of a 1408KB memory space that is shared between program and data space. L2 memory can be configured as mapped memory, cache, or combinations of the two.

The device has a 1000 Mbps Ethernet Switch Subsystem with a management data input/output (MDIO) module and two SGMII ports; a 4-bit transmit, 4-bit receive VLYNQ interface; an inter-integrated circuit (I2C) bus interface; a multichannel audio serial port (McASP) with ten serializers; two telecom serial interface ports (TSIP); four 64-bit general-purpose timers each configurable as two independent 32-bit timers; a user-configurable 16-bit or 32-bit host-port interface (HPI); 32 pins for general-purpose input/output (GPIO) with programmable interrupt/event generation modes, multiplexed with other peripherals; one UART; and two glueless external memory interfaces: a synchronous and asynchronous external memory interface (EMIFA) for slower memories/peripherals, and a higher DDR2 SDRAM interface.

The management data input/output (MDIO) module continuously polls all 32 MDIO addresses to enumerate all PHY devices in the system.

The I2C and VLYNQ ports allow the device to easily control peripheral modules and/or communicate with host processors.

The rich peripheral set provides the ability to control external peripheral devices and communicate with external processors. For details on each of the peripherals, see the related sections later in this document and the associated peripheral reference guides.

The devices have a complete set of development tools. These include C compilers, a DSP assembly optimizer to simplify programming and scheduling, and a Windows™ debugger interface for visibility into source code execution.

## 1.4 Functional Block Diagram

Figure 1-1 shows the functional block diagram of the device.

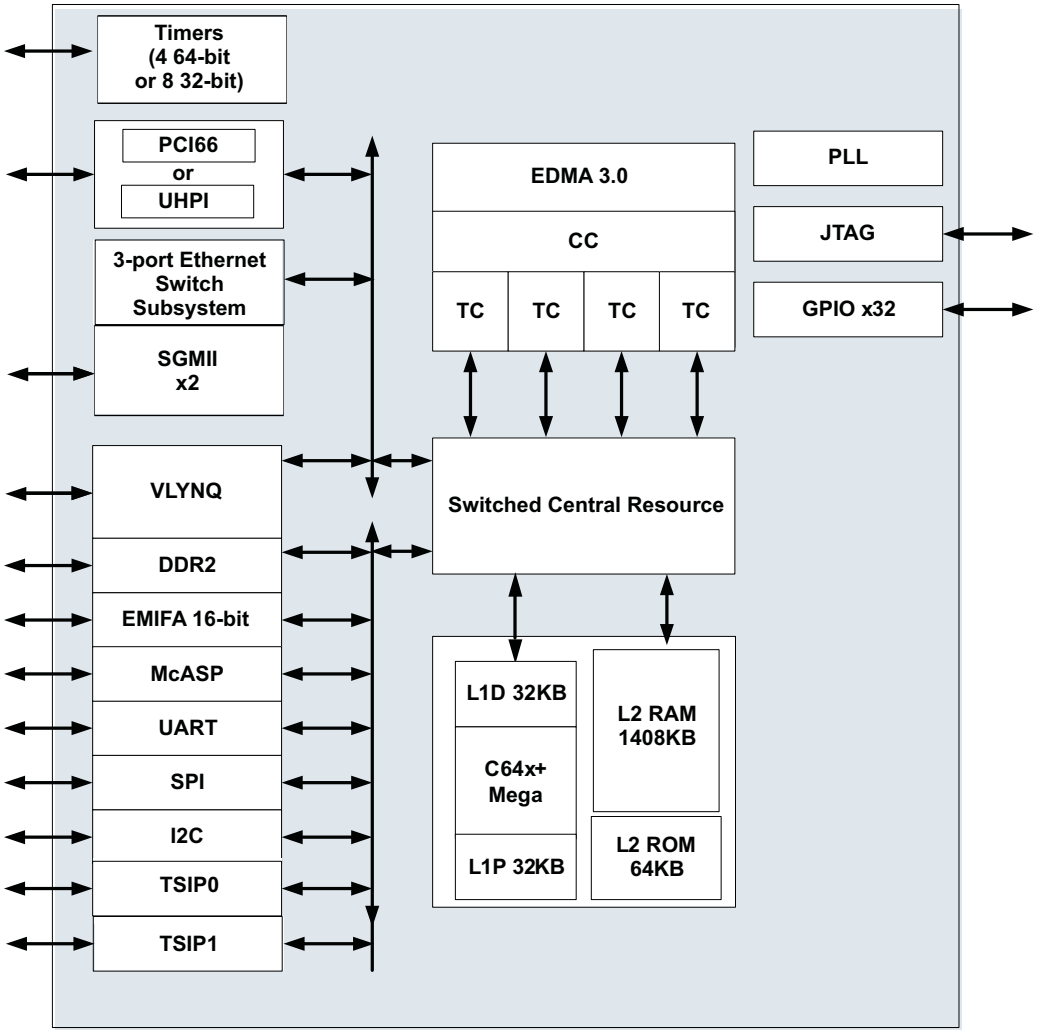



**Figure 1-1. Functional Block Diagram**

## Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

This data manual revision history highlights the technical changes made to the SPRS371E device-specific data manual to make it an SPRS371F revision. Applicable updates to the TMS320C645x Digital Signal Processors device family, specifically relating to the TMS320C6452 devices which are now in the production data (PD) stage of development, have been incorporated.

### TMS320C6452 Revision History

| SEE | ADDITIONS, DELETIONS, MODIFICATIONS |
|---|---|
| Section 1 | External Memory Interfaces (EMIFs):<br>• Deleted the footnote reference (1) from "512M-Byte Address Space (1.8-V I/O)" subbullet<br>• Deleted associated footnote **[devices are PD; SR1.0 was TMX]** |
| Section 1.1 | Added separate Applications section (new format structure) |
| Table 2-4 | Terminal Functions, Power Pins:<br>• Added "or left unconnected" to the $V_{CCMON}$ (L19) pin DESCRIPTION column [For the Advisory, see the Device-Specific Silicon Errata.] |
| Figure 6-5 | Corrected CLKDIR to VLYNQ to show 0 is VLYNQ, external |
| Section 6.4.1 | PLL1 Controller Device-Specific Information<br>• Added "[When SYSCLK4 is used as the EMIF input clock source, the actual clock goes through a divider and the frequency would be SYSCLK4 divide-by-2 (see Figure 6-5, *PLL Input Clock*).]" to the *SYSCLK4 is used as the EMIFA AECLKOUT* bullet **[Cleared Doc Feedback]** |
| Table 6-44 | Timing Requirements for Asynchronous Memory Cycles for EMIFA Module:<br>• Changed 3ns ***to*** 0ns for $t_h$(AOEH-EDV) |
| Figure 6-21 | Updated to show times 3 and 4 are referred to rising edge $\overline{\text{AAOE}}$/$\overline{\text{ASOE}}$. |
| Section 7.2 | Deleted duplicated Orderable Addendum table |

# 2 Device Overview

## 2.1 Device Characteristics

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

**Table 2-1. Characteristics of the Processor**

| HARDWARE FEATURES | | C6452 |
|---|---|---|
| Peripherals<br>Not all peripheral pins are available at the same time (For more detail, see Section 3.) | DDR2 memory controller (32-bit bus width) [1.8-V I/O] | 1 |
| | 16-bit bus width synchronous/asynchronous EMIF [EMIFA] | 1 |
| | EDMA3 (64 independent channels, 8 QDMA channels) | 1 |
| | Timers | 4 64-bit General Purpose<br>(each configurable as 1 64-bit ***or*** 2 32-bit) |
| | UART | (with RTS and CTS flow control) |
| | I2C | 1 (Master/Slave) |
| | SPI | 1 (4-wire, 2 chip select) |
| | McASP | 1 (10 serailizers) |
| | 3-port Ethernet Switch Subsystem supporting 10/100/1000 Base-T<br>Management data input/output (MDIO) | 2 SGMII ports available |
| | VLYNQ | 1 |
| | General-purpose input/output port (GPIO) | Up to 32 pins |
| | HPI (16/32-bit) | 1 |
| | PCI (32 bit) (33 MHz or 66 MHz) | 1 (PCI33 or PCI66) |
| | Telecom Serial Interface Port (TSIP) | 2 (TSIP0/1) |
| On-Chip Memory | Size (bytes) | 1408KB RAM, 64KB ROM |
| | Organization | 32KB L1 program (L1P)/cache<br>(Cache up to 32KB)<br>32KB L1 data (L1D)/cache<br>(Cache up to 32KB)<br>1408KB unified mapped RAM/Cache (L2)<br>64KB Boot ROM |
| MegaModule Rev ID | Revision ID Register (MM_REVID[15:0])<br>(address location 0x0181 2000) | 0x0003 |
| CPU ID + CPU Rev ID | Control Status Register (CSR.[31:16]) | 0x1000 |
| JTAG BSDL_ID | JTAGID register<br>(address location: 0x0204 9018) | 0x1B77 A02F |
| CPU Frequency | MHz | 720, 900 |
| Cycle Time | ns | 1.39 ns (-720), 1.11 ns (-900) |
| Voltage | Core (V) | 1.2 V (-720, -900) |
| | I/O (V) | 1.8 V, 3.3 V |
| PLL Options | CLKIN1 frequency multiplier | x1 (Bypass), x15, x20, x25, x30, x32 |
| BGA Package | | 529-Pin Flip Chip Plastic BGA (ZUT) |
| Process Technology | 0.09-µm/6-Level Cu Metal Process (CMOS) | 0.09 µm |
| Product Status(1) | Product Preview (PP), Advance Information (AI), or Production Data (PD) | PD |

(1) See Section 2.7 for a description of each stage of development.

## 2.2 CPU (DSP Core) Description

The C64x+ central processing unit (CPU) consists of eight functional units, two register files, and two data paths as shown in Figure 2-1. The two general-purpose register files (A and B) each contain 32 32-bit registers for a total of 64 registers. The general-purpose registers can be used for data or can be data address pointers. The data types supported include packed 8-bit data, packed 16-bit data, 32-bit data, 40-bit data, and 64-bit data. Values larger than 32 bits, such as 40-bit-long or 64-bit-long values are stored in register pairs, with the 32 LSBs of data placed in an even register and the remaining 8 or 32 MSBs in the next upper register (which is always an odd-numbered register).

The eight functional units (.M1, .L1, .D1, .S1, .M2, .L2, .D2, and .S2) are each capable of executing one instruction every clock cycle. The .M functional units perform all multiply operations. The .S and .L units perform a general set of arithmetic, logical, and branch functions. The .D units primarily load data from memory to the register file and store results from the register file into memory.

The C64x+ CPU extends the performance of the C64x core through enhancements and new features.

Each C64x+ .M unit can perform one of the following each clock cycle: one 32 x 32 bit multiply, one 16 x 32 bit multiply, two 16 x 16 bit multiplies, two 16 x 32 bit multiplies, two 16 x 16 bit multiplies with add/subtract capabilities, four 8 x 8 bit multiplies, four 8 x 8 bit multiplies with add operations, and four 16 x 16 multiplies with add/subtract capabilities (including a complex multiply). There is also support for Galois field multiplication for 8-bit and 32-bit data. Many communications algorithms such as FFTs and modems require complex multiplication. The complex multiply (CMPY) instruction takes for 16-bit inputs and produces a 32-bit real and a 32-bit imaginary output. There are also complex multiplies with rounding capability that produces one 32-bit packed output that contain 16-bit real and 16-bit imaginary values. The 32 x 32 bit multiply instructions provide the extended precision necessary for audio and other high-precision algorithms on a variety of signed and unsigned 32-bit data types.

The .L or (Arithmetic Logic Unit) now incorporates the ability to do parallel add/subtract operations on a pair of common inputs. Versions of this instruction exist to work on 32-bit data or on pairs of 16-bit data performing dual 16-bit add and subtracts in parallel. There are also saturated forms of these instructions.

The C64x+ core enhances the .S unit in several ways. In the C64x core, dual 16-bit MIN2 and MAX2 comparisons were available only on the .L units. On the C64x+ core they are also available on the .S unit, which increases the performance of algorithms that do searching and sorting. Finally, to increase data packing and unpacking throughput, the .S unit allows sustained high performance for the quad 8-bit/16-bit and dual 16-bit instructions. Unpack instructions prepare 8-bit data for parallel 16-bit operations. Pack instructions return parallel results to output precision including saturation support.

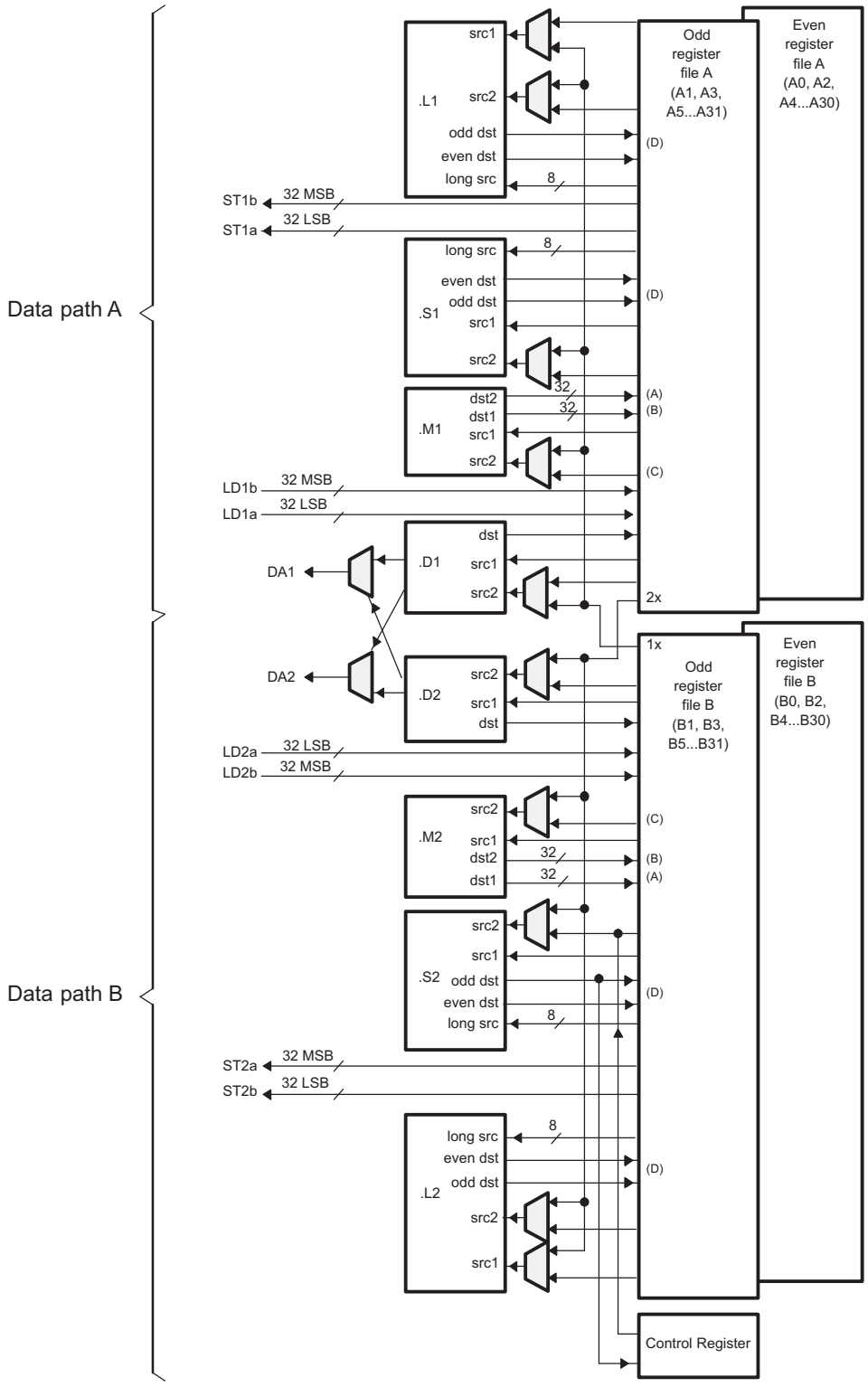
Other new features include:

- **SPLOOP** - A small instruction buffer in the CPU that aids in creation of software pipelining loops where multiple iterations of a loop are executed in parallel. The SPLOOP buffer reduces the code size associated with software pipelining. Furthermore, loops in the SPLOOP buffer are fully interruptible.
- **Compact Instructions** - The native instruction size for the C6000 devices is 32 bits. Many common instructions such as MPY, AND, OR, ADD, and SUB can be expressed as 16 bits if the C64x+ compiler can restrict the code to use certain registers in the register file. This compression is performed by the code generation tools.
- **Instruction Set Enhancement** - As noted above, there are new instructions such as 32-bit multiplications, complex multiplications, packing, sorting, bit manipulation, and 32-bit Galois field multiplication.
- **Exceptions Handling** - Intended to aid the programmer in isolating bugs. The C64x+ CPU is able to detect and respond to exceptions, both from internally detected sources (such as illegal opcodes).
- **Privilege** - Defines user and supervisor modes of operation, allowing the operating system to give a basic level of protection to sensitive resources. Local memory is divided into multiple pages, each with read, write, and execute permissions.

- **Time-Stamp Counter** - Primarily targeted for real-time operating system (RTOS) robustness, a free-running time-stamp counter that is ***not*** sensitive to system stalls is implemented in the CPU.

For more details on the C64x+ CPU and its enhancements over the C64x architecture, see the following documents:

- *TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide* (literature number SPRU732)
- *TMS320C64x+ DSP Megamodule Reference Guide* (literature number SPRU871)
- *TMS320C64x to TMS320C64x+ CPU Migration Guide Application Report* (literature number SPRAA84)
- *TMS320C64x+ DSP Cache User's Guide* (literature number SPRU862)




A. On .M unit, dst2 is 32 MSB.
B. On .M unit, dst1 is 32 LSB.
C. On C64x CPU .M unit, src2 is 32 bits; on C64x+ CPU .M unit,src2 is 64 bits.
D. On .L and .S units, odd dst connects to odd register files andeven dst connects to even register files.

**Figure 2-1. TMS320C64x+™ CPU (DSP Core) Data Paths**

## 2.3 C64x+ CPU

The C64x+ core uses a two-level cache-based architecture. The Level 1 program memory/cache (L1P) consists of 32KB memory space that can be configured as mapped memory or direct mapped cache. The Level 1 data memory/cache (L1D) consists of 32KB that can be configured as mapped memory or 2-way associated cache. The Level 2 memory/cache (L2) consists of a 1408KB memory space that is shared between program and data space. L2 memory can be configured as mapped memory, cache, or a combination of both.

Table 2-2 shows a memory map of the C64x+ CPU cache registers for the device.

**Table 2-2. C64x+ Cache Registers**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0184 0000 | L2CFG | L2 cache configuration register |
| 0x0184 0020 | L1PCFG | L1P size cache configuration register |
| 0x0184 0024 | L1PCC | L1P freeze mode cache configuration register |
| 0x0184 0040 | L1DCFG | L1D size cache configuration register |
| 0x0184 0044 | L1DCC | L1D freeze mode cache configuration register |
| 0x0184 0048 - 0x0184 0FFC | - | Reserved |
| 0x0184 1000 | - | Reserved |
| 0x0184 1004 - 0x0184 1FFC | - | Reserved |
| 0x0184 2000 | L2ALLOC0 | L2 allocation register 0 |
| 0x0184 2004 | L2ALLOC1 | L2 allocation register 1 |
| 0x0184 2008 | L2ALLOC2 | L2 allocation register 2 |
| 0x0184 200C | L2ALLOC3 | L2 allocation register 3 |
| 0x0184 2010 - 0x0184 3FFF | - | Reserved |
| 0x0184 4000 | L2WBAR | L2 writeback base address register |
| 0x0184 4004 | L2WWC | L2 writeback word count register |
| 0x0184 4010 | L2WIBAR | L2 writeback invalidate base address register |
| 0x0184 4014 | L2WIWC | L2 writeback invalidate word count register |
| 0x0184 4018 | L2IBAR | L2 invalidate base address register |
| 0x0184 401C | L2IWC | L2 invalidate word count register |
| 0x0184 4020 | L1PIBAR | L1P invalidate base address register |
| 0x0184 4024 | L1PIWC | L1P invalidate word count register |
| 0x0184 4030 | L1DWIBAR | L1D writeback invalidate base address register |
| 0x0184 4034 | L1DWIWC | L1D writeback invalidate word count register |
| 0x0184 4038 | - | Reserved |
| 0x0184 4040 | L1DWBAR | L1D block writeback |
| 0x0184 4044 | L1DWWC | L1D block writeback |
| 0x0184 4048 | L1DIBAR | L1D invalidate base address register |
| 0x0184 404C | L1DIWC | L1D invalidate word count register |
| 0x0184 4050 - 0x0184 4FFF | - | Reserved |
| 0x0184 5000 | L2WB | L2 writeback all register |
| 0x0184 5004 | L2WBINV | L2 writeback invalidate all register |
| 0x0184 5008 | L2INV | L2 global invalidate without writeback |
| 0x0184 500C - 0x0184 5027 | - | Reserved |
| 0x0184 5028 | L1PINV | L1P global invalidate |
| 0x0184 502C - 0x0184 5039 | - | Reserved |
| 0x0184 5040 | L1DWB | L1D global writeback |
| 0x0184 5044 | L1DWBINV | L1D global writeback with invalidate |
| 0x0184 5048 | L1DINV | L1D global invalidate without writeback |

**Table 2-2. C64x+ Cache Registers (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0184 8000 - 0x0184 80FC | MAR0 - MAR63 | Reserved 0x0000 0000 - 0x3FFF FFFF |
| 0x0184 80C0 - 0x0184 80FC | MAR48 - MAR63 | Reserved 0x3000 0000 - 0x3FFF FFFF |
| 0x0184 8100 - 0x0184 813C | MAR64 - MAR79 | Memory attribute registers for PCI Data 0x4000 0000 - 0x4FFF FFFF |
| 0x0184 8140 - 0x0184 827C | MAR80 - MAR159 | Reserved 0x5000 0000 - 0x9FFF FFFF |
| 0x0184 8280 - 0x0184 82BC | MAR160 - MAR175 | Memory attribute registers for EMIFA CE2 0xA000 0000 - 0xA3FF FFFF |
| 0x0184 82C0 - 0x0184 82FC | MAR176 - MAR191 | Memory attribute registers for EMIFA CE3 0xB000 0000 - 0xB3FF FFFF |
| 0x0184 8130 - 0x0184 813C | MAR76 - MAR79 | Memory Attribute Registers for VLYNQ 0x4C00 0000 - 0x4FFF FFFF |
| 0x0184 8300 - 0x0184 837C | MAR192 - MAR223 | Reserved 0xC000 0000 - 0xDFFF FFFF |
| 0x0184 8380 - 0x0184 83BC | MAR224 - MAR239 | Memory attribute registers for DDR2 0xE000 0000 - 0xEFFF FFFF |
| 0x0184 83C0 - 0x0184 83FC | MAR240 - MAR255 | Memory attribute registers for DDR2 0xF000 0000 - 0xFFFF FFFF |

## 2.4 Memory Map Summary

Table 2-3 shows the memory map address ranges of the device. The device has multiple on-chip memories associated with its processor and various subsystems. To help simplify software development, a unified memory map is used where possible to maintain a consistent view of device resources across all bus masters.

**Table 2-3. Memory Map Summary**

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ MEMORY MAP |
|---|---|---|---|
| 0x0000 0000 | 0x000F FFFF | 1M | Reserved |
| 0x0010 0000 | 0x0011 FFFF | 128K | Reserved |
| 0x0020 0000 | 0x007F FFFF | 6M | Reserved |
| 0x0080 0000 | 0x0080 FFFF | 64K | Internal ROM |
| 0x0080 1000 | 0x009F FFFF | 2M - 64K | Reserved |
| 0x0080 0000 | 0x0080 FFFF | 64K | Internal ROM |
| 0x0080 1000 | 0x009F FFFF | 2M - 64K | Reserved |
| 0x00A0 0000 | 0x00B5 FFFF | 1408K | L2 SRAM |
| 0x00B6 0000 | 0x00DF FFFF | 4M - 1408K | Reserved |
| 0x00A8 0000 | 0x00DF FFFF | 4M - 512K | Reserved |
| 0x00E0 0000 | 0x00E0 7FFF | 32K | L1P SRAM |
| 0x00E0 8000 | 0x00EF FFFF | 1M – 32K | Reserved |
| 0x00F0 0000 | 0x00F0 7FFF | 32K | L1D SRAM |
| 0x00F0 8000 | 0x00FF FFFF | 1M – 32K | Reserved |
| 0x0100 0000 | 0x017F FFFF | 8M | Reserved |
| 0x0180 0000 | 0x0180 FFFF | 64K | C64x+ Interrupt Controller |
| 0x0181 0000 | 0x0181 0FFF | 4K | C64x+ Power-down Control |
| 0x0181 1000 | 0x0181 1FFF | 4K | C64x+ Security ID |
| 0x0181 2000 | 0x0181 2FFF | 4K | C64x+ Revision ID |
| 0x0181 3000 | 0x0181 FFFF | 52K | Reserved |
| 0x0182 0000 | 0x0182 040F | 1040B | C64x+ EMC |
| 0x0182 0410 | 0x0182 FFFF | 64K – 16 | Reserved |
| 0x0183 0000 | 0x0183 FFFF | 64K | Reserved |
| 0x0184 0000 | 0x0184 FFFF | 64K | C64x+ Memory control |
| 0x0185 0000 | 0x01BB FFFF | 3, 520K | Reserved |
| 0x01BC 0000 | 0x01BC FFFF | 64K | Emulation |
| 0x01BD 0000 | 0x01BD FFFF | 64K | Reserved |
| 0x01BE 0000 | 0x01BF FFFF | 128K | Reserved |
| 0x01BE 0000 | 0x01FF FFFF | 4.125M | Reserved |
| 0x0200 0000 | 0x0200 007F | 128B | HPI Control |

**Table 2-3. Memory Map Summary (continued)**

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ MEMORY MAP |
|---|---|---|---|
| 0x0200 0080 | 0x0203 FFFF | 256K – 128 | Reserved |
| 0x0204 0000 | 0x0204 3FFF | 16K | McASP Control |
| 0x0204 4000 | 0x0204 43FF | 1K | McASP Data |
| 0x0204 4400 | 0x0204 47FF | 1K | Timer0 |
| 0x0204 4800 | 0x0204 4BFF | 1K | Timer1 |
| 0x0204 4C00 | 0x0204 4FFF | 1K | Timer2 |
| 0x0204 5000 | 0x0204 53FF | 1K | Timer3 |
| 0x0204 5400 | 0x0204 5FFF | 3K | Reserved |
| 0x0204 6000 | 0x0204 6FFF | 4K | PSC |
| 0x0204 7000 | 0x0204 73FF | 1K | UART |
| 0x0204 7400 | 0x0204 77FF | 1K | Reserved |
| 0x0204 7800 | 0x0204 7BFF | 1K | SPI |
| 0x0204 7C00 | 0x0204 7FFF | 1K | I2C Data and Control |
| 0x0204 8000 | 0x0204 83FF | 1K | GPIO |
| 0x0204 8400 | 0x0204 87FF | 1K | PCI Control |
| 0x0204 8800 | 0x0204 8FFF | 2K | Reserved |
| 0x0204 9000 | 0x0204 9FFF | 4K | Chip-Level Registers |
| 0x0204 A000 | 0x0207 FFFF | 216K | Reserved |
| 0x0208 0000 | 0x0209 FFFF | 128K | Reserved |
| 0x020A 0000 | 0x020D FFFF | 256K | Reserved |
| 0x020E 0000 | 0x020E 01FF | 512 | PLL Controller 1 |
| 0x020E 0200 | 0x0211 FFFF | 256K – 512 | Reserved |
| 0x0212 0000 | 0x0212 01FF | 512 | PLL Controller 2 |
| 0x0212 0200 | 0x0215 FFFF | 256K – 512 | Reserved |
| 0x0216 0000 | 0x029C FFFF | 9M - 576K | Reserved |
| 0x02A0 0000 | 0x02A0 7FFF | 32K | EDMA3CC |
| 0x02A0 8000 | 0x02A1 FFFF | 96K | Reserved |
| 0x02A2 0000 | 0x02A2 7FFF | 32K | EDMA3TC0 |
| 0x02A2 8000 | 0x02A2 FFFF | 32K | EDMA3TC1 |
| 0x02A3 0000 | 0x02A3 7FFF | 32K | EDMA3TC2 |
| 0x02A3 8000 | 0x02A3 FFFF | 32K | EDMA3TC3 |
| 0x02A4 0000 | 0x02A7 FFFF | 256K | Reserved |
| 0x02A8 0000 | 0x02A8 04FF | 1.25K | Reserved |
| 0x02A8 0500 | 0x02AB FFFF | 256K – 1.25K | Reserved |
| 0x02AC 0000 | 0x02AD FFFF | 128K | Reserved |
| 0x02AE 0000 | 0x02AF FFFF | 128K | Reserved |
| 0x02B0 0000 | 0x02B0 00FF | 256 | Reserved |
| 0x02B0 0100 | 0x02B0 3FFF | 16K – 256 | Reserved |
| 0x02B0 4000 | 0x02B0 407F | 128 | Reserved |
| 0x02B0 4080 | 0x02B3 FFFF | 256K – 128 | Reserved |
| 0x02B4 0000 | 0x02B4 01FF | 512 | Reserved |
| 0x02B4 0200 | 0x02B7 FFFF | 256K – 512 | Reserved |
| 0x02B8 0000 | 0x02B9 FFFF | 128K | Reserved |
| 0x02BA 0000 | 0x02BB FFFF | 128K | Reserved |
| 0x02BC 0000 | 0x02BF FFFF | 256K | Reserved |
| 0x02C0 0000 | 0x02C0 3FFF | 16K | Reserved |
| 0x02C0 4000 | 0x02C0 7FFF | 16K | Reserved |
| 0x02C0 8000 | 0x02C0 BFFF | 16K | Reserved |
| 0x02C0 C000 | 0x02C0 FFFF | 16K | Reserved |
| 0x02C1 0000 | 0x02C1 3FFF | 16K | Reserved |
| 0x02C1 4000 | 0x02C3 FFFF | 176K | Reserved |

**Table 2-3. Memory Map Summary (continued)**

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ MEMORY MAP |
|---|---|---|---|
| 0x02C4 0000 | 0x02C7 FFFF | 256K | TSIP1 |
| 0x02C8 0000 | 0x02CB FFFF | 256K | TSIP0 |
| 0x02CC 0000 | 0x02CF FFFF | 256K | Reserved |
| 0x02D0 0000 | 0x02D0 1FFF | 8K | Ethernet Subsystem CPPI RAM(1) |
| 0x02D0 2000 | 0x02D0 2FFF | 4K | Ethernet Subsystem Control |
| 0x02D0 3000 | 0x02D0 3FFF | 4K | Ethernet Subsystem 3PSW |
| 0x02D0 4000 | 0x02D0 47FF | 2K | Ethernet Subsystem MDIO |
| 0x02D0 4800 | 0x02D0 4BFF | 1K | Ethernet Subsystem SGMII0 |
| 0x02D0 4C00 | 0x02D0 4FFF | 1K | Ethernet Subsystem SGMII1 |
| 0x02D0 5000 | 0x02D0 57FF | 2K | Reserved |
| 0x02D0 5800 | 0x02DB FFFF | 746K | Reserved |
| 0x02DC 0000 | 0x02DF FFFF | 256K | Reserved |
| 0x02E0 0000 | 0x02E0 3FFF | 16K | Reserved |
| 0x02E0 4000 | 0x02FF FFFF | 2M – 16K | Reserved |
| 0x0300 0000 | 0x03FF FFFF | 16M | Reserved |
| 0x0400 0000 | 0x0FFF FFFF | 192M | Reserved |
| 0x1000 0000 | 0x1FFF FFFF | 256M | Reserved |
| 0x2000 0000 | 0x2FFF FFFF | 256M | Reserved |
| 0x3000 0000 | 0x3000 00FF | 256 | Reserved |
| 0x3000 0100 | 0x33FF FFFF | 64M – 256 | Reserved |
| 0x3400 0000 | 0x3400 00FF | 256 | Reserved |
| 0x3400 0100 | 0x37FF FFFF | 64M – 256 | Reserved |
| 0x3800 0000 | 0x3BFF FFFF | 64M | VLYNQ |
| 0x3C00 0000 | 0x3CFF FFFF | 16M | Reserved |
| 0x3D00 0000 | 0x3DFF FFFF | 16M | Reserved |
| 0x3E00 0000 | 0x3FFF FFFF | 32M | Reserved |
| 0x4000 0000 | 0x4FFF FFFF | 256M | PCI Data |
| 0x5000 0000 | 0x51FF FFFF | 32M | Reserved |
| 0x5200 0000 | 0x53FF FFFF | 32M | Reserved |
| 0x5400 0000 | 0x55FF FFFF | 32M | Reserved |
| 0x5600 0000 | 0x57FF FFFF | 32M | Reserved |
| 0x5800 0000 | 0x59FF FFFF | 32M | Reserved |
| 0x5A00 0000 | 0x5BFF FFFF | 32M | Reserved |
| 0x5C00 0000 | 0x5DFF FFFF | 32M | Reserved |
| 0x5E00 0000 | 0x5FFF FFFF | 32M | Reserved |
| 0x6000 0000 | 0x61FF FFFF | 32M | Reserved |
| 0x6200 0000 | 0x63FF FFFF | 32M | Reserved |
| 0x6400 0000 | 0x65FF FFFF | 32M | Reserved |
| 0x6600 0000 | 0x67FF FFFF | 32M | Reserved |
| 0x6800 0000 | 0x6FFF FFFF | 128M | Reserved |
| 0x7000 0000 | 0x77FF FFFF | 128M | EMIFA Configuration |
| 0x7800 0000 | 0x7FFF FFFF | 128M | DDR2 EMIF Configuration |
| 0x8000 0000 | 0x8FFF FFFF | 256M | Reserved |
| 0x9000 0000 | 0x9FFF FFFF | 256M | Reserved |
| 0xA000 0000 | 0xA3FF FFFF | 64M | EMIFA CE2 |
| 0xA400 0000 | 0xAFFF FFFF | 256 - 64M | Reserved |
| 0xB000 0000 | 0xB3FF FFFF | 64M | EMIFA CE3 |

(1) The 8K CPPI Descriptor memory is mapped to an address range 0x02C8 2000 - 0x02C8 3FFF, from the perspective of the Ethernet subsystem 3PSW. The buffer descriptors, when accessed from the C64x+, are addressed from 0x02D0 0000. However, within these buffer descriptors, when the pointer to the next buffer descriptor is programmed, the Ethernet subsystem 3PSW is interpreting this value. Thus, this programmed value should be in the address range starting from 0x02C8 2000.

**Table 2-3. Memory Map Summary (continued)**

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ MEMORY MAP |
|---|---|---|---|
| 0xB400 0000 | 0xBFFF FFFF | 256 - 64M | Reserved |
| 0xC000 0000 | 0xCFFF FFFF | 256M | Reserved |
| 0xD000 0000 | 0xDFFF FFFF | 256M | Reserved |
| 0xE000 0000 | 0xEFFF FFFF | 256M | DDR2 SDRAM |
| 0xF000 0000 | 0xFFFF FFFF | 256M | DDR2 SDRAM |

## 2.5 Pin Assignments

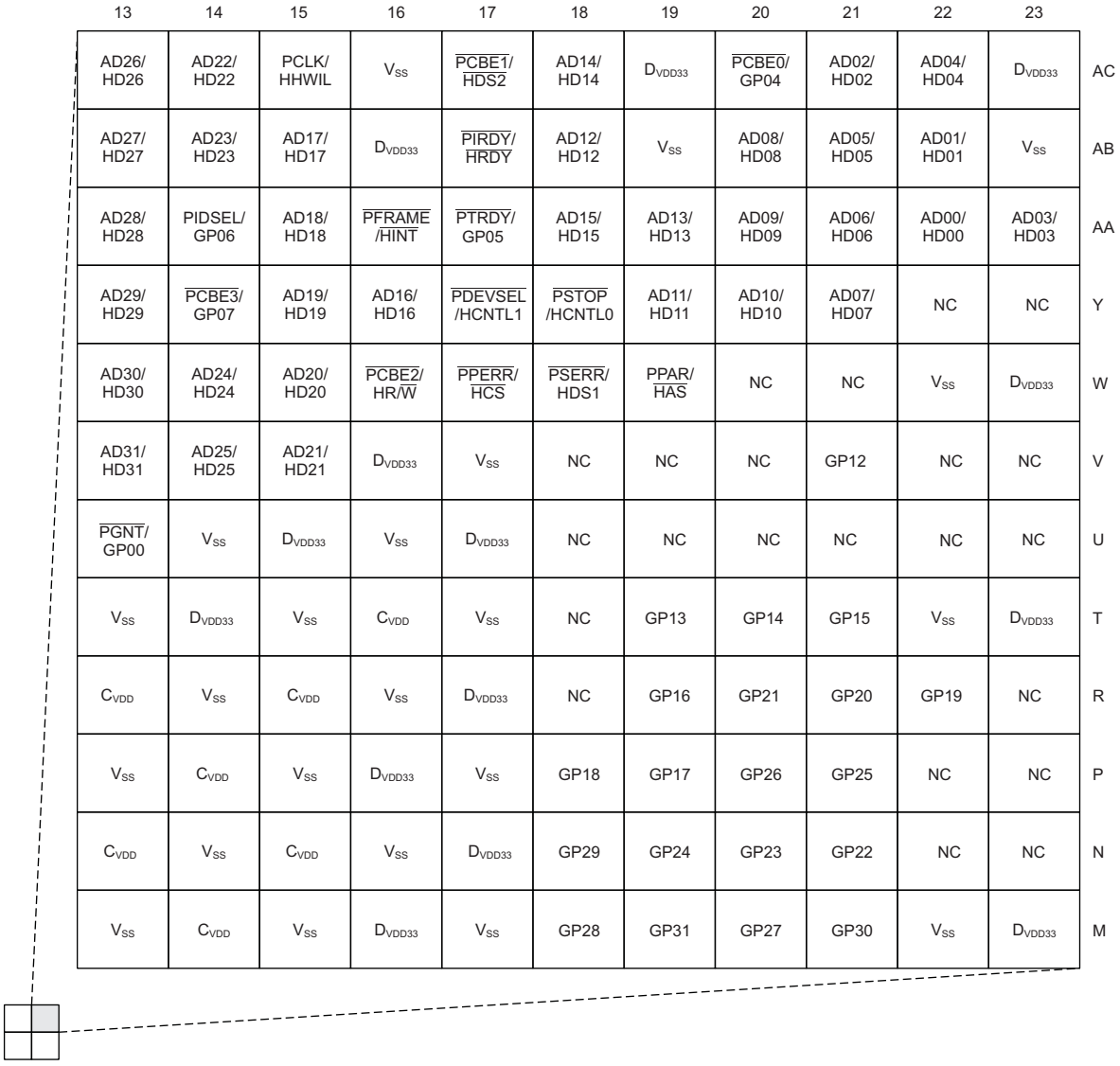
Extensive use of pin multiplexing is used to accommodate the largest number of peripheral functions in the smallest possible package. Pin multiplexing is controlled using a combination of hardware configuration at device reset and software programmable register settings. For more information on pin muxing, see Section 3.2.6, *PINMUX Register*.

### *2.5.1 Pin Map (Bottom View)*

Figure 2-2 through Figure 2-5 show the bottom view of the **ZUT** package pin assignments in four quadrants (A, B, C, and D).

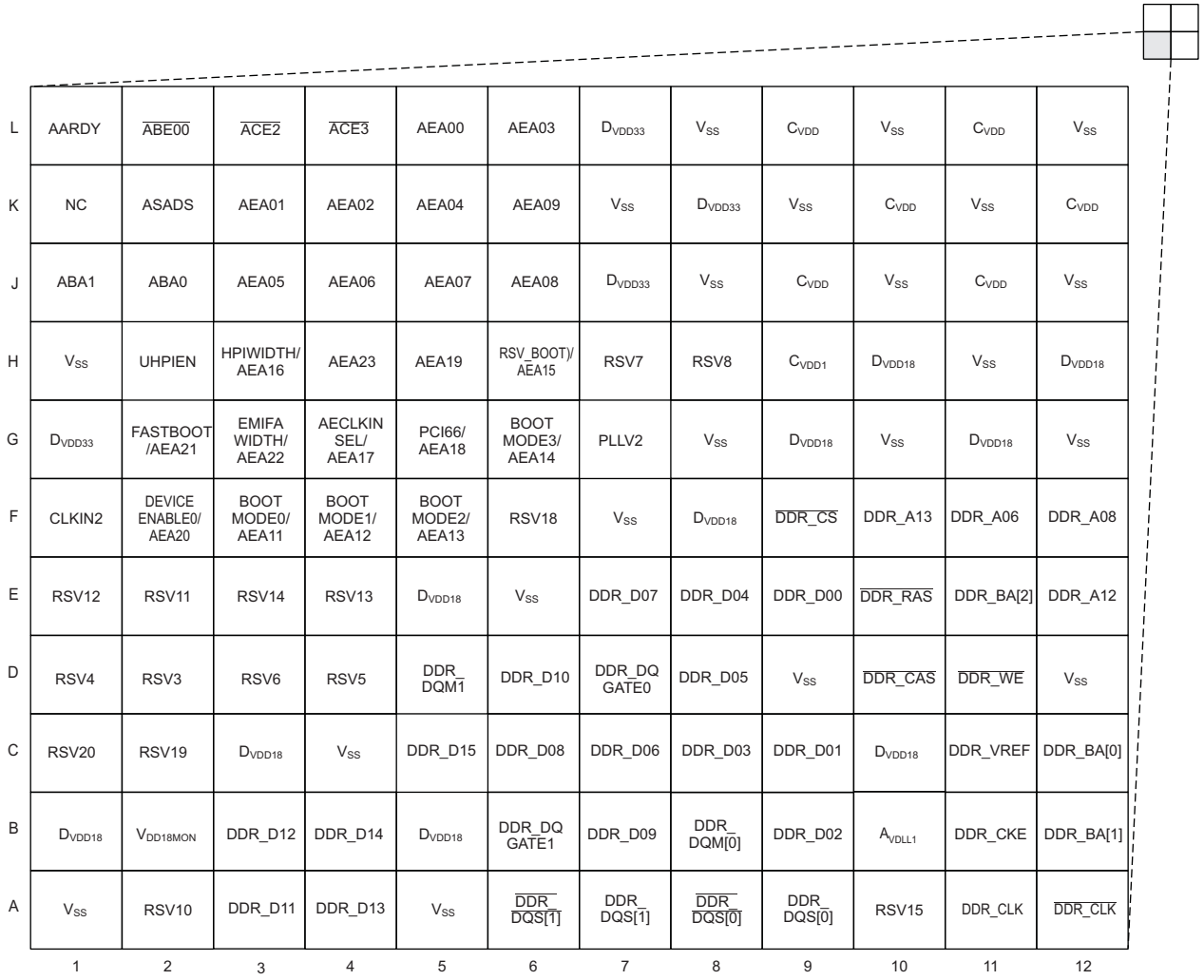
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AC | $V_{SS}$ | $D_{VDD33}$ | AHCLKX/ TSIP1CLKB | AHCLKR/ TSIP1CLKA | $D_{VDD33}$ | ACLKR/ TSIP0CLKA | ACLKX/ TSIP0CLKB | $V_{SS}$ | SGMII1RXN | $V_{SS}$ | REFCLKN | $V_{SS}$ |
| AB | NC | NC | AMUTEIN/ TSIP1FSA | AXR3/ TSIP0TX1 | $V_{SS}$ | AXR0/ TSIP0TR0 | $D_{VDD33}$ | $A_{VDDT}$ | SGMII1RXP | $A_{VDDR}$ | REFCLKP | $D_{VDD33}$ |
| AA | NC | NC | $V_{SS}$ | AXR6/ TSIP1TX0 | AXR9 | AXR2/ TSIP0TX0 | AFSX/ TSIP0FSB | $V_{SS}$ | SGMII0RXP | SGMII0RXN | $V_{SS}$ | $\overline{\text{PREQ}}$/ GP03 |
| Y | $\overline{\text{VSCRUN}}$ | NC | NC | $D_{VDD33}$ | AXR4/ TSIP1TR0 | AXR1/ TSIP0TR1 | AXR8 | $D_{VDD33}$ | $V_{SS}$ | $V_{SS}$ | SGMII0TXP | RSV21 |
| W | VCLK | VRXD0 | NC | NC | NC | AFSR/ TSIP0FSA | $V_{SS}$ | SGMII1TXP | SGMII1TXN | $A_{VDDA}$ | SGMII0TXN | RSV22 |
| V | $V_{SS}$ | $D_{VDD33}$ | VRXD1 | VRXD2 | NC | AXR7/ TSIP1TX1 | AXR5/ TSIP1TR1 | $C_{VDD}$ | $V_{SS}$ | RSV17 | $A_{VDDA}$ | $\overline{\text{PINTA}}$/ GP02 |
| U | VRXD3 | VTXD1 | VTXD0 | VTXD3 | VTXD2 | NC | AMUTE/ TSIP1FSB | MDIO | MDCLK | $D_{VDDD}$ | $A_{VDDT}$ | $\overline{\text{PRST}}$/ GP01 |
| T | AECLKIN | $\overline{\text{AAWE}}$/ $\overline{\text{ASWE}}$ | AED03 | AED02 | AED01 | AED00 | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $D_{VDDD}$ | $V_{SS}$ | $D_{VDD33}$ |
| R | AR/$\overline{\text{W}}$ | AED08 | AED07 | AED06 | AED05 | AED04 | $D_{VDD33}$ | $C_{VDD}$ | $C_{VDDESS}$ | $V_{SS}$ | $C_{VDDESS}$ | $V_{SS}$ |
| P | AECLKOUT | $\overline{\text{AAOE}}$/ $\overline{\text{ASOE}}$ | AED12 | AED11 | AED10 | AED09 | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ |
| N | $V_{SS}$ | $D_{VDD33}$ | PLLV1 | AED13 | AED15 | AED14 | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ |
| M | CLKIN1 | RSV9 | SYSCLK5 | $\overline{\text{ABE01}}$ | AEA10 | NC | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ |

**Figure 2-2. ZUT Pin Map [Top Left Quadrant]**



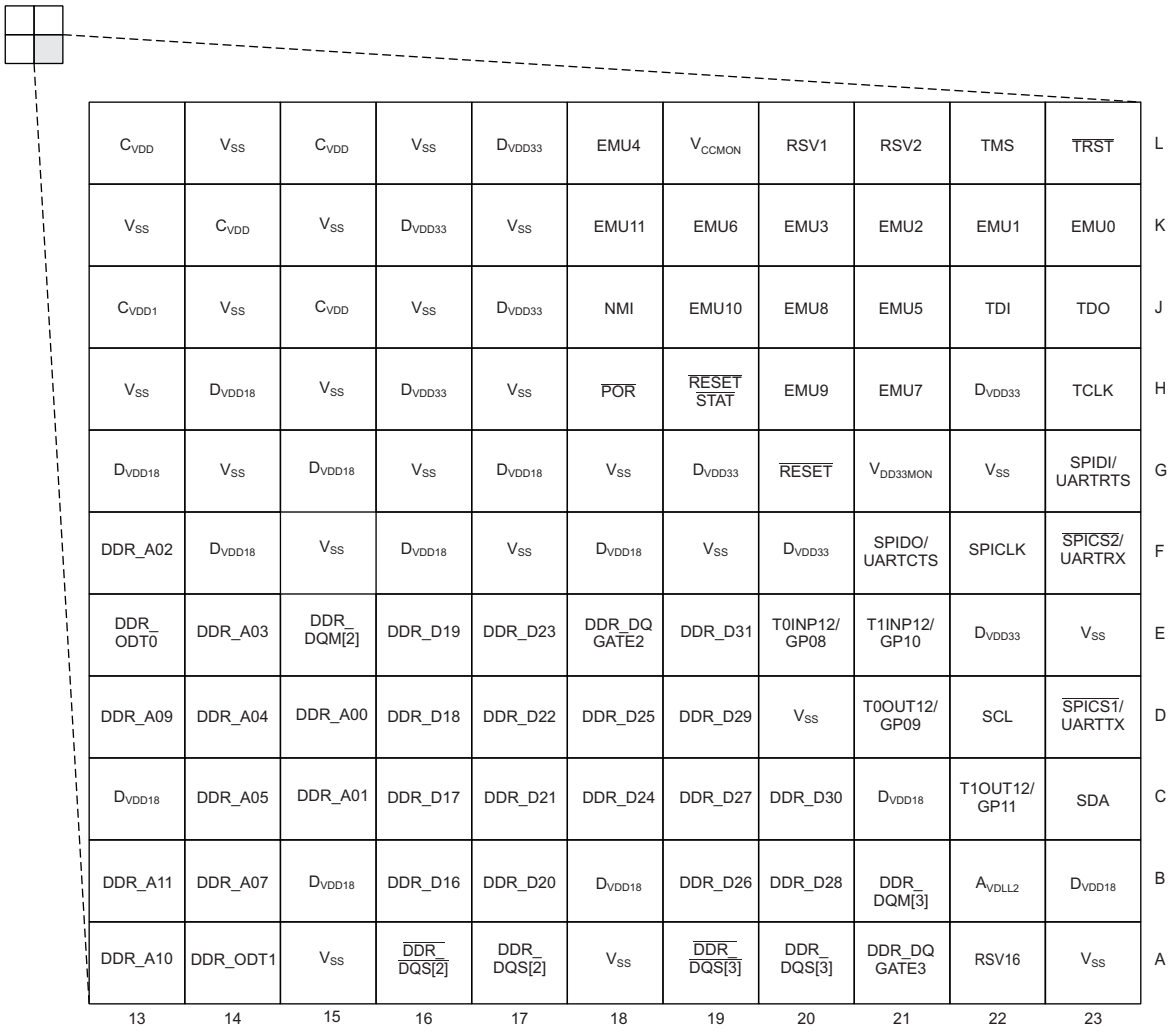
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| AD26/ HD26 | AD22/ HD22 | PCLK/ HHWIL | $V_{SS}$ | $\overline{PCBE1}$/ $\overline{HDS2}$ | AD14/ HD14 | $D_{VDD33}$ | $\overline{PCBE0}$/ GP04 | AD02/ HD02 | AD04/ HD04 | $D_{VDD33}$ | AC |
| AD27/ HD27 | AD23/ HD23 | AD17/ HD17 | $D_{VDD33}$ | $\overline{PIRDY}$/ $\overline{HRDY}$ | AD12/ HD12 | $V_{SS}$ | AD08/ HD08 | AD05/ HD05 | AD01/ HD01 | $V_{SS}$ | AB |
| AD28/ HD28 | PIDSEL/ GP06 | AD18/ HD18 | $\overline{PFRAME}$ /$\overline{HINT}$ | $\overline{PTRDY}$/ GP05 | AD15/ HD15 | AD13/ HD13 | AD09/ HD09 | AD06/ HD06 | AD00/ HD00 | AD03/ HD03 | AA |
| AD29/ HD29 | $\overline{PCBE3}$/ GP07 | AD19/ HD19 | AD16/ HD16 | $\overline{PDEVSEL}$ /HCNTL1 | $\overline{PSTOP}$ /HCNTL0 | AD11/ HD11 | AD10/ HD10 | AD07/ HD07 | NC | NC | Y |
| AD30/ HD30 | AD24/ HD24 | AD20/ HD20 | $\overline{PCBE2}$/ HR/$\overline{W}$ | $\overline{PPERR}$/ $\overline{HCS}$ | $\overline{PSERR}$/ HDS1 | PPAR/ $\overline{HAS}$ | NC | NC | $V_{SS}$ | $D_{VDD33}$ | W |
| AD31/ HD31 | AD25/ HD25 | AD21/ HD21 | $D_{VDD33}$ | $V_{SS}$ | NC | NC | NC | GP12 | NC | NC | V |
| $\overline{PGNT}$/ GP00 | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $D_{VDD33}$ | NC | NC | NC | NC | NC | NC | U |
| $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | NC | GP13 | GP14 | GP15 | $V_{SS}$ | $D_{VDD33}$ | T |
| $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | NC | GP16 | GP21 | GP20 | GP19 | NC | R |
| $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | GP18 | GP17 | GP26 | GP25 | NC | NC | P |
| $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | GP29 | GP24 | GP23 | GP22 | NC | NC | N |
| $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | GP28 | GP31 | GP27 | GP30 | $V_{SS}$ | $D_{VDD33}$ | M |

**Figure 2-3. ZUT Pin Map [Top Right Quadrant]**



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| L | AARDY | $\overline{ABE00}$ | $\overline{ACE2}$ | $\overline{ACE3}$ | AEA00 | AEA03 | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ |
| K | NC | ASADS | AEA01 | AEA02 | AEA04 | AEA09 | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ |
| J | ABA1 | ABA0 | AEA05 | AEA06 | AEA07 | AEA08 | $D_{VDD33}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ |
| H | $V_{SS}$ | UHPIEN | HPIWIDTH/ AEA16 | AEA23 | AEA19 | RSV_BOOT)/ AEA15 | RSV7 | RSV8 | $C_{VDD1}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ |
| G | $D_{VDD33}$ | FASTBOOT /AEA21 | EMIFA WIDTH/ AEA22 | AECLKIN SEL/ AEA17 | PCI66/ AEA18 | BOOT MODE3/ AEA14 | PLLV2 | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ |
| F | CLKIN2 | DEVICE ENABLE0/ AEA20 | BOOT MODE0/ AEA11 | BOOT MODE1/ AEA12 | BOOT MODE2/ AEA13 | RSV18 | $V_{SS}$ | $D_{VDD18}$ | $\overline{DDR\_CS}$ | DDR_A13 | DDR_A06 | DDR_A08 |
| E | RSV12 | RSV11 | RSV14 | RSV13 | $D_{VDD18}$ | $V_{SS}$ | DDR_D07 | DDR_D04 | DDR_D00 | $\overline{DDR\_RAS}$ | DDR_BA[2] | DDR_A12 |
| D | RSV4 | RSV3 | RSV6 | RSV5 | DDR_ $\overline{DQM1}$ | DDR_D10 | DDR_DQ $\overline{GATE0}$ | DDR_D05 | $V_{SS}$ | $\overline{DDR\_CAS}$ | $\overline{DDR\_WE}$ | $V_{SS}$ |
| C | RSV20 | RSV19 | $D_{VDD18}$ | $V_{SS}$ | DDR_D15 | DDR_D08 | DDR_D06 | DDR_D03 | DDR_D01 | $D_{VDD18}$ | DDR_VREF | DDR_BA[0] |
| B | $D_{VDD18}$ | $V_{DD18MON}$ | DDR_D12 | DDR_D14 | $D_{VDD18}$ | DDR_DQ $\overline{GATE1}$ | DDR_D09 | DDR_ DQM[0] | DDR_D02 | $A_{VDLL1}$ | DDR_CKE | DDR_BA[1] |
| A | $V_{SS}$ | RSV10 | DDR_D11 | DDR_D13 | $V_{SS}$ | $\overline{DDR\_}$ $\overline{DQS[1]}$ | DDR_ DQS[1] | $\overline{DDR\_}$ $\overline{DQS[0]}$ | DDR_ DQS[0] | RSV15 | DDR_CLK | $\overline{DDR\_CLK}$ |

**Figure 2-4. ZUT Pin Map [Bottom Left Quadrant]**



| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $C_{VDD}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | EMU4 | $V_{CCMON}$ | RSV1 | RSV2 | TMS | $\overline{TRST}$ | L |
| $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | EMU11 | EMU6 | EMU3 | EMU2 | EMU1 | EMU0 | K |
| $C_{VDD1}$ | $V_{SS}$ | $C_{VDD}$ | $V_{SS}$ | $D_{VDD33}$ | NMI | EMU10 | EMU8 | EMU5 | TDI | TDO | J |
| $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD33}$ | $V_{SS}$ | $\overline{POR}$ | $\overline{RESET}$ $\overline{STAT}$ | EMU9 | EMU7 | $D_{VDD33}$ | TCLK | H |
| $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD33}$ | $\overline{RESET}$ | $V_{DD33MON}$ | $V_{SS}$ | SPIDI/ UARTRTS | G |
| DDR_A02 | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD18}$ | $V_{SS}$ | $D_{VDD33}$ | SPIDO/ UARTCTS | SPICLK | $\overline{SPICS2}$/ UARTRX | F |
| DDR_ ODT0 | DDR_A03 | DDR_ DQM[2] | DDR_D19 | DDR_D23 | DDR_DQ GATE2 | DDR_D31 | T0INP12/ GP08 | T1INP12/ GP10 | $D_{VDD33}$ | $V_{SS}$ | E |
| DDR_A09 | DDR_A04 | DDR_A00 | DDR_D18 | DDR_D22 | DDR_D25 | DDR_D29 | $V_{SS}$ | T0OUT12/ GP09 | SCL | $\overline{SPICS1}$/ UARTTX | D |
| $D_{VDD18}$ | DDR_A05 | DDR_A01 | DDR_D17 | DDR_D21 | DDR_D24 | DDR_D27 | DDR_D30 | $D_{VDD18}$ | T1OUT12/ GP11 | SDA | C |
| DDR_A11 | DDR_A07 | $D_{VDD18}$ | DDR_D16 | DDR_D20 | $D_{VDD18}$ | DDR_D26 | DDR_D28 | DDR_ DQM[3] | $A_{VDLL2}$ | $D_{VDD18}$ | B |
| DDR_A10 | DDR_ODT1 | $V_{SS}$ | $\overline{DDR\_DQS[2]}$ | DDR_ DQS[2] | $V_{SS}$ | $\overline{DDR\_DQS[3]}$ | DDR_ DQS[3] | DDR_DQ GATE3 | RSV16 | $V_{SS}$ | A |

**Figure 2-5. ZUT Pin Map [Bottom Right Quadrant]**

## 2.6 Terminal Functions

The terminal functions tables (Table 2-4 through Table 2-5) identify the external signal names, the associated pin (ball) numbers along with the mechanical package designator, the pin type, whether the pin has any internal pullup or pulldown resistors, and a functional pin description. For more detailed information on device configuration, peripheral selection, multiplexed/shared pin, and debugging considerations, see Section 3.

All device boot and configuration pins are multiplexed with functional pins. These pins function as device boot and configuration pins only during device reset. When both the reset pin ($\overline{\text{RESET}}$) and the power-on reset pin ($\overline{\text{POR}}$) are deasserted, the input states of these multiplexed device boot and configuration pins are sampled and latched into the BOOTCFG register. For proper device operation, these pins must be pulled up/down to the desired value via an external resistor.

**Table 2-4. Terminal Functions**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| **Clock/PLL Configuration** | | | | | |
| CLKIN1 | M1 | I | IPD | 3.3 V | Clock Input for PLL1 |
| CLKIN2 | F1 | I | IPD | 3.3 V | Clock Input for PLL2 |
| REFCLKN(2) | AC11 | I | | | Differential Reference Clock input (negative) for SGMII |
| REFCLKP(2) | AB11 | I | | | Differential Reference Clock input (positive) for SGMII |
| PLLV1 | N3 | A | | 1.8 V | 1.8-V I/O Supply Voltage for PLL1 |
| PLLV2 | G7 | A | | 1.8 V | 1.8-V I/O Supply Voltage for PLL2 |
| SYSCLK5 | M3 | I/O/Z | IPD | 3.3 V | Clock out of device speed/4 |
| **JTAG** | | | | | |
| TCLK | H23 | I | IPU | 3.3 V | JTAG Test Port Clock |
| TDI | J22 | I | IPU | 3.3 V | JTAG Test Port Data In |
| TDO | J23 | OZ | IPU | 3.3 V | JTAG Test Port Data Out |
| TMS | L22 | I | IPU | 3.3 V | JTAG Test Port Mode Select |
| $\overline{\text{TRST}}$ | L23 | I | IPD | 3.3 V | JTAG Test Port Reset |
| EMU0 | K23 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 0 |
| EMU1 | K22 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 1 |
| EMU2 | K21 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 2 |
| EMU3 | K20 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 3 |
| EMU4 | L18 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 4 |
| EMU5 | J21 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 5 |
| EMU6 | K19 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 6 |
| EMU7 | H21 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 7 |
| EMU8 | J20 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 8 |
| EMU9 | H20 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 9 |
| EMU10 | J19 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 10 |
| EMU11 | K18 | I/O/Z | IPU | 3.3 V | JTAG Test Port Emulation 11 |
| **RESET/INTERRUPTS** | | | | | |
| NMI | J18 | I | IPD | 3.3 V | Nonmaskable Interrupt |
| $\overline{\text{RESETSTAT}}$ | H19 | O | | 3.3 V | Reset Status Pin |
| $\overline{\text{RESET}}$ | G20 | I | | 3.3 V | Device Reset |
| $\overline{\text{POR}}$ | H18 | I | | 3.3 V | Power On Reset |

(1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
(2) The clock input buffers on the REFCLKP/N pins are compatible with LVDS and LVPECL clock sources. These input buffers include a 100-Ω termination (P to N) and a common-mode biasing. Because the common-mode biasing is included, the clock source must be AC coupled.

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| **HOST-PORT INTERFACE (HPI) or PERIPHERAL COMPONENT INTERCONNECT (PCI) or GPIO[0:7]** | | | | | |
| AD00/HD00 | AA22 | I/O/Z | | 3.3 V | Host Port data [15:00] pin or PCI data-address bus [15:00] [default] |
| AD01/HD01 | AB22 | I/O/Z | | 3.3 V | |
| AD02/HD02 | AC21 | I/O/Z | | 3.3 V | |
| AD03/HD03 | AA23 | I/O/Z | | 3.3 V | |
| AD04/HD04 | AC22 | I/O/Z | | 3.3 V | |
| AD05/HD05 | AB21 | I/O/Z | | 3.3 V | |
| AD06/HD06 | AA21 | I/O/Z | | 3.3 V | |
| AD07/HD07 | Y21 | I/O/Z | | 3.3 V | |
| AD08/HD08 | AB20 | I/O/Z | | 3.3 V | |
| AD09/HD09 | AA20 | I/O/Z | | 3.3 V | |
| AD10/HD10 | Y20 | I/O/Z | | 3.3 V | |
| AD11/HD11 | Y19 | I/O/Z | | 3.3 V | |
| AD12/HD12 | AB18 | I/O/Z | | 3.3 V | |
| AD13/HD13 | AA19 | I/O/Z | | 3.3 V | |
| AD14/HD14 | AC18 | I/O/Z | | 3.3 V | |
| AD15/HD15 | AA18 | I/O/Z | | 3.3 V | |
| AD16/HD16 | Y16 | I/O/Z | | 3.3 V | Host Port data [31:16] pin or PCI data-address bus [31:16] [default] |
| AD17/HD17 | AB15 | I/O/Z | | 3.3 V | |
| AD18/HD18 | AA15 | I/O/Z | | 3.3 V | |
| AD19/HD19 | Y15 | I/O/Z | | 3.3 V | |
| AD20/HD20 | W15 | I/O/Z | | 3.3 V | |
| AD21/HD21 | V15 | I/O/Z | | 3.3 V | |
| AD22/HD22 | AC14 | I/O/Z | | 3.3 V | |
| AD23/HD23 | AB14 | I/O/Z | | 3.3 V | |
| AD24/HD24 | W14 | I/O/Z | | 3.3 V | |
| AD25/HD25 | V14 | I/O/Z | | 3.3 V | |
| AD26/HD26 | AC13 | I/O/Z | | 3.3 V | |
| AD27/HD27 | AB13 | I/O/Z | | 3.3 V | |
| AD28/HD28 | AA13 | I/O/Z | | 3.3 V | |
| AD29/HD29 | Y13 | I/O/Z | | 3.3 V | |
| AD30/HD30 | W13 | I/O/Z | | 3.3 V | |
| AD31/HD31 | V13 | I/O/Z | | 3.3 V | |
| PPAR/$\overline{\text{HAS}}$ | W19 | I/O/Z | | 3.3 V | Host Address Strobe (I) or PCI parity [default] |
| $\overline{\text{PSTOP}}$/HCNTL0 | Y18 | I/O/Z | | 3.3 V | Host Control selects between control, address, or data registers (I) or PCI Stop [default] |
| $\overline{\text{PDEVSEL}}$/HCNTL1 | Y17 | I/O/Z | | 3.3 V | Host Control selects between control, address, or data registers (I) or PCI Device Select [default] |
| $\overline{\text{PPERR}}$/$\overline{\text{HCS}}$ | W17 | I/O/Z | | 3.3 V | Host Chip Select (I) or PCI Parity Error [default] |
| $\overline{\text{PSERR}}$/ $\overline{\text{HDS1}}$ | W18 | I/O/Z | | 3.3 V | Host Data Strobe 1 (I) or PCI System Error [default] |
| $\overline{\text{PCBE0}}$/GP04 | AC20 | I/O/Z | | 3.3 V | PCI Command/Byte Enable 0 or GP[4] [default |
| $\overline{\text{PCBE1}}$/$\overline{\text{HDS2}}$ | AC17 | I | | 3.3 V | PCI Command/Byte Enable 1 or host data strobe 2 |
| $\overline{\text{PCBE2}}$/HR/$\overline{\text{W}}$ | W16 | I/O/Z | | 3.3 V | PCI Command/Byte Enable 2 or host read or write select (I) |
| $\overline{\text{PCBE3}}$/GP07 | Y14 | I/O/Z | | 3.3 V | PCI Command/Byte Enable 3 or GPIO[7] |
| PCLK/HHWIL | AC15 | I/O/Z | | 3.3 V | PCI Clock (I) [default] or host Half-word Select - first or second half-word (not necessarily high or low order) [For HPI16 bus width selection only] (I) |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| PFRAME/HINT | AA16 | I/O/Z | | 3.3 V | PCI Frame or host interrupt from DSP to host (O/Z) |
| PIRDY/HRDY | AB17 | I/O/Z | | 3.3 V | PCI Initiator Ready [default] or Host Ready from DSP to host (O/Z) |
| PGNT/GP00 | U13 | I/O/Z | | 3.3 V | PCI Bus Grant (I) or GPIO[0] |
| PRST/GP01 | U12 | I/O/Z | | 3.3 V | PCI Reset (I) or GPIO[1] |
| PINTA/GP02 | V12 | I/O/Z | | 3.3 V | PCI Interrupt A (O/Z) or GPIO[2] |
| PREQ/GP03 | AA12 | I/O/Z | | 3.3 V | PCI Bus Request (O/Z) or GPIO[3] |
| PTRDY/GP05 | AA17 | I/O/Z | | 3.3 V | PCI Target Ready or GPIO[5] |
| PIDSEL/GP06 | AA14 | I/O/Z | | 3.3 V | PCI Initialization Device Select (I) or GPIO[6] |
| **DDR2 MEMORY CONTROLLER** | | | | | |
| DDR_BA[0] | C12 | I/O/Z | | 1.8 V | DDR2 Memory Controller Bank Address Control |
| DDR_BA[1] | B12 | I/O/Z | | 1.8 V | |
| DDR_BA[2] | E11 | I/O/Z | | 1.8 V | |
| DDR_CS | F9 | I/O/Z | | 1.8 V | DDR2 Memory Controller Memory Space Enable |
| DDR_A00 | D15 | I/O/Z | | 1.8 V | DDR2 Memory Controller External Address |
| DDR_A01 | C15 | I/O/Z | | 1.8 V | |
| DDR_A02 | F13 | I/O/Z | | 1.8 V | |
| DDR_A03 | E14 | I/O/Z | | 1.8 V | |
| DDR_A04 | D14 | I/O/Z | | 1.8 V | |
| DDR_A05 | C14 | I/O/Z | | 1.8 V | |
| DDR_A06 | F11 | I/O/Z | | 1.8 V | |
| DDR_A07 | B14 | I/O/Z | | 1.8 V | |
| DDR_A08 | F12 | I/O/Z | | 1.8 V | |
| DDR_A09 | D13 | I/O/Z | | 1.8 V | |
| DDR_A10 | A13 | I/O/Z | | 1.8 V | |
| DDR_A11 | B13 | I/O/Z | | 1.8 V | |
| DDR_A12 | E12 | I/O/Z | | 1.8 V | |
| DDR_A13 | F10 | I/O/Z | | 1.8 V | |
| DDR_CLK | A12 | I/O/Z | | 1.8 V | Negative DDR2 Memory Controller Output Clock (CLKIN2 frequency x 10) |
| DDR_CLK | A11 | I/O/Z | | 1.8 V | DDR2 Memory Controller Output Clock (CLKIN2 frequency x 10) |
| DDR_D00 | E9 | I/O/Z | | 1.8 V | DDR2 Memory Controller External Data |
| DDR_D01 | C9 | I/O/Z | | 1.8 V | |
| DDR_D02 | B9 | I/O/Z | | 1.8 V | |
| DDR_D03 | C8 | I/O/Z | | 1.8 V | |
| DDR_D04 | E8 | I/O/Z | | 1.8 V | |
| DDR_D05 | D8 | I/O/Z | | 1.8 V | |
| DDR_D06 | C7 | I/O/Z | | 1.8 V | |
| DDR_D07 | E7 | I/O/Z | | 1.8 V | |
| DDR_D08 | C6 | I/O/Z | | 1.8 V | |
| DDR_D09 | B7 | I/O/Z | | 1.8 V | |
| DDR_D10 | D6 | I/O/Z | | 1.8 V | |
| DDR_D11 | A3 | I/O/Z | | 1.8 V | |
| DDR_D12 | B3 | I/O/Z | | 1.8 V | |
| DDR_D13 | A4 | I/O/Z | | 1.8 V | |
| DDR_D14 | B4 | I/O/Z | | 1.8 V | |
| DDR_D15 | C5 | I/O/Z | | 1.8 V | |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| DDR_D16 | B16 | I/O/Z | | 1.8 V | DDR2 Memory Controller External Data (continued) |
| DDR_D17 | C16 | I/O/Z | | 1.8 V | |
| DDR_D18 | D16 | I/O/Z | | 1.8 V | |
| DDR_D19 | E16 | I/O/Z | | 1.8 V | |
| DDR_D20 | B17 | I/O/Z | | 1.8 V | |
| DDR_D21 | C17 | I/O/Z | | 1.8 V | |
| DDR_D22 | D17 | I/O/Z | | 1.8 V | |
| DDR_D23 | E17 | I/O/Z | | 1.8 V | |
| DDR_D24 | C18 | I/O/Z | | 1.8 V | |
| DDR_D25 | D18 | I/O/Z | | 1.8 V | |
| DDR_D26 | B19 | I/O/Z | | 1.8 V | |
| DDR_D27 | C19 | I/O/Z | | 1.8 V | |
| DDR_D28 | B20 | I/O/Z | | 1.8 V | |
| DDR_D29 | D19 | I/O/Z | | 1.8 V | |
| DDR_D30 | C20 | I/O/Z | | 1.8 V | |
| DDR_D31 | E19 | I/O/Z | | 1.8 V | |
| DDR_ODT0 | E13 | I/O/Z | | 1.8 V | On-die termination signals to external DDR2 SDRAM. These pins are reserved for future use and should not be connected to the DDR2 SDRAM. |
| DDR_ODT1 | A14 | I/O/Z | | 1.8 V | |
| $\overline{\text{DDR\_CAS}}$ | D10 | I/O/Z | | 1.8 V | DDR2 Memory Controller SDRAM column address strobe |
| DDR_CKE | B11 | I/O/Z | | 1.8 V | DDR2 Memory Controller SDRAM clock-enable |
| DDR_DQGATE0 | D7 | I/O/Z | | 1.8 V | DDR2 Memory Controller Data Strobe Gate |
| DDR_DQGATE1 | B6 | I/O/Z | | 1.8 V | |
| DDR_DQGATE2 | E18 | I/O/Z | | 1.8 V | |
| DDR_DQGATE3 | A21 | I/O/Z | | 1.8 V | |
| DDR_DQM[0] | B8 | I/O/Z | | 1.8 V | DDR2 Memory Controller Byte-enable Controls. 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). |
| DDR_DQM[1] | D5 | I/O/Z | | 1.8 V | |
| DDR_DQM[2] | E15 | I/O/Z | | 1.8 V | |
| DDR_DQM[3] | B21 | I/O/Z | | 1.8 V | |
| DDR_DQS[0] | A9 | I/O/Z | | 1.8 V | DDR2 Memory Controller Data Strobe [3:0] |
| DDR_DQS[1] | A7 | I/O/Z | | 1.8 V | |
| DDR_DQS[2] | A17 | I/O/Z | | 1.8 V | |
| DDR_DQS[3] | A20 | I/O/Z | | 1.8 V | |
| $\overline{\text{DDR\_DQS[0]}}$ | A8 | I/O/Z | | 1.8 V | DDR2 Memory Controller Data Strobe [3:0] Negative |
| $\overline{\text{DDR\_DQS[1]}}$ | A6 | I/O/Z | | 1.8 V | |
| $\overline{\text{DDR\_DQS[2]}}$ | A16 | I/O/Z | | 1.8 V | |
| $\overline{\text{DDR\_DQS[3]}}$ | A19 | I/O/Z | | 1.8 V | |
| $\overline{\text{DDR\_RAS}}$ | E10 | I/O/Z | | 1.8 V | DDR2 Memory Controller SDRAM Row Address Strobe |
| $\overline{\text{DDR\_WE}}$ | D11 | I/O/Z | | 1.8 V | DDR2 Memory Controller SDRAM Write Enable |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| **CONFIGURATION AND EMIFA** | | | | | |
| DEVICEENABLE0/ AEA20 | F2 | I/O/Z | IPD | 3.3 V | EMIFA External Address 20 (word address). (O/Z) For proper device operation, this pin must be externally pulled up with a 1-kΩ resistor at device reset |
| EMIFAWIDTH/ AEA22 | G3 | I/O/Z | IPD | 3.3 V | EMIFA External Address 22 (word address). (O/Z) EMIFA data bus width selection pin state captured at the rising edge of RESET.<br>0 - sets EMIFA CS2 to 8-bit data bus width<br>1 - sets EMIFA CS2 to 16 bit data bus width. For details, see Section 3. |
| FASTBOOT/ AEA21 | G2 | I/O/Z | IPD | 3.3 V | EMIFA External Address 22 (word address). (O/Z) Enables FAST BOOT of the device. For details, see Section 3. |
| UHPIEN | H2 | I | IPD | 3.3 V | UHPI Enable Pin. This pin controls the selection (enable/disable) of the HPI and GPIO[0:7] muxed with PCI. For details, see Section 3. |
| HPIWIDTH/ AEA16 | H3 | I/O/Z | IPD | 3.3 V | EMIFA External Address 16 (word address) (O/Z) HPI peripheral bus width (HPI_WIDTH) select (Applies only when HPI is enabled; UHPIEN pin = 1) |
| RSV_BOOT/ AEA15 | H6 | I/O/Z | IPU | 3.3 V | EMIFA External Address 15 (word address) (O/Z) For proper device operation, this pin must be externally pulled up with a 1-kΩ resistor at device reset. |
| PCI66/AEA18 | G5 | I/O/Z | IPD | 3.3 V | PCI Frequency Selection (PCI66). The PCI peripheral must be enabled (UHPIEN = 0) to use this function. PCI66_AEA18 selects the PCI operating frequency of 66 MHz or 33 MHz. PCI operating frequency is selected at reset via the pullup/pulldown resistor on the PCI66 pin AEA18:<br>0 - PCI operates at 33 MHz (default)<br>1 - PCI operates at 66 MHz. |
| BOOTMODE0/AEA11<br>BOOTMODE1/AEA12<br>BOOTMODE2/AEA13<br>BOOTMODE3/AEA14 | F3<br>F4<br>F5<br>G6 | I/O/Z | IPD | 3.3 V | The BOOTMODE[3:0] defines what boot code is executed on device reset. See Section 3.2.1 for more details. |
| **INTER-INTEGRATED CIRCUIT (I2C)** | | | | | |
| SCL | D22 | I/O/Z | | 3.3 V | I2C clock. When the I2C module is used, use an external pullup resistor. |
| SDA | C23 | I/O/Z | | 3.3 V | I2C data. When I2C is used, make certain there is an external pullup resistor. |
| **SGMII0/1 and MDIO(1)** | | | | | |
| SGMII0RXN | AA10 | I | | 1.2 V | Differential SGMII Port 0 RX input (negative) |
| SGMII0RXP | AA9 | I | | 1.2 V | Differential SGMII Port 0 RX input (positive) |
| SGMII0TXN | W11 | O | | 1.2 V | Differential SGMII Port 0 TX output (negative) |
| SGMII0TXP | Y11 | O | | 1.2 V | Differential SGMII Port 0 TX output (positive) |
| SGMII1RXN | AC9 | I | | 1.2 V | Differential SGMII Port 1 RX input (negative) |
| SGMII1RXP | AB9 | I | | 1.2 V | Differential SGMII Port 1 RX input (positive) |
| SGMII1TXN | W9 | O | | 1.2 V | Differential SGMII Port 1 TX output (negative) |
| SGMII1TXP | W8 | O | | 1.2 V | Differential SGMII Port 1 TX output (positive) |
| MDCLK | U9 | OZ | IPD | 3.3 V | MDIO Serial Clock (MDCLK) |
| MDIO | U8 | I/O/Z | IPU | 3.3 V | MDIO Serial Data (MDIO) |
| **SPI or UART** | | | | | |

(1) If the Ethernet Subsystem is not used at all, these connections must be followed:
- Disconnect AA10, AA9, W11,Y11, AC9, AB9, W9, W8, and U9
- Connect AC11 to $CV_{DD}$
- Connect AB11 to $V_{SS}$
- Directly connect V11 ($V_{DDA}$), W10 ($V_{DDA}$), T10 ($V_{DDD}$), U10 ($V_{DDD}$) , AB8 ($V_{DDT}$), U11 ($V_{DDT}$), R9 (ESS core power), R11 (ESS core power) to $CV_{DD}$
- Directly connect AB10, ($V_{DDR}$) to $DV_{DD18}$

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| SPICLK | F22 | I/O/Z | IPU | 3.3 V | SPI Clock Output |
| $\overline{\text{SPICS1}}$/UARTTX | D23 | I/O/Z | IPU | 3.3 V | SPI Chip Select 1 or UART Transmit (O/Z) |
| $\overline{\text{SPICS2}}$/UARTRX | F23 | I/O/Z | IPU | 3.3 V | SPI Chip Select 2 or UART Receive |
| SPIDI/UARTRTS | G23 | I/O/Z | IPU | 3.3 V | SPI Data input or UART Ready to send (O/Z) |
| SPIDO/UARTCTS | F21 | I/O/Z | IPU | 3.3 V | SPI Data Output or UART Clear to send |
| **TIMER 0/1 or GPIO[8:11]** | | | | | |
| T0INP12/GP08 | E20 | I/O/Z | IPD | 3.3 V | Timer 0 input pin for lower 32-bit counter (I) or GPIO 8 |
| T0OUT12/GP09 | D21 | I/O/Z | IPD | 3.3 V | Timer 0 output pin for lower 32-bit counter (O/Z) or GPIO 9 |
| T1INPL/GP10 | E21 | I/O/Z | IPD | 3.3 V | Timer 1 input pin for lower 32-bit counter (I) or GPIO 10 |
| T1OUT12/GP11 | C22 | I/O/Z | IPD | 3.3 V | Timer 1 output pin for lower 32-bit counter(O/Z) or GPIO 11 |
| **McASP OR TISIP0/1** | | | | | |
| AHCLKR/ TSIP1CLKA | AC4 | I/O/Z | IPD | 3.3 V | McASP Receive high-frequency master clock or TSIP1 serial data clock A |
| AHCLKX/ TSIP1CLKB | AC3 | I/O/Z | IPD | 3.3 V | McASP Transmit high-frequency master clock or TSIP1 serial data clock B |
| ACLKR/ TSIP0CLKA | AC6 | I/O/Z | IPD | 3.3 V | McASP Receive master clock or TSIP0 serial data clock A |
| ACLKX/ TSIP0CLKB | AC7 | I/O/Z | IPD | 3.3 V | McASP Transmit master clock or TSIP0 serial data clock B |
| AFSR/TSIP0FSA | W6 | I/O/Z | IPD | 3.3 V | McASP Receive Frame sync or left/right clock (LRCLK) or TSIP0 frame sync A |
| AFSX/TSIP0FSB | AA7 | I/O/Z | IPD | 3.3 V | McASP Transmit Frame sync or left/right clock (LRCLK) or TSIP0 frame sync B |
| AXR0/TSIP0TR0 | AB6 | I/O/Z | IPD | 3.3 V | McASP data pin 0 or TSIP0 input serial data receive 0 |
| AXR1/TSIP0TR1 | Y6 | | IPD | 3.3 V | McASP data pin 1 or TSIP0 input serial data receive 1 |
| AXR2/TSIP0TX0 | AA6 | | IPD | 3.3 V | McASP data pin 2 or TSIP0 output serial data transmit 0 |
| AXR3/TSIP0TX1 | AB4 | | IPD | 3.3 V | McASP data pin 3 or TSIP0 output serial data transmit 1 |
| AXR4/TSIP1TR0 | Y5 | | IPD | 3.3 V | McASP data pin 4 or TSIP1 input serial data receive 0 |
| AXR5/TSIP1TR1 | V7 | | IPD | 3.3 V | McASP data pin 5 or TSIP1 input serial data receive 1 |
| AXR6/TSIP1TX0 | AA4 | | IPD | 3.3 V | McASP data pin 6 or TSIP1 output serial data transmit 0 |
| AXR7/TSIP1TX1 | V6 | | IPD | 3.3 V | McASP data pin 7 or TSIP1 output serial data transmit 1 |
| AMUTEIN/ TSIP1FSA | AB3 | I/O/Z | IPD | 3.3 V | McASP Mute Input or TSIP1 frame sync A |
| AMUTE/ TSIP1FSB | U7 | I/O/Z | IPD | 3.3 V | McASP Mute Output (O/Z) or TSIP1 frame sync B |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| **GPIO[12:15]** | | | | | |
| NC | Y23 | | | | No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| NC | V23 | | | | |
| NC | Y22 | | | | |
| NC | V22 | | | | |
| NC | U23 | | | | |
| NC | W20 | | | | |
| NC | V18 | | | | |
| NC | U18 | | | | |
| NC | V19 | | | | |
| NC | W21 | | | | |
| NC | T18 | | | | |
| NC | U19 | | | | |
| NC | V20 | | | | |
| GP12 | V21 | I/O/Z | IPD | 3.3 V | |
| GP13 | T19 | I/O/Z | IPD | 3.3 V | |
| GP14 | T20 | I/O/Z | IPD | 3.3 V | |
| GP15 | T21 | I/O/Z | IPD | 3.3 V | |
| NC | U20 | | | | |
| NC | U21 | | | | |
| NC | U22 | | | | |
| NC | R18 | | | | |
| **GPIO[16:31]** | | | | | |
| NC | P23 | | | | No Connect. This terminal is not used and must be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| NC | N23 | | | | |
| NC | R23 | | | | |
| NC | P22 | | | | |
| NC | N22 | | | | |
| GP16 | R19 | I/O/Z | IPD | 3.3 V | GPIO 16 |
| GP17 | P19 | I/O/Z | IPD | 3.3 V | GPIO 17 |
| GP18 | P18 | I/O/Z | IPD | 3.3 V | GPIO 18 |
| GP19 | R22 | I/O/Z | IPD | 3.3 V | GPIO 19 |
| GP20 | R21 | I/O/Z | IPD | 3.3 V | GPIO 20 |
| GP21 | R20 | I/O/Z | IPD | 3.3 V | GPIO 21 |
| GP22 | N21 | I/O/Z | IPD | 3.3 V | GPIO 22 |
| GP23 | N20 | I/O/Z | IPD | 3.3 V | GPIO 23 |
| GP24 | N19 | I/O/Z | IPD | 3.3 V | GPIO 24 |
| GP25 | P21 | I/O/Z | IPD | 3.3 V | GPIO 25 |
| GP26 | P20 | I/O/Z | IPD | 3.3 V | GPIO 26 |
| GP27 | M20 | I/O/Z | IPD | 3.3 V | GPIO 27 |
| GP28 | M18 | I/O/Z | IPD | 3.3 V | GPIO 28 |
| GP29 | N18 | I/O/Z | IPD | 3.3 V | GPIO 29 |
| GP30 | M21 | I/O/Z | IPD | 3.3 V | GPIO 30 |
| GP31 | M19 | I/O/Z | IPD | 3.3 V | GPIO 31 |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| **VLYNQ** | | | | | |
| NC | AB1 | | | | No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| VCLK | W1 | I/O/Z | IPU | 3.3 V | VLYNQ Clock (I/O) |
| NC | AA1 | | | | No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| NC | AB2 | | | | |
| $\overline{\text{VSCRUN}}$ | Y1 | I/O/Z | IPU | 3.3 V | VLYNQ serial clock run request (I/O) |
| NC | W5 | | | | No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| NC | AA2 | | | | |
| NC | Y3 | | | | |
| NC | U6 | | | | |
| NC | Y2 | | | | |
| NC | W3 | | | | |
| NC | V5 | | | | |
| NC | W4 | | | | |
| VRXD0 | W2 | I/O/Z | IPD | 3.3 V | VLYNQ receive data pin [0] (I) |
| VRXD1 | V3 | I/O/Z | IPD | 3.3 V | VLYNQ receive data pin [1] (I) |
| VRXD2 | V4 | I/O/Z | IPD | 3.3 V | VLYNQ receive data pin [2] (I) |
| VRXD3 | U1 | I/O/Z | IPD | 3.3 V | VLYNQ receive data pin [3] (I) |
| VTXD0 | U3 | I/O/Z | IPD | 3.3 V | VLYNQ transmit data pin [0] (O) |
| VTXD1 | U2 | I/O/Z | IPD | 3.3 V | VLYNQ transmit data pin [1] (O) |
| VTXD2 | U5 | I/O/Z | IPD | 3.3 V | VLYNQ transmit data pin [2] (O) |
| VTXD3 | U4 | I/O/Z | IPD | 3.3 V | VLYNQ transmit data pin [3] (O) |
| **EMIFA** | | | | | |
| AECLKIN | T1 | I | IPD | 3.3 V | EMIFA external input clock (I) |
| AECLKOUT | P1 | I/O/Z | IPD | 3.3 V | EMIFA output clock (O/Z) |
| $\overline{\text{AAWE}}$/$\overline{\text{ASWE}}$ | T2 | I/O/Z | IPU | 3.3 V | Asynchronous memory write enable/Programmable synchronous interface write-enable |
| AR/$\overline{\text{W}}$ | R1 | I/O/Z | IPU | 3.3 V | Asynchronous memory read/write (O/Z) |
| $\overline{\text{AAOE}}$/$\overline{\text{ASOE}}$ | P2 | I/O/Z | IPU | 3.3 V | Asynchronous/Programmable synchronous memory output-enable (O/Z) |
| AED00 | T6 | I/O/Z | IPU | 3.3 V | EMIFA External Data 0 |
| AED01 | T5 | I/O/Z | IPU | 3.3 V | EMIFA External Data 1 |
| AED02 | T4 | I/O/Z | IPU | 3.3 V | EMIFA External Data 2 |
| AED03 | T3 | I/O/Z | IPU | 3.3 V | EMIFA External Data 3 |
| AED04 | R6 | I/O/Z | IPU | 3.3 V | EMIFA External Data 4 |
| AED05 | R5 | I/O/Z | IPU | 3.3 V | EMIFA External Data 5 |
| AED06 | R4 | I/O/Z | IPU | 3.3 V | EMIFA External Data 6 |
| AED07 | R3 | I/O/Z | IPU | 3.3 V | EMIFA External Data 7 |
| AED08 | R2 | I/O/Z | IPU | 3.3 V | EMIFA External Data 8 |
| AED09 | P6 | I/O/Z | IPU | 3.3 V | EMIFA External Data 9 |
| AED10 | P5 | I/O/Z | IPU | 3.3 V | EMIFA External Data 10 |
| AED11 | P4 | I/O/Z | IPU | 3.3 V | EMIFA External Data 11 |
| AED12 | P3 | I/O/Z | IPU | 3.3 V | EMIFA External Data 12 |
| AED13 | N4 | I/O/Z | IPU | 3.3 V | EMIFA External Data 13 |
| AED14 | N6 | I/O/Z | IPU | 3.3 V | EMIFA External Data 14 |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| AED15 | N5 | I/O/Z | IPU | 3.3 V | EMIFA External Data 15 |
| **VIDEO PORT 4 OR EMIFA** | | | | | |
| VP4CLK0/AARDY | L1 | I | IPU | 3.3 V | Video Port 4 Clock 0 (I) or Asynchronous memory ready input (I) |
| VP4CLK1NC | K1 | I/O/Z | IPD | 3.3 V | Video Port 4 Clock 1No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| VP4CTL0/ABA0 | J2 | I/O/Z | IPD | 3.3 V | Video Port 4 Control 0 or EMIFA bank address control (ABA[1:0]) (O/Z). Active-low bank selects for the 16-bit EMIFA. When interfacing to 16-bit asynchronous devices, ABA1 carries bit 1 of the byte address. For an 8-bit asynchronous interface, ABA[1:0] are used to carry bits 1 and 0 of the byte address. |
| VP4CTL1/ABA1 | J1 | I/O/Z | IPD | 3.3 V | Video Port 4 Control 1 or EMIFA bank address control (ABA[1:0]) (O/Z). Active-low bank selects for the 16-bit EMIFA. WHEN interfacing to 16-bit asynchronous devices, ABA1 carries bit 1 of the byte address. For an 8-bit asynchronous interface, ABA[1:0] are used to carry bits 1 and 0 of the byte address. |
| VP4CTL2/ $\overline{\text{ASADS}}$/ $\overline{\text{ASRE}}$ | K2 | I/O/Z | IPD | 3.3 V | Video Port 4 Control 2 or Programmable synchronous address strobe or read-enable. For programmable synchronous interface, the r_enable field in the ChipSelect x Configuration Register selects between $\overline{\text{ASADS}}$ and $\overline{\text{ASRE}}$:<br>– If r_enable = 0, then the $\overline{\text{ASADS/ASRE}}$ signal functions as the $\overline{\text{ASADS}}$ signal.<br>– If r_enable = 1, then the $\overline{\text{ASADS/ASRE}}$ signal functions as the $\overline{\text{ASRE}}$ signal. |
| VP4D02/ $\overline{\text{ABE00}}$ | L2 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 2 or EMIFA byte-enable control 0. 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. |
| VP4D03/$\overline{\text{ABE01}}$ | M4 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 3 or EMIFA byte-enable control 1. Number of address bits or byte enables used depends on the width of external memory. Byte-write enables for most types of memory. |
| VP4D04/AEA10 | M5 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 4 or EMIFA External Address 10 (word address) (O/Z) |
| VP4D05NC | M6 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 5No Connect. This terminal is not used and should be left floating. Do not bias, connect to power or ground, or use as a routing point. This terminal may be connected internally. |
| VP4D06/$\overline{\text{ACE2}}$ | L3 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 6 or EMIFA memory space enable 2 |
| VP4D07/$\overline{\text{ACE3}}$ | L4 | I/O/Z | IPU | 3.3 V | Video Port 4 Data 7 or EMIFA memory space enable 3 |
| VP4D08/AEA00 | L5 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 8 or EMIFA External Address 0 (word address) (O/Z) |
| VP4D09/AEA01 | K3 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 9 or EMIFA External Address 1 (word address) (O/Z) |
| VP4D12/AEA02 | K4 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 12 or EMIFA External Address 2 (word address) (O/Z) |
| VP4D13/AEA03 | L6 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 13 or EMIFA External Address 3 (word address) (O/Z) |
| VP4D14/AEA04 | K5 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 14 or EMIFA External Address 4 (word address) (O/Z) |
| VP4D15/AEA05 | J3 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 15 or EMIFA External Address 5 (word address) (O/Z) |
| VP4D16/AEA06 | J4 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 16 or EMIFA External Address 6 (word address) (O/Z) |
| VP4D17/AEA07 | J5 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 17 or EMIFA External Address 7 (word address) (O/Z) |
| VP4D18/AEA08 | J6 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 18 or EMIFA External Address 8 (word address) (O/Z) |

**Table 2-4. Terminal Functions (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| VP4D19/AEA09 | K6 | I/O/Z | IPD | 3.3 V | Video Port 4 Data 19 or EMIFA External Address 9 (word address) (O/Z) |
| **EMIFA** | | | | | |
| AEA23 | H4 | OZ | IPD | 3.3 V | EMIFA External Address 23 (word address) (O/Z) |
| AEA19 | H5 | O/Z | IPU | 3.3 V | EMIFA External Address 19 (word address) (O/Z) |
| AECLKINSEL/AEA17 | G4 | I/O/Z | IPD | 3.3 V | Select EMIFA external clock (I) (The EMIFA input clock AECLKIN or SYSCLK4 is selected at reset via the pullup/pulldown resistor on this pin. Note: AECLKIN is the default for the EMIFA input clock.)<br>or EMIFA external address 17 (word address) (O/Z) |

**Table 2-5. Terminal Functions (Ground and Power Supply)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| $V_{SS}$ | A1 | | | | Ground |
| $V_{SS}$ | A5 | | | | Ground |
| $V_{SS}$ | A15 | | | | Ground |
| $V_{SS}$ | A18 | | | | Ground |
| $V_{SS}$ | A23 | | | | Ground |
| $V_{SS}$ | C4 | | | | Ground |
| $V_{SS}$ | D9 | | | | Ground |
| $V_{SS}$ | D12 | | | | Ground |
| $V_{SS}$ | D20 | | | | Ground |
| $V_{SS}$ | E6 | | | | Ground |
| $V_{SS}$ | E23 | | | | Ground |
| $V_{SS}$ | F7 | | | | Ground |
| $V_{SS}$ | F15 | | | | Ground |
| $V_{SS}$ | F17 | | | | Ground |
| $V_{SS}$ | F19 | | | | Ground |
| $V_{SS}$ | G8 | | | | Ground |
| $V_{SS}$ | G10 | | | | Ground |
| $V_{SS}$ | G12 | | | | Ground |
| $V_{SS}$ | G14 | | | | Ground |
| $V_{SS}$ | G16 | | | | Ground |
| $V_{SS}$ | G18 | | | | Ground |
| $V_{SS}$ | G22 | | | | Ground |
| $V_{SS}$ | H1 | | | | Ground |
| $V_{SS}$ | H11 | | | | Ground |
| $V_{SS}$ | H13 | | | | Ground |
| $V_{SS}$ | H15 | | | | Ground |
| $V_{SS}$ | H17 | | | | Ground |
| $V_{SS}$ | J8 | | | | Ground |
| $V_{SS}$ | J10 | | | | Ground |
| $V_{SS}$ | J12 | | | | Ground |
| $V_{SS}$ | J14 | | | | Ground |
| $V_{SS}$ | J16 | | | | Ground |
| $V_{SS}$ | K7 | | | | Ground |
| $V_{SS}$ | K9 | | | | Ground |
| $V_{SS}$ | K11 | | | | Ground |
| $V_{SS}$ | K13 | | | | Ground |
| $V_{SS}$ | K15 | | | | Ground |
| $V_{SS}$ | K17 | | | | Ground |
| $V_{SS}$ | L8 | | | | Ground |
| $V_{SS}$ | L10 | | | | Ground |
| $V_{SS}$ | L12 | | | | Ground |
| $V_{SS}$ | L14 | | | | Ground |
| $V_{SS}$ | L16 | | | | Ground |
| $V_{SS}$ | M7 | | | | Ground |
| $V_{SS}$ | M9 | | | | Ground |

(1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal

**Table 2-5. Terminal Functions (Ground and Power Supply) (continued)**

| TERMINAL NAME | NO | TYPE[1] | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| $V_{SS}$ | M11 | | | | Ground |
| $V_{SS}$ | M13 | | | | Ground |
| $V_{SS}$ | M15 | | | | Ground |
| $V_{SS}$ | M17 | | | | Ground |
| $V_{SS}$ | M22 | | | | Ground |
| $V_{SS}$ | N1 | | | | Ground |
| $V_{SS}$ | N8 | | | | Ground |
| $V_{SS}$ | N10 | | | | Ground |
| $V_{SS}$ | N12 | | | | Ground |
| $V_{SS}$ | N14 | | | | Ground |
| $V_{SS}$ | N16 | | | | Ground |
| $V_{SS}$ | P7 | | | | Ground |
| $V_{SS}$ | P9 | | | | Ground |
| $V_{SS}$ | P11 | | | | Ground |
| $V_{SS}$ | P13 | | | | Ground |
| $V_{SS}$ | P15 | | | | Ground |
| $V_{SS}$ | P17 | | | | Ground |
| $V_{SS}$ | R10 | | | | Ground |
| $V_{SS}$ | R12 | | | | Ground |
| $V_{SS}$ | R14 | | | | Ground |
| $V_{SS}$ | R16 | | | | Ground |
| $V_{SS}$ | T7 | | | | Ground |
| $V_{SS}$ | T9 | | | | Ground |
| $V_{SS}$ | T11 | | | | Ground |
| $V_{SS}$ | T13 | | | | Ground |
| $V_{SS}$ | T15 | | | | Ground |
| $V_{SS}$ | T17 | | | | Ground |
| $V_{SS}$ | T22 | | | | Ground |
| $V_{SS}$ | U14 | | | | Ground |
| $V_{SS}$ | U16 | | | | Ground |
| $V_{SS}$ | V1 | | | | Ground |
| $V_{SS}$ | V9 | | | | Ground |
| $V_{SS}$ | V17 | | | | Ground |
| $V_{SS}$ | W7 | | | | Ground |
| $V_{SS}$ | W22 | | | | Ground |
| $V_{SS}$ | Y9 | | | | Ground |
| $V_{SS}$ | Y10 | | | | Ground |
| $V_{SS}$ | AA3 | | | | Ground |
| $V_{SS}$ | AA8 | | | | Ground |
| $V_{SS}$ | AA11 | | | | Ground |
| $V_{SS}$ | AB5 | | | | Ground |
| $V_{SS}$ | AB19 | | | | Ground |
| $V_{SS}$ | AB23 | | | | Ground |
| $V_{SS}$ | AC1 | | | | Ground |
| $V_{SS}$ | AC8 | | | | Ground |
| $V_{SS}$ | AC10 | | | | Ground |

**Table 2-5. Terminal Functions (Ground and Power Supply) (continued)**

| TERMINAL NAME | NO | TYPE[1] | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| $V_{SS}$ | AC12 | | | | Ground |
| $V_{SS}$ | AC16 | | | | Ground |
| **POWER PINS** | | | | | |
| $CV_{DD}$ | J9 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | J11 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | J15 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | K10 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | K12 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | K14 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | L9 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | L11 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | L13 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | L15 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | M10 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | M12 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | M14 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | N11 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | N13 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | N15 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | P10 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | P12 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | P14 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | R13 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | N9 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | T16 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | R8 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | R15 | | | | 1.2-V Core Power Supply |
| $CV_{DD}$ | V8 | | | | 1.2-V Core Power Supply |
| $CV_{DDESS}$ | R11 | | | | 1.2-V Core Power Supply for Ethernet Subsystem |
| $CV_{DDESS}$ | R9 | | | | 1.2-V Core Power Supply for Ethernet Subsystem |
| $AV_{DLL1}$ | B10 | | | | 1.8-V I/O supply |
| $AV_{DLL2}$ | B22 | | | | 1.8-V I/O supply |
| $CV_{DD1}$ | H9 | | | | 1.2-V Power supply for DDR, DDR I/Os, EMIF-DDR Subsystem |
| $CV_{DD1}$ | J13 | | | | 1.2-V Power supply for DDR, DDR I/Os, EMIF-DDR Subsystem |
| $AV_{DDA}$ | V11 | | | | 1.2-V SerDes Analog supply |
| $AV_{DDA}$ | W10 | | | | 1.2-V SerDes Analog supply |
| $DV_{DDD}$ | T10 | | | | 1.2-V SerDes Digital Supply |
| $DV_{DDD}$ | U10 | | | | 1.2-V SerDes Digital Supply |
| $AV_{DDR}$ | AB10 | | | | 1.8-V SerDes Analog Supply (Regulator) |
| $AV_{DDT}$ | AB8 | | | | 1.2-V SerDes Analog Supply |
| $AV_{DDT}$ | U11 | | | | 1.2-V SerDes Analog Supply |
| $DV_{DD33}$ | E22 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | F20 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | G1 | | | | 3.3-V I/O supply voltage |

**Table 2-5. Terminal Functions (Ground and Power Supply) (continued)**

| TERMINAL NAME | NO | TYPE[1] | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| $DV_{DD33}$ | G19 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | J7 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | H16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | H22 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | J17 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | K8 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | K16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | L7 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | L17 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | M8 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | M16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | M23 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | N2 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | N7 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | N17 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | P8 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | P16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | R7 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | R17 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | T8 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | T12 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | T14 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | T23 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AB7 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | U15 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | U17 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | V2 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | V16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | W23 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | Y4 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | Y8 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AB16 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AC2 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AC5 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AB12 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AC19 | | | | 3.3-V I/O supply voltage |
| $DV_{DD33}$ | AC23 | | | | 3.3-V I/O supply voltage |
| $DV_{DD18}$ | B1 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | B5 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | B15 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | B18 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | B23 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | C3 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | C10 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | C13 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | C21 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |

**Table 2-5. Terminal Functions (Ground and Power Supply) (continued)**

| TERMINAL NAME | NO | TYPE[1] | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| $DV_{DD18}$ | E5 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | F8 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | F14 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | F16 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | F18 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | G9 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | G11 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | G13 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | G15 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | G17 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | H10 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | H12 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| $DV_{DD18}$ | H14 | | | | 1.8-V I/O supply voltage (DDR2 Memory Controller) |
| DDR_VREF | C11 | | | | ($DV_{DD18}$/2)-V reference for SSTL buffer (DDR2 Memory Controller0. This input voltage cn be generated directly from $DV_{DD18}$ using two 1-KΩ resistors to form a resister divider circuit. |
| $V_{CCMON}$ | L19 | | | | Die-side 1.2-V core supply voltage monitor pin. The monitor pins indicate the voltage on the die, and, therefore, provide the best probe point for voltage monitoring purposes. If the $V_{CCMON}$ pin is not used, it should be connected directly to the 1.2-V core supply or left unconnected. |
| $V_{DD18MON}$ | B2 | | | | Die-side 1.8-V I/O supply voltage monitor pin. The monitor pins indicate the voltage on the die and, therefore, provide the best probe point for voltage monitoring purposes. If the $V_{DD18MON}$ pin is not used, it should be connected directly to the 1.8-V I/O supply ($DV_{DD18}$). |
| $V_{DD33MON}$ | G21 | | | | Die-side 3.3-V I/O supply voltage monitor pin. The monitor pins indicate the voltage on the die and, therefore, provide the best probe point for voltage monitoring purposes. If the $V_{DD33MON}$ pin is not used, it should be connected directly to the 3.3-V I/O supply ($DV_{DD33}$). |
| **Reserved** | | | | | |
| RSV 1 | L20 | A | | | Reserved. Unconnected |
| RSV 2 | L21 | A | | | Reserved. Unconnected |
| RSV 3 | D2 | O | | | Reserved. Unconnected |
| RSV 4 | D1 | O | | | Reserved. Unconnected |
| RSV 5 | D4 | O | | | Reserved. Unconnected |
| RSV 6 | D3 | O | | | Reserved. Unconnected |
| RSV 7 | H7 | A | | | Reserved. These pins must be connected directly to $V_{SS}$ for proper device operation. |
| RSV 8 | H8 | A | | | Reserved. These pins must be connected directly to $V_{SS}$ for proper device operation. |
| RSV 9 | M2 | A | | | Reserved. Unconnected |
| RSV 10 | A2 | A | | | Reserved. Unconnected |
| RSV 11 | E2 | | | | Reserved This pin must be connected directly to $V_{SS}$ for proper device operation. |
| RSV 12 | E1 | | | | Reserved. This pin must be connected directly to 1.8-V I/O supply |
| RSV 13 | E4 | | | | Reserved This pin must be connected directly to $V_{SS}$ for proper device operation. |
| RSV 14 | E3 | | | | Reserved. This pin must be connected directly to 1.8-V I/O supply |

**Table 2-5. Terminal Functions (Ground and Power Supply) (continued)**

| TERMINAL NAME | NO | TYPE(1) | INTERNAL PULLUP/ PULLDOWN | OPER VOLT | DESCRIPTION |
|---|---|---|---|---|---|
| RSV 15 | A10 | A | | | Reserved. Unconnected |
| RSV 16 | A22 | A | | | Reserved. Unconnected |
| RSV 17 | V10 | A | | | Reserved. Unconnected |
| RSV 18 | F6 | I | | | Reserved. These pins must be connected directly to 1.8-V I/O supply($DV_{DD18}$) for proper device operation. |
| RSV 19 | C2 | | | | Reserved. This pin must be connected to the 1.8-V I/O supply ($DV_{DD18}$) via a 200-Ω resistor for proper device operation. NOTE: If the DDR2 Memory Controller is not used, the DDR_VREF, RSV19, and RSV20 pins can be directly connected to ground ($V_{SS}$) to save power. However, connecting these pins directly to ground will prevent boundary scan from functioning on the DDR2 Memory Controller pins. To preserve boundary-scan functionality on the DDR2 Memory Controller pins, see Section 6.3.6. |
| RSV 20 | C1 | | | | Reserved. This pin must be connected to ground ($V_{SS}$) via a 200-Ω resistor for proper device operation. NOTE: If the DDR2 Memory Controller is not used, the RSV 19 and RSV 20 pins can be directly connected to ground ($V_{SS}$) to save power. However, connecting these pins directly to ground will prevent boundary scan from functioning on the DDR2 Memory Controller pins. To preserve boundary-scan functionality on the DDR2 Memory Controller pins, see Section 6.3.6. |
| RSV 21 | Y12 | | | | Reserved. This pin must be connected via a 20-Ω resistor directly to 3.3-V I/O Supply ($DV_{DD33}$) for proper device operation. The resistor used should have a minimal rating of 250 mW. |
| RSV 22 | W12 | | | | Reserved. This pin must be connected via a 40-Ω resistor directly to ground ($V_{SS}$) for proper device operation. The resistor used should have a minimal rating of 100 mW. |

## 2.7 Device Support

### *2.7.1 Development Support*

TI offers an extensive line of development tools for the TMS320DM64x DMPTMS320DM64x 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 tools support documentation is electronically available within the Code Composer Studio™ Integrated Development Environment (IDE).

The following products support development of TMS320DM64xx DMP-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 SoC application.

**Hardware Development Tools:**

Extended Development System (XDS™) Emulator (supports TMS320DM64x multiprocessor system debug) EVM (Evaluation Module)

For a complete listing of development-support tools for the TMS320DM64x platform, visit the Texas Instruments website at www.ti.com. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

### 2.7.2 Device and Development-Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all DSP devices and support tools. Each DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., TMS320C6452ZUT7). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

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

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

**TMS** Fully-qualified production device.

Support tool development evolutionary flow:

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

**TMDS** Fully qualified development-support product.

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

"Developmental product is intended for internal evaluation purposes."

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

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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, ZUT), the temperature range (for example, "Blank" is the commercial temperature range), and the device speed range in megahertz (for example, "Blank" is the default [720-MHz]).

Figure 2-6 provides a legend for reading the complete device name for the devices.

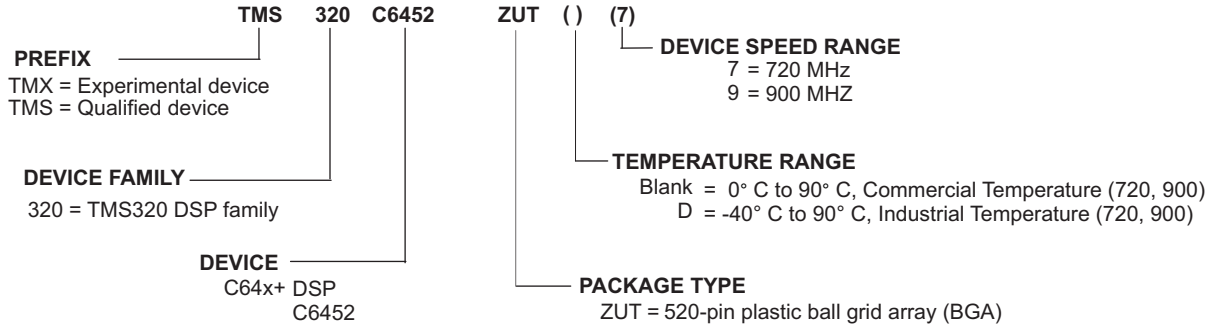



**Figure 2-6. Device Nomenclature**

### 2.7.3 Related Documentation From Texas Instruments

The following documents describe the devices. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

The following documents describe the TMS320C6452. Copies of these documents are available on the Internet at www.ti.com. Enter the literature number in the search box provided at www.ti.com.

**CPU**

**SPRU732** ***TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide*** describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C64x and TMS320C64x+ digital signal processors (DSPs) of the TMS320C6000 DSP family. The C64x/C64x+ DSP generation comprises fixed-point devices in the C6000 DSP platform. The C64x+ DSP is an enhancement of the C64x DSP with added functionality and an expanded instruction set.

**Reference Guides**

**SPRUF85** ***C6452 DSP DDR2 Memory Controller User's Guide*** describes the DDR2 memory controller in the TMS320C6452 Digital Signal Processor (DSP). The DDR2/mDDR memory controller is used to interface with JESD79D-2A standard compliant DDR2 SDRAM devices and standard Mobile DDR SDRAM devices.

**SPRUF86** ***C6452 Peripheral Component Interconnect (PCI) User's Guide*** describes the peripheral component interconnect (PCI) port in the TMS320C6452 Digital Signal Processor (DSP). The PCI port supports connection of the C642x DSP to a PCI host via the integrated PCI master/slave bus interface. The PCI port interfaces to the DSP via the enhanced DMA (EDMA) controller. This architecture allows for both PCI master and slave transactions, while keeping the EDMA channel resources available for other applications.

**SPRUF87** ***C6452 DSP Host Port Interface (UHPI) User's Guide*** describes the host port interface (HPI) in the TMS320C6452 Digital Signal Processor (DSP). The HPI is a parallel port through which a host processor can directly access the CPU memory space. The host device functions as a master to the interface, which increases ease of access. The host and CPU can exchange information via internal or external memory. The host also has direct access to memory-mapped peripherals. Connectivity to the CPU memory space is provided through the enhanced direct memory access (EDMA) controller.

**SPRUF89** ***C6452 DSP VLYNQ Port User's Guide*** describes the VLYNQ port in the TMS320C6452 Digital Signal Processor (DSP). The VLYNQ port is a high-speed point-to-point serial interface for connecting to host processors and other VLYNQ compatible devices. It is a full-duplex serial bus where transmit and receive operations occur separately and simultaneously without interference.

**SPRUF90** ***C6452 DSP 64-Bit Timer User's Guide*** describes the operation of the 64-bit timer in the C6452 Digital Signal Processor (DSP). The timer can be configured as a general-purpose 64-bit timer or dual general-purpose 32-bit timers.

**SPRUF91** ***C6452 DSP Multichannel Audio Serial Port (McASP) User's Guide*** describes the multichannel audio serial port (McASP) in the C6452 Digital Signal Processor (DSP). The McASP functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, Inter-Integrated Sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT).

**SPRUF92** ***C6452 DSP Serial Port Interface (SPI) User's Guide*** discusses the Serial Port Interface (SPI) in the C6452 Digital Signal Processor (DSP). This reference guide provides the specifications for a 16-bit configurable, synchronous serial peripheral interface. The SPI is a programmable-length shift register, used for high speed communication between external peripherals or other DSPs.

**SPRUF93** ***C6452 DSP Universal Asynchronous Receiver/Transmitter (UART) User's Guide*** describes the universal asynchronous receiver/transmitter (UART) peripheral in the C6452 Digital Signal Processor (DSP). The UART peripheral performs serial-to-parallel conversion on data received from a peripheral device, and parallel-to-serial conversion on data received from the CPU.

**SPRUF94** ***C6452 DSP Inter-Integrated Circuit (I2C) Module User's Guide*** describes the inter-integrated circuit (I2C) peripheral in the C6452 Digital Signal Processor (DSP). The I2C peripheral provides an interface between the DSP and other devices compliant with the I2C-bus specification and connected by way of an I2C-bus. External components attached to this 2-wire serial bus can transmit and receive up to 8-bit wide data to and from the DSP through the I2C peripheral. This document assumes the reader is familiar with the I2C-bus specification.

**SPRUF95** ***C6452 DSP General-Purpose Input/Output (GPIO) User's Guide*** describes the general-purpose input/output (GPIO) peripheral in the C6452 Digital Signal Processor (DSP). The GPIO peripheral provides dedicated general-purpose pins that can be configured as either inputs or outputs. When configured as an input, you can detect the state of the input by reading the state of an internal register. When configured as an output, you can write to an internal register to control the state driven on the output pin.

**SPRUF96** ***C6452 DSP Telecom Serial Interface Port (TSIP) User's Guide*** is a multi-link serial interface consisting of a maximum of two transmit data signals (or links), two receive data signals (or links), two frame sync input signals, and two serial clock inputs. Internally the TSIP offers single channel of timeslot data management and single DMA capability that allow individual timeslots to be selectively processed.

**SPRUF97** ***TMS320C6452 DSP 3 Port Switch (3PSW) Ethernet Subsystem User's Guide*** describes the operation of the 3 port switch (3PSW) ethernet subsystem in the TMS320C6452 Digital Signal Processor (DSP). The 3 port switch gigabit ethernet subsystem provides ethernet packet communication and can be configured as an ethernet switch. It provides the serial gigabit media independent interface (SGMII), the management data input output (MDIO) for physical layer device (PHY) management.

# 3 Device Configuration

## 3.1 System Module Registers

The system module includes status and control registers required for configuration of the device. Brief descriptions of the various registers are shown in Table 3-1. System Module registers required for device configuration are described in the following sections.

**Table 3-1. System Module Register Memory Map**

| HEX ADDRESS RANGE | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x0204 9000 | PINMUX | Pin multiplexing control 0 |
| 0x0204 9004 | Reserved | Reserved |
| 0x0204 9008 | DSPBOOTADDR | Boot Address of DSP, decoded by bootloader software for host boots |
| 0x0204 900C | BOOTCMPLT | Boot Complete |
| 0x0204 9010 | | Reserved |
| 0x0204 9014 | BOOTCFG | Device boot configuration |
| 0x0204 9018 | JTAGID | Device ID number. See Section 6.24 for details. |
| 0x0204 901C | PRI_ALLOC | Bus master priority control. See Section 4 for details |
| 0x0204 9020 -0x0204 9053 | Reserved | Reserved |
| 0x0204 9054 | KEY_REG | Key Register to protect against accidental writes. |
| 0x0204 9060 - 0x0204 90A7 | Reserved | Reserved |
| 0x0204 90A8 | CFGPLL | CFGPLL inputs for SerDes |
| 0x0204 90AC | Reserved | Reserved |
| 0x0204 90B0 | CFGRX0 | Configure SGMII0 RX (1) |
| 0x0204 90B4 | CFGRX1 | Configure SGMII1 RX (1) |
| 0x0204 90B8 | CFGTX0 | Configure SGMII0 TX.(1) |
| 0x0204 90BC | CFGTX1 | Configure SGMII1 TX (1) |
| 0x0204 90C0 | Reserved | Reserved |
| 0x0204 90C4 | MAC_ADDR_R0 | MAC Address Read Only Register 0 |
| 0x0204 90C8 | MAC_ADDR_R1 | MAC Address Read Only Register 1 |
| 0x0204 90CC | MAC_ADDR_RW0 | MAC Address Read/Write Register 0 |
| 0x0204 90D0 | MAC_ADDR_RW0 | MAC Address Read/Write Register 1 |
| 0x0204 90D4 | ESS_LOCK | Ethernet Sub System Lock Register |

(1) See the *TMS320C6452 DSP 3 Port Switch (3PSW) Ethernet Subsystem User's Guide* (SPRUF97) for details.

## 3.2 Bootmode Registers

The BOOTCFG and DSPBOOTADDR registers are described in the following sections. At reset, the status levels of various pins required for proper boot are stored within these registers.

### *3.2.1 Boot Configuration (BOOTCFG) Register*

Configuration pins latched at reset are presented in the BOOTCFG register accessible through the system module. This is a read-only register. The bits show the true latched value of the corresponding input at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion. This is desirable since the most important use of this MMR is for the user to debug/view the actual value driven on the pins during device reset.

**Figure 3-1. BOOTCFG Register**

| 31 | | | | | | | 24 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R-0 | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| AECLKINSEL | PC166 | HPIWIDTH | Reserved | FASTBOOT | Reserved | DUHPIEN | EMIFAWIDTH |
| | | R-L | | R-L | | R-L | R-L |
| 15 | | | | | | | 8 |
| Reserved | | | | | | | |
| R-0 | | | | | | | |
| 7 | | | 4 | 3 | | | 0 |
| Reserved | | | | BOOTMODE | | | |
| R-0 | | | | R-L | | | |

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

**Table 3-2. BOOTCFG Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:24 | Reserved | | Reserved |
| 23 | AECLKINSEL | | Controls the clock input for EMIFA. Latched from AECLKINSEL at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 1 | EMIFA clocked from internal SYSCLK |
| | | 0 | EMIFA clocked from outside from AECLKIN |
| 22 | PCI66 | | Controls PCI speed. PCI. Latched from PCI66 at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 0 | 33 MHz PCI |
| | | 1 | 66 MHz |
| 21 | HPIWIDTH | | Controls HPI bus width. Latched from HPIWIDTH at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 0 | 16 bit |
| | | 1 | 32 bit |
| 20 | Reserved | 1 | Reserved |
| 19 | FASTBOOT | | Fast Boot. Latched from FASTBOOT at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 0 | No Fast Boot |
| | | 1 | Fast Boot |
| 18 | Reserved | | Reserved |
| 17 | DUHPIEN | | PCI Enable Default. Latched from UHPIEN at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 0 | UHPI disabled |
| | | 1 | UHPI enabled |
| 16 | EMIFAWIDTH | | EMIFA CS2 Bus Width Default. Latched from EMIFAWIDTH at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |
| | | 0 | 8-bit |
| | | 1 | 16-bit |
| 15:4 | Reserved | | Reserved |

**Table 3-2. BOOTCFG Register Field Descriptions (continued)**

| Bit | Field | Value | Description |
|---|---|---|---|
| 3:0 | BOOTMODE | | Boot Mode. Latched from Bootmode at $\overline{\text{RESET}}$ or $\overline{\text{POR}}$ deassertion |

**Table 3-3. Boot Modes**

| DEVICE BOOT AND CONFIGURATION PINS | | | BOOT DESCRIPTION(1) | C6452 (MASTER/SLAVE) | DSPBOOTADDR (DEFAULT) |
|---|---|---|---|---|---|
| BOOTMODE[3:0] | UHPIEN | FASTBOOT | | | |
| 0000 | 0 or 1 | 0 or 1 | No boot (emulation boot) | - | 0x0080 0000 |
| 0001 | 0 | 1 | PCI boot no auto-initialization | Slave | 0x0080 0000 |
| | 1 | 0 or 1 | HPI boot | Slave | 0x0080 0000 |
| 0010 | 0 | 1 | PCI boot with auto-initialization | Slave | 0x0080 0000 |
| | 1 | 0 or 1 | HPI boot | Slave | 0x0080 0000 |
| 0011 | 0 or 1 | 0 | UART boot with no hardware flow control | Slave | 0x0080 0000 |
| 0100 | 0 or 1 | 0 | EMIFA ROM direct boot (PLL bypass mode) | Master | 0xA000 0000 |
| | | 1 | EMIFA ROM AIS boot | Master | 0x0080 0000 |
| 0101 | 0 or 1 | 0 or 1 | I2C Boot (standard mode) | Master | 0x0080 0000 |
| 0110 | 0 or 1 | 0 or 1 | SPI boot | Master | 0x0080 0000 |
| 0111 | 0 or 1 | 0 or 1 | Reserved | - | 0x0080 0000 |
| 1000 | 0 or 1 | 0 or 1 | SGMII0 - Boot port, no packet forwarding | Slave | 0x0080 0000 |
| 1001 | 0 or 1 | 0 or 1 | SGMII0 - Boot port, SGMII1 packet forwarding | Slave | 0x0080 0000 |
| 1010 | 0 or 1 | 0 or 1 | SGMII1 - Boot port, SGMII0 packet forwarding | Slave | 0x0080 0000 |
| 1011 | 0 or 1 | 0 or 1 | Reserved | - | 0x0080 0000 |
| 1100 | 0 or 1 | 0 or 1 | Reserved | - | 0x0080 0000 |
| 1101 | 0 or 1 | 0 or 1 | Reserved | - | 0x0080 0000 |
| 1110 | 0 or 1 | 0 | UART Boot with hardware flow control [UART0] | Slave | 0x0080 0000 |
| 1111 | 0 or 1 | 0 or 1 | Reserved | - | 0x0080 0000 |

(1) In all bootmodes other than EMIFA ROM Direct Boot (BOOTMODE[3:0] = 0100b, UHPIEN = 0b or 1b, FASTBOOT = 0b) all C64x+ cache is disabled (L1P,L1D,L2).

### 3.2.2 DSPBOOTADDR Register Description

The DSPBOOTADDR register contains the upper 22 bits of the C64x+ DSP reset vector. The register format is shown in Figure 3-2 and bit field descriptions are shown in Table 3-4. DSPBOOTADDR is readable and writable by software after reset. DSPBOOTADDR Decode: This decode logic determines the default of the DSPBOOTADDR Register. It can default to the base address of L2 ROM (0x00800000) or the base address of EMIFA CS2 (0xA0000000).

**Figure 3-2. DSPBOOTADDR Register**

| 31 | 10 | 9 | 0 |
|---|---|---|---|
| BOOTADDR | | Reserved | |
| R/W-0100 0010 0010 0000 0000 00 | | R-0 | |

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

**Table 3-4. DSPBOOTADDR Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:10 | BOOTADDR | | Upper 22 bits of the C64x+ DSP bootmode address |
| 9:0 | Reserved | | Reserved |

### 3.2.3 Boot Complete (BOOTCMPLT) register

The BOOTCMPLT register contains a BC (boot complete) field in bit 0, and a ERR (boot error) field in bits 19:16.

The BC field is written by the external host to indicates that it has completed boot. In the bootloader code, the CPU can poll for this bit. Once this bit = 1, the CPU can begin executing from DSPBOOTADDR.

The ERR field is written by the bootloader software if the software detects a boot error. Coming out of a boot, application software can read this field to determine if boot was accomplished. Actual error code is determined by software.

**Figure 3-3. BOOTCMPLT Register 3**

| 31 | 20 | 19 | 16 |
|---|---|---|---|
| Reserved | | ERR | |
| R-0 | | R-0 | |

| 15 | 1 | 0 |
|---|---|---|
| Reserved | | BC |
| R-0 | | R-0 |

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

**Table 3-5. BOOTCMPLT Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:20 | Reserved | | Reserved |
| 19:16 | ERR | | Boot error |
| | | 0000 | No error |
| | | 0001 – 1111 | Bootloader software detected boot error. |
| 15:1 | Reserved | | Reserved |
| 0 | BC | | Boot Complete Flag from host. This is applicable only to host boots. |
| | | 0 | Host has not completed booting this device. |
| | | 1 | Host has completed booting this device and the DSP can begin executing from DSPBOOTADDR. |

### 3.2.4 Priority Allocation (PRI_ALLOC)

Each of the masters (excluding the C64x+ Megamodule) is assigned a priority via the Priority Allocation Register (PRI_ALLOC), see Figure 3-4. The priority is enforced when several masters in the system are vying for the same endpoint. A value of 000b has the highest priority, while 111b has the lowest priority.

Note that the configuration SCR port on the data SCR is considered a single endpoint meaning priority will be enforced when multiple masters try to access the configuration SCR. Priority is also enforced on the configuration SCR side when a master (through the data SCR) tries to access the same endpoint as the C64x+ Megamodule.

The Ethernet Subsystem and VLYNQ fields specify the priority of the Ethernet Subsystem and VLYNQ peripherals, respectively. Similarly, the HOST field applies to the priority of the HPI and PCI peripherals. Other master peripherals are not present in the PRI_ALLOC register as they have their own registers to program their priorities. For more information on the default priority values in these peripheral registers, see the device-compatible peripheral reference guides.

TI recommends that these priority registers be reprogrammed during device initialization.

**Table 3-6. Default Master Priorities**

| MASTER | DEFAULT PRIORITY |
|---|---|
| EDMA3TC0 | 0 (EDMA CC QUEPRI Register) |
| EDMA3TC1 | 0 (EDMA CC QUEPRI Register) |
| EDMA3TC2 | 0 (EDMA CC QUEPRI Register) |
| EDMA3TC3 | 0 (EDMA CC QUEPRI Register) |
| 64x+_DMAP | 7 (C64x+ MDMAARBE.PRI Register bit field) |
| 64x+_CFGP | 1 (C64x+ MDMAARBE.PRI Register bit field) |
| Ethernet Subsystem | 3 (PRI_ALLOC register) |
| VLYNQ | 4 (PRI_ALLOC register) |
| UHPI | 4 (PRI_ALLOC register) |
| PCI | 4 (PRI_ALLOC register) |

**Figure 3-4. Priority Allocation Register (PRI_ALLOC)**

| 31 | 16 |
|---|---|
| Reserved | |
| R-0000000000001000 | |

| 15 | 12 | 11 | 9 | 8 | 6 | 5 | 3 | 2 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Reserved | | Reserved | | VLYNQ | | HOST | | Ethernet Subsystem | |
| R-0000 | | R/W-101 | | R/W-100 | | R/W-100 | | R/W-011 | |

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

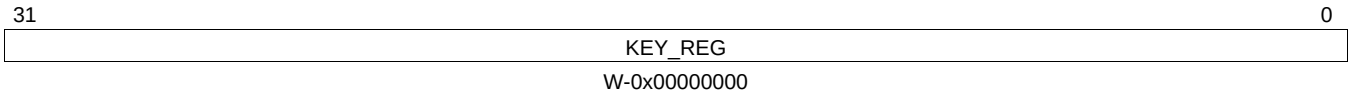
### 3.2.5 KEY_REG (write protection)

KEY_REG protects against accidental writes to certain system configuration registers. The complete set of registers protected by the KEY_REG is:

- PINMUX
- BOOTCFG
- PRI_ALLOC
- CFGPLL
- CFGRX0
- CFGTX0
- CFGRX1
- CFGTX1
- MAC_ADDR_RW0
- MAC_ADDR_RW1

Writes to these registers are locked/blocked by default. To enable writes to these registers, write 0xADDDECAF to the KEY_REG. After enabling writes to protected registers by doing the above, the register writes should occur within 10000 CPU/6 cycles, after which the key will be reset.

**Figure 3-5. KEY_REG**




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

### 3.2.6 PINMUX Register

All pin multiplexing options are controlled by software via PINMUX register (except the ones mentioned in , whose default is selected by configuration pins). This PINMUX register reside within the system module portion of the CFG bus memory map. The format of the registers and a description of the pins they control are in the following sections.

The PINMUX Register controls all the software-controlled pin muxing. The register format is shown in Figure 3-6. A brief description of each field is shown in Table 3-7.

**Figure 3-6. PINMUX Register**

| 31 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|
| Reserved | | GPIO_EN1 | | Reserved | | GPIO_EN2 | |
| R-0000 0000 00 | | R/W-00 | | R-00 | | R/W-00 | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EMIFA_EN | | SPI_UART_EN | | Reserved | | MCASP_EN | | Reserved | | VLYNQ_EN | | Reserved | | TIMER_EN |
| R/W-00 | | R/W-00 | | R-00 | | R/W-00 | | R-00 | | R/W-00 | | R-000 | | R/W-0 |

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

**Table 3-7. PINMUX Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:22 | Reserved | | Reserved |
| 21:20 | GPIO_EN1 | | Controls the pin state of GPIO[12:15] |
| | | | GPIO[12:15] PIN STATE(1) |
| | | 00 | 3-state |
| | | 01 | 3-state |
| | | 10 | Reserved |
| | | 11 | GP12-15 |
| 19:18 | Reserved | | Reserved |
| 17:16 | GPIO_EN2 | | Controls the pin state of GPIO[16:31] |
| | | | GPIO[16:31] PIN STATE |
| | | 00 | 3-state |
| | | 01 | 3-state |
| | | 10 | GP16-31 |
| | | 11 | Reserved |
| 15:14 | EMIFA_EN | | EMIFA (2) |
| | | | EMIFA PIN STATE(3) |
| | | 00 | 3-state |
| | | 01 | 3-state |
| | | 10 | EMIFA |
| | | 11 | Reserved |

(1) The complete list of pins: V21, T19, T20, T21.
(2) The value of EMIFA_EN depends on the BOOTMODE[3:0] pin value at reset. If the BOOTMODE[3:0] is 0100, the mux defaults to EMIFA enable (the value is 10b).
(3) The complete list of pins: T1, P1, T2, R1, P2, T6, T5, T4, T3, R6, R5, R4, R3, R2, P6, P5, P4, P3, N4, N6, N5, L1, J2, J1, K2, L2, M4, M5, L3, L4, L5, K3, K4, L6, K5, J3, J4, J5, J6, K6.

**Table 3-7. PINMUX Register Field Descriptions (continued)**

| Bit | Field | Value | Description | | |
|---|---|---|---|---|---|
| 13:12 | SPI_UART_EN | | Controls the pin muxing between SPI and UART | | |
| | | | | UNMUXED (4) | MUXED (5) |
| | | | | SPICLK | SPI or UART |
| | | 00 | | 3-state | 3-state |
| | | 01 | | Enable | SPI |
| | | 10 | | Disable | UART |
| | | 11 | | Enable | SPIDI<br>SPIDO<br>UART_TX<br>UART_RX |
| 11:10 | Reserved | | Reserved | | |
| 9:8 | MCASP_EN | | McASP | | |
| | | | | MUXED (6) | |
| | | | | McASP or TSIP0/1 | |
| | | 00 | | 3-state | |
| | | 01 | | McASP (all McASP Pins) | |
| | | 10 | | Reserved | |
| | | 11 | | TSIP0 and TSIP1 | |
| 7:6 | Reserved | | Reserved | | |
| 5:4 | VLYNQ_EN | | Controls the pin state of VLYNQ | | |
| | | | | VLYNQ | |
| | | 00 | | 3-state | |
| | | 01 | | 3-state | |
| | | 10 | | Reserved | |
| | | 11 | | VRXD0-3 and VTXD0-3, VCLK, VSCRUN | |
| 3:1 | Reserved | | Reserved | | |
| 0 | TIMER_EN | | Controls the pin muxing between TIMER and GPIO[8:11] | | |
| | | | | | MUXED(7) |
| | | 0 | | | GPIO[8:11] |
| | | 1 | | | Timer 0/1 |

(4) The complete list of pin:F22
(5) The complete list of pins: D23, F23, G23, F21
(6) The complete list of pins: AC4, AC3, AC6, AC7, W6, AA7, AB6, Y6, AA6, AB4, Y5, V7, AA4, V6, Y7, AA5, AB3, U7
(7) The complete list of pins:E20, D21, E21, C22

### 3.2.7 MAC Address Registers

- MAC_ADDR_R0
- MAC_ADDR_R1
- MAC_ADDR_RW0
- MAC_ADDR_RW1

Two sets of registers provide default MAC addresses for the device. One set - MAC_ADDR_R0 and MAC_ADDR_R1 - is read only and the other set - MAC_ADDR_RW0 and MAC_ADDR_RW1 - includes read and write registers.

**Figure 3-7. MAC_ADDR_R0 Register**

| 31 | 0 |
|---|---|
| MAC_ID | |
| R-MAC ADDRESS[31:0] | |

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

**Table 3-8. MAC_ADDR_R0 Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:0 | MAC_ID | Mac Address[31:0] of the device | Bit 0 of MAC_ID is bit 0 of MAC Address |

**Figure 3-8. MAC_ADDR_R1 Register**

| 31 | 24 | 23 | 16 |
|---|---|---|---|
| CRC | | Reserved | |
| R-CRC for the MAC_ID | | R-00000000 | |

| 15 | 0 |
|---|---|
| MAC_ID | |
| R-MAC ADDRESS[47:32] | |

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

**Table 3-9. MAC_ADDR_R1 Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:24 | CRC | CRC of the MAC ID | This field will hold the CRC of the MAC address of that particular device. |
| 23:16 | Reserved | 0x00 | Reserved |
| 15:0 | MAC_ID | Mac Address[47:32] of the device | Bit 0 of MAC_ID is Bit 32 of MAC Address |

**Figure 3-9. MAC_ADDR_RW0 Register**

| 31 | 0 |
|---|---|
| MAC_ADDR_R0 | |
| R/W - MAC ID[31:0] | |

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

**Table 3-10. MAC_ADDR_RW0 Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:0 | MAC_ID | Mac Address[31:0] of the device | Bit 0 of MAC_ID is bit 0 of MAC Address |

**Figure 3-10. MAC_ADDR_RW1 Register**

| 31 | 24 | 23 | 16 |
|---|---|---|---|
| CRC | | Reserved | |
| R/W-CRC for the MAC_ID | | R/W-00000000 | |

| 15 | 0 |
|---|---|
| MAC_ID | |
| R-MAC ADDRESS[47:32] | |

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

**Table 3-11. MAC_ADDR_RW1 Register Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:24 | CRC | CRC of the MAC ID | This field will hold the CRC of the MAC address of that particular device. |
| 23:16 | Reserved | 0x00 | Reserved |
| 15:0 | MAC_ID | Mac Address[47:32] of the device | Bit 0 of MAC_ID is Bit 32 of MAC Address |

## 3.3 Pullup/Pulldown Resistors

Proper board design should specify that input pins to the device always be at a valid logic level and not floating. This may be achieved via pullup/pulldown resistors. The device features internal pullup (IPU) and internal pulldown (IPD) resistors on most pins to eliminate the need, unless otherwise noted, for *external* pullup/pulldown resistors.

An external pullup/pulldown resistor must be used in the following situations:

- *Boot and Configuration Pins*: If the pin is both routed out and in high-impedance mode, an external pullup/pulldown resistor *must* be used, even if the IPU/IPD matches the desired value/state.
- *Other Input Pins*: If the IPU/IPD *does not* match the desired value/state, use an external pullup/pulldown resistor to pull the signal to the opposite rail.

If the boot and configuration pins are both routed out and in high-impedance mode, it is recommended that an external pullup/pulldown resistor be used. Although internal pullup/pulldown resistors exist on these pins and they may match the desired configuration value, providing external connectivity can help specify that valid logic levels are latched on these important boot configuration pins. In addition, applying external pullup/pulldown resistors on the boot and configuration pins adds convenience to the user in debugging and flexibility in switching operating modes.

Tips for choosing an external pullup/pulldown resistor:

- Select a resistor with the largest possible resistance
- Calculate the worst-case leakage current that flows through this external resistor. Worst-case leakage current can be calculated by adding up all the leakage current at the pin—e.g., the input current ($I_I$) from the device, and leakage current from the other device(s) to which this pin is connected.
- Specify that the voltage at the pin stays well within the low-/high-level input voltages ($V_{IL}$ or $V_{IH}$) when worst-case leakage current is flowing through this external resistor.
  - To oppose an IPU and pull the signal to a logic low, the voltage at the pin must stay well below $V_{IL}$.
  - To oppose an IPD and pull the signal to a logic high, the voltage at the pin must stay well above $V_{IH}$.

For most systems, a 1-kΩ resistor can be used to oppose the IPU/IPD while meeting the above criteria. Users should confirm this resistor value is correct for their specific application.

For most systems, a 20-kΩ resistor can be used to complement the IPU/IPD on the boot and configuration pins while meeting the above criteria. Users should confirm this resistor value is correct for their specific application.

For more detailed information on input current ($I_I$), and the low-/high-level input voltages ($V_{IL}$ and $V_{IH}$) , see Section 5.3, *Electrical Characteristics Over Recommended Ranges of Supply Voltage and Operating Temperature*.

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

# 4 System Interconnect

The C64x+ Megamodule, the EDMA3 transfer controllers, and the system peripherals are interconnected through two switch fabrics. The switch fabrics allow for low-latency, concurrent data transfers between master peripherals and slave peripherals. The switch fabrics allow for seamless arbitration between the system masters when accessing system slaves.

## 4.1 Internal Buses, Bridges, and Switch Fabrics

Two types of buses exist in the device: data buses and configuration buses. Some device peripherals have both a data bus and a configuration bus interface, while others only have one type of interface. Furthermore, the bus interface width and speed varies from peripheral to peripheral. Configuration buses are mainly used to access the register space of a peripheral and the data buses are used mainly for data transfers. However, in some cases, the configuration bus is also used to transfer data. For example, data is transferred to the UART or I2C via their configuration bus. Similarly, the data bus can also be used to access the register space of a peripheral. For example, the EMIFA and DDR2 memory controller registers are accessed through their data bus interface.

The C64x+ Megamodule, the EDMA3 traffic controllers, and the various system peripherals can be divided into two categories: masters and slaves. Masters are capable of initiating read and write transfers in the system and do not rely on the EDMA3 for their data transfers. Slaves, on the other hand, rely on the EDMA3 to perform transfers to and from them. Masters include the EDMA3 traffic controllers and PCI. Slaves include the McASP, and I2C.

The device contains two switch fabrics through which masters and slaves communicate. The data switch fabric, known as the data switched central resource (SCR), is a high-throughput interconnect mainly used to move data across the system (for more information, see Section 4.2). The data SCR connects masters to slaves via 128-bit data buses running at a SYSCLK1 frequency (SYSCLK1 is generated from PLL1 controller). Peripherals that have a 128-bit data bus interface running at this speed can connect directly to the data SCR; other peripherals require a bridge.

The configuration switch fabric, also known as the configuration switch central resource (SCR) is mainly used by the C64x+ Megamodule to access peripheral registers (for more information, see Section 4.3). The configuration SCR connects the C64x+ Megamodule to slaves via 32-bit configuration buses running at a SYSCLK1 frequency (SYSCLK1 is generated from the PLL1 controller). As with the data SCR, some peripherals require the use of a bridge to interface to the configuration SCR. Note that the data SCR also connects to the configuration SCR. Bridges perform a variety of functions:

- Conversion between configuration bus and data bus.
- Width conversion between peripheral bus width and SCR bus width
- Frequency conversion between peripheral bus frequency and SCR bus frequency

For example, the EMIFA memory controller require a bridge to convert their 64-bit data bus interface into a 128-bit interface so that they can connect to the data SCR.

Some peripherals can be accessed through the data SCR and also through the configuration SCR.

## 4.2 Data Switch Fabric Connections

Figure 4-1 shows the connection between slaves and masters through the data switched central resource (SCR). Masters are shown on the right and slaves on the left. The data SCR connects masters to slaves via 128-bit data buses running at a SYSCLK1 frequency. SYSCLK1 is supplied by the PLL1 controller and is fixed at a frequency equal to the CPU frequency divided by 3. Some peripherals, like the C64x+ Megamodule, have both slave and master ports. Each EDMA3 transfer controller has an independent connection to the data SCR.

Masters can access the configuration SCR through the data SCR. The configuration SCR is described in Section 4.3.

Not all masters on the device may connect to all slaves. Allowed connections are summarized in Table 4-1.

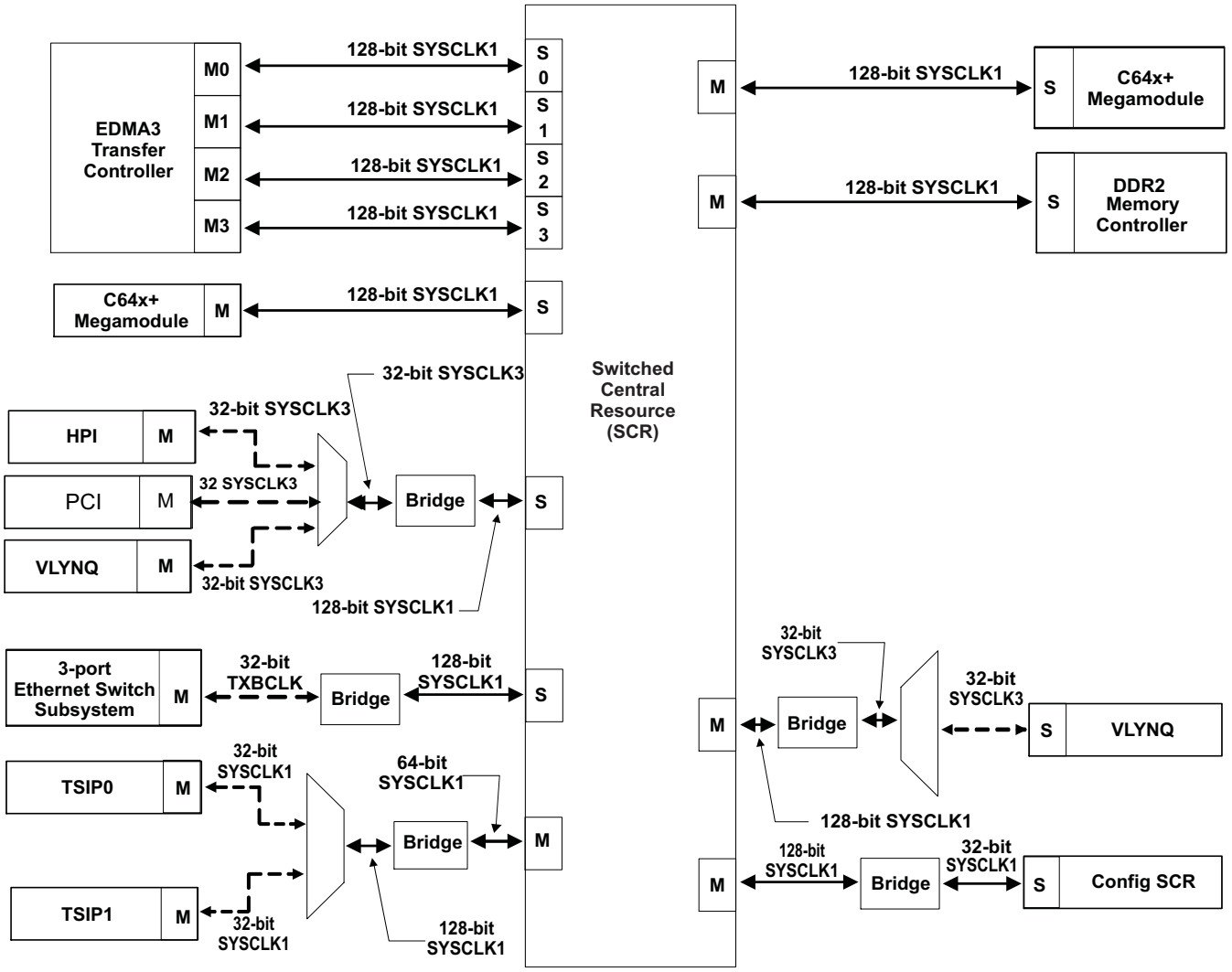



**Figure 4-1. Data SCR**

**Table 4-1. Connectivity Matrix for Data SCR**

| | MEGAMODULE | DDR2 EMIF | EMIFA | PCI | VLYNQ | Configuration SCR |
|---|---|---|---|---|---|---|
| TC0 | Y | Y | Y | Y | Y | Y |
| TC1 | Y | Y | Y | Y | Y | Y |
| TC2 | Y | Y | Y | Y | Y | Y |
| TC3 | Y | Y | Y | N | N | N |
| Megamodule | N | Y | Y | Y | Y | N |
| HPI | Y | Y | Y | Y | Y | Y |
| PCI | Y | Y | Y | Y | Y | Y |
| VLYNQ | Y | Y | Y | Y | Y | Y |
| Ethernet Subsystem | Y | Y | Y | N | N | N |

## 4.3 Configuration Switch Fabric

Figure 4-2 shows the connection between the C64x+ megamodule and the configuration SCR, which is mainly used by the C64x+ Megamodule to access peripheral registers. The data SCR also has a connection to the configuration SCR that allows masters to access most peripheral registers. The only registers not accessible by the data SCR through the configuration SCR are the device configuration registers and the PLL1 and PLL2 controller registers; these can be accessed only by the C64x+ Megamodule. The configuration SCR uses 32-bit configuration buses running at SYSCLK1 frequency. SYSCLK1 is supplied by the PLL1 controller and is fixed at a frequency equal to the CPU frequency divided by 3.

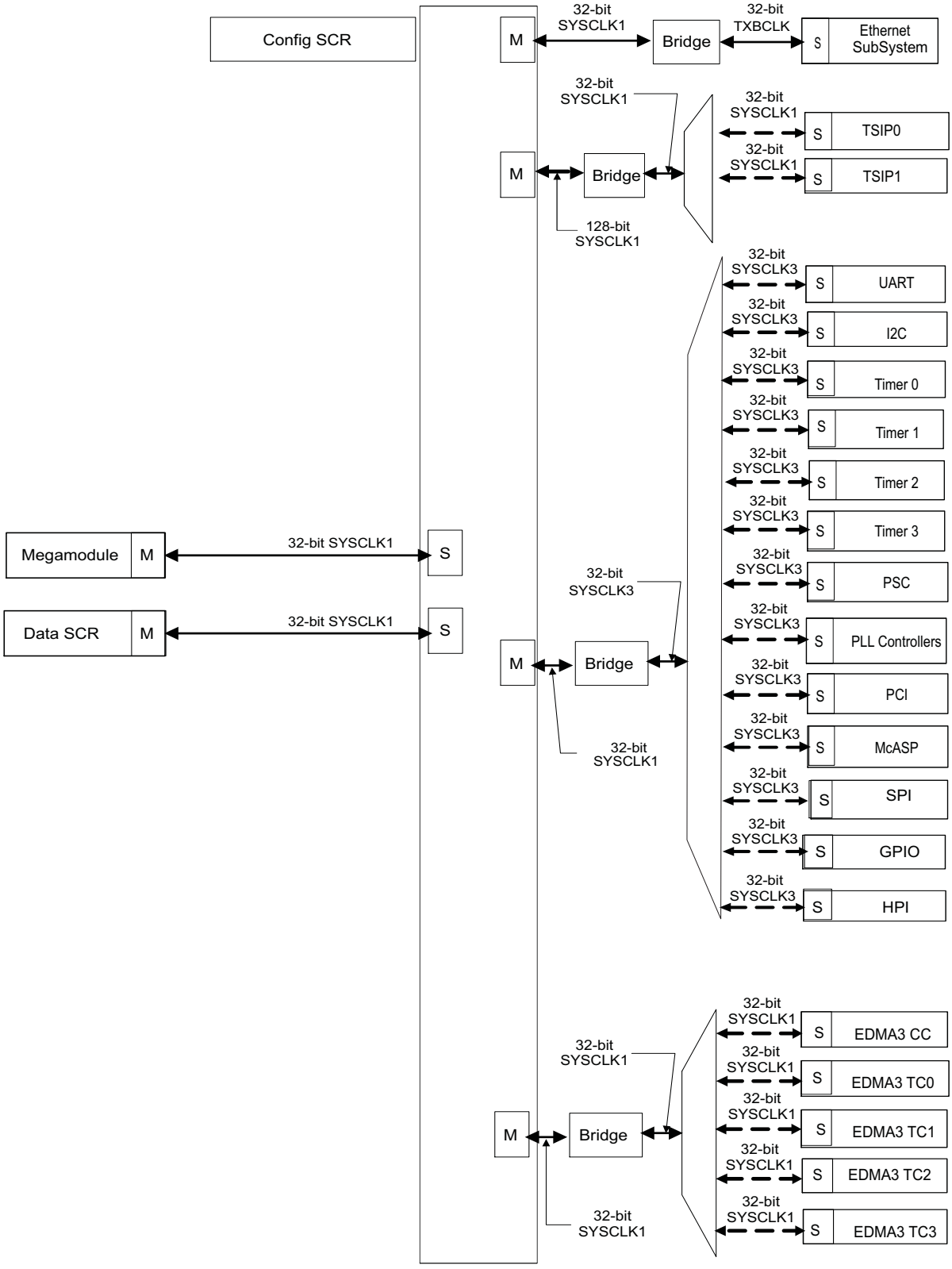



**Figure 4-2. Configuration SCR**

# 5 Device Operating Conditions

## 5.1 Absolute Maximum Ratings

Over Operating Temperature Range (Unless Otherwise Noted)(1)

| | | | |
|---|---|---|---|
| Supply voltage ranges: | Core ($CV_{DD}$, $CV_{DDESS}$, $CV_{DD1}$, $AV_{DDA}$, $DV_{DDD}$, $AV_{DDT}$)(2) | 1.20-V operation | -0.5 V to 1.5 V |
| | I/O, 3.3V ($DV_{DD33}$)(2) | | -0.5 V to 4.2 V |
| | I/O, 1.8V ($DV_{DD18}$, $AV_{DLL1}$, $AV_{DLL2}$, $AV_{DDR}$)(2) | | -0.5 to 2.5 V |
| Input voltage ranges: | $V_I$ I/O, 3.3-V pins | | -0.5 V to 4.2 V |
| | $V_I$ I/O, 1.8 V | | -0.5 V to 2.5 V |
| Output voltage ranges: | $V_O$ I/O, 3.3-V pins | | -0.5 V to 4.2 V |
| | $V_O$ I/O, 1.8 V | | -0.5 V to 2.5 V |
| Operating case temperature, $T_{case}$ | Commercial | | 0°C to 90°C |
| | Industrial | | -40°C to 90°C |
| Storage temperature range, $T_{stg}$ | (default) | | -65°C 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 $V_{SS}$.

## 5.2 Recommended Operating Conditions

| | | | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|---|
| $CV_{DD}$ | Supply voltage, Core (1) | (-720, -900 devices) | 1.14 | 1.2 | 1.26 | V |
| $CV_{DDESS}$ | Supply voltage, Ethernet Subsystem Core (1) | | | | | |
| $CV_{DD1}$ | Supply voltage, DDR Core (1) | | | | | |
| $AV_{DDA}$ | Supply voltage, SerDes Analog (1) | | | | | |
| $DV_{DDD}$ | Supply voltage, SerDes Digital (1) | | | | | |
| $AV_{DDT}$ | Supply voltage, SerDes Analog (1) | | | | | |
| $DV_{DD33}$ | Supply voltage, I/O, 3.3 V | | 3.14 | 3.3 | 3.46 | V |
| $DV_{DD18}$ | Supply voltage, DDR I/O, 1.8 V | | 1.71 | 1.8 | 1.89 | V |
| $AV_{DLL1}$ | Supply voltage, I/O, 1.8 V | | | | | |
| $AV_{DLL2}$ | Supply voltage, I/O, 1.8 V | | | | | |
| $AV_{DDR}$ | Supply voltage, 1.8-V SerDes Analog Supply (Regulator) | | | | | |
| $V_{SS}$ | Supply ground ($V_{SS}$) | | 0 | 0 | 0 | V |
| DDR_VREF | DDR2 reference voltage(2) | | $0.49DV_{DD18}$ | $0.5DV_{DD18}$ | $0.51DV_{DD18}$ | V |
| $V_{IH}$ | High-level input voltage, 3.3 V (except PCI-capable and I2C pins) | | 2 | | | V |
| | High-level input voltage, I2C | | $0.7DV_{DD33}$ | | | V |
| | PCI-capable pins | | $0.5DV_{DD33}$ | | $DV_{DD33} + 0.5$ | V |
| | DDR2 memory controller pins (DC) | | DDR_VREF + 0.125 | | $DV_{DD18} + 0.3$ | V |
| $V_{IL}$ | Low-level input voltage, 3.3 V(except PCI-capable and I2C pins) | | | | 0.8 | V |
| | Low-level input voltage, I2C | | 0 | | $0.3DV_{DD33}$ | V |
| | PCI-capable pins | | -0.5 | | $0.3DV_{DD33}$ | V |
| | DDR2 memory controller pins (DC) | | -0.3 | | DDR_VREF - 0.125 | V |
| $T_{case}$ | Operating case temperature | Commercial | 0 | | 90 | °C |
| | | Industrial | -40 | | 90 | |
| $F_{SYSCLK1}$ | DSP Operating Frequency (SYSCLK1) | (-900 devices) | 33.3 | | 900 | MHz |
| | | (-720 devices) | 33.3 | | 720 | MHz |

(1) Future variants of TI SOC devices may operate at voltages ranging from 0.9 V to 1.4 V to provide a range of system power/performance options. TI highly recommends that users design-in a supply that can handle multiple voltages within this range (i.e., 1.0 V, 1.05 V, 1.1 V, 1.14 V, 1.2, 1.26 V with ± 3% tolerances) by implementing simple board changes such as reference resistor values or input pin configuration modifications. Not incorporating a flexible supply may limit the system ability to easily adapt to future versions of TI SOC devices.

(2) DDR_VREF is expected to equal $0.5DV_{DDR2}$ of the transmitting device and to track variations in the $DV_{DD18}$.

## 5.3 Electrical Characteristics

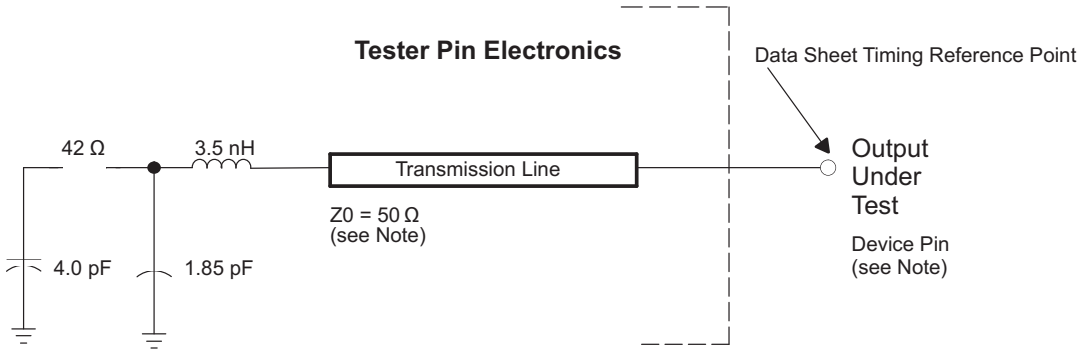
Over Recommended Ranges of Supply Voltage and Operating Temperature (Unless Otherwise Noted)

| PARAMETER | | TEST CONDITIONS (1) | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{OH}$ High-level output voltage | 3.3-V pins (except PCI-capable and I2C pins) | $DV_{DD33}$ = MIN, $I_{OH}$ = MAX | $0.8DV_{DD33}$ | | | V |
| | PCI-capable pins(2) | $DV_{DD33}$ = 3.3V, $I_{OH}$ = -0.5 mA | $0.9DV_{DD33}$ | | | |
| | DDR2 memory controller pins | | 1.4 | | | |
| $V_{OL}$ Low-level output voltage | 3.3-V pins (except PCI-capable and I2C pins) | $DV_{DD33}$ = MIN, $I_{OL}$ = MAX | | | $0.22DV_{DD33}$ | |
| | PCI-capable pins(2) | $DV_{DD33}$ = 3.3V, $I_{OL}$ = 1.5 mA | | | $0.1DV_{DD33}$ | |
| | I2C pins | Pulled up to 3.3 V, 3 mA sink current | | | 0.4 | |
| | DDR2 memory controller pins | | | | 0.4 | V |
| $I_I$ Input current [dc] | 3.3-V pins (except PCI-capable and I2C pins) | $V_I$ = $V_{SS}$ to $DV_{DD33}$ without internal pullup oe pulldown resistor | -1 | | 1 | µA |
| | | $V_I$ = $V_{SS}$ to $DV_{DD33}$ with internal pullup resistor (2) | 50 | 100 | 400 | µA |
| | | $V_I$ = $V_{SS}$ to $DV_{DD33}$ with opposing internal pulldown resistor (2) | -400 | -100 | -50 | µA |
| | Input current [dc] (I2C) | $0.1DV_{DD33} \le V_I \le 0.9DV_{DD33}$ | -10 | | 10 | µA |
| | PCI-capable pins(4) | | -600 | | 600 | µA |
| $I_{OH}$ High-level output current [dc] | All peripherals other than DDR2 and PCI | | | | -8 | mA |
| | DDR2 memory controller pins | | | | -4 | mA |
| | PCI-capable pins(2) | | | | -0.5 | mA |
| $I_{OL}$ Low-level output current [dc] | All peripherals other than DDR2 , PCI and I2C | | | | 8 | mA |
| | I2C pins | | | | 3 | mA |
| | DDR2 memory controller pins | | | | 4 | mA |
| | PCI-capable pins(2) | | | | 1.5 | mA |
| $I_{OZ}$ I/O Off-state output current [DC] | 3.3-V pins | | -20 | | 20 | µA |
| $I_{CDD}$ | Core ($CV_{DD}$, $V_{DDA\_1P1V}$) supply current(3) | $CV_{DD}$ = 1.2-V, DSP clock = 720 MHz | | 2353 | | mA |
| | | $CV_{DD}$ = 1.2-V, DSP clock = 900 MHz | | 2564 | | mA |
| $I_{DDD}$ | 3.3-V I/O ($DV_{DD33}$) supply current(3) | $DV_{DD}$ = 3.3-V, DSP clock = 720 MHz | | 249 | | mA |
| | | $DV_{DD}$ = 3.3-V, DSP clock = 900 MHz | | 249 | | mA |
| | 1.8-V I/O ($DV_{DDR2}$, DDR_VDDDLL, $PLLV_{PRW18}$, $V_{DDA\_1P8V}$, $MXV_{DD}$) supply current(3) | $DV_{DD}$ = 1.8-V, DSP clock = 720 MHz | | 309 | | mA |
| | | $DV_{DD}$ = 1.8-V, DSP clock = 900 MHz | | 309 | | mA |
| $C_I$ | Input capacitance | | | | 10 | pF |
| $C_o$ | Output capacitance | | | | 10 | pF |

(1) For test conditions shown as MIN, MAX, or NOM, use the appropriate value specified in the recommended operating conditions table.
(2) Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.
(3) Assumes the following conditions: 50% DSP CPU utilization; (peripheral configurations, other housekeeping activities) DDR2 at 50% utilization (266 MHz ), 50% writes, 32 bits, 100% bit switching, 110-MHz Video Ports at 100% utilization, MCASP operating at 25 MHz with 100% utilization with 10 serializers, Timer0,1 at 100% utilization, VICP with 100% utilization, PCI operating at 66 MHz with 50% writes at room temp (25°C) using ZUT package. (as in the power appnote) for the three items.

# 6 Peripheral Information and Electrical Specifications

## 6.1 Parameter Information




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

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

**Figure 6-1. Test Load Circuit for AC Timing Measurements**

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

### *6.1.1 3.3-V Signal Transition Levels*

All input and output timing parameters are referenced to $V_{ref}$ for both 0 and 1 logic levels. For 3.3-V I/O, $V_{ref}$ = 1.5 V. For 1.8-V I/O, $V_{ref}$ = 0.9 V.

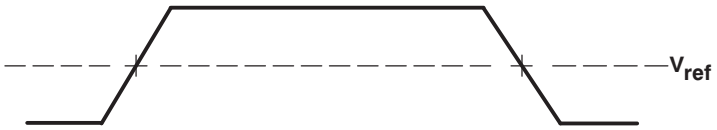



**Figure 6-2. Input and Output Voltage Reference Levels for ac Timing Measurements**

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

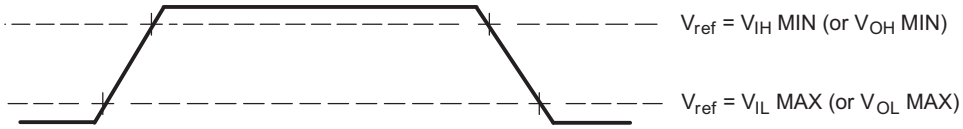



**Figure 6-3. Rise and Fall Transition Time Voltage Reference Levels**

### *6.1.2 3.3-V Signal Transition Rates*

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

### 6.1.3 Timing Parameters and Board Routing Analysis

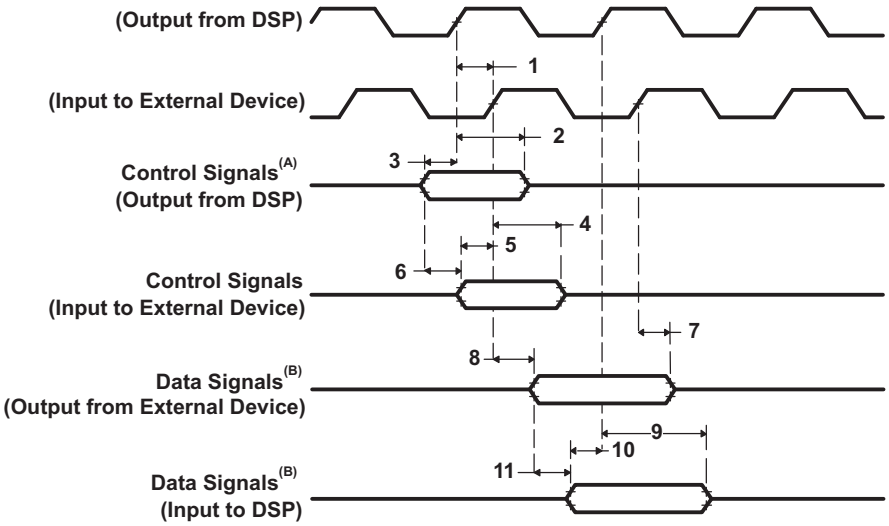
The timing parameter values specified in this data manual do *not* include delays caused by board routings. As a good board design practice, such delays must *always* be taken into account. Timing values may be adjusted by increasing/decreasing such delays. TI recommends utilizing the available I/O buffer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

For inputs, timing is most impacted by the round-trip propagation delay from the DSP to the external device and from the external device to the DSP. This round-trip delay tends to negatively impact the input setup time margin, but also tends to improve the input hold time margins (see Table 6-1 and Figure 6-4).

Figure 6-4 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 6-1. Board-Level Timing Example (see Figure 6-4)**

| 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 |




A. Control signals include data for writes.

B. Data signals are generated during reads from an external device.

**Figure 6-4. Board-Level Input/Output Timings**

## 6.2 Recommended Clock and Control Signal Transition Behavior

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

## 6.3 Power Supplies

For more information regarding TI's power management products and suggested devices to power TI DSPs, visit www.ti.com/dsppower.

### *6.3.1 Power-Supply Sequencing*

The device includes 1.2-V core supply ($CV_{DD}$, $CV_{DDESS}$, $CV_{DD1}$, $AV_{DDA}$, $DV_{DDD}$, $AV_{DDT}$), and two I/O supplies—3.3-V ($DV_{DD33}$) and 1.8-V ($DV_{DD18}$, $AV_{DLL1}$, $AV_{DLL2}$, $AV_{DDR}$). To ensure proper device operation, a specific power-up sequence must be followed. Some TI power-supply devices include features that facilitate power sequencing — for example, Auto-Track and Slow-Start/Enable features. For more information on TI power supplies and their features, visit www.ti.com/dsppower.

Following is a summary of the power sequencing requirements:

- The power ramp order must be 3.3-V ($DV_{DD33}$) before 1.8-V ($DV_{DD18}$, $AV_{DLL1}$, $AV_{DLL2}$, $AV_{DDR}$), and 1.8-V ($DV_{DD18}$, $AV_{DLL1}$, $AV_{DLL2}$, $AV_{DDR}$) before 1.2-V core supply ($CV_{DD}$, $CV_{DDESS}$, $CV_{DD1}$, $AV_{DDA}$, $DV_{DDD}$, $AV_{DDT}$) —meaning during power up, the voltage at the 1.8-V rail should never exceed the voltage at the 3.3-V rail. Similarly, the voltage at the 1.2-V rail should never exceed the voltage at the $DV_{DDR2}$ rail.
- From the time that power ramp begins, all power supplies (3.3 V, 1.8 V, 1.2 V) must be stable within 200 ms. The term "stable" means reaching the recommended operating condition (see Section 5.2).

### *6.3.2 Power-Supply Design Considerations*

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

### *6.3.3 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. 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. Larger caps for each supply can be placed further away for bulk decoupling. Large bulk caps (on the order of 100 µF) should be furthest away, but still as close as possible. Large caps for each supply should be placed outside of the BGA footprint.

### *6.3.4 Power and Sleep Controller (PSC)*

The power and sleep controller (PSC) controls power by turning off unused power domains or by gating off clocks to individual peripherals/modules. The device uses the clock-gating feature of the PSC only for power savings. The PSC consists of a global PSC (GPSC) and a set of local PSCs (LPSCs).

The GPSC contains memory mapped registers, PSC interrupt control, and a state machine for each peripheral/module. An LPSC is associated with each peripheral/module and provides clock and reset control. The LPSCs are shown in Table 6-2. The PSC register memory map is given in Table 6-3. For more details on the PSC, see the *TMS320C6452 DSP Subsystem Reference Guide* (literature number SPRUFB1).

**Table 6-2. LPSC Assignments**

| LPSC NUMBER | PERIPHERAL/ MODULE |
|---|---|
| 0 | EDMA3CC |
| 1 | Reserved |
| 2 | Reserved |
| 3 | Reserved |
| 4 | Reserved |
| 5 | TSIP0 |
| 6 | TSIP1 |
| 7 | DDR2 Memory Controller |
| 8 | UHPI |
| 9 | VLYNQ |
| 10 | GPIO |
| 11 | TIMER0 |
| 12 | TIMER1 |
| 13 | Reserved |
| 14 | Reserved |
| 15 | Reserved |
| 16 | Reserved |
| 17 | SPI |
| 18 | I2C |
| 19 | PCI |
| 20 | Reserved |
| 21 | Reserved |
| 22 | Reserved |
| 23 | Reserved |
| 24 | Reserved |
| 25 | EMIFA |
| 26 | TIMER2 |
| 27 | TIMER3 |
| 28 | Reserved |
| 29 | McASP |
| 30 | UART |
| 31 | Reserved |
| 32 | Reserved |
| 33 | C64x+ CPU |
| 34 | Ethernet Subsystem |

**Table 6-3. PSC Register Memory Map**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 6000 | PID | Peripheral Revision and Class Information Register |
| 0x0204 6004- 0x0204 600F | – | Reserved |
| 0x0204 6010 | – | Reserved |
| 0x0204 6014 | | Reserved |
| 0x0204 6018 | INTEVAL | Interrupt Evaluation Register |
| 0x0204 601C- 0x0204 603F | – | Reserved |
| 0x0204 6040 | – | Reserved |
| 0x0204 6044 | MERRPR1 | Module Error Pending 1 (mod 32- 63) Register |
| 0x0204 6048- 0x0204 604F | – | Reserved |

**Table 6-3. PSC Register Memory Map (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 6050 | – | Reserved |
| 0x0204 6054 | MERRCR1 | Module Error Clear 1 (mod 32 - 63) Register |
| 0x0204 6058- 0x0204 605F | – | Reserved |
| 0x0204 6060 | – | Reserved |
| 0x0204 6064- 0x0204 6067 | – | Reserved |
| 0x0204 6068 | – | Reserved |
| 0x0204 606C- 0x0204 611F | – | Reserved |
| 0x0204 6120 | PTCMD | Power Domain Transition Command Register |
| 0x0204 6124- 0x0204 6127 | – | Reserved |
| 0x0204 6128 | PTSTAT | Power Domain Transition Status Register |
| 0x0204 612C- 0x0204 61FF | – | Reserved |
| 0x0204 6200 | PDSTAT0 | Power Domain Status 0 Register (Always On) |
| 0x0204 6204- 0x0204 62FF | – | Reserved |
| 0x0204 6300 | PDCTL0 | Power Domain Control 0 Register (Always On) |
| 0x0204 6304- 0x1C4 150F | – | Reserved |
| 0x0204 6510 | – | Reserved |
| 0x0204 6514 | – | Reserved |
| 0x0204 6518- 0x0204 65FF | – | Reserved |
| 0x0204 6600- 0x0204 67FF | – | Reserved |
| 0x0204 6800 | MDSTAT0 | Module Status 0 Register (EDMACC) |
| 0x0204 6804 | – | Reserved |
| 0x0204 6808 | – | Reserved |
| 0x0204 680C | – | Reserved |
| 0x0204 6810 | – | Reserved |
| 0x0204 6814 | – | Reserved |
| 0x0204 6818 | – | Reserved |
| 0x0204 681C | MDSTAT7 | Module Status 7 Register (DDR2) |
| 0x0204 6820 | MDSTAT8 | Module Status 8 Register (HPI) |
| 0x0204 6824 | MDSTAT9 | Module Status 9 Register (VLYNQ) |
| 0x0204 6828 | MDSTAT10 | Module Status 10 Register (GPIO) |
| 0x0204 682C | MDSTAT11 | Module Status 11 Register (TIMER 0) |
| 0x0204 6830 | MDSTAT12 | Module Status 12 Register (TIMER 1) |
| 0x0204 6834 | – | Reserved |
| 0x0204 6838 | – | Reserved |
| 0x0204 683C | – | Reserved |
| 0x0204 6840 | – | Reserved |
| 0x0204 6844 | MDSTAT17 | Module Status 17 Register (SPI) |
| 0x0204 6848 | MDSTAT18 | Module Status 18 Register (I2C) |
| 0x0204 684C | MDSTAT19 | Module Status 19 Register (PCI) |
| 0x0204 6850 | MDSTAT20 | Reserved |
| 0x0204 6854 | MDSTAT21 | Reserved |
| 0x0204 6858 | MDSTAT22 | Reserved |
| 0x0204 685C | MDSTAT23 | Reserved |
| 0x0204 6860 | MDSTAT24 | Reserved |
| 0x0204 6864 | MDSTAT25 | Module Status 25 Register (EMIFA) |
| 0x0204 6868 | MDSTAT26 | Module Status 26 Register (TIMER 2) |
| 0x0204 686C | MDSTAT27 | Module Status 27 Register (TIMER 3) |

**Table 6-3. PSC Register Memory Map (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 6870 | MDSTAT28 | Reserved |
| 0x0204 6874 | MDSTAT29 | Module Status 29 Register (McASP) |
| 0x0204 6878 | MDSTAT30 | Module Status 30 Register (UART) |
| 0x0204 687C | MDSTAT31 | Reserved |
| 0x0204 6880 | – | Reserved |
| 0x0204 6884 | MDSTAT33 | Module Status 33 Register (C64x+ CPU) |
| 0x0204 6888 | MDSTAT34 | Module Status 34 Register (Ethernet Subsystem) |
| 0x0204 688C-0x0204 69FF | – | Reserved |
| 0x0204 6A00 | MDCTL0 | Module Control 0 Register (EDMACC) |
| 0x0204 6A04 | – | Reserved |
| 0x0204 6A08 | – | Reserved |
| 0x0204 6A0C | – | Reserved |
| 0x0204 6A10 | – | Reserved |
| 0x0204 6A14 | – | Reserved |
| 0x0204 6A18 | – | Reserved |
| 0x0204 6A1C | MDCTL7 | Module Control 7 Register (DDR2) |
| 0x0204 6A20 | MDCTL8 | Module Control 8 Register (HPI) |
| 0x0204 6A24 | MDCTL9 | Module Control 9 Register (VLYNQ) |
| 0x0204 6A28 | MDCTL10 | Module Control 10 Register (GPIO) |
| 0x0204 6A2C | MDCTL11 | Module Control 11 Register (TIMER 0) |
| 0x0204 6A30 | MDCTL12 | Module Control 12 Register (TIMER 1) |
| 0x0204 6A34 | – | Reserved |
| 0x0204 6A38 | – | Reserved |
| 0x0204 6A3C | – | Reserved |
| 0x0204 6A40 | – | Reserved |
| 0x0204 6A44 | MDCTL17 | Module Control 17 Register (SPI) |
| 0x0204 6A48 | MDCTL18 | Module Control 18 Register (I2C) |
| 0x0204 6A4C | MDCTL19 | Module Control 19 Register (PCI) |
| 0x0204 6A50 | MDCTL20 | Reserved |
| 0x0204 6A54 | MDCTL21 | Reserved |
| 0x0204 6A58 | MDCTL22 | Reserved |
| 0x0204 6A5C | MDCTL23 | Reserved |
| 0x0204 6A60 | MDCTL24 | Reserved |
| 0x0204 6A64 | MDCTL25 | Module Control 25 Register (EMIFA) |
| 0x0204 6A68 | MDCTL26 | Module Control 26 Register (TIMER 2) |
| 0x0204 6A6C | MDCTL27 | Module Control 27 Register (TIMER 3) |
| 0x0204 6A70 | MDCTL28 | Reserved |
| 0x0204 6A74 | MDCTL29 | Module Control 29 Register (McASP) |
| 0x0204 6A78 | MDCTL30 | Module Control 30 Register (UART) |
| 0x0204 6A7C | MDCTL31 | Reserved |
| 0x0204 6A80 | – | Reserved |
| 0x0204 6A84 | MDCTL33 | Module Control 33 Register (C64x+ CPU) |
| 0x0204 6A88 | MDCTL34 | Module Control 34 Register (Ethernet Subsystem) |
| 0x0204 6A90- 0x0204 6FFF | – | Reserved |

### 6.3.5 Power and Clock Domains

The device includes two power domains: the System Domain and the Ethernet Subsystem Domain. Both of these power domains are always on when the chip is on. Both of these domains are powered by the $CV_{DD}$ pins of the device.

The primary PLL controller generates the input clock to the C64x+ megamodule as well as most of the system peripherals such as the multichannel audio serial ports (McASPs) and the external memory interface (EMIFA). The secondary PLL controller generates interface clocks for the DDR2 memory controller. The Ethernet Subsystem is clocked through the SerDes module, which takes input from REFCLKP/N. The primary PLL controller (PLL1 controller) uses the device input clock CLKIN1 and the secondary PLL controller (PLL2 controller) uses the device input clock CLKIN2.

Table 6-4 provides a listing of the clock domains.

**Table 6-4. Power and Clock Domains**

| POWER DOMAIN | CLOCK DOMAIN | PERIPHERAL/MODULE/USAGE |
|---|---|---|
| System Domain | CLKDIV1 | C64x+ CPU |
| System Domain | CLKDIV3 | EDMA/SCR |
| System Domain | CLKDIV3 | TSIP0 |
| System Domain | CLKDIV3 | TSIP1 |
| System Domain | CLKDIV3 | DDR Subsystem |
| System Domain | CLKDIV3 | EMIFA |
| System Domain | CLKDIV6 | HPI |
| System Domain | CLKDIV6 | PCI |
| System Domain | CLKDIV6 | VLYNQ |
| System Domain | CLKDIV6 | UART |
| System Domain | CLKDIV6 | I2C |
| System Domain | CLKDIV6 | TIMER 0 |
| System Domain | CLKDIV6 | TIMER 1 |
| System Domain | CLKDIV6 | TIMER 2 |
| System Domain | CLKDIV6 | TIMER 3 |
| System Domain | CLKDIV6 | SPI |
| System Domain | CLKDIV6 | McASP |
| System Domain | CLKDIV6 | GPIO |
| System Domain | CLKDIV6 | PLL Controller 1 |
| System Domain | CLKDIV6 | PLL Controller 2 |
| System Domain | CLKDIV6 | Config SCR |
| System Domain | CLKDIV4 0 | Internal EMIFA Clock |
| System Domain | CLKDIV4 1 | Emulation and Trace |
| Ethernet Subsystem Domain | SerDes TXBCLK | Ethernet Subsystem |

The device architecture is divided into the power and clock domains shown in Table 6-5, which further shows the clock domains and their ratios.

**Table 6-5. Clock Domain Assignment**

| SUBSYSTEM | CLOCK DOMAIN | DOMAIN CLOCK SOURCE | FIXED RATIO vs SYSREFCLK FREQUENCY |
|---|---|---|---|
| DSP Subsystem | CLKDIV1 | PLLC1.REFSYSCLK | - |
| Peripherals (CLKDIV3 Domain) | CLKDIV3 | PLLC1.SYSCLK1 | 1:3 |
| Emulation/Trace | CLKDIV4 1 | PLLC1.SYSCLK2 | 1:4 |
| Peripherals (CLKDIV6 Domain) | CLKDIV6 | PLLC1.SYSCLK3 | 1:6 |

**Table 6-5. Clock Domain Assignment (continued)**

| SUBSYSTEM | CLOCK DOMAIN | DOMAIN CLOCK SOURCE | FIXED RATIO vs SYSREFCLK FREQUENCY |
|---|---|---|---|
| Internal EMIFA Clock | CLKDIV4 0 | PLLC1.SYSCLK4 | 1:4 (1) |
| SYSCLK5 pin | CLKDIV4 2 | PLLC1.SYSCLK5 | 1:4 |

(1) There is a /2 divider in the path of PLLC1.SYSCLK4 so the effective EMIFA clock is PLLC1.SYSCLK4/2. By default the internal EMIFA Clock is 1:8.

### 6.3.6 Preserving Boundary-Scan Functionality on DDR2 Memory Pins

Similarly, when the DDR2 Memory Controller is not used, the DDR_VREF, RSV19, and RSV20 pins can be connected directly to ground ($V_{SS}$) to save power. However, this will prevent boundary-scan from functioning on the DDR2 Memory Controller pins. To preserve boundary-scan functionality on the DDR2 Memory Controller pins, DDR_VREF, RSV19, and RSV20 should be connected as follows:

- DDR_VREF - connect to a voltage of $DV_{DD18}/2$. The $DV_{DD18}/2$ voltage can be generated directly from the $DV_{DD18}$ supply using two 1-kΩ resistors to form a resistor divider circuit.
- RSV19 - connect this pin to the 1.8-V I/O supply ($DV_{DD18}$) via a 200-Ω resistor
- RSV20 - connect this pin to ground ($V_{SS}$) via a 200-Ω resistor.

## 6.4 PLL1 Controller

The primary PLL controller generates the input clock to the C64x+ megamodule (including the CPU) as well as most of the system peripherals such as the multichannel audio serial ports (McASPs) and the external memory interface (EMIFA). Figure 6-5 shows a functional block diagram of the PLL Input Clock.

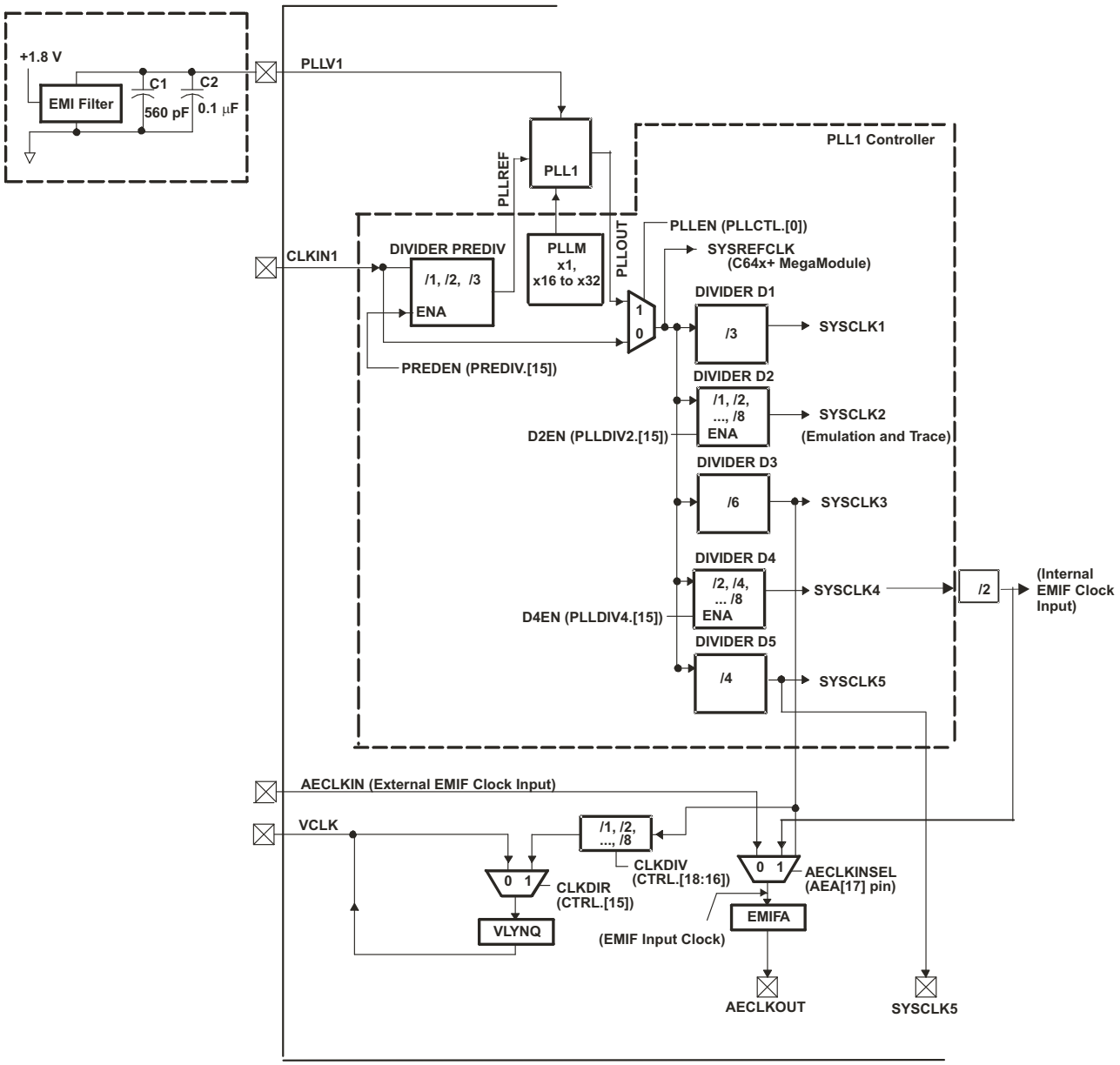



**Figure 6-5. PLL Input Clock**

As shown in Figure 6-5, the PLL1 controller features a software-programmable PLL multiplier controller (PLLM) and seven dividers (PREDIV, D1, D2, D3, D4, D5). The PLL1 controller uses the device input clock CLKIN1 to generate a system reference clock (SYSREFCLK) and five system clocks (SYSCLK1, SYSCLK2, SYSCLK3, SYSCLK4, and SYSCLK5). PLL1 power is supplied externally via the PLL1 power-supply pin (PLLV1). An external EMI filter circuit must be added to PLLV1. The 1.8-V supply of the EMI filter must be from the same 1.8-V power plane supplying the I/O power-supply pin, $DV_{DD18}$. TI requires EMI filter manufacturer Murata, part number NFM18CC222R1C3.

All PLL external components (C1, C2, and the EMI Filter) must be placed as close to the C64x+ 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. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (C1, C2, and the EMI Filter). The minimum CLKIN1 rise and fall times should also be observed. For the input clock timing requirements, see Section 6.4.4.

### 6.4.1 PLL1 Controller Device-Specific Information

As shown in Figure 6-5, the PLL1 controller generates several internal clocks including the system reference clock (SYSREFCLK), and the system clocks (SYSCLK1/2/3/4/5). The high-frequency clock signal SYSREFCLK is directly used to clock the C64x+ megamodule (including the CPU) and also serves as a reference clock for the rest of the DSP system. Dividers D1, D2, D3, D4, and D5 divide the high-frequency clock SYSREFCLK to generate SYSCLK1, SYSCLK2, SYSCLK3, SYSCLK4, and SYSCLK5, respectively.

The system clocks are used to clock different portions of the DSP as follows:

- SYSCLK1 is used for the following modules: 3PDMA, the SCR and the bridges, DDR Subsystem internal logic, and EMIFA internal logic.
- SYSCLK2 is used for Emulation and Trace
- SYSCLK3 is used for most of the peripherals. These modules are clocked from SYSCLK3: HPI, PCI, VLYNQ, UART, I2C, TIMER 0, TIMER 1, TIMER 2, TIMER 3, SPI, McASP, GPIO, PLL Controller 1, PLL Controller 2, Config SCR
- SYSCLK4 is used as the EMIFA AECLKOUT

  [When SYSCLK4 is used as the EMIF input clock source, the actual clock goes through a divider and the frequency would be SYSCLK4 divide-by-2 (see Figure 6-5, *PLL Input Clock*).]

The PLL multiplier controller (PLLM) must be programmed after reset. There is no hardware CLKMODE selection on the device. Since the divider ratio bits for dividers D1, D3, and D5 are fixed, the frequency of SYSCLK1, SYCLK3, and SYSCLK5 is tied to the frequency of SYSREFCLK. However, the frequency of SYSCLK2 and SYSCLK4 depends on the configuration of dividers D2 and D4. For example, with PLLM in the PLL1 multiply control register set to 10011b (x20 mode) and a 35-MHz CLKIN1 input, the PLL output PLLOUT is set to 700 MHz and SYSCLK1 and SYSCLK3 run at 233 MHz and 117 MHz, respectively. Divider D4 can be programmed through the PLLDIV4 register to divide SYSREFCLK by 8 (2 * (PLLDIV4.RATIO+1)) such that SYSCLK4 runs at 87.5 MHz.

Note that there is a minimum and maximum operating frequency for PLLREF, PLLOUT, SYSCLK4, and SYSCLK5. The PLL1 Controller must not be configured to exceed any of these constraints (certain combinations of external clock input, internal dividers, and PLL multiply ratios might not be supported). For the PLL clocks input and output frequency ranges, see Table 6-6.

**Table 6-6. PLL1 Clock Frequency Ranges**

| CLOCK SIGNAL | MIN | MAX | UNIT |
|---|---|---|---|
| CLKIN1 | 25 | 66.6 | MHz |
| PLLREF (PLLEN = 1)(1) | 25 | 66.6 | MHz |
| PLLOUT (1) | 400 | 720 (-720 devices) | MHz |
| | 400 | 900 (-900 devices) | MHz |
| SYSCLK4 | 16P(2) | 87.5(3)/112.5(4) | MHz |

(1) Only applies when the PLL1 Controller is set to PLL mode (PLLEN = 1 in the PLLCTL register).
(2) P = 1/CPU clock frequency in ns
(3) For 700 MHz PLLOUT
(4) For 900 MHz PLLOUT

### 6.4.2 PLL1 Controller Operating Modes

The PLL1 controller has two modes of operation: bypass mode and PLL mode. The mode of operation is determined by the PLLEN bit of the PLL control register (PLLCTL). In PLL mode, SYSREFCLK is generated from the device input clock CLKIN1 using the divider PREDIV and the PLL multiplier PLLM. In bypass mode, CLKIN1 is fed directly to SYSREFCLK.

All hosts (i.e., HPI) must hold off accesses to the DSP while the frequency of its internal clocks is changing. A mechanism must be in place such that the DSP notifies the host when the PLL configuration has completed.

### 6.4.3 PLL1 Stabilization, Lock, and Reset Times

The PLL stabilization time is the amount of time that must be allotted for the internal PLL regulators to become stable after device power-up. The PLL should not be operated until this stabilization time has finished.

The PLL reset time is the amount of wait time needed when resetting the PLL (writing PLLRST = 1) for the PLL to properly reset, before bringing the PLL out of reset (writing PLLRST = 0). For the PLL1 reset time value, see Table 6-7.

**Table 6-7. PLL1 Stabilization, Lock, and Reset Times**

| | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|
| PLL stabilization time | 150 | | | µs |
| PLL lock time | | | 2000 × C(1) | µs |
| PLL reset time | 128 × C(1) | | | µs |

(1) C = CLKIN1 cycle time in ns. For example, when CLKIN1 frequency is 50 MHz, use C = 20 ns.

### 6.4.4 PLL1 Controller Input and Output Clock Electrical Data/Timing

**Table 6-8. Timing Requirements for CLKIN1 (1) (2) (3) (see Figure 6-6)**

| NO. | | | 720 900 PLL MODES x1 (Bypass), x15, x20, x25, x30, x32 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(CLKIN1)}$ | Cycle time, CLKIN1 | 15 | 40 | ns |
| 2 | $t_{w(CLKIN1H)}$ | Pulse duration, CLKIN1 high | 0.4C | | ns |
| 3 | $t_{w(CLKIN1L)}$ | Pulse duration, CLKIN1 low | 0.4C | | ns |
| 4 | $t_{t(CLKIN1)}$ | Transition time, CLKIN1 | | 1.2 | ns |
| 5 | $t_{J(CLKIN1)}$ | Period jitter, (peak-to-peak), CLKIN1 | | 100 | ps |

(1) The reference points for the rise and fall transitions are measured at 3.3-V $V_{IL}$ MAX and $V_{IH}$ MIN.
(2) C = CLKIN1 cycle time in ns. For example, when CLKIN1 frequency is 50 MHz, use C = 20 ns.
(3) The PLL1 multiplier factors (x1 [BYPASS], x 15 to x32) further limit the MIN and MAX values for tc(CLKIN1). For more detailed information on these limitations, see Section 6.3.5, Power and Clock Domains.

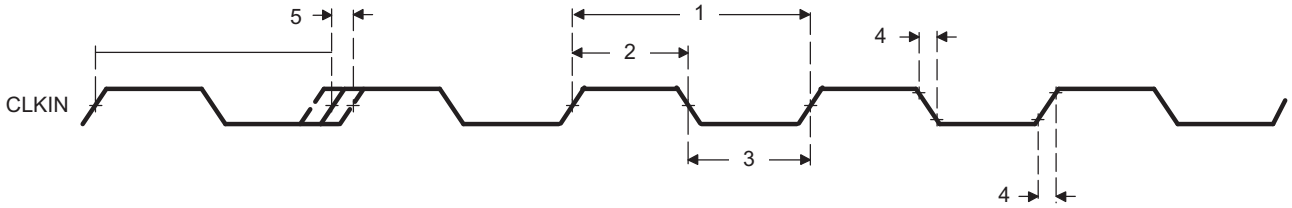



**Figure 6-6. CLKIN1 Timing**

### 6.4.5 PLL1 Controller Register Description(s)

A summary of the PLL1 controller registers is shown in Table 6-9.

**Table 6-9. PLL1 and Reset Controller Registers Memory Map**

| HEX ADDRESS RANGE | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x020E 0000 | PID | Peripheral Identification and Revision Information Register |
| 0x020E 00E4 | RSTYPE | Reset Type Register |
| 0x020E 0100 | PLLCTL | PLL Controller 1 Operations Control Register |
| 0x020E 0110 | PLLM | PLL Controller 1 Multiplier Control Register |
| 0x020E 0114 | PREDIV | PLL Pre-Divider Control Register |
| 0x020E 011C | PLLDIV2 | PLL Controller 1 Control-Divider 2 Register (SYSCLK2) |
| 0x020E 0138 | PLLCMD | PLL Controller 1 Command Register |
| 0x020E 013C | PLLSTAT | PLL Controller 1 Status Register (Shows PLLC1 Status) |
| 0x020E 0140 | ALNCTL | PLL Controller Clock Align Control Register |
| 0x020E 0144 | DCHANGE | PLLDIV Ratio Change Status Register |
| 0x020E 0150 | SYSTAT | PLL Controller 1 System Clock Status 1 Register (Indicates SYSCLK on/off Status) |
| 0x020E 0160 | PLLDIV4 | PLL Controller 1 Control-Divider 4 Register (SYSCLK4) |

## 6.5 PLL2 Controller

The secondary PLL controller generates interface clocks for the DDR2 memory controller.

As shown in Figure 6-7, the PLL2 controller features a PLL multiplier controller. The PLL multiplier is fixed to a x20 multiplier rate. PLL2 power is supplied externally via the PLL2 power supply (PLLV2). An external PLL filter circuit must be added to PLLV2 as shown in Figure 6-7. The 1.8-V supply for the EMI filter must be from the same 1.8-V power plane supplying the I/O power-supply pin, $DV_{DD18}$. TI requires EMI filter manufacturer Murata, part number NFM18CC222R1C3.

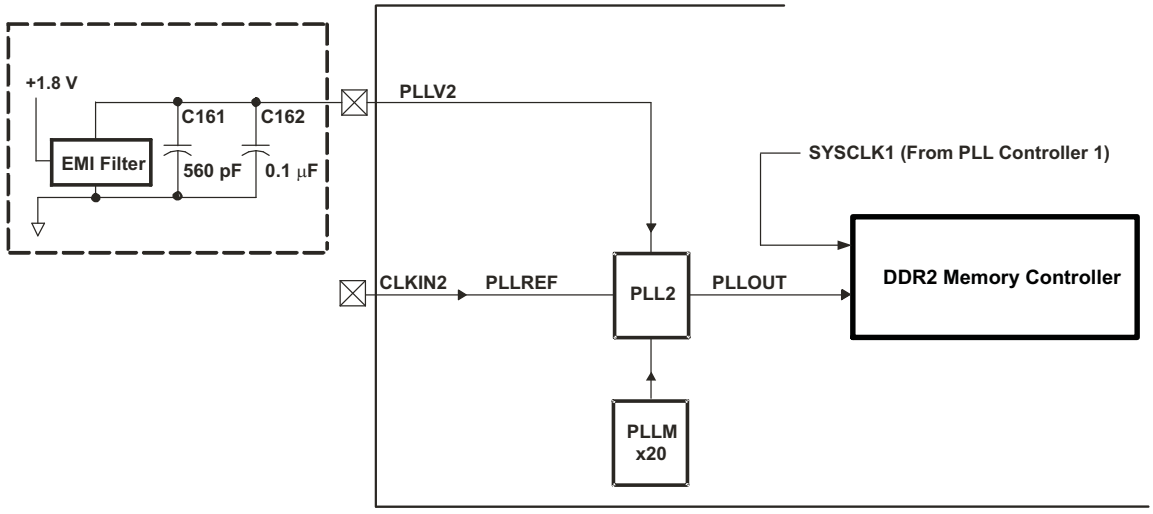



**Figure 6-7. PLL Controller**

All PLL external components (C161, C162, and the EMI Filter) should be placed as close to the C64x+ DSP device as possible. For the best performance, TI requires that all the PLL external components be on a single side of the board without jumpers, switches, or components other than the ones shown. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (C161, C162, and the EMI Filter). The minimum CLKIN2 rise and fall times should also be observed. For the input clock timing requirements, see Section 6.5.3, PLL2 Controller Input Clock Electrical Data/Timing.

### 6.5.1 PLL2 Controller Device-Specific Information

As shown in Figure 6-7, the output of PLL2, PLLOUT, is directly fed to the DDR2 memory controller. This clock is used by the DDR2 memory controller to generate DDR_CLK and $\overline{\text{DDR\_CLK}}$. Note that, internally, the data bus interface of the DDR2 memory controller is clocked by SYSCLK1 of the PLL1 controller.

Note that there is a minimum and maximum operating frequency for PLLREF and PLLOUT. The clock generator must not be configured to exceed any of these constraints. For the PLL clocks input and output frequency ranges, see Table 6-10.

**Table 6-10. PLL2 Clock Frequency Ranges**

| CLOCK SIGNAL | REQUIRED FREQUENCY | UNIT |
|---|---|---|
| PLLREF (CLKIN2 ) | 20 - 26.6 | MHz |
| PLLOUT (DDR2 clock) | 400 - 533 (1) | MHz |

(1) This clock is the 2x of the DDR clock.

### 6.5.2 PLL2 Controller Operating Modes

Unlike the PLL1 controller that can operate in bypass and a PLL mode, the PLL2 controller only operates in PLL mode. PLL2 is unlocked only during the power-up sequence (see Section 6.7) and is locked by the time the $\overline{\text{RESETSTAT}}$ pin goes high. It does not lose lock during any of the other resets.

### 6.5.3 PLL2 Controller Input Clock Electrical Data/Timing

**Table 6-11. Timing Requirements for CLKIN2(1) (2) (see Figure 6-8)**

| NO. | | | 720 900 PLL MODE x20 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(CLKIN2)}$ | Cycle time, CLKIN2 | 37.5 | 50 | ns |
| 2 | $t_{w(CLKIN2H)}$ | Pulse duration, CLKIN2 high | 0.4C | | ns |
| 3 | $t_{w(CLKIN2L)}$ | Pulse duration, CLKIN2 low | 0.4C | | ns |
| 4 | $t_{t(CLKIN2)}$ | Transition time, CLKIN2 | | 1.2 | ns |
| 5 | $t_{J(CLKIN2)}$ | Period jitter, (peak-to-peak) CLKIN2 | | 100 | ps |

(1) The reference points for the rise and fall transitions are measured at 3.3-V $V_{IL}$ MAX and $V_{IH}$ MIN.
(2) C = CLKIN2 cycle time in ns. For example, when CLKIN2 frequency is 25 MHz, use C = 40 ns.

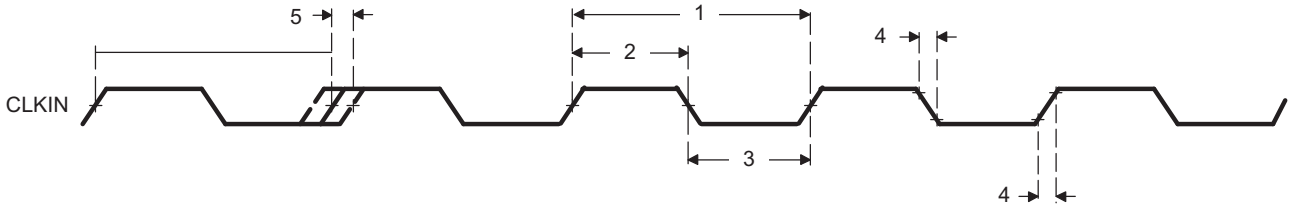



**Figure 6-8. CLKIN2 Timing**

### 6.5.4 PLL2 Controller Register Description(s)

A summary of the PLL2 controller registers is shown in Table 6-12.

**Table 6-12. PLL2 and Reset Controller Registers Memory Map**

| HEX ADDRESS RANGE | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x0212 0000 | PID | Peripheral Identification and Revision Information Register |
| 0x0212 0100 | PLLCTL | PLL Controller 2 Operations Control Register |
| 0x0212 0110 | PLLM | PLL Controller 2 Multiplier Control Register |
| 0x0212 0138 | PLLCMD | PLL Controller 2 Command Register |
| 0x0212 013C | PLLSTAT | PLL Controller 2 Status Register (Shows PLLC1 Status) |

## 6.6 Enhanced Direct Memory Access (EDMA3) Controller

The EDMA controller handles all data transfers between memories and the device slave peripherals on the device. These data transfers include cache servicing, non-cacheable memory accesses, user-programmed data transfers, and host accesses. These are summarized as follows:

- Transfer to/from on-chip memories
  - DSP L1D memory
  - DSP L2 memory
- Transfer to/from external storage
  - DDR2 SDRAM
  - Synchronous/Asynchronous EMIF (EMIFA)
- Transfer to/from peripherals/hosts
  - VLYNQ
  - HPI
  - McASP
  - UART
  - Timer 0/1/2/3
  - SPI
  - I2C

### *6.6.1 EDMA3 Channel Synchronization Events*

The EDMA supports up to 64 EDMA channels that service peripheral devices and external memory. Table 6-13 lists the source of EDMA synchronization events associated with each of the programmable EDMA channels. .

**Table 6-13. EDMA Channel Synchronization Events**

| TPCC CHANNEL | DEFAULT EVENT# | BINARY | DEFAULT EVENT | TPCC CHANNEL | DEFAULT EVENT # | BINARY | DEFAULT EVENT |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 000 0000 | HPI/PCI : DSPINT | 32 | 32 | 010 0000 | Reserved |
| 1 | 1 | 000 0001 | TIMER0 : TINT0L | 33 | 33 | 010 0001 | Reserved |
| 2 | 2 | 000 0010 | TIMER0 : TINT0H | 34 | 34 | 010 0010 | Reserved |
| 3 | 3 | 000 0011 | TIMER2 : TINT2L | 35 | 35 | 010 0011 | Reserved |
| 4 | 4 | 000 0100 | TIMER2 : TINT2H | 36 | 36 | 010 0100 | Reserved |
| 5 | 5 | 000 0101 | TIMER3 : TINT3L | 37 | 37 | 010 0101 | Reserved |
| 6 | 6 | 000 0110 | TIMER3 : TINT3H | 38 | 38 | 010 0110 | Reserved |
| 7 | 7 | 000 0111 | Reserved | 39 | 39 | 010 0111 | Reserved |
| 8 | 8 | 000 1000 | Reserved | 40 | 40 | 010 1000 | Reserved |
| 9 | 9 | 000 1001 | Reserved | 41 | 41 | 010 1001 | Reserved |
| 10 | 10 | 000 1010 | McASP: AXEVTE | 42 | 42 | 010 1010 | Reserved |
| 11 | 11 | 000 1011 | McASP: AXEVTO | 43 | 43 | 010 1011 | Reserved |
| 12 | 12 | 000 1100 | McASP: AXEVT | 44 | 44 | 010 1100 | ICREVT |
| 13 | 13 | 000 1101 | McASP: AREVTE | 45 | 45 | 010 1101 | ICXEVT |
| 14 | 14 | 000 1110 | McASP: AREVTO | 46 | 46 | 010 1110 | SPI: SPIXEVT |
| 15 | 15 | 000 1111 | McASP: AREVT | 47 | 47 | 010 1111 | SPI: SPIREVT |
| 16 | 16 | 001 0000 | TIMER1 : TINT1L | 48 | 48 | 011 0000 | Reserved |
| 17 | 17 | 001 0001 | TIMER1 : TINT1H | 49 | 49 | 011 0001 | Reserved |
| 18 | 18 | 001 0010 | UART: URXEVT | 50 | 50 | 011 0010 | Reserved |
| 19 | 19 | 001 0011 | UART: UTXEVT | 51 | 51 | 011 0011 | Reserved |
| 20 | 20 | 001 0100 | Reserved | 52 | 52 | 011 0100 | Reserved |
| 21 | 21 | 001 0101 | Reserved | 53 | 53 | 011 0101 | Reserved |

**Table 6-13. EDMA Channel Synchronization Events (continued)**

| TPCC CHANNEL | DEFAULT EVENT# | BINARY | DEFAULT EVENT | TPCC CHANNEL | DEFAULT EVENT # | BINARY | DEFAULT EVENT |
|---|---|---|---|---|---|---|---|
| 22 | 22 | 001 0110 | Reserved | 54 | 54 | 011 0110 | GPIO : GPINT6 |
| 23 | 23 | 001 0111 | Reserved | 55 | 55 | 011 0111 | GPIO : GPINT7 |
| 24 | 24 | 001 1000 | Reserved | 56 | 56 | 011 1000 | GPIO : GPINT8 |
| 25 | 25 | 001 1001 | Reserved | 57 | 57 | 011 1001 | GPIO : GPINT9 |
| 26 | 26 | 001 1010 | Reserved | 58 | 58 | 011 1010 | GPIO : GPINT10 |
| 27 | 27 | 001 1011 | Reserved | 59 | 59 | 011 1011 | GPIO : GPINT11 |
| 28 | 28 | 001 1100 | Reserved | 60 | 60 | 011 1100 | GPIO : GPINT12 |
| 29 | 29 | 001 1101 | Reserved | 61 | 61 | 011 1101 | GPIO : GPINT13 |
| 30 | 30 | 001 1110 | Reserved | 62 | 62 | 011 1110 | GPIO : GPINT14 |
| 31 | 31 | 001 1111 | Reserved | 63 | 63 | 011 1111 | GPIO : GPINT15 |

### 6.6.2 EDMA Peripheral Register Description(s)

Table 6-14 lists the EDMA registers, their corresponding acronyms, and device memory locations.

**Table 6-14. EDMA Channel Controller Registers**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 0000 | PID | Peripheral ID Register |
| 0x02A0 0004 | CCCFG | EDMA3CC Configuration Register |
| 0x02A0 0008 - 0x02A0 00FC | | Reserved |
| 0x02A0 0100 | DCHMAP0 | DMA Channel 0 Mapping Register |
| 0x02A0 0104 | DCHMAP1 | DMA Channel 1 Mapping Register |
| 0x02A0 0108 | DCHMAP2 | DMA Channel 2 Mapping Register |
| 0x02A0 010C | DCHMAP3 | DMA Channel 3 Mapping Register |
| 0x02A0 0110 | DCHMAP4 | DMA Channel 4 Mapping Register |
| 0x02A0 0114 | DCHMAP5 | DMA Channel 5 Mapping Register |
| 0x02A0 0118 | DCHMAP6 | DMA Channel 6 Mapping Register |
| 0x02A0 011C | DCHMAP7 | DMA Channel 7 Mapping Register |
| 0x02A0 0120 | DCHMAP8 | DMA Channel 8 Mapping Register |
| 0x02A0 0124 | DCHMAP9 | DMA Channel 9 Mapping Register |
| 0x02A0 0128 | DCHMAP10 | DMA Channel 10 Mapping Register |
| 0x02A0 012C | DCHMAP11 | DMA Channel 11 Mapping Register |
| 0x02A0 0130 | DCHMAP12 | DMA Channel 12 Mapping Register |
| 0x02A0 0134 | DCHMAP13 | DMA Channel 13 Mapping Register |
| 0x02A0 0138 | DCHMAP14 | DMA Channel 14 Mapping Register |
| 0x02A0 013C | DCHMAP15 | DMA Channel 15 Mapping Register |
| 0x02A0 0140 | DCHMAP16 | DMA Channel 16 Mapping Register |
| 0x02A0 0144 | DCHMAP17 | DMA Channel 17 Mapping Register |
| 0x02A0 0148 | DCHMAP18 | DMA Channel 18 Mapping Register |
| 0x02A0 014C | DCHMAP19 | DMA Channel 19 Mapping Register |
| 0x02A0 0150 | DCHMAP20 | DMA Channel 20 Mapping Register |
| 0x02A0 0154 | DCHMAP21 | DMA Channel 21 Mapping Register |
| 0x02A0 0158 | DCHMAP22 | DMA Channel 22 Mapping Register |
| 0x02A0 015C | DCHMAP23 | DMA Channel 23 Mapping Register |
| 0x02A0 0160 | DCHMAP24 | DMA Channel 24 Mapping Register |
| 0x02A0 0164 | DCHMAP25 | DMA Channel 25 Mapping Register |
| 0x02A0 0168 | DCHMAP26 | DMA Channel 26 Mapping Register |

**Table 6-14. EDMA Channel Controller Registers (continued)**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 016C | DCHMAP27 | DMA Channel 27 Mapping Register |
| 0x02A0 0170 | DCHMAP28 | DMA Channel 28 Mapping Register |
| 0x02A0 0174 | DCHMAP29 | DMA Channel 29 Mapping Register |
| 0x02A0 0178 | DCHMAP30 | DMA Channel 30 Mapping Register |
| 0x02A0 017C | DCHMAP31 | DMA Channel 31 Mapping Register |
| 0x02A0 0180 | DCHMAP32 | DMA Channel 32 Mapping Register |
| 0x02A0 0184 | DCHMAP33 | DMA Channel 33 Mapping Register |
| 0x02A0 0188 | DCHMAP34 | DMA Channel 34 Mapping Register |
| 0x02A0 018C | DCHMAP35 | DMA Channel 35 Mapping Register |
| 0x02A0 0190 | DCHMAP36 | DMA Channel 36 Mapping Register |
| 0x02A0 0194 | DCHMAP37 | DMA Channel 37 Mapping Register |
| 0x02A0 0198 | DCHMAP38 | DMA Channel 38 Mapping Register |
| 0x02A0 019C | DCHMAP39 | DMA Channel 39 Mapping Register |
| 0x02A0 01A0 | DCHMAP40 | DMA Channel 40 Mapping Register |
| 0x02A0 01A4 | DCHMAP41 | DMA Channel 41 Mapping Register |
| 0x02A0 01A8 | DCHMAP42 | DMA Channel 42 Mapping Register |
| 0x02A0 01AC | DCHMAP43 | DMA Channel 43 Mapping Register |
| 0x02A0 01B0 | DCHMAP44 | DMA Channel 44 Mapping Register |
| 0x02A0 01B4 | DCHMAP45 | DMA Channel 45 Mapping Register |
| 0x02A0 01B8 | DCHMAP46 | DMA Channel 46 Mapping Register |
| 0x02A0 01BC | DCHMAP47 | DMA Channel 47 Mapping Register |
| 0x02A0 01C0 | DCHMAP48 | DMA Channel 48 Mapping Register |
| 0x02A0 01C4 | DCHMAP49 | DMA Channel 49 Mapping Register |
| 0x02A0 01C8 | DCHMAP50 | DMA Channel 50 Mapping Register |
| 0x02A0 01CC | DCHMAP51 | DMA Channel 51 Mapping Register |
| 0x02A0 01D0 | DCHMAP52 | DMA Channel 52 Mapping Register |
| 0x02A0 01D4 | DCHMAP53 | DMA Channel 53 Mapping Register |
| 0x02A0 01D8 | DCHMAP54 | DMA Channel 54 Mapping Register |
| 0x02A0 01DC | DCHMAP55 | DMA Channel 55 Mapping Register |
| 0x02A0 01E0 | DCHMAP56 | DMA Channel 56 Mapping Register |
| 0x02A0 01E4 | DCHMAP57 | DMA Channel 57 Mapping Register |
| 0x02A0 01E8 | DCHMAP58 | DMA Channel 58 Mapping Register |
| 0x02A0 01EC | DCHMAP59 | DMA Channel 59 Mapping Register |
| 0x02A0 01F0 | DCHMAP60 | DMA Channel 60 Mapping Register |
| 0x02A0 01F4 | DCHMAP61 | DMA Channel 61 Mapping Register |
| 0x02A0 01F8 | DCHMAP62 | DMA Channel 62 Mapping Register |
| 0x02A0 01FC | DCHMAP63 | DMA Channel 63 Mapping Register |
| 0x02A0 0200 | QCHMAP0 | QDMA Channel 0 Mapping to PaRAM Register |
| 0x02A0 0204 | QCHMAP1 | QDMA Channel 1 Mapping to PaRAM Register |
| 0x02A0 0208 | QCHMAP2 | QDMA Channel 2 Mapping to PaRAM Register |
| 0x02A0 020C | QCHMAP3 | QDMA Channel 3 Mapping to PaRAM Register |
| 0x02A0 0210 | QCHMAP4 | QDMA Channel 4 Mapping to PaRAM Register |
| 0x02A0 0214 | QCHMAP5 | QDMA Channel 5 Mapping to PaRAM Register |
| 0x02A0 0218 | QCHMAP6 | QDMA Channel 6 Mapping to PaRAM Register |
| 0x02A0 021C | QCHMAP7 | QDMA Channel 7 Mapping to PaRAM Register |
| 0x02A0 0220 - 0x02A0 021C | - | Reserved |
| 0x02A0 0220 - 0x02A0 023C | - | Reserved |

**Table 6-14. EDMA Channel Controller Registers (continued)**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 0240 | DMAQNUM0 | DMA Queue Number Register 0 (Channels 00 to 07) |
| 0x02A0 0244 | DMAQNUM1 | DMA Queue Number Register 1 (Channels 08 to 15) |
| 0x02A0 0248 | DMAQNUM2 | DMA Queue Number Register 2 (Channels 16 to 23) |
| 0x02A0 024C | DMAQNUM3 | DMA Queue Number Register 3 (Channels 24 to 31) |
| 0x02A0 0250 | DMAQNUM4 | DMA Queue Number Register 4 (Channels 32 to 39) |
| 0x02A0 0254 | DMAQNUM5 | DMA Queue Number Register 5 (Channels 40 to 47) |
| 0x02A0 0258 | DMAQNUM6 | DMA Queue Number Register 6 (Channels 48 to 55) |
| 0x02A0 025C | DMAQNUM7 | DMA Queue Number Register 7 (Channels 56 to 63) |
| 0x02A0 0260 | QDMAQNUM | CC QDMA Queue Number |
| 0x02A0 0264 - 0x02A0 0280 | - | Reserved |
| 0x02A0 0284 | QUEPRI | Queue Priority Register |
| 0x02A0 0288 - 0x02A0 02FC | - | Reserved |
| 0x02A0 0300 | EMR | Event Missed Register |
| 0x02A0 0304 | EMRH | Event Missed Register High |
| 0x02A0 0308 | EMCR | Event Missed Clear Register |
| 0x02A0 030C | EMCRH | Event Missed Clear Register High |
| 0x02A0 0310 | QEMR | QDMA Event Missed Register |
| 0x02A0 0314 | QEMCR | QDMA Event Missed Clear Register |
| 0x02A0 0318 | CCERR | EDMA3CC Error Register |
| 0x02A0 031C | CCERRCLR | EDMA3CC Error Clear Register |
| 0x02A0 0320 | EEVAL | Error Evaluate Register |
| 0x02A0 0324 - 0x02A0 033C | - | Reserved |
| 0x02A0 0340 | DRAE0 | DMA Region Access Enable Register for Region 0 |
| 0x02A0 0344 | DRAEH0 | DMA Region Access Enable Register High for Region 0 |
| 0x02A0 0348 | DRAE1 | DMA Region Access Enable Register for Region 1 |
| 0x02A0 034C | DRAEH1 | DMA Region Access Enable Register High for Region 1 |
| 0x02A0 0350 | DRAE2 | DMA Region Access Enable Register for Region 2 |
| 0x02A0 0354 | DRAEH2 | DMA Region Access Enable Register High for Region 2 |
| 0x02A0 0358 | DRAE3 | DMA Region Access Enable Register for Region 3 |
| 0x02A0 035C | DRAEH3 | DMA Region Access Enable Register High for Region 3 |
| 0x02A0 0360 | DRAE4 | DMA Region Access Enable Register for Region 4 |
| 0x02A0 0364 | DRAEH4 | DMA Region Access Enable Register High for Region 4 |
| 0x02A0 0368 | DRAE5 | DMA Region Access Enable Register for Region 5 |
| 0x02A0 036C | DRAEH5 | DMA Region Access Enable Register High for Region 5 |
| 0x02A0 0370 | DRAE6 | DMA Region Access Enable Register for Region 6 |
| 0x02A0 0374 | DRAEH6 | DMA Region Access Enable Register High for Region 6 |
| 0x02A0 0378 | DRAE7 | DMA Region Access Enable Register for Region 7 |
| 0x02A0 037C | DRAEH7 | DMA Region Access Enable Register High for Region 7 |
| 0x02A0 0380 | QRAE0 | QDMA Region Access Enable Register for Region 0 |
| 0x02A0 0384 | QRAE1 | QDMA Region Access Enable Register for Region 1 |
| 0x02A0 0388 | QRAE2 | QDMA Region Access Enable Register for Region 2 |
| 0x02A0 038C | QRAE3 | QDMA Region Access Enable Register for Region 3 |
| 0x02A0 0390 | QRAE4 | QDMA Region Access Enable Register for Region 4 |
| 0x02A0 0394 | QRAE5 | QDMA Region Access Enable Register for Region 5 |
| 0x02A0 0398 | QRAE6 | QDMA Region Access Enable Register for Region 6 |
| 0x02A0 039C | QRAE7 | QDMA Region Access Enable Register for Region 7 |
| 0x02A0 0400 | Q0E0 | Event Queue 0 Entry Register 0 |

**Table 6-14. EDMA Channel Controller Registers (continued)**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 0404 | Q0E1 | Event Queue 0 Entry Register 1 |
| 0x02A0 0408 | Q0E2 | Event Queue 0 Entry Register 2 |
| 0x02A0 040C | Q0E3 | Event Queue 0 Entry Register 3 |
| 0x02A0 0410 | Q0E4 | Event Queue 0 Entry Register 4 |
| 0x02A0 0414 | Q0E5 | Event Queue 0 Entry Register 5 |
| 0x02A0 0418 | Q0E6 | Event Queue 0 Entry Register 6 |
| 0x02A0 041C | Q0E7 | Event Queue 0 Entry Register 7 |
| 0x02A0 0420 | Q0E8 | Event Queue 0 Entry Register 8 |
| 0x02A0 0424 | Q0E9 | Event Queue 0 Entry Register 9 |
| 0x02A0 0428 | Q0E10 | Event Queue 0 Entry Register 10 |
| 0x02A0 042C | Q0E11 | Event Queue 0 Entry Register 11 |
| 0x02A0 0430 | Q0E12 | Event Queue 0 Entry Register 12 |
| 0x02A0 0434 | Q0E13 | Event Queue 0 Entry Register 13 |
| 0x02A0 0438 | Q0E14 | Event Queue 0 Entry Register 14 |
| 0x02A0 043C | Q0E15 | Event Queue 0 Entry Register 15 |
| 0x02A0 0440 | Q1E0 | Event Queue 1 Entry Register 0 |
| 0x02A0 0444 | Q1E1 | Event Queue 1 Entry Register 1 |
| 0x02A0 0448 | Q1E2 | Event Queue 1 Entry Register 2 |
| 0x02A0 044C | Q1E3 | Event Queue 1 Entry Register 3 |
| 0x02A0 0450 | Q1E4 | Event Queue 1 Entry Register 4 |
| 0x02A0 0454 | Q1E5 | Event Queue 1 Entry Register 5 |
| 0x02A0 0458 | Q1E6 | Event Queue 1 Entry Register 6 |
| 0x02A0 045C | Q1E7 | Event Queue 1 Entry Register 7 |
| 0x02A0 0460 | Q1E8 | Event Queue 1 Entry Register 8 |
| 0x02A0 0464 | Q1E9 | Event Queue 1 Entry Register 9 |
| 0x02A0 0468 | Q1E10 | Event Queue 1 Entry Register 10 |
| 0x02A0 046C | Q1E11 | Event Queue 1 Entry Register 11 |
| 0x02A0 0470 | Q1E12 | Event Queue 1 Entry Register 12 |
| 0x02A0 0474 | Q1E13 | Event Queue 1 Entry Register 13 |
| 0x02A0 0478 | Q1E14 | Event Queue 1 Entry Register 14 |
| 0x02A0 047C | Q1E15 | Event Queue 1 Entry Register 15 |
| 0x02A0 0480 | Q2E0 | Event Queue 2 Entry Register 0 |
| 0x02A0 0484 | Q2E1 | Event Queue 2 Entry Register 1 |
| 0x02A0 0488 | Q2E2 | Event Queue 2 Entry Register 2 |
| 0x02A0 048C | Q2E3 | Event Queue 2 Entry Register 3 |
| 0x02A0 0490 | Q2E4 | Event Queue 2 Entry Register 4 |
| 0x02A0 0494 | Q2E5 | Event Queue 2 Entry Register 5 |
| 0x02A0 0498 | Q2E6 | Event Queue 2 Entry Register 6 |
| 0x02A0 049C | Q2E7 | Event Queue 2 Entry Register 7 |
| 0x02A0 04A0 | Q2E8 | Event Queue 2 Entry Register 8 |
| 0x02A0 04A4 | Q2E9 | Event Queue 2 Entry Register 9 |
| 0x02A0 04A8 | Q2E10 | Event Queue 2 Entry Register 10 |
| 0x02A0 04AC | Q2E11 | Event Queue 2 Entry Register 11 |
| 0x02A0 04B0 | Q2E12 | Event Queue 2 Entry Register 12 |
| 0x02A0 04B4 | Q2E13 | Event Queue 2 Entry Register 13 |
| 0x02A0 04B8 | Q2E14 | Event Queue 2 Entry Register 14 |
| 0x02A0 04BC | Q2E15 | Event Queue 2 Entry Register 15 |

**Table 6-14. EDMA Channel Controller Registers (continued)**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 04C0 | Q3E0 | Event Queue 3 Entry Register 0 |
| 0x02A0 04C4 | Q3E1 | Event Queue 3 Entry Register 1 |
| 0x02A0 04C8 | Q3E2 | Event Queue 3 Entry Register 2 |
| 0x02A0 04CC | Q3E3 | Event Queue 3 Entry Register 3 |
| 0x02A0 04D0 | Q3E4 | Event Queue 3 Entry Register 4 |
| 0x02A0 04D4 | Q3E5 | Event Queue 3 Entry Register 5 |
| 0x02A0 04D8 | Q3E6 | Event Queue 3 Entry Register 6 |
| 0x02A0 04DC | Q3E7 | Event Queue 3 Entry Register 7 |
| 0x02A0 04E0 | Q3E8 | Event Queue 3 Entry Register 8 |
| 0x02A0 04E4 | Q3E9 | Event Queue 3 Entry Register 9 |
| 0x02A0 04E8 | Q3E10 | Event Queue 3 Entry Register 10 |
| 0x02A0 04EC | Q3E11 | Event Queue 3 Entry Register 11 |
| 0x02A0 04F0 | Q3E12 | Event Queue 3 Entry Register 12 |
| 0x02A0 04F4 | Q3E13 | Event Queue 3 Entry Register 13 |
| 0x02A0 04F8 | Q3E14 | Event Queue 3 Entry Register 14 |
| 0x02A0 04FC | Q3E15 | Event Queue 3 Entry Register 15 |
| 0x02A0 0500 - 0x02A0 051C | - | Reserved |
| 0x02A0 0520 - 0x02A0 05FC | - | Reserved |
| 0x02A0 0600 | QSTAT0 | Queue 0 Status Register |
| 0x02A0 0604 | QSTAT1 | Queue 1 Status Register |
| 0x02A0 0608 | QSTAT2 | Queue Status Register 2 |
| 0x02A0 060C | QSTAT3 | Queue Status Register 3 |
| 0x02A0 0610 - 0x02A0 061C | - | Reserved |
| 0x02A0 0620 | QWMTHRA | Queue Watermark Threshold A Register for Q[3:0] |
| 0x02A0 0624 | - | Reserved |
| 0x02A0 0640 | CCSTAT | EDMA3CC Status Register |
| 0x02A0 0644 - 0x02A0 06FC | - | Reserved |
| 0x02A0 0700 - 0x02A0 07FC | - | Reserved |
| 0x02A0 0800 | MPFAR | Memory Protection Fault Address Register |
| 0x02A0 0804 | MPFSR | Memory Protection Fault Status Register |
| 0x02A0 0808 | MPFCR | Memory Protection Fault Command Register |
| 0x02A0 080C | MPPA0 | Memory Protection Page Attribute Register 0 |
| 0x02A0 0810 | MPPA1 | Memory Protection Page Attribute Register 1 |
| 0x02A0 0814 | MPPA2 | Memory Protection Page Attribute Register 2 |
| 0x02A0 0818 | MPPA3 | Memory Protection Page Attribute Register 3 |
| 0x02A0 081C | MPPA4 | Memory Protection Page Attribute Register 4 |
| 0x02A0 0820 | MPPA5 | Memory Protection Page Attribute Register 5 |
| 0x02A0 0824 | MPPA6 | Memory Protection Page Attribute Register 6 |
| 0x02A0 0828 | MPPA7 | Memory Protection Page Attribute Register 7 |
| 0x02A0 082C - 0x02A0 0FFC | - | Reserved |
| 0x02A0 1000 | ER | Event Register |
| 0x02A0 1004 | ERH | Event Register High |
| 0x02A0 1008 | ECR | Event Clear Register |
| 0x02A0 100C | ECRH | Event Clear Register High |
| 0x02A0 1010 | ESR | Event Set Register |
| 0x02A0 1014 | ESRH | Event Set Register High |
| 0x02A0 1018 | CER | Chained Event Register |

**Table 6-14. EDMA Channel Controller Registers (continued)**

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x02A0 101C | CERH | Chained Event Register High |
| 0x02A0 1020 | EER | Event Enable Register |
| 0x02A0 1024 | EERH | Event Enable Register High |
| 0x02A0 1028 | EECR | Event Enable Clear Register |
| 0x02A0 102C | EECRH | Event Enable Clear Register High |
| 0x02A0 1030 | EESR | Event Enable Set Register |
| 0x02A0 1034 | EESRH | Event Enable Set Register High |
| 0x02A0 1038 | SER | Secondary Event Register |
| 0x02A0 103C | SERH | Secondary Event Register High |
| 0x02A0 1040 | SECR | Secondary Event Clear Register |
| 0x02A0 1044 | SECRH | Secondary Event Clear Register High |
| 0x02A0 1048 - 0x02A0 104C | | Reserved |
| 0x02A0 1050 | IER | Interrupt Enable Register |
| 0x02A0 1054 | IERH | Interrupt Enable Register High |
| 0x02A0 1058 | IECR | Interrupt Enable Clear Register |
| 0x02A0 105C | IECRH | Interrupt Enable Clear Register High |
| 0x02A0 1060 | IESR | Interrupt Enable Set Register |
| 0x02A0 1064 | IESRH | Interrupt Enable Set Register High |
| 0x02A0 1068 | IPR | Interrupt Pending Register |
| 0x02A0 106C | IPRH | Interrupt Pending Register High |
| 0x02A0 1070 | ICR | Interrupt Clear Register |
| 0x02A0 1074 | ICRH | Interrupt Clear Register High |
| 0x02A0 1078 | IEVAL | Interrupt Evaluate Register |
| 0x02A0 107C | - | Reserved |
| 0x02A0 1080 | QER | QDMA Event Register |
| 0x02A0 1084 | QEER | QDMA Event Enable Register |
| 0x02A0 1088 | QEECR | QDMA Event Enable Clear Register |
| 0x02A0 108C | QEESR | QDMA Event Enable Set Register |
| 0x02A0 1090 | QSER | QDMA Secondary Event Register |
| 0x02A0 1094 | QSECR | QDMA Secondary Event Clear Register |
| 0x02A0 1098 - 0x02A0 1FFF | - | Reserved |
| 0x02A0 2000- 0x02A0 2097 | - | Shadow Region 0 Channel Registers |
| 0x02A0 2098 - 0x02A0 21FF | - | Reserved |
| 0x02A0 2200 - 0x02A0 2297 | - | Shadow Region 1 Channel Registers |
| 0x02A0 2298 - 0x02A0 23FF | - | Reserved |
| 0x02A0 2400 - 0x02A0 2497 | - | Shadow Region 2 Channel Registers |
| 0x02A0 2498 - 0x02A0 25FF | - | Reserved |
| 0x02A0 2600 - 0x02A0 2697 | - | Shadow Region 3 Channel Registers |
| 0x02A0 2698 - 0x02A0 27FF | - | Reserved |
| 0x02A0 2800 - 0x02A0 2897 | - | Shadow Region 4 Channel Registers |
| 0x02A0 2898 - 0x02A0 29FF | - | Reserved |
| 0x02A0 2A00 - 0x02A0 2A97 | - | Shadow Region 5 Channel Registers |
| 0x02A0 2A98 - 0x02A0 2BFF | - | Reserved |
| 0x02A0 2C00 - 0x02A0 2C97 | - | Shadow Region 6 Channel Registers |
| 0x02A0 2C98 - 0x02A0 2DFF | - | Reserved |
| 0x02A0 2E00 - 0x02A0 2E97 | - | Shadow Region 7 Channel Registers |
| 0x02A0 2E98 - 0x02A0 2FFF | - | Reserved |

Table 6-15 shows an abbreviation of the set of registers that make up the parameter set for each of 128 EDMA events. Each of the parameter register sets consist of eight 32-bit word entries. Table 6-16 shows the parameter set entry registers with relative memory address locations within each of the parameter sets.

**Table 6-15. EDMA Parameter Set RAM**

| HEX ADDRESS RANGE | DESCRIPTION |
|---|---|
| 0x02A0 4000 - 0x02A0 401F | Parameters Set 0 (8 32-bit words) |
| 0x02A0 4020 - 0x02A0 403F | Parameters Set 1 (8 32-bit words) |
| 0x02A0 4040 - 0x02A0 405F | Parameters Set 2 (8 32-bit words) |
| 0x02A0 4060 - 0x02A0 407F | Parameters Set 3 (8 32-bit words) |
| 0x02A0 4080 - 0x02A0 409F | Parameters Set 4 (8 32-bit words) |
| 0x02A0 40A0 - 0x02A0 40BF | Parameters Set 5 (8 32-bit words) |
| ... | ... |
| 0x02A0 4FC0 - 0x02A0 4FDF | Parameters Set 126 (8 32-bit words) |
| 0x02A0 4FE0 - 0x02A0 4FFF | Parameters Set 127 (8 32-bit words) |
| ... | ... |
| 0x02A0 5FC0 - 0x02A0 5FDF | Parameters Set 254 (8 32-bit words) |
| 0x02A0 5FE0 - 0x02A0 5FFF | Parameters Set 255 (8 32-bit words) |
| ... | ... |
| 0x02A0 7FC0 - 0x02A0 7FDF | Parameters Set 510 (8 32-bit words) |
| 0x02A0 7FE0 - 0x02A0 7FFF | Parameters Set 511 (8 32-bit words) |

**Table 6-16. Parameter Set Entries**

| HEX OFFSET ADDRESS WITHIN THE PARAMETER SET | ACRONYM | PARAMETER ENTRY |
|---|---|---|
| 0x0000 | OPT | Option |
| 0x0004 | SRC | Source Address |
| 0x0008 | A_B_CNT | A Count, B Count |
| 0x000C | DST | Destination Address |
| 0x0010 | SRC_DST_BIDX | Source B Index, Destination B Index |
| 0x0014 | LINK_BCNTRLD | Link Address, B Count Reload |
| 0x0018 | SRC_DST_CIDX | Source C Index, Destination C Index |
| 0x001C | CCNT | C Count |

**Table 6-17. EDMA3 Transfer Controller 0 Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A2 0000 | PID | Peripheral Identification Register |
| 02A2 0004 | TCCFG | EDMA3TC Configuration Register |
| 02A2 0008 - 02A2 00FC | - | Reserved |
| 02A2 0100 | TCSTAT | EDMA3TC Channel Status Register |
| 02A2 0104 - 02A2 011C | - | Reserved |
| 02A2 0120 | ERRSTAT | Error Register |
| 02A2 0124 | ERREN | Error Enable Register |
| 02A2 0128 | ERRCLR | Error Clear Register |
| 02A2 012C | ERRDET | Error Details Register |
| 02A2 0130 | ERRCMD | Error Interrupt Command Register |
| 02A2 0134 - 02A2 013C | - | Reserved |
| 02A2 0140 | RDRATE | Read Rate Register |

**Table 6-17. EDMA3 Transfer Controller 0 Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A2 0144 - 02A2 023C | - | Reserved |
| 02A2 0240 | SAOPT | Source Active Options Register |
| 02A2 0244 | SASRC | Source Active Source Address Register |
| 02A2 0248 | SACNT | Source Active Count Register |
| 02A2 024C | SADST | Source Active Destination Address Register |
| 02A2 0250 | SABIDX | Source Active Source B-Index Register |
| 02A2 0254 | SAMPPRXY | Source Active Memory Protection Proxy Register |
| 02A2 0258 | SACNTRLD | Source Active Count Reload Register |
| 02A2 025C | SASRCBREF | Source Active Source Address B-Reference Register |
| 02A2 0260 | SADSTBREF | Source Active Destination Address B-Reference Register |
| 02A2 0264 - 02A2 027C | - | Reserved |
| 02A2 0280 | DFCNTRLD | Destination FIFO Set Count Reload |
| 02A2 0284 | DFSRCBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A2 0288 | DFDSTBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A2 028C - 02A2 02FC | - | Reserved |
| 02A2 0300 | DFOPT0 | Destination FIFO Options Register 0 |
| 02A2 0304 | DFSRC0 | Destination FIFO Source Address Register 0 |
| 02A2 0308 | DFCNT0 | Destination FIFO Count Register 0 |
| 02A2 030C | DFDST0 | Destination FIFO Destination Address Register 0 |
| 02A2 0310 | DFBIDX0 | Destination FIFO BIDX Register 0 |
| 02A2 0314 | DFMPPRXY0 | Destination FIFO Memory Protection Proxy Register 0 |
| 02A2 0318 - 02A2 033C | - | Reserved |
| 02A2 0340 | DFOPT1 | Destination FIFO Options Register 1 |
| 02A2 0344 | DFSRC1 | Destination FIFO Source Address Register 1 |
| 02A2 0348 | DFCNT1 | Destination FIFO Count Register 1 |
| 02A2 034C | DFDST1 | Destination FIFO Destination Address Register 1 |
| 02A2 0350 | DFBIDX1 | Destination FIFO BIDX Register 1 |
| 02A2 0354 | DFMPPRXY1 | Destination FIFO Memory Protection Proxy Register 1 |
| 02A2 0358 - 02A2 037C | - | Reserved |
| 02A2 0380 | DFOPT2 | Destination FIFO Options Register 2 |
| 02A2 0384 | DFSRC2 | Destination FIFO Source Address Register 2 |
| 02A2 0388 | DFCNT2 | Destination FIFO Count Register 2 |
| 02A2 038C | DFDST2 | Destination FIFO Destination Address Register 2 |
| 02A2 0390 | DFBIDX2 | Destination FIFO BIDX Register 2 |
| 02A2 0394 | DFMPPRXY2 | Destination FIFO Memory Protection Proxy Register 2 |
| 02A2 0398 - 02A2 03BC | - | Reserved |
| 02A2 03C0 | DFOPT3 | Destination FIFO Options Register 3 |
| 02A2 03C4 | DFSRC3 | Destination FIFO Source Address Register 3 |
| 02A2 03C8 | DFCNT3 | Destination FIFO Count Register 3 |
| 02A2 03CC | DFDST3 | Destination FIFO Destination Address Register 3 |
| 02A2 03D0 | DFBIDX3 | Destination FIFO BIDX Register 3 |
| 02A2 03D4 | DFMPPRXY3 | Destination FIFO Memory Protection Proxy Register 3 |
| 02A2 03D8 - 02A2 7FFF | - | Reserved |

**Table 6-18. EDMA3 Transfer Controller 1 Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A2 8000 | PID | Peripheral Identification Register |
| 02A2 8004 | TCCFG | EDMA3TC Configuration Register |
| 02A2 8008 - 02A2 80FC | - | Reserved |
| 02A2 8100 | TCSTAT | EDMA3TC Channel Status Register |
| 02A2 8104 - 02A2 811C | - | Reserved |
| 02A2 8120 | ERRSTAT | Error Register |
| 02A2 8124 | ERREN | Error Enable Register |
| 02A2 8128 | ERRCLR | Error Clear Register |
| 02A2 812C | ERRDET | Error Details Register |
| 02A2 8130 | ERRCMD | Error Interrupt Command Register |
| 02A2 8134 - 02A2 813C | - | Reserved |
| 02A2 8140 | RDRATE | Read Rate Register |
| 02A2 8144 - 02A2 823C | - | Reserved |
| 02A2 8240 | SAOPT | Source Active Options Register |
| 02A2 8244 | SASRC | Source Active Source Address Register |
| 02A2 8248 | SACNT | Source Active Count Register |
| 02A2 824C | SADST | Source Active Destination Address Register |
| 02A2 8250 | SABIDX | Source Active Source B-Index Register |
| 02A2 8254 | SAMPPRXY | Source Active Memory Protection Proxy Register |
| 02A2 8258 | SACNTRLD | Source Active Count Reload Register |
| 02A2 825C | SASRCBREF | Source Active Source Address B-Reference Register |
| 02A2 8260 | SADSTBREF | Source Active Destination Address B-Reference Register |
| 02A2 8264 - 02A2 827C | - | Reserved |
| 02A2 8280 | DFCNTRLD | Destination FIFO Set Count Reload |
| 02A2 8284 | DFSRCBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A2 8288 | DFDSTBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A2 828C - 02A2 82FC | - | Reserved |
| 02A2 8300 | DFOPT0 | Destination FIFO Options Register 0 |
| 02A2 8304 | DFSRC0 | Destination FIFO Source Address Register 0 |
| 02A2 8308 | DFCNT0 | Destination FIFO Count Register 0 |
| 02A2 830C | DFDST0 | Destination FIFO Destination Address Register 0 |
| 02A2 8310 | DFBIDX0 | Destination FIFO BIDX Register 0 |
| 02A2 8314 | DFMPPRXY0 | Destination FIFO Memory Protection Proxy Register 0 |
| 02A2 8318 - 02A2 833C | - | Reserved |
| 02A2 8340 | DFOPT1 | Destination FIFO Options Register 1 |
| 02A2 8344 | DFSRC1 | Destination FIFO Source Address Register 1 |
| 02A2 8348 | DFCNT1 | Destination FIFO Count Register 1 |
| 02A2 834C | DFDST1 | Destination FIFO Destination Address Register 1 |
| 02A2 8350 | DFBIDX1 | Destination FIFO BIDX Register 1 |
| 02A2 8354 | DFMPPRXY1 | Destination FIFO Memory Protection Proxy Register 1 |
| 02A2 8358 - 02A2 837C | - | Reserved |
| 02A2 8380 | DFOPT2 | Destination FIFO Options Register 2 |
| 02A2 8384 | DFSRC2 | Destination FIFO Source Address Register 2 |
| 02A2 8388 | DFCNT2 | Destination FIFO Count Register 2 |
| 02A2 838C | DFDST2 | Destination FIFO Destination Address Register 2 |
| 02A2 8390 | DFBIDX2 | Destination FIFO BIDX Register 2 |

**Table 6-18. EDMA3 Transfer Controller 1 Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A2 8394 | DFMPPRXY2 | Destination FIFO Memory Protection Proxy Register 2 |
| 02A2 8398 - 02A2 83BC | - | Reserved |
| 02A2 83C0 | DFOPT3 | Destination FIFO Options Register 3 |
| 02A2 83C4 | DFSRC3 | Destination FIFO Source Address Register 3 |
| 02A2 83C8 | DFCNT3 | Destination FIFO Count Register 3 |
| 02A2 83CC | DFDST3 | Destination FIFO Destination Address Register 3 |
| 02A2 83D0 | DFBIDX3 | Destination FIFO BIDX Register 3 |
| 02A2 83D4 | DFMPPRXY3 | Destination FIFO Memory Protection Proxy Register 3 |
| 02A2 83D8 - 02A2 FFFF | - | Reserved |

**Table 6-19. EDMA3 Transfer Controller 2 Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A3 0000 | PID | Peripheral Identification Register |
| 02A3 0004 | TCCFG | EDMA3TC Configuration Register |
| 02A3 0008 - 02A3 00FC | - | Reserved |
| 02A3 0100 | TCSTAT | EDMA3TC Channel Status Register |
| 02A3 0104 - 02A3 011C | - | Reserved |
| 02A3 0120 | ERRSTAT | Error Register |
| 02A3 0124 | ERREN | Error Enable Register |
| 02A3 0128 | ERRCLR | Error Clear Register |
| 02A3 012C | ERRDET | Error Details Register |
| 02A3 0130 | ERRCMD | Error Interrupt Command Register |
| 02A3 0134 - 02A3 013C | - | Reserved |
| 02A3 0140 | RDRATE | Read Rate Register |
| 02A3 0144 - 02A3 023C | - | Reserved |
| 02A3 0240 | SAOPT | Source Active Options Register |
| 02A3 0244 | SASRC | Source Active Source Address Register |
| 02A3 0248 | SACNT | Source Active Count Register |
| 02A3 024C | SADST | Source Active Destination Address Register |
| 02A3 0250 | SABIDX | Source Active Source B-Index Register |
| 02A3 0254 | SAMPPRXY | Source Active Memory Protection Proxy Register |
| 02A3 0258 | SACNTRLD | Source Active Count Reload Register |
| 02A3 025C | SASRCBREF | Source Active Source Address B-Reference Register |
| 02A3 0260 | SADSTBREF | Source Active Destination Address B-Reference Register |
| 02A3 0264 - 02A3 027C | - | Reserved |
| 02A3 0280 | DFCNTRLD | Destination FIFO Set Count Reload |
| 02A3 0284 | DFSRCBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A3 0288 | DFDSTBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A3 028C - 02A3 02FC | - | Reserved |
| 02A3 0300 | DFOPT0 | Destination FIFO Options Register 0 |
| 02A3 0304 | DFSRC0 | Destination FIFO Source Address Register 0 |
| 02A3 0308 | DFCNT0 | Destination FIFO Count Register 0 |
| 02A3 030C | DFDST0 | Destination FIFO Destination Address Register 0 |
| 02A3 0310 | DFBIDX0 | Destination FIFO BIDX Register 0 |
| 02A3 0314 | DFMPPRXY0 | Destination FIFO Memory Protection Proxy Register 0 |
| 02A3 0318 - 02A3 033C | - | Reserved |
| 02A3 0340 | DFOPT1 | Destination FIFO Options Register 1 |

**Table 6-19. EDMA3 Transfer Controller 2 Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A3 0344 | DFSRC1 | Destination FIFO Source Address Register 1 |
| 02A3 0348 | DFCNT1 | Destination FIFO Count Register 1 |
| 02A3 034C | DFDST1 | Destination FIFO Destination Address Register 1 |
| 02A3 0350 | DFBIDX1 | Destination FIFO BIDX Register 1 |
| 02A3 0354 | DFMPPRXY1 | Destination FIFO Memory Protection Proxy Register 1 |
| 02A3 0358 - 02A3 037C | - | Reserved |
| 02A3 0380 | DFOPT2 | Destination FIFO Options Register 2 |
| 02A3 0384 | DFSRC2 | Destination FIFO Source Address Register 2 |
| 02A3 0388 | DFCNT2 | Destination FIFO Count Register 2 |
| 02A3 038C | DFDST2 | Destination FIFO Destination Address Register 2 |
| 02A3 0390 | DFBIDX2 | Destination FIFO BIDX Register 2 |
| 02A3 0394 | DFMPPRXY2 | Destination FIFO Memory Protection Proxy Register 2 |
| 02A3 0398 - 02A3 03BC | - | Reserved |
| 02A3 03C0 | DFOPT3 | Destination FIFO Options Register 3 |
| 02A3 03C4 | DFSRC3 | Destination FIFO Source Address Register 3 |
| 02A3 03C8 | DFCNT3 | Destination FIFO Count Register 3 |
| 02A3 03CC | DFDST3 | Destination FIFO Destination Address Register 3 |
| 02A3 03D0 | DFBIDX3 | Destination FIFO BIDX Register 3 |
| 02A3 03D4 | DFMPPRXY3 | Destination FIFO Memory Protection Proxy Register 3 |
| 02A3 03D8 - 02A3 7FFF | - | Reserved |

**Table 6-20. EDMA3 Transfer Controller 3 Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A3 8000 | PID | Peripheral Identification Register |
| 02A3 8004 | TCCFG | EDMA3TC Configuration Register |
| 02A3 8008 - 02A3 80FC | - | Reserved |
| 02A3 8100 | TCSTAT | EDMA3TC Channel Status Register |
| 02A3 8104 - 02A3 811C | - | Reserved |
| 02A3 8120 | ERRSTAT | Error Register |
| 02A3 8124 | ERREN | Error Enable Register |
| 02A3 8128 | ERRCLR | Error Clear Register |
| 02A3 812C | ERRDET | Error Details Register |
| 02A3 8130 | ERRCMD | Error Interrupt Command Register |
| 02A3 8134 - 02A3 813C | - | Reserved |
| 02A3 8140 | RDRATE | Read Rate Register |
| 02A3 8144 - 02A3 823C | - | Reserved |
| 02A3 8240 | SAOPT | Source Active Options Register |
| 02A3 8244 | SASRC | Source Active Source Address Register |
| 02A3 8248 | SACNT | Source Active Count Register |
| 02A3 824C | SADST | Source Active Destination Address Register |
| 02A3 8250 | SABIDX | Source Active Source B-Index Register |
| 02A3 8254 | SAMPPRXY | Source Active Memory Protection Proxy Register |
| 02A3 8258 | SACNTRLD | Source Active Count Reload Register |
| 02A3 825C | SASRCBREF | Source Active Source Address B-Reference Register |
| 02A3 8260 | SADSTBREF | Source Active Destination Address B-Reference Register |
| 02A3 8264 - 02A3 827C | - | Reserved |
| 02A3 8280 | DFCNTRLD | Destination FIFO Set Count Reload |

**Table 6-20. EDMA3 Transfer Controller 3 Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 02A3 8284 | DFSRCBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A3 8288 | DFDSTBREF | Destination FIFO Set Destination Address B Reference Register |
| 02A3 828C - 02A3 82FC | - | Reserved |
| 02A3 8300 | DFOPT0 | Destination FIFO Options Register 0 |
| 02A3 8304 | DFSRC0 | Destination FIFO Source Address Register 0 |
| 02A3 8308 | DFCNT0 | Destination FIFO Count Register 0 |
| 02A3 830C | DFDST0 | Destination FIFO Destination Address Register 0 |
| 02A3 8310 | DFBIDX0 | Destination FIFO BIDX Register 0 |
| 02A3 8314 | DFMPPRXY0 | Destination FIFO Memory Protection Proxy Register 0 |
| 02A3 8318 - 02A3 833C | - | Reserved |
| 02A3 8340 | DFOPT1 | Destination FIFO Options Register 1 |
| 02A3 8344 | DFSRC1 | Destination FIFO Source Address Register 1 |
| 02A3 8348 | DFCNT1 | Destination FIFO Count Register 1 |
| 02A3 834C | DFDST1 | Destination FIFO Destination Address Register 1 |
| 02A3 8350 | DFBIDX1 | Destination FIFO BIDX Register 1 |
| 02A3 8354 | DFMPPRXY1 | Destination FIFO Memory Protection Proxy Register 1 |
| 02A3 8358 - 02A3 837C | - | Reserved |
| 02A3 8380 | DFOPT2 | Destination FIFO Options Register 2 |
| 02A3 8384 | DFSRC2 | Destination FIFO Source Address Register 2 |
| 02A3 8388 | DFCNT2 | Destination FIFO Count Register 2 |
| 02A3 838C | DFDST2 | Destination FIFO Destination Address Register 2 |
| 02A3 8390 | DFBIDX2 | Destination FIFO BIDX Register 2 |
| 02A3 8394 | DFMPPRXY2 | Destination FIFO Memory Protection Proxy Register 2 |
| 02A3 8398 - 02A3 83BC | - | Reserved |
| 02A3 83C0 | DFOPT3 | Destination FIFO Options Register 3 |
| 02A3 83C4 | DFSRC3 | Destination FIFO Source Address Register 3 |
| 02A3 83C8 | DFCNT3 | Destination FIFO Count Register 3 |
| 02A3 83CC | DFDST3 | Destination FIFO Destination Address Register 3 |
| 02A3 83D0 | DFBIDX3 | Destination FIFO BIDX Register 3 |
| 02A3 83D4 | DFMPPRXY3 | Destination FIFO Memory Protection Proxy Register 3 |
| 02A3 83D8 - 02A3 FFFF | - | Reserved |

## 6.7 Reset Controller

The reset controller detects the different types of resets supported on the device and manages the distribution of those resets throughout the device.

The device has several types of resets: power-on reset, warm reset, max reset and system reset. Table 6-21 explains further the types of reset, the reset initiator, and the effects of each reset on the chip. See Section 6.7.9 for more information on the effects of each reset on the PLL controllers and their clocks.

**Table 6-21. Device-Level Reset Types**

| TYPE | INITIATOR | EFFECT(s) |
|---|---|---|
| Power-on Reset | $\overline{\text{POR}}$ pin | Resets the entire chip including the test and emulation logic |
| Warm Reset | $\overline{\text{RESET}}$ pin | Resets everything except for the test and emulation logic and the Ethernet Subsystem |
| Max Reset | Emulator | Same as a warm reset |
| System Reset | PCI via the $\overline{\text{PRST}}$ pin | A system reset maintains memory contents and does not reset the test and emulation circuit and the Ethernet Subsystem. The device configuration pins are also not re-latched and system reset does not affect the state of the peripherals (enable/disable). |

In addition to device-level global resets, the PSC provides the capability to cause local resets to peripherals and/or the CPU.

### *6.7.1 Power-on Reset ($\overline{\text{POR}}$ Pin)*

Power-on reset (POR) is initiated by the $\overline{\text{POR}}$ pin and is used to reset the entire chip, including the test and emulation logic. Power-on reset is also referred to as a cold reset since the device usually goes through a power-up cycle. During power-up, the $\overline{\text{POR}}$ pin must be asserted (driven low) until the power supplies have reached their normal operating conditions. Note that a device power-up cycle is not required to initiate a power-on reset.

The following sequence must be followed during a power-on reset:

1. Wait for all power supplies to reach normal operating conditions while keeping the $\overline{\text{POR}}$ pin asserted (driven low). While $\overline{\text{POR}}$ is asserted, all pins will be in high-impedance mode. After the $\overline{\text{POR}}$ pin is deasserted (driven high), all Z-group pins, low-group pins, and high-group pins are set to their reset state and will remain at their reset state until configured by their respective peripheral. The clock and reset of each peripheral is determined by the default settings of the power and sleep controller (PSC).
2. Once all the power supplies are within valid operating conditions, the $\overline{\text{POR}}$ pin must remain asserted (low) for a minimum number of 256 CLKIN2 cycles. The PLL1 controller input clock, CLKIN1, and the PCI input clock, PCLK, must be valid during this time. PCLK is needed only if the PCI module is being used. If the DDR2 memory controller and the Ethernet Subsystem are not needed, CLKIN2 can be tied low and REFCLKP/REFCLKN can be connected to $V_{SS}$ and $CV_{DD}$ respectively. In this case, the $\overline{\text{POR}}$ pin must remain asserted (low) for a minimum of 256 CLKIN1 cycles after all power supplies have reached valid operating conditions. Within the low period of the $\overline{\text{POR}}$ pin, the following occurs:
   (a) The reset signals flow to the entire chip (including the test and emulation logic), resetting modules that use reset asynchronously.
   (b) The PLL1 controller clocks are started at the frequency of the system reference clock. The clocks are propagated throughout the chip to reset modules that use reset synchronously. By default, PLL1 is in reset and unlocked.
   (c) The PLL2 controller clocks are started at the frequency of the system reference clock. PLL2 is held in reset. Since the PLL2 controller always operates in PLL mode, the system reference clock and all the system clocks are invalid at this point.
   (d) The $\overline{\text{RESETSTAT}}$ pin stays asserted (low), indicating the device is in reset.
3. The $\overline{\text{POR}}$ pin may now be deasserted (driven high). When the $\overline{\text{POR}}$ pin is deasserted, the configuration pin values are latched, and the PLL controllers change their system clocks to their default divide-down values. PLL2 is taken out of reset and automatically starts its locking sequence. Other

device initialization is also started.

4. After device initialization is complete, the RESETSTAT pin is deasserted (driven high). By this time, PLL2 has already completed its locking sequence and is outputting a valid clock. The system clocks of both PLL controllers are allowed to finish their current cycles and then paused for 10 cycles of their respective system reference clocks. After the pause, the system clocks are restarted at their default divide-by settings.

The device is now out of reset; device execution begins as dictated by the selected boot mode.

### 6.7.2 Warm Reset (RESET Pin)

A warm reset has the same effect as a power-on reset, except that in this case, the test and emulation logic are not reset.

The following sequence must be followed during a warm reset:

1. Hold the RESET pin low for a minimum of 24 CLKIN1 cycles. Within the low period of the RESET pin, the following occurs:
   (a) The Z-group pins, low-group pins, and the high-group pins are set to their reset state
   (b) The reset signals flow to the entire chip (excluding the test and emulation logic), resetting modules that use reset asynchronously
   (c) The PLL Controllers are reset. PLL1 switches back to PLL bypass mode, resetting all their registers to default values. Both PLL1 and PLL2 are placed in reset and lose lock. The PLL1 controller clocks start running at the frequency of the system reference clock. The clocks are propagated throughout the chip to reset modules that use reset synchronously.
   (d) The RESETSTAT pin becomes active (low), indicating the device is in reset.
2. The RESET pin may now be released (driven inactive high). When the RESET pin is released, the configuration pin values are latched and the PLL controllers immediately change their system clocks to their default divide-down values. Other device initialization is also started.

After device initialization is complete, the RESETSTAT pin goes inactive (high). All system clocks are allowed to finish their current cycles and then paused for 10 cycles of their respective system reference clocks. After the pause the system clocks are restarted at their default divide-by settings.

The clock and reset of each peripheral is determined by the default settings of the PSC.

The device is now out of reset, device execution begins as dictated by the selected boot mode.

### 6.7.3 Maximum Reset

A maximum (max) reset is initiated by the emulator. The effects are the same as a warm reset, except the device boot and configuration pins are not re-latched. The emulator initiates a maximum reset via the ICEPICK module. This ICEPICK initiated reset is nonmaskable.

The max reset sequence is as follows:

1. Max reset is initiated by the emulator. During this time, the following happens:
   (a) The reset signals flow to the entire chip, resetting all the modules on chip except the test and emulation logic.
   (b) The PLL controllers are reset, PLL1 switches back to PLL bypass mode, resetting all their registers to default values. Both PLL1 and PLL2 are placed in reset and lose lock.
   (c) The RESETSTAT pin becomes asserted (low), indicating the device is in reset.
2. After device initialization is complete, the PLL Controllers pause the system clocks for 10 cycles. At the end of these 10 cycles, the RESETSTAT pin is deasserted (driven high). At this point, the following occurs:
   (a) The I/O pins are controlled by the default peripherals (default peripherals are determined by PINMUX register).
   (b) The clock and reset of each peripheral is determined by the default settings of the power and sleep controller (PSC).

(c) The C64x+ begins executing from DSPBOOTADDR (determined by bootmode selection).

After the reset sequence, the boot sequence begins. Since the boot and configuration pins are not latched with a max reset, the previous values (as shown in the BOOTCFG register) are used to select the bootmode. After the boot sequence, follow the software initialization sequence.

### 6.7.4 System Reset

A system reset maintains memory contents and does not reset the clock logic or the test and emulation circuitry. The device configuration pins are also not re-latched and the state of the peripherals (enabled/disabled) is also not affected. A system reset is initiated by the $\overline{\text{PRST}}$ pin of PCI peripheral.

During a system reset, the following happens:

1. The $\overline{\text{RESETSTAT}}$ pin goes low to indicate an internal reset is being generated. The reset is allowed to propagate through the system. Internal system clocks are not affected.
2. After the internal reset signal has propagated, the PLL controllers pause and restart their system clocks for about 10 cycles of their system reference clocks, but retain their configuration. The PLLs also remain locked.
3. The boot sequence is started after the system clocks are restarted. Since the configuration pins (including the BOOTMODE[3:0] pins) are not latched with a system reset, the previous values, as shown in the BOOTCFG register, are used to select the boot mode.

### 6.7.5 Peripheral Local Reset

The user can configure the local reset and clock state of a peripheral through programming the PSC. Table 6-2 identifies the LPSC numbers and the peripherals capable of being locally reset by the PSC. For more detailed information on the programming of these peripherals by the PSC, see the *TMS320C6452 DSP Subsystem Reference Guide* (literature number SPRUFB1).

### 6.7.6 Reset Priority

If any of the above reset sources occur simultaneously, the PLLCTRL processes only the highest priority reset request. The reset request priorities are as follows (high to low):

- Power-on Reset
- Maximum Reset
- Warm Reset
- System Reset

### 6.7.7 Reset Controller Register

The reset type status (RSTYPE) register is the only register for the reset controller.

The RSTYPE register latches the cause of the last reset. If multiple reset sources occur simultaneously, this register latches the highest priority reset source. The reset type status register is shown in Figure 6-9 and described in Table 6-22.

**Figure 6-9. Reset Type Status Register (RSTYPE)**

| 31 | | | | | 16 |
|---|---|---|---|---|---|
| Reserved | | | | | |
| R-0 | | | | | |

| 15 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|
| Reserved | | SRST | MRST | WRST | POR |
| R-0 | | R-0 | R-0 | R-0 | R-0 |

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

**Table 6-22. Reset Type Status Register (RSTYPE) Field Descriptions**

| Bit | Field | Value | Description |
|---|---|---|---|
| 31:4 | Reserved | | Reserved. The reserved bit location is always read as 0. A value written to this field has no effect. |
| 3 | SRST | | System reset |
| | | 0 | System Reset was not the last reset to occur. |
| | | 1 | System Reset was the last reset to occur. |
| 2 | MRST | | Max reset |
| | | 0 | Max Reset was not the last reset to occur. |
| | | 1 | Max Reset was the last reset to occur. |
| 1 | WRST | | Warm reset |
| | | 0 | Warm Reset was not the last reset to occur. |
| | | 1 | Warm Reset was the last reset to occur. |
| 0 | POR | | Power-on reset |
| | | 0 | Power-on Reset was not the last reset to occur. |
| | | 1 | Power-on Reset was the last reset to occur. |

### 6.7.8 Pin Behaviors at Reset

During normal operation, devices pins are controlled by the selected peripheral. During device level global reset, the pin behaviors are classified into the following Reset Groups:

- **Z Group:** These pins are 3-stated when a device-level global reset source (e.g., $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins remain 3-stated until configured otherwise by their respective peripheral (after the peripheral is enabled by the PSC).
- **Z/High Group:** These pins are 3-stated when a device-level global reset source (e.g., $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins drive a logic High.
- **Z/Low Group:** These pins are 3-stated when a device-level global reset source (e.g. $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins drive a logic Low.
- **DDR2 Z/High Group:** These pins are 3-stated when a device-level global reset source (e.g. $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins are Driven High.
- **DDR2 Low/High Group:** These pins are driven Low when a device-level global reset source (e.g. $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins are Driven High.
- **DDR2 High/Low Group:** These pins are driven High when a device-level global reset source (e.g. $\overline{\text{POR}}$, $\overline{\text{RESET}}$, or Max Reset) is asserted. When the reset source is de-asserted, these pins are Driven Low.

- **Clock Group:** These clock pins are toggling by default. They pause momentarily before $\overline{\text{RESETSTAT}}$ is de-asserted (high).

Table 6-23 lists the Reset Group for each pin.

**Table 6-23. Pin Behaviors at Reset**

| Pin Name | Reset Group |
|---|---|
| CLKIN1 | Z Group |
| CLKIN2 | Z Group |
| REFCLKN | Z Group |
| REFCLKP | Z Group |
| SYSCLK5 | Z Group |
| TCLK | Z Group |
| TDI | Z Group |
| TDO | Z Group |
| TMS | Z Group |
| $\overline{\text{TRST}}$ | Z Group |
| EMU0 | Z Group |
| EMU1 | Z Group |
| EMU2 | Z Group |
| EMU3 | Z Group |
| EMU4 | Z Group |
| EMU5 | Z Group |
| EMU6 | Z Group |
| EMU7 | Z Group |
| EMU8 | Z Group |
| EMU9 | Z Group |
| EMU10 | Z Group |
| EMU11 | Z Group |
| NMI | Z Group |
| $\overline{\text{RESET}}$ | Z Group |
| $\overline{\text{POR}}$ | Z Group |
| AD00/HD00 | Z Group |
| AD01/HD01 | Z Group |
| AD02/HD02 | Z Group |
| AD03/HD03 | Z Group |
| AD04/HD04 | Z Group |
| AD05/HD05 | Z Group |
| AD06/HD06 | Z Group |
| AD07/HD07 | Z Group |
| AD08/HD08 | Z Group |
| AD09/HD09 | Z Group |
| AD10/HA10 | Z Group |
| AD11/HD11 | Z Group |
| AD12/HD12 | Z Group |
| AD13/HD13 | Z Group |
| AD14/HD14 | Z Group |
| AD15/HD15 | Z Group |
| AD16/HD16 | Z Group |
| AD17/HD17 | Z Group |

**Table 6-23. Pin Behaviors at Reset (continued)**

| Pin Name | Reset Group |
|---|---|
| AD18/HS18 | Z Group |
| AD19/HD19 | Z Group |
| AD20/HD20 | Z Group |
| AD21/HD21 | Z Group |
| AD22/HD22 | Z Group |
| AD23/HD23 | Z Group |
| AD24/HD24 | Z Group |
| AD25/HS25 | Z Group |
| AD26/HD26 | Z Group |
| AD27/AD27 | Z Group |
| AD28/HD28 | Z Group |
| AD29/HD29 | Z Group |
| AD30/HD30 | Z Group |
| AD31/HD31 | Z Group |
| PPAR/$\overline{\text{HAS}}$ | Z Group |
| $\overline{\text{PSTOP}}$/HCNTL0 | Z Group |
| $\overline{\text{PDEVSEL}}$/HCNTL1 | Z Group |
| $\overline{\text{PPERR}}$/$\overline{\text{HCS}}$ | Z Group |
| $\overline{\text{PSERR}}$/$\overline{\text{HDS1}}$ | Z Group |
| $\overline{\text{PCBE0}}$/GP04 | Z Group |
| $\overline{\text{PCBE2}}$/HR/$\overline{\text{W}}$ | Z Group |
| $\overline{\text{PCBE3}}$/GP07 | Z Group |
| PCLK/HHWIL | Z Group |
| $\overline{\text{PGNT}}$/GPO0 | Z Group |
| $\overline{\text{PRST}}$/GPO1 | Z Group |
| $\overline{\text{PINTA}}$/GPO2 | Z Group |
| $\overline{\text{PREQ}}$/GPO3 | Z Group |
| $\overline{\text{PTRDY}}$/GPO5 | Z Group |
| PIDSEL/GPO6 | Z Group |
| DEVICEENABLE0/AEA20 | Z Group |
| EMIBWIDTH/AEA22 | Z Group |
| FASTBOOT/AEA21 | Z Group |
| UHPIEN | Z Group |
| HPIWIDTH/AEA16 | Z Group |
| RSV_BOOT/AEA15 | Z Group |
| PCI66/AEA18 | Z Group |
| BOOTMODE0/AEA11 | Z Group |
| BOOTMODE1/AEA12 | Z Group |
| BOOTMODE2/AEA13 | Z Group |
| BOOTMODE3/AEA14 | Z Group |
| SCL | Z Group |
| SDA | Z Group |
| SGMII0RXN | Z Group |
| SGMII0RXP | Z Group |
| SGMII1RXN | Z Group |
| SGMII1RXP | Z Group |
| SGMII0TXN | Z Group |

**Table 6-23. Pin Behaviors at Reset (continued)**

| Pin Name | Reset Group |
|---|---|
| SGMII0TXP | Z Group |
| SGMII1TXN | Z Group |
| SGMII1TXP | Z Group |
| MDIO | Z Group |
| SPICLK | Z Group |
| $\overline{\text{SPICS1}}$/UARTTX | Z Group |
| $\overline{\text{SPICS2}}$/UARTRX | Z Group |
| SPIDI/UARTRTS | Z Group |
| SPIDO/UARTCTS | Z Group |
| T0INP12/GP08 | Z Group |
| T0OUT12/GP09 | Z Group |
| T1INPL/GP10 | Z Group |
| T1OUT12/GP11 | Z Group |
| AHCLKR | Z Group |
| ALHCLKX | Z Group |
| ACLKR | Z Group |
| ACLKX | Z Group |
| AFSR | Z Group |
| AFSX | Z Group |
| AXR0/TSIP0TR0 | Z Group |
| AXR1/TSIP0TR1 | Z Group |
| AXR2/TSIP0TX0 | Z Group |
| AXR3/TSIP0TX1 | Z Group |
| AXR4/TSIP1TR0 | Z Group |
| AXR5/TSIP1TR1 | Z Group |
| AXR6/TSIP1TX0 | Z Group |
| AXR7/TSIP1TX1 | Z Group |
| AMUTEIN/TISP1FSA | Z Group |
| AMUTE/TSIP1FSB | Z Group |
| GP16 | Z Group |
| GP17 | Z Group |
| GP18 | Z Group |
| GP19 | Z Group |
| GP20 | Z Group |
| GP21 | Z Group |
| GP22 | Z Group |
| GP23 | Z Group |
| GP24 | Z Group |
| GP25 | Z Group |
| GP26 | Z Group |
| GP27 | Z Group |
| GP28 | Z Group |
| GP29 | Z Group |
| GP30 | Z Group |
| GP31 | Z Group |
| AECLKOUT | Z Group |
| $\overline{\text{AAWE}}$/$\overline{\text{ASWE}}$ | Z Group |

**Table 6-23. Pin Behaviors at Reset (continued)**

| Pin Name | Reset Group |
|---|---|
| AR/$\overline{W}$ | Z Group |
| $\overline{AAOE}$/$\overline{ASOE}$ | Z Group |
| AED00 | Z Group |
| AED01 | Z Group |
| AED02 | Z Group |
| AED03 | Z Group |
| AED04 | Z Group |
| AED05 | Z Group |
| AED06 | Z Group |
| AED07 | Z Group |
| AED08 | Z Group |
| AED09 | Z Group |
| AED10 | Z Group |
| AED11 | Z Group |
| AED12 | Z Group |
| AED13 | Z Group |
| AED14 | Z Group |
| AED15 | Z Group |
| ABA0 | Z Group |
| ABA1 | Z Group |
| $\overline{ASADS}$/$\overline{ASRE}$ | Z Group |
| $\overline{ABE00}$ | Z Group |
| $\overline{ABE01}$ | Z Group |
| AEA10 | Z Group |
| $\overline{ACE2}$ | Z Group |
| $\overline{ACE3}$ | Z Group |
| AEA00 | Z Group |
| AEA01 | Z Group |
| AEA02 | Z Group |
| AEA03 | Z Group |
| AEA04 | Z Group |
| AEA05 | Z Group |
| AEA06 | Z Group |
| AEA07 | Z Group |
| AEA08 | Z Group |
| AEA09 | Z Group |
| AEA23 | Z Group |
| AEA19 | Z Group |
| AEA17 | Z Group |
| MDCLK | Z/High Group |
| AARDY | Z/High Group |
| $\overline{PFRAME}$/$\overline{HINT}$ | Z/High Group |
| $\overline{PIRDY}$/$\overline{HRDY}$ | Z/Low Group |
| VTXD0 | Z/Low Group |
| VTXD1 | Z/Low Group |
| VTXD2 | Z/Low Group |
| VTXD3 | Z/Low Group |

**Table 6-23. Pin Behaviors at Reset (continued)**

| Pin Name | Reset Group |
|---|---|
| DDR_D00 | DDR2 Z Group |
| DDR_D01 | DDR2 Z Group |
| DDR_D02 | DDR2 Z Group |
| DDR_D03 | DDR2 Z Group |
| DDR_D04 | DDR2 Z Group |
| DDR_D05 | DDR2 Z Group |
| DDR_D06 | DDR2 Z Group |
| DDR_D07 | DDR2 Z Group |
| DDR_D08 | DDR2 Z Group |
| DDR_D09 | DDR2 Z Group |
| DDR_D10 | DDR2 Z Group |
| DDR_D11 | DDR2 Z Group |
| DDR_D12 | DDR2 Z Group |
| DDR_D13 | DDR2 Z Group |
| DDR_D14 | DDR2 Z Group |
| DDR_D15 | DDR2 Z Group |
| DDR_D16 | DDR2 Z Group |
| DDR_D17 | DDR2 Z Group |
| DDR_D18 | DDR2 Z Group |
| DDR_D19 | DDR2 Z Group |
| DDR_D20 | DDR2 Z Group |
| DDR_D21 | DDR2 Z Group |
| DDR_D22 | DDR2 Z Group |
| DDR_D23 | DDR2 Z Group |
| DDR_D24 | DDR2 Z Group |
| DDR_D25 | DDR2 Z Group |
| DDR_D26 | DDR2 Z Group |
| DDR_D27 | DDR2 Z Group |
| DDR_D28 | DDR2 Z Group |
| DDR_D29 | DDR2 Z Group |
| DDR_D30 | DDR2 Z Group |
| DDR_D31 | DDR2 Z Group |
| DDR_DQGATE1 | DDR2 Z Group |
| DDR_DQGATE3 | DDR2 Z Group |
| DDR_DQS[0]P | DDR2 Z Group |
| DDR_DQS[1]P | DDR2 Z Group |
| DDR_DQS[2]P | DDR2 Z Group |
| DDR_DQS[3]P | DDR2 Z Group |
| DDR_DQS[0]N | DDR2 Z Group |
| DDR_DQS[1]N | DDR2 Z Group |
| DDR_DQS[2]N | DDR2 Z Group |
| DDR_DQS[3]N | DDR2 Z Group |
| DDR_DQM[0] | DDR2 Z/High Group |
| DDR_DQM[1] | DDR2 Z/High Group |
| DDR_DQM[2] | DDR2 Z/High Group |
| DDR_DQM[3] | DDR2 Z/High Group |
| $\overline{\text{DDR\_CAS}}$ | DDR2 High Group |

**Table 6-23. Pin Behaviors at Reset (continued)**

| Pin Name | Reset Group |
|---|---|
| $\overline{\text{DDR\_RAS}}$ | DDR2 High Group |
| $\overline{\text{DDR\_WE}}$ | DDR2 High Group |
| DDR_A00 | DDR2 Low Group |
| DDR_A01 | DDR2 Low Group |
| DDR_A02 | DDR2 Low Group |
| DDR_A03 | DDR2 Low Group |
| DDR_A04 | DDR2 Low Group |
| DDR_A05 | DDR2 Low Group |
| DDR_A06 | DDR2 Low Group |
| DDR_A07 | DDR2 Low Group |
| DDR_A08 | DDR2 Low Group |
| DDR_A09 | DDR2 Low Group |
| DDR_A10 | DDR2 Low Group |
| DDR_A11 | DDR2 Low Group |
| DDR_A12 | DDR2 Low Group |
| DDR_A13 | DDR2 Low Group |
| DDR_ODT0 | DDR2 Low Group |
| DDR_ODT1 | DDR2 Low Group |
| DDR_CKE | DDR2 Low Group |
| DDR_DQGATE0 | DDR2 Low Group |
| DDR_DQGATE2 | DDR2 Low Group |
| DDR_BA[0] | DDR2 Low/High Group |
| DDR_BA[1] | DDR2 Low/High Group |
| DDR_BA[2] | DDR2 Low/High Group |
| $\overline{\text{DDR\_CS}}$ | DDR2 High/Low Group |
| DDR_CLKP | Clock |
| DDR_CLKN | Clock |
| AECLKIN | Clock |
| $\overline{\text{RESETSTAT}}$ | Reflects $\overline{\text{POR}}$ or $\overline{\text{RESET}}$ value |

### 6.7.9 Reset Electrical Data/Timing

**NOTE**

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

**Table 6-24. Timing Requirements for Reset(1) (2) (see Figure 6-10)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 5 | $t_{w(POR)}$ | Pulse duration, $\overline{\text{POR}}$ low(3) | 256D | | ns |
| 6 | $t_{w(RESET)}$ | Pulse duration, $\overline{\text{RESET}}$ low | 24C | | ns |
| 7 | $t_{su(boot)}$ | Setup time, boot mode and configuration pins valid before $\overline{\text{POR}}$ high or $\overline{\text{RESET}}$ high (4) | 6P | | ns |

(1) C = 1/CLKIN1 clock frequency in ns
(2) D = 1/CLKIN2 clock frequency in ns
(3) If CLKIN2 is not used, $t_{w(POR)}$ must be measured in terms of CLKIN1 cycles; otherwise, use the slower of the CLKIN1,CLKIN2 cycles.
(4) P = 1/CPU clock frequency in nanoseconds

**Table 6-24. Timing Requirements for Reset(1) (2) (see Figure 6-10) (continued)**

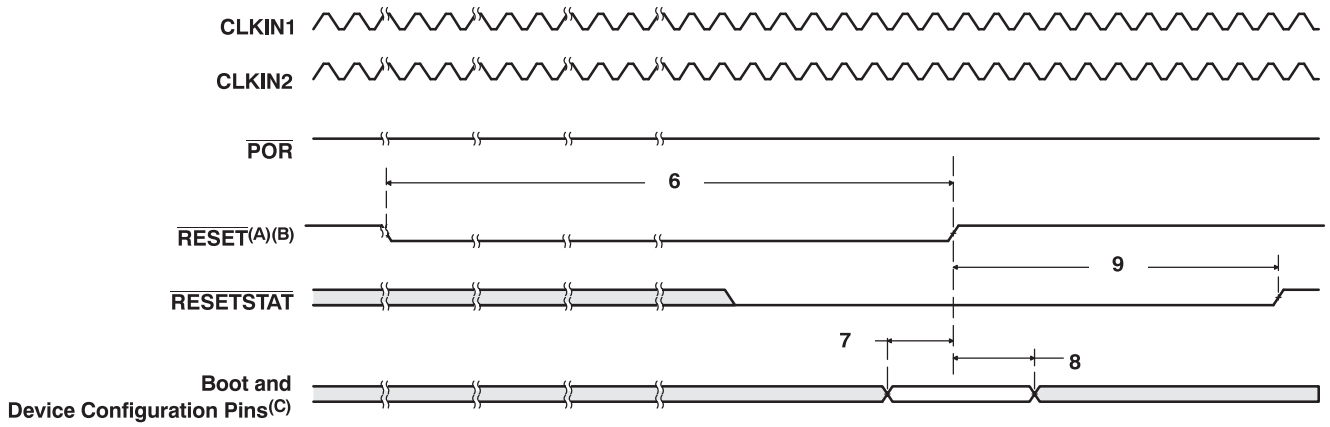
| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 8 | $t_{h(boot)}$ | Hold time, boot mode and configuration pins valid after $\overline{POR}$ high or $\overline{RESET}$ high (4) | 6P | | ns |

**Table 6-25. Switching Characteristics Over Recommended Operating Conditions During Reset(1) (see Figure 6-10)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 9 | $t_{d(PORH-RSTATH)}$ | Delay time, $\overline{POR}$ high AND $\overline{RESET}$ high to $\overline{RESETSTAT}$ high | | 1500C | ns |

(1) C = 1/CLKIN1 clock frequency in ns.

- Z group consists of: all I/O/Z and O/Z pins, except for Low and High group pins. Pins become high impedance as soon as their respective power supply has reached normal operating conditions. Pins remain in high impedance until configured otherwise by their respective peripherals.
- Low group consists of pins that become low as soon as their respective power supply has reached normal operating conditions. Pins remain low until configured otherwise by their respective peripheral.
- High group consists of pins that become high as soon as their respective power supply has reached normal operating conditions. Pins remain high until configured otherwise by their respective peripheral.
- All peripherals must be enabled through software following a power-on reset; for more details, see Section 6.7.1, *Power-on Reset*.
- For power-supply sequence requirements, see Section 6.3.1.




A. $\overline{RESET}$ should be used only after the device has been powered up. For more details on the use of the $\overline{RESET}$ pin, see Section 6.7, *Reset Controller*.

B. A reset signal is generated internally during a Warm Reset. This internal reset signal has the same effect as the $\overline{RESET}$ pin during a Warm Reset.

C. Boot and Device Configuration Inputs (during reset) include AEA[22:11], and UHPIEN.

**Figure 6-10. Warm Reset and Max Reset Timing**

## 6.8 Interrupts

The C64x+ DSP interrupt controller combines device events into 12 prioritized interrupts. The source for each of the 12 CPU interrupts is user programmable. Also, the interrupt controller controls the generation of the CPU exception, NMI, and emulation interrupts and the generation of AEG events. Table 6-27 summarizes the C64x+ interrupt controller registers and memory locations. For more details on DSP interrupt control, see *TMS320C6452 DSP Subsystem Reference Guide* (literature number SPRUFB1).

**Table 6-26. DSP Interrupt Events**

| DSP INTERRUPT EVENT NUMBER | EVENT | INTERRUPT SOURCE |
|---|---|---|
| 0 | EVT0 | Output of event combiner 0, for events 1 – 31 |
| 1 | EVT1 | Output of event combiner 1, for events 32 – 63 |
| 2 | EVT2 | Output of event combiner 2, for events 64 – 95 |
| 3 | EVT33 | Output of event combiner 3, for events 96 – 127 |
| 4-8 | | Reserved |
| 9 | EMU_DTDMA | ECM interrupt for:<br>• Host scan access event<br>• DTDMA transfer complete event<br>• AET interrupt event |
| 10 | Reserved | Reserved |
| 11 | EMU_RTDXRX | RTDX receive complete event |
| 12 | EMU_RTDXTX | RTDX transmit complete event |
| 13 | IDMA0 EMC | C64x+ EMC 0 event |
| 14 | IDMA1 EMC | C64x+ EMC 1 event |
| 15 | DSPINT | Host (PCI/HPI) to DSP interrupt event |
| 16 | I2CINT | I2C interrupt event |
| 17 | Reserved | Reserved |
| 18 | AEASYNCERR event | EMIFA Error Interrupt event |
| 19 | TINT2L | Timer interrupt low event |
| 20 | TINT2H | Timer interrupt high event |
| 21 | TINT3L | Timer interrupt low event |
| 22 | TINT3H | Timer interrupt high event |
| 23 | PSCINT | PSC-ALLINT event |
| 24 | TPCC_GINT | EDMA3 channel global completion interrupt event |
| 25 | SPIINT0 | SPI Interrupt |
| 26 | SPIINT1 | SPI Interrupt |
| 27 | Reserved | Reserved |
| 28 | Reserved | Reserved |
| 29 | Reserved | Reserved |
| 30 -31 | Reserved | Reserved |
| 32 | RX_PULSE | Ethernet Subsystem RX pulse interrupt event |
| 33 | RX_THRESH_PULSE | Ethernet Subsystem RX threshold interrupt event |
| 34 | TX_PULSE | Ethernet Subsystem TX pulse interrupt event |
| 35 | MISC_PULSE | Ethernet Subsystem MISC pulse interrupt event |
| 36 | UART_INT | UART Interrupt |
| 37 | Reserved | Reserved |
| 38 | Reserved | Reserved |
| 39 | Reserved | Reserved |
| 40 | Reserved | Reserved |

**Table 6-26. DSP Interrupt Events (continued)**

| DSP INTERRUPT EVENT NUMBER | EVENT | INTERRUPT SOURCE |
|---|---|---|
| 41 | Reserved | Reserved |
| 42 | GPIO_BNK1_INT | (GPIO16:31) GPIO Bank 1 Interrupt event |
| 43 | AXINT | TX Interrupt McASP |
| 44 | ARINT | RX Interrupt McASP |
| 45-49 | | Reserved |
| 50 | VINT | VLYNQ Pulse Interrupt event |
| 51 | GPINT0 | GPIO Interrupt event |
| 52 | GPINT1 | GPIO Interrupt event |
| 53 | GPINT2 | GPIO Interrupt event |
| 54 | GPINT3 | GPIO Interrupt event |
| 55 | GPINT4 | GPIO Interrupt event |
| 56 | GPINT5 | GPIO Interrupt event |
| 57 | GPINT6 | GPIO Interrupt event |
| 58 | GPINT7 | GPIO Interrupt event |
| 59 | GPINT8 | GPIO Interrupt event |
| 60 | GPINT9 | GPIO Interrupt event |
| 61 | GPINT10 | GPIO Interrupt event |
| 62 | GPINT11 | GPIO Interrupt event |
| 63 | GPINT12 | GPIO Interrupt |
| 64 | GPINT13 | GPIO Interrupt |
| 65 | GPINT14 | GPIO Interrupt event |
| 66 | GPINT15 | GPIO Interrupt event |
| 67 | TINT0L | Timer interrupt low event |
| 68 | TINT0H | Timer interrupt high event |
| 69 | TINT1L | Timer interrupt low event |
| 70 | TINT1H | Timer interrupt high event |
| 71 | EDMA3CC_INT0 | EDMA3CC Completion Interrupt - Mask0 event |
| 72 | EDMA3CC_INT1 | EDMA3CC Completion Interrupt – Mask1 event |
| 73 | EDMA3CC_INT2 | EDMA3CC Completion Interrupt – Mask2 event |
| 74 | EDMA3CC_INT3 | EDMA3CC Completion Interrupt – Mask3 event |
| 75 | EDMA3CC_INT4 | EDMA3CC Completion Interrupt – Mask4 event |
| 76 | EDMA3CC_INT5 | EDMA3CC Completion Interrupt – Mask5 event |
| 77 | EDMA3CC_INT6 | EDMA3CC Completion Interrupt – Mask6 event |
| 78 | EDMA3CC_INT7 | EDMA3CC Completion Interrupt – Mask7 event |
| 79 | EDMA3CC_ERRINT | EDMA3CC Error Interrupt event |
| 80 | EDMA3CC_MPINT | EDMA3CC Memory Protection Interrupt event |
| 81 | EDMA3TC0_ERRINT | EDMA3TC0 Error Interrupt event |
| 82 | EDMA3TC1_ERRINT | EDMA3TC1 Error Interrupt event |
| 83 | EDMA3TC2_ERRINT | EDMA3TC2 Error Interrupt event |
| 84 | EDMA3TC3_ERRINT | EDMA3TC3 Error Interrupt event |
| 85 | Reserved | Reserved |
| 86 | Reserved | Reserved |
| 87 | Reserved | Reserved |
| 88 | Reserved | Reserved |
| 89 | Reserved | Reserved |
| 90 | Reserved | Reserved |

**Table 6-26. DSP Interrupt Events (continued)**

| DSP INTERRUPT EVENT NUMBER | EVENT | INTERRUPT SOURCE |
|---|---|---|
| 91 | Reserved | Reserved |
| 92 | Reserved | Reserved |
| 93 | Reserved | Reserved |
| 94 | Reserved | Reserved |
| 95 | Reserved | Reserved |
| 96 | INTERR | C64x+ Interrupt Controller Dropped CPU Interrupt Event |
| 97 | EMC_IDMAERR | C64x+ EMC Invalid IDMA Parameters event |
| 98 | Reserved | Reserved |
| 99 | Reserved | Reserved |
| 100 | EFIINTA | EFI Interrupt from side A event |
| 101 | EFIINTB | EFI Interrupt from side B event |
| 102 - 112 | Reserved | Reserved |
| 113 | L1P_ED | L1P Single bit error detected during DMA read event |
| 114-115 | Reserved | Reserved |
| 116 | L2_ED1 | L2 single bit error detected event |
| 117 | L2_ED2 | L2 two bit error detected event |
| 118 | PDC_INT | Power Down sleep interrupt event |
| 119 | Reserved | Reserved |
| 120 | L1P_CMPA | L1P CPU memory protection fault event |
| 121 | L1P_DMPA | L1P DMA memory protection fault event |
| 122 | L1D_CMPA | L1D CPU memory protection fault event |
| 123 | L1D_DMPA | L1D DMA memory protection fault event |
| 124 | L2_CMPA | L2 CPU memory protection fault event |
| 125 | L2_DMPA | L2 DMA memory protection fault event |
| 126 | IDMA_CMPA | IDMA CPU memory protection fault event |
| 127 | IDMA_BUSERR | IDMA bus error interrupt event |

**Table 6-27. C64x+ Interrupt Controller Registers**

| HEX ADDRESS | ACRONYM | REGISTER DESCRIPTION |
|---|---|---|
| 0x0180 0000 | EVTFLAG0 | Event flag register 0 |
| 0x0180 0004 | EVTFLAG1 | Event flag register 1 |
| 0x0180 0008 | EVTFLAG2 | Event flag register 2 |
| 0x0180 000C | EVTFLAG3 | Event flag register 3 |
| 0x0180 0020 | EVTSET0 | Event set register 0 |
| 0x0180 0024 | EVTSET1 | Event set register 1 |
| 0x0180 0028 | EVTSET2 | Event set register 2 |
| 0x0180 002C | EVTSET3 | Event set register 3 |
| 0x0180 0040 | EVTCLR0 | Event clear register 0 |
| 0x0180 0044 | EVTCLR1 | Event clear register 1 |
| 0x0180 0048 | EVTCLR2 | Event clear register 2 |
| 0x0180 004C | EVTCLR3 | Event clear register 3 |
| 0x0180 0080 | EVTMASK0 | Event mask register 0 |
| 0x0180 0084 | EVTMASK1 | Event mask register 1 |
| 0x0180 0088 | EVTMASK2 | Event mask register 2 |
| 0x0180 008C | EVTMASK3 | Event mask register 3 |
| 0x0180 00A0 | MEVTFLAG0 | Masked event flag register 0 |
| 0x0180 00A4 | MEVTFLAG1 | Masked event flag register 1 |
| 0x0180 00A8 | MEVTFLAG2 | Masked event flag register 2 |
| 0x0180 00AC | MEVTFLAG3 | Masked event flag register 3 |
| 0x0180 00C0 | EXPMASK0 | Exception mask register 0 |
| 0x0180 00C4 | EXPMASK1 | Exception mask register 1 |
| 0x0180 00C8 | EXPMASK2 | Exception mask register 2 |
| 0x0180 00CC | EXPMASK3 | Exception mask register 3 |
| 0x0180 00E0 | MEXPFLAG0 | Masked exception flag register 0 |
| 0x0180 00E4 | MEXPFLAG1 | Masked exception flag register 1 |
| 0x0180 00E8 | MEXPFLAG2 | Masked exception flag register 2 |
| 0x0180 00EC | MEXPFLAG3 | Masked exception flag register 3 |
| 0x0180 0104 | INTMUX1 | Interrupt mux register 1 |
| 0x0180 0108 | INTMUX2 | Interrupt mux register 2 |
| 0x0180 010C | INTMUX3 | Interrupt mux register 3 |
| 0x0180 0140 | AEGMUX0 | Advanced event generator mux register 0 |
| 0x0180 0144 | AEGMUX1 | Advanced event generator mux register 1 |
| 0x0180 0180 | INTXSTAT | Interrupt exception status |
| 0x0180 0184 | INTXCLR | Interrupt exception clear |
| 0x0180 0188 | INTDMASK | Dropped interrupt mask register |
| 0x0180 01C0 | EVTASRT | Event assert register |

## 6.9 DDR2 Memory Controller

The 32-bit DDR2 memory controller bus of the device is used to interface to JESD79D-2A standard-compliant DDR2 SDRAM devices. The DDR2 external bus interfaces only to DDR2 SDRAM devices; it does not share the bus with any other types of peripherals. The decoupling of DDR2 memories from other devices simplifies board design and provides I/O concurrency from a second external memory interface, EMIFA.

The internal data bus clock frequency and DDR2 bus clock frequency directly affect the maximum throughput of the DDR2 bus. The data rate of the DDR2 bus is equal to the CLKIN2 frequency multiplied by 20. The internal data bus clock frequency of the DDR2 memory controller is fixed at a divide-by-three ratio of the CPU frequency. The maximum DDR2 throughput is determined by the smaller of the two bus frequencies. For example, if the internal data bus frequency is 300 MHz (CPU frequency is 900 MHz) and the DDR2 data rate is 533 MHz (266 MHz clock rate as CLKIN2 frequency is 26.6 MHz), the maximum data rate achievable by the DDR2 memory controller is 2.13 Gbytes/sec.

### *6.9.1 DDR2 Memory Controller Device-Specific Information*

The approach to specifying interface timing for the DDR2 memory bus is different than on other interfaces such as EMIF and HPI. For these other interfaces, the device timing was specified in terms of data manual specifications and I/O buffer information specification (IBIS) models.

For the DDR2 memory bus, the approach is to specify compatible DDR2 devices and provide the printed circuit board (PCB) solution and guidelines directly to the user. Texas Instruments (TI) has performed the simulation and system characterization to be sure all DDR2 interface timings in this solution are met.

The ODT[1:0] pins of the memory controller must be left unconnected. The ODT pins on the DDR2 memory device(s) must be connected to ground.

The DDR2 memory controller on the device supports the following memory topologies:

- A 32-bit wide configuration interfacing to two 16-bit wide DDR2 SDRAM devices.
- A 16-bit wide configuration interfacing to a single 16-bit wide DDR2 SDRAM device.

A race condition may exist when certain masters write data to the DDR2 memory controller. For example, if master A passes a software message via a buffer in external memory and does not wait for indication that the write completes, when master B attempts to read the software message, then the master B read may bypass the master A write and, thus, master B may read stale data and, therefore, receive an incorrect message.

Some master peripherals (e.g., EDMA3 transfer controllers) will always wait for the write to complete before signaling an interrupt to the system, thus avoiding this race condition. For masters that do not have hardware specification of write-read ordering, it may be necessary to specify data ordering via software.

If master A does not wait for indication that a write is complete, it must perform the following workaround:

1. Perform the required write.
2. Perform a dummy write to the DDR2 memory controller module ID and revision register.
3. Perform a dummy read to the DDR2 memory controller module ID and revision register.
4. Indicate to master B that the data is ready to be read after completion of the read in step 3. The completion of the read in step 3 ensures that the previous write was done.

The master peripherals that need to implement this workaround are HPI, PCI, and VLYNQ.

### 6.9.2 DDR2 Memory Controller Peripheral Registers

**Table 6-28. DDR2 Memory Controller Registers(1)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x7800 0000 | MIDR | DDR2 Memory Controller Module and Revision Register |
| 0x7800 0004 | DMCSTAT | DDR2 Memory Controller Status Register |
| 0x7800 0008 | SDCFG | DDR2 Memory Controller SDRAM Configuration Register |
| 0x7800 000C | SDRFC | DDR2 Memory Controller SDRAM Refresh Control Register |
| 0x7800 0010 | SDTIM1 | DDR2 Memory Controller SDRAM Timing 1 Register |
| 0x7800 0014 | SDTIM2 | DDR2 Memory Controller SDRAM Timing 2 Register |
| 0x7800 0018 | - | Reserved |
| 0x7800 0020 | BPRIO | DDR2 Memory Controller Burst Priority Register |
| 0x7800 0024 - 0x7800 004C | - | Reserved |
| 0x7800 0050 - 0x7800 0078 | - | Reserved |
| 0x7800 007C - 0x7800 00BC | - | Reserved |
| 0x7800 00C0 - 0x7800 00E0 | - | Reserved |
| 0x7800 00E4 | DMCCTL | DDR2 Memory Controller Control Register |
| 0x7800 00E8 - 0x7800 00FC | - | Reserved |
| 0x7800 0100 - 0x7FFF FFFF | - | Reserved |

(1) For details about the DDR2 registers and their modes, see the *C6452 DSP DDR2 Memory Controller (DDR2) User's Guide* (literature number SPRUF85.

### 6.9.3 DDR2 Interface

This section provides the timing information for the DDR2 interface as a PCB design and manufacturing specification. The design rules constrain PCB trace length, PCB trace skew, signal integrity, cross-talk, and signal timing. These rules, when followed, result in a reliable DDR2 memory system without the need for a complex timing closure process. For more information regarding guidelines for using this DDR2 specification, *Understanding TI's PCB Routing Rule-Based DDR2 Timing Specification* (SPRAAV0).

#### 6.9.3.1 DDR2 Interface Schematic

Figure 6-11 shows the DDR2 interface schematic for a x32 DDR2 memory system. The x16 DDR2 system schematic shown in Figure 6-12 is identical except that the high word DDR2 device is deleted.

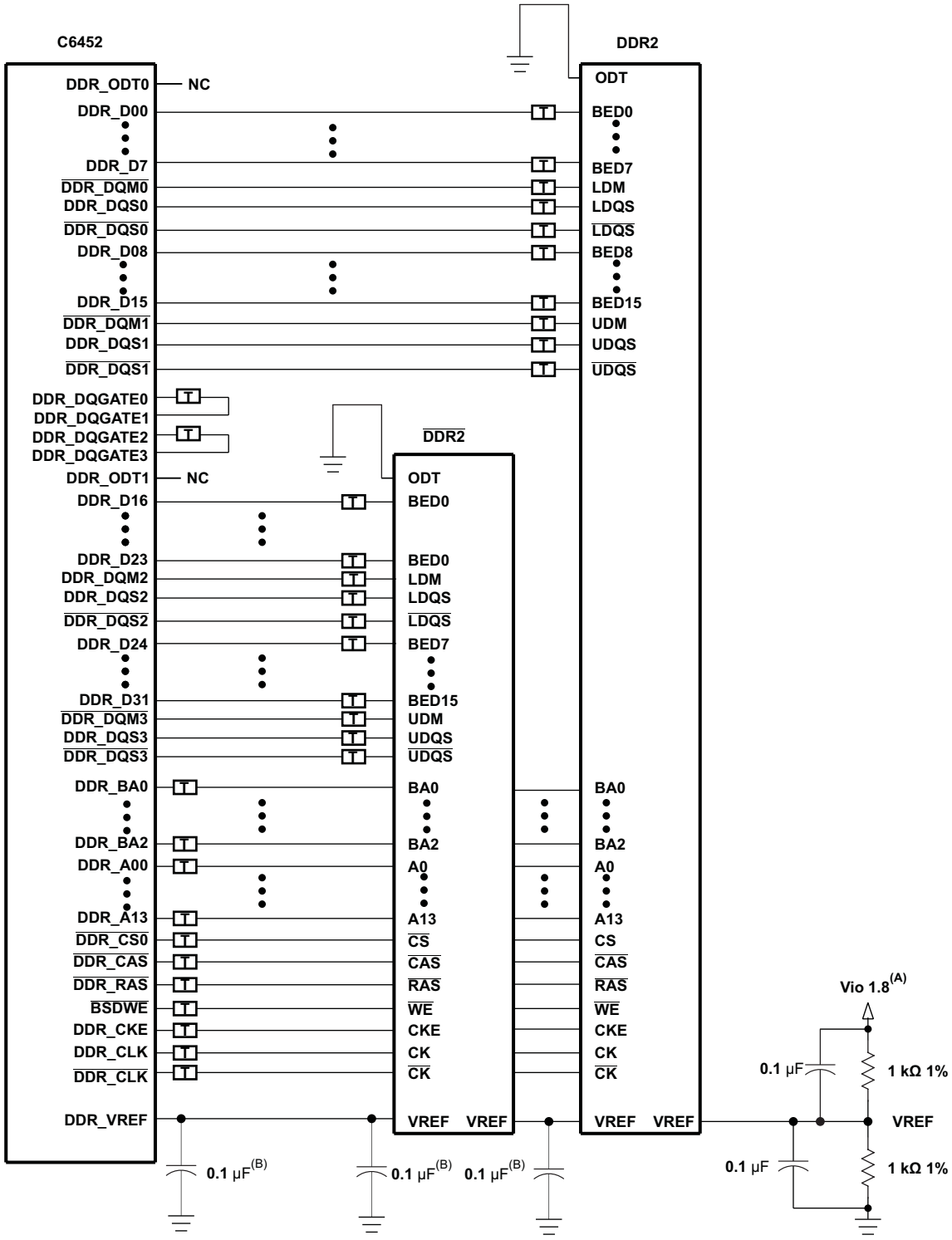
#### 6.9.3.2 Compatible JEDEC DDR2 Devices

Table 6-29 shows the parameters of the JEDEC DDR2 devices that are compatible with this interface. Generally, the DDR2 interface is compatible with x16 DDR2-533 speed grade DDR2 devices.

**Table 6-29. Compatible JEDEC DDR2 Devices**

| NO. | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| 1 | JEDEC DDR2 Device Speed Grade(1) | DDR2-533 | | |
| 2 | JEDEC DDR2 Device Bit Width | x16 | x16 | Bits |
| 3 | JEDEC DDR2 Device Count(2) | 1 | 2 | Devices |
| 4 | JEDEC DDR2 Device Ball Count(3) | 84 | 92 | Balls |

(1) Higher DDR2 speed grades are supported due to inherent JEDEC DDR2 backwards compatibility.
(2) One DDR2 device is used for 16-bit DDR2 memory system. Two DDR2 devices are used for 32-bit DDR2 memory system.
(3) 92 ball devices retained for legacy support. New designs will migrate to 84 ball DDR2 devices. Electrically, the 92 and 84 ball DDR2 devices are the same.

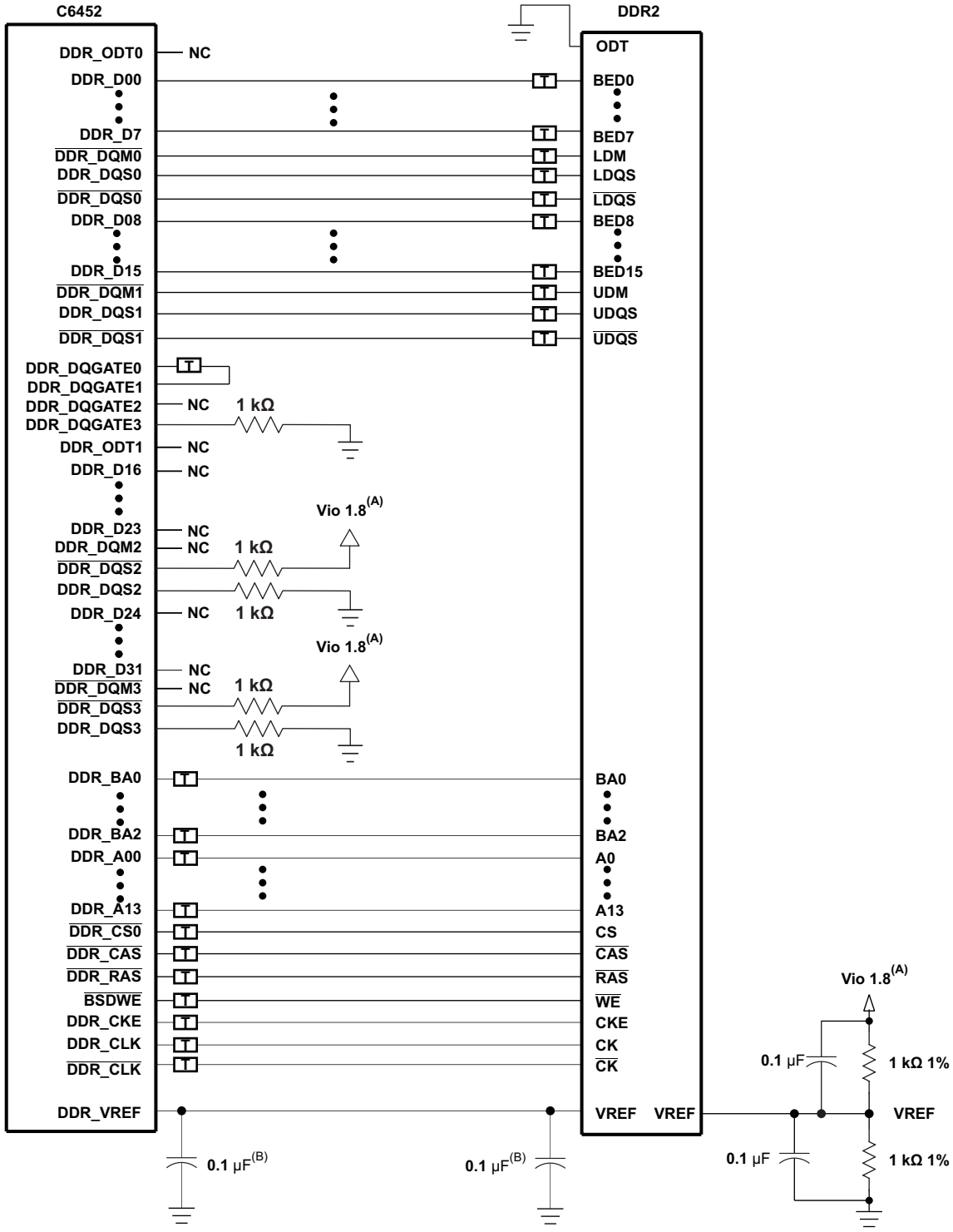



T Terminator, if desired. See terminator comments.

A. $V_{io}$ 1.8 is the power supply for the DDR2 memory interface.

B. One of these capacitors can be eliminated if the divider and its capacitors are placed near a device VREF pin.

**Figure 6-11. C6452 32-Bit DDR2 High-Level Schematic**




T Terminator, if desired. See terminator comments.

A. One of these capacitors can be eliminated if the divider and its capacitors are placed near a device VREF pin.

B. Vio 1.8 is the power supply for the DDR2 memory interface.

**Figure 6-12. C6452 16-Bit DDR2 High-Level Schematic**

#### 6.9.3.3 PCB Stackup

The minimum stackup required for routing the C6452 device is a six-layer stack as shown in Table 6-30. Additional layers may be added to the PCB stack up to accommodate other circuity or to reduce the size of the PCB footprint.

**Table 6-30. Minimum PCB Stack Up**

| LAYER | TYPE | DESCRIPTION |
|---|---|---|
| 1 | Signal | Top routing mostly horizontal |
| 2 | Plane | Ground |
| 3 | Plane | Power |
| 4 | Signal | Internal routing |
| 5 | Plane | Ground |
| 6 | Signal | Bottom routing mostly vertical |

Complete stack up specifications are provided in Table 6-31.

**Table 6-31. PCB Stack Up Specifications**

| NO. | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | PCB Routing/Plane Layers | 6 | | | |
| 2 | Signal Routing Layers | 3 | | | |
| 3 | Full ground layers under DDR2 routing Region | 2 | | | |
| 4 | Number of ground plane cuts allowed within DDR routing region | | | 0 | |
| 5 | Number of ground reference planes required for each DDR2 routing layer | 1 | | | |
| 6 | Number of layers between DDR2 routing layer and reference ground plane | | | 0 | |
| 7 | PCB Routing Feature Size | | 4 | | Mils |
| 8 | PCB Trace Width w | | 4 | | Mils |
| 8 | PCB BGA escape via pad size | | 18 | | Mils |
| 9 | PCB BGA escape via hole size | | 8 | | Mils |
| 10 | DSP Device BGA pad size (1) | | | | |
| 11 | DDR2 Device BGA pad size (2) | | | | |
| 12 | Single Ended Impedance, Zo | 50 | | 75 | Ω |
| 13 | Impedance Control (3) | Z-5 | Z | Z+5 | Ω |

(1) See the *Flip Chip Ball Grid Array Package Reference Guide* (SPRU811) for DSP device BGA pad size.
(2) See the DDR2 device manufacturer documenation for the DDR2 device BGA pad size.
(3) Z is the nominal singled ended impedance selected for the PCB specified by item 12.

#### 6.9.3.4 Placement

Figure 6-13 shows the required placement for the C6452 device as well as the DDR2 devices. The dimensions for Figure 6-13 are defined in Table 6-32. The placement does not restrict the side of the PCB where the devices are mounted. The purpose of the placement is to limit the maximum trace lengths and allow for proper routing space. For 16-bit DDR memory systems, the high word DDR2 device is omitted from the placement.

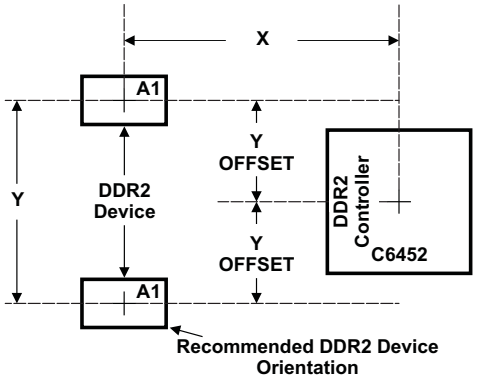



**Figure 6-13. DDR2 Device Placement**

**Table 6-32. Placement Specifications**

| NO. | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| 1 | X(1) (2) | | 1660 | Mils |
| 2 | Y(1) (2) | | 1280 | Mils |
| 3 | Y Offset(1) (2) (3) | | 650 | Mils |
| 4 | DDR2 Keepout Region(4) | | | |
| 5 | Clearance from non-DDR2 signal to DDR2 Keepout Region(5) | 4 | | w |

(1) See Figure 6-11 for dimension defintions.
(2) Measurements from center of DSP device to center of DDR2 device.
(3) For 16-bit memory systems, it is recommended that Y Offset be as small as possible.
(4) DDR2 Keepout region to encompass entire DDR2 routing area
(5) Non-DDR2 signals allowed within DDR2 keepout region provided they are separated from DDR2 routing layers by a ground plane.

### 6.9.3.5 DDR2 Keep Out Region

The region of the PCB used for the DDR2 circuitry must be isolated from other signals. The DDR2 keep out region is defined for this purpose and is shown in Figure 6-14. The size of this region varies with the placement and DDR routing. Additional clearances required for the keep out region are shown in Table 6-32.

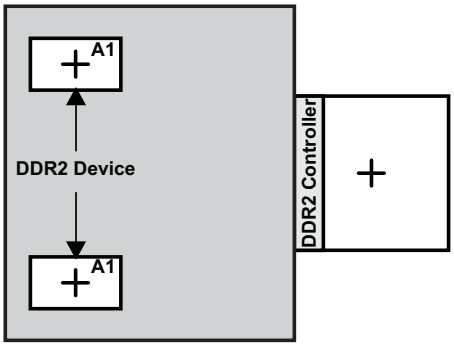



**Figure 6-14. DDR2 Keepout Region**

**NOTE**

The region (see Figure 6-14) should encompass all DDR2 circuitry and varies depending on placement. Non-DDR2 signals should not be routed on the DDR signal layers within the DDR2 keep out region. Non-DDR2 signals may be routed in the region provided they are routed on layers separated from DDR2 signal layers by a ground layer. No breaks should be allowed in the reference ground layers in this region. In addition, the 1.8-V power plane should cover the entire keep out region.

#### 6.9.3.6 Bulk Bypass Capacitors

Bulk bypass capacitors are required for moderate speed bypassing of the DDR2 and other circuitry. Table 6-33 contains the minimum numbers and capacitance required for the bulk bypass capacitors. This table covers only the bypass needs of the DSP and DDR2 interfaces. Additional bulk bypass capacitance may be needed for other circuitry.

**Table 6-33. Bulk Bypass Capacitors**

| NO. | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| 1 | $DV_{DD18}$ Bulk Bypass Capacitor Count(1) | 3 | | Devices |
| 2 | $DV_{DD18}$ Bulk Bypass Total Capacitance | 30 | | µF |
| 3 | DDR#1 Bulk Bypass Capacitor Count(1) | 1 | | Devices |
| 4 | DDR#1 Bulk Bypass Total Capacitance | 10 | | µF |
| 5 | DDR#2 Bulk Bypass Capacitor Count(2) (3) | 1 | | Devices |
| 6 | DDR#2 Bulk Bypass Total Capacitance(3) | 10 | | µF |

(1) These devices should be placed near the device they are bypassing, but preference should be given to the placement of the high-speed (HS) bypass caps.
(2) These devices should be placed near the device they are bypassing, but preference should be given to the placement of the high-speed (HS) bypass caps.
(3) Only used on 32-bit wide DDR2 memory systems

#### 6.9.3.7 High-Speed Bypass Capacitors

High-Speed (HS) bypass capacitors are critical for proper DDR2 interface operation. It is particularly important to minimize the parasitic series inductance of the HS bypass cap, DSP/DDR power, and DSP/DDR ground connections. Table 6-34 contains the specification for the HS bypass capacitors as well as for the power connections on the PCB.

#### 6.9.3.8 Net Classes

Table 6-35 lists the clock net classes for the DDR2 interface. Table 6-36 lists the signal net classes, and associated clock net classes, for the signals in the DDR2 interface. These net classes are used for the termination and routing rules that follow.

**Table 6-34. High-Speed Bypass Capacitors**

| NO. | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| 1 | HS Bypass Capacitor Package Size(1) | | 0402 | 10 Mils |
| 2 | Distance from HS bypass capacitor to device being bypassed | | 250 | Mils |
| 3 | Number of connection vias for each HS bypass capacitor(2) | 2 | | Vias |
| 4 | Trace length from bypass capacitor contact to connection via | 1 | 30 | Mils |
| 5 | Number of connection vias for each DDR2 device power or ground balls | 1 | | Vias |
| 6 | Trace length from DDR2 device power ball to connection via | | 35 | Mils |
| 7 | $DV_{DD18}$ HS Bypass Capacitor Count(3) | 20 | | Devices |
| 8 | $DV_{DD18}$ HS Bypass Capacitor Total Capacitance | 1.2 | | µF |

(1) L × W, 10 mil units ( i.e., a 0402 is a 40 × 20 mil surface mount capacitor)
(2) An additional HS bypass capacitor can share the connection vias only if it is mounted on the opposite side of the board.
(3) These devices should be placed as close as possible to the device being bypassed.

**Table 6-34. High-Speed Bypass Capacitors (continued)**

| NO. | PARAMETER | MIN | MAX | UNIT |
|---|---|---|---|---|
| 9 | DDR#1 HS Bypass Capacitor Count(3) | 8 | | Devices |
| 10 | DDR#1 HS Bypass Capacitor Total Capacitance | 0.4 | | µF |
| 11 | DDR#2 HS Bypass Capacitor Count(3) (4) | 8 | | Devices |
| 12 | DDR#2 HS Bypass Capacitor Total Capacitance(4) | 0.4 | | µF |

(4) Only used on 32-bit wide DDR2 memory systems

**Table 6-35. Clock Net Class Definitions**

| CLOCK NET CLASS | DSP PIN NAMES |
|---|---|
| CK | DDR_CLK/$\overline{\text{DDR\_CLK}}$ |
| DQS0 | DDR_DQS0/$\overline{\text{DDR\_DQS0}}$ |
| DQS1 | DDR_DQS1/$\overline{\text{DDR\_DQS1}}$ |
| DQS2(1) | DDR_DQS2/$\overline{\text{DDR\_DQS2}}$ |
| DQS3(1) | DDR_DQS3/$\overline{\text{DDR\_DQS3}}$ |

(1) Only used on 32-bit wide DDR2 memory systems.

**Table 6-36. Signal Net Class Definitions**

| CLOCK NET CLASS | ASSOCIATED CLOCK NET CLASS | DSP PIN NAMES |
|---|---|---|
| ADDR_CTRL | CK | $\overline{\text{DDR\_BA}}$[2:0], DDR_A[13:0], $\overline{\text{DDR\_CS}}$, $\overline{\text{DDR\_CAS}}$, $\overline{\text{DDR\_RAS}}$, $\overline{\text{DDR\_WE}}$, DDR_CKE |
| DQ0 | DQS0 | DDR_D[7:0], DDR_DQM0 |
| DQ1 | DQS1 | DDR_D[15:8], DDR_DQM1 |
| DQ2(1) | DQS2 | DDR_D[23:16], DDR_DQM2 |
| DQ3(1) | DQS3 | DDR_D[31:24], DDR_DQM3 |
| DQGATEL | CK, DQS0, DQS1 | DDR_DQGATE0, DDR_DQGATE1 |
| DQGATEH(1) | CK, DQS2, DQS3 | DDR_DQGATE2, DDR_DQGATE3 |

(1) Only used on 32-bit wide DDR2 memory systems.

### 6.9.3.9 DDR2 Signal Termination

No terminations of any kind are required in order to meet signal integrity and overshoot requirements. Serial terminators are permitted, if desired, to reduce EMI risk; however, serial terminations are the only type permitted. Table 6-37 shows the specifications for the series terminators.

**Table 6-37. DDR2 Signal Terminations**

| NO. | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | CK Net Class(1) | 0 | | 10 | Ω |
| 2 | ADDR_CTRL Net Class(1) (2) (3) | 0 | 22 | Zo | Ω |
| 3 | Data Byte Net Classes (DQS0-DQS3, DQ0-DQ3)(1) (2) (3) (4) | 0 | 22 | Zo | Ω |
| 4 | DQGATE Net Classes (DQGATEL, DQGATEH)(1) (2) (3) | 0 | 10 | Zo | Ω |

(1) Only series termination is permitted, parallel or SST specifically disallowed.
(2) Terminator values larger than typical only recommended to address EMI issues.
(3) Termination value should be uniform across net class.
(4) When no termination is used on data lines (0 Ωs), the DDR2 devices must be programmed to operate in 60% strength mode.

### 6.9.3.10 VREF Routing

VREF is used as a reference by the input buffers of the DDR2 memories as well as the device's. VREF is intended to be 1/2 the DDR2 power supply voltage and should be created using a resistive divider as shown in Figure 6-11. Other methods of creating VREF are not recommended. Figure 6-15 shows the layout guidelines for VREF.

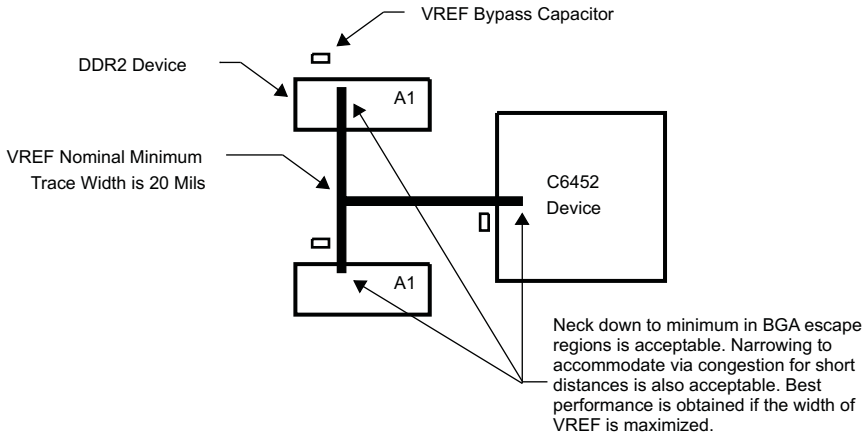



**Figure 6-15. VREF Routing and Topology**

### 6.9.3.11 DDR2 CK and ADDR_CTRL Routing

Figure 6-16 shows the topology of the routing for the CK and ADDR_CTRL net classes. The route is a balanced *T* as it is intended that the length of segments B and C be equal. In addition, the length of A should be maximized.

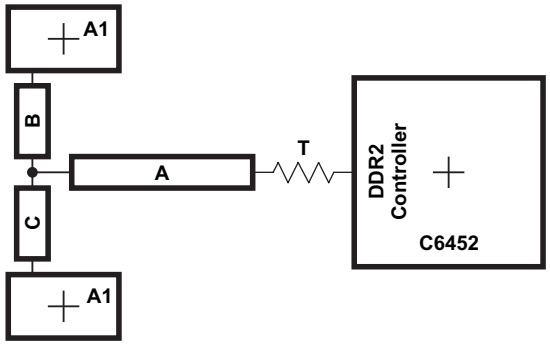



**Figure 6-16. CK and ADDR_CTRL Routing and Topology**

**Table 6-38. CK and ADDR_CTRL Routing Specification(1)**

| NO | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | Center to center CK-$\overline{CK}$ spacing | | | 2w | |
| 2 | CK A to B/A to C Skew Length Mismatch(1) | | | 25 | Mils |
| 3 | CK B to C Skew Length Mismatch | | | 25 | Mils |
| 4 | Center to center CK to other DDR2 trace spacing(2) | 4w | | | |
| 5 | CK/ADDR_CTRL nominal trace length(3) | CACLM-50 | CACLM | CACLM+50 | Mils |
| 6 | ADDR_CTRL to CK Skew Length Mismatch | | | 100 | Mils |
| 7 | ADDR_CTRL to ADDR_CTRL Skew Length Mismatch | | | 100 | Mils |
| 8 | Center to center ADDR_CTRL to other DDR2 trace spacing(2) | 4w | | | |
| 9 | Center to center ADDR_CTRL to other ADDR_CTRL trace spacing(2) | 3w | | | |
| 10 | ADDR_CTRL A to B/A to C Skew Length Mismatch(1) | | | 100 | Mils |
| 11 | ADDR_CTRL B to C Skew Length Mismatch | | | 100 | Mils |

(1) Series terminator, if used, should be located closest to DSP.
(2) Center to center spacing is allowed to fall to minimum (w) for up to 500 mils of routed length to accommodate BGA escape and routing congestion.
(3) CACLM is the longest Manhattan distance of the CK and ADDR_CTRL net classes.

Figure 6-17 shows the topology and routing for the DQS and DQ net classes; the routes are point to point. Skew matching across bytes is not needed nor recommended.

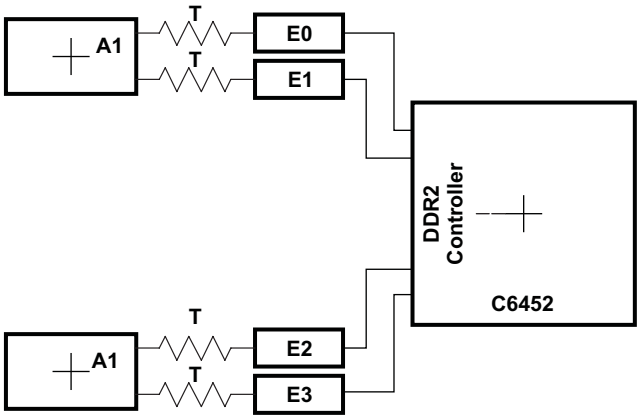



**Figure 6-17. DQS and DQ Routing and Toplogy**

**Table 6-39. DQS and DQ Routing Specification (1)**

| NO. | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | Center to center DQS-$\overline{DQS}$ spacing | | | 2 w | |
| 2 | DQS E Skew Length Mismatch | | | 25 | Mils |
| 3 | Center to center DQS to other DDR2 trace spacing(2) | 4 w | | | |
| 4 | DQS/DQ nominal trace length(1) (3) (4) (5) | DQLM-50 | DQLM | DQLM+50 | Mils |
| 5 | DQ to DQS Skew Length Mismatch(3) (4) (5) | | | 100 | Mils |
| 6 | DQ to DQ Skew Length Mismatch(3) (4) (5) | | | 100 | Mils |
| 7 | Center to center DQ to other DDR2 trace spacing(2) (6) | 4 w | | | |

(1) Series terminator, if used, should be located closest to DDR.
(2) Center to center spacing is allowed to fall to minimum (w) for up to 500 mils of routed length to accommodate BGA escape and routing congestion.
(3) A 16-bit DDR memory system has two sets of data net classes, one for data byte 0, and one for data byte 1, each with an associated DQS (2 DQSs).
(4) A 32-bit DDR memory system will have four sets of data net classes, one each for data bytes 0 through 3, and each associated with a DQS (4 DQSs).
(5) There is no need and it is not recommended to skew match across data bytes, i.e., from DQS0 and data byte 0 to DQS1 and data byte 1.
(6) DQs from other DQS domains are considered *other DDR2 trace*.

**Table 6-39. DQS and DQ Routing Specification (1) (continued)**

| NO. | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 8 | Center to Center DQ to other DQ trace spacing (7) (2) | 3w | | | |
| 9 | DQ/DQS E Skew Length Mismatch (3) (4) (5) | | | 100 | Mils |

(7) DQLM is the longest Manhattan distance of each of the DQS and DQ net classes.

Figure 6-18 shows the routing for the DQGATE net classes. Table 6-40 contains the routing specification.

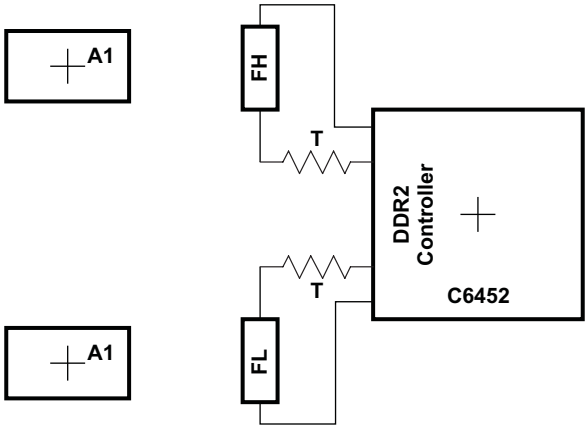



**Figure 6-18. DQGATE Routing**

**Table 6-40. DQGATE Routing Specification**

| NO. | PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | DQGATEL Length F (1) | | CKB0B1 | | |
| 2 | DQGATEH Length F (2) (3) | | CKB2B3 | | |
| 3 | Center to center DQGATE to any other trace spacing | 4 w | | | |
| 4 | DQS/DQ nominal trace length | DQLM - 50 | DQLM | DQLM + 50 | Mils |
| 5 | DQGATEL Skew (4) | | | 100 | Mils |
| 6 | DQGATEH Skew (5) (3) | | | 100 | Mils |

(1) CKB0B1 is the sum of the length of the CK net plus the average length of the DQS0 and DQS1 nets.
(2) CKB2B3 is the sum of the length of the CK net plus the average length of the DQS2 and DQS3 nets.
(3) Only used on 32-bit-wide DDR2 memory systems.
(4) Skew from CKB0B1
(5) Skew from CKB2B3

## 6.10 External Memory Interface A (EMIFA)

The EMIFA can interface to a variety of external devices or ASICs, including:

- Pipelined and flow-through synchronous-burst SRAM (SBSRAM)
- ZBT (zero bus turnaround) SRAM and late write SRAM
- Synchronous FIFOs
- Asynchronous memory, including SRAM, ROM, and Flash

### 6.10.1 EMIFA Device-Specific Information

Timing analysis must be done to verify all ac timing requirements are met. TI recommends utilizing I/O buffer information specification (IBIS) to analyze all ac timing.

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

To maintain signal integrity, serial termination resistors should be inserted into all EMIFA output signal lines.

A race condition may exist when certain masters write data to the EMIFA. For example, if master A passes a software message via a buffer in external memory and does not wait for indication that the write completes, when master B attempts to read the software message, then the master B read may bypass the master A write and, thus, master B may read stale data and, therefore, receive an incorrect message.

Some master peripherals (e.g., EDMA3 transfer controllers) will always wait for the write to complete before signaling an interrupt to the system, thus avoiding this race condition. For masters that do not have hardware specification of write-read ordering, it may be necessary to specify data ordering via software.

If master A does not wait for indication that a write is complete, it must perform the following workaround:

1. Perform the required write.
2. Perform a dummy write to the EMIFA module ID and revision register.
3. Perform a dummy read to the EMIFA module ID and revision register.
4. Indicate to master B that the data is ready to be read after completion of the read in step 3. The completion of the read in step 3 ensures that the previous write was done.

### 6.10.2 EMIFA Peripheral Register Description(s)

For more information on the EMIF registers shown in Table 6-41, see *TMS320C6452 DSP External Memory Interface (EMIF) User's Guide* (literature number SPRUFF8).

**Table 6-41. EMIFA Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x7000 0000 | MIDR | Module ID and Revision Register |
| 0x7000 0004 | STAT | Status Register |
| 0x7000 0008 | - | Reserved |
| 0x7000 000C - 0x7000 001C | - | Reserved |
| 0x7000 0020 | BPRIO | Burst Priority Register |
| 0x7000 0024 - 0x7000 004C | - | Reserved |
| 0x7000 0050 - 0x7000 007C | - | Reserved |
| 0x7000 0080 | CE2CFG | EMIFA CE2 Configuration Register |
| 0x7000 0084 | CE3CFG | EMIFA CE3 Configuration Register |
| 0x7000 0088 | - | Reserved |
| 0x7000 008C | - | Reserved |
| 0x7000 0090 - 0x7000 009C | - | Reserved |
| 0x7000 00A0 | AWCC | EMIFA Async Wait Cycle Configuration Register |

**Table 6-41. EMIFA Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x7000 00A4 - 0x7000 00BC | - | Reserved |
| 0x7000 00C0 | INTRAW | EMIFA Interrupt RAW Register |
| 0x7000 00C4 | INTMSK | EMIFA Interrupt Masked Register |
| 0x7000 00C8 | INTMSKSET | EMIFA Interrupt Mask Set Register |
| 0x7000 00CC | INTMSKCLR | EMIFA Interrupt Mask Clear Register |
| 0x7000 00D0 - 0x7000 00DC | - | Reserved |
| 0x7000 00E0 - 0x77FF FFFF | - | Reserved |

### 6.10.3 EMIFA Electrical Data/Timing

**Table 6-42. Timing Requirements for AECLKIN for EMIFA(1) (2) (see Figure 6-19)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(EKI)}$ | Cycle time, AECLKIN | 6(3) | 16P(4) | ns |
| 2 | $t_{w(EKIH)}$ | Pulse duration, AECLKIN high | 2.7 | | ns |
| 3 | $t_{w(EKIL)}$ | Pulse duration, AECLKIN low | 2.7 | | ns |
| 4 | $t_{t(EKI)}$ | Transition time, AECLKIN | | 2 | ns |
| 5 | $t_{J(EKI)}$ | Period Jitter, AECLKIN | | 0.02E(5) | ns |

(1) The reference points for the rise and fall transitions are measured at $V_{IL}$ MAX and $V_{IH}$ MIN.
(2) E = the EMIF input clock (AECLKIN or SYSCLK4/2) period in ns for EMIFA.
(3) Minimum AECLKIN cycle times *must* be met, even when AECLKIN is generated by an internal clock source. Minimum AECLKIN times are based on internal logic speed; the maximum useable speed of the EMIF may be lower due to AC timing requirements.
(4) P is P = 1/CPU clock frequency in ns.
(5) This timing applies only when AECLKIN is used for EMIFA.

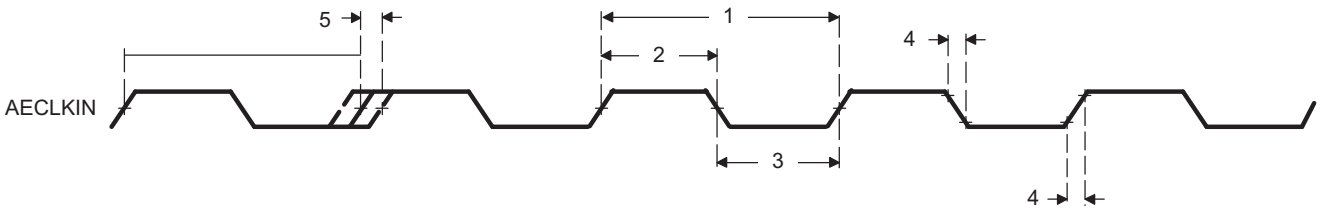
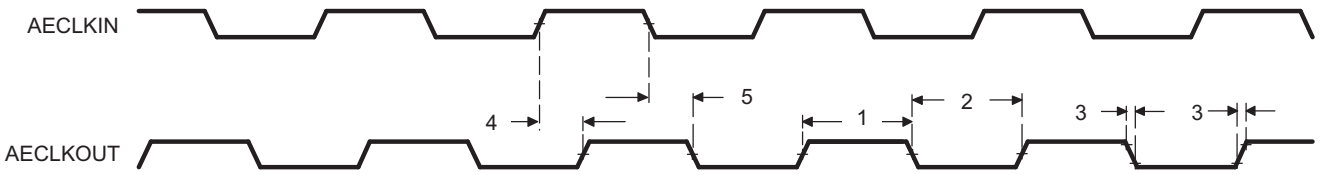



**Figure 6-19. AECLKIN Timing for EMIFA**

**Table 6-43. Switching Characteristics Over Recommended Operating Conditions for AECLKOUT for the EMIFA Module(1) (2) (3) (see Figure 6-20)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(EKO)}$ | Cycle time, AECLKOUT | E - 0.7 | E + 0.7 | ns |
| 2 | $t_{w(EKOH)}$ | Pulse duration, AECLKOUT high | EH - 0.7 | EH + 0.7 | ns |
| 3 | $t_{w(EKOL)}$ | Pulse duration, AECLKOUT low | EL - 0.7 | EL + 0.7 | ns |
| 4 | $t_{t(EKO)}$ | Transition time, AECLKOUT | | 1 | ns |
| 5 | $t_{d(EKIH-EKOH)}$ | Delay time, AECLKIN high to AECLKOUT high | 1 | 8 | ns |
| 6 | $t_{d(EKIL-EKOL)}$ | Delay time, AECLKIN low to AECLKOUT low | 1 | 8 | ns |

(1) E = the EMIF input clock (AECLKIN or SYSCLK4/2) period in ns for EMIFA.
(2) The reference points for the rise and fall transitions are measured at $V_{OL}$ MAX and $V_{OH}$ MIN.
(3) EH is the high period of E (EMIF input clock period) in ns and EL is the low period of E (EMIF input clock period) in ns for EMIFA.




A. E = the EMIF input clock (AECLKIN or SYSCLK4/2) period in ns for EMIFA.
B. The reference points for the rise and fall transitions are measured at $V_{OL}$ MAX and $V_{OH}$ MIN.
C. EH is the high period of E (EMIF input clock period) in ns and EL is the low period of E (EMIF input clock period) in ns for EMIFA.

**Figure 6-20. AECLKOUT Timing for the EMIFA Module**

### 6.10.3.1 Asynchronous Memory Timing

**Table 6-44. Timing Requirements for Asynchronous Memory Cycles for EMIFA Module(1) (2) (3) (see Figure 6-21 and Figure 6-22)**

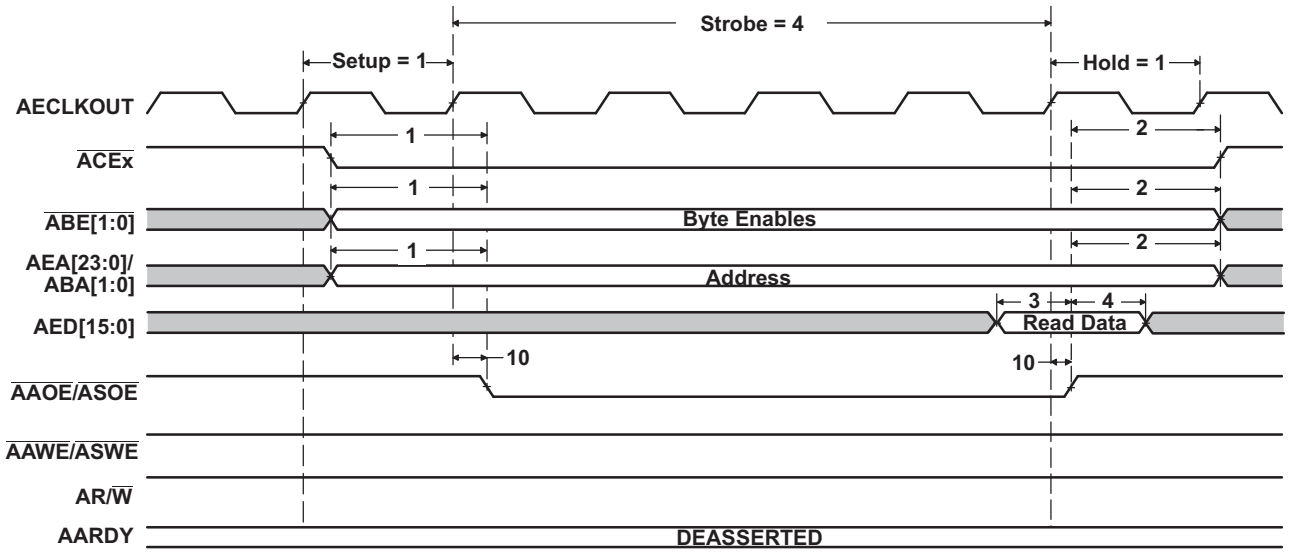
| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 3 | $t_{su(EDV-A\ OEH)}$ | Setup time, AEDx valid before $\overline{AAOE}/\overline{ASOE}$ high | 6.5 | | ns |
| 4 | $t_{h(AOEH-EDV)}$ | Hold time, AEDx valid after $\overline{AAOE}/\overline{ASOE}$ high | 0 | | ns |
| 5 | $t_{su(ARDY-EKOH)}$ | Setup time, AARDY valid before AECLKOUT low | 1 | | ns |
| 6 | $t_{h(EKOH-ARDY)}$ | Hold time, AARDY valid after AECLKOUT low | 2 | | ns |
| 7 | $t_{w(ARDY\ )}$ | Pulse width, AARDY assertion and deassertion | 2E + 5 | | ns |
| 8 | $t_{d(ARDY-HOLD)}$ | Delay time, from AARDY sampled deasserted on AECLKOUT falling to beginning of programmed hold period | | 4E | ns |
| 9 | $t_{su(ARDY-HOLD)}$ | Setup time, before end of programmed strobe period by which AARDY should be asserted in order to insert extended strobe wait states. | 2E | | ns |

(1) E = AECLKOUT period in ns for EMIFA
(2) To specify data setup time, simply program the strobe width wide enough.
(3) AARDY is internally synchronized. To use AARDY as an asynchronous input, the pulse width of the AARDY signal should be at least 2E to specify setup and hold time is met.

**Table 6-45. Switching Characteristics Over Recommended Operating Conditions for Asynchronous Memory Cycles for EMIFA Module(1) (2) (3) (see Figure 6-21 and Figure 6-22)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{osu(SELV-AOEL)}$ | Output setup time, select signals valid to $\overline{AAOE}/\overline{ASOE}$ low | RS × E - 1.5 | | ns |
| 2 | $t_{oh(AOEH-SELIV)}$ | Output hold time, $\overline{AAOE}/\overline{ASOE}$ high to select signals invalid | RS × E - 1.9 | | ns |
| 10 | $t_{d(EKOH-AOEV)}$ | Delay time, AECLKOUT high to $\overline{AAOE}/\overline{ASOE}$ valid | 1 | 7.28 | ns |
| 11 | $t_{osu(SELV-AWEL)}$ | Output setup time, select signals valid to $\overline{AAWE}/\overline{ASWE}$ low | WS × E - 1.7 | | ns |
| 12 | $t_{oh(AWEH-SELIV)}$ | Output hold time, $\overline{AAWE}/\overline{ASWE}$ high to select signals invalid | WH × E - 1.8 | | ns |
| 13 | $t_{d(EKOH-AWEV)}$ | Delay time, AECLKOUT high to $\overline{AAWE}/\overline{ASWE}$ valid | 1.3 | 7.1 | ns |

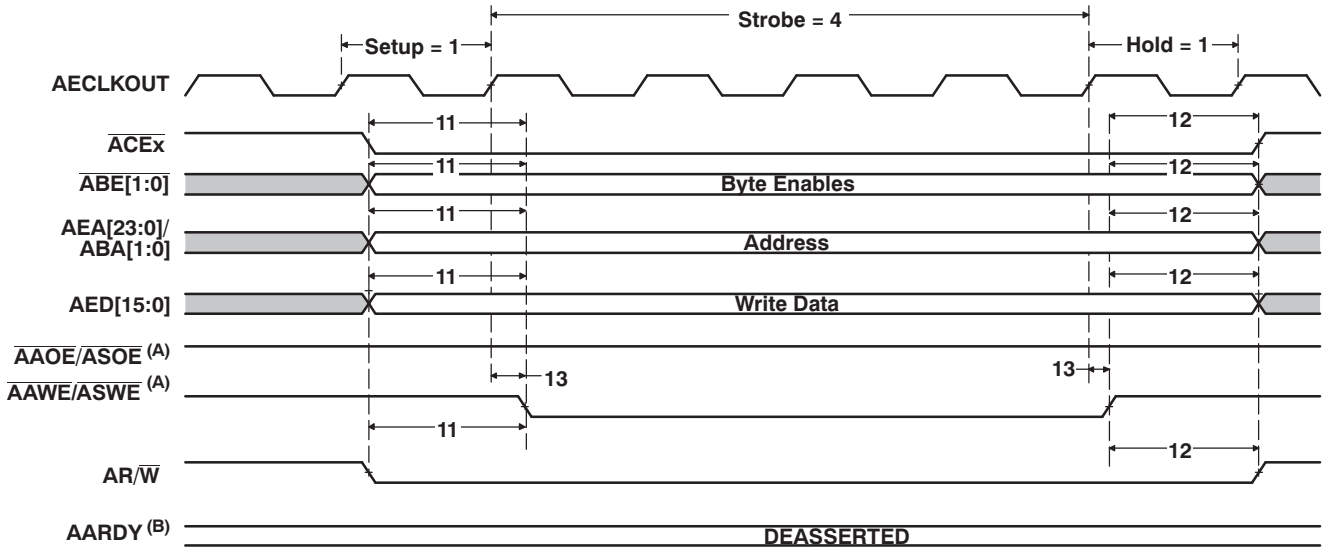
(1) E = AECLKOUT period in ns for EMIFA
(2) RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold. These parameters are programmed via the EMIFA CE Configuration registers (CEnCFG).
(3) Select signals for EMIFA include: $\overline{ACEx}$, $\overline{ABE}$[1:0], AEA[23:0], ABA[1:0]; and for EMIFA writes, also include AR/$\overline{W}$, AED[15:0].




A. AAOE/ASOE and AAWE/ASWE operate as AAOE/ASOE (identified under select signals) and AAWE/ASWE, respectively, during asynchronous memory accesses.

B. Polarity of the AARDY signal is programmable through the AP field of the EMIFA Async Wait Cycle Configuration register (AWCC).

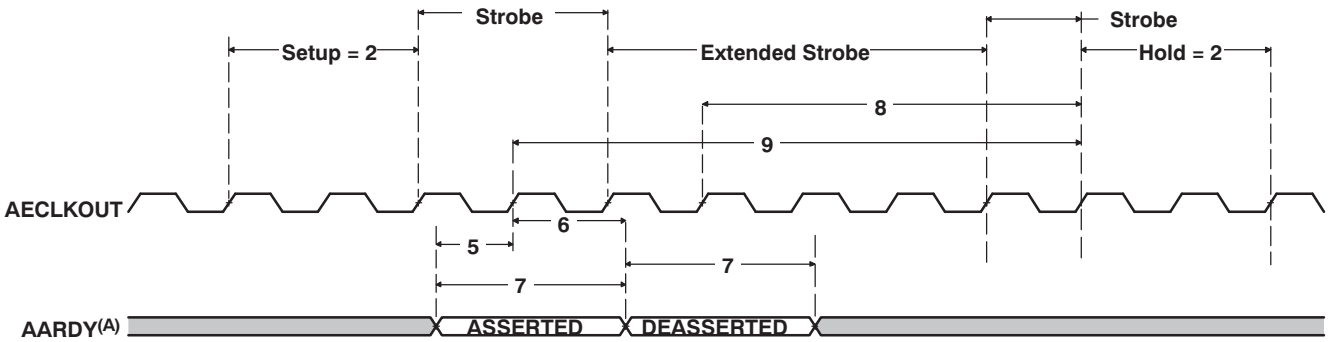
**Figure 6-21. Asynchronous Memory Read Timing for EMIFA**




A. AAOE/ASOE and AAWE/ASWE operate as AAOE/ASOE and AAWE/ASWE (identified under select signals) during asynchronous memory accesses.

B. Polarity of the AARDY signal is programmable through the AP field of the EMIFA Async Wait Cycle Configuration register (AWCC).

**Figure 6-22. Asynchronous Memory Write Timing for EMIFA**




A. Polarity of the AARDY signal is programmable through the AP field of the EMIFA Async Wait Cycle Configuration register (AWCC).

**Figure 6-23. AARDY Timing**

### 6.10.3.2 Programmable Synchronous Interface Timing

**Table 6-46. Timing Requirements for Programmable Synchronous Interface Cycles for EMIFA Module (see Figure 6-24)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 6 | $t_{su(EDV-EKOH)}$ | Setup time, read AEDx valid before AECLKOUT high | 2 | | ns |
| 7 | $t_{h(EKOH-EDV)}$ | Hold time, read AEDx valid after AECLKOUT high | 1.5 | | ns |

**Table 6-47. Switching Characteristics Over Recommended Operating Conditions for Programmable Synchronous Interface Cycles for EMIFA Module(1) (see Figure 6-24-Figure 6-26)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{d(EKOH-CEV)}$ | Delay time, AECLKOUT high to $\overline{ACE}$x valid | 1.3 | 4.9 | ns |
| 2 | $t_{d(EKOH-BEV)}$ | Delay time, AECLKOUT high to $\overline{ABE}$x valid | | 4.9 | ns |
| 3 | $t_{d(EKOH-BEIV)}$ | Delay time, AECLKOUT high to $\overline{ABE}$x invalid | 1.3 | | ns |
| 4 | $t_{d(EKOH-EAV)}$ | Delay time, AECLKOUT high to AEAx valid | | 4.9 | ns |
| 5 | $t_{d(EKOH-EAIV)}$ | Delay time, AECLKOUT high to AEAx invalid | 1.3 | | ns |
| 8 | $t_{d(EKOH-ADSV)}$ | Delay time, AECLKOUT high to $\overline{ASADS}$/$\overline{ASRE}$ valid | 1.3 | 4.9 | ns |
| 9 | $t_{d(EKOH-OEV)}$ | Delay time, AECLKOUT high to $\overline{AAOE}$/$\overline{ASOE}$ valid | 1.3 | 4.9 | ns |
| 10 | $t_{d(EKOH-EDV)}$ | Delay time, AECLKOUT high to AEDx valid | | 4.9 | ns |
| 11 | $t_{d(EKOH-EDIV)}$ | Delay time, AECLKOUT high to AEDx invalid | 1.3 | | ns |
| 12 | $t_{d(EKOH-WEV)}$ | Delay time, AECLKOUT high to $\overline{AAWE}$/$\overline{ASWE}$ valid | 1.3 | 4.9 | ns |

(1) The following parameters are programmable via the EMIFA CE Configuration registers (CEnCFG):
- Read latency (R_LTNCY): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (W_LTNCY): 0-, 1-, 2-, or 3-cycle write latency
- $\overline{ACE}$x assertion length (CE_EXT): For standard SBSRAM or ZBT SRAM interface, $\overline{ACE}$x goes inactive after the final command has been issued (CE_EXT = 0). For synchronous FIFO interface with glue, $\overline{ACE}$x is active when $\overline{AAOE}$/$\overline{ASOE}$ is active (CE_EXT = 1).
- Function of $\overline{ASADS}$/$\overline{ASRE}$ (R_ENABLE): For standard SBSRAM or ZBT SRAM interface, $\overline{ASADS}$/$\overline{ASRE}$ has deselect cycles (R_ENABLE = 0). For FIFO interface, $\overline{ASADS}$/$\overline{ASRE}$ has NO deselect cycles (R_ENABLE = 1).

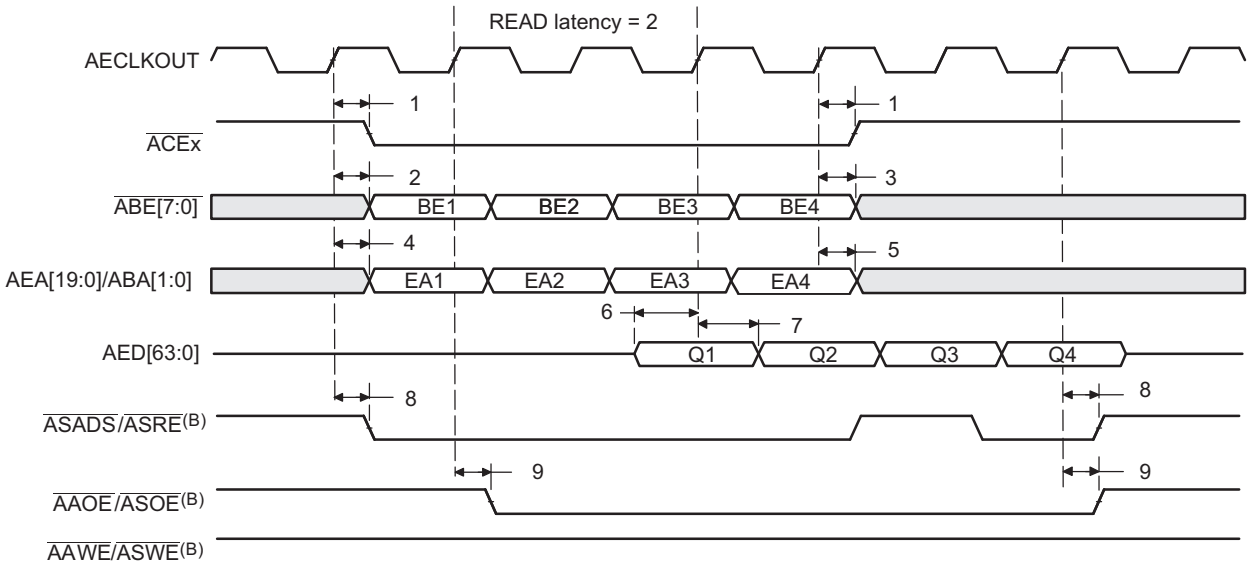


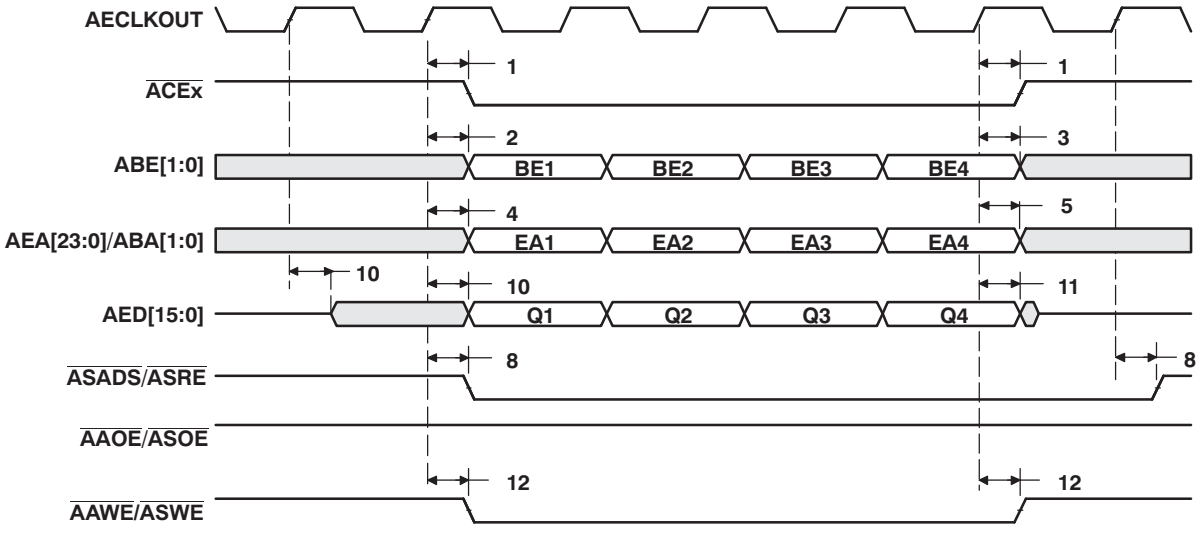

**Figure 6-24. Programmable Synchronous Interface Read Timing for EMIFA (With Read Latency = 2)**

**NOTE**

This information applies to Figure 6-25 and Figure 6-26.

The following parameters are programmable via the EMIF Chip Select n Configuration Register (CESECn):

- Read latency (R_LTNCY): 1-, 2-, or 3-cycle read latency
- Write latency (W_LTNCY): 0-, 1-, 2-, or 3-cycle write latency
- $\overline{\text{ACEx}}$ assertion length (CE_EXT): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{ACEx}}$ goes inactive after the final command has been issued (CE_EXT = 0). For synchronous FIFO interface, $\overline{\text{ACEx}}$ is active when $\overline{\text{AAOE}}/\overline{\text{ASOE}}$ is active (CE_EXT = 1).
- Function of $\overline{\text{ASADS}}/\overline{\text{ASRE}}$ (R_ENABLE): For standard SBSRAM or ZBT SRAM interface, $\overline{\text{ASADS}}/\overline{\text{ASRE}}$ has deselect cycles (R_ENABLE = 0). For FIFO interface, $\overline{\text{ASADS}}/\overline{\text{ASRE}}$ has NO deselect cycles (R_ENABLE = 1).




A. In this figure, W_LTNCY = 0, CE_EXT = 0, R_ENABLE = 0, and SSEL = 1.

**Figure 6-25. Programmable Synchronous Interface Write Timing for EMIFA (With Write Latency = 0)**

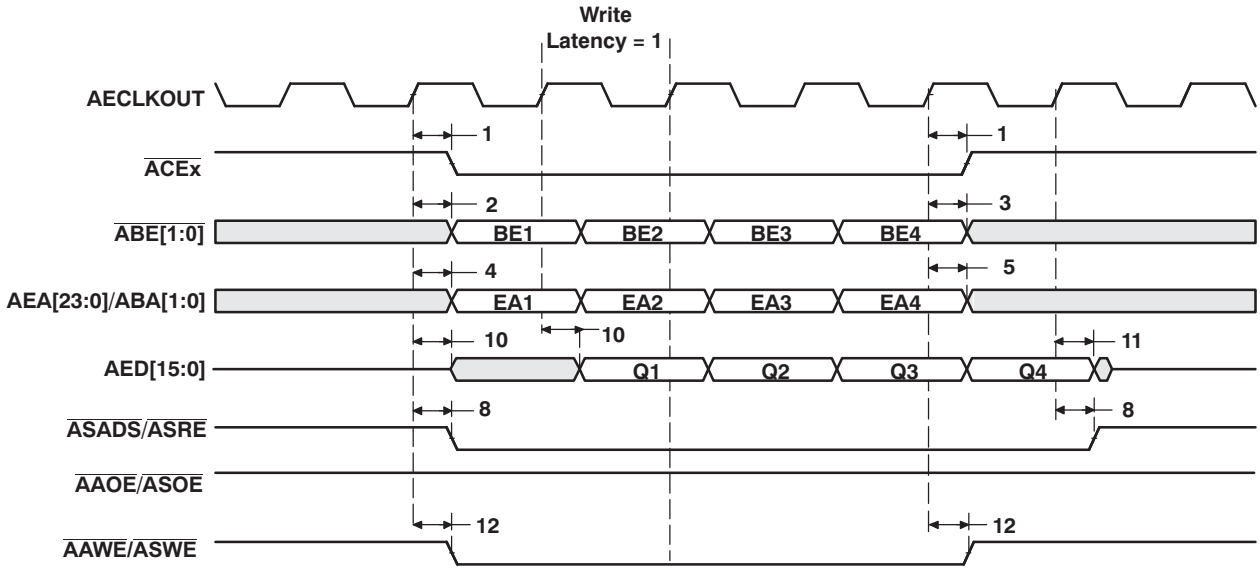



**Figure 6-26. Programmable Synchronous Interface Write Timing for EMIFA (With Write Latency = 1)**

## 6.11 Universal Asynchronous Receiver/Transmitter (UART)

The device has a UART peripheral. The UART has the following features:

- 16-byte storage space for both the transmitter and receiver FIFOs
- 1, 4, 8, or 14 byte selectable receiver FIFO trigger level for autoflow control and DMA
- DMA signaling capability for both received and transmitted data
- Programmable auto-rts and auto-cts for autoflow control
- Frequency pre-scale values from 1 to 65, 535 to generate appropriate baud rates
- Prioritized interrupts
- Programmable serial data formats
  - 5, 6, 7, or 8-bit characters
  - Even, odd, or no parity bit generation and detection
  - 1, 1.5, or 2 stop bit generation
- False start bit detection
- Line break generation and detection
- Internal diagnostic capabilities
  - Loopback controls for communications link fault isolation
  - Break, parity, overrun, and framing error simulation
- Modem control functions (CTS, RTS).

The UART registers are listed in Table 6-48 .

### *6.11.1 UART Peripheral Register Description(s)*

**Table 6-48. UART Register Descriptions**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 7000 | RBR | UART Receiver Buffer Register (Read Only) |
| 0x0204 7000 | THR | UART Transmitter Holding Register (Write Only) |
| 0x0204 7004 | IER | UART Interrupt Enable Register |
| 0x0204 7008 | IIR | UART Interrupt Identification Register (Read Only) |
| 0x0204 7008 | FCR | UART FIFO Control Register (Write Only) |
| 0x0204 700C | LCR | UART Line Control Register |
| 0x0204 7010 | MCR | UART Modem Control Register |
| 0x0204 7014 | LSR | UART Line Status Register |
| 0x0204 7018 | - | Reserved |
| 0x0204 701C | - | Reserved |
| 0x0204 7020 | DLL | UART Divisor Latch (LSB) |
| 0x0204 7024 | DLH | UART Divisor Latch (MSB) |
| 0x0204 7028 | PID | Peripheral Identification Register |
| 0x0204 702C | | Reserved |
| 0x0204 7030 | PWREMU_MGMT | UART Power and Emulation Management Register |
| 0x0204 7034 | MDR | Mode Definition Register |
| 0x0204 7038 - 0x0204 73FF | - | Reserved |

### 6.11.2 UART Electrical Data/Timing

**Table 6-49. Timing Requirements for UARTx Receive[1] (see Figure 6-27)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 4 | $t_{w(URXDB)}$ | Pulse duration, receive data bit (RXDn) [15/30/100 pF] | 0.96U | 1.05U | ns |
| 5 | $t_{w(URXSB)}$ | Pulse duration, receive start bit [15/30/100 pF] | 0.96U | 1.05U | ns |

(1) U = UART baud time = 1/programmed baud rate.

**Table 6-50. Switching Characteristics Over Recommended Operating Conditions for UARTx Transmit[1] (see Figure 6-27)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $f_{(baud)}$ | Maximum programmable baud rate | | 5 | MHz |
| 2 | $t_{w(UTXDB)}$ | Pulse duration, transmit data bit (TXDn) [15/30/100 pF] | U - 2 | U + 2 | ns |
| 3 | $t_{w(UTXSB)}$ | Pulse duration, transmit start bit [15/30/100 pF] | U - 2 | U + 2 | ns |

(1) U = UART baud time = 1/programmed baud rate.

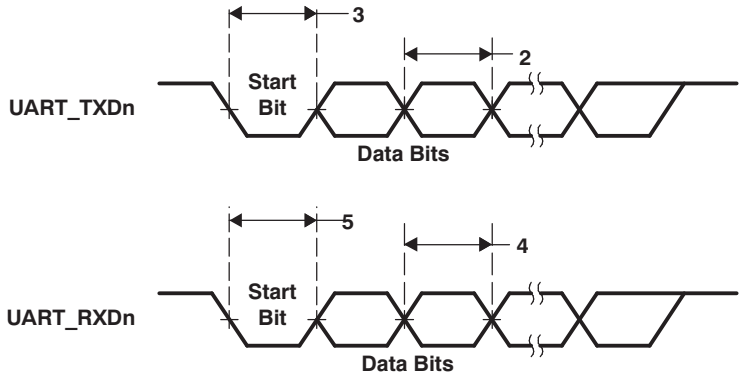



**Figure 6-27. UART Transmit/Receive Timing**

## 6.12 Serial Peripheral Interface Port (SPI)

### 6.12.1 SPI Device-Specific Information

Figure 6-28 is a block diagram of the SPI module, which is a simple shift register and buffer plus control logic. Data is written to the shift register before transmission occurs and is read from the buffer at the end of transmission. The SPI can operate only as a master, in which case, it initiates a transfer and drives the SPICLK pin. Four clock phase and polarity options are supported as well as many data formatting options.

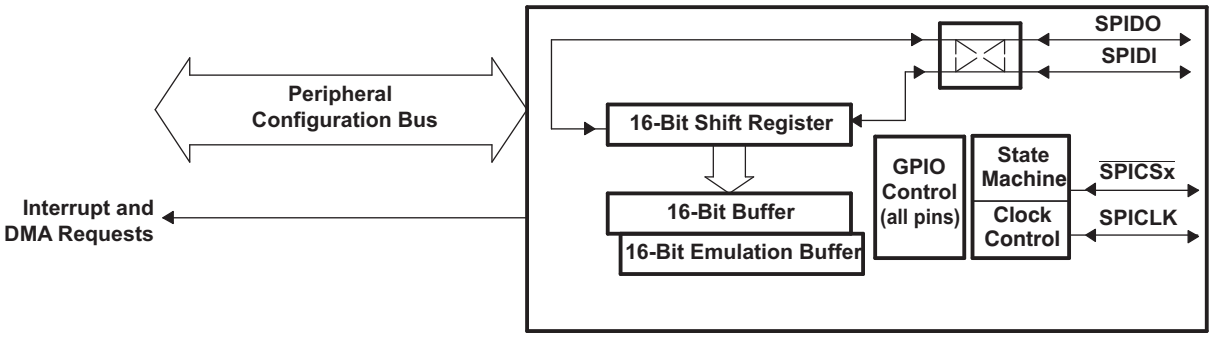



**Figure 6-28. Block Diagram of SPI Module**

The SPI supports 3- and 4-pin operation with three basic pins (SPICLK, SPIDO, and SPIDI) and two optional pins (SPICSx).

The optional SPICSx (Slave Chip Select) pin is most useful to enable in master mode when there are more than one slave devices on the same SPI port. The device only shifts data and drives the SPIDI pin when SPICSx is held low.

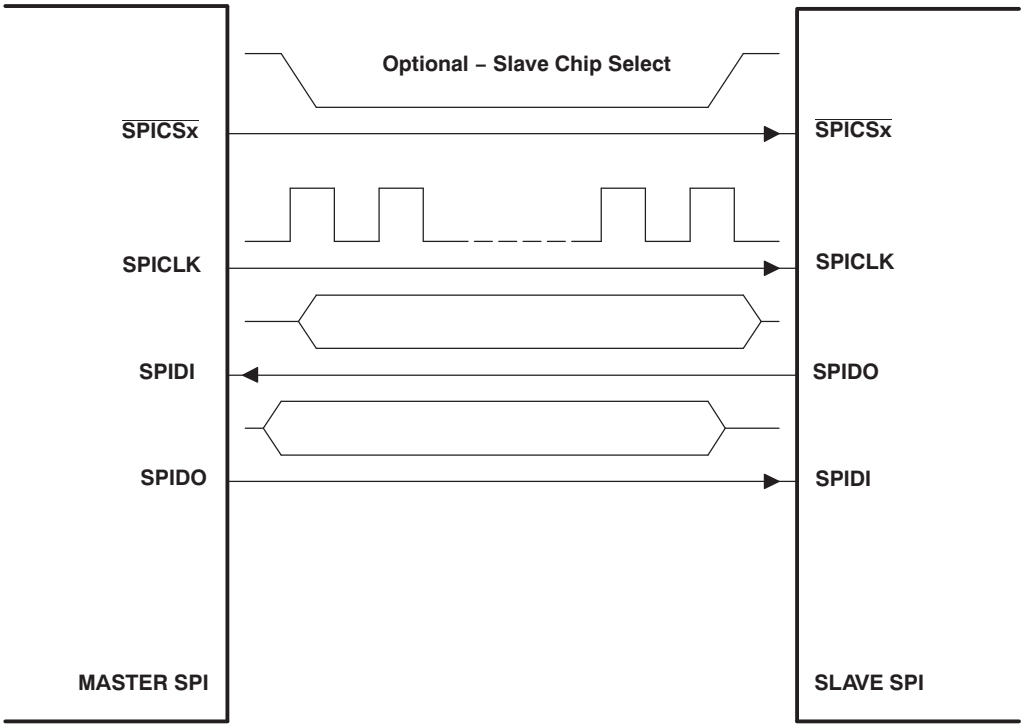



**Figure 6-29. Illustration of SPI Master-to-SPI Slave Connection**

### 6.12.2 SPI Peripheral Register Descriptions

Table 6-51 is a list of the SPI registers.

**Table 6-51. SPI Configuration Registers**

| SPI0 BYTE ADDRESS | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x0204 7800 | SPIGCR0 | Global Control Register 0 |
| 0x0204 7804 | SPIGCR1 | Global Control Register 1 |
| 0x0204 7808 | SPIINT0 | Interrupt Register |
| 0x0204 780C | SPILVL | Interrupt Level Register |
| 0x0204 7810 | SPIFLG | Flag Register |
| 0x0204 7814 | SPIPC0 | Pin Control Register 0 (Pin Function) |
| 0x0204 7818 | SPIPC1 | Pin Control Register 1 (Pin Direction) |
| 0x0204 781C | SPIPC2 | Pin Control Register 2 (Pin Data In) |
| 0x0204 783C | SPIDAT1 | Shift Register 1 (with format select) |
| 0x0204 7840 | SPIBUF | Buffer Register |
| 0x0204 7844 | SPIEMU | Emulation Register |
| 0x0204 7848 | SPIDELAY | Delay Register |
| 0x0204 784C | SPIDEF | Default Chip Select Register |
| 0x0204 7850 | SPIFMT0 | Format Register 0 |
| 0x0204 7854 | SPIFMT1 | Format Register 1 |
| 0x0204 7858 | SPIFMT2 | Format Register 2 |
| 0x0204 785C | SPIFMT3 | Format Register 3 |
| 0x0204 7860 | TGINTVECT0 | Interrupt Vector for SPI INT0 |
| 0x0204 7864 | TGINTVECT1 | Interrupt Vector for SPI INT1 |

## 6.12.3 SPI Electrical Data/Timing

### 6.12.3.1 Serial Peripheral Interface (SPI) Timing

Table 6-52 assumes testing over recommended operating conditions (see Figure 6-30).

**Table 6-52. General Timing Requirements for SPIx Master Modes(1)**

| NO. | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|
| 1 | $t_{c(SPC)M}$ | Cycle Time, SPICLK, All Master Modes | | greater of 8P or 100 ns | 256P | ns |
| 2 | $t_{w(SPCH)M}$ | Pulse Width High, SPICLK, All Master Modes | | greater of 4P or 45 ns | | ns |
| 3 | $t_{w(SPCL)M}$ | Pulse Width Low, SPICLK, All Master Modes | | greater of 4P or 45 ns | | ns |
| 4 | $t_{d(SIMO\_SPC)M}$ | Delay, initial data bit valid on SPIDO to initial edge on SPICLK(2) | Polarity = 0, Phase = 0, to SPICLK rising | 4P | | ns |
| | | | Polarity = 0, Phase = 1, to SPICLK rising | $0.5t_{c(SPC)M}$ + 4P | | |
| | | | Polarity = 1, Phase = 0, to SPICLK falling | 4P | | |
| | | | Polarity = 1, Phase = 1, to SPICLK falling | $0.5t_{c(SPC)M}$ + 4P | | |
| 5 | $t_{d(SPC\_SIMO)M}$ | Delay, subsequent bits valid on SPIDO after transmit edge of SPICLK | Polarity = 0, Phase = 0, from SPICLK rising | | 15 | ns |
| | | | Polarity = 0, Phase = 1, from SPICLK falling | | 15 | |
| | | | Polarity = 1, Phase = 0, from SPICLK falling | | 15 | |
| | | | Polarity = 1, Phase = 1, from SPICLK rising | | 15 | |
| 6 | $t_{oh(SPC\_SIMO)M}$ | Output hold time, SPIDO valid after receive edge of SPICLK, except for final bit(3) | Polarity = 0, Phase = 0, from SPICLK falling | $0.5t_{c(SPC)M}$ - 10 | | ns |
| | | | Polarity = 0, Phase = 1, from SPICLK rising | $0.5t_{c(SPC)M}$ - 10 | | |
| | | | Polarity = 1, Phase = 0, from SPICLK rising | $0.5t_{c(SPC)M}$ - 10 | | |
| | | | Polarity = 1, Phase = 1, from SPICLK falling | $0.5t_{c(SPC)M}$ - 10 | | |
| 7 | $t_{su(SOMI\_SPC)M}$ | Input Setup Time, SPIDI valid before receive edge of SPICLK | Polarity = 0, Phase = 0, to SPICLK falling | 0.5P + 15 | | ns |
| | | | Polarity = 0, Phase = 1, to SPICLK rising | 0.5P + 15 | | |
| | | | Polarity = 1, Phase = 0, to SPICLK rising | 0.5P + 15 | | |
| | | | Polarity = 1, Phase = 1, to SPICLK falling | 0.5P + 15 | | |
| 8 | $t_{ih(SPC\_SOMI)M}$ | Input Hold Time, SPIDI valid after receive edge of SPICLK | Polarity = 0, Phase = 0, from SPICLK falling | 0.5P + 5 | | ns |
| | | | Polarity = 0, Phase = 1, from SPICLK rising | 0.5P + 5 | | |
| | | | Polarity = 1, Phase = 0, from SPICLK rising | 0.5P + 5 | | |
| | | | Polarity = 1, Phase = 1, from SPICLK falling | 0.5P + 5 | | |

(1) P = SYSCLK3 period
(2) First bit may be MSB or LSB depending upon SPI configuration. MO(0) refers to first bit and MO(n) refers to last bit output on SPIDO. MI(0) refers to the first bit input and MI(n) refers to the last bit input on SPIDI.
(3) The final data bit will be held on the SPIDO pin until the SPIDAT1 register is written with new data.

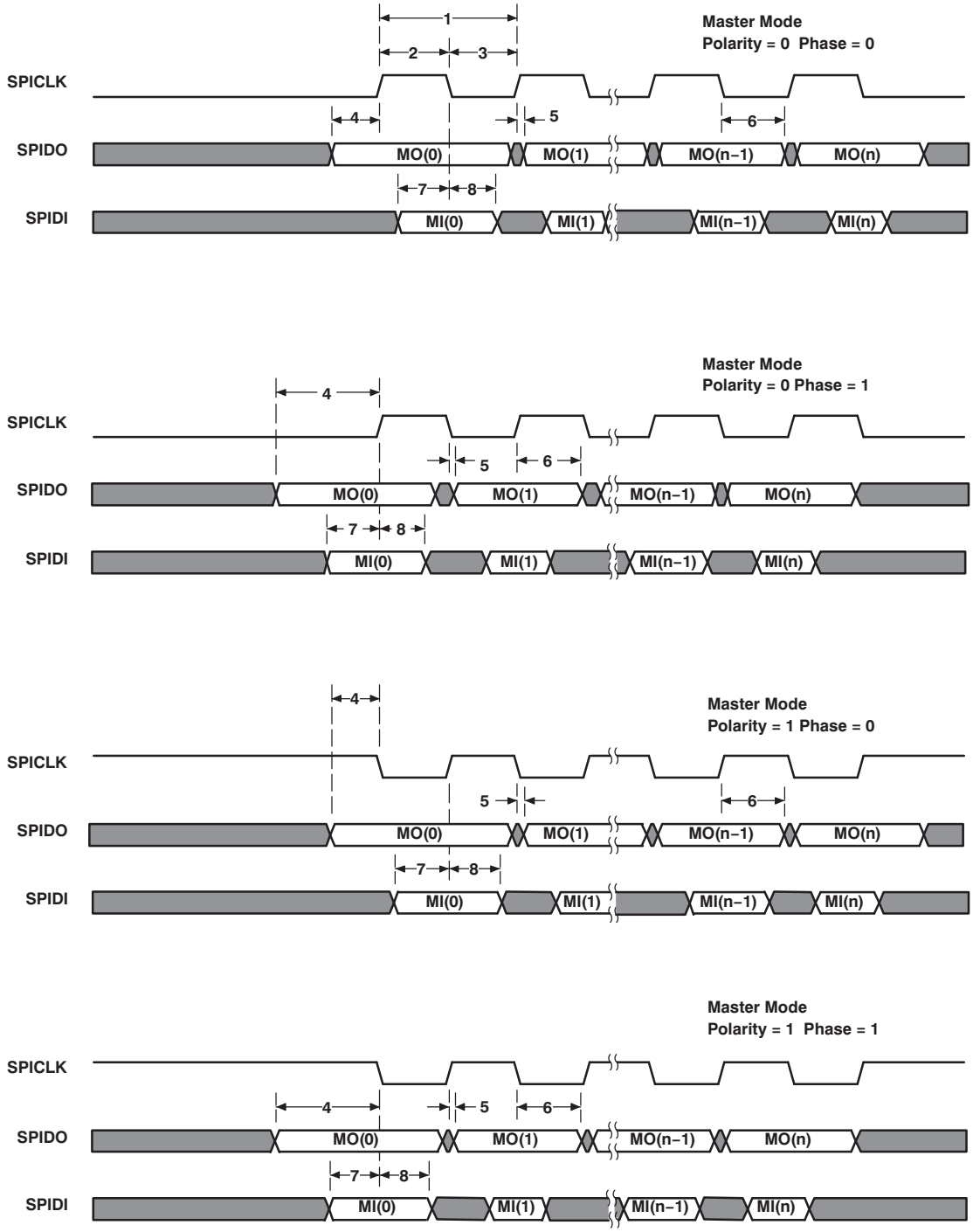



**Figure 6-30. SPI Timings—Master Mode**

## 6.13 Inter-Integrated Circuit (I2C)

The inter-integrated circuit (I2C) module provides an interface between the C6452 device and other devices compliant with Philips Semiconductors Inter-IC bus (I2C-bus™) specification version 2.1. External components attached to this 2-wire serial bus can transmit/receive up to 8-bit data to/from the DSP through the I2C module. The I2C port *does not* support CBUS-compatible devices.

The I2C port supports:

- Compatible with Philips I2C Specification Revision 2.1 (January 2000)
- Fast Mode up to 400 Kbps (no fail-safe I/O buffers)
- Noise Filter to Remove Noise 50 ns or less
- Seven- and Ten-Bit Device Addressing Modes
- Master (Transmit/Receive) and Slave (Transmit/Receive) Functionality
- Events: DMA, Interrupt, or Polling
- Slew-Rate Limited Open-Drain Output Buffers

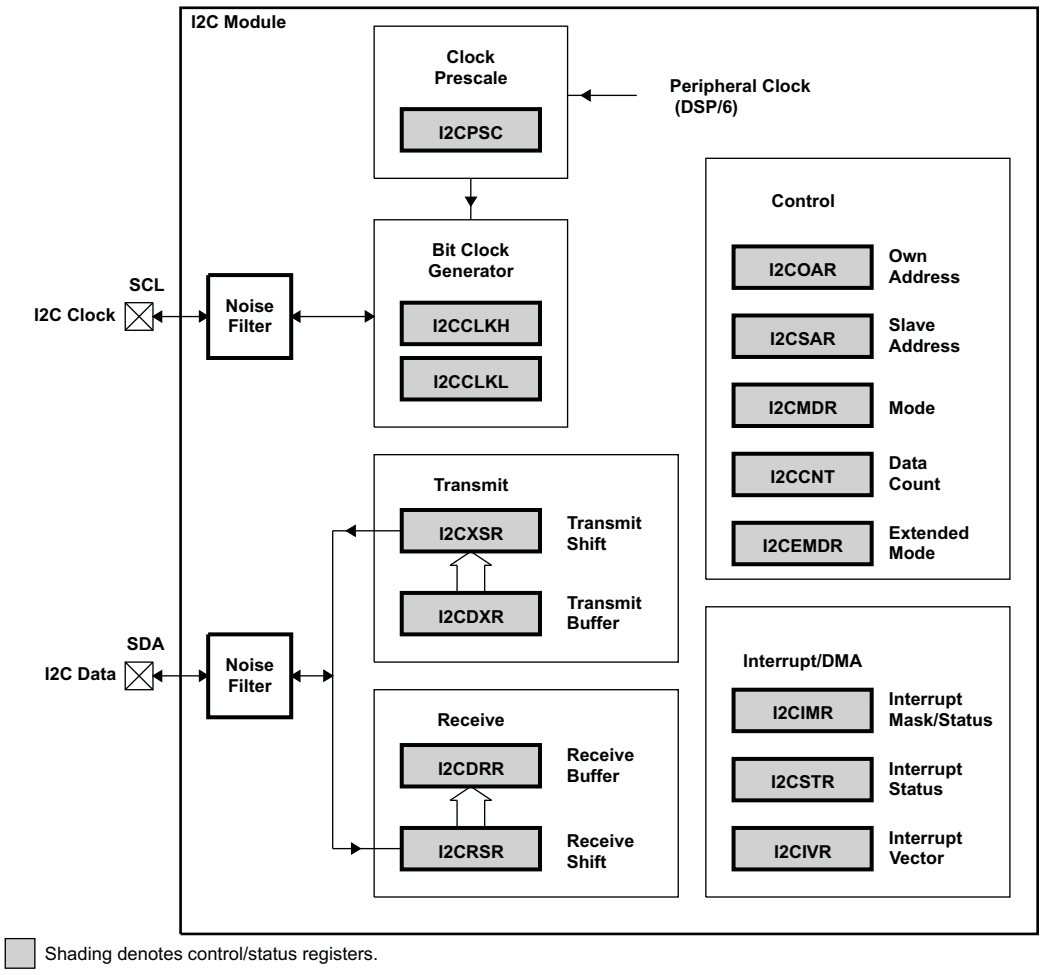



**Figure 6-31. I2C Module Block Diagram**

For more detailed information on the I2C peripheral, see the *TMS320C6452 DSP Inter-Integrated Circuit (I2C) Module User's Guide* (literature number SPRUF94).

### 6.13.1 I2C Peripheral Register Description(s)

**Table 6-53. I2C Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 7C00 | ICOAR | I2C Own Address Register |
| 0x0204 7C04 | ICIMR | I2C Interrupt Mask Register |
| 0x0204 7C08 | ICSTR | I2C Interrupt Status Register |
| 0x0204 7C0C | ICCLKL | I2C Clock Divider Low Register |
| 0x0204 7C10 | ICCLKH | I2C Clock Divider High Register |
| 0x0204 7C14 | ICCNT | I2C Data Count Register |
| 0x0204 7C18 | ICDRR | I2C Data Receive Register |
| 0x0204 7C1C | ICSAR | I2C Slave Address Register |
| 0x0204 7C20 | ICDXR | I2C Data Transmit Register |
| 0x0204 7C24 | ICMDR | I2C Mode Register |
| 0x0204 7C28 | ICIVR | I2C Interrupt Vector Register |
| 0x0204 7C2C | ICEMDR | I2C Extended Mode Register |
| 0x0204 7C30 | ICPSC | I2C Prescaler Register |
| 0x0204 7C34 | ICDMAC | I2C DMA Control Register |

## 6.13.2 I2C Electrical Data/Timing

### 6.13.2.1 Inter-Integrated Circuits (I2C) Timing

**Table 6-54. Timing Requirements for I2C Timings(1) (see Figure 6-32)**

| NO. | | | 720 900 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | STANDARD MODE | | FAST MODE | | |
| | | | MIN | MAX | MIN | MAX | |
| 1 | $t_{c(SCL)}$ | Cycle time, SCL | 10 | | 2.5 | | µs |
| 2 | $t_{su(SCLH-SDAL)}$ | Setup time, SCL high before SDA low (for a repeated START condition) | 4.7 | | 0.6 | | µs |
| 3 | $t_{h(SCLL-SDAL)}$ | Hold time, SCL low after SDA low (for a START and a repeated START condition) | 4 | | 0.6 | | µs |
| 4 | $t_{w(SCLL)}$ | Pulse duration, SCL low | 4.7 | | 1.3 | | µs |
| 5 | $t_{w(SCLH)}$ | Pulse duration, SCL high | 4 | | 0.6 | | µs |
| 6 | $t_{su(SDAV-SCLH)}$ | Setup time, SDA valid before SCL high | 250 | | 100(2) | | ns |
| 7 | $t_{h(SDA-SCLL)}$ | Hold time, SDA valid after SCL low | 0(3) | | 0(3) | 0.9(4) | µs |
| 8 | $t_{w(SDAH)}$ | Pulse duration, SDA high between STOP and START conditions | 4.7 | | 1.3 | | µs |
| 9 | $t_{r(SDA)}$ | Rise time, SDA | | 1000 | 20 + 0.1$C_b$ (5) | 300 | ns |
| 10 | $t_{r(SCL)}$ | Rise time, SCL | | 1000 | 20 + 0.1$C_b$ (5) | 300 | ns |
| 11 | $t_{f(SDA)}$ | Fall time, SDA | | 300 | 20 + 0.1$C_b$ (5) | 300 | ns |
| 12 | $t_{f(SCL)}$ | Fall time, SCL | | 300 | 20 + 0.1$C_b$ (5) | 300 | ns |
| 13 | $t_{su(SCLH-SDAH)}$ | Setup time, SCL high before SDA high (for STOP condition) | 4 | | 0.6 | | µs |
| 14 | $t_{w(SP)}$ | Pulse duration, spike (must be suppressed) | | | 0 | 50 | ns |
| 15 | $C_b$ (5) | Capacitive load for each bus line | | 400 | | 400 | pF |

(1) The I2C pins SDA and SCL do not feature fail-safe I/O buffers. These pins could potentially draw current when the device is powered down.
(2) A Fast-mode I2C-bus™ device can be used in a standard-mode I2C-bus system, but the requirement $t_{su(SDA-SCLH)} \geq 250$ ns must then be met. This will be the case automatically if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_r$ max + $t_{su(SDA-SCLH)}$ = 1000 + 250 = 1250 ns (according to the Standard-mode I2C-Bus Specification) before the SCL line is released.
(3) A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the $V_{IHmin}$ of the SCL signal) to bridge the undefined region of the falling edge of SCL.
(4) The maximum $t_{h(SDA-SCLL)}$ has to be met only if the device does not stretch the low period [$t_{w(SCLL)}$] of the SCL signal.
(5) $C_b$ = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times are allowed.

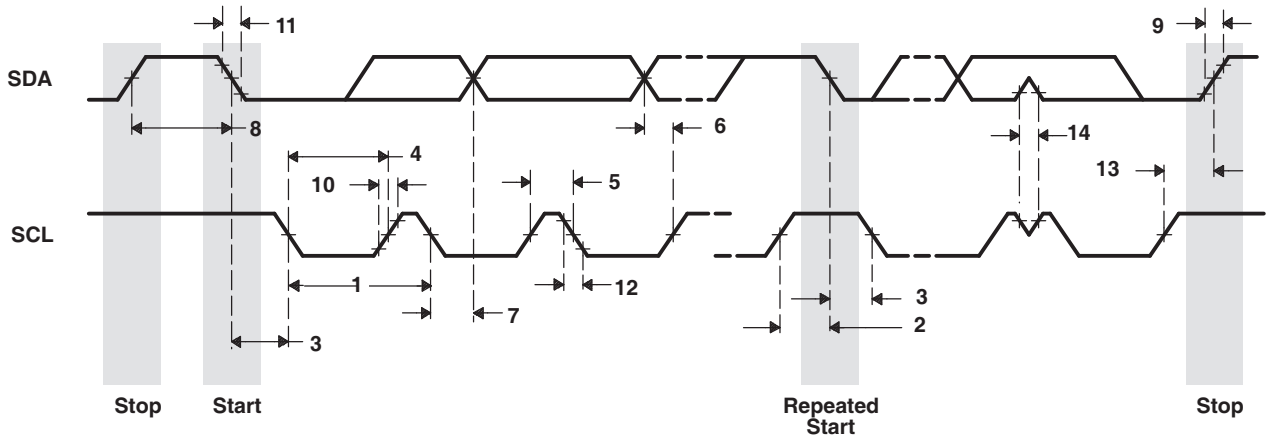



**Figure 6-32. I2C Receive Timings**

**Table 6-55. Switching Characteristics for I2C Timings[1] (see Figure 6-33)**

| NO. | PARAMETER | | 720 900 | | | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | STANDARD MODE | | FAST MODE | | |
| | | | MIN | MAX | MIN | MAX | |
| 16 | $t_{c(SCL)}$ | Cycle time, SCL | 10 | | 2.5 | | µs |
| 17 | $t_{d(SCLH-SDAL)}$ | Delay time, SCL high to SDA low (for a repeated START condition) | 4.7 | | 0.6 | | µs |
| 18 | $t_{d(SDAL-SCLL)}$ | Delay time, SDA low to SCL low (for a START and a repeated START condition) | 4 | | 0.6 | | µs |
| 19 | $t_{w(SCLL)}$ | Pulse duration, SCL low | 4.7 | | 1.3 | | µs |
| 20 | $t_{w(SCLH)}$ | Pulse duration, SCL high | 4 | | 0.6 | | µs |
| 21 | $t_{d(SDAV-SCLH)}$ | Delay time, SDA valid to SCL high | 250 | | 100 | | ns |
| 22 | $t_{v(SCLL-SDAV)}$ | Valid time, SDA valid after SCL low | 0 | | 0 | 0.9 | µs |
| 23 | $t_{w(SDAH)}$ | Pulse duration, SDA high between STOP and START conditions | 4.7 | | 1.3 | | µs |
| 24 | $t_{r(SDA)}$ | Rise time, SDA | | 1000 | $20 + 0.1C_b$ (2) | 300 | ns |
| 25 | $t_{r(SCL)}$ | Rise time, SCL | | 1000 | $20 + 0.1C_b$ (2) | 300 | ns |
| 26 | $t_{f(SDA)}$ | Fall time, SDA | | 300 | $20 + 0.1C_b$ (2) | 300 | ns |
| 27 | $t_{f(SCL)}$ | Fall time, SCL | | 300 | $20 + 0.1C_b$ (2) | 300 | ns |
| 28 | $t_{d(SCLH-SDAH)}$ | Delay time, SCL high to SDA high (for STOP condition) | 4 | | 0.6 | | µs |
| 29 | $C_p$ | Capacitance for each I2C pin | | 10 | | 10 | pF |

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

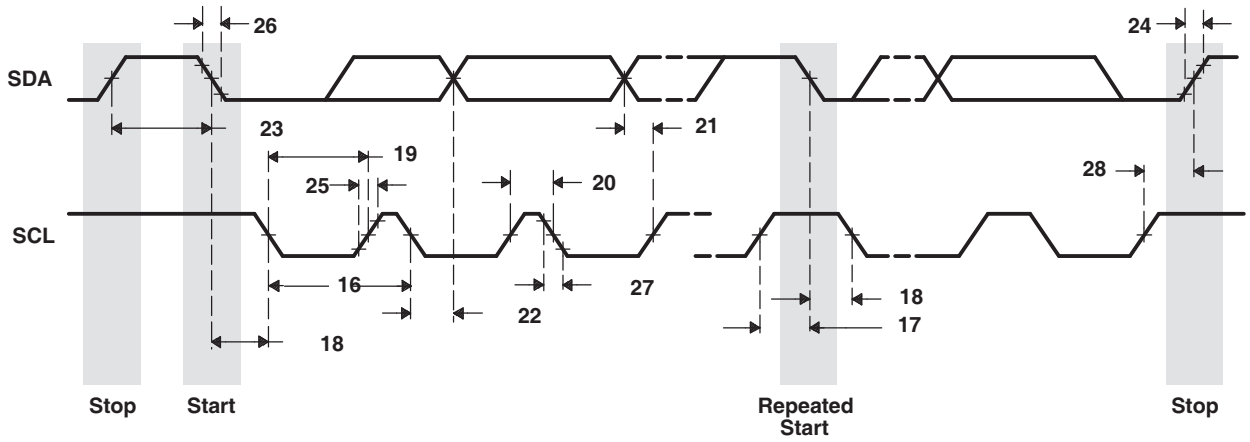



**Figure 6-33. I2C Transmit Timings**

## 6.14 Host-Port Interface (HPI) Peripheral

### 6.14.1 HPI Device-Specific Information

The device includes a user-configurable 16-bit or 32-bit host-port interface (HPI16/HPI32). The AEA14 pin controls the HPI_WIDTH, allowing the user to configure the HPI as a 16-bit or 32-bit peripheral.

Software handshaking via the HRDY bit of the Host Port Control Register (HPIC) is not supported.

An HPI boot is terminated using a DSP interrupt. The DSP interrupt is registered in bit 0 (channel 0) of the EDMA Event Register (ER). This event must be cleared by software before triggering transfers on DMA channel 0.

### 6.14.2 HPI Peripheral Register Description(s)

**Table 6-56. HPI Control Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|---|---|---|---|
| 0x0200 0000 | PID | Peripheral Identification Register | |
| 0x0200 0004 | PWREMU_MGMT | HPI power and emulation management register | PWREMU_MGMT has both host/CPU read/write access. |
| 0x0200 0008 - 0x0200 0024 | - | Reserved | |
| 0x0200 0028 | - | Reserved | |
| 0x0200 002C | - | Reserved | |
| 0x0200 0030 | HPIC | HPI control register | The host and the CPU have read/write access to the HPIC register.(1) |
| 0x0200 0034 | HPIA (HPIAW)(2) | HPI address register (Write) | The host has read/write access to the HPIA registers. The CPU has read access only to the HPIA registers. |
| 0x0200 0038 | HPIA (HPIAR)(2) | HPI address register (Read) | |
| 0x0200 003C - 0x0200 007F | - | Reserved | |

(1) The CPU can write 1 to the $\overline{\text{HINT}}$ bit to generate an interrupt to the host and it can write 1 to the DSPINT bit to clear/acknowledge an interrupt from the host.

(2) There are two 32-bit HPIA registers: HPIAR for read operations and HPIAW for write operations. The HPI can be configured such that HPIAR and HPIAW act as a single 32-bit HPIA (single-HPIA mode) or as two separate 32-bit HPIAs (dual-HPIA mode) from the perspective of the host. The CPU can access HPIAW and HPIAR independently. For details about the HPIA registers and their modes, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87).

### 6.14.3 HPI Electrical Data/Timing

**Table 6-57. Timing Requirements for Host-Port Interface Cycles(1) (2) (see Figure 6-34 through Figure 6-41)**

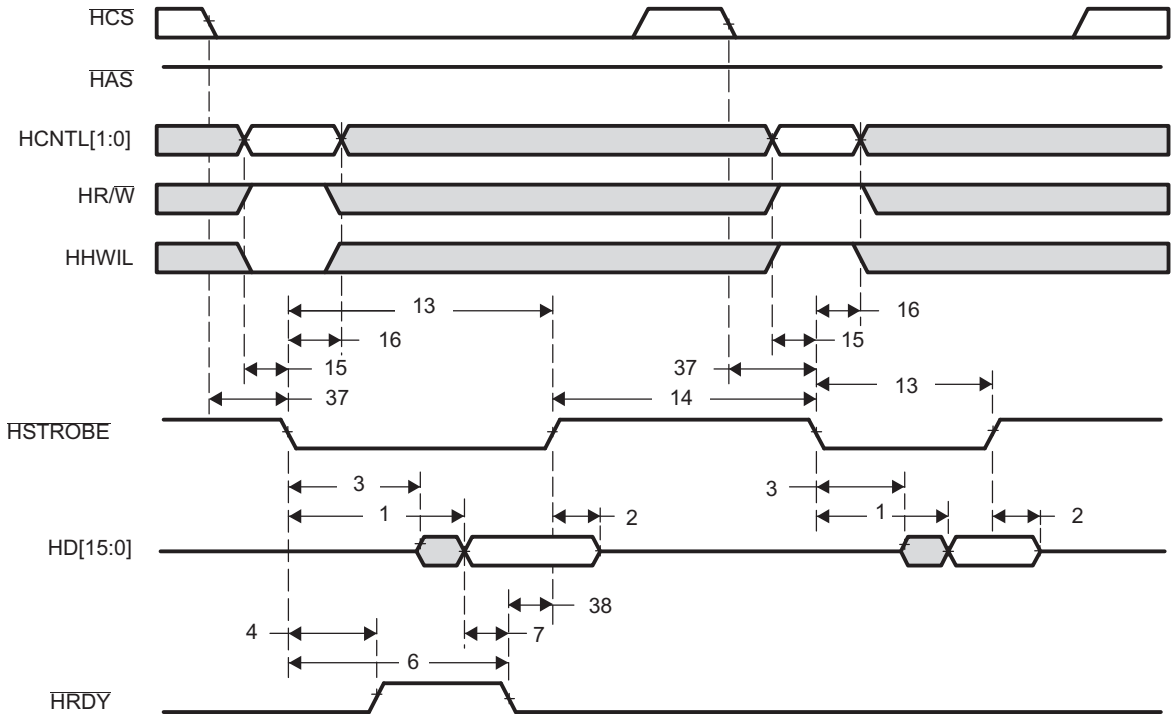
| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 9 | $t_{su(HASL-HSTBL)}$ | Setup time, $\overline{HAS}$ low before $\overline{HSTROBE}$ low | 5 | | ns |
| 10 | $t_{h(HSTBL-HASL)}$ | Hold time, $\overline{HAS}$ low after $\overline{HSTROBE}$ low | 2 | | ns |
| 11 | $t_{su(SELV-HASL)}$ | Setup time, select signals(3) valid before $\overline{HAS}$ low | 5 | | ns |
| 12 | $t_{h(HASL-SELV)}$ | Hold time, select signals(3) valid after $\overline{HAS}$ low | 5 | | ns |
| 13 | $t_{w(HSTBL)}$ | Pulse duration, $\overline{HSTROBE}$ low | 2M | | ns |
| 14 | $t_{w(HSTBH)}$ | Pulse duration, $\overline{HSTROBE}$ high between consecutive accesses | 2M | | ns |
| 15 | $t_{su(SELV-HSTBL)}$ | Setup time, select signals(3) valid before $\overline{HSTROBE}$ low | 5 | | ns |
| 16 | $t_{h(HSTBL-SELV)}$ | Hold time, select signals(3) valid after $\overline{HSTROBE}$ low | 5 | | ns |
| 17 | $t_{su(HDV-HSTBH)}$ | Setup time, host data valid before $\overline{HSTROBE}$ high | 5 | | ns |
| 18 | $t_{h(HSTBH-HDV)}$ | Hold time, host data valid after $\overline{HSTROBE}$ high | 1 | | ns |
| 37 | $t_{su(HCSL-HSTBL)}$ | Setup time, $\overline{HCS}$ low before $\overline{HSTROBE}$ low | 0 | | ns |
| 38 | $t_{h(HRDYL-HSTBL)}$ | Hold time, $\overline{HSTROBE}$ low after $\overline{HRDY}$ low. $\overline{HSTROBE}$ should not be inactivated until $\overline{HRDY}$ is active (low); otherwise, HPI writes will not complete properly. | 1.1 | | ns |

(1) $\overline{HSTROBE}$ refers to the following logical operation on $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$: [NOT ($\overline{HDS1}$ XOR $\overline{HDS2}$)] OR $\overline{HCS}$.
(2) M = SYSCLK3 period = 6/CPU clock frequency in ns.
(3) Select signals include: HCNTL[1:0] and HR/$\overline{W}$. For HPI16 mode only, select signals also include HHWIL.

**Table 6-58. Switching Characteristics for Host-Port Interface Cycles(1) (2) (see Figure 6-34 through Figure 6-41)**

| NO. | PARAMETER | | | 720 900 MIN | 720 900 MAX | UNIT |
|---|---|---|---|---|---|---|
| 1 | $t_{d(HSTBL-HDV)}$ | Delay time, $\overline{HSTROBE}$ low to DSP data valid | Case 1. HPIC or HPIA read | 5 | 15 | ns |
| | | | Case 2. HPID read with no auto-increment(3) | | 9 × M + 20 | |
| | | | Case 3. HPID read with auto-increment and read FIFO initially empty(3) | | 9 × M + 20 | |
| | | | Case 4. HPID read with auto-increment and data previously prefetched into the read FIFO | 5 | 15 | |
| 2 | $t_{dis(HSTBH-HDV)}$ | Disable time, HD high-impedance from $\overline{HSTROBE}$ high | | 1 | 4 | ns |
| 3 | $t_{en(HSTBL-HD)}$ | Enable time, HD driven from $\overline{HSTROBE}$ low | | 3 | 15 | ns |
| 4 | $t_{d(HSTBL-HRDYH)}$ | Delay time, $\overline{HSTROBE}$ low to $\overline{HRDY}$ high | | | 12 | ns |
| 5 | $t_{d(HSTBH-HRDYH)}$ | Delay time, $\overline{HSTROBE}$ high to $\overline{HRDY}$ high | | | 12 | ns |
| 6 | $t_{d(HSTBL-HRDYL)}$ | Delay time, $\overline{HSTROBE}$ low to $\overline{HRDY}$ low | Case 1. HPID read with no auto-increment(3) | | 10 × M + 20 | ns |
| | | | Case 2. HPID read with auto-increment and read FIFO initially empty(3) | | 10 × M + 20 | |
| 7 | $t_{d(HDV-HRDYL)}$ | Delay time, HD valid to $\overline{HRDY}$ low | | 0 | | ns |
| 34 | $t_{d(DSH-HRDYL)}$ | Delay time, $\overline{HSTROBE}$ high to $\overline{HRDY}$ low | Case 1. HPIA write(3) | | 5 × M + 20 | ns |
| | | | Case 2. HPID write with no auto-increment(3) | | 5 × M + 20 | |
| 35 | $t_{d(HSTBL-HRDYL)}$ | Delay time, $\overline{HSTROBE}$ low to $\overline{HRDY}$ low for HPIA write and FIFO not empty(3) | | | 40 × M + 20 | ns |
| 36 | $t_{d(HASL-HRDYH)}$ | Delay time, $\overline{HAS}$ low to $\overline{HRDY}$ high | | | 12 | ns |

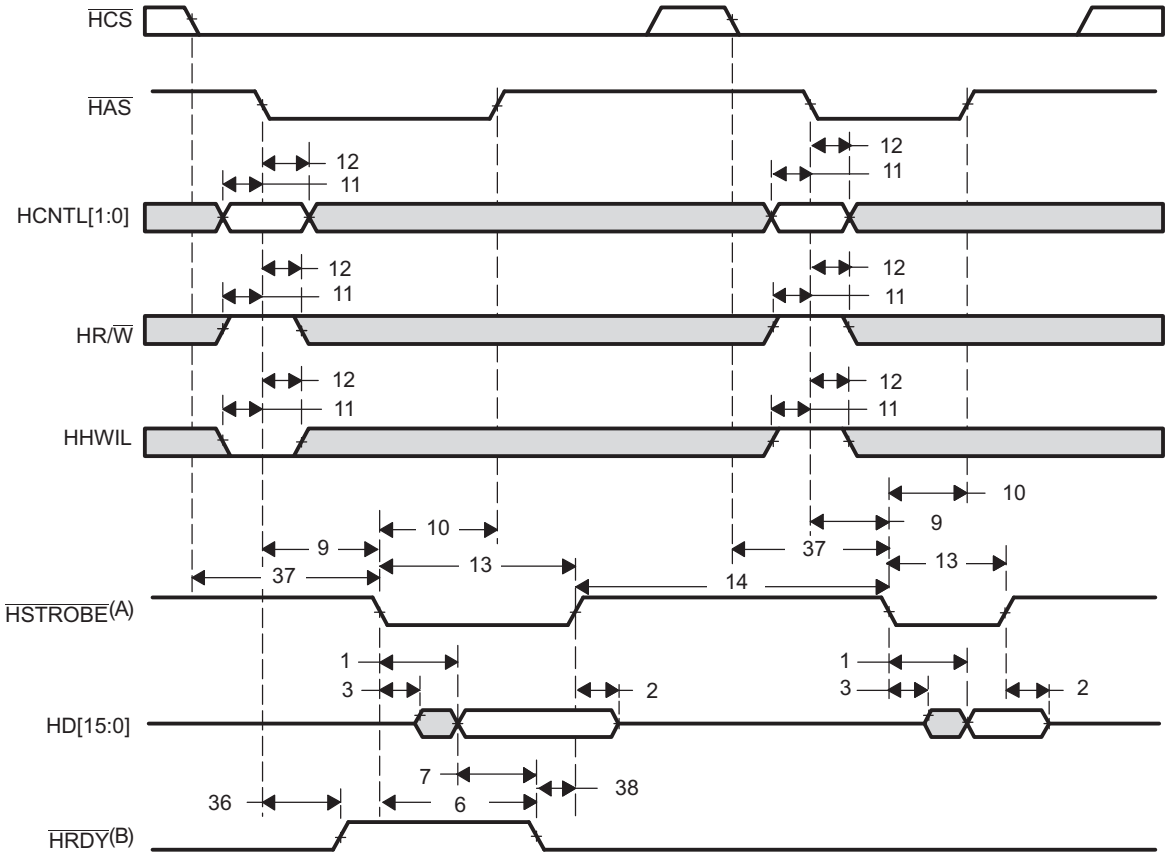
(1) M = SYSCLK3 period = 6/CPU clock frequency in ns.
(2) $\overline{HSTROBE}$ refers to the following logical operation on $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$: [NOT($\overline{HDS1}$ XOR $\overline{HDS2}$)] OR $\overline{HCS}$.
(3) Assumes the HPI is accessing L2/L1 memory and no other master is accessing the same memory location.




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87).

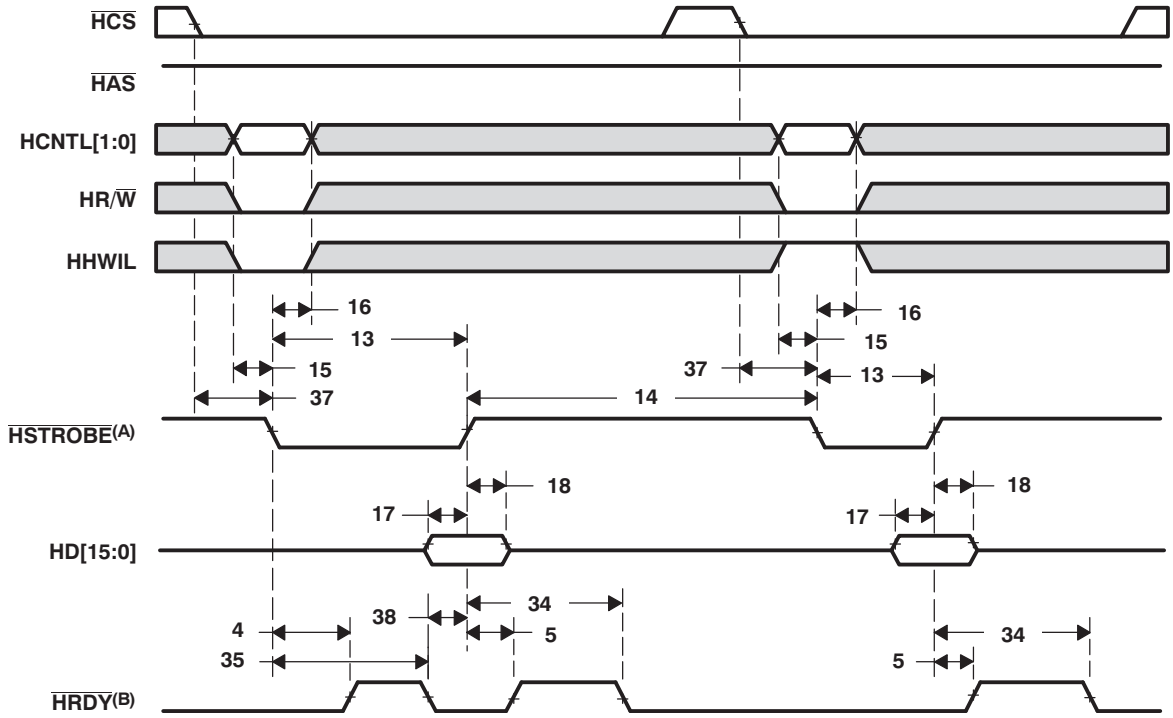
**Figure 6-34. HPI16 Read Timing (HAS Not Used, Tied High)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87).

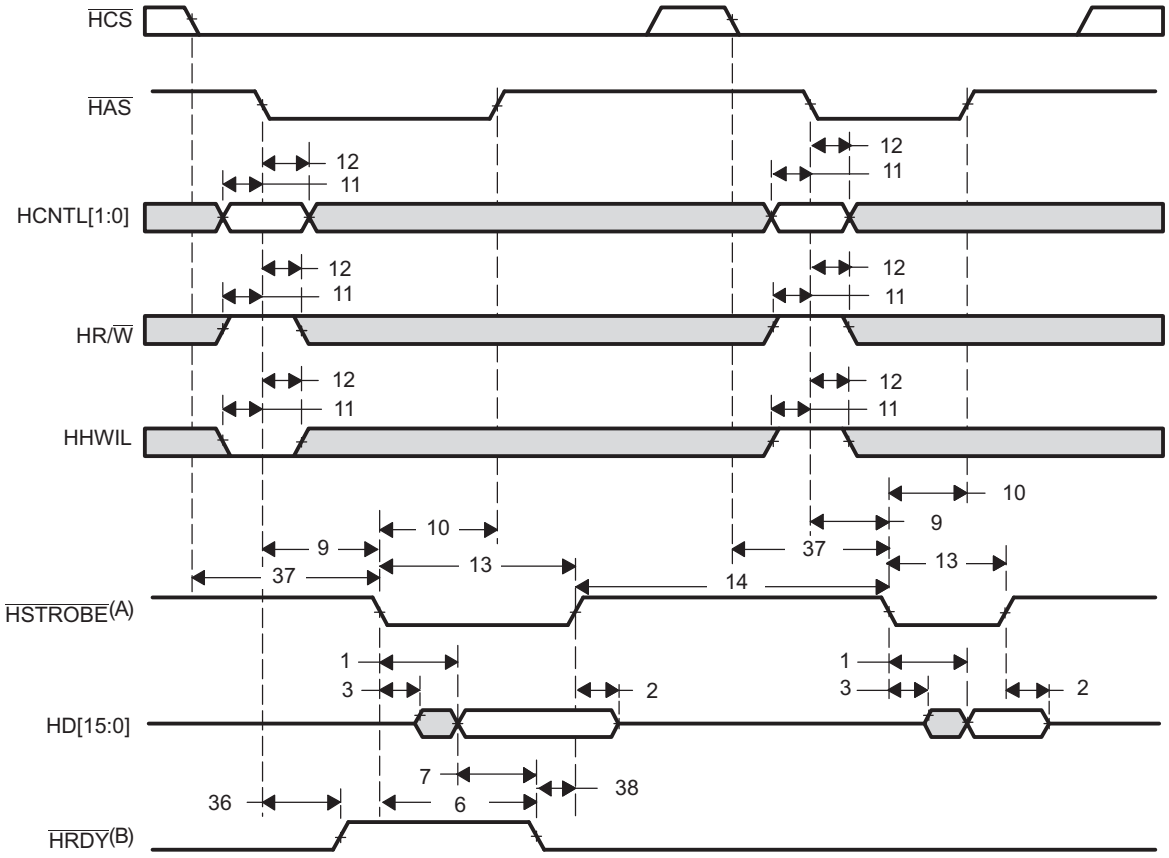
**Figure 6-35. HPI16 Read Timing (HAS Used)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87).

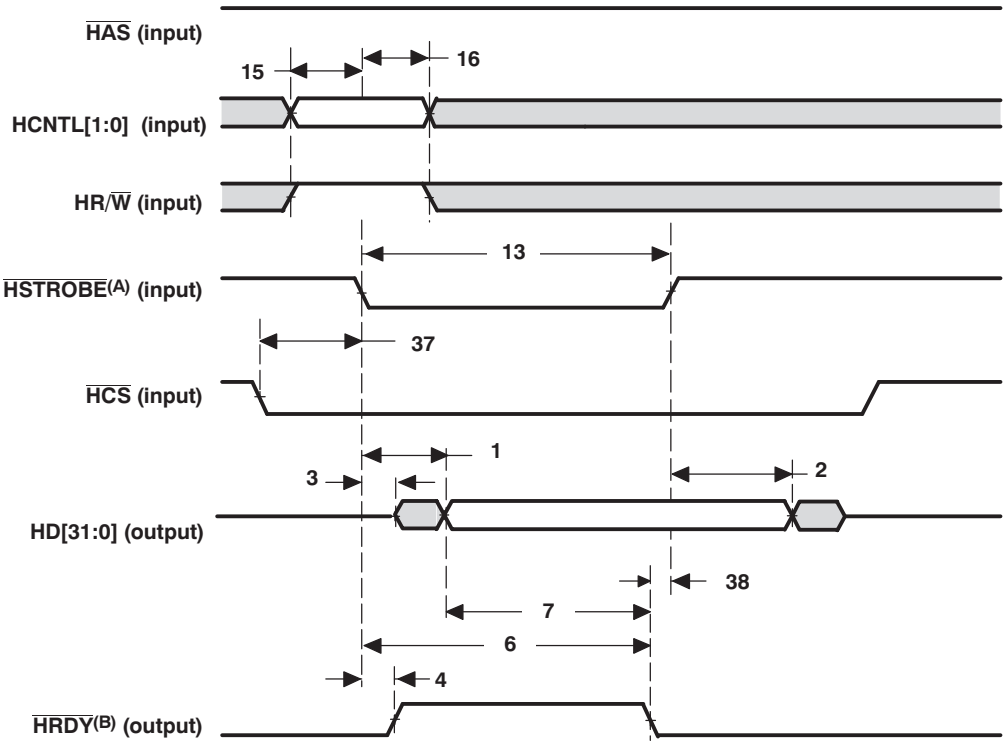
**Figure 6-36. HPI16 Write Timing (HAS Not Used, Tied High)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87).

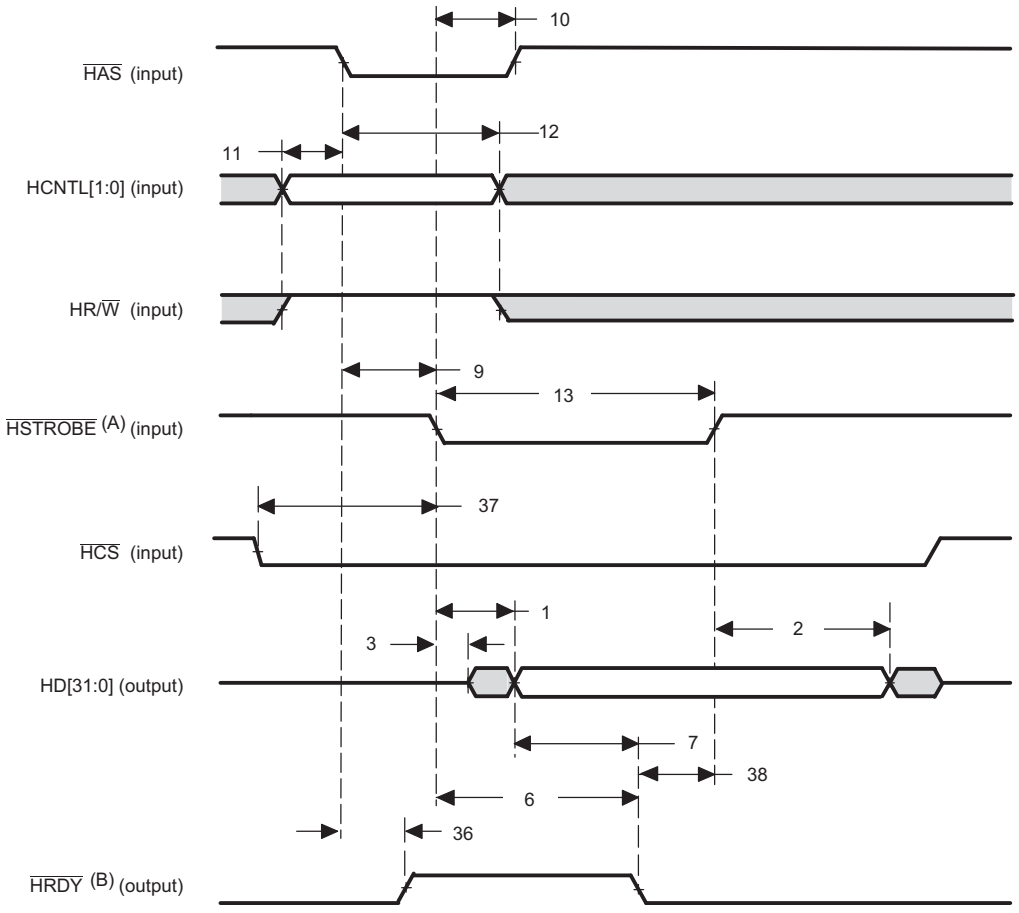
**Figure 6-37. HPI16 Write Timing (HAS Used)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87)

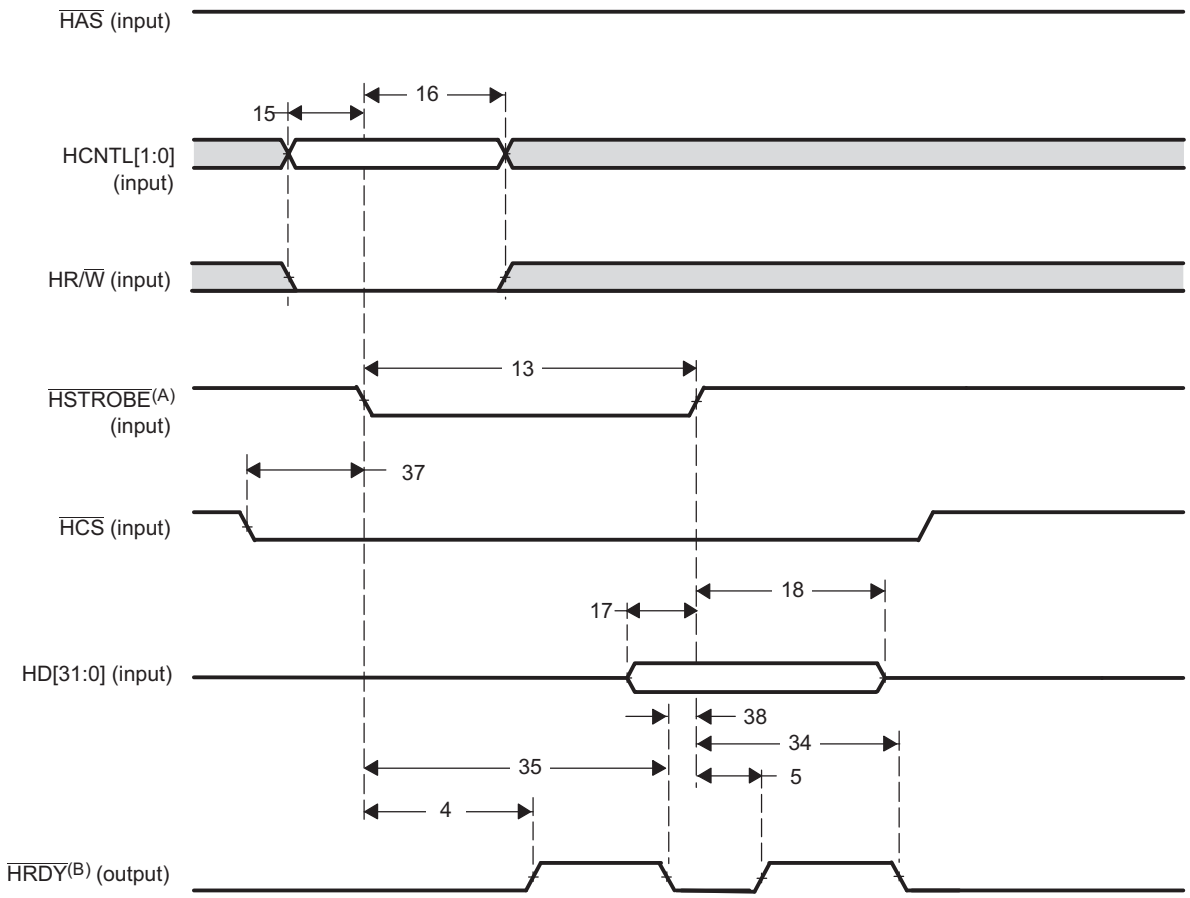
**Figure 6-38. HPI32 Read Timing (HAS Not Used, Tied High)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87)

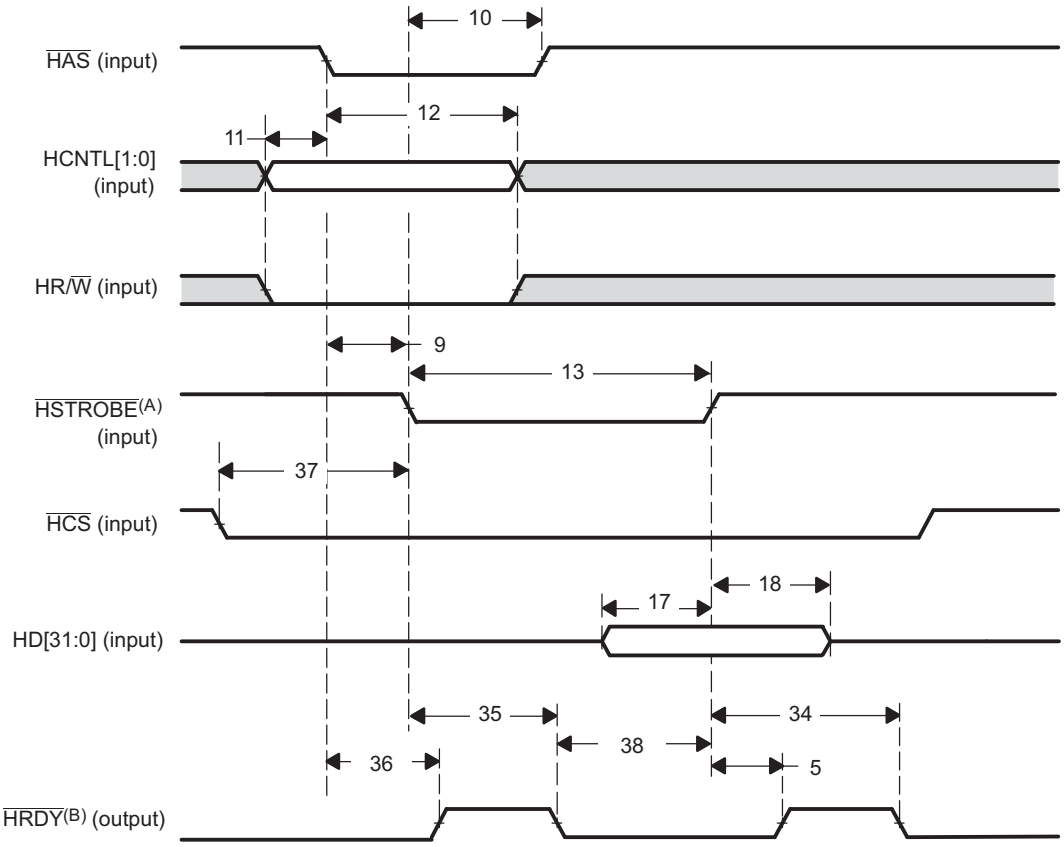
**Figure 6-39. HPI32 Read Timing (HAS Used)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87)

**Figure 6-40. HPI32 Write Timing (HAS Not Used, Tied High)**




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

B. Depending on the type of write or read operation (HPID without auto-incrementing; HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C6452 DSP Host Port Interface (HPI) User's Guide* (literature number SPRUF87)

**Figure 6-41. HPI32 Write Timing (HAS Used)**

## 6.15 Peripheral Component Interconnect (PCI)

The device supports connections to a PCI backplane via the integrated PCI master/slave bus interface. The PCI port interfaces to DSP internal resources via the data switched central resource. .

For more detailed information on the PCI port peripheral module, see the *TMS320C6452 Peripheral Component Interconnect (PCI) User's Guide* (literature number SPRUF86).

### *6.15.1 PCI Device-Specific Information*

The PCI peripheral conforms to the *PCI Local Bus Specification* (version 2.3). The PCI peripheral can act both as a PCI bus master and as a target. It supports PCI bus operation of speeds up to 66 MHz and uses a 32-bit data/address bus.

The pins of the PCI peripheral are multiplexed with the pins of the HPI, and GPIO peripherals. PCI functionality for these pins is controlled (enabled/disabled) by the UHPIEN pin (H2). The maximum speed of the PCI, 33 MHz or 66 MHz, is controlled through the PCI66 pin (G5). For more detailed information on the peripheral control, see Section 3.

The device provides an initialization mechanism through which the default values for some of the PCI configuration registers can be read from an I2C EEPROM. Table 6-59 shows the registers which can be initialized through the PCI auto-initialization. Also shown is the default value of these registers when PCI auto-initialization is not used. PCI auto-initialization is enabled by selecting PCI boot with auto-initialization. For more information on this feature, see the *TMS320C6452 Peripheral Component Interconnect (PCI) User's Guide* (literature number SPRUF86).

**Table 6-59. Default Values for PCI Configuration Registers**

| REGISTER | DEFAULT VALUE |
|---|---|
| Vendor ID/Device ID Register (PCIVENDEV) | 104C B003h |
| Class Code/Revision ID Register (PCICLREV) | 0000 0001h |
| Subsystem Vendor ID/Subsystem ID Register (PCISUBID) | 0000 0000h |
| Max Latency/Min Grant/Interrupt Pin/Interrupt Line Register (PCILGINT) | 0000 0100h |

### 6.15.2 PCI Peripheral Register Description(s)

**Table 6-60. PCI Configuration Registers**

| PCI HOST ACCESS HEX ADDRESS OFFSET | ACRONYM | PCI HOST ACCESS REGISTER NAME |
|---|---|---|
| 0x00 | PCIVENDEV | Vendor ID/Device ID |
| 0x04 | PCICSR | Command/Status |
| 0x08 | PCICLREV | Class Code/Revision ID |
| 0x0C | PCICLINE | BIST/Header Type/Latency Timer/Cacheline Size |
| 0x10 | PCIBAR0 | Base Address 0 |
| 0x14 | PCIBAR1 | Base Address 1 |
| 0x18 | PCIBAR2 | Base Address 2 |
| 0x1C | PCIBAR3 | Base Address 3 |
| 0x20 | PCIBAR4 | Base Address 4 |
| 0x24 | PCIBAR5 | Base Address 5 |
| 0x28 - 0x2B | - | Reserved |
| 0x2C | PCISUBID | Subsystem Vendor ID/Subsystem ID |
| 0x30 | - | Reserved |
| 0x34 | PCICPBPTR | Capabilities Pointer |
| 0x38 - 0x3B | - | Reserved |
| 0x3C | PCILGINT | Max Latency/Min Grant/Interrupt Pin/Interrupt Line |
| 0x40 - 0x7F | - | Reserved |

**Table 6-61. PCI Back End Configuration Registers**

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---|---|---|
| 0x0204 8400 - 0x0204 840F | - | Reserved |
| 0x0204 8410 | PCISTATSET | PCI Status Set Register |
| 0x0204 8414 | PCISTATCLR | PCI Status Clear Register |
| 0x0204 8418 - 0x0204 841F | - | Reserved |
| 0x0204 8420 | PCIHINTSET | PCI Host Interrupt Enable Set Register |
| 0x0204 8424 | PCIHINTCLR | PCI Host Interrupt Enable Clear Register |
| 0x0204 8428 - 0x0204 842F | - | Reserved |
| 0x0204 8430 | PCIBINTSET | PCI Back End Application Interrupt Enable Set Register |
| 0x0204 8434 | PCIBINTCLR | PCI Back End Application Interrupt Enable Clear Register |
| 0x0204 8438 | PCIBCLKMGT | PCI Back End Application Clock Management Register |
| 0x0204 843C - 0x0204 84FF | - | Reserved |
| 0x0204 8500 | PCIVENDEVMIR | PCI Vendor ID/Device ID Mirror Register |
| 0x0204 8504 | PCICSRMIR | PCI Command/Status Mirror Register |
| 0x0204 8508 | PCICLREVMIR | PCI Class Code/Revision ID Mirror Register |
| 0x0204 850C | PCICLINEMIR | PCI BIST/Header Type/Latency Timer/Cacheline Size Mirror Register |
| 0x0204 8510 | PCIBAR0MSK | PCI Base Address Mask Register 0 |
| 0x0204 8514 | PCIBAR1MSK | PCI Base Address Mask Register 1 |
| 0x0204 8518 | PCIBAR2MSK | PCI Base Address Mask Register 2 |
| 0x0204 851C | PCIBAR3MSK | PCI Base Address Mask Register 3 |
| 0x0204 8520 | PCIBAR4MSK | PCI Base Address Mask Register 4 |
| 0x0204 8524 | PCIBAR5MSK | PCI Base Address Mask Register 5 |
| 0x0204 8528 - 0x0204 852B | - | Reserved |
| 0x0204 852C | PCISUBIDMIR | PCI Subsystem Vendor ID/Subsystem ID Mirror Register |
| 0x0204 8530 | - | Reserved |

**Table 6-61. PCI Back End Configuration Registers (continued)**

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---|---|---|
| 0x0204 8534 | PCICPBPTRMIR | PCI Capabilities Pointer Mirror Register |
| 0x0204 8538 - 0x0204 853B | - | Reserved |
| 0x0204 853C | PCILGINTMIR | PCI Max Latency/Min Grant/Interrupt Pin/Interrupt Line Mirror Register |
| 0x0204 8540 - 0x0204 857F | - | Reserved |
| 0x0204 8580 | PCISLVCNTL | PCI Slave Control Register |
| 0x0204 8584 - 0x0204 85BF | - | Reserved |
| 0x0204 85C0 | PCIBAR0TRL | PCI Slave Base Address 0 Translation Register |
| 0x0204 85C4 | PCIBAR1TRL | PCI Slave Base Address 1 Translation Register |
| 0x0204 85C8 | PCIBAR2TRL | PCI Slave Base Address 2 Translation Register |
| 0x0204 85CC | PCIBAR3TRL | PCI Slave Base Address 3 Translation Register |
| 0x0204 85D0 | PCIBAR4TRL | PCI Slave Base Address 4 Translation Register |
| 0x0204 85D4 | PCIBAR5TRL | PCI Slave Base Address 5 Translation Register |
| 0x0204 85D8 - 0x0204 85DF | - | Reserved |
| 0x0204 85E0 | PCIBAR0MIR | PCI Base Address Register 0 Mirror Register |
| 0x0204 85E4 | PCIBAR1MIR | PCI Base Address Register 1 Mirror Register |
| 0x0204 85E8 | PCIBAR2MIR | PCI Base Address Register 2 Mirror Register |
| 0x0204 85EC | PCIBAR3MIR | PCI Base Address Register 3 Mirror Register |
| 0x0204 85F0 | PCIBAR4MIR | PCI Base Address Register 4 Mirror Register |
| 0x0204 85F4 | PCIBAR5MIR | PCI Base Address Register 5 Mirror Register |
| 0x0204 85F8 - 0x0204 86FF | - | Reserved |
| 0x0204 8700 | PCIMCFGDAT | PCI Master Configuration/IO Access Data Register |
| 0x0204 8704 | PCIMCFGADR | PCI Master Configuration/IO Access Address Register |
| 0x0204 8708 | PCIMCFGCMD | PCI Master Configuration/IO Access Command Register |
| 0x0204 870C - 0x0204 870F | - | Reserved |
| 0x0204 8710 | PCIMSTCFG | PCI Master Configuration Register |

**Table 6-62. PCI Hook Configuration Registers**

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---|---|---|
| 0x0204 8794 | PCIVENDEVPRG | PCI Vendor ID and Device ID Program Register |
| 0x0204 8798 | PCICMDSTATPRG | PCI Command and Status Program Register |
| 0x0204 879C | PCICLREVPRG | PCI Class Code and Revision ID Program Register |
| 0x0204 87A0 | PCISUBIDPRG | PCI Subsystem Vendor ID and Subsystem ID Program Register |
| 0x0204 87A4 | PCIMAXLGPRG | PCI Max Latency and Min Grant Program Register |
| 0x0204 87A8 | PCILRSTREG | PCI LRESET Register |
| 0x0204 87AC | PCICFGDONE | PCI Configuration Done Register |
| 0x0204 87B0 | PCIBAR0MPRG | PCI Base Address Mask Register 0 Program Register |
| 0x0204 87B4 | PCIBAR1MPRG | PCI Base Address Mask Register 1 Program Register |
| 0x0204 87B8 | PCIBAR2MPRG | PCI Base Address Mask Register 2 Program Register |
| 0x0204 87BC | PCIBAR3MPRG | PCI Base Address Mask Register 3 Program Register |
| 0x0204 87C0 | PCIBAR4MPRG | PCI Base Address Mask Register 4 Program Register |
| 0x0204 87C4 | PCIBAR5MPRG | PCI Base Address Mask Register 5 Program Register |
| 0x0204 87C8 | PCIBAR0PRG | PCI Base Address Register 0 Program Register |
| 0x0204 87CC | PCIBAR1PRG | PCI Base Address Register 1 Program Register |
| 0x0204 87D0 | PCIBAR2PRG | PCI Base Address Register 2 Program Register |
| 0x0204 87D4 | PCIBAR3PRG | PCI Base Address Register 3 Program Register |
| 0x0204 87D8 | PCIBAR4PRG | PCI Base Address Register 4 Program Register |

**Table 6-62. PCI Hook Configuration Registers (continued)**

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---|---|---|
| 0x0204 87DC | PCIBAR5PRG | PCI Base Address Register 5 Program Register |
| 0x0204 87E0 | PCIBAR0TRLPRG | PCI Base Address Translation Register 0 Program Register |
| 0x0204 87E4 | PCIBAR1TRLPRG | PCI Base Address Translation Register 1 Program Register |
| 0x0204 87E8 | PCIBAR2TRLPRG | PCI Base Address Translation Register 2 Program Register |
| 0x0204 87EC | PCIBAR3TRLPRG | PCI Base Address Translation Register 3 Program Register |
| 0x0204 87F0 | PCIBAR4TRLPRG | PCI Base Address Translation Register 4 Program Register |
| 0x0204 87F4 | PCIBAR5TRLPRG | PCI Base Address Translation Register 5 Program Register |
| 0x0204 87F8 | PCIBASENPRG | PCI Base En Prog Register |
| 0x0204 87FC - 0x0204 87FF | - | Reserved |

**Table 6-63. PCI External Memory Space**

| HEX ADDRESS OFFSET | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x4000 0000 - 0x407F FFFF | - | PCI Master Window 0 |
| 0x4080 0000 - 0x40FF FFFF | - | PCI Master Window 1 |
| 0x4100 0000 - 0x417F FFFF | - | PCI Master Window 2 |
| 0x4180 0000 - 0x41FF FFFF | - | PCI Master Window 3 |
| 0x4200 0000 - 0x427F FFFF | - | PCI Master Window 4 |
| 0x4280 0000 - 0x42FF FFFF | - | PCI Master Window 5 |
| 0x4300 0000 - 0x437F FFFF | - | PCI Master Window 6 |
| 0x4380 0000 - 0x43FF FFFF | - | PCI Master Window 7 |
| 0x4400 0000 - 0x447F FFFF | - | PCI Master Window 8 |
| 0x4480 0000 - 0x44FF FFFF | - | PCI Master Window 9 |
| 0x4500 0000 - 0x457F FFFF | - | PCI Master Window 10 |
| 0x4580 0000 - 0x45FF FFFF | - | PCI Master Window 11 |
| 0x4600 0000 - 0x467F FFFF | - | PCI Master Window 12 |
| 0x4680 0000 - 0x46FF FFFF | - | PCI Master Window 13 |
| 0x4700 0000 - 0x477F FFFF | - | PCI Master Window 14 |
| 0x4780 0000 - 0x47FF FFFF | - | PCI Master Window 15 |
| 0x4800 0000 - 0x487F FFFF | - | PCI Master Window 16 |
| 0x4880 0000 - 0x48FF FFFF | - | PCI Master Window 17 |
| 0x4900 0000 - 0x497F FFFF | - | PCI Master Window 18 |
| 0x4980 0000 - 0x49FF FFFF | - | PCI Master Window 19 |
| 0x4A00 0000 - 0x4A7F FFFF | - | PCI Master Window 20 |
| 0x4A80 0000 - 0x4AFF FFFF | - | PCI Master Window 21 |
| 0x4B00 0000 - 0x4B7F FFFF | - | PCI Master Window 22 |
| 0x4B80 0000 - 0x4BFF FFFF | - | PCI Master Window 23 |
| 0x4C00 0000 - 0x4C7F FFFF | - | PCI Master Window 24 |
| 0x4C80 0000 - 0x4CFF FFFF | - | PCI Master Window 25 |
| 0x4D00 0000 - 0x4D7F FFFF | - | PCI Master Window 26 |
| 0x4D80 0000 - 0x4DFF FFFF | - | PCI Master Window 27 |
| 0x4E00 0000 - 0x4E7F FFFF | - | PCI Master Window 28 |
| 0x4E80 0000 - 0x4EFF FFFF | - | PCI Master Window 29 |
| 0x4F00 0000 - 0x4F7F FFFF | - | PCI Master Window 30 |
| 0x4F80 0000 - 0x4FFF FFFF | - | PCI Master Window 31 |

### 6.15.3 PCI Electrical Data/Timing

Texas Instruments (TI) has performed the simulation and system characterization to be sure that the PCI peripheral meets all ac timing specifications as required by the *PCI Local Bus Specification* (version 2.3). The ac timing specifications are not reproduced here. For more information on the ac timing specifications, see Section 4.2.3, Timing Specification (33 MHz timing), and Section 7.6.4, Timing Specification (66 MHz timing), of the *PCI Local Bus Specification* (version 2.3). Note that the PCI peripheral only supports 3.3-V signaling.

## 6.16 Multichannel Audio Serial Port (McASP)

The McASP functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, Inter-Integrated Sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT).

### *6.16.1 McASP Device-Specific Information*

The device includes one multichannel audio serial port (McASP) interface peripheral. The McASP is a serial port optimized for the needs of multichannel audio applications.

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

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

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

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

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

For more detailed information on and the functionality of the McASP peripheral, see the *TMS320C6452 Multichannel Audio Serial Port (McASP) User's Guide* (literature number SPRUF91).

#### 6.16.1.1 McASP Block Diagram

Figure 6-42 illustrates the major blocks along with external signals of the McASP peripheral; and shows the 10 serial data [AXR] pins.

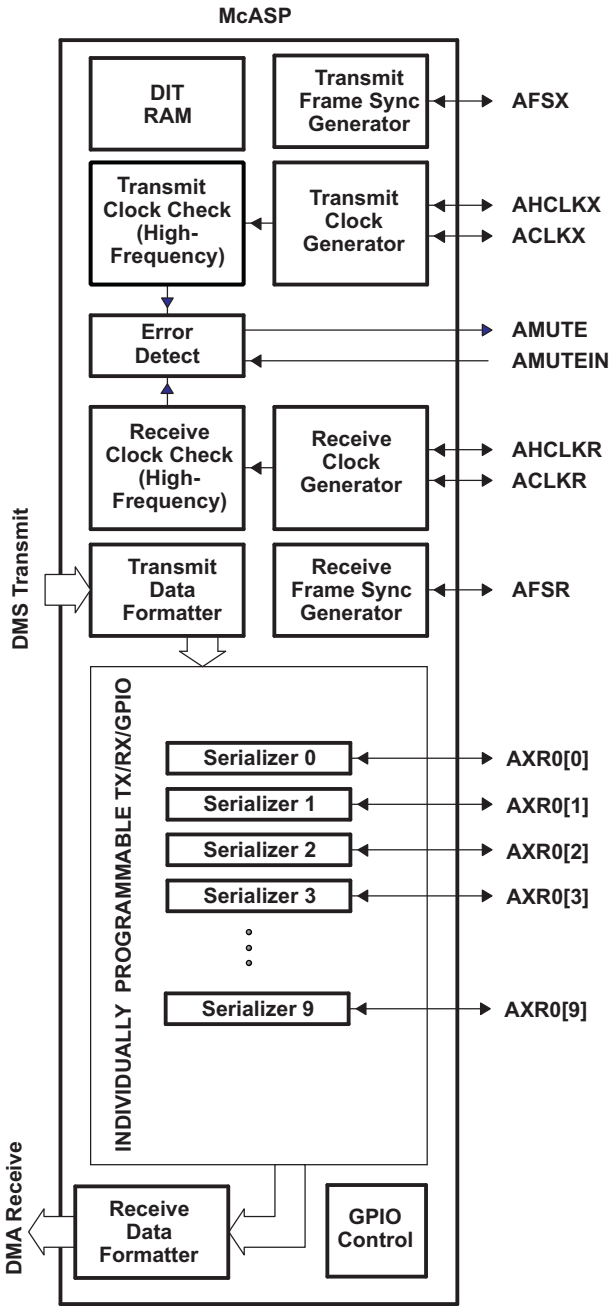



**Figure 6-42. McASP Configuration**

### 6.16.1.2 McASP Peripheral Register Description(s)

**Table 6-64. McASP Control Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 0000 | PID | Peripheral Identification register [Register value: 0x0010 0101] |
| 0x0204 0004 | PWRDEMU | Power down and emulation management register |
| 0x0204 0008 | - | Reserved |
| 0x0204 000C | - | Reserved |
| 0x0204 0010 | PFUNC | Pin function register |
| 0x0204 0014 | PDIR | Pin direction register |
| 0x0204 0018 | PDOUT | Pin data out register |
| 0x0204 001C | PDIN/PDSET | Pin data in/data set register<br>Read returns: PDIN<br>Writes affect: PDSET |
| 0x0204 0020 | PDCLR | Pin data clear register |
| 0x0204 0024 - 0x0204 0040 | - | Reserved |
| 0x0204 0044 | GBLCTL | Global control register |
| 0x0204 0048 | AMUTE | Mute control register |
| 0x0204 004C | DLBCTL | Digital Loop-back control register |
| 0x0204 0050 | DITCTL | DIT mode control register |
| 0x0204 0054 - 0x0204 005C | - | Reserved |
| 0x0204 0060 | RGBLCTL | Alias of GBLCTL containing only Receiver Reset bits, allows transmit to be reset independently from receive. |
| 0x0204 0064 | RMASK | Receiver format UNIT bit mask register |
| 0x0204 0068 | RFMT | Receive bit stream format register |
| 0x0204 006C | AFSRCTL | Receive frame sync control register |
| 0x0204 0070 | ACLKRCTL | Receive clock control register |
| 0x0204 0074 | AHCLKRCTL | High-frequency receive clock control register |
| 0x0204 0078 | RTDM | Receive TDM slot 0-31 register |
| 0x0204 007C | RINTCTL | Receiver interrupt control register |
| 0x0204 0080 | RSTAT | Status register - Receiver |
| 0x0204 0084 | RSLOT | Current receive TDM slot register |
| 0x0204 0088 | RCLKCHK | Receiver clock check control register |
| 0x0204 008C - 0x0204 009C | - | Reserved |
| 0x0204 00A0 | XGBLCTL | Alias of GBLCTL containing only Transmitter Reset bits, allows transmit to be reset independently from receive. |
| 0x0204 00A4 | XMASK | Transmit format UNIT bit mask register |
| 0x0204 00A8 | XFMT | Transmit bit stream format register |
| 0x0204 00AC | AFSXCTL | Transmit frame sync control register |
| 0x0204 00B0 | ACLKXCTL | Transmit clock control register |
| 0x0204 00B4 | AHCLKXCTL | High-frequency Transmit clock control register |
| 0x0204 00B8 | XTDM | Transmit TDM slot 0-31 register |
| 0x0204 00BC | XINTCTL | Transmit interrupt control register |
| 0x0204 00C0 | XSTAT | Status register - Transmitter |
| 0x0204 00C4 | XSLOT | Current transmit TDM slot |
| 0x0204 00C8 | XCLKCHK | Transmit clock check control register |

**Table 6-64. McASP Control Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 00CC - 0x0204 00FC | - | Reserved |
| 0x0204 0100 | DITCSRA0 | Left (even TDM slot) channel status register file |
| 0x0204 0104 | DITCSRA1 | Left (even TDM slot) channel status register file |
| 0x0204 0108 | DITCSRA2 | Left (even TDM slot) channel status register file |
| 0x0204 010C | DITCSRA3 | Left (even TDM slot) channel status register file |
| 0x0204 0110 | DITCSRA4 | Left (even TDM slot) channel status register file |
| 0x0204 0114 | DITCSRA5 | Left (even TDM slot) channel status register file |
| 0x0204 0118 | DITCSRB0 | Right (odd TDM slot) channel status register file |
| 0x0204 011C | DITCSRB1 | Right (odd TDM slot) channel status register file |
| 0x0204 0120 | DITCSRB2 | Right (odd TDM slot) channel status register file |
| 0x0204 0124 | DITCSRB3 | Right (odd TDM slot) channel status register file |
| 0x0204 0128 | DITCSRB4 | Right (odd TDM slot) channel status register file |
| 0x0204 012C | DITCSRB5 | Right (odd TDM slot) channel status register file |
| 0x0204 0130 | DITUDRA0 | Left (even TDM slot) user data register file |
| 0x0204 0134 | DITUDRA1 | Left (even TDM slot) user data register file |
| 0x0204 0138 | DITUDRA2 | Left (even TDM slot) user data register file |
| 0x0204 013C | DITUDRA3 | Left (even TDM slot) user data register file |
| 0x0204 0140 | DITUDRA4 | Left (even TDM slot) user data register file |
| 0x0204 0144 | DITUDRA5 | Left (even TDM slot) user data register file |
| 0x0204 0148 | DITUDRB0 | Right (odd TDM slot) user data register file |
| 0x0204 014C | DITUDRB1 | Right (odd TDM slot) user data register file |
| 0x0204 0150 | DITUDRB2 | Right (odd TDM slot) user data register file |
| 0x0204 0154 | DITUDRB3 | Right (odd TDM slot) user data register file |
| 0x0204 0158 | DITUDRB4 | Right (odd TDM slot) user data register file |
| 0x0204 015C | DITUDRB5 | Right (odd TDM slot) user data register file |
| 0x0204 0160 - 0x0204 017C | - | Reserved |
| 0x0204 0180 | SRCTL0 | Serializer 0 control register |
| 0x0204 0184 | SRCTL1 | Serializer 1 control register |
| 0x0204 0188 | SRCTL2 | Serializer 2 control register |
| 0x0204 018C | SRCTL3 | Serializer 3 control register |
| 0x0204 0190 | SRCTL4 | Serializer 4 control register |
| 0x0204 0194 | SRCTL5 | Serializer 5 control register |
| 0x0204 0198 | SRCTL6 | Serializer 6 control register |
| 0x0204 019C | SRCTL7 | Serializer 7 control register |
| 0x0204 01A0 | SRCTL8 | Serializer 8 control register |
| 0x0204 01A4 | SRCTL9 | Serializer 9 control register |
| 0x0204 01A8 - 0x0204 01FC | - | Reserved |
| 0x0204 0200 | XBUF0 | Transmit Buffer for Serializer 0 |
| 0x0204 0204 | XBUF1 | Transmit Buffer for Serializer 1 |
| 0x0204 0208 | XBUF2 | Transmit Buffer for Serializer 2 |
| 0x0204 020C | XBUF3 | Transmit Buffer for Serializer 3 |
| 0x0204 0210 | XBUF4 | Transmit Buffer for Serializer 4 |
| 0x0204 0214 | XBUF5 | Transmit Buffer for Serializer 5 |
| 0x0204 0218 | XBUF6 | Transmit Buffer for Serializer 6 |
| 0x0204 021C | XBUF7 | Transmit Buffer for Serializer 7 |

**Table 6-64. McASP Control Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 021A | XBUF8 | Transmit Buffer for Serializer 8 |
| 0x0204 0220 | XBUF9 | Transmit Buffer for Serializer 9 |
| 0x0204 0224-0x0204 027C | - | Reserved |
| 0x0204 0280 | RBUF0 | Receive Buffer for Serializer 0 |
| 0x0204 0284 | RBUF1 | Receive Buffer for Serializer 1 |
| 0x0204 0288 | RBUF2 | Receive Buffer for Serializer 2 |
| 0x0204 028C | RBUF3 | Receive Buffer for Serializer 3 |
| 0x0204 0290 | RBUF4 | Receive Buffer for Serializer 4 |
| 0x0204 0294 | RBUF5 | Receive Buffer for Serializer 5 |
| 0x0204 0298 | RBUF6 | Receive Buffer for Serializer 6 |
| 0x0204 029C | RBUF7 | Receive Buffer for Serializer 7 |
| 0x0204 02A0 | RBUF8 | Receive Buffer for Serializer 8 |
| 0x0204 02A4 | RBUF9 | Receive Buffer for Serializer 9 |
| 0x0204 02A8-0x0204 3FFF | - | Reserved |

**Table 6-65. McASP Data Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|---|---|---|---|
| 0204 4000 - 0204 43FF | RBUF/XBUF | McASP receive buffers or McASP transmit buffers via the Peripheral Data Bus. | (Used when RSEL or XSEL bits = 0 [these bits are located in the RFMT or XFMT registers, respectively].) |

### 6.16.1.3 McASP Electrical Data/Timing

#### *6.16.1.3.1 Multichannel Audio Serial Port (McASP) Timing*

**Table 6-66. Timing Requirements for McASP (see Figure 6-43 and Figure 6-44)(1)**

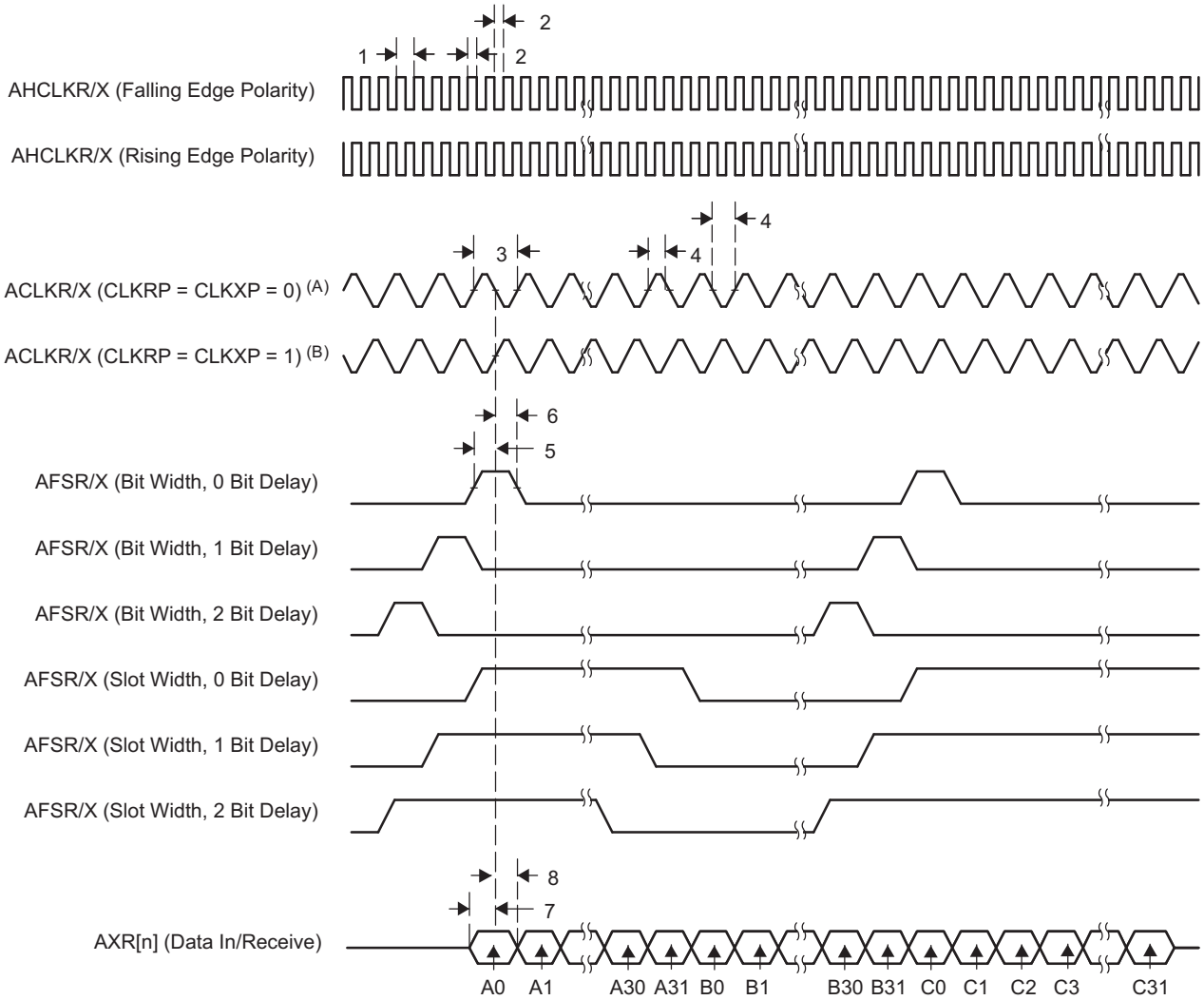
| NO. | | | | 720 900 | | UNIT |
|---|---|---|---|---|---|---|
| | | | | MIN | MAX | |
| 1 | $t_{c(AHCKRX)}$ | Cycle time, AHCLKR/X | | 20 | | ns |
| 2 | $t_{w(AHCKRX)}$ | Pulse duration, AHCLKR/X high or low | | 10 | | ns |
| 3 | $t_{c(CKRX)}$ | Cycle time, ACLKR/X | ACLKR/X ext | 33 | | ns |
| 4 | $t_{w(CKRX)}$ | Pulse duration, ACLKR/X high or low | ACLKR/X ext | 16.5 | | ns |
| 5 | $t_{su(FRX-CKRX)}$ | Setup time, AFSR/X input valid before ACLKR/X latches data | ACLKR/X int | 5 | | ns |
| | | | ACLKR/X ext | 5 | | ns |
| 6 | $t_{h(CKRX-FRX)}$ | Hold time, AFSR/X input valid after ACLKR/X latches data | ACLKR/X int | 5 | | ns |
| | | | ACLKR/X ext | 12.08 | | ns |
| 7 | $t_{su(AXR-CKRX)}$ | Setup time, AXR input valid before ACLKR/X latches data | ACLKR/X int | 5 | | ns |
| | | | ACLKR/X ext | 5 | | ns |
| 8 | $t_{h(CKRX-AXR)}$ | Hold time, AXR input valid after ACLKR/X latches data | ACLKR/X int | 5 | | ns |
| | | | ACLKR/X ext | 7.35 | | ns |

(1) ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1
ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1

**Table 6-67. Switching Characteristics Over Recommended Operating Conditions for McASP (see Figure 6-43 and Figure 6-44)(1)**

| NO. | PARAMETER | | | 720 900 | | UNIT |
|---|---|---|---|---|---|---|
| | | | | MIN | MAX | |
| 9 | $t_{c(AHCKRX)}$ | Cycle time, AHCLKR/X | | 20 | | ns |
| 10 | $t_{w(AHCKRX)}$ | Pulse duration, AHCLKR/X high or low | | 10 | | ns |
| 11 | $t_{c(CKRX)}$ | Cycle time, ACLKR/X | ACLKR/X int | 33 | | ns |
| 12 | $t_{w(CKRX)}$ | Pulse duration, ACLKR/X high or low | ACLKR/X int | 16.5 | | ns |
| 13 | $t_{d(CKRX-FRX)}$ | Delay time, ACLKR/X transmit edge to AFSX/R output valid | ACLKR/X int | | 5 | ns |
| | | | ACLKR/X ext | | 14.28 | ns |
| 14 | $t_{d(CKX-AXRV)}$ | Delay time, ACLKX transmit edge to AXR output valid | ACLKX int | | 10 | ns |
| | | | ACLKX ext | | 12.5 | ns |
| 15 | $t_{dis(CKRX-AXRHZ)}$ | Disable time, AXR high impedance following last data bit from ACLKR/X transmit edge | ACLKR/X int | | 10 | ns |
| | | | ACLKR/X ext | | 12.5 | ns |

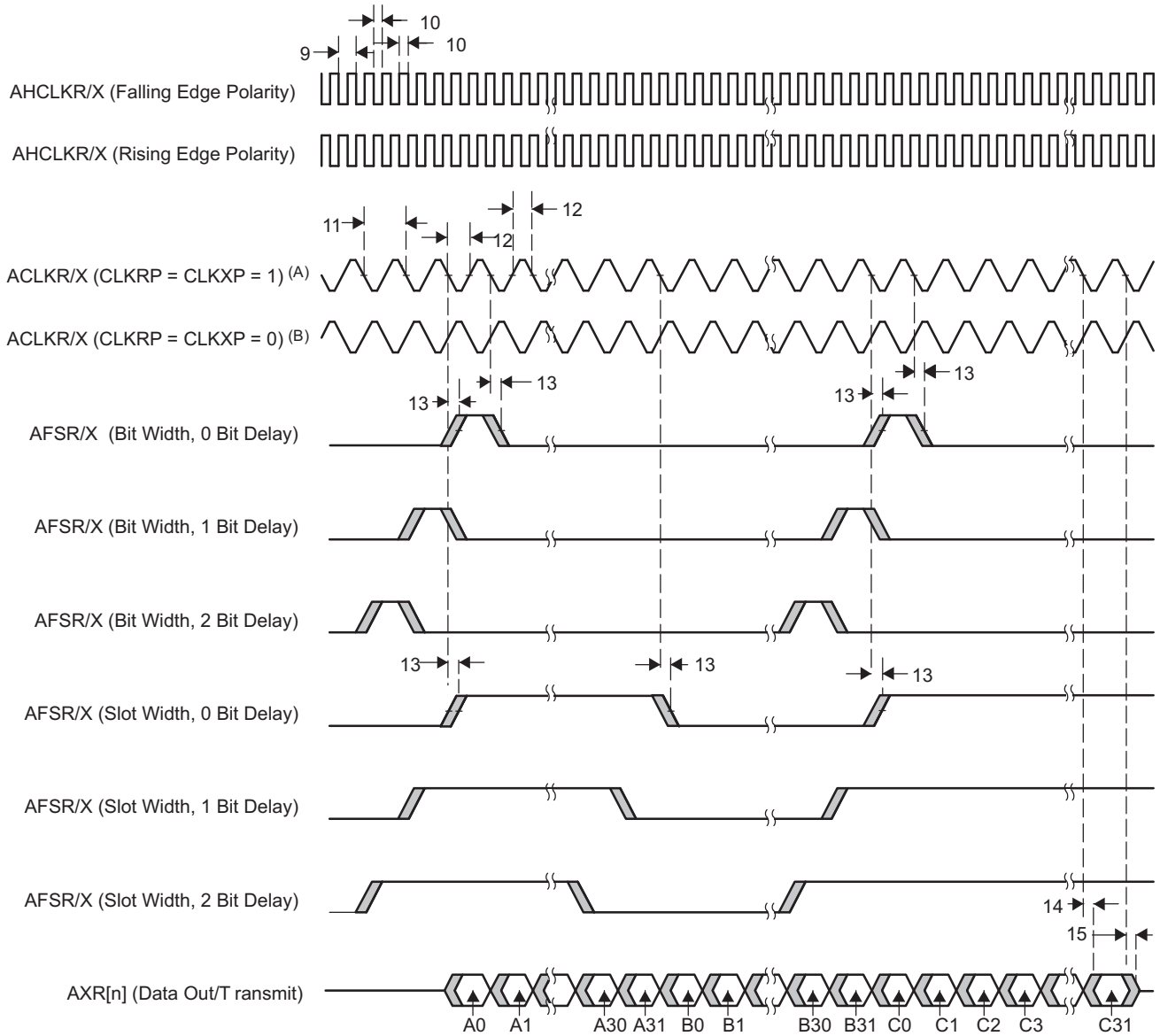
(1) ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1
ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1




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

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

**Figure 6-43. McASP Input Timing**




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

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

**Figure 6-44. McASP Output Timing**

## 6.17 Telecom Serial Interface Port (TSIP)

The TSIP provides a glueless interface to common telecom serial data streams, including H-MVIP and H.100/H.110. For transmit and receive, 256 channels are provided. The clock and frame sync for all transmit channels is common, and the clock and frame sync for all receive links is also common. A single clock and frame sync may be used for transmit and receive together. The clock polarity for data and frame sync is independently selectable. The clock frequency can be set for 2X operation (16.384 MHz for 8.192 Mbps data) or 1X operation (8.192 MHz for 8.192 Mbps data).

### 6.17.1 Features

- Dual Transmit & Receive Telecom Serial Interface Ports (TSIP)
  - Each TSIP module supports a transmit/receive data rate of 16.384 Mbps
    - One transmit/receive data lane per module at 16.384 Mbps each
    - Two transmit/receive data lanes per module at 8.192 Mbps each
- Flexible Clock and Frame Sync Inputs
  - Configurable for independent TX and RX clock & Frame Sync
  - Configurable for common TX and RX clock & Frame Sync (second clock & Frame Sync can be used as a redundant source)
  - Independent clock polarity selection for data & Frame Sync
  - Selection of 2X or 1X data clock frequency
- Timeslot Data Management and Multi-Channel DMA Capability
  - Independent timeslot enable/disable per DMA channel
  - Selectively unpacks and packs timeslot data for transmit and receive based on a channel timeslot definition
  - A-law and μ-law support on a per timeslot basis
- Frame and SuperFrame Interrupts
- 8-terminal External Interface (min 4 terminals/TSIP x 2 ports)

### 6.17.2 TSIP External Signals

The device contains two TSIP modules. The external interface of each TSIP consists of eight signals: two clock-input signals, two frame-sync input signals, two receive-data input signals, and two transmit-data output signals.

**Table 6-68. Signal Descriptions**

| SIGNAL NAME | DIRECTION | DESCRIPTION(1) |
|---|---|---|
| TSIPxCLKA | Input | TSIP serial data clock A |
| TSIPxCLKB | Input | TSIP serial data clock B |
| TSIPxFSA | Input | TSIP frame sync A |
| TSIPxFSB | Input | TSIP frame sync B |
| TSIPxTR[1:0] | Input | TSIP serial receive data. The serial data clock and frame sync is common for all channels. |
| TSIPxTX[1:0] | Output | TSIP serial transmit data. The serial data clock and frame sync is common for all channels. |

(1) In, 256 time slot mode (16.384 Mbps), only CLKA, FSA, TR0 and TX0 are used

### Table 6-69. TSIP Module Registers

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00000000 | PID | PID Register |
| 0x00000004 | EMUTST | Emulation and Test Register |
| 0x00000008 | RST | Reset Register |
| 0x0000000C -- 0x0000007C | - | Reserved |

### Table 6-70. Serial Interface Registers

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00000080 | SIUCTL | SIU Global Control Register |
| 0x00000084 -- 0x0000009C | - | Reserved |
| 0x000000A0 | XCLK | Transmit Clock Source Register |
| 0x000000A4 | XCTL | Transmit Control Register |
| 0x000000A8 | XSIZE | Transmit Size Register |
| 0x0000000AC -- 0x000000BC | - | Reserved |
| 0x0000000C0 | RCLK | Receive Clock Source Register |
| 0x000000C4 | RCTL | Receive Control Register |
| 0x000000C8 | RSIZE | Receive Size Register |
| 0x0000000CC -- 0x000000FC | - | Reserved |

### Table 6-71. TDMU Global Registers

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00000100 | TDMUCTL | TDMU Global Control Register |
| 0x00000104 | XFRFC | Transmit Free Running Frame Counter |
| 0x00000108 | RFRFC | Receive Free Running Frame Counter |
| 0x0000010C | TDMUCFG | TDMU Global Configuration Register |
| 0x00000110 | XBMST | Transmit Channel Active Status Register |
| 0x000000114 | RBMST | Receive Channel Active Status Register |
| 0x000000118 -- 0x0000017C | - | Reserved |

### Table 6-72. DMATCU Global Registers

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00000180 | DMACTL | DMATCU Global Control Register |
| 0x00000184 | XDLY | Transmit Timeslot Delay Counter |
| 0x00000188 | RDLY | Receive Timeslot Delay Counter |
| 0x0000018C | - | Reserved |
| 0x00000190 | XCHST | Transmit Channel Active Status Register |
| 0x000000194 | RCHST | Receive Channel Active Status Register |
| 0x000000198 -- 0x000001FC | - | Reserved |

### Table 6-73. TDMU Channel Error Registers

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00000200 | ERRCTL | TX/RX Channel 0 Error Control Register |
| 0x00000204 | ERRCNT | TX/RX Channel 0 Error Count Register |
| 0x00000208 | ERRQ | TX/RX Channel 0 Error Queue Register |
| 0x0000020C | - | Reserved |

**Table 6-74. TDMU Channel Registers**

| OFFSET ADDRESS | ACRONYM | REGISTER DESCRIPTION |
|---|---|---|
| 0x00000800 -- 0x0000081C | XCHEN | Transmit channel 0 registers |
| 0x00000820 -- 0x00000BFC | - | Reserved |
| 0x00000C00 -- 0x00000C1C | RCHEN | Receive channel 0 registers |
| 0x00000C20 -- 0x00000FFC | - | Reserved |

**Table 6-75. DMATCU Transmit Channel 0 Registers**

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00001000 | ABASE | Memory Base Address Register A |
| 0x00001004 | AFALLOC | Frame Allocation Register A |
| 0x00001008 | AFSIZE | Frame Size Register A |
| 0x0000100C | AFCNT | Frame Count Register A |
| 0x000001010 -- 0x0000101C | - | Reserved |
| 0x000001020 | BBASE | Memory Base Address Register B |
| 0x000001024 | BFALLOC | Frame Allocation Register B |
| 0x000001028 | BFSIZE | Frame Size Register B |
| 0x00000102C | BFCNT | Frame Count Register B |
| 0x000001030 -- 0x000017FC | - | Reserved |

**Table 6-76. DMATCU Receive Channel 0 Registers**

| OFFSET ADDRESS | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x00001800 | ABASE | Memory Base Address Register A |
| 0x00001804 | AFALLOC | Frame Allocation Register A |
| 0x00001808 | AFSIZE | Frame Size Register A |
| 0x0000180C | AFCNT | Frame Count Register A |
| 0x000001810 -- 0x0000181C | - | Reserved |
| 0x000001820 | BBASE | Memory Base Address Register B |
| 0x000001824 | BFALLOC | Frame Allocation Register B |
| 0x000001828 | BFSIZE | Frame Size Register B |
| 0x00000182C | BFCNT | Frame Count Register B |
| 0x000001830 -- 0x00001FFC | - | Reserved |

**Table 6-77. TDMU Channel Bitmap Registers**

| OFFSET ADDRESS | ACRONYM | REGISTER DESCRIPTION |
|---|---|---|
| 0x00008000 -- 0x000080FC | XBMA | Transmit channel 0 bitmap A |
| 0x00008100 -- 0x000081FC | XBMB | Transmit channel 0 bitmap B |
| 0x00008200 -- 0x0000BFFC | - | Reserved |
| 0x0000C000 -- 0x0000C0FC | RBMA | Receive channel 0 bitmap A |
| 0x0000C100 -- 0x0000C1FC | RBMB | Receive channel 0 bitmap B |
| 0x0000CC00 -- 0x0000FFFC | - | Reserved |

### 6.17.3 TSIP Timing

TSIP supports two basic timing modes, 2X and 1X. In the 2X timing mode, there are two serial data clock periods per data bit. In the 1X timing mode, there is one serial clock period per data bit. Both modes offer programmable delay from the recognition of the frame sync pulse that indicates the start of a frame and the first (most significant) bit of timeslot 0 for that frame.

### 6.17.4 2X Mode Timing

The 2X mode timing is illustrated and defined in Table 6-78, Table 6-79, and Figure 6-45. The nominal frequency for the selected serial data clock (CLK_A or CLK_B) is 16.384 MHz (± 0.1%), 32.768 MHz (± 0.1%), or 65.536 MHz (± 0.1%), depending on the data rate option. The nominal frequency for the selected frame sync (FS_A or FS_B) is 8 kHz.

**Table 6-78. Timing Requirements for TSIP 2X Mode**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| 1 | $t_{c(clk)}$ | Cycle time, CLK rising edge to next CLK rising edge | 61 (1) | | ns |
| 2 | $t_{w(clkl)}$ | Pulse duration, CLK low | 0.4 $t_{c(clk)}$ | | ns |
| 3 | $t_{w(clkh)}$ | Pulse duration, CLK high | 0.4 $t_{c(clk)}$ | | ns |
| 4 | $t_{t(clk)}$ | Transition time, CLK high to low or CLK low to high | | 2 | ns |
| 5 | $t_{su(fs-clk)}$ | Setup time, $f_s$ valid before rising CLK | 5 | | ns |
| 6 | $t_{h(clk-fs)}$ | Hold time, $f_s$ valid after rising CLK | 5 | | ns |
| 7 | $t_{su(tr-clk)}$ | Setup time, $t_r$ valid before rising CLK | 5 | | ns |
| 8 | $t_{h(clk-tr)}$ | Hold time, $t_r$ valid before rising CLK | 5 | | ns |

(1) Timing shown is for 8.192 Mbps links. Timing for 16.384 Mbps and 32.768 Mbps links is 30.5 ns and 15.2 ns, respectively.

**Table 6-79. 2x Switching Characteristics**

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

| | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| 9 | $t_{d(clkl-tx)}$ | Delay time, CLK low to TX valid | | 1 | | 12 | ns |

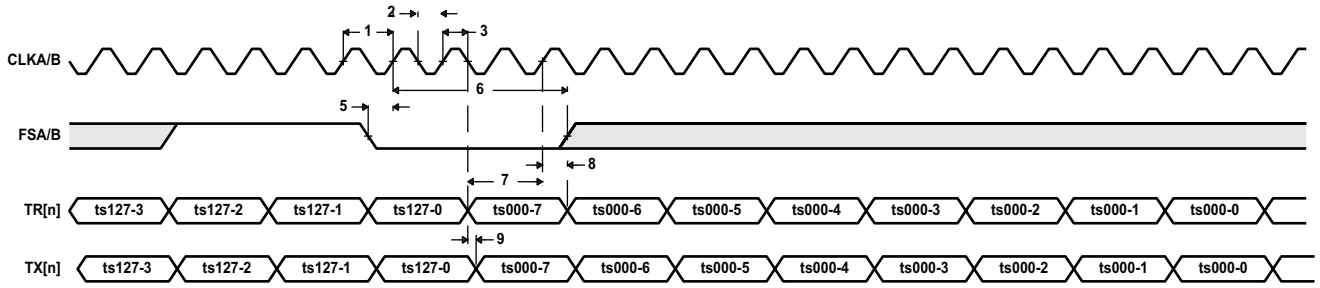



**Figure 6-45. TSIP 2x Timing Diagram**

### 6.17.5 1X Mode Timing

The 1X mode timing is illustrated and defined in Table 6-80, Table 6-81, and Figure 6-46. The nominal frequency for the selected serial data clock (CLK_A or CLK_B) is 8.192 MHz (± 0.1%), 16.384 MHz (± 0.1%), or 32.768 MHz (± 0.1%), depending on the data rate option. The nominal frequency for the selected frame sync (FS_A or FS_B) is 8 kHz.

**Table 6-80. Timing Requirements for TSIP 1X Mode**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| 11 | $t_{c(clk)}$ | Cycle time, CLK rising edge to next CLK rising edge | 122.1 (1) | | ns |
| 12 | $t_{w(clkl)}$ | Pulse duration, CLK low | 0.4 $t_{c(clk)}$ | | ns |
| 13 | $t_{w(clkh)}$ | Pulse duration, CLK high | 0.4 $t_{c(clk)}$ | | ns |
| 14 | $t_{t(clk)}$ | Transition time, CLK high to low or CLK low to high | | 2 | ns |
| 15 | $t_{su(fs-clk)}$ | Setup time, $f_s$ valid before rising CLK | 5 | | ns |
| 16 | $t_{h(clk-fs)}$ | Hold time, $f_s$ valid after rising CLK | 5 | | ns |
| 17 | $t_{su(tr-clk)}$ | Setup time, $t_r$ valid before falling CLK | 5 | | ns |
| 18 | $t_{h(clk-tr)}$ | Hold time, $t_r$ valid before falling CLK | 5 | | ns |

(1) Timing shown is for 8.192 Mbps links. Timing for 16.384 Mbps and 32.768 Mbps links is 61 ns and 30.5 ns, respectively.

**Table 6-81. Switching Characteristics for TSIP 1X Mode**

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

| | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| 19 | $t_{d(clkh-tx)}$ | Delay time, CLK high to TX valid (1024 (1) clock cycles plus) | | 1 | | 12 | ns |

(1) Delay is exactly one frame plus the switching delay for 8.192 Mbps links. Data delays of 1 - 1024, 1 - 2048, and 14096 are suitable for 8.192 Mbps, 16.384 Mbps, and 32.768 Mbps links, respectively.

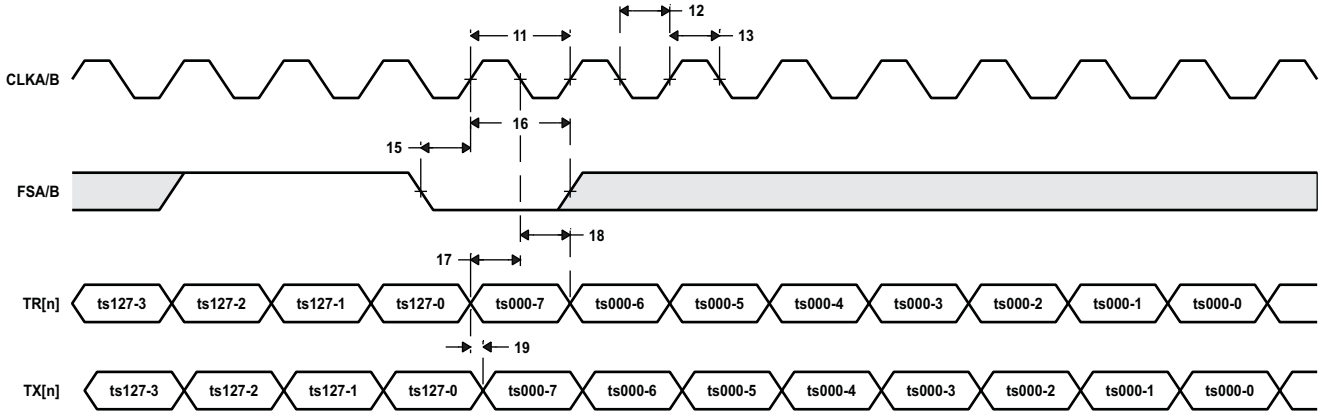



**Figure 6-46. 1X Timing Diagram**

## 6.18 3-Port Ethernet Switch Subsystem (3PSW)

The Ethernet module controls the flow of packet data between the device and two external Ethernet PHYs or one external Ethernet PHY, with hardware flow control and quality-of-service (QOS) support. See Figure 6-47 for a block diagram of the Ethernet module. The Ethernet Subsystem contains a 3-port gigabit switch, where one port is internally connected to the C64x+ DSP (via the switched central resource) and the other two ports are brought out externally. Each of the external Ethernet ports support the modes shown in Table 6-82.

The Ethernet module controls the flow of packet data between the device and two external Ethernet PHYs , with hardware flow control and quality-of-service (QOS) support. See Figure 6-47 for a block diagram of the Ethernet module. The Ethernet Subsystem contains a 3-port gigabit switch, where one port is internally connected to the C64x+ DSP (via the switched central resource) and the other two ports are brought out externally. Each of the external Ethernet ports support the modes shown in Table 6-82.

**Table 6-82. Ethernet Operating Modes**

| DESCRIPTION | DATA RATE | OPERATING MODE |
|---|---|---|
| 10Base-T | 10 Mbits/second (Mbps) | half- or full-duplex |
| 100Base-T | 100 Mbits/second (Mbps) | half- or full-duplex |
| 1000Base-T | 1000 Mbits/second (Mbps) | full-duplex |

The Ethernet Subsystem provides these functions:

- Ethernet communication/routing by way of two dedicated 10/100/1000 ports with SGMII interfaces
  - Wire-rate switching (802.1d), non-blocking switch fabric
  - Four priority levels of QoS TX support (802.1p) in hardware
  - Programmable interrupt pacing on RX/TX plus interrupt threshold on RX
  - Supports forwarding frame sizes of 64-2020 bytes
- Address Lookup
  - 1024 total address lookup engine (ALE) entries of VLANs and/or MAC addresses
  - L2 address lock and L2 filtering support
  - Multicast/broadcast filtering and forwarding state control
  - Receive-based or destination-based multicast and broadcast rate limits
  - MAC address blocking
  - Source port locking
  - OUI (Vendor ID) host accept/deny feature
  - Host controlled time-based aging
  - MAC authentication (802.1x)
  - Remapping of priority level of VLAN or ports
  - Multiple spanning tree support (spanning tree per VLAN)

- VLAN support
  - 802.1Q compliant
    - Auto add port VLAN for untagged frames on ingress
    - Auto VLAN removal on egress and auto pad to minimum frame size
  - Flow control (IEEE 802.3x)
  - Programmable priority escalation to specify delivery of lower priority level packets in the event of over-subscribed TX high priority traffic
  - Host pass CRC mode (enables CRC protection through host)
  - Write-protect option for Ethernet module registers (3PGSW, CPPI RAM, MDIO, SGMII0, SGMII1, control)
  - Ethernet statistics:
    - EtherStats and 802.3 Stats RMON statistics gathering (shared)
    - Programmable statistics interrupt mask when a statistic is above one half its 32-bit value
  - MDIO module for PHY management
  - SGMII gigabit current mode logic (CML) differential SERializer/DESerializer (SerDes) I/O receiver/transmitters
    - Adaptive active equalization for superior data dependent jitter tolerance in the presence of a lossy channel
    - Loss of signal detector with programmable threshold levels in receive channels
    - Integrated receiver and transmitter termination
    - IEEE 802.3 gigabit Ethernet conformant

### 6.18.1 Ethernet Subsystem Functions

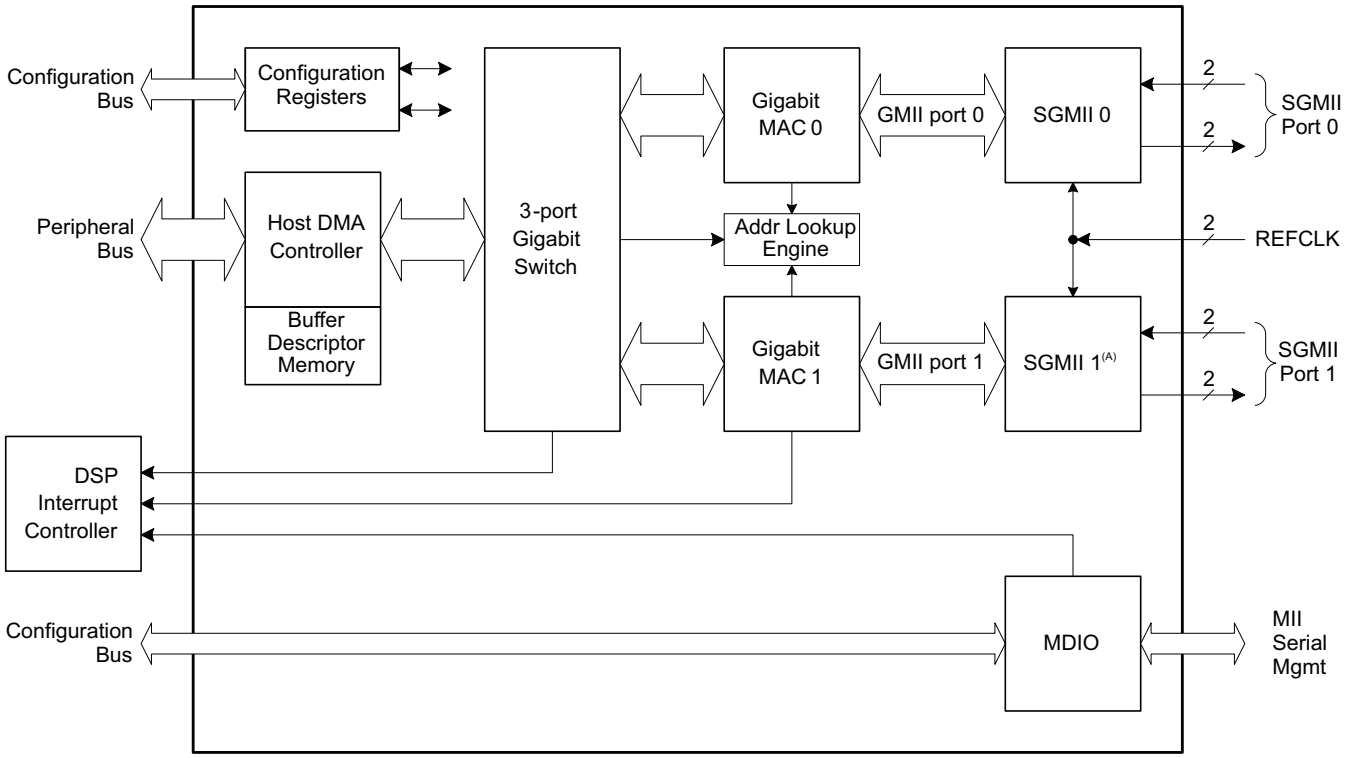



**Figure 6-47. Ethernet Subsystem Block Diagram**

The Ethernet Subsystem conforms to the IEEE 802.3-2002 standard. Deviating from this standard, the GMAC module does not use the transmit coding error signal MTXER. Instead of driving the error pin when an underflow condition occurs on a transmitted frame, the GMAC generates an incorrect checksum by inverting the frame CRC, so that the transmitted frame will be detected as an error by the network.

In networking systems, packet transmission and reception are critical tasks. The communications port programming interface (CPPI) protocol maximizes the efficiency of interaction between the host software and communications modules. .

After reset, initialization, configuration, and auto-negotiation, the host C64x+ DSP may initiate Ethernet transmit and receive operations.

- Transmit operations are initiated by C64x+ DSP writes to the appropriate transmit channel head descriptor pointer contained in the CPDMA block. The CPDMA TX controller then fetches the first packet in the packet chain from memory in accordance with the CPPI protocol for the GMAC to process before sending to the SGMII.
- Receive operations are initiated by C64x+ DSP writes to the appropriate receive channel head descriptor pointer. The CPDMA RX controller then writes packets to memory in accordance with the CPPI protocol.

DSP writes may be write-protected to the Ethernet Subsystem configuration registers from addresses 0x02D0 0000 - 0x02D0 4FFF (3PGSW, MDIO, SGMII0, SGMII1, control), and the CPPI RAM. The Ethernet Subsystem setting in the PSC is also write-protected. A specific 32-bit lock code (0x4C6F436B) and a 32-bit unlock code (0x6F50654E) written to ESS_LOCK register will activate or clear this option, respectively. See Section 3.2.5 and Section 3.2.7

The 3-port gigabit switch block contains the following functions:

- 3-port gigabit switch: performs packet forwarding and routing functions, one port is internally connected to the C64x+ DSP and two ports are brought out externally
- CPDMA: performs high-speed DMA transfers with RX and TX CPPI buffers in local memory, including channel setup and channel teardown
- GMAC (Gigabit Ethernet MAC):
  - Uses Rx packet FIFO, and a TX packet FIFO to improve data transfer efficiency
  - Handles processing of Ethernet packet data, frames, and headers
  - Includes flow control
  - Provides statistics collection and reporting
- The address lookup engine (ALE) processes all received packets to determine where (that is, which packet location) to forward the packet. The ALE uses the incoming packet received port number, destination address, source address, length/type, and VLAN information to determine how the packet should be forwarded. The ALE outputs the port mask to the switch fabric that indicates to which port(s) the packet should be forwarded.

### 6.18.2 Interrupt Controller and Pacing Interrupts

The interrupt control block selects the interrupts from the 3-port gigabit switch and MDIO modules for output to the C64x+ DSP. The miscellaneous interrupt is an immediate (non-paced) interrupt selected from the miscellaneous interrupts (host error level, statistics level, MDIO User [2], MDIO link [2]).

The eight RX interrupts and eight TX interrupts can be paced. The 8 RX threshold interrupts and the miscellaneous interrupts are not paced. The interrupt pacing feature limits the number of interrupts that occur during a given period of time. For heavily loaded systems in which interrupts can occur at a very high rate, the performance benefit is significant due to minimizing the overhead associated with servicing each interrupt. Interrupt pacing increases the C64x+ DSP cache hit ratio by minimizing the number of times that large interrupt service routines are moved to and from the DSP instruction cache.

**MDIO**

The MDIO module manages the PHY configuration and monitors status. For a list of supported registers and register fields, see Table 6-84. In 10/100 mode, the GMII_MTXD(7:0) data bus uses only the lower nibble.

**SGMII**

The SGMII/SerDes module contains:

- Gigabit differential current mode logic (CML) receiver/transmitters
- An integrated RX/TX PLL to provide the required high-quality/high-speed internal clocks
- Phase-interpolator-based clock/data recovery
- A bandgap reference for transmitter swing settings
- Parallel-to-serial converter
- Serial-to-parallel converter
- Integrated receiver and transmitter termination
- Configuration logic
- 802.3 auto-negotiation functionality (as defined in Clause 37 of the IEEE Specification 802.3)

The SGMII receive interface converts the encoded receive signals from the differential receive input terminals (SGMII0RXN: SGMII0RXP, SGMII1RXN: SGMII1RXP) into the required GMAC GMII signals. The SGMII transmit interface converts the GMAC GMII data into the required encoded differential transmit output terminals (SGMII0TXN: SGMII0TXP, SGMII1TXN: SGMII1TXP). The GMAC does not source the transmit error signal. Any transmit frame from the GMAC with an error (ie., underrun) will be indicated as an error by an error CRC.

### 6.18.3 Peripheral Register Description(s)

Table 6-83 through Table 6-86 list the registers.

**Table 6-83. Ethernet Switch Registers**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 3000 | CPSW_ID_VER | CPSW Identification and Version Register |
| 0x02D0 3004 | CPSW_CONTROL | CPSW Switch Control Register |
| 0x02D0 3008 | CPSW_SOFT_RESET | CPSW Soft Reset Register |
| 0x02D0 300C | CPSW_STAT_PORT_EN | CPSW Statistics Port Enable Register |
| 0x02D0 3010 | CPSW_PTYPE | CPSW Transmit Priority Type Register |
| 0x02D0 3014 | P0_MAX_BLKS | CPSW Port 0 Maximum FIFO blocks Register |
| 0x02D0 3018 | P0_BLK_CNT | CPSW Port 0 FIFO Block Usage Count Register (read only) |
| 0x02D0 301C | P0_FLOW_THRESH | CPSW Port 0 Flow Control Threshold Register |
| 0x02D0 3020 | P0_PORT_VLAN | CPSW Port 0 VLAN Register |
| 0x02D0 3024 | P0_TX_PRI_MAP | CPSW Port 0 Tx Header Pri to Switch Pri Mapping Register |
| 0x02D0 3028 | GMAC0_GAP_THRESH | CPSW GMAC0 Short Gap Threshold Register |
| 0x02D0 302C | GMAC0_SA_LO | CPSW GMAC0 Source Address Low Register |
| 0x02D0 3030 | GMAC0_SA_HI | CPSW GMAC0 Source Address High Register |
| 0x02D0 3034 | P1_MAX_BLKS | CPSW Port 1 Maximum FIFO blocks Register |
| 0x02D0 3038 | P1_BLK_CNT | CPSW Port 1 FIFO Block Usage Count Register (read only) |
| 0x02D0 303C | P1_FLOW_THRESH | CPSW Port 1 Flow Control Threshold Register |
| 0x02D0 3040 | P1_PORT_VLAN | CPSW Port 1 VLAN Register |
| 0x02D0 3044 | P1_TX_PRI_MAP | CPSW Port 1 Tx Header Priority to Switch Pri Mapping Register |
| 0x02D0 3048 | GMAC1_GAP_THRESH | CPSW GMAC1 Short Gap Threshold Register |
| 0x02D0 304C | GMAC1_SA_LO | CPSW GMAC1 Source Address Low Register |
| 0x02D0 3050 | GMAC1_SA_HI | CPSW GMAC1 Source Address High Register |

**Table 6-83. Ethernet Switch Registers (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 3054 | P2_MAX_BLKS | CPSW Port 2 Maximum FIFO blocks Register |
| 0x02D0 3058 | P2_BLK_CNT | CPSW Port 2 FIFO Block Usage Count Register (read only) |
| 0x02D0 305C | P2_FLOW_THRESH | CPSW Port 2 Flow Control Threshold Register |
| 0x02D0 3060 | P2_PORT_VLAN | CPSW Port 2 VLAN Register |
| 0x02D0 3064 | P2_TX_PRI_MAP | CPSW Port 2 Tx (CPDMA Rx) Header Priority to Switch Pri Mapping Register |
| 0x02D0 3068 | CPDMA_TX_PRI_MAP | CPSW CPDMA Tx (Port 2 Rx) Pkt Priority to Header Priority Mapping Register |
| 0x02D0 306C | CPDMA_RX_CH_MAP | CPSW CPDMA Rx (Port 2 Tx) Switch Priority to DMA channel Mapping Register |
| 0x02D0 3070 - 0x02D0 307C | reserved | |
| 0x02D0 3080 | GMAC0_IDVER | GMAC0 Identification and Version Register |
| 0x02D0 3084 | GMAC0_MACCONTROL | GMAC0 Mac Control Register |
| 0x02D0 3088 | GMAC0_MACSTATUS | GMAC0 Mac Status Register |
| 0x02D0 308C | GMAC0_SOFT_RESET | GMAC0 Soft Reset Register |
| 0x02D0 3090 | GMAC0_RX_MAXLEN | GMAC0 RX Maximum Length Register |
| 0x02D0 3094 | GMAC0_BOFFTEST | GMAC0 Backoff Test Register |
| 0x02D0 3098 | reserved | |
| 0x02D0 309C | reserved | |
| 0x02D0 30A0 | GMAC0_EMCONTROL | GMAC0 Emulation Control Register |
| 0x02D0 30A4 | GMAC0_RX_PRI_MAP | GMAC0 Rx Pkt Priority to Header Priority Mapping Register |
| 0x02D0 30A8 - 0x02D0 30BC | reserved | |
| 0x02D0 30C0 | GMAC1_IDVER | GMAC1 Identification and Version Register |
| 0x02D0 30C4 | GMAC1_MACCONTROL | GMAC1 Mac Control Register |
| 0x02D0 30C8 | GMAC1_MACSTATUS | GMAC1 Mac Status Register |
| 0x02D0 30CC | GMAC1_SOFT_RESET | GMAC1 Soft Reset Register |
| 0x02D0 30D0 | GMAC1_RX_MAXLEN | GMAC1 RX Maximum Length Register |
| 0x02D0 30D4 | GMAC1_BOFFTEST | GMAC1 Backoff Test Register |
| 0x02D0 30D8 | reserved | |
| 0x02D0 30DC | reserved | |
| 0x02D0 30E0 | GMAC1_EMCONTROL | GMAC1 Emulation Control Register |
| 0x02D0 30E4 | GMAC1_RX_PRI_MAP | GMAC1 Rx Pkt Priority to Header Priority Mapping Register |
| 0x02D0 30E8 - 0x02D0 30FC | reserved | |
| 0x02D0 3100 | TX_IDVER | CPDMA Tx Identification and Version Register |
| 0x02D0 3104 | TX_CONTROL | CPDMA Tx Control Register |
| 0x02D0 3108 | TX_TEARDOWN | CPDMA Tx Teardown Register |
| 0x02D0 310C | reserved | |
| 0x02D0 3110 | RX_IDVER | CPDMA Rx Identification and Version Register |
| 0x02D0 3114 | RX_CONTROL | CPDMA Rx Control Register |
| 0x02D0 3118 | RX_TEARDOWN | CPDMA Rx Teardown Register |
| 0x02D0 311C | SOFT_RESET | CPDMA Soft Reset Register |
| 0x02D0 3120 | DMACONTROL | CPDMA Control Register |
| 0x02D0 3124 | DMASTATUS | CPDMA Status Register |
| 0x02D0 3128 | RX_BUFFER_OFFSET | CPDMA Rx Buffer Offset Register |
| 0x02D0 312C | EMCONTROL | CPDMA Emulation Control Register |
| 0x02D0 3130 - 0x02D0 317C | reserved | |
| 0x02D0 3180 | TX_INTSTAT_RAW | CPDMA Tx interrupt Status Register (raw value) |
| 0x02D0 3184 | TX_INTSTAT_MASKED | CPDMA Tx Interrupt Status Register (masked value) |

**Table 6-83. Ethernet Switch Registers (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 3188 | TX_INTMASK_SET | CPDMA Tx Interrupt Mask Set Register |
| 0x02D0 318C | TX_INTMASK_CLEAR | CPDMA Tx Interrupt Mask Clear Register |
| 0x02D0 3190 | CPDMA_IN_VECTOR | CPDMA Input Vector Register (read only) |
| 0x02D0 3194 | CPDMA_EOI_VECTOR | CPDMA End Of Interrupt Vector Register |
| 0x02D0 3198 - 0x02D0 319C | reserved | |
| 0x02D0 31A0 | RX_INTSTAT_RAW | CPDMA Rx Interrupt Status Register (raw value) |
| 0x02D0 31A4 | RX_INTSTAT_MASKED | CPDMA Rx Interrupt Status Register (masked value) |
| 0x02D0 31A8 | RX_INTMASK_SET | CPDMA Rx Interrupt Mask Set Register |
| 0x02D0 31AC | RX_INTMASK_CLEAR | CPDMA Rx Interrupt Mask Clear Register |
| 0x02D0 31B0 | DMA_INTSTAT_RAW | CPDMA DMA Interrupt Status Register (raw value) |
| 0x02D0 31B4 | DMA_INTSTAT_MASKED | CPDMA DMA Interrupt Status Register (masked value) |
| 0x02D0 31B8 | DMA_INTMASK_SET | CPDMA DMA Interrupt Mask Set Register |
| 0x02D0 31BC | DMA_INTMASK_CLEAR | CPDMA DMA Interrupt Mask Clear Register |
| 0x02D0 31C0 | RX0_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 0 |
| 0x02D0 31C4 | RX1_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 1 |
| 0x02D0 31C8 | RX2_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 2 |
| 0x02D0 31CC | RX3_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 3 |
| 0x02D0 31D0 | RX4_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 4 |
| 0x02D0 31D4 | RX5_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 5 |
| 0x02D0 31D8 | RX6_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 6 |
| 0x02D0 31DC | RX7_PENDTHRESH | CPDMA Rx Threshold Pending Register Channel 7 |
| 0x02D0 31E0 | RX0_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 0 |
| 0x02D0 31E4 | RX1_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 1 |
| 0x02D0 31E8 | RX2_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 2 |
| 0x02D0 31EC | RX3_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 3 |
| 0x02D0 31F0 | RX4_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 4 |
| 0x02D0 31F4 | RX5_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 5 |
| 0x02D0 31F8 | RX6_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 6 |
| 0x02D0 31FC | RX7_FREEBUFFER | CPDMA Rx Free Buffer Register Channel 7 |
| 0x02D0 3200 | TX0_HDP | CPDMA Tx Channel 0 Head Desc Pointer |
| 0x02D0 3204 | TX1_HDP | CPDMA Tx Channel 1 Head Desc Pointer |
| 0x02D0 3208 | TX2_HDP | CPDMA Tx Channel 2 Head Desc Pointer |
| 0x02D0 320C | TX3_HDP | CPDMA Tx Channel 3 Head Desc Pointer |
| 0x02D0 3210 | TX4_HDP | CPDMA Tx Channel 4 Head Desc Pointer |
| 0x02D0 3214 | TX5_HDP | CPDMA Tx Channel 5 Head Desc Pointer |
| 0x02D0 3218 | TX6_HDP | CPDMA Tx Channel 6 Head Desc Pointer |
| 0x02D0 321C | TX7_HDP | CPDMA Tx Channel 7 Head Desc Pointer |
| 0x02D0 3220 | RX0_HDP | CPDMA Rx 0 Channel 0 Head Desc Pointer |
| 0x02D0 3224 | RX1_HDP | CPDMA Rx 1 Channel 1 Head Desc Pointer |
| 0x02D0 3228 | RX2_HDP | CPDMA Rx 2 Channel 2 Head Desc Pointer |
| 0x02D0 322C | RX3_HDP | CPDMA Rx 3 Channel 3 Head Desc Pointer |
| 0x02D0 3230 | RX4_HDP | CPDMA Rx 4 Channel 4 Head Desc Pointer |
| 0x02D0 3234 | RX5_HDP | CPDMA Rx 5 Channel 5 Head Desc Pointer |
| 0x02D0 3238 | RX6_HDP | CPDMA Rx 6 Channel 6 Head Desc Pointer |
| 0x02D0 323C | RX7_HDP | CPDMA Rx 7 Channel 7 Head Desc Pointer |
| 0x02D0 3240 | TX0_CP | CPDMA Tx Channel 0 Completion Pointer Register |
| 0x02D0 3244 | TX1_CP | CPDMA Tx Channel 1 Completion Pointer Register |

**Table 6-83. Ethernet Switch Registers (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 3248 | TX2_CP | CPDMA Tx Channel 2 Completion Pointer Register |
| 0x02D0 324C | TX3_CP | CPDMA Tx Channel 3 Completion Pointer Register |
| 0x02D0 3250 | TX4_CP | CPDMA Tx Channel 4 Completion Pointer Register |
| 0x02D0 3254 | TX5_CP | CPDMA Tx Channel 5 Completion Pointer Register |
| 0x02D0 3258 | TX6_CP | CPDMA Tx Channel 6 Completion Pointer Register |
| 0x02D0 325C | TX7_CP | CPDMA Tx Channel 7 Completion Pointer Register |
| 0x02D0 3260 | RX0_CP | CPDMA Rx Channel 0 Completion Pointer Register |
| 0x02D0 3264 | RX1_CP | CPDMA Rx Channel 1 Completion Pointer Register |
| 0x02D0 3268 | RX2_CP | CPDMA Rx Channel 2 Completion Pointer Register |
| 0x02D0 326C | RX3_CP | CPDMA Rx Channel 3 Completion Pointer Register |
| 0x02D0 3270 | RX4_CP | CPDMA Rx Channel 4 Completion Pointer Register |
| 0x02D0 3274 | RX5_CP | CPDMA Rx Channel 5 Completion Pointer Register |
| 0x02D0 3278 | RX6_CP | CPDMA Rx Channel 6 Completion Pointer Register |
| 0x02D0 327C | RX7_CP | CPDMA Rx Channel 7 Completion Pointer Register |
| 0x02D0 3280 - 0x02D0 32BF | reserved | |
| 0x02D0 32C0 - 0x02D0 32FC | reserved | |
| 0x02D0 3300 - 0x02D0 337C | reserved | |
| 0x02D0 3380 - 0x02D0 33FC | reserved | |
| 0x02D0 3400 | RXGOODFRAMES | CPSW_STATS Total number of good frames received |
| 0x02D0 3404 | RXBROADCASTFRAMES | CPSW_STATS Total number of good broadcast frames received |
| 0x02D0 3408 | RXMULTICASTFRAMES | CPSW_STATS Total number of good multicast frames received |
| 0x02D0 340C | RXPAUSEFRAMES | CPSW_STATS PauseRxFrames |
| 0x02D0 3410 | RXCRCERRORS | CPSW_STATS Total number of CRC errors frames received |
| 0x02D0 3414 | RXALIGNCODEERRORS | CPSW_STATS Total number of alignment/code errors received |
| 0x02D0 3418 | RXOVERSIZEDFRAMES | CPSW_STATS Total number of oversized frames received |
| 0x02D0 341C | RXJABBERFRAMES | CPSW_STATS Total number of jabber frames received |
| 0x02D0 3420 | RXUNDERSIZEDFRAMES | CPSW_STATS Total number of undersized frames received |
| 0x02D0 3424 | RXFRAGMENTS | CPSW_STATS RxFragments received |
| 0x02D0 3428 | reserved | |
| 0x02D0 342C | reserved | |
| 0x02D0 3430 | RXOCTETS | CPSW_STATS Total number of received bytes in good frames |
| 0x02D0 3434 | TXGOODFRAMES | CPSW_STATS GoodTxFrames |
| 0x02D0 3438 | TXBROADCASTFRAMES | CPSW_STATS BroadcastTxFrames |
| 0x02D0 343C | TXMULTICASTFRAMES | CPSW_STATS MulticastTxFrames |
| 0x02D0 3440 | TXPAUSEFRAMES | CPSW_STATS PauseTxFrames |
| 0x02D0 3444 | TXDEFERREDFRAMES | CPSW_STATS Deferred Frames |
| 0x02D0 3448 | TXCOLLISIONFRAMES | CPSW_STATS Collisions |
| 0x02D0 344C | TXSINGLECOLLFRAMES | CPSW_STATS SingleCollisionTxFrames |
| 0x02D0 3450 | TXMULTCOLLFRAMES | CPSW_STATS MultipleCollisionTxFrames |
| 0x02D0 3454 | TXEXCESSIVECOLLISIONS | CPSW_STATS ExcessiveCollisions |
| 0x02D0 3458 | TXLATECOLLISIONS | CPSW_STATS LateCollisions |
| 0x02D0 345C | TXUNDERRUN | CPSW_STATS Transmit Underrun Error |
| 0x02D0 3460 | TXCARRIERSENSEERRORS | CPSW_STATS CarrierSenseErrors |
| 0x02D0 3464 | TXOCTETS | CPSW_STATS TxOctets |

**Table 6-83. Ethernet Switch Registers (continued)**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 3468 | OCTETFRAMES64 | CPSW_STATS 64octetFrames |
| 0x02D0 346C | OCTETFRAMES65T127 | CPSW_STATS 65-127octetFrames |
| 0x02D0 3470 | OCTETFRAMES128T255 | CPSW_STATS 128-255octetFrames |
| 0x02D0 3474 | OCTETFRAMES256T511 | CPSW_STATS 256-511octetFrames |
| 0x02D0 3478 | OCTETFRAMES512T1023 | CPSW_STATS 512-1023octetFrames |
| 0x02D0 347C | OCTETFRAMES1024TUP | CPSW_STATS 1023-1518octetFrames |
| 0x02D0 3480 | NETOCTETS | CPSW_STATS NetOctets |
| 0x02D0 3484 | RXSOFOVERRUNS | CPSW_STATS Receive FIFO or DMA Start of Frame Overruns |
| 0x02D0 3488 | RXMOFOVERRUNS | CPSW_STATS Receive FIFO or DMA Mid of Frame Overruns |
| 0x02D0 348C | RXDMAOVERRUNS | CPSW_STATS Receive DMA Start of Frame and Middle of Frame Overruns |
| 0x02D0 3490 - 0x02D0 34FC | reserved | |
| 0x02D0 3500 | ALE_IDVER | ALE Identification and Version Register |
| 0x02D0 3504 | reserved | |
| 0x02D0 3508 | ALE_CONTROL | ALE Control Register |
| 0x02D0 350C | reserved | |
| 0x02D0 3510 | ALE_PRESCALE | ALE Prescale Register |
| 0x02D0 3514 | reserved | |
| 0x02D0 3518 | ALE_UNKNOWN_VLAN | ALE Unknown VLAN Register |
| 0x02D0 351C | reserved | |
| 0x02D0 3520 | ALE_TBLCTL | ALE Table Control Register |
| 0x02D0 3524 - 0x02D0 3530 | reserved | |
| 0x02D0 3534 | ALE_TBLW2 | ALE Table Word 2 Register |
| 0x02D0 3538 | ALE_TBLW1 | ALE Table Word 1 Register |
| 0x02D0 353C | ALE_TBLW0 | ALE Table Word 0 Register |
| 0x02D0 3540 | ALE_PORTCTL0 | ALE Port 0 Control Register |
| 0x02D0 3544 | ALE_PORTCTL1 | ALE Port 1 Control Register |
| 0x02D0 3548 | ALE_PORTCTL2 | ALE Port 2 Control Register |
| 0x02D0 354C - 0x02D0 37FF | reserved | |

**Table 6-84. Ethernet Subsystem Registers**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 2000 | IDVER | Identification and Version Register |
| 0x02D0 2004 | SOFT_RESET | Soft Reset Register |
| 0x02D0 2008 | EM_CONTROL | Emulation Control Register |
| 0x02D0 200C | INT_CONTROL | Interrupt Control Register |
| 0x02D0 2010 | RX_THRESH_EN | Receive Threshold Interrupt Enable Register |
| 0x02D0 2014 | RX_EN | Receive Interrupt Enable Register |
| 0x02D0 2018 | TX_EN | Transmit Interrupt Enable Register |
| 0x02D0 201C | MISC_EN | Misc Interrupt Enable Register |
| 0x02D0 2020 | RX_THRESH_STAT | Receive Threshold Masked Interrupt Status Register |
| 0x02D0 2024 | RX_STAT | Receive Interrupt Masked Interrupt Status Register |
| 0x02D0 2028 | TX_STAT | Transmit Interrupt Masked Interrupt Status Register |
| 0x02D0 202C | MISC_STAT | Misc Interrupt Masked Interrupt Status Register |
| 0x02D0 2030 | RX_IMAX | Receive Interrupts Per Millisecond |
| 0x02D0 2034 | TX_IMAX | Transmit Interrupts Per Millisecond |

**Table 6-85. SGMII0 Registers**

| HEX ADDRESS RANGE | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x02D0 4800 | IDVER | Identification and Version Register |
| 0x02D0 4804 | SOFT_RESET | Soft Reset Register |
| 0x02D0 4808 - 0x02D0 480C | Reserved | Reserved |
| 0x02D0 4810 | CONTROL | Control Register |
| 0x02D0 4814 | STATUS | Status Register (read only) |
| 0x02D0 4818 | MR_ADV_ABILITY | Advertised Ability Register |
| 0x02D0 481C | MR_NP_TX | Transmit Next Page Register |
| 0x02D0 4820 | MR_LP_ADV_ABILITY | Link Partner Advertised Ability (read only) |
| 0x02D0 4824 | MR_NP_RX | Link Partner Receive Next Page Register (read only) |
| 0x02D0 4828 - 0x02D0 482C | Reserved | Reserved |
| 0x02D0 4830 | Reserved | Reserved |
| 0x02D0 4834 | Reserved | Reserved |
| 0x02D0 4838 | Reserved | Reserved |
| 0x02D0 483C | Reserved | Reserved |
| 0x02D0 4840 | DIAG_CLEAR | Diagnostics Clear Register |
| 0x02D0 4844 | DIAG_CONTROL | Diagnostics Control Register |
| 0x02D0 4848 | DIAG_STATUS | Diagnostics Status Register (read only) |
| 0x02D0 484C - 0x02D0 487F | Reserved | Reserved |

**Table 6-86. SGMII1 Registers**

| HEX ADDRESS RANGE | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x02D0 4C00 | IDVER | Identification and Version Register |
| 0x02D0 4C04 | SOFT_RESET | Soft Reset Register |
| 0x02D0 4C08 - 0x02D0 4C0C | Reserved | Reserved |
| 0x02D0 4C10 | CONTROL | Control Register |
| 0x02D0 4C14 | STATUS | Status Register (read only) |
| 0x02D0 4C18 | MR_ADV_ABILITY | Advertised Ability Register |
| 0x02D0 4C1C | MR_NP_TX | Transmit Next Page Register |
| 0x02D0 4C20 | MR_LP_ADV_ABILITY | Link Partner Advertised Ability (read only) |
| 0x02D0 4C24 | MR_NP_RX | Link Partner Receive Next Page Register (read only) |
| 0x02D0 4C28 - 0x02D0 4C2C | Reserved | Reserved |
| 0x02D0 4C30 | Reserved | Reserved |
| 0x02D0 4C34 | Reserved | Reserved |
| 0x02D0 4C38 | Reserved | Reserved |
| 0x02D0 4C3C | Reserved | Reserved |
| 0x02D0 4C40 | DIAG_CLEAR | Diagnostics Clear Register |
| 0x02D0 4C44 | DIAG_CONTROL | Diagnostics Control Register |
| 0x02D0 4C48 | DIAG_STATUS | Diagnostics Status Register (read only) |
| 0x02D0 4C4C - 0x02D0 4C7F | Reserved | Reserved |

### 6.18.4 Ethernet Subsystem Timing

**Table 6-87. Ethernet Subsystem Timing Requirements**

| | PARAMETER(1) | MIN | NOM | MAX | UNITS |
|---|---|---|---|---|---|
| $t_{01}$ | REFCLKP/N period, mode | | | | |
| | x 10 mode | | 8 | | ns |

(1) C = REFCLKP/N period in µs.

**Table 6-87. Ethernet Subsystem Timing Requirements (continued)**

| | PARAMETER[1] | MIN | NOM | MAX | UNITS |
|---|---|---|---|---|---|
| | x 20 mode | | 16 | | ns |
| | x 25 mode | | 20 | | ns |
| $t_{02}$ | REFCLKP/N duty cycle | 40 | | 60 | % |
| $t_{03}$ | REFCLKP/N rise/fall | | 700 | | ps |
| $t_{04}$ | PLL Clock Period, x *n* Mode | | $t_{01}$ / *n* | | ns |
| $t_{05}$ | PLL power up | | | 1 + 200 * C | µs |

### REFCLKP/N Jitter and PLL Loop Bandwidth

Jitter on the reference clock will degrade both the transmit eye and receiver jitter tolerance thereby impairing system performance. A good quality, low jitter reference clock is necessary to achieve compliance with most if not all physical layer standards (see Table 6-88).

**Table 6-88. REFCLKP/N Jitter Requirements for Standards Compliance**

| Standard | Line Rate (Gbps) | Total REFCLKP/N Jitter (within PLL bandwidth) |
|---|---|---|
| Gigabit Ethernet | 1.25 | 50 ps pk-pk |

## 6.19 Management Data Input/Output (MDIO)

The management data input/output (MDIO) module continuously polls all 32 MDIO addresses to enumerate all PHY devices in the system. It contains two user access registers to control and monitor up to two PHYs simultaneously.

The MDIO module implements the 802.3 serial management interface to interrogate and control two Ethernet PHYs simultaneously using a shared two-wire bus.

### *6.19.1 MII Management Interface*

Host software uses the MDIO module to configure the auto-negotiation parameters of each PHY attached to the Ethernet Switch Subsystem, retrieve the negotiation results, and configure required parameters in the Ethernet Switch Subsystem module for correct operation. The module is designed to allow almost transparent operation of the MDIO interface, with very little maintenance from the core processor. Only a maximum of two PHYs may be connected at any given time.

For more detailed information on the Ethernet Switch Subsystem, see the *TMS320C6452 DSP 3 Port Switch (3PSW) Ethernet Subsystem User's Guide* (SPRUF97). For a list of supported registers and register fields, see Table 6-89.

### *6.19.2 MDIO Register Descriptions*

**Table 6-89. MDIO Registers**

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---|---|---|
| 0x02D0 4000 | MDIOVER | Module version register |
| 0x02D0 4004 | MDIOCONTROL | Module control register |
| 0x02D0 4008 | MDIOALIVE | PHY acknowledge status register |
| 0x02D0 400C | MDIOLINK | PHY link status register |
| 0x02D0 4010 | MDIOLINKINTRAW | Link status change interrupt register (raw value) |
| 0x02D0 4014 | MDIOLINKINTMASKED | Link status change interrupt register (masked value) |
| 0x02D0 4018 - 0x02D0 401C | reserved | |
| 0x02D0 4020 | MDIOUSERINTRAW | User command complete interrupt register (raw value) |
| 0x02D0 4024 | MDIOUSERINTMASKED | User command complete interrupt register (masked value) |
| 0x02D0 4028 | MDIOUSERINTMASKSET | User interrupt mask set register |
| 0x02D0 402C | MDIOUSERINTMASKCLR | User interrupt mask clear register |
| 0x02D0 4030 - 0x02D0 407C | reserved | |
| 0x02D0 4080 | MDIOUSERACCESS0 | User access register0 |
| 0x02D0 4084 | MDIOUSERPHYSEL0 | User PHY select register0 |
| 0x02D0 4088 | MDIOUSERACCESS1 | User access register1 |
| 0x02D0 408C | MDIOUSERPHYSEL1 | User PHY select register1 |
| 0x02D0 4090 - 0x02D0 40FF | reserved | |

## 6.20 Timers

The device has four 64-bit general-purpose timers of which only Timer 0 and Timer 1 have external input/output. The timers can be used to: time events, count events, generate pulses, interrupt the CPU, and send synchronization events to the EDMA3 channel controller.

### 6.20.1 General-Purpose Timers

Each timer can be programmed as a 64-bit timer or as two separate 32-bit timers. Each timer is made up of two 32-bit counters: a high counter and a low counter. The timer pins, TINPLx and TOUTLx are connected to the low counter. The high counter does not have any external device pins.

For more detailed information, see the *TMS320C6452 DSP 64-Bit Timer User's Guide* (literature number SPRUF90).

### 6.20.2 Timer Peripheral Register Descriptions

**Table 6-90. Timer 0 Registers**

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 4400 | PID12 | Peripheral Identification Register |
| 0x0204 4404 | EMUMGT_CLKSPD | Timer 0 Emulation Management/Clock Speed Register |
| 0x0204 4410 | TIM12 | Timer 0 Counter Register 12 |
| 0x0204 4414 | TIM34 | Timer 0 Counter Register 34 |
| 0x0204 4418 | PRD12 | Timer 0 Period Register 12 |
| 0x0204 441C | PRD34 | Timer 0 Period Register 34 |
| 0x0204 4420 | TCR | Timer 0 Control Register |
| 0x0204 4424 | TGCR | Timer 0 Global Control Register |
| 0x0x0204 4428 - 0x0204 44FF | - | Reserved |

**Table 6-91. Timer 1 Registers**

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 4800 | PID12 | Peripheral Identification Register |
| 0x0204 4804 | EMUMGT_CLKSPD | Timer 1 Emulation Management/Clock Speed Register |
| 0x0204 4810 | TIM12 | Timer 1 Counter Register 12 |
| 0x0204 4814 | TIM34 | Timer 1 Counter Register 34 |
| 0x0204 4818 | PRD12 | Timer 1 Period Register 12 |
| 0x0204 481C | PRD34 | Timer 1 Period Register 34 |
| 0x0204 4820 | TCR | Timer 1 Control Register |
| 0x0204 4824 | TGCR | Timer 1 Global Control Register |
| 0x0204 4828 - 0x0204 48FF | - | Reserved |

**Table 6-92. Timer 2 Registers**

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 4C00 | PID12 | Peripheral Identification Register |
| 0x0204 4C04 | EMUMGT_CLKSPD | Timer 2 Emulation Management/Clock Speed Register |
| 0x0204 4C10 | TIM12 | Timer 2 Counter Register 12 |
| 0x0204 4C14 | TIM34 | Timer 2 Counter Register 34 |
| 0x0204 4C18 | PRD12 | Timer 2 Period Register 12 |
| 0x0204 4C1C | PRD34 | Timer 2 Period Register 34 |
| 0x0204 4C20 | TCR | Timer 2 Control Register |
| 0x0204 4C24 | TGCR | Timer 2 Global Control Register |
| 0x0204 4C28 - 0x0204 4CFF | - | Reserved |

**Table 6-93. Timer 3 Registers**

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---|---|---|
| 0x0204 5000 | PID12 | Peripheral Identification Register |
| 0x0204 5004 | EMUMGT_CLKSPD | Timer 3 Emulation Management/Clock Speed Register |
| 0x0204 5010 | TIM12 | Timer 3 Counter Register 12 |
| 0x0204 5014 | TIM34 | Timer 3 Counter Register 34 |
| 0x0204 5018 | PRD12 | Timer 3 Period Register 12 |
| 0x0204 501C | PRD34 | Timer 3 Period Register 34 |
| 0x0204 5020 | TCR | Timer 3 Control Register |
| 0x0204 5024 | TGCR | Timer 3 Global Control Register |
| 0x0204 5028 - 0x0204 50FF | - | Reserved |

### 6.20.3 Timer Electrical Data/Timing

**Table 6-94. Timing Requirements for Timer Input(1) (see Figure 6-48)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{w(TIMIxH)}$ | Pulse duration, TIMIxH high | 12P(1) | | ns |
| 2 | $t_{w(TIMIxL)}$ | Pulse duration, TIMIxL low | 12P | | ns |

(1) P = 1/CPU clock frequency in ns.

**Table 6-95. Switching Characteristics for Timer Output**

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

| NO. | PARAMETER | | 720 900 | | | UNIT |
|---|---|---|---|---|---|---|
| | | | MIN | TYP | MAX | |
| 3 | $t_{w(TIMOxH)}$ | Pulse duration, TIMOxH high | 12P(1) | | | |
| 4 | $t_{w(TIMOxL)}$ | Pulse duration, TIMOxL low | 12P | | | |

(1) P = 1/CPU clock frequency in ns.

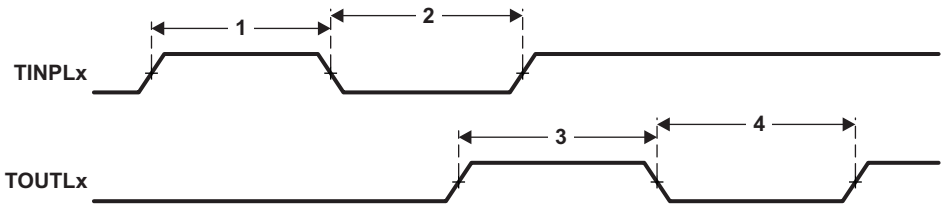



**Figure 6-48. Timer Timing**

## 6.21 VLYNQ Peripheral

### 6.21.1 VLYNQ Device-Specific Information

The VLYNQ peripheral conforms to the *VLYNQ Module Specification (revision 2.x)*. By default, the VLYNQ peripheral is initialized with a device ID of 0x22.

### 6.21.2 VLYNQ Peripheral Register Descriptions

**Table 6-96. VLYNQ Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x3800 0000 | - | Reserved |
| 0x3800 0004 | CTRL | VLYNQ Local Control Register |
| 0x3800 0008 | STAT | VLYNQ Local Status Register |
| 0x3800 000C | INTPRI | VLYNQ Local Interrupt Priority Vector Status/Clear Register |
| 0x3800 0010 | INTSTATCLR | VLYNQ Local Interrupt Status/Clear Register |
| 0x3800 0014 | INTPENDSET | VLYNQ Local Interrupt Pending/Set Register |
| 0x3800 0018 | INTPTR | VLYNQ Local Interrupt Pointer Register |
| 0x3800 001C | XAM | VLYNQ Local Transmit Address Map |
| 0x3800 0020 | RAMS1 | VLYNQ Local Receive Address Map Size 1 |
| 0x3800 0024 | RAMO1 | VLYNQ Local Receive Address Map Offset 1 |
| 0x3800 0028 | RAMS2 | VLYNQ Local Receive Address Map Size 2 |
| 0x3800 002C | RAMO2 | VLYNQ Local Receive Address Map Offset 2 |
| 0x3800 0030 | RAMS3 | VLYNQ Local Receive Address Map Size 3 |
| 0x3800 0034 | RAMO3 | VLYNQ Local Receive Address Map Offset 3 |
| 0x3800 0038 | RAMS4 | VLYNQ Local Receive Address Map Size 4 |
| 0x3800 003C | RAMO4 | VLYNQ Local Receive Address Map Offset 4 |
| 0x3800 0040 | CHIPVER | VLYNQ Local Chip Version Register |
| 0x3800 0044 | AUTNGO | VLYNQ Local Auto Negotiation Register |
| 0x3800 0048 | MANNGO | VLYNQ Local Manual Negotiation Register |
| 0x3800 004C | NGOSTAT | VLYNQ Local Negotiation Status Register |
| 0x3800 0050 - 0x3800 005C | - | Reserved |
| 0x3800 0060 | INTVEC0 | VLYNQ Local Interrupt Vector 3 - 0 |
| 0x3800 0064 | INTVEC1 | VLYNQ Local Interrupt Vector 7 - 4 |
| 0x3800 0068 - 0x3800 007C | - | Reserved *for future use* [Local Interrupt Vectors 8 - 31] |
| 0x3800 0080 | RREVID | VLYNQ Remote Revision Register |
| 0x3800 0084 | RCTRL | VLYNQ Remote Control Register |
| 0x3800 0088 | RSTAT | VLYNQ Remote Status Register |
| 0x3800 008C | RINTPRI | VLYNQ Remote Interrupt Priority Vector Status/Clear Register |
| 0x3800 0090 | RINTSTATCLR | VLYNQ Remote Interrupt Status/Clear Register |
| 0x3800 0094 | RINTPENDSET | VLYNQ Remote Interrupt Pending/Set Register |
| 0x3800 0098 | RINTPTR | VLYNQ Remote Interrupt Pointer Register |
| 0x3800 009C | RXAM | VLYNQ Remote Transmit Address Map |
| 0x3800 00A0 | RRAMS1 | VLYNQ Remote Receive Address Map Size 1 |
| 0x3800 00A4 | RRAMO1 | VLYNQ Remote Receive Address Map Offset 1 |
| 0x3800 00A8 | RRAMS2 | VLYNQ Remote Receive Address Map Size 2 |
| 0x3800 00AC | RRAMO2 | VLYNQ Remote Receive Address Map Offset 2 |
| 0x3800 00B0 | RRAMS3 | VLYNQ Remote Receive Address Map Size 3 |
| 0x3800 00B4 | RRAMO3 | VLYNQ Remote Receive Address Map Offset 3 |
| 0x3800 00B8 | RRAMS4 | VLYNQ Remote Receive Address Map Size 4 |
| 0x3800 00BC | RRAMO4 | VLYNQ Remote Receive Address Map Offset 4 |

**Table 6-96. VLYNQ Registers (continued)**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x3800 00C0 | RCHIPVER | VLYNQ Remote Chip Version Register |
| 0x3800 00C4 | RAUTNGO | VLYNQ Remote Auto Negotiation Register |
| 0x3800 00C8 | RMANNGO | VLYNQ Remote Manual Negotiation Register |
| 0x3800 00CC | RNGOSTAT | VLYNQ Remote Negotiation Status Register |
| 0x3800 00D0 - 0x3800 00DC | - | Reserved |
| 0x3800 00E0 | RINTVEC0 | VLYNQ Remote Interrupt Vector 3 - 0 |
| 0x3800 00E4 | RINTVEC1 | VYLNQ Remote Interrupt Vector 7 - 4 |
| 0x3800 00E8 - 0x3800 00FC | - | Reserved *for future use* [Remote Interrupt Vectors 8 - 31] |

### 6.21.3 VLYNQ Electrical Data/Timing

**Table 6-97. Timing Requirements for VCLK for VLYNQ (see Figure 6-49)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(VCLK)}$ | Cycle time, VCLK | 8 | | ns |
| 2 | $t_{w(VCLKH)}$ | Pulse duration, VCLK high, VCLK Input | 2 | | ns |
| | | Pulse duration, VCLK high, VCLK Output | 3 | | ns |
| 3 | $t_{w(VCLKL)}$ | Pulse duration, VCLK low, VCLK Input | 2 | | ns |
| | | Pulse duration, VCLK low, VCLK Output | 3 | | ns |
| 4 | $t_{t(VCLK)}$ | Transition time, VCLK | | | ns |

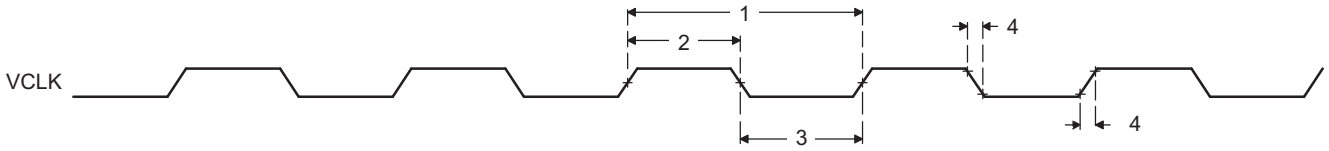



**Figure 6-49. VCLK Timing for VLYNQ**

### Table 6-98. Switching Characteristics

Over Recommended Operating Conditions for Transmit Data for the VLYNQ Module (see Figure 6-50)

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{d(VCLKH-TXDI)}$ | Delay time, VCLK high to VTXD[3:0] invalid [**SLOW** Mode] | 2.25 | | ns |
| 1 | $t_{d(VCLKH-TXDI)}$ | Delay time, VCLK high to VTXD[3:0] invalid [**FAST** Mode] | 0.86 | | ns |
| 2 | $t_{d(VCLKH-TXDV)}$ | Delay time, VCLK to VTXD[3:0] valid | | 6.85 | ns |

### Table 6-99. Timing Requirements for Receive Data for the VLYNQ Module (see Figure 6-50)

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 3 | $t_{su(RXDV-VCLKH)}$ | RTM disabled | 0.2 | | ns |
| | Setup time, VRXD[3:0] valid before VCLK high | RTM enabled, RXD Flop = 0 | 1.25 | | ns |
| | | RTM enabled, RXD Flop = 1 | 0.91 | | ns |
| | | RTM enabled, RXD Flop = 2 | 0.64 | | ns |
| | | RTM enabled, RXD Flop = 3 | 0.36 | | ns |
| | | RTM enabled, RXD Flop = 4 | 0.09 | | ns |
| | | RTM enabled, RXD Flop = 5 | -0.18 | | ns |
| | | RTM enabled, RXD Flop = 6 | -0.44 | | ns |
| | | RTM enabled, RXD Flop = 7 | -0.69 | | ns |
| 4 | $t_{h(VCLKH-RXDV)}$ | RTM disabled | 2 | | ns |
| | Hold time, VRXD[3:0] valid after VCLK high | RTM enabled, RXD Flop = 0 | 0.95 | | ns |
| | | RTM enabled, RXD Flop = 1 | 1.33 | | ns |
| | | RTM enabled, RXD Flop = 2 | 1.72 | | ns |
| | | RTM enabled, RXD Flop = 3 | 2.15 | | ns |
| | | RTM enabled, RXD Flop = 4 | 2.58 | | ns |
| | | RTM enabled, RXD Flop = 5 | 3.03 | | ns |
| | | RTM enabled, RXD Flop = 6 | 3.46 | | ns |
| | | RTM enabled, RXD Flop = 7 | 3.89 | | ns |

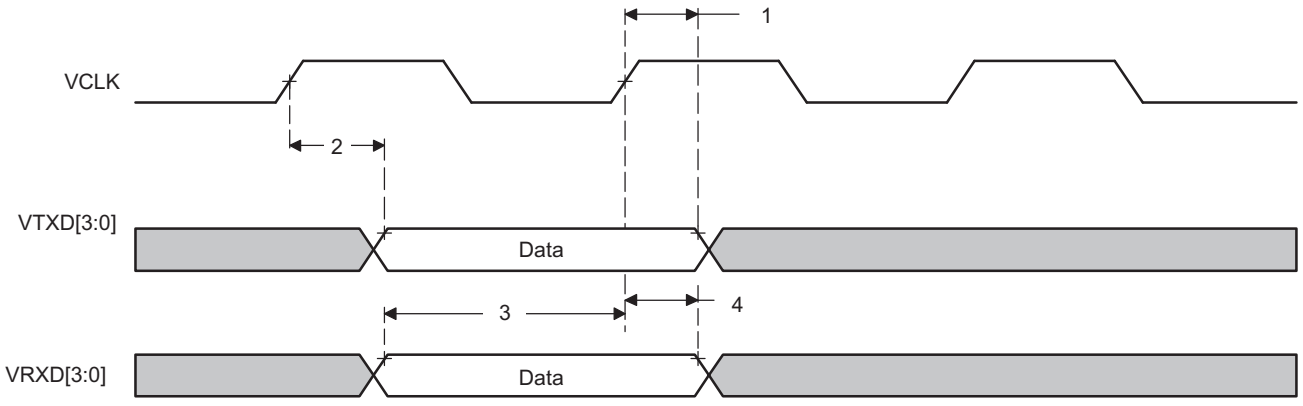



**Figure 6-50. VLYNQ Transmit/Receive Timing**

## 6.22 General-Purpose Input/Output (GPIO)

The GPIO peripheral provides general-purpose pins that can be configured as either inputs or outputs. When configured as an output, a write to an internal register can control the state driven on the output pin. When configured as an input, the state of the input is detectable by reading the state of an internal register. In addition, the GPIO peripheral can produce CPU interrupts and EDMA events in different interrupt/event generation modes. The GPIO peripheral provides generic connections to external devices. The GPIO pins are grouped into banks of 16 pins per bank (i.e., bank 0 consists of GPIO [0:15]).

The GPIO peripheral supports the following:

- Up to 3.3-V GPIO pins
- Interrupts:
  - Up to 16 unique GPIO[0:15] interrupts from Bank 0
  - One GPIO bank (aggregated) interrupt signal from the GPIOs in Bank 1
  - Interrupts can be triggered by rising and/or falling edge, specified for each interrupt capable GPIO signal
- DMA events:
  - Up to 10 unique GPIO DMA events from Bank 0
- Set/clear functionality: Firmware writes 1 to corresponding bit position(s) to set or to clear GPIO signal(s). This allows multiple firmware processes to toggle GPIO output signals without critical section protection (disable interrupts, program GPIO, re-enable interrupts, to prevent context switching to anther process during GPIO programming).
- Separate Input/Output registers
- Output register in addition to set/clear so that, if preferred by firmware, some GPIO output signals can be toggled by direct write to the output register(s).
- Output register, when read, reflects output drive status. This, in addition to the input register reflecting pin status and open-drain I/O cell, allows wired logic be implemented.

The memory map for the GPIO registers is shown in Table 6-100.

For more detailed information on GPIOs, see the *TMS320C6452 DSP General-Purpose Input/Output (GPIO) User's Guide* (literature number SPRUF95).

### 6.22.1 GPIO Peripheral Register Descriptions

**Table 6-100. GPIO Registers**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---|---|---|
| 0x0204 8000 | PID | Peripheral Identification Register |
| 0x0204 8004 | - | Reserved |
| 0x0204 8008 | BINTEN | GPIO interrupt per-bank enable |
| **GPIO Banks 0 and 1** | | |
| 0x0204 800C | - | Reserved |
| 0x0204 8010 | DIR | GPIO Banks 0 and 1 Direction Register (GPIO[0:31]) |
| 0x0204 8014 | OUT_DATA | GPIO Banks 0 and 1 Output Data Register (GPIO[0:31]) |
| 0x0204 8018 | SET_DATA | GPIO Banks 0 and 1 Set Data Register (GPIO[0:31]) |
| 0x0204 801C | CLR_DATA | GPIO Banks 0 and 1 Clear data for banks 0 and 1 (GPIO[0:31]) |
| 0x0204 8020 | IN_DATA | GPIO Banks 0 and 1 Input Data Register (GPIO[0:31]) |
| 0x0204 8024 | SET_RIS_TRIG | GPIO Banks 0 and 1 Set Rising Edge Interrupt Register (GPIO[0:31]) |
| 0x0204 8028 | CLR_RIS_TRIG | GPIO Banks 0 and 1 Clear Rising Edge Interrupt Register (GPIO[0:31]) |
| 0x0204 802C | SET_FAL_TRIG | GPIO Banks 0 and 1 Set Falling Edge Interrupt Register (GPIO[0:31]) |
| 0x0204 8030 | CLR_FAL_TRIG | GPIO Banks 0 and 1 Clear Falling Edge Interrupt Register (GPIO[0:31]) |
| 0x0204 8034 | INSTAT | GPIO Banks 0 and 1 Interrupt Status Register (GPIO[0:31]) |

### 6.22.2 GPIO Peripheral Input/Output Electrical Data/Timing

**Table 6-101. Timing Requirements for GPIO Inputs[1] (see Figure 6-51)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{w(GPIH)}$ | Pulse duration, GPIx high | 12P | | ns |
| 2 | $t_{w(GPIL)}$ | Pulse duration, GPIx low | 12P | | ns |

(1) The pulse width given is sufficient to generate a CPU interrupt or an EDMA event. However, if a user wants to have the device recognize the GPIx changes through software polling of the GPIO register, the GPIx duration must be extended to allow the device enough time to access the GPIO register through the internal bus. P = 1/CPU clock frequency in ns.

**Table 6-102. Switching Characteristics Over Recommended Operating Conditions for GPIO Outputs (see Figure 6-51)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 3 | $t_{w(GPOH)}$ | Pulse duration, GPOx high | 6P[1] | | ns |
| 4 | $t_{w(GPOL)}$ | Pulse duration, GPOx low | 6P[1] | | ns |

(1) This parameter value should not be used as a maximum performance specification. Actual performance of back-to-back accesses of the GPIO is dependent upon internal bus activity. P = 1/CPU clock frequency in ns.

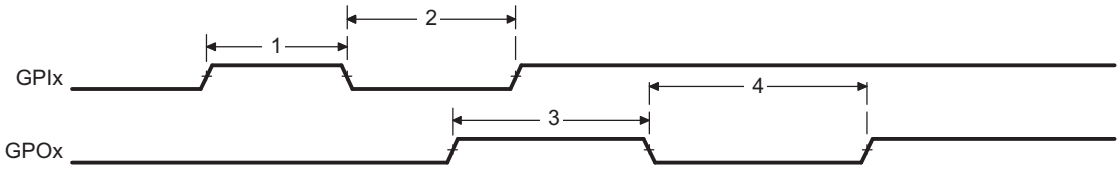



**Figure 6-51. GPIO Port Timing**

## 6.23 Emulation Features and Capability

### 6.23.1 Advanced Event Triggering (AET)

The device supports Advanced Event Triggering (AET). This capability can be used to debug complex problems as well as understand performance characteristics of user applications. AET provides the following capabilities:

- **Hardware Program Breakpoints:** specify addresses or address ranges that can generate events such as halting the processor or triggering the trace capture.
- **Data Watchpoints:** specify data variable addresses, address ranges, or data values that can generate events such as halting the processor or triggering the trace capture.
- **Counters:** count the occurrence of an event or cycles for performance monitoring.
- **State Sequencing:** allows combinations of hardware program breakpoints and data watchpoints to precisely generate events for complex sequences.

For more information on AET, see the following documents:

- *Using Advanced Event Triggering to Find and Fix Intermittent Real-Time Bugs Application Report* (literature number SPRA753)
- *Using Advanced Event Triggering to Debug Real-Time Problems in High Speed Embedded Microprocessor Systems Application Report* (literature number SPRA387)

### 6.23.2 Trace

The device supports Trace. Trace is a debug technology that provides a detailed, historical account of application code execution, timing, and data accesses. Trace collects, compresses, and exports debug information for analysis. Trace works in real-time and does not impact the execution of the system.

For more information on board design guidelines for Trace Advanced Emulation, see the *Emulation and Trace Headers Technical Reference Manual* (literature number SPRU655).

## 6.24 IEEE 1149.1 JTAG

The JTAG (2) interface is used for BSDL testing and emulation of the device.

$\overline{\text{TRST}}$ needs to be released only when it is necessary to use a JTAG controller to debug the device or exercise the device's boundary scan functionality. Note: $\overline{\text{TRST}}$ is synchronous and ***must*** be clocked by TCK; otherwise, the boundary scan logic may not respond as expected after $\overline{\text{TRST}}$ is asserted.

For maximum reliability, the device includes an internal pulldown (IPD) on the $\overline{\text{TRST}}$ pin to make certain that $\overline{\text{TRST}}$ will always be asserted upon power up and the device's internal emulation logic will always be properly initialized.

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

When using this type of JTAG controller, assert $\overline{\text{TRST}}$ to initialize the device after powerup and externally drive $\overline{\text{TRST}}$ high before attempting any emulation or boundary scan operations.

### 6.24.1 JTAG Peripheral Register Description(s) - JTAG ID Register

**Table 6-103. JTAG ID Register**

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|---|---|---|---|
| 0x0204 9018 | JTAGID | JTAG Identification Register | Read-only. Provides 32-bit JTAG ID of the device. |

(2) IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

The JTAG ID register is a read-only register that identifies to the customer the JTAG/Device ID. The JTAG ID register resides at address location 0x0204 9018. The register hex value is: 0x0B77 A02F . For the actual register bit names and their associated bit field descriptions, see Figure 6-52 and Table 6-104.

**Figure 6-52. JTAGID Register (0x0204 9018)**

| 31-28 | 27-12 | 11-1 | 0 |
|---|---|---|---|
| VERSION | PART NUMBER (16-Bit) | MANUFACTURER (11-Bit) | LSB |
| R-0001 | R-1011 0111 0111 1010 | R-0000 0010 111 | R-1 |

LEGEND: R = Read, W = Write, n = value at reset

**Table 6-104. JTAGID Register Selection Bit Descriptions**

| BIT | NAME | DESCRIPTION |
|---|---|---|
| 31:28 | VERSION | Silicon version |
| 27:12 | PART NUMBER | Part Number (16-Bit) value |
| 11-1 | MANUFACTURER | Manufacturer (11-Bit) value. |
| 0 | LSB | LSB |

### 6.24.2 JTAG Electrical Data/Timing

**Table 6-105. Timing Requirements for JTAG Test Port (see Figure 6-53)**

| NO. | | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 1 | $t_{c(TCK)}$ | Cycle time, TCK | 35 | | ns |
| 3 | $t_{su(TDIV-TCKH)}$ | Setup time, TDI/TMS/$\overline{\text{TRST}}$ valid before TCK high | 2 | | ns |
| 4 | $t_{h(TCKH-TDIV)}$ | Hold time, TDI/TMS/$\overline{\text{TRST}}$ valid after TCK high | 0 | | ns |

**Table 6-106. Switching Characteristics Over Recommended Operating Conditions for JTAG Test Port (see Figure 6-53)**

| NO. | PARAMETER | | 720 900 | | UNIT |
|---|---|---|---|---|---|
| | | | MIN | MAX | |
| 2 | $t_{d(TCKL-TDOV)}$ | Delay time, TCK low to TDO valid | 0 | 0.25 × $t_{c(TCK)}$ | ns |

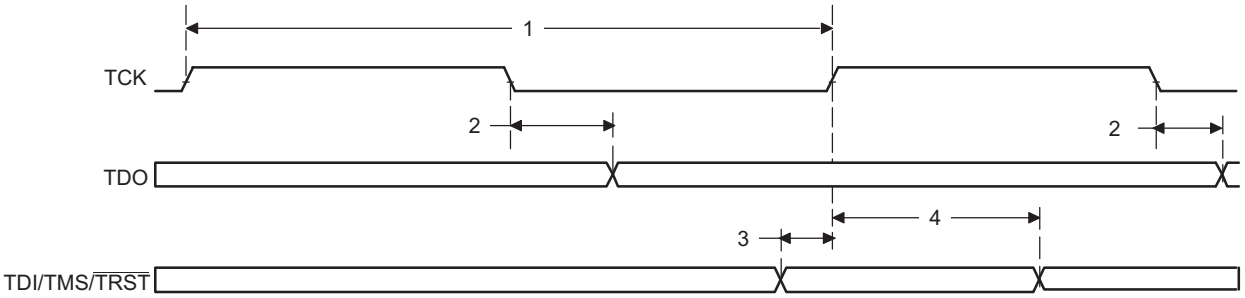



**Figure 6-53. JTAG Test-Port Timing**

# 7 Mechanical Data

The following table(s) show the thermal resistance characteristics for the ZUT mechanical package.

## 7.1 Thermal Data for ZUT

**Table 7-1. Thermal Resistance Characteristics (PBGA Package) [ZUT]**

| | | °C/W(1) | AIR FLOW (m/s)(2) |
|---|---|---|---|
| $R\Theta_{JC}$ | RΘJC Junction-to-case | 1.7 | N/A |
| $R\Theta_{JB}$ | RΘJB Junction-to-board | 8.6 | N/A |
| $R\Theta_{JA}$ | RΘJA Junction-to-free air | 17.2 | 0.0 |
| | | 13.7 | 1.0 |
| | | 12.4 | 2.0 |
| | | 11.5 | 3.0 |
| $\psi_{JT}$ | ψJT Junction-to-package top | 0.6 | 0.0 |
| | | 0.6 | 1.0 |
| | | 0.6 | 2.0 |
| | | 0.7 | 3.0 |
| $\psi i_{JB}$ | ψJB Junction-to-board | 8.4 | 0.0 |
| | | 7.4 | 1.0 |
| | | 7.0 | 2.0 |
| | | 6.7 | 3.0 |

(1) The junction-to-case measurement was conducted in a JEDEC defined 1S0P system. Other measurements were conducted in a JEDEC defined 1S2P system and will change based on environment as well as application.
For more information, see these three EIA/JEDEC standards:
- EIA/JESD51-2, *Integrated Circuits Thermal Test Method Environment Conditions - Natural Convection (Still Air)*
- EIA/JESD51-3, *Low Effective Thermal Conductivity Test Board for Leaded Surface Mount Packages*
- JESD51-9, *Test Boards for Area Array Surface Mount Package Thermal Measurements*

(2) m/s = meters per second

## 7.2 Packaging Information

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

## PACKAGING INFORMATION

| Orderable Device | Status (1) | Package Type | Package Drawing | Pins | Package Qty | Eco Plan (2) | Lead finish/ Ball material (6) | MSL Peak Temp (3) | Op Temp (°C) | Device Marking (4/5) | Samples |
|---|---|---|---|---|---|---|---|---|---|---|---|
| TMS320C6452CUT9 | NRND | FCBGA | CUT | 529 | | TBD | Call TI | Call TI | 0 to 0 | | |
| TMS320C6452ZUT9 | ACTIVE | FCBGA | ZUT | 529 | 84 | RoHS-Exempt & Green | SNAGCU | Level-4-245C-72HR | 0 to 0 | TMS320C6452ZUT @2007 TI 900MHZ | Samples |

(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) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
**RoHS Exempt:** TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
**Green:** TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

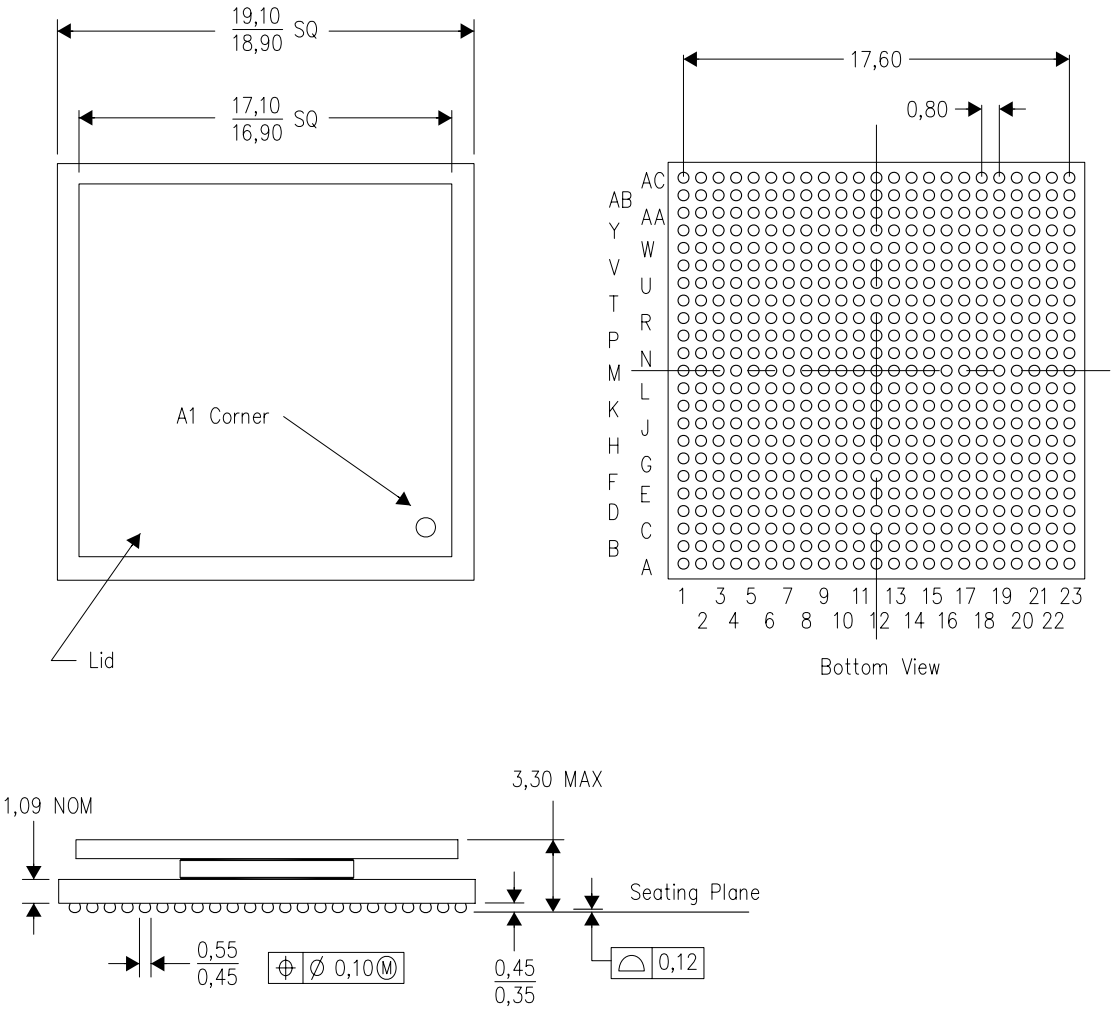
(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

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

ZUT (S–PBGA–N529) PLASTIC BALL GRID ARRAY




4207941/C 02/10

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

# IMPORTANT NOTICE AND DISCLAIMER

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

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

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

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

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

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