

TMS320C6424 Fixed-Point Digital Signal Processor

Check for Samples: [TMS320C6424](#)

1 TMS320C6424 Fixed-Point Digital Signal Processor

1.1 Features

- **High-Performance Digital Signal Processor (C6424)**
 - 2.5-, 2-, 1.67-, 1.43-ns Instruction Cycle Time
 - 400-, 500-, 600-, 700-MHz C64x+™ Clock Rate
 - Eight 32-Bit C64x+ Instructions/Cycle
 - 3200, 4000, 4800, 5600 MIPS
 - Fully Software-Compatible With C64x
 - Commercial and Automotive (Q or S suffix) Grades
 - Low-Power Device (L suffix)
- **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 [Flexible Allocation]
 - 640K-Bit (80K-Byte) L1D Data RAM/Cache [Flexible Allocation]
 - 1M-Bit (128K-Byte) L2 Unified Mapped RAM/Cache [Flexible Allocation]
- **Endianess: Supports Both Little Endian and Big Endian**
- **External Memory Interfaces (EMIFs)**
 - 32-Bit DDR2 SDRAM Memory Controller With 256M-Byte Address Space (1.8-V I/O)
 - Supports up to 333-MHz (data rate) bus and interfaces to DDR2-400 SDRAM
 - Asynchronous 16-Bit Wide EMIF (EMIFA) With up to 128M-Byte Address Reach
 - Flash Memory Interfaces
 - NOR (8-/16-Bit-Wide Data)
 - NAND (8-/16-Bit-Wide Data)
- **Enhanced Direct-Memory-Access (EDMA) Controller (64 Independent Channels)**
- **Two 64-Bit General-Purpose Timers (Each Configurable as Two 32-Bit Timers)**
- **One 64-Bit Watch Dog Timer**
- **Two UARTs (One with RTS and CTS Flow Control)**
- **Master/Slave Inter-Integrated Circuit (I²C Bus™)**
- **Two Multichannel Buffered Serial Ports (McBSPs)**
 - I2S and TDM
 - AC97 Audio Codec Interface
 - SPI
 - Standard Voice Codec Interface (AIC12)
 - Telecom Interfaces – ST-Bus, H-100
 - 128 Channel Mode
- **Multichannel Audio Serial Port (McASP0)**
 - Four Serializers and SPDIF (DIT) Mode
- **16-Bit Host-Port Interface (HPI)**
- **32-Bit 33-MHz, 3.3-V Peripheral Component Interconnect (PCI) Master/Slave Interface**



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

- **10/100 Mb/s Ethernet MAC (EMAC)**
 - IEEE 802.3 Compliant
 - Supports Multiple Media Independent Interfaces (MII, RMII)
 - Management Data Input/Output (MDIO) Module
- **VLYNQ™ Interface (FPGA Interface)**
- **Three Pulse Width Modulator (PWM) Outputs**
- **On-Chip ROM Bootloader**
- **Individual Power-Savings Modes**
- **Flexible PLL Clock Generators**
- **IEEE-1149.1 (JTAG™) Boundary-Scan-Compatible**
- **Up to 111 General-Purpose I/O (GPIO) Pins (Multiplexed With Other Device Functions)**
- **Packages:**
 - 361-Pin Pb-Free PBGA Package (ZWT Suffix), 0.8-mm Ball Pitch
 - 376-Pin Plastic BGA Package (ZDU Suffix), 1.0-mm Ball Pitch
- **0.09- μ m/6-Level Cu Metal Process (CMOS)**
- **3.3-V and 1.8-V I/O, 1.2-V Internal (-7/-6/-5/-4/Q6/-Q5/-Q4)**
- **3.3-V and 1.8-V I/O, 1.05-V Internal (-7/-6/-5/-4/-L/-Q5)**
- **Applications:**
 - Telecom
 - Audio
 - Industrial Applications
- **Community Resources**
 - [TI E2E Community](#)
 - [TI Embedded Processors Wiki](#)

1.2 Description

The TMS320C64x+™ DSPs (including the TMS320C6424 device) are the highest-performance fixed-point DSP generation in the TMS320C6000™ DSP platform. The C6424 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 signal processor 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 5600 million instructions per second (MIPS) at a clock rate of 700 MHz, 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 telecom, audio, and industrial applications. The DSP core can produce four 16-bit multiply-accumulates (MACs) per cycle for a total of 2800 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 5600 MMACS. For more details on the C64x+ DSP, see the *TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide* (literature number [SPRU732](#)).

The C6424 also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000 DSP platform devices. The C6424 core uses a two-level cache-based architecture. The Level 1 program memory/cache (L1P) consists of a 256K-bit memory space that can be configured as mapped memory or direct mapped cache, and the Level 1 data (L1D) consists of a 640K-bit memory space—384K-bit of which is mapped memory and 256K-bit of which can be configured as mapped memory or 2-way set-associative cache. The Level 2 memory/cache (L2) consists of a 1M-bit 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 peripheral set includes: a 10/100 Mb/s Ethernet MAC (EMAC) with a management data input/output (MDIO) module; a 4-bit transmit, 4-bit receive VLYNQ interface; an inter-integrated circuit (I2C) Bus interface; two multichannel buffered serial ports (McBSPs); a multichannel audio serial port (McASP0) with 4 serializers; 2 64-bit general-purpose timers each configurable as 2 independent 32-bit timers; 1 64-bit watchdog timer; a user-configurable 16-bit host-port interface (HPI); up to 111-pins of general-purpose

input/output (GPIO) with programmable interrupt/event generation modes, multiplexed with other peripherals; 2 UARTs with hardware handshaking support on 1 UART; 3 pulse width modulator (PWM) peripherals; 1 peripheral component interconnect (PCI) [33 MHz]; and 2 glueless external memory interfaces: an asynchronous external memory interface (EMIFA) for slower memories/peripherals, and a higher speed synchronous memory interface for DDR2.

The Ethernet Media Access Controller (EMAC) provides an efficient interface between the C6424 and the network. The C6424 EMAC supports 10Base-T and 100Base-TX, or 10 Mbits/second (Mbps) and 100 Mbps in either half- or full-duplex mode, with hardware flow control and quality of service (QOS) support.

The Management Data Input/Output (MDIO) module continuously polls all 32 MDIO addresses in order to enumerate all PHY devices in the system.

The I2C and VLYNQ ports allow C6424 to easily control peripheral devices 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 C6424 has 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.

1.3 Functional Block Diagram

Figure 1-1 shows the functional block diagram of the C6424 device.

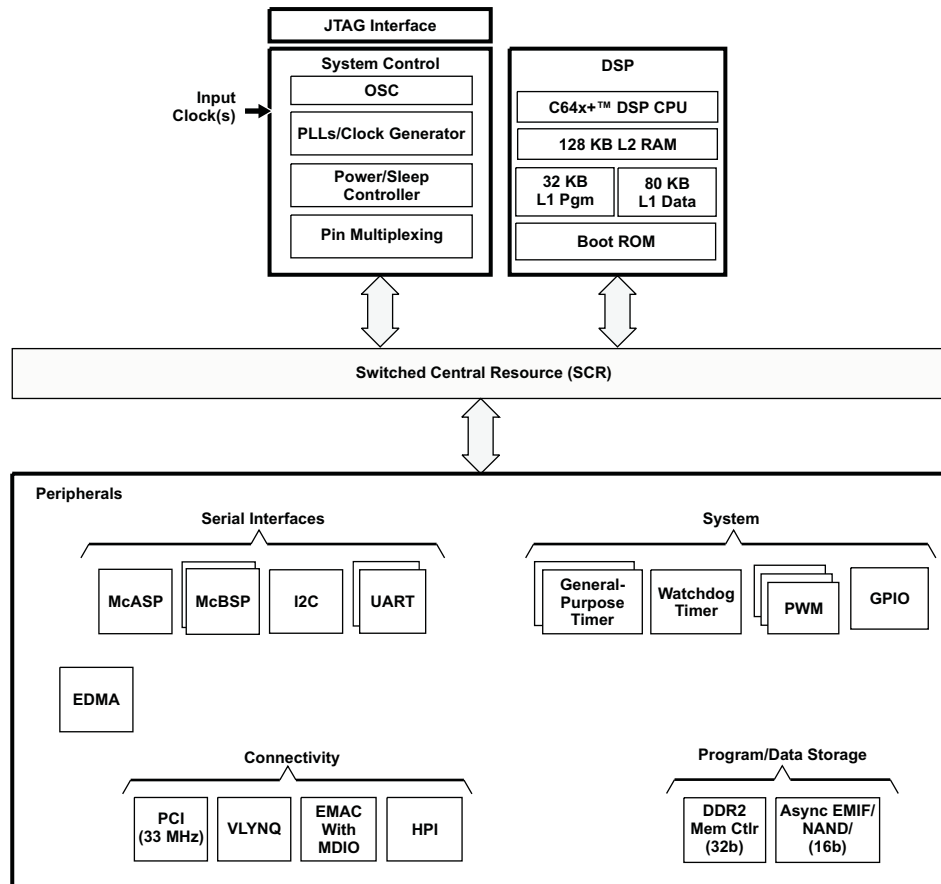


Figure 1-1. TMS320C6424 Functional Block Diagram

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 SPRS347C device-specific data manual to make it an SPRS347D revision.

Scope: Applicable updates to the TMS320C642x DSP, specifically relating to the TMS320C6424 device, have been incorporated.

- Added 700-MHz C64x+™ device speed.
- Added designators for low-power (-L) devices.
- Clarified the device operating frequencies that can run at 1.05 V core voltage.

| SEE | ADDITIONS/MODIFICATIONS/DELETIONS |
|---|--|
| Table 2-1 Characteristics of the C6424 Processor | Updated Table 2-1 , <i>Characteristics of the C6424 Processor</i> , to reflect changes to the CPU Frequency and Cycle Time at $V_{DD} = 1.05$ V. |
| Figure 2-14 Device Nomenclature | Updated/Changed Figure 2-14 , <i>Device Nomenclature</i> , to reflect device speed range. |

2 Device Overview

2.1 Device Characteristics

Table 2-1, provides an overview of the TMS320C6424 DSP. The tables show significant features of the C6424 device, 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 C6424 Processor

| HARDWARE FEATURES | | C6424 | |
|---|--|--|--|
| Peripherals Not all peripherals pins are available at the same time (for more detail, see the Device Configuration section). | DDR2 Memory Controller | (16-/32-bit bus width) [1.8 V I/O] | |
| | Asynchronous EMIF [EMIFA] | Asynchronous (8-/16-bit bus width), RAM, Flash, (NOR or NAND) | |
| | EDMA3 | 1 (64 independent channels, 8 QDMA channels) | |
| | Timers | 2 64-bit General Purpose (configurable as 2 64-bit or 4 32-bit) 1 64-bit Watch Dog | |
| | UART s | 2 (one with RTS and CTS flow control) | |
| | I2C | 1 (Master/Slave) | |
| | McBSPs | 2 | |
| | McASP | 1 (4 Serializers) | |
| | 10/100 Ethernet MAC (EMAC) with Management Data Input/Output (MDIO) | 1 | |
| | VLYNQ | 1 | |
| | General-Purpose Input/Output Port (GPIO) | Up to 111 pins | |
| | PWM | 3 outputs | |
| | HPI (16-bit) | 1 | |
| | PCI (32-bit), [33-MHz] | 1 | |
| On-Chip Memory | Size (Bytes) | 240KB RAM, 64KB ROM | |
| | Organization | 32K-Byte (32KB) L1 Program (L1P) RAM/Cache (Cache up to 32KB) 80KB L1 Data (L1D) RAM/Cache (Cache up to 32KB) 128KB Unified Mapped RAM/Cache (L2) 64KB Boot ROM | |
| MegaModule Rev ID | Revision ID Register (MM_REVID.[15:0]) (address location: 0x0181 2000) | See the <i>TMS320C6424/21 Digital Signal Processor (DSP) [Silicon Revisions 1.1 and 1.0] Silicon Errata</i> (literature number SPRZ252). | |
| CPU ID + CPU Rev ID | Control Status Register (CSR.[31:16]) | | |
| JTAG BSDL_ID | JTAGID register (address location: 0x01C4 0028) | See Section 6.22.1, JTAG ID (JTAGID) Register Description(s) | |
| CPU Frequency ⁽¹⁾ | MHz | CV _{DD} = 1.2V | 700 (-7) 600 (-6/-Q6) 500 (-5/-Q5) 400 (-4/-Q4) |
| | | CV _{DD} = 1.05V | 560 (-7) 450 (-6) 400 (-5/-Q5/-L) 350 (-4) |

(1) Applies to "tape and reel" part number counterparts as well. For more information, see [Section 2.7, Device and Development-Support Tool Nomenclature](#).

Table 2-1. Characteristics of the C6424 Processor (continued)

| HARDWARE FEATURES | | C6424 |
|-------------------------------|--|--|
| Cycle Time ⁽¹⁾ | ns | CV _{DD} = 1.2 V 1.43 (-7) 1.67 (-6/-Q6) 2 (-5/-Q5) 2.5 (-4/-Q4) |
| | | CV _{DD} = 1.05 V 1.79 (-7) 2.22 (-6) 2.5 (-5/-Q5/-L) 2.8 (-4) |
| Voltage ⁽¹⁾ | Core (V) | 1.2 V (-7/-6/-5/-4/-Q6/-Q5/-Q4) |
| | I/O (V) | 1.05 V (-7/-6/-5/-4/-L/-Q5) 1.8 V, 3.3 V |
| PLL Options | MXI/CLKIN frequency multiplier (15-30 MHz reference) | x1 (Bypass), x14 to x32 |
| BGA Package(s) | 16 x 16 mm, 0.8 mm pitch | 361-Pin BGA (ZWT) |
| | 23 x 23 mm, 1.0 mm pitch | 376-Pin BGA (ZDU) |
| Process Technology | µm | 0.09 µm |
| Product Status ⁽²⁾ | Product Preview (PP), Advance Information (AI), or Production Data (PD) | PD |

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

2.2 C64x+ Megamodule

The C64x+ Megamodule (Figure 2-1) consists of the following components:

- TMS320C64x+ CPU
- Internal memory controllers:
 - Level-1 program memory controller (L1P controller)
 - Level-1 data memory controller (L1D controller)
 - Level-2 unified memory controller (L2 controller)
 - External memory controller (EMC)
 - Internal direct memory access (IDMA) controller
- Internal peripherals
 - Interrupt controller (INTC)
 - Power-down controller (PDC)

The following C64x+ Megamodule standard features **are not** supported on the C6424 device:

- Memory Protection
- Power-down of the internal memories (L1P, L1D, and L2)

For more detailed information on the C64x+ Megamodule, see the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number [SPRU871](#)).

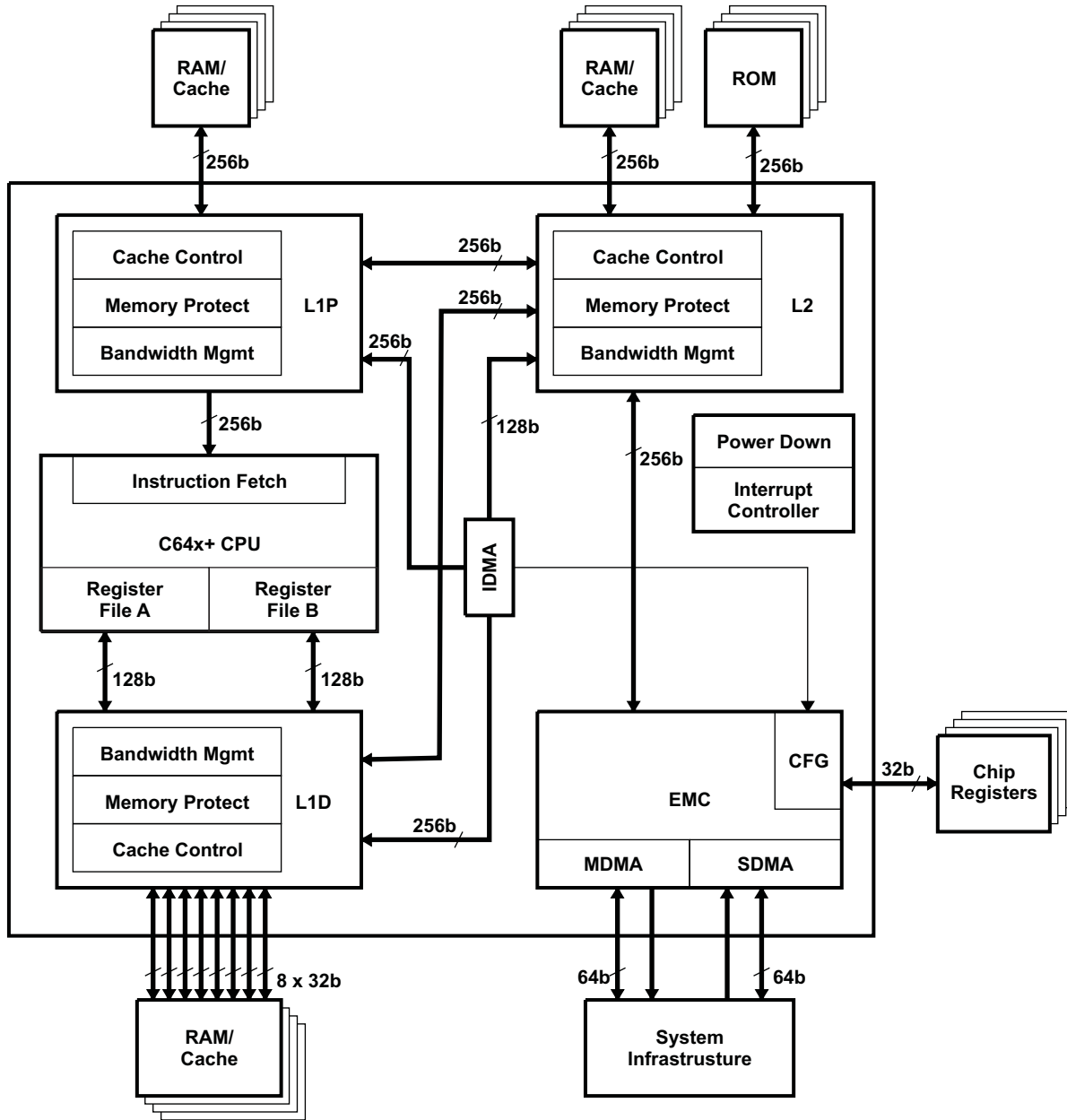


Figure 2-1. TMS320C64x+ Megamodule Block Diagram

2.2.1 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 2-2](#). 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 only available 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.

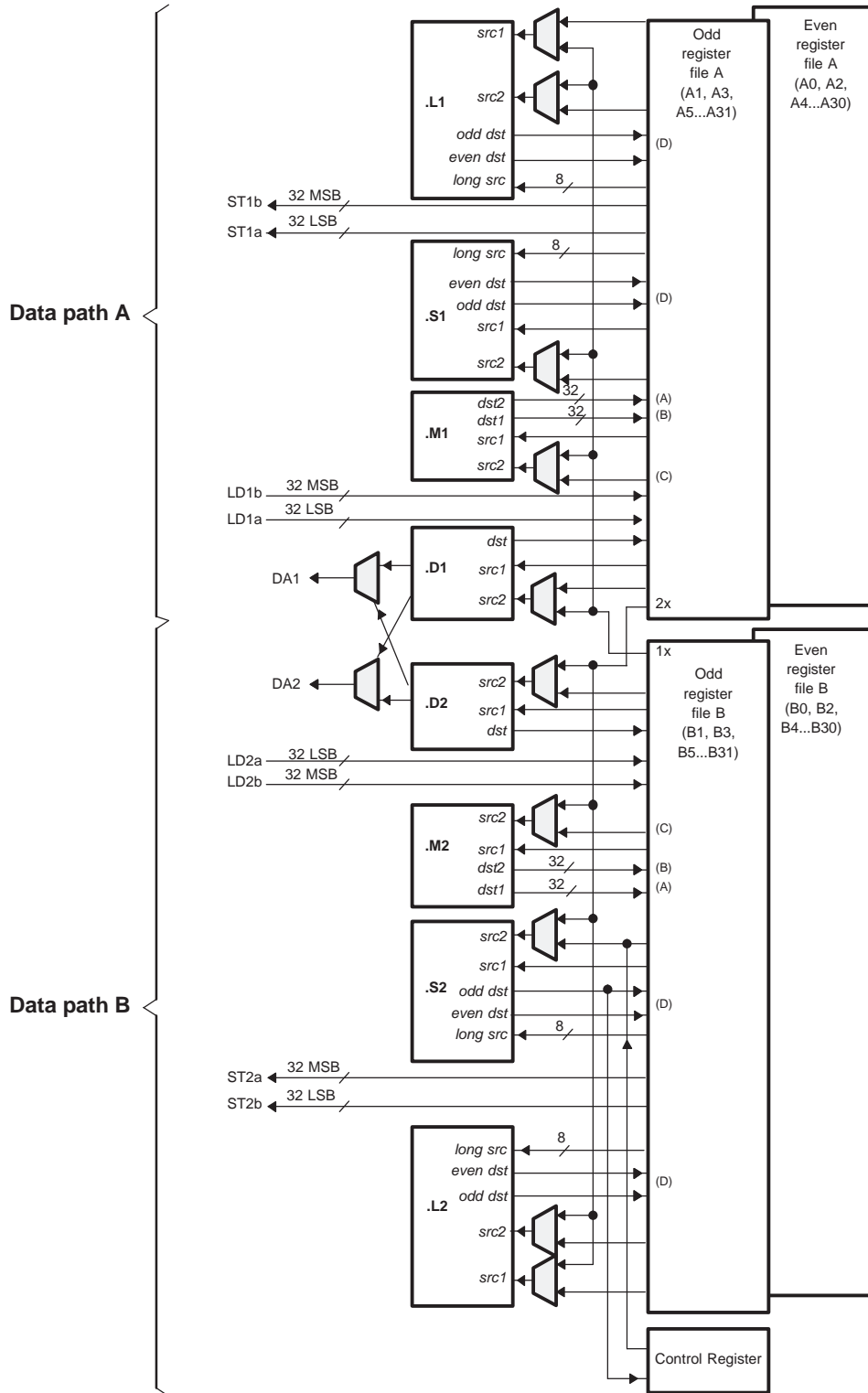
Other new features include:

- **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 op-codes) and from system events (such as a watchdog time expiration).
- **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 is implemented in the CPU which is *not* sensitive to system stalls.

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 and *even dst* connects to even register files.

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

2.2.2 C64x+ Memory Architecture

The C64x+ Megamodule implements a two-level internal cache-based memory architecture with external memory support. The Level 1 Program memory/cache (L1P) consists of 32 KB memory space that can be configured as mapped memory or direct mapped cache. The Level 1 Data memory/cache (L1D) consists of 80 KB—48 KB of which is mapped memory and 32 KB of which can be configured as mapped memory or 2-way set associated cache. The Level 2 memory/cache (L2) consists of a 128 KB 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 register for the device.

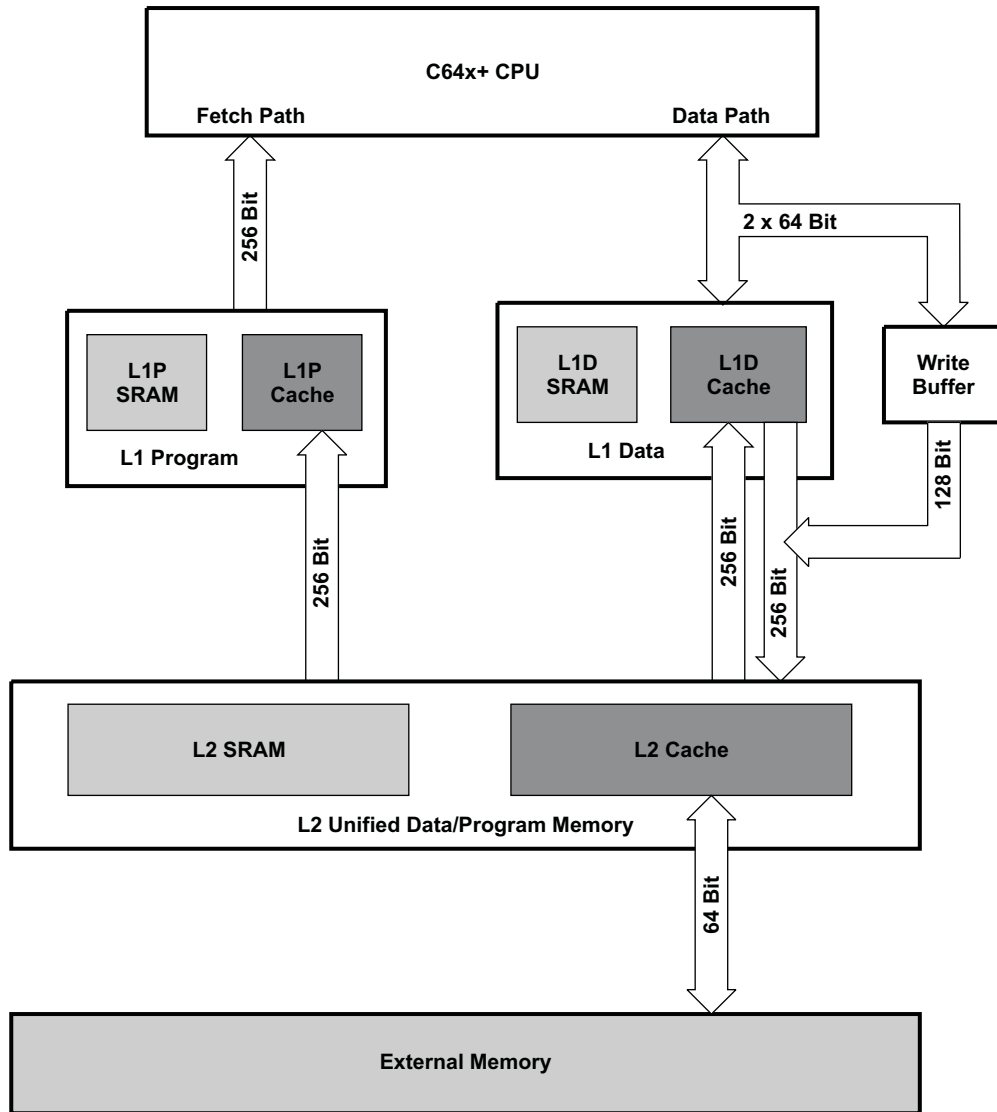
Figure 2-3, shows a diagram of the C64x+ Cache Memory Architecture.

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 (see Section 2.2.2.1, L1P Configuration Register (L1PCFG) Description) |
| 0x0184 0024 | L1PCC | L1P Freeze Mode Cache configuration register |
| 0x0184 0040 | L1DCFG | L1D Size Cache configuration register (see Section 2.2.2.2, L1D Configuration Register (L1DCFG) Description) |
| 0x0184 0044 | L1DCC | L1D Freeze Mode Cache configuration register |
| 0x0184 0048 - 0x0184 0FFC | - | Reserved |
| 0x0184 1000 | EDMAWEIGHT | L2 EDMA access control register |
| 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 |

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

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---------------------------|------------------|---|
| 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 |
| 0x0184 8000 - 0x0184 80BC | MAR0 - MAR47 | Reserved (corresponds to byte address 0x0000 0000 - 0x2FFF FFFF) |
| 0x0184 80C0 - 0x0184 80FC | MAR48 - MAR63 | Memory Attribute Registers for PCI Data (corresponds to byte address 0x3000 0000 - 0x3FFF FFFF) |
| 0x0184 8100 - 0x0184 8104 | MAR64 - MAR65 | Reserved (corresponds to byte address 0x4000 0000 - 0x41FF FFFF) |
| 0x0184 8108 - 0x0184 8124 | MAR66 - MAR73 | Memory Attribute Registers for EMIFA (corresponds to byte address 0x4200 0000 - 0x49FF FFFF) |
| 0x0184 8128 - 0x0184 812C | MAR74 - MAR75 | Reserved (corresponds to byte address 0x4A00 0000 - 0x4BFF FFFF) |
| 0x0184 8130 - 0x0184 813C | MAR76 - MAR79 | Memory Attribute Registers for VLYNQ (corresponds to byte address 0x4C00 0000 - 0x4FFF FFFF) |
| 0x0184 8140 - 0x0184 81FC | MAR80 - MAR127 | Reserved (corresponds to byte address 0x5000 0000 - 0x7FFF FFFF) |
| 0x0184 8200 - 0x0184 823C | MAR128 - MAR143 | Memory Attribute Registers for DDR2 (corresponds to byte address 0x8000 0000 - 0x8FFF FFFF) |
| 0x0184 8240 - 0x0184 83FC | MAR144 - MAR255 | Reserved (corresponds to byte address 0x9000 0000 - 0xFFFF FFFF) |



Legend:



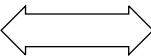
-  Addressable Memory
-  Cache Memory
-  Data Paths Managed By Cache Controller

Figure 2-3. C64x+ Cache Memory Architecture

The L1P is divided into two regions—denoted as L1P Region 0 and L1P Region 1. This is the L1P architecture on the C6424:

- L1P Region 0: 0K-Byte Memory
- L1P Region 1: 32K-Byte Memory
 - L1P Region 1 can be configured as mapped memory or cache and has a 0 wait state latency. This region is shown as "L1P RAM/Cache" in [Table 2-5, Memory Map Summary](#).

The C6424 **does not** support memory protection on L1P.

The L1D is divided into two regions—denoted as L1D Region 0 and L1D Region 1. This is the L1D architecture on the C6424:

- L1D Region 0: 48K-Byte Memory
 - This region is shown as "L1D RAM" in [Table 2-5, Memory Map Summary](#).
- L1D Region 1: 32K-Byte Memory
 - L1D Region 1 can be configured as mapped memory or cache. This region is shown as "L1D RAM/Cache" in [Table 2-5, Memory Map Summary](#).

The C6424 **does not** support memory protection on L1D.

L2 memory implements two separate memory ports. This is the L2 architecture on the C6424:

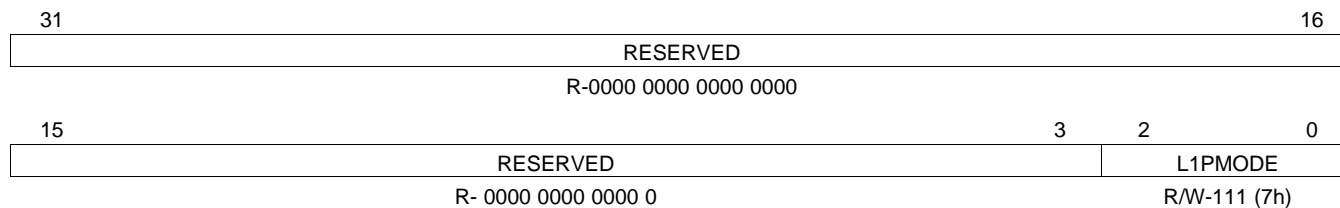
- Port 0
 - This port is shown as "L2 RAM/Cache" in [Table 2-5, Memory Map Summary](#).
 - Banking Scheme: 2 x 128-bit banks
 - Latency: 1 cycle (0 wait state)
- Port 1
 - This port is shown as "Boot ROM" in [Table 2-5, Memory Map Summary](#).
 - Banking Scheme: 1 x 256-bit bank
 - Latency: 1 cycle (0 wait state)

The C6424 **does not** support memory protection on L2.

For more detailed information about the C64x+ Cache Memory Architecture, see the *TMS320C64x+ DSP Cache User's Guide* (literature number [SPRU862](#)) and the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number [SPRU871](#)).

2.2.2.1 L1P Configuration Register (L1PCFG) Description

The L1P Configuration Register (L1PCFG) controls/defines the size of the L1P cache. On the C6424, the L1PCFG register is device-specific and varies from what is shown in the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number [SPRU871](#)). The format and bit field descriptions of the L1PCFG register for the C6424 are shown in [Figure 2-4](#) and [Table 2-3](#), respectively.



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

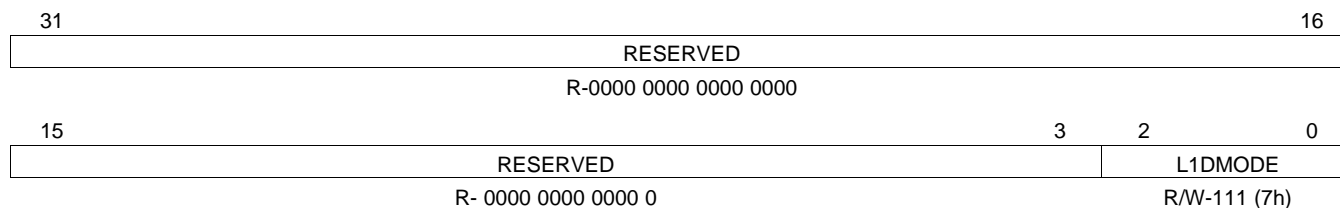
Figure 2-4. L1PCFG Register

Table 2-3. L1PCFG Register Description

| Bit | Field Name | Description |
|------|------------|--|
| 31:3 | RESERVED | Reserved. Read-only, writes have no effect. |
| 2:0 | L1PMODE | L1PMODE select. 000 [0h] = L1P Cache Disabled 001 [1h] = 4 KB 010 [2h] = 8 KB 011 [3h] = 16KB 100 [4h] = 32KB 101 [5h] = 32KB 110 [6h] = 32KB 111 [7h] = 32KB [default]. |

2.2.2.2 L1D Configuration Register (L1DCFG) Description

The L1D Configuration Register (L1DCFG) controls/defines the size of the L1D cache. On the C6424, the L1DCFG register is device-specific and varies from what is shown in the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number SPRU871). The format and bit field descriptions of the L1DCFG register for the C6424 are shown in [Figure 2-5](#) and [Table 2-4](#), respectively.



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

Figure 2-5. L1DCFG Register

Table 2-4. L1DCFG Register Description

| Bit | Field Name | Description |
|------|------------|--|
| 31:3 | RESERVED | Reserved. Read-only, writes have no effect. |
| 2:0 | L1DMODE | L1DMODE select. 000 [0h] = L1D Cache Disabled 001 [1h] = 4 KB 010 [2h] = 8 KB 011 [3h] = 16KB 100 [4h] = 32KB 101 [5h] = 32KB 110 [6h] = 32KB 111 [7h] = 32KB [default]. |

2.3 Memory Map Summary

Table 2-5 shows the memory map address ranges of the device. Table 2-6 depicts the expanded map of the Configuration Space (0x0180 0000 through 0x0FFF FFFF). The device has multiple on-chip memories associated with its two processors 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 2-5. Memory Map Summary

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ MEMORY MAP | EDMA PERIPHERAL MEMORY MAP | PCI MEMORY MAP |
|---------------|-------------|--------------|--|--|-------------------|
| 0x0000 0000 | 0x000F FFFF | 1M | Reserved | Reserved | Reserved |
| 0x0010 0000 | 0x001F FFFF | 64K | Boot ROM | | |
| 0x0011 0000 | 0x007F FFFF | 7M-64K | Reserved | | |
| 0x0080 0000 | 0x0081 FFFF | 128K | L2 RAM/Cache ⁽¹⁾ | | |
| 0x0082 0000 | 0x00E0 7FFF | 6048K | Reserved | | |
| 0x00E0 8000 | 0x00E0 FFFF | 32K | L1P RAM/Cache ⁽¹⁾ | | |
| 0x00E1 0000 | 0x00F0 3FFF | 976K | Reserved | | |
| 0x00F0 4000 | 0x00F0 FFFF | 48K | L1D RAM | | |
| 0x00F1 0000 | 0x00F1 7FFF | 32K | L1D RAM/Cache ⁽¹⁾ | | |
| 0x00F1 8000 | 0x017F FFFF | 9120K | Reserved | | |
| 0x0180 0000 | 0x01BF FFFF | 4M | CFG Space | | |
| 0x01C0 0000 | 0x01FF FFFF | 4M | CFG Bus Peripherals | | |
| 0x0200 0000 | 0x100F FFFF | 225M | Reserved | Reserved | Reserved |
| 0x1010 0000 | 0x1010 FFFF | 64K | Boot ROM | | |
| 0x1011 0000 | 0x107F FFFF | 7M-48K | Reserved | | |
| 0x1080 0000 | 0x1081 FFFF | 128K | L2 RAM/Cache ⁽¹⁾ | | |
| 0x1082 0000 | 0x10E0 7FFF | 6048K | Reserved | | |
| 0x10E0 8000 | 0x10E0 FFFF | 32K | L1P RAM/Cache ⁽¹⁾ | | |
| 0x10E1 0000 | 0x10F0 3FFF | 976K | Reserved | | |
| 0x10F0 4000 | 0x10F0 FFFF | 48K | L1D RAM | | |
| 0x10F1 0000 | 0x10F1 7FFF | 32K | L1D RAM/Cache ⁽¹⁾ | | |
| 0x10F1 8000 | 0x10FF FFFF | 1M-96K | Reserved | | |
| 0x1100 0000 | 0x1FFF FFFF | 240M | Reserved | | |
| 0x2000 0000 | 0x2000 7FFF | 32K | DDR2 Control Regs | DDR2 Control Regs | DDR2 Control Regs |
| 0x2000 8000 | 0x2FFF FFFF | 256M-32K | Reserved | Reserved | Reserved |
| 0x3000 0000 | 0x3FFF FFFF | 256M | PCI Data | PCI Data | |
| 0x4000 0000 | 0x41FF FFFF | 32M | Reserved | Reserved | |
| 0x4200 0000 | 0x43FF FFFF | 32M | EMIFA Data ($\overline{CS2}$) ⁽²⁾ | EMIFA Data ($\overline{CS2}$) ⁽²⁾ | |
| 0x4400 0000 | 0x45FF FFFF | 32M | EMIFA Data ($\overline{CS3}$) ⁽²⁾ | EMIFA Data ($\overline{CS3}$) ⁽²⁾ | |
| 0x4600 0000 | 0x47FF FFFF | 32M | EMIFA Data ($\overline{CS4}$) ⁽²⁾ | EMIFA Data ($\overline{CS4}$) ⁽²⁾ | |
| 0x4800 0000 | 0x49FF FFFF | 32M | EMIFA Data ($\overline{CS5}$) ⁽²⁾ | EMIFA Data ($\overline{CS5}$) ⁽²⁾ | |
| 0x4A00 0000 | 0x4BFF FFFF | 32M | Reserved | Reserved | |
| 0x4C00 0000 | 0x4FFF FFFF | 64M | VLYNQ (Remote Data) | VLYNQ (Remote Data) | |
| 0x5000 0000 | 0x7FFF FFFF | 768M | Reserved | Reserved | |
| 0x8000 0000 | 0x8FFF FFFF | 256M | DDR2 Memory Controller | DDR2 Memory Controller | |
| 0x9000 0000 | 0xFFFF FFFF | 1792M | Reserved | Reserved | Reserved |

- For all boot modes that default to DSPBOOTADDR = 0x0010 0000 (i.e., all boot modes except the EMIFA ROM Direct Boot, BOOTMODE[3:0] = 0100, FASTBOOT = 0), the bootloader code disables all C64x+ cache (L2, L1P, and L1D) so that upon exit from the bootloader code, all C64x+ memories are configured as all RAM (L2CFG.L2MODE = 0h, L1PCFG.L1PMODE = 0h, and L1DCFG.L1DMODE = 0h). If cache use is required, the application code must explicitly enable the cache. For more information on boot modes, see Section 3.4.1, *Boot Modes*. For more information on the bootloader, see the *Using the TMS320C642x Bootloader Application Report* (literature number SPRAAK5). For the EMIFA ROM Direct Boot (BOOTMODE[3:0] = 0100, FASTBOOT = 0), the bootloader is not executed—that is, L2 RAM/Cache defaults to all RAM (L2CFG.L2MODE = 0h); L1P RAM/Cache defaults to all cache (L1PCFG.L1PMODE = 7h); and L1D RAM/Cache defaults to all cache (L1DCFG.L1DMODE = 7h).
- The EMIFA CS0 and CS1 are **not** functionally supported on the C6424 device, and therefore, are **not** pinned out.

Table 2-6. Configuration Memory Map Summary

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ |
|---------------|-------------|--------------|-------------------------------------|
| 0x0180 0000 | 0x0180 FFFF | 64K | C64x+ Interrupt Controller |
| 0x0181 0000 | 0x0181 0FFF | 4K | C64x+ Powerdown Controller |
| 0x0181 1000 | 0x0181 1FFF | 4K | C64x+ Security ID |
| 0x0181 2000 | 0x0181 2FFF | 4K | C64x+ Revision ID |
| 0x0182 0000 | 0x0182 FFFF | 64K | C64x+ EMC |
| 0x0183 0000 | 0x0183 FFFF | 64K | Reserved |
| 0x0184 0000 | 0x0184 FFFF | 64K | C64x+ Memory System |
| 0x0185 0000 | 0x0187 FFFF | 192K | Reserved |
| 0x0188 0000 | 0x01BB FFFF | 3328K | Reserved |
| 0x01BC 0000 | 0x01BC 00FF | 256 | Reserved |
| 0x01BC 0100 | 0x01BC 01FF | 256 | Pin Manager and Trace |
| 0x01BC 0400 | 0x01BF FFFF | 255K | Reserved |
| 0x01C0 0000 | 0x01C0 FFFF | 64K | EDMA CC |
| 0x01C1 0000 | 0x01C1 03FF | 1K | EDMA TC0 |
| 0x01C1 0400 | 0x01C1 07FF | 1K | EDMA TC1 |
| 0x01C1 0800 | 0x01C1 0BFF | 1K | EDMA TC2 |
| 0x01C1 0C00 | 0x01C1 9FFF | 5K | Reserved |
| 0x01C1 A000 | 0x01C1 A7FF | 2K | PCI Control Register ⁽¹⁾ |
| 0x01C1 A800 | 0x01C1 FFFF | 22K | Reserved |
| 0x01C2 0000 | 0x01C2 03FF | 1K | UART0 |
| 0x01C2 0400 | 0x01C2 07FF | 1K | UART1 |
| 0x01C2 0800 | 0x01C2 0FFF | 2K | Reserved |
| 0x01C2 1000 | 0x01C2 13FF | 1K | I2C |
| 0x01C2 1400 | 0x01C2 17FF | 1K | Timer0 |
| 0x01C2 1800 | 0x01C2 1BFF | 1K | Timer1 |
| 0x01C2 1C00 | 0x01C2 1FFF | 1K | Timer2 (Watchdog) |
| 0x01C2 2000 | 0x01C2 23FF | 1K | PWM0 |
| 0x01C2 2400 | 0x01C2 27FF | 1K | PWM1 |
| 0x01C2 2800 | 0x01C2 2BFF | 1K | PWM2 |
| 0x01C2 2C00 | 0x01C3 FFFF | 117K | Reserved |
| 0x01C4 0000 | 0x01C4 07FF | 2K | System Module |
| 0x01C4 0800 | 0x01C4 0BFF | 1K | PLL Controller 1 |
| 0x01C4 0C00 | 0x01C4 0FFF | 1K | PLL Controller 2 |
| 0x01C4 1000 | 0x01C4 1FFF | 4K | Power and Sleep Controller |
| 0x01C4 2000 | 0x01C6 6FFF | 148K | Reserved |
| 0x01C6 7000 | 0x01C6 77FF | 2K | GPIO |
| 0x01C6 7800 | 0x01C6 7FFF | 2K | HPI |
| 0x01C6 8000 | 0x01C7 FFFF | 96K | Reserved |
| 0x01C8 0000 | 0x01C8 0FFF | 4K | EMAC Control Registers |
| 0x01C8 1000 | 0x01C8 1FFF | 4K | EMAC Control Module Registers |
| 0x01C8 2000 | 0x01C8 3FFF | 8K | EMAC Control Module RAM |
| 0x01C8 4000 | 0x01C8 47FF | 2K | MDIO Control Registers |
| 0x01C8 4800 | 0x01CF FFFF | 494K | Reserved |
| 0x01D0 0000 | 0x01D0 07FF | 2K | McBSP0 |
| 0x01D0 0800 | 0x01D0 0FFF | 2K | McBSP1 |

(1) Access to certain PCI registers when there is no active PCI clock may hang the device. For more information, see the *TMS320C6424 Peripheral Component Interconnect (PCI) Reference Guide* (literature number SPRUEN3).

Table 2-6. Configuration Memory Map Summary (continued)

| START ADDRESS | END ADDRESS | SIZE (Bytes) | C64x+ |
|---------------|-------------|--------------|-------------------------|
| 0x01D0 1000 | 0x01D0 13FF | 1K | McASP0 Control |
| 0x01D0 1400 | 0x01D0 17FF | 1K | McASP0 Data |
| 0x01D0 1800 | 0x01DF FFFF | 1018K | Reserved |
| 0x01E0 0000 | 0x01E0 0FFF | 4K | EMIFA Control |
| 0x01E0 1000 | 0x01E0 1FFF | 4K | VLYNQ Control Registers |
| 0x01E0 2000 | 0x0FFF FFFF | 226M-8K | Reserved |

2.4 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 3.7](#), *Multiplexed Pin Configurations* of this document.

2.4.1 Pin Map (Bottom View)

[Figure 2-6](#) through [Figure 2-9](#) show the bottom view of the **ZWT** package pin assignments in four quadrants (A, B, C, and D). [Figure 2-10](#) through [Figure 2-13](#) show the bottom view of the **ZDU** package pin assignments in four quadrants (A, B, C, and D).

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
|---|-----------------------------|-----------------------------|------------------------------|-----------------------------|--------------------|--------------------|------------------------------|------------------------------|--------------------|--------------------|---|
| W | V _{SS} | V _{SS} | DDR_D[7] | DDR_D[9] | DDR_D[12] | DDR_D[14] | DDR_CLK | $\overline{\text{DDR_CLK}}$ | DDR_A[12] | DDR_A[11] | W |
| V | DV _{DDR2} | DDR_D[4] | DDR_D[6] | DDR_D[8] | DDR_D[11] | DDR_D[13] | DDR_D[15] | DDR_CKE | DDR_BA[1] | DDR_A[8] | V |
| U | DDR_D[2] | DDR_D[3] | DDR_D[5] | DDR_DQS[0] | DDR_D[10] | DDR_DQS[1] | $\overline{\text{DDR_RAS}}$ | DDR_BA[0] | DDR_BA[2] | DDR_A[10] | U |
| T | DDR_D[0] | DDR_D[1] | PCIEN | DDR_DQM[0] | DV _{DDR2} | DDR_DQM[1] | DDR_CAS | DDR_WE | DDR_CS | DDR_ZN | T |
| R | V _{SS} | $\overline{\text{TRST}}$ | TMS | DV _{DDR2} | V _{SS} | DV _{DDR2} | V _{SS} | DV _{DDR2} | V _{SS} | DV _{DDR2} | R |
| P | DV _{DD33} | EMU0 | TDO | TDI | DV _{DDR2} | V _{SS} | DV _{DDR2} | V _{SS} | DV _{DDR2} | V _{SS} | P |
| N | TCK | EMU1 | $\overline{\text{RESETOUT}}$ | $\overline{\text{POR}}$ | V _{SS} | DV _{DD33} | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | N |
| M | CLKOUT0/ PWM2/ GP[84] | SCL | SDA | $\overline{\text{RESET}}$ | DV _{DD33} | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | V _{SS} | M |
| L | UCTS0/ GP[87] | URXD0/ GP[85] | URTS0/ PWM0/ GP[88] | TINP1L/ URXD1/ GP[56] | RSV3 | DV _{DD33} | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | L |
| K | V _{SS} | CLKS1/ TINP0L/ GP[98] | UTXD0/ GP[86] | TOUT1L/ UTXD1/ GP[55] | RSV2 | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | V _{SS} | K |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |

Figure 2-6. ZWT Pin Map [Quadrant A]

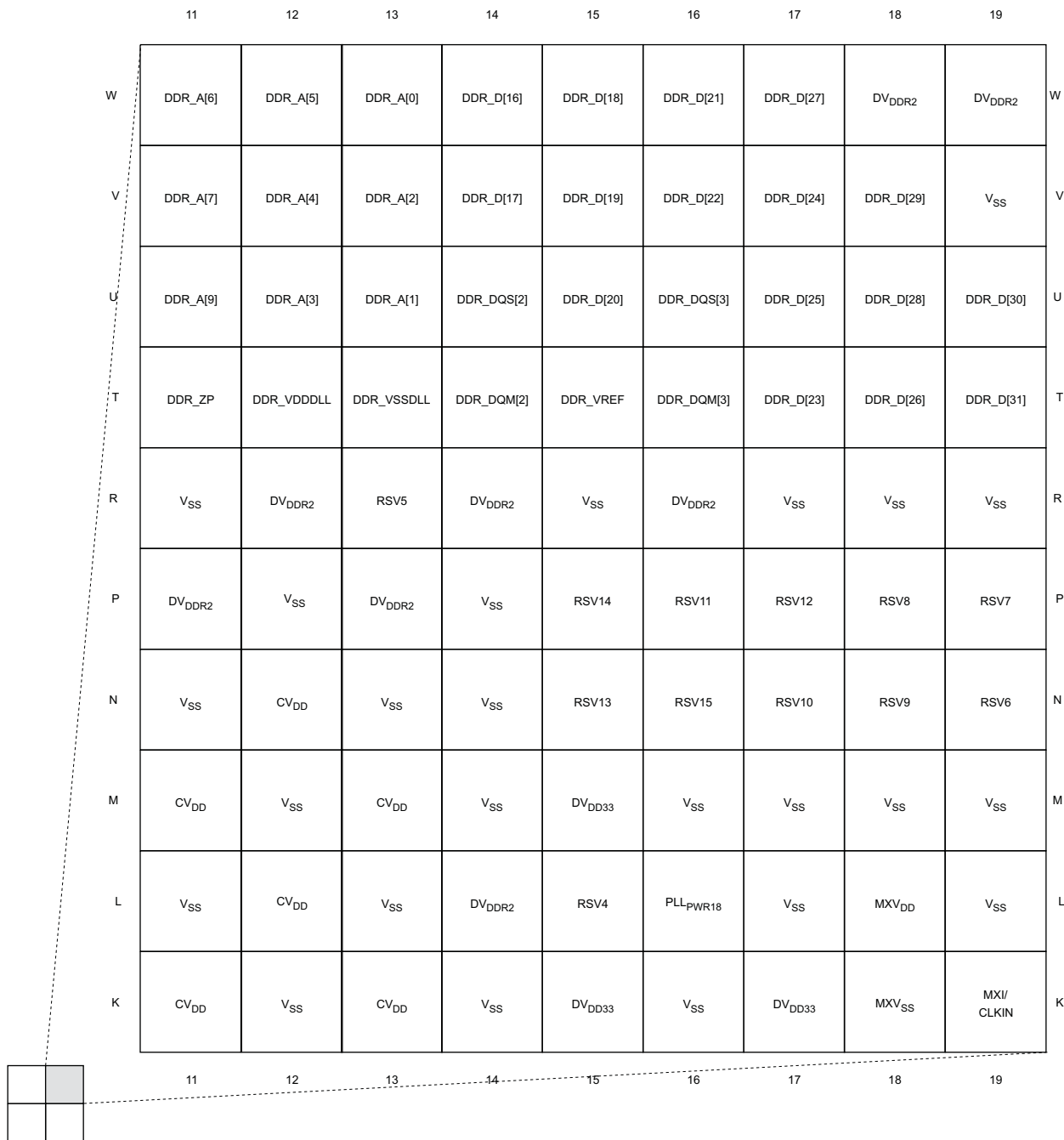


Figure 2-7. ZWT Pin Map [Quadrant B]

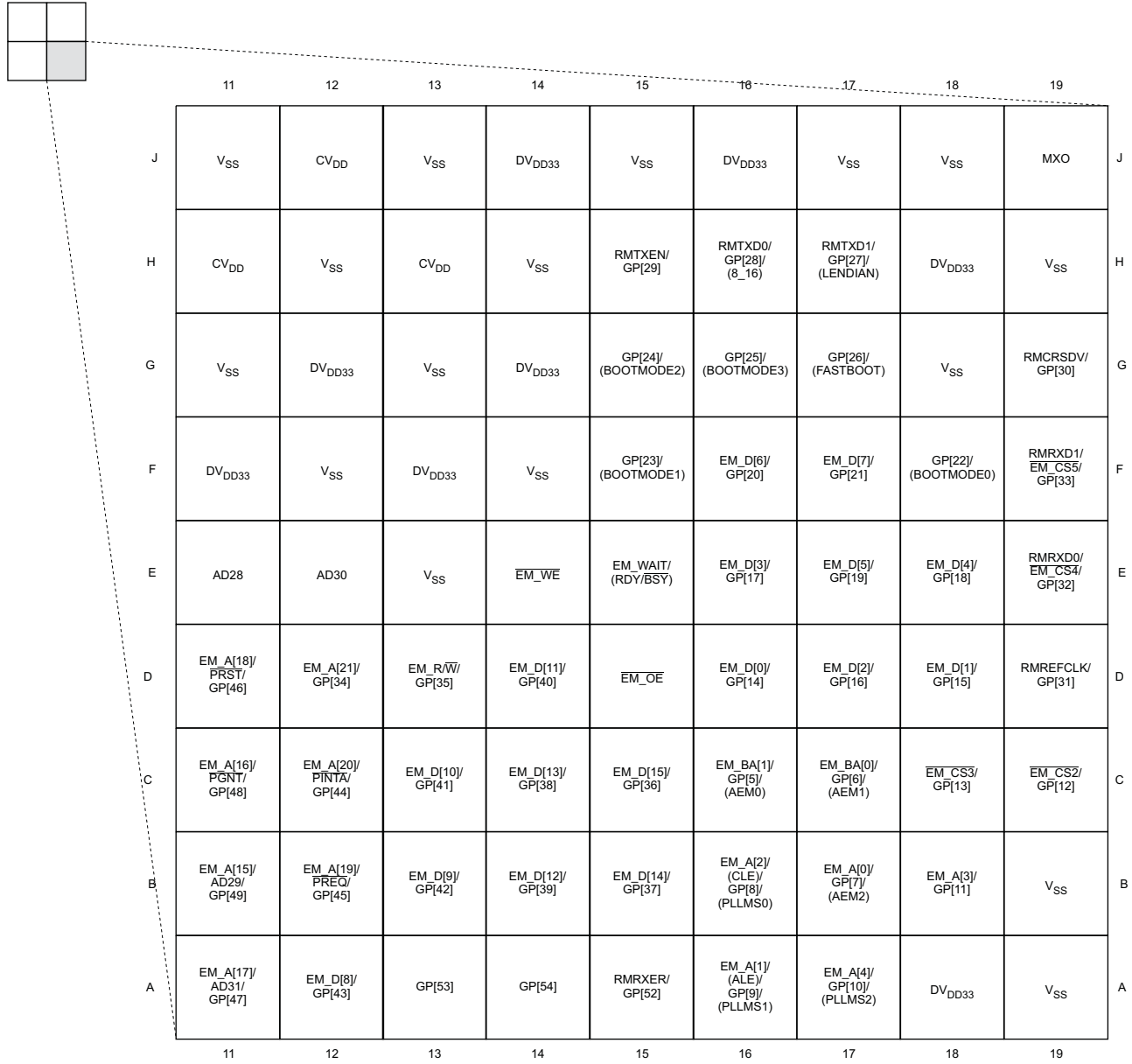


Figure 2-8. ZWT Pin Map [Quadrant C]



| | | | | | | | | | | | |
|---|----------------------------------|-------------------------------------|--------------------------------------|-------------------------------------|---|---|--|---|-------------------------------|-------------------------------|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| J | DV _{DD33} | AHCLKR0/ CLKR0/ GP[101] | AXR0[1]/ DX0/ GP[104] | CLKS0/ TOUT0L/ GP[97] | V _{SS} | DV _{DD33} | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | J |
| H | ACLKR0/ CLKX0/ GP[99] | AXR0[0]/ FSR1/ GP[105] | AXR0[2]/ FSX0/ GP[103] | AFSR0/ DR0/ GP[100] | DV _{DD33} | V _{SS} | CV _{DD} | V _{SS} | CV _{DD} | V _{SS} | H |
| G | AHCLKX0/ CLKR1/ GP[108] | AFSX0/ DX1/ GP[107] | AMUTE0/ DR1/ GP[110] | AXR0[3]/ FSR0/ GP[102] | V _{SS} | DV _{DD33} | V _{SS} | DV _{DD33} | V _{SS} | DV _{DD33} | G |
| F | ACLKX0/ CLKX1/ GP[106] | AMUTEIN0/ FSX1/ GP[109] | GP[4]/ PWM1 | V _{SS} | DV _{DD33} | V _{SS} | DV _{DD33} | V _{SS} | DV _{DD33} | V _{SS} | F |
| E | AD0/ GP[0] | AD1/ GP[1] | AD2/ GP[2] | AD4/ GP[3] | RSV1 | DV _{DD33} | V _{SS} | DV _{DD33} | V _{SS} | AD26 | E |
| D | FAS/ MDIO/ AD3/ GP[83] | FRDY/ MRXD2/ PCBE6/ GP[80] | HCNTL1/ MTXEN/ AD11/ GP[75] | HD14/ MTXD0/ AD15/ GP[72] | HD12/ MTXD2/ PPAR/ GP[70] | HD6/ VLYNQ_TXD1/ PTRDY/ GP[64] | HD1/ VLYNQ_RXD0/ AD16/ GP[59] | EM_A[6]/ AD20/ GP[95] | EM_A[9]/ PIDSEL/ GP[92] | EM_A[12]/ PCBE3/ GP[89] | D |
| C | FCS/ MDCLK/ AD5/ GP[81] | HINT/ MRXD3/ AD6/ GP[82] | HDS2/ MRXD0/ AD9/ GP[78] | HHWIL/ MRXDV/ AD13/ GP[74] | HD11/ MTXD3/ PCBE1/ GP[69] | HD9/ MCOL/ PSTOP/ GP[67] | HD4/ VLYNQ_RXD3/ PFRAME/ GP[62] | HD0/ VLYNQ_SCRUN/ AD18/ GP[58] | EM_A[7]/ AD22/ GP[94] | EM_A[11]/ AD24/ GP[90] | C |
| B | V _{SS} | HDS1/ MRXD1/ AD7/ GP[79] | HCNTL0/ MRXER/ AD10/ GP[76] | HD13/ MTXD1/ AD14/ GP[71] | HD10/ MCRS/ PSERR/ GP[68] | HD7/ VLYNQ_TXD2/ PDEVSEL/ GP[65] | HD3/ VLYNQ_RXD2/ PCBE2/ GP[61] | EM_A[5]/ AD19/ GP[96] | EM_A[8]/ AD21/ GP[93] | EM_A[13]/ AD25/ GP[51] | B |
| A | DV _{DD33} | DV _{DD33} | HR/W/ MRXCLK/ AD8/ GP[77] | HD15/ MTXCLK/ AD12/ GP[73] | HD8/ VLYNQ_TXD3/ PPERR/ GP[66] | HD5/ VLYNQ_TXD0/ PIRDY/ GP[63] | VLYNQ_CLOCK/ PCICLK/ GP[57] | HD2/ VLYNQ_RXD1/ AD17/ GP[60] | EM_A[10]/ AD23/ GP[91] | EM_A[14]/ AD27/ GP[50] | A |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |

Figure 2-9. ZWT Pin Map [Quadrant D]



Figure 2-10. ZDU Pin Map [Quadrant A]

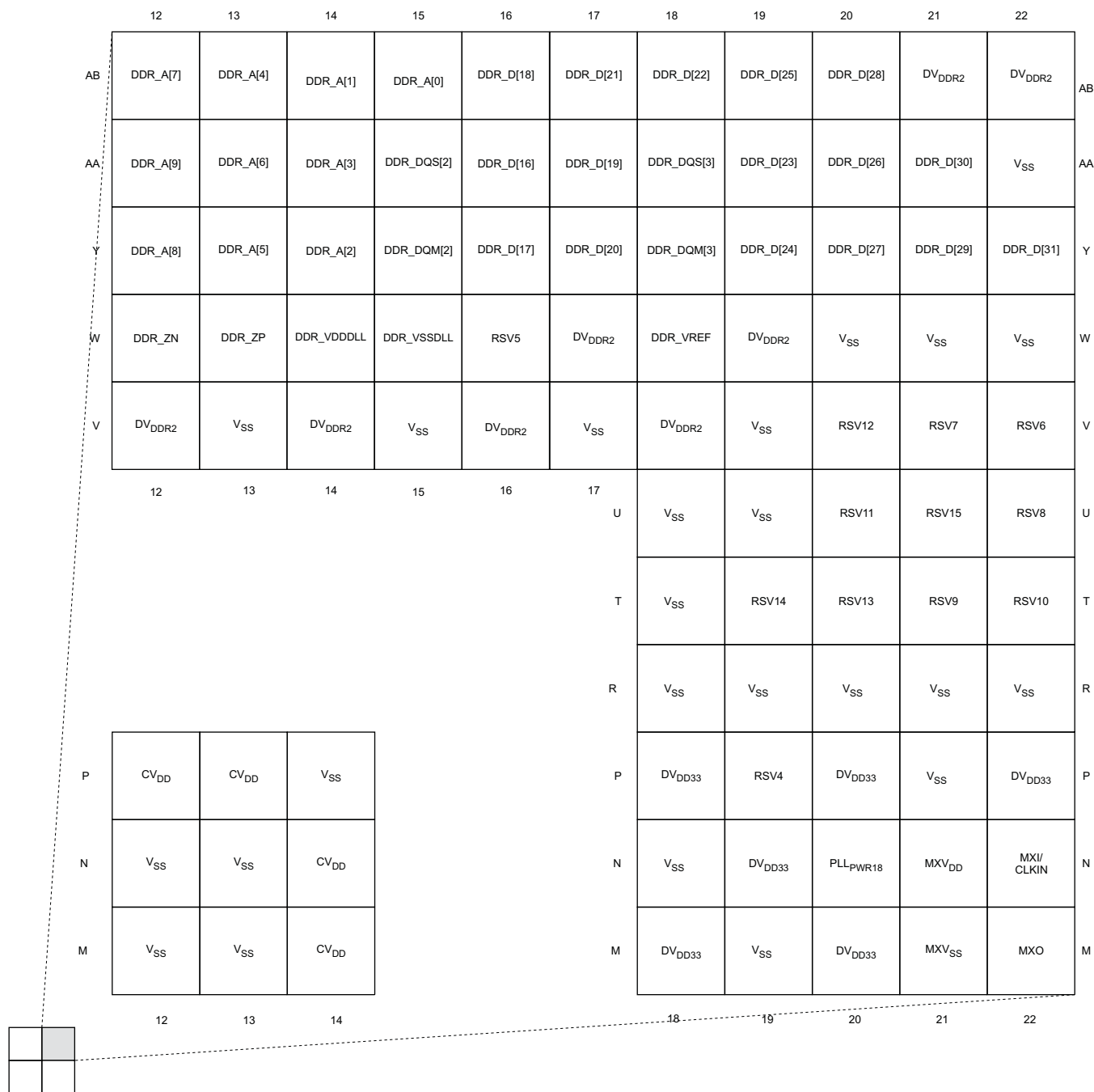


Figure 2-11. ZDU Pin Map [Quadrant B]

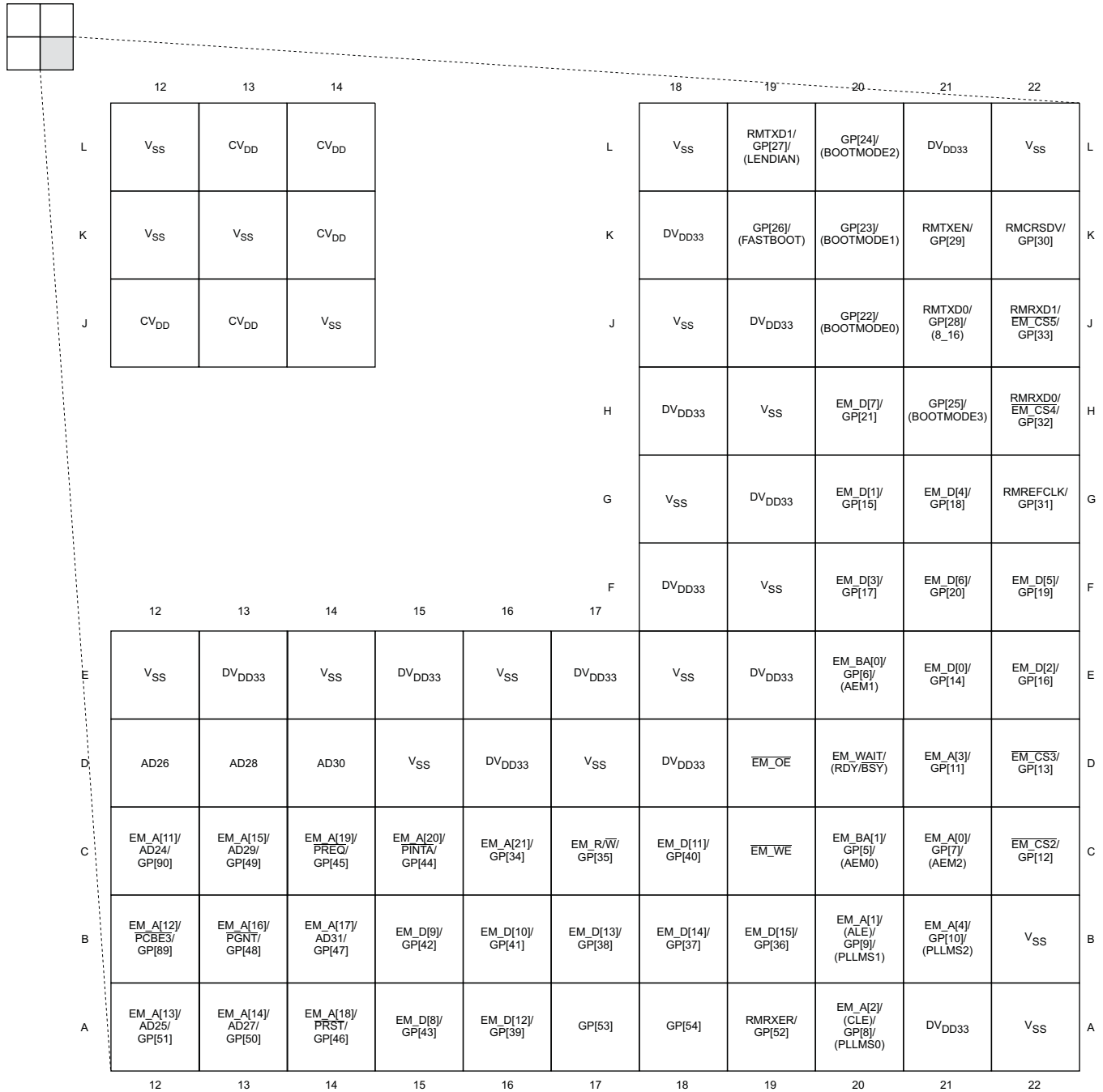


Figure 2-12. ZDU Pin Map [Quadrant C]



Figure 2-13. ZDU Pin Map [Quadrant D]

2.5 Terminal Functions

The terminal functions tables (Table 2-7 through Table 2-29) 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 the Device Configurations section of this data manual.

All device boot and configuration pins (except PCIEN) are multiplexed configuration pins— meaning they are multiplexed with functional pins. These pins function as device boot and configuration pins only during device reset. The input states of these pins are sampled and latched into the BOOTCFG register when device reset is deasserted (see **Note** below). After device reset is deasserted, the values on these multiplexed pins no longer have to hold the configuration.

The PCIEN pin is a standalone configuration pin. Its value is latched into the BOOTCFG register when device reset is deasserted (see **Note** below). Unlike the multiplexed device boot and configuration pins, the value on the PCIEN pin even after device reset is deasserted **must** hold the configuration.

For proper device operation, external pullup/pulldown resistors may be required on these device boot and configuration pins. Section 3.9.1, *Pullup/Pulldown Resistors* discusses situations where external pullup/pulldown resistors are required.

Note: Internal to the chip, the two device reset pins $\overline{\text{RESET}}$ and $\overline{\text{POR}}$ are logically AND'd together for the purpose of latching device boot and configuration pins. The values on all device boot and configuration pins are latched into the BOOTCFG register when the logical AND of $\overline{\text{RESET}}$ and $\overline{\text{POR}}$ transitions from low-to-high.

Table 2-7. BOOT Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ₍₃₎ | DESCRIPTION |
|----------------------------------|---------|---------|---------------------|--|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| BOOT | | | | | |
| GP[25]/ (BOOTMODE3) | G16 | H21 | I/O/Z | IPD DV _{DD33} | Bootmode configuration bits. These bootmode functions along with the FASTBOOT function determine what device bootmode configuration is selected. The C6424 device supports several types of bootmodes along with a FASTBOOT option; for more details on the types/options, see Section 3.4.1, <i>Boot Modes</i> . |
| GP[24]/ (BOOTMODE2) | G15 | L20 | | | |
| GP[23]/ (BOOTMODE1) | F15 | K20 | | | |
| GP[22]/ (BOOTMODE0) | F18 | J20 | | | |
| GP[26]/ (FASTBOOT) | G17 | K19 | I/O/Z | IPD DV _{DD33} | Fast Boot 0 = Not Fast Boot 1 = Fast Boot |
| EM_A[4]/GP[10]/(PL LMS2) | A17 | B21 | I/O/Z | IPD DV _{DD33} | Fast Boot PLL Multiplier Select (PLLMS) |
| EM_A[1]/(ALE)/ GP[9]/(PLLMS1) | A16 | B20 | I/O/Z | IPD DV _{DD33} | These pins select the PLL multiplier for Fast Boot. For more details, see Section 3.5.1.3, <i>Fast Boot PLL Multiplier Select (PLLMS)</i> . |
| EM_A[2]/(CLE)/ GP[8]/(PLLMS0) | B16 | A20 | I/O/Z | IPD DV _{DD33} | |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see Section 3.9.1, *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-7. BOOT Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ₍₃₎ | DESCRIPTION |
|-----------------------------|---------|---------|---------------------|--|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_A[0]/ GP[7]/(AEM2) | B17 | C21 | I/O/Z | IPD DV _{DD33} | Selects EMIFA Pinout Mode |
| EM_BA[0]/ GP[6]/(AEM1) | C17 | E20 | I/O/Z | IPD DV _{DD33} | The C6424 supports the following EMIFA Pinout Modes: AEM[2:0] = 000, No EMIFA AEM[2:0] = 010, EMIFA (Async) Pinout Mode 2 AEM[2:0] = 101, EMIFA (NAND) Pinout Mode 5 |
| EM_BA[1]/ GP[5]/(AEM0) | C16 | C20 | I/O/Z | IPD DV _{DD33} | |
| RMTXD0/GP[28]/ (8_16) | H16 | J21 | I/O/Z | IPD DV _{DD33} | Selects the EMIFA $\overline{\text{EM_CS2}}$ default bus width 0 = 8-bit bus width 1 = 16-bit bus width This signal affects the EMIFA module's $\overline{\text{EM_CS2}}$ space default setting. It does not affect the pin out. To determine the actual EMIFA pinout, the (AEM[2:0]) boot configuration settings must be set appropriately. |
| PCIEN | T3 | W3 | I | IPD DV _{DD33} | PCI Enable 0 = PCI pin function is disabled 1 = PCI pin function is enabled |
| RMTXD1/GP[27]/ (LENDIAN) | H17 | L19 | I/O/Z | IPU DV _{DD33} | Endian selection 0 = Big Endian 1 = Little Endian |

Table 2-8. Oscillator/PLL Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ | DESCRIPTION |
|------------------------|---------|---------|---------------------|----------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| OSCILLATOR, PLL | | | | | |
| MXI/ CLKIN | K19 | N22 | I | MXV _{DD} | Crystal input MXI for MX oscillator (system oscillator, 15-30 MHz). If the internal oscillator is bypassed, this is the external oscillator clock input. ⁽³⁾ |
| MXO | J19 | M22 | O | MXV _{DD} | Crystal output for MX oscillator |
| MXV _{DD} | L18 | N21 | S | (4) | 1.8 V power supply for MX oscillator. On the board, this pin can be connected to the same 1.8 V power supply as DV _{DDR2} . |
| MXV _{SS} | K18 | M21 | GND | (4) | Ground for MX oscillator |
| PLL _{PWR18} | L16 | N20 | S | (4) | 1.8 V power supply for PLLs |

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

(2) Specifies the operating I/O supply voltage for each signal

(3) For more information on external board connections, see [Section 6.6](#), *External Clock Input From MXI/CLKIN Pin*.

(4) For more information, see the *Recommended Operating Conditions* table

Table 2-9. Clock Generator Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| CLOCK GENERATOR | | | | | |
| CLKOUT0/ PWM2/GP[84] | M1 | R1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the System Clock generator (PLL1), PWM2, and GPIO. For the System Clock generator (PLL1), it is clock output CLKOUT0. This is configurable for toggling at the device input clock frequency (MXI/CLKIN frequency) or other divided-down (/1 to /32) clock outputs. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-10. RESET and JTAG Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ (³) | DESCRIPTION |
|------------------------------|---------|---------|---------------------|--|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| RESET | | | | | |
| $\overline{\text{RESET}}$ | M4 | R3 | I | IPU DV _{DD33} | Device reset |
| $\overline{\text{RESETOUT}}$ | N3 | T3 | O/Z | – DV _{DD33} | Reset output status pin. The $\overline{\text{RESETOUT}}$ pin indicates when the device is in reset. |
| $\overline{\text{POR}}$ | N4 | R2 | I | IPU DV _{DD33} | Power-on reset. |
| JTAG | | | | | |
| TMS | R3 | V3 | I | IPU DV _{DD33} | JTAG test-port mode select input. For proper device operation, do not oppose the IPU on this pin. |
| TDO | P3 | U2 | O/Z | – DV _{DD33} | JTAG test-port data output |
| TDI | P4 | U3 | I | IPU DV _{DD33} | JTAG test-port data input |
| TCK | N1 | U1 | I | IPU DV _{DD33} | JTAG test-port clock input |
| $\overline{\text{TRST}}$ | R2 | V2 | I | IPD DV _{DD33} | JTAG test-port reset. For IEEE 1149.1 JTAG compatibility, see the IEEE 1149.1 JTAG compatibility statement portion of this data sheet |
| EMU1 | N2 | T2 | I/O/Z | IPU DV _{DD33} | Emulation pin 1 |
| EMU0 | P2 | T1 | I/O/Z | IPU DV _{DD33} | Emulation pin 0 |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-11. EMIFA Terminal Functions (Boot Configuration)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ (3) | DESCRIPTION |
|----------------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| EMIFA: BOOT CONFIGURATION | | | | | |
| EM_BA[1]/ GP[5]/(AEM0) | C16 | C20 | I/O/Z | IPD DV _{DD33} | These pins are multiplexed between the EMIFA, and GPIO. When RESET or POR is asserted, these pins function as EMIFA configuration pins. At reset, the input states of AEM[2:0] are sampled to set the EMIFA Pinout Mode. For more details, see Section 3.5.1, Configurations at Reset . After reset, these pins function as EMIFA or GPIO pin functions based on pin mux selection. For more details on the AEM functions, see Section 3.5.1.2, EMIFA Pinout Mode (AEM[2:0]) . |
| EM_BA[0]/ GP[6]/(AEM1) | C17 | E20 | I/O/Z | IPD DV _{DD33} | |
| EM_A[0]/ GP[7]/(AEM2) | B17 | C21 | I/O/Z | IPD DV _{DD33} | |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

(3) Specifies the operating I/O supply voltage for each signal.

Table 2-12. EMIFA Terminal Functions (EMIFA Pinout Mode 2, AEM[2:0] = 010)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| EMIFA FUNCTIONAL PINS: 16-Bit ASYNC/NOR (EMIFA Pinout Mode 2, AEM[2:0] = 010) | | | | | |
| Actual pin functions are determined by the PINMUX0 and PINMUX1 register bit settings (e.g., PCIEN, AEM[2:0], etc.). For more details, see Section 3.7, Multiplexed Pin Configurations . | | | | | |
| $\overline{\text{EM_CS2}}$ / GP[12] | C19 | C22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, and GPIO. For EMIFA, this pin is Chip Select 2 output $\overline{\text{EM_CS2}}$ for use with asynchronous memories (i.e., NOR flash). This is the chip select for the default boot and ROM boot modes. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| $\overline{\text{EM_CS3}}$ / GP[13] | C18 | D22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, and GPIO. For EMIFA, this pin is Chip Select 3 output $\overline{\text{EM_CS3}}$ for use with asynchronous memories (i.e., NOR flash). Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| RMRXD0/ $\overline{\text{EM_CS4}}$ / GP[32] | E19 | H22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, and GPIO. For EMIFA, it is Chip Select 4 output $\overline{\text{EM_CS4}}$ for use with asynchronous memories (i.e., NOR flash). Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| RMRXD1/ $\overline{\text{EM_CS5}}$ / GP[33] | F19 | J22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, and GPIO. For EMIFA, it is Chip Select 5 output $\overline{\text{EM_CS5}}$ for use with asynchronous memories (i.e., NOR flash). Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| $\overline{\text{EM_R/W}}$ / GP[35] | D13 | C17 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA, it is read/write output $\overline{\text{EM_R/W}}$. |
| $\overline{\text{EM_WAIT}}$ / (RDY/BSY) | E15 | D20 | I/O/Z | IPU DV _{DD33} | For EMIFA (ASYNC/NOR), this pin is wait state extension input $\overline{\text{EM_WAIT}}$. |
| $\overline{\text{EM_OE}}$ | D15 | D19 | I/O/Z | IPU DV _{DD33} | For EMIFA, it is output enable output $\overline{\text{EM_OE}}$. |
| $\overline{\text{EM_WE}}$ | E14 | C19 | I/O/Z | IPU DV _{DD33} | For EMIFA, it is write enable output $\overline{\text{EM_WE}}$. |
| $\overline{\text{EM_BA[0]}}$ / GP[6]/(AEM1) | C17 | E20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA, this is the Bank Address 0 output ($\overline{\text{EM_BA[0]}}$). When connected to an 8-bit asynchronous memory, this pin is the lowest order bit of the byte address. When connected to a 16-bit asynchronous memory, this pin has the same function as EMIFA address pin 22 ($\overline{\text{EM_A[22]}}$). |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

(3) Specifies the operating I/O supply voltage for each signal.

Table 2-12. EMIFA Terminal Functions (EMIFA Pinout Mode 2, AEM[2:0] = 010) (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_BA[1]/ GP[5]/(AEM0) | C16 | C20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA, this is the Bank Address 1 output EM_BA[1]. When connected to an 8-bit asynchronous memory, this pin is the 2nd bit of the address. When connected to a 16-bit asynchronous memory, this pin is the lowest order bit of the 16-bit word address. |
| EM_A[21]/GP[34] | D12 | C16 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 21 output EM_A[21]. |
| EM_A[20]/PINTA/ GP[44] | C12 | C15 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 20 output EM_A[20]. |
| EM_A[19]/PREQ/ GP[45] | B12 | C14 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 19 output EM_A[19]. |
| EM_A[18]/PRST/ GP[46] | D11 | A14 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 18 output EM_A[18]. |
| EM_A[17]/AD31/ GP[47] | A11 | B14 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 17 output EM_A[17]. |
| EM_A[16]/PGNT/ GP[48] | C11 | B13 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 16 output EM_A[16]. |
| EM_A[15]/AD29/ GP[49] | B11 | C13 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 15 output EM_A[15]. |
| EM_A[14]/AD27/ GP[50] | A10 | A13 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 14 output EM_A[14]. |
| EM_A[13]/AD25/ GP[51] | B10 | A12 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 13 output EM_A[13]. |
| EM_A[12]/PCBE3/ GP[89] | D10 | B12 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 12 output EM_A[12]. |
| EM_A[11]/AD24/ GP[90] | C10 | C12 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 11 output EM_A[11]. |
| EM_A[10]/AD23/ GP[91] | A9 | B11 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 10 output EM_A[10]. |
| EM_A[9]/PIDSEL/ GP[92] | D9 | C11 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 9 output EM_A[9]. |
| EM_A[8]/AD21/ GP[93] | B9 | A11 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 8 output EM_A[8]. |
| EM_A[7]/AD22/ GP[94] | C9 | C10 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 7 output EM_A[7]. |

Table 2-12. EMIFA Terminal Functions (EMIFA Pinout Mode 2, AEM[2:0] = 010) (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|----------------------------------|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_A[6]/AD20/ GP[95] | D8 | B10 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 6 output EM_A[6]. |
| EM_A[5]/AD19/ GP[96] | B8 | A10 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 5 output EM_A[5]. |
| EM_A[4]/ GP[10]/(PLLMS2) | A17 | B21 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 4 output EM_A[4]. |
| EM_A[3]/ GP[11] | B18 | D21 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 3 output EM_A[3]. |
| EM_A[2]/(CLE)/ GP[8]/(PLLMS0) | B16 | A20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address bit 2 output EM_A[2]. |
| EM_A[1]/(ALE)/ GP[9]/(PLLMS1) | A16 | B20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address output EM_A[1]. |
| EM_A[0]/ GP[7]/(AEM2) | B17 | C21 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), this pin is address output EM_A[0], which is the least significant bit on a 32-bit word address. For an 8-bit asynchronous memory, this pin is the 3rd bit of the address. When connected to a 16-bit asynchronous memory, this pin is the 2nd bit of the address. |
| EM_D0/ GP[14] | D16 | E21 | I/O/Z | IPD DV _{DD33} | These pins are multiplexed between EMIFA and GPIO. For EMIFA (AEM[2:0] = 010), these pins are the 16-bit bi-directional data bus (EM_D[15:0]). |
| EM_D1/ GP[15] | D18 | G20 | I/O/Z | IPD DV _{DD33} | |
| EM_D2/ GP[16] | D17 | E22 | I/O/Z | IPD DV _{DD33} | |
| EM_D3/ GP[17] | E16 | F20 | I/O/Z | IPD DV _{DD33} | |
| EM_D4/ GP[18] | E18 | G21 | I/O/Z | IPD DV _{DD33} | |
| EM_D5/ GP[19] | E17 | F22 | I/O/Z | IPD DV _{DD33} | |
| EM_D6/ GP[20] | F16 | F21 | I/O/Z | IPD DV _{DD33} | |
| EM_D7/ GP[21] | F17 | H20 | I/O/Z | IPD DV _{DD33} | |
| EM_D8/ GP[43] | A12 | A15 | I/O/Z | IPD DV _{DD33} | |
| EM_D9/ GP[42] | B13 | B15 | I/O/Z | IPD DV _{DD33} | |
| EM_D10/ GP[41] | C13 | B16 | I/O/Z | IPD DV _{DD33} | |
| EM_D11/ GP[40] | D14 | C18 | I/O/Z | IPD DV _{DD33} | |
| EM_D12/ GP[39] | B14 | A16 | I/O/Z | IPD DV _{DD33} | |
| EM_D13/ GP[38] | C14 | B17 | I/O/Z | IPD DV _{DD33} | |
| EM_D14/ GP[37] | B15 | B18 | I/O/Z | IPD DV _{DD33} | |
| EM_D15/ GP[36] | C15 | B19 | I/O/Z | IPD DV _{DD33} | |

Table 2-12. EMIFA Terminal Functions (EMIFA Pinout Mode 2, AEM[2:0] = 010) (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EMIFA FUNCTIONAL PINS: 16-Bit NAND (EMIFA Pinout Mode 2, AEM[2:0] = 010) | | | | | |
| EM_A[1]/(ALE)/ GP[9]/(PLLMS1) | A16 | B20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. When used for EMIFA (NAND), this pin is the Address Latch Enable output (ALE). |
| EM_A[2]/(CLE)/ GP[8]/(PLLMS0) | B16 | A20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. When used for EMIFA (NAND), this pin is the Command Latch Enable output (CLE). |
| EM_WAIT/ (RDY/BSY) | E15 | D20 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), it is ready/busy input (RDY/ $\overline{\text{BSY}}$). |
| $\overline{\text{EM_OE}}$ | D15 | D19 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), this pin is read enable output ($\overline{\text{RE}}$). |
| $\overline{\text{EM_WE}}$ | E14 | C19 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), this pin is write enable output ($\overline{\text{WE}}$). |
| $\overline{\text{EM_CS2}}$ / GP[12] | C19 | C22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), this pin is Chip Select 2 output $\overline{\text{EM_CS2}}$ for use with NAND flash. This is the chip select for the default boot and ROM boot modes. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| $\overline{\text{EM_CS3}}$ / GP[13] | C18 | D22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), this pin is Chip Select 3 output $\overline{\text{EM_CS3}}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| RMRXD0/ $\overline{\text{EM_CS4}}$ / GP[32] | E19 | H22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND), EMIFA (NAND), and GPIO. For EMIFA (NAND), it is Chip Select 4 output $\overline{\text{EM_CS4}}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |
| RMRXD1/ $\overline{\text{EM_CS5}}$ / GP[33] | F19 | J22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND), EMIFA (NAND), and GPIO. For EMIFA (NAND), it is Chip Select 5 output $\overline{\text{EM_CS5}}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state. |

Table 2-12. EMIFA Terminal Functions (EMIFA Pinout Mode 2, AEM[2:0] = 010) (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_D0/ GP[14] | D16 | E21 | I/O/Z | IPD DV _{DD33} | <p>These pins are multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND) AEM[2:0] = 010, these are the 16-bit bi-directional data bus (EM_D[15:0]).</p> |
| EM_D1/ GP[15] | D18 | G20 | I/O/Z | IPD DV _{DD33} | |
| EM_D2/ GP[16] | D17 | E22 | I/O/Z | IPD DV _{DD33} | |
| EM_D3/ GP[17] | E16 | F20 | I/O/Z | IPD DV _{DD33} | |
| EM_D4/ GP[18] | E18 | G21 | I/O/Z | IPD DV _{DD33} | |
| EM_D5/ GP[19] | E17 | F22 | I/O/Z | IPD DV _{DD33} | |
| EM_D6/ GP[20] | F16 | F21 | I/O/Z | IPD DV _{DD33} | |
| EM_D7/ GP[21] | F17 | H20 | I/O/Z | IPD DV _{DD33} | |
| EM_D8/ GP[43] | A12 | A15 | I/O/Z | IPD DV _{DD33} | |
| EM_D9/ GP[42] | B13 | B15 | I/O/Z | IPD DV _{DD33} | |
| EM_D10/ GP[41] | C13 | B16 | I/O/Z | IPD DV _{DD33} | |
| EM_D11/ GP[40] | D14 | C18 | I/O/Z | IPD DV _{DD33} | |
| EM_D12/ GP[39] | B14 | A16 | I/O/Z | IPD DV _{DD33} | |
| EM_D13/ GP[38] | C14 | B17 | I/O/Z | IPD DV _{DD33} | |
| EM_D14/ GP[37] | B15 | B18 | I/O/Z | IPD DV _{DD33} | |
| EM_D15/ GP[36] | C15 | B19 | I/O/Z | IPD DV _{DD33} | |

Table 2-13. EMIFA Terminal Functions (EMIFA Pinout Mode 5, AEM[2:0] = 101)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EMIFA FUNCTIONAL PINS: 8-Bit NAND (EMIFA Pinout Mode 5, AEM[2:0] = 101) | | | | | |
| Actual pin functions are determined by the PINMUX0 and PINMUX1 register bit settings (e.g., PCIEN, AEM[2:0], etc.). For more details, see Section 3.7, Multiplexed Pin Configurations . | | | | | |
| EM_A[1]/(ALE)/ GP[9]/(PLLMS1) | A16 | B20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. When used for EMIFA (NAND), this pin is the Address Latch Enable output (ALE). |
| EM_A[2]/(CLE)/ GP[8]/(PLLMS0) | B16 | A20 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. When used for EMIFA (NAND), this pin is the Command Latch Enable output (CLE). |
| EM_WAIT/ (RDY/BSY) | E15 | D20 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), it is ready/busy input (RDY/ \overline{BSY}). |
| $\overline{EM_OE}$ | D15 | D19 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), this pin is read enable output (\overline{RE}). |
| $\overline{EM_WE}$ | E14 | C19 | I/O/Z | IPU DV _{DD33} | When used for EMIFA (NAND), this pin is write enable output (\overline{WE}). |
| $\overline{EM_CS2}$ / GP[12] | C19 | C22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), this pin is Chip Select 2 output $\overline{EM_CS2}$ for use with NAND flash. This is the chip select for the default boot and ROM boot modes. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{EM_CSx}$ function defaults to an inactive (high) state. |
| $\overline{EM_CS3}$ / GP[13] | C18 | D22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), this pin is Chip Select 3 output $\overline{EM_CS3}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{EM_CSx}$ function defaults to an inactive (high) state. |
| RMRXD0/ $\overline{EM_CS4}$ / GP[32] | E19 | H22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), it is Chip Select 4 output $\overline{EM_CS4}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{EM_CSx}$ function defaults to an inactive (high) state. |
| RMRXD1/ $\overline{EM_CS5}$ / GP[33] | F19 | J22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND), it is Chip Select 5 output $\overline{EM_CS5}$ for use with NAND flash. Note: This pin features an internal pulldown (IPD). If this pin is connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor must be used to ensure the $\overline{EM_CSx}$ function defaults to an inactive (high) state. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

(3) Specifies the operating I/O supply voltage for each signal

Table 2-13. EMIFA Terminal Functions (EMIFA Pinout Mode 5, AEM[2:0] = 101) (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_D0/ GP[14] | D16 | E21 | I/O/Z | IPD DV _{DD33} | These pins are multiplexed between EMIFA (NAND) and GPIO. For EMIFA (NAND) AEM[2:0] = 101, these pins are the 8-bit bi-directional data bus (EM_D[7:0]). |
| EM_D1/ GP[15] | D18 | G20 | I/O/Z | IPD DV _{DD33} | |
| EM_D2/ GP[16] | D17 | E22 | I/O/Z | IPD DV _{DD33} | |
| EM_D3/ GP[17] | E16 | F20 | I/O/Z | IPD DV _{DD33} | |
| EM_D4/ GP[18] | E18 | G21 | I/O/Z | IPD DV _{DD33} | |
| EM_D5/ GP[19] | E17 | F22 | I/O/Z | IPD DV _{DD33} | |
| EM_D6/ GP[20] | F16 | F21 | I/O/Z | IPD DV _{DD33} | |
| EM_D7/ GP[21] | F17 | H20 | I/O/Z | IPD DV _{DD33} | |

Table 2-14. DDR2 Memory Controller Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ₍₃₎ | DESCRIPTION |
|-------------------------------|---------|---------|---------------------|--|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| DDR2 Memory Controller | | | | | |
| DDR_CLK | W7 | AB7 | I/O/Z | DV _{DDR2} | DDR2 Clock Output |
| $\overline{\text{DDR_CLK}}$ | W8 | AB8 | I/O/Z | DV _{DDR2} | DDR2 Differential Clock Output |
| DDR_CKE | V8 | AA8 | I/O/Z | DV _{DDR2} | DDR2 Clock Enable Output |
| $\overline{\text{DDR_CS}}$ | T9 | Y11 | I/O/Z | DV _{DDR2} | DDR2 Active Low Chip Select Output |
| $\overline{\text{DDR_WE}}$ | T8 | Y10 | I/O/Z | DV _{DDR2} | DDR2 Active Low Write Enable Output |
| DDR_DQM[3] | T16 | Y18 | I/O/Z | DV _{DDR2} | DDR2 Data Mask Outputs DQM3: For upper byte data bus DDR_D[31:24] DQM2: For DDR_D[23:16] DQM1: For DDR_D[15:8] DQM0: For lower byte DDR_D[7:0] |
| DDR_DQM[2] | T14 | Y15 | I/O/Z | DV _{DDR2} | |
| DDR_DQM[1] | T6 | Y7 | I/O/Z | DV _{DDR2} | |
| DDR_DQM[0] | T4 | Y4 | I/O/Z | DV _{DDR2} | |
| $\overline{\text{DDR_RAS}}$ | U7 | Y8 | I/O/Z | DV _{DDR2} | DDR2 Row Access Signal Output |
| $\overline{\text{DDR_CAS}}$ | T7 | Y9 | I/O/Z | DV _{DDR2} | DDR2 Column Access Signal Output |
| DDR_DQS[0] | U4 | AA4 | I/O/Z | DV _{DDR2} | Data strobe input/outputs for each byte of the 32-bit data bus. They are outputs to the DDR2 memory when writing and inputs when reading. They are used to synchronize the data transfers. DQS3 : For upper byte DDR_D[31:24] DQS2: For DDR_D[23:16] DQS1: For DDR_D[15:8] DQS0: For bottom byte DDR_D[7:0] |
| DDR_DQS[1] | U6 | AA7 | I/O/Z | DV _{DDR2} | |
| DDR_DQS[2] | U14 | AA15 | I/O/Z | DV _{DDR2} | |
| DDR_DQS[3] | U16 | AA18 | I/O/Z | DV _{DDR2} | |
| DDR_BA[0] | U8 | AA9 | I/O/Z | DV _{DDR2} | Bank Select Outputs (BS[2:0]). Two are required to support 1Gb DDR2 memories. |
| DDR_BA[1] | V9 | AB9 | | | |
| DDR_BA[2] | U9 | AB10 | | | |
| DDR_A[12] | W9 | AA10 | I/O/Z | DV _{DDR2} | DDR2 Address Bus Output |
| DDR_A[11] | W10 | AA11 | | | |
| DDR_A[10] | U10 | AB11 | | | |
| DDR_A[9] | U11 | AA12 | | | |
| DDR_A[8] | V10 | Y12 | | | |
| DDR_A[7] | V11 | AB12 | | | |
| DDR_A[6] | W11 | AA13 | | | |
| DDR_A[5] | W12 | Y13 | | | |
| DDR_A[4] | V12 | AB13 | | | |
| DDR_A[3] | U12 | AA14 | | | |
| DDR_A[2] | V13 | Y14 | | | |
| DDR_A[1] | U13 | AB14 | | | |
| DDR_A[0] | W13 | AB15 | | | |

(1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
 (2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.
 (3) For more information, see the *Recommended Operating Conditions* table

Table 2-14. DDR2 Memory Controller Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ⁽³⁾ | DESCRIPTION |
|------------|------------|------------|---------------------|--|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| DDR_D[31] | T19 | Y22 | I/O/Z | DV _{DDR2} | DDR2 bi-directional data bus can be configured as 32-bits wide or 16-bits wide. |
| DDR_D[30] | U19 | AA21 | | | |
| DDR_D[29] | V18 | Y21 | | | |
| DDR_D[28] | U18 | AB20 | | | |
| DDR_D[27] | W17 | Y20 | | | |
| DDR_D[26] | T18 | AA20 | | | |
| DDR_D[25] | U17 | AB19 | | | |
| DDR_D[24] | V17 | Y19 | | | |
| DDR_D[23] | T17 | AA19 | | | |
| DDR_D[22] | V16 | AB18 | | | |
| DDR_D[21] | W16 | AB17 | | | |
| DDR_D[20] | U15 | Y17 | | | |
| DDR_D[19] | V15 | AA17 | | | |
| DDR_D[18] | W15 | AB16 | | | |
| DDR_D[17] | V14 | Y16 | | | |
| DDR_D[16] | W14 | AA16 | | | |
| DDR_D[15] | V7 | AB6 | | | |
| DDR_D[14] | W6 | Y6 | | | |
| DDR_D[13] | V6 | AA6 | | | |
| DDR_D[12] | W5 | AB5 | | | |
| DDR_D[11] | V5 | Y5 | | | |
| DDR_D[10] | U5 | AA5 | | | |
| DDR_D[9] | W4 | W5 | | | |
| DDR_D[8] | V4 | AB4 | | | |
| DDR_D[7] | W3 | W4 | | | |
| DDR_D[6] | V3 | AB3 | | | |
| DDR_D[5] | U3 | Y3 | | | |
| DDR_D[4] | V2 | AA3 | | | |
| DDR_D[3] | U2 | AA2 | | | |
| DDR_D[2] | U1 | W2 | | | |
| DDR_D[1] | T2 | Y2 | | | |
| DDR_D[0] | T1 | Y1 | | | |
| DDR_VREF | T15 | W18 | I | ⁽³⁾ | Reference voltage input for the SSTL_18 I/O buffers |
| DDR_VSSDLL | T13 | W15 | GND | ⁽³⁾ | Ground for the DDR2 DLL |
| DDR_VDDDLL | T12 | W14 | S | ⁽³⁾ | Power (1.8 Volts) for the DDR2 Digital Locked Loop |
| DDR_ZN | T10 | W12 | | ⁽³⁾ | Impedance control for DDR2 outputs. This must be connected via a 200-Ω resistor to DV _{DDR2} . |
| DDR_ZP | T11 | W13 | | ⁽³⁾ | Impedance control for DDR2 outputs. This must be connected via a 200-Ω resistor to V _{SS} . |

Table 2-15. Peripheral Component Interconnect (PCI) Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-----------------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| PCI | | | | | |
| EM_A[16]/PGNT/ GP[48] | C11 | B13 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the EMIFA, PCI, and GPIO. In PCI mode, this pin is PCI bus grant (I) |
| EM_A[18]/PRST/ GP[46] | D11 | A14 | I/O/Z | IPD DV _{DD33} | THIS pin is multiplexed between the EMIFA, PCI, and GPIO. In PCI mode, this pin is PCI reset (I) |
| EM_A[19]/PREG/ GP[45] | B12 | C14 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the EMIFA, PCI, and GPIO. In PCI mode, this pin is the PCI bus request (O/Z) |
| EM_A[20]/PINTA/ GP[44] | C12 | C15 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the EMIFA, PCI, and GPIO. In PCI mode, this pin is the PCI interrupt A (O/Z) |
| EM_A[12]/PCBE3/ GP[89] | D10 | B12 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. In PCI mode, this pin is the PCI command/byte enable 3 (I/O/Z). |
| HD3/VLYNQ_RXD2/ PCBE2 /GP[61] | B7 | B8 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI command/byte enable 2 (I/O/Z) |
| HD11/MTXD3/ PCBE1/GP[69] | C5 | A5 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In PCI mode, this pin is the PCI command/byte enable 1 (I/O/Z) |
| HRDY/MRXD2/ PCBE0/GP[80] | D2 | C3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In PCI mode, this pin is the PCI command/byte enable 0 (I/O/Z) |
| EM_A[9]/PIDSEL/ GP[92] | D9 | C11 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMIFA, PCI, and GPIO. In PCI mode, this pin is the PCI initialization device select (I) |
| VLYNQ_CLOCK/ PCICK/GP[57] | A7 | A8 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI clock (I) |
| HD4/VLYNQ_RXD3/ PFRAME/GP[62] | C7 | C8 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI frame (I/O/Z) |
| HD5/VLYNQ_TXD0/ PIRDY/GP[63] | A6 | A7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI initiator ready (I/O/Z) |
| HD6/VLYNQ_TXD1/ PTRDY/GP[64] | D6 | C7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI target ready (I/O/Z) |
| HD7/VLYNQ_TXD2/ PDEVSEL/GP[65] | B6 | B7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI device select (I/O/Z) |
| HD8/VLYNQ_TXD3/ PPERR/GP[66] | A5 | A6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. In PCI mode, this pin is the PCI parity error (I/O/Z) |
| HD9/MCOL/ PSTOP/GP[67] | C6 | C6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In PCI mode, this pin is the PCI stop (I/O/Z) |
| HD10/MCRS/ PSERR/GP[68] | B5 | B6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In PCI mode, this pin is the PCI system error (I/O/Z) |
| HD12/MTXD2/ PPAR/GP[70] | D5 | C5 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In PCI mode, this pin is the PCI parity (I/O/Z) |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-15. Peripheral Component Interconnect (PCI) Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---------------------------------|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EM_A[17]/AD31/ GP[47] | A11 | B14 | I/O/Z | IPD DV _{DD33} | <p>These pins are multiplexed between PCI, EMIFA, HPI, VLYNQ, EMAC (MII), and GPIO. For PCI, these pins are PCI data-address bus [31:0] (I/O/Z)</p> |
| AD30 | E12 | D14 | I/O/Z | IPD DV _{DD33} | |
| EM_A[15]/AD29/ GP[49] | B11 | C13 | I/O/Z | IPD DV _{DD33} | |
| AD28 | E11 | D13 | I/O/Z | IPD DV _{DD33} | |
| EM_A[14]/AD27/ GP[50] | A10 | A13 | I/O/Z | IPD DV _{DD33} | |
| AD26 | E10 | D12 | I/O/Z | IPD DV _{DD33} | |
| EM_A[13]/AD25/ GP[51] | B10 | A12 | I/O/Z | IPD DV _{DD33} | |
| EM_A[11]/AD24/GP[90] | C10 | C12 | I/O/Z | IPD DV _{DD33} | |
| EM_A[10]/AD23/GP[91] | A9 | B11 | I/O/Z | IPD DV _{DD33} | |
| EM_A[7]/AD22/GP[94] | C9 | C10 | I/O/Z | IPD DV _{DD33} | |
| EM_A[8]/AD21/GP[93] | B9 | A11 | I/O/Z | IPD DV _{DD33} | |
| EM_A[6]/AD20/GP[95] | D8 | B10 | I/O/Z | IPD DV _{DD33} | |
| EM_A[5]/AD19/GP[96] | B8 | A10 | I/O/Z | IPD DV _{DD33} | |
| HD0/VLYNQ_SCRUN/ AD18/GP[58] | C8 | B9 | I/O/Z | IPU DV _{DD33} | |
| HD2/VLYNQ_RXD1/ AD17/GP[60] | A8 | A9 | I/O/Z | IPD DV _{DD33} | |
| HD1/VLYNQ_RXD0/ AD16/GP[59] | D7 | C9 | I/O/Z | IPD DV _{DD33} | |
| HD14/MTXD0/ AD15/GP[72] | D4 | B5 | I/O/Z | IPD DV _{DD33} | |
| HD13/MTXD1/ AD14/GP[71] | B4 | B4 | I/O/Z | IPD DV _{DD33} | |
| HHWIL/MRXDV/ AD13/GP[74] | C4 | D3 | I/O/Z | IPD DV _{DD33} | |
| HD15/MTXCLK/ AD12/GP[73] | A4 | A4 | I/O/Z | IPD DV _{DD33} | |
| HCNTL1/MTXEN/ AD11/GP[75] | D3 | C4 | I/O/Z | IPD DV _{DD33} | |

Table 2-15. Peripheral Component Interconnect (PCI) Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|--|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| HCNTL0/MRXER/ AD10/GP[76] | B3 | B2 | I/O/Z | IPD DV _{DD33} | These pins are multiplexed between PCI, EMIFA, HPI, VLYNQ, EMAC (MII), and GPIO. For PCI, these pins are PCI data-address bus [31:0] (I/O/Z) |
| HDS2/MRXD0/ AD9/GP[78] | C3 | C2 | I/O/Z | IPU DV _{DD33} | |
| HR \overline{W} /MRXCLK/ AD8/GP[77] | A3 | A3 | I/O/Z | IPD DV _{DD33} | |
| HDS1/MRXD1/ AD7/GP[79] | B2 | B3 | I/O/Z | IPU DV _{DD33} | |
| HINT/MRXD3/ AD6/GP[82] | C2 | D2 | I/O/Z | IPU DV _{DD33} | |
| HCS/MDCLK/ AD5/GP[81] | C1 | D1 | I/O/Z | IPU DV _{DD33} | |
| AD4/GP[3] | E4 | F2 | I/O/Z | IPD DV _{DD33} | |
| HAS \overline{S} /MDIO/ AD3/GP[83] | D1 | C1 | I/O/Z | IPU DV _{DD33} | |
| AD2/GP[2] | E3 | F1 | I/O/Z | IPD DV _{DD33} | |
| AD1/GP[1] | E2 | E2 | I/O/Z | IPD DV _{DD33} | |
| AD0/GP[0] | E1 | E1 | I/O/Z | IPD DV _{DD33} | |

Table 2-16. EMAC (MII/RMII) and MDIO Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-------------------------------------|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| EMAC (MII) | | | | | |
| HCNTL1/MTXEN/ AD11/GP[75] | D3 | C4 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Enable output MTXEN. |
| HD15/MTXCLK/ AD12/GP[73] | A4 | A4 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Clock input MTXCLK. |
| HD9/MCOL/ PSTOP/GP[67] | C6 | C6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Collision Detect input MCOL. |
| HD11/MTXD3/ PCBE1/GP[69] | C5 | A5 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Data 3 output MTXD3. |
| HD12/MTXD2/ PPAR/GP[70] | D5 | C5 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Data 2 output MTXD2. |
| HD13/MTXD1/ AD14/GP[71] | B4 | B4 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Data 1 output MTXD1. |
| HD14/MTXD0/ AD15/GP[72] | D4 | B5 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Transmit Data 0 output MTXD0. |
| HR \bar{W} /MRXCLK/ AD8/GP[77] | A3 | A3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Clock input MRXCLK. |
| HHWIL/MRXDV/ AD13/GP[74] | C4 | D3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Data Valid input MRXDV. |
| HCNTL0/MRXER/ AD10/GP[76] | B3 | B2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Error input MRXER. |
| HD10/MCRS/ PSERR/GP[68] | B5 | B6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MI) mode, it is Carrier Sense input MCRS. |
| HINT/MRXD3/ AD6/GP[82] | C2 | D2 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Data 3 input MRXD3. |
| HRDY/MRXD2/ PCBE0/GP[80] | D2 | C3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Data 2 input MRXD2. |
| HDS1/MRXD1/ AD7/GP[79] | B2 | B3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive data 1 input MRXD1. |
| HDS2/MRXD0/ AD9/GP[78] | C3 | C2 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In Ethernet MAC (MII) mode, it is Receive Data 0 input MRXD0. |
| EMAC (RMII) | | | | | |
| RMCRSDV/GP[30] | G19 | K22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC carrier sense/receive data valid (RMCRSDV) [I]. |
| RMRXER/GP[52] | A15 | A19 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC receive error (RMRXER) [I]. |
| RMTXD1/GP[27]/ (LENDIAN) | H17 | L19 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC transmit data pin 1 (RMTXD1) [O/Z]. |
| RMTXD0/GP[28]/ | H16 | J21 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC transmit data pin 0 (RMTXD0) [O/Z]. |
| RMREFCLK/GP[31] | D19 | G22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC RMII reference clock (RMREFCLK) [I]. |
| RMTXEN/GP[29] | H15 | K21 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII) and GPIO. In Ethernet MAC (RMII) mode, it is EMAC transmit enable (RMTXEN) [O/Z]. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see Section 3.9.1, Pullup/Pulldown Resistors.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-16. EMAC (MII/RMII) and MDIO Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| RMRXD1/ $\overline{\text{EM_CS5}}$ / GP[33] | F19 | J22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII), EMIFA, and GPIO. In Ethernet MAC (RMII) mode, it is EMAC receive data pin 1 (RMRXD1) [1]. |
| RMRXD0/ $\overline{\text{EM_CS4}}$ / GP[32] | E19 | H22 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between EMAC (RMII), EMIFA, and GPIO. In Ethernet MAC (RMII) mode, it is EMAC receive data pin 0 (RMRXD0) [1]. |
| MDIO | | | | | |
| $\overline{\text{HCS}}$ /MDCLK/ AD5/GP[81] | C1 | D1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, MDIO, PCI, and GPIO. In Ethernet MAC mode, it is Management Data Clock output MDCLK. |
| $\overline{\text{HAS}}$ /MDIO/ AD3/GP[83] | D1 | C1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, MDIO, PCI, and GPIO. In Ethernet MAC mode, it is Management Data IO MDIO [1/O/Z]. |

Table 2-17. VLYNQ Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-----------------------------------|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| VLYNQ | | | | | |
| VLYNQ_CLOCK/ PCICLK/GP[57] | A7 | A8 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between VLYNQ, PCI, and GPIO. For VLYNQ, it is the clock VLYNQ_CLOCK [I/O/Z]. |
| HD0/VLYNQ_SCRUN/ AD18/GP[58] | C8 | B9 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is the Serial Clock run request VLYNQ_SCRUN [I/O/Z]. |
| HD8/VLYNQ_TXD3/ PPERR/GP[66] | A5 | A6 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is transmit bus bit 3 output VLYNQ_TXD3. |
| HD7/VLYNQ_TXD2/ PDEVSEL/GP[65] | B6 | B7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is transmit bus bit 2 output VLYNQ_TXD2. |
| HD6/VLYNQ_TXD1/ PTRDY/GP[64] | D6 | C7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is transmit bus bit 1 output VLYNQ_TXD1. |
| HD5/VLYNQ_TXD0/ PIRDY/GP[63] | A6 | A7 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is transmit bus bit 0 output VLYNQ_TXD0. |
| HD4/VLYNQ_RXD3/ PFRAME/GP[62] | C7 | C8 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is receive bus bit 3 input VLYNQ_RXD3. |
| HD3/VLYNQ_RXD2/ PCBE2/GP[61] | B7 | B8 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is receive bus bit 2 input VLYNQ_RXD2. |
| HD2/VLYNQ_RXD1/ AD17/GP[60] | A8 | A9 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is receive bus bit 1 input VLYNQ_RXD1. |
| HD1/VLYNQ_RXD0/ AD16/GP[59] | D7 | C9 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, VLYNQ, PCI, and GPIO. For VLYNQ, it is receive bus bit 0 input VLYNQ_RXD0. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-18. Host-Port Interface Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-------------------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| Host-Port Interface (HPI) | | | | | |
| HD0/VLYNQ_SCRUN/ AD18/GP[58] | C8 | B9 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, VLYNQ or EMAC (MII), PCI, and GPIO. In HPI mode, these pins are host-port data pins HD[15:0] (I/O/Z) and are multiplexed internally with the HPI address lines. |
| HD1/VLYNQ_RXD0/ AD16/GP[59] | D7 | C9 | | | |
| HD2/VLYNQ_RXD1/ AD17/GP[60] | A8 | A9 | | | |
| HD3/VLYNQ_RXD2/ PCBE2/GP[61] | B7 | B8 | | | |
| HD4/VLYNQ_RXD3/ PFRAME/GP[62] | C7 | C8 | | | |
| HD5/VLYNQ_TXD0/ PIRDY/GP[63] | A6 | A7 | | | |
| HD6/VLYNQ_TXD1/ PTRDY/GP[64] | D6 | C7 | | | |
| HD7/VLYNQ_TXD2/ PDEVSEL/GP[65] | B6 | B7 | | | |
| HD8/VLYNQ_TXD3/ PPERR/GP[66] | A5 | A6 | | | |
| HD9/MCOL/ PSTOP/GP[67] | C6 | C6 | | | |
| HD10/MCRS/ PSERR/GP[68] | B5 | B6 | | | |
| HD11/MTXD3/ PCBE1/GP[69] | C5 | A5 | | | |
| HD12/MTXD2/ PPAR/GP[70] | D5 | C5 | | | |
| HD13/MTXD1/ AD14/GP[71] | B4 | B4 | | | |
| HD14/MTXD0/ AD15/GP[72] | D4 | B5 | | | |
| HD15/MTXCLK/ AD12/GP[73] | A4 | A4 | | | |
| HHWIL/MRXDV/ AD13/GP[74] | C4 | D3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is half-word identification input HHWIL (I). |
| HCNTL1/MTXEN/ AD11/GP[75] | D3 | C4 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is control input 1 HCNTL1 (I). The state of HCNTL1 and HCNTL0 determines if address, data, or control information is being transmitted between an external host and the C6424. |
| HCNTL0/MRXER/ AD10/GP[76] | B3 | B2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is control input 0 HCNTL0 (I). The state of HCNTL1 and HCNTL0 determines if address, data, or control information is being transmitted between an external host and the C6424. |
| HR \bar{W} /MRXCLK/ AD8/GP[77] | A3 | A3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is host read or write select HR \bar{W} (I). |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-18. Host-Port Interface Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|--|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| $\overline{\text{HDS2}}/\text{MRXD0}/\text{AD9}/\text{GP}[78]$ | C3 | C2 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is host data strobe input 2 $\overline{\text{HDS2}}$ (I). |
| $\overline{\text{HDS1}}/\text{MRXD1}/\text{AD7}/\text{GP}[79]$ | B2 | B3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is host data strobe input 1 $\overline{\text{HDS1}}$ (I). |
| $\overline{\text{HRDY}}/\text{MRXD2}/\text{PCBE0}/\text{GP}[80]$ | D2 | C3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is host ready output from DSP to host (O/Z). |
| $\overline{\text{HCS}}/\text{MDCLK}/\text{AD5}/\text{GP}[81]$ | C1 | D1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, MDIO, PCI, and GPIO. In HPI mode, this pin is HPI active low chip select input HCS (I). |
| $\overline{\text{HINT}}/\text{MRXD3}/\text{AD6}/\text{GP}[82]$ | C2 | D2 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, EMAC (MII), PCI, and GPIO. In HPI mode, this pin is host interrupt output $\overline{\text{HINT}}$ (O/Z). |
| $\overline{\text{HAS}}/\text{MDIO}/\text{AD3}/\text{GP}[83]$ | D1 | C1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between HPI, MDIO, PCI, and GPIO. In HPI mode, this pin is host address strobe HAS (I). For proper HPI operation, if this pin is routed out, it must be pulled up via an external resistor. |

Table 2-19. I2C Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|------------|---------|---------|---------------------|--------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| I2C | | | | | |
| SCL | M2 | N2 | I/O/Z | DV _{DD33} | For I2C, this pin is I2C clock. In I2C master mode, this pin is an output. In I2C slave mode, this pin is an input. When the I2C module is used, for proper device operation, this pin must be pulled up via an external resistor. |
| SDA | M3 | P2 | I/O/Z | DV _{DD33} | For I2C, this pin is the I2C bi-directional data signal. When the I2C module is used, for proper device operation, this pin must be pulled up via an external resistor. |

- (1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.
(3) Specifies the operating I/O supply voltage for each signal

Table 2-20. Multichannel Buffered Serial Port 0 and Multichannel Buffered Serial Port 1 (McBSP0 and McBSP1) Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| Multichannel Buffered Serial Port 0 (McBSP0) | | | | | |
| For more details on pin multiplexing, see Section 3.7, Multiplexed Pin Configurations . | | | | | |
| CLKS0/TOUT0L/GP[97] | J4 | L3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McBSP0, Timer0, and GPIO. For McBSP0, it is McBSP0 external clock source (I). |
| ACLKR0/CLKX0/GP[99] | H1 | J1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 transmit clock CLKX0 (I/O/Z). |
| AHCLKR0/CLKR0/GP[101] | J2 | K1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 receive clock CLKR0 (I/O/Z). |
| AXR0[2]/FSX0/GP[103] | H3 | J2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 transmit frame synchronization FSX0 (I/O/Z). |
| AXR0[3]/FSR0/GP[102] | G4 | J3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 receive frame synchronization FSR0 (I/O/Z). |
| AXR0[1]/DX0/GP[104] | J3 | K2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 data transmit output DX0 (O/Z). |
| AFSR0/DR0/GP[100] | H4 | K3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McBSP0, it is McBSP0 data receive input DR0 (I). |
| Multichannel Buffered Serial Port 1 (McBSP1) | | | | | |
| For more details on pin multiplexing, see Section 3.7, Multiplexed Pin Configurations . | | | | | |
| CLKS1/TINP0L/GP[98] | K2 | L2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McBSP0, Timer0, and GPIO. For McBSP1, it is McBSP1 external clock source (I). |
| ACLKX0/CLKX1/GP[106] | F1 | G1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 transmit clock CLKX1 (I/O/Z). |
| AHCLKX0/CLKR1/GP[108] | G1 | H1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 receive clock CLKR1 (I/O/Z). |
| AMUTEIN0/FSX1/GP[109] | F2 | G3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 transmit frame synchronization FSX1 (I/O/Z). |
| AXR0[0]/FSR1/GP[105] | H2 | H2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 receive frame synchronization FSR1 (I/O/Z). |
| AFSX0/DX1/GP[107] | G2 | G2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 data transmit output DX1 (O/Z). |
| AMUTE0/DR1/GP[110] | G3 | H3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McBSP1, it is McBSP1 data receive input DR1 (I). |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

(3) Specifies the operating I/O supply voltage for each signal

Table 2-21. Multichannel Audio Serial Port (McASP0) Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---------------------------|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| McASP0 | | | | | |
| AMUTEIN0/FSX1/ GP[109] | F2 | G3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 mute input AMUTEIN0 (I). |
| AMUTE0/DR1/ GP[110] | G3 | H3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 mute output AMUTE0 (O/Z). |
| ACLKR0/CLKX0/ GP[99] | H1 | J1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 receive bit clock ACLKR0 (I/O/Z). |
| AHCLKR0/CLKR0/ GP[101] | J2 | K1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 receive high-frequency master clock AHCLKR0 (I/O/Z). |
| ACLKX0/CLKX1/ GP[106] | F1 | G1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 transmit bit clock ACLKX0 (I/O/Z). |
| AHCLKX0/CLKR1/ GP[108] | G1 | H1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 transmit high-frequency master clock AHCLKX0 (I/O/Z). |
| AFSR0/DR0/ GP[100] | H4 | K3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 receive frame synchronization AFSR0 (I/O/Z). |
| AFSX0/DX1/ GP[107] | G2 | G2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 transmit frame synchronization AFSX0 (I/O/Z). |
| AXR0[3]/FSR0/ GP[102] | G4 | J3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 transmit/receive (TX/RX) data pin 3 AXR0[3] (I/O/Z). |
| AXR0[2]/FSX0/ GP[103] | H3 | J2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 transmit/receive (TX/RX) data pin 2 AXR0[2] (I/O/Z). |
| AXR0[1]/DX0/ GP[104] | J3 | K2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP0, and GPIO. For McASP0, it is McASP0 transmit/receive (TX/RX) data pin 1 AXR0[1] (I/O/Z). |
| AXR0[0]/FSR1/ GP[105] | H2 | H2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between McASP0, McBSP1, and GPIO. For McASP0, it is McASP0 transmit/receive (TX/RX) data pin 0 AXR0[0] (I/O/Z). |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-22. UART0 and UART1 Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ⁽³⁾ | DESCRIPTION |
|-----------------------------|---------|---------|---------------------|-------------------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| UART1 | | | | | |
| TINP1L/ URXD1/ GP[56] | L4 | P3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the Timer 1, UART1 (Data), and GPIO. Fo UART1 this pin is the receive data input URXD1. |
| TOUT1L/ UTXD1/ GP[55] | K4 | N3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the Timer 1, UART1 (Data), and GPIO. Fo UART1 this pin is the transmit data output UTXD1. |
| UART0 | | | | | |
| URXD0/ GP[85] | L2 | M2 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between UART0 (Data) and GPIO. When used by UART0 this pin is the receive data input URXD0. |
| UTXD0/ GP[86] | K3 | N1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between UART0 (Data) and GPIO. In UART0 mode, this pin is the transmit data output UTXD0. |
| UCTS0/ GP[87] | L1 | P1 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the UART0 (Flow Control) and GPIO. In UART0 mode, this pin is the clear to send input UCTS0. |
| URTS0/ PWM0/ GP[88] | L3 | M3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the UART0 (Flow Control), PWM0, and GPIO. In UART0 mode, this pin is the ready to send output URTS0. |

- (1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.
(3) Specifies the operating I/O supply voltage for each signal

Table 2-23. PWM0, PWM1, and PWM2 Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-------------------------|---------|---------|---------------------|---------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| PWM2 | | | | | |
| CLKOUT0/PWM2/ GP[84] | M1 | R1 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the System Clock generator (PLL1), PWM2, and GPIO. For PWM2, this pin is output PWM2. |
| PWM1 | | | | | |
| GP[4]/PWM1 | F3 | F3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between GPIO and PWM1. For PWM1, this pin is output PWM1. |
| PWM0 | | | | | |
| URTS0/PWM0/ GP[88] | L3 | M3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the UART0 (Flow Control), PWM0, and GPIO. For PWM0, this pin is output PWM0. |

- (1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.
(3) Specifies the operating I/O supply voltage for each signal

Table 2-24. Timer 0, Timer 1, and Timer 2 Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|---|---------|---------|---------------------|---------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| Timer 2 | | | | | |
| No external pins. The Timer 2 (watchdog) peripheral pins are not pinned out as external pins. | | | | | |
| Timer 1 | | | | | |
| TINP1L/ URXD1/ GP[56] | L4 | P3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the Timer 1, UART1 (Data), and GPIO. For Timer 1, this pin is the timer 1 input pin for the lower 32-bit counter |
| TOUT1L/ UTXD1/ GP[55] | K4 | N3 | I/O/Z | IPU DV _{DD33} | This pin is multiplexed between the Timer 1, UART1, and GPIO. For Timer 1, this pin is the timer 1 output pin for the lower 32-bit counter |
| Timer 0 | | | | | |
| CLKS1/ TINP0L/ GP[98] | K2 | L2 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the McBSP1, Timer 0, and GPIO. For Timer 0, this pin is the timer 0 input pin for the lower 32-bit counter |
| CLKS0/ TOUT0L/ GP[97] | J4 | L3 | I/O/Z | IPD DV _{DD33} | This pin is multiplexed between the McBSP0, Timer 0, and GPIO. For Timer 0, this pin is the timer 0 output pin for the lower 32-bit counter |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-25. Multiplexed GPIO 3.3 V Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|--|---------|---------|---------------------|--------------------------|-------------|
| NAME | ZWT NO. | ZDU NO. | | | |
| GPIO | | | | | |
| 104 out of 111 GPIO pins on the C6424 device are multiplexed with other peripherals pin functions (e.g., PCI, HPI, VLYNQ, EMAC/MDIO, McASP0, McBSP0, McBSP1, Timer 0, Timer 1, UART0, UART1, PWM0, PWM1, PWM2, EMIFA, and the CLKOUT0 pin), see the peripheral-specific Terminal Functions tables for the GPIO multiplexing. | | | | | |

- (1) I = Input, O = Output, Z = High impedance, S = Supply voltage, GND = Ground, A = Analog signal
 (2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.
 (3) Specifies the operating I/O supply voltage for each signal

Table 2-26. Standalone GPIO 3.3 V Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ⁽²⁾ ⁽³⁾ | DESCRIPTION |
|------------------------------|---------|---------|---------------------|-------------------------------------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| Standalone GPIO 3.3 V | | | | | |
| GP[22]/ (BOOTMODE0) | F18 | J20 | I/O/Z | IPD DV _{DD33} | These pins function as boot configuration pins during device reset. After device reset, these pins function as standalone general-purpose input/output (GPIO). |
| GP[23]/ (BOOTMODE1) | F15 | K20 | I/O/Z | IPD DV _{DD33} | |
| GP[24]/ (BOOTMODE2) | G15 | L20 | I/O/Z | IPD DV _{DD33} | |
| GP[25]/ (BOOTMODE3) | G16 | H21 | I/O/Z | IPD DV _{DD33} | |
| GP[26]/ (FASTBOOT) | G17 | K19 | I/O/Z | IPD DV _{DD33} | |
| GP[53] | A13 | A17 | I/O/Z | IPD DV _{DD33} | This pin is standalone and functions as GPIO pin 53. |
| GP[54] | A14 | A18 | I/O/Z | IPD DV _{DD33} | This pin is standalone and functions as GPIO pin 54. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-27. Reserved Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER ^{(2) (3)} | DESCRIPTION |
|-----------------|---------|---------|---------------------|--------------------------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| RESERVED | | | | | |
| RSV1 | E5 | D4 | | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV2 | K5 | L4 | | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV3 | L5 | M4 | | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV4 | L15 | P19 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV5 | R13 | W16 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV6 | N19 | V22 | A I | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |
| RSV7 | P19 | V21 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV8 | P18 | U22 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV9 | N18 | T21 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV10 | N17 | T22 | A O | | Reserved. (Leave unconnected, do not connect to power or ground) |
| RSV11 | P16 | U20 | | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |
| RSV12 | P17 | V20 | | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |
| RSV13 | N15 | T20 | | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |
| RSV14 | P15 | T19 | | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |
| RSV15 | N16 | U21 | | | Reserved. This pin must be tied directly to V _{SS} for normal device operation. |

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

(2) IPD = Internal pulldown, IPU = Internal pullup. For more detailed information on pullup/pulldown resistors and situations where external pullup/pulldown resistors are required, see [Section 3.9.1](#), *Pullup/Pulldown Resistors*.

(3) Specifies the operating I/O supply voltage for each signal

Table 2-28. Supply Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER | DESCRIPTION |
|----------------------------|---------|---------|---------------------|-------|--|
| NAME | ZWT NO. | ZDU NO. | | | |
| SUPPLY VOLTAGE PINS | | | | | |
| DV _{DD33} | A1 | A2 | S | | 3.3 V I/O supply voltage (see Section 6.3.3 , Power-Supply Decoupling.) |
| | A2 | A21 | | | |
| | A18 | B1 | | | |
| | E6 | D6 | | | |
| | E8 | D8 | | | |
| | F5 | D10 | | | |
| | F7 | D16 | | | |
| | F9 | D18 | | | |
| | F11 | E3 | | | |
| | F13 | E5 | | | |
| | G6 | E7 | | | |
| | G8 | E9 | | | |
| | G10 | E11 | | | |
| | G12 | E13 | | | |
| | G14 | E15 | | | |
| | H5 | E17 | | | |
| | H18 | E19 | | | |
| | J1 | F4 | | | |
| | J6 | F18 | | | |
| | J14 | G5 | | | |
| | J16 | G19 | | | |
| | K15 | H4 | | | |
| | K17 | H18 | | | |
| | L6 | J5 | | | |
| | M5 | J19 | | | |
| | M15 | K4 | | | |
| | N6 | K18 | | | |
| | P1 | L1 | | | |
| | | L5 | | | |
| | | L21 | | | |
| | M18 | | | | |
| | M20 | | | | |
| | N5 | | | | |
| | N19 | | | | |
| | P4 | | | | |
| | P18 | | | | |
| | P20 | | | | |
| | P22 | | | | |
| | R5 | | | | |
| | T4 | | | | |

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

Table 2-28. Supply Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER | DESCRIPTION |
|--------------------|---------|---------|---------------------|-------|---|
| NAME | ZWT NO. | ZDU NO. | | | |
| DV _{DDR2} | L14 | U5 | S | | 1.8 V DDR2 I/O supply voltage (see the Power-Supply Decoupling section of this data manual) |
| | P5 | V1 | | | |
| | P7 | V4 | | | |
| | P9 | V6 | | | |
| | P11 | V8 | | | |
| | P13 | V10 | | | |
| | R4 | V12 | | | |
| | R6 | V14 | | | |
| | R8 | V16 | | | |
| | R10 | V18 | | | |
| | R12 | W7 | | | |
| | R14 | W9 | | | |
| | R16 | W11 | | | |
| | T5 | W17 | | | |
| | V1 | W19 | | | |
| | W18 | AA1 | | | |
| W19 | AB21 | | | | |
| | AB22 | | | | |
| CV _{DD} | H7 | J10 | S | | 1.20 V supply voltage (-7/-6/-5/-4/-Q6/-Q5/-Q4 devices) 1.05 V core supply voltage (-7/-6/-5/-4/-L/-Q5 devices) (see the Power-Supply Decoupling section of this data manual) |
| | H9 | J11 | | | |
| | H11 | J12 | | | |
| | H13 | J13 | | | |
| | J8 | K9 | | | |
| | J10 | K14 | | | |
| | J12 | L9 | | | |
| | K7 | L13 | | | |
| | K9 | L14 | | | |
| | K11 | M9 | | | |
| | K13 | M10 | | | |
| | L8 | M14 | | | |
| | L10 | N9 | | | |
| | L12 | N14 | | | |
| | M7 | P10 | | | |
| | M9 | P11 | | | |
| | M11 | P12 | | | |
| | M13 | P13 | | | |
| N8 | | | | | |
| N10 | | | | | |
| N12 | | | | | |

Table 2-29. Ground Terminal Functions

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER | DESCRIPTION |
|--------------------|---------|---------|---------------------|-------|-------------|
| NAME | ZWT NO. | ZDU NO. | | | |
| GROUND PINS | | | | | |
| V_{SS} | A19 | A1 | GND | | Ground pins |
| | B1 | A22 | | | |
| | B19 | B22 | | | |
| | E7 | D5 | | | |
| | E9 | D7 | | | |
| | E13 | D9 | | | |
| | F4 | D11 | | | |
| | F6 | D15 | | | |
| | F8 | D17 | | | |
| | F10 | E4 | | | |
| | F12 | E6 | | | |
| | F14 | E8 | | | |
| | G5 | E10 | | | |
| | G7 | E12 | | | |
| | G9 | E14 | | | |
| | G11 | E16 | | | |
| | G13 | E18 | | | |
| | G18 | F5 | | | |
| | H6 | F19 | | | |
| | H8 | G4 | | | |
| | H10 | G18 | | | |
| | H12 | H5 | | | |
| | H14 | H19 | | | |
| | H19 | J4 | | | |
| | J5 | J9 | | | |
| | J7 | J14 | | | |
| | J9 | J18 | | | |
| | J11 | K5 | | | |
| | J13 | K10 | | | |
| | J15 | K11 | | | |
| | J17 | K12 | | | |
| | J18 | K13 | | | |
| K1 | L10 | | | | |
| K6 | L11 | | | | |
| K8 | L12 | | | | |
| K10 | L18 | | | | |
| K12 | L22 | | | | |
| K14 | M1 | | | | |
| K16 | M5 | | | | |

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

Table 2-29. Ground Terminal Functions (continued)

| SIGNAL | | | TYPE ⁽¹⁾ | OTHER | DESCRIPTION |
|-----------------|---------|---------|---------------------|-------|-------------|
| NAME | ZWT NO. | ZDU NO. | | | |
| V _{SS} | L7 | M11 | GND | | Ground pins |
| | L9 | M12 | | | |
| | L11 | M13 | | | |
| | L13 | M19 | | | |
| | L17 | N4 | | | |
| | L19 | N10 | | | |
| | M6 | N11 | | | |
| | M8 | N12 | | | |
| | M10 | N13 | | | |
| | M12 | N18 | | | |
| | M14 | P5 | | | |
| | M16 | P9 | | | |
| | M17 | P14 | | | |
| | M18 | P21 | | | |
| | M19 | R4 | | | |
| | N5 | R18 | | | |
| | N7 | R19 | | | |
| | N9 | R20 | | | |
| | N11 | R21 | | | |
| | N13 | R22 | | | |
| | N14 | T5 | | | |
| | P6 | T18 | | | |
| | P8 | U4 | | | |
| | P10 | U18 | | | |
| | P12 | U19 | | | |
| | P14 | V5 | | | |
| | R1 | V7 | | | |
| | R5 | V9 | | | |
| | R7 | V11 | | | |
| | R9 | V13 | | | |
| | R11 | V15 | | | |
| | R15 | V17 | | | |
| | R17 | V19 | | | |
| | R18 | W1 | | | |
| R19 | W6 | | | | |
| V19 | W8 | | | | |
| W1 | W10 | | | | |
| W2 | W20 | | | | |
| | W21 | | | | |
| | W22 | | | | |
| | AA22 | | | | |
| | AB1 | | | | |
| | AB2 | | | | |

2.6 Device Support

2.6.1 Development Support

TI offers an extensive line of development tools for the TMS320C642x DSP platform, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules. The tool's support documentation is electronically available within the Code Composer Studio™ Integrated Development Environment (IDE).

The following products support development of TMS320C642x DSP-based applications:

Software Development Tools:

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

Hardware Development Tools:

Extended Development System (XDS™) Emulator (supports TMS320C642x DSP multiprocessor system debug) EVM (Evaluation Module)

For a complete listing of development-support tools for the TMS320C642x DSP platform, visit the Texas Instruments web site on the Worldwide Web at <http://www.ti.com> uniform resource locator (URL). For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

2.7 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., **TMX320C6424ZWTQ6**). 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:

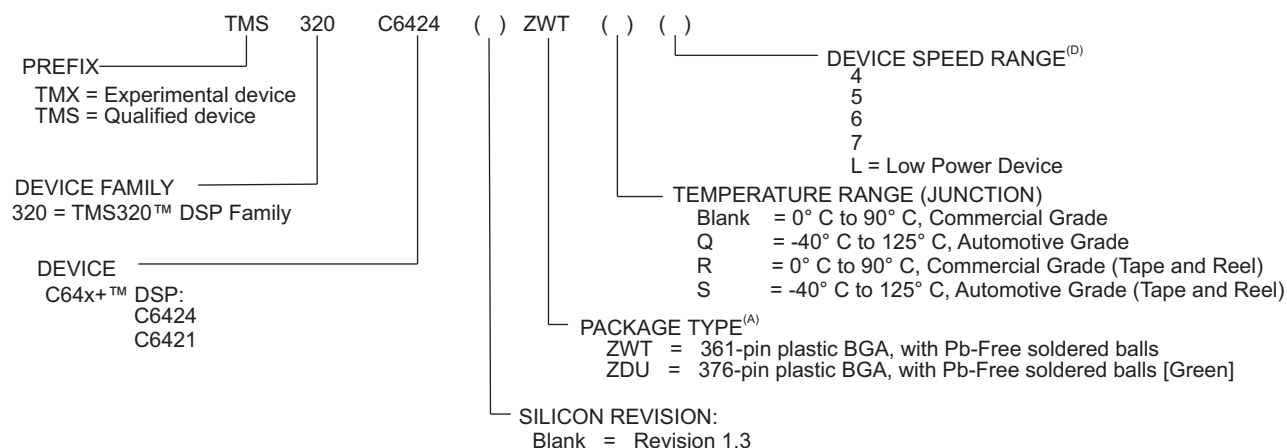
"Developmental product is intended for internal evaluation purposes."

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, ZWT), the temperature range (for example, "Blank" is the commercial temperature range), and the device speed range in megahertz (for example, "Blank" is the default [600-MHz]).

Figure 2-14 provides a legend for reading the complete device name for any TMS320C642x DSP platform member.



- A. BGA = Ball Grid Array
- B. Not all combinations are available. For more information, see the *Orderable Devices* table in the Packing Information section.
- C. The device speed range symbolization indicates the maximum CPU frequency at the highest CV_{DD} voltage supported. To determine the maximum CPU frequency at other supported CV_{DD} voltages, see Section 6.7.1, PLL1 and PLL2.
- D. For the actual device operating frequency (speed), see the $F_{SYSCLK1}$ in Section 5.2, Recommended Operating Conditions.

Figure 2-14. Device Nomenclature^{(C)(D)}

2.8 Documentation Support

2.8.1 Related Documentation From Texas Instruments

The following documents describe the TMS320C642x Fixed-Point Digital Signal Processor (DSP). 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 current documentation that describes the C642x DSP, related peripherals, and other technical collateral, is available in the C6000 DSP product folder at: www.ti.com/c6000.

[SPRUJEM3](#) *TMS320C642x DSP Peripherals Overview Reference Guide.* Provides an overview and briefly describes the peripherals available on the TMS320C642x Digital Signal Processor (DSP).

[SPRAA84](#) *TMS320C64x to TMS320C64x+ CPU Migration Guide.* Describes migrating from the Texas Instruments TMS320C64x digital signal processor (DSP) to the TMS320C64x+ DSP. The objective of this document is to indicate differences between the two cores. Functionality in the devices that is identical is not included.

[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.

[SPRU871](#) *TMS320C64x+ DSP Megamodule Reference Guide.* Describes the TMS320C64x+ digital signal processor (DSP) megamodule. Included is a discussion on the internal direct memory access (IDMA) controller, the interrupt controller, the power-down controller, memory protection, bandwidth management, and the memory and cache.

3 Device Configurations

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 configurations are discussed in the following sections.

Table 3-1. System Module Register Memory Map

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---------------------------|------------------|---|
| 0x01C4 0000 | PINMUX0 | Pin Multiplexing Control 0 (see Section 3.7.2.1 , <i>PINMUX0 Register Description</i>). |
| 0x01C4 0004 | PINMUX1 | Pin Multiplexing Control 1 (see Section 3.7.2.2 , <i>PINMUX1 Register Description</i>). |
| 0x01C4 0008 | DSPBOOTADDR | DSP Boot Address (see Section 3.4.2.3 , <i>DSPBOOTADDR Register</i>). |
| 0x01C4 000C | BOOTCOMPLT | Boot Complete (see Section 3.4.2.2 , <i>BOOTCOMPLT Register</i>). |
| 0x01C4 0010 | – | Reserved |
| 0x01C4 0014 | BOOTCFG | Device Boot Configuration (see Section 3.4.2.1 , <i>BOOTCFG Register</i>). |
| 0x01C4 0018 - 0x01C4 0027 | – | Reserved |
| 0x01C4 0028 | JTAGID | JTAG ID (see Section 6.22.1 , <i>JTAG ID (JTAGID) Register Description(s)</i>). |
| 0x01C4 002C | – | Reserved |
| 0x01C4 0030 | HPICTL | HPI Control (see Section 3.6.2.1 , <i>HPI Control Register</i>). |
| 0x01C4 0034 | – | Reserved |
| 0x01C4 0038 | – | Reserved |
| 0x01C4 003C | MSTPRI0 | Bus Master Priority Control 0 (see Section 3.6.1 , <i>Switch Central Resource (SCR) Bus Priorities</i>). |
| 0x01C4 0040 | MSTPRI1 | Bus Master Priority Control 1 (see Section 3.6.1 , <i>Switch Central Resource (SCR) Bus Priorities</i>). |
| 0x01C4 0044 | – | Reserved |
| 0x01C4 0048 | VDD3P3V_PWDN | VDD 3.3-V I/O Powerdown Control (see Section 3.2 , <i>Power Considerations</i>). |
| 0x01C4 004C | DDRVTPER | DDR2 VTP Enable Register (see Section 6.9.4 , <i>DDR2 Memory Controller</i>). |
| 0x01C4 0050 - 0x01C4 0080 | – | Reserved |
| 0x01C4 0084 | TIMERCTL | Timer Control (see Section 3.6.2.2 , <i>Timer Control Register</i>). |
| 0x01C4 0088 | EDMATCCFG | EDMA Transfer Controller Default Burst Size Configuration (see Section 3.6.2.3 , <i>EDMA TC Configuration Register</i>). |
| 0x01C4 008C | – | Reserved |

3.2 Power Considerations

The C6424 provides several means of managing power consumption.

As described in the [Section 6.3.4](#), *C6424 Power and Clock Domains*, the C6424 has one single power domain—the “Always On” power domain. Within this power domain, the C6424 utilizes local clock gating via the Power and Sleep Controller (PSC) to achieve power savings. For more details on the PSC, see [Section 6.3.5](#), *Power and Sleep Controller (PSC)* and the *TMS320C642x Power and Sleep Controller (PSC) User's Guide* (literature number [SPRUEN8](#)).

Some of the C6424 peripherals support additional power saving features. For more details on power saving features supported, see the *TMS320C642x DSP Peripherals Overview Reference Guide* (literature number [SPRUEM3](#)).

Most C6424 3.3-V I/Os can be powered-down to reduce power consumption. The VDD3P3V_PWDN register in the System Module (see [Figure 3-1](#)) is used to selectively power down unused 3.3-V I/O pins. For independent control, the 3.3-V I/Os are separated into functional groups—most of which are named according to the pin multiplexing groups (see [Table 3-2](#)). For these I/O groups, only the I/O buffers needed for Host/EMIFA Boot or Power-Up Operations are powered up by default (CLKOUT Block, EMIFA Block, Host Block, PCI Data Block, and GPIO Block).

Note: To save power, all other I/O buffers are powered down by default. Before using these pins, the user *must* program the VDD3P3V_PWDN register to power up the corresponding I/O buffers.

For a list of multiplexed pins on the device and the pin mux group each pin belongs to, see [Section 3.7.3.1, Multiplexed Pins on C6424](#).

Note: The VDD3P3V_PWDN register *only* controls the power to the I/O buffers. The Power and Sleep Controller (PSC) determines the clock/power state of the peripheral.

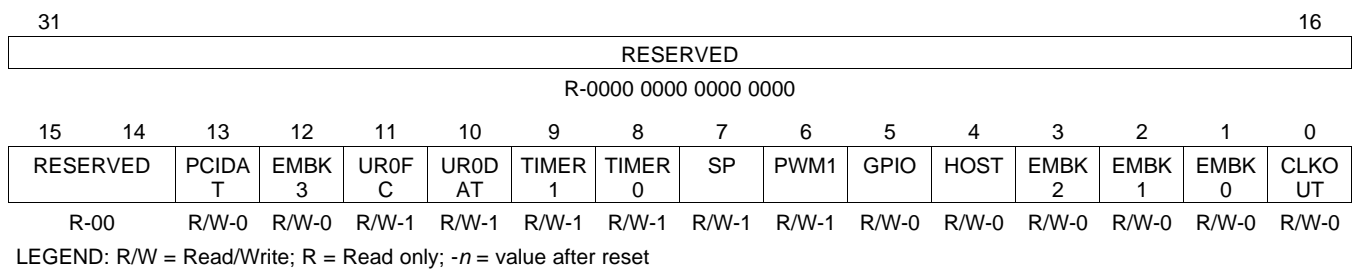


Figure 3-1. VDD3P3V_PWDN Register— 0x01C4 0048

Table 3-2. VDD3P3V_PWDN Register Bit Descriptions⁽¹⁾

| BIT | NAME | DESCRIPTION |
|-------|----------|--|
| 31:14 | RESERVED | Reserved. Read-only, writes have no effect. |
| 13 | PCIDAT | PCI Data Block I/O Power Down Control. Controls the power of the 3 I/O pins in the PCI Data Block. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 12 | EMBK3 | EMIFA Sub-Block 3 I/O Power Down Control. Controls the power of the 8 I/O pins in the EMIFA Sub-Block 3. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 11 | UR0FC | UART0 Flow Control Block I/O Power Down Control. Controls the power of the 2 I/O pins in the UART0 Flow Control Block. 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |
| 10 | UR0DAT | UART0 Data Block I/O Power Down Control. Controls the power of the 2 I/O pins in the UART0 Data Block. 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |
| 9 | TIMER1 | Timer1 Block I/O Power Down Control. Controls the power of the 2 I/O pins in the Timer1 Block. 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |

(1) For more details on I/O pins belonging to each pin mux block, see [Section 3.7, Multiplexed Pin Configurations](#).

Table 3-2. VDD3P3V_PWDN Register Bit Descriptions ⁽¹⁾ (continued)

| BIT | NAME | DESCRIPTION |
|-----|--------|--|
| 8 | TIMER0 | Timer0 Block I/O Power Down Control. Controls the power of the 2 I/O pins in the Timer0 Block. 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |
| 7 | SP | Serial Port Block I/O Power Down Control. Controls the power of the 12 I/O pins in the Serial Port Block (Serial Port Sub-Block 0 and Serial Port Sub-Block 1). 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |
| 6 | PWM1 | PWM1 Block I/O Power Down Control. Controls the power of the 1 I/O pin in the PWM1 Block. 0 = I/O pins powered up. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z) [default]. |
| 5 | GPIO | GPIO Block I/O Power Down Control. Controls the power of the 4 I/O pins in the GPIO Block. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 4 | HOST | Host Block I/O Power Down Control. Controls the power of the 27 I/O pins in the Host Block. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 3 | EMBK2 | EMIFA Sub-Block 2 I/O Power Down Control. Controls the power of the 3 I/O pins in the EMIFA Sub-Block 2. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 2 | EMBK1 | EMIFA Sub-Block 1 I/O Power Down Control. Controls the power of the 29 I/O pins in the EMIFA Sub-Block 1. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 1 | EMBK0 | EMIFA Sub-Block 0 I/O Power Down Control. Controls the power of the 21 I/O pins in the EMIFA Sub-Block 0. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |
| 0 | CLKOUT | CLKOUT Block I/O Power Down Control. Controls the power of the 1 I/O pin in the CLKOUT Block. 0 = I/O pins powered up [default]. 1 = I/O pins powered down and not operational. Outputs are 3-stated (Hi-Z). |

3.3 Clock Considerations

Global device and local peripheral clocks are controlled by the PLL Controllers (PLL1 and PLL2) and the Power and Sleep Controller (PSC).

3.3.1 Clock Configurations after Device Reset

After device reset, the user is responsible for programming the PLL Controllers (PLL1 and PLL2) and the Power and Sleep Controller (PSC) to bring the device up to the desired clock frequency and the desired peripheral clock state (clock gating or not).

For additional power savings, some of the C6424 peripherals support clock gating within the peripheral boundary. For more details on clock gating and power saving features supported by a specific peripheral, see the *TMS320C642x DSP Peripherals Overview Reference Guide* (literature number [SPRUEM3](#)).

3.3.1.1 Device Clock Frequency

The C6424 defaults to PLL bypass mode. To bring the device up to the desired clock frequency, the user should program PLLC1 and PLLC2 after device reset.

C6424 supports a FASTBOOT option, where upon exit from device reset the internal bootloader code automatically programs the PLLC1 into PLL mode with a specific PLL multiplier and divider to speed up device boot. While the FASTBOOT option is beneficial for faster boot, the PLL multiplier and divider selected for boot *may not* be the exact frequency desired for the run-time application. It is the user's responsibility to reconfigure PLLC1 after fastboot to bring the device into the desired clock frequency. [Section 3.4.1, Boot Modes](#), discusses the different fast boot modes in more detail.

The user **must** adhere to the various clock requirements when programming the PLLC1 and PLLC2:

- Fixed frequency ratio requirements between CLKDIV1, CLKDIV3, and CLKDIV6 clock domains. For more details on the frequency ratio requirements, see [Section 6.3.4, C6424 Power and Clock Domains](#).
- PLL multiplier and frequency ranges. For more details on PLL multiplier and frequency ranges, see [Section 6.7.1, PLL1 and PLL2](#).

3.3.1.2 Module Clock State

The clock and reset state for each of the modules is controlled by the Power and Sleep Controller (PSC). [Table 3-3](#) shows the default state of each module after a device-level global reset. The C6424 device has four different module states—Enable, Disable, SyncReset, or SwRstDisable. For more information on the definitions of the module states, the PSC, and PSC programming, see [Section 6.3.5, Power and Sleep Controller \(PSC\)](#) and the *TMS320C642x Power and Sleep Controller (PSC) User's Guide* (literature number [SPRUEN8](#)).

Table 3-3. C6424 Default Module States

| LPSC # | MODULE NAME | DEFAULT MODULE STATE [PSC Register MDSTATn.STATE] |
|--------|------------------------|--|
| 2 | EDMACC | SwRstDisable |
| 3 | EDMATC0 | SwRstDisable |
| 4 | EDMATC1 | SwRstDisable |
| 5 | EDMATC2 | SwRstDisable |
| 6 | EMAC Memory Controller | SwRstDisable |
| 7 | MDIO | SwRstDisable |
| 8 | EMAC | SwRstDisable |
| 9 | McASP0 | SwRstDisable |
| 11 | VLYNQ | SwRstDisable |
| 12 | HPI | SwRstDisable |
| 13 | DDR2 Memory Controller | SwRstDisable |
| 14 | EMIFA | SwRstDisable, if configuration pins AEM[2:0] = 000b Enable, if configuration pins AEM[2:0] = Others [010b and 101b] |
| 15 | PCI | SwRstDisable |
| 16 | McBSP0 | SwRstDisable |
| 17 | McBSP1 | SwRstDisable |
| 18 | I2C | SwRstDisable |
| 19 | UART0 | SwRstDisable |
| 20 | UART1 | SwRstDisable |

Table 3-3. C6424 Default Module States (continued)

| LPSC # | MODULE NAME | DEFAULT MODULE STATE [PSC Register MDSTATn.STATE] |
|--------|-------------|--|
| 23 | PWM0 | SwRstDisable |
| 24 | PWM1 | SwRstDisable |
| 25 | PWM2 | SwRstDisable |
| 26 | GPIO | SwRstDisable |
| 27 | TIMER0 | SwRstDisable |
| 28 | TIMER1 | SwRstDisable |
| 39 | C64x+ CPU | Enable |

3.4 Boot Sequence

The boot sequence is a process by which the device's memory is loaded with program and data sections, and by which some of the device's internal registers are programmed with predetermined values. The boot sequence is started automatically after each device-level global reset. For more details on device-level global resets, see [Section 6.5, Reset](#).

There are several methods by which the memory and register initialization can take place. Each of these methods is referred to as a boot mode. The boot mode to be used is selected at reset. For more information on the bootmode selections, see [Section 3.4.1, Boot Modes](#).

The device is booted through multiple means—primary bootloaders within internal ROM or EMIFA, and secondary user bootloaders from peripherals or external memories. Boot modes, pin configurations, and register configurations required for booting the device, are described in the following subsections.

3.4.1 Boot Modes

The C6424 boot modes are determined by these device boot and configuration pins. For information on how these pins are sampled at device reset, see [Section 6.5.1.2, Latching Boot and Configuration Pins](#).

- BOOTMODE[3:0]
- PCIEN
- FASTBOOT
- PLLMS[2:0]

BOOTMODE[3:0] and PCIEN determine the type of boot (e.g., I2C Boot, EMIFA Boot, HPI Boot, or PCI Boot, etc.). FASTBOOT determines if the PLL is enabled during boot to speed up the boot process.

PLLMS[2:0] is used by bootloader code to determine the PLL multiplier used during fastboot modes (FASTBOOT = 1).

The C6424 boot modes are grouped into two categories—Non-Fastboot Modes and User-Select Multiplier Fastboot Modes.

- **Non-Fastboot Modes (FASTBOOT = 0):** The device operates in default PLL bypass mode during boot. The Non-Fastboot bootmodes available on the C6424 are shown in [Table 3-4](#).
- **User-Select Multiplier Fastboot Modes (FASTBOOT = 1):** The bootloader code speeds up the device during boot. The PLL multiplier is selected by the user via the PLLMS[2:0] pins. The User-Select Multiplier Fastboot bootmodes available on the C6424 are shown in [Table 3-5](#).

All other modes *not* shown in these tables are reserved and invalid settings.

Table 3-4. Non-Fastboot Modes (FASTBOOT = 0)

| DEVICE BOOT AND CONFIGURATION PINS | | BOOT DESCRIPTION ⁽¹⁾ | C6424 DMP (Master/Slave) | PLL1 CLOCK SETTING AT BOOT | | | DSPBOOTADDR (DEFAULT) ⁽¹⁾ |
|------------------------------------|--------|---|--------------------------|----------------------------|----------------------------------|----------------------------|--------------------------------------|
| BOOTMODE[3:0] | PCIE1N | | | PLL MODE ⁽²⁾ | CLKDIV1 DOMAIN (SYSCLK1 DIVIDER) | DEVICE FREQUENCY (SYSCLK1) | |
| 0000 | 0 or 1 | No Boot (Emulation Boot) | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 0001 | 0 or 1 | Reserved | – | – | – | – | – |
| 0010 | 0 | HPI Boot | Slave | Bypass | /1 | CLKIN | 0x0010 0000 |
| | 1 | Reserved | – | – | – | – | – |
| 0011 | 0 or 1 | Reserved | – | – | – | – | – |
| 0100 | 0 or 1 | EMIFA ROM Direct Boot [PLL Bypass Mode] | Master | Bypass | /1 | CLKIN | 0x4200 000 |
| 0101 | 0 or 1 | I2C Boot [STANDARD MODE] ⁽³⁾ | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 0110 | 0 or 1 | 16-bit SPI Boot [McBSP0] | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 0111 | 0 or 1 | NAND Flash Boot | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 1000 | 0 or 1 | UART Boot without Hardware Flow Control [UART0] | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 1001 | 0 or 1 | Reserved | – | – | – | – | – |
| 1010 | 0 or 1 | VLYNQ Boot | Slave | Bypass | /1 | CLKIN | 0x0010 0000 |
| 1011 | 0 or 1 | Reserved | – | – | – | – | – |
| 1100 | 0 or 1 | Reserved | – | – | – | – | – |
| 1101 | 0 or 1 | Reserved | – | – | – | – | – |
| 1110 | 0 or 1 | UART Boot with Hardware Flow Control [UART0] | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 1111 | 0 or 1 | 24-bit SPI Boot (McBSP0 + GP[97]) | Master | Bypass | /1 | CLKIN | 0x0010 0000 |

- (1) For all boot modes that default to DSPBOOTADDR = 0x0010 0000 (i.e., all boot modes except the EMIFA ROM Direct Boot, BOOTMODE[3:0] = 0100, FASTBOOT = 0), the bootloader code disables all C64x+ cache (L2, L1P, and L1D) so that upon exit from the bootloader code, all C64x+ memories are configured as all RAM. If cache use is required, the application code must explicitly enable the cache. For more information on the bootloader, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).
- (2) The PLL MODE for Non-Fastboot Modes is fixed as shown in this table; therefore, the PLLMS[2:0] configuration pins have no effect on the PLL MODE.
- (3) I2C Boot (BOOTMODE[3:0] = 0101b) is *only* available if the MXI/CLKIN frequency is between 21 MHz to 30 MHz. I2C Boot is **not** available for MXI/CLKIN frequencies less than 21 MHz.

Table 3-5. User-Select Multiplier Fastboot Modes (FASTBOOT = 1)

| DEVICE BOOT AND CONFIGURATION PINS | | BOOT DESCRIPTION ⁽¹⁾ | C6424 DMP (Master/Slave) | PLL1 CLOCK SETTING AT BOOT | | | DSPBOOTADDR (DEFAULT) ⁽¹⁾ |
|------------------------------------|--------|---|--------------------------|----------------------------|----------------------------------|----------------------------|--------------------------------------|
| BOOTMODE[3:0] | PCIEN | | | PLL MODE ⁽²⁾ | CLKDIV1 DOMAIN (SYSCLK1 DIVIDER) | DEVICE FREQUENCY (SYSCLK1) | |
| 0000 | 0 or 1 | No Boot (Emulation Boot) | Master | Bypass | /1 | CLKIN | 0x0010 0000 |
| 0001 | 0 | Reserved | – | – | – | – | – |
| | 1 | PCI Boot without Auto Initialization | Slave | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 0010 | 0 | HPI Boot | Slave | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| | 1 | PCI Boot with Auto Initialization | Slave | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 0011 | 0 or 1 | Reserved | – | – | – | – | – |
| 0100 | 0 or 1 | EMIFA ROM FASTBOOT with AIS | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 0101 | 0 or 1 | I2C Boot [I2C MODE] ⁽³⁾ | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 0110 | 0 or 1 | 16-bit SPI Boot [McBSP0] | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 0111 | 0 or 1 | NAND Flash Boot | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 1000 | 0 or 1 | UART Boot without Hardware Flow Control [UART0] | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 1001 | 0 or 1 | EMIFA ROM FASTBOOT without AIS | Master | Table 3-6 | /2 | Table 3-6 | – |
| 1010 | 0 or 1 | VLYNQ Boot | Slave | x20 | /2 | CLKIN x20 / 2 | 0x0010 0000 |
| 1011 | 0 or 1 | Reserved | – | – | – | – | – |
| 1100 | 0 or 1 | Reserved | – | – | – | – | – |
| 1101 | 0 or 1 | Reserved | – | – | – | – | – |
| 1110 | 0 or 1 | UART Boot with Hardware Flow Control [UART0] | Master | Table 3-6 | /2 | Table 3-6 | 0x0010 0000 |
| 1111 | 0 or 1 | 24-bit SPI Boot (McBSP0 + GP[97]) | Master | x20 | /2 | CLKIN x20 / 2 | 0x0010 0000 |

- (1) For all boot modes that default to DSPBOOTADDR = 0x0010 0000, the bootloader code disables all C64x+ cache (L2, L1P, and L1D) so that upon exit from the bootloader code, all C64x+ memories are configured as all RAM. If cache use is required, the application code must explicitly enable the cache. For more information on the bootloader, see the *Using the TMS320C642x Bootloader Application Report* (literature number SPRAAK5).
- (2) Any supported PLL MODE is available. [See Table 3-6 for supported C6424 PLL MODE options].
- (3) I2C Boot (BOOTMODE[3:0] = 0101b) is *only* available if the MXI/CLKIN frequency is between 21 MHz to 30 MHz. I2C Boot is *not* available for MXI/CLKIN frequencies less than 21 MHz.

Table 3-6. PLL Multiplier Selection (PLLMS[2:0]) in User-Select Multiplier Fastboot Modes (FASTBOOT = 1)

| DEVICE BOOT AND CONFIGURATION PINS | PLL1 CLOCK SETTING AT BOOT | | |
|------------------------------------|----------------------------|----------------------------------|----------------------------|
| PLLMS[2:0] | PLL MODE | CLKDIV1 DOMAIN (SYSCLK1 DIVIDER) | DEVICE FREQUENCY (SYSCLK1) |
| 000 | x20 | /2 | CLKIN x20 / 2 |
| 001 | x15 | /2 | CLKIN x15 / 2 |
| 010 | x16 | /2 | CLKIN x16 / 2 |
| 011 | x18 | /2 | CLKIN x18 / 2 |
| 100 | x22 | /2 | CLKIN x22 / 2 |
| 101 | x25 | /2 | CLKIN x25 / 2 |
| 110 | x27 | /2 | CLKIN x27 / 2 |
| 111 | x30 | /2 | CLKIN x30 / 2 |

As shown in [Table 3-4](#) and [Table 3-5](#), at device reset the Boot Controller defaults the DSPBOOTADDR to one of two values based on the boot mode selected. In all boot modes, the C64x+ is immediately released from reset and begins executing from address location indicated in DSPBOOTADDR.

- **Internal Bootloader ROM (0x0010 0000):** For most boot modes, the DSPBOOTADDR defaults to the internal Bootloader ROM so that the DSP can immediately execute the bootloader code in the internal ROM. The bootloader code decodes the captured BOOTMODE, FASTBOOT, PCIEN, and PLLMS information (in the BOOTCFG register) to determine the proper boot operation.

Note: For all boot modes that default to DSPBOOTADDR = 0x0010 0000, the bootloader code disables all C64x+ cache (L2, L1P, and L1D) so that upon exit from the bootloader code, all C64x+ memories are configured as all RAM. If cache use is required, the application code must explicitly enable the cache. For more information on boot modes, see [Section 3.4.1, Boot Modes](#). For more information on the bootloader, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).

- **EMIFA Chip Select Space 2 (0x4200 0000):** The EMIFA ROM Direct Boot in PLL Bypass Mode (BOOTCFG settings BOOTMODE[3:0] = 0100b, FASTBOOT = 0) is the **only** exception where the DSPBOOTADDR defaults to the EMIFA Chip Select Space 2. The DSP begins execution directly from the external ROM at this EMIFA space.

For more information how the bootloader code handles each boot mode, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).

3.4.1.1 FASTBOOT

When C6424 exits pin reset ($\overline{\text{RESET}}$ or $\overline{\text{POR}}$ released), the PLL Controllers (PLL1 and PLL2) default to PLL Bypass Mode. This means the PLLs are disabled, and the MXI/CLKIN clock input is driving the chip. All the clock domain divider ratios discussed in [Section 6.3.4, C6424 Power and Clock Domains](#), still apply. For example, assume an MXI/CLKIN frequency of 25 MHz—meaning the internal clock source for EMIFA is at CLKDIV3 domain = 25 MHz/3 = 8.3 MHz, a very slow clock. In addition, the EMIFA registers are reset to the slowest configuration which translates to very slow peripheral operation/boot.

To optimize boot time, the user should reprogram clock settings via the PLLC as early as possible during the boot process. The FASTBOOT pin facilitates this operation by allowing the device to boot at a faster clock rate.

Except for the EMIFA ROM Direct Boot in PLL Bypass Mode (BOOTCFG settings BOOTMODE[3:0] = 0100b, FASTBOOT = 0), all other boot modes default to executing from the Internal Bootloader ROM. The first action that the bootloader code takes is to decode the boot mode. If the FASTBOOT option is selected (BOOTCFG.FASTBOOT = 1), the bootloader software begins by programming the PLLC1 (System PLLC) to PLL Mode to give the device a slightly faster operation before fetching code from external devices. The exact PLL multiplier that the bootloader uses is determined by the PLLMS[2:0] settings, as shown in [Table 3-5](#) and [Table 3-6](#).

Some boot modes *must* be accompanied with FASTBOOT = 1 so that the corresponding peripheral can run at a reasonable rate to communicate to the external device(s). This includes PCI boot.

Note: PLLC2 still stays in PLL Bypass Mode, the bootloader *does not* reconfigure it.

3.4.1.2 Selecting FASTBOOT PLL Multiplier

[Table 3-5](#) and [Table 3-6](#) show the PLL multipliers used by the bootloader code during fastboot (FASTBOOT = 1) and the resulting device frequency. The user is responsible for selecting the bootmode with the appropriate PLL multiplier for their MXI/CLKIN clock source so that the device speed and PLL frequency range requirements are met. For the PLLC1 Clock Frequency Ranges, see [Table 6-15, PLLC1 Clock Frequency Ranges](#) in [Section 6.7.1, PLL1 and PLL2](#).

The following are guidelines for PLL output frequency and device speed (frequency):

- **PLL Output Frequency:** (PLLOUT = CLKIN frequency * boot PLL Multiplier) **must** stay within the PLLOUT frequency range in [Table 6-15](#), *PLL_{C1} Clock Frequency Ranges*.
- **Device Frequency:** (SYSCLK1) calculated from [Table 3-5](#) **must not** exceed the SYSCLK1 maximum frequency in [Table 6-15](#), *PLL_{C1} Clock Frequency Ranges*.

For example, for a 600-MHz device with a CLKIN = 25 MHz, in order to stay within the PLLOUT frequency range and SYSCLK1 maximum frequency from [Table 6-15](#), *PLL_{C1} Clock Frequency Ranges*, the user **must** select a boot mode with a PLL1 multiplier between x16 and x24.

3.4.1.3 EMIFA Boot Modes

As shown in [Table 3-4](#) and [Table 3-5](#), there are different types of EMIFA Boot Modes. This subsection summarizes these types of EMIFA boot modes. For further detailed information, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).

- **EMIFA ROM Direct Boot in PLL Bypass Mode** (FASTBOOT = 0, BOOTMODE[3:0] = 0100b)
 - The C64x+ fetches the code directly from EMIFA Chip Select 2 Space [$\overline{\text{EM_CS2}}$] (address 0x42000000)
 - The PLL is in Bypass Mode
 - EMIFA is configured as Asynchronous EMIF. The user is responsible for ensuring the desirable Asynchronous EMIF pins are available through configuration pins AEM[2:0]. AEM[2:0] **must** be configured to 010b [EMIFA (Async) Pinout Mode 2].
- **EMIFA ROM Fastboot with AIS** (FASTBOOT = 1, BOOTMODE[3:0] = 0100b)
 - The C64x+ begins execution from the internal bootloader ROM at address 0x00100000.
 - The bootloader code programs PLLC1 to PLL Mode to speed up the boot process. The PLL multiplier value is determined by the PLLMS[2:0] configuration as shown in [Table 3-5](#).
 - The bootloader code reads code from the EMIFA $\overline{\text{EM_CS2}}$ space using the application image script (AIS) format.
 - EMIFA is configured as Asynchronous EMIF. The user is responsible for ensuring the desirable Asynchronous EMIF pins are available through configuration pins AEM[2:0]. AEM[2:0] **must** be configured to 010b [EMIFA (Async) Pinout Mode 2].
- **EMIFA ROM Fastboot without AIS:** (FASTBOOT = 1, BOOTMODE[3:0] = 1001b)
 - The C64x+ begins execution from the internal bootloader ROM at address 0x00100000.
 - The bootloader code programs PLLC1 to PLL Mode to speed up the boot process. The PLL multiplier value is determined by the PLLMS[2:0] configuration as shown in [Table 3-5](#).
 - The bootloader code then jumps to the EMIFA $\overline{\text{EM_CS2}}$ space, at which point the C64x+ fetches the code directly from address 0x42000000.
 - EMIFA is configured as Asynchronous EMIF. The user is responsible for ensuring the desirable Asynchronous EMIF pins are available through configuration pins AEM[2:0]. AEM[2:0] **must** be configured to 010b [EMIFA (Async) Pinout Mode 2].
- **NAND Flash Boot:** (FASTBOOT = 0 or 1, BOOTMODE[3:0] = 0111b)
 - The C64x+ begins execution from the internal bootloader ROM at address 0x00100000.
 - Depending on the FASTBOOT and PLLMS[2:0] settings, the bootloader code may program the PLLC1 to PLL Mode to speed up the boot process. See [Table 3-4](#) and [Table 3-5](#).
 - The bootloader code reads the code from EMIFA (NAND) $\overline{\text{EM_CS2}}$ (address 0x42000000) using AIS format.
 - EMIFA is configured in NAND mode. The user is responsible for ensuring the desirable Asynchronous EMIF pins are available through configuration pins AEM[2:0]. AEM[2:0] **can** be configured to 010b [EMIFA (Async) Pinout Mode 2] or 101b [EMIFA (NAND) Pinout Mode 5].

3.4.1.4 Serial Boot Modes (I2C, UART[UART0], SPI[McBSP0])

This subsection discusses how the bootloader configures the clock dividers for the serial boot modes—I2C boot, UART boot, and SPI boot.

3.4.1.4.1 I2C Boot

If FASTBOOT = 0, then I2C Boot (BOOTMODE = 0101) is performed in Standard-Mode (up-to 100 kbps). If FASTBOOT = 1, then I2C Boot is performed in Fast-Mode (up-to 400 kbps). The actual I2C data transfer rate is dependent on the MXI/CLKIN frequency.

This is how the bootloader programs the I2C:

- I2C Boot in Fast-Mode (BOOTMODE[3:0] = 0101b, FASTBOOT = 1)
 - I2C register settings: ICPSC.IPSC = 2₁₀, ICCLKL.ICCL = 8₁₀, ICCKH.ICCH = 8₁₀
 - Resulting in the following I2C prescaled module clock frequency (internal I2C clock):
 - (CLKIN frequency in MHz) / 3
 - Resulting in the following I2C serial clock (SCL):
 - SCL frequency (in kHz) = (CLKIN frequency in MHz) / 78 * 1000
 - SCL low pulse duration (in μs) = 39 / (CLKIN frequency in MHz)
 - SCL high pulse duration (in μs) = 39 / (CLKIN frequency in MHz)
- I2C Boot in Standard-Mode (BOOTMODE[3:0] = 0101b, FASTBOOT = 0)
 - I2C register settings: ICPSC.IPSC = 2₁₀, ICCLKL.ICCL = 45₁₀, ICCKH.ICCH = 45₁₀
 - Resulting in the following I2C prescaled module clock frequency (internal I2C clock):
 - (CLKIN frequency in MHz) / 3
 - Resulting in the following I2C serial clock (SCL):
 - SCL frequency (in kHz) = (CLKIN frequency in MHz) / 300 * 1000
 - SCL low pulse duration (in μs) = 150 / (CLKIN frequency in MHz)
 - SCL high pulse duration (in μs) = 150 / (CLKIN frequency in MHz)

Note: the I2C peripheral requires that the prescaled module clock frequency *must* be between 7 to 12 MHz. Therefore, the I2C boot is *only* available for MXI/CLKIN frequency between 21 MHz and 30 MHz.

For more details on the I2C peripheral configurations and clock requirements, see the *TMS320C642x DSP Inter-Integrated Circuit (I2C) Peripheral User's Guide* (literature number [SPRUEN0](#)).

3.4.1.4.2 UART Boot

For UART Boot (BOOTMODE[3:0] = 1000b or 1110b), the bootloader programs the UART0 peripheral as follows:

- UART0 divisor is set to 15₁₀
- Resulting in this UART0 baud rate in kilobit per second (kbps):
 - (CLKIN frequency in MHz) * 1000 / (15 * 16)

The user is responsible for ensuring the resulting baud rate is appropriate for the system. The UART0 divisor (/15) is optimized for CLKIN frequency between 27 to 29 MHz to stay within 5% of the 115200-bps baud rate.

For more details on the UART peripheral configurations and clock generation, see the *TMS320C642x DSP Universal Asynchronous Receiver/Transmitter (UART) User's Guide* (literature number [SPRUEN6](#)).

3.4.1.4.3 SPI Boot

Both 16-bit address SPI Boot (BOOTMODE = 0110) and 24-bit address SPI boot are performed through the McBSP0 peripheral. The bootloader programs the McBSP0 peripheral as follows:

- McBSP0 register settings: SRGR.CLKGDV = 2₁₀
- Resulting in this SPI serial clock frequency:
 - (SYSCLK3 frequency in MHz) / 3

SYSCLK3 frequency = SYSCLK1 frequency / 6. SYSCLK1 frequency during boot can be found in [Table 3-4](#), [Table 3-5](#), and/or [Table 3-6](#) based on the boot mode selection.

For example, if BOOTMODE[3:0] = 0110b, FASTBOOT = 1, the MXI/CLKIN frequency = 30 MHz, PLLMS[2:0] = 100b, the combination of Table 3-5 and Table 3-6 indicates that the device frequency (SYSCLK1) is CLKIN x 22 / 2 = 330 MHz. This means SYSCLK3 frequency is 330 / 6 = 55 MHz, resulting in SPI serial clock frequency of 55 / 3 = 18.3 MHz.

3.4.1.5 Host Boot Modes

The C6424 supports two types of host boots—PCI Boot or HPI Boot.

The PCI Boot (BOOTMODE[3:0] = 0001b or 0010b, PCIEN = 1) is *only* available in fastboot (FASTBOOT = 1), as shown in Table 3-5.

The HPI Boot is available in fastboot and non-fastboot, as shown in Table 3-4 and Table 3-5.

Note: The HPI $\overline{\text{HSTROBE}}$ inactive pulse duration timing requirement [$t_{w(\text{HSTBH})}$] is dependent on the HPI internal clock source (SYSCLK3) frequency (see Section 6.12.3, *HPI Electrical Data/Timing*). The external host *must* be aware of the SYSCLK3 frequency during boot to ensure the $\overline{\text{HSTROBE}}$ pulse duration timing requirement is met.

3.4.2 Bootmode Registers

3.4.2.1 BOOTCFG Register

The Device Bootmode (see Section 3.4.1, *Boot Modes*) and Configuration pins (see Section 3.5.1, *Device and Peripheral Configurations at Device Reset*) latched at reset are captured in the Device Boot Configuration (BOOTCFG) register which is accessible through the System Module. This is a *read-only* register. The bits show the values latched from the corresponding configuration pins sampled at device reset. For more information on how these pins are sampled at device reset, see Section 6.5.1.2, *Latching Boot and Configuration Pins*. For the corresponding device boot and configuration pins, see Table 2-7, *BOOT Terminal Functions*.

| | | | | | | | | | | | | | | | | |
|------------------|----|-------|----|-----|-------|---|----------|---|---|----------|----------|-----|---------|------|----|----|
| 31 | | | | | | | | | | 21 | | 20 | 19 | 18 | 17 | 16 |
| RESERVED | | | | | | | | | | LENDIAN | FASTBOOT | RSV | DPCIE N | 8_16 | | |
| R-0000 0000 0001 | | | | | | | | | | R-L | R-L | R-0 | R-L | R-L | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| RSV | | PLLMS | | RSV | DAEM | | RESERVED | | | BOOTMODE | | | | | | |
| R-0 | | R-LLL | | R-0 | R-LLL | | R-0000 | | | R-LLLL | | | | | | |

LEGEND: R = Read only; L = pin state latched at reset rising edge; -n = value after reset

Figure 3-2. BOOTCFG Register—0x01C4 0014

Table 3-7. BOOTCFG Register Description

| Bit | Field Name | Description |
|-------|------------|---|
| 31:21 | RESERVED | Reserved. Writes have no effect. |
| 20 | LENDIAN | Little Endian Selection (see Section 3.5.1.4, <i>Endianess Selection (LENDIAN)</i>) This field determines the device endian mode. 0 = Device is Big Endian 1 = Device is Little Endian The default value is latched from LENDIAN configuration pin. |

Table 3-7. BOOTCFG Register Description (continued)

| Bit | Field Name | Description |
|-------|------------|--|
| 19 | FASTBOOT | <p>Fastboot (see Section 3.4.1.1, FASTBOOT)</p> <p>This field is used by the device bootloader code to determine if it needs to speed up the device to PLL mode before booting.</p> <p>0 = No Fastboot 1 = Fastboot</p> <p>The default value is latched from FASTBOOT configuration pin.</p> |
| 18 | RSV | Reserved. Writes have no effect. |
| 17 | DPCIEN | <p>PINMUX1.PCIEN Default (see Section 3.5.1.5, PCI Enable) For more details on the PCIEN settings, see Section 3.7.2.2, PINMUX1 Register Description.</p> <p>This field affects the pin mux control by setting the default of PINMUX1.PCIEN. This field determines if the internal pullup/pulldown resistors on the PCI capable pins are enabled/disabled. This field <i>does not</i> affect PCI register setting.</p> <p>The user must keep the value on the PCIEN pin constant throughout the operation.</p> <p>The default value is from the PCIEN configuration pin.</p> |
| 16 | 8_16 | <p>EMIFA CS2 Bus Width Default Configuration (see Section 3.5.1.1, EMIFA CS2 Bus Width (8_16))</p> <p>0 = EMIFA CS2 space defaults to 8-bit data bus width 1 = EMIFA CS2 space defaults to 16-bit bus width</p> <p><i>This field does not affect pin mux control. This field affects the EMIFA register setting—the default of EMIFA register field A1CR.ASIZE is set to this 8_16 value.</i></p> <p>The default value is latched from the 8_16 configuration pin.</p> |
| 15 | RSV | Reserved. Writes have no effect. |
| 14:12 | PLLMS | <p>Fastboot PLL Multiplier Select [PLLMS] (see Section 3.5.1.3, Fast Boot PLL Multiplier Select [PLLMS])</p> <p>If FASTBOOT = 1, this field selects the FASTBOOT PLL Multiplier according to Table 3-6.</p> <p>The default value is latched from the PLLMS[2:0] configuration pins.</p> |
| 11 | RSV | Reserved. Writes have no effect. |
| 10:8 | DAEM | <p>PINMUX0.AEM default [DAEM] (see Section 3.5.1.2, EMIFA Pinout Mode (AEM[2:0]))</p> <p>For more details on the AEM settings, see Section 3.7.2.1, PINMUX0 Register Description.</p> <p>This field affects pin mux control by setting the default of PINMUX0.AEM. This field <i>does not</i> affect EMIFA Register settings.</p> <p>The default value is latched from the AEM[2:0] configuration pins.</p> |
| 7:4 | RESERVED | Reserved. Writes have no effect. |
| 3:0 | BOOTMODE | <p>Boot Mode (see Section 3.4.1, Boot Modes)</p> <p>This field is used in conjunction with FASTBOOT, PCIEN, and PLLMS to determine the device boot mode.</p> <p>The default value is latched from the BOOTMODE[3:0] configuration pins.</p> |

3.4.2.2 BOOTCMPLT Register

If the bootloader code detects an error during boot, it records the error status in the Boot Complete (BOOTCMPLT) register.

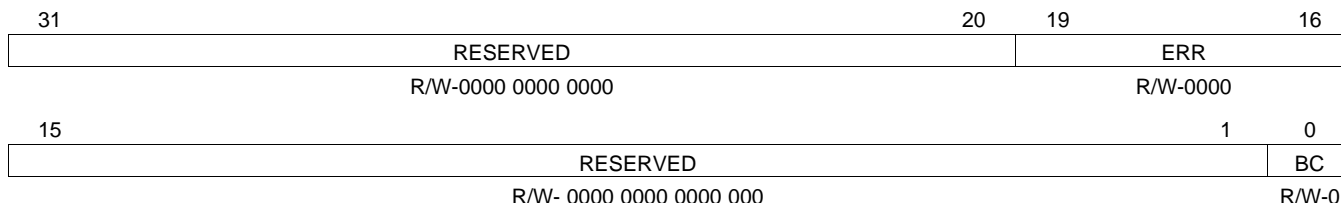
In addition, the BOOTCMPLT register is used for communication between the external host and the bootloader code during a Host Boot (PCI Boot or HPI boot). Once the external host has completed boot, it **must** perform the following communication with the bootloader code:

- Write the desired 32-bit CPU starting address in the DSPBOOTADDR register (see [Section 3.4.2.3, DSPBOOTADDR Register](#)).

- Write a '1' to the Boot Complete (BC) bit field in the BOOTCMPLT register to indicate that the host has completed booting this device.

Once the bootloader code detects BC = 1, it directs the CPU to begin executing from the DSPBOOTADDR register.

The BOOTCMPLT register is reset by any device-level global reset. For the list of device-level global resets, see [Section 6.5, Reset](#).



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

Figure 3-3. BOOTCMPLT Register— 0x01C4 000C

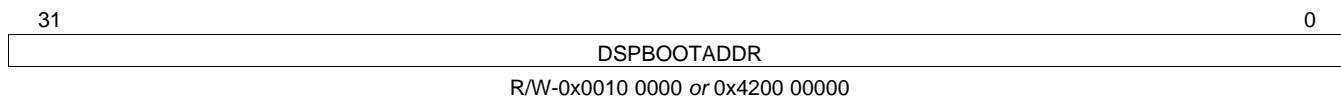
Table 3-8. BOOTCMPLT Register Description

| Bit | Field Name | Description |
|-------|------------|---|
| 31:20 | RESERVED | Reserved. For proper device operation, the user should <i>only</i> write "0" to these bits. |
| 19:16 | ERR | Boot Error 0000 = No Error (default). 0001 - 1111 = bootloader software detected a boot error and aborted the boot. For the error codes, see the <i>Using the TMS320C642x DSP Bootloader Application Report</i> (literature number SPRAAK5). |
| 15:1 | RESERVED | Reserved. For proper device operation, the user should <i>only</i> write "0" to these bits. |
| 0 | BC | Boot Complete Flag from Host This field is <i>only</i> applicable to Host Boots. 0 = Host <i>has not</i> completed booting this device (default). 1 = Host <i>has</i> completed booting this device. DSP can begin executing from the DSPBOOTADDR register value. |

3.4.2.3 DSPBOOTADDR Register

The DSP Boot Address (DSPBOOTADDR) register contains the starting address for the C64x+ CPU. Whenever the C64x+ is released from reset, it begins executing from the location pointed to by DSPBOOTADDR register. For Host boots (HPI Boot or PCI Boot), the DSPBOOTADDR register is also used for communication between the Host and the bootloader code during boot.

The DSPBOOTADDR register is reset by any device-level global reset. For the list of device-level global resets, see [Section 6.5, Reset](#).



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

Figure 3-4. DSPBOOTADDR Register— 0x01C4 0008

Table 3-9. DSPBOOTADDR Register Description

| Bit | Field Name | Description |
|------|-------------|--|
| 31:0 | DSPBOOTADDR | <p>DSP Boot Address</p> <p>After boot, the C64x+ CPU begins execution from this 32-bit address location 0x00100000 (for Internal Bootloader ROM).</p> <p>or</p> <p>0x42000000 (for EMIFA $\overline{CS2}$ Space).</p> <p>The lower 10 bits (bits 9:0) should <i>always</i> be programmed to "0" as they are ignored by the C64x+.</p> <p>Default depends on boot mode selected.</p> <p>See Table 3-4, <i>Non-Fastboot Modes</i> and Table 3-5, <i>User-Select Multiplier Fastboot Modes</i>.</p> |

At device reset, the Boot Controller defaults DSPBOOTADDR to one of two values (either Internal Bootloader ROM at address 0x00100000 or EMIFA $\overline{CS2}$ Space 0x42000000) based on the boot mode selected (for the boot mode selections, see [Table 3-4](#) and [Table 3-5](#)).

For Non-Host Boot Modes, software can leave the DSPBOOTADDR register at default.

For Host Boots (HPI Boot or PCI Boot), the DSPBOOTADDR register is also used for communication between the Host and the bootloader code during boot. For Host Boots, the DSPBOOTADDR register defaults to Internal Bootloader ROM, and the C64x+ CPU is immediately released from reset so that it can begin executing the bootloader code in this internal ROM. The bootloader code waits for the Host to boot the device. Once the Host is done booting the device, it **must** write a new starting address into the DSPBOOTADDR register, and follow with writing BOOTCMPLT.BC = 1 to indicate the boot is complete. As soon as the bootloader code detects BOOTCMPLT.BC = 1, it instructs the CPU to jump to this new DSPBOOTADDR address. At this point, the CPU continues the rest of the code execution starting from the new DSPBOOTADDR location and the boot is completed.

3.5 Configurations At Reset

Some device configurations are determined at reset. The following subsections give more details.

3.5.1 Device and Peripheral Configurations at Device Reset

[Table 2-7](#), BOOT Terminal Functions, lists the device boot and configuration pins that are latched at device reset for configuring basic device settings for proper device operation. [Table 3-10](#) summarizes the device boot and configuration pins, and the device functions that they affect.

Table 3-10. Default Functions Affected by Device Boot and Configuration Pins

| DEVICE BOOT AND CONFIGURATION PINS | BOOT SELECTED | PIN MUX CONTROL | GLOBAL SETTING | PERIPHERAL SETTING |
|------------------------------------|---|--|---|--|
| BOOTMODE[3:0] | Boot Mode | PINMUX0/PINMUX1 Registers: Based on BOOTMODE[3:0], the bootloader code programs PINMUX0 and PINMUX1 registers to select the appropriate pin functions required for boot. | I/O Pin Power: Based on BOOTMODE[3:0], the bootloader code programs VDD3P3V_PWDN register to power up the I/O pins required for boot. | PSC/Peripherals: Based on BOOTMODE[3:0], the bootloader code programs the PSC to put boot-related peripheral(s) in the Enable State, and programs the peripheral(s) for boot operation. |
| FASTBOOT | Fastboot | – | Sets Device Frequency: Based on BOOTMODE, FASTBOOT, and PLLMS; the bootloader code programs PLLC1. | – |
| PLLMS[2:0] | If FASTBOOT = 1, the PLLMS[2:0] selects the FASTBOOT PLL Multiplier. | – | Sets Device Frequency: Based on BOOTMODE, FASTBOOT, and PLLMS; the bootloader code programs PLLC1. | – |
| AEM[2:0] | – | PINMUX0.AEM: Sets the default of this field to control the EMIFA Pinout Mode. Affects the pin muxing in EMIFA Sub-Block 0, 1, and 3. | – | PSC/EMIFA: The EMIFA module state defaults to SwRstDisable if AEM = 0; otherwise, the EMIFA module state defaults to Enable. |
| 8_16 | EMIFA Boot: 8_16 determines the EMIFA CS ² Data Bus Width. | – | – | EMIFA: Sets default of register field A1CR.ASIZE to determine the default data bus width for EMIFA CS ² space. |
| PCIEN ⁽¹⁾ | Host Boot: PCIEN selects the type of Host Boot (HPI Boot or PCI Boot) | PINMUX1.PCIEN: sets this field to control the PCI pin muxing in Host Block, PCI Data Block, GPIO Block, EMIFA Sub-Block 0 and Sub-Block 3. (1) (2) | – | PSC/Peripheral (Applicable to Host Boot only): Based on the Host Boot type (PCI or HPI), the bootloader code programs the PSC to put the corresponding peripheral in the Enable State, and programs the peripheral for boot operation. |
| LENDIAN | – | – | Device endianness | – |

(1) Software can modify all PINMUX0 and PINMUX1 bit fields from their defaults, **except** for PINMUX1.PCIEN.

(2) In addition to pin mux control, PCIEN also affects the internal pullup/down resistors of the PCI capable pins. When PCIEN = 0, internal pullup/down resistors on the PCI capable pins are enabled. When PCIEN = 1, internal pullup/down resistors on the PCI capable pins are disabled to be compliant to the *PCI Local Bus Specification Revision 2.3*.

For proper device operation, external pullup/pulldown resistors may be required on these device boot and configuration pins. For discussion situations where external pullup/pulldown resistors are required, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

Note: Except for PCIEN, all other C6424 configuration inputs (BOOTMODE[3:0], FASTBOOT, PLLMS[2:0], AEM[2:0], 8_16, and LENDIAN) are multiplexed with other functional pins. These pins function as device boot and configuration pins only during device reset. The user **must** take care of any potential data contention in the system. To help avoid system data contention, the C6424 puts these configuration pins into a high-impedance state (Hi-Z) when device reset ($\overline{\text{RESET}}$ or $\overline{\text{POR}}$) is asserted, and continues to hold them in a high-impedance state until the internal global reset is removed; at which point, the default peripheral (either GPIO or EMIFA based on default of AEM[2:0]) will now control these pins.

All of the device boot and configuration pin settings are captured in the corresponding bit fields in the BOOTCFG register (see [Section 3.4.2.1](#), *BOOTCFG Register*).

The following subsections provide more details on the device configurations determined at device reset: AEM, PLLMS, PCIEN, 8_16, and LENDIAN.

3.5.1.1 EMIFA CS2 Bus Width (8_16)

The default width of the EMIFA chip select space 2 ($\overline{\text{EM_CS2}}$) is determined by the 8_16 input captured in BOOTCFG.8_16. If 8_16 = 0, the $\overline{\text{EM_CS2}}$ space data bus defaults to 8-bits wide. If 8_16 = 1, it defaults to 16-bits wide. The C6424 sets the default of the EMIFA register A1CR.ASIZE based on the 8_16 input.

For EMIFA boot modes ([Section 3.4.1.3](#), *EMIFA Boot Modes*), the user must set 8_16 appropriately to set the default bus width for boot.

Note: 8_16 only selects the default bus width for EMIFA $\overline{\text{CS2}}$ space. The $\overline{\text{CS2}}$ bus width may be changed at any time by software through programming the EMIFA register A1CR.ASIZE field. All other chip select spaces default to 8-bits wide and must be modified using the corresponding EMIFA control register if 16-bit operation is desired.

Note: 8_16 does not affect pin mux control. It is the user's responsibility to configure AEM[2:0] appropriately so that the desired EMIFA data pins are selected.

3.5.1.2 EMIFA Pinout Mode (AEM[2:0])

To support different usage scenarios, the C6424 provides intricate pin multiplexing between the EMIFA and other peripherals. The PINMUX0.AEM register bit field in the System Module determines the EMIFA Pinout Mode. The AEM[2:0] pins only select the default EMIFA Pinout Mode. It is latched at device reset de-assertion (high) into the BOOTCFG.DAEM bit field. The AEM[2:0] value also sets the default of the PINMUX0.AEM bit field. While the BOOTCFG.DAEM bit field shows the actual latched value and *cannot* be modified, the PINMUX0.AEM value can be changed by software to modify the EMIFA Pinout Mode.

Note: The AEM[2:0] value *does not* affect the operation of the EMIFA module itself. It *only* affects which EMIFA pins are brought out to the device pins. For more details on the AEM settings, see [Section 3.7](#), *Multiplexed Pin Configurations*.

3.5.1.3 FASTBOOT PLL Multiplier Select (PLLMS)

If FASTBOOT = 1, the PLLMS[2:0] pins select PLL multiplier for Fastboot modes. If FASTBOOT = 0, the PLLMS[2:0] pins are ignored.

The PLLMS[2:0] pin values are latched at device reset de-assertion into the BOOTCFG.PLLMS field and cannot be modified by software. This value is only applicable during fast boot.

For more information on boot modes and the FASTBOOT PLL multiplier selection, see [Section 3.4.1](#), *Boot Modes*.

3.5.1.4 Endianness Selection (LENDIAN)

The LENDIAN configuration pin latched at reset determines the endianness setting of the device. If LENDIAN = 1, little endian is selected. If LENDIAN = 0, big endian is selected.

The setting is latched and stored in the BOOTCFG.LENDIAN field and cannot be modified by software.

3.5.1.5 PCI Enable (PCIEN)

The PCIEN configuration pin determines if the PCI peripheral is used on this device. If PCIEN = 1 indicating the PCI *is* used, then the PCI multiplexed pins default to PCI functions, and the pins' corresponding internal pullup/pulldown resistors are disabled. If PCIEN = 0 indicating the PCI *is not* used, then the PCI muxed pins default to non-PCI functions, and the pins' corresponding internal pullup/pulldown resistors are enabled.

The PCIEN setting is captured and stored in the BOOTCFG.DPCIEN bit field, and also in the PINMUX1.PCIEN bit field. These values *cannot* be changed by software. Furthermore, for proper device operation, the user **must** hold the desired setting at the PCIEN pin throughout device operation.

3.6 Configurations After Reset

The following sections provide details on configuring the device after reset.

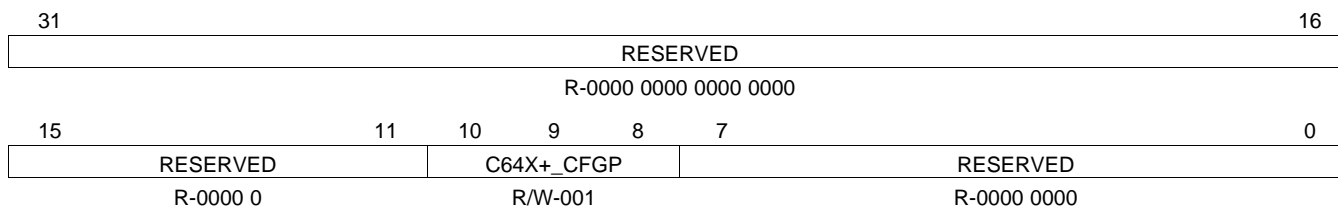
Multiplexed pins are configured both at and after reset. [Section 3.5.1, Device and Peripheral Configurations at Device Reset](#), discusses multiplexed pin control at reset. For more details on multiplexed pins control after reset, see [Section 3.7, Multiplexed Pin Configurations](#).

3.6.1 Switch Central Resource (SCR) Bus Priorities

Prioritization within the Switched Central Resource (SCR) is programmable for each master. The register bit fields and default priority levels for C6424 bus masters are shown in [Table 3-11, C6424 Default Bus Master Priorities](#). The priority levels should be tuned to obtain the best system performance for a particular application. Lower values indicate higher priority. For most masters, their priority values are programmed at the system level by configuring the MSTPRI0 and MSTPRI1 registers. Details on the MSTPRI0/1 registers are shown in [Figure 3-5](#) and [Figure 3-6](#). The C64x+ and EDMA masters contain registers that control their own priority values.

Table 3-11. C6424 Default Bus Master Priorities

| Priority Bit Field | Bus Master | Default Priority Level |
|--------------------|-------------|-------------------------------|
| EDMATC0P | EDMATC0 | 0 (EDMACC QUEPRI Register) |
| EDMATC1P | EDMATC1 | 0 (EDMACC QUEPRI Register) |
| EDMATC2P | EDMATC2 | 0 (EDMACC QUEPRI Register) |
| C64X+_DMAP | C64X+ (DMA) | 7 (C64x + MDMAARBE.PRI field) |
| C64X+_CFGP | C64X+ (CFG) | 1 (MSTPRI0 Register) |
| EMACP | EMAC | 4 (MSTPRI1 Register) |
| VLYNQP | VLYNQ | 4 (MSTPRI1 Register) |
| HPIP | HPI | 4 (MSTPRI1 Register) |
| PCIP | PCI | 4 (MSTPRI1 Register) |



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

Figure 3-5. MSTPRI0 Register— 0x01C4 003C

Table 3-12. MSTPRI0 Register Description

| Bit | Field Name | Description |
|-------|------------|---|
| 31:11 | RESERVED | Reserved. Read-only, writes have no effect. |

Table 3-12. MSTPRI0 Register Description (continued)

| Bit | Field Name | Description |
|------|------------|---|
| 10:8 | C64X+_CFG | C64X+_CFG master port priority in System Infrastructure. 000 = Priority 0 (Highest) 100 = Priority 4 001 = Priority 1 101 = Priority 5 010 = Priority 2 110 = Priority 6 011 = Priority 3 111 = Priority 7 (Lowest) |
| 7:0 | RESERVED | Reserved. Read-only, writes have no effect. |

| | | | | | | | | | | | | |
|---------------------|----------|---------|----|-----|---------|----|-----|---------|----|---------|----|----|
| 31 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| RESERVED | | PCIP | | RSV | HPIP | | RSV | VLYNQ | | | | |
| R-0000 0 | | R/W-100 | | R-0 | R/W-100 | | R-0 | R/W-100 | | | | |
| 15 | RESERVED | | | | | | | | 3 | 2 | 1 | 0 |
| | | | | | | | | | | EMACP | | |
| R- 0000 0000 0000 0 | | | | | | | | | | R/W-100 | | |

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

Figure 3-6. MSTPRI1 Register— 0x01C4 0040**Table 3-13. MSTPRI1 Register Description**

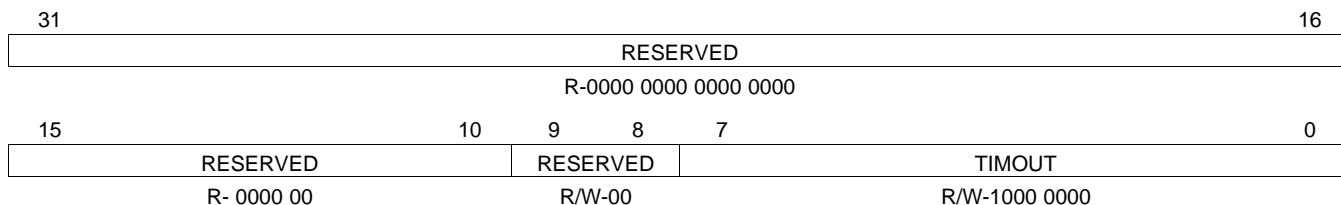
| Bit | Field Name | Description |
|-------|------------|---|
| 31:27 | RESERVED | Reserved. Read-only, writes have no effect. |
| 26:24 | PCIP | PCI master port priority in System Infrastructure. 000 = Priority 0 (Highest) 100 = Priority 4 001 = Priority 1 101 = Priority 5 010 = Priority 2 110 = Priority 6 011 = Priority 3 111 = Priority 7 (Lowest) |
| 23 | RSV | Reserved. Read-only, writes have no effect. |
| 22:20 | HPIP | HPI master port priority in System Infrastructure. 000 = Priority 0 (Highest) 100 = Priority 4 001 = Priority 1 101 = Priority 5 010 = Priority 2 110 = Priority 6 011 = Priority 3 111 = Priority 7 (Lowest) |
| 19 | RSV | Reserved. Read-only, writes have no effect. |
| 18:16 | VLYNQ | VLYNQ master port priority in System Infrastructure. 000 = Priority 0 (Highest) 100 = Priority 4 001 = Priority 1 101 = Priority 5 010 = Priority 2 110 = Priority 6 011 = Priority 3 111 = Priority 7 (Lowest) |
| 15:3 | RESERVED | Reserved. Read-only, writes have no effect. |
| 2:0 | EMACP | EMACP master port priority in System Infrastructure. 000 = Priority 0 (Highest) 100 = Priority 4 001 = Priority 1 101 = Priority 5 010 = Priority 2 110 = Priority 6 011 = Priority 3 111 = Priority 7 (Lowest) |

3.6.2 Peripheral Selection After Device Reset

After device reset, most peripheral configurations are done within the peripheral's registers. This section discusses some additional peripheral controls in the System Module. For information on multiplexed pin controls that determine what peripheral pins are brought out to the pins, see [Section 3.7, Multiplexed Pin Configurations](#).

3.6.2.1 HPI Control Register (HPICTL)

The HPI Control (HPICTL) register determines the Host Burst Write Time-Out value. **The user should only modify this register once during device initialization. When modifying this register, the user must ensure the HPI FIFOs are empty and there are no on-going HPI transactions.**



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

Figure 3-7. HPICTL Register— 0x01C4 0030

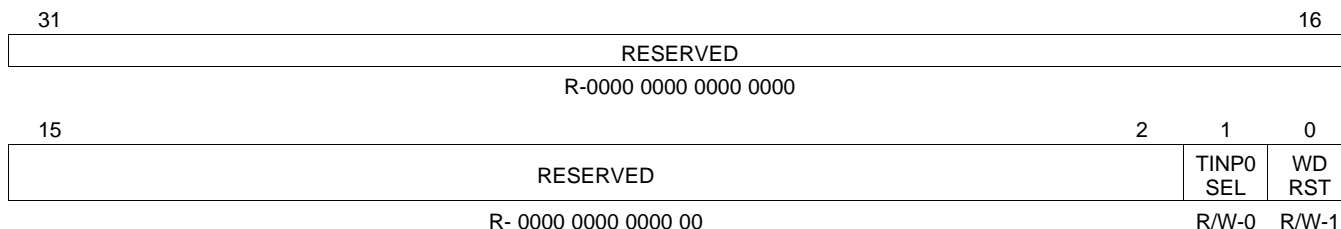
Table 3-14. HPICTL Register Description

| Bit | Field Name | Description |
|-------|------------|---|
| 31:10 | RESERVED | Reserved. Read-only, writes have no effect. |
| 9:8 | RESERVED | Reserved. For proper device operation, the user should <i>only</i> write "0" to these bits (default). |
| 7:0 | TIMOUT | Host Burst Write Timeout Value When the HPI time-out counter reaches the value programmed here, the HPI write FIFO content is flushed. For more details on the time-out counter and its use in write bursting, see the <i>TMS320C642x DSP Host Port Interface (HPI) User's Guide</i> (literature number SPRUEM9). |

3.6.2.2 Timer Control Register (TIMERCTL)

The Timer Control Register (TIMERCTL) provides additional control for Timer0 and Timer2. **The user should only modify this register once during device initialization, when the corresponding Timer is not in use.**

- Timer 2 Control: The TIMERCTL.WDRST bit determines if the WatchDog timer event (Timer 2) can cause a device max reset. For more details on the description of a maximum reset, see [Section 6.5.3, Maximum Reset](#).
- Timer 0 Control: The TINP0SEL bit selects the clock source connected to Timer0's TIN0 input.



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

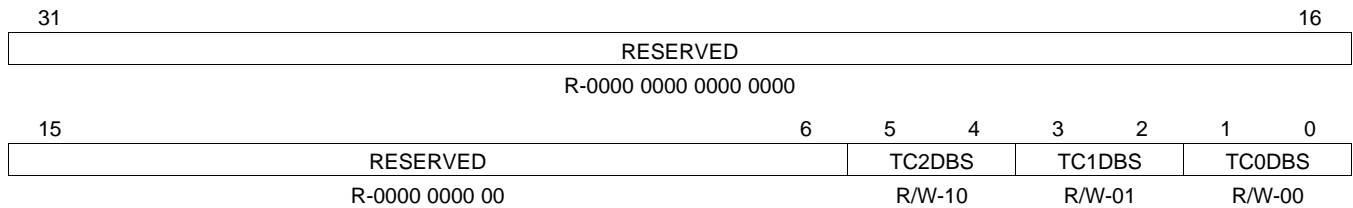
Figure 3-8. TIMERCTL Register— 0x01C4 0084

Table 3-15. TIMERCTL Register Description

| Bit | Field Name | Description |
|------|------------|--|
| 31:2 | RESERVED | Reserved. Read-Only, writes have no effect. |
| 1 | TINP0SEL | Timer0 External Input (TIN0) Select 0 = Timer0 external input comes directly from the TINP0L pin (default). 1 = Timer0 external input is TINP0L pin divided by 6. For example, if TINP0L = 25MHz, Timer0 input TIN0 is 25MHz / 6 = 4.2 MHz. |
| 0 | WDRST | WatchDog Reset Enable 0 = WatchDog Timer Event (WDINT from Timer2) does not cause device reset. 1 = WatchDog Timer Event (WDINT from Timer2) causes a device max reset (default). |

3.6.2.3 EDMA TC Configuration Register (EDMATCCFG)

The EDMA Transfer Controller Configuration (EDMATCCFG) register configures the default burst size (DBS) for EDMA TC0, EDMA TC1, and EDMA TC2. For more information on the correct usage of DBS, see the *TMS320C642x DSP Enhanced Direct Memory Access (EDMA) Controller User's Guide* (literature number [SPRUEM5](#)). **The user should only modify this register once during device initialization and when the corresponding EDMA TC is not in use.**



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

Figure 3-9. EDMA TC Configuration Register— 0x01C4 0088

Table 3-16. EDMA TC Configuration Register Description

| Bit | Field | Description |
|------|----------|---|
| 31:6 | RESERVED | Reserved. Read-Only, writes have no effect. |
| 5:4 | TC2DBS | EDMA TC2 Default Burst Size 00 = 16 byte 01 = 32 byte 10 = 64 byte (default) 11 = reserved EDMA TC2 is intended for PCI or miscellaneous transfers. TC2 FIFO size is 128 bytes, regardless of Default Burst Size setting. |
| 3:2 | TC1DBS | EDMA TC1 Default Burst Size 00 = 16 byte 01 = 32 byte (default) 10 = 64 byte 11 = reserved EDMA TC1 is intended for high throughput bulk transfers. TC1 FIFO size is 256 bytes, regardless of Default Burst Size setting. |
| 1:0 | TC0DBS | EDMA TC0 Default Burst Size 00 = 16 byte (default) 01 = 32 byte 10 = 64 byte 11 = reserved EDMA TC0 is intended for short burst transfers with stringent deadlines (e.g., McBSP, McASP). TC0 FIFO size is 128 bytes, regardless of Default Burst Size setting. |

3.7 Multiplexed Pin Configurations

C6424 makes extensive use of pin multiplexing to accommodate a large number of peripheral functions in the smallest possible package, providing ultimate flexibility for end applications.

The Pin Multiplex Registers PINMUX0 and PINMUX1 in the System Module are responsible for controlling all pin multiplexing functions on the C6424. The default setting of some of the PINMUX0 and PINMUX1 bit fields are configured by configuration pins latched at reset (see [Section 3.5.1, Device and Peripheral Configurations at Device Reset](#)). After reset, software may program the PINMUX0 and PINMUX1 registers to switch pin functionalities.

The following peripherals have multiplexed pins: EMIFA, PCI, HPI, VLYNQ, EMAC, McASP0, McBSP0, McBSP1, PWM0, PWM1, PWM2, Timer0, Timer1, UART0, UART1, and GPIO.

The device is divided into the following Pin Multiplexed Blocks (Pin Mux Blocks):

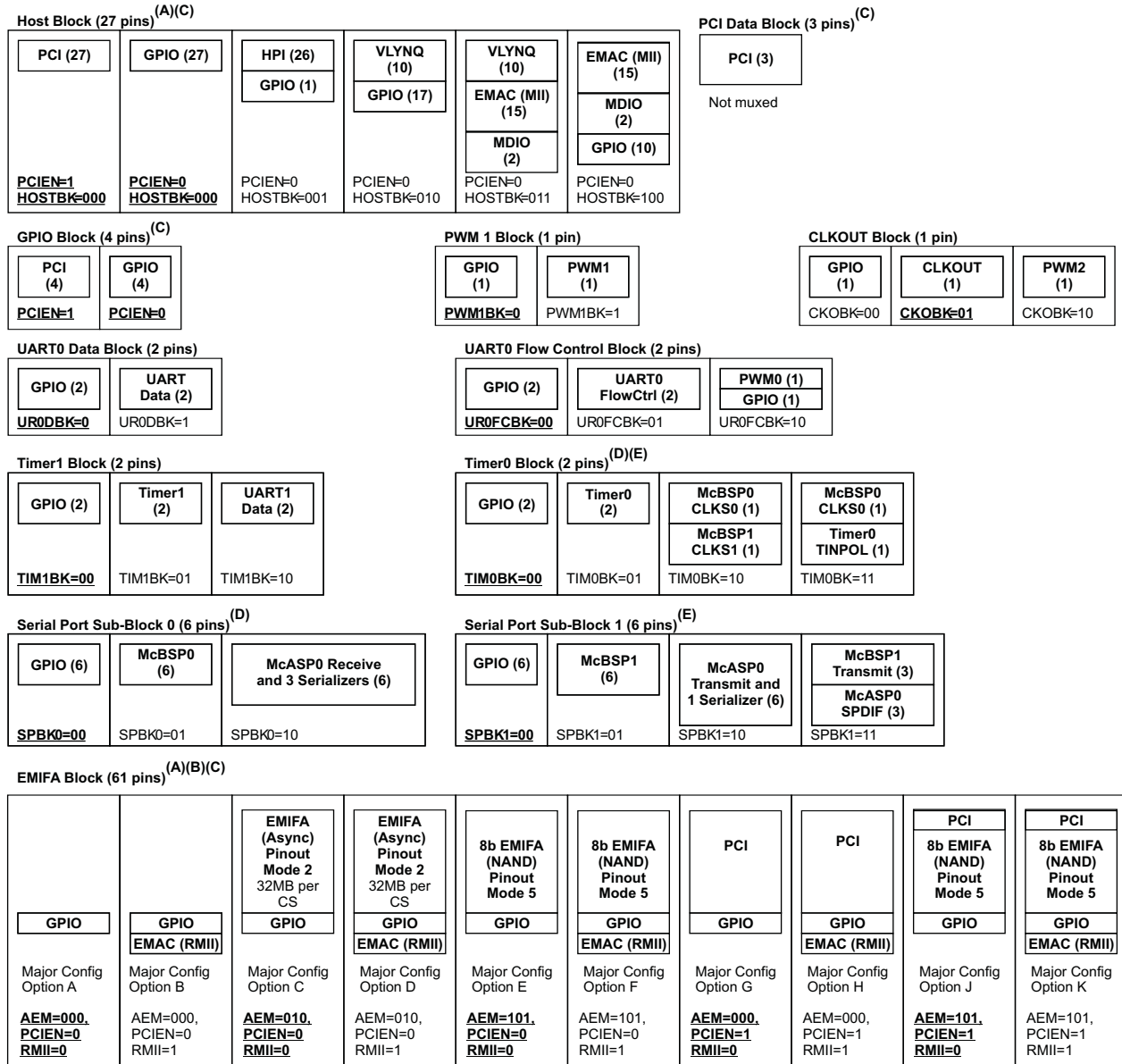
- **EMIFA Block:** EMIFA, part of PCI, GPIO. This block is further subdivided into these sub-blocks:
 - **Sub-Block 0:** part of EMIFA (data, address, control), part of PCI, part of EMAC(RMII), and GPIO
 - **Sub-Block 1:** part of EMIFA (data, address, control), part of EMAC(RMII), and GPIO
 - **Sub-Block 2:** part of EMIFA (control signals EM_WAIT/(RDY/BSY), EM_OE, and EM_WE)
 - **Sub-Block 3:** part of EMIFA (address EM_A[12:5]), part of PCI, and GPIO
- **Host Block:** HPI, VLYNQ, EMAC(MII), part of PCI, and GPIO
- **PCI Data Block:** part of PCI
- **GPIO Block:** part of PCI and GPIO
- **Serial Port Block:** McBSP0, McBSP1, McASP0, and GPIO. This block is further sub-divided into sub-blocks.
 - **Serial Port Sub-Block 0:** McBSP0, part of McASP0, and GPIO
 - **Serial Port Sub-Block 1:** McBSP1, part of McASP0, and GPIO
- **UART0 Flow Control Block:** UART0 flow control, PWM0, and GPIO
- **UART0 Data Block:** UART0 data and GPIO
- **Timer0 Block:** Timer0 and McBSPs' CLKS pins
- **Timer1 Block:** Timer1 and UART1 data
- **PWM1 Block:** PWM1 and GPIO
- **CLKOUT Block:** CLKOUT0, PWM2, and GPIO

As shown in the list above, the PCI, McBSP0, McBSP1, UART0, and EMAC peripherals span multiple Pin Mux Blocks. To use these peripherals, they must be selected in all relevant Pin Mux Blocks. For more details, see [Section 3.7.3, Pin Multiplexing Details](#), and [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#).

Note: There is no actual pin multiplexing in EMIFA Sub-Block 2 and the PCI Data Block. However these are still considered "pin mux blocks" because they contain part of the pins necessary for EMIFA and PCI, respectively.

A high level view of the Pin Mux Blocks is shown in [Figure 3-10](#). In each Pin Mux Block, the PINMUX0/PINMUX1 default settings are underlined.

Note: Some default pin functions are determined by configuration pins (PCIEN and AEM[2:0]); therefore, more than one configuration setting can serve as default based on the configuration pin settings latched at device reset.



- Default settings for PINMUX0 and PINMUX1 registers are underlined.
- EMIFA Block: This figure only shows the Major Config Options in the EMIFA Block based on the AEM, RMII, and PCIEN settings. Actual pin functions in the EMIFA Block are further determined by other PINMUX fields.
- PCI pins span multiple blocks (Host Block, GPIO Block, EMIFA Block, and PCI Data Block). For PCI to be operational, PCI pins must be selected in all of these Pin Mux Blocks. For the EMIFA Block, PCI is only supported if AEM = 000b or 101b.
- McBSP0 pins span multiple blocks (Serial Port Sub-Block0 and Timer0 Block). Serial Port Sub-Block0 contains most of the pins needed for McBSP0 operation. Timer0 Block contains the optional external clock source input CLKS0.
- McBSP1 spans multiple blocks (Serial Port Sub-Block1, Timer0 Block). Serial Port Sub-Block1 contains most of the pins needed for McBSP1 operation. Timer0 Block contains the optional external clock source input CLKS1.

Figure 3-10. Pin Mux Block Selection

3.7.1 Pin Muxing Selection At Reset

This section summarizes pin mux selection at reset.

The configuration pins AEM[2:0] and PCIEN latched at device reset determine default pin muxing for the following Pin Mux Blocks:

- EMIFA Block: default pin mux determined by AEM[2:0], RMII, and PCIEN. **After reset, software may modify settings in the PINMUX0 register to fine tune pin muxing in this block. However, after reset, software is not allowed to modify PINMUX1.PCIEN setting to change PCI pinout.**
 - PCIEN = 0, AEM[2:0] = 000b, RMII = 0: Major Config Option A is selected. This block defaults to 58 GPIO pins.
 - PCIEN = 0, AEM[2:0] = 010b, RMII = 0: Major Config Option C is selected. This block defaults to 16-bit EMIFA (Async) Pinout Mode 2, plus 13 to 16 GPIO pins.
 - PCIEN = 0, AEM[2:0] = 101b, RMII = 0: Major Config Option E is selected. This block defaults to 8-bit EMIFA (NAND) Pinout Mode 5, plus 44 to 47 GPIO pins.
 - PCIEN = 1, AEM[2:0] = 000b, RMII = 0: Major Config Option G is selected. This block defaults to PCI and 42 GPIO pins.
 - PCIEN = 1, AEM[2:0] = 101b, RMII = 0: Major Config Option J is selected. This block defaults to PCI, 8-bit EMIFA (NAND) Pinout Mode 5, and 28 to 31 GPIO pins.
- Host Block: default pin mux determined by PCIEN.
 - PCIEN = 0: the 27 pins in Host Block default to GPIO function. Software may program PINMUX1.HOSTBK to modify pin functions after reset.
 - PCIEN = 1: the 27 pins in Host Block serve as PCI pins. **Software is not allowed to modify this setting after reset.**
- GPIO Block: pin function determined by PCIEN configuration pin.
 - PCIEN = 0: the 4 pins in GPIO Block serve as GPIO pins. **Software is not allowed to modify this setting after reset.**
 - PCIEN = 1: the 4 pins in GPIO Block serve as PCI pins. **Software is not allowed to modify this setting after reset.**
- PCI Data Block: pin function determined by PCIEN.
 - PCIEN = 0: the 3 pins in PCI Data Block have no function and should be left unconnected. **Software is not allowed to modify this setting after reset.**
 - PCIEN = 1: the 3 pins in PCI Data Block serve as PCI pins. **Software is not allowed to modify this setting after reset.**

For a description of the PINMUX0 and PINMUX1 registers and more details on pin muxing, see [Section 3.7.2](#).

3.7.2 Pin Muxing Selection After Reset

The PINMUX0 and PINMUX1 registers in the System Module allow software to select the pin functions in the Pin Mux Blocks. The pin control of some of the Pin Mux Blocks requires a combination of PINMUX0/PINMUX1 bit fields. For more details on the combination of the PINMUX bit fields that control each muxed pin, see [Section 3.7.3.1, Multiplexed Pins on C6424](#).

This section only provides an overview of the PINMUX0 and PINMUX1 registers. For more detailed discussion on how to program each Pin Mux Block, see [Section 3.7.3, Pin Multiplexing Details](#).

3.7.2.1 PINMUX0 Register Description

The Pin Multiplexing 0 Register (PINMUX0) controls the pin function in the EMIFA Block. The PINMUX0

Table 3-17. PINMUX0 Register Description (continued)

| Bit | Field Name | Description | Pins Controlled |
|-----|--------------------|---|---|
| 2:0 | AEM ⁽¹⁾ | <p>EMIFA Pinout Modes This field does not affect the actual EMIFA operation. It only determines what multiplexed pins in the EMIFA Block serves as EMIFA pins. The pin mux control for some of these pins are controlled by a combination of AEM and PCIEEN fields:</p> <p>000b = No EMIFA Mode None of the multiplexed pins in the EMIFA Block serves as EMIFA pins. They serve as GPIO or PCI pins based on PCIEEN setting.</p> <p>001b = Reserved.</p> <p>010b = EMIFA (Async) Pinout Mode Pinout allows up to a maximum of these functions from EMIFA Block: 16-bit EMIFA (Async or NAND) + GPIO + EMAC(RMII). All of the pins listed under the "Pins Controlled" column serve as EMIFA pins. PINMUX1.PCIEEN must be set to 0. PINMUX0.RMII can be set to 0 or 1.</p> <p>011b = Reserved.</p> <p>100b = Reserved.</p> <p>101b = 8-bit EMIFA (NAND) Pinout Mode Pinout allows up to a maximum of these functions from EMIFA Block: 8-bit EMIFA (NAND) + GPIO + PCI + EMAC(RMII). PINMUX1.PCIEEN can be set to 0 or 1. PINMUX0.RMII can be set to 0 or 1.</p> <p>110b through 111b = Reserved.</p> | EM_D[15]/GP[36] EM_D[14]/GP[37] EM_D[13]/GP[38] EM_D[12]/GP[39] EM_D[11]/GP[40] EM_D[10]/GP[41] EM_D[9]/GP[42] EM_D[8]/GP[43] EM_D[7]/GP[21] EM_D[6]/GP[20] EM_D[5]/GP[19] EM_D[4]/GP[18] EM_D[3]/GP[17] EM_D[2]/GP[16] EM_D[1]/GP[15] EM_D[0]/GP[14] EM_R/WGP[35] EM_A[21]/GP[34] EM_A[20]/PINTA/GP[44] EM_A[19]/PREQ/GP[45] EM_A[18]/PRST/GP[46] EM_A[17]/AD31/GP[47] EM_A[16]/PGNT/GP[48] EM_A[15]/AD29/GP[49] EM_A[14]/AD27/GP[50] EM_A[13]/AD25/GP[51] EM_A[12]/PCBE3/GP[89] EM_A[11]/AD24/GP[90] EM_A[10]/AD23/GP[91] EM_A[9]/PIDSEL/GP[92] EM_A[8]/AD21/GP[93] EM_A[7]/AD22/GP[94] EM_A[6]/AD20/GP[95] EM_A[5]/AD19/GP[96] EM_A[4]/GP[10]/(PLLMS2) EM_A[3]/GP[11] EM_A[2]/(CLE)/GP[8]/(PLLMS0) EM_A[1]/(ALE)/GP[9]/(PLLMS1) EM_A[0]/GP[7]/(AEM2) EM_CS2/GP[12] EM_BA[0]/GP[6]/(AEM1) EM_BA[1]/GP[5]/(AEM0) The pin mux for these pins are controlled by a combination of AEM and other fields, including PCIEEN. For the full set of valid configurations of these pins, see Section 3.7.3.13.2, EMIFA Block Pin-by-Pin Multiplexing Summary . |

(1) The AEM default value is latched at reset from AEM[2:0] configuration inputs. The latched values are also shown at BOOTCFG.DAEM (read-only).

3.7.2.2 PINMUX1 Register Description

The Pin Multiplexing 1 Register (PINMUX1) controls the pin multiplexing of all Pin Mux Blocks. The PINMUX1 register format is shown in Figure 3-12 and the bit field descriptions are given in Table 3-18. Some muxed pins are controlled by more than one PINMUX bit field. For the combination of PINMUX bit fields that control each muxed pin, see Section 3.7.3.1, *Multiplexed Pins on C6424*.

| | | | | | | | | | | | | | | | | | |
|-------------|----------|-------|---------|---------|----|--------|---------|--------|---------|--------|----|----------|----|--------|-------|--------|----|
| 31 | RESERVED | | | | | | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| R/W-0000 00 | | | | | | R/W-00 | | R/W-00 | | R/W-00 | | R/W-00 | | R/W-00 | | R/W-00 | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| CKOBK | | RSV | PWM1 BK | UR0FCBK | | RSV | UR0D BK | RSV | HOSTBK | | | RESERVED | | | PCIEN | | |
| R/W-01 | | R/W-0 | R/W-0 | R/W-00 | | R/W-0 | R/W-0 | R/W-0 | R/W-000 | | | R/W-000 | | | R-P | | |

LEGEND: R/W = Read/Write; R = Read only; P = specified pin state; -n = value after reset

(1) For proper C6424 device operation, **always** write a value of "0" to all RESERVED/RSV bits.

Figure 3-12. PINMUX1 Register—0x01C4 0004 (1)

Table 3-18. PINMUX1 Register Description

| Bit | Field Name | Description | Pins Controlled |
|-------|------------|---|--|
| 31:26 | RESERVED | Reserved. For proper device operation, the user should only write "0" to this bit (default). | — |
| 25:24 | SPBK1 | Serial Port Sub-Block 1 Pin Select. Selects the function of the multiplexed pins in the Serial Port Sub-Block 1. 00 = GPIO Mode (default). Pins function as GPIO (GP[110:105]). 01 = McBSP1 Mode. Pins function as McBSP1: CLKX1, FSX1, DX1, CLKR1, FSR1, and DR1. 10 = McASP0 Transmit and 1 serializer. Pins function as McASP0: AXR0[0], ACLKX0, AFSX0, AHCLKX0, AMUTEIN0, and AMUTE0. 11 = McBSP1 Transmit + McASP0 SPDIF Mode. Pins function as McBSP1 transmit (CLKX1, FSX1, DX1) and McASP0 SPDIF (AXR0[0], AHCLKX0, AMUTE0). | Serial Port Sub-Block 1: AXR0[0]/FSR1/GP[105] ACLKX0/CLKX1/GP[106] AFSX0/DX1/GP[107] AHCLKX0/CLKR1/GP[108] AMUTEIN0/FSX1/GP[109] AMUTE0/DR1/GP[110] |
| 23:22 | SPBK0 | Serial Port Sub-Block 0 Pin Select. Selects the function of the multiplexed pins in the Serial Port Sub-Block 0. 00 = GPIO Mode (default). Pins function as GPIO (GP[104:99]). 01 = McBSP0 Mode. Pins function as McBSP0 CLKX0, FSX0, DX0, CLKR0, FSR0, and DR0. 10 = McASP0 Receive and 3 serializers. Pins function as McASP0 ACLKR0, AFSR0, AHCLKR0, AXR0_3, AXR0_2, and AXR0_1. 11 = Reserved | Serial Port Sub-Block 0: ACLKR0/CLKX0/GP[99] AFSR0/DR0/GP[100] AHCLKR0/CLKR0/GP[101] AXR0[3]/FSR0/GP[102] AXR0[2]/FSX0/GP[103] AXR0[1]/DX0/GP[104] |
| 21:20 | TIM1BK | Timer1 Block Pin Select. Selects the function of the multiplexed pins in the Timer1 Block. 00 = GPIO Mode (default). Pins function as GPIO (GP[56:55]). 01 = Timer1 Mode. Pins function as Timer1 TINP1L and TOUT1L. 10 = UART1 Data Mode. Pins function as UART1 data pins URXD1 and UTXD1. 11 = Reserved. | Timer1 Block: TINP1L/URXD1/GP[56] TOUT1L/UTXD1/GP[55] |
| 19:18 | RSV | Reserved. For proper device operation, the user should only write "0" to this bit (default). | — |

Table 3-18. PINMUX1 Register Description (continued)

| Bit | Field Name | Description | Pins Controlled |
|-------|------------|--|--|
| 17:16 | TIM0BK | <p>Timer0 Block Pin Select. Selects the function of the multiplexed pins in the Timer0 Block.</p> <p>00 = GPIO Mode (default). Pins function as GPIO (GP[98:97]).</p> <p>01 = Timer0 Mode. Pins function as Timer0 TINP0L and TOUT0L.</p> <p>10 = McBSP0 and McBSP1 External Clock Source Mode. Pins function as McBSP0 and McBSP1 external clock source pins, CLKS0 and CLKS1, respectively.</p> <p>11 = McBSP0 External Clock Source + Timer0 Input Mode. Pins function as McBSP0 external clock source CLKS0, and Timer0 input TINP0L.</p> | <p>Timer0 Block: CLKS1/TINP0L/GP[98] CLKS0/TOUT0L/GP[97]</p> |
| 15:14 | CK0BK | <p>CLKOUT Block Pin Select. Selects the function of the multiplexed pins in the CLKOUT Block.</p> <p>00 = GPIO Mode. Pin functions as GPIO (GP[84]).</p> <p>01 = CLKOUT Mode (default). Pin functions as device clock output CLKOUT0, sourced from PLLC1 OBSCLK.</p> <p>10 = PWM2 Mode. Pin functions as PWM2.</p> <p>11 = Reserved</p> | <p>CLKOUT Block: CLKOUT0/PWM2/GP[84]</p> |
| 13 | RSV | Reserved. For proper device operation, the user should only write "0" to this bit (default). | — |
| 12 | PWM1BK | <p>PWM1 Block Pin Select. Selects the function of the multiplexed pins in the PWM1 Block.</p> <p>0 = GPIO Mode (default). Pin functions as GPIO (GP[4]).</p> <p>1 = PWM1 Mode. Pin functions as PWM1.</p> | <p>PWM1 Block: GP[4]/PWM1</p> |
| 11:10 | UR0FCBK | <p>UART0 Flow Control Block Pin Select. Selects the function of the multiplexed pins in the UART0 Flow Control Block.</p> <p>00 = GPIO Mode (default). Pins function as GPIO (GP[88:87]).</p> <p>01 = UART0 Flow Control Mode. Pins function as UART0 Flow Control UCTS0 and URTS0.</p> <p>10 = PWM0 + GPIO Mode. Pins function as PWM0 and GPIO (GP[87]).</p> <p>11 = Reserved</p> | <p>UART0 Flow Control Block: UCTS0/GP[87] URTS0/PWM0/GP[88]</p> |
| 9 | RSV | Reserved. For proper device operation, the user should only write "0" to this bit (default). | — |
| 8 | UR0DBK | <p>UART0 Data Block Pin Select. Selects the function of the multiplexed pins in the UART0 Data Block.</p> <p>0 = GPIO Mode (default). Pins function as GPIO (GP[86:85]).</p> <p>1 = UART0 Data Mode. Pins function as UART0 data URXD0 and UTXD0.</p> | <p>UART0 Data Block: URXD0/GP[85] UTXD0/GP[86]</p> |

Table 3-18. PINMUX1 Register Description (continued)

| Bit | Field Name | Description | Pins Controlled |
|-----|------------|---|---|
| 7 | RSV | Reserved. For proper device operation, the user should only write "0" to this bit (default). | – |
| 6:4 | HOSTBK | <p>Host Block Pin Select. If EMAC operation is desired, EMAC must be placed in reset before programming PINMUX1. HOSTBK or PINMUX0.RMII to select EMAC pins.</p> <p>PCIEEN = 0 and HOSTBK = 000: GPIO Mode (default if PCIEEN = 0). Pins function as GPIO (GP[83:57]).</p> <p>PCIEEN = 0 and HOSTBK = 001: HPI + 1 GPIO Mode. Pins function as HPI and GPIO (GP[57]).</p> <p>PCIEEN = 0 and HOSTBK = 010: VLYNQ + 17 GPIO Mode. Pins function as VLYNQ (VLYNQ_CLOCK, VLYNQ_SCRUN, VLYNQ_RXD[3:0], VLYNQ_TXD[3:0]), and GP[83:67].</p> <p>PCIEEN = 0 and HOSTBK = 011: VLYNQ + MII + MDIO Mode. Pins function as VLYNQ (VLYNQ_CLOCK, VLYNQ_SCRUN, VLYNQ_RXD[3:0], VLYNQ_TXD[3:0]), MII (TXCLK, CRS, COL, TXD[3:0], RXVD, TXEN, RXER, RXCLK, RXD[3:0]), and MDIO (MDIO, MDC). When EMAC(MII) is selected, EMAC(RMII) must not be selected. PINMUX0.RMII must be set to 0.</p> <p>PCIEEN = 0 and HOSTBK = 100: MII + MDIO +10 GPIO Mode. Pins function as MII (TXCLK, CRS, COL, TXD[3:0], RXVD, TXEN, RXER, RXCLK, RXD[3:0]), MDIO (MDIO, MDC), and GP[66:57]. When EMAC(MII) is selected, EMAC(RMII) must not be selected. PINMUX0.RMII must be set to 0.</p> <p>PCIEEN = 1 and HOSTBK = 000: PCI Mode (default if PCIEEN = 1). Pins function as PCI pins: PCICLK, PCBE2, PCBE1, PCBE0, PFRAME, PIDRDY, PTRDY, PDEVSEL, PPER, PSTOP, PSERR, PPAR, AD[18:5], and AD03.</p> <p>All other PCIEEN and HOSTBK combinations reserved.</p> | <p>Host Block: VLYNQ_CLOCK/PCICLK/GP[57] HD0/VLYNQ_SCRUN/AD18/GP[58] HD1/VLYNQ_RXD0/AD16/GP[59] HD2/VLYNQ_RXD1/AD17/GP[60] HD3/VLYNQ_RXD2/PCBE2/GP[61] HD4/VLYNQ_RXD3/PFRAME/GP[62] HD5/VLYNQ_TXD0/PIRDY/GP[63] HD6/VLYNQ_TXD1/PTRDY/GP[64] HD7/VLYNQ_TXD2/PDEVSEL/GP[65] HD8/VLYNQ_TXD3/PPER/GP[66] HD9/MCOL/PSTOP/GP[67] HD10/MCRS/PSERR/GP[68] HD11/MTXD3/PCBE1/GP[69] HD12/MTXD2/PPAR/GP[70] HD13/MTXD1/AD14/GP[71] HD14/MTXD0/AD15/GP[72] HD15/MTXCLK/AD12/GP[73] HHWIL/MRXDV/AD13/GP[74] HCNTL1/MTXEN/AD11/GP[75] HCNTL0/MRXER/AD10/GP[76] HR/W/MRXCLK/AD8/GP[77] HDS2/MRXD0/AD9/GP[78] HDS1/MRXD1/AD7/GP[79] HRDY/MRXD2/PCBE0/GP[80] HCS/MDCLK/AD5/GP[81] HINT/MRXD3/AD6/GP[82] HAS/MDIO/AD3/GP[83]</p> <p>The combination of PINMUX1 fields PCIEEN and HOSTBK select the function of these 27 pins.</p> |
| 3:1 | RESERVED | Reserved. For proper device operation, the user should only write "0" to this bit (default). | – |

Table 3-18. PINMUX1 Register Description (continued)

| Bit | Field Name | Description | Pins Controlled |
|-----|------------|--|---|
| 0 | PCIEN | <p>PCI Enable.</p> <p>The PINMUX1.PCIEN reflects the state of the PCIEN pin. PINMUX1.PCIEN is read only and cannot be modified by software. For proper device operation, the user must hold the desired setting at the PCIEN pin throughout device operation.</p> <p>PCIEN = 0: No PCI supported. Internal pullup/pulldown (IPU/IPD) on these pins are enabled.</p> <p>For PCI multiplexed pins in the GPIO Block, when PCIEN = 0, the pins function as GPIO (GP[3:0]).</p> <p>For PCI multiplexed pins in the Host Block, refer to PINMUX1.HOSTBK field for the actual pin functions.</p> <p>For PCI multiplexed pins in the EMIFA Block, refer to the PINMUX0.AEM field for the actual pin functions.</p> <p>For PCI pins in the PCI Data Block, when PCIEN = 0, the pins have no function and should be left unconnected.</p> <p>PCIEN = 1: PCI supported. Internal pullup/pulldown (IPU/IPD) on all PCI pins are disabled.</p> <p>All pins function as PCI pins. Applicable <i>only</i> for PINMUX0.AEM = 000b or 101b.</p> | <p>Host Block*: See list of 27 pins in HOSTBK bit field description</p> <p>PCI Data Block: AD26 AD28 AD30</p> <p>GPIO Block: AD0/GP[0] AD1/GP[1] AD2/GP[2] AD4/GP[3]</p> <p>EMIFA Sub-Block 0*: EM_A[13]/AD25/GP[51] EM_A[14]/AD27/GP[50] EM_A[15]/AD29/GP[49] EM_A[16]/PGNT/GP[48] EM_A[17]/AD31/GP[47] EM_A[18]/PRST/GP[46] EM_A[19]/PREQ/GP[45] EM_A[20]/PINTA/GP[44]</p> <p>EMIFA Sub-Block 3*: EM_A[12]/PCBE3/GP[89] EM_A[11]/AD24/GP[90] EM_A[10]/AD23/GP[91] EM_A[9]/PIDSEL/GP[92] EM_A[8]/AD21/GP[93] EM_A[7]/AD22/GP[94] EM_A[6]/AD20/GP[95] EM_A[5]/AD19/GP[96]</p> <p>*The pin mux for the EMIFA Sub-Block 0, EMIFA Sub-Block 3, and Host Block pins are controlled by a combination of PCIEN and other PINMUX0/1 fields, including HOSTBK and AEM. See Section 3.7.3.13.2, EMIFA Block Pin-By-Pin Multiplexing Summary, for the full set of valid configurations of EMIFA Block pins. For the full set of valid configurations of Host Block pins, see Section 3.7.3.3, Host Block Muxing.</p> |

3.7.3 Pin Multiplexing Details

This section discusses how to program each Pin Mux Block to select the desired peripheral functions.

The following steps can be used to determine pin muxing suitable for the application:

1. Understand the major configuration choices available for the specific application.
 - (a) Device Major Configuration Choices: [Figure 3-10](#) shown in [Section 3.7](#), *Multiplexed Pin Configurations*, provides a high-level view of the device pin muxing and can be used to determine the possible mix of peripheral options for a specific application.
 - (b) EMIFA Block Major Configuration Choices: The EMIFA block features extensive pin multiplexing to accommodate a variety of applications. In addition to [Figure 3-10](#), [Section 3.7.3.13](#), *EMIFA Block Muxing*, provides more details on the Major Configuration choices for this block.
2. See [Section 3.7.3.1](#), *Multiplexed Pins on C6424*, for a summary of all the multiplexed pins on this device and the pin mux group they belong to.
3. Refer to the individual pin mux sections ([Section 3.7.3.3](#), *Host Block Muxing* to [Section 3.7.3.13](#), *EMIFA Block Muxing*) for pin muxing details for a specific pin mux block.
 - (a) For peripherals that span multiple pin mux blocks, the user must select the appropriate pins for that peripheral in all relevant pin mux blocks. For more details, see [Section 3.7.3.2](#), *Peripherals Spanning Multiple Pin Mux Blocks*.

For details on PINMUX0 and PINMUX1 registers, see [Section 3.7.2](#).

3.7.3.1 Multiplexed Pins on C6424

[Table 3-19](#) summarizes all of the multiplexed pins on C6424, the pin mux group for each pin, and the PINMUX register fields that control the pin. For pin mux details, see the specific pin mux group section ([Section 3.7.3.3](#), *Host Block Muxing* to [Section 3.7.3.13](#), *EMIFA Block Muxing*). For a description of the PINMUX register fields, see [Section 3.7.2](#).

Table 3-19. Multiplexed Pins on C6424

| SIGNAL | | | PINMUX DESCRIPTION | |
|-----------------------|---------|---------|--------------------|---------------------------------|
| NAME | ZWT NO. | ZDU NO. | PINMUX GROUP | CONTROLLED BY PINMUX BIT FIELDS |
| GP[54] | A14 | A18 | EMIFA Sub-Block 0 | - |
| GP[53] | A13 | A17 | EMIFA Sub-Block 0 | - |
| RMRXR/GP[52] | A15 | A19 | EMIFA Sub-Block 0 | RMII |
| EM_A[13]/AD25/GP[51] | B10 | A12 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_A[14]/AD27/GP[50] | A10 | A13 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_A[15]/AD29/GP[49] | B11 | C13 | EMIFASub-Block 0 | PCIEN, AEM |
| EM_A[16]/PGNT/GP[48] | C11 | B13 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_A[17]/AD31/GP[47] | A11 | B14 | EMIFASub-Block 0 | PCIEN, AEM |
| EM_A[18]/PRST/GP[46] | D11 | A14 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_A[19]/PREQ/GP[45] | B12 | C14 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_A[20]/PINTA/GP[44] | C12 | C15 | EMIFA Sub-Block 0 | PCIEN, AEM |
| EM_D[8]/GP[43] | A12 | A15 | EMIFA Sub-Block 0 | AEM |
| EM_D[9]/GP[42] | B13 | B15 | EMIFA Sub-Block 0 | AEM |
| EM_D[10]/GP[41] | C13 | B16 | EMIFA Sub-Block 0 | AEM |
| EM_D[11]/GP[40] | D14 | C18 | EMIFA Sub-Block 0 | AEM |
| EM_D[12]/GP[39] | B14 | A16 | EMIFA Sub-Block 0 | AEM |
| EM_D[13]/GP[38] | C14 | B17 | EMIFA Sub-Block 0 | AEM |
| EM_D[14]/GP[37] | B15 | B18 | EMIFA Sub-Block 0 | AEM |
| EM_D[15]/GP[36] | C15 | B19 | EMIFA Sub-Block 0 | AEM |
| EM_RW/GP[35] | D13 | C17 | EMIFA Sub-Block 0 | AEM |
| EM_A[21]/GP[34] | D12 | C16 | EMIFA Sub-Block 0 | AEM |
| RMRXD1/EM_CS5/GP[33] | F19 | J22 | EMIFA Sub-Block 1 | RMII, CS5SEL |

Table 3-19. Multiplexed Pins on C6424 (continued)

| SIGNAL | | | PINMUX DESCRIPTION | |
|-------------------------------|---------|---------|--------------------|---------------------------------|
| NAME | ZWT NO. | ZDU NO. | PINMUX GROUP | CONTROLLED BY PINMUX BIT FIELDS |
| RMRXD0/EM_CS4/GP[32] | E19 | H22 | EMIFA Sub-Block 1 | RMII, CS4SEL |
| REFCLK/GP[31] | D19 | G22 | EMIFA Sub-Block 1 | RMII |
| RMCRSDV/GP[30] | G19 | K22 | EMIFA Sub-Block 1 | RMII |
| RMTXEN/GP[29] | H15 | K21 | EMIFA Sub-Block 1 | RMII |
| RMTXD0/GP[28] | H16 | J21 | EMIFA Sub-Block 1 | RMII |
| RMTXD1/GP[27] | H17 | L19 | EMIFA Sub-Block 1 | RMII |
| GP[26]/(FASTBOOT) | G17 | K19 | EMIFA Sub-Block 1 | - |
| GP[25]/(BOOTMODE3) | G16 | H21 | EMIFA Sub-Block 1 | - |
| GP[24]/(BOOTMODE2) | G15 | L20 | EMIFA Sub-Block 1 | - |
| GP[23]/(BOOTMODE1) | F15 | K20 | EMIFA Sub-Block 1 | - |
| GP[22]/(BOOTMODE0) | F18 | J20 | EMIFA Sub-Block 1 | - |
| EM_D[7]/GP[21] | F17 | H20 | EMIFA Sub-Block 1 | AEM |
| EM_D[6]/GP[20] | F16 | F21 | EMIFA Sub-Block 1 | AEM |
| EM_D[5]/GP[19] | E17 | F22 | EMIFA Sub-Block 1 | AEM |
| EM_D[4]/GP[18] | E18 | G21 | EMIFA Sub-Block 1 | AEM |
| EM_D[3]/GP[17] | E16 | F20 | EMIFA Sub-Block 1 | AEM |
| EM_D[2]/GP[16] | D17 | E22 | EMIFA Sub-Block 1 | AEM |
| EM_D[1]/GP[15] | D18 | G20 | EMIFA Sub-Block 1 | AEM |
| EM_D[0]/GP[14] | D16 | E21 | EMIFA Sub-Block 1 | AEM |
| EM_CS3/GP[13] | C18 | D22 | EMIFA Sub-Block 1 | CS3SEL |
| EM_CS2/GP[12] | C19 | C22 | EMIFA Sub-Block 1 | AEM |
| EM_A[3]/GP[11] | B18 | D21 | EMIFA Sub-Block 1 | AEM |
| EM_A[4]/GP[10]/(PLLMS2) | A17 | B21 | EMIFA Sub-Block 1 | AEM |
| EM_A[1]/(ALE)/GP[9]/(PLLMS1) | A16 | B20 | EMIFA Sub-Block 1 | AEM |
| EM_A[2]/(CLE)/GP[8]/(PLLMS0) | B16 | A20 | EMIFA Sub-Block 1 | AEM |
| EM_A[0]/GP[7]/(AEM2) | B17 | C21 | EMIFA Sub-Block 1 | AEM |
| EM_BA[0]/GP[6]/(AEM1) | C17 | E20 | EMIFA Sub-Block 1 | AEM |
| EM_BA[1]/GP[5]/(AEM0) | C16 | C20 | EMIFA Sub-Block 1 | AEM |
| EM_A[12]/PCBE3/GP[89] | D10 | B12 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[11]/AD24/GP[90] | C10 | C12 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[10]/AD23/GP[91] | A9 | B11 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[9]/PIDSEL/GP[92] | D9 | C11 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[8]/AD21/GP[93] | B9 | A11 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[7]/AD22/GP[94] | C9 | C10 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[6]/AD20/GP[95] | D8 | B10 | EMIFA Sub-Block 3 | PCIEN, AEM |
| EM_A[5]/AD19/GP[96] | B8 | A10 | EMIFA Sub-Block 3 | PCIEN, AEM |
| VLYNQ_CLOCK/PCICLK/GP[57] | A7 | A8 | Host Block | PCIEN, HOSTBK |
| HD0/VLYNQ_SCRUN/AD18/GP[58] | C8 | B9 | Host Block | PCIEN, HOSTBK |
| HD1/VLYNQ_RXD0/AD16/GP[59] | D7 | C9 | Host Block | PCIEN, HOSTBK |
| HD2/VLYNQ_RXD1/AD17/GP[60] | A8 | A9 | Host Block | PCIEN, HOSTBK |
| HD3/VLYNQ_RXD2/PCBE2/GP[61] | B7 | B8 | Host Block | PCIEN, HOSTBK |
| HD4/VLYNQ_RXD3/PFRAME/GP[62] | C7 | C8 | Host Block | PCIEN, HOSTBK |
| HD5/VLYNQ_TXD0/PTRDY/GP[63] | A6 | A7 | Host Block | PCIEN, HOSTBK |
| HD6/VLYNQ_TXD1/PTRDY/GP[64] | D6 | C7 | Host Block | PCIEN, HOSTBK |
| HD7/VLYNQ_TXD2/PDEVSEL/GP[65] | B6 | B7 | Host Block | PCIEN, HOSTBK |
| HD8/VLYNQ_TXD3/PPERR/GP[66] | A5 | A6 | Host Block | PCIEN, HOSTBK |
| HD9/MCOL/PSTOP/GP[67] | C6 | C6 | Host Block | PCIEN, HOSTBK |
| HD10/MCRS/PSERR/GP[68] | B5 | B6 | Host Block | PCIEN, HOSTBK |
| HD11/MTXD3/PCBE1/GP[69] | C5 | A5 | Host Block | PCIEN, HOSTBK |

Table 3-19. Multiplexed Pins on C6424 (continued)

| SIGNAL | | | PINMUX DESCRIPTION | |
|--------------------------------------|---------|---------|--------------------------|---------------------------------|
| NAME | ZWT NO. | ZDU NO. | PINMUX GROUP | CONTROLLED BY PINMUX BIT FIELDS |
| HD12/MTXD2/PPAR/GP[70] | D5 | C5 | Host Block | PCIEN, HOSTBK |
| HD13/MTXD1/AD14/GP[71] | B4 | B4 | Host Block | PCIEN, HOSTBK |
| HD14/MTXD0/AD15/GP[72] | D4 | B5 | Host Block | PCIEN, HOSTBK |
| HD15/MTXCLK/AD12/GP[73] | A4 | A4 | Host Block | PCIEN, HOSTBK |
| HHWIL/MRXDV/AD13/GP[74] | C4 | D3 | Host Block | PCIEN, HOSTBK |
| HCNTL1/MTXEN/AD11/GP[75] | D3 | C4 | Host Block | PCIEN, HOSTBK |
| HCNTL0/MRXER/AD10/GP[76] | B3 | B2 | Host Block | PCIEN, HOSTBK |
| HR \overline{W} /MRXCLK/AD8/GP[77] | A3 | A3 | Host Block | PCIEN, HOSTBK |
| HDS2/MRXD0/AD9/GP[78] | C3 | C2 | Host Block | PCIEN, HOSTBK |
| HDS1/MRXD1/AD7/GP[79] | B2 | B3 | Host Block | PCIEN, HOSTBK |
| HRDY/MRXD2/PCBE0/GP[80] | D2 | C3 | Host Block | PCIEN, HOSTBK |
| HCS/MDCLK/AD5/GP[81] | C1 | D1 | Host Block | PCIEN, HOSTBK |
| HINT/MRXD3/AD6/GP[82] | C2 | D2 | Host Block | PCIEN, HOSTBK |
| HAS/MDIO/AD3/GP[83] | D1 | C1 | Host Block | PCIEN, HOSTBK |
| AD0/GP[0] | E1 | E1 | GPIO Block | PCIEN |
| AD1/GP[1] | E2 | E2 | GPIO Block | PCIEN |
| AD2/GP[2] | E3 | F1 | GPIO Block | PCIEN |
| AD4/GP[3] | E4 | F2 | GPIO Block | PCIEN |
| GP[4]/PWM1 | F3 | F3 | PWM1Block | PWM1BK |
| ACLKR0/CLKX0/GP[99] | H1 | J1 | Serial Port Sub-Block 0 | SPBK0 |
| AFSR0/DR0/GP[100] | H4 | K3 | Serial Port Sub-Block 0 | SPBK0 |
| AHCLKR0/CLKR0/GP[101] | J2 | K1 | Serial Port Sub-Block 0 | SPBK0 |
| AXR0[3]/FSR0/GP[102] | G4 | J3 | Serial Port Sub-Block 0 | SPBK0 |
| AXR0[2]/FSX0/GP[103] | H3 | J2 | Serial Port Sub-Block 0 | SPBK0 |
| AXR0[1]/DX0/GP[104] | J3 | K2 | Serial Port Sub-Block 0 | SPBK0 |
| AXR0[0]/FSR1/GP[105] | H2 | H2 | Serial Port Sub-Block 1 | SPBK1 |
| ACLKX0/CLKX1/GP[106] | F1 | G1 | Serial Port Sub-Block 1 | SPBK1 |
| AFSX0/DX1/GP[107] | G2 | G2 | Serial Port Sub-Block 1 | SPBK1 |
| AHCLKX0/CLKR1/GP[108] | G1 | H1 | Serial Port Sub-Block 1 | SPBK1 |
| AMUTEIN0/FSX1/GP[109] | F2 | G3 | Serial Port Sub-Block 1 | SPBK1 |
| AMUTE0/DR1/GP[110] | G3 | H3 | Serial Port Sub-Block 1 | SPBK1 |
| TINP1L/URXD1/GP[56] | L4 | P3 | Timer 1 Block | TIM1BK |
| TOUT1L/UTXD1/GP[55] | K4 | N3 | Timer 1 Block | TIM1BK |
| CLKS1/TINP0L/GP[98] | K2 | L2 | Timer 0 Block | TIM0BK |
| CLKS0/TOUT0L/GP[97] | J4 | L3 | Timer 0 Block | TIM0BK |
| URXD0/GP[85] | L2 | M2 | UART0 Data Block | UR0DBK |
| UTXD0/GP[86] | K3 | N1 | UART0 Data Block | UR0DBK |
| UCTS0/GP[87] | L1 | P1 | UART0 Flow Control Block | UR0FCBK |
| URTS0/PWM0/GP[88] | L3 | M3 | UART0 Flow Control Block | UR0FCBK |
| CLKOUT0/PWM2/GP[84] | M1 | R1 | CLKOUT Block | CKOBK |

Note: PINMUX groups EMIFA Sub-Block 2 and PCI Data Block are not shown in the above table because there is **no** actual pin multiplexing in those blocks. But these two blocks are still considered "pin mux blocks" because they contain some of the pins necessary for EMIFA and PCI, respectively. The pins in these blocks are as follows:

- **EMIFA Sub-Block 2**
 - EM_WAIT/(RDY/BSY)
 - EM_OE
 - EM_WE

- **PCI Data Block**
 - AD30
 - AD28
 - AD26

3.7.3.2 Peripherals Spanning Multiple Pin Mux Blocks

Some peripherals span multiple Pin Mux Blocks. To use these peripherals, they must be selected in all of the relevant Pin Mux Blocks. The following is the list of peripherals that span multiple Pin Mux Blocks:

- **PCI:** PCI pins span across the Host Block, EMIFA Block Sub-Block 0 and Sub-Block 3, PCI Data Block, and GPIO Block. To select PCI pins, program PINMUX registers as follows:
 - **Host Block:** PCIEN = 1, HOSTBK = 000
 - **EMIFA Block:** Select either Major Configuration Option G, H, J, or K. For more details on the PINMUX settings associated with Major Configuration Options G, H, J, or K, see [Section 3.7.3.13, EMIFA Block Muxing](#).
 - **PCI Data Block:** PCIEN = 1
 - **GPIO Block:** PCIEN = 1
- **McBSP0:** Six McBSP0 pins are located in the Serial Port Sub-Block 0, but the CLKS0 pin is muxed in the Timer0 Block. To select McBSP0 pins, program PINMUX registers as follows:
 - **Serial Port Sub-Block 0:** SPBK0 = 01
 - **Timer0 Block:** If CLKS0 pin is desired, program TIM0BK = 10 or 11.
- **McBSP1:** Six McBSP1 pins are located in the Serial Port Sub-Block 1, but the CLKS1 pin is muxed in the Timer0 Block. To select McBSP1 pins, program PINMUX registers as follows:
 - **Serial Port Sub-Block 1:** SPBK1 = 01 (if both McBSP1 transmit and receive pins are needed) or 11 (if only McBSP1 transmit pins are needed)
 - **Timer0 Block:** If CLKS1 pin is desired, program TIM0BK = 10.

- **UART0:** The two UART0 data pins are located in the UART0 Data Block, but the two UART0 flow control pins are located in the UART0 Flow Control Block. To select UART0, program PINMUX registers as follows:
 - **UART0 Data Block:** UR0BK = 1
 - **UART0 Flow Control Block:** If flow control pins are desired, program UR0FCBK = 01.

3.7.3.3 Host Block Muxing

This block of 27 pins consists of PCI, HPI, VLYNQ, EMAC(MII), MDIO, and GPIO muxed pins. The following register fields select the pin functions in the Host Block:

- PINMUX1.PCIEN
- PINMUX1.HOSTBK

[Table 3-20](#) summarizes the 27 pins in the Host Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-20. Host Block Muxed Pins Selection

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | | | | | |
|-------------------------------|-----------------------|----------------------------|--|--------|-------------|--|----------|----------------------------|----------|--|
| | HPI | | EMAC(MII)/MDIO | | VLYNQ | | PCI | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| VLYNQ_CLOCK/PCICLK/GP[57] | – | – | – | – | VLYNQ_CLOCK | | PCICLK | | GP[57] | PCIE = 0, and HOSTBK = 000 or HOSTBK = 001 or HOSTBK = 100 |
| HD0/VLYNQ_SCRUN/AD18/GP[58] | HD0 | PCIE = 0, and HOSTBK = 001 | – | – | VLYNQ_SCRUN | PCIE = 0, and HOSTBK = 010 or HOSTBK = 011 | AD18 | PCIE = 1, and HOSTBK = 000 | GP[58] | PCIE = 0, and HOSTBK = 000 or HOSTBK = 100 |
| HD1/VLYNQ_RXD0/AD16/GP[59] | HD1 | | – | – | VLYNQ_RXD0 | | AD16 | | GP[59] | |
| HD2/VLYNQ_RXD1/AD17/GP[60] | HD2 | | – | – | VLYNQ_RXD1 | | AD17 | | GP[60] | |
| HD3/VLYNQ_RXD2/PCBE2/GP[61] | HD3 | | – | – | VLYNQ_RXD2 | | PCBE2 | | GP[61] | |
| HD4/VLYNQ_RXD3/PFRAME/GP[62] | HD4 | | – | – | VLYNQ_RXD3 | | PFRAME | | GP[62] | |
| HD5/VLYNQ_TXD0/PIRDY/GP[63] | HD5 | | – | – | VLYNQ_TXD0 | | PIRDY | | GP[63] | |
| HD6/VLYNQ_TXD1/PTRDY/GP[64] | HD6 | | – | – | VLYNQ_TXD1 | | PTRDY | | GP[64] | |
| HD7/VLYNQ_TXD2/PDEVSEL/GP[65] | HD7 | | – | – | VLYNQ_TXD2 | | PDEVSEL | | GP[65] | |
| HD8/VLYNQ_TXD3/PPERR/GP[66] | HD8 | | – | – | VLYNQ_TXD3 | | PPERR | | GP[66] | |
| HD9/MCOL/PSTOP/GP[67] | HD9 | | MCOL | | – | | – | | PSTOP | |
| HD10/MCRS/PSERR/GP[68] | HD10 | MCRS | | – | – | PSERR | GP[68] | | | |
| HD11/MTXD3/PCBE1/GP[69] | HD11 | MTXD3 | PCIE = 0, and HOSTBK = 011 or HOSTBK = 100 | – | – | PCBE1 | GP[69] | | | |
| HD12/MTXD2/PPAR/GP[70] | HD12 | MTXD2 | | – | – | PPAR | GP[70] | | | |
| HD13/MTXD1/AD14/GP[71] | HD13 | MTXD1 | | – | – | AD14 | GP[71] | | | |
| HD14/MTXD0/AD15/GP[72] | HD14 | MTXD0 | | – | – | AD15 | GP[72] | | | |
| HD15/MTXCLK/AD12/GP[73] | HD15 | MTXCLK | | – | – | AD12 | GP[73] | | | |
| HHWIL/MRXDV/AD13/GP[74] | HHWIL | MRXDV | | – | – | AD13 | GP[74] | | | |
| HCNTL1/MTXEN/AD11/GP[75] | HCNTL1 | MTXEN | | – | – | AD11 | GP[75] | | | |
| HCNTL0/MRXER/AD10/GP[76] | HCNTL0 | MRXER | | – | – | AD10 | GP[76] | | | |
| HRW/MRXCLK/AD8/GP[77] | HRW | MRXCLK | | – | – | AD8 | GP[77] | | | |
| HDS2/MRXD0/AD9/GP[78] | HDS2 | MRXD0 | | – | – | AD9 | GP[78] | | | |
| HDS1/MRXD1/AD7/GP[79] | HDS1 | MRXD1 | – | – | AD7 | GP[79] | | | | |
| HRDY/MRXD2/PCBE0/GP[80] | HRDY | MRXD2 | – | – | PCBE0 | GP[80] | | | | |
| HCS/MDCLK/AD5/GP[81] | HCS | MDCLK | – | – | AD5 | GP[81] | | | | |
| HINT/MRXD3/AD6/GP[82] | HINT | MRXD3 | – | – | AD6 | GP[82] | | | | |
| HAS/MDIO/AD3/GP[83] | HAS | MDIO | – | – | AD3 | GP[83] | | | | |

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), PCI pins span across the following Pin Mux Blocks: Host Block, EMIFA Block Sub-Block 0 and Sub-Block 3, PCI Data Block, and GPIO Block. For proper PCI operation, PCI must be selected in all of these Pin Mux Blocks.

There is only one EMAC peripheral on the C6424 device, even though the pins for MII mode and the pins for RMII modes are brought out to different locations. The EMAC MII mode pins are in the Host Block, while EMAC RMII mode pins are only in the EMIFA Block. The user is only allowed to select either the MII pins or the RMII pins. The operation is undefined if the user attempts to select both MII pins and RMII pins.

[Table 3-21](#) provides a different view of the Host Block pin muxing, showing the Host Block function based on PINMUX1 settings. The selection options are also shown pictorially in [Figure 3-10](#).

If EMAC operation is desired, EMAC must be placed in reset before programming PINMUX1.HOSTBK to select EMAC pins.

Table 3-21. Host Block Function Selection

| PINMUX1 SETTING | | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------------|------------|-------------------------------------|--|
| PCIEN ⁽¹⁾ | HOSTBK | | |
| 1 | 000 | PCI (Default if PCIEN = 1) | PCI: PCICLK, $\overline{PCBE2}$, $\overline{PCBE1}$, $\overline{PCBE0}$, \overline{PFRAME} , \overline{PIDRDY} , \overline{PTRDY} , $\overline{PDEVSEL}$, \overline{PSTOP} , \overline{PPER} , \overline{PSERR} , \overline{PPAR} , AD[18:05], AD[03] Internal pullup/pulldown on all these pins are disabled. |
| 1 | 001 to 111 | Reserved | Reserved |
| 0 | 000 | GPIO (27) (Default if PCIEN = 0) | GPIO: GP[83:57] |
| 0 | 001 | HPI + GPIO (1) | HPI: HHWIL, HCNTL[1:0], $\overline{HR/W}$, $\overline{HDS2}$, $\overline{HDS1}$, \overline{HRDY} , \overline{HCS} , \overline{HINT} , \overline{HAS} , HD[15:0] GPIO: GP[57] |
| 0 | 010 | VLYNQ + GPIO (17) | VLYNQ: VLYNQ_CLOCK, VLYNQ_SCRUN, VLYNQ_RXD[3:0], VLYNQ_TXD[3:0] GPIO: GP[83:67] |
| 0 | 011 | VLYNQ + EMAC (MII) + MDIO | VLYNQ: VLYNQ_CLOCK, VLYNQ_SCRUN, VLYNQ_RXD[3:0], VLYNQ_TXD[3:0] EMAC (MII): TXCLK, CRS, COL, TXD[3:0], RXDV, TXEN, RXER, RXCLK, RXD[3:0] MDIO: MDC, MDIO If EMAC operation is desired, EMAC must be placed in reset before programming PINMUX1.HOSTBK or PINMUX0.RMII to select EMAC pins. |

(1) If PCIEN = 1, the internal pullup/pulldown on all Host Block pins are disabled. If PCIEN = 0, the internal pullup/pulldown on all Host Block pins are enabled.

Table 3-21. Host Block Function Selection (continued)

| PINMUX1 SETTING | | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------------|------------|-------------------------------|---|
| PCIEN ⁽¹⁾ | HOSTBK | | |
| 0 | 100 | EMAC (MII) + MDIO + GPIO (10) | EMAC (MII): TXCLK, CRS, COL, TXD[3:0], RXDV, TXEN, RXER, RXCLK, RXD[3:0] MDIO: MDC, MDIO GPIO: GP[66:57] If EMAC operation is desired, EMAC must be placed in reset before programming PINMUX1.HOSTBK or PINMUX0.RMII to select EMAC pins. |
| 0 | 101 to 111 | Reserved | Reserved |

The PINMUX1.PCIEN field is read-only, and its setting is determined by the PCIEN configuration pin. Based on the PCIEN configuration pin setting, the 27 pins in the Host Block defaults to either PCI or GPIO function.

In addition, the VDD3P3V_PWDN.HOST field determines the power state of the Host Block pins. The Host Block pins default to powered up. For more details on the VDD3P3V_PWDN.HOST field, see [Section 3.2, Power Considerations](#).

3.7.3.4 PCI Data Block

This block of 3 pins consists of 3 PCI Address/Data pins—AD30, AD28, AD26. The PINMUX1.PCIEN register field affects the pin functions in the PCI Data Block.

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), PCI pins span across the following Pin Mux Blocks: Host Block, EMIFA Block Sub-Block 0 and Sub-Block 3, PCI Data Block, and GPIO Block. For proper PCI operation, PCI must be selected in all of these Pin Mux Blocks.

The 3 pins in the PCI Data Block are not muxed with any other peripherals. However, the PINMUX1.PCIEN field controls the internal pullup/pulldown resistors on these pins. For PCI operation (PCIEN = 1), the internal pullup/pulldown resistors are disabled. If the device does not support PCI (PCIEN = 0), the internal pullup/pulldown resistors on these pins are enabled so that the user can leave these pins unconnected on the board.

[Table 3-22](#) shows the Host Block pin selection based on PINMUX1.PCIEN setting.

Table 3-22. PCI Data Block Pin Control

| PINMUX1.PCIEN | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|---|--|
| 0 | No Connect Pins (Default if PCIEN = 0) | No Connect Pins <i>Internal pullup/pulldown enabled. Leave these three pins unconnected on the board.</i> |
| 1 | PCI (Default if PCIEN = 1) | PCI: AD26, AD28, AD30 |

3.7.3.5 GPIO Block Muxing

This block of 4 pins consists of PCI and GPIO muxed pins. The PINMUX1.PCIEN register field selects the pin functions in the GPIO Block.

[Table 3-23](#) summarizes the 4 pins in the GPIO Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-23. GPIO Block Muxed Pins Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | |
|-----------|-----------------------|--------------------------|----------|--------------------------|
| | PCI | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT |
| AD0/GP[0] | AD0 | PCIEN = 1 ⁽¹⁾ | GP[0] | PCIEN = 0 ⁽¹⁾ |
| AD1/GP[1] | AD1 | | GP[1] | |
| AD2/GP[2] | AD2 | | GP[2] | |
| AD4/GP[3] | AD4 | | GP[3] | |

(1) If PCIEN = 1, the internal pullup/pulldown on all GPIO Block pins are disabled. If PCIEN = 0, the internal pullup/pulldown on all GPIO Block pins are enabled.

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), PCI pins span across the following Pin Mux Blocks: Host Block, EMIFA Block Sub-Block 0 and Sub-Block 3, PCI Data Block, and GPIO Block. For proper PCI operation, PCI **must** be selected in all of these Pin Mux Blocks.

[Table 3-24](#) provides a different view of the GPIO Block pin muxing, showing the GPIO Block function based on PINMUX1.PCIEN setting. The selection options are also shown pictorially in [Figure 3-10](#).

Table 3-24. GPIO Block Function Selection

| PINMUX1.PCIEN | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|------------------------------------|-------------------------|
| 0 | PCI (Default if PCIEN = 1) | PCI: AD0, AD1, AD2, AD4 |
| 1 | GPIO (4) (Default if PCIEN = 0) | GPIO: GP[3:0] |

The PINMUX1.PCIEN field is read-only, and its setting is determined by the PCIEN configuration pin. Based on the PCIEN configuration pin setting, the 4 pins in the GPIO Block defaults to either PCI or GPIO function.

In addition, the VDD3P3V_PWDN.GPIO field determines the power state of the GPIO Block pins. The GPIO Block pins default to powered up. For more details on the VDD3P3V_PWDN.GPIO field, see [Section 3.2, Power Considerations](#).

3.7.3.6 UART0 Data Block Muxing

This block of 2 pins consists of UART0 Data, and GPIO muxed pins. The PINMUX1.UR0DBK register field select the pin functions in the UART0 Data Block.

[Table 3-25](#) summarizes the 2 pins in the UART0 Data Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-25. UART0 Data Block Muxed Pins Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | |
|--------------|-----------------------|------------|----------|------------|
| | UART0 | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT |
| URXD0/GP[85] | URXD0 | UR0DBK = 1 | GP[85] | UR0DBK = 0 |
| UTXD0/GP[86] | UTXD0 | | GP[86] | |

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), the UART0 pins span across two Pin Mux Blocks: UART0 Data Block, and UART0 Flow Control Block. For proper UART0 operation, the two pins in the UART0 Data Block must be configured for UART0 data functions. The two pins in the UART0 Flow Control Block are optional.

[Table 3-26](#) provides a different view of the UART0 Data Block pin muxing, showing the UART0 Data Block function based on PINMUX1.UR0DBK setting. The selection options are also shown pictorially in [Figure 3-10](#).

Table 3-26. UART0 Data Block Function Selection

| PINMUX1.UR0DBK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------|--------------------|-------------------------|
| 0 | GPIO (2) (Default) | GPIO: GP[86:85] |
| 1 | UART0 Data | UART0: URXD0, UTXD0 |

In addition, the VDD3P3V_PWDN.UR0DAT field determines the power state of the UART0 Data Block pins. The UART0 Data Block pins default to powered down and not operational. To use these pins, user must first program VDD3P3V_PWDN.UR0DAT = 0 to power up the pins. For more details on the VDD3P3V_PWDN.UR0DAT field, see [Section 3.2, Power Considerations](#).

The UART0 Data Block features internal pullup resistors, which matches the UART inactive polarity.

3.7.3.7 UART0 Flow Control Block

This block of 2 pins consists of UART0 Flow Control, PWM0, and GPIO muxed pins. The PINMUX1.UR0FCBK register field selects the pin functions in the UART0 Flow Control Block.

Table 3-27 summarizes the 2 pins in the UART0 Flow Control Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-27. UART0 Flow Control Block Muxed Pins Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | | | |
|---------------------------|-----------------------|--------------|----------|--------------|----------|-----------------|
| | UART0 | | PWM0 | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| UCTS0/ GP[87] | UCTS0 | UR0FCBK = 01 | – | – | GP[87] | UR0FCBK = 00/10 |
| URTS0/ PWM0/ GP[88] | URTS0 | | PWM0 | UR0FCBK = 10 | GP[88] | UR0FCBK = 00 |

As discussed in Section 3.7.3.2, *Peripherals Spanning Multiple Pin Mux Blocks*, the UART0 pins span across two Pin Mux Blocks: UART0 Data Block, and UART0 Flow Control Block. For proper UART0 operation, the two pins in the UART0 Data Block must be configured for UART0 data functions. The two pins in the UART0 Flow Control Block are optional.

Table 3-28 provides a different view of the UART0 Flow Control Block pin muxing, showing the UART0 Flow Control Block function based on PINMUX1.UR0FCBK setting. The selection options are also shown pictorially in Figure 3-10.

Table 3-28. UART0 Flow Control Block Function Selection

| PINMUX1.UR0FCBK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|-----------------|--------------------|----------------------------|
| 00 | GPIO (2) (default) | GPIO: GP[88:87] |
| 01 | UART0 Flow Control | UART0: UCTS0, URTS0 |
| 10 | PWM0 + GPIO (1) | PWM0: PWM0 GPIO: GP[87] |
| 11 | Reserved | Reserved |

In addition, the VDD3P3V_PWDN.UR0FC field determines the power state of the UART0 Flow Control Block pins. The UART0 Flow Control Block pins default to powered down and not operational. To use these pins, user must first program VDD3P3V_PWDN.UR0FC = 0 to power up the pins. For more details on the VDD3P3V_PWDN.UR0FC field, see Section 3.2, *Power Considerations*.

The UART0 Flow Control Block features internal pullup resistors, which matches the UART inactive polarity.

3.7.3.8 Timer0 Block

This block of 2 pins consists of Timer0, McBSP0, McBSP1, and GPIO muxed pins. The PINMUX1.TIM0BK register field selects the pin functions in the Timer0 Block.

Table 3-29 summarizes the 2 pins in the Timer0 Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-29. Timer0 Block Muxed Pins Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | | | |
|-----------------------------|-----------------------|----------------|----------|----------------|----------|-------------|
| | McBSP | | Timer0 | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| CLKS1/ TINPOL/ GP[98] | CLKS1 | TIM0BK = 10 | TINPOL | TIM0BK = 01/11 | GP[98] | TIM0BK = 00 |
| CLKS0/ TOUT0L/ GP[97] | CLKS0 | TIM0BK = 10/11 | TOUT0L | TIM0BK = 01 | GP[97] | |

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), the McBSP0 pins span across two Pin Mux Blocks: Serial Port Sub-Block0, and Timer0 Block. For proper McBSP0 operation, the Serial Port Sub-Block0 must be programmed to select McBSP0 function. The McBSP0 CLKS0 pin in the Timer0 Block is optional for McBSP0 operation. CLKS0 is only needed if you desire using CLKS0 as an external clock source to the McBSP0 internal sample rate generator.

Similarly, as discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), the McBSP1 pins span across two Pin Mux Blocks: Serial Port Sub-Block1, and Timer0 Block. For proper McBSP1 operation, the Serial Port Sub-Block1 must be programmed to select McBSP1 function. The McBSP1 CLKS1 pin in the Timer0 Block is optional for McBSP1 operation. CLKS1 is only needed if you desire using CLKS1 as an external clock source to the McBSP1 internal sample rate generator.

[Table 3-30](#) provides a different view of the Timer0 Block pin muxing, showing the Timer0 Block function based on PINMUX1.TIM0BK setting. The selection options are also shown pictorially in [Figure 3-10](#).

Table 3-30. Timer0 Block Function Selection

| PINMUX1.TIM0BK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------|---|---------------------------------|
| 00 | GPIO (2) (default) | GPIO: GP[98:97] |
| 01 | Timer0 | Timer0: TINP0L, TOUT0L |
| 10 | McBSP0 External Clock Source, McBSP1 External Clock Source | McBSP0: CLKS0 McBSP1: CLKS1 |
| 11 | McBSP0 External Clock Source, Timer0 Input | McBSP0: CLKS0 Timer0: TINP0L |

In addition, the VDD3P3V_PWDN.TIMER0 field determines the power state of the Timer0 Block pins. The Timer0 Block pins default to powered down and not operational. To use these pins, user must first program VDD3P3V_PWDN.TIMER0 = 0 to power up the pins. For more details on the VDD3P3V_PWDN.TIMER0 field, see [Section 3.2, Power Considerations](#).

3.7.3.9 Timer1 Block

This block of 2 pins consists of Timer1, UART1 Data, and GPIO muxed pins. The PINMUX1.TIM1BK register field selects the pin functions in the Timer1 Block.

[Table 3-31](#) summarizes the 2 pins in the Timer1 Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-31. Timer1 Block Muxed Pins Selection

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | |
|-----------------------------|-----------------------|-------------|----------|-------------|----------|-------------|
| | TIMER1 | | UART1 | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| TINP1L/ URXD1/ GP[56] | TINP1L | TIM1BK = 01 | URXD1 | TIM1BK = 10 | GP[56] | TIM1BK = 00 |
| TOUT1L/ UTXD1/ GP[55] | TOUT1L | | UTXD1 | | GP[55] | |

Unlike UART0, UART1 only supports data pins but not flow control pins.

[Table 3-32](#) provides a different view of the Timer1 Block pin muxing, showing the Timer1 Block function based on PINMUX1.TIM1BK setting. The selection options are also shown pictorially in [Figure 3-10](#).

Table 3-32. Timer1 Block Function Selection

| PINMUX1.TIM1BK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------|--------------------|-------------------------|
| 00 | GPIO (2) (default) | GPIO: GP[56:55] |
| 01 | Timer1 | Timer1: TINP1L, TOUT1L |

Table 3-32. Timer1 Block Function Selection (continued)

| PINMUX1.TIM1BK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------|----------------|-------------------------|
| 10 | UART1 Data | UART1: URXD1, UTXD1 |
| 11 | Reserved | Reserved |

In addition, the VDD3P3V_PWDN.TIMER1 field determines the power state of the Timer1 Block pins. The Timer1 Block pins default to powered down and not operational. To use these pins, user must first program VDD3P3V_PWDN.TIMER1 = 0 to power up the pins. For more details on the VDD3P3V_PWDN.TIMER1 field, see [Section 3.2, Power Considerations](#).

The Timer1 Block features internal pull up resistors, which matches the UART inactive polarity.

3.7.3.10 Serial Port Block

This block of 12 pins consists of McASP0, McBSP0, McBSP1, and GPIO muxed pins. The following register fields select the pin functions in the Serial Port Block:

- PINMUX1.SPBK0
- PINMUX1.SPBK1

The Serial Port Block is further subdivided into these sub-blocks:

- Serial Port Sub-Block 0: McBSP0, part of McASP0, GPIO.
- Serial Port Sub-Block 1: McBSP1, part of McASP0, GPIO.

[Table 3-33](#) summarizes the 12 pins in the Serial Port Block, the multiplexed function on each pin, and the PINMUX configurations to select the corresponding function.

Table 3-33. Serial Port Block Muxed Pins Selection

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | |
|--------------------------------|-----------------------|---------------|----------|---------------|----------|------------|
| | McASP0 | | McBSP0/1 | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| Serial Port Sub-block 0 | | | | | | |
| ACLKR0/CLKX0/GP[99] | ACLKR0 | SPBK0 = 10 | CLKX0 | SPBK0 = 01 | GP[99] | SPBK0 = 00 |
| AFSR0/DR0/GP[100] | AFSR0 | | DR0 | | GP[100] | |
| AHCLKR0/CLKR0/GP[101] | AHCLKR0 | | CLKR0 | | GP[101] | |
| AXR0[3]/FSR0/GP[102] | AXR0[3] | | FSR0 | | GP[102] | |
| AXR0[2]/FSX0/GP[103] | AXR0[2] | | FSX0 | | GP[103] | |
| AXR0[1]/DX0/GP[104] | AXR0[1] | | DX0 | | GP[104] | |
| Serial Port Sub-block 1 | | | | | | |
| AXR0[0]/FSR1/GP[105] | AXR0[0] | SPBK1 = 10/11 | FSR1 | SPBK1 = 01 | GP[105] | SPBK1 = 00 |
| ACLKX0/CLKX1/GP[106] | ACLKX0 | SPBK1 = 10 | CLKX1 | SPBK1 = 01/11 | GP[106] | |
| AFSX0/DX1/GP[107] | AFSX0 | SPBK1 = 10 | DX1 | SPBK1 = 01/11 | GP[107] | |
| AHCLKX0/CLKR1/GP[108] | AHCLKX0 | SPBK1 = 10/11 | CLKR1 | SPBK1 = 01 | GP[108] | |
| AMUTEIN0/FSX1/GP[109] | AMUTEIN0 | SPBK1 = 10 | FSX1 | SPBK1 = 01/11 | GP[109] | |
| AMUTE0/DR1/GP[110] | AMUTE0 | SPBK1 = 10/11 | DR1 | SPBK1 = 01 | GP[110] | |

As discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), the McBSP0 pins span across two Pin Mux Blocks: Serial Port Sub-Block0, and Timer0 Block. For proper McBSP0 operation, the Serial Port Sub-Block0 must be programmed to select McBSP0 function. The McBSP0 CLKS0 pin in the Timer0 Block is optional for McBSP0 operation. CLKS0 is only needed if you desire using CLKS0 as an external clock source to the McBSP0 internal sample rate generator.

Similarly, as discussed in [Section 3.7.3.2, Peripherals Spanning Multiple Pin Mux Blocks](#), the McBSP1 pins span across two Pin Mux Blocks: Serial Port Sub-Block1, and Timer0 Block. For proper McBSP1 operation, the Serial Port Sub-Block1 must be programmed to select McBSP1 function. The McBSP1 CLKS1 pin in the Timer0 Block is optional for McBSP1 operation. CLKS1 is only needed if you desire using CLKS1 as an external clock source to the McBSP1 internal sample rate generator.

[Table 3-34](#) and [Table 3-35](#) provide a different view of the Serial Port Block. [Table 3-34](#) shows the Serial Port Sub-Block 0 function based on PINMUX1.SPBK0 setting. [Table 3-35](#) shows the Serial Port Sub-Block 1 function based on PINMUX1.SPBK1 setting. These selection options are also shown pictorially in [Figure 3-10](#).

Table 3-34. Serial Port Sub-Block 0 Function Selection

| PINMUX1.SPBK0 | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|-------------------------------|---|
| 00 | GPIO (6) (default) | GPIO: GP[104:99] |
| 01 | McBSP0 | McBSP0: CLKX0, FSX0, DX0, CLKR0, FSR0, DR0 |
| 10 | McASP0 Receive, 3 Serializers | McASP0: ACLKR0, AFSR0, AHCLKR0, AXR0[3], AXR0[2], AXR0[1] |
| 11 | Reserved | Reserved |

Table 3-35. Serial Port Sub-Block 1 Function Selection

| PINMUX1.SPBK1 | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|--------------------|--|
| 00 | GPIO (6) (default) | GPIO: GP[110:105] |
| 01 | McBSP1 | McBSP1: CLKX1, FSX1, DX1, CLKR1, FSR1, DR1 |

Table 3-35. Serial Port Sub-Block 1 Function Selection (continued)

| PINMUX1.SPBK1 | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|--|--|
| 10 | McASP0 Transmit with 1 Serializer and Mute Control | McASP0: AXR0[0], ACLKX0, AFSX0, AHCLKX0, AMUTEIN0 ⁽¹⁾ , AMUTE0 |
| 11 | McBSP1 Transmit, McASP0 SPDIF | McBSP1: CLKX1, FSX1, DX1 McASP0: AXR0[0], AHCLKX0, AMUTE0 |

(1) The input from the AMUTEIN0/FSX1/GP[109] pin is connected to both the McASP0 and GPIO.

In addition, the VDD3P3V_PWDN.SP field determines the power state of the Serial Port Block pins. The Serial Port Block pins default to powered down and not operational. To use these pins, user must first program VDD3P3V_PWDN.SP = 0 to power up the pins. For more details on the VDD3P3V_PWDN.SP field, see [Section 3.2, Power Considerations](#).

To facilitate McASP0 operation, the input from the AMUTEIN0/FSX1/GP[109] pin is connected to both the McASP0 and the GPIO module. Therefore when an external mute event occurs, in addition to notifying the McASP0, it can also cause an interrupt through the GPIO module.

3.7.3.11 PWM1 Block

This block of 1 pin consists of PWM1 and GPIO muxed pins (GP[4]/PWM1). The PINMUX1.PWM1BK register field selects the pin function in the PWM1 Block.

[Table 3-36](#) summarizes the 1 pin in the PWM1 Block, its multiplexed function, and the PINMUX configurations to select the corresponding function.

Table 3-36. PWM1 Block Muxed Pin Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | |
|------------|-----------------------|------------|----------|------------|
| | PWM1 | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT |
| GP[4]/PWM1 | PWM1 | PWM1BK = 1 | GP[4] | PWM1BK = 0 |

[Table 3-37](#) provides a different view of the PWM1 Block pin muxing, showing the PWM1 Block function based on PINMUX1.PWM1BK setting. The selection options are also shown pictorially in [Figure 3-10](#).

Table 3-37. PWM1 Block Function Selection

| PINMUX1.PWM1BK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|----------------|--------------------|-------------------------|
| 0 | GPIO (1) (default) | GPIO: GP[4] |
| 1 | PWM1 | PWM1: PWM1 |

In addition, the VDD3P3V_PWDN.PWM1 field determines the power state of the PWM1 Block pin. The PWM1 Block pin defaults to powered down and not operational. To use this pin, user must first program VDD3P3V_PWDN.PWM1 = 0 to power up the pin. For more details on the VDD3P3V_PWDN.PWM1 field, see [Section 3.2, Power Considerations](#).

3.7.3.12 CLKOUT Block

This block of 1 pin consists of CLKOUT, PWM2, and GPIO muxed pin (CLKOUT0/PWM2/GP[84]). The PINMUX1.CKOBK register field selects the pin function in the CLKOUT Block.

[Table 3-38](#) summarizes the 1 pin in the CLKOUT Block, its multiplexed function, and the PINMUX configurations to select the corresponding function.

Table 3-38. CLKOUT Block Multiplexed Pin Selection

| SIGNAL | MULTIPLEXED FUNCTIONS | | | | | |
|-----------------------------|-----------------------|------------|----------|------------|----------|------------|
| | CLKOUT0 | | PWM2 | | GPIO | |
| NAME | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| CLKOUT0/ PWM2/ GP[84] | CLKOUT0 | CKOBK = 01 | PWM2 | CKOBK = 10 | GP[84] | CKOBK = 00 |

Table 3-39 provides a different view of the CLKOUT Block pin muxing, showing the CLKOUT Block function based on PINMUX1.CKOBK setting. The selection options are also shown pictorially in Figure 3-10.

Table 3-39. CLKOUT Block Function Selection

| PINMUX1.CKOBK | BLOCK FUNCTION | RESULTING PIN FUNCTIONS |
|---------------|------------------|---------------------------|
| 00 | GPIO (1) | GPIO: GP[84] |
| 01 | CLKOUT (default) | Device Clock-Out: CLKOUT0 |
| 10 | PWM2 | PWM2: PWM2 |
| 11 | Reserved | Reserved |

This block defaults to CLKOUT0 pin function.

In addition, the VDD3P3V_PWDN.CLKOUT field determines the power state of the CLKOUT Block pin. The CLKOUT Block pin defaults to powered up. For more details on the VDD3P3V_PWDN.CLKOUT field, see Section 3.2, *Power Considerations*.

3.7.3.13 EMIFA Block Muxing

This block of 61 pins consists of EMIFA, EMAC(RMII), PCI, and GPIO muxed pins. The following register fields affect the pin functions in the EMIFA Block:

- All PINMUX0 register fields: AEM, CS5SEL, CS4SEL, CS3SEL, and RMII.
- PINMUX1.PCIEN

As discussed in Section 3.7.3.2, *Peripherals Spanning Multiple Pin Mux Blocks*, PCI pins span across the following Pin Mux Blocks: Host Block, EMIFA Block Sub-Block 0 and Sub-Block 3, PCI Data Block, and GPIO Block. For proper PCI operation, PCI must be selected in all of these Pin Mux Blocks.

There is only one EMAC peripheral on the C6424 device, even though the pins for MII mode and the pins for RMII modes are brought out to different locations. The EMAC MII mode pins are in the Host Block, while EMAC RMII mode pins are only in the EMIFA Block. The user is only allowed to select either the MII pins or the RMII pins. The operation is undefined if the user attempts to select both MII pins and RMII pins.

The EMIFA Block is divided into multiple sub-blocks for ultimate flexibility in pin multiplexing to accommodate a wide variety of applications, and for the purpose of I/O pins power control:

- Sub-Block 0: multiplexed between EMIFA data/address/control pins, PCI, part of EMAC(RMII), and GPIO.
- Sub-Block 1: multiplexed between EMIFA data/address/control pins, part of EMAC(RMII), and GPIO.
- Sub-Block 2: no multiplexing. EMIFA control pins EM_WAIT/(RDY/BSY), EM_OE, EM_WE.
- Sub-Block 3: multiplexed between EMIFA address pins EM_A[12:6], PCI, and GPIO.

The EMBK0, EMBK1, EMBK2, EMBK3 fields in the VDD3P3V_PWDN register determine the power state of the EMIFA Block pins. The EMIFA Block pins default to powered up. For more details on the EMBK0, EMBK1, EMBK2, EMBK3 fields in the VDD3P3V_PWDN register, see Section 3.2, *Power Considerations*.

To understand pin multiplexing in the EMIFA Block, see [Section 3.7.3.13.1](#), *EMIFA Block Major Configuration Choices* to determine the major configuration choices (A,B,C,D,E,F,G,H,J, or K). [Section 3.7.3.13.2](#), *EMIFA Block Pin-By-Pin Multiplexing Summary*, provides a pin-by-pin muxing summary for the EMIFA Block. For more information on the PINMUX0 and PINMUX1 registers, see [Section 3.7.2](#), *Pin Muxing Selection After Reset*.

3.7.3.13.1 EMIFA Block Major Configuration Choices

[Table 3-40](#) shows the major configuration choices in the EMIFA Block. Use this table to determine all of the PINMUX settings for the EMIFA Block: AEM, PCIEN, RMII, CS5SEL, CS4SEL, and CS3SEL.

Table 3-40. EMIFA Block Major Configuration Choices

| MAJOR CONFIG. OPTION | PINMUX SELECTION FIELDS | | | | | | RESULTING PERIPHERALS/PINS | | | |
|----------------------|-------------------------|-----|------|--------|--------|--------|----------------------------|--|---|--|
| | PCIEN | AEM | RMII | CS3SEL | CS4SEL | CS5SEL | PCI ⁽¹⁾ | EMIFA ⁽²⁾ | RMII | GPIO |
| A | 0 | 000 | 0 | 0 | 0 | 0 | - | - | - | 58 GP Pins: GP[96:89], GP[54:5] |
| B | 0 | 000 | 1 | 0 | 0 | 0 | - | - | RMII: RMRXER, RMRXD[1:0], RMTXD[1:0], REFCLK, RMCRSDV, RMTXEN | 50 GP Pins: GP[96:89], GP[54:53], GP[51:34], GP[26:5], |
| C | 0 | 010 | 0 | 0 or 1 | 0 or 1 | 0 or 1 | - | 16-bit EMIFA (Async) Pinout Mode 2: EM_A[21:0], EM_D[15:0], EM_R/W, EM_CS2, EM_BA[1:0], EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS5 (CS5SEL = 1), EM_CS4 (CS4SEL = 1), EM_CS3 (CS3SEL = 1) | - | 13 GP pins: GP[54:52], GP[31:22] <i>Optional Selection:</i> GP[33] (CS5SEL = 0), GP[32] (CS4SEL = 0), GP[13] (CS3SEL = 0) |
| D | 0 | 010 | 1 | 0 or 1 | 0 | 0 | - | 16-bit EMIFA (Async) Pinout Mode 2: EM_A[21:0], EM_D[15:0], EM_R/W, EM_CS2, EM_BA[1:0], EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS3 (CS3SEL = 1) | RMII: RMRXER, RMRXD[1:0], RMTXD[1:0], REFCLK, RMCRSDV, RMTXEN | 7 GP pins: GP[54:53], GP[26:22] <i>Optional Selection:</i> GP[13] (CS3SEL = 0) |
| E | 0 | 101 | 0 | 0 or 1 | 0 or 1 | 0 or 1 | - | 8-bit EMIFA (NAND) Pinout Mode 5: EM_D[7:0], EM_A[2:1], EM_CS2, EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS5 (CS5SEL = 1), EM_CS4 (CS4SEL = 1), EM_CS3 (CS3SEL = 1) | - | 44 GP pins: GP[96:89], GP[54:34], GP[31:22], GP[11:10], GP[7:5] <i>Optional Selection:</i> GP[33] (CS5SEL = 0), GP[32] (CS4SEL = 0), GP[13] (CS3SEL = 0) |
| F | 0 | 101 | 1 | 0 or 1 | 0 | 0 | - | 8-bit EMIFA (NAND) Pinout Mode 5: EM_D[7:0], EM_A[2:1], EM_CS2, EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS3 (CS3SEL = 1) | RMII: RMRXER, RMRXD[1:0], RMTXD[1:0], REFCLK, RMCRSDV, RMTXEN | 38 GP pins: GP[96:89], GP[54:34], GP[26:22], GP[11:10], GP[7:5] <i>Optional Selection:</i> GP[13] (CS3SEL = 0) |

(1) PCI pins span across multiple Pin Mux Blocks ([Section 3.7.3.2](#) , *Peripherals Spanning Multiple Pin Mux Blocks*). This table only refers to the PCI pins in the EMIFA Block.

(2) The EMIFA pins EM_WAIT/(RDY/BSY), EM_OE, and EM_WE are non-multiplexed pins. They are available in all the configuration options. However, they are only useful if additional EMIFA pins are functional. Therefore in this table, these pins are only listed in configuration options C,D,E,F,J, and K.

Table 3-40. EMIFA Block Major Configuration Choices (continued)

| MAJOR CONFIG. OPTION | PINMUX SELECTION FIELDS | | | | | | RESULTING PERIPHERALS/PINS | | | |
|----------------------|-------------------------|-----|------|--------|--------|--------|--|---|--|--|
| | PCIEN | AEM | RMII | CS3SEL | CS4SEL | CS5SEL | PCI ⁽¹⁾ | EMIFA ⁽²⁾ | RMII | GPIO |
| G | 1 | 000 | 0 | 0 | 0 | 0 | PCI: PGNT, PRST, PREQ, PINTA, PCBE3, PIDSEL AD31, AD29, AD27, AD[25:19] | - | - | 42 GP pins: GP[54:52], GP[43:5] |
| H | 1 | 000 | 1 | 0 | 0 | 0 | PCI: PGNT, PRST, PREQ, PINTA, PCBE3, PIDSEL AD31, AD29, AD27, AD[25:19] | - | RMII: RMRXR, RMRXD[1:0], RMTXD[1:0], REFCLK, RMCSDV, RMTXEN | 34 GP Pins: GP[54:53], GP[43:34], GP[26:5] |
| J | 1 | 101 | 0 | 0 or 1 | 0 or 1 | 0 or 1 | PCI: PGNT, PRST, PREQ, PINTA, PCBE3, PIDSEL AD31, AD29, AD27, AD[25:19] | 8-bit EMIFA (NAND) Pinout Mode 5: EM_D[7:0], EM_A[2:1], EM_CS2, EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS5 (CS5SEL = 1), EM_CS4 (CS4SEL = 1), EM_CS3 (CS3SEL = 1) | - | 28 GP Pins: GP[54:52], GP[43:34], GP[31:22], GP[11:10], GP[7:5] <i>Optional Selection:</i> GP[33] (CS5SEL = 0), GP[32] (CS4SEL = 0), GP[13] (CS3SEL = 0) |
| K | 1 | 101 | 1 | 0 or 1 | 0 | 0 | PCI: PGNT, PRST, PREQ, PINTA, PCBE3, PIDSEL AD31, AD29, AD27, AD[25:19] | 8-bit EMIFA (NAND) Pinout Mode 5: EM_D[7:0], EM_A[2:1], EM_CS2, EM_WAIT/(RDY/BSY), EM_WE, EM_OE <i>Optional Selection:</i> EM_CS3 (CS3SEL = 1) | RMII: RMRXR, RMRXD[1:0], RMTXD[1:0], REFCLK, RMCSDV, RMTXEN | 22 GP Pins: GP[54:53], GP[43:34], GP[26:22], GP[11:10], GP[7:5] <i>Optional Selection:</i> GP[13] (CS3SEL = 0) |

The following is an example on how to read [Table 3-40](#). For example, the "PINMUX Selection Fields" columns indicate that Major Configuration Choice C is selected through setting PINMUX1.PCIEN = 0, PINMUX0.AEM = 010b, and PINMUX0.RMII = 0. Other PINMUX0 fields CS3SEL, CS4SEL, and CS5SEL can be set to either 0 or 1 based on the system's EMIFA Chip Select space need. The "Resulting Peripherals/Pins" columns indicate that Major Configuration Option C can support the following combination of pin functions:

- No PCI pins
- Pins for 16-bit EMIFA (Async or NAND) function with EMIFA Chip Select space 2 ($\overline{\text{EM_CS2}}$). If additional Chip Select spaces are needed, set the corresponding PINMUX bit (CS5SEL, CS4SEL, and/or CS3SEL) to 1.
- At least 13 GPIO pins. If the additional Chip Select spaces from $\overline{\text{EM_CS3}}$, $\overline{\text{EM_CS4}}$, or $\overline{\text{EM_CS5}}$ are not needed, the corresponding PINMUX bit (CS3SEL, CS4SEL, and/or CS5SEL) can be set to 0 to get additional GPIO pins.

3.7.3.13.2 EMIFA Block Pin-By-Pin Multiplexing Summary

This section summarizes the EMIFA Block muxing on a pin-by-pin basis. It provides an alternative view to pin muxing in the EMIFA Block. It summarizes the EMIFA Block pin muxing by dividing up the EMIFA Block based on the PINMUX field that controls the pins. To determine the actual EMIFA Major Configuration Option for the application need, see [Section 3.7.3.13.1, EMIFA Block Major Configuration Choices](#).

Table 3-41 shows the pin multiplexing control for each pin in the EMIFA Sub-Block 0. These PINMUX0 and PINMUX1 register fields control the multiplexing in this sub-block:

- PINMUX0: AEM, RMII
- PINMUX1: PCIEN

Table 3-42 shows the pin multiplexing control for each pin in the EMIFA Sub-Block 1. These PINMUX0 register fields control the multiplexing in this sub-block:

- PINMUX0: AEM, CS5SEL, CS4SEL, CS3SEL, RMII

EMIFA Sub-Block 2 is dedicated to EMIFA pins EM_WAIT/(RDY/BSY), EM_OE, and EM_WE. There is no pin multiplexing in this block. These pins always function as EMIFA control pins.

Table 3-43 shows the pin multiplexing control for each pin in the EMIFA Sub-Block 3. These PINMUX0 and PINMUX1 register fields control the multiplexing in this sub-block:

- PINMUX0: AEM
- PINMUX1: PCIEN

Table 3-41. EMIFA Sub-Block 0 Pin-By-Pin Mux Control

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | | | |
|-----------------------|-----------------------|----------------------|----------|------------------------|------------|----------|----------|------------------------|
| | EMIFA | | PCI | | EMAC(RMII) | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| GP[54] | – | – | – | – | – | – | GP[54] | – |
| GP[53] | – | – | – | – | – | – | GP[53] | – |
| RMRXER/GP[52] | – | – | – | – | RMXER | RMII = 1 | GP[52] | RMII = 0 |
| EM_A[13]/AD25/GP[51] | EM_A[13] | PCIEN = 0 AEM = 2 | AD25 | PCIEN = 1 AEM = 0/5 | – | – | GP[51] | PCIEN = 0 AEM = 0/5 |
| EM_A[14]/AD27/GP[50] | EM_A[14] | | AD27 | | – | – | GP[50] | |
| EM_A[15]/AD29/GP[49] | EM_A[15] | | AD29 | | – | – | GP[49] | |
| EM_A[16]/PGNT/GP[48] | EM_A[16] | | PGNT | | – | – | GP[48] | |
| EM_A[17]/AD31/GP[47] | EM_A[17] | | AD31 | | – | – | GP[47] | |
| EM_A[18]/PRST/GP[46] | EM_A[18] | | PRST | | – | – | GP[46] | |
| EM_A[19]/PREQ/GP[45] | EM_A[19] | | PREQ | | – | – | GP[45] | |
| EM_A[20]/PINTA/GP[44] | EM_A[20] | | PINTA | | – | – | GP[44] | |

Table 3-41. EMIFA Sub-Block 0 Pin-By-Pin Mux Control (continued)

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | | | |
|-----------------------------|-----------------------|---------|----------|--------|------------|--------|----------|-----------|
| | EMIFA | | PCI | | EMAC(RMII) | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| EM_D[8]/GP[43] | EM_D[8] | AEM = 2 | – | – | – | – | GP[43] | AEM = 0,5 |
| EM_D[9]/GP[42] | EM_D[9] | | – | – | – | – | GP[42] | |
| EM_D[10]/GP[41] | EM_D[10] | | – | – | – | – | GP[41] | |
| EM_D[11]/GP[40] | EM_D[11] | | – | – | – | – | GP[40] | |
| EM_D[12]/GP[39] | EM_D[12] | | – | – | – | – | GP[39] | |
| EM_D[13]/GP[38] | EM_D[13] | | – | – | – | – | GP[38] | |
| EM_D[14]/GP[37] | EM_D[14] | | – | – | – | – | GP[37] | |
| EM_D[15]/GP[36] | EM_D[15] | | – | – | – | – | GP[36] | |
| EM_R \overline{W} /GP[35] | EM_R \overline{W} | | – | – | – | – | GP[35] | |
| EM_A[21]/GP[34] | EM_A[21] | – | – | – | – | GP[34] | | |

Table 3-42. EMIFA Sub-Block 1 Pin-By-Pin Mux Control

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | |
|----------------------|-----------------------|------------------------|------------|------------------------|----------|------------------------|
| | EMIFA | | EMAC(RMII) | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| RMRXD1/EM_CS5/GP[33] | EM_CS5 | RMII = 0 CS5SEL = 1 | RMRXD1 | RMII = 1 CS5SEL = 0 | GP[33] | RMII = 0 CS5SEL = 0 |
| RMRXD0/EM_CS4/GP[32] | EM_CS4 | RMII = 0 CS4SEL = 1 | RMRXD0 | RMII = 1 CS4SEL = 0 | GP[32] | RMII = 0 CS4SEL = 0 |
| REFCLK/GP[31] | – | – | REFCLK | RMII = 1 | GP[31] | RMII = 0 |
| RMCRSDV/GP[30] | – | – | RMCRSDV | | GP[30] | |
| RMTXEN/GP[29] | – | – | RMTXEN | | GP[29] | |
| RMTXD0/GP[28] | – | – | RMTXD0 | | GP[28] | |
| RMTXD1/GP[27] | – | – | RMTXD1 | | GP[27] | |
| GP[26]/(FASTBOOT) | – | – | – | – | GP[26] | – |
| GP[25]/(BOOTMODE3) | – | – | – | – | GP[25] | – |
| GP[24]/(BOOTMODE2) | – | – | – | – | GP[24] | – |
| GP[23]/(BOOTMODE1) | – | – | – | – | GP[23] | – |
| GP[22]/(BOOTMODE0) | – | – | – | – | GP[22] | – |
| EM_D[7]/GP[21] | EM_D[7] | AEM = 2/5 | – | – | GP[21] | AEM = 0 |
| EM_D[6]/GP[20] | EM_D[6] | | – | – | GP[20] | |
| EM_D[5]/GP[19] | EM_D[5] | | – | – | GP[19] | |
| EM_D[4]/GP[18] | EM_D[4] | | – | – | GP[18] | |
| EM_D[3]/GP[17] | EM_D[3] | | – | – | GP[17] | |
| EM_D[2]/GP[16] | EM_D[2] | | – | – | GP[16] | |
| EM_D[1]/GP[15] | EM_D[1] | | – | – | GP[15] | |
| EM_D[0]/GP[14] | EM_D[0] | | – | – | GP[14] | |

Table 3-42. EMIFA Sub-Block 1 Pin-By-Pin Mux Control (continued)

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | |
|------------------------------|-----------------------|------------|------------|--------|----------|------------|
| | EMIFA | | EMAC(RMII) | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| EM_CS3/GP[13] | EM_CS3 | CS3SEL = 1 | – | – | GP[13] | CS3SEL = 0 |
| EM_CS2/GP[12] | EM_CS2 | AEM = 2/5 | – | – | GP[12] | AEM = 0 |
| EM_A[3]/GP[11] | EM_A[3] | AEM = 2 | – | – | GP[11] | AEM = 0/5 |
| EM_A[4]/GP[10]/(PLLMS2) | EM_A[4] | | GP[10] | | | |
| EM_A[1]/(ALE)/GP[9]/(PLLMS1) | EM_A[1]/(ALE) | AEM = 2/5 | – | – | GP[9] | AEM = 0 |
| EM_A[2]/(CLE)/GP[8]/(PLLMS0) | EM_A[2]/(CLE) | | GP[8] | | | |
| EM_A[0]/GP[7]/(AEM2) | EM_A[0] | AEM = 2 | – | – | GP[7] | AEM = 0/5 |
| EM_BA[0]/GP[6]/(AEM1) | EM_BA[0] | | GP[6] | | | |
| EM_BA[1]/GP[5]/(AEM0) | EM_BA[1] | | GP[5] | | | |

Table 3-43. EMIFA Sub-Block 3 Pin-By-Pin Mux Control

| SIGNAL NAME | MULTIPLEXED FUNCTIONS | | | | | |
|-----------------------|-----------------------|-----------------------|----------|-------------------------|----------|-------------------------|
| | EMIFA | | PCI | | GPIO | |
| | FUNCTION | SELECT | FUNCTION | SELECT | FUNCTION | SELECT |
| EM_A[12]/PCBE3/GP[89] | EM_A[12] | PCIEN = 0, AEM = 2 | PCBE3 | PCIEN = 1, AEM = 0/5 | GP[89] | PCIEN = 0, AEM = 0/5 |
| EM_A[11]/AD24/GP[90] | EM_A[11] | | AD24 | | GP[90] | |
| EM_A[10]/AD23/GP[91] | EM_A[10] | | AD23 | | GP[91] | |
| EM_A[9]/PIDSEL/GP[92] | EM_A[9] | | PIDSEL | | GP[92] | |
| EM_A[8]/AD21/GP[93] | EM_A[8] | | AD21 | | GP[93] | |
| EM_A[7]/AD22/GP[94] | EM_A[7] | | AD22 | | GP[94] | |
| EM_A[6]/AD20/GP[95] | EM_A[6] | | AD20 | | GP[95] | |
| EM_A[5]/AD19/GP[96] | EM_A[5] | | AD19 | | GP[96] | |

3.8 Device Initialization Sequence After Reset

Software should follow this initialization sequence after coming out of device reset.

1. Complete the boot sequence as needed. For more details on the boot sequence, see the *Using the TMS320C642x Bootloader* Application Report (literature number [SPRAAK5](#)).
2. If the device is not already at the desired operating frequency, program the PLL Controllers (PLL1 and PLL2) to configure the device frequency. For details on how to program the PLLC, see the *C642x DSP Phase-Locked Loop Controller (PLLC) User's Guide* (literature number [SPRUES0](#)).
3. Program PINMUX0 and PINMUX1 registers to select device pin functions. For more details on programming the PINMUX0 and PINMUX1 registers to select device pin functions, see [Section 3.7, Multiplexed Pin Configurations](#).

Note: If EMAC operation is desired, the EMAC **must** be placed in reset before programming PINMUX1.HOSTBK and PINMUX1.RMII to select EMAC pins.
4. Program the VDD3P3V_PWDN register to power up the necessary I/O pins. For more details on programming the VDD3P3V_PWDN register, see [Section 3.2, Power Considerations](#).
5. As needed by the application, program the following System Module registers when there are no active transactions on the respective peripherals:
 - (a) HPICTL ([Section 3.6.2.1, HPI Control Register](#)): applicable for HPI **only** if a different host burst write timeout value from default is desired.
 - (b) TIMERCTL ([Section 3.6.2.2, Timer Control Register](#)): applicable for Timer0 and Watchdog Timer2 **only**.
 - (c) EDMATCCFG ([Section 3.6.2.3, EDMA TC Configuration Register](#)): applicable for EDMA **only**. The recommendation is to leave the EDMATCCFG register at its default.
6. Program the Power and Sleep Controller (PSC) to enable the desired peripherals. For details on how to program the PSC, see the *TMS320C642x Power and Sleep Controller (PSC) User's Guide* (literature number [SPRUEN8](#)).
7. Program the Switched Central Resource (SCR) bus priorities for the master peripherals ([Section 3.6.1](#)). This **must** be configured when there are no active transactions on the respective peripherals:
 - (a) Program the MSTPRI0 and MSTPRI1 registers in the System Module. These registers can be programmed **before or after** the respective peripheral is enabled by the PSC in step 6.
 - (b) Program the EDMACC QUEPRI register, the C64x+ MDMAARBE.PRI field. These registers can only be programmed **after** the respective peripheral is enabled by the PSC in step 6.
8. Configure the C64x+ Megamodule and the peripherals.
 - (a) For details on C64x+ Megamodule configuration, see the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number [SPRU871](#)).

Special considerations: Bootloader disables C64x+ cache—For all boot modes that default to DSPBOOTADDR = 0x0010 0000 (i.e., all boot modes except the EMIFA ROM Direct Boot, BOOTMODE[3:0] = 0100, FASTBOOT = 0), the bootloader code disables all C64x+ cache (L2, L1P, and L1D) so that upon exit from the bootloader code, all C64x+ memories are configured as all RAM (L2CFG.L2MODE = 0h, L1PCFG.L1PMODE = 0h, and L1DCFG.L1DMODE = 0h). If cache use is required, the application code must explicitly enable the cache. For more information on boot modes, see [Section 3.4.1, Boot Modes](#). For more information on the bootloader, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).

(b) Peripherals configuration: see the respective peripheral user's guide.

Special considerations: DDR2 memory controller—the Peripheral Bus Burst Priority Register (PBBPR) should be programmed to ensure good DDR2 throughput and to prevent command starvation (prevention of certain commands from being processed by the DDR2 memory controller). For more details, see the *TMS320C642x DSP DDR2 Memory Controller User's Guide* (literature number [SPRUEM4](#)). A hex value of 0x20 is recommended for the PBBPR PR_OLD_COUNT field to provide a good DSP performance and still allow good utilization by other modules.

3.9 Debugging Considerations

3.9.1 Pullup/Pulldown Resistors

Proper board design should ensure that input pins to the C642x DSP device always be at a valid logic level and not floating. This may be achieved via pullup/pulldown resistors. The C642x DSP 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 needs to be used in the following situations:

- *Boot and Configuration Pins:* If the pin is both routed out and 3-stated (not driven), an external pullup/pulldown resistor is **strongly recommended**, 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.
- *EMIFA Chip Select Outputs:* On C6424, the EMIFA chip select pins ($\overline{\text{EM_CS2}}$, $\overline{\text{EM_CS3}}$, $\overline{\text{EM_CS4}}$, and $\overline{\text{EM_CS5}}$) feature an internal pulldown (IPD) resistor. If these pins are connected and used as an EMIFA chip select signal, for proper device operation, an external pullup resistor **must** be used to ensure the $\overline{\text{EM_CSx}}$ function defaults to an inactive (high) state.

For the boot and configuration pins (listed in [Table 2-7, Boot Terminal Functions](#)), if they are both routed out and 3-stated (not driven), it is **strongly recommended** that an external pullup/pulldown resistor be implemented. Although, internal pullup/pulldown resistors exist on these pins and they may match the desired configuration value, providing external connectivity can help ensure that valid logic levels are latched on these device boot and 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:

- Consider the total amount of current that may pass through the pullup or pulldown resistor. Make sure to include the leakage currents of all the devices connected to the net, as well as any internal pullup or pulldown resistors.
- Decide a target value for the net. For a pulldown resistor, this should be below the lowest V_{IL} level of all inputs connected to the net. For a pullup resistor, this should be above the highest V_{IH} level of all inputs on the net. A reasonable choice would be to target the V_{OL} or V_{OH} levels for the logic family of the limiting device; which, by definition, have margin to the V_{IL} and V_{IH} levels.
- Select a pullup/pulldown resistor with the largest possible value; but, which can still ensure that the net will reach the target pulled value when maximum current from all devices on the net is flowing through the resistor. The current to be considered includes leakage current plus, any other internal and external pullup/pulldown resistors on the net.
- For bidirectional nets, there is an additional consideration which sets a lower limit on the resistance value of the external resistor. Verify that the resistance is small enough that the weakest output buffer can drive the net to the opposite logic level (including margin).
- Remember to include tolerances when selecting the resistor value.
- For pullup resistors, also remember to include tolerances on the DV_{DD} rail.

For most systems, a 1-k Ω 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-k Ω resistor can be used to compliment 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}) for the C642x DSP, 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

On the C6424 device, the C64x+ Megamodule, the EDMA3 transfer controllers, and the system peripherals are interconnected through a switch fabric architecture (see [Figure 4-1](#)). The switch fabric is composed of multiple switched central resources (SCRs) and multiple bridges. The SCRs establish low-latency connectivity between master peripherals and slave peripherals. Additionally, the SCRs provide priority-based arbitration and facilitate concurrent data movement between master and slave peripherals. Through an SCR, the DSP can send data to the DDR2 Memory Controller without affecting a data transfer between the EMAC and L2 memory. Bridges are mainly used to perform bus-width conversion as well as bus operating frequency conversion. For example, in [Figure 4-1](#), Bridge 6 performs a frequency conversion between a bus operating at DSP/3 clock rate and a bus operating at DSP/6 clock rate. Furthermore, Bridge 5 performs a bus-width conversion between a 64-bit bus and a 32-bit bus.

The C64x+ Megamodule, the EDMA3 transfer controllers (EDMA3TC[2:0]), and the various system peripherals can be classified into two categories: master peripherals and slave peripherals. Master peripherals are typically capable of initiating read and write transfers in the system and **do not** rely on the EDMA3 or on the CPU to perform transfers to and from them. The system master peripherals include the C64x+ Megamodule, the EDMA3 transfer controllers, VLYNQ, EMAC, HPI, and PCI. Not all master peripherals may connect to all slave peripherals. The supported connections are designated by an Y in [Table 4-1](#).

Table 4-1. System Connection Matrix

| MASTER PERIPHERALS/MODULES | SLAVE PERIPHERALS/MODULES | | | | |
|------------------------------|---------------------------|------------------------|---------------------------|---------------------|---------------------------------------|
| | C64x+ SDMA | DDR2 MEMORY CONTROLLER | PCI (MASTER BACK-END I/F) | SCR4 ⁽¹⁾ | SCR2, SCR6, SCR7, SCR8 ⁽¹⁾ |
| C64x+ MDMA | – | Y | Y | – | Y |
| PCI (SLAVE BACK-END I/F) | Y | Y | – | Y | Y |
| VLYNQ | Y | Y | – | Y | Y |
| EMAC | Y | Y | – | Y | Y |
| HPI | Y | Y | – | Y | Y |
| EDMA3TC's (EDMA3TC2/TC1/TC0) | Y | Y | Y | Y | Y |
| C64x+ CFG | – | – | – | Y | Y |

(1) All the peripherals/modules that support a connection to SCR2, SCR4, SCR6, SCR7, and SCR8 have access to all peripherals/modules connected to those respective SCRs.

4.1 System Interconnect Block Diagram

[Figure 4-1](#) displays the C6424 system interconnect block diagram. The following is a list that helps in the interpretation of this diagram:

- The direction of the arrows indicates either a bus master *or* bus slave.
- The arrow originates at a bus master and terminates at a bus slave.
- The direction of the arrows **does not** indicate the direction of data flow. Data flow is typically bi-directional for each of the documented bus paths.
- The pattern of each arrow's line indicates the clock rate at which it is operating— i.e., either DSP/3, DSP/6, or MXI/CLKIN clock rate.
- A peripheral may have multiple instances shown in [Figure 4-1](#) for the following reason:
 - The peripheral/module has master port(s) for data transfers, as well as slave port(s) for register access, data access, and/or memory access. Examples of these peripherals are C64x+ Megamodule, EDMA3, VLYNQ, HPI, EMAC, and PCI.

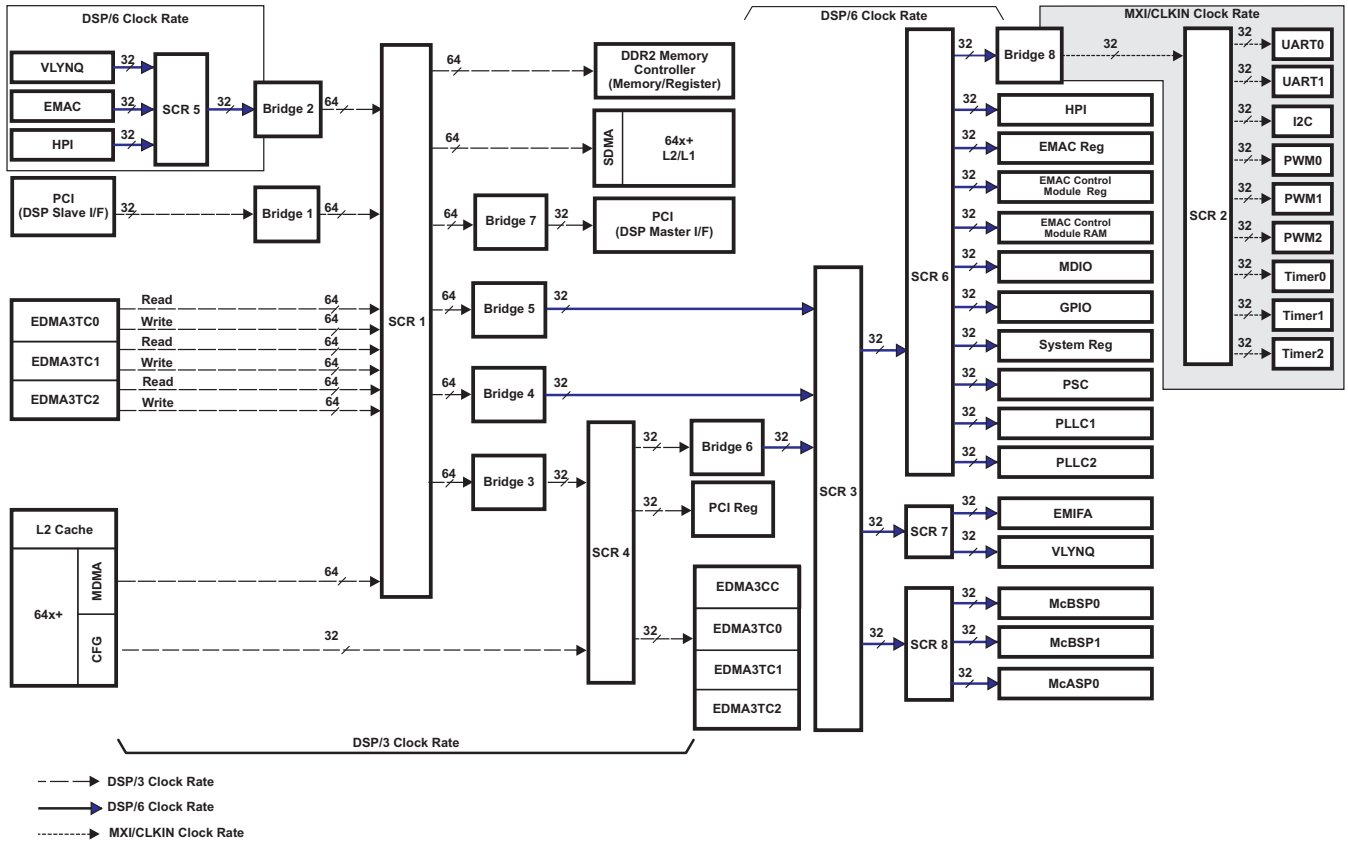


Figure 4-1. System Interconnect Block Diagram

5 Device Operating Conditions

5.1 Absolute Maximum Ratings Over Operating Temperature Range (Unless Otherwise Noted)⁽¹⁾

| | | |
|---|---|--------------------------------------|
| Supply voltage ranges: | Core (CV _{DD}) ⁽²⁾ | –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 _{DDR2} , DDR_VDDDLL, PLL_PWR18, MXV _{DD}) ⁽²⁾ | –0.5 to 2.5 V |
| Input voltage ranges: | V _I I/O, 3.3-V pins (except PCI-capable pins) | –0.5 V to 4.2 V |
| | V _I I/O, 3.3-V pins PCI-capable pins | –0.5 V to DV _{DD33} + 0.5 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 (except PCI-capable pins) | –0.5 V to 4.2 V |
| | V _O I/O, 3.3-V pins PCI-capable pins | –0.5 V to DV _{DD33} + 0.5 V |
| | V _O I/O, 1.8 V | –0.5 V to 2.5 V |
| Operating Junction temperature ranges, T _J : | Commercial | 0°C to 90°C |
| | Automotive (Q or S suffix) | –40°C to 125°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⁽¹⁾

| | | | MIN | NOM | MAX | UNIT | |
|------------------------------------|---|-----------------------------------|------------------------|-----------------------|------------------------|--------------------------|-----|
| CV _{DD} | Supply voltage, Core (CV _{DD}) ⁽²⁾ | (-7/-6/-5/-4/-Q6/-Q5/-Q4 devices) | 1.14 | 1.2 | 1.26 | V | |
| | | (-7/-6/-5/-4/-L/-Q5 devices) | 1.0 | 1.05 | 1.1 | V | |
| DV _{DD} | Supply voltage, I/O, 3.3V (DV _{DD33}) | | 2.97 | 3.3 | 3.63 | V | |
| | Supply voltage, I/O, 1.8V (DV _{DDR2} , DDR_VDDDLL, PLL_PWR18, MXV _{DD}) ⁽³⁾ | | 1.71 | 1.8 | 1.89 | V | |
| V _{SS} | Supply ground (V _{SS} , DDR_VSSDLL, MXV _{SS}) ⁽⁴⁾ | | 0 | 0 | 0 | V | |
| DDR_VREF | DDR2 reference voltage ⁽⁵⁾ | | 0.49DV _{DDR2} | 0.5DV _{DDR2} | 0.51DV _{DDR2} | V | |
| DDR_ZP | DDR2 impedance control, connected via 200 Ω resistor to V _{SS} | | V _{SS} | | | V | |
| DDR_ZN | DDR2 impedance control, connected via 200 Ω resistor to DV _{DDR2} | | DV _{DDR2} | | | V | |
| V _{IH} | High-level input voltage, 3.3V (except PCI-capable and I2C pins) | | 2 | | | V | |
| | High-level input voltage, MXI/ CLKIN | | 0.65MXV _{DD} | | | V | |
| | High-level input voltage, PCI | | 0.5DV _{DD33} | | | DV _{DD33} + 0.5 | V |
| | High-level input voltage, I2C | | 0.7DV _{DD33} | | | | |
| V _{IL} | Low-level input voltage, 3.3V (except PCI-capable and I2C pins) | | | | | 0.8 | V |
| | Low-level input voltage, MXI/ CLKIN | | | | | 0.35MXV _{DD} | V |
| | Low-level input voltage, PCI | | -0.5 | | | 0.3DV _{DD33} | V |
| | Low-level input voltage, I2C | | 0 | | | 0.3DV _{DD33} | V |
| T _J | Operating Junction temperature ^{(6) (7)} | Commercial | 0 | | | 90 | °C |
| | | Automotive (Q or S suffix) | -40 | | | 125 | °C |
| T _A | Operating Ambient temperature ⁽⁷⁾ | Commercial | 0 | | | 70 | °C |
| | | Automotive (Q or S suffix) | -40 | | | 85 | °C |
| F _{SYCLK1} ⁽²⁾ | DSP Operating Frequency (SYCLK1), CV _{DD} = 1.2 V | -7 devices | | | | 700 | MHz |
| | | -6/-Q6 devices | | | | 600 | MHz |
| | | -5/-Q5 devices | | | | 500 | MHz |
| | | -4/-Q4 devices | | | | 400 | MHz |
| | DSP Operating Frequency (SYCLK1), CV _{DD} = 1.05 V | -7 devices | | | | 560 | MHz |
| | | -6 devices | | | | 450 | MHz |
| | | -5/-Q5/-L devices | | | | 400 | MHz |
| | | -4 devices | | | | 350 | MHz |

- (1) The actual voltage *must* be determined at device power-up, and **not** be changed dynamically during run-time.
- (2) Applies to "tape and reel" part number counterparts as well. For more information, see [Section 2.7](#), *Device and Development-Support Tool Nomenclature*.
- (3) Oscillator 1.8 V power supply (MXV_{DD}) can be connected to the same 1.8 V power supply as DV_{DDR2}.
- (4) Oscillator ground (MXV_{SS}) must be kept separate from other grounds and connected directly to the crystal load capacitor ground.
- (5) DDR_VREF is expected to equal 0.5DV_{DDR2} of the transmitting device and to track variations in the DV_{DDR2}.
- (6) In the absence of a heat sink or direct thermal attachment on the top of the device, use the following formula to determine the device junction temperature: $T_J = T_C + (\text{Power} \times \text{Psi}_{JT})$. Power and T_C can be measured by the user. [Section 7.1](#), *Thermal Data for ZWT and Section 7.2*, *Thermal Data for ZDU* provide the junction-to-package top (Psi_{JT}) value based on airflow in the system. In the presence of a heat sink or direct thermal attachment on the top of the device, additional calculations and considerations **must** be taken into account. For more detailed information on thermal considerations, measurements, and calculations, see the *Thermal Considerations for the DM64xx, DM64x, and C6000 Devices* Application Report (literature number [SPRAAL9](#)).
- (7) Applications must meet **both** the Operating Junction Temperature and Operating Ambient Temperature requirements. For more detailed information on thermal considerations, measurements, and calculations, see the *Thermal Considerations for the DM64xx, DM64x, and C6000 Devices* Application Report (literature number [SPRAAL9](#)).

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 (3.3V I/O except PCI-capable and I2C pins) | DV _{DD33} = MIN, I _{OH} = MAX | 2.4 | | | V |
| | High-level output voltage (3.3V I/O PCI-capable pins) | I _{OH} = -0.5 mA, DV _{DD33} = 3.3 V | 0.9DV _{DD33} ⁽²⁾ | | | V |
| V _{OL} | Low-level output voltage (3.3V I/O except PCI-capable and I2C pins) | DV _{DD33} = MIN, I _{OL} = MAX | | | 0.4 | V |
| | Low-level output voltage (3.3V I/O PCI-capable pins) | I _{OH} = 1.5 mA, DV _{DD33} = 3.3 V | | 0.1DV _{DD33} ⁽²⁾ | | V |
| | Low-level output voltage (3.3V I/O I2C pins) | I _O = 3 mA | 0 | | 0.4 | V |
| I _I ⁽³⁾ | Input current [DC] (except I2C and PCI capable pins) | V _I = V _{SS} to DV _{DD33} with internal pullup resistor ⁽⁴⁾ | 50 | 100 | 250 | μA |
| | | V _I = V _{SS} to DV _{DD33} with internal pulldown resistor ⁽⁴⁾ | -250 | -100 | -50 | μA |
| | Input current [DC] (I2C) | V _I = V _{SS} to DV _{DD33} | | | ±10 | μA |
| | Input current (PCI capable pins) [DC] ⁽⁵⁾ | 0 < V _I < DV _{DD33} = 3.3 V without internal resistor | | | ±50 | μA |
| | | 0 < V _I < DV _{DD33} = 3.3 V with internal pullup resistor ⁽⁴⁾ | 50 | | 250 | μA |
| 0 < V _I < DV _{DD33} = 3.3 V with internal pulldown resistor ⁽⁴⁾ | | -250 | | -50 | μA | |
| I _{OH} | High-level output current [DC] | CLK_OUT0 /PWM2/GPIO[84] and VLYNQ_CLOCK/PCICLK/GP[57] | | | -8 | mA |
| | | DDR2 | | | -13.4 | mA |
| | | PCI-capable pins | | | -0.5 ⁽²⁾ | mA |
| | | All other peripherals | | | -4 | mA |
| I _{OL} | Low-level output current [DC] | CLK_OUT0 /PWM2/GPIO[84] and VLYNQ_CLOCK/PCICLK/GP[57] | | | 8 | mA |
| | | DDR2 | | | 13.4 | mA |
| | | PCI-capable pins | | | 1.5 ⁽²⁾ | mA |
| | | All other peripherals | | | 4 | mA |
| I _{OZ} ⁽⁶⁾ | I/O Off-state output current | V _O = DV _{DD33} or V _{SS} ; internal pull disabled | | | ±50 | μA |
| | | V _O = DV _{DD33} or V _{SS} ; internal pull enabled | | ±100 | | μA |
| I _{CDD} | Core (CV _{DD}) supply current ⁽⁷⁾ | CV _{DD} = 1.2 V, DSP clock = 700 MHz | | 597 | | mA |
| | | CV _{DD} = 1.2 V, DSP clock = 600 MHz | | 524 | | mA |
| | | CV _{DD} = 1.2 V, DSP clock = 500 MHz | | 460 | | mA |
| | | CV _{DD} = 1.2 V, DSP clock = 400 MHz | | 392 | | mA |
| | | CV _{DD} = 1.05 V, DSP clock = 560 MHz | | 442 | | mA |
| | | CV _{DD} = 1.05 V, DSP clock = 450 MHz | | 372 | | mA |
| | | CV _{DD} = 1.05 V, DSP clock = 400 MHz | | 341 | | mA |

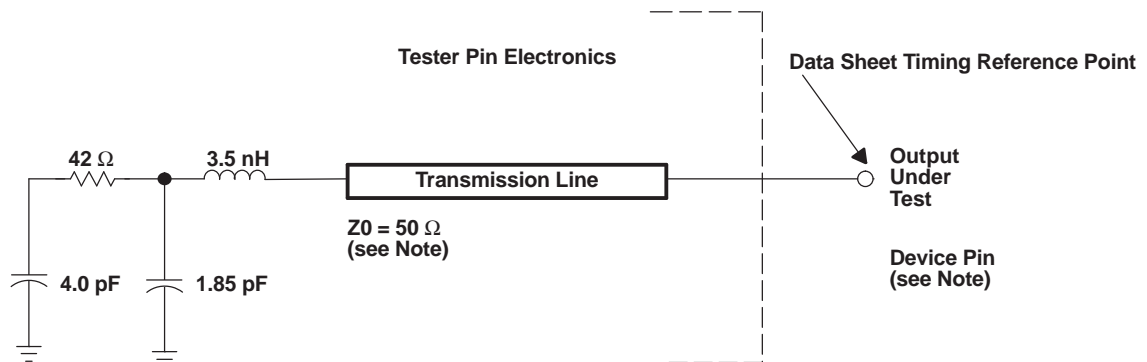
- (1) For test conditions shown as MIN, MAX, or NOM, use the appropriate value specified in the recommended operating conditions table.
- (2) These rated numbers are from the *PCI Local Bus Specification Revision 2.3*. The DC specifications and AC specifications are defined in Table 4-3 (DC Specifications for 3.3V Signaling) and Table 4-4 (AC Specifications for 3.3V Signaling), respectively.
- (3) I_I applies to input-only pins and bi-directional pins. For input-only pins, I_I indicates the input leakage current. For bi-directional pins, I_I indicates the input leakage current and off-state (Hi-Z) output leakage current.
- (4) Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.
- (5) PCI input leakage currents include Hi-Z output leakage for all bidirectional buffers with 3-state outputs.
- (6) I_{OZ} applies to output-only pins, indicating off-state (Hi-Z) output leakage current.
- (7) Measured under the following conditions: 60% DSP CPU utilization doing typical activity (peripheral configurations, other housekeeping activities); DDR2 Memory Controller at 50% utilization (135 MHz), 50% writes, 32 bits, 50% bit switching; 2 MHz McBSP0 at 100% utilization and 50% switching; Timer0 at 100% utilization. At room temperature (25 °C) for typical process ZWT devices. The actual current draw varies across manufacturing processes and is highly application-dependent. C642x DSP devices are offered in two basic options: lower-power option and high-performance option. Low-power devices offer lower power consumption across temperature and voltage when compared with high-performance devices. However, high-performance devices offer higher operating speeds. For more details on core and I/O activity, as well as information relevant to board power supply design, see the *TMS320C642x Power Consumption Summary* Application Report (literature number [SPRAA09](#)).

**Electrical Characteristics Over Recommended Ranges of Supply Voltage and Operating Temperature
(Unless Otherwise Noted) (continued)**

| PARAMETER | | TEST CONDITIONS ⁽¹⁾ | MIN | TYP | MAX | UNIT |
|-----------------|--|--|-----|-----|-----|------|
| I _{DD} | 3.3V I/O (DV _{DD33}) supply current ⁽⁷⁾ | DV _{DD} = 3.3 V, CV _{DD} = 1.2 V, DSP clock = 700 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.2 V, DSP clock = 600 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.2 V, DSP clock = 500 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.2 V, DSP clock = 400 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.05 V, DSP clock = 560 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.05 V, DSP clock = 450 MHz | | 13 | | mA |
| | | DV _{DD} = 3.3 V, CV _{DD} = 1.05 V, DSP clock = 400 MHz | | 13 | | mA |
| I _{DD} | 1.8V I/O (DV _{DDR2} , DDR_VDDDLL, PLLV _{PRW18} , MXV _{DD}) supply current ⁽⁷⁾ | DV _{DD} = 1.8 V, CV _{DD} = 1.2 V, DSP clock = 700 MHz | | 94 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.2 V, DSP clock = 600 MHz | | 93 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.2 V, DSP clock = 500 MHz | | 92 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.2 V, DSP clock = 400 MHz | | 91 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.05 V, DSP clock = 560 MHz | | 74 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.05 V, DSP clock = 450 MHz | | 73 | | mA |
| | | DV _{DD} = 1.8 V, CV _{DD} = 1.05 V, DSP clock = 400 MHz | | 72 | | mA |
| C _I | Input capacitance | | | | 5 | pF |
| C _O | Output capacitance | | | | 5 | pF |

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.

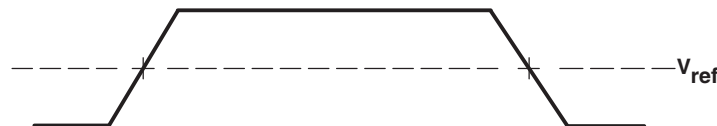


Figure 6-2. Input and Output Voltage Reference Levels for AC Timing Measurements

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.

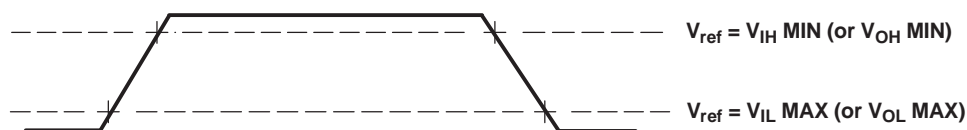


Figure 6-3. Rise and Fall Transition Time Voltage Reference Levels

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 sheet do *not* include delays by board routings. As a good board design practice, such delays must *always* be taken into account. Timing values may be

adjusted by increasing/decreasing such delays. TI recommends utilizing the available I/O buffer information specification (IBIS) models to analyze the timing characteristics correctly. To properly use IBIS models to attain accurate timing analysis for a given system, see the *Using IBIS Models for Timing Analysis* application report (literature number SPRA839). If needed, external logic hardware such as buffers may be used to compensate any timing differences.

For the DDR2 memory controller interface, it is *not* necessary to use the IBIS models to analyze timing characteristics. TI provides a PCB routing rules solution that describes the routing rules to ensure the DDR2 memory controller interface timings are met. See the *Implementing DDR2 PCB Layout on the TMS320C6421/4 DMSoC* Application Report (literature number [SPRAAL7](#)).

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 C6424 includes one core supply (CV_{DD}), and two I/O supplies— DV_{DD33} and DV_{DDR2} . 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.

Here is a summary of the power sequencing requirements:

- The power ramp order **must** be DV_{DD33} before DV_{DDR2} , and DV_{DDR2} before CV_{DD} —meaning during power up, the voltage at the DV_{DDR2} rail should never exceed the voltage at the DV_{DD33} rail. Similarly, the voltage at the CV_{DD} rail should never exceed the voltage at the DV_{DDR2} rail.
- From the time that power ramp begins, all power supplies (DV_{DD33} , DV_{DDR2} , CV_{DD}) **must** be stable within 200 ms. The term "stable" means reaching the recommended operating condition (see [Section 5.2, Recommended Operating Conditions](#) table).

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 C6424 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 μ F) should be furthest away, but still as close as possible. Large caps for each supply should be placed outside of the BGA footprint.

As with the selection of any component, verification of capacitor availability over the product's production lifetime should be considered.

For more details on capacitor usage and placement, see the *Implementing DDR2 PCB Layout on the TMS320C642x DSP* Application Report (literature number [SPRAAK5](#)).

6.3.4 C6424 Power and Clock Domains

The C6424 includes one single power domain — the "Always On" power domain. The "Always On" power domain is always on when the chip is on. The "Always On" domain is powered by the CV_{DD} pins of the C6424. All C6424 modules lie within the "Always On" power domain. [Table 6-1](#) provides a listing of the C6424 clock domains.

One primary reference clock is required for the C6424 device. It can be either a crystal input or driven by external oscillators. A 15–30-MHz crystal is recommended for the PLLs, which generate the internal clocks for the digital signal processor (DSP), peripherals, and the EDMA3. For further description of the C6424 clock domains, see [Table 6-2](#) and [Figure 6-4](#).

The C6424 architecture is divided into the power and clock domains shown in [Table 6-1](#). [Table 6-2](#) and [Table 6-3](#) further discuss the clock domains and their ratios. [Figure 6-4](#) shows the clock domain block diagram.

Table 6-1. C6424 Power and Clock Domains

| Power Domain | Clock Domain | Peripheral/Module |
|--------------|--------------|-------------------|
| Always On | CLKIN | UART0 |
| Always On | CLKIN | UART1 |
| Always On | CLKIN | I2C |
| Always On | CLKIN | Timer0 |
| Always On | CLKIN | Timer1 |
| Always On | CLKIN | Timer2 |
| Always On | CLKIN | PWM0 |
| Always On | CLKIN | PWM1 |
| Always On | CLKIN | PWM2 |
| Always On | CLKDIV3 | DDR2 |
| Always On | CLKDIV3 | EDMA |
| Always On | CLKDIV3 | PCI |
| Always On | CLKDIV3 | SCR |
| Always On | CLKDIV6 | GPSC |
| Always On | CLKDIV6 | LPSCs |
| Always On | CLKDIV6 | PLL1 |
| Always On | CLKDIV6 | PLL2 |
| Always On | CLKDIV6 | Ice Pick |
| Always On | CLKDIV6 | EMIFA |
| Always On | CLKDIV6 | HPI |
| Always On | CLKDIV6 | VLYNQ |
| Always On | CLKDIV6 | EMAC |
| Always On | CLKDIV6 | McASP0 |
| Always On | CLKDIV6 | McBSP0 |
| Always On | CLKDIV6 | McBSP1 |
| Always On | CLKDIV6 | GPIO |
| Always On | CLKDIV1 | C64x+ CPU |

Table 6-2. C6424 Clock Domains

| SUBSYSTEM | CLOCK DOMAIN | DOMAIN CLOCK SOURCE | FIXED RATIO vs. SYSCLK1 FREQUENCY | EXAMPLE FREQUENCY (MHz) |
|------------------------------|--------------|----------------------------|-----------------------------------|-------------------------|
| Peripherals (CLKIN Domain) | CLKIN | PLL1 AUXCLK ⁽¹⁾ | – | 25 MHz |
| DSP | CLKDIV1 | PLL1 SYSCLK1 | 1:1 | 600 MHz |
| EDMA3 | CLKDIV3 | PLL1 SYSCLK2 | 1:3 | 200 MHz |
| Peripherals (CLKDIV3 Domain) | CLKDIV3 | PLL1 SYSCLK2 | 1:3 | 200 MHz |
| Peripherals (CLKDIV6 Domain) | CLKDIV6 | PLL1 SYSCLK3 | 1:6 | 100 MHz |

(1) PLL1 AUXCLK runs at exactly the same frequency as the device clock source from the MXI/CLKIN pin.

The CLKDIV1:CLKDIV3:CLKDIV6 ratio must be strictly followed by programming the PLL Controller 1 (PLL1) PLLDIV1, PLLDIV2, and PLLDIV3 registers appropriately (see [Table 6-3](#)).

Table 6-3. PLL1 Programming for CLKDIV1, CLKDIV3, CLKDIV6 Domains

| | CLKDIV1 DOMAIN (SYSCLK1) | | CLKDIV3 DOMAIN (SYSCLK2) | | CLKDIV6 DOMAIN (SYSCLK3) | |
|------|--------------------------|---------------|--------------------------|---------------|--------------------------|---------------|
| | PLL1 Divide-Down | PLLDIV1.RATIO | PLL1 Divide-Down | PLLDIV2.RATIO | PLL1 Divide-Down | PLLDIV3.RATIO |
| DIV1 | /1 | 0 | /3 | 2 | /6 | 5 |
| DIV2 | /2 | 1 | /6 | 5 | /12 | 11 |
| DIV3 | /3 | 2 | /9 | 8 | /18 | 17 |

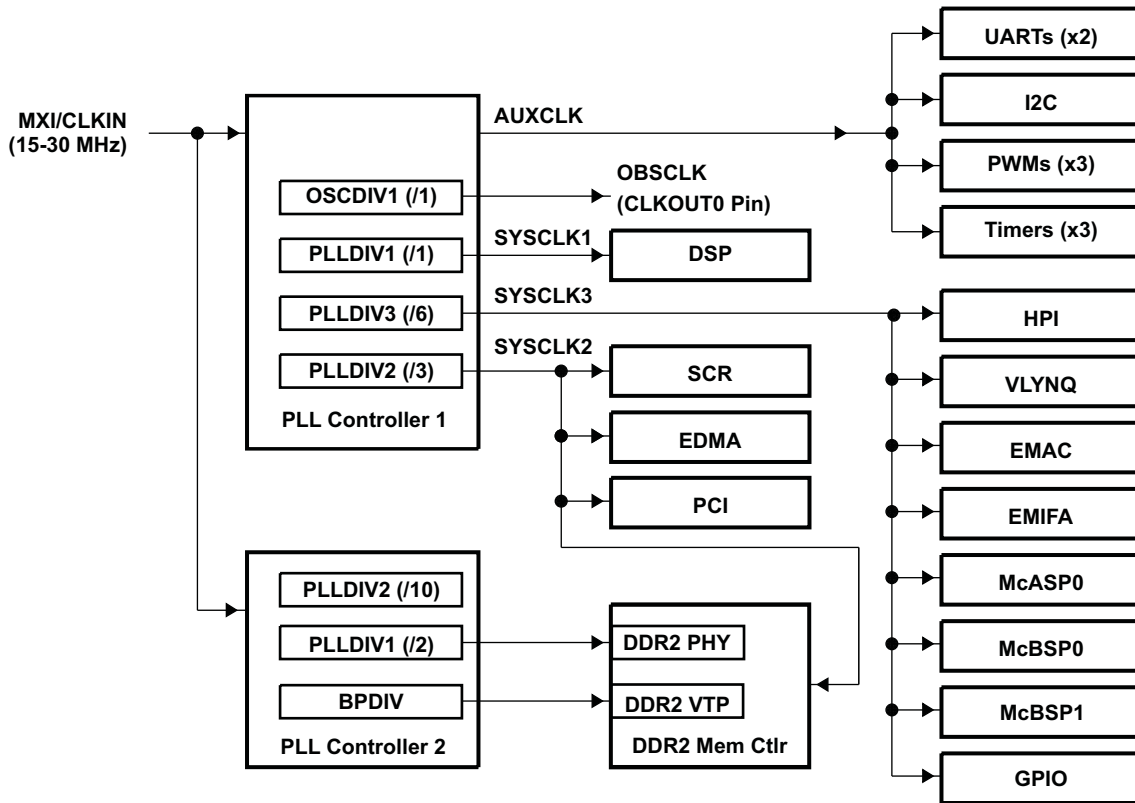


Figure 6-4. PLL1 and PLL2 Clock Domain Block Diagram

For further detail on PLL1 and PLL2, see the structure block diagrams [Figure 6-5](#) and [Figure 6-6](#), respectively.

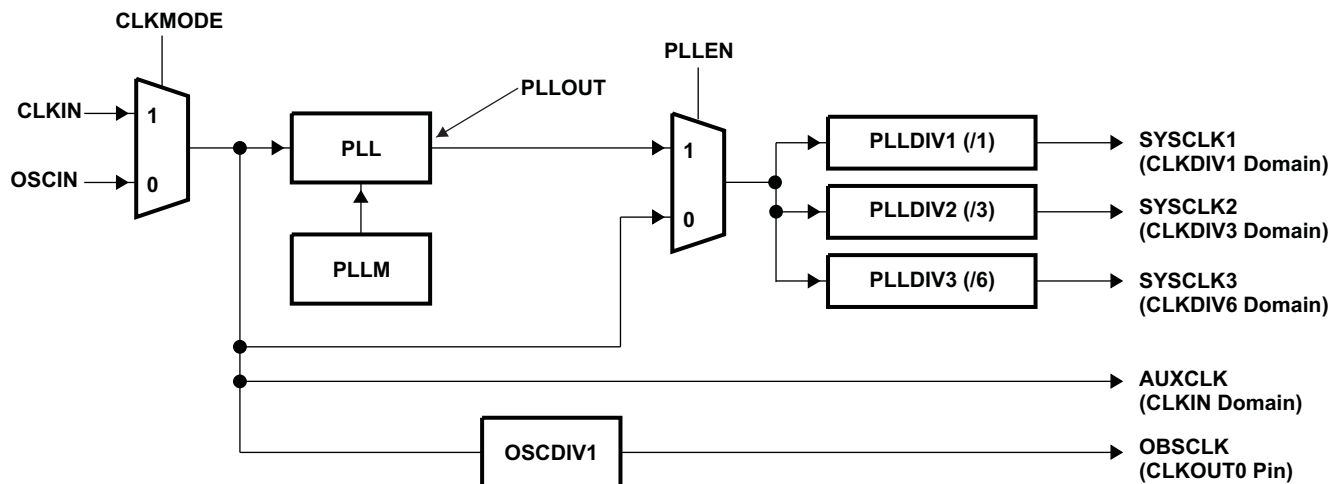


Figure 6-5. PLL1 Structure Block Diagram

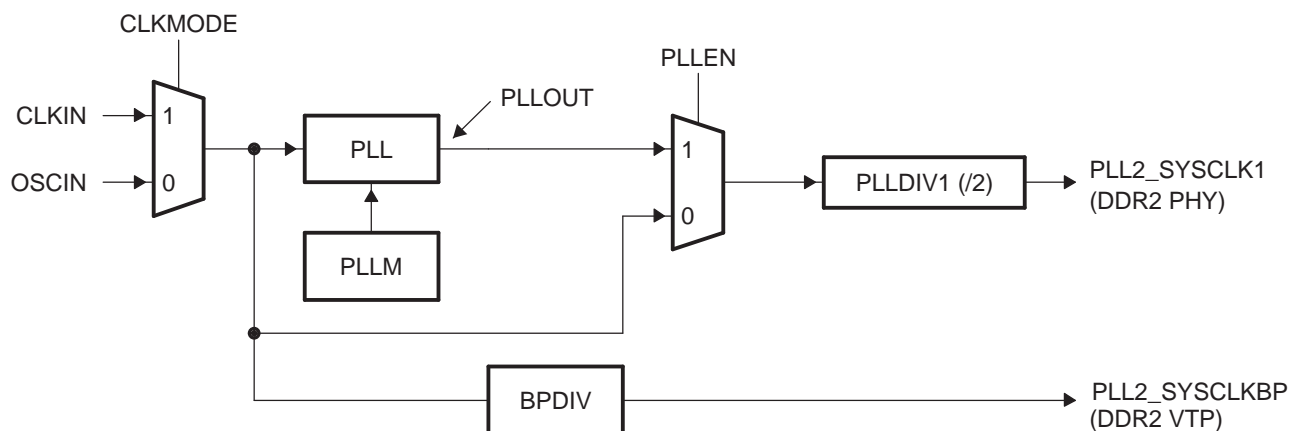


Figure 6-6. PLL2 Structure Block Diagram

6.3.5 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 C6424 device only utilizes the clock gating feature of the PSC 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 for C6424 are shown in [Table 6-4](#). The PSC Register memory map is given in [Table 6-5](#). For more details on the PSC, see the *TMS320C642x Power and Sleep Controller (PSC) Reference Guide* (literature number SPRUEN8).

Table 6-4. C6424 LPSC Assignments

| LPSC Number | Peripheral/Module | LPSC Number | Peripheral/Module | LPSC Number | Peripheral/Module |
|-------------|------------------------|-------------|-------------------|-------------|-------------------|
| 0 | Reserved | 14 | EMIFA | 28 | TIMER1 |
| 1 | Reserved | 15 | PCI | 29 | Reserved |
| 2 | EDMACC | 16 | McBSP0 | 30 | Reserved |
| 3 | EDMATC0 | 17 | McBSP1 | 31 | Reserved |
| 4 | EDMATC1 | 18 | I2C | 32 | Reserved |
| 5 | EDMATC2 | 19 | UART0 | 33 | Reserved |
| 6 | EMAC Memory Controller | 20 | UART1 | 34 | Reserved |
| 7 | MDIO | 21 | Reserved | 35 | Reserved |
| 8 | EMAC | 22 | Reserved | 36 | Reserved |
| 9 | McASP0 | 23 | PWM0 | 37 | Reserved |
| 10 | Reserved | 24 | PWM1 | 38 | Reserved |
| 11 | VLYNQ | 25 | PWM2 | 39 | C64x+ CPU |
| 12 | HPI | 26 | GPIO | 40 | Reserved |
| 13 | DDR2 Memory Controller | 27 | TIMER0 | | |

Table 6-5. PSC Register Memory Map

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---------------------------|------------------|--|
| 0x01C4 1000 | PID | Peripheral Revision and Class Information Register |
| 0x01C4 1004 - 0x01C4 100F | - | Reserved |
| 0x01C4 1010 | - | Reserved |
| 0x01C4 1014 | - | Reserved |
| 0x01C4 1018 | INTEVAL | Interrupt Evaluation Register |
| 0x01C4 101C - 0x01C4 103F | - | Reserved |
| 0x01C4 1040 | - | Reserved |
| 0x01C4 1044 | MERRPR1 | Module Error Pending 1 (mod 32- 63) Register |
| 0x01C4 1048 - 0x01C4 104F | - | Reserved |
| 0x01C4 1050 | - | Reserved |
| 0x01C4 1054 | MERRCR1 | Module Error Clear 1 (mod 32 - 63) Register |
| 0x01C4 1058 - 0x01C4 105F | - | Reserved |
| 0x01C4 1060 | - | Reserved |
| 0x01C4 1064 - 0x01C4 1067 | - | Reserved |
| 0x01C4 1068 | - | Reserved |
| 0x01C4 106C - 0x01C4 111F | - | Reserved |
| 0x01C4 1120 | PTCMD | Power Domain Transition Command Register |
| 0x01C4 1124 - 0x01C4 1127 | - | Reserved |
| 0x01C4 1128 | PTSTAT | Power Domain Transition Status Register |
| 0x01C4 112C - 0x01C4 11FF | - | Reserved |
| 0x01C4 1200 | PDSTAT0 | Power Domain Status 0 Register (Always On) |
| 0x01C4 1204 - 0x01C4 12FF | - | Reserved |
| 0x01C4 1300 | PDCTL0 | Power Domain Control 0 Register (Always On) |
| 0x01C4 1304 - 0x1C4 150F | - | Reserved |
| 0x01C4 1510 | - | Reserved |
| 0x01C4 1514 | - | Reserved |
| 0x01C4 1518 - 0x01C4 15FF | - | Reserved |
| 0x01C4 1600 - 0x01C4 17FF | - | Reserved |

Table 6-5. PSC Register Memory Map (continued)

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---------------------------|------------------|--|
| 0x01C4 1800 | - | Reserved |
| 0x01C4 1804 | - | Reserved |
| 0x01C4 1808 | MDSTAT2 | Module Status 2 Register (EDMACC) |
| 0x01C4 180C | MDSTAT3 | Module Status 3 Register (EDMATC0) |
| 0x01C4 1810 | MDSTAT4 | Module Status 4 Register (EDMATC1) |
| 0x01C4 1814 | MDSTAT5 | Module Status 5 Register (EDMATC2) |
| 0x01C4 1818 | MDSTAT6 | Module Status 6 Register (EMAC Memory Controller) |
| 0x01C4 181C | MDSTAT7 | Module Status 7 Register (MDIO) |
| 0x01C4 1820 | MDSTAT8 | Module Status 8 Register (EMAC) |
| 0x01C4 1824 | MDSTAT9 | Module Status 9 Register (McASP0) |
| 0x01C4 1828 | - | Reserved |
| 0x01C4 182C | MDSTAT11 | Module Status 11 Register (VLYNQ) |
| 0x01C4 1830 | MDSTAT12 | Module Status 12 Register (HPI) |
| 0x01C4 1834 | MDSTAT13 | Module Status 13 Register (DDR2) |
| 0x01C4 1838 | MDSTAT14 | Module Status 14 Register (EMIFA) |
| 0x01C4 183C | MDSTAT15 | Module Status 15 Register (PCI) |
| 0x01C4 1840 | MDSTAT16 | Module Status 16 Register (McBSP0) |
| 0x01C4 1844 | MDSTAT17 | Module Status 17 Register (McBSP1) |
| 0x01C4 1848 | MDSTAT18 | Module Status 18 Register (I2C) |
| 0x01C4 184C | MDSTAT19 | Module Status 19 Register (UART0) |
| 0x01C4 1850 | MDSTAT20 | Module Status 20 Register (UART1) |
| 0x01C4 1854 | - | Reserved |
| 0x01C4 1858 | - | Reserved |
| 0x01C4 185C | MDSTAT23 | Module Status 23 Register (PWM0) |
| 0x01C4 1860 | MDSTAT24 | Module Status 24 Register (PWM1) |
| 0x01C4 1864 | MDSTAT25 | Module Status 25 Register (PWM2) |
| 0x01C4 1868 | MDSTAT26 | Module Status 26 Register (GPIO) |
| 0x01C4 186C | MDSTAT27 | Module Status 27 Register (TIMER0) |
| 0x01C4 1870 | MDSTAT28 | Module Status 28 Register (TIMER1) |
| 0x01C4 1874 - 0x01C4 189B | - | Reserved |
| 0x01C4 189C | MDSTAT39 | Module Status 39 Register (C64x+ CPU) |
| 0x01C4 18A0 | - | Reserved |
| 0x01C4 18A4 - 0x01C4 19FF | - | Reserved |
| 0x01C4 1A00 | - | Reserved |
| 0x01C4 1A04 | - | Reserved |
| 0x01C4 1A08 | MDCTL2 | Module Control 2 Register (EDMACC) |
| 0x01C4 1A0C | MDCTL3 | Module Control 3 Register (EDMATC0) |
| 0x01C4 1A10 | MDCTL4 | Module Control 4 Register (EDMATC1) |
| 0x01C4 1A14 | MDCTL5 | Module Control 5 Register (EDMATC2) |
| 0x01C4 1A18 | MDCTL6 | Module Control 6 Register (EMAC Memory Controller) |
| 0x01C4 1A1C | MDCTL7 | Module Control 7 Register (MDIO) |
| 0x01C4 1A20 | MDCTL8 | Module Control 8 Register (EMAC) |
| 0x01C4 1A24 | MDCTL9 | Module Control 9 Register (McASP0) |
| 0x01C4 1A28 | - | Reserved |
| 0x01C4 1A2C | MDCTL11 | Module Control 11 Register (VLYNQ) |
| 0x01C4 1A30 | MDCTL12 | Module Control 12 Register (HPI) |

Table 6-5. PSC Register Memory Map (continued)

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|---------------------------|------------------|--|
| 0x01C4 1A34 | MDCTL13 | Module Control 13 Register (DDR2) |
| 0x01C4 1A38 | MDCTL14 | Module Control 14 Register (EMIFA) |
| 0x01C4 1A3C | MDCTL15 | Module Control 15 Register (PCI) |
| 0x01C4 1A40 | MDCTL16 | Module Control 16 Register (McBSP0) |
| 0x01C4 1A44 | MDCTL17 | Module Control 17 Register (McBSP1) |
| 0x01C4 1A48 | MDCTL18 | Module Control 18 Register (I2C) |
| 0x01C4 1A4C | MDCTL19 | Module Control 19 Register (UART0) |
| 0x01C4 1A50 | MDCTL20 | Module Control 20 Register (UART1) |
| 0x01C4 1A54 | - | Reserved |
| 0x01C4 1A58 | - | Reserved |
| 0x01C4 1A5C | MDCTL23 | Module Control 23 Register (PWM0) |
| 0x01C4 1A60 | MDCTL24 | Module Control 24 Register (PWM1) |
| 0x01C4 1A64 | MDCTL25 | Module Control 25 Register (PWM2) |
| 0x01C4 1A68 | MDCTL26 | Module Control 26 Register (GPIO) |
| 0x01C4 1A6C | MDCTL27 | Module Control 27 Register (TIMER0) |
| 0x01C4 1A70 | MDCTL28 | Module Control 28 Register (TIMER1) |
| 0x01C4 1A74 - 0x01C4 1A9B | - | Reserved |
| 0x01C4 1A9C | MDCTL39 | Module Control 39 Register (C64x+ CPU) |
| 0x01C4 1AA0 | - | Reserved |
| 0x01C4 1AA4 - 0x01C4 1FFF | - | Reserved |

6.4 Enhanced Direct Memory Access (EDMA3) Controller

The EDMA controller handles all data transfers between memories and the device slave peripherals on the C6424 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
 - NAND flash
 - Asynchronous EMIF (EMIFA)
- Transfer to/from peripherals/hosts
 - VLYNQ
 - HPI
 - McBSP0/1
 - McASP0
 - PWM
 - UART0/1
 - PCI

The EDMA supports two addressing modes: constant addressing and increment addressing. On the C6424, constant addressing mode is **not** supported by any peripheral or internal memory. For more information on these two addressing modes, see the *TMS320C642x DSP Enhanced DMA (EDMA) Controller User's Guide* (literature number SPRUEM5).

6.4.1 EDMA3 Channel Synchronization Events

The EDMA supports up to 64 EDMA channels which service peripheral devices and external memory. [Table 6-6](#) lists the source of EDMA synchronization events associated with each of the programmable EDMA channels. For the C6424 device, the association of an event to a channel is fixed; each of the EDMA channels has one specific event associated with it. These specific events are captured in the EDMA event registers (ER, ERH) even if the events are disabled by the EDMA event enable registers (EER, EERH). For more detailed information on the EDMA module and how EDMA events are enabled, captured, processed, linked, chained, and cleared, etc., see the *TMS320C642x DSP Enhanced DMA (EDMA) Controller User's Guide* (literature number SPRUEM5).

Table 6-6. C6424 EDMA Channel Synchronization Events⁽¹⁾

| EDMA CHANNEL | EVENT NAME | EVENT DESCRIPTION |
|--------------|------------|----------------------------|
| 0-1 | – | Reserved |
| 2 | XEVT0 | McBSP0 Transmit Event |
| 3 | REVT0 | McBSP0 Receive Event |
| 4 | XEVT1 | McBSP1 Transmit Event |
| 5 | REVT1 | McBSP1 Receive Event |
| 6 | – | Reserved |
| 7 | – | Reserved |
| 8 | – | Reserved |
| 9 | – | Reserved |
| 10 | AXEVTE0 | McASP0 Transmit Event Even |
| 11 | AXEVTO0 | McASP0 Transmit Event Odd |
| 12 | AXEVT0 | McASP0 Transmit Event |
| 13 | AREVTE0 | McASP0 Receive Event Even |
| 14 | AREVTO0 | McASP0 Receive Event Odd |
| 15 | AREVT0 | McASP0 Receive Event |
| 16-21 | – | Reserved |
| 22 | URXEVT0 | UART 0 Receive Event |
| 23 | UTXEVT0 | UART 0 Transmit Event |
| 24 | URXEVT1 | UART 1 Receive Event |
| 25 | UTXEVT1 | UART 1 Transmit Event |
| 26 | – | Reserved |
| 27 | – | Reserved |
| 28 | ICREVT | I2C Receive Event |
| 29 | ICXEVT | I2C Transmit Event |
| 30-31 | – | Reserved |
| 32 | GPINT0 | GPIO 0 Interrupt |
| 33 | GPINT1 | GPIO 1 Interrupt |
| 34 | GPINT2 | GPIO 2 Interrupt |
| 35 | GPINT3 | GPIO 3 Interrupt |
| 36 | GPINT4 | GPIO 4 Interrupt |
| 37 | GPINT5 | GPIO 5 Interrupt |
| 38 | GPINT6 | GPIO 6 Interrupt |
| 39 | GPINT7 | GPIO 7 Interrupt |
| 40 | GPBKINT0 | GPIO Bank 0 Interrupt |
| 41 | GPBKINT1 | GPIO Bank 1 Interrupt |

(1) In addition to the events shown in this table, each of the 64 channels can also be synchronized with the transfer completion or alternate transfer completion events. For more detailed information on EDMA event-transfer chaining, see the *TMS320C642x DSP Enhanced DMA (EDMA) Controller User's Guide* (literature number SPRUEM5).

Table 6-6. C6424 EDMA Channel Synchronization Events ⁽¹⁾ (continued)

| EDMA CHANNEL | EVENT NAME | EVENT DESCRIPTION |
|--------------|------------|------------------------------|
| 42 | GPBNKINT2 | GPIO Bank 2 Interrupt |
| 43 | GPBNKINT3 | GPIO Bank 3 Interrupt |
| 44 | GPBNKINT4 | GPIO Bank 4 Interrupt |
| 45 | GPBNKINT5 | GPIO Bank 5 Interrupt |
| 46 | GPBNKINT6 | GPIO Bank 6 Interrupt |
| 47 | – | Reserved |
| 48 | TEVTL0 | Timer 0 Event Low Interrupt |
| 49 | TEVTH0 | Timer 0 Event High Interrupt |
| 50 | TEVTL1 | Timer 1 Event Low Interrupt |
| 51 | TEVTH1 | Timer 1 Event High Interrupt |
| 52 | PWM0 | PWM 0 Event |
| 53 | PWM1 | PWM 1 Event |
| 54 | PWM2 | PWM 2 Event |
| 55-63 | – | Reserved |

6.4.2 EDMA Peripheral Register Description(s)

Table 6-7 lists the EDMA registers, their corresponding acronyms, and C6424 device memory locations.

Table 6-7. C6424 EDMA Registers

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|-------------------------------------|----------|---|
| Channel Controller Registers | | |
| 0x01C0 0000 - 0x01C0 0003 | | Reserved |
| 0x01C0 0004 | CCCFG | EDMA3CC Configuration Register |
| 0x01C0 0008 - 0x01C0 01FF | | Reserved |
| Global Registers | | |
| 0x01C0 0200 | QCHMAP0 | QDMA Channel 0 Mapping to PaRAM Register |
| 0x01C0 0204 | QCHMAP1 | QDMA Channel 1 Mapping to PaRAM Register |
| 0x01C0 0208 | QCHMAP2 | QDMA Channel 2 Mapping to PaRAM Register |
| 0x01C0 020C | QCHMAP3 | QDMA Channel 3 Mapping to PaRAM Register |
| 0x01C0 0210 | QCHMAP4 | QDMA Channel 4 Mapping to PaRAM Register |
| 0x01C0 0214 | QCHMAP5 | QDMA Channel 5 Mapping to PaRAM Register |
| 0x01C0 0218 | QCHMAP6 | QDMA Channel 6 Mapping to PaRAM Register |
| 0x01C0 021C | QCHMAP7 | QDMA Channel 7 Mapping to PaRAM Register |
| 0x01C0 0240 | DMAQNUM0 | DMA Queue Number Register 0 (Channels 00 to 07) |
| 0x01C0 0244 | DMAQNUM1 | DMA Queue Number Register 1 (Channels 08 to 15) |
| 0x01C0 0248 | DMAQNUM2 | DMA Queue Number Register 2 (Channels 16 to 23) |
| 0x01C0 024C | DMAQNUM3 | DMA Queue Number Register 3 (Channels 24 to 31) |
| 0x01C0 0250 | DMAQNUM4 | DMA Queue Number Register 4 (Channels 32 to 39) |
| 0x01C0 0254 | DMAQNUM5 | DMA Queue Number Register 5 (Channels 40 to 47) |
| 0x01C0 0258 | DMAQNUM6 | DMA Queue Number Register 6 (Channels 48 to 55) |
| 0x01C0 025C | DMAQNUM7 | DMA Queue Number Register 7 (Channels 56 to 63) |
| 0x01C0 0260 | QDMAQNUM | CC QDMA Queue Number |
| 0x01C0 0264 - 0x01C0 0283 | – | Reserved |
| 0x01C0 0284 | QUEPRI | Queue Priority Register |
| 0x01C0 0288 - 0x01C0 02FF | – | Reserved |
| 0x01C0 0300 | EMR | Event Missed Register |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---------------------------|----------|---|
| 0x01C0 0304 | EMRH | Event Missed Register High |
| 0x01C0 0308 | EMCR | Event Missed Clear Register |
| 0x01C0 030C | EMCRH | Event Missed Clear Register High |
| 0x01C0 0310 | QEMR | QDMA Event Missed Register |
| 0x01C0 0314 | QEMCR | QDMA Event Missed Clear Register |
| 0x01C0 0318 | CCERR | EDMA3CC Error Register |
| 0x01C0 031C | CCERRCLR | EDMA3CC Error Clear Register |
| 0x01C0 0320 | EEVAL | Error Evaluate Register |
| 0x01C0 0340 | DRAE0 | DMA Region Access Enable Register for Region 0 |
| 0x01C0 0344 | DRAEH0 | DMA Region Access Enable Register High for Region 0 |
| 0x01C0 0348 | DRAE1 | DMA Region Access Enable Register for Region 1 |
| 0x01C0 034C | DRAEH1 | DMA Region Access Enable Register High for Region 1 |
| 0x01C0 0350 | – | Reserved |
| 0x01C0 0354 | – | Reserved |
| 0x01C0 0358 | – | Reserved |
| 0x01C0 035C | – | Reserved |
| 0x01C0 0360 - 0x01C0 037C | – | Reserved |
| 0x01C0 0380 | QRAE0 | QDMA Region Access Enable Register for Region 0 |
| 0x01C0 0384 | QRAE1 | QDMA Region Access Enable Register for Region 1 |
| 0x01C0 0388 | – | Reserved |
| 0x01C0 038C | – | Reserved |
| 0x01C0 0390 - 0x01C0 039C | – | Reserved |
| 0x01C0 0400 | Q0E0 | Event Q0 Entry 0 Register |
| 0x01C0 0404 | Q0E1 | Event Q0 Entry 1 Register |
| 0x01C0 0408 | Q0E2 | Event Q0 Entry 2 Register |
| 0x01C0 040C | Q0E3 | Event Q0 Entry 3 Register |
| 0x01C0 0410 | Q0E4 | Event Q0 Entry 4 Register |
| 0x01C0 0414 | Q0E5 | Event Q0 Entry 5 Register |
| 0x01C0 0418 | Q0E6 | Event Q0 Entry 6 Register |
| 0x01C0 041C | Q0E7 | Event Q0 Entry 7 Register |
| 0x01C0 0420 | Q0E8 | Event Q0 Entry 8 Register |
| 0x01C0 0424 | Q0E9 | Event Q0 Entry 9 Register |
| 0x01C0 0428 | Q0E10 | Event Q0 Entry 10 Register |
| 0x01C0 042C | Q0E11 | Event Q0 Entry 11 Register |
| 0x01C0 0430 | Q0E12 | Event Q0 Entry 12 Register |
| 0x01C0 0434 | Q0E13 | Event Q0 Entry 13 Register |
| 0x01C0 0438 | Q0E14 | Event Q0 Entry 14 Register |
| 0x01C0 043C | Q0E15 | Event Q0 Entry 15 Register |
| 0x01C0 0440 | Q1E0 | Event Q1 Entry 0 Register |
| 0x01C0 0444 | Q1E1 | Event Q1 Entry 1 Register |
| 0x01C0 0448 | Q1E2 | Event Q1 Entry 2 Register |
| 0x01C0 044C | Q1E3 | Event Q1 Entry 3 Register |
| 0x01C0 0450 | Q1E4 | Event Q1 Entry 4 Register |
| 0x01C0 0454 | Q1E5 | Event Q1 Entry 5 Register |
| 0x01C0 0458 | Q1E6 | Event Q1 Entry 6 Register |
| 0x01C0 045C | Q1E7 | Event Q1 Entry 7 Register |
| 0x01C0 0460 | Q1E8 | Event Q1 Entry 8 Register |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---------------------------|---------|---|
| 0x01C0 0464 | Q1E9 | Event Q1 Entry 9 Register |
| 0x01C0 0468 | Q1E10 | Event Q1 Entry 10 Register |
| 0x01C0 046C | Q1E11 | Event Q1 Entry 11 Register |
| 0x01C0 0470 | Q1E12 | Event Q1 Entry 12 Register |
| 0x01C0 0474 | Q1E13 | Event Q1 Entry 13 Register |
| 0x01C0 0478 | Q1E14 | Event Q1 Entry 14 Register |
| 0x01C0 047C | Q1E15 | Event Q1 Entry 15 Register |
| 0x01C0 0480 | Q2E0 | Event Q2 Entry 0 Register |
| 0x01C0 0484 | Q2E1 | Event Q2 Entry 1 Register |
| 0x01C0 0488 | Q2E2 | Event Q2 Entry 2 Register |
| 0x01C0 048C | Q2E3 | Event Q2 Entry 3 Register |
| 0x01C0 0490 | Q2E4 | Event Q2 Entry 4 Register |
| 0x01C0 0494 | Q2E5 | Event Q2 Entry 5 Register |
| 0x01C0 0498 | Q2E6 | Event Q2 Entry 6 Register |
| 0x01C0 049C | Q2E7 | Event Q2 Entry 7 Register |
| 0x01C0 04A0 | Q2E8 | Event Q2 Entry 8 Register |
| 0x01C0 04A4 | Q2E9 | Event Q2 Entry 9 Register |
| 0x01C0 04A8 | Q2E10 | Event Q2 Entry 10 Register |
| 0x01C0 04AC | Q2E11 | Event Q2 Entry 11 Register |
| 0x01C0 04B0 | Q2E12 | Event Q2 Entry 12 Register |
| 0x01C0 04B4 | Q2E13 | Event Q2 Entry 13 Register |
| 0x01C0 04B8 | Q2E14 | Event Q2 Entry 14 Register |
| 0x01C0 04BC | Q2E15 | Event Q2 Entry 15 Register |
| 0x01C0 04C0 - 0x01C0 05FF | | Reserved |
| 0x01C0 0600 | QSTAT0 | Queue 0 Status Register |
| 0x01C0 0604 | QSTAT1 | Queue 1 Status Register |
| 0x01C0 0608 | QSTAT2 | Queue 2 Status Register |
| 0x01C0 060C - 0x01C0 061F | | Reserved |
| 0x01C0 0620 | QWMTHRA | Queue Watermark Threshold A Register for Q[2:0] |
| 0x01C0 0624 | – | Reserved |
| 0x01C0 0640 | CCSTAT | EDMA3CC Status Register |
| 0x01C0 0644 - 0x01C0 0FFF | | Reserved |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|--|---------|--------------------------------------|
| Global Channel Registers | | |
| 0x01C0 1000 | ER | Event Register |
| 0x01C0 1004 | ERH | Event Register High |
| 0x01C0 1008 | ECR | Event Clear Register |
| 0x01C0 100C | ECRH | Event Clear Register High |
| 0x01C0 1010 | ESR | Event Set Register |
| 0x01C0 1014 | ESRH | Event Set Register High |
| 0x01C0 1018 | CER | Chained Event Register |
| 0x01C0 101C | CERH | Chained Event Register High |
| 0x01C0 1020 | EER | Event Enable Register |
| 0x01C0 1024 | EERH | Event Enable Register High |
| 0x01C0 1028 | EECR | Event Enable Clear Register |
| 0x01C0 102C | EECRH | Event Enable Clear Register High |
| 0x01C0 1030 | EESR | Event Enable Set Register |
| 0x01C0 1034 | EESRH | Event Enable Set Register High |
| 0x01C0 1038 | SER | Secondary Event Register |
| 0x01C0 103C | SERH | Secondary Event Register High |
| 0x01C0 1040 | SECR | Secondary Event Clear Register |
| 0x01C0 1044 | SECRH | Secondary Event Clear Register High |
| 0x01C0 1048 - 0x01C0 104F | | Reserved |
| 0x01C0 1050 | IER | Interrupt Enable Register |
| 0x01C0 1054 | IERH | Interrupt Enable Register High |
| 0x01C0 1058 | IECR | Interrupt Enable Clear Register |
| 0x01C0 105C | IECRH | Interrupt Enable Clear Register High |
| 0x01C0 1060 | IESR | Interrupt Enable Set Register |
| 0x01C0 1064 | IESRH | Interrupt Enable Set Register High |
| 0x01C0 1068 | IPR | Interrupt Pending Register |
| 0x01C0 106C | IPRH | Interrupt Pending Register High |
| 0x01C0 1070 | ICR | Interrupt Clear Register |
| 0x01C0 1074 | ICRH | Interrupt Clear Register High |
| 0x01C0 1078 | IEVAL | Interrupt Evaluate Register |
| 0x01C0 1080 | QER | QDMA Event Register |
| 0x01C0 1084 | QEER | QDMA Event Enable Register |
| 0x01C0 1088 | QEECR | QDMA Event Enable Clear Register |
| 0x01C0 108C | QEESR | QDMA Event Enable Set Register |
| 0x01C0 1090 | QSER | QDMA Secondary Event Register |
| 0x01C0 1094 | QSECR | QDMA Secondary Event Clear Register |
| 0x01C0 1098 - 0x01C0 1FFF | | Reserved |
| Shadow Region 0 Channel Registers | | |
| 0x01C0 2000 | ER | Event Register |
| 0x01C0 2004 | ERH | Event Register High |
| 0x01C0 2008 | ECR | Event Clear Register |
| 0x01C0 200C | ECRH | Event Clear Register High |
| 0x01C0 2010 | ESR | Event Set Register |
| 0x01C0 2014 | ESRH | Event Set Register High |
| 0x01C0 2018 | CER | Chained Event Register |
| 0x01C0 201C | CERH | Chained Event Register High |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|--|---------|--------------------------------------|
| 0x01C0 2020 | EER | Event Enable Register |
| 0x01C0 2024 | EERH | Event Enable Register High |
| 0x01C0 2028 | EECR | Event Enable Clear Register |
| 0x01C0 202C | EECRH | Event Enable Clear Register High |
| 0x01C0 2030 | EESR | Event Enable Set Register |
| 0x01C0 2034 | EESRH | Event Enable Set Register High |
| 0x01C0 2038 | SER | Secondary Event Register |
| 0x01C0 203C | SERH | Secondary Event Register High |
| 0x01C0 2040 | SECR | Secondary Event Clear Register |
| 0x01C0 2044 | SECRH | Secondary Event Clear Register High |
| 0x01C0 2048 - 0x01C0 204C | - | Reserved |
| 0x01C0 2050 | IER | Interrupt Enable Register |
| 0x01C0 2054 | IERH | Interrupt Enable Register High |
| 0x01C0 2058 | IECR | Interrupt Enable Clear Register |
| 0x01C0 205C | IECRH | Interrupt Enable Clear Register High |
| 0x01C0 2060 | IESR | Interrupt Enable Set Register |
| 0x01C0 2064 | IESRH | Interrupt Enable Set Register High |
| 0x01C0 2068 | IPR | Interrupt Pending Register |
| 0x01C0 206C | IPRH | Interrupt Pending Register High |
| 0x01C0 2070 | ICR | Interrupt Clear Register |
| 0x01C0 2074 | ICRH | Interrupt Clear Register High |
| 0x01C0 2078 | IEVAL | Interrupt Evaluate Register |
| 0x01C0 207C | - | Reserved |
| 0x01C0 2080 | QER | QDMA Event Register |
| 0x01C0 2084 | QEER | QDMA Event Enable Register |
| 0x01C0 2088 | QEECR | QDMA Event Enable Clear Register |
| 0x01C0 208C | QEESR | QDMA Event Enable Set Register |
| 0x01C0 2090 | QSER | QDMA Secondary Event Register |
| 0x01C0 2094 | QSECR | QDMA Secondary Event Clear Register |
| 0x01C0 2098 - 0x01C0 21FC | - | Reserved |
| Shadow Region 1 Channel Registers | | |
| 0x01C0 2200 | ER | Event Register |
| 0x01C0 2204 | ERH | Event Register High |
| 0x01C0 2208 | ECR | Event Clear Register |
| 0x01C0 220C | ECRH | Event Clear Register High |
| 0x01C0 2210 | ESR | Event Set Register |
| 0x01C0 2214 | ESRH | Event Set Register High |
| 0x01C0 2218 | CER | Chained Event Register |
| 0x01C0 221C | CERH | Chained Event Register High |
| 0x01C0 2220 | EER | Event Enable Register |
| 0x01C0 2224 | EERH | Event Enable Register High |
| 0x01C0 2228 | EECR | Event Enable Clear Register |
| 0x01C0 222C | EECRH | Event Enable Clear Register High |
| 0x01C0 2230 | EESR | Event Enable Set Register |
| 0x01C0 2234 | EESRH | Event Enable Set Register High |
| 0x01C0 2238 | SER | Secondary Event Register |
| 0x01C0 223C | SERH | Secondary Event Register High |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|--|---------|--|
| 0x01C0 2240 | SECR | Secondary Event Clear Register |
| 0x01C0 2244 | SECRH | Secondary Event Clear Register High |
| 0x01C0 2248 - 0x01C0 224C | - | Reserved |
| 0x01C0 2250 | IER | Interrupt Enable Register |
| 0x01C0 2254 | IERH | Interrupt Enable Register High |
| 0x01C0 2258 | IECR | Interrupt Enable Clear Register |
| 0x01C0 225C | IECRH | Interrupt Enable Clear Register High |
| 0x01C0 2260 | IESR | Interrupt Enable Set Register |
| 0x01C0 2264 | IESRH | Interrupt Enable Set Register High |
| 0x01C0 2268 | IPR | Interrupt Pending Register |
| 0x01C0 226C | IPRH | Interrupt Pending Register High |
| 0x01C0 2270 | ICR | Interrupt Clear Register |
| 0x01C0 2274 | ICRH | Interrupt Clear Register High |
| 0x01C0 2278 | IEVAL | Interrupt Evaluate Register |
| 0x01C0 227C | - | Reserved |
| 0x01C0 2280 | QER | QDMA Event Register |
| 0x01C0 2284 | QEER | QDMA Event Enable Register |
| 0x01C0 2288 | QEECR | QDMA Event Enable Clear Register |
| 0x01C0 228C | QEESR | QDMA Event Enable Set Register |
| 0x01C0 2290 | QSER | QDMA Secondary Event Register |
| 0x01C0 2294 | QSECR | QDMA Secondary Event Clear Register |
| 0x01C0 2298 - 0x01C0 23FC | - | Reserved |
| 0x01C0 2400 - 0x01C0 25FC | - | Reserved |
| 0x01C0 2600 - 0x01C0 27FC | - | Reserved |
| 0x01C0 2800 - 0x01C0 29FC | - | Reserved |
| 0x01C0 2A00 - 0x01C0 2BFC | - | Reserved |
| 0x01C0 2C00 - 0x01C0 2DFC | - | Reserved |
| 0x01C0 2E00 - 0x01C0 2FFC | - | Reserved |
| 0x01C0 2FFD - 0x01C0 3FFF | - | Reserved |
| 0x01C0 4000 - 0x01C0 4FFF | - | Parameter Set RAM (see Table 6-8) |
| 0x01C0 5000 - 0x01C0 7FFF | - | Reserved |
| 0x01C0 8000 - 0x01C0 FFFF | - | Reserved |
| Transfer Controller 0 Registers | | |
| 0x01C1 0000 | - | Reserved |
| 0x01C1 0004 | TCCFG | EDMA3 TC0 Configuration Register |
| 0x01C1 0008 - 0x01C1 00FF | - | Reserved |
| 0x01C1 0100 | TCSTAT | EDMA3 TC0 Channel Status Register |
| 0x01C1 0104 - 0x01C1 0110 | - | Reserved |
| 0x01C1 0114 - 0x01C1 011F | - | Reserved |
| 0x01C1 0120 | ERRSTAT | EDMA3 TC0 Error Status Register |
| 0x01C1 0124 | ERREN | EDMA3 TC0 Error Enable Register |
| 0x01C1 0128 | ERRCLR | EDMA3 TC0 Error Clear Register |
| 0x01C1 012C | ERRDET | EDMA3 TC0 Error Details Register |
| 0x01C1 0130 | ERRCMD | EDMA3 TC0 Error Interrupt Command Register |
| 0x01C1 0134 - 0x01C1 013F | - | Reserved |
| 0x01C1 0140 | RDRATE | EDMA3 TC0 Read Command Rate Register |
| 0x01C1 0144 - 0x01C1 01FF | - | Reserved |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|--|-----------|---|
| 0x01C1 0200 - 0x01C1 023F | - | Reserved |
| 0x01C1 0240 | SAOPT | EDMA3 TC0 Source Active Options Register |
| 0x01C1 0244 | SASRC | EDMA3 TC0 Source Active Source Address Register |
| 0x01C1 0248 | SACNT | EDMA3 TC0 Source Active Count Register |
| 0x01C1 024C | SADST | EDMA3 TC0 Source Active Destination Address Register |
| 0x01C1 0250 | SABIDX | EDMA3 TC0 Active B-Index Register |
| 0x01C1 0254 | SAMPPRXY | EDMA3 TC0 Source Active Memory Protection Proxy Register |
| 0x01C1 0258 | SACNTRLD | EDMA3 TC0 Source Active Count Reload Register |
| 0x01C1 025C | SASRCBREF | EDMA3 TC0 Source Active Source Address B-Reference Register |
| 0x01C1 0260 | SADSTBREF | EDMA3 TC0 Source Active Destination Address B-Reference Register |
| 0x01C1 0264 - 0x01C1 027F | - | Reserved |
| 0x01C1 0280 | DFCNTRLD | EDMA3 TC0 Destination FIFO Set Count Reload Register |
| 0x01C1 0284 | DFSRCBREF | EDMA3 TC0 Destination FIFO Set Source Address B-Reference Register |
| 0x01C1 0288 | DFDSTBREF | EDMA3 TC0 Destination FIFO Set Destination Address B-Reference Register |
| 0x01C1 028C - 0x01C1 02FF | - | Reserved |
| 0x01C1 0300 | DFOPT0 | EDMA3 TC0 Destination FIFO Options Register 0 |
| 0x01C1 0304 | DFSRC0 | EDMA3 TC0 Destination FIFO Source Address Register 0 |
| 0x01C1 0308 | DFCNT0 | EDMA3 TC0 Destination FIFO Count Register 0 |
| 0x01C1 030C | DFDST0 | EDMA3 TC0 Destination FIFO Destination Address Register 0 |
| 0x01C1 0310 | DFBIDX0 | EDMA3 TC0 Destination FIFO B-Index Register 0 |
| 0x01C1 0314 | DFMPPRXY0 | EDMA3 TC0 Destination FIFO Memory Protection Proxy Register 0 |
| 0x01C1 0318 - 0x01C1 033F | - | Reserved |
| 0x01C1 0340 | DFOPT1 | EDMA3 TC0 Destination FIFO Options Register 1 |
| 0x01C1 0344 | DFSRC1 | EDMA3 TC0 Destination FIFO Source Address Register 1 |
| 0x01C1 0348 | DFCNT1 | EDMA3 TC0 Destination FIFO Count Register 1 |
| 0x01C1 034C | DFDST1 | EDMA3 TC0 Destination FIFO Destination Address Register 1 |
| 0x01C1 0350 | DFBIDX1 | EDMA3 TC0 Destination FIFO B-Index Register 1 |
| 0x01C1 0354 | DFMPPRXY1 | EDMA3 TC0 Destination FIFO Memory Protection Proxy Register 1 |
| 0x01C1 0358 - 0x01C1 037F | - | Reserved |
| 0x01C1 0380 | DFOPT2 | EDMA3 TC0 Destination FIFO Options Register 2 |
| 0x01C1 0384 | DFSRC2 | EDMA3 TC0 Destination FIFO Source Address Register 2 |
| 0x01C1 0388 | DFCNT2 | EDMA3 TC0 Destination FIFO Count Register 2 |
| 0x01C1 038C | DFDST2 | EDMA3 TC0 Destination FIFO Destination Address Register 2 |
| 0x01C1 0390 | DFBIDX2 | EDMA3 TC0 Destination FIFO B-Index Register 2 |
| 0x01C1 0394 | DFMPPRXY2 | EDMA3 TC0 Destination FIFO Memory Protection Proxy Register 2 |
| 0x01C1 0398 - 0x01C1 03BF | - | Reserved |
| 0x01C1 03C0 | DFOPT3 | EDMA3 TC0 Destination FIFO Options Register 3 |
| 0x01C1 03C4 | DFSRC3 | EDMA3 TC0 Destination FIFO Source Address Register 3 |
| 0x01C1 03C8 | DFCNT3 | EDMA3 TC0 Destination FIFO Count Register 3 |
| 0x01C1 03CC | DFDST3 | EDMA3 TC0 Destination FIFO Destination Address Register 3 |
| 0x01C1 03D0 | DFBIDX3 | EDMA3 TC0 Destination FIFO B-Index Register 3 |
| 0x01C1 03D4 | DFMPPRXY3 | EDMA3 TC0 Destination FIFO Memory Protection Proxy Register 3 |
| 0x01C1 03D8 - 0x01C1 03FF | - | Reserved |
| Transfer Controller 1 Registers | | |
| 0x01C1 0400 | - | Reserved |
| 0x01C1 0404 | TCCFG | EDMA3 TC1 Configuration Register |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---------------------------|-----------|---|
| 0x01C1 0408 - 0x01C1 04FF | - | Reserved |
| 0x01C1 0500 | TCSTAT | EDMA3 TC1 Channel Status Register |
| 0x01C1 0504 - 0x01C1 0510 | - | Reserved |
| 0x01C1 0514 - 0x01C1 051F | - | Reserved |
| 0x01C1 0520 | ERRSTAT | EDMA3 TC1 Error Status Register |
| 0x01C1 0524 | ERREN | EDMA3 TC1 Error Enable Register |
| 0x01C1 0528 | ERRCLR | EDMA3 TC1 Error Clear Register |
| 0x01C1 052C | ERRDET | EDMA3 TC1 Error Details Register |
| 0x01C1 0530 | ERRCMD | EDMA3 TC1 Error Interrupt Command Register |
| 0x01C1 0534 - 0x01C1 053F | - | Reserved |
| 0x01C1 0540 | RDRATE | EDMA3 TC1 Read Command Rate Register |
| 0x01C1 0544 - 0x01C1 05FF | - | Reserved |
| 0x01C1 0600 - 0x01C1 063F | - | Reserved |
| 0x01C1 0640 | SAOPT | EDMA3 TC1 Source Active Options Register |
| 0x01C1 0644 | SASRC | EDMA3 TC1 Source Active Source Address Register |
| 0x01C1 0648 | SACNT | EDMA3 TC1 Source Active Count Register |
| 0x01C1 064C | SADST | EDMA3 TC1 Source Active Destination Address Register |
| 0x01C1 0650 | SABIDX | EDMA3 TC1 Active B-Index Register |
| 0x01C1 0654 | SAMPPRXY | EDMA3 TC1 Source Active Memory Protection Proxy Register |
| 0x01C1 0658 | SACNTRLD | EDMA3 TC1 Source Active Count Reload Register |
| 0x01C1 065C | SASRCBREF | EDMA3 TC1 Source Active Source Address B-Reference Register |
| 0x01C1 0660 | SADSTBREF | EDMA3 TC1 Source Active Destination Address B-Reference Register |
| 0x01C1 0664 - 0x01C1 067F | - | Reserved |
| 0x01C1 0680 | DFCNTRLD | EDMA3 TC1 Destination FIFO Set Count Reload Register |
| 0x01C1 0684 | DFSRCBREF | EDMA3 TC1 Destination FIFO Set Source Address B-Reference Register |
| 0x01C1 0688 | DFDSTBREF | EDMA3 TC1 Destination FIFO Set Destination Address B-Reference Register |
| 0x01C1 068C - 0x01C1 06FF | - | Reserved |
| 0x01C1 0700 | DFOPT0 | EDMA3 TC1 Destination FIFO Options Register 0 |
| 0x01C1 0704 | DFSRC0 | EDMA3 TC1 Destination FIFO Source Address Register 0 |
| 0x01C1 0708 | DFCNT0 | EDMA3 TC1 Destination FIFO Count Register 0 |
| 0x01C1 070C | DFDST0 | EDMA3 TC1 Destination FIFO Destination Address Register 0 |
| 0x01C1 0710 | DFBIDX0 | EDMA3 TC1 Destination FIFO B-Index Register 0 |
| 0x01C1 0714 | DFMPPRXY0 | EDMA3 TC1 Destination FIFO Memory Protection Proxy Register 0 |
| 0x01C1 0718 - 0x01C1 073F | - | Reserved |
| 0x01C1 0740 | DFOPT1 | EDMA3 TC1 Destination FIFO Options Register 1 |
| 0x01C1 0744 | DFSRC1 | EDMA3 TC1 Destination FIFO Source Address Register 1 |
| 0x01C1 0748 | DFCNT1 | EDMA3 TC1 Destination FIFO Count Register 1 |
| 0x01C1 074C | DFDST1 | EDMA3 TC1 Destination FIFO Destination Address Register 1 |
| 0x01C1 0750 | DFBIDX1 | EDMA3 TC1 Destination FIFO B-Index Register 1 |
| 0x01C1 0754 | DFMPPRXY1 | EDMA3 TC1 Destination FIFO Memory Protection Proxy Register 1 |
| 0x01C1 0758 - 0x01C1 077F | - | Reserved |
| 0x01C1 0780 | DFOPT2 | EDMA3 TC1 Destination FIFO Options Register 2 |
| 0x01C1 0784 | DFSRC2 | EDMA3 TC1 Destination FIFO Source Address Register 2 |
| 0x01C1 0788 | DFCNT2 | EDMA3 TC1 Destination FIFO Count Register 2 |
| 0x01C1 078C | DFDST2 | EDMA3 TC1 Destination FIFO Destination Address Register 2 |
| 0x01C1 0790 | DFBIDX2 | EDMA3 TC1 Destination FIFO B-Index Register 2 |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|--|-----------|---|
| 0x01C1 0794 | DFMPPRXY2 | EDMA3 TC1 Destination FIFO Memory Protection Proxy Register 2 |
| 0x01C1 0798 - 0x01C1 07BF | - | Reserved |
| 0x01C1 07C0 | DFOPT3 | EDMA3 TC1 Destination FIFO Options Register 3 |
| 0x01C1 07C4 | DFSRC3 | EDMA3 TC1 Destination FIFO Source Address Register 3 |
| 0x01C1 07C8 | DFCNT3 | EDMA3 TC1 Destination FIFO Count Register 3 |
| 0x01C1 07CC | DFDST3 | EDMA3 TC1 Destination FIFO Destination Address Register 3 |
| 0x01C1 07D0 | DFBIDX3 | EDMA3 TC1 Destination FIFO B-Index Register 3 |
| 0x01C1 07D4 | DFMPPRXY3 | EDMA3 TC1 Destination FIFO Memory Protection Proxy Register 3 |
| 0x01C1 07D8 - 0x01C1 07FF | - | Reserved |
| Transfer Controller 2 Registers | | |
| 0x01C1 0800 | - | Reserved |
| 0x01C1 0804 | TCCFG | EDMA3 TC2 Configuration Register |
| 0x01C1 0808 - 0x01C1 08FF | - | Reserved |
| 0x01C1 0900 | TCSTAT | EDMA3 TC2 Channel Status Register |
| 0x01C1 0904 - 0x01C1 0910 | - | Reserved |
| 0x01C1 0914 - 0x01C1 091F | - | Reserved |
| 0x01C1 0920 | ERRSTAT | EDMA3 TC2 Error Status Register |
| 0x01C1 0924 | ERREN | EDMA3 TC2 Error Enable Register |
| 0x01C1 0928 | ERRCLR | EDMA3 TC2 Error Clear Register |
| 0x01C1 092C | ERRDET | EDMA3 TC2 Error Details Register |
| 0x01C1 0930 | ERRCMD | EDMA3 TC2 Error Interrupt Command Register |
| 0x01C1 0934 - 0x01C1 093F | - | Reserved |
| 0x01C1 0940 | RDRATE | EDMA3 TC2 Read Command Rate Register |
| 0x01C1 0944 - 0x01C1 09FF | - | Reserved |
| 0x01C1 0A00 - 0x01C1 0A3F | - | Reserved |
| 0x01C1 0A40 | SAOPT | EDMA3 TC2 Source Active Options Register |
| 0x01C1 0A44 | SASRC | EDMA3 TC2 Source Active Source Address Register |
| 0x01C1 0A48 | SACNT | EDMA3 TC2 Source Active Count Register |
| 0x01C1 0A4C | SADST | EDMA3 TC2 Source Active Destination Address Register |
| 0x01C1 0A50 | SABIDX | EDMA3 TC2 Active B-Index Register |
| 0x01C1 0A54 | SAMPPRXY | EDMA3 TC2 Source Active Memory Protection Proxy Register |
| 0x01C1 0A58 | SACNTRLD | EDMA3 TC2 Source Active Count Reload Register |
| 0x01C1 0A5C | SASRCBREF | EDMA3 TC2 Source Active Source Address B-Reference Register |
| 0x01C1 0A60 | SADSTBREF | EDMA3 TC2 Source Active Destination Address B-Reference Register |
| 0x01C1 0A64 - 0x01C1 0A7F | - | Reserved |
| 0x01C1 0A80 | DFCNTRLD | EDMA3 TC2 Destination FIFO Set Count Reload Register |
| 0x01C1 0A84 | DFSRCBREF | EDMA3 TC2 Destination FIFO Set Source Address B-Reference Register |
| 0x01C1 0A88 | DFDSTBREF | EDMA3 TC2 Destination FIFO Set Destination Address B-Reference Register |
| 0x01C1 0A8C - 0x01C1 0AFF | - | Reserved |
| 0x01C1 0B00 | DFOPT0 | EDMA3 TC2 Destination FIFO Options Register 0 |
| 0x01C1 0B04 | DFSRC0 | EDMA3 TC2 Destination FIFO Source Address Register 0 |
| 0x01C1 0B08 | DFCNT0 | EDMA3 TC2 Destination FIFO Count Register 0 |
| 0x01C1 0B0C | DFDST0 | EDMA3 TC2 Destination FIFO Destination Address Register 0 |
| 0x01C1 0B10 | DFBIDX0 | EDMA3 TC2 Destination FIFO B-Index Register 0 |
| 0x01C1 0B14 | DFMPPRXY0 | EDMA3 TC2 Destination FIFO Memory Protection Proxy Register 0 |
| 0x01C1 0B18 - 0x01C1 0B3F | - | Reserved |

Table 6-7. C6424 EDMA Registers (continued)

| HEX ADDRESS | ACRONYM | REGISTER NAME |
|---------------------------|-----------|---|
| 0x01C1 0B40 | DFOPT1 | EDMA3 TC2 Destination FIFO Options Register 1 |
| 0x01C1 0B44 | DFSRC1 | EDMA3 TC2 Destination FIFO Source Address Register 1 |
| 0x01C1 0B48 | DFCNT1 | EDMA3 TC2 Destination FIFO Count Register 1 |
| 0x01C1 0B4C | DFDST1 | EDMA3 TC2 Destination FIFO Destination Address Register 1 |
| 0x01C1 0B50 | DFBIDX1 | EDMA3 TC2 Destination FIFO B-Index Register 1 |
| 0x01C1 0B54 | DFMPPRXY1 | EDMA3 TC2 Destination FIFO Memory Protection Proxy Register 1 |
| 0x01C1 0B58 - 0x01C1 0B7F | - | Reserved |
| 0x01C1 0B80 | DFOPT2 | EDMA3 TC2 Destination FIFO Options Register 2 |
| 0x01C1 0B84 | DFSRC2 | EDMA3 TC2 Destination FIFO Source Address Register 2 |
| 0x01C1 0B88 | DFCNT2 | EDMA3 TC2 Destination FIFO Count Register 2 |
| 0x01C1 0B8C | DFDST2 | EDMA3 TC2 Destination FIFO Destination Address Register 2 |
| 0x01C1 0B90 | DFBIDX2 | EDMA3 TC2 Destination FIFO B-Index Register 2 |
| 0x01C1 0B94 | DFMPPRXY2 | EDMA3 TC2 Destination FIFO Memory Protection Proxy Register 2 |
| 0x01C1 0B98 - 0x01C1 0BBF | - | Reserved |
| 0x01C1 0BC0 | DFOPT3 | EDMA3 TC2 Destination FIFO Options Register 3 |
| 0x01C1 0BC4 | DFSRC3 | EDMA3 TC2 Destination FIFO Source Address Register 3 |
| 0x01C1 0BC8 | DFCNT3 | EDMA3 TC2 Destination FIFO Count Register 3 |
| 0x01C1 0BCC | DFDST3 | EDMA3 TC2 Destination FIFO Destination Address Register 3 |
| 0x01C1 0BD0 | DFBIDX3 | EDMA3 TC2 Destination FIFO B-Index Register 3 |
| 0x01C1 0BD4 | DFMPPRXY3 | EDMA3 TC2 Destination FIFO Memory Protection Proxy Register 3 |
| 0x01C1 0BD8 - 0x01C1 0BFF | - | Reserved |

Table 6-8 shows an abbreviation of the set of registers which make up the parameter set for each of 128 EDMA events. Each of the parameter register sets consist of 8 32-bit word entries. Table 6-9 shows the parameter set entry registers with relative memory address locations within each of the parameter sets.

Table 6-8. EDMA Parameter Set RAM

| HEX ADDRESS RANGE | DESCRIPTION |
|---------------------------|-------------------------------------|
| 0x01C0 4000 - 0x01C0 401F | Parameters Set 0 (8 32-bit words) |
| 0x01C0 4020 - 0x01C0 403F | Parameters Set 1 (8 32-bit words) |
| 0x01C0 4040 - 0x01C0 405F | Parameters Set 2 (8 32-bit words) |
| 0x01C0 4060 - 0x01C0 407F | Parameters Set 3 (8 32-bit words) |
| 0x01C0 4080 - 0x01C0 409F | Parameters Set 4 (8 32-bit words) |
| 0x01C0 40A0 - 0x01C0 40BF | Parameters Set 5 (8 32-bit words) |
| ... | ... |
| 0x01C0 4FC0 - 0x01C0 4FDF | Parameters Set 126 (8 32-bit words) |
| 0x01C0 4FE0 - 0x01C0 4FFF | Parameters Set 127 (8 32-bit words) |

Table 6-9. 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 |

6.5 Reset

The reset controller detects the different type of resets supported on the C6424 device and manages the distribution of those resets throughout the device.

The C6424 device has several types of device-level resets - power-on reset, warm reset, max reset, and CPU reset. [Table 6-10](#) explains further the types of reset, the reset initiator, and the effects of each reset on the chip. See [Section 6.5.9, Reset Electrical Data/Timing](#), for more information on the effects of each reset on the PLL controllers and their clocks.

Table 6-10. Device-Level Global Reset Types

| TYPE | INITIATOR | EFFECT(s) |
|----------------------|-------------------------------|---|
| Power-on Reset (POR) | $\overline{\text{POR}}$ pin | Global chip reset (Cold reset). Activates the POR signal on chip, which resets the entire chip including the emulation logic. The power-on reset (POR) pin must be driven low during power ramp of the device. Device boot and configuration pins are latched. |
| Warm Reset | $\overline{\text{RESET}}$ pin | Resets everything except for the emulation logic. Emulator stays alive during Warm Reset. Device boot and configuration pins are latched. |
| Max Reset | Emulator, WD Timer (Timer 2) | Same as a Warm Reset, except the C6424 device boot and configuration pins are not re-latched. |

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

6.5.1 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

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. If an external 15–30-MHz oscillator is used on the MXI/CLKIN pin, the external clock should also be running at the correct frequency prior to de-asserting the $\overline{\text{POR}}$ pin. **Note:** 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 the power supplies to reach normal operating conditions while keeping the $\overline{\text{POR}}$ pin asserted (driven low).
2. Wait for the input clock source to be stable while keeping the $\overline{\text{POR}}$ pin asserted (low).
3. Once the power supplies and the input clock source are stable, the $\overline{\text{POR}}$ pin **must** remain asserted (low) for a minimum of 12 MXI cycles.

Within the low period of the $\overline{\text{POR}}$ pin, the following happens:

- The reset signals flow to the entire chip (including the emulation logic), resetting the modules on chip.
 - The PLL Controller clocks start at the frequency of the MXI clock. The clocks are propagated throughout the chip to reset the chip synchronously. By default, both PLL1 and PLL2 are in reset and unlocked. The PLL Controllers default to PLL Bypass Mode.
 - The $\overline{\text{RESETOUT}}$ pin stays asserted (low), indicating the device is in reset.
4. The $\overline{\text{POR}}$ pin may now be deasserted (driven high).

When the $\overline{\text{POR}}$ pin is deasserted (high), the configuration pin values are latched and the PLL Controllers changed their system clocks to their default divide-down values. Both PLL Controllers are still in PLL Bypass Mode. Other device initialization also begins.

5. After device initialization is complete, the PLL Controllers pause the system clocks for 10 cycles. At the end of these 10 cycles, the $\overline{\text{RESETOUT}}$ pin is deasserted (driven high).

At this point:

- The I/O pins are controlled by the default peripherals (default peripherals are determined by PINMUX0 and PINMUX1 registers).
- The clock and reset of each peripheral is determined by the default settings of the Power and Sleep Controller (PSC).
- The PLL Controllers are operating in PLL Bypass Mode.
- The C64x+ begins executing from DSPBOOTADDR (determined by bootmode selection).

After the reset sequence, the boot sequence begins. For more details on the boot sequence, see the *Using the TMS320C642x Bootloader* Application Report (literature number SPRAAK5).

After the boot sequence, follow the software initialization sequence described in [Section 3.8, Device Initialization Sequence After Reset](#).

6.5.1.1 Usage of $\overline{\text{POR}}$ versus $\overline{\text{RESET}}$ Pins

$\overline{\text{POR}}$ and $\overline{\text{RESET}}$ are independent resets.

If the device needs to go through a power-up cycle, $\overline{\text{POR}}$ (not $\overline{\text{RESET}}$) **must** be used to fully reset the device.

In functional end-system, emulation/debugger logic is typically *not* needed; therefore, the recommendation for functional end-system is to use the $\overline{\text{POR}}$ pin for full device reset. If $\overline{\text{RESET}}$ pin is *not* needed, it can be pulled inactive (high) via an external pullup resistor.

In a debug system, it is typically desirable to allow the reset of the device without crashing an emulation session. In this case, the user can use the $\overline{\text{POR}}$ pin to achieve full device reset and use the $\overline{\text{RESET}}$ pin to achieve a debug reset—which resets the entire device except emulation logic.

6.5.1.2 Latching Boot and Configuration Pins

Internal to the chip, the two device reset pins $\overline{\text{RESET}}$ and $\overline{\text{POR}}$ are logically AND'd together *only* for the purpose of latching device boot and configuration pins. The values on all device and boot configuration pins are latched into the BOOTCFG register when the logical AND of $\overline{\text{RESET}}$ and $\overline{\text{POR}}$ transitions from low-to-high.

6.5.2 Warm Reset ($\overline{\text{RESET}}$ Pin)

A Warm Reset is activated by driving the $\overline{\text{RESET}}$ pin active low. This resets everything in the device except the emulation logic. An emulator session will stay alive during warm reset.

For more information on $\overline{\text{POR}}$ vs. $\overline{\text{RESET}}$ usage, see [Section 6.5.1.1](#), *Usage of $\overline{\text{POR}}$ versus $\overline{\text{RESET}}$ Pins* and [Section 6.5.1.2](#), *Latching Boot and Configuration Pins*.

The following sequence **must** be followed during a Warm Reset:

1. Power supplies and input clock source should already be stable.
2. The RESET pin **must** be asserted (low) for a minimum of 12 MXI cycles.

Within the low period of the $\overline{\text{RESET}}$ pin, the following happens:

- The reset signals flow to the entire chip resetting all the modules on chip, except the emulation logic.
- The PLL Controllers are reset thereby, switching back to PLL Bypass Mode and resetting all their registers to default values. Both PLL1 and PLL2 are placed in reset and lose lock.
- The $\overline{\text{RESETOUT}}$ pin becomes asserted (low), indicating the device is in reset.

3. The $\overline{\text{RESET}}$ pin may now be deasserted (driven high).

When the $\overline{\text{RESET}}$ pin is deasserted (high), the configuration pin values are latched and the PLL Controllers changed their system clocks to their default divide-down values. Both PLL Controllers are still in PLL Bypass Mode. Other device initialization also begins.

4. After device initialization is complete, the PLL Controllers pause the system clocks for 10 cycles. At the end of these 10 cycles, the $\overline{\text{RESETOUT}}$ pin is deasserted (driven high).

At this point:

- The I/O pins are controlled by the default peripherals (default peripherals are determined by PINMUX0 and PINMUX1 registers).
- The clock and reset of each peripheral is determined by the default settings of the Power and Sleep Controller (PSC).
- The PLL Controllers are operating in PLL Bypass Mode.
- The C64x+ begins executing from DSPBOOTADDR (determined by bootmode selection).

After the reset sequence, the boot sequence begins. For more details on the boot sequence, see the *Using the TMS320C642x Bootloader* Application Report (literature number [SPRAAK5](#)).

After the boot sequence, follow the software initialization sequence described in [Section 3.8](#), *Device Initialization Sequence After Reset*.

6.5.3 Maximum Reset

A Maximum (Max) Reset is initiated by the emulator or the watchdog timer (Timer 2). 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 non-maskable. When the watchdog timer counter reaches zero, this will also initiate a maximum reset to recover from a runaway condition. The watchdog timeout reset condition is masked if the TIMERCTL.WDRST bit is cleared to "0".

To invoke the maximum reset via the ICEPICK module, the user can perform the following from the Code Composer Studio™ IDE menu: Debug → Advanced Resets → System Reset

This is the Max Reset sequence:

1. Max Reset is initiated by the emulator or the watchdog timer.
During this time, the following happens:
 - The reset signals flow to the entire chip resetting all the modules on chip except the emulation logic.
 - The PLL Controllers are reset thereby, switching back to PLL Bypass Mode and resetting all their registers to default values. Both PLL1 and PLL2 are placed in reset and lose lock.
 - The $\overline{\text{RESETOUT}}$ 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 $\overline{\text{RESETOUT}}$ pin is deasserted (driven high).

At this point:

- The I/O pins are controlled by the default peripherals (default peripherals are determined by PINMUX0 and PINMUX1 registers).
- The clock and reset of each peripheral is determined by the default settings of the Power and Sleep Controller (PSC).
- The PLL Controllers are operating in PLL Bypass Mode.
- 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 boot mode. For more details, see the *Using the TMS320C642x Bootloader* Application Report (literature number [SPRAAK5](#)).

After the boot sequence, follow the software initialization sequence described in [Section 3.8](#), *Device Initialization Sequence After Reset*.

6.5.4 CPU Local Reset

The C64x+ DSP CPU has an internal reset input that allows a host (PCI/HPI) to control it. This reset is configured through a register bit (MDCTL[39]. $\overline{\text{LRST}}$) in the Power Sleep Controller (PSC) module. When in C64x+ local reset, the slave DMA port on C64x+ will remain active and the internal memory will be accessible. For procedures on asserting and de-asserting CPU local reset by the host, see the *TMS320C642x Power and Sleep Controller (PSC) Reference Guide* (literature number [SPRUEN8](#)).

For information on peripheral selection at the rising edge of $\overline{\text{POR}}$ or $\overline{\text{RESET}}$, see [Section 3](#), *Device Configurations* of this data manual.

6.5.5 Peripheral Local Reset

The user can configure the local reset and clock state of a peripheral through programming the PSC. [Table 6-4](#), C6424 LPSC Assignments 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 *TMS320C642x Power and Sleep Controller (PSC) Reference Guide* (literature number [SPRUEN8](#)).

6.5.6 Reset Priority

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

- Power-on Reset
- Maximum Reset
- Warm Reset
- CPU Reset

6.5.7 Reset Controller Register

The reset type status (RSTYPE) register (01C4 00E4) is the only register for the reset controller. This register falls in the same memory range as the PLL1 controller registers (see [Section 6.7.2](#), for the PLL1 Controller Registers (including Reset Controller)). For more details on the RSTYPE register, see the *TMS320C642x DSP Phase-Locked Loop Controller (PLL) User's Guide* (literature number [SPRUJES0](#)).

6.5.8 Pin Behaviors at Reset

During normal operations, pins are controlled by the respective peripheral selected in the PINMUX0 or PINMUX1 register. During device level global reset, the pin behaves as follows:

Multiplexed Boot and Configuration Pins

These pins are forced 3-stated when $\overline{\text{RESETOUT}}$ is asserted (low). This is to ensure the proper boot and configuration values can be latched on these multiplexed pins. This is particularly useful in the case where the boot and configuration values are driven by an external control device. After $\overline{\text{RESETOUT}}$ is deasserted (high), these pins are controlled by their respective default peripheral.

- **Boot and Configuration Pins Group:** RMTXD0/GP[28]/(8_16), RMTXD1/GP[27](LENDIAN), GP[26]/(FASTBOOT), GP[25]/(BOOTMODE3), GP[24]/(BOOTMODE2), GP[23]/(BOOTMODE1), GP[22]/(BOOTMODE0), EM_A[4]/GP[10]/(PLLMS2), EM_A[1]/(ALE)/GP[9]/(PLLMS1), EM_A[2]/(CLE)/GP[8]/(PLLMS0), EM_A[0]/GP[7]/(AEM2), EM_BA[0]/GP[6]/(AEM1), and EM_BA[1]/GP[5]/(AEM0).

For information on whether external pullup/pulldown resistors should be used on the boot and configuration pins, see [Section 3.9.1, Pullup/Pulldown Resistors](#).

Default Power Down Pins

As discussed in [Section 3.2, Power Considerations](#), the VDD3P3V_PWDN register controls power to the 3.3-V pins. The VDD3P3V_PWDN register defaults to powering down some 3.3-V pins to save power. For more details on the VDD3P3V_PWDN register and which 3.3-V pins default to powerup or powerdown, [Section 3.2, Power Considerations](#). The pins that default to powerdown, are both reset to powerdown and high-impedance. They remain in that state until configured otherwise by VDD3P3_PWDN and PINMUX0/PINMUX1 programming.

- **Default Power Down Pin Group:** GP[4]/PWM1, ACLKR0/CLKX0/GP[99], AFSR0/DR0/GP[100], AHCLKR0/CLKR0/GP[101], AXR0[3]/FSR0/GP[102], AXR0[2]/FSX0/GP[103], AXR0[1]/DX0/GP[104], AXR0/FSR1/GP[105], ACLKX0/CLKX1/GP[106], AFSX0/DX1/GP[107], AHCLKX0/CLKR1/GP[108], AMUTEIN0/FSX1/GP[109], AMUTE0/DR1/GP[110], TOUT1L/UTXD1/GP[55], TINP1L/URXD1/GP[56], CLKS0/TOUT0L/GP[97], CLKS1/TINP0L/GP[98], URXD0/GP[85], UTXD0/GP[86], UCTS0/GP[87], and URTS0/PWM0/GP[88].

All Other Pins

During $\overline{\text{RESETOUT}}$ assertion (low), all other pins are controlled by the default peripheral. The default peripheral is determined by the default settings of the PINMUX0 or PINMUX1 registers.

Some of the PINMUX0/PINMUX1 settings are determined by configuration pins latched at reset. To determine the reset behavior of these pins, see [Section 3.7, Multiplexed Pin Configurations](#) and read the rest of the this subsection to understand how that default peripheral controls the pin.

The reset behaviors for all these other pins are categorized as follows (also see [Figure 6-7](#) and [Figure 6-8](#) in [Section 6.5.9, Reset Electrical Data/Timing](#)):

- **Z+/Low Group (Z Longer-to-Low Group):** These pins are 3-stated when device-level global reset source (e.g., POR, $\overline{\text{RESET}}$ or Max Reset) is asserted. These pins remain 3-stated throughout $\overline{\text{RESETOUT}}$ assertion. When $\overline{\text{RESETOUT}}$ is deasserted, these pins drive a logic low.
- **Z+/High Group (Z Longer-to-High Group):** These pins are 3-stated when device-level global reset source (e.g., POR, $\overline{\text{RESET}}$ or Max Reset) is asserted. These pins remain 3-stated throughout $\overline{\text{RESETOUT}}$ assertion. When $\overline{\text{RESETOUT}}$ is deasserted, these pins drive a logic high.
- **Z+/Invalid Group (Z Longer-to-Invalid Group):** These pins are 3-stated when device-level global reset source (e.g., POR, $\overline{\text{RESET}}$ or Max Reset) is asserted. These pins remain 3-stated throughout $\overline{\text{RESETOUT}}$ assertion. When $\overline{\text{RESETOUT}}$ is deasserted, these pins drive an invalid value until configured otherwise by their respective peripheral (after the peripheral is enabled by the PSC).
- **Z Group:** These pins are 3-stated by default, and these pins remain 3-stated throughout $\overline{\text{RESETOUT}}$ assertion. When $\overline{\text{RESETOUT}}$ is deasserted, these pins remain 3-stated until configured otherwise by

their respective peripheral (after the peripheral is enabled by the PSC).

- **Low Group:** These pins are low by default, and remain low until configured otherwise by their respective peripheral (after the peripheral is enabled by the PSC).
- **High Group:** These pins are high by default, and remain high until configured otherwise by their respective peripheral (after the peripheral is enabled by the PSC).
- **Z/Low Group (Z-to-Low Group):** These pins are 3-stated when device-level global reset source (e.g., POR, RESET or Max Reset) is asserted. When the reset source is deasserted, these pins drive a logic low.
- **Z/High Group (Z-to-High Group):** These pins are 3-stated when device-level global reset source (e.g., POR, RESET or Max Reset) is asserted. When reset source is deasserted, these pins drive a logic high.
- **Clock Group:** These clock pins are toggling by default. They paused momentarily before RESETOUT is deasserted (high). The *only* pin in the Clock Group is CLKOUT0.

This is a list of possible default peripherals and how they control the pins during reset:

- **GPIO:** All GPIO pins behave according to Z Group.
Note: The following EMIFA list *only* includes pins that can default to function as EMIFA signals.
- **EMIFA:** These EMIFA signals are multiplexed with boot and configuration pins: EM_A[4], EM_A[2:0], EM_BA[0], EM_BA[1]; therefore, they are forced 3-stated throughout RESETOUT.
 - **Z+/Low Group:** EM_A[4], EM_A[2:0]
 - **Z+/High Group:** EM_BA[0], EM_BA[1], $\overline{\text{EM_OE}}$, $\overline{\text{EM_WE}}$
 - **Z+/Invalid Group:** EM_D[15:0]
 - **Z/Low Group:** EM_A[21:5], EM_A[3], EM_R $\overline{\text{W}}$
 - **Z/High Group:** $\overline{\text{EM_CS2}}$
 - **Z Group:** EM_WAIT
- **DDR2 Memory Controller:**
 - **Clock Group:** DDR_CLK, $\overline{\text{DDR_CLK}}$
 - **DDR2 Z Group:** DDR_DQM[3:0], DDR_DQS[3:0], DDR_D[31:0]
 - **DDR2 Low Group:** DDR_CKE, DDR_BS[2:0], DDR_A[12:0]
 - **DDR2 High Group:** $\overline{\text{DDR_CS}}$, $\overline{\text{DDR_WE}}$, $\overline{\text{DDR_RAS}}$, $\overline{\text{DDR_CAS}}$
- **PCI:** All PCI pins behave according to Z Group.
- **I2C:** All I2C pins behave according to Z Group.
- **JTAG:** TDO, EMU0, and EMU1 pins behave according to Z Group. TCK, TDI, TMS, and $\overline{\text{TRST}}$ are input-only pins.
- **Clock:** CLKOUT0

For more information on the pin behaviors during device-level global reset, see [Figure 6-7](#) and [Figure 6-8](#) in [Section 6.5.9, Reset Electrical Data/Timing](#).

6.5.9 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 6-11. Timing Requirements for Reset (see Figure 6-7 and Figure 6-8)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|------------------|--|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 1 | $t_{w(RESSET)}$ | Pulse duration, \overline{POR} low or \overline{RESET} low | 12C ⁽¹⁾ | | ns |
| 4 | $t_{su(CONFIG)}$ | Setup time, boot and configuration pins valid before \overline{POR} high or \overline{RESET} high ⁽²⁾ | 12C ⁽¹⁾ | | ns |
| 5 | $t_{h(CONFIG)}$ | Hold time, boot and configuration pins valid after \overline{POR} high or \overline{RESET} high ⁽²⁾ | 0 | | ns |

(1) C = 1/MXI clock frequency in ns. The device clock source *must* be stable and at a valid frequency prior to meeting the $t_{w(RESSET)}$ requirement.

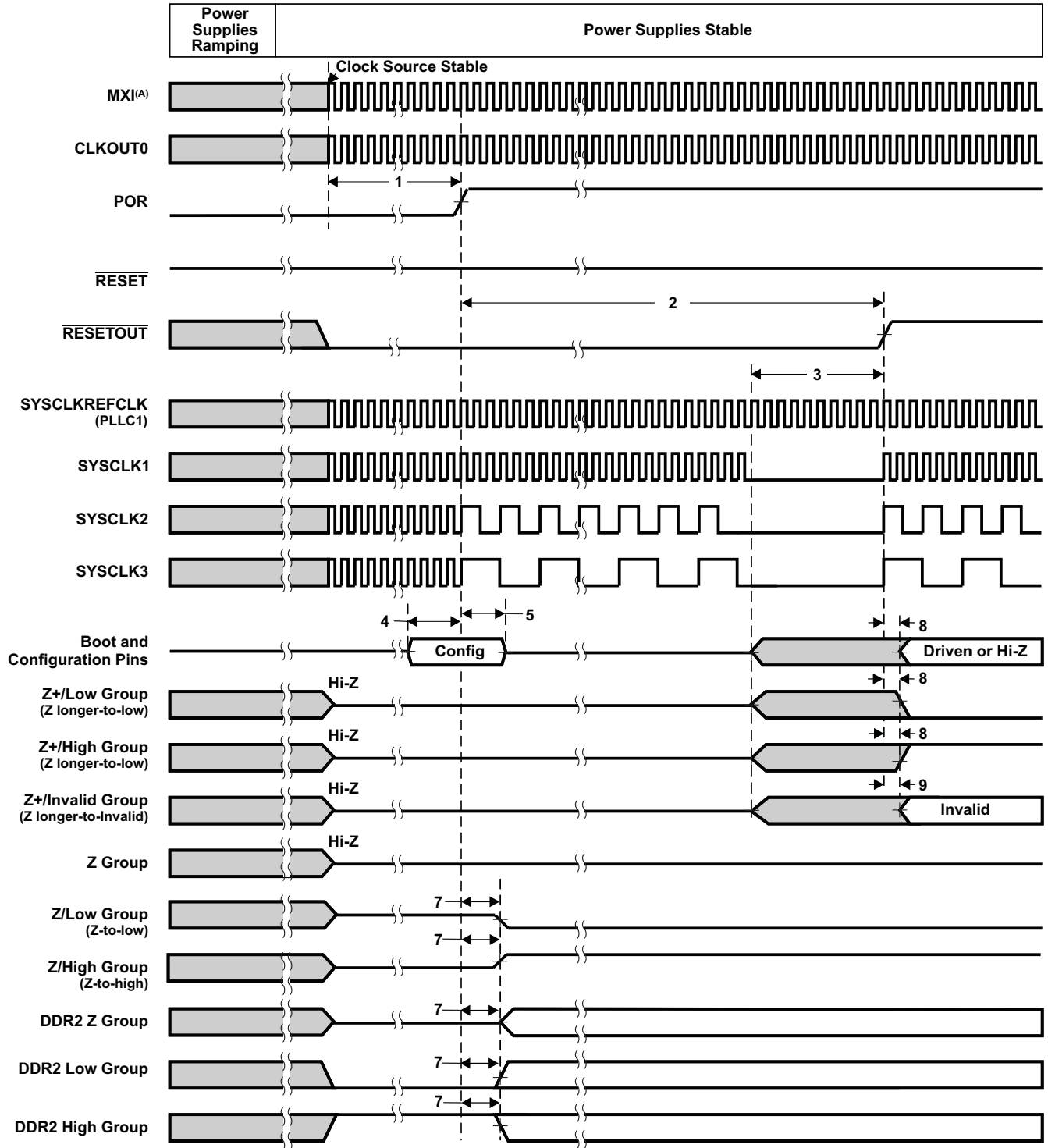
(2) For the list of boot and configuration pins, see Table 2-7, *Boot Terminal Functions*.

Table 6-12. Switching Characteristics Over Recommended Operating Conditions During Reset⁽¹⁾
(see Figure 6-8)

| NO. | PARAMETER | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---------------------|--|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 2 | $t_d(RSTH-RSTOUTH)$ | Delay time, \overline{POR} high or \overline{RESET} high to $\overline{RESETOUT}$ high | 1900C | | ns |
| 3 | $t_w(PAUSE)$ | Pulse duration, SYSCLKs paused (low) before $\overline{RESETOUT}$ high | 10C | 10C | ns |
| 6 | $t_d(RSTL-IV)$ | Delay time, \overline{POR} low or \overline{RESET} low to pins invalid | 20 | | ns |
| 7 | $t_d(RSTH-V)$ | Delay time, \overline{POR} high or \overline{RESET} high to pins valid | 20 | | ns |
| 8 | $t_d(RSTOUTH-V)$ | Delay time, $\overline{RESETOUT}$ high to pins valid | 0 | | ns |
| 9 | $t_d(RSTOUTH-IV)$ | Delay time, $\overline{RESETOUT}$ high to pins invalid | 12C | | ns |

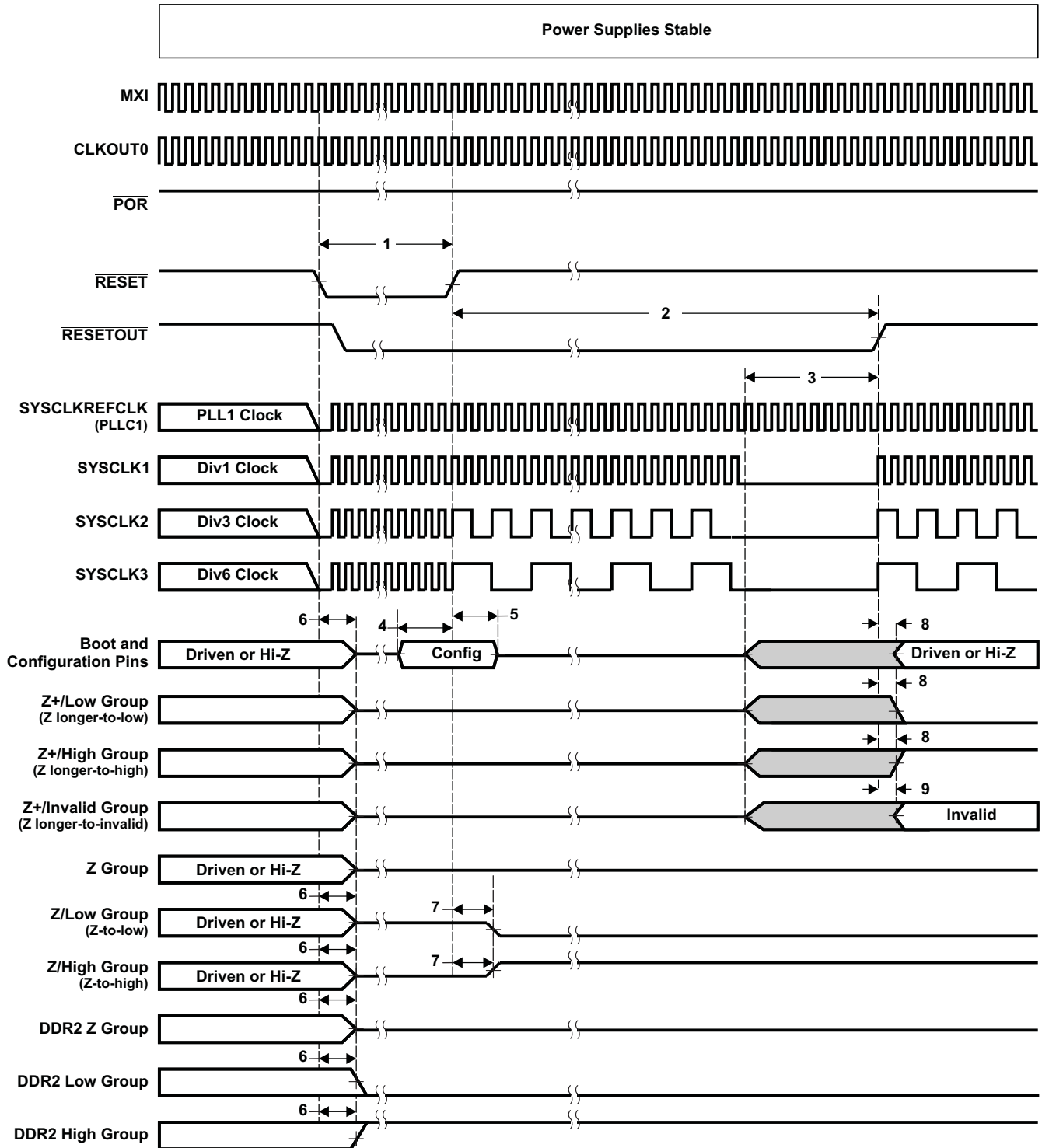
(1) C = 1/CLKIN1 clock frequency in ns.

Figure 6-7 shows the Power-Up Timing. Figure 6-8 shows the Warm Reset (\overline{RESET}) Timing and Max Reset Timing are identical to Warm Reset Timing, except the boot and configuration pins are *not* relatched *and* the BOOTCFG register retains its previous value latched before the Max Reset were initiated.



- A. Power supplies and MXI must be stable before the start of $t_{W(RESET)}$.
- B. Pin reset behavior depends on which peripheral defaults to controlling the multiplexed pin. For more details on what pin group (e.g., Z Group, Z/Low Group, Z/High Group, etc.) each pin belongs to, see [Section 6.5.8, Pin Behaviors at Reset](#).

Figure 6-7. Power-Up Timing



- A. Pin reset behavior depends on which peripheral defaults to controlling the multiplexed pin. For more details on what pin group (e.g., Z Group, Z/Low Group, Z/High Group, etc.) each pin belongs to, see [Section 6.5.8, Pin Behaviors at Reset](#).

Figure 6-8. Warm Reset ($\overline{\text{RESET}}$) Timing

6.6 External Clock Input From MXI/CLKIN Pin

The C6424 device includes two options to provide an external clock input:

- Use an on-chip oscillator with external crystal.
- Use an external 1.8-V LVCMOS-compatible clock input.

The optimal external clock input frequency is 15–30 MHz. [Section 6.6.1](#) provides more details on Option 1, using an on-chip oscillator with external crystal. [Section 6.6.2](#) provides details on Option 2, using an external 1.8-V LVCMOS-compatible clock input.

6.6.1 Clock Input Option 1- Crystal

In this option, a crystal is used as the external clock input to the C6424.

The 15–30-MHz oscillator provides the reference clock for all C6424 subsystems and peripherals. The on-chip oscillator requires an external 15–30-MHz crystal connected across the MXI and MXO pins, along with two load capacitors, as shown in [Figure 6-9](#). The external crystal load capacitors **must** be connected only to the 15–30-MHz oscillator ground pin (MXV_{SS}). **Do not** connect to board ground (V_{SS}). The MXV_{DD} pin can be connected to the same 1.8 V power supply as DV_{DDR2}.

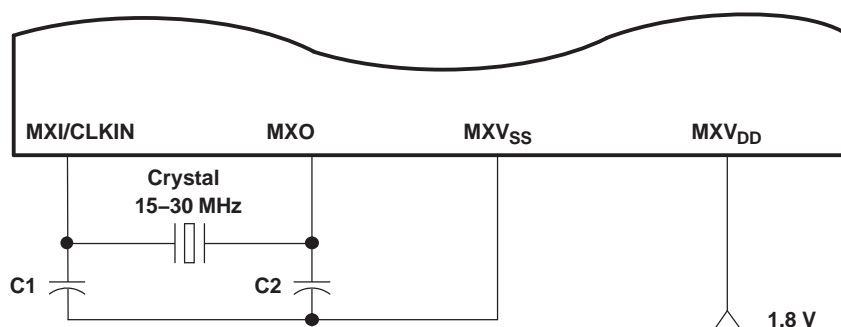


Figure 6-9. 15–30-MHz System Oscillator

The load capacitors, C1 and C2, should be chosen such that the equation is satisfied (typical values are C1 = C2 = 10 pF). C_L in the equation is the load specified by the crystal manufacturer. All discrete components used to implement the oscillator circuit should be placed as close as possible to the associated oscillator pins (MXI and MXO) and to the MXV_{SS} pin.

$$C_L = \frac{C_1 C_2}{(C_1 + C_2)}$$

Table 6-13. Input Requirements for Crystal ⁽¹⁾

| PARAMETER | | MIN | TYP | MAX | UNIT |
|-----------|---|-----|-----|-----|------|
| | Start-up time (from power up until oscillating at stable frequency of 30 MHz) | | | 4 | ms |
| | Oscillation frequency | 15 | | 30 | MHz |
| | ESR | | | 60 | Ω |

(1) For audio applications, stability of the input clock is very important. The user should select crystals with low enough ppm to ensure good audio quality for the specific application.

6.6.2 Clock Input Option 2—1.8-V LVCMOS-Compatible Clock Input

In this option, a 1.8-V LVCMOS-Compatible Clock Input is used as the external clock input to the C6424. The external connections are shown in [Figure 6-10](#). The MXI/CLKIN pin is connected to the 1.8-V LVCMOS-Compatible clock source. The MXO pin is left unconnected. The MXV_{SS} pin is connected to board ground (V_{SS}). The MXV_{DD} pin can be connected to the same 1.8-V power supply as DV_{DDR2}.

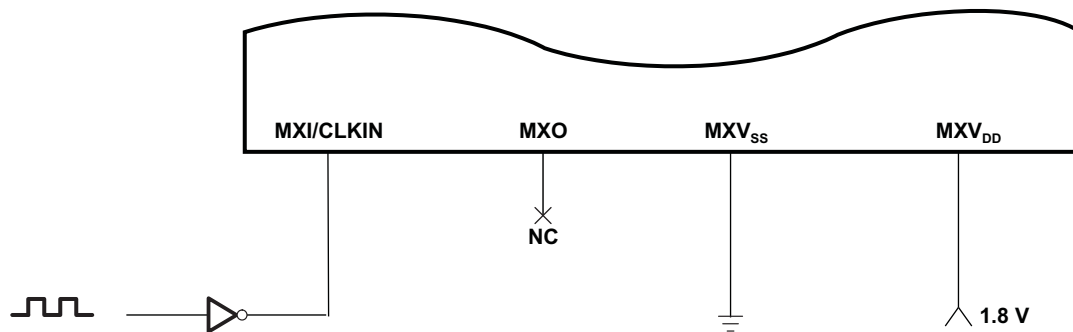


Figure 6-10. 1.8-V LVCMOS-Compatible Clock Input

The clock source **must** meet the MXI/CLKIN timing requirements in [Section 6.7.4, Clock PLL Electrical Data/Timing \(Input and Output Clocks\)](#).

6.7 Clock PLLs

There are two independently controlled PLLs on C6424. PLL1 generates the frequencies required for the DSP, DMA, and other peripherals. PLL2 generates the frequencies required for the DDR2 interface. The recommended reference clock for both PLLs is the 15–30-MHz crystal input.

6.7.1 PLL1 and PLL2

Both PLL1 and PLL2 power is supplied externally via the 1.8 V PLL power-supply pin (PLL_{PWR18}). An external EMI filter circuit **must** be added to PLL_{PWR18}, as shown in Figure 6-11. The 1.8-V supply of the EMI filter must be from the same 1.8-V power plane supplying the device's 1.8-V I/O power-supply pins (DV_{DDDR2}). TI recommends EMI filter manufacturer Murata, part number NFM18CC222R1C3.

All PLL external components (C1, C2, and the EMI Filter) **must** be placed as close to the 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 in Figure 6-11. For reduced PLL jitter, maximize the spacing between switching signals and the PLL external components (C1, C2, and the EMI Filter).

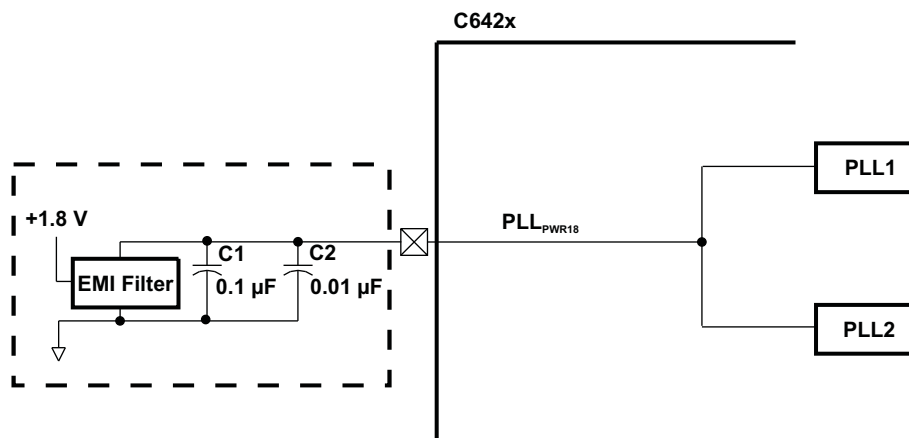


Figure 6-11. PLL1 and PLL2 External Connection

The minimum CLKIN rise and fall times should also be observed. For the input clock timing requirements, see Section 6.7.4, *Clock PLL Electrical Data/Timing (Input and Output Clocks)*.

There is an allowable range for PLL multiplier (PLLM). There is a minimum and maximum operating frequency for MXI/CLKIN, PLL_{OUT}, and the device clocks (SYSCLKs). The PLL Controllers **must** be configured not to exceed any of these constraints documented in this section (certain combinations of external clock inputs, internal dividers, and PLL multiply ratios might not be supported). For these constraints (ranges), see Table 6-14 through Table 6-16.

Table 6-14. PLL1 and PLL2 Multiplier Ranges

| PLL MULTIPLIER (PLLM) | MIN | MAX |
|-----------------------|-----|-----|
| PLL1 Multiplier | x14 | x32 |
| PLL2 Multiplier | x14 | x32 |

Table 6-15. PLLC1 Clock Frequency Ranges

| CLOCK SIGNAL NAME | | MIN | MAX | UNIT |
|---|------------------------------|-----|-----|------|
| MXI/CLKIN ⁽¹⁾ | | 15 | 30 | MHz |
| PLLOUT CV _{DD} = 1.2 V | -7 devices | 300 | 700 | MHz |
| | -6/-5/-4/-Q6/-Q5/-Q4 devices | 300 | 600 | MHz |
| PLLOUT CV _{DD} = 1.05 V | -7 devices | 300 | 520 | MHz |
| | -6/-5/-4/-L/-Q5 devices | 300 | 520 | MHz |
| SYSCLK1 ⁽²⁾ (CLKDIV1 Domain), CV _{DD} = 1.2 V | -7 devices | | 700 | MHz |
| | -6/-Q6 devices | | 600 | MHz |
| | -5/-Q5 devices | | 500 | MHz |
| SYSCLK1 ⁽²⁾ (CLKDIV1 Domain), CV _{DD} = 1.05 V | -4/-Q4 devices | | 400 | MHz |
| | -7 devices | | 520 | MHz |
| | -6 devices | | 450 | MHz |
| | -5/-Q5/-L devices | | 400 | MHz |
| | -4 devices | | 350 | MHz |

(1) MXI/CLKIN input clock is used for both PLL Controllers (PLLC1 and PLLC2).

(2) Applies to "tape and reel" part number counterparts as well. For more information, see [Section 2.7](#), *Device and Development-Support Tool Nomenclature*.

Table 6-16. PLLC2 Clock Frequency Ranges

| CLOCK SIGNAL NAME | | MIN | MAX | UNIT |
|----------------------------|----------------------------|-----|-----|------|
| MXI/CLKIN ⁽¹⁾ | | 15 | 30 | MHz |
| PLLOUT | At 1.2-V CV _{DD} | 300 | 900 | MHz |
| | At 1.05-V CV _{DD} | 300 | 666 | MHz |
| PLL2_SYSCLK1 (to DDR2 PHY) | | | 333 | MHz |

(1) MXI/CLKIN input clock is used for both PLL Controllers (PLLC1 and PLLC2).

Both PLL1 and PLL2 have stabilization, lock, and reset timing requirements that **must** be followed.

The PLL stabilization time is the amount of time that **must** be allotted for the internal PLL regulators to become stable after the PLL is powered up (after PLLCTL.PLLPWRDN bit goes through a 1-to-0 transition). The PLL should *not* be operated until this stabilization time has expired. This stabilization step **must** be applied after these resets—a Power-on Reset, a Warm Reset, or a Max Reset, as the PLLCTL.PLLPWRDN bit resets to a "1". For the PLL stabilization time value, see [Table 6-17](#).

The PLL reset time is the amount of wait time needed for the PLL to properly reset (writing PLLRST = 0) before bringing the PLL out of reset (writing PLLRST = 1). For the PLL reset time value, see [Table 6-17](#).

The PLL lock time is the amount of time needed from when the PLL is taken out of reset (PLLRST = 1 with PLEN = 0) to when to when the PLL controller can be switched to PLL mode (PLEN = 1). For the PLL lock time value, see [Table 6-17](#).

Table 6-17. PLL1 and PLL2 Stabilization, Lock, and Reset Times

| PLL STABILIZATION/LOCK/RESET TIME | MIN | TYP | MAX | UNIT |
|-----------------------------------|---------------------|-----|----------------------|------|
| PLL Stabilization Time | 150 | | | μs |
| PLL Lock Time | | | 2000C ⁽¹⁾ | ns |
| PLL Reset Time | 128C ⁽¹⁾ | | | ns |

(1) C = CLKIN cycle time in ns. For example, when MXI/CLKIN frequency is 25 MHz, use C = 40 ns.

For details on the PLL initialization software sequence, see the *TMS320C642x DSP Phase-Locked Loop Controller (PLL) User's Guide* (literature number [SPRUES0](#)).

For more information on the clock domains and their clock ratio restrictions, see [Section 6.3.4, C6424 Power and Clock Domains](#).

6.7.2 PLL Controller Register Description(s)

A summary of the PLL controller registers is shown in [Table 6-18](#). For more details, see the *TMS320C642x DSP Phase-Locked Loop Controller (PLL) User's Guide* (literature number [SPRUES0](#)).

Table 6-18. PLL and Reset Controller Registers Memory Map

| HEX ADDRESS RANGE | REGISTER ACRONYM | DESCRIPTION |
|----------------------------------|------------------|---|
| PLL1 Controller Registers | | |
| 0x01C4 0800 | PID | Peripheral ID Register |
| 0x01C4 08E4 | RSTYPE | Reset Type Register |
| 0x01C4 0900 | PLLCTL | PLL Controller 1 PLL Control Register |
| 0x01C4 0910 | PLLM | PLL Controller 1 PLL Multiplier Control Register |
| 0x01C4 0918 | PLLDIV1 | PLL Controller 1 Divider 1 Register (SYSCLK1) |
| 0x01C4 091C | PLLDIV2 | PLL Controller 1 Divider 2 Register (SYSCLK2) |
| 0x01C4 0920 | PLLDIV3 | PLL Controller 1 Divider 3 Register (SYSCLK3) |
| 0x01C4 0924 | OSCDIV1 | PLL Controller 1 Oscillator Divider 1 Register (OBSCLK) [CLKOUT0 pin] |
| 0x01C4 0928 | – | Reserved |
| 0x01C4 092C | – | Reserved |
| 0x01C4 0938 | PLLCMD | PLL Controller 1 Command Register |
| 0x01C4 093C | PLLSTAT | PLL Controller 1 Status Register (Shows PLLC1 Status) |
| 0x01C4 0940 | ALNCTL | PLL Controller 1 Clock Align Control Register (Indicates Which SYSCLKs Need to be Aligned for Proper Device Operation) |
| 0x01C4 0944 | DCHANGE | PLL Controller 1 PLLDIV Divider Ratio Change Status Register (Indicates if SYSCLK Divide Ratio has Been Modified) |
| 0x01C4 0948 | CKEN | PLL Controller 1 Clock Enable Control Register |
| 0x01C4 094C | CKSTAT | PLL Controller 1 Clock Status Register (For All Clocks Except SYSCLKx) |
| 0x01C4 0950 | SYSTAT | PLL Controller 1 SYSCLK Status Register (Indicates SYSCLK on/off Status) |
| 0x01C4 0960 | – | Reserved |
| 0x01C4 0964 | – | Reserved |
| PLL2 Controller Registers | | |
| 0x01C4 0C00 | PID | Peripheral ID Register |
| 0x01C4 0D00 | PLLCTL | PLL Controller 2 PLL Control Register |
| 0x01C4 0D10 | PLLM | PLL Controller 2 PLL Multiplier Control Register |
| 0x01C4 0D18 | PLLDIV1 | PLL Controller 2 Divider 1 Register (SYSCLK1) |
| 0x01C4 0D1C | – | Reserved |
| 0x01C4 0D20 - 0x01C4 0D2C | – | Reserved |
| 0x01C4 0D2C | BPDIV | PLL Controller 2 Bypass Divider Register (SYSCLKBP) |
| 0x01C4 0D38 | PLLCMD | PLL Controller 2 Command Register |
| 0x01C4 0D3C | PLLSTAT | PLL Controller 2 Status Register (Shows PLLC2 Status) |
| 0x01C4 0D40 | ALNCTL | PLL Controller 2 Clock Align Control Register (Indicates Which SYSCLKs Need to be Aligned for Proper Device Operation) |
| 0x01C4 0D44 | DCHANGE | PLL Controller 2 PLLDIV Divider Ratio Change Status Register (Indicates if SYSCLK Divide Ratio has Been Modified) |
| 0x01C4 0D48 | – | Reserved |
| 0x01C4 0D4C | CKSTAT | PLL Controller 2 Clock Status Register (For All Clocks Except SYSCLKx) |
| 0x01C4 0D50 | SYSTAT | PLL Controller 2 SYSCLK Status Register (Indicates SYSCLK on/off Status) |
| 0x01C4 0D54 - 0x01C4 0FFF | – | Reserved |

6.7.3 Clock PLL Considerations with External Clock Sources

If the internal oscillator is bypassed, to minimize the clock jitter a single clean power supply should power both the C6424 device and the external clock oscillator circuit. The minimum CLKIN rise and fall times should also be observed. For the input clock timing requirements, see [Section 6.7.4, Clock PLL Electrical Data/Timing \(Input and Output Clocks\)](#).

Rise/fall times, duty cycles (high/low pulse durations), and the load capacitance of the external clock source must meet the device requirements in this data manual (see [Section 5.3, Electrical Characteristics Over Recommended Ranges of Supply Voltage and Operating Case Temperature](#) and [Section 6.7.4, Clock PLL Electrical Data/Timing \(Input and Output Clocks\)](#)).

6.7.4 Clock PLL Electrical Data/Timing (Input and Output Clocks)

Table 6-19. Timing Requirements for MXI/CLKIN^{(1) (2) (3) (4)} (see [Figure 6-12](#))

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------------|--------------------------------|-------------------------------|-------|------|
| | | | MIN | MAX | |
| 1 | $t_c(\text{MXI})$ | Cycle time, MXI/CLKIN | 33.3 | 66.7 | ns |
| 2 | $t_w(\text{MXIH})$ | Pulse duration, MXI/CLKIN high | 0.45C | 0.55C | ns |
| 3 | $t_w(\text{MXIL})$ | Pulse duration, MXI/CLKIN low | 0.45C | 0.55C | ns |
| 4 | $t_t(\text{MXI})$ | Transition time, MXI/CLKIN | | 0.05C | ns |
| 5 | $t_J(\text{MXI})$ | Period jitter, MXI/CLKIN | | 0.02C | ns |

- (1) The MXI/CLKIN frequency and PLL multiply factor should be chosen such that the resulting clock frequency is within the specific range for CPU operating frequency. For example, for a -6 speed device with a 25 MHz CLKIN frequency, the PLL multiply factor should be ≤ 24 .
- (2) The reference points for the rise and fall transitions are measured at V_{IL} MAX and V_{IH} MIN.
- (3) For more details on the PLL multiplier factors, see *TMS320C642x DSP Phase-Locked Loop Controller (PLL) User's Guide* (literature number [SPRUJES0](#)).
- (4) C = CLKIN cycle time in ns. For example, when MXI/CLKIN frequency is 30 MHz, use C = 33.3 ns.

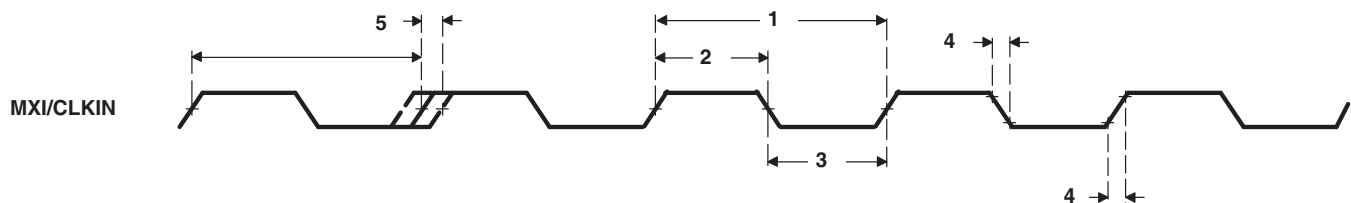


Figure 6-12. MXI/CLKIN Timing

Table 6-20. Switching Characteristics Over Recommended Operating Conditions for CLKOUT0^{(1) (2)}
(see [Figure 6-13](#))

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---|-------------------------------|-------|------|
| | | MIN | MAX | |
| 1 | t_c Cycle time, CLKOUT0 | 33.3 | 66.7 | ns |
| 2 | $t_w(\text{CLKOUT0H})$ Pulse duration, CLKOUT0 high | 0.45P | 0.55P | ns |
| 3 | $t_w(\text{CLKOUT0L})$ Pulse duration, CLKOUT0 low | 0.45P | 0.55P | ns |
| 4 | $t_t(\text{CLKOUT0})$ Transition time, CLKOUT0 | | 0.05P | ns |

- (1) The reference points for the rise and fall transitions are measured at $V_{OL\ MAX}$ and $V_{OH\ MIN}$.
 (2) $P = 1/\text{CLKOUT0 clock frequency}$ in nanoseconds (ns). For example, when CLKOUT0 frequency is 30 MHz, use $P = 33.3$ ns.

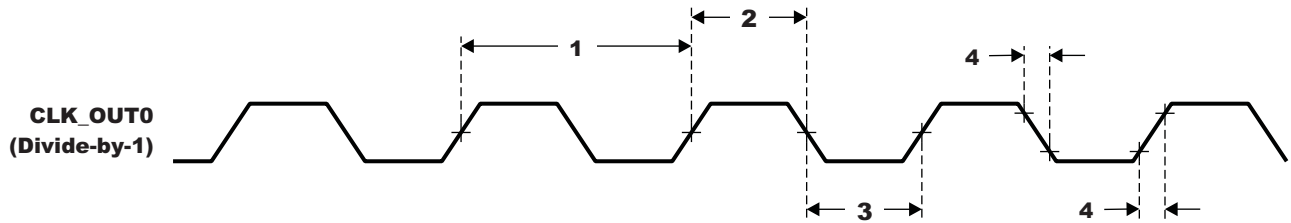


Figure 6-13. CLKOUT0 Timing

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 and is listed in [Table 6-21](#). Also, the interrupt controller controls the generation of the CPU exception and emulation interrupts. The NMI input to the C64x+ DSP interrupt controller is not connected internally, therefore the NMI interrupt is not available. [Table 6-22](#) summarizes the C64x+ interrupt controller registers and memory locations. For more details on DSP interrupt control, see the *TMS320C64x+ DSP Megamodule Reference Guide* (literature number [SPRU871](#)).

Table 6-21. C6424 DSP System Event Mapping

| DSP SYSTEM EVENT NUMBER | ACRONYM | SOURCE | DSP SYSTEM EVENT NUMBER | ACRONYM | SOURCE |
|-------------------------|-----------------|---------------------------|-------------------------|-------------|--|
| 0 | EVT0 | C64x+ Int Ctl 0 | 64 | GPIO0 | GPIO |
| 1 | EVT1 | C64x+ Int Ctl 1 | 65 | GPIO1 | GPIO |
| 2 | EVT2 | C64x+ Int Ctl 2 | 66 | GPIO2 | GPIO |
| 3 | EVT3 | C64x+ Int Ctl 3 | 67 | GPIO3 | GPIO |
| 4 | TINTL0 | Timer 0 – TINT12 | 68 | GPIO4 | GPIO |
| 5 | TINTH0 | Timer 0 – TINT34 | 69 | GPIO5 | GPIO |
| 6 | TINTL1 | Timer 1 – TINT12 | 70 | GPIO6 | GPIO |
| 7 | TINTH1 | Timer 1 – TINT34 | 71 | GPIO7 | GPIO |
| 8 | WDINT | Timer 2 – TINT12 | 72 | GPIOBNK0 | GPIO |
| 9 | EMU_DTDMA | C64x+ EMC | 73 | GPIOBNK1 | GPIO |
| 10 | | Reserved | 74 | GPIOBNK2 | GPIO |
| 11 | EMU_RTDXR | C64x+ RTDX | 75 | GPIOBNK3 | GPIO |
| 12 | EMU_RTDXTX | C64x+ RTDX | 76 | GPIOBNK4 | GPIO |
| 13 | IDMAINT0 | C64x+ EMC 0 | 77 | GPIOBNK5 | GPIO |
| 14 | IDMAINT1 | C64x+ EMC 1 | 78 | GPIOBNK6 | GPIO |
| 15 | | Reserved | 79 | | Reserved |
| 16 | | Reserved | 80 | PWM0 | PWM0 |
| 17 | | Reserved | 81 | PWM1 | PWM1 |
| 18 | | Reserved | 82 | PWM2 | PWM2 |
| 19 | | Reserved | 83 | IICINT0 | I2C |
| 20 | | Reserved | 84 | UARTINT0 | UART0 |
| 21 | | Reserved | 85 | UARTINT1 | UART1 |
| 22 | | Reserved | 86 | | Reserved |
| 23 | | Reserved | 87 | | Reserved |
| 24 | | Reserved | 88 | | Reserved |
| 25 | | Reserved | 89 | | Reserved |
| 26 | | Reserved | 90 | | Reserved |
| 27 | | Reserved | 91 | | Reserved |
| 28 | | Reserved | 92 | | Reserved |
| 29 | | Reserved | 93 | | Reserved |
| 30 | | Reserved | 94 | | Reserved |
| 31 | | Reserved | 95 | | Reserved |
| 32 | | Reserved | 96 | INTERR | C64x+ Interrupt Controller Dropped CPU Interrupt Event |
| 33 | | Reserved | 97 | EMC_IDMAERR | C64x+ EMC Invalid IDMA Parameters |
| 34 | EDMA3CC_GINT | EDMACC Global Interrupt | 98 | | Reserved |
| 35 | EDMA3CC_INT0 | EDMACC Interrupt Region 0 | 99 | | Reserved |
| 36 | EDMA3CC_INT1 | EDMACC Interrupt Region 1 | 100 | | Reserved |
| 37 | EDMA3CC_ERRINT | EDMA CC Error | 101 | | Reserved |
| 38 | EDMA3TC_ERRINT0 | EDMA TC0 Error | 102 | | Reserved |
| 39 | EDMA3TC_ERRINT1 | EDMA TC1 Error | 103 | | Reserved |
| 40 | EDMA3TC_ERRINT2 | EDMA TC2 Error | 104 | | Reserved |
| 41 | PSCINT | PSC ALLINT | 105 | | Reserved |

Table 6-21. C6424 DSP System Event Mapping (continued)

| DSP SYSTEM EVENT NUMBER | ACRONYM | SOURCE | DSP SYSTEM EVENT NUMBER | ACRONYM | SOURCE |
|-------------------------|---------|------------------------|-------------------------|-----------|-------------|
| 42 | | Reserved | 106 | | Reserved |
| 43 | EMACINT | EMAC Memory Controller | 107 | | Reserved |
| 44 | | Reserved | 108 | | Reserved |
| 45 | | Reserved | 109 | | Reserved |
| 46 | | Reserved | 110 | | Reserved |
| 47 | HPIINT | HPI | 111 | | Reserved |
| 48 | MBXINT0 | McBSP0 Transmit | 112 | | Reserved |
| 49 | MBRINT0 | McBSP0 Receive | 113 | PMC_ED | C64x+ PMC |
| 50 | MBXINT1 | McBSP1 Transmit | 114 | | Reserved |
| 51 | MBRINT1 | McBSP1 Receive | 115 | | Reserved |
| 52 | | Reserved | 116 | UMCED1 | C64x+ UMC 1 |
| 53 | DDRINT | DDR2 Memory Controller | 117 | UMCED2 | C64x+ UMC 2 |
| 54 | EMIFAIN | EMIFA | 118 | PDCINT | C64x+ PDC |
| 55 | VLQINT | VLYNQ | 119 | SYSCMPA | C64x+ SYS |
| 56 | PCIINT | PCI | 120 | PMCCMPA | C64x+ PMC |
| 57 | | Reserved | 121 | PMCDMPA | C64x+ PMC |
| 58 | | Reserved | 122 | DMCCMPA | C64x+ DMC |
| 59 | AXINT0 | McASP0 Transmit | 123 | DMCDMPA | C64x+ DMC |
| 60 | ARINT0 | McASP0 Receive | 124 | UMCCMPA | C64x+ UMC |
| 61 | | Reserved | 125 | UMCDMPA | C64x+ UMC |
| 62 | | Reserved | 126 | EMCCMPA | C64x+ EMC |
| 63 | | Reserved | 127 | EMCBUSERR | C64x+ EMC |

Table 6-22. 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 0180 | INTXSTAT | Interrupt exception status |
| 0x0180 0184 | INTXCLR | Interrupt exception clear |
| 0x0180 0188 | INTDMASK | Dropped interrupt mask register |

6.9 External Memory Interface (EMIF)

C6424 supports several memory and external device interfaces, including:

- Asynchronous EMIF (EMIFA) for interfacing to NOR Flash, SRAM, etc.
- NAND Flash

6.9.1 Asynchronous EMIF (EMIFA)

The C6424 Asynchronous EMIF (EMIFA) provides a 16-bit data bus, an address bus width up to 24-bits, and 4 chip selects, along with memory control signals. These signals are multiplexed between these peripherals:

- EMIFA and NAND interfaces
- EMAC (RMII)
- PCI
- GPIO

6.9.2 NAND (NAND, SmartMedia, xD)

The EMIFA interface provides both the asynchronous EMIF and NAND interfaces. Four chip selects are provided and each are individually configurable to provide either EMIFA or NAND support. The NAND features supported are as follows.

- NAND flash on up to 4 asynchronous chip selects.
- 8- or 16-bit data bus width
- Programmable cycle timings.
- Performs ECC calculation.
- NAND Mode also supports SmartMedia and xD memory cards
- Boot ROM supports booting of the C6424 from NAND flash located at CS2

The memory map for EMIFA and NAND registers is shown in [Table 6-23](#). For more details on the EMIFA and NAND interfaces, the *TMS320C642x DSP Peripherals Overview Reference Guide* (literature number SPRUEM3) and the *TMS320C642x Asynchronous External Memory Interface (EMIF) User's Guide* (literature number SPRUEM7).

Table 6-23. EMIFA/NAND Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|-----------|---|
| 0x01E0 0000 | RCSR | Revision Code and Status Register |
| 0x01E0 0004 | AWCCR | Asynchronous Wait Cycle Configuration Register |
| 0x01E0 0008 - 0x01E0 000F | | Reserved |
| 0x01E0 0010 | A1CR | Asynchronous 1 Configuration Register (CS2 Space) |
| 0x01E0 0014 | A2CR | Asynchronous 2 Configuration Register (CS3 Space) |
| 0x01E0 0018 | A3CR | Asynchronous 3 Configuration Register (CS4 Space) |
| 0x01E0 001C | A4CR | Asynchronous 4 Configuration Register (CS5 Space) |
| 0x01E0 0020 - 0x01E0 003F | - | Reserved |
| 0x01E0 0040 | EIRR | EMIF Interrupt Raw Register |
| 0x01E0 0044 | EIMR | EMIF Interrupt Mask Register |
| 0x01E0 0048 | EIMSR | EMIF Interrupt Mask Set Register |
| 0x01E0 004C | EIMCR | EMIF Interrupt Mask Clear Register |
| 0x01E0 0050 - 0x01E0 005F | - | Reserved |
| 0x01E0 0060 | NANDFCR | NAND Flash Control Register |
| 0x01E0 0064 | NANDFSR | NAND Flash Status Register |
| 0x01E0 0070 | NANDF1ECC | NAND Flash 1 ECC Register (CS2 Space) |
| 0x01E0 0074 | NANDF2ECC | NAND Flash 2 ECC Register (CS3 Space) |
| 0x01E0 0078 | NANDF3ECC | NAND Flash 3 ECC Register (CS4 Space) |
| 0x01E0 007C | NANDF4ECC | NAND Flash 4 ECC Register (CS5 Space) |
| 0x01E0 0080 - 0x01E0 0FFF | - | Reserved |

6.9.3 EMIFA Electrical Data/Timing

Table 6-24. Timing Requirements for Asynchronous Memory Cycles for EMIFA Module⁽¹⁾
(see [Figure 6-14](#) and [Figure 6-15](#))

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | UNIT |
|-------------------------|------------------------|---|--------|-----|------|
| | | MIN | NOM | MAX | |
| READS and WRITES | | | | | |
| 2 | $t_{w(EM_WAIT)}$ | Pulse duration, EM_WAIT assertion and deassertion | 2E | | ns |
| READS | | | | | |
| 12 | $t_{su(EMDV-EMOEH)}$ | Setup time, EM_D[15:0] valid before $\overline{EM_OE}$ high | 5 | | ns |
| 13 | $t_{h(EMOEH-EMDIV)}$ | Hold time, EM_D[15:0] valid after $\overline{EM_OE}$ high | 0 | | ns |
| 14 | $t_{su(EMWAIT-EMOEH)}$ | Setup time, EM_WAIT asserted before $\overline{EM_OE}$ high ⁽²⁾ | 4E + 5 | | ns |
| WRITES | | | | | |
| 28 | $t_{su(EMWAIT-EMWEH)}$ | Setup time, EM_WAIT asserted before $\overline{EM_WE}$ high ⁽²⁾ | 4E + 5 | | ns |

(1) E = SYSCLK3 period in ns for EMIFA. For example, when running the DSP CPU at 600 MHz, use E = 10 ns.

(2) Setup before end of STROBE phase (if no extended wait states are inserted) by which EM_WAIT must be asserted to add extended wait states. [Figure 6-16](#) and [Figure 6-17](#) describe EMIF transactions that include extended wait states inserted during the STROBE phase. However, cycles inserted as part of this extended wait period should not be counted; the 4E requirement is to the start of where the HOLD phase would begin if there were no extended wait cycles.

Table 6-25. Switching Characteristics Over Recommended Operating Conditions for Asynchronous Memory Cycles for EMIFA Module⁽¹⁾ ⁽²⁾ (see Figure 6-14 and Figure 6-15)

| NO | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | UNIT |
|-------------------------|--|--|-----|--------------------------------------|-----------------------|
| | | MIN | NOM | MAX | |
| READS and WRITES | | | | | |
| 1 | $t_d(\text{TURNAROUND})$ Turn around time | | | $(TA + 1) * E$ | ns |
| READS | | | | | |
| 3 | $t_c(\text{EMRCYCLE})$ EMIF read cycle time | | | $(RS + RST + RH + TA + 4) * E^{(3)}$ | ns |
| 4 | $t_{su}(\text{EMCSL-EMOEL})$ | Output setup time, $\overline{EM_CS}[5:2]$ low to $\overline{EM_OE}$ low (SS = 0) | | $(RS + 1) * E - 4$ | $(RS + 1) * E + 4$ ns |
| | | Output setup time, $\overline{EM_CS}[5:2]$ low to $\overline{EM_OE}$ low (SS = 1) | | -4 | 4 ns |
| 5 | $t_h(\text{EMOEH-EMCSH})$ | Output hold time, $\overline{EM_OE}$ high to $\overline{EM_CS}[5:2]$ high (SS = 0) | | $(RH + 1) * E - 4$ | $(RH + 1) * E + 4$ ns |
| | | Output hold time, $\overline{EM_OE}$ high to $\overline{EM_CS}[5:2]$ high (SS = 1) | | -4 | 4 ns |
| 6 | $t_{su}(\text{EMBAV-EMOEL})$ | Output setup time, $\overline{EM_BA}[1:0]$ valid to $\overline{EM_OE}$ low | | $(RS + 1) * E - 4$ | $(RS + 1) * E + 4$ ns |
| 7 | $t_h(\text{EMOEH-EMBAIV})$ | Output hold time, $\overline{EM_OE}$ high to $\overline{EM_BA}[1:0]$ invalid | | $(RH + 1) * E - 4$ | $(RH + 1) * E + 4$ ns |
| 8 | $t_{su}(\text{EMBAV-EMOEL})$ | Output setup time, $\overline{EM_A}[21:0]$ valid to $\overline{EM_OE}$ low | | $(RS + 1) * E - 4$ | $(RS + 1) * E + 4$ ns |
| 9 | $t_h(\text{EMOEH-EMBAIV})$ | Output hold time, $\overline{EM_OE}$ high to $\overline{EM_A}[21:0]$ invalid | | $(RH + 1) * E - 4$ | $(RH + 1) * E + 4$ ns |
| 10 | $t_w(\text{EMOEL})$ | $\overline{EM_OE}$ active low width | | $(RST + 1) * E^{(3)}$ | ns |
| 11 | $t_d(\text{EMWAITH-EMOEH})$ | Delay time from EM_WAIT deasserted to $\overline{EM_OE}$ high | | | $4E + 4$ ns |
| WRITES | | | | | |
| 15 | $t_c(\text{EMWCYCLE})$ EMIF write cycle time | | | $(WS + WST + WH + TA + 4) * E^{(3)}$ | ns |
| 16 | $t_{su}(\text{EMCSL-EMWEL})$ | Output setup time, $\overline{EM_CS}[5:2]$ low to $\overline{EM_WE}$ low (SS = 0) | | $(WS + 1) * E - 4$ | $(WS + 1) * E + 4$ ns |
| | | Output setup time, $\overline{EM_CS}[5:2]$ low to $\overline{EM_WE}$ low (SS = 1) | | -4 | 4 ns |
| 17 | $t_h(\text{EMWEH-EMCSH})$ | Output hold time, $\overline{EM_WE}$ high to $\overline{EM_CS}[5:2]$ high (SS = 0) | | $(WH + 1) * E - 4$ | $(WH + 1) * E + 4$ ns |
| | | Output hold time, $\overline{EM_WE}$ high to $\overline{EM_CS}[5:2]$ high (SS = 1) | | -4 | 4 ns |
| 18 | $t_{su}(\text{EMRNW-EMWEL})$ | Output setup time, $\overline{EM_R/W}$ valid to $\overline{EM_WE}$ low | | $(WS + 1) * E - 4$ | $(WS + 1) * E + 4$ ns |
| 19 | $t_h(\text{EMWEH-EMRNW})$ | Output hold time, $\overline{EM_WE}$ high to $\overline{EM_R/W}$ invalid | | $(WH + 1) * E - 4$ | $(WH + 1) * E + 4$ ns |
| 20 | $t_{su}(\text{EMBAV-EMWEL})$ | Output setup time, $\overline{EM_BA}[1:0]$ valid to $\overline{EM_WE}$ low | | $(WS + 1) * E - 4$ | $(WS + 1) * E + 4$ ns |
| 21 | $t_h(\text{EMWEH-EMBAIV})$ | Output hold time, $\overline{EM_WE}$ high to $\overline{EM_BA}[1:0]$ invalid | | $(WH + 1) * E - 4$ | $(WH + 1) * E + 4$ ns |
| 22 | $t_{su}(\text{EMAV-EMWEL})$ | Output setup time, $\overline{EM_A}[21:0]$ valid to $\overline{EM_WE}$ low | | $(WS + 1) * E - 4$ | $(WS + 1) * E + 4$ ns |
| 23 | $t_h(\text{EMWEH-EMAV})$ | Output hold time, $\overline{EM_WE}$ high to $\overline{EM_A}[21:0]$ invalid | | $(WH + 1) * E - 4$ | $(WH + 1) * E + 4$ ns |

(1) RS = Read setup, RST = Read STrobe, RH = Read Hold, WS = Write Setup, WST = Write STrobe, WH = Write Hold, TA = Turn Around, EW = Extend Wait mode, SS = Select Strobe mode. These parameters are programmed via the Asynchronous *n* Configuration and Asynchronous Wait Cycle Configuration Registers.
 (2) E = SYSCLK3 period in ns for EMIFA. For example, when running the DSP CPU at 600 MHz, use E = 10 ns.
 (3) When EW = 1, the EMIF will extend the strobe period up to 4,096 additional cycles when the EM_WAIT pin is asserted by the external device.

Table 6-25. Switching Characteristics Over Recommended Operating Conditions for Asynchronous Memory Cycles for EMIFA Module ⁽¹⁾ ⁽²⁾ (see Figure 6-14 and Figure 6-15) (continued)

| NO | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | UNIT |
|----|---|-------------------------------|-----------------------|--------------------|------|
| | | MIN | NOM | MAX | |
| 24 | $t_{w(EMWE)}$ $\overline{EM_WE}$ active low width | | $(WST + 1) * E^{(3)}$ | | ns |
| 25 | $t_{d(EMWAITH-EMWEH)}$ Delay time from EM_WAIT deasserted to EM_WE high | | | 4E + 4 | ns |
| 26 | $t_{su(EMDV-EMWEL)}$ Output setup time, EM_D[15:0] valid to $\overline{EM_WE}$ low | $(WS + 1) * E - 4$ | | $(WS + 1) * E + 4$ | ns |
| 27 | $t_{h(EMWEH-EMDIV)}$ Output hold time, $\overline{EM_WE}$ high to EM_D[15:0] invalid | $(WH + 1) * E - 4$ | | $(WH + 1) * E + 4$ | ns |

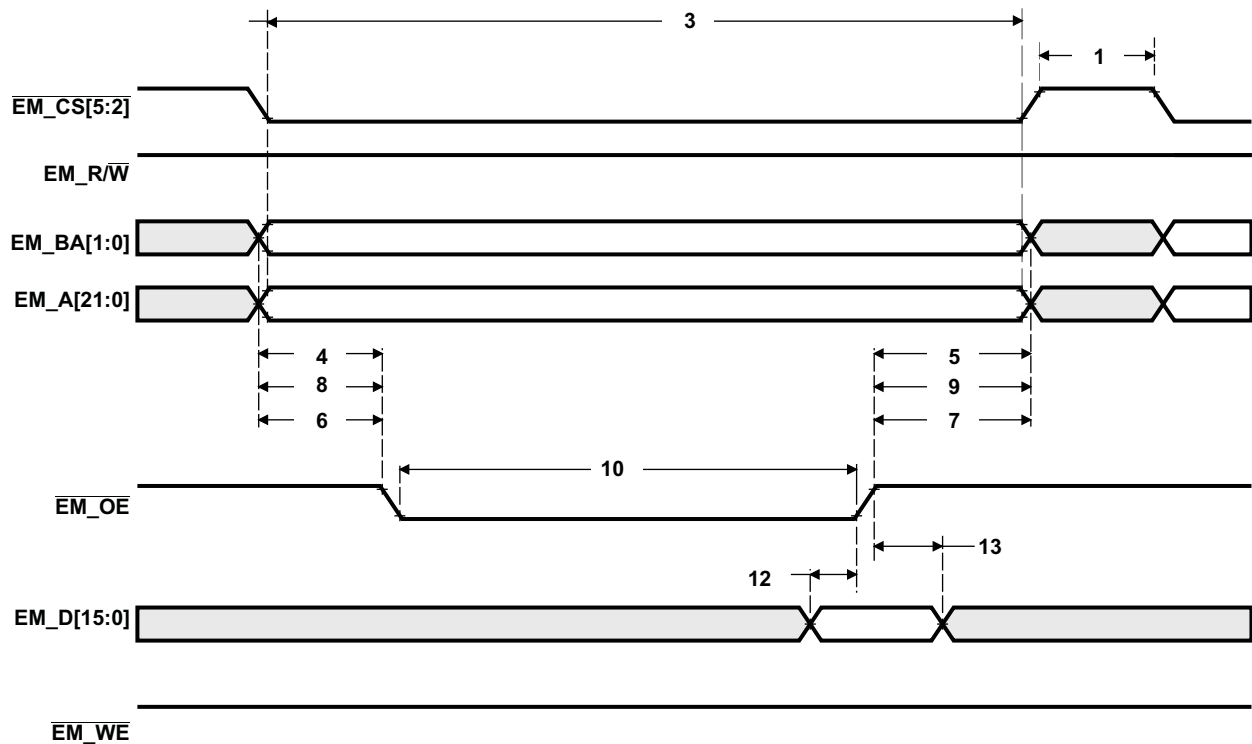


Figure 6-14. Asynchronous Memory Read Timing for EMIFA

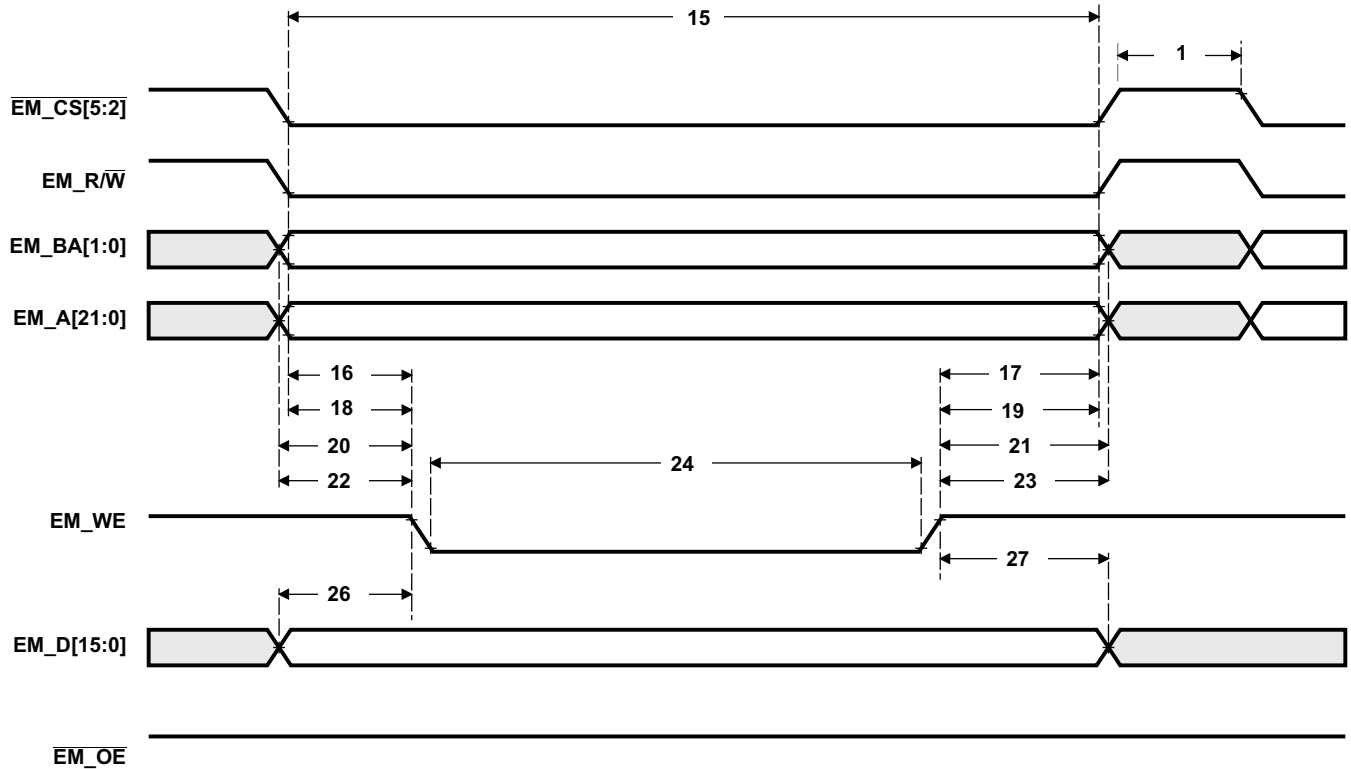


Figure 6-15. Asynchronous Memory Write Timing for EMIF

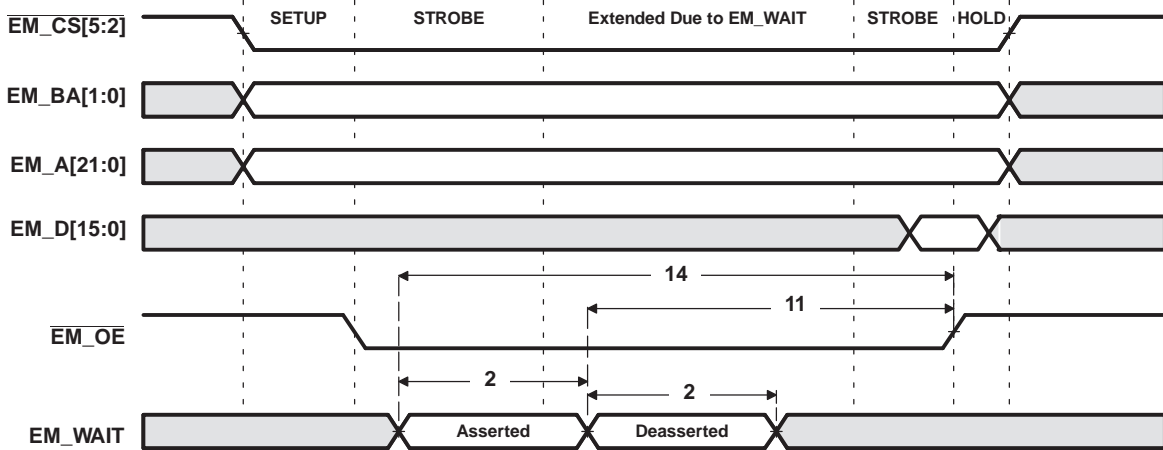


Figure 6-16. EM_WAIT Read Timing Requirements

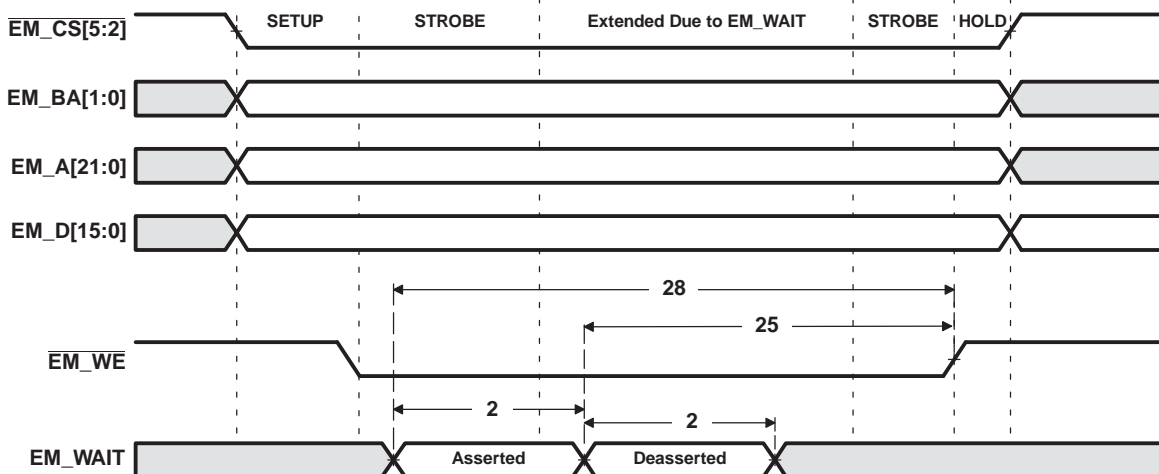


Figure 6-17. EM_WAIT Write Timing Requirements

6.9.4 DDR2 Memory Controller

The DDR2 Memory Controller is a dedicated interface to DDR2 SDRAM. It supports JESD79D-2A standard compliant DDR2 SDRAM Devices and can interface to either 16-bit or 32-bit DDR2 SDRAM devices. For details on the DDR2 Memory Controller, see the *TMS320C642x DSP Peripherals Overview Reference Guide* (literature number [SPRUEM3](#)) and the *TMS320C642x DSP DDR2 Memory Controller User's Guide* (literature number [SPRUEM4](#)).

A memory map of the DDR2 Memory Controller registers is shown in [Table 6-26](#).

Table 6-26. DDR2 Memory Controller Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|----------|--|
| 0x01C4 004C | DDRVTPER | DDR2 VTP Enable Register |
| 0x01C4 2038 | DDRVTPR | DDR2 VTP Register |
| 0x2000 0000 - 0x2000 0003 | - | Reserved |
| 0x2000 0004 | SDRSTAT | SDRAM Status Register |
| 0x2000 0008 | SDBCR | SDRAM Bank Configuration Register |
| 0x2000 000C | SDRCR | SDRAM Refresh Control Register |
| 0x2000 0010 | SDTIMR | SDRAM Timing Register |
| 0x2000 0014 | SDTIMR2 | SDRAM Timing Register 2 |
| 0x2000 0020 | PBBPR | Peripheral Bus Burst Priority Register |
| 0x2000 0024 - 0x2000 00BF | - | Reserved |
| 0x2000 00C0 | IRR | Interrupt Raw Register |
| 0x2000 00C4 | IMR | Interrupt Masked Register |
| 0x2000 00C8 | IMSR | Interrupt Mask Set Register |
| 0x2000 00CC | IMCR | Interrupt Mask Clear Register |
| 0x2000 00D0 - 0x2000 00E3 | - | Reserved |
| 0x2000 00E4 | DDRPHYCR | DDR PHY Control Register |
| 0x2000 00E8 - 0x2000 00EF | - | Reserved |
| 0x2000 00F0 | VTPIOCR | DDR VTP IO Control Register |
| 0x2000 00E8 - 0x2000 7FFF | - | Reserved |

6.9.4.1 DDR2 Memory Controller Electrical Data/Timing

The *Implementing DDR2 PCB Layout on the TMS320C6421/4 DMSoC* Application Report (literature number [SPRAAL7](#)) specifies a complete DDR2 interface solution for the C6424 as well as a list of compatible DDR2 devices. TI has performed the simulation and system characterization to ensure all DDR2 interface timings in this solution are met.

TI only supports board designs that follow the guidelines outlined in the *Implementing DDR2 PCB Layout on the TMS320C6421/4 DMSoC* Application Report (literature number [SPRAAL7](#)).

Table 6-27. Switching Characteristics Over Recommended Operating Conditions for DDR2 Memory Controller⁽¹⁾ ⁽²⁾(see [Figure 6-18](#))

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---------------------------------------|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{c(DDR_CLK)}$ Cycle time, DDR_CLK | 6 | 8 | ns |

(1) DDR_CLK cycle time = 2 x PLL2 _SYSCLK1 cycle time.

(2) The PLL2 Controller **must** be programmed such that the resulting DDR_CLK clock frequency is within the specified range.

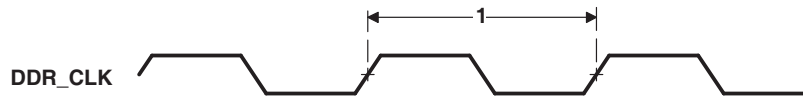


Figure 6-18. DDR2 Memory Controller Clock Timing

6.10 Universal Asynchronous Receiver/Transmitter (UART)

C6424 has 2 UART peripherals. Each 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) on **UART0 only**.

The UART0/1 registers are listed in [Table 6-28](#) and [Table 6-29](#).

6.10.1 UART Peripheral Register Description(s)

Table 6-28. UART0 Register Descriptions

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|-------------|---|
| 0x01C2 0000 | RBR | UART0 Receiver Buffer Register (Read Only) |
| 0x01C2 0000 | THR | UART0 Transmitter Holding Register (Write Only) |
| 0x01C2 0004 | IER | UART0 Interrupt Enable Register |
| 0x01C2 0008 | IIR | UART0 Interrupt Identification Register (Read Only) |
| 0x01C2 0008 | FCR | UART0 FIFO Control Register (Write Only) |
| 0x01C2 000C | LCR | UART0 Line Control Register |
| 0x01C2 0010 | MCR | UART0 Modem Control Register |
| 0x01C2 0014 | LSR | UART0 Line Status Register |
| 0x01C2 0018 | - | Reserved |
| 0x01C2 001C | - | Reserved |
| 0x01C2 0020 | DLL | UART0 Divisor Latch (LSB) |
| 0x01C2 0024 | DLH | UART0 Divisor Latch (MSB) |
| 0x01C2 0028 | PID1 | Peripheral Identification Register 1 |
| 0x01C2 002C | PID2 | Peripheral Identification Register 2 |
| 0x01C2 0030 | PWREMU_MGMT | UART0 Power and Emulation Management Register |
| 0x01C2 0034 - 0x01C2 03FF | - | Reserved |

Table 6-29. UART1 Register Descriptions

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|---------|---|
| 0x01C2 0400 | RBR | UART1 Receiver Buffer Register (Read Only) |
| 0x01C2 0400 | THR | UART1 Transmitter Holding Register (Write Only) |
| 0x01C2 0404 | IER | UART1 Interrupt Enable Register |

Table 6-29. UART1 Register Descriptions (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|-------------|---|
| 0x01C2 0408 | IIR | UART1 Interrupt Identification Register (Read Only) |
| 0x01C2 0408 | FCR | UART1 FIFO Control Register (Write Only) |
| 0x01C2 040C | LCR | UART1 Line Control Register |
| 0x01C2 0410 | MCR | UART1 Modem Control Register |
| 0x01C2 0414 | LSR | UART1 Line Status Register |
| 0x01C2 0418 | - | Reserved |
| 0x01C2 041C | - | Reserved |
| 0x01C2 0420 | DLL | UART1 Divisor Latch (LSB) |
| 0x01C2 0424 | DLH | UART1 Divisor Latch (MSB) |
| 0x01C2 0428 | PID1 | Peripheral Identification Register 1 |
| 0x01C2 042C | PID2 | Peripheral Identification Register 2 |
| 0x01C2 0430 | PWREMU_MGMT | UART1 Power and Emulation Management Register |
| 0x01C2 0434 - 0x01C2 07FF | - | Reserved |

6.10.2 UART Electrical Data/Timing

Table 6-30. Timing Requirements for UARTx Receive⁽¹⁾ (see Figure 6-19)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------|--|-------------------------------|-------|------|
| | | | MIN | MAX | |
| 4 | $t_{w(URXDB)}$ | Pulse duration, UART receive data bit (URXDx) [15/30/100 pF] | 0.96U | 1.05U | ns |
| 5 | $t_{w(URXSB)}$ | Pulse duration, UART receive start bit [15/30/100 pF] | 0.96U | 1.05U | ns |

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

Table 6-31. Switching Characteristics Over Recommended Operating Conditions for UARTx Transmit⁽¹⁾ (see Figure 6-19)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------|--------------------------------|-------|---------|
| | | MIN | MAX | |
| 1 | $f_{(baud)}$ | Maximum programmable baud rate | | 128 kHz |
| 2 | $t_{w(UTXDB)}$ | U - 2 | U + 2 | ns |
| 3 | $t_{w(UTXSB)}$ | U - 2 | U + 2 | ns |

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

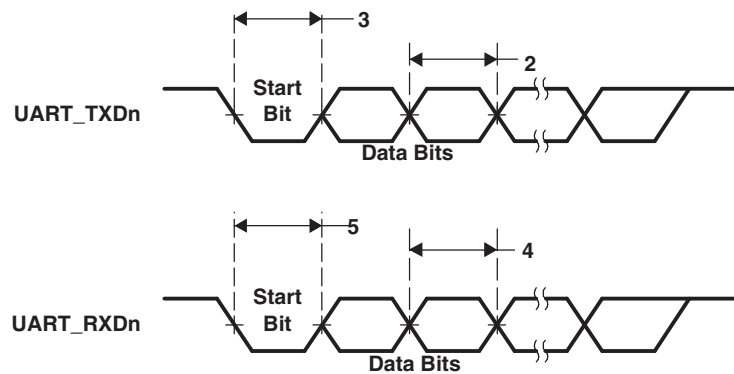


Figure 6-19. UARTx Transmit/Receive Timing

6.11 Inter-Integrated Circuit (I2C)

The inter-integrated circuit (I2C) module provides an interface between C6424 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

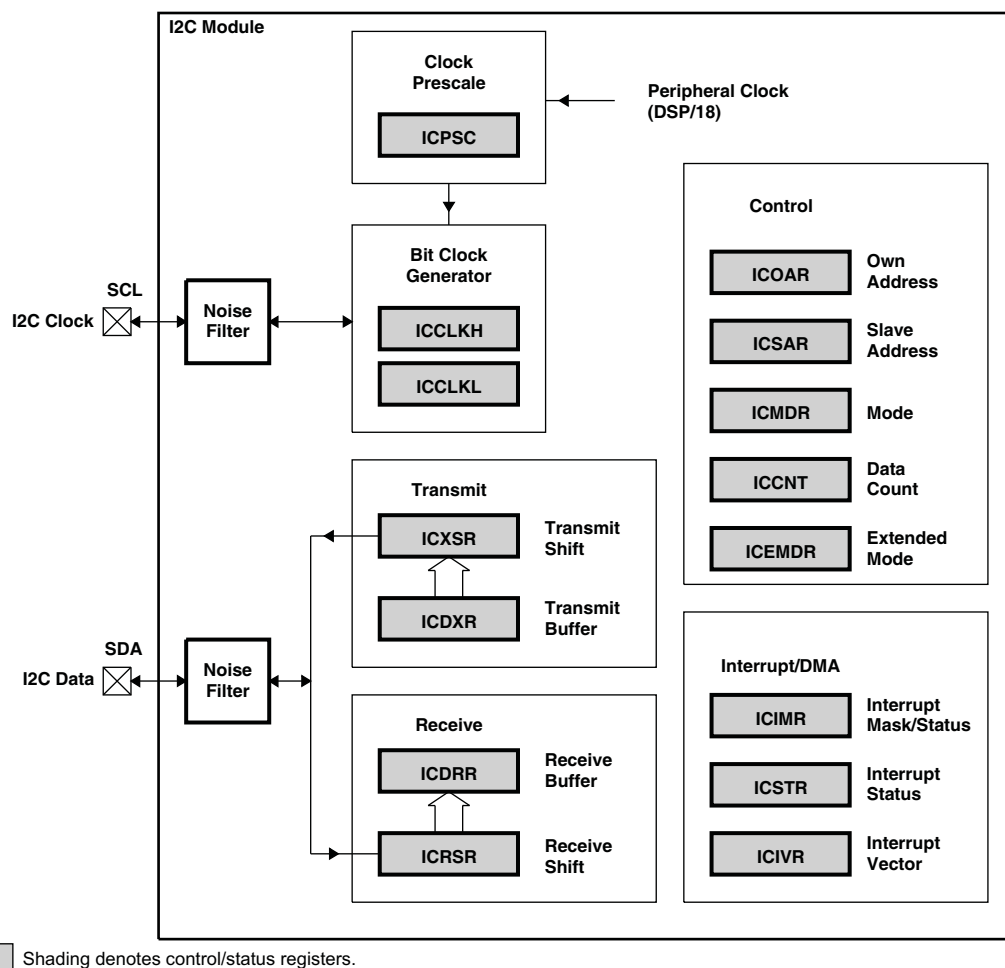


Figure 6-20. I2C Module Block Diagram

For more detailed information on the I2C peripheral, see the *TMS320C642x DSP Peripherals Overview Reference Guide* (literature number SPRUEM3).

6.11.1 I2C Peripheral Register Description(s)

Table 6-32. I2C Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|---------|--|
| 0x1C2 1000 | ICOAR | I2C Own Address Register |
| 0x1C2 1004 | ICIMR | I2C Interrupt Mask Register |
| 0x1C2 1008 | ICSTR | I2C Interrupt Status Register |
| 0x1C2 100C | ICCLKL | I2C Clock Divider Low Register |
| 0x1C2 1010 | ICCLKH | I2C Clock Divider High Register |
| 0x1C2 1014 | ICCNT | I2C Data Count Register |
| 0x1C2 1018 | ICDRR | I2C Data Receive Register |
| 0x1C2 101C | ICSAR | I2C Slave Address Register |
| 0x1C2 1020 | ICDXR | I2C Data Transmit Register |
| 0x1C2 1024 | ICMDR | I2C Mode Register |
| 0x1C2 1028 | ICIVR | I2C Interrupt Vector Register |
| 0x1C2 102C | ICEMDR | I2C Extended Mode Register |
| 0x1C2 1030 | ICPSC | I2C Prescaler Register |
| 0x1C2 1034 | ICPID1 | I2C Peripheral Identification Register 1 |
| 0x1C2 1038 | ICPID2 | I2C Peripheral Identification Register 2 |

6.11.2 I2C Electrical Data/Timing

6.11.2.1 Inter-Integrated Circuits (I2C) Timing

Table 6-33. Timing Requirements for I2C Timings⁽¹⁾ (see Figure 6-21)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|---------------------|---|-------------------------------|------|---------------------|--------------------|---------|
| | | | STANDARD MODE | | FAST MODE | | |
| | | | MIN | MAX | MIN | MAX | |
| 1 | $t_{c(SCL)}$ | Cycle time, SCL | 10 | | 2.5 | | μ s |
| 2 | $t_{su(SCLH-SDAL)}$ | Setup time, SCL high before SDA low (for a repeated START condition) | 4.7 | | 0.6 | | μ s |
| 3 | $t_{h(SCLL-SDAL)}$ | Hold time, SCL low after SDA low (for a START and a repeated START condition) | 4 | | 0.6 | | μ s |
| 4 | $t_{w(SCLL)}$ | Pulse duration, SCL low | 4.7 | | 1.3 | | μ s |
| 5 | $t_{w(SCLH)}$ | Pulse duration, SCL high | 4 | | 0.6 | | μ s |
| 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⁾ | μ s |
| 8 | $t_{w(SDAH)}$ | Pulse duration, SDA high between STOP and START conditions | 4.7 | | 1.3 | | μ s |
| 9 | $t_{r(SDA)}$ | Rise time, SDA | | 1000 | $20 + 0.1C_b^{(5)}$ | 300 | ns |
| 10 | $t_{r(SCL)}$ | Rise time, SCL | | 1000 | $20 + 0.1C_b^{(5)}$ | 300 | ns |
| 11 | $t_{f(SDA)}$ | Fall time, SDA | | 300 | $20 + 0.1C_b^{(5)}$ | 300 | ns |
| 12 | $t_{f(SCL)}$ | Fall time, SCL | | 300 | $20 + 0.1C_b^{(5)}$ | 300 | ns |
| 13 | $t_{su(SCLH-SDAH)}$ | Setup time, SCL high before SDA high (for STOP condition) | 4 | | 0.6 | | μ s |
| 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 automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_r \text{ max} + t_{su(SDA-SCLH)} = 1000 + 250 = 1250$ 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 only to be met 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.

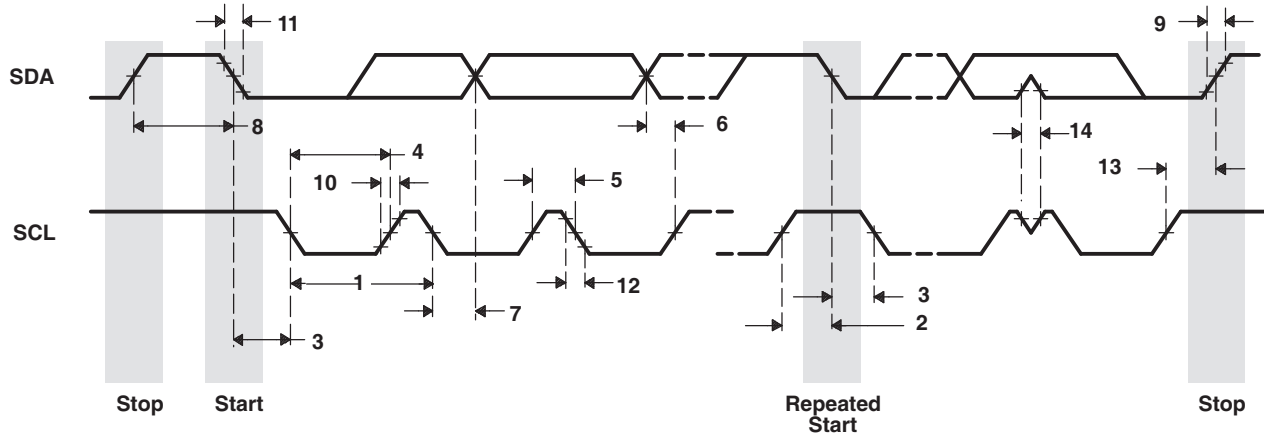


Figure 6-21. I2C Receive Timings

Table 6-34. Switching Characteristics for I2C Timings⁽¹⁾ (see Figure 6-22)

| NO. | PARAMETER | -7I/-6I/-5I/-4 -L/-Q6/-Q5/-Q4 | | | | 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 | $20 + 0.1C_{b(1)}$ | 300 | ns |
| 25 | $t_{r(SCL)}$ Rise time, SCL | | 1000 | $20 + 0.1C_{b(1)}$ | 300 | ns |
| 26 | $t_{f(SDA)}$ Fall time, SDA | | 300 | $20 + 0.1C_{b(1)}$ | 300 | ns |
| 27 | $t_{f(SCL)}$ Fall time, SCL | | 300 | $20 + 0.1C_{b(1)}$ | 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.

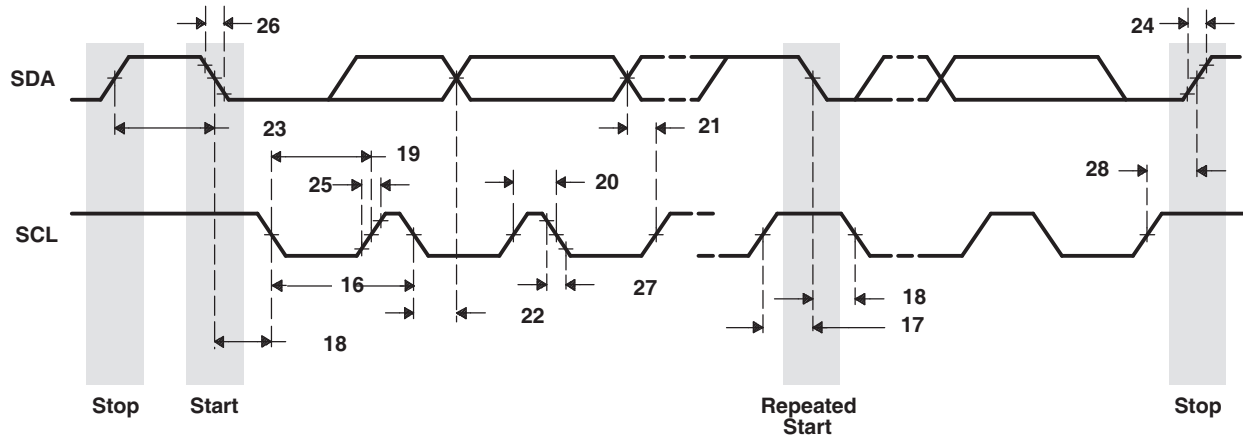


Figure 6-22. I2C Transmit Timings

6.12 Host-Port Interface (HPI) Peripheral

6.12.1 HPI Device-Specific Information

The C6424 device includes a user-configurable 16-bit Host-port interface (HPI16).

Software handshaking via the HRDY bit of the Host Port Control Register (HPIC) is not supported on the C6424.

6.12.2 HPI Peripheral Register Description(s)

Table 6-35. HPI Control Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-----------------------|--------------------------------|---|---|
| 01C6 7800 | PID | Peripheral Identification Register | |
| 01C6 7804 | PWREMU_MGMT | HPI power and emulation management register | The CPU has read/write access to the PWREMU_MGMT register. |
| 01C6 7808 - 01C6 7824 | - | Reserved | |
| 01C6 7828 | - | Reserved | |
| 01C6 782C | - | Reserved | |
| 01C6 7830 | HPIC | HPI control register | The Host and the CPU both have read/write access to the HPIC register. |
| 01C6 7834 | HPIA (HPIAW) ⁽¹⁾ | HPI address register (Write) | The Host has read/write access to the HPIA registers. The CPU has only read access to the HPIA registers. |
| 01C6 7838 | HPIA (HPIAR) ⁽¹⁾ | HPI address register (Read) | |
| 01C6 783C - 01C6 7FFF | - | Reserved | |

- (1) 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 more details about the HPIA registers and their modes, see the *TMS320C642x DSP Host Port Interface (HPI) User's Guide* (literature number [SPRUEM9](#)).

6.12.3 HPI Electrical Data/Timing

Table 6-36. Timing Requirements for Host-Port Interface Cycles⁽¹⁾ ⁽²⁾ (see [Figure 6-23](#) through [Figure 6-24](#))

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{su}(SELV-HSTBL)$ Setup time, select signals ⁽³⁾ valid before $\overline{HSTROBE}$ low | 5 | | ns |
| 2 | $t_h(HSTBL-SELV)$ Hold time, select signals ⁽³⁾ valid after $\overline{HSTROBE}$ low | 2 | | ns |
| 3 | $t_w(HSTBL)$ Pulse duration, $\overline{HSTROBE}$ active low | 15 | | ns |
| 4 | $t_w(HSTBH)$ Pulse duration, $\overline{HSTROBE}$ inactive high between consecutive accesses | 2M | | ns |
| 11 | $t_{su}(HDV-HSTBH)$ Setup time, host data valid before $\overline{HSTROBE}$ high | 5 | | ns |
| 12 | $t_h(HSTBH-HDV)$ Hold time, host data valid after $\overline{HSTROBE}$ high | 0 | | ns |
| 13 | $t_h(HRDYL-HSTBL)$ Hold time, $\overline{HSTROBE}$ high after \overline{HRDY} low. $\overline{HSTROBE}$ should not be inactivated until \overline{HRDY} is active (low); otherwise, HPI writes will not complete properly. | 0 | | ns |

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

(2) $M = \text{SYSCLK3 period} = (\text{CPU clock frequency})/6$ in ns. For example, when running parts at 600 MHz, use $M = 10$ ns.

(3) Select signals include: $\overline{HCNTL}[1:0]$, $\overline{HR}/\overline{W}$ and \overline{HHWIL} .

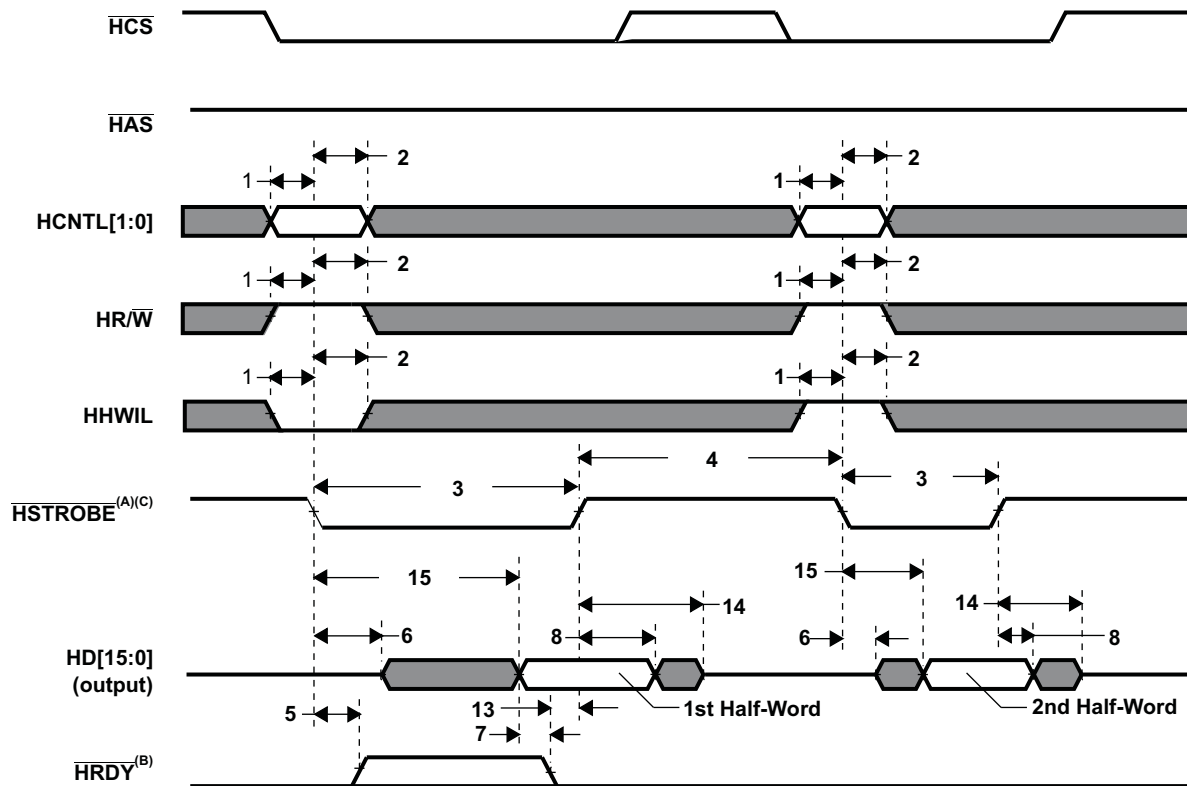
Table 6-37. Switching Characteristics for Host-Port Interface Cycles^{(1) (2) (3)}
 (see [Figure 6-23](#) through [Figure 6-24](#))

| NO. | PARAMETER | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT | |
|-----|----------------------|--|---|-----|------|----|
| | | | MIN | MAX | | |
| 5 | $t_{d(HSTBL-HRDYV)}$ | Delay time, $\overline{HSTROBE}$ low to \overline{HRDY} valid | For HPI Write, \overline{HRDY} can go high (<i>not ready</i>) for these HPI Write conditions; otherwise, \overline{HRDY} stays low (<i>ready</i>): Case 1: Back-to-back HPIA writes (can be either first or second half-word) Case 2: HPIA write following a PREFETCH command (can be either first or second half-word) Case 3: HPID write when FIFO is full or flushing (can be either first or second half-word) Case 4: HPIA write and Write FIFO not empty For HPI Read, \overline{HRDY} can go high (<i>not ready</i>) for these HPI Read conditions: Case 1: HPID read (with auto-increment) and data not in Read FIFO (can only happen to first half-word of HPID access) Case 2: First half-word access of HPID Read without auto-increment For HPI Read, \overline{HRDY} stays low (<i>ready</i>) for these HPI Read conditions: Case 1: HPID read with auto-increment and data is already in Read FIFO (applies to either half-word of HPID access) Case 2: HPID read without auto-increment and data is already in Read FIFO (always applies to second half-word of HPID access) Case 3: HPIC or HPIA read (applies to either half-word access) | | 12 | ns |
| 6 | $t_{en(HSTBL-HD)}$ | Enable time, HD driven from $\overline{HSTROBE}$ low | 2 | | ns | |
| 7 | $t_{d(HRDYL-HDV)}$ | Delay time, \overline{HRDY} low to HD valid | 0 | | ns | |
| 8 | $t_{oh(HSTBH-HDV)}$ | Output hold time, HD valid after $\overline{HSTROBE}$ high | 1.5 | | ns | |
| 14 | $t_{dis(HSTBH-HDV)}$ | Disable time, HD high-impedance from $\overline{HSTROBE}$ high | 12 | | ns | |
| 15 | $t_{d(HSTBL-HDV)}$ | Delay time, $\overline{HSTROBE}$ low to HD valid | For HPI Read. Applies to conditions where data is already residing in HPID/FIFO: Case 1: HPIC or HPIA read Case 2: First half-word of HPID read with auto-increment and data is already in Read FIFO Case 3: Second half-word of HPID read with or without auto-increment | | 15 | ns |
| 18 | $t_{d(HSTBH-HRDYV)}$ | Delay time, $\overline{HSTROBE}$ high to \overline{HRDY} valid | For HPI Write, \overline{HRDY} can go high (<i>not ready</i>) for these HPI Write conditions; otherwise, \overline{HRDY} stays low (<i>ready</i>): Case 1: HPID write when Write FIFO is full (can happen to either half-word) Case 2: HPIA write (can happen to either half-word) Case 3: HPID write without auto-increment (only happens to second half-word) | | 12 | ns |

(1) $M = \text{SYSCLK3 period} = (\text{CPU clock frequency})/6$ in ns. For example, when running parts at 600 MHz, use $M = 10$ ns.

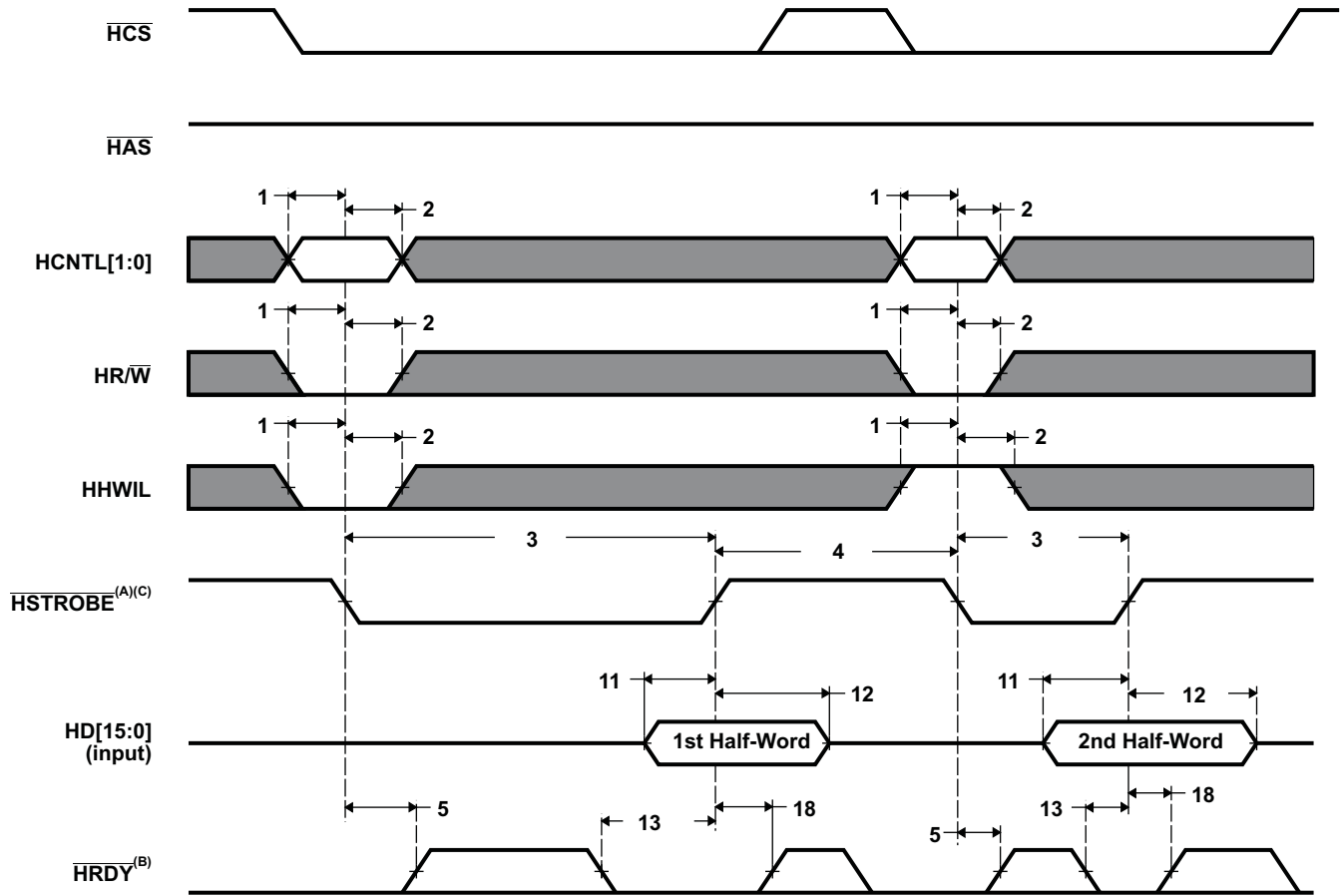
(2) $\overline{HSTROBE}$ refers to the following logical operation on \overline{HCS} , $\overline{HDS1}$, and $\overline{HDS2}$: $[\text{NOT}(\overline{HDS1} \text{ XOR } \overline{HDS2})] \text{ OR } \overline{HCS}$.

(3) By design, whenever \overline{HCS} is driven inactive (high), HPI will drive \overline{HRDY} active low.



- A. $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: $[\text{NOT}(\overline{\text{HDS1}} \text{ XOR } \overline{\text{HDS2}})] \text{ OR } \overline{\text{HCS}}$.
- 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 $\overline{\text{HRDY}}$ may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C642x Host Port Interface (HPI) User's Guide* (literature number SPRUEM9).
- C. $\overline{\text{HCS}}$ reflects typical $\overline{\text{HCS}}$ behavior when $\overline{\text{HSTROBE}}$ assertion is caused by $\overline{\text{HDS1}}$ or $\overline{\text{HDS2}}$. $\overline{\text{HCS}}$ timing requirements are reflected by parameters for $\overline{\text{HSTROBE}}$.

Figure 6-23. HPI16 Read Timing ($\overline{\text{HAS}}$ Not Used, Tied High)



- A. $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: $[\text{NOT}(\overline{\text{HDS1}} \text{ XOR } \overline{\text{HDS2}})] \text{ OR } \overline{\text{HCS}}$.
- B. Depending on the type of write or read operation (HPID without auto-incrementing; HPID with auto-incrementing) and the state of the FIFO, transitions on $\overline{\text{HRDY}}$ may or may not occur. For more detailed information on the HPI peripheral, see the *TMS320C642x Host Port Interface (HPI) User's Guide* (literature number SPRUEM9).
- C. $\overline{\text{HCS}}$ reflects typical $\overline{\text{HCS}}$ behavior when $\overline{\text{HSTROBE}}$ assertion is caused by $\overline{\text{HDS1}}$ or $\overline{\text{HDS2}}$. $\overline{\text{HCS}}$ timing requirements are reflected by parameters for $\overline{\text{HSTROBE}}$.

Figure 6-24. HPI16 Write Timing ($\overline{\text{HAS}}$ Not Used, Tied High)

6.13 Multichannel Buffered Serial Port (McBSP)

The McBSP provides these functions:

- Full-duplex communication
- Double-buffered data registers, which allow a continuous data stream
- Independent framing and clocking for receive and transmit
- Direct interface to industry-standard codecs, analog interface chips (AICs), and other serially connected analog-to-digital (A/D) and digital-to-analog (D/A) devices
- External shift clock or an internal, programmable frequency shift clock for data transfer

If internal clock source is used, the CLKGDV field of the Sample Rate Generator Register (SRGR) must always be set to a value of 1 or greater.

For more detailed information on the McBSP peripheral, see the *TMS320C642x DSP Multichannel Buffered Serial Port (McBSP) User's Guide* (literature number SPRUEN2).

6.13.1 McBSP Peripheral Register Description(s)

Table 6-38. McBSP 0 Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-----------------------|---------|--|--|
| 01D0 0000 | DRR0 | McBSP0 Data Receive Register | The CPU and EDMA3 controller can only read this register; they cannot write to it. |
| 01D0 0004 | DXR0 | McBSP0 Data Transmit Register | |
| 01D0 0008 | SPCR0 | McBSP0 Serial Port Control Register | |
| 01D0 000C | RCR0 | McBSP0 Receive Control Register | |
| 01D0 0010 | XCR0 | McBSP0 Transmit Control Register | |
| 01D0 0014 | SRGR0 | McBSP0 Sample Rate Generator register | |
| 01D0 0018 | MCR0 | McBSP0 Multichannel Control Register | |
| 01D0 001C | RCERE00 | McBSP0 Enhanced Receive Channel Enable Register 0 Partition A/B | |
| 01D0 0020 | XCERE00 | McBSP0 Enhanced Transmit Channel Enable Register 0 Partition A/B | |
| 01D0 0024 | PCR0 | McBSP0 Pin Control Register | |
| 01D0 0028 | RCERE10 | McBSP0 Enhanced Receive Channel Enable Register 1 Partition C/D | |
| 01D0 002C | XCERE10 | McBSP0 Enhanced Transmit Channel Enable Register 1 Partition C/D | |
| 01D0 0030 | RCERE20 | McBSP0 Enhanced Receive Channel Enable Register 2 Partition E/F | |
| 01D0 0034 | XCERE20 | McBSP0 Enhanced Transmit Channel Enable Register 2 Partition E/F | |
| 01D0 0038 | RCERE30 | McBSP0 Enhanced Receive Channel Enable Register 3 Partition G/H | |
| 01D0003C | XCERE30 | McBSP0 Enhanced Transmit Channel Enable Register 3 Partition G/H | |
| 01D0 0040 - 01D0 07FF | - | Reserved | |

Table 6-39. McBSP 1 Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-------------------|---------|------------------------------|---|
| 01D0 0800 | DRR1 | McBSP1 Data Receive Register | The CPU and EDMA controller can only read this register; they cannot write to it. |

Table 6-39. McBSP 1 Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-----------------------|---------|---|----------|
| 01D0 0804 | DXR1 | McBSP1 Data Transmit Register | |
| 01D0 0808 | SPCR1 | McBSP1 serial port control register | |
| 01D0 080C | RCR1 | McBSP1 Receive Control Register | |
| 01D0 0810 | XCR1 | McBSP1 Transmit Control Register | |
| 01D0 0814 | SRGR1 | McBSP1 sample rate generator register | |
| 01D0 0818 | MCR1 | McBSP1 multichannel control register | |
| 01D0 081C | RCERE01 | McBSP1 Enhanced Receive Channel Enable Register 0 Partition A/B | |
| 01D0 0820 | XCERE01 | McBSP1 Enhanced Transmit Channel Enable Register 0 Partition A/B | |
| 01D0 0824 | PCR1 | McBSP1 Pin Control Register | |
| 01D0 0828 | RCERE11 | McBSP1 Enhanced Receive Channel Enable Register 1 Partition C/D | |
| 01D0 082C | XCERE11 | McBSP1 Enhanced Transmit Channel Enable Register 1 Partition C/D | |
| 01D0 0830 | RCERE21 | McBSP1 Enhanced Receive Channel Enable Register 2 Partition E/F | |
| 01D0 0834 | XCERE21 | McBSP1 Enhanced Transmit Channel Enable Register 2 Partition E/F | |
| 01D0 0838 | RCERE31 | McBSP1 Enhanced Receive Channel Enable Register 3 Partition G/H | |
| 01D0 083C | XCERE31 | McBSP1 Enhanced Transmit Channel Enable Register 3 Partition G/H | |
| 01D0 0840 - 01D0 0FFF | - | Reserved | |

6.13.2 McBSP Electrical Data/Timing

6.13.2.1 Multichannel Buffered Serial Port (McBSP) Timing

Table 6-40. Timing Requirements for McBSP⁽¹⁾ (see Figure 6-25)

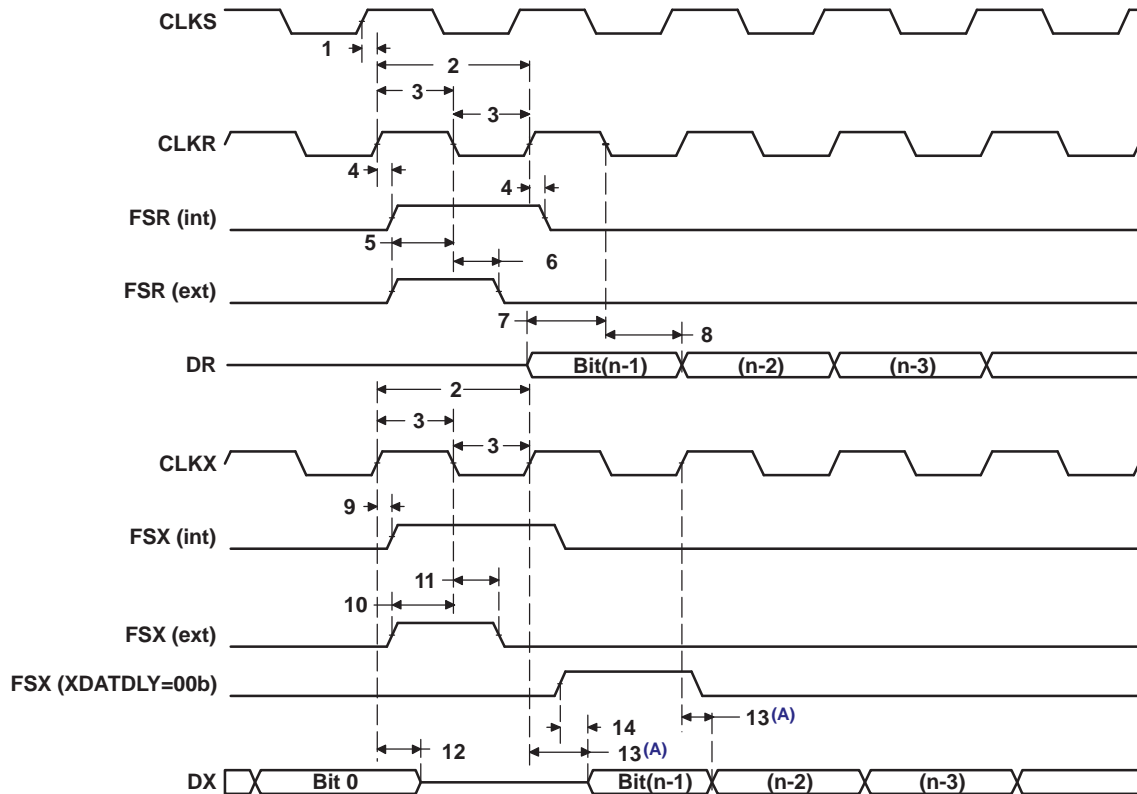
| NO. | | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------------|---|------------|-------------------------------|-----|------|
| | | | | MIN | MAX | |
| 2 | $t_{c(CKRX)}$ | Cycle time, CLKR/X | CLKR/X ext | 2P ⁽²⁾ (3) | | ns |
| 3 | $t_{w(CKRX)}$ | Pulse duration, CLKR/X high or CLKR/X low | CLKR/X ext | P - 1 ⁽⁴⁾ | | ns |
| 5 | $t_{su(FRH-CKRL)}$ | Setup time, external FSR high before CLKR low | CLKR int | 14 | | ns |
| | | | CLKR ext | 4 | | |
| 6 | $t_h(CKRL-FRH)$ | Hold time, external FSR high after CLKR low | CLKR int | 6 | | ns |
| | | | CLKR ext | 4 | | |
| 7 | $t_{su(DRV-CKRL)}$ | Setup time, DR valid before CLKR low | CLKR int | 14 | | ns |
| | | | CLKR ext | 4 | | |
| 8 | $t_h(CKRL-DRV)$ | Hold time, DR valid after CLKR low | CLKR int | 3.5 | | ns |
| | | | CLKR ext | 3 | | |
| 10 | $t_{su(FXH-CKXL)}$ | Setup time, external FSX high before CLKX low | CLKX int | 14 | | ns |
| | | | CLKX ext | 4 | | |
| 11 | $t_h(CKXL-FXH)$ | Hold time, external FSX high after CLKX low | CLKX int | 6 | | ns |
| | | | CLKX ext | 3 | | |

- (1) CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
- (2) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (3) Use whichever value is greater. Minimum CLKR/X cycle times *must* be met, even when CLKR/X is generated by an internal clock source. The minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.
- (4) This parameter applies to the maximum McBSP frequency. Operate serial clocks (CLKR/X) in the reasonable range of 40/60 duty cycle.

Table 6-41. Switching Characteristics Over Recommended Operating Conditions for McBSP⁽¹⁾ (2)
(see [Figure 6-25](#))

| NO. | PARAMETER | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------------|--|--|-----|------|
| | | | MIN | MAX | |
| 1 | $t_{d(CKSH-CKRXH)}$ | Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input | 3 | 10 | ns |
| 2 | $t_{c(CKRX)}$ | Cycle time, CLKR/X | CLKR/X int $2P^{(3)} \text{ }^{(4)} \text{ }^{(5)}$ | | ns |
| 3 | $t_w(CKRX)$ | Pulse duration, CLKR/X high or CLKR/X low | CLKR/X int $C - 2^{(6)} \quad C + 2^{(6)}$ | | ns |
| 4 | $t_{d(CKRH-FRV)}$ | Delay time, CLKR high to internal FSR valid | CLKR int -4 5.5 | | ns |
| 9 | $t_{d(CKXH-FXV)}$ | Delay time, CLKX high to internal FSX valid | CLKX int -4 5.5 | | ns |
| | | | CLKX ext 2.5 14.5 | | |
| 12 | $t_{dis(CKXH-DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX high | CLKX int -5.5 7.5 | | ns |
| | | | CLKX ext -2.1 16 | | |
| 13 | $t_{d(CKXH-DXV)}$ | Delay time, CLKX high to DX valid | CLKX int $-4 + D1^{(7)} \quad 5.5 + D2^{(7)}$ | | ns |
| | | | CLKX ext $2.5 + D1^{(7)} \quad 14.5 + D2^{(7)}$ | | |
| 14 | $t_{d(FXH-DXV)}$ | Delay time, FSX high to DX valid ONLY applies when in data delay 0 (XDATDLY = 00b) mode | FSX int -4 ⁽⁸⁾ 5 ⁽⁸⁾ | | ns |
| | | | FSX ext 1 ⁽⁸⁾ 14.5 ⁽⁸⁾ | | |

- (1) CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
- (2) Minimum delay times also represent minimum output hold times.
- (3) Minimum CLKR/X cycle times must be met, even when CLKR/X is generated by an internal clock source. Minimum CLKR/X cycle times are based on internal logic speed; the maximum usable speed may be lower due to EDMA limitations and AC timing requirements.
- (4) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (5) Use whichever value is greater.
- (6) C = H or L
 S = sample rate generator input clock = P if CLKSM = 1 (P = SYSCLK3 period)
 S = sample rate generator input clock = P_clks if CLKSM = 0 (P_clks = CLKS period)
 H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
 H = (CLKGDV + 1)/2 * S if CLKGDV is odd
 L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
 L = (CLKGDV + 1)/2 * S if CLKGDV is odd
 CLKGDV should be set appropriately to ensure the McBSP bit rate does not exceed the maximum limit (see (4) above).
- (7) Extra delay from CLKX high to DX valid **applies only to the first data bit of a device**, if and only if DXENA = 1 in SPCR.
 if DXENA = 0, then D1 = D2 = 0
 if DXENA = 1, then D1 = 6P, D2 = 12P
- (8) Extra delay from FSX high to DX valid **applies only to the first data bit of a device**, if and only if DXENA = 1 in SPCR.
 if DXENA = 0, then D1 = D2 = 0
 if DXENA = 1, then D1 = 6P, D2 = 12P



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

Figure 6-25. McBSP Timing^(B)

Table 6-42. Timing Requirements for FSR When GSYNC = 1 (see Figure 6-26)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------------|---------------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{su}(FRH-CKSH)$ | Setup time, FSR high before CLKS high | | ns |
| 2 | $t_h(CKSH-FRH)$ | Hold time, FSR high after CLKS high | | ns |

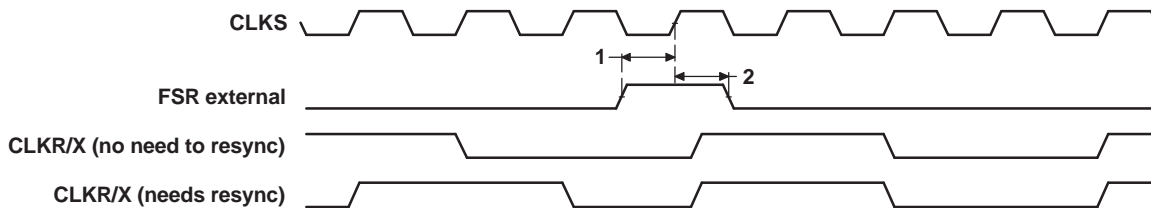


Figure 6-26. FSR Timing When GSYNC = 1

Table 6-43. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0⁽¹⁾ (2) (see Figure 6-27)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|--------------------|--------------------------------------|-----|-------|--------|------|
| | | MASTER | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su}(DRV-CKXL)$ | Setup time, DR valid before CLKX low | | 14 | 2 - 3P | ns |

(1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.

(2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.

Table 6-43. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0 ^{(1) (2)}
(see Figure 6-27) (continued)

| NO. | | -7I-6I-5I-4 -LI-Q6I-Q5I-Q4 | | | | UNIT |
|-----|---------------------------|------------------------------------|-----|-------|--------|------|
| | | MASTER | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 5 | t _h (CKXL-DRV) | Hold time, DR valid after CLKX low | | 4 | 5 + 6P | ns |

Table 6-44. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0 ^{(1) (2)} (see Figure 6-27)

| NO. | PARAMETER | -7I-6I-5I-4 -LI-Q6I-Q5I-Q4 | | | | UNIT |
|-----|------------------------------|---|-----|-------|---------------------|---------------------|
| | | MASTER ⁽³⁾ | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 1 | t _h (CKXL-FXL) | Hold time, FSX low after CLKX low ⁽⁴⁾ | | T - 4 | T + 5.5 | ns |
| 2 | t _d (FXL-CKXH) | Delay time, FSX low to CLKX high ⁽⁵⁾ | | L - 4 | L + 4 | ns |
| 3 | t _d (CKXH-DXV) | Delay time, CLKX high to DX valid | | -4 | 5.5 | 3P + 2.8 5P + 17 |
| 6 | t _{dis} (CKXL-DXHZ) | Disable time, DX high impedance following last data bit from CLKX low | | L - 6 | L + 7.5 | ns |
| 7 | t _{dis} (FXH-DXHZ) | Disable time, DX high impedance following last data bit from FSX high | | | P + 3 3P + 17 | ns |
| 8 | t _d (FXL-DXV) | Delay time, FSX low to DX valid | | | 2P + 1.8 4P + 17 | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.
- (3) S = Sample rate generator input clock = 2P if CLKSM = 1 (P = SYSCLK3 period)
S = Sample rate generator input clock = 2P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
H = (CLKGDV + 1)/2 * S if CLKGDV is odd
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
L = (CLKGDV + 1)/2 * S if CLKGDV is odd
- (4) FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
- (5) FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

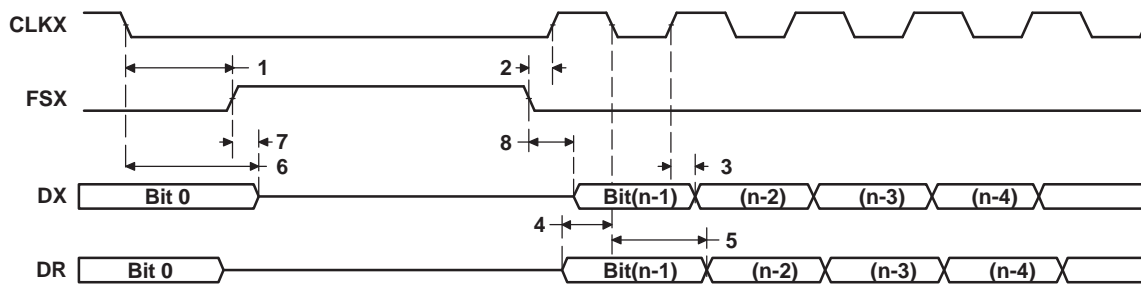


Figure 6-27. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

Table 6-45. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0⁽¹⁾ ⁽²⁾
(see Figure 6-28)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT | |
|-----|--------------------|---------------------------------------|-----|-------|-----|--------|----|
| | | MASTER | | SLAVE | | | |
| | | MIN | MAX | MIN | MAX | | |
| 4 | $t_{su}(DRV-CKXH)$ | Setup time, DR valid before CLKX high | | 14 | | 2 - 3P | ns |
| 5 | $t_h(CKXH-DRV)$ | Hold time, DR valid after CLKX high | | 4 | | 5 + 6P | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.

Table 6-46. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0⁽¹⁾ ⁽²⁾ (see Figure 6-28)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT | | |
|-----|----------------------|---|-----|-------|---------|----------|---------|----|
| | | MASTER ⁽³⁾ | | SLAVE | | | | |
| | | MIN | MAX | MIN | MAX | | | |
| 1 | $t_h(CKXL-FXL)$ | Hold time, FSX low after CLKX low ⁽⁴⁾ | | L - 4 | L + 5.5 | | ns | |
| 2 | $t_d(FXL-CKXH)$ | Delay time, FSX low to CLKX high ⁽⁵⁾ | | T - 4 | T + 4 | | ns | |
| 3 | $t_d(CKXL-DXV)$ | Delay time, CLKX low to DX valid | | -4 | 5.5 | 3P + 2.8 | 5P + 17 | ns |
| 6 | $t_{dis}(CKXL-DXHZ)$ | Disable time, DX high impedance following last data bit from CLKX low | | -6 | 7.5 | 3P + 2 | 5P + 17 | ns |
| 7 | $t_d(FXL-DXV)$ | Delay time, FSX low to DX valid | | H - 4 | H + 5.5 | 2P + 2 | 4P + 17 | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.
- (3) S = Sample rate generator input clock = 2P if CLKSM = 1 (P = SYSCLK3 period)
S = Sample rate generator input clock = 2P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
H = (CLKGDV + 1)/2 * S if CLKGDV is odd
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
L = (CLKGDV + 1)/2 * S if CLKGDV is odd
- (4) FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
- (5) FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

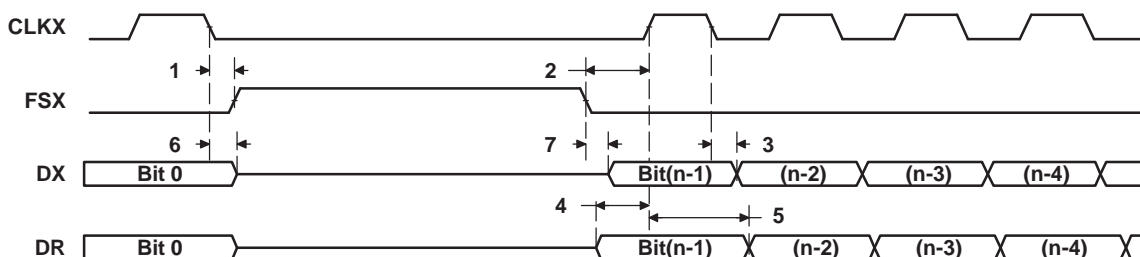


Figure 6-28. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0

Table 6-47. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1⁽¹⁾ ⁽²⁾
(see Figure 6-29)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|--------------------|---------------------------------------|-----|-------|--------|------|
| | | MASTER | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su(DRV-CKXH)}$ | Setup time, DR valid before CLKX high | | 14 | 2 - 3P | ns |
| 5 | $t_h(CKXH-DRV)$ | Hold time, DR valid after CLKX high | | 4 | 5 + 6P | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.

Table 6-48. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1⁽¹⁾ ⁽²⁾ (see Figure 6-29)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|----------------------|--|-----|-------|-------------------|---------------------|
| | | MASTER ⁽³⁾ | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 1 | $t_h(CKXH-FXL)$ | Hold time, FSX low after CLKX high ⁽⁴⁾ | | T - 4 | T + 5.5 | ns |
| 2 | $t_d(FXL-CKXL)$ | Delay time, FSX low to CLKX low ⁽⁵⁾ | | H - 4 | H + 4 | ns |
| 3 | $t_d(CKXL-DXV)$ | Delay time, CLKX low to DX valid | | -4 | 5.5 | 3P + 2.8 5P + 17 |
| 6 | $t_{dis}(CKXH-DXHZ)$ | Disable time, DX high impedance following last data bit from CLKX high | | H - 6 | H + 7.5 | ns |
| 7 | $t_{dis}(FXH-DXHZ)$ | Disable time, DX high impedance following last data bit from FSX high | | | P + 3 3P + 17 | ns |
| 8 | $t_d(FXL-DXV)$ | Delay time, FSX low to DX valid | | | 2P + 2 4P + 17 | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.
- (3) S = Sample rate generator input clock = 2P if CLKSM = 1 (P = SYSCLK3 period)
S = Sample rate generator input clock = 2P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
H = (CLKGDV + 1)/2 * S if CLKGDV is odd
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
L = (CLKGDV + 1)/2 * S if CLKGDV is odd
- (4) FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
- (5) FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

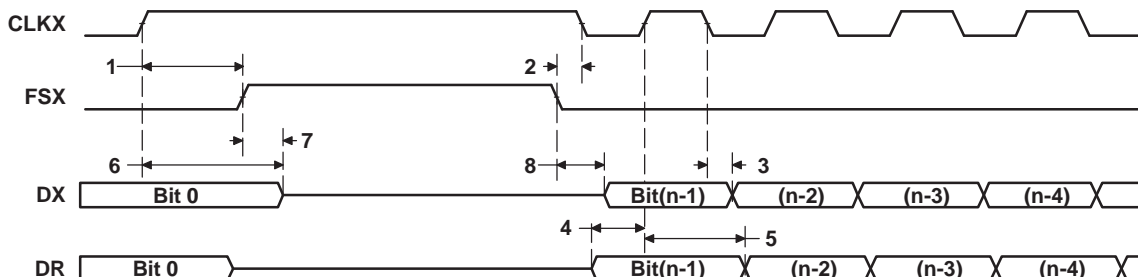


Figure 6-29. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

Table 6-49. Timing Requirements for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1⁽¹⁾ (2)
(see Figure 6-30)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|--------------------|---------------------------------------|-----|-------|--------|------|
| | | MASTER | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 4 | $t_{su}(DRV-CKXH)$ | Setup time, DR valid before CLKX high | | 14 | 2 - 3P | ns |
| 5 | $t_h(CKXH-DRV)$ | Hold time, DR valid after CLKX high | | 4 | 5+ 6P | ns |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.

Table 6-50. Switching Characteristics Over Recommended Operating Conditions for McBSP as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1⁽¹⁾ (2) (see Figure 6-30)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|----------------------|--|-----|-------|---------|---------------------|
| | | MASTER ⁽³⁾ | | SLAVE | | |
| | | MIN | MAX | MIN | MAX | |
| 1 | $t_h(CKXH-FXL)$ | Hold time, FSX low after CLKX high ⁽⁴⁾ | | H - 4 | H + 5.5 | ns |
| 2 | $t_d(FXL-CKXL)$ | Delay time, FSX low to CLKX low ⁽⁵⁾ | | T - 4 | T + 4 | ns |
| 3 | $t_d(CKXH-DXV)$ | Delay time, CLKX high to DX valid | | -4 | 5.5 | 3P + 2.8 5P + 17 |
| 6 | $t_{dis}(CKXH-DXHZ)$ | Disable time, DX high impedance following last data bit from CLKX high | | -6 | 7.5 | 3P + 2 5P + 17 |
| 7 | $t_d(FXL-DXV)$ | Delay time, FSX low to DX valid | | L - 4 | L + 5.5 | 2P + 2 4P + 17 |

- (1) P = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use P = 10ns.
- (2) For all SPI Slave modes, the rate of the internal clock CLKG must be at least 8 times faster than that of the SPI data rate. User should program sample rate generator to achieve maximum CLKG by setting CLKSM = CLKGDV = 1.
- (3) S = Sample rate generator input clock = 2P if CLKSM = 1 (P = SYSCLK3 period)
S = Sample rate generator input clock = 2P_clks if CLKSM = 0 (P_clks = CLKS period)
T = CLKX period = (1 + CLKGDV) * S
H = CLKX high pulse width = (CLKGDV/2 + 1) * S if CLKGDV is even
H = (CLKGDV + 1)/2 * S if CLKGDV is odd
L = CLKX low pulse width = (CLKGDV/2) * S if CLKGDV is even
L = (CLKGDV + 1)/2 * S if CLKGDV is odd
- (4) FSRP = FSXP = 1. As a SPI Master, FSX is inverted to provide active-low slave-enable output. As a Slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for Master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for Slave McBSP
- (5) FSX should be low before the rising edge of clock to enable Slave devices and then begin a SPI transfer at the rising edge of the Master clock (CLKX).

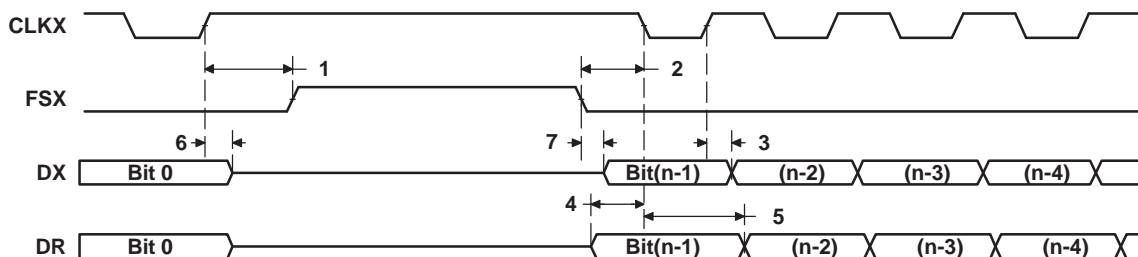


Figure 6-30. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

6.14 Multichannel Audio Serial Port (McASP0) Peripheral

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).

6.14.1 McASP0 Device-Specific Information

The C6424 device includes one multichannel audio serial port (McASP) interface peripheral (McASP0). The McASP0 is a serial port optimized for the needs of multichannel audio applications.

The McASP0 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 McASP0 peripheral, see the *TMS320C642x DSP Multichannel Audio Serial Port (McASP) User's Guide* (literature number SPRUEN1).

6.14.1.1 McASP Block Diagram

[Figure 6-31](#) illustrates the major blocks along with external signals of the C6424 McASP0 peripheral; and shows the 4 serial data [AXR] pins.

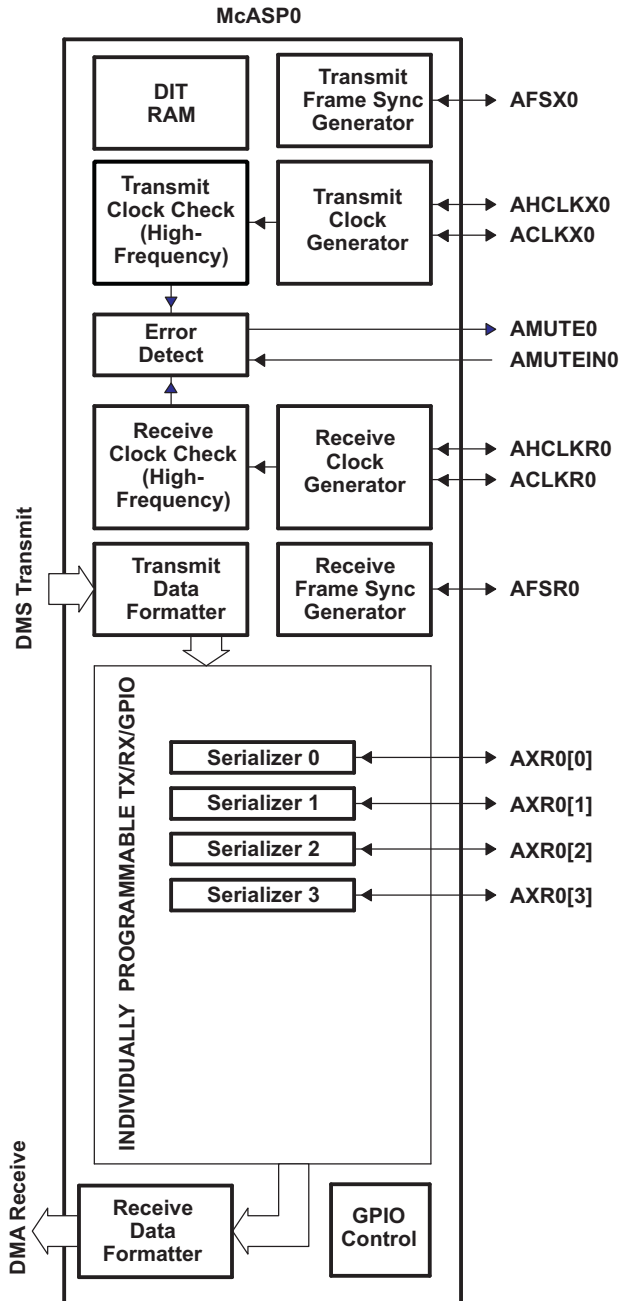


Figure 6-31. McASP0 Configuration

6.14.1.2 McASP0 Peripheral Register Description(s)

Table 6-51. McASP0 Control Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-----------------------|-----------|---|
| 01D0 1000 | PID | Peripheral Identification register [Register value: 0x0010 0101] |
| 01D0 1004 | – | Reserved |
| 01D0 1008 | – | Reserved |
| 01D0 100C | – | Reserved |
| 01D0 1010 | PFUNC | Pin function register |
| 01D0 1014 | PDIR | Pin direction register |
| 01D0 1018 | – | Reserved |
| 01D0 101C | – | Reserved |
| 01D0 1020 | – | Reserved |
| 01D0 1024 – 01D0 1040 | – | Reserved |
| 01D0 1044 | GBLCTL | Global control register |
| 01D0 1048 | AMUTE | Mute control register |
| 01D0 104C | DLBCTL | Digital Loop-back control register |
| 01D0 1050 | DITCTL | DIT mode control register |
| 01D0 1054 – 01D0 105C | – | Reserved |
| 01D0 1060 | RGBLCTL | Alias of GBLCTL containing only Receiver Reset bits, allows transmit to be reset independently from receive. |
| 01D0 1064 | RMASK | Receiver format UNIT bit mask register |
| 01D0 1068 | RFMT | Receive bit stream format register |
| 01D0 106C | AFSRCTL | Receive frame sync control register |
| 01D0 1070 | ACLKRCTL | Receive clock control register |
| 01D0 1074 | AHCLKRCTL | High-frequency receive clock control register |
| 01D0 1078 | RTDM | Receive TDM slot 0–31 register |
| 01D0 107C | RINTCTL | Receiver interrupt control register |
| 01D0 1080 | RSTAT | Status register – Receiver |
| 01D0 1084 | RSLOT | Current receive TDM slot register |
| 01D0 1088 | RCLKCHK | Receiver clock check control register |
| 01D0 108C – 01D0 109C | – | Reserved |
| 01D0 10A0 | XGBLCTL | Alias of GBLCTL containing only Transmitter Reset bits, allows transmit to be reset independently from receive. |
| 01D0 10A4 | XMASK | Transmit format UNIT bit mask register |
| 01D0 10A8 | XFMT | Transmit bit stream format register |
| 01D0 10AC | AFSXCTL | Transmit frame sync control register |
| 01D0 10B0 | ACLKXCTL | Transmit clock control register |
| 01D0 10B4 | AHCLKXCTL | High-frequency Transmit clock control register |
| 01D0 10B8 | XTDM | Transmit TDM slot 0–31 register |
| 01D0 10BC | XINTCTL | Transmit interrupt control register |
| 01D0 10C0 | XSTAT | Status register – Transmitter |
| 01D0 10C4 | XSLOT | Current transmit TDM slot |
| 01D0 10C8 | XCLKCHK | Transmit clock check control register |

Table 6-51. McASP0 Control Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-----------------------|----------|---|
| 01D0 10CC – 01D0 10FC | – | Reserved |
| 01D0 1100 | DITCSRA0 | Left (even TDM slot) channel status register file |
| 01D0 1104 | DITCSRA1 | Left (even TDM slot) channel status register file |
| 01D0 1108 | DITCSRA2 | Left (even TDM slot) channel status register file |
| 01D0 110C | DITCSRA3 | Left (even TDM slot) channel status register file |
| 01D0 1110 | DITCSRA4 | Left (even TDM slot) channel status register file |
| 01D0 1114 | DITCSRA5 | Left (even TDM slot) channel status register file |
| 01D0 1118 | DITCSRB0 | Right (odd TDM slot) channel status register file |
| 01D0 111C | DITCSRB1 | Right (odd TDM slot) channel status register file |
| 01D0 1120 | DITCSRB2 | Right (odd TDM slot) channel status register file |
| 01D0 1124 | DITCSRB3 | Right (odd TDM slot) channel status register file |
| 01D0 1128 | DITCSRB4 | Right (odd TDM slot) channel status register file |
| 01D0 112C | DITCSRB5 | Right (odd TDM slot) channel status register file |
| 01D0 1130 | DITUDRA0 | Left (even TDM slot) user data register file |
| 01D0 1134 | DITUDRA1 | Left (even TDM slot) user data register file |
| 01D0 1138 | DITUDRA2 | Left (even TDM slot) user data register file |
| 01D0 113C | DITUDRA3 | Left (even TDM slot) user data register file |
| 01D0 1140 | DITUDRA4 | Left (even TDM slot) user data register file |
| 01D0 1144 | DITUDRA5 | Left (even TDM slot) user data register file |
| 01D0 1148 | DITUDRB0 | Right (odd TDM slot) user data register file |
| 01D0 114C | DITUDRB1 | Right (odd TDM slot) user data register file |
| 01D0 1150 | DITUDRB2 | Right (odd TDM slot) user data register file |
| 01D0 1154 | DITUDRB3 | Right (odd TDM slot) user data register file |
| 01D0 1158 | DITUDRB4 | Right (odd TDM slot) user data register file |
| 01D0 115C | DITUDRB5 | Right (odd TDM slot) user data register file |
| 01D0 1160 – 01D0 117C | – | Reserved |
| 01D0 1180 | SRCTL0 | Serializer 0 control register |
| 01D0 1184 | SRCTL1 | Serializer 1 control register |
| 01D0 1188 | SRCTL2 | Serializer 2 control register |
| 01D0 118C | SRCTL3 | Serializer 3 control register |
| 01D0 1190 – 01D0 11FC | – | Reserved |
| 01D0 1200 | XBUF0 | Transmit Buffer for Serializer 0 |
| 01D0 1204 | XBUF1 | Transmit Buffer for Serializer 1 |
| 01D0 1208 | XBUF2 | Transmit Buffer for Serializer 2 |
| 01D0 120C | XBUF3 | Transmit Buffer for Serializer 3 |
| 01D0 1210 – 01D0 127C | – | Reserved |
| 01D0 1280 | RBUF0 | Receive Buffer for Serializer 0 |
| 01D0 1284 | RBUF1 | Receive Buffer for Serializer 1 |
| 01D0 1288 | RBUF2 | Receive Buffer for Serializer 2 |
| 01D0 128C | RBUF3 | Receive Buffer for Serializer 3 |
| 01D0 1290 – 01D0 13FF | – | Reserved |

Table 6-52. McASP0 Data Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-----------------------|------------|--|---|
| 01D0 1400 – 01D0 17FF | RBUF /XBUF | McASP0 receive buffers or McASP0 transmit buffers via the Peripheral Data Bus. | (Used when RBUSEL or XBUSEL bits = 0 [these bits are located in the RFMT or XFMT registers, respectively].) |

6.14.1.3 McASP0 Electrical Data/Timing

6.14.1.3.1 Multichannel Audio Serial Port (McASP) Timing

Table 6-53. Timing Requirements for McASP (see Figure 6-32 and Figure 6-33)⁽¹⁾

| NO. | | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---------------------------|--|--------------------|-------------------------------|-----|------|
| | | | | MIN | MAX | |
| 1 | $t_c(\text{AHCKRX})$ | Cycle time, AHCLKR/X | | | 25 | ns |
| 2 | $t_w(\text{AHCKRX})$ | Pulse duration, AHCLKR/X high or low | | | 10 | ns |
| 3 | $t_c(\text{CKRX})$ | Cycle time, ACLKR/X ⁽²⁾ | ACLKR/X ext | | 25 | ns |
| 4 | $t_w(\text{CKRX})$ | Pulse duration, ACLKR/X high or low | ACLKR/X ext | | 10 | ns |
| 5 | $t_{su}(\text{FRX-CKRX})$ | Setup time, AFSR/X input valid before ACLKR/X latches data | ACLKR/X int | | 11 | ns |
| | | | ACLKR/X ext | | 3 | ns |
| 6 | $t_h(\text{CKRX-FRX})$ | Hold time, AFSR/X input valid after ACLKR/X latches data | ACLKR/X int | | 0 | ns |
| | | | ACLKR/X ext input | | 4 | ns |
| | | | ACLKR/X ext output | | 6 | ns |
| 7 | $t_{su}(\text{AXR-CKRX})$ | Setup time, AXR input valid before ACLKR/X latches data | ACLKR/X int | | 11 | ns |
| | | | ACLKR/X ext | | 3 | ns |
| 8 | $t_h(\text{CKRX-AXR})$ | Hold time, AXR input valid after ACLKR/X latches data | ACLKR/X int | | 3 | ns |
| | | | ACLKR/X ext input | | 4 | ns |
| | | | ACLKR/X ext output | | 6 | 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

- (2) There is a clock ratio requirement between the system infrastructure clock, SYSCLK3, and the McASP0 bit clocks, ACLKR/ACLKX. For proper device operation, the ACLKR/ACLKX frequency **must be no** faster than ½ of SYSCLK3 frequency.

Table 6-54. Switching Characteristics Over Recommended Operating Conditions for McASP⁽¹⁾ (2)
(see [Figure 6-32](#) and [Figure 6-33](#))⁽³⁾

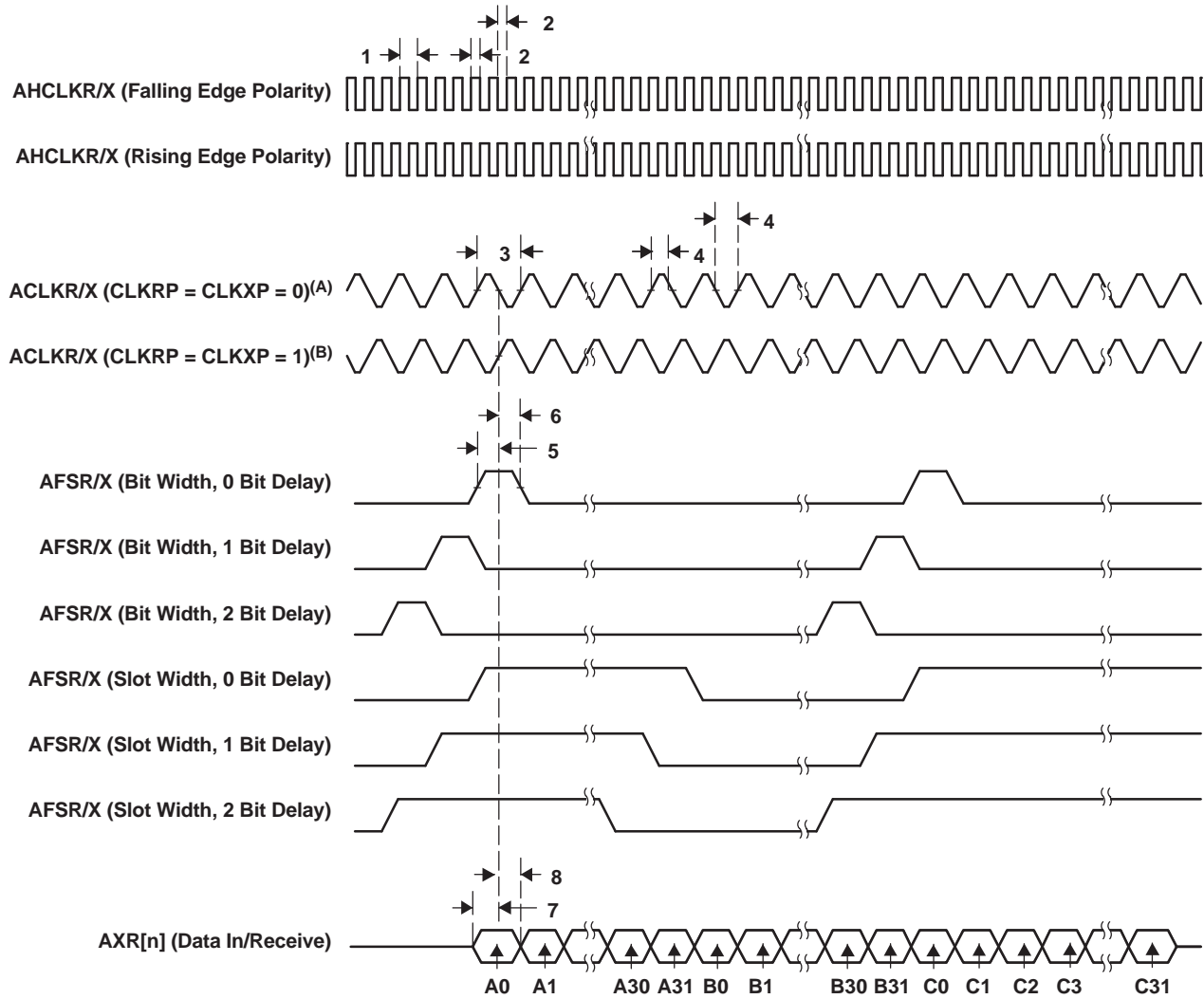
| NO. | PARAMETER | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT | |
|-----|-----------------------|---|-------------------------------|-------|------|----|
| | | | MIN | MAX | | |
| 9 | $t_{c(AHCKRX)}$ | Cycle time, AHCLKR/X | 25 | | ns | |
| 10 | $t_{w(AHCKRX)}$ | Pulse duration, AHCLKR/X high or low | AH - 2.5 | | ns | |
| 11 | $t_{c(CKRX)}$ | Cycle time, ACLKR/X ⁽⁴⁾ | ACLKR/X int 25 | | ns | |
| 12 | $t_{w(CKRX)}$ | Pulse duration, ACLKR/X high or low | ACLKR/X int A - 2.5 | | ns | |
| 13 | $t_{d(CKRX-FRX)}$ | Delay time, ACLKR/X transmit edge to AFSX/R output valid | ACLKR/X int | -2.25 | 5.5 | ns |
| | | | ACLKR/X ext input | 0 | 12.5 | ns |
| | | | ACLKR/X ext output | 0 | 14 | ns |
| 14 | $t_{d(CKX-AXRV)}$ | Delay time, ACLKX transmit edge to AXR output valid | ACLKX int | -2.25 | 5.5 | ns |
| | | | ACLKX ext input | 0 | 12.5 | ns |
| | | | ACLKX ext output | 0 | 14 | ns |
| 15 | $t_{dis(CKRX-AXRHZ)}$ | Disable time, AXR high impedance following last data bit from ACLKR/X transmit edge | ACLKR/X int | -4.5 | 8 | ns |
| | | | ACLKR/X ext | -4.5 | 12.5 | ns |

(1) A = (ACLKR/X period)/2 in ns. For example, when ACLKR/X period is 25 ns, use A = 12.5 ns.

(2) AH = (AHCLKR/X period)/2 in ns. For example, when AHCLKR/X period is 25 ns, use AH = 12.5 ns.

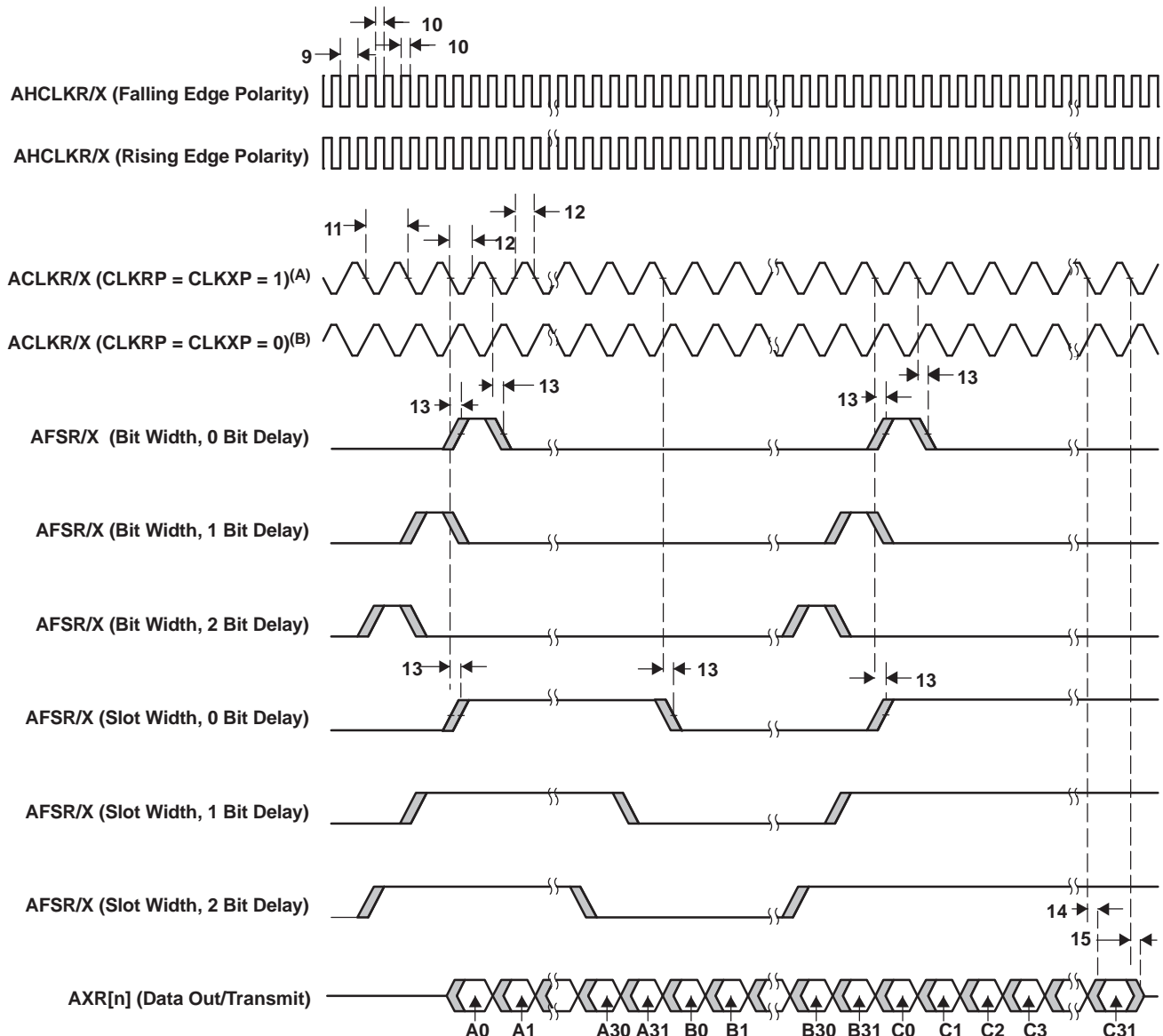
(3) ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
 ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
 ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1
 ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
 ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
 ACLKR external output: ACLKRCTL.CLKRM=0,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

(4) There is a clock ratio requirement between the system infrastructure clock, SYSCLK3, and the McASP0 bit clocks, ACLKR/ACLKX. For proper device operation, the ACLKR/ACLKX frequency **must be no** faster than ½ of SYSCLK3 frequency.



- 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 6-32. McASP Input Timings



- 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 6-33. McASP Output Timings

6.15 Ethernet Media Access Controller (EMAC)

The Ethernet Media Access Controller (EMAC) provides an efficient interface between C6424 and the network. The C6424 EMAC supports two interface modes – Media Independent Interface (MII) and Reduced Media Independent Interface (RMII). The MII mode supports both 10Base-T (10 Mbps/second [Mbps]) and 100Base-TX (100 Mbps) in either half- or full-duplex mode. The RMII mode supports both 10Base-T (10 Mbps/second [Mbps]) and 100Base-TX (100 Mbps) in full-duplex mode only. The EMAC module also supports hardware flow control and quality of service (QOS).

The EMAC controls the flow of packet data from the C6424 device to the PHY. The MDIO module controls PHY configuration and status monitoring.

The EMAC module conforms to the IEEE 802.3-2002 standard, describing the “Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer” specifications. The IEEE 802.3 standard has also been adopted by ISO/IEC and re-designated as ISO/IEC 8802-3:2000(E).

Deviation from this standard, the EMAC 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 EMAC will intentionally generate an incorrect checksum by inverting the frame CRC, so that the transmitted frame will be detected as an error by the network.

Both the EMAC and the MDIO modules interface to the C6424 device through a custom interface that allows efficient data transmission and reception. This custom interface is referred to as the EMAC control module, and is considered integral to the EMAC/MDIO peripheral. The control module is also used to multiplex and control interrupts.

For more details on the C6424 EMAC peripheral, see the *TMS320C6424 Ethernet Media Access Controller (EMAC)/Management Data Input/Output (MDIO) Module User's Guide* (literature number [SPRUJEM6](#)). For a list of supported registers and register fields, see [Table 6-55](#) [Ethernet MAC (EMAC) Control Registers] and [Table 6-56](#) [EMAC Statistics Registers] in this data manual.

6.15.1 EMAC Device-Specific Information

Interface Modes

The EMAC module on the TMS320C6424 supports two interface modes: Media Independent Interface (MII) and Reduced Media Independent Interface (RMII). The MII interface mode is defined in the IEEE 802.3-2002 standard.

The RMII mode of the EMAC conforms to the RMII Specification (revision 1.2), as written by the RMII Consortium. As the name implies, the Reduced Media Independent Interface (RMII) mode is a reduced pin count version of the MII mode and only supports full-duplex mode.

Interface Mode Select

Although, the EMAC uses different pins for the MII and RMII modes, only one mode can be used at a time because both modes share the same EMAC peripheral module. It is the user's responsibility to select only one mode via the PINMUX1 register settings (specifically, the PCIEN, HOSTBK, and RMII bit fields). For a detailed description of pin functions, see [Section 2.5](#), *Terminal Functions*.

Note: In addition, the EMAC *must* be placed in reset (via the Power and Sleep Controller [PSC]) before programming the PINMUX0 and PINMUX1 registers to select the EMAC pins.

Using the RMII Mode of the EMAC

The EMAC contains logic that allows it to communicate using the Reduced Media Independent Interface (RMII) protocol. This logic must be taken out of reset before being used. To use the RMII mode of the EMAC follow these guidelines:

- Supply a 50 MHz reference clock to the RMREFCLK input pin.
- The PINMUX1 register RMII bit field **must** be programmed to "1" to select the RMII pins and the RMII mode of operation.
- MACCONTROL.RMIISPEED **must** be programmed to the desired operating speed for the RMII interface.
- MACCONTROL.FULLDUPLEX **must** be programmed to "1", selecting full duplex mode for RMII.

6.15.2 EMAC Peripheral Register Description(s)

Table 6-55. Ethernet MAC (EMAC) Control Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|-------------------|---|
| 01C8 0000 | TXIDVER | Transmit Identification and Version Register |
| 01C8 0004 | TXCONTROL | Transmit Control Register |
| 01C8 0008 | TXTEARDOWN | Transmit Teardown Register |
| 01C8 0010 | RXIDVER | Receive Identification and Version Register |
| 01C8 0014 | RXCONTROL | Receive Control Register |
| 01C8 0018 | RXTEARDOWN | Receive Teardown Register |
| 01C8 0080 | TXINTSTATRAW | Transmit Interrupt Status (Unmasked) Register |
| 01C8 0084 | TXINTSTATMASKED | Transmit Interrupt Status (Masked) Register |
| 01C8 0088 | TXINTMASKSET | Transmit Interrupt Mask Set Register |
| 01C8 008C | TXINTMASKCLEAR | Transmit Interrupt Mask Clear Register |
| 01C8 0090 | MACINVECTOR | MAC Input Vector Register |
| 01C8 00A0 | RXINTSTATRAW | Receive Interrupt Status (Unmasked) Register |
| 01C8 00A4 | RXINTSTATMASKED | Receive Interrupt Status (Masked) Register |
| 01C8 00A8 | RXINTMASKSET | Receive Interrupt Mask Set Register |
| 01C8 00AC | RXINTMASKCLEAR | Receive Interrupt Mask Clear Register |
| 01C8 00B0 | MACINTSTATRAW | MAC Interrupt Status (Unmasked) Register |
| 01C8 00B4 | MACINTSTATMASKED | MAC Interrupt Status (Masked) Register |
| 01C8 00B8 | MACINTMASKSET | MAC Interrupt Mask Set Register |
| 01C8 00BC | MACINTMASKCLEAR | MAC Interrupt Mask Clear Register |
| 01C8 0100 | RXMBPENABLE | Receive Multicast/Broadcast/Promiscuous Channel Enable Register |
| 01C8 0104 | RXUNICASTSET | Receive Unicast Enable Set Register |
| 01C8 0108 | RXUNICASTCLEAR | Receive Unicast Clear Register |
| 01C8 010C | RXMAXLEN | Receive Maximum Length Register |
| 01C8 0110 | RXBUFFEROFFSET | Receive Buffer Offset Register |
| 01C8 0114 | RXFILTERLOWTHRESH | Receive Filter Low Priority Frame Threshold Register |
| 01C8 0120 | RX0FLOWTHRESH | Receive Channel 0 Flow Control Threshold Register |
| 01C8 0124 | RX1FLOWTHRESH | Receive Channel 1 Flow Control Threshold Register |
| 01C8 0128 | RX2FLOWTHRESH | Receive Channel 2 Flow Control Threshold Register |
| 01C8 012C | RX3FLOWTHRESH | Receive Channel 3 Flow Control Threshold Register |
| 01C8 0130 | RX4FLOWTHRESH | Receive Channel 4 Flow Control Threshold Register |
| 01C8 0134 | RX5FLOWTHRESH | Receive Channel 5 Flow Control Threshold Register |
| 01C8 0138 | RX6FLOWTHRESH | Receive Channel 6 Flow Control Threshold Register |
| 01C8 013C | RX7FLOWTHRESH | Receive Channel 7 Flow Control Threshold Register |
| 01C8 0140 | RX0FREEBUFFER | Receive Channel 0 Free Buffer Count Register |

Table 6-55. Ethernet MAC (EMAC) Control Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-----------------------|-----------------------------------|--|
| 01C8 0144 | RX1FREEBUFFER | Receive Channel 1 Free Buffer Count Register |
| 01C8 0148 | RX2FREEBUFFER | Receive Channel 2 Free Buffer Count Register |
| 01C8 014C | RX3FREEBUFFER | Receive Channel 3 Free Buffer Count Register |
| 01C8 0150 | RX4FREEBUFFER | Receive Channel 4 Free Buffer Count Register |
| 01C8 0154 | RX5FREEBUFFER | Receive Channel 5 Free Buffer Count Register |
| 01C8 0158 | RX6FREEBUFFER | Receive Channel 6 Free Buffer Count Register |
| 01C8 015C | RX7FREEBUFFER | Receive Channel 7 Free Buffer Count Register |
| 01C8 0160 | MACCONTROL | MAC Control Register |
| 01C8 0164 | MACSTATUS | MAC Status Register |
| 01C8 0168 | EMCONTROL | Emulation Control Register |
| 01C8 016C | FIFOCONTROL | FIFO Control Register (Transmit and Receive) |
| 01C8 0170 | MACCONFIG | MAC Configuration Register |
| 01C8 0174 | SOFTRESET | Soft Reset Register |
| 01C8 01D0 | MACSRCADDRLO | MAC Source Address Low Bytes Register (Lower 32-bits) |
| 01C8 01D4 | MACSRCADDRHI | MAC Source Address High Bytes Register (Upper 16-bits) |
| 01C8 01D8 | MACHASH1 | MAC Hash Address Register 1 |
| 01C8 01DC | MACHASH2 | MAC Hash Address Register 2 |
| 01C8 01E0 | BOFFTEST | Back Off Test Register |
| 01C8 01E4 | TPACETEST | Transmit Pacing Algorithm Test Register |
| 01C8 01E8 | RXPAUSE | Receive Pause Timer Register |
| 01C8 01EC | TXPAUSE | Transmit Pause Timer Register |
| 01C8 0200 - 01C8 02FC | (see Table 6-56) | EMAC Statistics Registers |
| 01C8 0500 | MACADDRLO | MAC Address Low Bytes Register |
| 01C8 0504 | MACADDRHI | MAC Address High Bytes Register |
| 01C8 0508 | MACINDEX | MAC Index Register |
| 01C8 0600 | TX0HDP | Transmit Channel 0 DMA Head Descriptor Pointer Register |
| 01C8 0604 | TX1HDP | Transmit Channel 1 DMA Head Descriptor Pointer Register |
| 01C8 0608 | TX2HDP | Transmit Channel 2 DMA Head Descriptor Pointer Register |
| 01C8 060C | TX3HDP | Transmit Channel 3 DMA Head Descriptor Pointer Register |
| 01C8 0610 | TX4HDP | Transmit Channel 4 DMA Head Descriptor Pointer Register |
| 01C8 0614 | TX5HDP | Transmit Channel 5 DMA Head Descriptor Pointer Register |
| 01C8 0618 | TX6HDP | Transmit Channel 6 DMA Head Descriptor Pointer Register |
| 01C8 061C | TX7HDP | Transmit Channel 7 DMA Head Descriptor Pointer Register |
| 01C8 0620 | RX0HDP | Receive Channel 0 DMA Head Descriptor Pointer Register |
| 01C8 0624 | RX1HDP | Receive Channel 1 DMA Head Descriptor Pointer Register |
| 01C8 0628 | RX2HDP | Receive Channel 2 DMA Head Descriptor Pointer Register |
| 01C8 062C | RX3HDP | Receive Channel 3 DMA Head Descriptor Pointer Register |
| 01C8 0630 | RX4HDP | Receive Channel 4 DMA Head Descriptor Pointer Register |
| 01C8 0634 | RX5HDP | Receive Channel 5 DMA Head Descriptor Pointer Register |
| 01C8 0638 | RX6HDP | Receive Channel 6 DMA Head Descriptor Pointer Register |
| 01C8 063C | RX7HDP | Receive Channel 7 DMA Head Descriptor Pointer Register |
| 01C8 0640 | TX0CP | Transmit Channel 0 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0644 | TX1CP | Transmit Channel 1 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0648 | TX2CP | Transmit Channel 2 Completion Pointer (Interrupt Acknowledge) Register |

Table 6-55. Ethernet MAC (EMAC) Control Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|---------|--|
| 01C8 064C | TX3CP | Transmit Channel 3 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0650 | TX4CP | Transmit Channel 4 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0654 | TX5CP | Transmit Channel 5 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0658 | TX6CP | Transmit Channel 6 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 065C | TX7CP | Transmit Channel 7 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0660 | RX0CP | Receive Channel 0 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0664 | RX1CP | Receive Channel 1 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0668 | RX2CP | Receive Channel 2 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 066C | RX3CP | Receive Channel 3 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0670 | RX4CP | Receive Channel 4 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0674 | RX5CP | Receive Channel 5 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 0678 | RX6CP | Receive Channel 6 Completion Pointer (Interrupt Acknowledge) Register |
| 01C8 067C | RX7CP | Receive Channel 7 Completion Pointer (Interrupt Acknowledge) Register |

Table 6-56. EMAC Statistics Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|-------------------|--|
| 01C8 0200 | RXGOODFRAMES | Good Receive Frames Register |
| 01C8 0204 | RXBCASTFRAMES | Broadcast Receive Frames Register (Total number of good broadcast frames received) |
| 01C8 0208 | RXMCASTFRAMES | Multicast Receive Frames Register (Total number of good multicast frames received) |
| 01C8 020C | RXPAUSEFRAMES | Pause Receive Frames Register |
| 01C8 0210 | RXCRCERRORS | Receive CRC Errors Register (Total number of frames received with CRC errors) |
| 01C8 0214 | RXALIGNCODEERRORS | Receive Alignment/Code Errors Register (Total number of frames received with alignment/code errors) |
| 01C8 0218 | RXOVERSIZED | Receive Oversized Frames Register (Total number of oversized frames received) |
| 01C8 021C | RXJABBER | Receive Jabber Frames Register (Total number of jabber frames received) |
| 01C8 0220 | RXUNDERSIZED | Receive Undersized Frames Register (Total number of undersized frames received) |
| 01C8 0224 | RXFRAGMENTS | Receive Frame Fragments Register |
| 01C8 0228 | RXFILTERED | Filtered Receive Frames Register |
| 01C8 022C | RXQOSFILTERED | Received QOS Filtered Frames Register |
| 01C8 0230 | RXOCTETS | Receive Octet Frames Register (Total number of received bytes in good frames) |
| 01C8 0234 | TXGOODFRAMES | Good Transmit Frames Register (Total number of good frames transmitted) |
| 01C8 0238 | TXBCASTFRAMES | Broadcast Transmit Frames Register |
| 01C8 023C | TXMCASTFRAMES | Multicast Transmit Frames Register |
| 01C8 0240 | TXPAUSEFRAMES | Pause Transmit Frames Register |
| 01C8 0244 | TXDEFERRED | Deferred Transmit Frames Register |
| 01C8 0248 | TXCOLLISION | Transmit Collision Frames Register |
| 01C8 024C | TXSINGLECOLL | Transmit Single Collision Frames Register |
| 01C8 0250 | TXMULTICOLL | Transmit Multiple Collision Frames Register |
| 01C8 0254 | TXEXCESSIVECOLL | Transmit Excessive Collision Frames Register |
| 01C8 0258 | TXLATECOLL | Transmit Late Collision Frames Register |
| 01C8 025C | TXUNDERRUN | Transmit Underrun Error Register |
| 01C8 0260 | TXCARRIERSENSE | Transmit Carrier Sense Errors Register |
| 01C8 0264 | TXOCTETS | Transmit Octet Frames Register |
| 01C8 0268 | FRAME64 | Transmit and Receive 64 Octet Frames Register |
| 01C8 026C | FRAME65T127 | Transmit and Receive 65 to 127 Octet Frames Register |
| 01C8 0270 | FRAME128T255 | Transmit and Receive 128 to 255 Octet Frames Register |
| 01C8 0274 | FRAME256T511 | Transmit and Receive 256 to 511 Octet Frames Register |
| 01C8 0278 | FRAME512T1023 | Transmit and Receive 512 to 1023 Octet Frames Register |
| 01C8 027C | FRAME1024TUP | Transmit and Receive 1024 to 1518 Octet Frames Register |
| 01C8 0280 | NETOCTETS | Network Octet Frames Register |
| 01C8 0284 | RXSOFOVERRUNS | Receive FIFO or DMA Start of Frame Overruns Register |
| 01C8 0288 | RXMOFOVERRUNS | Receive FIFO or DMA Middle of Frame Overruns Register |
| 01C8 028C | RXDMAOVERRUNS | Receive DMA Start of Frame and Middle of Frame Overruns Register |

Table 6-57. EMAC Control Module Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|-----------|----------------------------|
| 0x01C8 1004 | EWCTL | Interrupt control register |
| 0x01C8 1008 | EWINTTCNT | Interrupt timer count |

Table 6-58. EMAC Control Module RAM

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|---------|---------------------------------------|
| 0x01C8 2000 - 0x01C8 3FFF | | EMAC Control Module Descriptor Memory |

6.15.3 EMAC Electrical Data/Timing (MII and RMII)

6.15.3.1 EMAC MII Electrical Data/Timing

Table 6-59. Timing Requirements for MRCLK - MII Operation (see Figure 6-34)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|-----------------|----------------------------------|-----|----------|-----|------|
| | | 10 Mbps | | 100 Mbps | | |
| | | MIN | MAX | MIN | MAX | |
| 1 | $t_{c(MRCLK)}$ | Cycle time, MRCLK ⁽¹⁾ | | 400 | 40 | ns |
| 2 | $t_{w(MRCLKH)}$ | Pulse duration, MRCLK high | | 140 | 14 | ns |
| 3 | $t_{w(MRCLKL)}$ | Pulse duration, MRCLK low | | 140 | 14 | ns |

(1) There is a clock ratio requirement between the system infrastructure clock, SYSCLK3, and the EMAC receive/transmit input clocks, MRCLK and MTCLK. For proper device operation, the SYSCLK3 frequency **must be** faster than 12.5 MHz.

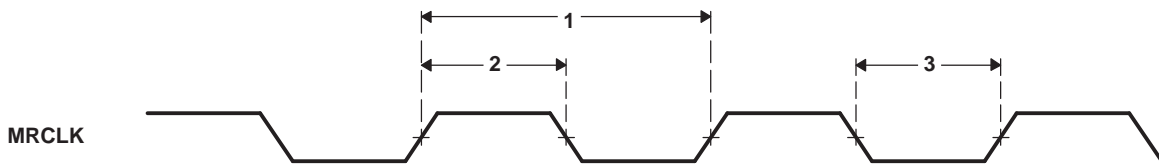


Figure 6-34. MRCLK Timing (EMAC - Receive) [MII Operation]

Table 6-60. Timing Requirements for MTCLK - MII Operation (see Figure 6-34)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | | | UNIT |
|-----|-----------------|----------------------------------|-----|----------|-----|------|
| | | 10 Mbps | | 100 Mbps | | |
| | | MIN | MAX | MIN | MAX | |
| 1 | $t_{c(MTCLK)}$ | Cycle time, MTCLK ⁽¹⁾ | | 400 | 40 | ns |
| 2 | $t_{w(MTCLKH)}$ | Pulse duration, MTCLK high | | 140 | 14 | ns |
| 3 | $t_{w(MTCLKL)}$ | Pulse duration, MTCLK low | | 140 | 14 | ns |

(1) There is a clock ratio requirement between the system infrastructure clock, SYSCLK3, and the EMAC receive/transmit input clocks, MRCLK and MTCLK. For proper device operation, the SYSCLK3 frequency **must be** faster than 12.5 MHz.

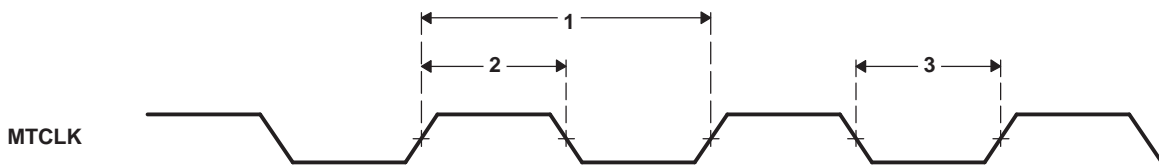


Figure 6-35. MTCLK Timing (EMAC - Transmit) [MII Operation]

Table 6-61. Timing Requirements for EMAC MII Receive 10/100 Mbit/s⁽¹⁾ (see Figure 6-36)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT | |
|-----|-----------------------|--|-----|------|----|
| | | MIN | MAX | | |
| 1 | $t_{su(MRXD-MRCLKH)}$ | Setup time, receive selected signals valid before MRCLK high | | 8 | ns |
| 2 | $t_{h(MRCLKH-MRXD)}$ | Hold time, receive selected signals valid after MRCLK high | | 8 | ns |

(1) Receive selected signals include: MRXD3-MRXD0, MRXDV, and MRXER.

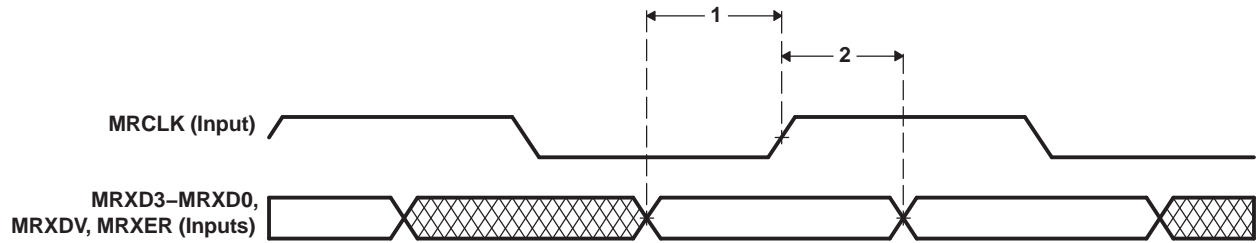


Figure 6-36. EMAC Receive Interface Timing [MII Operation]

Table 6-62. Switching Characteristics Over Recommended Operating Conditions for EMAC MII Transmit 10/100 Mbit/s⁽¹⁾ (see Figure 6-37)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{d(MTCLKH-MTXD)}$ Delay time, MTCLK high to transmit selected signals valid | 2 | 25 | ns |

(1) Transmit selected signals include: MTXD3-MTXD0, and MTXEN.

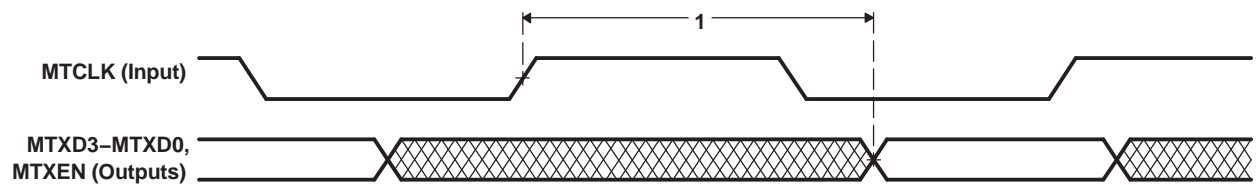


Figure 6-37. EMAC Transmit Interface Timing [MII Operation]

6.15.3.2 EMAC RMII Electrical Data/Timing

The RMREFCLK pin is used to source a clock to the EMAC when it is configured for RMII operation. The RMREFCLK frequency should be 50 MHz \pm 50 PPM with a duty cycle between 35% and 65%, inclusive.

Table 6-63. Timing Requirements for RMREFCLK - RMII Operation (see Figure 6-38)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{c(RMREFCLK)}$ Cycle time, RMREFCLK ⁽¹⁾ | 20 | | ns |
| 2 | $t_{w(RMREFCLKH)}$ Pulse duration, RMREFCLK high | 7 | 13 | ns |
| 3 | $t_{w(RMREFCLKL)}$ Pulse duration, RMREFCLK low | 7 | 13 | ns |
| 4 | $t_{t(RMREFCLK)}$ Transition time, RMREFCLK | | 2 | ns |

(1) There is a clock ratio requirement between the system infrastructure clock, SYSCLK3, and the EMAC RMII reference clock, RMREFCLK. For proper device operation, the SYSCLK3 frequency **must be** faster than 12.5 MHz.

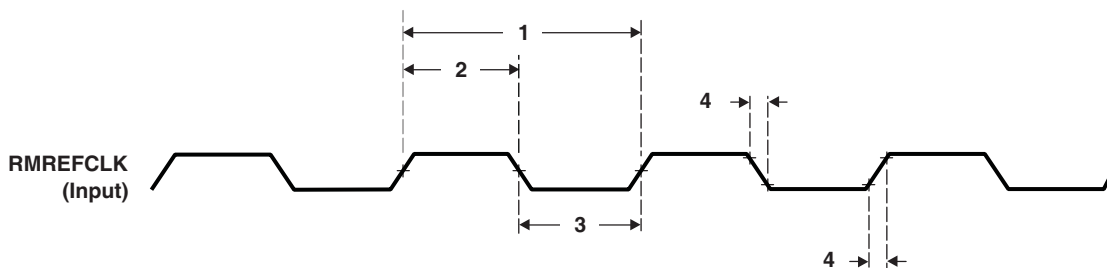


Figure 6-38. RMREFCLK Timing [RMII Operation]

Table 6-64. Timing Requirements for EMAC RMII Receive 10/100 Mbit/s⁽¹⁾ (see Figure 6-39)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_{su}(RMRXD-REFCLKH)$ Setup time, receive selected signals valid before RMREFCLK high | 4 | | ns |
| 2 | $t_h(REFCLKH-RMRXD)$ Hold time, receive selected signals valid after RMREFCLK high | 2 | | ns |

(1) Receive selected signals include: RMRXD1-RMRXD0, RMCRSDV, and RMRXER.

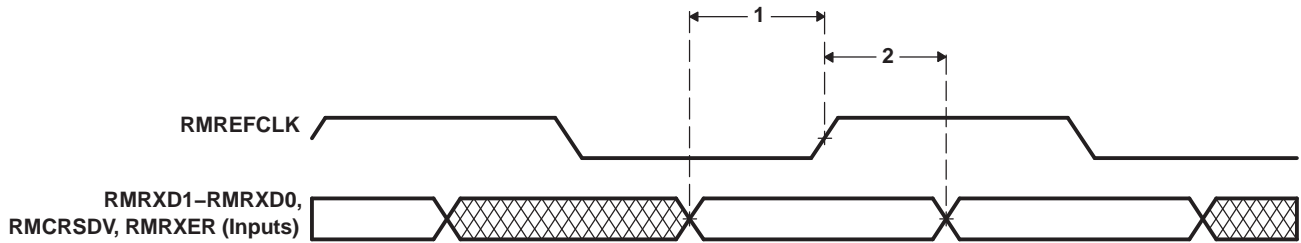


Figure 6-39. EMAC Receive Interface Timing [RMII Operation]

Table 6-65. Switching Characteristics Over Recommended Operating Conditions for EMAC RMII Transmit 10/100 Mbit/s⁽¹⁾ (see Figure 6-40)

| NO. | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--|-------------------------------|------|------|
| | | MIN | MAX | |
| 1 | $t_d(REFCLKH-MTXD)$ Delay time, RMREFCLK high to transmit selected signals valid | 2.2 | 15.5 | ns |

(1) Transmit selected signals include: RMTXD1-RMTXD0, and RMTXEN.

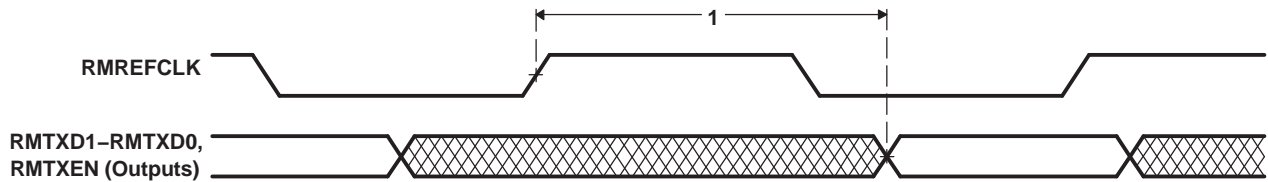


Figure 6-40. EMAC Transmit Interface Timing [RMII Operation]

6.16 Management Data Input/Output (MDIO)

The Management Data Input/Output (MDIO) module continuously polls all 32 MDIO addresses in order to enumerate all PHY devices in the system.

The Management Data Input/Output (MDIO) module implements the 802.3 serial management interface to interrogate and control Ethernet PHY(s) using a shared two-wire bus. Host software uses the MDIO module to configure the auto-negotiation parameters of each PHY attached to the EMAC, retrieve the negotiation results, and configure required parameters in the EMAC 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 one PHY may be connected at any given time.

For more detailed information on the MDIO peripheral, see the *TMS320C642x Ethernet Media Access Controller (EMAC)/Management Data Input/Output (MDIO) Module User's Guide* (literature number SPRUEM6). For a list of supported registers and register fields, see [Table 6-66](#) [MDIO Registers] in this data manual.

6.16.1 Peripheral Register Description(s)

Table 6-66. MDIO Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|------------------|--|
| 0x01C8 4000 | – | Reserved |
| 0x01C8 4004 | CONTROL | MDIO Control Register |
| 0x01C8 4008 | ALIVE | MDIO PHY Alive Status Register |
| 0x01C8 400C | LINK | MDIO PHY Link Status Register |
| 0x01C8 4010 | LINKINTRAW | MDIO Link Status Change Interrupt (Unmasked) Register |
| 0x01C8 4014 | LINKINTMASKED | MDIO Link Status Change Interrupt (Masked) Register |
| 0x01C8 4018 | – | Reserved |
| 0x01C8 4020 | USERINTRAW | MDIO User Command Complete Interrupt (Unmasked) Register |
| 0x01C8 4024 | USERINTMASKED | MDIO User Command Complete Interrupt (Masked) Register |
| 0x01C8 4028 | USERINTMASKSET | MDIO User Command Complete Interrupt Mask Set Register |
| 0x01C8 402C | USERINTMASKCLEAR | MDIO User Command Complete Interrupt Mask Clear Register |
| 0x01C8 4030 - 0x01C8 407C | – | Reserved |
| 0x01C8 4080 | USERACCESS0 | MDIO User Access Register 0 |
| 0x01C8 4084 | USERPHYSEL0 | MDIO User PHY Select Register 0 |
| 0x01C8 4088 | USERACCESS1 | MDIO User Access Register 1 |
| 0x01C8 408C | USERPHYSEL1 | MDIO User PHY Select Register 1 |
| 0x01C8 4090 - 0x01C8 47FF | – | Reserved |

6.16.2 Management Data Input/Output (MDIO) Electrical Data/Timing

Table 6-67. Timing Requirements for MDIO Input (see Figure 6-41 and Figure 6-42)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|-----------------------|---|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 1 | $t_{c(MDCLK)}$ | Cycle time, MDCLK | 400 | | ns |
| 2 | $t_{w(MDCLK)}$ | Pulse duration, MDCLK high/low | 180 | | ns |
| 3 | $t_t(MDCLK)$ | Transition time, MDCLK | | 5 | ns |
| 4 | $t_{su(MDIO-MDCLKH)}$ | Setup time, MDIO data input valid before MDCLK high | 10 | | ns |
| 5 | $t_h(MDCLKH-MDIO)$ | Hold time, MDIO data input valid after MDCLK high | 10 | | ns |

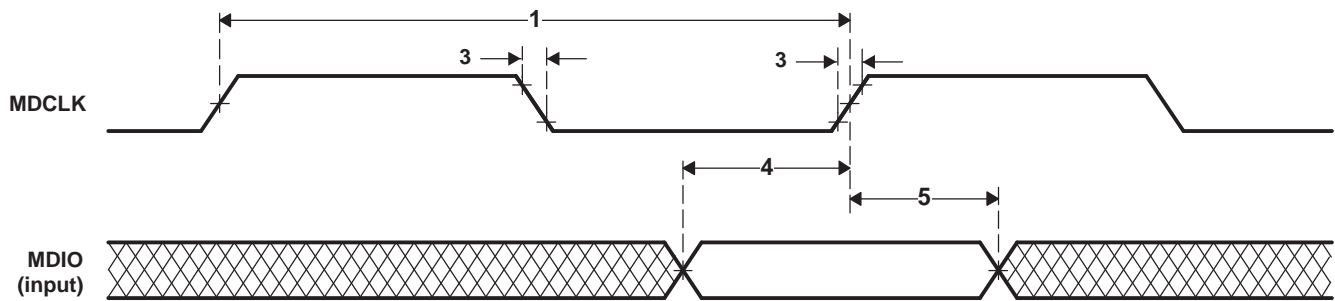


Figure 6-41. MDIO Input Timing

Table 6-68. Switching Characteristics Over Recommended Operating Conditions for MDIO Output (see Figure 6-42)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------------|---|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 7 | $t_d(MDCLKL-MDIO)$ | Delay time, MDCLK low to MDIO data output valid | | 100 | ns |

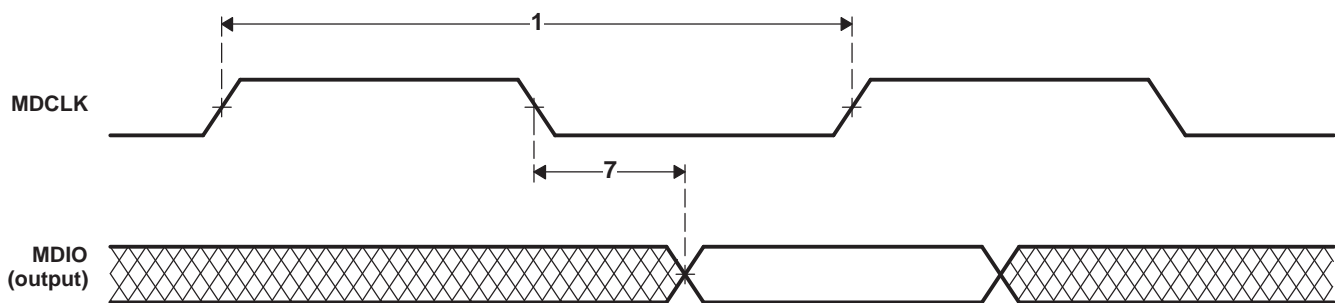


Figure 6-42. MDIO Output Timing

6.17 Timers

The C6424 device has 3 64-bit general-purpose timers which have the following features:

- 64-bit count-up counter
- Timer modes:
 - 64-bit general-purpose timer mode (Timer 0 and 1)
 - Dual 32-bit general-purpose timer mode (Timer 0 and 1)
 - Watchdog timer mode (Timer 2)
- 2 possible clock sources:
 - Internal clock
 - External clock input via timer input pin TINPL (Timer 0 and 1 only)
- 2 operation modes:
 - One-time operation (timer runs for one period then stops)
 - Continuous operation (timer automatically resets after each period)
- Generates interrupts to the DSP
- Generates sync event to EDMA
- Causes device global reset upon watchdog timer timeout (Timer 2 only)

For more detailed information, see the *TMS320C642x DSP 64-Bit Timer User's Guide* (literature number SPRUEN5).

6.17.1 Timer Peripheral Register Description(s)

Table 6-69. Timer 0 Registers

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---------------------------|---------------|---|
| 0x01C2 1400 | - | Reserved |
| 0x01C2 1404 | EMUMGT_CLKSPD | Timer 0 Emulation Management/Clock Speed Register |
| 0x01C2 1410 | TIM12 | Timer 0 Counter Register 12 |
| 0x01C2 1414 | TIM34 | Timer 0 Counter Register 34 |
| 0x01C2 1418 | PRD12 | Timer 0 Period Register 12 |
| 0x01C2 141C | PRD34 | Timer 0 Period Register 34 |
| 0x01C2 1420 | TCR | Timer 0 Control Register |
| 0x01C2 1424 | TGCR | Timer 0 Global Control Register |
| 0x01C2 1428 - 0x01C2 17FF | - | Reserved |

Table 6-70. Timer 1 Registers

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---------------------------|---------------|---|
| 0x01C2 1800 | - | Reserved |
| 0x01C2 1804 | EMUMGT_CLKSPD | Timer 1 Emulation Management/Clock Speed Register |
| 0x01C2 1810 | TIM12 | Timer 1 Counter Register 12 |
| 0x01C2 1814 | TIM34 | Timer 1 Counter Register 34 |
| 0x01C2 1818 | PRD12 | Timer 1 Period Register 12 |
| 0x01C2 181C | PRD34 | Timer 1 Period Register 34 |
| 0x01C2 1820 | TCR | Timer 1 Control Register |
| 0x01C2 1824 | TGCR | Timer 1 Global Control Register |
| 0x01C2 1828 - 0x01C2 1BFF | - | Reserved |

Table 6-71. Timer 2 (Watchdog) Registers

| HEX ADDRESS RANGE | ACRONYM | DESCRIPTION |
|---------------------------|---------------|---|
| 0x01C2 1C00 | - | Reserved |
| 0x01C2 1C04 | EMUMGT_CLKSPD | Timer 2 Emulation Management/Clock Speed Register |
| 0x01C2 1C10 | TIM12 | Timer 2 Counter Register 12 |
| 0x01C2 1C14 | TIM34 | Timer 2 Counter Register 34 |
| 0x01C2 1C18 | PRD12 | Timer 2 Period Register 12 |
| 0x01C2 1C1C | PRD34 | Timer 2 Period Register 34 |
| 0x01C2 1C20 | TCR | Timer 2 Control Register |
| 0x01C2 1C24 | TGCR | Timer 2 Global Control Register |
| 0x01C2 1C28 | WDTCR | Timer 2 Watchdog Timer Control Register |
| 0x01C2 1C2C - 0x01C2 1FFF | - | Reserved |

6.17.2 Timer Electrical Data/Timing

Table 6-72. Timing Requirements for Timer Input⁽¹⁾ ⁽²⁾ ⁽³⁾ (see Figure 6-43)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------|-----------------------------|--|-------|------|
| | | | MIN | MAX | |
| 1 | $t_{w(TINPH)}$ | Pulse duration, TINPxL high | TINP0L, if TIMERCTL.TINPOSEL = 0 [default] | 2P | ns |
| | | | TINP0L, if TIMERCTL.TINPOSEL = 1 | 0.33P | ns |
| | | | TINP1L | 2P | ns |
| 2 | $t_{w(TINPL)}$ | Pulse duration, TINPxL low | TINP0L, if TIMERCTL.TINPOSEL = 0 [default] | 2P | ns |
| | | | TINP0L, if TIMERCTL.TINPOSEL = 1 | 0.33P | ns |
| | | | TINP1L | 2P | ns |

- (1) P = MXI/CLKIN cycle time in ns. For example, when MXI/CLKIN frequency is 25 MHz, use P = 40 ns.
- (2) The TIMERCTL.TINPOSEL field in the System Module determines if the TINP0L input directly goes to Timer 0 (TIMERCTL.TINPOSEL=0), or if the TINP0L input is first divided down by 6 before going to Timer 0 (TIMERCTL.TINPOSEL=1).
- (3) TINP1L input goes directly to Timer 1.

Table 6-73. Switching Characteristics Over Recommended Operating Conditions for Timer Output⁽¹⁾ (see Figure 6-43)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------|-----------------------------|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 3 | $t_{w(TOUTH)}$ | Pulse duration, TOUTxL high | P | | ns |
| 4 | $t_{w(TOURL)}$ | Pulse duration, TOUTxL low | P | | ns |

- (1) P = MXI/CLKIN cycle time in ns. For example, when MXI/CLKIN frequency is 25 MHz, use P = 40 ns.

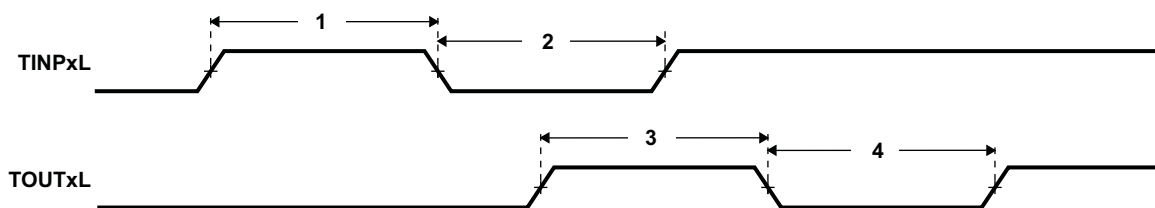


Figure 6-43. Timer Timing

6.18 Peripheral Component Interconnect (PCI)

The C6424 DSP supports connections to PCI-compliant devices via the integrated PCI master/slave bus interface. The PCI port interfaces to DSP internal resources via the data switched central resource. The data switched central resource is described in more detail in [Section 4, System Interconnect](#).

For more detailed information on the PCI port peripheral module, see the *TMS320C642x DSP Peripheral Component Interconnect (PCI) User's Guide* (literature number SPRUEN3).

6.18.1 PCI Device-Specific Information

The PCI peripheral on the C6424 DSP conforms to the *PCI Local Bus Specification Revision 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 33 MHz and uses a 32-bit data/address bus.

On the C6424 device, the pins of the PCI peripheral are multiplexed with the pins of the EMIFA, GPIO, HPI, VLYNQ, and EMAC peripherals. For more detailed information on how to select PCI, see [Section 3, Device Configurations](#).

The C6424 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 6-74](#) shows the registers which can be initialized through the PCI auto-initialization. The default value of these registers when PCI auto-initialization is **not** used is shown in [Table 6-74](#). PCI auto-initialization is enabled by selecting PCI boot with auto-initialization. For information on how to select PCI boot with auto-initialization, see [Section 3.4.1, Boot Modes](#). For more information on PCI auto-initialization, see the *TMS320C642x DSP Peripheral Component Interconnect (PCI) User's Guide* (literature number SPRUEN3) and the *Using the TMS320C42x Bootloader Application Report* (literature number [SPRAAK5](#)).

The PCI peripheral is a master peripheral within the C6424 DSP.

Table 6-74. Default Values for PCI Configuration Registers

| REGISTER | DEFAULT VALUE (HEX) |
|------------------|---------------------|
| Vendor ID | 104C |
| Device ID | B001 |
| Class Code | 11 8000 |
| Revision ID | 01 |
| System Vendor ID | 0000 |
| Subsystem ID | 0000 |
| Max Latency | 00 |
| Min Grant | 00 |
| Interrupt Pin | 00 |
| Interrupt Line | 00 |

6.18.2 PCI Peripheral Register Description(s)

Table 6-75. PCI Memory-Mapped Registers

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---------------------------------|--------------|--|
| 01C1 A000 - 01C1 A00F | - | Reserved |
| 01C1 A010 | PCISTATSET | PCI Status Set Register |
| 01C1 A014 | PCISTATCLR | PCI Status Clear Register |
| 01C1 A018 - 01C1 A01F | - | Reserved |
| 01C1 A020 | PCIHINTSET | PCI Host Interrupt Enable Set Register |
| 01C1 A024 | PCIHINTCLR | PCI Host Interrupt Enable Clear Register |
| 01C1 A028 - 01C1 A02F | - | Reserved |
| 01C1 A030 | PCIDINTSET | PCI DSP Interrupt Enable Set Register |
| 01C1 A034 | PCIDINTCLR | PCI DSP Interrupt Enable Clear Register |
| 01C1 A038 - 01C1 A0FF | - | Reserved |
| 01C1 A100 | PCIVENDEVMIR | PCI Vendor ID/Device ID Mirror Register |
| 01C1 A104 | PCICSRMIR | PCI Command/Status Mirror Register |
| 01C1 A108 | PCICLREVMIR | PCI Class Code/Revision ID Mirror Register |
| 01C1 A10C | PCICLINEMIR | PCI BIST/Header Type/Latency Timer/Cacheline Size Mirror Register |
| 01C1 A110 | PCIBAR0MSK | PCI Base Address Mask Register 0 |
| 01C1 A114 | PCIBAR1MSK | PCI Base Address Mask Register 1 |
| 01C1 A118 | PCIBAR2MSK | PCI Base Address Mask Register 2 |
| 01C1 A11C | PCIBAR3MSK | PCI Base Address Mask Register 3 |
| 01C1 A120 | PCIBAR4MSK | PCI Base Address Mask Register 4 |
| 01C1 A124 | PCIBAR5MSK | PCI Base Address Mask Register 5 |
| 01C1 A128 - 01C1 A12B | - | Reserved |
| 01C1 A12C | PCISUBIDMIR | PCI Subsystem Vendor ID/Subsystem ID Mirror Register |
| 01C1 A130 | - | Reserved |
| 01C1 A134 | PCICBPTRMIR | PCI Capabilities Pointer Mirror Register |
| 01C1 A138 - 01C1 A13B | - | Reserved |
| 01C1 A13C | PCILGINTMIR | PCI Max Latency/Min Grant/Interrupt Pin/Interrupt Line Mirror Register |
| 01C1 A140 - 01C1 A17F | - | Reserved |
| 01C1 A180 | PCISLVCNTL | PCI Slave Control Register |
| 01C1 A184 - 01C1 A1BF | - | Reserved |
| 01C1 A1C0 | PCIBAR0TRL | PCI Slave Base Address 0 Translation Register |
| 01C1 A1C4 | PCIBAR1TRL | PCI Slave Base Address 1 Translation Register |
| 01C1 A1C8 | PCIBAR2TRL | PCI Slave Base Address 2 Translation Register |
| 01C1 A1CC | PCIBAR3TRL | PCI Slave Base Address 3 Translation Register |
| 01C1 A1D0 | PCIBAR4TRL | PCI Slave Base Address 4 Translation Register |
| 01C1 A1D4 | PCIBAR5TRL | PCI Slave Base Address 5 Translation Register |
| 01C1 A1D8 - 01C1 A1DF | - | Reserved |
| 01C1 A1E0 | PCIBAR0MIR | PCI Base Address Register 0 Mirror Register |
| 01C1 A1E4 | PCIBAR1MIR | PCI Base Address Register 1 Mirror Register |
| 01C1 A1E8 | PCIBAR2MIR | PCI Base Address Register 2 Mirror Register |
| 01C1 A1EC | PCIBAR3MIR | PCI Base Address Register 3 Mirror Register |
| 01C1 A1F0 | PCIBAR4MIR | PCI Base Address Register 4 Mirror Register |
| 01C1 A1F4 | PCIBAR5MIR | PCI Base Address Register 5 Mirror Register |
| 01C1 A1F8 - 01C1 A2FF | - | Reserved |
| 01C1 A300 | PCIMCFGDAT | PCI Master Configuration/IO Access Data Register |

Table 6-75. PCI Memory-Mapped Registers (continued)

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---------------------------------|-------------|---|
| 01C1 A304 | PCIMCFGADR | PCI Master Configuration/IO Access Address Register |
| 01C1 A308 | PCIMCFGCMD | PCI Master Configuration/IO Access Command Register |
| 01C1 A30C - 01C1 A30F | - | Reserved |
| 01C1 A310 | PCIMSTCFG | PCI Master Configuration Register |
| 01C1 A314 | PCIADDSUB0 | PCI Address Substitution 0 Register |
| 01C1 A318 | PCIADDSUB1 | PCI Address Substitution 1 Register |
| 01C1 A31C | PCIADDSUB2 | PCI Address Substitution 2 Register |
| 01C1 A320 | PCIADDSUB3 | PCI Address Substitution 3 Register |
| 01C1 A324 | PCIADDSUB4 | PCI Address Substitution 4 Register |
| 01C1 A328 | PCIADDSUB5 | PCI Address Substitution 5 Register |
| 01C1 A32C | PCIADDSUB6 | PCI Address Substitution 6 Register |
| 01C1 A330 | PCIADDSUB7 | PCI Address Substitution 7 Register |
| 01C1 A334 | PCIADDSUB8 | PCI Address Substitution 8 Register |
| 01C1 A338 | PCIADDSUB9 | PCI Address Substitution 9 Register |
| 01C1 A33C | PCIADDSUB10 | PCI Address Substitution 10 Register |
| 01C1 A340 | PCIADDSUB11 | PCI Address Substitution 11 Register |
| 01C1 A344 | PCIADDSUB12 | PCI Address Substitution 12 Register |
| 01C1 A348 | PCIADDSUB13 | PCI Address Substitution 13 Register |
| 01C1 A34C | PCIADDSUB14 | PCI Address Substitution 14 Register |
| 01C1 A350 | PCIADDSUB15 | PCI Address Substitution 15 Register |
| 01C1 A354 | PCIADDSUB16 | PCI Address Substitution 16 Register |
| 01C1 A358 | PCIADDSUB17 | PCI Address Substitution 17 Register |
| 01C1 A35C | PCIADDSUB18 | PCI Address Substitution 18 Register |
| 01C1 A360 | PCIADDSUB19 | PCI Address Substitution 19 Register |
| 01C1 A364 | PCIADDSUB20 | PCI Address Substitution 20 Register |
| 01C1 A368 | PCIADDSUB21 | PCI Address Substitution 21 Register |
| 01C1 A36C | PCIADDSUB22 | PCI Address Substitution 22 Register |
| 01C1 A370 | PCIADDSUB23 | PCI Address Substitution 23 Register |
| 01C1 A374 | PCIADDSUB24 | PCI Address Substitution 24 Register |
| 01C1 A378 | PCIADDSUB25 | PCI Address Substitution 25 Register |
| 01C1 A37C | PCIADDSUB26 | PCI Address Substitution 26 Register |
| 01C1 A380 | PCIADDSUB27 | PCI Address Substitution 27 Register |
| 01C1 A384 | PCIADDSUB28 | PCI Address Substitution 28 Register |
| 01C1 A388 | PCIADDSUB29 | PCI Address Substitution 29 Register |
| 01C1 A38C | PCIADDSUB30 | PCI Address Substitution 30 Register |
| 01C1 A390 | PCIADDSUB31 | PCI Address Substitution 31 Register |

Table 6-76. PCI Hook Configuration Registers

| DSP ACCESS HEX ADDRESS RANGE | ACRONYM | DSP ACCESS REGISTER NAME |
|---------------------------------|--------------|---|
| 01C1 A394 | PCIVENDEVPRG | PCI Vendor ID and Device ID Program Register |
| 01C1 A398 | - | Reserved |
| 01C1 A39C | PCICLREVPGRG | PCI Class Code and Revision ID Program Register |
| 01C1 A3A0 | PCISUBIDPRG | PCI Subsystem Vendor ID and Subsystem ID Program Register |
| 01C1 A3A4 | PCIMAXLGPRG | PCI Max Latency and Min Grant Program Register |
| 01C1 A3A8 | - | Reserved |
| 01C1 A3AC | PCICFGDONE | PCI Configuration Done Register |
| 01C1 A3B0 - 01C1 A7FF | - | Reserved |

Table 6-77. PCI External Memory Space

| HEX ADDRESS RANGE | ACRONYM | PCI MASTER WINDOW NAME |
|-----------------------|---------|------------------------|
| 3000 0000 - 307F FFFF | - | PCI Master Window 0 |
| 3080 0000 - 30FF FFFF | - | PCI Master Window 1 |
| 3100 0000 - 317F FFFF | - | PCI Master Window 2 |
| 3180 0000 - 31FF FFFF | - | PCI Master Window 3 |
| 3200 0000 - 327F FFFF | - | PCI Master Window 4 |
| 3280 0000 - 32FF FFFF | - | PCI Master Window 5 |
| 3300 0000 - 337F FFFF | - | PCI Master Window 6 |
| 3380 0000 - 33FF FFFF | - | PCI Master Window 7 |
| 3400 0000 - 347F FFFF | - | PCI Master Window 8 |
| 3480 0000 - 34FF FFFF | - | PCI Master Window 9 |
| 3500 0000 - 357F FFFF | - | PCI Master Window 10 |
| 3580 0000 - 35FF FFFF | - | PCI Master Window 11 |
| 3600 0000 - 367F FFFF | - | PCI Master Window 12 |
| 3680 0000 - 36FF FFFF | - | PCI Master Window 13 |
| 3700 0000 - 377F FFFF | - | PCI Master Window 14 |
| 3780 0000 - 37FF FFFF | - | PCI Master Window 15 |
| 3800 0000 - 387F FFFF | - | PCI Master Window 16 |
| 3880 0000 - 38FF FFFF | - | PCI Master Window 17 |
| 3900 0000 - 397F FFFF | - | PCI Master Window 18 |
| 3980 0000 - 39FF FFFF | - | PCI Master Window 19 |
| 3A00 0000 - 3A7F FFFF | - | PCI Master Window 20 |
| 3A80 0000 - 3AFF FFFF | - | PCI Master Window 21 |
| 3B00 0000 - 3B7F FFFF | - | PCI Master Window 22 |
| 3B80 0000 - 3BFF FFFF | - | PCI Master Window 23 |
| 3C00 0000 - 3C7F FFFF | - | PCI Master Window 24 |
| 3C80 0000 - 3CFF FFFF | - | PCI Master Window 25 |
| 3D00 0000 - 3D7F FFFF | - | PCI Master Window 26 |
| 3D80 0000 - 3DFF FFFF | - | PCI Master Window 27 |
| 3E00 0000 - 3E7F FFFF | - | PCI Master Window 28 |
| 3E80 0000 - 3EFF FFFF | - | PCI Master Window 29 |
| 3F00 0000 - 3F7F FFFF | - | PCI Master Window 30 |
| 3F80 0000 - 3FFF FFFF | - | PCI Master Window 31 |

6.18.3 PCI Electrical Data/Timing

Texas Instruments (TI) has performed the simulation and system characterization to ensure that the PCI peripheral meets all AC timing specifications as required by the *PCI Local Bus Specification Revision 2.3*. Therefore, 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)* of the *PCI Local Bus Specification Revision 2.3*.

Note: The C6424 PCI peripheral **only** supports 3.3-V signaling and 33-MHz operation.

There is a clock ratio requirement between the system infrastructure clock, SYSCLK2, and the PCI clock, PCICLK. For proper device operation, the SYSCLK2 frequency **must be** faster than the PCICLK frequency (33 MHz).

6.19 Pulse Width Modulator (PWM)

The 3 C6424 Pulse Width Modulator (PWM) peripherals support the following features:

- Period counter
- First-phase duration counter
- Repeat count for one-shot operation
- Configurable to operate in either one-shot or continuous mode
- Buffered period and first-phase duration registers
- One-shot operation triggerable by hardware events with programmable edge transitions. (low-to-high or high-to-low).
- One-shot operation generates N+1 periods of waveform, N being the repeat count register value
- Emulation support

The register memory maps for PWM0/1/2 are shown in [Table 6-78](#), [Table 6-79](#), and [Table 6-80](#).

Table 6-78. PWM0 Register Memory Map

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|---------|------------------------------------|
| 0x01C2 2000 | | Reserved |
| 0x01C2 2004 | PCR | PWM0 Peripheral Control Register |
| 0x01C2 2008 | CFG | PWM0 Configuration Register |
| 0x01C2 200C | START | PWM0 Start Register |
| 0x01C2 2010 | RPT | PWM0 Repeat Count Register |
| 0x01C2 2014 | PER | PWM0 Period Register |
| 0x01C2 2018 | PH1D | PWM0 First-Phase Duration Register |
| 0x01C2 201C - 0x01C2 23FF | - | Reserved |

Table 6-79. PWM1 Register Memory Map

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|--------------------------|---------|------------------------------------|
| 0x01C2 2400 | | Reserved |
| 0x01C2 2404 | PCR | PWM1 Peripheral Control Register |
| 0x01C2 2408 | CFG | PWM1 Configuration Register |
| 0x01C2 240C | START | PWM1 Start Register |
| 0x01C2 2410 | RPT | PWM1 Repeat Count Register |
| 0x01C2 2414 | PER | PWM1 Period Register |
| 0x01C2 2418 | PH1D | PWM1 First-Phase Duration Register |
| 0x01C2 241C -0x01C2 27FF | - | Reserved |

Table 6-80. PWM2 Register Memory Map

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|---------|------------------------------------|
| 0x01C2 2800 | | Reserved |
| 0x01C2 2804 | PCR | PWM2 Peripheral Control Register |
| 0x01C2 2808 | CFG | PWM2 Configuration Register |
| 0x01C2 280C | START | PWM2 Start Register |
| 0x01C2 2810 | RPT | PWM2 Repeat Count Register |
| 0x01C2 2814 | PER | PWM2 Period Register |
| 0x01C2 2818 | PH1D | PWM2 First-Phase Duration Register |
| 0x01C2 281C - 0x01C2 2BFF | - | Reserved |

6.19.1 PWM0/1/2 Electrical Data/Timing

Table 6-81. Switching Characteristics Over Recommended Operating Conditions for PWM0/1/2 Outputs
(see [Figure 6-44](#))

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_w(\text{PWMH})$ Pulse duration, PWMx high | 37 | | ns |
| 2 | $t_w(\text{PWML})$ Pulse duration, PWMx low | 37 | | ns |
| 3 | $t_t(\text{PWM})$ Transition time, PWMx | | 5 | ns |

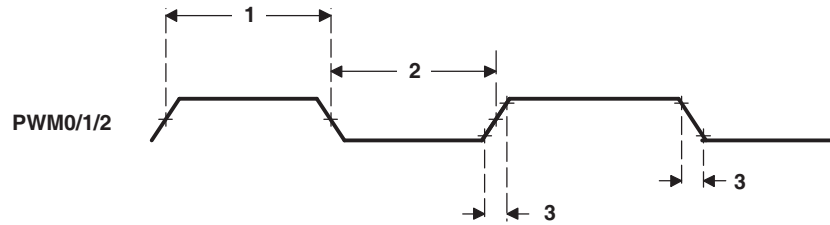


Figure 6-44. PWM Output Timing

6.20 VLYNQ

The C6424 VLYNQ peripheral provides a high speed serial communications interface with the following features.

- Low Pin Count
- Scalable Performance / Support
- Simple Packet Based Transfer Protocol for Memory Mapped Access
 - Write Request / Data Packet
 - Read Request Packet
 - Read Response Data Packet
 - Interrupt Request Packet
- Supports both Symmetric and Asymmetric Operation
 - Tx pins on first device connect to Rx pins on second device and vice versa
 - Data pin widths are automatically detected after reset
 - Request packets, response packets, and flow control information are all multiplexed and sent across the same physical pins
 - Supports both Host/Peripheral and Peer to Peer communication
- Simple Block Code Packet Formatting (8b/10b)
- In Band Flow Control
 - No extra pins needed
 - Allows receiver to momentarily throttle back transmitter when overflow is about to occur
 - Uses built in special code capability of block code to seamlessly interleave flow control information with user data
 - Allows system designer to balance cost of data buffering versus performance
- Multiple outstanding transactions
- Automatic packet formatting optimizations
- Internal loop-back mode

6.20.1 VLYNQ Peripheral Register Description(s)

Table 6-82. VLYNQ Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|-------------------|------------|---|
| 0x01E0 1000 | - | Reserved |
| 0x01E0 1004 | CTRL | VLYNQ Local Control Register |
| 0x01E0 1008 | STAT | VLYNQ Local Status Register |
| 0x01E0 100C | INTPRI | VLYNQ Local Interrupt Priority Vector Status/Clear Register |
| 0x01E0 1010 | INTSTATCLR | VLYNQ Local Unmasked Interrupt Status/Clear Register |
| 0x01E0 1014 | INTPENDSET | VLYNQ Local Interrupt Pending/Set Register |
| 0x01E0 1018 | INTPTR | VLYNQ Local Interrupt Pointer Register |
| 0x01E0 101C | XAM | VLYNQ Local Transmit Address Map Register |
| 0x01E0 1020 | RAMS1 | VLYNQ Local Receive Address Map Size 1 Register |
| 0x01E0 1024 | RAMO1 | VLYNQ Local Receive Address Map Offset 1 Register |
| 0x01E0 1028 | RAMS2 | VLYNQ Local Receive Address Map Size 2 Register |
| 0x01E0 102C | RAMO2 | VLYNQ Local Receive Address Map Offset 2 Register |
| 0x01E0 1030 | RAMS3 | VLYNQ Local Receive Address Map Size 3 Register |
| 0x01E0 1034 | RAMO3 | VLYNQ Local Receive Address Map Offset 3 Register |
| 0x01E0 1038 | RAMS4 | VLYNQ Local Receive Address Map Size 4 Register |
| 0x01E0 103C | RAMO4 | VLYNQ Local Receive Address Map Offset 4 Register |

Table 6-82. VLYNQ Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|-------------|--|
| 0x01E0 1040 | CHIPVER | VLYNQ Local Chip Version Register |
| 0x01E0 1044 | AUTNGO | VLYNQ Local Auto Negotiation Register |
| 0x01E0 1048 | - | Reserved |
| 0x01E0 104C | - | Reserved |
| 0x01E0 1050 - 0x01E0 105C | - | Reserved |
| 0x01E0 1060 | - | Reserved |
| 01E0 10C00 0064 | - | Reserved |
| 0x01E0 1068 - 0x01E0 107C | - | Reserved for future use |
| 0x01E0 1080 | RREVID | VLYNQ Remote Revision Register |
| 0x01E0 1084 | RCTRL | VLYNQ Remote Control Register |
| 0x01E0 1088 | RSTAT | VLYNQ Remote Status Register |
| 0x01E0 108C | RINTPRI | VLYNQ Remote Interrupt Priority Vector Status/Clear Register |
| 0x01E0 1090 | RINTSTATCLR | VLYNQ Remote Unmasked Interrupt Status/Clear Register |
| 0x01E0 1094 | RINTPENDSET | VLYNQ Remote Interrupt Pending/Set Register |
| 0x01E0 1098 | RINTPTR | VLYNQ Remote Interrupt Pointer Register |
| 0x01E0 109C | RXAM | VLYNQ Remote Transmit Address Map Register |
| 0x01E0 10A0 | RRAMS1 | VLYNQ Remote Receive Address Map Size 1 Register |
| 0x01E0 10A4 | RRAMO1 | VLYNQ Remote Receive Address Map Offset 1 Register |
| 0x01E0 10A8 | RRAMS2 | VLYNQ Remote Receive Address Map Size 2 Register |
| 0x01E0 10AC | RRAMO2 | VLYNQ Remote Receive Address Map Offset 2 Register |
| 0x01E0 10B0 | RRAMS3 | VLYNQ Remote Receive Address Map Size 3 Register |
| 0x01E0 10B4 | RRAMO3 | VLYNQ Remote Receive Address Map Offset 3 Register |
| 0x01E0 10B8 | RRAMS4 | VLYNQ Remote Receive Address Map Size 4 Register |
| 0x01E0 10BC | RRAMO4 | VLYNQ Remote Receive Address Map Offset 4 Register |
| 0x01E0 10C0 | RCHIPVER | VLYNQ Remote Chip Version Register (values on the device_id and device_rev pins of remote VLYNQ) |
| 0x01E0 10C4 | RAUTNGO | VLYNQ Remote Auto Negotiation Register |
| 0x01E0 10C8 | RMANNGO | VLYNQ Remote Manual Negotiation Register |
| 0x01E0 10CC | RNGOSTAT | VLYNQ Remote Negotiation Status Register |
| 0x01E0 10D0 - 0x01E0 10DC | - | Reserved |
| 0x01E0 10E0 | RINTVEC0 | VLYNQ Remote Interrupt Vectors 3 - 0 (sourced from vlynq_int_i[3:0] port of remote VLYNQ) |
| 0x01E0 10E4 | RINTVEC1 | VLYNQ Remote Interrupt Vectors 7 - 4 (sourced from vlynq_int_i[7:4] port of remote VLYNQ) |
| 0x01E0 10E8 - 0x01E0 10FC | - | Reserved for future use |
| 0x01E0 1100 - 0x01E0 1FFF | - | Reserved |

6.20.2 VLYNQ Electrical Data/Timing

Table 6-83. Timing Requirements for VLYNQ_CLK Input (see Figure 6-45)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------|--------------------------------|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 1 | $t_c(VCLK)$ | Cycle time, VLYNQ_CLK | 10 | | ns |
| 2 | $t_w(VCLKH)$ | Pulse duration, VLYNQ_CLK high | 3 | | ns |
| 3 | $t_w(VCLKL)$ | Pulse duration, VLYNQ_CLK low | 3 | | ns |

Table 6-84. Switching Characteristics Over Recommended Operating Conditions for VLYNQ_CLK Output (see Figure 6-45)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------|--------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | $t_c(VCLK)$ | Cycle time, VLYNQ_CLK | 10 | ns |
| 2 | $t_w(VCLKH)$ | Pulse duration, VLYNQ_CLK high | 4 | ns |
| 3 | $t_w(VCLKL)$ | Pulse duration, VLYNQ_CLK low | 4 | ns |

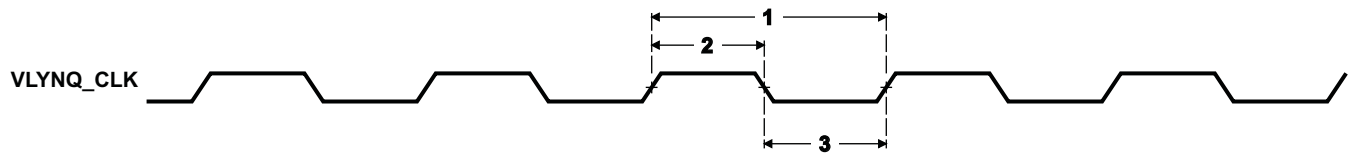


Figure 6-45. VLYNQ_CLK Timing for VLYNQ

Table 6-85. Switching Characteristics Over Recommended Operating Conditions for Transmit Data for the VLYNQ Module (see Figure 6-46)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|--------------------|--|------|------|
| | | MIN | MAX | |
| 1 | $t_d(VCLKH-TXD I)$ | Delay time, VLYNQ_CLK high to VLYNQ_TXD[3:0] invalid | 2.25 | ns |
| 2 | $t_d(VCLKH-TXD V)$ | Delay time, VLYNQ_CLK high to VLYNQ_TXD[3:0] valid | 12 | ns |

Table 6-86. Timing Requirements for Receive Data for the VLYNQ Module⁽¹⁾ (see Figure 6-46)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|----------------------|--|-------------------------------|------|------|
| | | | MIN | MAX | |
| 3 | $t_{su}(RXDV-VCLKH)$ | Setup time, VLYNQ_RXD[3:0] valid before VLYNQ_CLK high | RTM disabled, RTM sample = 3 | 1.75 | ns |
| | | | RTM enabled | (1) | ns |
| 4 | $t_h(VCLKH-RXDV)$ | Hold time, VLYNQ_RXD[3:0] valid after VLYNQ_CLK high | RTM disabled, RTM sample = 3 | 3 | ns |
| | | | RTM enabled | (1) | ns |

(1) The VLYNQ receive timing manager (RTM) is a serial receive logic designed to eliminate setup and hold violations that could occur in traditional input signals. RTM logic automatically selects the setup and hold timing from one of eight data flops (see Table 6-87). When RTM logic is disabled, the setup and hold timing from the default data flop (3) is used.

Table 6-87. RTM RX Data Flop Hold/Setup Timing Constraints (Typical Values)

| RX Data Flop | HOLD (Y) | SETUP (X) |
|--------------|----------|-----------|
| 0 | 1.3 | 0.9 |
| 1 | 1.4 | 0.7 |
| 2 | 1.5 | -0.4 |
| 3 | 1.6 | -0.6 |
| 4 | 1.8 | -0.8 |
| 5 | 2.0 | -1.0 |
| 6 | 2.2 | -1.1 |
| 7 | 2.4 | -1.2 |

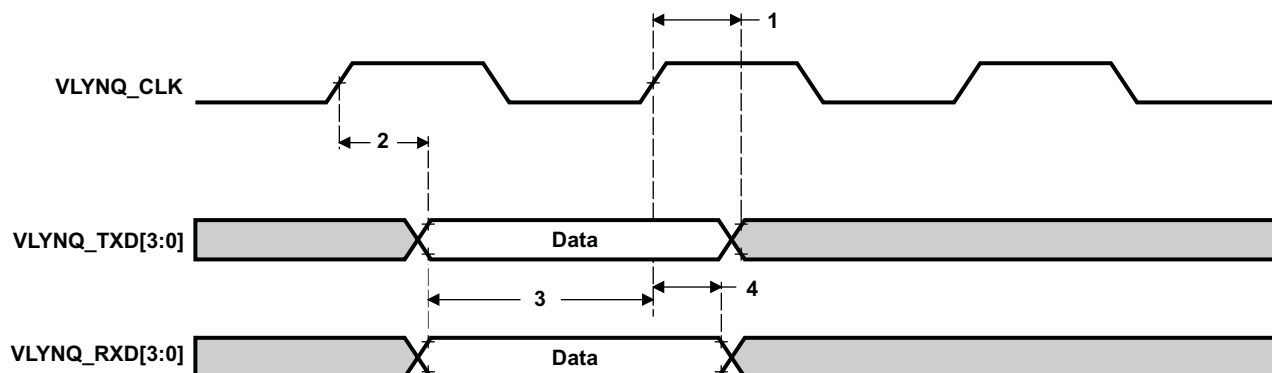


Figure 6-46. VLYNQ Transmit/Receive Timing

6.21 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 GP[0:15]).

The C6424 GPIO peripheral supports the following:

- Up to 111 3.3-V GPIO pins, GP[0:110]
- Interrupts:
 - Up to 8 unique GP[0:7] interrupts from Bank 0
 - 7 GPIO bank (aggregated) interrupt signals from each of the 7 banks of GPIOs
 - Interrupts can be triggered by rising and/or falling edge, specified for each interrupt capable GPIO signal
- DMA events:
 - Up to 8 unique GPIO DMA events from Bank 0
 - 7 GPIO bank (aggregated) DMA event signals from each of the 7 banks of GPIOs
- 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 another 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 6-88](#). For more detailed information on GPIOs, see the *TMS320C642x DSP General-Purpose Input/Output (GPIO) User's Guide* (literature number SPRUEM8).

6.21.1 GPIO Peripheral Register Description(s)

Table 6-88. GPIO Registers

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|----------------|--|
| 0x01C6 7000 | PID | Peripheral Identification Register |
| 0x01C6 7004 | - | Reserved |
| 0x01C6 7008 | BINTEN | GPIO interrupt per-bank enable |
| GPIO Banks 0 and 1 | | |
| 0x01C6 700C | - | Reserved |
| 0x01C6 7010 | DIR01 | GPIO Banks 0 and 1 Direction Register (GP[0:31]) |
| 0x01C6 7014 | OUT_DATA01 | GPIO Banks 0 and 1 Output Data Register (GP[0:31]) |
| 0x01C6 7018 | SET_DATA01 | GPIO Banks 0 and 1 Set Data Register (GP[0:31]) |
| 0x01C6 701C | CLR_DATA01 | GPIO Banks 0 and 1 Clear data for banks 0 and 1 (GP[0:31]) |
| 0x01C6 7020 | IN_DATA01 | GPIO Banks 0 and 1 Input Data Register (GP[0:31]) |
| 0x01C6 7024 | SET_RIS_TRIG01 | GPIO Banks 0 and 1 Set Rising Edge Interrupt Register (GP[0:31]) |
| 0x01C6 7028 | CLR_RIS_TRIG01 | GPIO Banks 0 and 1 Clear Rising Edge Interrupt Register (GP[0:31]) |
| 0x01C6 702C | SET_FAL_TRIG01 | GPIO Banks 0 and 1 Set Falling Edge Interrupt Register (GP[0:31]) |
| 0x01C6 7030 | CLR_FAL_TRIG01 | GPIO Banks 0 and 1 Clear Falling Edge Interrupt Register (GP[0:31]) |
| 0x01C6 7034 | INSTAT01 | GPIO Banks 0 and 1 Interrupt Status Register (GP[0:31]) |
| GPIO Banks 2 and 3 | | |
| 0x01C6 7038 | DIR23 | GPIO Banks 2 and 3 Direction Register (GP[32:63]) |
| 0x01C6 703C | OUT_DATA23 | GPIO Banks 2 and 3 Output Data Register (GP[32:63]) |
| 0x01C6 7040 | SET_DATA23 | GPIO Banks 2 and 3 Set Data Register (GP[32:63]) |
| 0x01C6 7044 | CLR_DATA23 | GPIO Banks 2 and 3 Clear Data Register (GP[32:63]) |
| 0x01C6 7048 | IN_DATA23 | GPIO Banks 2 and 3 Input Data Register (GP[32:63]) |
| 0x01C6 704C | SET_RIS_TRIG23 | GPIO Banks 2 and 3 Set Rising Edge Interrupt Register (GP[32:63]) |
| 0x01C6 7050 | CLR_RIS_TRIG23 | GPIO Banks 2 and 3 Clear Rising Edge Interrupt Register (GP[32:63]) |
| 0x01C6 7054 | SET_FAL_TRIG23 | GPIO Banks 2 and 3 Set Falling Edge Interrupt Register (GP[32:63]) |
| 0x01C6 7058 | CLR_FAL_TRIG23 | GPIO Banks 2 and 3 Clear Falling Edge Interrupt Register (GP[32:63]) |
| 0x01C6 705C | INSTAT23 | GPIO Banks 2 and 3 Interrupt Status Register (GP[32:63]) |
| GPIO Bank 4 and 5 | | |
| 0x01C6 7060 | DIR45 | GPIO Bank 4 and 5 Direction Register (GP[64:95]) |
| 0x01C6 7064 | OUT_DATA45 | GPIO Bank 4 and 5 Output Data Register (GP[64:95]) |
| 0x01C6 7068 | SET_DATA45 | GPIO Bank 4 and 5 Set Data Register (GP[64:95]) |
| 0x01C6 706C | CLR_DATA45 | GPIO Bank 4 and 5 Clear Data Register (GP[64:95]) |
| 0x01C6 7070 | IN_DATA45 | GPIO Bank 4 and 5 Input Data Register (GP[64:95]) |
| 0x01C6 7074 | SET_RIS_TRIG45 | GPIO Bank 4 and 5 Set Rising Edge Interrupt Register (GP[64:95]) |
| 0x01C6 7078 | CLR_RIS_TRIG45 | GPIO Bank 4 and 5 Clear Rising Edge Interrupt Register (GP[64:95]) |
| 0x01C6 707C | SET_FAL_TRIG45 | GPIO Bank 4 and 5 Set Falling Edge Interrupt Register (GP[64:95]) |
| 0x01C6 7080 | CLR_FAL_TRIG45 | GPIO Bank 4 and 5 Clear Falling Edge Interrupt Register (GP[64:95]) |
| 0x01C6 7084 | INSTAT45 | GPIO Bank 4 and 5 Interrupt Status Register (GP[64:95]) |
| GPIO Bank 6 | | |
| 0x01C6 7088 | DIR6 | GPIO Bank 6 Direction Register (GP[96:110]) |
| 0x01C6 708C | OUT_DATA6 | GPIO Bank 6 Output Data Register (GP[96:110]) |
| 0x01C6 7090 | SET_DATA6 | GPIO Bank 6 Set Data Register (GP[96:110]) |
| 0x01C6 7094 | CLR_DATA6 | GPIO Bank 6 Clear Data Register (GP[96:110]) |
| 0x01C6 7098 | IN_DATA6 | GPIO Bank 6 Input Data Register (GP[96:110]) |
| 0x01C6 709C | SET_RIS_TRIG6 | GPIO Bank 6 Set Rising Edge Interrupt Register (GP[96:110]) |
| 0x01C6 70A0 | CLR_RIS_TRIG6 | GPIO Bank 6 Clear Rising Edge Interrupt Register (GP[96:110]) |

Table 6-88. GPIO Registers (continued)

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME |
|---------------------------|---------------|--|
| 0x01C6 70A4 | SET_FAL_TRIG6 | GPIO Bank 6 Set Falling Edge Interrupt Register (GP[96:110]) |
| 0x01C6 70A8 | CLR_FAL_TRIG6 | GPIO Bank 6 Clear Falling Edge Interrupt Register (GP[96:110]) |
| 0x01C6 70AC | INSTAT6 | GPIO Bank 6 Interrupt Status Register (GP[96:110]) |
| 0x01C6 70B0 - 0x01C6 7FFF | - | Reserved |

6.21.2 GPIO Peripheral Input/Output Electrical Data/Timing

Table 6-89. Timing Requirements for GPIO Inputs⁽¹⁾ (see Figure 6-47)

| NO. | | | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---------------|----------------------------------|-------------------------------|-----|------|
| | | | MIN | MAX | |
| 1 | $t_{w(GPIH)}$ | Pulse duration, GP[x] input high | 2C ⁽²⁾ | | ns |
| 2 | $t_{w(GPIL)}$ | Pulse duration, GP[x] input low | 2C ⁽²⁾ | | ns |

- (1) The pulse width given is sufficient to generate a CPU interrupt or an EDMA event. However, if a user wants to have C6424 recognize the GP[x] input changes through software polling of the GPIO register, the GP[x] input duration must be extended to allow C6424 enough time to access the GPIO register through the internal bus.
- (2) C = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use C = 10ns.

Table 6-90. Switching Characteristics Over Recommended Operating Conditions for GPIO Outputs (see Figure 6-47)

| NO. | PARAMETER | -7/-6/-5/-4 -L/-Q6/-Q5/-Q4 | | UNIT |
|-----|---------------|-------------------------------|-----|------|
| | | MIN | MAX | |
| 3 | $t_{w(GPOH)}$ | 2C ^{(1) (2)} | | ns |
| 4 | $t_{w(GPOL)}$ | 2C ^{(1) (2)} | | 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.
- (2) C = SYSCLK3 period in ns. For example, when running parts at 600 MHz, use C = 10ns.

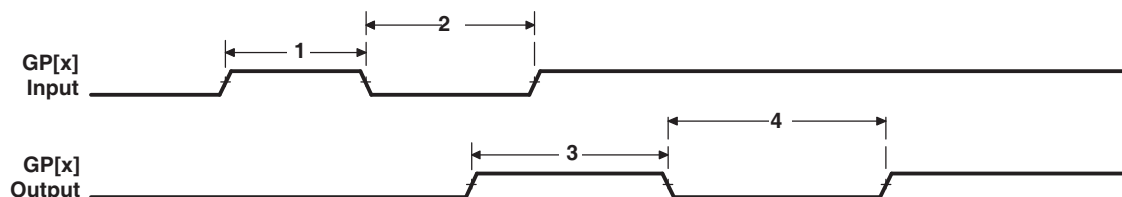


Figure 6-47. GPIO Port Timing

6.22 IEEE 1149.1 JTAG

The JTAG⁽³⁾ interface is used for BSDL testing and emulation of the C6424 device.

$\overline{\text{TRST}}$ only needs to be released 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, C6424 includes an internal pulldown (IPD) on the $\overline{\text{TRST}}$ pin to ensure that $\overline{\text{TRST}}$ will always be asserted upon power up and the 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.

6.22.1 JTAG ID (JTAGID) Register Description(s)

Table 6-91. JTAG ID Register

| HEX ADDRESS RANGE | ACRONYM | REGISTER NAME | COMMENTS |
|-------------------|---------|------------------------------|---|
| 0x01C4 0028 | JTAGID | JTAG Identification Register | Read-only. Provides 32-bit JTAG ID of the device. |

(3) 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. For the C6424 device, the JTAG ID register resides at address location 0x01C4 0028. For the actual register bit names and their associated bit field descriptions, see [Figure 6-48](#) and [Table 6-92](#).

| 31-28 | 27-12 | 11-1 | 0 |
|-----------------|-----------------------|-----------------------|-----|
| VARIANT (4-Bit) | PART NUMBER (16-Bit) | MANUFACTURER (11-Bit) | LSB |
| R-n | R-1011 0111 0010 0001 | R-0000 0010 111 | R-1 |

LEGEND: R = Read, W = Write, n = value at reset

Figure 6-48. JTAG ID (JTAGID) Register—0x01C4 0028

Table 6-92. JTAG ID Register Selection Bit Descriptions

| BIT | NAME | DESCRIPTION |
|-------|--------------|--|
| 31:28 | VARIANT | Variant (4-Bit) value. A read from this field always returns 0b0000. |
| 27:12 | PART NUMBER | Part Number (16-Bit) value. C6424 value: 1011 0111 0010 0001. |
| 11-1 | MANUFACTURER | Manufacturer (11-Bit) value. C6424 value: 0000 0010 111. |
| 0 | LSB | LSB. This bit is read as a "1" for C6424. |

6.22.2 JTAG Electrical Data/Timing

Table 6-93. Timing Requirements for JTAG Test Port (see Figure 6-49)

| NO. | PARAMETER | -7I-6I-5I-4 -LI-Q6I-Q5I-Q4 | | UNIT |
|-----|---|-------------------------------|-----|------|
| | | MIN | MAX | |
| 1 | t_c (TCK) Cycle time, TCK | 33 | | ns |
| 3 | t_{su} (TDIV-TCKH) Setup time, TDI/TMS/TRST valid before TCK high | 2.5 | | ns |
| 4 | t_h (TCKH-TDIV) Hold time, TDI/TMS/TRST valid after TCK high | 16.5 | | ns |

Table 6-94. Switching Characteristics Over Recommended Operating Conditions for JTAG Test Port (see Figure 6-49)

| NO. | PARAMETER | -7I-6I-5I-4 -LI-Q6I-Q5I-Q4 | | UNIT |
|-----|--|-------------------------------|-----|------|
| | | MIN | MAX | |
| 2 | t_d (TCKL-TDOV) Delay time, TCK low to TDO valid | 0 | 14 | ns |

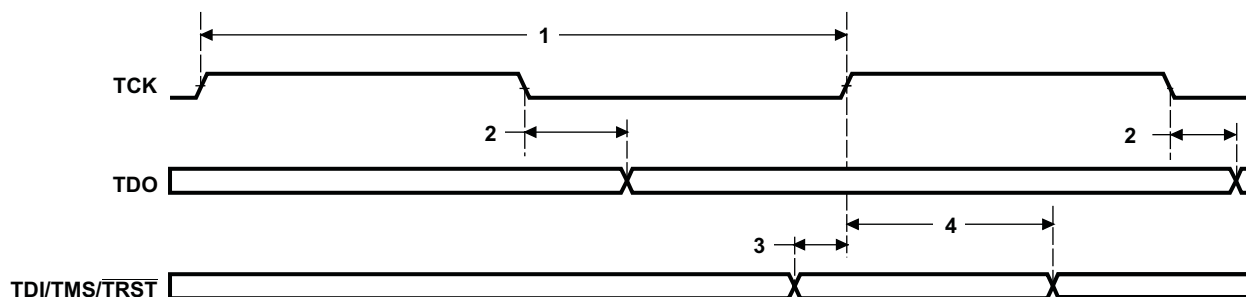


Figure 6-49. JTAG Test-Port Timing

7 Mechanical Data

The following table(s) show the thermal resistance characteristics for the PBGA–ZWT and ZDU mechanical package(s).

7.1 Thermal Data for ZWT

Table 7-1. Thermal Resistance Characteristics (PBGA Package) [ZWT]

| NO. | | | °C/W ⁽¹⁾ | AIR FLOW (m/s) ⁽²⁾ |
|-----|-----------------|-------------------------|---------------------|-------------------------------|
| 1 | R θ_{JC} | Junction-to-case | 5.4 | N/A |
| 2 | R θ_{JB} | Junction-to-board | 16.0 | N/A |
| 3 | | | 26.6 | 0.00 |
| 4 | R θ_{JA} | Junction-to-free air | 21.9 | 1.0 |
| 5 | | | 20.4 | 2.00 |
| 7 | | | 0.0 | 0.00 |
| 8 | Psi $_{JT}$ | Junction-to-package top | 0.1 | 1.0 |
| 9 | | | 0.2 | 2.00 |
| 11 | | | 15.9 | 0.00 |
| 12 | Psi $_{JB}$ | Junction-to-board | 15.8 | 1.0 |
| 13 | | | 15.3 | 2.00 |

(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*
- JESD51-7, *High Effective Thermal Conductivity Test Board for Leaded Surface Mount Packages*

(2) m/s = meters per second

7.2 Thermal Data for ZDU

Table 7-2. Thermal Resistance Characteristics (PBGA Package) [ZDU]

| NO. | | | °C/W ⁽¹⁾ | AIR FLOW (m/s) ⁽²⁾ |
|-----|--------------------------|-------------------------|---------------------|-------------------------------|
| 1 | R θ _{JC} | Junction-to-case | 7.7 | N/A |
| 2 | R θ _{JB} | Junction-to-board | 10.5 | N/A |
| 3 | R θ _{JA} | Junction-to-free air | 19.7 | 0.00 |
| 4 | | | 15.5 | 1.0 |
| 5 | | | 14.3 | 2.00 |
| 7 | Psi _{JT} | Junction-to-package top | 4.9 | 0.00 |
| 8 | | | 5.1 | 1.0 |
| 9 | | | 5.2 | 2.00 |
| 11 | Psi _{JB} | Junction-to-board | 10.4 | 0.00 |
| 12 | | | 9.8 | 1.0 |
| 13 | | | 9.6 | 2.00 |

(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*
- JESD51-7, *High Effective Thermal Conductivity Test Board for Leaded Surface Mount Packages*

(2) m/s = meters per second

7.3 Packaging Information

The following packaging information and addendum reflect 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/Ball Finish (6) | MSL Peak Temp (3) | Op Temp (°C) | Device Marking (4/5) | Samples |
|------------------|---------------|--------------|--------------------|------|----------------|----------------------------|-------------------------|----------------------|--------------|--------------------------------|-------------------------|
| TMS320C6424ZDU4 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZDU TMS320 4 | Samples |
| TMS320C6424ZDU5 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZDU TMS320 5 | Samples |
| TMS320C6424ZDU6 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | C6424ZDU6 | Samples |
| TMS320C6424ZDU7 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZDU TMS320 7 | Samples |
| TMS320C6424ZDUL | ACTIVE | BGA | ZDU | 376 | | TBD | Call TI | Call TI | 0 to 90 | C6424ZDUL | Samples |
| TMS320C6424ZDUQ5 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | -40 to 125 | L1 C6424ZDUQ TMS320 | Samples |
| TMS320C6424ZDUQ6 | ACTIVE | BGA | ZDU | 376 | 60 | Green (RoHS & no Sb/Br) | SNAGCU | Level-3-260C-168 HR | -40 to 125 | L1 C6424ZDUQ TMS320 6 | Samples |
| TMS320C6424ZWT4 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZWT TMS320 4 | Samples |
| TMS320C6424ZWT5 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZWT TMS320 5 | Samples |
| TMS320C6424ZWT6 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZWT TMS320 | Samples |
| TMS320C6424ZWT7 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | 0 to 90 | L2 C6424ZWT TMS320 | Samples |

| Orderable Device | Status (1) | Package Type | Package Drawing | Pins | Package Qty | Eco Plan (2) | Lead/Ball Finish (6) | MSL Peak Temp (3) | Op Temp (°C) | Device Marking (4/5) | Samples |
|------------------|---------------|--------------|-----------------|------|-------------|-----------------|-------------------------|----------------------|--------------|--------------------------------|----------------|
| | | | | | | | | | | 7 | |
| TMS320C6424ZWTL | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | | L2 C6424ZWTL TMS320 | Samples |
| TMS320C6424ZWTQ5 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | -40 to 125 | L1 C6424ZWTQ TMS320 5 | Samples |
| TMS320C6424ZWTQ6 | ACTIVE | NFBGA | ZWT | 361 | 90 | Pb-Free (RoHS) | SNAGCU | Level-3-260C-168 HR | -40 to 125 | L1 C6424ZWTQ TMS320 6 | Samples |
| TMX320C6424AZWT | OBSOLETE | NFBGA | ZWT | 361 | | TBD | Call TI | Call TI | -40 to 125 | 320C6424A ZWT TMX | |
| TMX320C6424CZDU | OBSOLETE | BGA | ZDU | 376 | | TBD | Call TI | Call TI | 0 to 90 | | |

(1) The marketing status values are defined as follows:

ACTIVE: Product device recommended for new designs.

LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

PREVIEW: Device has been announced but is not in production. Samples may or may not be available.

OBSOLETE: TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check <http://www.ti.com/productcontent> for the latest availability information and additional product content details.

TBD: The Pb-Free/Green conversion plan has not been defined.

Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.

Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.

Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

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

(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.

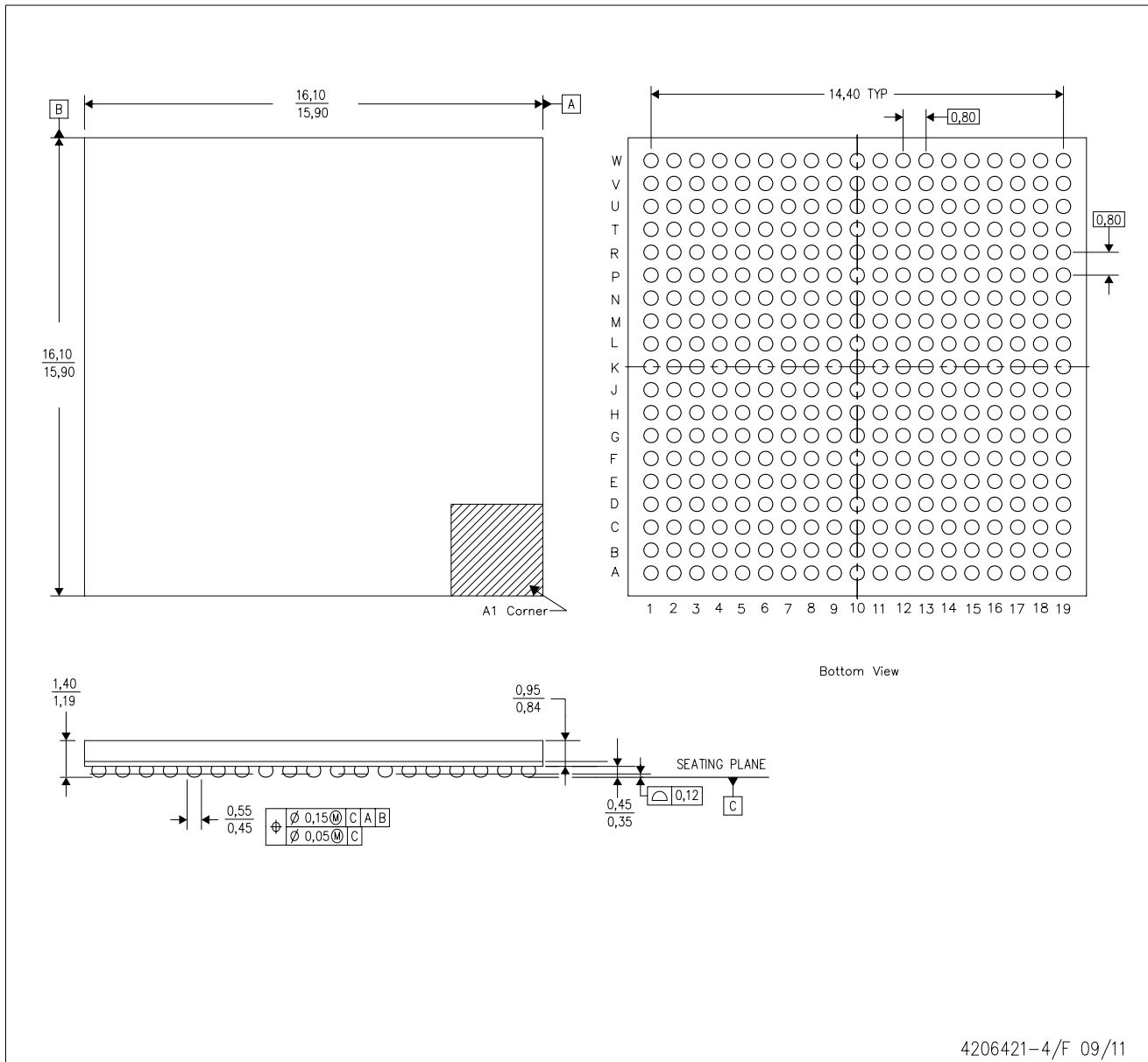
⁽⁶⁾ Lead/Ball Finish - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead/Ball Finish values may wrap to two lines if the finish value exceeds the maximum column width.

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

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

ZWT (S-PBGA-N361)

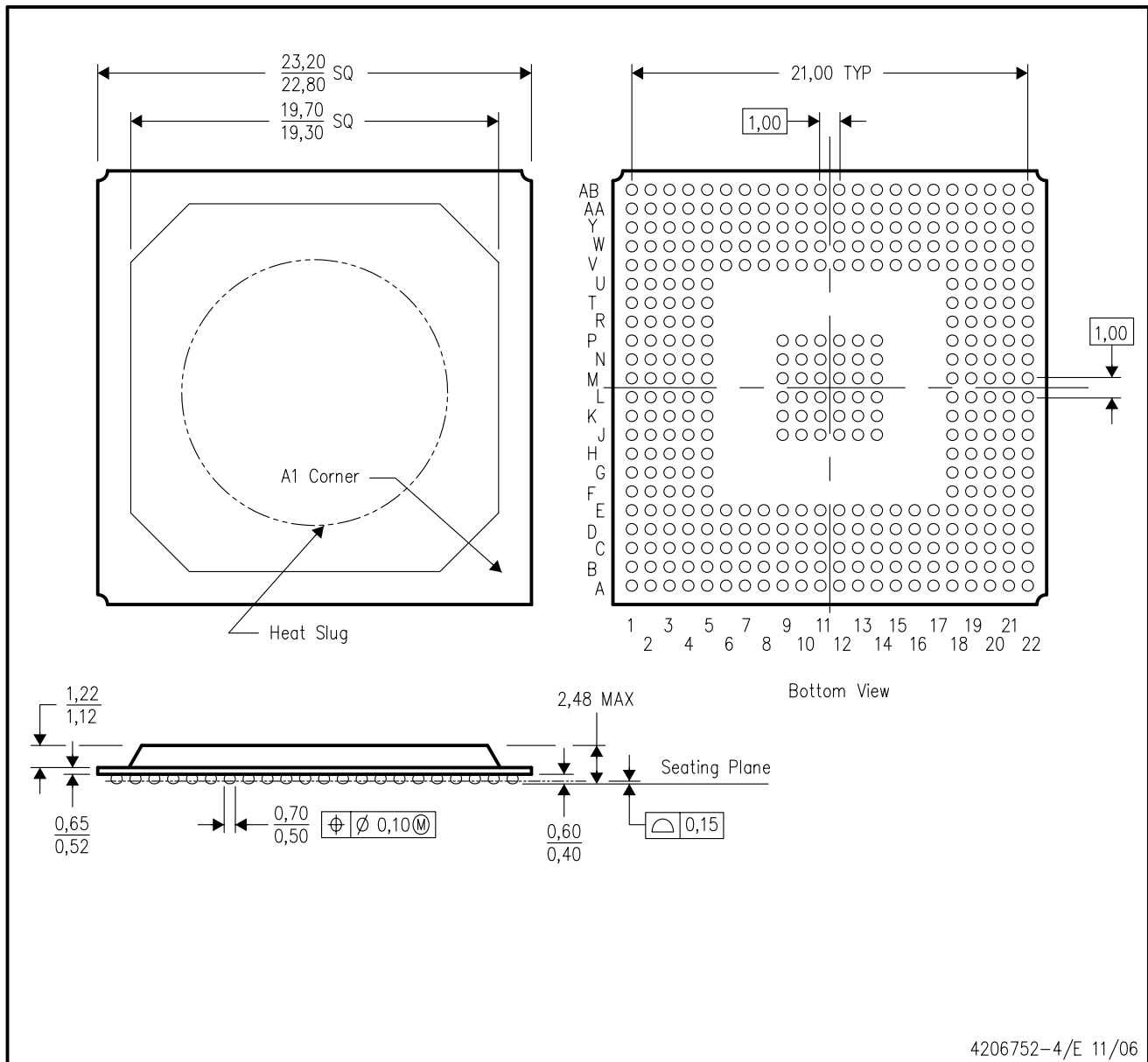
PLASTIC BALL GRID ARRAY



- NOTES:
- A. All linear dimensions are in millimeters. Dimensioning and tolerancing per ASME Y14.5M-1994.
 - B. This drawing is subject to change without notice.
 - C. This is a Pb-free solder ball design.
 - D. Falls within JEDEC MO-275.

ZDU (S-PBGA-N376)

PLASTIC BALL GRID ARRAY



- NOTES:
- A. All linear dimensions are in millimeters.
 - B. This drawing is subject to change without notice.
 - C. Falls within JEDEC MO-151
 - D. Thermally enhanced molded plastic package with heat slug (HSL).
 - E. This is a lead-free solder ball design.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

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

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

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

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

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

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

| | |
|------------------------------|--|
| Audio | www.ti.com/audio |
| Amplifiers | amplifier.ti.com |
| Data Converters | dataconverter.ti.com |
| DLP® Products | www.dlp.com |
| DSP | dsp.ti.com |
| Clocks and Timers | www.ti.com/clocks |
| Interface | interface.ti.com |
| Logic | logic.ti.com |
| Power Mgmt | power.ti.com |
| Microcontrollers | microcontroller.ti.com |
| RFID | www.ti-rfid.com |
| OMAP Applications Processors | www.ti.com/omap |
| Wireless Connectivity | www.ti.com/wirelessconnectivity |

Applications

| | |
|-------------------------------|--|
| Automotive and Transportation | www.ti.com/automotive |
| Communications and Telecom | www.ti.com/communications |
| Computers and Peripherals | www.ti.com/computers |
| Consumer Electronics | www.ti.com/consumer-apps |
| Energy and Lighting | www.ti.com/energy |
| Industrial | www.ti.com/industrial |
| Medical | www.ti.com/medical |
| Security | www.ti.com/security |
| Space, Avionics and Defense | www.ti.com/space-avionics-defense |
| Video and Imaging | www.ti.com/video |

TI E2E Community

e2e.ti.com