



## AM1705 ARM® Microprocessor

### 1 Device Overview

#### 1.1 Features

- 375- and 456-MHz ARM926EJ-S™ RISC Core
  - 32-Bit and 16-Bit (Thumb®) Instructions
  - Single-Cycle MAC
  - ARM Jazelle® Technology
  - Embedded ICE-RT™ for Real-Time Debug
- ARM9™ Memory Architecture
  - 16KB of Instruction Cache
  - 16KB of Data Cache
  - 8KB of RAM (Vector Table)
  - 64KB of ROM
- Enhanced Direct Memory Access Controller 3 (EDMA3):
  - 2 Transfer Controllers
  - 32 Independent DMA Channels
  - 8 Quick DMA Channels
  - Programmable Transfer Burst Size
- 128KB of RAM Memory
- 3.3-V LVCMOS I/Os (Except for USB Interface)
- Two External Memory Interfaces:
  - EMIFA
    - NOR (8-Bit-Wide Data)
    - NAND (8-Bit-Wide Data)
  - EMIFB
    - 16-Bit SDRAM With 128-MB Address Space
- Three Configurable 16550-Type UART Modules:
  - UART0 With Modem Control Signals
  - 16-Byte FIFO
  - 16x or 13x Oversampling Option
  - Autoflow Control Signals (CTS, RTS) on UART0 Only
- Two Serial Peripheral Interfaces (SPIs) Each With One Chip Select
- Programmable Real-Time Unit Subsystem (PRUSS)
  - Two Independent Programmable Real-Time Unit (PRU) Cores
    - 32-Bit Load-Store RISC Architecture
    - 4KB of Instruction RAM per Core
    - 512 Bytes of Data RAM per Core
    - PRUSS can be Disabled Through Software to Save Power
  - Standard Power-Management Mechanism
    - Clock Gating
    - Entire Subsystem Under a Single PSC Clock Gating Domain
- Dedicated Interrupt Controller
- Dedicated Switched Central Resource
- Multimedia Card (MMC)/Secure Digital (SD) Card Interface With Secure Data I/O (SDIO)
- Two Master and Slave Inter-Integrated Circuit (I<sup>2</sup>C Bus™)
- USB 2.0 OTG Port With Integrated PHY (USB0)
  - USB 2.0 Full-Speed Client
  - USB 2.0 Full- and Low-Speed Host
  - End Point 0 (Control)
  - End Points 1, 2, 3, and 4 (Control, Bulk, Interrupt, or ISOC) RX and TX
- Two Multichannel Audio Serial Ports (McASPs):
  - Six Clock Zones and 28 Serial Data Pins
  - Supports TDM, I2S, and Similar Formats
  - FIFO Buffers for Transmit and Receive
- 10/100 Mbps Ethernet MAC (EMAC):
  - IEEE 802.3 Compliant (3.3-V I/O Only)
  - RMI Media-Independent Interface
  - Management Data I/O (MDIO) Module
- One 64-Bit General-Purpose Timer (Configurable as Two 32-Bit Timers)
- One 64-Bit General-Purpose Watchdog Timer (Configurable as Two 32-Bit General-Purpose Timers)
- Three Enhanced Pulse Width Modulators (eHRPWMs):
  - Dedicated 16-Bit Time-Base Counter With Period and Frequency Control
  - 6 Single-Edge, 6 Dual-Edge Symmetric, or 3 Dual-Edge Asymmetric Outputs
  - Dead-Band Generation
  - PWM Chopping by High-Frequency Carrier
  - Trip Zone Input
- Three 32-Bit Enhanced Capture (eCAP) Modules:
  - Configurable as 3 Capture Inputs or 3 Auxiliary Pulse Width Modulator (APWM) Outputs
  - Single-Shot Capture of up to Four Event Timestamps
- Two 32-Bit Enhanced Quadrature Encoder Pulse (eQEP) Modules
- 176-Pin PowerPAD™ Plastic Quad Flat Pack [PTP suffix], 0.5-mm Pin Pitch
- Commercial, Industrial, or Extended Temperature



## 1.2 Applications

- Industrial Automation
- Home Automation
- Test and Measurement
- Portable Data Terminals

## 1.3 Description

The AM1705 is a low-power ARM microprocessor based on an ARM926EJ-S.

The device enables original-equipment manufacturers (OEMs) and original-design manufacturers (ODMs) to quickly bring to market devices with robust operating systems, rich user interfaces, and high processor performance through the maximum flexibility of a fully integrated, mixed processor solution.

The ARM926EJ-S is a 32-bit RISC processor core that performs 32-bit or 16-bit instructions and processes 32-, 16-, or 8-bit data. The core uses pipelining so that all parts of the processor and memory system can operate continuously.

The ARM core has a coprocessor 15 (CP15), protection module, and data and program memory management units (MMUs) with table look-aside buffers. The ARM core has separate 16KB of instruction and 16-KB data caches. Both memory blocks are 4-way associative with virtual index virtual tag (VIVT). The ARM core also has 8KB of RAM (Vector Table) and 64KB of ROM.

The peripheral set includes: a 10/100 Mbps Ethernet MAC (EMAC) with a management data input/output (MDIO) module; two I<sup>2</sup>C Bus interfaces; three multichannel audio serial ports (McASPs) with serializers and FIFO buffers; two 64-bit general-purpose timers each configurable (one configurable as watchdog); up to 8 banks of 16 pins of general-purpose input/output (GPIO) with programmable interrupt/event generation modes, multiplexed with other peripherals; three UART interfaces (one with both  $\overline{RTS}$  and  $\overline{CTS}$ ); three enhanced high-resolution pulse width modulator (eHRPWM) peripherals; three 32-bit enhanced capture (eCAP) module peripherals which can be configured as 3 capture inputs or 3 auxiliary pulse width modulator (APWM) outputs; two 32-bit enhanced quadrature encoded pulse (eQEP) peripherals; and 2 external memory interfaces: an asynchronous and SDRAM external memory interface (EMIFA) for slower memories or peripherals, and a higher speed memory interface (EMIFB) for SDRAM.

The Ethernet Media Access Controller (EMAC) provides an efficient interface between the device and the network. The EMAC supports both 10Base-T and 100Base-TX, or 10 Mbps and 100 Mbps in either half- or full-duplex mode. Additionally, an MDIO interface is available for PHY configuration.

The I<sup>2</sup>C, SPI, and USB2.0 ports allow the device 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 peripheral, see the related sections later in this document and the associated peripheral reference guides.

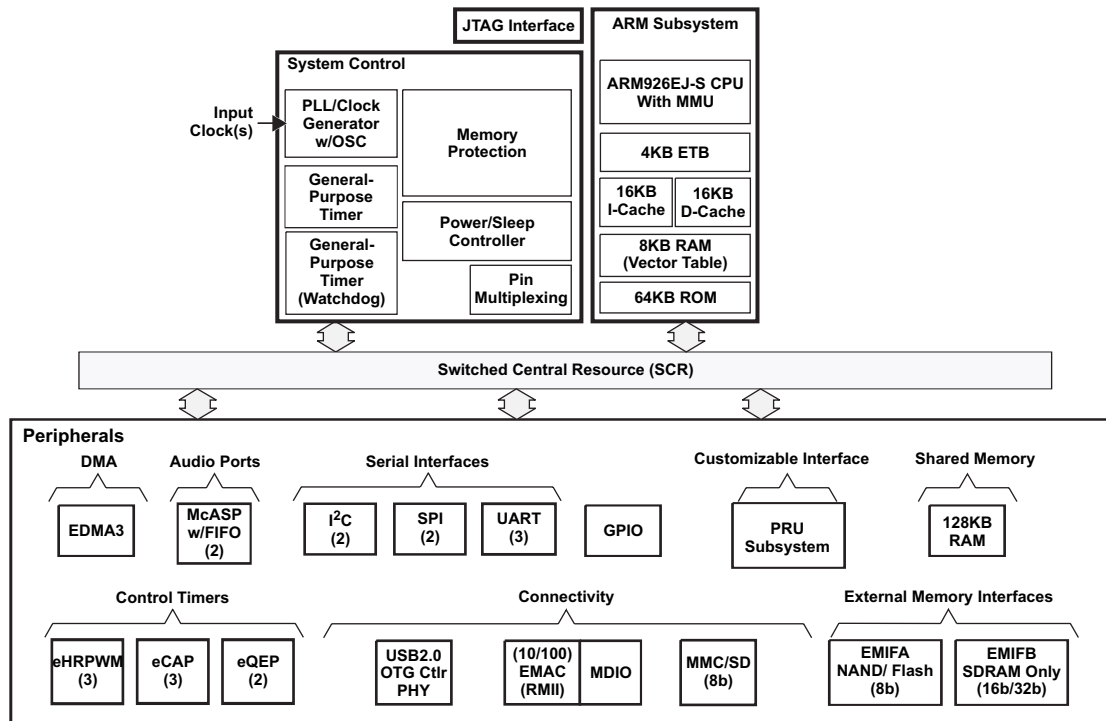
The device has a complete set of development tools for the ARM processor. These tools include C compilers and a Windows® debugger interface for visibility into source code execution.

**Device Information<sup>(1)</sup>**

| PART NUMBER | PACKAGE     | BODY SIZE           |
|-------------|-------------|---------------------|
| AM1705      | HLQFP (176) | 24.00 mm × 24.00 mm |

(1) For more information on these devices, see [Section 8](#).

## 1.4 Functional Block Diagram



**Figure 1-1. AM1705 Functional Block Diagram**

## Table of Contents

|                                                                                                                                      |                           |                                                                       |                            |
|--------------------------------------------------------------------------------------------------------------------------------------|---------------------------|-----------------------------------------------------------------------|----------------------------|
| <b>1 Device Overview</b>                                                                                                             | <b><a href="#">1</a></b>  | 6.10 External Memory Interface A (EMIFA)                              | <a href="#">60</a>         |
| 1.1 Features                                                                                                                         | <a href="#">1</a>         | 6.11 External Memory Interface B (EMIFB)                              | <a href="#">66</a>         |
| 1.2 Applications                                                                                                                     | <a href="#">2</a>         | 6.12 Memory Protection Units                                          | <a href="#">73</a>         |
| 1.3 Description                                                                                                                      | <a href="#">2</a>         | 6.13 MMC / SD / SDIO (MMCSD)                                          | <a href="#">76</a>         |
| 1.4 Functional Block Diagram                                                                                                         | <a href="#">3</a>         | 6.14 Ethernet Media Access Controller (EMAC)                          | <a href="#">79</a>         |
| <b>2 Revision History</b>                                                                                                            | <b><a href="#">5</a></b>  | 6.15 Management Data Input/Output (MDIO)                              | <a href="#">84</a>         |
| <b>3 Device Comparison</b>                                                                                                           | <b><a href="#">6</a></b>  | 6.16 Multichannel Audio Serial Ports (McASP0, McASP1)                 | <a href="#">86</a>         |
| 3.1 Device Characteristics                                                                                                           | <a href="#">6</a>         | 6.17 Serial Peripheral Interface Ports (SPI0, SPI1)                   | <a href="#">97</a>         |
| 3.2 Device Compatibility                                                                                                             | <a href="#">7</a>         | 6.18 Enhanced Capture (eCAP) Peripheral                               | <a href="#">115</a>        |
| 3.3 ARM Subsystem                                                                                                                    | <a href="#">7</a>         | 6.19 Enhanced Quadrature Encoder (eQEP) Peripheral                    | <a href="#">118</a>        |
| 3.4 Memory Map Summary                                                                                                               | <a href="#">10</a>        | 6.20 Enhanced High-Resolution Pulse-Width Modulator (eHRPWM)          | <a href="#">120</a>        |
| 3.5 Pin Assignments                                                                                                                  | <a href="#">12</a>        | 6.21 Timers                                                           | <a href="#">124</a>        |
| 3.6 Terminal Functions                                                                                                               | <a href="#">13</a>        | 6.22 Inter-Integrated Circuit Serial Ports (I2C0, I2C1)               | <a href="#">126</a>        |
| <b>4 Device Configuration</b>                                                                                                        | <b><a href="#">29</a></b> | 6.23 Universal Asynchronous Receiver/Transmitter (UART)               | <a href="#">131</a>        |
| 4.1 Boot Modes                                                                                                                       | <a href="#">29</a>        | 6.24 USB0 OTG (USB2.0 OTG)                                            | <a href="#">133</a>        |
| 4.2 SYSCFG Module                                                                                                                    | <a href="#">30</a>        | 6.25 Power and Sleep Controller (PSC)                                 | <a href="#">141</a>        |
| 4.3 Pullup/Pulldown Resistors                                                                                                        | <a href="#">32</a>        | 6.26 Programmable Real-Time Unit Subsystem (PRUSS)                    | <a href="#">144</a>        |
| <b>5 Device Operating Conditions</b>                                                                                                 | <b><a href="#">33</a></b> | 6.27 Emulation Logic                                                  | <a href="#">147</a>        |
| 5.1 Absolute Maximum Ratings Over Operating Junction Temperature Range (Unless Otherwise Noted)                                      | <a href="#">33</a>        | 6.28 IEEE 1149.1 JTAG                                                 | <a href="#">153</a>        |
| 5.2 Handling Ratings                                                                                                                 | <a href="#">33</a>        | <b>7 Device and Documentation Support</b>                             | <b><a href="#">155</a></b> |
| 5.3 Recommended Operating Conditions                                                                                                 | <a href="#">34</a>        | 7.1 Device Nomenclature                                               | <a href="#">155</a>        |
| 5.4 Notes on Recommended Power-On Hours (POH)                                                                                        | <a href="#">35</a>        | 7.2 Tools and Software                                                | <a href="#">156</a>        |
| 5.5 Electrical Characteristics Over Recommended Ranges of Supply Voltage and Operating Junction Temperature (Unless Otherwise Noted) | <a href="#">36</a>        | 7.3 Documentation Support                                             | <a href="#">156</a>        |
| <b>6 Peripheral Information and Electrical Specifications</b>                                                                        | <b><a href="#">37</a></b> | 7.4 Community Resources                                               | <a href="#">156</a>        |
| 6.1 Parameter Information                                                                                                            | <a href="#">37</a>        | 7.5 Trademarks                                                        | <a href="#">156</a>        |
| 6.2 Recommended Clock and Control Signal Transition Behavior                                                                         | <a href="#">38</a>        | 7.6 Electrostatic Discharge Caution                                   | <a href="#">156</a>        |
| 6.3 Power Supplies                                                                                                                   | <a href="#">38</a>        | 7.7 Export Control Notice                                             | <a href="#">157</a>        |
| 6.4 Reset                                                                                                                            | <a href="#">39</a>        | 7.8 Glossary                                                          | <a href="#">157</a>        |
| 6.5 Crystal Oscillator or External Clock Input                                                                                       | <a href="#">42</a>        | <b>8 Mechanical Packaging and Orderable Information</b>               | <b><a href="#">158</a></b> |
| 6.6 Clock PLLs                                                                                                                       | <a href="#">44</a>        | 8.1 Thermal Data for PTP                                              | <a href="#">158</a>        |
| 6.7 Interrupts                                                                                                                       | <a href="#">48</a>        | 8.2 Supplementary Information About the 176-pin PTP PowerPAD™ Package | <a href="#">158</a>        |
| 6.8 General-Purpose Input/Output (GPIO)                                                                                              | <a href="#">52</a>        | 8.3 Packaging Information                                             | <a href="#">159</a>        |
| 6.9 EDMA                                                                                                                             | <a href="#">55</a>        |                                                                       |                            |

## 2 Revision History

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

| Changes from June 15, 2014 to January 15, 2017                                                                                                                                                                                                                                                                                                    | Page                                     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------|
| <ul style="list-style-type: none"> <li>Updated/Changed the following registers to Reserved in <a href="#">Table 6-17</a>: 0x6800 0018, 0x6800 001C, 0x6800 0078, 0x6800 007C.....</li> <li>Updated/Changed <a href="#">Figure 6-13</a>, <a href="#">Figure 6-14</a>, <a href="#">Figure 6-15</a>, and <a href="#">Figure 6-16</a>.....</li> </ul> | <a href="#">61</a><br><a href="#">64</a> |

## 3 Device Comparison

### 3.1 Device Characteristics

Table 3-1 provides an overview of the device. The table shows significant features of the device, including the capacity of on-chip RAM, peripherals, and the package type with pin count.

**Table 3-1. Characteristics of the Device**

| HARDWARE FEATURES                                                                                                                    |                                                                         | AM1705                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Peripherals<br><br>Not all peripherals pins are available at the same time (for more detail, see the Device Configurations section). | EMIFB                                                                   | 16-bit, up to 128 MB SDRAM                                                                                                    |
|                                                                                                                                      | EMIFA                                                                   | Asynchronous (8-bit bus width) RAM, Flash, NOR, NAND                                                                          |
|                                                                                                                                      | Flash Card Interface                                                    | MMC and SD cards supported                                                                                                    |
|                                                                                                                                      | EDMA3                                                                   | 32 independent channels, 8 QDMA channels, 2 Transfer controllers                                                              |
|                                                                                                                                      | Timers                                                                  | 2 64-Bit General Purpose (configurable as 2 separate 32-bit timers, 1 configurable as Watch Dog)                              |
|                                                                                                                                      | UART                                                                    | 3 (one with RTS and CTS flow control)                                                                                         |
|                                                                                                                                      | SPI                                                                     | 2 (Each with one hardware chip select)                                                                                        |
|                                                                                                                                      | I <sup>2</sup> C                                                        | 2 (both Master/Slave)                                                                                                         |
|                                                                                                                                      | Multichannel Audio Serial Port [McASP]                                  | 2(each with transmit/receive, FIFO buffer, 16/12/4 serializers)                                                               |
|                                                                                                                                      | 10/100 Ethernet MAC with Management Data I/O                            | 1 (RMII Interface)                                                                                                            |
|                                                                                                                                      | eHRPWM                                                                  | 6 Single Edge, 6 Dual Edge Symmetric, or 3 Dual Edge Asymmetric Outputs                                                       |
|                                                                                                                                      | eCAP                                                                    | 3 32-bit capture inputs or 3 32-bit auxiliary PWM outputs                                                                     |
|                                                                                                                                      | eQEP                                                                    | 2 32-bit QEP channels with 4 inputs/channel                                                                                   |
|                                                                                                                                      | USB 2.0 (USB0)                                                          | Full-Speed/Low-Speed OTG Controller with on-chip OTG PHY                                                                      |
|                                                                                                                                      | General-Purpose Input/Output Port                                       | 8 banks of 16-bit                                                                                                             |
| On-Chip Memory                                                                                                                       | PRU Subsystem (PRUSS)                                                   | 2 Programmable PRU Cores                                                                                                      |
|                                                                                                                                      | Size (Bytes)                                                            | 168KB RAM, 64KB ROM                                                                                                           |
|                                                                                                                                      | Organization                                                            | <b>ARM</b><br>16KB I-Cache<br>16KB D-Cache<br>8KB RAM (Vector Table)<br>64KB ROM<br><br><b>ADDITIONAL MEMORY</b><br>128KB RAM |
|                                                                                                                                      |                                                                         |                                                                                                                               |
| JTAG BSDL_ID                                                                                                                         | DEVIDR0 register                                                        | 0x8B7D F02F (Silicon Revision 1.1)<br>0x9B7D F02F (Silicon Revisions 3.0, 2.1, and 2.0)                                       |
| CPU Frequency                                                                                                                        | MHz                                                                     | ARM926 375 MHz (1.2V) or 456 MHz (1.3V)                                                                                       |
| Voltage                                                                                                                              | Core (V)                                                                | 1.2 V nominal for 375 MHz version<br>1.3 V nominal for 456 MHz version                                                        |
|                                                                                                                                      | I/O (V)                                                                 | 3.3 V                                                                                                                         |
| Package                                                                                                                              |                                                                         | 24 mm x 24 mm, 176-Pin, 0.5 mm pitch, TQFP (PTP)                                                                              |
| Product Status <sup>(1)</sup>                                                                                                        | Product Preview (PP), Advance Information (AI), or Production Data (PD) | 375 MHz Versions - PD<br>456 MHz Version - PD                                                                                 |

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

## 3.2 Device Compatibility

The ARM926EJ-S RISC CPU is compatible with other ARM9 CPUs from ARM Holdings plc.

## 3.3 ARM Subsystem

The ARM Subsystem includes the following features:

- ARM926EJ-S RISC processor
- ARMv5TEJ (32/16-bit) instruction set
- Little endian
- System Control Co-Processor 15 (CP15)
- MMU
- 16KB Instruction cache
- 16KB Data cache
- Write Buffer
- Embedded Trace Module and Embedded Trace Buffer (ETM/ETB)
- ARM Interrupt controller

### 3.3.1 ARM926EJ-S RISC CPU

The ARM Subsystem integrates the ARM926EJ-S processor. The ARM926EJ-S processor is a member of ARM9 family of general-purpose microprocessors. This processor is targeted at multi-tasking applications where full memory management, high performance, low die size, and low power are all important. The ARM926EJ-S processor supports the 32-bit ARM and 16 bit THUMB instruction sets, enabling the user to trade off between high performance and high code density. Specifically, the ARM926EJ-S processor supports the ARMv5TEJ instruction set, which includes features for efficient execution of Java byte codes, providing Java performance similar to Just in Time (JIT) Java interpreter, but without associated code overhead.

The ARM926EJ-S processor supports the ARM debug architecture and includes logic to assist in both hardware and software debug. The ARM926EJ-S processor has a Harvard architecture and provides a complete high performance subsystem, including:

- ARM926EJ -S integer core
- CP15 system control coprocessor
- Memory Management Unit (MMU)
- Separate instruction and data caches
- Write buffer
- Separate instruction and data (internal RAM) interfaces
- Separate instruction and data AHB bus interfaces
- Embedded Trace Module and Embedded Trace Buffer (ETM/ETB)

For more complete details on the ARM9, refer to the ARM926EJ-S Technical Reference Manual, available at <http://www.arm.com>

### 3.3.2 CP15

The ARM926EJ-S system control coprocessor (CP15) is used to configure and control instruction and data caches, Memory Management Unit (MMU), and other ARM subsystem functions. The CP15 registers are programmed using the MRC and MCR ARM instructions, when the ARM in a privileged mode such as supervisor or system mode.

### 3.3.3 MMU

A single set of two level page tables stored in main memory is used to control the address translation, permission checks and memory region attributes for both data and instruction accesses. The MMU uses a single unified Translation Lookaside Buffer (TLB) to cache the information held in the page tables. The MMU features are:

- Standard ARM architecture v4 and v5 MMU mapping sizes, domains and access protection scheme.
- Mapping sizes are:
  - 1MB (sections)
  - 64KB (large pages)
  - 4KB (small pages)
  - 1KB (tiny pages)
- Access permissions for large pages and small pages can be specified separately for each quarter of the page (subpage permissions)
- Hardware page table walks
- Invalidate entire TLB, using CP15 register 8
- Invalidate TLB entry, selected by MVA, using CP15 register 8
- Lockdown of TLB entries, using CP15 register 10

### 3.3.4 Caches and Write Buffer

The size of the Instruction cache is 16KB, Data cache is 16KB. Additionally, the caches have the following features:

- Virtual index, virtual tag, and addressed using the Modified Virtual Address (MVA)
- Four-way set associative, with a cache line length of eight words per line (32-bytes per line) and with two dirty bits in the Dcache
- Dcache supports write-through and write-back (or copy back) cache operation, selected by memory region using the C and B bits in the MMU translation tables
- Critical-word first cache refilling
- Cache lockdown registers enable control over which cache ways are used for allocation on a line fill, providing a mechanism for both lockdown, and controlling cache corruption
- Dcache stores the Physical Address TAG (PA TAG) corresponding to each Dcache entry in the TAG RAM for use during the cache line write-backs, in addition to the Virtual Address TAG stored in the TAG RAM. This means that the MMU is not involved in Dcache write-back operations, removing the possibility of TLB misses related to the write-back address.
- Cache maintenance operations provide efficient invalidation of, the entire Dcache or Icache, regions of the Dcache or Icache, and regions of virtual memory.

The write buffer is used for all writes to a noncachable bufferable region, write-through region and write misses to a write-back region. A separate buffer is incorporated in the Dcache for holding write-back for cache line evictions or cleaning of dirty cache lines. The main write buffer has 16-word data buffer and a four-address buffer. The Dcache write-back has eight data word entries and a single address entry.

### 3.3.5 Advanced High-Performance Bus (AHB)

The ARM Subsystem uses the AHB port of the ARM926EJ-S to connect the ARM to the Config bus and the external memories. Arbiters are employed to arbitrate access to the separate D-AHB and I-AHB by the Config Bus and the external memories bus.

### 3.3.6 Embedded Trace Macrocell (ETM) and Embedded Trace Buffer (ETB)

To support real-time trace, the ARM926EJ-S processor provides an interface to enable connection of an Embedded Trace Macrocell (ETM). The ARM926EJ-S Subsystem in the device also includes the Embedded Trace Buffer (ETB). The ETM consists of two parts:

- Trace Port provides real-time trace capability for the ARM9.
- Triggering facilities provide trigger resources, which include address and data comparators, counter, and sequencers.

The device trace port is not pinned out and is instead only connected to the Embedded Trace Buffer. The ETB has a 4KB buffer memory. ETB enabled debug tools are required to read/interpret the captured trace data.

This device uses ETM9™ version r2p2 and ETB version r0p1. Documentation on the ETM and ETB is available from ARM Ltd. Reference the 'CoreSight™ ETM9™ Technical Reference Manual, revision r0p1' and the 'ETM9 Technical Reference Manual, revision r2p2'.

### **3.3.7 ARM Memory Mapping**

By default the ARM has access to most on and off chip memory areas, EMIFA, EMIFB, and the additional 128K byte on chip SRAM. Likewise almost all of the on chip peripherals are accessible to the ARM by default.

To improve security and/or robustness, the device has extensive memory and peripheral protection units which can be configured to limit access rights to the various on/off chip resources to specific hosts; including the ARM as well as other master peripherals. This allows the system tasks to be partitioned between the ARM and DSP as best suites the particular application; while enhancing the overall robustness of the solution.

See [Table 3-2](#) for a detailed top level device memory map that includes the ARM memory space.

### 3.4 Memory Map Summary

**Table 3-2. AM1705 Top Level Memory Map**

| Start Address | End Address | Size | ARM Mem Map                      | EDMA Mem Map | PRUSS Mem Map             | Master Peripheral Mem Map |
|---------------|-------------|------|----------------------------------|--------------|---------------------------|---------------------------|
| 0x0000 0000   | 0x0000 0FFF | 4K   | -                                |              | PRUSS Local Address Space |                           |
| 0x0000 1000   | 0x01BB FFFF |      | -                                |              |                           |                           |
| 0x01BC 0000   | 0x01BC 0FFF | 4K   | ARM ETB memory                   | -            |                           |                           |
| 0x01BC 1000   | 0x01BC 17FF | 2K   | ARM ETB reg                      | -            |                           |                           |
| 0x01BC 1800   | 0x01BC 18FF | 256  | ARM Ice Crusher                  | -            |                           |                           |
| 0x01BC 1900   | 0x01BF FFFF |      | -                                |              |                           |                           |
| 0x01C0 0000   | 0x01C0 7FFF | 32K  | EDMA3 Channel Controller         |              |                           |                           |
| 0x01C0 8000   | 0x01C0 83FF | 1024 | EDMA3 Transfer Controller 0      |              |                           |                           |
| 0x01C0 8400   | 0x01C0 87FF | 1024 | EDMA3 Transfer Controller 1      |              |                           |                           |
| 0x01C0 8800   | 0x01C0 FFFF |      | -                                |              |                           |                           |
| 0x01C1 0000   | 0x01C1 0FFF | 4K   | PSC 0                            |              |                           |                           |
| 0x01C1 1000   | 0x01C1 1FFF | 4K   | PLL Controller                   |              |                           |                           |
| 0x01C1 2000   | 0x01C1 3FFF |      | -                                |              |                           |                           |
| 0x01C1 4000   | 0x01C1 4FFF | 4K   | SYSCFG                           |              |                           |                           |
| 0x01C1 5000   | 0x01C1 FFFF |      | -                                |              |                           |                           |
| 0x01C2 0000   | 0x01C2 0FFF | 4K   | Timer64P 0                       |              |                           |                           |
| 0x01C2 1000   | 0x01C2 1FFF | 4K   | Timer64P 1                       |              |                           |                           |
| 0x01C2 2000   | 0x01C2 2FFF | 4K   | I2C 0                            |              |                           |                           |
| 0x01C2 3000   | 0x01C2 3FFF |      |                                  |              |                           |                           |
| 0x01C2 4000   | 0x01C3 FFFF |      |                                  |              |                           |                           |
| 0x01C4 0000   | 0x01C4 0FFF | 4K   | MMC/SD 0                         |              |                           |                           |
| 0x01C4 1000   | 0x01C4 1FFF | 4K   | SPI 0                            |              |                           |                           |
| 0x01C4 2000   | 0x01C4 2FFF | 4K   | UART 0                           |              |                           |                           |
| 0x01C4 3000   | 0x01CF FFFF |      | -                                |              |                           |                           |
| 0x01D0 0000   | 0x01D0 0FFF | 4K   | McASP 0 Control                  |              |                           |                           |
| 0x01D0 1000   | 0x01D0 1FFF | 4K   | McASP 0 AFIFO Control            |              |                           |                           |
| 0x01D0 2000   | 0x01D0 2FFF | 4K   | McASP 0 Data                     |              |                           |                           |
| 0x01D0 3000   | 0x01D0 3FFF |      | -                                |              |                           |                           |
| 0x01D0 4000   | 0x01D0 4FFF | 4K   | McASP 1 Control                  |              |                           |                           |
| 0x01D0 5000   | 0x01D0 5FFF | 4K   | McASP 1 AFIFO Control            |              |                           |                           |
| 0x01D0 6000   | 0x01D0 6FFF | 4K   | McASP 1 Data                     |              |                           |                           |
| 0x01D0 7000   | 0x01D0 BFFF |      | -                                |              |                           |                           |
| 0x01D0 C000   | 0x01D0 CFFF | 4K   | UART 1                           |              |                           |                           |
| 0x01D0 D000   | 0x01D0 DFFF | 4K   | UART 2                           |              |                           |                           |
| 0x01D0 E000   | 0x01DF FFFF |      |                                  |              |                           |                           |
| 0x01E0 0000   | 0x01E0 FFFF | 64K  | USB0                             |              |                           |                           |
| 0x01E1 0000   | 0x01E1 1FFF |      | -                                |              |                           |                           |
| 0x01E1 2000   | 0x01E1 2FFF | 4K   | SPI 1                            |              |                           |                           |
| 0x01E1 3000   | 0x01E1 3FFF |      | -                                |              |                           |                           |
| 0x01E1 4000   | 0x01E1 4FFF | 4K   | Memory Protection Unit 1 (MPU 1) |              |                           |                           |
| 0x01E1 5000   | 0x01E1 5FFF | 4K   | Memory Protection Unit 2 (MPU 2) |              |                           |                           |
| 0x01E1 6000   | 0x01E1 FFFF |      | -                                |              |                           |                           |
| 0x01E2 0000   | 0x01E2 1FFF | 8K   | EMAC Control Module RAM          |              |                           |                           |
| 0x01E2 2000   | 0x01E2 2FFF | 4K   | EMAC Control Module Registers    |              |                           |                           |

**Table 3-2. AM1705 Top Level Memory Map (continued)**

| Start Address | End Address  | Size | ARM Mem Map              | EDMA Mem Map | PRUSS Mem Map              | Master Peripheral Mem Map |
|---------------|--------------|------|--------------------------|--------------|----------------------------|---------------------------|
| 0x01E2 3000   | 0x01E2 3FFF  | 4K   | EMAC Control Registers   |              |                            |                           |
| 0x01E2 4000   | 0x01E2 4FFF  | 4K   | EMAC MDIO port           |              |                            |                           |
| 0x01E2 5000   | 0x01E2 5FFF  |      | -                        |              |                            |                           |
| 0x01E2 6000   | 0x01E2 6FFF  | 4K   | GPIO                     |              |                            |                           |
| 0x01E2 7000   | 0x01E2 7FFF  | 4K   | PSC 1                    |              |                            |                           |
| 0x01E2 8000   | 0x01E2 8FFF  | 4K   | I2C 1                    |              |                            |                           |
| 0x01E2 9000   | 0x01EF FFFF  |      | -                        |              |                            |                           |
| 0x01F0 0000   | 0x01F0 0FFF  | 4K   | eHRPWM 0                 |              |                            |                           |
| 0x01F0 1000   | 0x01F0 1FFF  | 4K   | HRPWM 0                  |              |                            |                           |
| 0x01F0 2000   | 0x01F0 2FFF  | 4K   | eHRPWM 1                 |              |                            |                           |
| 0x01F0 3000   | 0x01F0 3FFF  | 4K   | HRPWM 1                  |              |                            |                           |
| 0x01F0 4000   | 0x01F0 4FFF  | 4K   | eHRPWM 2                 |              |                            |                           |
| 0x01F0 5000   | 0x01F0 5FFF  | 4K   | HRPWM 2                  |              |                            |                           |
| 0x01F0 6000   | 0x01F0 6FFF  | 4K   | ECAP 0                   |              |                            |                           |
| 0x01F0 7000   | 0x01F0 7FFF  | 4K   | ECAP 1                   |              |                            |                           |
| 0x01F0 8000   | 0x01F0 8FFF  | 4K   | ECAP 2                   |              |                            |                           |
| 0x01F0 9000   | 0x01F0 9FFF  | 4K   | EQEP 0                   |              |                            |                           |
| 0x01F0 A000   | 0x01F0 AFFF  | 4K   | EQEP 1                   |              |                            |                           |
| 0x01F0 B000   | 0x5FFF FFFF  |      | -                        |              |                            |                           |
| 0x6000 0000   | 0x61FF FFFF  | 32M  | EMIFA async data (CS2)   |              |                            |                           |
| 0x6200 0000   | 0x63FF FFFF  | 32M  | EMIFA async data (CS3)   |              |                            |                           |
| 0x6400 0000   | 0x65FF FFFF  | 32M  | EMIFA async data (CS4)   |              |                            |                           |
| 0x6600 0000   | 0x67FF FFFF  | 32M  | EMIFA async data (CS5)   |              |                            |                           |
| 0x6800 0000   | 0x6800 7FFF  | 32K  | EMIFA Control Registers  |              |                            |                           |
| 0x6800 8000   | 0x7FFF FFFF  |      | -                        |              |                            |                           |
| 0x8000 0000   | 0x8001 FFFF  | 128K | On-chip RAM              |              |                            |                           |
| 0x8002 0000   | 0xAFFF FFFF  |      | -                        |              |                            |                           |
| 0xB000 0000   | 0xB000 7FFF  | 32K  | EMIFB Control Registers  |              |                            |                           |
| 0xB000 8000   | 0xBFFF FFFF  |      | -                        |              |                            |                           |
| 0xC000 0000   | 0xC7FF FFFF  | 128M | EMIFB SDRAM Data         |              |                            |                           |
| 0xC800 0000   | 0xFFFFC FFFF |      |                          |              |                            |                           |
| 0xFFFFD 0000  | 0xFFFFD FFFF | 64K  | ARM local ROM            | -            |                            |                           |
| 0xFFFFE 0000  | 0xFFFFE DFFF |      | -                        |              |                            |                           |
| 0xFFFFE E000  | 0xFFFFE FFFF | 8K   | ARM Interrupt Controller | -            |                            |                           |
| 0xFFFFF 0000  | 0xFFFFF 1FFF | 8K   | ARM local RAM            | -            | ARM local RAM (PRU 0 Only) | -                         |
| 0xFFFFF 2000  | 0xFFFFF FFFF |      | -                        |              |                            |                           |

## 3.5 Pin Assignments

Extensive use of pin multiplexing is used to accommodate the largest number of peripheral functions in the smallest possible package. Pin multiplexing is controlled using a combination of hardware configuration at device reset and software programmable register settings.

### 3.5.1 Pin Map (Bottom View)

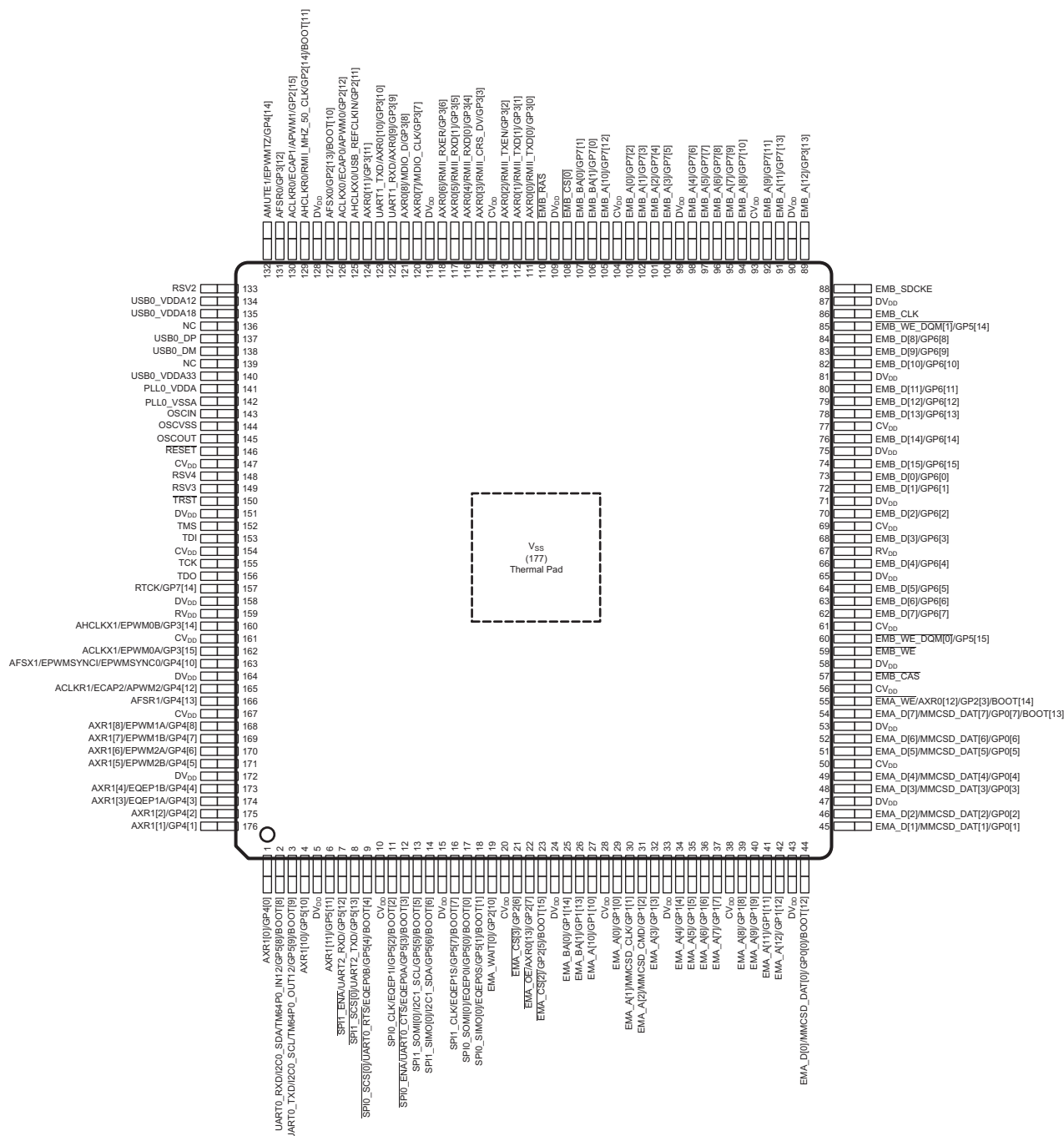


Figure 3-1. Pin Map (PTP)

### 3.6 Terminal Functions

Table 3-3 to Table 3-20 identify the external signal names, the associated pin/ball numbers along with the mechanical package designator, the pin type (I, O, IO, OZ, or PWR), whether the pin/ball has any internal pullup/pulldown resistors, whether the pin/ball is configurable as an IO in GPIO mode, and a functional pin description.

#### 3.6.1 Device Reset and JTAG

**Table 3-3. Reset and JTAG Terminal Functions**

| SIGNAL NAME    | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | DESCRIPTION                         |
|----------------|--------|---------------------|---------------------|-------------------------------------|
|                | PTP    |                     |                     |                                     |
| RESET          |        |                     |                     |                                     |
| RESET          | 146    | I                   |                     | Device reset input                  |
| JTAG           |        |                     |                     |                                     |
| TMS            | 152    | I                   | IPU                 | JTAG test mode select               |
| TDI            | 153    | I                   | IPU                 | JTAG test data input                |
| TDO            | 156    | O                   | IPD                 | JTAG test data output               |
| TCK            | 155    | I                   | IPU                 | JTAG test clock                     |
| TRST           | 150    | I                   | IPD                 | JTAG test reset                     |
| RTCK / GP7[14] | 157    | I/O                 | IPD                 | JTAG Test Clock Return Clock Output |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

#### 3.6.2 High-Frequency Oscillator and PLL

**Table 3-4. High-Frequency Oscillator and PLL Terminal Functions**

| SIGNAL NAME      | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | DESCRIPTION                                        |
|------------------|--------|---------------------|---------------------|----------------------------------------------------|
|                  | PTP    |                     |                     |                                                    |
| 1.2-V OSCILLATOR |        |                     |                     |                                                    |
| OSCIN            | 143    | I                   |                     | Oscillator input                                   |
| OSCOUT           | 145    | O                   |                     | Oscillator output                                  |
| OSCVSS           | 144    | GND                 |                     | Oscillator ground                                  |
| 1.2-V PLL        |        |                     |                     |                                                    |
| PLL0_VDDA        | 141    | PWR                 |                     | PLL analog V <sub>DD</sub> (1.2-V filtered supply) |
| PLL0_VSSA        | 142    | GND                 |                     | PLL analog V <sub>SS</sub> (for filter)            |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.3 External Memory Interface A (ASYNC)

**Table 3-5. External Memory Interface A (EMIFA) Terminal Functions**

| SIGNAL NAME                            | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED              | DESCRIPTION                |
|----------------------------------------|--------|---------------------|---------------------|--------------------|----------------------------|
|                                        | PTP    |                     |                     |                    |                            |
| EMA_D[7]/MMCS_D_DAT[7]/GP0[7]/BOOT[13] | 54     | I/O                 | IPU                 | MMC/SD, GPIO, BOOT | EMIFA data bus             |
| EMA_D[6]/MMCS_D_DAT[6]/GP0[6]          | 52     | I/O                 | IPU                 | MMC/SD, GPIO       |                            |
| EMA_D[5]/MMCS_D_DAT[5]/GP0[5]          | 51     | I/O                 | IPU                 |                    |                            |
| EMA_D[4]/MMCS_D_DAT[4]/GP0[4]          | 49     | I/O                 | IPU                 |                    |                            |
| EMA_D[3]/MMCS_D_DAT[3]/GP0[3]          | 48     | I/O                 | IPU                 |                    |                            |
| EMA_D[2]/MMCS_D_DAT[2]/GP0[2]          | 46     | I/O                 | IPU                 |                    |                            |
| EMA_D[1]/MMCS_D_DAT[1]/GP0[1]          | 45     | I/O                 | IPU                 | MMC/SD, GPIO, BOOT |                            |
| EMA_D[0]/MMCS_D_DAT[0]/GP0[0]/BOOT[12] | 44     | I/O                 | IPU                 |                    | EMIFA address bus          |
| EMA_A[12]/GP1[12]                      | 42     | O                   | IPU                 | GPIO               |                            |
| EMA_A[11]/ GP1[11]                     | 41     | O                   | IPU                 |                    |                            |
| EMA_A[10]/GP1[10]                      | 27     | O                   | IPU                 |                    |                            |
| EMA_A[9]/GP1[9]                        | 40     | O                   | IPU                 |                    |                            |
| EMA_A[8]/GP1[8]                        | 39     | O                   | IPU                 |                    |                            |
| EMA_A[7]/GP1[7]                        | 37     | O                   | IPD                 |                    |                            |
| EMA_A[6]/GP1[6]                        | 36     | O                   | IPD                 |                    |                            |
| EMA_A[5]/GP1[5]                        | 35     | O                   | IPD                 |                    |                            |
| EMA_A[4]/GP1[4]                        | 34     | O                   | IPD                 |                    |                            |
| EMA_A[3]/GP1[3]                        | 32     | O                   | IPD                 |                    |                            |
| EMA_A[2]/MMCS_D_CMD/GP1[2]             | 31     | O                   | IPU                 | MMCS_D, GPIO       | EMIFA address bus          |
| EMA_A[1]/MMCS_D_CLK/GP1[1]             | 30     | O                   | IPU                 |                    |                            |
| EMA_A[0]/GP1[0]                        | 29     | O                   | IPD                 | GPIO               | EMIFA bank address         |
| EMA_BA[1]/GP1[13]                      | 26     | O                   | IPU                 |                    |                            |
| EMA_BA[0]/GP1[14]                      | 25     | O                   | IPU                 | GPIO               |                            |
| EMA_CS[3] /GP2[6]                      | 21     | O                   | IPU                 | GPIO               | EMIFA Async Chip Select    |
| EMA_CS[2]/GP2[5]/BOOT[15]              | 23     | O                   | IPU                 | GPIO, BOOT         |                            |
| EMA_OE /AXR0[13]/GP2[7]                | 22     | O                   | IPU                 | McASP0, GPIO       | EMIFA output enable        |
| EMA_WAIT[0]/ GP2[10]                   | 19     | I                   | IPU                 | GPIO               | EMIFA wait input/interrupt |

(1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.

**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.

(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.4 External Memory Interface B (SDRAM only)

**Table 3-6. External Memory Interface B (EMIFB) Terminal Functions**

| SIGNAL NAME            | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED | DESCRIPTION                            |
|------------------------|--------|---------------------|---------------------|-------|----------------------------------------|
|                        | PTP    |                     |                     |       |                                        |
| EMB_D[15]/GP6[15]      | 74     | I/O                 | IPD                 | GPIO  | EMIFB SDRAM data bus                   |
| EMB_D[14]/GP6[14]      | 76     | I/O                 | IPD                 |       |                                        |
| EMB_D[13]/GP6[13]      | 78     | I/O                 | IPD                 |       |                                        |
| EMB_D[12]/GP6[12]      | 79     | I/O                 | IPD                 |       |                                        |
| EMB_D[11]/GP6[11]      | 80     | I/O                 | IPD                 |       |                                        |
| EMB_D[10]/GP6[10]      | 82     | I/O                 | IPD                 |       |                                        |
| EMB_D[9]/GP6[9]        | 83     | I/O                 | IPD                 |       |                                        |
| EMB_D[8]/GP6[8]        | 84     | I/O                 | IPD                 |       |                                        |
| EMB_D[7]/GP6[7]        | 62     | I/O                 | IPD                 |       |                                        |
| EMB_D[6]/GP6[6]        | 63     | I/O                 | IPD                 |       |                                        |
| EMB_D[5]/GP6[5]        | 64     | I/O                 | IPD                 |       |                                        |
| EMB_D[4]/GP6[4]        | 66     | I/O                 | IPD                 |       |                                        |
| EMB_D[3]/GP6[3]        | 68     | I/O                 | IPD                 |       |                                        |
| EMB_D[2]/GP6[2]        | 70     | I/O                 | IPD                 |       |                                        |
| EMB_D[1]/GP6[1]        | 72     | I/O                 | IPD                 |       |                                        |
| EMB_D[0]/GP6[0]        | 73     | I/O                 | IPD                 |       |                                        |
| EMB_A[12]/GP3[13]      | 89     | O                   | IPD                 | GPIO  | EMIFB SDRAM row/column address bus     |
| EMB_A[11]/GP7[13]      | 91     | O                   | IPD                 |       |                                        |
| EMB_A[10]/GP7[12]      | 105    | O                   | IPD                 |       |                                        |
| EMB_A[9]/GP7[11]       | 92     | O                   | IPD                 |       |                                        |
| EMB_A[8]/GP7[10]       | 94     | O                   | IPD                 |       |                                        |
| EMB_A[7]/GP7[9]        | 95     | O                   | IPD                 |       |                                        |
| EMB_A[6]/GP7[8]        | 96     | O                   | IPD                 |       |                                        |
| EMB_A[5]/GP7[7]        | 97     | O                   | IPD                 |       |                                        |
| EMB_A[4]/GP7[6]        | 98     | O                   | IPD                 | GPIO  | EMIFB SDRAM row/column address         |
| EMB_A[3]/GP7[5]        | 100    | O                   | IPD                 |       |                                        |
| EMB_A[2]/GP7[4]        | 101    | O                   | IPD                 |       |                                        |
| EMB_A[1]/GP7[3]        | 102    | O                   | IPD                 |       |                                        |
| EMB_A[0]/GP7[2]        | 103    | O                   | IPD                 |       | EMIFB SDRAM bank address               |
| EMB_BA[1]/GP7[0]       | 106    | O                   | IPU                 |       |                                        |
| EMB_BA[0]/GP7[1]       | 107    | O                   | IPU                 |       |                                        |
| EMB_CLK                | 86     | O                   | IPU                 |       | EMIF SDRAM clock                       |
| EMB_SDCKE              | 88     | O                   | IPU                 |       | EMIFB SDRAM clock enable               |
| EMB_WE                 | 59     | O                   | IPU                 |       | EMIFB write enable                     |
| EMB_RAS                | 110    | O                   | IPU                 |       | EMIFB SDRAM row address strobe         |
| EMB_CAS                | 57     | O                   | IPU                 |       | EMIFB column address strobe            |
| EMB_CS[0]              | 108    | O                   | IPU                 |       | EMIFB SDRAM chip select 0              |
| EMB_WE_DQM[1] /GP5[14] | 85     | O                   | IPU                 | GPIO  | EMIFB write enable/data mask for EMB_D |
| EMB_WE_DQM[0] /GP5[15] | 60     | O                   | IPU                 |       |                                        |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.5 Serial Peripheral Interface Modules (SPI0, SPI1)

**Table 3-7. Serial Peripheral Interface (SPI) Terminal Functions**

| SIGNAL NAME                                  | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                     | DESCRIPTION                   |
|----------------------------------------------|--------|---------------------|---------------------|---------------------------|-------------------------------|
|                                              | PTP    |                     |                     |                           |                               |
| SPI0                                         |        |                     |                     |                           |                               |
| SPI0_SCS[0] /UART0_RTS/EQEP0B/GP5[4]/BOOT[4] | 9      | I/O                 | IPU                 | UART0, EQEP0B, GPIO, BOOT | SPI0 chip select              |
| SPI0_ENA /UART0_CTS/EQEP0A/GP5[3]/BOOT[3]    | 12     | I/O                 | IPU                 | UART0, EQEP0A, GPIO, BOOT | SPI0 enable                   |
| SPI0_CLK/EQEP1I/GP5[2]/BOOT[2]               | 11     | I/O                 | IPD                 | eQEP1, GPIO, BOOT         | SPI0 clock                    |
| SPI0_SIMO[0]/EQEP0S/GP5[1]/BOOT[1]           | 18     | I/O                 | IPD                 | eQEP0, GPIO, BOOT         | SPI0 data slave-in-master-out |
| SPI0_SOMI[0]/EQEP0I/GP5[0]/BOOT[0]           | 17     | I/O                 | IPD                 |                           | SPI0 data slave-out-master-in |
| SPI1                                         |        |                     |                     |                           |                               |
| SPI1_SCS[0] /UART2_TXD/GP5[13]               | 8      | I/O                 | IPU                 | UART2, GPIO               | SPI1 chip select              |
| SPI1_ENA /UART2_RXD/GP5[12]                  | 7      | I/O                 | IPU                 |                           | SPI1 enable                   |
| SPI1_CLK/EQEP1S/GP5[7]/BOOT[7]               | 16     | I/O                 | IPD                 | eQEP1, GPIO, BOOT         | SPI1 clock                    |
| SPI1_SIMO[0]/I2C1_SDA/GP5[6]/BOOT[6]         | 14     | I/O                 | IPU                 | I2C1, GPIO, BOOT          | SPI1 data slave-in-master-out |
| SPI1_SOMI[0]/I2C1_SCL/GP5[5]/BOOT[5]         | 13     | I/O                 | IPU                 |                           | SPI1 data slave-out-master-in |

(1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.

**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.

(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.6 Enhanced Capture/Auxiliary PWM Modules (eCAP0, eCAP1, eCAP2)

The eCAP Module pins function as either input captures or auxiliary PWM 32-bit outputs, depending upon how the eCAP module is programmed.

**Table 3-8. Enhanced Capture Module (eCAP) Terminal Functions**

| SIGNAL NAME                         | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED        | DESCRIPTION                                        |
|-------------------------------------|--------|---------------------|---------------------|--------------|----------------------------------------------------|
|                                     | PTP    |                     |                     |              |                                                    |
| eCAP0                               |        |                     |                     |              |                                                    |
| ACLKX0/ <b>ECAP0</b> /APWM0/GP2[12] | 126    | I/O                 | IPD                 | McASP0, GPIO | enhanced capture 0 input or auxiliary PWM 0 output |
| eCAP1                               |        |                     |                     |              |                                                    |
| ACLKR0/ <b>ECAP1</b> /APWM1/GP2[15] | 130    | I/O                 | IPD                 | McASP0, GPIO | enhanced capture 1 input or auxiliary PWM 1 output |
| eCAP2                               |        |                     |                     |              |                                                    |
| ACLKR1/ <b>ECAP2</b> /APWM2/GP4[12] | 165    | I/O                 | IPD                 | McASP1, GPIO | enhanced capture 2 input or auxiliary PWM 2 output |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.7 Enhanced Pulse Width Modulators (eHRPWM0, eHRPWM1, eHRPWM2)

**Table 3-9. Enhanced Pulse Width Modulator (eHRPWM) Terminal Functions**

| SIGNAL NAME                       | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                          | DESCRIPTION                                                 |
|-----------------------------------|--------|---------------------|---------------------|--------------------------------|-------------------------------------------------------------|
|                                   | PTP    |                     |                     |                                |                                                             |
| eHRPWM0                           |        |                     |                     |                                |                                                             |
| ACLKX1/EPWM0A/GP3[15]             | 162    | I/O                 | IPD                 | McASP1, GPIO                   | eHRPWM0 A output (with high-resolution)                     |
| AHCLKX1/EPWM0B/GP3[14]            | 160    | I/O                 | IPD                 |                                | eHRPWM0 B output                                            |
| AMUTE1/EPWMTZ/GP4[14]             | 132    | I/O                 | IPD                 | McASP1, eHRPWM1, GPIO, eHRPWM2 | eHRPWM0 trip zone input                                     |
| AFSX1/EPWMSYNCI/EPWMSYNCO/GP4[10] | 163    | I/O                 | IPD                 | McASP1, eHRPWM0, GPIO          | Sync input to eHRPWM0 module or sync output to external PWM |
| eHRPWM1                           |        |                     |                     |                                |                                                             |
| AXR1[8]/EPWM1A/GP4[8]             | 168    | I/O                 | IPD                 | McASP1, GPIO                   | eHRPWM1 A (with high-resolution)                            |
| AXR1[7]/EPWM1B/GP4[7]             | 169    | I/O                 | IPD                 |                                | eHRPWM1 B output                                            |
| AMUTE1/EPWMTZ/GP4[14]             | 132    | I/O                 | IPD                 | McASP1, eHRPWM0, GPIO, eHRPWM2 | eHRPWM1 trip zone input                                     |
| eHRPWM2                           |        |                     |                     |                                |                                                             |
| AXR1[6]/EPWM2A/GP4[6]             | 170    | I/O                 | IPD                 | McASP1, GPIO                   | eHRPWM2 A (with high-resolution)                            |
| AXR1[5]/EPWM2B/GP4[5]             | 171    | I/O                 | IPD                 |                                | eHRPWM2 B output                                            |
| AMUTE1/EPWMTZ/GP4[14]             | 132    | I/O                 | IPD                 | McASP1, eHRPWM0, GPIO, eHRPWM2 | eHRPWM2 trip zone input                                     |

(1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.

**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.

(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.8 Enhanced Quadrature Encoder Pulse Module (eQEP)

**Table 3-10. Enhanced Quadrature Encoder Pulse Module (eQEP) Terminal Functions**

| SIGNAL NAME                                 | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                   | DESCRIPTION             |
|---------------------------------------------|--------|---------------------|---------------------|-------------------------|-------------------------|
|                                             | PTP    |                     |                     |                         |                         |
| eQEP0                                       |        |                     |                     |                         |                         |
| SPI0_ENA/UART0_CTS/EQEP0A/GP5[3]/BOOT[3]    | 12     | I                   | IPU                 | SPI0, UART0, GPIO, BOOT | eQEP0A quadrature input |
| SPI0_SCS[0]/UART0_RTS/EQEP0B/GP5[4]/BOOT[4] | 9      | I                   | IPU                 |                         | eQEP0B quadrature input |
| SPI0_SOMI[0]/EQEP0I/GP5[0]/BOOT[0]          | 17     | I                   | IPD                 | SPI0, GPIO, BOOT        | eQEP0 index             |
| SPI0_SIMO[0]/EQEP0S/GP5[1]/BOOT[1]          | 18     | I                   | IPD                 |                         | eQEP0 strobe            |
| eQEP1                                       |        |                     |                     |                         |                         |
| AXR1[3]/EQEP1A/GP4[3]                       | 174    | I                   | IPD                 | McASP1, GPIO            | eQEP1A quadrature input |
| AXR1[4]/EQEP1B/GP4[4]                       | 173    | I                   | IPD                 |                         | eQEP1B quadrature input |
| SPI0_CLK/EQEP1I/GP5[2]/BOOT[2]              | 11     | I                   | IPD                 | SPI0, GPIO, BOOT        | eQEP1 index             |
| SPI1_CLK/EQEP1S/GP5[7]/BOOT[7]              | 16     | I                   | IPD                 | SPI1, GPIO, BOOT        | eQEP1 strobe            |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.9 Boot

**Table 3-11. Boot Terminal Functions<sup>(1)</sup>**

| SIGNAL NAME                                            | PIN NO | TYPE <sup>(2)</sup> | PULL <sup>(3)</sup> | MUXED                     | DESCRIPTION            |
|--------------------------------------------------------|--------|---------------------|---------------------|---------------------------|------------------------|
|                                                        | PTP    |                     |                     |                           |                        |
| EMA_CS[2]/GP2[5]/ <b>BOOT[15]</b>                      | 23     | I                   | IPU                 | EMIFA, GPIO               | Boot Selection Signals |
| EMA_WE/AXR0[12]/GP2[3]/ <b>BOOT[14]</b>                | 55     | I                   | IPU                 | EMIFA, McASP0, GPIO       |                        |
| EMA_D[7]/MMCSA_DAT[7]/GP0[7]/ <b>BOOT[13]</b>          | 54     | I                   | IPU                 | EMIFA, MMC/SD, GPIO       |                        |
| EMA_D[0]/MMCSA_DAT[0]/GP0[0]/ <b>BOOT[12]</b>          | 44     | I                   | IPU                 |                           |                        |
| AHCLKR0/RMII_MHZ_50_CLK/GP2[14]/ <b>BOOT[11]</b>       | 129    | I                   | IPD                 | McASP0, EMAC, GPIO        |                        |
| AFSX0/GP2[13]/ <b>BOOT[10]</b>                         | 127    | I                   | IPD                 | McASP0, GPIO              |                        |
| UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/ <b>BOOT[9]</b> | 3      | I                   | IPU                 | UART0, I2C0, Timer0, GPIO |                        |
| UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/ <b>BOOT[8]</b>  | 2      | I                   | IPU                 | UART0, I2C0, Timer0, GPIO |                        |
| SPI1_CLK/EQEP1S/GP5[7]/ <b>BOOT[7]</b>                 | 16     | I                   | IPD                 | SPI1, eQEP1, GPIO         |                        |
| SPI1_SIMO[0]/I2C1_SDA/GP5[6]/ <b>BOOT[6]</b>           | 14     | I                   | IPU                 | SPI1, I2C1, GPIO          |                        |
| SPI1_SOMI[0]/I2C1_SCL/GP5[5]/ <b>BOOT[5]</b>           | 13     | I                   | IPU                 |                           |                        |
| SPI0_SCS[0]/UART0_RTS/EQEP0B/GP5[4]/ <b>BOOT[4]</b>    | 9      | I                   | IPU                 | SPI0, UART0, eQEP0, GPIO  |                        |
| SPI0_ENA/UART0_CTS/EQEP0A/GP5[3]/ <b>BOOT[3]</b>       | 12     | I                   | IPU                 | SPI0, UART0, eQEP0, GPIO  |                        |
| SPI0_CLK/EQEP1I/GP5[2]/ <b>BOOT[2]</b>                 | 11     | I                   | IPD                 | SPI0, eQEP1, GPIO         |                        |
| SPI0_SIMO[0]/EQEP0S/GP5[1]/ <b>BOOT[1]</b>             | 18     | I                   | IPD                 | SPI0, eQEP0, GPIO         |                        |
| SPI0_SOMI[0]/EQEP0I/GP5[0]/ <b>BOOT[0]</b>             | 17     | I                   | IPD                 |                           |                        |

(1) Boot decoding will be defined in the ROM datasheet.

(2) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.

**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.

(3) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.10 Universal Asynchronous Receiver/Transmitters (UART0, UART1, UART2)

**Table 3-12. Universal Asynchronous Receiver/Transmitter (UART) Terminal Functions**

| SIGNAL NAME                                          | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                     | DESCRIPTION                |
|------------------------------------------------------|--------|---------------------|---------------------|---------------------------|----------------------------|
|                                                      | PTP    |                     |                     |                           |                            |
| UART0                                                |        |                     |                     |                           |                            |
| UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/BOOT[8]        | 2      | I                   | IPU                 | I2C0, BOOT, Timer0, GPIO, | UART0 receive data         |
| UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/BOOT[9]       | 3      | O                   | IPU                 | I2C0, Timer0, GPIO, BOOT  | UART0 transmit data        |
| SPI0_SCS[0]/ <u>UART0_RTS</u> /EQEP0B/GP5[4]/BOOT[4] | 9      | O                   | IPU                 | SPI0, eQEP0, GPIO, BOOT   | UART0 ready-to-send output |
| SPI0_ENA/ <u>UART0_CTS</u> /EQEP0A/GP5[3]/BOOT[3]    | 12     | I                   | IPU                 |                           | UART0 clear-to-send input  |
| UART1                                                |        |                     |                     |                           |                            |
| UART1_RXD/AXR0[9]/GP3[9] <sup>(3)</sup>              | 122    | I                   | IPD                 | McASP0, GPIO              | UART1 receive data         |
| UART1_TXD/AXR0[10]/GP3[10] <sup>(3)</sup>            | 123    | O                   | IPD                 |                           | UART1 transmit data        |
| UART2                                                |        |                     |                     |                           |                            |
| <u>SPI1_ENA</u> /UART2_RXD/GP5[12]                   | 7      | I                   | IPU                 | SPI1, GPIO                | UART2 receive data         |
| <u>SPI1_SCS[0]</u> /UART2_TXD/GP5[13]                | 8      | O                   | IPU                 |                           | UART2 transmit data        |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor
- (3) As these signals are internally pulled down while the device is in reset, it is necessary to externally pull them high with resistors if UART1 boot mode is used.

### 3.6.11 Inter-Integrated Circuit Modules(I2C0, I2C1)

**Table 3-13. Inter-Integrated Circuit (I2C) Terminal Functions**

| SIGNAL NAME                                    | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                     | DESCRIPTION       |
|------------------------------------------------|--------|---------------------|---------------------|---------------------------|-------------------|
|                                                | PTP    |                     |                     |                           |                   |
| I2C0                                           |        |                     |                     |                           |                   |
| UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/BOOT[8]  | 2      | I/O                 | IPU                 | UART0, Timer0, GPIO, BOOT | I2C0 serial data  |
| UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/BOOT[9] | 3      | I/O                 | IPU                 | UART0, Timer0, GPIO, BOOT | I2C0 serial clock |
| I2C1                                           |        |                     |                     |                           |                   |
| SPI1_SIMO[0]/I2C1_SDA/GP5[6]/BOOT[6]           | 14     | I/O                 | IPU                 | SPI1, GPIO, BOOT          | I2C1 serial data  |
| SPI1_SOMI[0]/I2C1_SCL/GP5[5]/BOOT[5]           | 13     | I/O                 | IPU                 |                           | I2C1 serial clock |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.12 Timers

**Table 3-14. Timers Terminal Functions**

| SIGNAL NAME                                                                       | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                   | DESCRIPTION         |
|-----------------------------------------------------------------------------------|--------|---------------------|---------------------|-------------------------|---------------------|
|                                                                                   | PTP    |                     |                     |                         |                     |
| TIMER0                                                                            |        |                     |                     |                         |                     |
| UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/BOOT[8]                                     | 2      | I                   | IPU                 | UART0, I2C0, GPIO, BOOT | Timer0 lower input  |
| UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/BOOT[9]                                    | 3      | O                   | IPU                 |                         | Timer0 lower output |
| TIMER1 (Watchdog )                                                                |        |                     |                     |                         |                     |
| No external pins. The Timer1 peripheral pins are not pinned out as external pins. |        |                     |                     |                         |                     |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.13 Multichannel Audio Serial Ports (McASP0, McASP1)

**Table 3-15. Multichannel Audio Serial Ports (McASPs) Terminal Functions**

| SIGNAL NAME                              | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED             | DESCRIPTION                  |
|------------------------------------------|--------|---------------------|---------------------|-------------------|------------------------------|
|                                          | PTP    |                     |                     |                   |                              |
| McASP0                                   |        |                     |                     |                   |                              |
| EMA_OE/AXR0[13]/GP2[7]                   | 22     | I/O                 | IPU                 | EMIFA, GPIO       | McASP0 serial data           |
| EMA_WE/AXR0[12]/GP2[3]/BOOT[14]          | 55     | I/O                 | IPU                 | EMIFA, GPIO, BOOT |                              |
| AXR0[11] / GP3[11]                       | 124    | I/O                 | IPD                 | McASP2, GPIO      |                              |
| UART1_TXD/AXR0[10]/GP3[10]               | 123    | I/O                 | IPD                 | GPIO              |                              |
| UART1_RXD/AXR0[9]/GP3[9]                 | 122    | I/O                 | IPD                 | GPIO              |                              |
| AXR0[8]/MDIO_D/GP3[8]                    | 121    | I/O                 | IPU                 | MDIO, GPIO        |                              |
| AXR0[7]/MDIO_CLK/GP3[7]                  | 120    | I/O                 | IPD                 |                   |                              |
| AXR0[6]/RMII_RXER/GP3[6]                 | 118    | I/O                 | IPD                 | EMAC, GPIO        |                              |
| AXR0[5]/RMII_RXD[1]/GP3[5]               | 117    | I/O                 | IPD                 |                   |                              |
| AXR0[4]/RMII_RXD[0]/GP3[4]               | 116    | I/O                 | IPD                 |                   |                              |
| AXR0[3]/RMII_CRS_DV/GP3[3]               | 115    | I/O                 | IPD                 |                   |                              |
| AXR0[2]/RMII_TXEN/GP3[2]                 | 113    | I/O                 | IPD                 |                   |                              |
| AXR0[1]/RMII_TXD[1]/GP3[1]               | 112    | I/O                 | IPD                 |                   |                              |
| AXR0[0]/RMII_TXD[0]/GP3[0]               | 111    | I/O                 | IPD                 |                   |                              |
| AHCLKX0/USB_REFCLKIN/GP2[11]             | 125    | I/O                 | IPD                 | USB, GPIO         | McASP0 transmit master clock |
| ACLKX0/ECAP0/APWM0/GP2[12]               | 126    | I/O                 | IPD                 | eCAP0, GPIO       | McASP0 transmit bit clock    |
| AFSX0/GP2[13]/BOOT[10]                   | 127    | I/O                 | IPD                 | GPIO, BOOT        | McASP0 transmit frame sync   |
| AHCLKR0/RMII_MHZ_50_CLK/GP2[14]/BOOT[11] | 129    | I/O                 | IPD                 | EMAC, GPIO, BOOT  | McASP0 receive master clock  |
| ACLKR0/ECAP1/APWM1/GP2[15]               | 130    | I/O                 | IPD                 | eCAP1, GPIO       | McASP0 receive bit clock     |
| AFSR0/GP3[12]                            | 131    | I/O                 | IPD                 | GPIO              | McASP0 receive frame sync    |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

**Table 3-15. Multichannel Audio Serial Ports (McASPs) Terminal Functions (continued)**

| SIGNAL NAME                       | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED                           | DESCRIPTION                  |
|-----------------------------------|--------|---------------------|---------------------|---------------------------------|------------------------------|
|                                   | PTP    |                     |                     |                                 |                              |
| McASP1                            |        |                     |                     |                                 |                              |
| AXR1[11]/GP5[11]                  | 6      | I/O                 | IPU                 | GPIO                            | McASP1 serial data           |
| AXR1[10]/GP5[10]                  | 4      | I/O                 | IPU                 |                                 |                              |
| AXR1[8]/EPWM1A/GP4[8]             | 168    | I/O                 | IPD                 | eHRPWM1 A, GPIO                 |                              |
| AXR1[7]/EPWM1B/GP4[7]             | 169    | I/O                 | IPD                 | eHRPWM1 B, GPIO                 |                              |
| AXR1[6]/EPWM2A/GP4[6]             | 170    | I/O                 | IPD                 | eHRPWM2 A, GPIO                 |                              |
| AXR1[5]/EPWM2B/GP4[5]             | 171    | I/O                 | IPD                 | eHRPWM2 B, GPIO                 |                              |
| AXR1[4]/EQEP1B/GP4[4]             | 173    | I/O                 | IPD                 | eQEP, GPIO                      |                              |
| AXR1[3]/EQEP1A/GP4[3]             | 174    | I/O                 | IPD                 |                                 |                              |
| AXR1[2]/GP4[2]                    | 175    | I/O                 | IPD                 | GPIO                            |                              |
| AXR1[1]/GP4[1]                    | 176    | I/O                 | IPD                 |                                 |                              |
| AXR1[0]/GP4[0]                    | 1      | I/O                 | IPD                 |                                 |                              |
| AHCLKX1/EPWM0B/GP3[14]            | 160    | I/O                 | IPD                 | eHRPWM0, GPIO                   | McASP1 transmit master clock |
| ACLKX1/EPWM0A/GP3[15]             | 162    | I/O                 | IPD                 | eHRPWM0, GPIO                   | McASP1 transmit bit clock    |
| AFSX1/EPWMSYNCI/EPWMSYNCO/GP4[10] | 163    | I/O                 | IPD                 | eHRPWM0, GPIO                   | McASP1 transmit frame sync   |
| ACLKR1/ECAP2/APWM2/GP4[12]        | 165    | I/O                 | IPD                 | eCAP2, GPIO                     | McASP1 receive bit clock     |
| AFSR1/GP4[13]                     | 166    | I/O                 | IPD                 | GPIO                            | McASP1 receive frame sync    |
| AMUTE1/EPWMTZ/GP4[14]             | 132    | O                   | IPD                 | eHRPWM0, eHRPWM1, eHRPWM2, GPIO | McASP1 mute output           |

### 3.6.14 Universal Serial Bus Modules (USB0)

**Table 3-16. Universal Serial Bus (USB) Terminal Functions**

| SIGNAL NAME                  | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | DESCRIPTION                                                                                                                                                                          |
|------------------------------|--------|---------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                              | PTP    |                     |                     |                                                                                                                                                                                      |
| USB0 2.0 OTG (USB0)          |        |                     |                     |                                                                                                                                                                                      |
| USB0_DM                      | 138    | A                   |                     | USB0 PHY data minus                                                                                                                                                                  |
| USB0_DP                      | 137    | A                   |                     | USB0 PHY data plus                                                                                                                                                                   |
| USB0_VDDA33                  | 140    | PWR                 |                     | USB0 PHY 3.3-V supply                                                                                                                                                                |
| USB0_VDDA18                  | 135    | PWR                 |                     | USB0 PHY 1.8-V supply input                                                                                                                                                          |
| USB0_VDDA12 <sup>(3)</sup>   | 134    | PWR                 |                     | USB0 PHY 1.2-V LDO output for bypass cap. For proper device operation, this pin is recommended to be connected via a 0.22 μF capacitor to VSS (GND), even if USB0 is not being used. |
| AHCLKX0/USB_REFCLKIN/GP2[11] | 125    | I                   | IPD                 | USB_REFCLKIN. Optional clock input.                                                                                                                                                  |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor
- (3) Core power supply LDO output for USB PHY. This pin must be connected via a 0.22  $\mu$ F capacitor to VSS.

### 3.6.15 Ethernet Media Access Controller (EMAC)

**Table 3-17. Ethernet Media Access Controller (EMAC) Terminal Functions**

| SIGNAL NAME                              | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED              | DESCRIPTION                        |
|------------------------------------------|--------|---------------------|---------------------|--------------------|------------------------------------|
|                                          | PTP    |                     |                     |                    |                                    |
| RMII                                     |        |                     |                     |                    |                                    |
| AHCLKR0/RMII_MHZ_50_CLK/GP2[14]/BOOT[11] | 129    | I/O                 | IPD                 | McASP0, GPIO, BOOT | EMAC 50-MHz clock input or output  |
| AXR0[6]/RMII_RXER/GP3[6]                 | 118    | I                   | IPD                 | McASP0, GPIO       | EMAC RMII receiver error           |
| AXR0[5]/RMII_RXD[1]/GP3[5]               | 117    | I                   | IPD                 |                    | EMAC RMII receive data             |
| AXR0[4]/RMII_RXD[0]/GP3[4]               | 116    | I                   | IPD                 |                    | EMAC RMII carrier sense data valid |
| AXR0[3]/RMII_CRS_DV/GP3[3]               | 115    | I                   | IPD                 |                    | EMAC RMII transmit enable          |
| AXR0[2]/RMII_TXEN/GP3[2]                 | 113    | O                   | IPD                 |                    | EMAC RMII trasmit data             |
| AXR0[1]/RMII_TXD[1]/GP3[1]               | 112    | O                   | IPD                 |                    |                                    |
| AXR0[0]/RMII_TXD[0]/GP3[0]               | 111    | O                   | IPD                 |                    |                                    |
| MDIO                                     |        |                     |                     |                    |                                    |
| AXR0[8]/MDIO_D/GP3[8]                    | 121    | I/O                 | IPU                 | McASP0, GPIO       | MDIO data clock                    |
| AXR0[7]/MDIO_CLK/GP3[7]                  | 120    | O                   | IPD                 |                    |                                    |

- (1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.  
**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.
- (2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.16 Multimedia Card/Secure Digital (MMC/SD)

**Table 3-18. Multimedia Card/Secure Digital (MMC/SD) Terminal Functions**

| SIGNAL NAME                           | PIN NO | TYPE <sup>(1)</sup> | PULL <sup>(2)</sup> | MUXED             | DESCRIPTION   |
|---------------------------------------|--------|---------------------|---------------------|-------------------|---------------|
|                                       | PTP    |                     |                     |                   |               |
| EMA_A[1]/MMCSD_CLK/GP1[1]             | 30     | O                   | IPU                 | EMIFA, GPIO       | MMCSD Clock   |
| EMA_A[2]/MMCSD_CMD/GP1[2]             | 31     | I/O                 | IPU                 |                   | MMCSD Command |
| EMA_D[7]/MMCSD_DAT[7]/GP0[7]/BOOT[13] | 54     | I/O                 | IPU                 | EMIFA, GPIO, BOOT | MMC/SD data   |
| EMA_D[6]/MMCSD_DAT[6]/GP0[6]          | 52     | I/O                 | IPU                 | EMIFA, GPIO       |               |
| EMA_D[5]/MMCSD_DAT[5]/GP0[5]          | 51     | I/O                 | IPU                 |                   |               |
| EMA_D[4]/MMCSD_DAT[4]/GP0[4]          | 49     | I/O                 | IPU                 |                   |               |
| EMA_D[3]/MMCSD_DAT[3]/GP0[3]          | 48     | I/O                 | IPU                 |                   |               |
| EMA_D[2]/MMCSD_DAT[2]/GP0[2]          | 46     | I/O                 | IPU                 |                   |               |
| EMA_D[1]/MMCSD_DAT[1]/GP0[1]          | 45     | I/O                 | IPU                 |                   |               |
| EMA_D[0]/MMCSD_DAT[0]/GP0[0]/BOOT[12] | 44     | I/O                 | IPU                 | EMIFA, GPIO, BOOT |               |

(1) I = Input, O = Output, I/O = Bidirectional, Z = High impedance, PWR = Supply voltage, GND = Ground, A = Analog signal.

**Note:** The pin type shown refers to the input, output or high-impedance state of the pin function when configured as the the signal name highlighted in bold. All multiplexed signals may enter a high-impedance state when the configured function is input-only or the configured function supports high-Z operation. All GPIO signals can be used as input or output. For multiplexed pins where functions have different types (i.e., input versus output), the table reflects the pin function direction for that particular peripheral.

(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

### 3.6.17 Reserved and No Connect

**Table 3-19. Reserved and No Connect Terminal Functions**

| SIGNAL NAME | PIN NO | TYPE <sup>(1)</sup> | DESCRIPTION                                                                                                                                            |
|-------------|--------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | PTP    |                     |                                                                                                                                                        |
| <b>RSV2</b> | 133    | -                   | Reserved. For proper device operation, this pin <b>must</b> be tied directly to CV <sub>DD</sub> .                                                     |
| <b>RSV3</b> | 149    | PWR                 | Reserved. For proper device operation, this pin <b>must</b> be tied directly to CV <sub>DD</sub> or left unconnected [do not connect to ground (VSS)]. |
| <b>RSV4</b> | 148    | I                   | Reserved. For proper device operation, this pin must be tied low or to CV <sub>DD</sub> .                                                              |
| <b>NC</b>   | 136    | -                   | No Connect (leave unconnected)                                                                                                                         |
| <b>NC</b>   | 139    | -                   | No Connect (leave unconnected)                                                                                                                         |

(1) PWR = Supply voltage.

### 3.6.18 Supply and Ground

**Table 3-20. Supply and Ground Terminal Functions**

| SIGNAL NAME                       | PIN NO                                                                                       | TYPE <sup>(1)</sup> | DESCRIPTION                      |
|-----------------------------------|----------------------------------------------------------------------------------------------|---------------------|----------------------------------|
|                                   | PTP                                                                                          |                     |                                  |
| <b>CVDD</b> (Core supply)         | 10, 20, 28, 38, 50, 56, 61, 69, 77, 93, 104, 114, 147, 154, 161, 167                         | PWR                 | Core supply voltage pins         |
| <b>RVDD</b> (Internal RAM supply) | 67, 159                                                                                      | PWR                 | Internal ram supply voltage pins |
| <b>DVDD</b> (I/O supply)          | 5, 15, 24, 33, 43, 47, 53, 58, 65, 71, 75, 81, 87, 90, 99, 109, 119, 128, 151, 158, 164, 172 | PWR                 | I/O supply voltage pins          |
| <b>VSS</b> (Ground)               | 177                                                                                          | GND                 | Ground pins                      |

(1) PWR = Supply voltage, GND - Ground.

### 3.6.19 Unused USB0 (USB2.0) Pin Configurations

**Table 3-21. Unused USB0 Pin Configurations**

| SIGNAL NAME                      | Configuration<br>(When USB0 is not used)                                                                   |
|----------------------------------|------------------------------------------------------------------------------------------------------------|
| USB0_DM                          | No connect                                                                                                 |
| USB0_DP                          | No connect                                                                                                 |
| USB0_VDDA33                      | No connect                                                                                                 |
| USB0_VDDA18                      | No connect                                                                                                 |
| USB0_VDDA12                      | Internal USB0 PHY output connected to an external 0.22 $\mu$ F filter capacitor, even if USB0 is not used. |
| AHCLKX0/USB_REFCLKIN/<br>GP2[11] | No connect or use as alternate function                                                                    |

## 4 Device Configuration

### 4.1 Boot Modes

This device supports a variety of boot modes through an internal ROM bootloader. This device does not support dedicated hardware boot modes; therefore, all boot modes utilize the internal ROM. The input states of the BOOT pins are sampled and latched into the BOOTCFG register, which is part of the system configuration (SYSCFG) module, when device reset is deasserted. Boot mode selection is determined by the values of the BOOT pins

The following boot modes are supported:

- NAND Flash boot
  - 8-bit NAND
- NOR Flash boot
  - NOR Direct boot (8-bit)
  - NOR Legacy boot (8-bit)
  - NOR AIS boot (8-bit)
- I2C0 / I2C1 Boot
  - EEPROM (Master Mode)
  - External Host (Slave Mode)
- SPI0 / SPI1 Boot
  - Serial Flash (Master Mode)
  - SERIAL EEPROM (Master Mode)
  - External Host (Slave Mode)
- UART0 / UART1 / UART2 Boot
  - External Host

## 4.2 SYSCFG Module

The following system level features of the chip are controlled by the SYSCFG peripheral:

- Readable Device, Die, and Chip Revision ID
- Control of Pin Multiplexing
- Priority of bus accesses different bus masters in the system
- Capture at power on reset the chip BOOT[15:0] pin values and make them available to software
- Special case settings for peripherals:
  - Locking of PLL controller settings
  - Default burst sizes for EDMA3 TC0 and TC1
  - Selection of the source for the eCAP module input capture (including on chip sources)
  - McASP AMUTEIN selection and clearing of AMUTE status for the three McASP peripherals
  - Control of the reference clock source and other side-band signals for both of the integrated USB PHYs
  - Clock source selection for EMIFA and EMIFB
- Selects the source of emulation suspend signal of peripherals supporting this function.

Many registers are accessible only by a host (ARM) when it is operating in its privileged mode. (ex. from the kernel, but not from user space code).

**Table 4-1. System Configuration (SYSCFG) Module Register Access**

| BYTE ADDRESS | ACRONYM   | REGISTER DESCRIPTION                     | ACCESS          |
|--------------|-----------|------------------------------------------|-----------------|
| 0x01C1 4000  | REVID     | Revision Identification Register         | —               |
| 0x01C1 4008  | DIEIDR0   | Device Identification Register 0         | —               |
| 0x01C1 400C  | DIEIDR1   | Device Identification Register 1         | —               |
| 0x01C1 4010  | DIEIDR2   | Device Identification Register 2         | —               |
| 0x01C1 4014  | DIEIDR3   | Device Identification Register 3         | —               |
| 0x01C1 4018  | DEVIDR0   | JTAG Identification Register             | —               |
| 0x01C1 4020  | BOOTCFG   | Boot Configuration Register              | Privileged mode |
| 0x01C1 4024  | CHIPREVID | Silicon Revision Identification Register | Privileged mode |
| 0x01C1 4038  | KICK0R    | Kick 0 Register                          | Privileged mode |
| 0x01C1 403C  | KICK1R    | Kick 1 Register                          | Privileged mode |
| 0x01C1 4040  | HOST0CFG  | Host 0 Configuration Register            | —               |
| 0x01C1 4044  | HOST1CFG  | Host 1 Configuration Register            | —               |
| 0x01C1 40E0  | IRAWSTAT  | Interrupt Raw Status/Set Register        | Privileged mode |
| 0x01C1 40E4  | IENSTAT   | Interrupt Enable Status/Clear Register   | Privileged mode |
| 0x01C1 40E8  | IENSET    | Interrupt Enable Register                | Privileged mode |
| 0x01C1 40EC  | IENCLR    | Interrupt Enable Clear Register          | Privileged mode |
| 0x01C1 40F0  | EOI       | End of Interrupt Register                | Privileged mode |
| 0x01C1 40F4  | FLTADDRR  | Fault Address Register                   | Privileged mode |
| 0x01C1 40F8  | FLTSTAT   | Fault Status Register                    | —               |
| 0x01C1 4110  | MSTPRI0   | Master Priority 0 Register               | Privileged mode |
| 0x01C1 4114  | MSTPRI1   | Master Priority 1 Register               | Privileged mode |
| 0x01C1 4118  | MSTPRI2   | Master Priority 2 Register               | Privileged mode |
| 0x01C1 4120  | PINMUX0   | Pin Multiplexing Control 0 Register      | Privileged mode |
| 0x01C1 4124  | PINMUX1   | Pin Multiplexing Control 1 Register      | Privileged mode |
| 0x01C1 4128  | PINMUX2   | Pin Multiplexing Control 2 Register      | Privileged mode |
| 0x01C1 412C  | PINMUX3   | Pin Multiplexing Control 3 Register      | Privileged mode |
| 0x01C1 4130  | PINMUX4   | Pin Multiplexing Control 4 Register      | Privileged mode |
| 0x01C1 4134  | PINMUX5   | Pin Multiplexing Control 5 Register      | Privileged mode |

**Table 4-1. System Configuration (SYSCFG) Module Register Access (continued)**

| BYTE ADDRESS | ACRONYM  | REGISTER DESCRIPTION                 | ACCESS          |
|--------------|----------|--------------------------------------|-----------------|
| 0x01C1 4138  | PINMUX6  | Pin Multiplexing Control 6 Register  | Privileged mode |
| 0x01C1 413C  | PINMUX7  | Pin Multiplexing Control 7 Register  | Privileged mode |
| 0x01C1 4140  | PINMUX8  | Pin Multiplexing Control 8 Register  | Privileged mode |
| 0x01C1 4144  | PINMUX9  | Pin Multiplexing Control 9 Register  | Privileged mode |
| 0x01C1 4148  | PINMUX10 | Pin Multiplexing Control 10 Register | Privileged mode |
| 0x01C1 414C  | PINMUX11 | Pin Multiplexing Control 11 Register | Privileged mode |
| 0x01C1 4150  | PINMUX12 | Pin Multiplexing Control 12 Register | Privileged mode |
| 0x01C1 4154  | PINMUX13 | Pin Multiplexing Control 13 Register | Privileged mode |
| 0x01C1 4158  | PINMUX14 | Pin Multiplexing Control 14 Register | Privileged mode |
| 0x01C1 415C  | PINMUX15 | Pin Multiplexing Control 15 Register | Privileged mode |
| 0x01C1 4160  | PINMUX16 | Pin Multiplexing Control 16 Register | Privileged mode |
| 0x01C1 4164  | PINMUX17 | Pin Multiplexing Control 17 Register | Privileged mode |
| 0x01C1 4168  | PINMUX18 | Pin Multiplexing Control 18 Register | Privileged mode |
| 0x01C1 416C  | PINMUX19 | Pin Multiplexing Control 19 Register | Privileged mode |
| 0x01C1 4170  | SUSPSRC  | Suspend Source Register              | Privileged mode |
| 0x01C1 4174  | -        | Reserved                             | —               |
| 0x01C1 4178  | -        | Reserved                             | —               |
| 0x01C1 417C  | CFGCHIP0 | Chip Configuration 0 Register        | Privileged mode |
| 0x01C1 4180  | CFGCHIP1 | Chip Configuration 1 Register        | Privileged mode |
| 0x01C1 4184  | CFGCHIP2 | Chip Configuration 2 Register        | Privileged mode |
| 0x01C1 4188  | CFGCHIP3 | Chip Configuration 3 Register        | Privileged mode |
| 0x01C1 418C  | CFGCHIP4 | Chip Configuration 4 Register        | Privileged mode |

### 4.3 Pullup/Pulldown Resistors

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

An external pullup/pulldown resistor 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.

For the boot and configuration pins, 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 IO supply rail.
- For most systems, a 1-k $\Omega$  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 $\Omega$  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 device, see [Section 5.3](#), Recommended Operating Conditions.
- For the internal pullup/pulldown resistors for all device pins, see the peripheral/system-specific terminal functions table.

## 5 Device Operating Conditions

### 5.1 Absolute Maximum Ratings Over Operating Junction Temperature Range (Unless Otherwise Noted) <sup>(1)</sup>

|                                                          |                                                                                                                                                              |                                                   |
|----------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------|
| Supply voltage ranges                                    | Core<br>(CVDD, RVDD, PLL0_VDDA) <sup>(2)</sup>                                                                                                               | -0.5 V to 1.4 V                                   |
|                                                          | I/O, 1.8V<br>(USB0_VDDA18) <sup>(2)</sup>                                                                                                                    | -0.5 V to 2 V                                     |
|                                                          | I/O, 3.3V<br>(DVDD, USB0_VDDA33) <sup>(2)</sup>                                                                                                              | -0.5 V to 3.8V                                    |
| Input voltage ranges                                     | V <sub>I</sub> I/O, 1.2V<br>(OSCIN)                                                                                                                          | -0.3 V to CVDD + 0.3V                             |
|                                                          | V <sub>I</sub> I/O, 3.3V<br>(Steady State)                                                                                                                   | -0.3V to DVDD + 0.35V                             |
|                                                          | V <sub>I</sub> I/O, 3.3V<br>(Transient)                                                                                                                      | DVDD + 20%<br>up to 20% of Signal<br>Period       |
|                                                          | V <sub>I</sub> I/O, USB 5V Tolerant Pins:<br>(USB0_DM, USB0_DP)                                                                                              | 5.25V <sup>(3)</sup>                              |
| Output voltage ranges                                    | V <sub>O</sub> I/O, 3.3V<br>(Steady State)                                                                                                                   | -0.5 V to DVDD + 0.3V                             |
|                                                          | V <sub>O</sub> I/O, 3.3V<br>(Transient Overshoot/Undershoot)                                                                                                 | 20% of DVDD for up to<br>20% of the signal period |
| Clamp Current                                            | Input or Output Voltages 0.3V above or below their respective power rails. Limit clamp current that flows through the I/O's internal diode protection cells. | ±20mA                                             |
| Operating Junction Temperature ranges,<br>T <sub>J</sub> | Commercial (default)                                                                                                                                         | 0°C to 90°C                                       |
|                                                          | Industrial (D version)                                                                                                                                       | -40°C to 90°C                                     |
|                                                          | Extended (A version)                                                                                                                                         | -40°C to 105°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 VSS, PLL0\_VSSA, OSCVSS
- (3) Up to a max of 24 hours.

### 5.2 Handling Ratings

|                                                     |                                           |            | UNIT |
|-----------------------------------------------------|-------------------------------------------|------------|------|
| Storage temperature range, T <sub>stg</sub>         | (default)                                 | -55 to 150 | °C   |
| ESD Stress Voltage, V <sub>ESD</sub> <sup>(1)</sup> | Human Body Model (HBM) <sup>(2)</sup>     | >2000      | V    |
|                                                     | Charged Device Model (CDM) <sup>(3)</sup> | >500       | V    |

- (1) Electrostatic discharge (ESD) to measure device sensitivity/immunity to damage caused by electrostatic discharges into the device.
- (2) Level listed above is the passing level per ANSI/ESDA/JEDEC JS-001-2010. JEDEC document JEP155 states that 500V HBM allows safe manufacturing with a standard ESD control process, and manufacturing with less than 500V HBM is possible if necessary precautions are taken. Pins listed as 1000V may actually have higher performance.
- (3) Level listed above is the passing level per EIA-JEDEC JESD22-C101E. JEDEC document JEP157 states that 250V CDM allows safe manufacturing with a standard ESD control process. Pins listed as 250V may actually have higher performance.

### 5.3 Recommended Operating Conditions

|                                |                                                                                                   |                       | MIN      | NOM | MAX                        | UNIT |
|--------------------------------|---------------------------------------------------------------------------------------------------|-----------------------|----------|-----|----------------------------|------|
| CVDD                           | Supply voltage, Core (CVDD, PLL0_VDDA)                                                            | 375 MHz version       | 1.14     | 1.2 | 1.32                       | V    |
|                                |                                                                                                   | 456 MHz version       | 1.25     | 1.3 | 1.35                       | V    |
| RVDD                           | Supply Voltage, Internal RAM                                                                      | 375 MHz version       | 1.14     | 1.2 | 1.32                       | V    |
|                                |                                                                                                   | 456 MHz version       | 1.25     | 1.3 | 1.35                       | V    |
| DVDD                           | Supply voltage, I/O, 1.8V (USB0_VDDA18)                                                           |                       | 1.71     | 1.8 | 1.89                       | V    |
|                                | Supply voltage, I/O, 3.3V (DVDD, USB0_VDDA33)                                                     |                       | 3.0      | 3.3 | 3.45                       | V    |
| VSS                            | Supply ground (VSS, PLL0_VSSA, OSCVSS <sup>(1)</sup> )                                            |                       | 0        | 0   | 0                          | V    |
| V <sub>IH</sub> <sup>(2)</sup> | High-level input voltage, I/O, 3.3V                                                               |                       | 2        |     |                            | V    |
|                                | High-level input voltage, OSCIN                                                                   |                       | 0.7*CVDD |     |                            | V    |
| V <sub>IL</sub> <sup>(2)</sup> | Low-level input voltage, I/O, 3.3V                                                                |                       |          |     | 0.8                        | V    |
|                                | Low-level input voltage, OSCIN                                                                    |                       |          |     | 0.3*CVDD                   | V    |
| V <sub>HYS</sub>               | Input Hysteresis                                                                                  |                       |          | 160 |                            | mV   |
| t <sub>t</sub>                 | Transition time, 10%-90%, All Inputs (unless otherwise specified in the electrical data sections) |                       |          |     | 0.25P or 10 <sup>(3)</sup> | ns   |
| F <sub>SYSClk6</sub>           | ARM Operating Frequency (SYSClk6)                                                                 | Commercial (default)  | 0        |     | 375 (1.2V)<br>456 (1.3V)   | MHz  |
|                                |                                                                                                   | Industrial (D suffix) | 0        |     | 456 (1.3V)                 | MHz  |
|                                |                                                                                                   | Extended (A suffix)   | 0        |     | 375(1.2V)                  | MHz  |

- (1) When an external crystal is used, oscillator (OSC\_VSS) ground must be kept separate from other grounds and connected directly to the crystal load capacitor ground. These pins are shorted to VSS on the device itself and should not be connected to VSS on the circuit board. If a crystal is not used and the clock input is driven directly, then the oscillator VSS may be connected to board ground.
- (2) Unless specifically indicated, these I/O specifications do not apply to USB I/Os. USB0 I/Os adhere to USB2.0 specification.
- (3) Whichever is smaller. P = the period of the applied signal. Maintaining transition times as fast as possible is recommended to improve noise immunity on input signals.

## 5.4 Notes on Recommended Power-On Hours (POH)

The information in the section below is provided solely for your convenience and does not extend or modify the warranty provided under TI's standard terms and conditions for TI semiconductor products.

To avoid significant degradation, the device power-on hours (POH) must be limited to the following:

**Table 5-1. Recommended Power-On Hours**

| Silicon Revision | Speed Grade | Operating Junction Temperature (Tj) | Nominal CVDD Voltage (V) | Power-On Hours [POH] (hours) |
|------------------|-------------|-------------------------------------|--------------------------|------------------------------|
| D                | 375 MHz     | 0 to 90 °C                          | 1.2V                     | 100,000                      |
| D                | 375 MHz     | -40 to 105 °C                       | 1.2V                     | 75,000 <sup>(1)</sup>        |
| D                | 456 MHz     | 0 to 90 °C                          | 1.3V                     | 100,000                      |
| D                | 456 MHz     | -40 to 90 °C                        | 1.3V                     | 100,000                      |

(1) 100,000 POH can be achieved at this temperature condition if the device operation is limited to 345 MHz.

**Note:** Logic functions and parameter values are not assured out of the range specified in the recommended operating conditions.

**The above notations cannot be deemed a warranty or deemed to extend or modify the warranty under TI's standard terms and conditions for TI semiconductor products.**

## 5.5 Electrical Characteristics Over Recommended Ranges of Supply Voltage and Operating Junction Temperature (Unless Otherwise Noted)

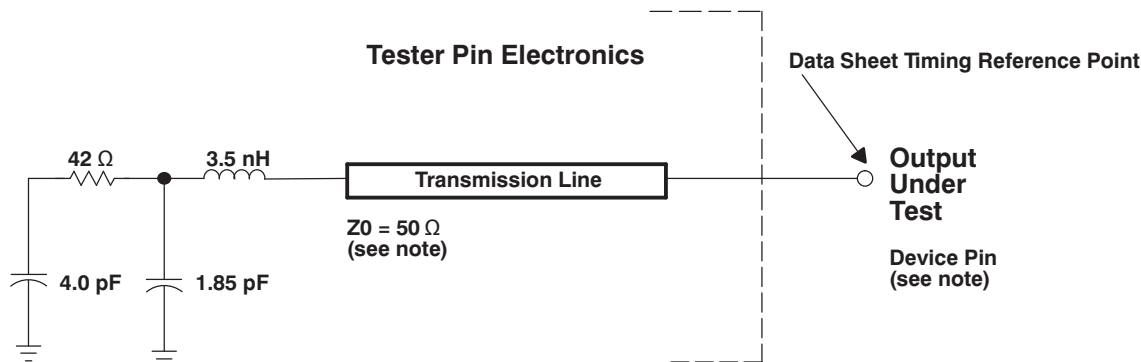
| PARAMETER      |                                      | TEST CONDITIONS                                                                | MIN  | TYP | MAX      | UNIT    |
|----------------|--------------------------------------|--------------------------------------------------------------------------------|------|-----|----------|---------|
| $V_{OH}^{(1)}$ | High-level output voltage (3.3V I/O) | DVDD = 3.15V, $I_{OH} = -4$ mA                                                 | 2.4  |     |          | V       |
|                |                                      | DVDD = 3.15V, $I_{OH} = 100$ $\mu$ A                                           | 2.95 |     |          | V       |
| $V_{OL}^{(1)}$ | Low-level output voltage (3.3V I/O)  | DVDD = 3.15V, $I_{OL} = 4$ mA                                                  |      |     | 0.4      | V       |
|                |                                      | DVDD = 3.15V, $I_{OL} = -100$ $\mu$ A                                          |      |     | 0.2      | V       |
| $I_I^{(2)(1)}$ | Input current                        | $V_I = V_{SS}$ to DVDD without opposing internal resistor                      |      |     | $\pm 35$ | $\mu$ A |
|                |                                      | $V_I = V_{SS}$ to DVDD with opposing internal pullup resistor <sup>(3)</sup>   | -30  |     | -200     | $\mu$ A |
|                |                                      | $V_I = V_{SS}$ to DVDD with opposing internal pulldown resistor <sup>(3)</sup> | 50   |     | 300      | $\mu$ A |
| $I_{OH}^{(1)}$ | High-level output current            |                                                                                |      |     | -4       | mA      |
| $I_{OL}^{(1)}$ | Low-level output current             |                                                                                |      |     | 4        | mA      |
| $I_{OZ}^{(4)}$ | I/O Off-state output current         | VO = VDD or VSS; Internal pull disabled                                        |      |     | $\pm 35$ | $\mu$ A |
| $C_I$          | Input capacitance                    | LVC MOS signals                                                                |      |     | 3        | pF      |
|                |                                      | OSCIN                                                                          |      |     | 2        | pF      |
| $C_O$          | Output capacitance                   | LVC MOS signals                                                                |      |     | 3        | pF      |

- (1) These I/O specifications apply to regular 3.3V I/Os and do not apply to USB0 unless specifically indicated. USB0 I/Os adhere to the USB 2.0 specification.
- (2)  $I_I$  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.
- (3) Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.
- (4)  $I_{OZ}$  applies to output-only pins, indicating off-state (Hi-Z) output leakage current.

## 6 Peripheral Information and Electrical Specifications

### 6.1 Parameter Information

#### 6.1.1 Parameter Information Device-Specific Information



- A. The data sheet provides timing at the device pin. For output timing analysis, the tester pin electronics and its transmission line effects must be taken into account. A transmission line with a delay of 2 ns or longer can be used to produce the desired transmission line effect. The transmission line is intended as a load only. It is not necessary to add or subtract the transmission line delay (2 ns or longer) from the data sheet timings. Input requirements in this data sheet are tested with an input slew rate of < 4 Volts per nanosecond (4 V/ns) at the device pin and the input signals are driven between 0V and the appropriate IO supply rail for the signal.

**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.

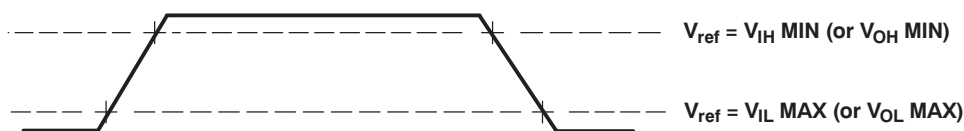
##### 6.1.1.1 Signal Transition Levels

All input and output timing parameters are referenced to  $V_{ref}$  for both "0" and "1" logic levels. For 3.3 V I/O,  $V_{ref} = 1.65$  V. For 1.8 V I/O,  $V_{ref} = 0.9$  V. For 1.2 V I/O,  $V_{ref} = 0.6$  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**

## 6.2 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

### 6.3.1 Power-on Sequence

The device should be powered-on in the following order:

1. Core logic supplies:
  - (a) CVDD core logic supply
  - (b) Other 1.2V logic supplies (PLL0\_VDDA). Groups 1a) and 1b) may be powered up together or 1a) first followed by 1b).
2. All 1.8V IO supplies (USB0\_VDDA18).
3. All digital IO and analog 3.3V PHY supplies (DVDD, USB0\_VDDA33 ).

Group 2) and group 3) may be powered on in either order [2 then 3, or 3 then 2] but group 3) must be powered-on after the core logic supplies.

There is no specific required voltage ramp rate for any of the supplies.

RESET must be maintained active until all power supplies have reached their nominal values.

### 6.3.2 Power-off Sequence

The power supplies can be powered-off in any order as long as the 3.3V supplies do not remain powered with the other supplies unpowered.

## 6.4 Reset

### 6.4.1 Power-On Reset (POR)

A power-on reset (POR) is required to place the device in a known good state after power-up. Power-On Reset is initiated by bringing  $\overline{\text{RESET}}$  and  $\overline{\text{TRST}}$  low at the same time. POR sets all of the device internal logic to its default state. All pins are tri-stated with the exception of RTCK/GP7[14]. If an emulator is driving TCK into the device during reset, then RTCK/GP7[14] will drive out RTCK. If TCK is not being driven into the device during reset, then RTCK/GP7[14] will drive low.

While both  $\overline{\text{TRST}}$  and  $\overline{\text{RESET}}$  need to be asserted upon power up, only  $\overline{\text{RESET}}$  needs to be released for the device to boot properly.  $\overline{\text{TRST}}$  may be asserted indefinitely for normal operation, keeping the JTAG port interface and device's emulation logic in the reset state.

$\overline{\text{TRST}}$  only needs to be released when it is necessary to use a JTAG controller to debug the 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.

$\overline{\text{RESET}}$  must be released only in order for boundary-scan JTAG to read the variant field of IDCODE correctly. Other boundary-scan instructions work correctly independent of current state of  $\overline{\text{RESET}}$ . For maximum reliability, the device includes an internal pulldown 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.

RTCK/GP7[14] is maintained active through a POR.

A summary of the effects of Power-On Reset is given below:

- All internal logic (including emulation logic and the PLL logic) is reset to its default state
- Internal memory is not maintained through a POR
- All device pins go to a high-impedance state

**CAUTION: A watchdog reset triggers a POR.**

### 6.4.2 Warm Reset

A warm reset provides a limited reset to the device. Warm Reset is initiated by bringing only  $\overline{\text{RESET}}$  low ( $\overline{\text{TRST}}$  is maintained high through a warm reset). Warm reset sets certain portions of the device to their default state while leaving others unaltered. All pins are 3-stated with the exception of RTCK/GP7[14]. If an emulator is driving TCK into the device during reset, then RTCK/GP7[14] will drive out RTCK. If TCK is not being driven into the device during reset, then RTCK/GP7[14] will drive low.

During emulation, the emulator will maintain  $\overline{\text{TRST}}$  high and hence only warm reset (not POR) is available during emulation debug and development.

RTCK/GP7[14] is maintained active through a warm reset.

A summary of the effects of Warm Reset is given below:

- All internal logic (except for the emulation logic and the PLL logic) is reset to its default state
- Internal memory is maintained through a warm reset
- All device pins go to a high-impedance state

### 6.4.3 Reset Electrical Data Timings

Table 6-1 assumes testing over the recommended operating conditions.

Table 6-1. Reset Timing Requirements<sup>(1)</sup>

| N<br>O. |                    |                                                                            | MIN | MAX | UNIT |
|---------|--------------------|----------------------------------------------------------------------------|-----|-----|------|
| 1       | $t_{w(RSTL)}$      | Pulse width, $\overline{RESET}/\overline{TRST}$ low                        | 100 |     | ns   |
| 2       | $t_{su(BPV-RSTH)}$ | Setup time, boot pins valid before $\overline{RESET}/\overline{TRST}$ high | 20  |     | ns   |
| 3       | $t_{h(RSTH-BPV)}$  | Hold time, boot pins valid after $\overline{RESET}/\overline{TRST}$ high   | 20  |     | ns   |

(1) For power-on reset (POR), the reset timings in this table refer to  $\overline{RESET}$  and  $\overline{TRST}$  together. For warm reset, the reset timings in this table refer to  $\overline{RESET}$  only ( $\overline{TRST}$  is held high).

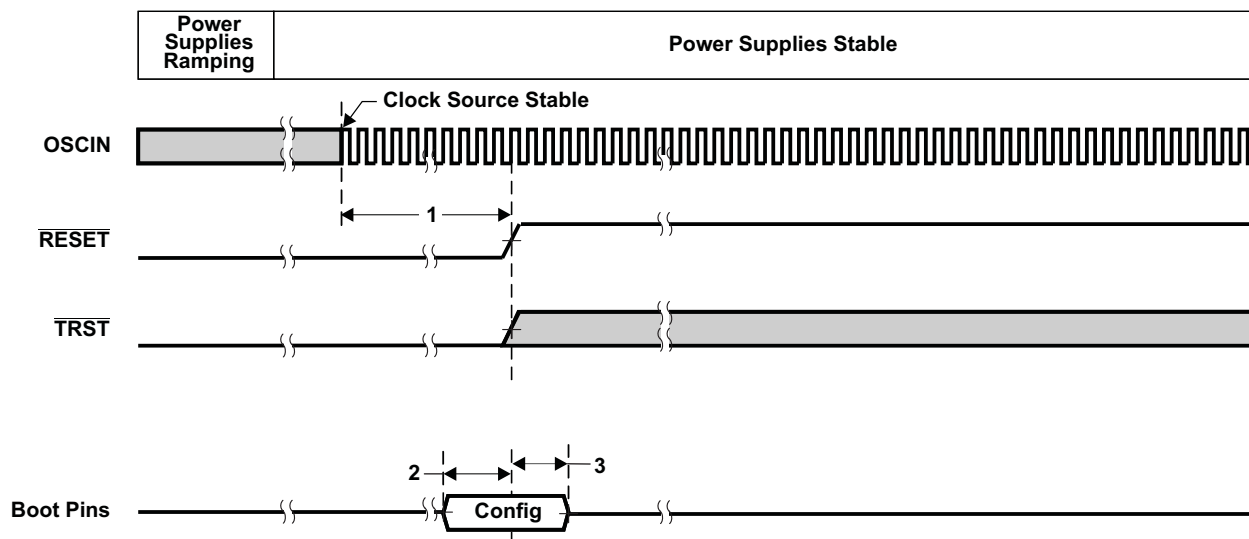


Figure 6-4. Power-On Reset (RESET and TRST active) Timing

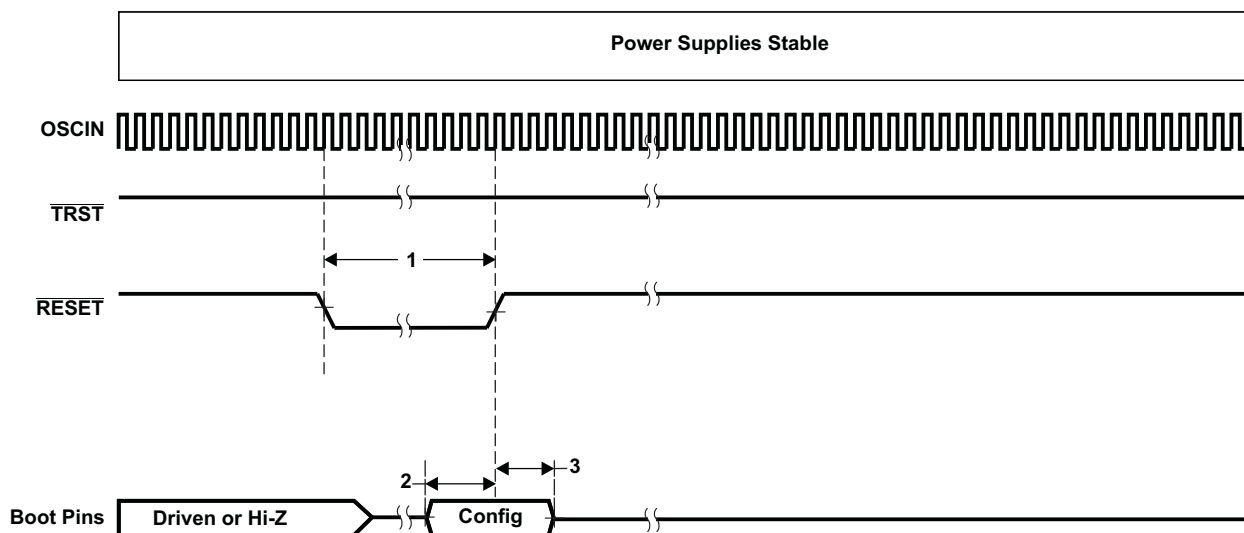


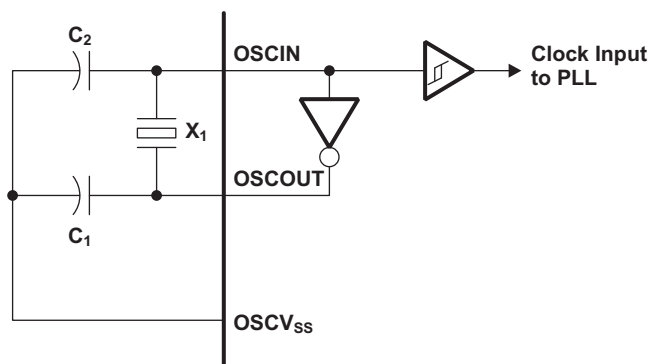
Figure 6-5. Warm Reset (RESET active, TRST high) Timing

## 6.5 Crystal Oscillator or External Clock Input

The device includes two choices to provide an external clock input, which is fed to the on-chip PLL to generate high-frequency system clocks. These options are illustrated in [Figure 6-6](#) and [Figure 6-7](#). For input clock frequencies between 12 and 20 MHz, a crystal with 80 ohm max ESR is recommended. For input clock frequencies between 20 and 30 MHz, a crystal with 60 ohm max ESR is recommended. Typical load capacitance values are 10-20 pF, where the load capacitance is the series combination of C1 and C2.

The CLKMODE bit in the PLLCTL register must be 0 to use the on-chip oscillator. If CLKMODE is set to 1, the internal oscillator is disabled.

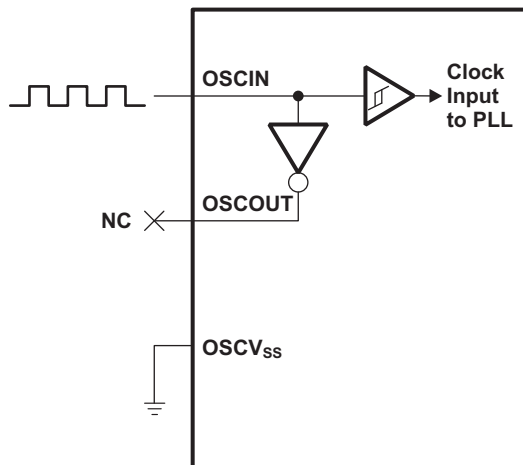
- [Figure 6-6](#) illustrates the option that uses on-chip 1.2V oscillator with external crystal circuit.
- [Figure 6-7](#) illustrates the option that uses an external 1.2V clock input.



**Figure 6-6. On-Chip 1.2V Oscillator**

**Table 6-2. Oscillator Timing Requirements**

| PARAMETER |                                           | MIN | MAX | UNIT |
|-----------|-------------------------------------------|-----|-----|------|
| $f_{osc}$ | Oscillator frequency range (OSCIN/OSCOUT) | 12  | 30  | MHz  |



**Figure 6-7. External 1.2V Clock Source**

**Table 6-3. OSCIN Timing Requirements**

| PARAMETER                     |                                            | MIN                              | MAX                   | UNIT |
|-------------------------------|--------------------------------------------|----------------------------------|-----------------------|------|
| $f_{\text{OSCIN}}$            | OSCIN frequency range (OSCIN)              | 12                               | 50                    | MHz  |
| $t_{\text{c}}(\text{OSCIN})$  | Cycle time, external clock driven on OSCIN | 20                               |                       | ns   |
| $t_{\text{w}}(\text{OSCINH})$ | Pulse width high, external clock on OSCIN  | $0.4 t_{\text{c}}(\text{OSCIN})$ |                       | ns   |
| $t_{\text{w}}(\text{OSCINL})$ | Pulse width low, external clock on OSCIN   | $0.4 t_{\text{c}}(\text{OSCIN})$ |                       | ns   |
| $t_{\text{t}}(\text{OSCIN})$  | Transition time, OSCIN                     |                                  | $0.25P$ or $10^{(1)}$ | ns   |
| $t_{\text{j}}(\text{OSCIN})$  | Period jitter, OSCIN                       |                                  | $0.02P$               | ns   |

(1) Whichever is smaller. P = the period of the applied signal. Maintaining transition times as fast as possible is recommended to improve noise immunity on input signals.

## 6.6 Clock PLLs

The device has one PLL controller that provides clock to different parts of the system. PLL0 provides clocks (though various dividers) to most of the components of the device.

The PLL controller provides the following:

- Glitch-Free Transitions (on changing clock settings)
- Domain Clocks Alignment
- Clock Gating
- PLL power down

The various clock outputs given by the controller are as follows:

- Domain Clocks: SYSCLK [1:n]
- Auxiliary Clock from reference clock source: AUXCLK

Various dividers that can be used are as follows:

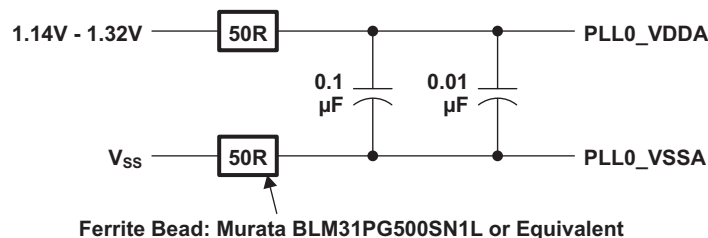
- Post-PLL Divider: POSTDIV
- SYSCLK Divider: D1,  $\frac{1}{4}$ , Dn

Various other controls supported are as follows:

- PLL Multiplier Control: PLLM
- Software programmable PLL Bypass: PLEN

### 6.6.1 PLL Device-Specific Information

The PLL requires some external filtering components to reduce power supply noise as shown in [Figure 6-8](#).



**Figure 6-8. PLL External Filtering Components**

The input to the PLL is either from the on-chip oscillator (OSCIN pin) or from an external clock on the OSCIN pin. The PLL outputs seven clocks that have programmable divider options. [Figure 6-9](#) illustrates the PLL Topology.

The PLL is disabled by default after a device reset. It must be configured by software according to the allowable operating conditions listed in [Table 6-4](#) before enabling the processor to run from the PLL by setting PLEN = 1.

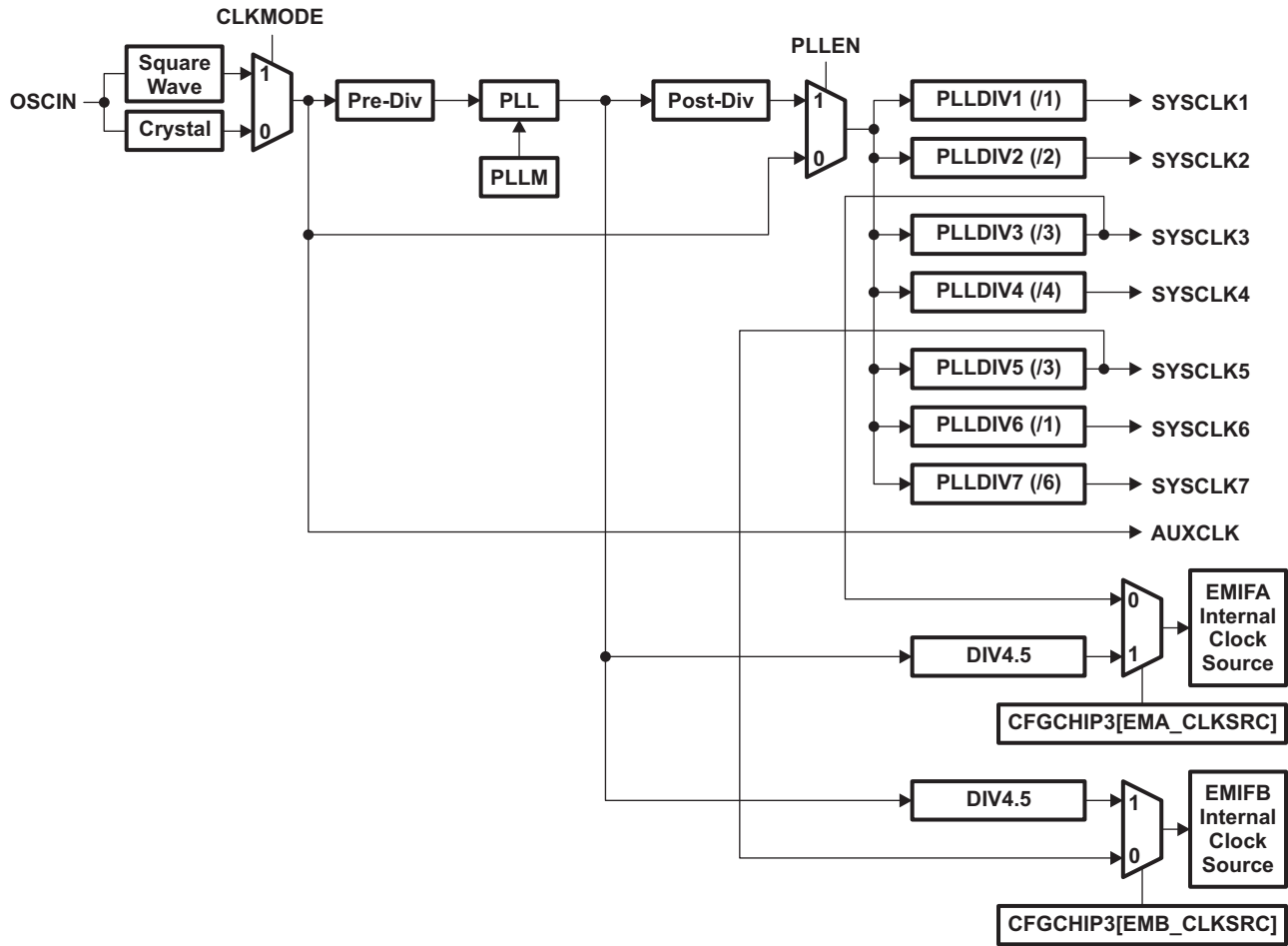


Figure 6-9. PLL Topology

**Table 6-4. Allowed PLL Operating Conditions**

| No. | PARAMETER                                                                                                                                                  | Default Value | MIN  | MAX                                                                                                    | UNIT         |
|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|------|--------------------------------------------------------------------------------------------------------|--------------|
| 1   | PLL <sub>RST</sub> : Assertion time during initialization                                                                                                  | N/A           | 1000 | N/A                                                                                                    | ns           |
| 2   | Lock time: The time that the application has to wait for the PLL to acquire locks before setting PLL <sub>EN</sub> , after changing PREDIV, PLLM, or OSCIN | N/A           | N/A  | $\text{Max PLL Lock Time} = \frac{2000 N}{\sqrt{M}}$ where N = Pre-Divider Ratio<br>M = PLL Multiplier | OSCIN cycles |
| 3   | PREDIV                                                                                                                                                     | /1            | /1   | /32                                                                                                    |              |
| 4   | PLL input frequency ( PLLREF )                                                                                                                             |               | 12   | 30 (if internal oscillator is used)<br>50 (if external clock source is used)                           | MHz          |
| 5   | PLL multiplier values (PLLM) <sup>(1)</sup>                                                                                                                | x20           | x4   | x32                                                                                                    |              |
| 6   | PLL output frequency. ( PLLOUT )                                                                                                                           | N/A           | 300  | 600                                                                                                    | MHz          |
| 7   | POSTDIV                                                                                                                                                    | /1            | /1   | /32                                                                                                    |              |

- (1) The multiplier values must be chosen such that the PLL output frequency (at PLL<sub>OUT</sub>) is between 300 and 600 MHz, but the frequency going into the SYSCLK dividers (after the post divider) cannot exceed the maximum clock frequency defined for the device at a given voltage operating point.

### 6.6.2 Device Clock Generation

PLL0 is controlled by PLL Controller 0. The PLLC0 manages the clock ratios, alignment, and gating for the system clocks to the chip. The PLLC is responsible for controlling all modes of the PLL through software, in terms of pre-division of the clock inputs, multiply factor within the PLL, and post-division for each of the chip-level clocks from the PLL output. The PLLC also controls reset propagation through the chip, clock alignment, and test points.

### 6.6.3 PLL Controller 0 Registers

**Table 6-5. PLL Controller 0 Registers**

| BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION                        |
|--------------|---------|---------------------------------------------|
| 0x01C1 1000  | REVID   | Revision Identification Register            |
| 0x01C1 10E4  | RSTYPE  | Reset Type Status Register                  |
| 0x01C1 1100  | PLLCTL  | PLL Control Register                        |
| 0x01C1 1104  | -       | Reserved                                    |
| 0x01C1 1110  | PLLM    | PLL Multiplier Control Register             |
| 0x01C1 1114  | PREDIV  | PLL Pre-Divider Control Register            |
| 0x01C1 1118  | PLLDIV1 | PLL Controller Divider 1 Register           |
| 0x01C1 111C  | PLLDIV2 | PLL Controller Divider 2 Register           |
| 0x01C1 1120  | PLLDIV3 | PLL Controller Divider 3 Register           |
| 0x01C1 1124  | -       | Reserved                                    |
| 0x01C1 1128  | POSTDIV | PLL Post-Divider Control Register           |
| 0x01C1 1138  | PLLCMD  | PLL Controller Command Register             |
| 0x01C1 113C  | PLLSTAT | PLL Controller Status Register              |
| 0x01C1 1140  | ALNCTL  | PLL Controller Clock Align Control Register |
| 0x01C1 1144  | DCHANGE | PLLDIV Ratio Change Status Register         |
| 0x01C1 1148  | CKEN    | Clock Enable Control Register               |
| 0x01C1 114C  | CKSTAT  | Clock Status Register                       |
| 0x01C1 1150  | SYSTAT  | SYSCLK Status Register                      |
| 0x01C1 1160  | PLLDIV4 | PLL Controller Divider 4 Register           |
| 0x01C1 1164  | PLLDIV5 | PLL Controller Divider 5 Register           |
| 0x01C1 1168  | PLLDIV6 | PLL Controller Divider 6 Register           |
| 0x01C1 116C  | PLLDIV7 | PLL Controller Divider 7 Register           |

## 6.7 Interrupts

### 6.7.1 ARM CPU Interrupts

The ARM9 CPU core supports 2 direct interrupts: FIQ and IRQ. The ARM Interrupt Controller extends the number of interrupts to 100, and provides features like programmable masking, priority, hardware nesting support, and interrupt vector generation.

#### 6.7.1.1 ARM Interrupt Controller (AINTC) Interrupt Signal Hierarchy

The ARM Interrupt controller organizes interrupts into the following hierarchy:

- Peripheral Interrupt Requests
  - Individual Interrupt Sources from Peripherals
- 100 System Interrupts
  - One or more Peripheral Interrupt Requests are combined (fixed configuration) to generate a System Interrupt.
  - After prioritization, the AINTC will provide an interrupt vector based unique to each System Interrupt
- 32 Interrupt Channels
  - Each System Interrupt is mapped to one of the 32 Interrupt Channels
  - Channel Number determines the first level of prioritization, Channel 0 is highest priority and 31 lowest.
  - If more than one system interrupt is mapped to a channel, priority within the channel is determined by system interrupt number (0 highest priority)
- Host Interrupts (FIQ and IRQ)
  - Interrupt Channels 0 and 1 generate the ARM FIQ interrupt
  - Interrupt Channels 2 through 31 Generate the ARM IRQ interrupt
- Debug Interrupts
  - Two Debug Interrupts are supported and can be used to trigger events in the debug subsystem
  - Sources can be selected from any of the System Interrupts or Host Interrupts

#### 6.7.1.2 AINTC Hardware Vector Generation

The AINTC also generates an interrupt vector in hardware for both IRQ and FIQ host interrupts. This may be used to accelerate interrupt dispatch. A unique vector is generated for each of the 100 system interrupts. The vector is computed in hardware as:

$$\text{VECTOR} = \text{BASE} + (\text{SYSTEM INTERRUPT NUMBER} \times \text{SIZE})$$

Where BASE and SIZE are programmable. The computed vector is a 32-bit address which may be dispatched to using a single instruction of type LDR PC, [PC, #-<offset\_12>] at the FIQ and IRQ vector locations (0xFFFF0018 and 0xFFFF001C respectively).

#### 6.7.1.3 AINTC Hardware Interrupt Nesting Support

Interrupt nesting occurs when an interrupt service routine re-enables interrupts, to allow the CPU to interrupt the ISR if a higher priority event occurs. The AINTC provides hardware support to facilitate interrupt nesting. It supports both global and per host interrupt (FIQ and IRQ in this case) automatic nesting. If enabled, the AINTC will automatically update an internal nesting register that temporarily masks interrupts at and below the priority of the current interrupt channel. Then if the ISR re-enables interrupts; only higher priority channels will be able to interrupt it. The nesting level is restored by the ISR by writing to the nesting level register on completion. Support for nesting can be enabled/disabled by software, with the option of automatic nesting on a global or per host interrupt basis; or manual nesting.

#### 6.7.1.4 AINTC System Interrupt Assignments on the device

System Interrupt assignments for the device are listed in [Table 6-6](#)

**Table 6-6. AINTC System Interrupt Assignments**

| System Interrupt | Interrupt Name     | Source                                                   |
|------------------|--------------------|----------------------------------------------------------|
| 0                | COMMTX             | ARM                                                      |
| 1                | COMMRX             | ARM                                                      |
| 2                | NINT               | ARM                                                      |
| 3                | PRU_EVTOUT0        | PRUSS Interrupt                                          |
| 4                | PRU_EVTOUT1        | PRUSS Interrupt                                          |
| 5                | PRU_EVTOUT2        | PRUSS Interrupt                                          |
| 6                | PRU_EVTOUT3        | PRUSS Interrupt                                          |
| 7                | PRU_EVTOUT4        | PRUSS Interrupt                                          |
| 8                | PRU_EVTOUT5        | PRUSS Interrupt                                          |
| 9                | PRU_EVTOUT6        | PRUSS Interrupt                                          |
| 10               | PRU_EVTOUT7        | PRUSS Interrupt                                          |
| 11               | EDMA3_CC0_CCINT    | EDMA CC Region 0                                         |
| 12               | EDMA3_CC0_CCERRINT | EDMA Channel Controller                                  |
| 13               | EDMA3_TC0_TCERRINT | EDMA Transfer Controller 0                               |
| 14               | EMIFA_INT          | EMIFA                                                    |
| 15               | IIC0_INT           | I2C0                                                     |
| 16               | MMCS0_INT0         | MMCS0                                                    |
| 17               | MMCS0_INT1         | MMCS0                                                    |
| 18               | PSC0_ALLINT        | PSC0                                                     |
| 19               | -                  | Reserved                                                 |
| 20               | SPI0_INT           | SPI0                                                     |
| 21               | T64P0_TINT12       | Timer64P0 Interrupt 12                                   |
| 22               | T64P0_TINT34       | Timer64P0 Interrupt 34                                   |
| 23               | T64P1_TINT12       | Timer64P1 Interrupt 12                                   |
| 24               | T64P1_TINT34       | Timer64P1 Interrupt 34                                   |
| 25               | UART0_INT          | UART0                                                    |
| 26               | -                  | Reserved                                                 |
| 27               | MPU_BOOTCFG_ERR    | Shared MPU and SYSCFG Address/Protection Error Interrupt |
| 28 - 31          | -                  | Reserved                                                 |
| 32               | EDMA3_TC1_TCERRINT | EDMA Transfer Controller 1                               |
| 33               | EMAC_C0RXTHRESH    | EMAC - Core 0 Receive Threshold Interrupt                |
| 34               | EMAC_C0RX          | EMAC - Core 0 Receive Interrupt                          |
| 35               | EMAC_C0TX          | EMAC - Core 0 Transmit Interrupt                         |
| 36               | EMAC_C0MISC        | EMAC - Core 0 Miscellaneous Interrupt                    |
| 37               | EMAC_C1RXTHRESH    | EMAC - Core 1 Receive Threshold Interrupt                |
| 38               | EMAC_C1RX          | EMAC - Core 1 Receive Interrupt                          |
| 39               | EMAC_C1TX          | EMAC - Core 1 Transmit Interrupt                         |
| 40               | EMAC_C1MISC        | EMAC - Core 1 Miscellaneous Interrupt                    |
| 41               | EMIF_MEMERR        | EMIFB                                                    |
| 42               | GPIO_B0INT         | GPIO Bank 0 Interrupt                                    |
| 43               | GPIO_B1INT         | GPIO Bank 1 Interrupt                                    |
| 44               | GPIO_B2INT         | GPIO Bank 2 Interrupt                                    |
| 45               | GPIO_B3INT         | GPIO Bank 3 Interrupt                                    |
| 46               | GPIO_B4INT         | GPIO Bank 4 Interrupt                                    |
| 47               | GPIO_B5INT         | GPIO Bank 5 Interrupt                                    |
| 48               | GPIO_B6INT         | GPIO Bank 6 Interrupt                                    |

**Table 6-6. AINTC System Interrupt Assignments (continued)**

| System Interrupt | Interrupt Name | Source                                   |
|------------------|----------------|------------------------------------------|
| 49               | GPIO_B7INT     | GPIO Bank 7 Interrupt                    |
| 50               | -              | Reserved                                 |
| 51               | IIC1_INT       | I2C1                                     |
| 52               | -              | Reserved                                 |
| 53               | UART_INT1      | UART1                                    |
| 54               | MCASP_INT      | McASP0, 1, 2 Combined RX / TX Interrupts |
| 55               | PSC1_ALLINT    | PSC1                                     |
| 56               | SPI1_INT       | SPI1                                     |
| 57               | -              | Reserved                                 |
| 58               | USB0_INT       | USB0 Interrupt                           |
| 59 - 60          | -              | Reserved                                 |
| 61               | UART2_INT      | UART2                                    |
| 62               | -              | Reserved                                 |
| 63               | EHRPWM0        | HiResTimer / PWM0 Interrupt              |
| 64               | EHRPWM0TZ      | HiResTimer / PWM0 Trip Zone Interrupt    |
| 65               | EHRPWM1        | HiResTimer / PWM1 Interrupt              |
| 66               | EHRPWM1TZ      | HiResTimer / PWM1 Trip Zone Interrupt    |
| 67               | EHRPWM2        | HiResTimer / PWM2 Interrupt              |
| 68               | EHRPWM2TZ      | HiResTimer / PWM2 Trip Zone Interrupt    |
| 69               | ECAP0          | ECAP0                                    |
| 70               | ECAP1          | ECAP1                                    |
| 71               | ECAP2          | ECAP2                                    |
| 72               | EQEP0          | EQEP0                                    |
| 73               | EQEP1          | EQEP1                                    |
| 74               | T64P0_CMPINT0  | Timer64P0 - Compare 0                    |
| 75               | T64P0_CMPINT1  | Timer64P0 - Compare 1                    |
| 76               | T64P0_CMPINT2  | Timer64P0 - Compare 2                    |
| 77               | T64P0_CMPINT3  | Timer64P0 - Compare 3                    |
| 78               | T64P0_CMPINT4  | Timer64P0 - Compare 4                    |
| 79               | T64P0_CMPINT5  | Timer64P0 - Compare 5                    |
| 80               | T64P0_CMPINT6  | Timer64P0 - Compare 6                    |
| 81               | T64P0_CMPINT7  | Timer64P0 - Compare 7                    |
| 82               | T64P1_CMPINT0  | Timer64P1 - Compare 0                    |
| 83               | T64P1_CMPINT1  | Timer64P1 - Compare 1                    |
| 84               | T64P1_CMPINT2  | Timer64P1 - Compare 2                    |
| 85               | T64P1_CMPINT3  | Timer64P1 - Compare 3                    |
| 86               | T64P1_CMPINT4  | Timer64P1 - Compare 4                    |
| 87               | T64P1_CMPINT5  | Timer64P1 - Compare 5                    |
| 88               | T64P1_CMPINT6  | Timer64P1 - Compare 6                    |
| 89               | T64P1_CMPINT7  | Timer64P1 - Compare 7                    |
| 90               | ARMCLKSTOPREQ  | PSC0                                     |
| 91 - 100         | -              | Reserved                                 |

### 6.7.1.5 AINTC Memory Map

**Table 6-7. AINTC Memory Map**

| BYTE ADDRESS                | ACRONYM             | REGISTER DESCRIPTION                              |
|-----------------------------|---------------------|---------------------------------------------------|
| 0xFFFFE E000                | REV                 | Revision Register                                 |
| 0xFFFFE E004                | CR                  | Control Register                                  |
| 0xFFFFE E008 - 0xFFFFE E00F | -                   | Reserved                                          |
| 0xFFFFE E010                | GER                 | Global Enable Register                            |
| 0xFFFFE E014 - 0xFFFFE E01B | -                   | Reserved                                          |
| 0xFFFFE E01C                | GNLR                | Global Nesting Level Register                     |
| 0xFFFFE E020                | SISR                | System Interrupt Status Indexed Set Register      |
| 0xFFFFE E024                | SICR                | System Interrupt Status Indexed Clear Register    |
| 0xFFFFE E028                | EISR                | System Interrupt Enable Indexed Set Register      |
| 0xFFFFE E02C                | EICR                | System Interrupt Enable Indexed Clear Register    |
| 0xFFFFE E030                | -                   | Reserved                                          |
| 0xFFFFE E034                | HIEISR              | Host Interrupt Enable Indexed Set Register        |
| 0xFFFFE E038                | HIEICR              | Host Interrupt Enable Indexed Clear Register      |
| 0xFFFFE E03C - 0xFFFFE E04F | -                   | Reserved                                          |
| 0xFFFFE E050                | VBR                 | Vector Base Register                              |
| 0xFFFFE E054                | VSR                 | Vector Size Register                              |
| 0xFFFFE E058                | VNR                 | Vector Null Register                              |
| 0xFFFFE E05C - 0xFFFFE E07F | -                   | Reserved                                          |
| 0xFFFFE E080                | GPIR                | Global Prioritized Index Register                 |
| 0xFFFFE E084                | GPVR                | Global Prioritized Vector Register                |
| 0xFFFFE E088 - 0xFFFFE E1FF | -                   | Reserved                                          |
| 0xFFFFE E200 - 0xFFFFE E20B | SRSR[1] - SRSR[3]   | System Interrupt Status Raw / Set Registers       |
| 0xFFFFE E20C - 0xFFFFE E27F | -                   | Reserved                                          |
| 0xFFFFE E280 - 0xFFFFE E28B | SECR[1] - SECR[3]   | System Interrupt Status Enabled / Clear Registers |
| 0xFFFFE E28C - 0xFFFFE E2FF | -                   | Reserved                                          |
| 0xFFFFE E300 - 0xFFFFE E30B | ESR[1] - ESR[3]     | System Interrupt Enable Set Registers             |
| 0xFFFFE E30C - 0xFFFFE E37F | -                   | Reserved                                          |
| 0xFFFFE E380 - 0xFFFFE E38B | ECR[1] - ECR[3]     | System Interrupt Enable Clear Registers           |
| 0xFFFFE E38C - 0xFFFFE E3FF | -                   | Reserved                                          |
| 0xFFFFE E400 - 0xFFFFE E458 | CMR[0] - CMR[22]    | Channel Map Registers (Byte Wide Registers)       |
| 0xFFFFE E459 - 0xFFFFE E7FF | -                   | Reserved                                          |
| 0xFFFFE E800 - 0xFFFFE E81F | -                   | Reserved                                          |
| 0xFFFFE E820 - 0xFFFFE E8FF | -                   | Reserved                                          |
| 0xFFFFE E900 - 0xFFFFE E904 | HIPIR[1] - HIPIR[2] | Host Interrupt Prioritized Index Registers        |
| 0xFFFFE E908 - 0xFFFFE EEFF | -                   | Reserved                                          |
| 0xFFFFE EF00 - 0xFFFFE EF04 | -                   | Reserved                                          |
| 0xFFFFE EF08 - 0xFFFFE F0FF | -                   | Reserved                                          |
| 0xFFFFE F100 - 0xFFFFE F104 | HINLR[1] - HINLR[2] | Host Interrupt Nesting Level Registers            |
| 0xFFFFE F108 - 0xFFFFE F4FF | -                   | Reserved                                          |
| 0xFFFFE F500                | HIER                | Host Interrupt Enable Register                    |
| 0xFFFFE F504 - 0xFFFFE F5FF | -                   | Reserved                                          |
| 0xFFFFE F600                | HIPVR[1] - HIPVR[2] | Host Interrupt Prioritized Vector Registers       |
| 0xFFFFE F608 - 0xFFFFE FFFF | -                   | Reserved                                          |

## 6.8 General-Purpose Input/Output (GPIO)

The GPIO peripheral provides general-purpose pins that can be configured as either inputs or outputs. When configured as an output, a write to an internal register can control the state driven on the output pin. When configured as an input, the state of the input is detectable by reading the state of an internal register. In addition, the GPIO peripheral can produce CPU interrupts and EDMA events in different interrupt/event generation modes. The GPIO peripheral provides generic connections to external devices. The GPIO pins are grouped into banks of 16 pins per bank (i.e., bank 0 consists of GPIO [0:15]).

The device GPIO peripheral supports the following:

- External Interrupt and DMA request Capability
  - Every GPIO pin may be configured to generate an interrupt request on detection of rising and/or falling edges on the pin.
  - The interrupt requests within each bank are combined (logical or) to create eight unique bank level interrupt requests.
  - The bank level interrupt service routine may poll the INTSTATx register for its bank to determine which pin(s) have triggered the interrupt.
  - GPIO Banks 0, 1, 2, 3, 4, 5, 6, and 7 Interrupts assigned to ARM INTC Interrupt Requests 42, 43, 44, 45, 46, 47, 48, and 49 respectively
  - Additionally, GPIO Banks 0, 1, 2, 3, 4, and 5 Interrupts assigned to EDMA events 6, 7, 22, 23, 28, and 29 respectively.
- 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 6-8](#).

## 6.8.1 GPIO Register Description(s)

**Table 6-8. GPIO Registers**

| BYTE ADDRESS              | ACRONYM        | REGISTER DESCRIPTION                                     |
|---------------------------|----------------|----------------------------------------------------------|
| 0x01E2 6000               | REV            | Peripheral Revision Register                             |
| 0x01E2 6004               | -              | Reserved                                                 |
| 0x01E2 6008               | BINTEN         | GPIO Interrupt Per-Bank Enable Register                  |
| <b>GPIO BANKS 0 AND 1</b> |                |                                                          |
| 0x01E2 6010               | DIR01          | GPIO Banks 0 and 1 Direction Register                    |
| 0x01E2 6014               | OUT_DATA01     | GPIO Banks 0 and 1 Output Data Register                  |
| 0x01E2 6018               | SET_DATA01     | GPIO Banks 0 and 1 Set Data Register                     |
| 0x01E2 601C               | CLR_DATA01     | GPIO Banks 0 and 1 Clear Data Register                   |
| 0x01E2 6020               | IN_DATA01      | GPIO Banks 0 and 1 Input Data Register                   |
| 0x01E2 6024               | SET_RIS_TRIG01 | GPIO Banks 0 and 1 Set Rising Edge Interrupt Register    |
| 0x01E2 6028               | CLR_RIS_TRIG01 | GPIO Banks 0 and 1 Clear Rising Edge Interrupt Register  |
| 0x01E2 602C               | SET_FAL_TRIG01 | GPIO Banks 0 and 1 Set Falling Edge Interrupt Register   |
| 0x01E2 6030               | CLR_FAL_TRIG01 | GPIO Banks 0 and 1 Clear Falling Edge Interrupt Register |
| 0x01E2 6034               | INTSTAT01      | GPIO Banks 0 and 1 Interrupt Status Register             |
| <b>GPIO BANKS 2 AND 3</b> |                |                                                          |
| 0x01E2 6038               | DIR23          | GPIO Banks 2 and 3 Direction Register                    |
| 0x01E2 603C               | OUT_DATA23     | GPIO Banks 2 and 3 Output Data Register                  |
| 0x01E2 6040               | SET_DATA23     | GPIO Banks 2 and 3 Set Data Register                     |
| 0x01E2 6044               | CLR_DATA23     | GPIO Banks 2 and 3 Clear Data Register                   |
| 0x01E2 6048               | IN_DATA23      | GPIO Banks 2 and 3 Input Data Register                   |
| 0x01E2 604C               | SET_RIS_TRIG23 | GPIO Banks 2 and 3 Set Rising Edge Interrupt Register    |
| 0x01E2 6050               | CLR_RIS_TRIG23 | GPIO Banks 2 and 3 Clear Rising Edge Interrupt Register  |
| 0x01E2 6054               | SET_FAL_TRIG23 | GPIO Banks 2 and 3 Set Falling Edge Interrupt Register   |
| 0x01E2 6058               | CLR_FAL_TRIG23 | GPIO Banks 2 and 3 Clear Falling Edge Interrupt Register |
| 0x01E2 605C               | INTSTAT23      | GPIO Banks 2 and 3 Interrupt Status Register             |
| <b>GPIO BANKS 4 AND 5</b> |                |                                                          |
| 0x01E2 6060               | DIR45          | GPIO Banks 4 and 5 Direction Register                    |
| 0x01E2 6064               | OUT_DATA45     | GPIO Banks 4 and 5 Output Data Register                  |
| 0x01E2 6068               | SET_DATA45     | GPIO Banks 4 and 5 Set Data Register                     |
| 0x01E2 606C               | CLR_DATA45     | GPIO Banks 4 and 5 Clear Data Register                   |
| 0x01E2 6070               | IN_DATA45      | GPIO Banks 4 and 5 Input Data Register                   |
| 0x01E2 6074               | SET_RIS_TRIG45 | GPIO Banks 4 and 5 Set Rising Edge Interrupt Register    |
| 0x01E2 6078               | CLR_RIS_TRIG45 | GPIO Banks 4 and 5 Clear Rising Edge Interrupt Register  |
| 0x01E2 607C               | SET_FAL_TRIG45 | GPIO Banks 4 and 5 Set Falling Edge Interrupt Register   |
| 0x01E2 6080               | CLR_FAL_TRIG45 | GPIO Banks 4 and 5 Clear Falling Edge Interrupt Register |
| 0x01E2 6084               | INTSTAT45      | GPIO Banks 4 and 5 Interrupt Status Register             |
| <b>GPIO BANKS 6 AND 7</b> |                |                                                          |
| 0x01E2 6088               | DIR67          | GPIO Banks 6 and 7 Direction Register                    |
| 0x01E2 608C               | OUT_DATA67     | GPIO Banks 6 and 7 Output Data Register                  |
| 0x01E2 6090               | SET_DATA67     | GPIO Banks 6 and 7 Set Data Register                     |
| 0x01E2 6094               | CLR_DATA67     | GPIO Banks 6 and 7 Clear Data Register                   |
| 0x01E2 6098               | IN_DATA67      | GPIO Banks 6 and 7 Input Data Register                   |
| 0x01E2 609C               | SET_RIS_TRIG67 | GPIO Banks 6 and 7 Set Rising Edge Interrupt Register    |
| 0x01E2 60A0               | CLR_RIS_TRIG67 | GPIO Banks 6 and 7 Clear Rising Edge Interrupt Register  |
| 0x01E2 60A4               | SET_FAL_TRIG67 | GPIO Banks 6 and 7 Set Falling Edge Interrupt Register   |

**Table 6-8. GPIO Registers (continued)**

| BYTE ADDRESS | ACRONYM        | REGISTER DESCRIPTION                                     |
|--------------|----------------|----------------------------------------------------------|
| 0x01E2 60A8  | CLR_FAL_TRIG67 | GPIO Banks 6 and 7 Clear Falling Edge Interrupt Register |
| 0x01E2 60AC  | INTSTAT67      | GPIO Banks 6 and 7 Interrupt Status Register             |

### 6.8.2 GPIO Peripheral Input/Output Electrical Data/Timing

**Table 6-9. Timing Requirements for GPIO Inputs<sup>(1)</sup> (see Figure 6-10)**

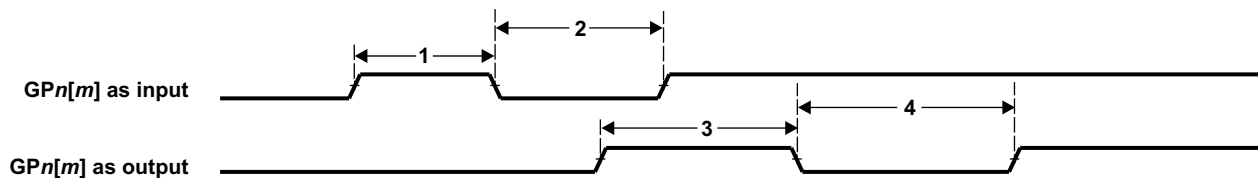
| No. | PARAMETER                                            | MIN           | MAX | UNIT |
|-----|------------------------------------------------------|---------------|-----|------|
| 1   | $t_{w(GPIH)}$ Pulse duration, $GPn[m]$ as input high | $2C^{(1)(2)}$ |     | ns   |
| 2   | $t_{w(GPIL)}$ Pulse duration, $GPn[m]$ as input low  | $2C^{(1)(2)}$ |     | ns   |

- (1) The pulse width given is sufficient to generate a CPU interrupt or an EDMA event. However, if a user wants to have the device recognize the GPIx changes through software polling of the GPIO register, the GPIx duration must be extended to allow the device enough time to access the GPIO register through the internal bus.
- (2)  $C = \text{SYSCLK4 period in ns}$ .

**Table 6-10. Switching Characteristics Over Recommended Operating Conditions for GPIO Outputs (see Figure 6-10)**

| No. | PARAMETER                                             | MIN           | MAX | UNIT |
|-----|-------------------------------------------------------|---------------|-----|------|
| 3   | $t_{w(GPOH)}$ Pulse duration, $GPn[m]$ as output high | $2C^{(1)(2)}$ |     | ns   |
| 4   | $t_{w(GPOL)}$ Pulse duration, $GPn[m]$ as output low  | $2C^{(1)(2)}$ |     | 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)  $C = \text{SYSCLK4 period in ns}$ .

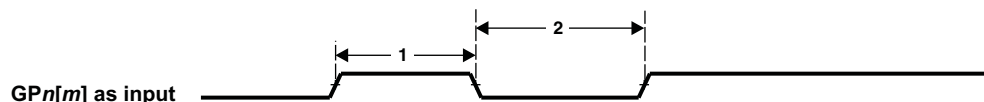
**Figure 6-10. GPIO Port Timing**

### 6.8.3 GPIO Peripheral External Interrupts Electrical Data/Timing

**Table 6-11. Timing Requirements for External Interrupts<sup>(1)</sup> (see Figure 6-11)**

| No. | PARAMETER                                                | MIN           | MAX | UNIT |
|-----|----------------------------------------------------------|---------------|-----|------|
| 1   | $t_{w(LOW)}$ Width of the external interrupt pulse low   | $2C^{(1)(2)}$ |     | ns   |
| 2   | $t_{w(HIGH)}$ Width of the external interrupt pulse high | $2C^{(1)(2)}$ |     | ns   |

- (1) The pulse width given is sufficient to generate an interrupt or an EDMA event. However, if a user wants to have device recognize the GPIO changes through software polling of the GPIO register, the GPIO duration must be extended to allow the device enough time to access the GPIO register through the internal bus.
- (2)  $C = \text{SYSCLK4 period in ns}$ .

**Figure 6-11. GPIO External Interrupt Timing**

## 6.9 EDMA

Table 6-12 is the list of EDMA3 Channel Controller Registers and Table 6-13 is the list of EDMA3 Transfer Controller registers.

**Table 6-12. EDMA3 Channel Controller (EDMA3CC) Registers**

| BYTE ADDRESS                    | ACRONYM    | REGISTER DESCRIPTION                            |
|---------------------------------|------------|-------------------------------------------------|
| 0x01C0 0000                     | PID        | Peripheral Identification Register              |
| 0x01C0 0004                     | CCCFG      | EDMA3CC Configuration Register                  |
| <b>GLOBAL REGISTERS</b>         |            |                                                 |
| 0x01C0 0200                     | QCHMAP0    | QDMA Channel 0 Mapping Register                 |
| 0x01C0 0204                     | QCHMAP1    | QDMA Channel 1 Mapping Register                 |
| 0x01C0 0208                     | QCHMAP2    | QDMA Channel 2 Mapping Register                 |
| 0x01C0 020C                     | QCHMAP3    | QDMA Channel 3 Mapping Register                 |
| 0x01C0 0210                     | QCHMAP4    | QDMA Channel 4 Mapping Register                 |
| 0x01C0 0214                     | QCHMAP5    | QDMA Channel 5 Mapping Register                 |
| 0x01C0 0218                     | QCHMAP6    | QDMA Channel 6 Mapping Register                 |
| 0x01C0 021C                     | QCHMAP7    | QDMA Channel 7 Mapping Register                 |
| 0x01C0 0240                     | DMAQNUM0   | DMA Channel Queue Number Register 0             |
| 0x01C0 0244                     | DMAQNUM1   | DMA Channel Queue Number Register 1             |
| 0x01C0 0248                     | DMAQNUM2   | DMA Channel Queue Number Register 2             |
| 0x01C0 024C                     | DMAQNUM3   | DMA Channel Queue Number Register 3             |
| 0x01C0 0260                     | QDMAQNUM   | QDMA Channel Queue Number Register              |
| 0x01C0 0284                     | QUEPRI     | Queue Priority Register <sup>(1)</sup>          |
| 0x01C0 0300                     | EMR        | Event Missed Register                           |
| 0x01C0 0308                     | EMCR       | Event Missed Clear Register                     |
| 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 0348                     | DRAE1      | DMA Region Access Enable Register for Region 1  |
| 0x01C0 0350                     | DRAE2      | DMA Region Access Enable Register for Region 2  |
| 0x01C0 0358                     | DRAE3      | DMA Region Access Enable Register for Region 3  |
| 0x01C0 0380                     | QRAE0      | QDMA Region Access Enable Register for Region 0 |
| 0x01C0 0384                     | QRAE1      | QDMA Region Access Enable Register for Region 1 |
| 0x01C0 0388                     | QRAE2      | QDMA Region Access Enable Register for Region 2 |
| 0x01C0 038C                     | QRAE3      | QDMA Region Access Enable Register for Region 3 |
| 0x01C0 0400 - 0x01C0 043C       | Q0E0-Q0E15 | Event Queue Entry Registers Q0E0-Q0E15          |
| 0x01C0 0440 - 0x01C0 047C       | Q1E0-Q1E15 | Event Queue Entry Registers Q1E0-Q1E15          |
| 0x01C0 0600                     | QSTAT0     | Queue 0 Status Register                         |
| 0x01C0 0604                     | QSTAT1     | Queue 1 Status Register                         |
| 0x01C0 0620                     | QWMTHRA    | Queue Watermark Threshold A Register            |
| 0x01C0 0640                     | CCSTAT     | EDMA3CC Status Register                         |
| <b>GLOBAL CHANNEL REGISTERS</b> |            |                                                 |
| 0x01C0 1000                     | ER         | Event Register                                  |
| 0x01C0 1008                     | ECR        | Event Clear Register                            |

(1) On previous architectures, the EDMA3TC priority was controlled by the queue priority register (QUEPRI) in the EDMA3CC memory-map. However for this device, the priority control for the transfer controllers is controlled by the chip-level registers in the System Configuration Module. You should use the chip-level registers and not QUEPRI to configure the TC priority.

**Table 6-12. EDMA3 Channel Controller (EDMA3CC) Registers (continued)**

| BYTE ADDRESS                             | ACRONYM | REGISTER DESCRIPTION                |
|------------------------------------------|---------|-------------------------------------|
| 0x01C0 1010                              | ESR     | Event Set Register                  |
| 0x01C0 1018                              | CER     | Chained Event Register              |
| 0x01C0 1020                              | EER     | Event Enable Register               |
| 0x01C0 1028                              | EECR    | Event Enable Clear Register         |
| 0x01C0 1030                              | EESR    | Event Enable Set Register           |
| 0x01C0 1038                              | SER     | Secondary Event Register            |
| 0x01C0 1040                              | SECR    | Secondary Event Clear Register      |
| 0x01C0 1050                              | IER     | Interrupt Enable Register           |
| 0x01C0 1058                              | IECR    | Interrupt Enable Clear Register     |
| 0x01C0 1060                              | IESR    | Interrupt Enable Set Register       |
| 0x01C0 1068                              | IPR     | Interrupt Pending Register          |
| 0x01C0 1070                              | ICR     | Interrupt Clear Register            |
| 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 |
| <b>SHADOW REGION 0 CHANNEL REGISTERS</b> |         |                                     |
| 0x01C0 2000                              | ER      | Event Register                      |
| 0x01C0 2008                              | ECR     | Event Clear Register                |
| 0x01C0 2010                              | ESR     | Event Set Register                  |
| 0x01C0 2018                              | CER     | Chained Event Register              |
| 0x01C0 2020                              | EER     | Event Enable Register               |
| 0x01C0 2028                              | EECR    | Event Enable Clear Register         |
| 0x01C0 2030                              | EESR    | Event Enable Set Register           |
| 0x01C0 2038                              | SER     | Secondary Event Register            |
| 0x01C0 2040                              | SECR    | Secondary Event Clear Register      |
| 0x01C0 2050                              | IER     | Interrupt Enable Register           |
| 0x01C0 2058                              | IECR    | Interrupt Enable Clear Register     |
| 0x01C0 2060                              | IESR    | Interrupt Enable Set Register       |
| 0x01C0 2068                              | IPR     | Interrupt Pending Register          |
| 0x01C0 2070                              | ICR     | Interrupt Clear Register            |
| 0x01C0 2078                              | IEVAL   | Interrupt Evaluate Register         |
| 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 |
| <b>SHADOW REGION 1 CHANNEL REGISTERS</b> |         |                                     |
| 0x01C0 2200                              | ER      | Event Register                      |
| 0x01C0 2208                              | ECR     | Event Clear Register                |
| 0x01C0 2210                              | ESR     | Event Set Register                  |
| 0x01C0 2218                              | CER     | Chained Event Register              |
| 0x01C0 2220                              | EER     | Event Enable Register               |

**Table 6-12. EDMA3 Channel Controller (EDMA3CC) Registers (continued)**

| BYTE ADDRESS              | ACRONYM | REGISTER DESCRIPTION                |
|---------------------------|---------|-------------------------------------|
| 0x01C0 2228               | EECR    | Event Enable Clear Register         |
| 0x01C0 2230               | EESR    | Event Enable Set Register           |
| 0x01C0 2238               | SER     | Secondary Event Register            |
| 0x01C0 2240               | SECR    | Secondary Event Clear Register      |
| 0x01C0 2250               | IER     | Interrupt Enable Register           |
| 0x01C0 2258               | IECR    | Interrupt Enable Clear Register     |
| 0x01C0 2260               | IESR    | Interrupt Enable Set Register       |
| 0x01C0 2268               | IPR     | Interrupt Pending Register          |
| 0x01C0 2270               | ICR     | Interrupt Clear Register            |
| 0x01C0 2278               | IEVAL   | Interrupt Evaluate Register         |
| 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 4000 - 0x01C0 4FFF | —       | Parameter RAM (PaRAM)               |

**Table 6-13. EDMA3 Transfer Controller (EDMA3TC) Registers**

| TRANSFER CONTROLLER 0<br>BYTE ADDRESS | TRANSFER CONTROLLER 1<br>BYTE ADDRESS | ACRONYM   | REGISTER DESCRIPTION                                          |
|---------------------------------------|---------------------------------------|-----------|---------------------------------------------------------------|
| 0x01C0 8000                           | 0x01C0 8400                           | PID       | Peripheral Identification Register                            |
| 0x01C0 8004                           | 0x01C0 8404                           | TCCFG     | EDMA3TC Configuration Register                                |
| 0x01C0 8100                           | 0x01C0 8500                           | TCSTAT    | EDMA3TC Channel Status Register                               |
| 0x01C0 8120                           | 0x01C0 8520                           | ERRSTAT   | Error Status Register                                         |
| 0x01C0 8124                           | 0x01C0 8524                           | ERREN     | Error Enable Register                                         |
| 0x01C0 8128                           | 0x01C0 8528                           | ERRCLR    | Error Clear Register                                          |
| 0x01C0 812C                           | 0x01C0 852C                           | ERRDET    | Error Details Register                                        |
| 0x01C0 8130                           | 0x01C0 8530                           | ERRCMD    | Error Interrupt Command Register                              |
| 0x01C0 8140                           | 0x01C0 8540                           | RDRATE    | Read Command Rate Register                                    |
| 0x01C0 8240                           | 0x01C0 8640                           | SAOPT     | Source Active Options Register                                |
| 0x01C0 8244                           | 0x01C0 8644                           | SASRC     | Source Active Source Address Register                         |
| 0x01C0 8248                           | 0x01C0 8648                           | SACNT     | Source Active Count Register                                  |
| 0x01C0 824C                           | 0x01C0 864C                           | SADST     | Source Active Destination Address Register                    |
| 0x01C0 8250                           | 0x01C0 8650                           | SABIDX    | Source Active B-Index Register                                |
| 0x01C0 8254                           | 0x01C0 8654                           | SAMPPRXY  | Source Active Memory Protection Proxy Register                |
| 0x01C0 8258                           | 0x01C0 8658                           | SACNTRLD  | Source Active Count Reload Register                           |
| 0x01C0 825C                           | 0x01C0 865C                           | SASRCBREF | Source Active Source Address B-Reference Register             |
| 0x01C0 8260                           | 0x01C0 8660                           | SADSTBREF | Source Active Destination Address B-Reference Register        |
| 0x01C0 8280                           | 0x01C0 8680                           | DFCNTRLD  | Destination FIFO Set Count Reload Register                    |
| 0x01C0 8284                           | 0x01C0 8684                           | DFSRCBREF | Destination FIFO Set Source Address B-Reference Register      |
| 0x01C0 8288                           | 0x01C0 8688                           | DFDSTBREF | Destination FIFO Set Destination Address B-Reference Register |
| 0x01C0 8300                           | 0x01C0 8700                           | DFOPT0    | Destination FIFO Options Register 0                           |
| 0x01C0 8304                           | 0x01C0 8704                           | DFSRC0    | Destination FIFO Source Address Register 0                    |
| 0x01C0 8308                           | 0x01C0 8708                           | DFCNT0    | Destination FIFO Count Register 0                             |
| 0x01C0 830C                           | 0x01C0 870C                           | DFDST0    | Destination FIFO Destination Address Register 0               |
| 0x01C0 8310                           | 0x01C0 8710                           | DFBIDX0   | Destination FIFO B-Index Register 0                           |

**Table 6-13. EDMA3 Transfer Controller (EDMA3TC) Registers (continued)**

| TRANSFER CONTROLLER 0<br>BYTE ADDRESS | TRANSFER CONTROLLER 1<br>BYTE ADDRESS | ACRONYM   | REGISTER DESCRIPTION                                |
|---------------------------------------|---------------------------------------|-----------|-----------------------------------------------------|
| 0x01C0 8314                           | 0x01C0 8714                           | DFMPPRXY0 | Destination FIFO Memory Protection Proxy Register 0 |
| 0x01C0 8340                           | 0x01C0 8740                           | DFOPT1    | Destination FIFO Options Register 1                 |
| 0x01C0 8344                           | 0x01C0 8744                           | DFSRC1    | Destination FIFO Source Address Register 1          |
| 0x01C0 8348                           | 0x01C0 8748                           | DFCNT1    | Destination FIFO Count Register 1                   |
| 0x01C0 834C                           | 0x01C0 874C                           | DFDST1    | Destination FIFO Destination Address Register 1     |
| 0x01C0 8350                           | 0x01C0 8750                           | DFBIDX1   | Destination FIFO B-Index Register 1                 |
| 0x01C0 8354                           | 0x01C0 8754                           | DFMPPRXY1 | Destination FIFO Memory Protection Proxy Register 1 |
| 0x01C0 8380                           | 0x01C0 8780                           | DFOPT2    | Destination FIFO Options Register 2                 |
| 0x01C0 8384                           | 0x01C0 8784                           | DFSRC2    | Destination FIFO Source Address Register 2          |
| 0x01C0 8388                           | 0x01C0 8788                           | DFCNT2    | Destination FIFO Count Register 2                   |
| 0x01C0 838C                           | 0x01C0 878C                           | DFDST2    | Destination FIFO Destination Address Register 2     |
| 0x01C0 8390                           | 0x01C0 8790                           | DFBIDX2   | Destination FIFO B-Index Register 2                 |
| 0x01C0 8394                           | 0x01C0 8794                           | DFMPPRXY2 | Destination FIFO Memory Protection Proxy Register 2 |
| 0x01C0 83C0                           | 0x01C0 87C0                           | DFOPT3    | Destination FIFO Options Register 3                 |
| 0x01C0 83C4                           | 0x01C0 87C4                           | DFSRC3    | Destination FIFO Source Address Register 3          |
| 0x01C0 83C8                           | 0x01C0 87C8                           | DFCNT3    | Destination FIFO Count Register 3                   |
| 0x01C0 83CC                           | 0x01C0 87CC                           | DFDST3    | Destination FIFO Destination Address Register 3     |
| 0x01C0 83D0                           | 0x01C0 87D0                           | DFBIDX3   | Destination FIFO B-Index Register 3                 |
| 0x01C0 83D4                           | 0x01C0 87D4                           | DFMPPRXY3 | Destination FIFO Memory Protection Proxy Register 3 |

Table 6-14 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 6-15 shows the parameter set entry registers with relative memory address locations within each of the parameter sets.

**Table 6-14. EDMA Parameter Set RAM**

| BYTE ADDRESS              | 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 6-15. Parameter Set Entries**

| BYTE OFFSET ADDRESS<br>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 |

**Table 6-15. Parameter Set Entries (continued)**

| BYTE OFFSET ADDRESS<br>WITHIN THE PARAMETER SET | ACRONYM | PARAMETER ENTRY |
|-------------------------------------------------|---------|-----------------|
| 0x001C                                          | CCNT    | C Count         |

**Table 6-16. EDMA Events**

| Event | Event Name / Source    | Event | Event Name / Source   |
|-------|------------------------|-------|-----------------------|
| 0     | McASP0 Receive         | 16    | MMCS0 Receive         |
| 1     | McASP0 Transmit        | 17    | MMCS0 Transmit        |
| 2     | McASP1 Receive         | 18    | SPI1 Receive          |
| 3     | McASP1 Transmit        | 19    | SPI1 Transmit         |
| 4     | Reserved               | 20    | PRU_EVTOUT6           |
| 5     | Reserved               | 21    | PRU_EVTOUT7           |
| 6     | GPIO Bank 0 Interrupt  | 22    | GPIO Bank 2 Interrupt |
| 7     | GPIO Bank 1 Interrupt  | 23    | GPIO Bank 3 Interrupt |
| 8     | UART0 Receive          | 24    | I2C0 Receive          |
| 9     | UART0 Transmit         | 25    | I2C0 Transmit         |
| 10    | Timer64P0 Event Out 12 | 26    | I2C1 Receive          |
| 11    | Timer64P0 Event Out 34 | 27    | I2C1 Transmit         |
| 12    | UART1 Receive          | 28    | GPIO Bank 4 Interrupt |
| 13    | UART1 Transmit         | 29    | GPIO Bank 5 Interrupt |
| 14    | SPI0 Receive           | 30    | UART2 Receive         |
| 15    | SPI0 Transmit          | 31    | UART2 Transmit        |

## 6.10 External Memory Interface A (EMIFA)

EMIFA is one of two external memory interfaces supported on the device. It supports asynchronous memory types, such as NAND and NOR flash and Asynchronous SRAM.

### 6.10.1 EMIFA Asynchronous Memory Support

EMIFA supports asynchronous:

- SRAM memories
- NAND Flash memories
- NOR Flash memories

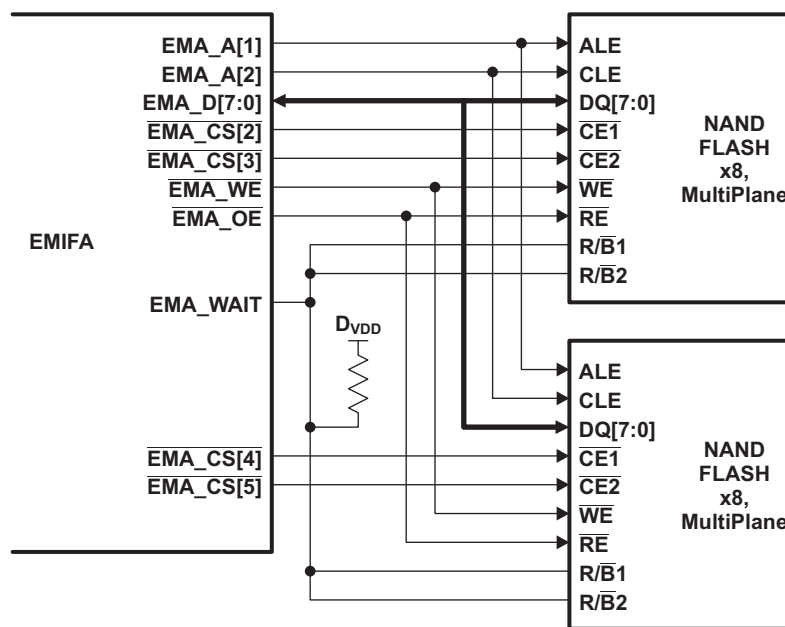
The device supports up to 13 address lines and an external wait/interrupt input. Up to 2 asynchronous chip selects are supported by EMIFA (EMA\_CS[3:2]).

Each chip select has the following individually programmable attributes:

- Data Bus Width
- Read cycle timings: setup, hold, strobe
- Write cycle timings: setup, hold, strobe
- Bus turn around time
- Extended Wait Option With Programmable Timeout
- Select Strobe Option
- NAND flash controller supports 1-bit and 4-bit ECC calculation on blocks of 512 bytes.

### 6.10.2 EMIFA Connection Examples

A likely use case with more than one EMIFA chip select used for NAND flash is illustrated in [Figure 6-12](#). This figure shows how two multiplane NAND flash devices with two chip selects each would connect to the EMIFA. In this case if NAND is the boot memory, then the boot image needs to be stored in the NAND area selected by EMA\_CS[3]. Part of the application image could spill over into the NAND regions selected by other EMIFA chip selects; but would rely on the code stored in the EMA\_CS[3] area to bootload it.



**Figure 6-12. AM1705 EMIFA Connection Diagram: Multiple NAND Flash Planes**

### 6.10.3 External Memory Interface A (EMIFA) Registers

Table 6-17 is a list of the EMIF registers.

**Table 6-17. External Memory Interface (EMIFA) Registers**

| BYTE ADDRESS | ACRONYM         | REGISTER DESCRIPTION                           |
|--------------|-----------------|------------------------------------------------|
| 0x6800 0000  | MIDR            | Module ID Register                             |
| 0x6800 0004  | AWCC            | Asynchronous Wait Cycle Configuration Register |
| 0x6800 0008  | -               | Reserved                                       |
| 0x6800 000C  | -               | Reserved                                       |
| 0x6800 0010  | CE2CFG          | Asynchronous 1 Configuration Register          |
| 0x6800 0014  | CE3CFG          | Asynchronous 2 Configuration Register          |
| 0x6800 0018  | -               | Reserved                                       |
| 0x6800 001C  | -               | Reserved                                       |
| 0x6800 0020  | -               | Reserved                                       |
| 0x6800 003C  | -               | Reserved                                       |
| 0x6800 0040  | INTRAW          | EMIFA Interrupt Raw Register                   |
| 0x6800 0044  | INTMSK          | EMIFA Interrupt Mask Register                  |
| 0x6800 0048  | INTMSKSET       | EMIFA Interrupt Mask Set Register              |
| 0x6800 004C  | INTMSKCLR       | EMIFA Interrupt Mask Clear Register            |
| 0x6800 0060  | NANDFCR         | NAND Flash Control Register                    |
| 0x6800 0064  | NANDFSR         | NAND Flash Status Register                     |
| 0x6800 0070  | NANDF1ECC       | NAND Flash 1 ECC Register (CS2 Space)          |
| 0x6800 0074  | NANDF2ECC       | NAND Flash 2 ECC Register (CS3 Space)          |
| 0x6800 0078  | -               | Reserved                                       |
| 0x6800 007C  | -               | Reserved                                       |
| 0x6800 00BC  | NAND4BITECCLOAD | NAND Flash 4-Bit ECC Load Register             |
| 0x6800 00C0  | NAND4BITECC1    | NAND Flash 4-Bit ECC Register 1                |
| 0x6800 00C4  | NAND4BITECC2    | NAND Flash 4-Bit ECC Register 2                |
| 0x6800 00C8  | NAND4BITECC3    | NAND Flash 4-Bit ECC Register 3                |
| 0x6800 00CC  | NAND4BITECC4    | NAND Flash 4-Bit ECC Register 4                |
| 0x6800 00D0  | NANDERRADD1     | NAND Flash 4-Bit ECC Error Address Register 1  |
| 0x6800 00D4  | NANDERRADD2     | NAND Flash 4-Bit ECC Error Address Register 2  |
| 0x6800 00D8  | NANDERRVAL1     | NAND Flash 4-Bit ECC Error Value Register 1    |
| 0x6800 00DC  | NANDERRVAL2     | NAND Flash 4-Bit ECC Error Value Register 2    |

### 6.10.4 EMIFA Electrical Data/Timing

The following assume testing over recommended operating conditions.

**Table 6-18. EMIFA Asynchronous Memory Timing Requirements<sup>(1)</sup>**

| No.                     | PARAMETER                                                                                     | MIN  | NOM | MAX | UNIT |
|-------------------------|-----------------------------------------------------------------------------------------------|------|-----|-----|------|
| <b>READS and WRITES</b> |                                                                                               |      |     |     |      |
| E                       | $t_{c(CLK)}$ Cycle time, EMIFA module clock                                                   | 10   |     |     | ns   |
| 2                       | $t_{w(EM\_WAIT)}$ Pulse duration, EM_WAIT assertion and deassertion                           | 2E   |     |     | ns   |
| <b>READS</b>            |                                                                                               |      |     |     |      |
| 12                      | $t_{su(EMDV-EMOE H)}$ Setup time, EM_D[15:0] valid before $\overline{EM\_OE}$ high            | 3    |     |     | ns   |
| 13                      | $t_{h(EMOE H-EMDIV)}$ Hold time, EM_D[15:0] valid after $\overline{EM\_OE}$ high              | 0    |     |     | ns   |
| 14                      | $t_{su(EMOEL-EMWAIT)}$ Setup Time, EM_WAIT asserted before end of Strobe Phase <sup>(2)</sup> | 4E+3 |     |     | ns   |
| <b>WRITES</b>           |                                                                                               |      |     |     |      |
| 28                      | $t_{su(EMWEL-EMWAIT)}$ Setup Time, EM_WAIT asserted before end of Strobe Phase <sup>(2)</sup> | 4E+3 |     |     | ns   |

- (1) E = EMA\_CLK period or in ns. EMA\_CLK is selected either as SYSCLK3 or the PLL output clock divided by 4.5. As an example, when SYSCLK3 is selected and set to 100MHz, E=10ns.
- (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 6-15 and Figure 6-16 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 6-19. EMIFA Asynchronous Memory Switching Characteristics<sup>(1) (2) (3)</sup>**

| No.              | PARAMETER                     |                                                                             | MIN                       | NOM                    | MAX                       | UNIT |
|------------------|-------------------------------|-----------------------------------------------------------------------------|---------------------------|------------------------|---------------------------|------|
| READS and WRITES |                               |                                                                             |                           |                        |                           |      |
| 1                | t <sub>d</sub> (TURNAROUND)   | Turn around time                                                            | (TA)*E - 3                | (TA)*E                 | (TA)*E + 3                | ns   |
| READS            |                               |                                                                             |                           |                        |                           |      |
| 3                | t <sub>c</sub> (EMRCYCLE)     | EMIF read cycle time (EW = 0)                                               | (RS+RST+RH)*E<br>- 3      | (RS+RST+RH)*E          | (RS+RST+RH)*E<br>+ 3      | ns   |
|                  |                               | EMIF read cycle time (EW = 1)                                               | (RS+RST+RH+E<br>WC)*E - 3 | (RS+RST+RH+EWC<br>) *E | (RS+RST+RH+E<br>WC)*E + 3 | ns   |
| 4                | t <sub>su</sub> (EMCEL-EMOEL) | Output setup time, $\overline{EMA\_CE}[5:2]$ low to<br>EMA_OE low (SS = 0)  | (RS)*E-3                  | (RS)*E                 | (RS)*E+3                  | ns   |
|                  |                               | Output setup time, $\overline{EMA\_CE}[5:2]$ low to<br>EMA_OE low (SS = 1)  | -3                        | 0                      | +3                        | ns   |
| 5                | t <sub>h</sub> (EMOEH-EMCEH)  | Output hold time, $\overline{EMA\_OE}$ high to<br>EMA_CE[5:2] high (SS = 0) | (RH)*E - 3                | (RH)*E                 | (RH)*E + 3                | ns   |
|                  |                               | Output hold time, $\overline{EMA\_OE}$ high to<br>EMA_CE[5:2] high (SS = 1) | -3                        | 0                      | +3                        | ns   |
| 6                | t <sub>su</sub> (EMBAV-EMOEL) | Output setup time, $\overline{EMA\_BA}[1:0]$ valid to<br>EMA_OE low         | (RS)*E-3                  | (RS)*E                 | (RS)*E+3                  | ns   |
| 7                | t <sub>h</sub> (EMOEH-EMBAIV) | Output hold time, $\overline{EMA\_OE}$ high to<br>EMA_BA[1:0] invalid       | (RH)*E-3                  | (RH)*E                 | (RH)*E+3                  | ns   |
| 8                | t <sub>su</sub> (EMBAV-EMOEL) | Output setup time, $\overline{EMA\_A}[13:0]$ valid to<br>EMA_OE low         | (RS)*E-3                  | (RS)*E                 | (RS)*E+3                  | ns   |
| 9                | t <sub>h</sub> (EMOEH-EMAIV)  | Output hold time, $\overline{EMA\_OE}$ high to<br>EMA_A[13:0] invalid       | (RH)*E-3                  | (RH)*E                 | (RH)*E+3                  | ns   |
| 10               | t <sub>w</sub> (EMOEL)        | EMA_OE active low width (EW = 0)                                            | (RST)*E-3                 | (RST)*E                | (RST)*E+3                 | ns   |
|                  |                               | $\overline{EMA\_OE}$ active low width (EW = 1)                              | (RST+EWC)*E-3             | (RST+EWC)*E            | (RST+EWC)*E+<br>3         | ns   |

- (1) TA = Turn around, RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold, MEWC = Maximum external wait cycles. These parameters are programmed via the Asynchronous Bank and Asynchronous Wait Cycle Configuration Registers. These support the following range of values: TA[4-1], RS[16-1], RST[64-1], RH[8-1], WS[16-1], WST[64-1], WH[8-1], and MEW[1-256].
- (2) E = EMA\_CLK period or in ns. EMA\_CLK is selected either as SYSCLK3 or the PLL output clock divided by 4.5. As an example, when SYSCLK3 is selected and set to 100MHz, E=10ns.
- (3) EWC = external wait cycles determined by EMA\_WAIT input signal. EWC supports the following range of values EWC[256-1]. Note that the maximum wait time before timeout is specified by bit field MEWC in the Asynchronous Wait Cycle Configuration Register.

**Table 6-19. EMIFA Asynchronous Memory Switching Characteristics<sup>(1) (2) (3)</sup> (continued)**

| No.           | PARAMETER              |                                                                                        | MIN                    | NOM                  | MAX                    | UNIT |
|---------------|------------------------|----------------------------------------------------------------------------------------|------------------------|----------------------|------------------------|------|
| 11            | $t_{d(EMWAITH-EMOEH)}$ | Delay time from EMA_WAIT deasserted to EMA_OE high                                     | 3E-3                   | 4E                   | 4E+3                   | ns   |
| <b>WRITES</b> |                        |                                                                                        |                        |                      |                        |      |
| 15            | $t_{c(EMWCYCLE)}$      | EMIF write cycle time (EW = 0)                                                         | $(WS+WST+WH)^*E-3$     | $(WS+WST+WH)^*E$     | $(WS+WST+WH)^*E+3$     | ns   |
|               |                        | EMIF write cycle time (EW = 1)                                                         | $(WS+WST+WH+EWC)^*E-3$ | $(WS+WST+WH+EWC)^*E$ | $(WS+WST+WH+EWC)^*E+3$ | ns   |
| 16            | $t_{su(EMCEL-EMWEL)}$  | Output setup time, $\overline{EMA\_CE[5:2]}$ low to $\overline{EMA\_WE}$ low (SS = 0)  | $(WS)^*E-3$            | $(WS)^*E$            | $(WS)^*E+3$            | ns   |
|               |                        | Output setup time, $\overline{EMA\_CE[5:2]}$ low to $\overline{EMA\_WE}$ low (SS = 1)  | -3                     | 0                    | +3                     | ns   |
| 17            | $t_{h(EMWEH-EMCEH)}$   | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_CE[5:2]}$ high (SS = 0) | $(WH)^*E-3$            | $(WH)^*E$            | $(WH)^*E+3$            | ns   |
|               |                        | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_CE[5:2]}$ high (SS = 1) | -3                     | 0                    | +3                     | ns   |
| 18            | $t_{su(EMDQMV-EMWEL)}$ | Output setup time, $\overline{EMA\_BA[1:0]}$ valid to $\overline{EMA\_WE}$ low         | $(WS)^*E-3$            | $(WS)^*E$            | $(WS)^*E+3$            | ns   |
| 19            | $t_{h(EMWEH-EMDQMV)}$  | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_BA[1:0]}$ invalid       | $(WH)^*E-3$            | $(WH)^*E$            | $(WH)^*E+3$            | ns   |
| 20            | $t_{su(EMBAV-EMWEL)}$  | Output setup time, $\overline{EMA\_BA[1:0]}$ valid to $\overline{EMA\_WE}$ low         | $(WS)^*E-3$            | $(WS)^*E$            | $(WS)^*E+3$            | ns   |
| 21            | $t_{h(EMWEH-EMBAV)}$   | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_BA[1:0]}$ invalid       | $(WH)^*E-3$            | $(WH)^*E$            | $(WH)^*E+3$            | ns   |
| 22            | $t_{su(EMAV-EMWEL)}$   | Output setup time, $\overline{EMA\_A[13:0]}$ valid to $\overline{EMA\_WE}$ low         | $(WS)^*E-3$            | $(WS)^*E$            | $(WS)^*E+3$            | ns   |
| 23            | $t_{h(EMWEH-EMAV)}$    | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_A[13:0]}$ invalid       | $(WH)^*E-3$            | $(WH)^*E$            | $(WH)^*E+3$            | ns   |
| 24            | $t_{w(EMWEL)}$         | $\overline{EMA\_WE}$ active low width (EW = 0)                                         | $(WST)^*E-3$           | $(WST)^*E$           | $(WST)^*E+3$           | ns   |
|               |                        | $\overline{EMA\_WE}$ active low width (EW = 1)                                         | $(WST+EWC)^*E-3$       | $(WST+EWC)^*E$       | $(WST+EWC)^*E+3$       | ns   |
| 25            | $t_{d(EMWAITH-EMWEH)}$ | Delay time from EMA_WAIT deasserted to $\overline{EMA\_WE}$ high                       | 3E-3                   | 4E                   | 4E+3                   | ns   |
| 26            | $t_{su(EMDV-EMWEL)}$   | Output setup time, $\overline{EMA\_D[15:0]}$ valid to $\overline{EMA\_WE}$ low         | $(WS)^*E-3$            | $(WS)^*E$            | $(WS)^*E+3$            | ns   |
| 27            | $t_{h(EMWEH-EMDIV)}$   | Output hold time, $\overline{EMA\_WE}$ high to $\overline{EMA\_D[15:0]}$ invalid       | $(WH)^*E-3$            | $(WH)^*E$            | $(WH)^*E+3$            | ns   |

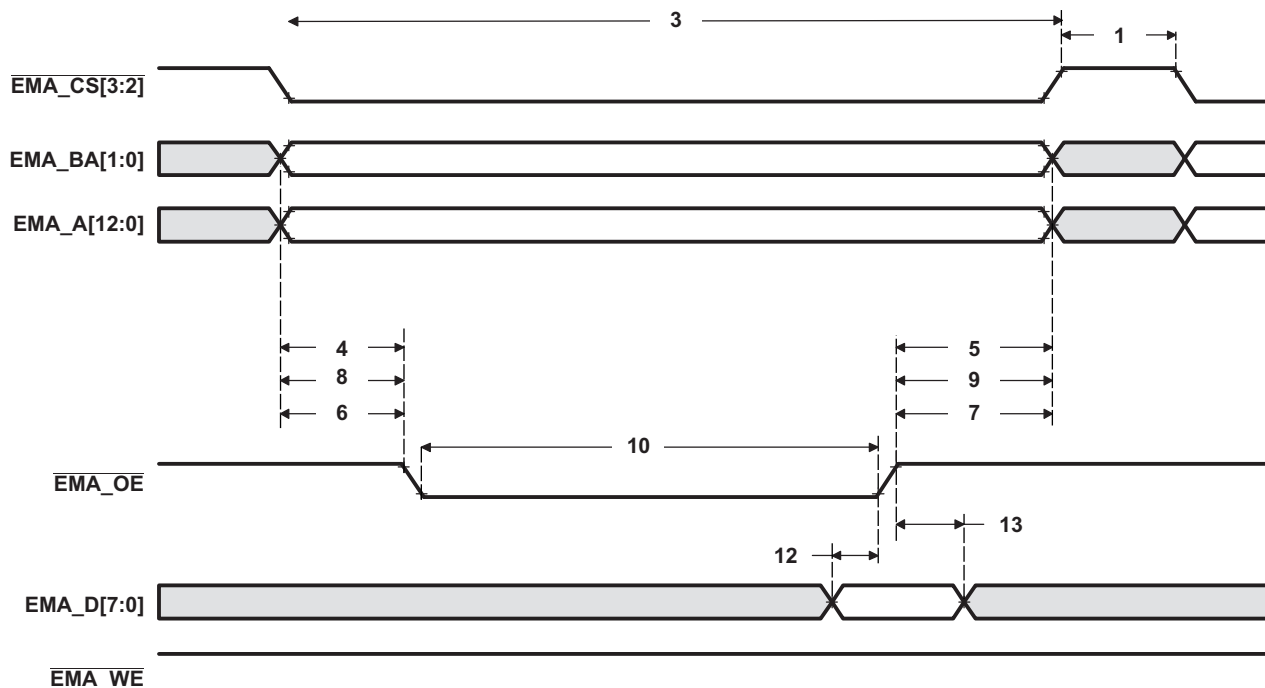


Figure 6-13. Asynchronous Memory Read Timing for EMIFA

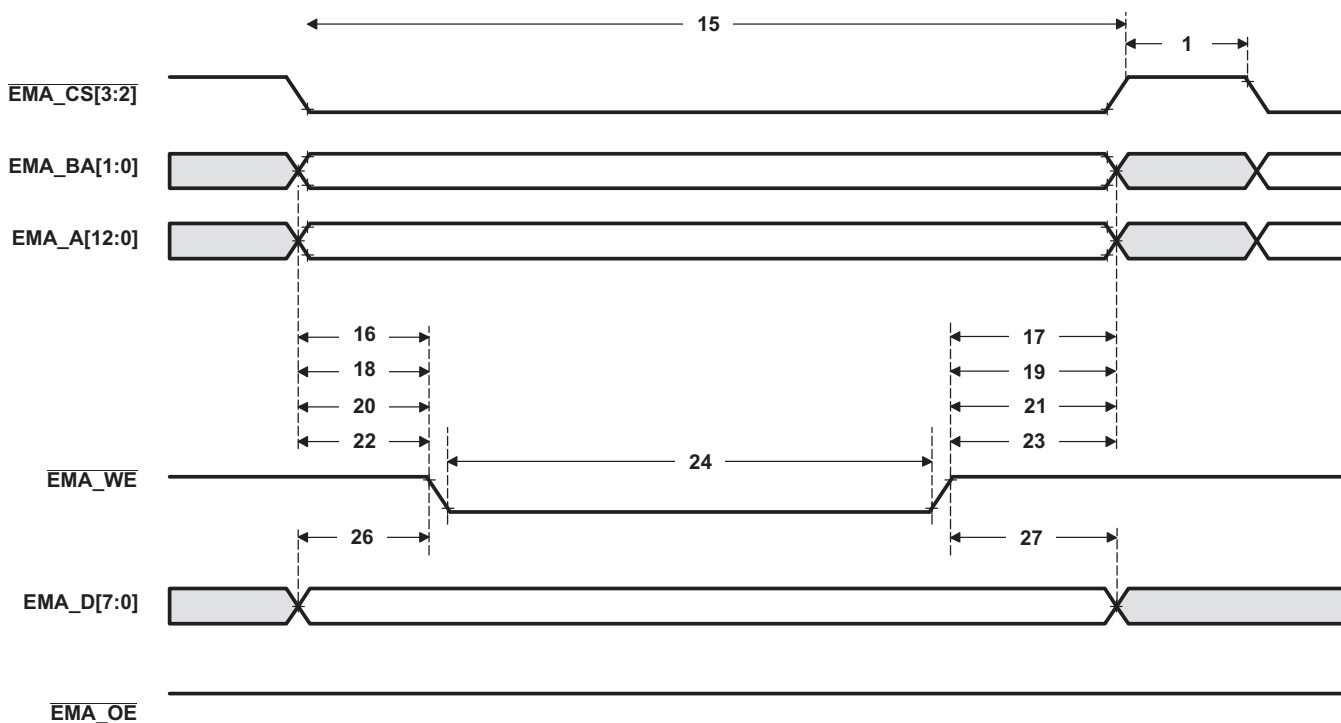


Figure 6-14. Asynchronous Memory Write Timing for EMIFA

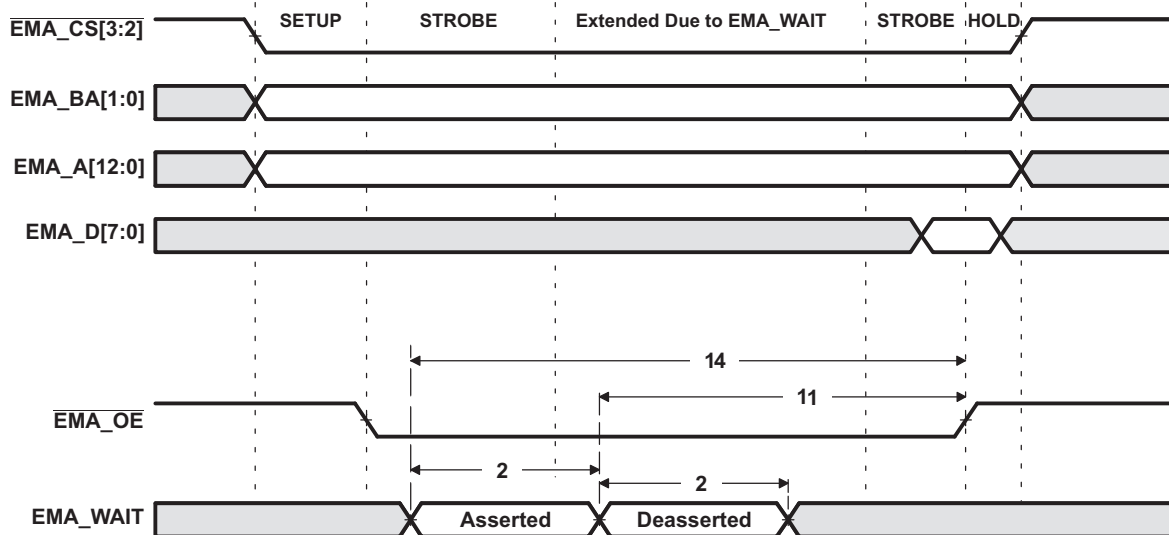


Figure 6-15. EMA\_WAIT Read Timing Requirements

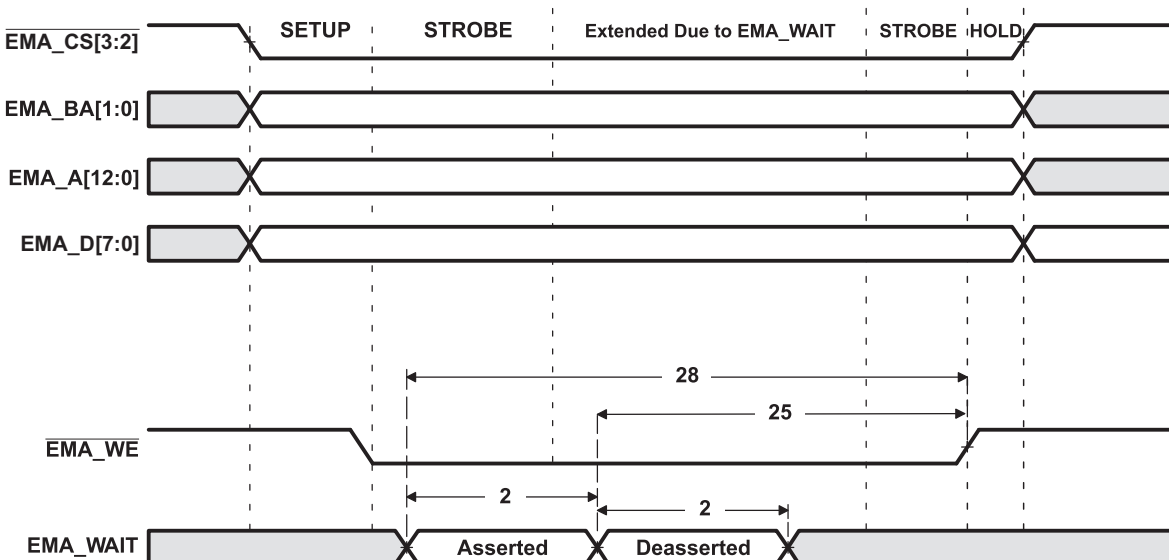
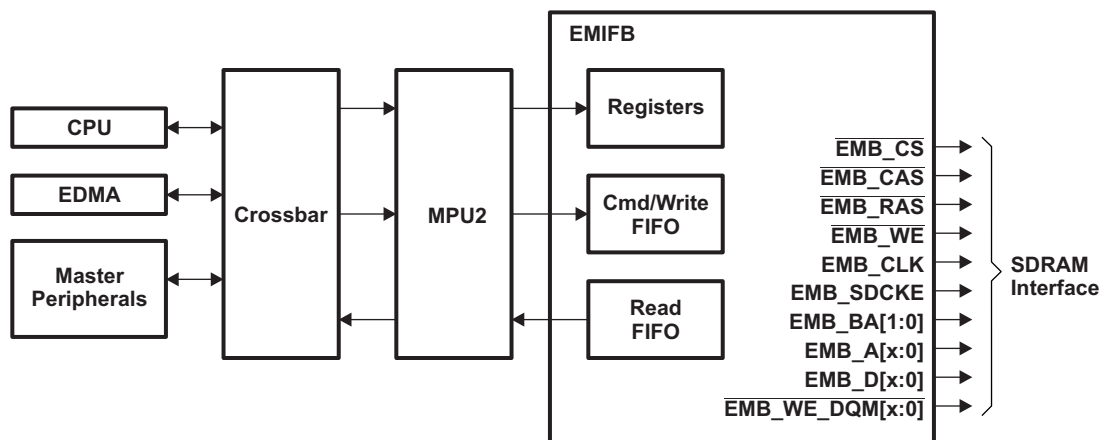


Figure 6-16. EMA\_WAIT Write Timing Requirements

## 6.11 External Memory Interface B (EMIFB)

The following EMIFB Functional Block Diagram illustrates a high-level view of the EMIFB and its connections within the device. Multiple requesters have access to EMIFB through a switched central resource (indicated as an overbar in the figure). The EMIFB implements a split transaction internal bus, allowing concurrence between reads and writes from the various requesters.



**Figure 6-17. EMIFB Functional Block Diagram**

EMIFB supports a 3.3V LVCMOS Interface.

### 6.11.1 EMIFB SDRAM Loading Limitations

EMIFB supports SDRAM up to 152 MHz with up to two SDRAM or asynchronous memory loads. Additional loads will limit the SDRAM operation to lower speeds and the maximum speed should be confirmed by board simulation using IBIS models.

### 6.11.2 Interfacing to SDRAM

The EMIFB supports a glueless interface to SDRAM devices with the following characteristics:

- Pre-charge bit is A[10]
- Supports 8, 9, 10 or 11 column address bits
- Supports up to 13 row address bits
- Supports 1, 2 or 4 internal banks

Table 6-20 shows the supported SDRAM configurations for EMIFB.

**Table 6-20. EMIFB Supported SDRAM Configurations<sup>(1)</sup>**

| SDRAM Memory Data Bus Width (bits) | Number of Memories | EMIFB Data Bus Size | Rows | Columns | Banks | Total Memory (Mbits) | Total Memory (Mbytes) | Memory Density (Mbits) |
|------------------------------------|--------------------|---------------------|------|---------|-------|----------------------|-----------------------|------------------------|
| 16                                 | 1                  | 16                  | 13   | 8       | 1     | 32                   | 4                     | 32                     |
|                                    | 1                  | 16                  | 13   | 8       | 2     | 64                   | 8                     | 64                     |
|                                    | 1                  | 16                  | 13   | 8       | 4     | 128                  | 16                    | 128                    |
|                                    | 1                  | 16                  | 13   | 9       | 1     | 64                   | 8                     | 64                     |
|                                    | 1                  | 16                  | 13   | 9       | 2     | 128                  | 16                    | 128                    |
|                                    | 1                  | 16                  | 13   | 9       | 4     | 256                  | 32                    | 256                    |
|                                    | 1                  | 16                  | 13   | 10      | 1     | 128                  | 16                    | 128                    |
|                                    | 1                  | 16                  | 13   | 10      | 2     | 256                  | 32                    | 256                    |
|                                    | 1                  | 16                  | 13   | 10      | 4     | 512                  | 64                    | 512                    |
|                                    | 1                  | 16                  | 13   | 11      | 1     | 256                  | 32                    | 256                    |
|                                    | 1                  | 16                  | 13   | 11      | 2     | 512                  | 64                    | 512                    |
|                                    | 1                  | 16                  | 13   | 11      | 4     | 1024                 | 128                   | 1024                   |
| 8                                  | 2                  | 16                  | 13   | 8       | 1     | 32                   | 4                     | 16                     |
|                                    | 2                  | 16                  | 13   | 8       | 2     | 64                   | 8                     | 32                     |
|                                    | 2                  | 16                  | 13   | 8       | 4     | 128                  | 16                    | 64                     |
|                                    | 2                  | 16                  | 13   | 9       | 1     | 64                   | 8                     | 32                     |
|                                    | 2                  | 16                  | 13   | 9       | 2     | 128                  | 16                    | 64                     |
|                                    | 2                  | 16                  | 13   | 9       | 4     | 256                  | 32                    | 128                    |
|                                    | 2                  | 16                  | 13   | 10      | 1     | 128                  | 16                    | 64                     |
|                                    | 2                  | 16                  | 13   | 10      | 2     | 256                  | 32                    | 128                    |
|                                    | 2                  | 16                  | 13   | 10      | 4     | 512                  | 64                    | 256                    |
|                                    | 2                  | 16                  | 13   | 11      | 1     | 256                  | 32                    | 128                    |
|                                    | 2                  | 16                  | 13   | 11      | 2     | 512                  | 64                    | 256                    |
|                                    | 2                  | 16                  | 13   | 11      | 4     | 1024                 | 128                   | 512                    |

(1) The shaded cells indicate configurations that are possible on the EMIFA interface but as of this writing SDRAM memories capable of supporting these densities are not available in the market.

Figure 6-18 shows an interface between the EMIFB and a  $2\text{M} \times 16 \times 4$  bank SDRAM device. Refer to Table 6-21, as an example that shows additional list of commonly-supported SDRAM devices and the required connections for the address pins. Note that in Table 6-21, page size/column size (not indicated in the table) is varied to get the required addressability range.

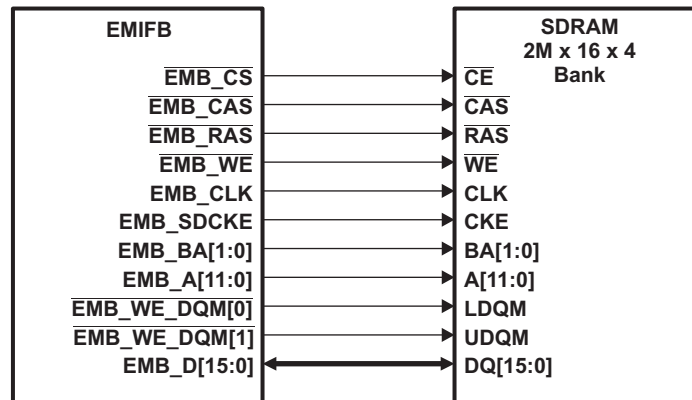


Figure 6-18. EMIFB to  $2\text{M} \times 16 \times 4$  bank SDRAM Interface

Table 6-21. Example of 16-bit EMIFB Address Pin Connections

| SDRAM Size | Width       | Banks |       | Address Pins |
|------------|-------------|-------|-------|--------------|
| 64M bits   | $\times 16$ | 4     | SDRAM | A[11:0]      |
|            |             |       | EMIFB | EMB_A[11:0]  |
| 128M bits  | $\times 16$ | 4     | SDRAM | A[11:0]      |
|            |             |       | EMIFB | EMB_A[11:0]  |
| 256M bits  | $\times 16$ | 4     | SDRAM | A[12:0]      |
|            |             |       | EMIFB | EMB_A[12:0]  |
| 512M bits  | $\times 16$ | 4     | SDRAM | A[12:0]      |
|            |             |       | EMIFB | EMB_A[12:0]  |

### 6.11.3 EMIFB Registers

Table 6-22 is a list of the EMIFB registers.

Table 6-22. EMIFB Controller Registers

| BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION                              |
|--------------|---------|---------------------------------------------------|
| 0xB000 0000  | MIDR    | Module ID Register                                |
| 0xB000 0008  | SDCFG   | SDRAM Configuration Register                      |
| 0xB000 000C  | SDRFC   | SDRAM Refresh Control Register                    |
| 0xB000 0010  | SDTIM1  | SDRAM Timing Register 1                           |
| 0xB000 0014  | SDTIM2  | SDRAM Timing Register 2                           |
| 0xB000 001C  | SDCFG2  | SDRAM Configuration 2 Register                    |
| 0xB000 0020  | BPRIO   | Peripheral Bus Burst Priority Register            |
| 0xB000 0040  | PC1     | Performance Counter 1 Register                    |
| 0xB000 0044  | PC2     | Performance Counter 2 Register                    |
| 0xB000 0048  | PCC     | Performance Counter Configuration Register        |
| 0xB000 004C  | PCMRS   | Performance Counter Master Region Select Register |
| 0xB000 0050  | PCT     | Performance Counter Time Register                 |
| 0xB000 00C0  | IRR     | Interrupt Raw Register                            |

**Table 6-22. EMIFB Controller Registers (continued)**

| BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION          |
|--------------|---------|-------------------------------|
| 0xB000 00C4  | IMR     | Interrupt Mask Register       |
| 0xB000 00C8  | IMSR    | Interrupt Mask Set Register   |
| 0xB000 00CC  | IMCR    | Interrupt Mask Clear Register |

### 6.11.4 EMIFB Electrical Data/Timing

**Table 6-23. EMIFB SDRAM Interface Timing Requirements**

| NO. |                   |                                                                        | CVDD = 1.3 V <sup>(1)</sup> |     | CVDD = 1.2V <sup>(2)</sup> |     | UNIT |
|-----|-------------------|------------------------------------------------------------------------|-----------------------------|-----|----------------------------|-----|------|
|     |                   |                                                                        | MIN                         | MAX | MIN                        | MAX |      |
| 19  | $t_{DV-CLKH}$     | Input setup time, read data valid on EMB_D[31:0] before EMB_CLK rising | 0.59                        |     | 0.8                        |     | ns   |
| 20  | $t_{h(CLKH-DIV)}$ | Input hold time, read data valid on EMB_D[31:0] after EMB_CLK rising   | 1.25                        |     | 1.5                        |     | ns   |

(1) Commercial (default), Industrial and Extended temperature range rated devices for 456 MHz max CPU operating frequency as applicable to the device

(2) Commercial (default), Industrial, Extended and Automotive temperature range rated devices for 400/375/300/266/200 MHz max CPU operating frequencies as applicable to the device

**Table 6-24. EMIFB SDRAM Interface Switching Characteristics for Commercial (Default) Temperature Range**

| NO. | PARAMETER            |                                                                         | CVDD = 1.3 V <sup>(1)</sup> |      | CVDD = 1.2V <sup>(2)</sup> |     | UNIT |
|-----|----------------------|-------------------------------------------------------------------------|-----------------------------|------|----------------------------|-----|------|
|     |                      |                                                                         | MIN                         | MAX  | MIN                        | MAX |      |
| 1   | $t_c(CLK)$           | Cycle time, EMIF clock EMB_CLK                                          | 6.579                       |      | 7.5                        |     | ns   |
| 2   | $t_w(CLK)$           | Pulse width, EMIF clock EMB_CLK high or low                             | 2.63                        |      | 3                          |     | ns   |
| 3   | $t_d(CLKH-CSV)$      | Delay time, EMB_CLK rising to $\overline{EMB\_CS}[0]$ valid             |                             | 4.25 |                            | 5.1 | ns   |
| 4   | $t_{oh}(CLKH-CSIV)$  | Output hold time, EMB_CLK rising to $\overline{EMB\_CS}[0]$ invalid     | 1.1                         |      | 1.1                        |     | ns   |
| 5   | $t_d(CLKH-DQMV)$     | Delay time, EMB_CLK rising to EMB_WE_DQM[3:0] valid                     |                             | 4.25 |                            | 5.1 | ns   |
| 6   | $t_{oh}(CLKH-DQMIV)$ | Output hold time, EMB_CLK rising to EMB_WE_DQM[3:0] invalid             | 1.1                         |      | 1.1                        |     | ns   |
| 7   | $t_d(CLKH-AV)$       | Delay time, EMB_CLK rising to EMB_A[12:0] and EMB_BA[1:0] valid         |                             | 4.25 |                            | 5.1 | ns   |
| 8   | $t_{oh}(CLKH-AIV)$   | Output hold time, EMB_CLK rising to EMB_A[12:0] and EMB_BA[1:0] invalid | 1.1                         |      | 1.1                        |     | ns   |
| 9   | $t_d(CLKH-DV)$       | Delay time, EMB_CLK rising to EMB_D[31:0] valid                         |                             | 4.25 |                            | 5.1 | ns   |
| 10  | $t_{oh}(CLKH-DIV)$   | Output hold time, EMB_CLK rising to EMB_D[31:0] invalid                 | 1.1                         |      | 1.1                        |     | ns   |
| 11  | $t_d(CLKH-RASV)$     | Delay time, EMB_CLK rising to $\overline{EMB\_RAS}$ valid               |                             | 4.25 |                            | 5.1 | ns   |
| 12  | $t_{oh}(CLKH-RASIV)$ | Output hold time, EMB_CLK rising to $\overline{EMB\_RAS}$ invalid       | 1.1                         |      | 1.1                        |     | ns   |
| 13  | $t_d(CLKH-CASV)$     | Delay time, EMB_CLK rising to $\overline{EMB\_CAS}$ valid               |                             | 4.25 |                            | 5.1 | ns   |
| 14  | $t_{oh}(CLKH-CASIV)$ | Output hold time, EMB_CLK rising to $\overline{EMB\_CAS}$ invalid       | 1.1                         |      | 1.1                        |     | ns   |
| 15  | $t_d(CLKH-WEV)$      | Delay time, EMB_CLK rising to $\overline{EMB\_WE}$ valid                |                             | 4.25 |                            | 5.1 | ns   |
| 16  | $t_{oh}(CLKH-WEIV)$  | Output hold time, EMB_CLK rising to $\overline{EMB\_WE}$ invalid        | 1.1                         |      | 1.1                        |     | ns   |
| 17  | $t_{dis}(CLKH-DHZ)$  | Delay time, EMB_CLK rising to EMB_D[31:0] tri-stated                    |                             | 4.25 |                            | 5.1 | ns   |
| 18  | $t_{CLKH-DLZ}$       | Output hold time, EMB_CLK rising to EMB_D[31:0] driving                 | 1.1                         |      | 1.1                        |     | ns   |

(1) Commercial (default) temperature range rated devices for 456 MHz max CPU operating frequency as applicable to the device

(2) Commercial (default) temperature range rated devices for 400/375/300/266/200 MHz max CPU operating frequencies as applicable to the device

**Table 6-25. EMIFB SDRAM Interface Switching Characteristics for Industrial, Extended, and Automotive Temperature Ranges**

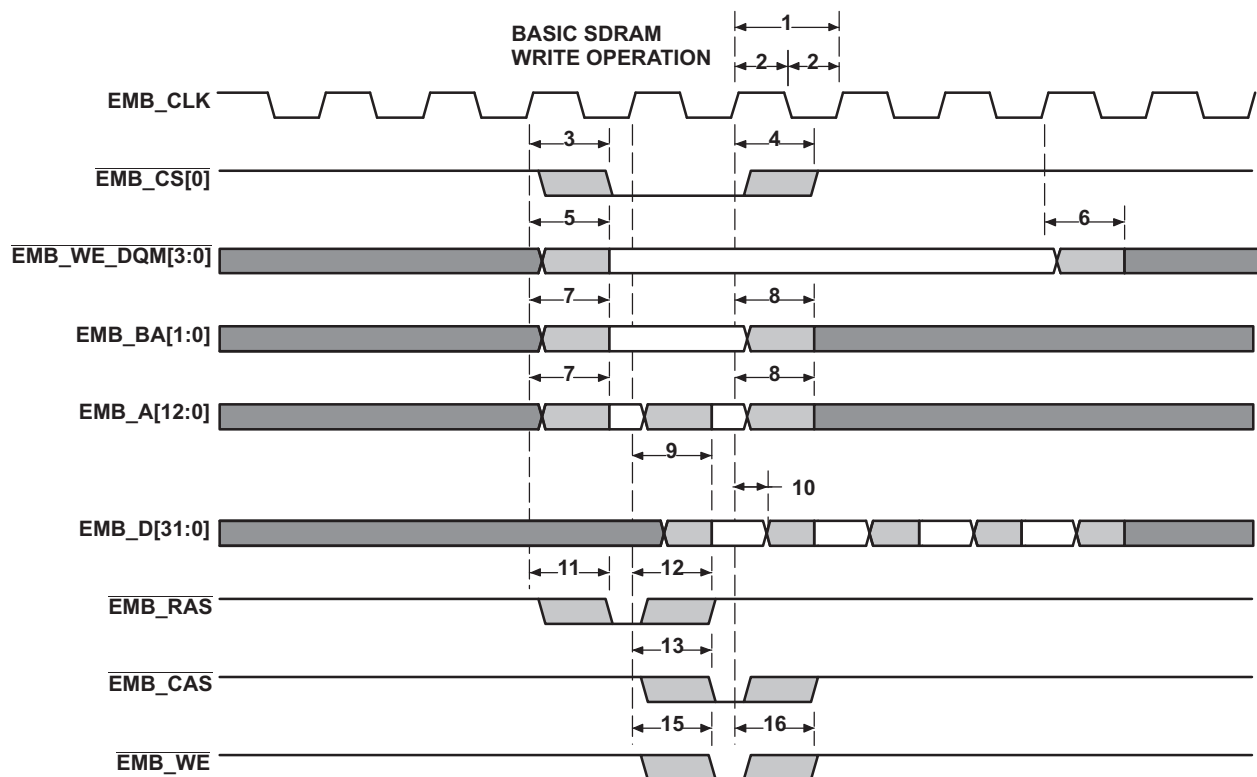
| NO. | PARAMETER           |                                                                     | CVDD = 1.3 V <sup>(1)</sup> |      | CVDD = 1.2V <sup>(2)</sup> |     | UNIT |
|-----|---------------------|---------------------------------------------------------------------|-----------------------------|------|----------------------------|-----|------|
|     |                     |                                                                     | MIN                         | MAX  | MIN                        | MAX |      |
| 1   | $t_c(CLK)$          | Cycle time, EMIF clock EMB_CLK                                      | 6.579                       |      | 7.5                        |     | ns   |
| 2   | $t_w(CLK)$          | Pulse width, EMIF clock EMB_CLK high or low                         | 2.63                        |      | 3                          |     | ns   |
| 3   | $t_d(CLKH-CSV)$     | Delay time, EMB_CLK rising to $\overline{EMB\_CS}[0]$ valid         |                             | 4.25 |                            | 5.1 | ns   |
| 4   | $t_{oh}(CLKH-CSIV)$ | Output hold time, EMB_CLK rising to $\overline{EMB\_CS}[0]$ invalid | 1.1                         |      | 0.9                        |     | ns   |

(1) Industrial temperature range rated devices for 456 MHz max CPU operating frequency as applicable to the device

(2) Industrial, Extended and Automotive temperature range rated devices for 400/375/300/266/200 MHz max CPU operating frequencies as applicable to the device

**Table 6-25. EMIFB SDRAM Interface Switching Characteristics for Industrial, Extended, and Automotive Temperature Ranges (continued)**

| NO. | PARAMETER                                                                                 | CVDD = 1.3 V <sup>(1)</sup> |      | CVDD = 1.2V <sup>(2)</sup> |     | UNIT |
|-----|-------------------------------------------------------------------------------------------|-----------------------------|------|----------------------------|-----|------|
|     |                                                                                           | MIN                         | MAX  | MIN                        | MAX |      |
| 5   | $t_{d(CLKH-DQM)}$ Delay time, EMB_CLK rising to EMB_WE_DQM[3:0] valid                     |                             | 4.25 |                            | 5.1 | ns   |
| 6   | $t_{oh(CLKH-DQM)}$ Output hold time, EMB_CLK rising to EMB_WE_DQM[3:0] invalid            | 1.1                         |      | 0.9                        |     | ns   |
| 7   | $t_{d(CLKH-AV)}$ Delay time, EMB_CLK rising to EMB_A[12:0] and EMB_BA[1:0] valid          |                             | 4.25 |                            | 5.1 | ns   |
| 8   | $t_{oh(CLKH-AV)}$ Output hold time, EMB_CLK rising to EMB_A[12:0] and EMB_BA[1:0] invalid | 1.1                         |      | 0.9                        |     | ns   |
| 9   | $t_{d(CLKH-DV)}$ Delay time, EMB_CLK rising to EMB_D[31:0] valid                          |                             | 4.25 |                            | 5.1 | ns   |
| 10  | $t_{oh(CLKH-DV)}$ Output hold time, EMB_CLK rising to EMB_D[31:0] invalid                 | 1.1                         |      | 0.9                        |     | ns   |
| 11  | $t_{d(CLKH-RAS)}$ Delay time, EMB_CLK rising to EMB_RAS valid                             |                             | 4.25 |                            | 5.1 | ns   |
| 12  | $t_{oh(CLKH-RAS)}$ Output hold time, EMB_CLK rising to EMB_RAS invalid                    | 1.1                         |      | 0.9                        |     | ns   |
| 13  | $t_{d(CLKH-CAS)}$ Delay time, EMB_CLK rising to EMB_CAS valid                             |                             | 4.25 |                            | 5.1 | ns   |
| 14  | $t_{oh(CLKH-CAS)}$ Output hold time, EMB_CLK rising to EMB_CAS invalid                    | 1.1                         |      | 0.9                        |     | ns   |
| 15  | $t_{d(CLKH-WE)}$ Delay time, EMB_CLK rising to EMB_WE valid                               |                             | 4.25 |                            | 5.1 | ns   |
| 16  | $t_{oh(CLKH-WE)}$ Output hold time, EMB_CLK rising to EMB_WE invalid                      | 1.1                         |      | 0.9                        |     | ns   |
| 17  | $t_{dis(CLKH-DHZ)}$ Delay time, EMB_CLK rising to EMB_D[31:0] tri-stated                  |                             | 4.25 |                            | 5.1 | ns   |
| 18  | $t_{h(CLKH-DLZ)}$ Output hold time, EMB_CLK rising to EMB_D[31:0] driving                 | 1.1                         |      | 0.9                        |     | ns   |



**Figure 6-19. EMIFB Basic SDRAM Write Operation**

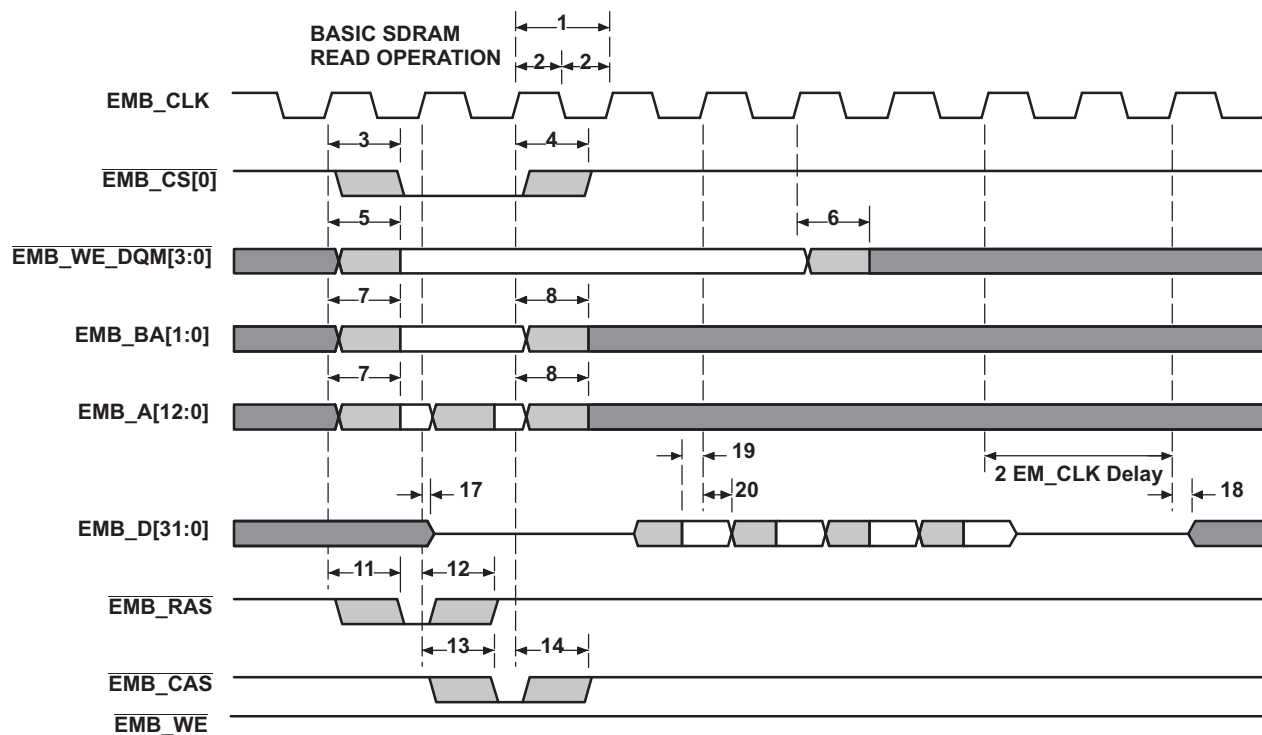


Figure 6-20. EMIFB Basic SDRAM Read Operation

## 6.12 Memory Protection Units

The MPU performs memory protection checking. It receives requests from a bus master in the system and checks the address against the fixed and programmable regions to see if the access is allowed. If allowed, the transfer is passed unmodified to its output bus (to the targeted address). If the transfer is illegal (fails the protection check) then the MPU does not pass the transfer to the output bus but rather services the transfer internally back to the input bus (to prevent a hang) returning the fault status to the requestor as well as generating an interrupt about the fault. The following features are supported by the MPU:

- Provides memory protection for fixed and programmable address ranges
- Supports multiple programmable address region
- Supports secure and debug access privileges
- Supports read, write, and execute access privileges
- Supports privid(8) associations with ranges
- Generates an interrupt when there is a protection violation, and saves violating transfer parameters
- MMR access is also protected

**Table 6-26. MPU1 Configuration Registers**

| MPU1<br>BYTE ADDRESS      | ACRONYM     | REGISTER DESCRIPTION                                    |
|---------------------------|-------------|---------------------------------------------------------|
| 0x01E1 4000               | REVID       | Revision ID                                             |
| 0x01E1 4004               | CONFIG      | Configuration                                           |
| 0x01E1 4010               | IRAWSTAT    | Interrupt raw status/set                                |
| 0x01E1 4014               | IENSTAT     | Interrupt enable status/clear                           |
| 0x01E1 4018               | IENSET      | Interrupt enable                                        |
| 0x01E1 401C               | IENCLR      | Interrupt enable clear                                  |
| 0x01E1 4020 - 0x01E1 41FF | -           | Reserved                                                |
| 0x01E1 4200               | PROG1_MPSAR | Programmable range 1, start address                     |
| 0x01E1 4204               | PROG1_MPEAR | Programmable range 1, end address                       |
| 0x01E1 4208               | PROG1_MPPA  | Programmable range 1, memory page protection attributes |
| 0x01E1 420C - 0x01E1 420F | -           | Reserved                                                |
| 0x01E1 4210               | PROG2_MPSAR | Programmable range 2, start address                     |
| 0x01E1 4214               | PROG2_MPEAR | Programmable range 2, end address                       |
| 0x01E1 4218               | PROG2_MPPA  | Programmable range 2, memory page protection attributes |
| 0x01E1 421C - 0x01E1 421F | -           | Reserved                                                |
| 0x01E1 4220               | PROG3_MPSAR | Programmable range 3, start address                     |
| 0x01E1 4224               | PROG3_MPEAR | Programmable range 3, end address                       |
| 0x01E1 4228               | PROG3_MPPA  | Programmable range 3, memory page protection attributes |
| 0x01E1 422C - 0x01E1 422F | -           | Reserved                                                |
| 0x01E1 4230               | PROG4_MPSAR | Programmable range 4, start address                     |
| 0x01E1 4234               | PROG4_MPEAR | Programmable range 4, end address                       |
| 0x01E1 4238               | PROG4_MPPA  | Programmable range 4, memory page protection attributes |
| 0x01E1 423C - 0x01E1 423F | -           | Reserved                                                |
| 0x01E1 4240               | PROG5_MPSAR | Programmable range 5, start address                     |
| 0x01E1 4244               | PROG5_MPEAR | Programmable range 5, end address                       |
| 0x01E1 4248               | PROG5_MPPA  | Programmable range 5, memory page protection attributes |
| 0x01E1 424C - 0x01E1 424F | -           | Reserved                                                |
| 0x01E1 4250               | PROG6_MPSAR | Programmable range 6, start address                     |
| 0x01E1 4254               | PROG6_MPEAR | Programmable range 6, end address                       |
| 0x01E1 4258               | PROG6_MPPA  | Programmable range 6, memory page protection attributes |
| 0x01E1 425C - 0x01E1 42FF | -           | Reserved                                                |

**Table 6-26. MPU1 Configuration Registers (continued)**

| MPU1<br>BYTE ADDRESS      | ACRONYM  | REGISTER DESCRIPTION |
|---------------------------|----------|----------------------|
| 0x01E14300                | FLTADDRR | Fault address        |
| 0x01E1 4304               | FLTSTAT  | Fault status         |
| 0x01E1 4308               | FLTCLR   | Fault clear          |
| 0x01E1 430C - 0x01E1 4FFF | -        | Reserved             |

**Table 6-27. MPU2 Configuration Registers**

| MPU2<br>BYTE ADDRESS      | ACRONYM     | REGISTER DESCRIPTION                                    |
|---------------------------|-------------|---------------------------------------------------------|
| 0x01E1 5000               | REVID       | Revision ID                                             |
| 0x01E1 5004               | CONFIG      | Configuration                                           |
| 0x01E1 5010               | IRAWSTAT    | Interrupt raw status/set                                |
| 0x01E1 5014               | IENSTAT     | Interrupt enable status/clear                           |
| 0x01E1 5018               | IENSET      | Interrupt enable                                        |
| 0x01E1 501C               | IENCLR      | Interrupt enable clear                                  |
| 0x01E1 5020 - 0x01E1 50FF | -           | Reserved                                                |
| 0x01E1 5100               | FXD_MPSAR   | Fixed range start address                               |
| 0x01E1 5104               | FXD_MPEAR   | Fixed range end start address                           |
| 0x01E1 5108               | FXD_MPPA    | Fixed range memory page protection attributes           |
| 0x01E1 510C - 0x01E1 51FF | -           | Reserved                                                |
| 0x01E1 5200               | PROG1_MPSAR | Programmable range 1, start address                     |
| 0x01E1 5204               | PROG1_MPEAR | Programmable range 1, end address                       |
| 0x01E1 5208               | PROG1_MPPA  | Programmable range 1, memory page protection attributes |
| 0x01E1 520C - 0x01E1 520F | -           | Reserved                                                |
| 0x01E1 5210               | PROG2_MPSAR | Programmable range 2, start address                     |
| 0x01E1 5214               | PROG2_MPEAR | Programmable range 2, end address                       |
| 0x01E1 5218               | PROG2_MPPA  | Programmable range 2, memory page protection attributes |
| 0x01E1 521C - 0x01E1 521F | -           | Reserved                                                |
| 0x01E1 5220               | PROG3_MPSAR | Programmable range 3, start address                     |
| 0x01E1 5224               | PROG3_MPEAR | Programmable range 3, end address                       |
| 0x01E1 5228               | PROG3_MPPA  | Programmable range 3, memory page protection attributes |
| 0x01E1 522C - 0x01E1 522F | -           | Reserved                                                |
| 0x01E1 5230               | PROG4_MPSAR | Programmable range 4, start address                     |
| 0x01E1 5234               | PROG4_MPEAR | Programmable range 4, end address                       |
| 0x01E1 5238               | PROG4_MPPA  | Programmable range 4, memory page protection attributes |
| 0x01E1 523C - 0x01E1 523F | -           | Reserved                                                |
| 0x01E1 5240               | PROG5_MPSAR | Programmable range 5, start address                     |
| 0x01E1 5244               | PROG5_MPEAR | Programmable range 5, end address                       |
| 0x01E1 5248               | PROG5_MPPA  | Programmable range 5, memory page protection attributes |
| 0x01E1 524C - 0x01E1 524F | -           | Reserved                                                |
| 0x01E1 5250               | PROG6_MPSAR | Programmable range 6, start address                     |
| 0x01E1 5254               | PROG6_MPEAR | Programmable range 6, end address                       |
| 0x01E1 5258               | PROG6_MPPA  | Programmable range 6, memory page protection attributes |
| 0x01E1 525C - 0x01E1 525F | -           | Reserved                                                |
| 0x01E1 5260               | PROG7_MPSAR | Programmable range 7, start address                     |
| 0x01E1 5264               | PROG7_MPEAR | Programmable range 7, end address                       |
| 0x01E1 5268               | PROG7_MPPA  | Programmable range 7, memory page protection attributes |
| 0x01E1 526C - 0x01E1 526F | -           | Reserved                                                |

**Table 6-27. MPU2 Configuration Registers (continued)**

| MPU2<br>BYTE ADDRESS      | ACRONYM      | REGISTER DESCRIPTION                                     |
|---------------------------|--------------|----------------------------------------------------------|
| 0x01E1 5270               | PROG8_MPSAR  | Programmable range 8, start address                      |
| 0x01E1 5274               | PROG8_MPEAR  | Programmable range 8, end address                        |
| 0x01E1 5278               | PROG8_MPPA   | Programmable range 8, memory page protection attributes  |
| 0x01E1 527C - 0x01E1 527F | -            | Reserved                                                 |
| 0x01E1 5280               | PROG9_MPSAR  | Programmable range 9, start address                      |
| 0x01E1 5284               | PROG9_MPEAR  | Programmable range 9, end address                        |
| 0x01E1 5288               | PROG9_MPPA   | Programmable range 9, memory page protection attributes  |
| 0x01E1 528C - 0x01E1 528F | -            | Reserved                                                 |
| 0x01E1 5290               | PROG10_MPSAR | Programmable range 10, start address                     |
| 0x01E1 5294               | PROG10_MPEAR | Programmable range 10, end address                       |
| 0x01E1 5298               | PROG10_MPPA  | Programmable range 10, memory page protection attributes |
| 0x01E1 529C - 0x01E1 529F | -            | Reserved                                                 |
| 0x01E1 52A0               | PROG11_MPSAR | Programmable range 11, start address                     |
| 0x01E1 52A4               | PROG11_MPEAR | Programmable range 11, end address                       |
| 0x01E1 52A8               | PROG11_MPPA  | Programmable range 11, memory page protection attributes |
| 0x01E1 52AC - 0x01E1 52AF | -            | Reserved                                                 |
| 0x01E1 52B0               | PROG12_MPSAR | Programmable range 12, start address                     |
| 0x01E1 52B4               | PROG12_MPEAR | Programmable range 12, end address                       |
| 0x01E1 52B8               | PROG12_MPPA  | Programmable range 12, memory page protection attributes |
| 0x01E1 52BC - 0x01E1 52FF | -            | Reserved                                                 |
| 0x01E1 5300               | FLTADDRR     | Fault address                                            |
| 0x01E1 5304               | FLTSTAT      | Fault status                                             |
| 0x01E1 5308               | FLTCLR       | Fault clear                                              |
| 0x01E1 530C - 0x01E1 5FFF | -            | Reserved                                                 |

## 6.13 MMC / SD / SDIO (MMCSd)

### 6.13.1 MMCSd Peripheral Description

The device includes an MMCSd controller which is compliant with MMC V4.0, Secure Digital Part 1 Physical Layer Specification V1.1 and Secure Digital Input Output (SDIO) V2.0 specifications.

The MMC/SD Controller has following features:

- MultiMediaCard (MMC) support
- Secure Digital (SD) Memory Card support
- MMC/SD protocol support
- SD high capacity support
- SDIO protocol support
- Programmable clock frequency
- 512 bit Read/Write FIFO to lower system overhead
- Slave EDMA transfer capability

The device MMC/SD Controller does not support SPI mode.

### 6.13.2 MMCSd Peripheral Register Description(s)

**Table 6-28. Multimedia Card/Secure Digital (MMC/SD) Card Controller Registers**

| BYTE ADDRESS | ACRONYM          | REGISTER DESCRIPTION                  |
|--------------|------------------|---------------------------------------|
| 0x01C4 0000  | MMCCTL           | MMC Control Register                  |
| 0x01C4 0004  | MMCCLK           | MMC Memory Clock Control Register     |
| 0x01C4 0008  | MMCST0           | MMC Status Register 0                 |
| 0x01C4 000C  | MMCST1           | MMC Status Register 1                 |
| 0x01C4 0010  | MMCIM            | MMC Interrupt Mask Register           |
| 0x01C4 0014  | MMCTOR           | MMC Response Time-Out Register        |
| 0x01C4 0018  | MMCTOD           | MMC Data Read Time-Out Register       |
| 0x01C4 001C  | MMCBLEN          | MMC Block Length Register             |
| 0x01C4 0020  | MMCNBLK          | MMC Number of Blocks Register         |
| 0x01C4 0024  | MMCNBLC          | MMC Number of Blocks Counter Register |
| 0x01C4 0028  | MMCDRR           | MMC Data Receive Register             |
| 0x01C4 002C  | MMCDXR           | MMC Data Transmit Register            |
| 0x01C4 0030  | MMCCMD           | MMC Command Register                  |
| 0x01C4 0034  | MMCARGHL         | MMC Argument Register                 |
| 0x01C4 0038  | MMCRSP01         | MMC Response Register 0 and 1         |
| 0x01C4 003C  | MMCRSP23         | MMC Response Register 2 and 3         |
| 0x01C4 0040  | MMCRSP45         | MMC Response Register 4 and 5         |
| 0x01C4 0044  | MMCRSP67         | MMC Response Register 6 and 7         |
| 0x01C4 0048  | MMCDRSP          | MMC Data Response Register            |
| 0x01C4 0050  | MMCCIDX          | MMC Command Index Register            |
| 0x01C4 0064  | SDIOCTL          | SDIO Control Register                 |
| 0x01C4 0068  | SDIOST0          | SDIO Status Register 0                |
| 0x01C4 006C  | SDIOIEN          | SDIO Interrupt Enable Register        |
| 0x01C4 0070  | SDIOIST          | SDIO Interrupt Status Register        |
| 0x01C4 0074  | MMCFIFOCTL $\pi$ | MMC FIFO Control Register             |

### 6.13.3 MMC/SD Electrical Data/Timing

**Table 6-29. Timing Requirements for MMC/SD Module**  
(see [Figure 6-22](#) and [Figure 6-24](#))

| No. | PARAMETER                                                              | MIN | MAX | UNIT |
|-----|------------------------------------------------------------------------|-----|-----|------|
| 1   | $t_{su}(CMDV-CLKH)$ Setup time, MMCSD_CMD valid before MMCSD_CLK high  | 3.2 |     | ns   |
| 2   | $t_h(CLKH-CMDV)$ Hold time, MMCSD_CMD valid after MMCSD_CLK high       | 1.5 |     | ns   |
| 3   | $t_{su}(DATV-CLKH)$ Setup time, MMCSD_DATx valid before MMCSD_CLK high | 3.2 |     | ns   |
| 4   | $t_h(CLKH-DATV)$ Hold time, MMCSD_DATx valid after MMCSD_CLK high      | 1.5 |     | ns   |

**Table 6-30. Switching Characteristics Over Recommended Operating Conditions for MMC/SD Module**  
(see [Figure 6-21](#) through [Figure 6-24](#))

| No. | PARAMETER                                                          | MIN  | MAX | UNIT |
|-----|--------------------------------------------------------------------|------|-----|------|
| 7   | $f_{(CLK)}$ Operating frequency, MMCSD_CLK                         | 0    | 52  | MHz  |
| 8   | $f_{(CLK\_ID)}$ Identification mode frequency, MMCSD_CLK           | 0    | 400 | KHz  |
| 9   | $t_{W(CLKL)}$ Pulse width, MMCSD_CLK low                           | 6.5  |     | ns   |
| 10  | $t_{W(CLKH)}$ Pulse width, MMCSD_CLK high                          | 6.5  |     | ns   |
| 11  | $t_r(CLK)$ Rise time, MMCSD_CLK                                    |      | 3   | ns   |
| 12  | $t_f(CLK)$ Fall time, MMCSD_CLK                                    |      | 3   | ns   |
| 13  | $t_d(CLKL-CMD)$ Delay time, MMCSD_CLK low to MMCSD_CMD transition  | -4.5 | 2.5 | ns   |
| 14  | $t_d(CLKL-DAT)$ Delay time, MMCSD_CLK low to MMCSD_DATx transition | -4.5 | 2.5 | ns   |

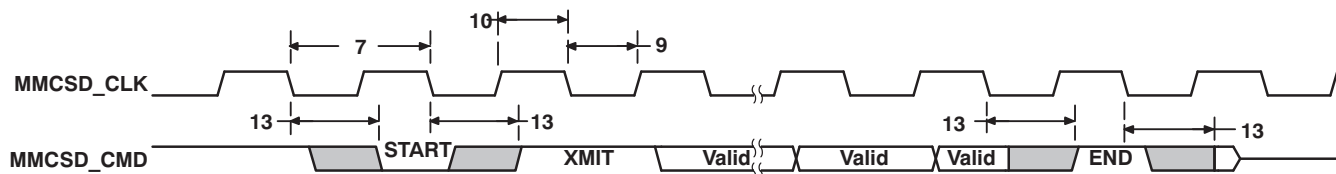


Figure 6-21. MMC/SD Host Command Timing

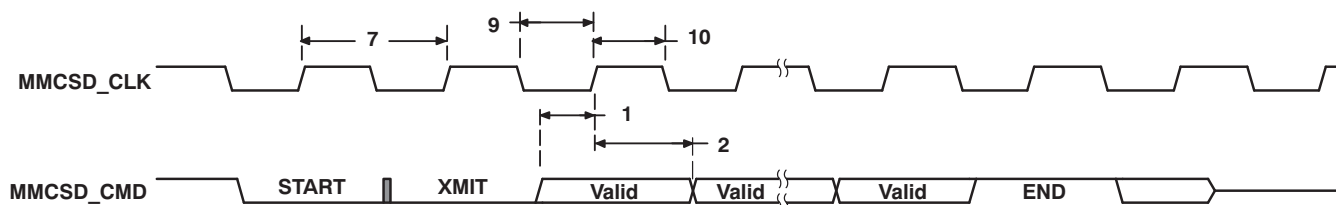


Figure 6-22. MMC/SD Card Response Timing

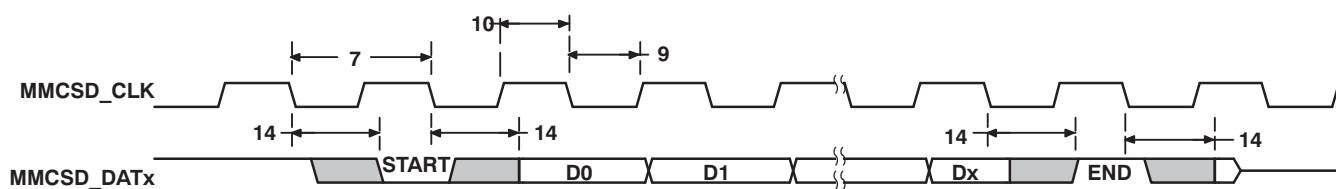


Figure 6-23. MMC/SD Host Write Timing

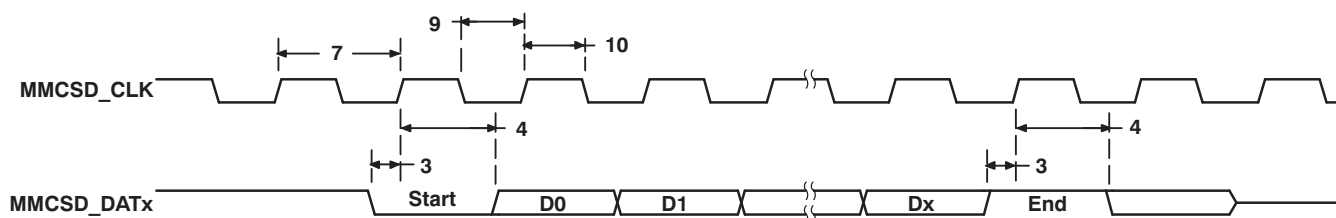


Figure 6-24. MMC/SD Host Read and Card CRC Status Timing

## 6.14 Ethernet Media Access Controller (EMAC)

The Ethernet Media Access Controller (EMAC) provides an efficient interface between the device and the network. The EMAC supports both 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 EMAC controls the flow of packet data from the device to the PHY. The MDIO module controls PHY configuration and status monitoring.

Both the EMAC and the MDIO modules interface to the 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.

### 6.14.1 EMAC Peripheral Register Description(s)

**Table 6-31. Ethernet Media Access Controller (EMAC) Registers**

| BYTE ADDRESS | ACRONYM           | REGISTER DESCRIPTION                                            |
|--------------|-------------------|-----------------------------------------------------------------|
| 0x01E2 3000  | TXREV             | Transmit Revision Register                                      |
| 0x01E2 3004  | TXCONTROL         | Transmit Control Register                                       |
| 0x01E2 3008  | TXTEARDOWN        | Transmit Teardown Register                                      |
| 0x01E2 3010  | RXREV             | Receive Revision Register                                       |
| 0x01E2 3014  | RXCONTROL         | Receive Control Register                                        |
| 0x01E2 3018  | RXTEARDOWN        | Receive Teardown Register                                       |
| 0x01E2 3080  | TXINTSTATRAW      | Transmit Interrupt Status (Unmasked) Register                   |
| 0x01E2 3084  | TXINTSTATMASKED   | Transmit Interrupt Status (Masked) Register                     |
| 0x01E2 3088  | TXINTMASKSET      | Transmit Interrupt Mask Set Register                            |
| 0x01E2 308C  | TXINTMASKCLEAR    | Transmit Interrupt Mask Clear Register                          |
| 0x01E2 3090  | MACINVECTOR       | MAC Input Vector Register                                       |
| 0x01E2 3094  | MACEOIVECTOR      | MAC End Of Interrupt Vector Register                            |
| 0x01E2 30A0  | RXINTSTATRAW      | Receive Interrupt Status (Unmasked) Register                    |
| 0x01E2 30A4  | RXINTSTATMASKED   | Receive Interrupt Status (Masked) Register                      |
| 0x01E2 30A8  | RXINTMASKSET      | Receive Interrupt Mask Set Register                             |
| 0x01E2 30AC  | RXINTMASKCLEAR    | Receive Interrupt Mask Clear Register                           |
| 0x01E2 30B0  | MACINTSTATRAW     | MAC Interrupt Status (Unmasked) Register                        |
| 0x01E2 30B4  | MACINTSTATMASKED  | MAC Interrupt Status (Masked) Register                          |
| 0x01E2 30B8  | MACINTMASKSET     | MAC Interrupt Mask Set Register                                 |
| 0x01E2 30BC  | MACINTMASKCLEAR   | MAC Interrupt Mask Clear Register                               |
| 0x01E2 3100  | RXMBPENABLE       | Receive Multicast/Broadcast/Promiscuous Channel Enable Register |
| 0x01E2 3104  | RXUNICASTSET      | Receive Unicast Enable Set Register                             |
| 0x01E2 3108  | RXUNICASTCLEAR    | Receive Unicast Clear Register                                  |
| 0x01E2 310C  | RXMAXLEN          | Receive Maximum Length Register                                 |
| 0x01E2 3110  | RXBUFFEROFFSET    | Receive Buffer Offset Register                                  |
| 0x01E2 3114  | RXFILTERLOWTHRESH | Receive Filter Low Priority Frame Threshold Register            |
| 0x01E2 3120  | RX0FLOWTHRESH     | Receive Channel 0 Flow Control Threshold Register               |
| 0x01E2 3124  | RX1FLOWTHRESH     | Receive Channel 1 Flow Control Threshold Register               |
| 0x01E2 3128  | RX2FLOWTHRESH     | Receive Channel 2 Flow Control Threshold Register               |
| 0x01E2 312C  | RX3FLOWTHRESH     | Receive Channel 3 Flow Control Threshold Register               |
| 0x01E2 3130  | RX4FLOWTHRESH     | Receive Channel 4 Flow Control Threshold Register               |
| 0x01E2 3134  | RX5FLOWTHRESH     | Receive Channel 5 Flow Control Threshold Register               |
| 0x01E2 3138  | RX6FLOWTHRESH     | Receive Channel 6 Flow Control Threshold Register               |
| 0x01E2 313C  | RX7FLOWTHRESH     | Receive Channel 7 Flow Control Threshold Register               |

**Table 6-31. Ethernet Media Access Controller (EMAC) Registers (continued)**

| BYTE ADDRESS              | ACRONYM                           | REGISTER DESCRIPTION                                              |
|---------------------------|-----------------------------------|-------------------------------------------------------------------|
| 0x01E2 3140               | RX0FREEBUFFER                     | Receive Channel 0 Free Buffer Count Register                      |
| 0x01E2 3144               | RX1FREEBUFFER                     | Receive Channel 1 Free Buffer Count Register                      |
| 0x01E2 3148               | RX2FREEBUFFER                     | Receive Channel 2 Free Buffer Count Register                      |
| 0x01E2 314C               | RX3FREEBUFFER                     | Receive Channel 3 Free Buffer Count Register                      |
| 0x01E2 3150               | RX4FREEBUFFER                     | Receive Channel 4 Free Buffer Count Register                      |
| 0x01E2 3154               | RX5FREEBUFFER                     | Receive Channel 5 Free Buffer Count Register                      |
| 0x01E2 3158               | RX6FREEBUFFER                     | Receive Channel 6 Free Buffer Count Register                      |
| 0x01E2 315C               | RX7FREEBUFFER                     | Receive Channel 7 Free Buffer Count Register                      |
| 0x01E2 3160               | MACCONTROL                        | MAC Control Register                                              |
| 0x01E2 3164               | MACSTATUS                         | MAC Status Register                                               |
| 0x01E2 3168               | EMCONTROL                         | Emulation Control Register                                        |
| 0x01E2 316C               | FIFOCONTROL                       | FIFO Control Register                                             |
| 0x01E2 3170               | MACCONFIG                         | MAC Configuration Register                                        |
| 0x01E2 3174               | SOFTRESET                         | Soft Reset Register                                               |
| 0x01E2 31D0               | MACSRCADDRLO                      | MAC Source Address Low Bytes Register                             |
| 0x01E2 31D4               | MACSRCADDRHI                      | MAC Source Address High Bytes Register                            |
| 0x01E2 31D8               | MACHASH1                          | MAC Hash Address Register 1                                       |
| 0x01E2 31DC               | MACHASH2                          | MAC Hash Address Register 2                                       |
| 0x01E2 31E0               | BOFFTEST                          | Back Off Test Register                                            |
| 0x01E2 31E4               | TPACETEST                         | Transmit Pacing Algorithm Test Register                           |
| 0x01E2 31E8               | RXPAUSE                           | Receive Pause Timer Register                                      |
| 0x01E2 31EC               | TXPAUSE                           | Transmit Pause Timer Register                                     |
| 0x01E2 3200 - 0x01E2 32FC | (see <a href="#">Table 6-32</a> ) | EMAC Statistics Registers                                         |
| 0x01E2 3500               | MACADDRLO                         | MAC Address Low Bytes Register, Used in Receive Address Matching  |
| 0x01E2 3504               | MACADDRHI                         | MAC Address High Bytes Register, Used in Receive Address Matching |
| 0x01E2 3508               | MACINDEX                          | MAC Index Register                                                |
| 0x01E2 3600               | TX0HDP                            | Transmit Channel 0 DMA Head Descriptor Pointer Register           |
| 0x01E2 3604               | TX1HDP                            | Transmit Channel 1 DMA Head Descriptor Pointer Register           |
| 0x01E2 3608               | TX2HDP                            | Transmit Channel 2 DMA Head Descriptor Pointer Register           |
| 0x01E2 360C               | TX3HDP                            | Transmit Channel 3 DMA Head Descriptor Pointer Register           |
| 0x01E2 3610               | TX4HDP                            | Transmit Channel 4 DMA Head Descriptor Pointer Register           |
| 0x01E2 3614               | TX5HDP                            | Transmit Channel 5 DMA Head Descriptor Pointer Register           |
| 0x01E2 3618               | TX6HDP                            | Transmit Channel 6 DMA Head Descriptor Pointer Register           |
| 0x01E2 361C               | TX7HDP                            | Transmit Channel 7 DMA Head Descriptor Pointer Register           |
| 0x01E2 3620               | RX0HDP                            | Receive Channel 0 DMA Head Descriptor Pointer Register            |
| 0x01E2 3624               | RX1HDP                            | Receive Channel 1 DMA Head Descriptor Pointer Register            |
| 0x01E2 3628               | RX2HDP                            | Receive Channel 2 DMA Head Descriptor Pointer Register            |
| 0x01E2 362C               | RX3HDP                            | Receive Channel 3 DMA Head Descriptor Pointer Register            |
| 0x01E2 3630               | RX4HDP                            | Receive Channel 4 DMA Head Descriptor Pointer Register            |
| 0x01E2 3634               | RX5HDP                            | Receive Channel 5 DMA Head Descriptor Pointer Register            |
| 0x01E2 3638               | RX6HDP                            | Receive Channel 6 DMA Head Descriptor Pointer Register            |
| 0x01E2 363C               | RX7HDP                            | Receive Channel 7 DMA Head Descriptor Pointer Register            |
| 0x01E2 3640               | TX0CP                             | Transmit Channel 0 Completion Pointer Register                    |
| 0x01E2 3644               | TX1CP                             | Transmit Channel 1 Completion Pointer Register                    |
| 0x01E2 3648               | TX2CP                             | Transmit Channel 2 Completion Pointer Register                    |
| 0x01E2 364C               | TX3CP                             | Transmit Channel 3 Completion Pointer Register                    |
| 0x01E2 3650               | TX4CP                             | Transmit Channel 4 Completion Pointer Register                    |

**Table 6-31. Ethernet Media Access Controller (EMAC) Registers (continued)**

| BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION                           |
|--------------|---------|------------------------------------------------|
| 0x01E2 3654  | TX5CP   | Transmit Channel 5 Completion Pointer Register |
| 0x01E2 3658  | TX6CP   | Transmit Channel 6 Completion Pointer Register |
| 0x01E2 365C  | TX7CP   | Transmit Channel 7 Completion Pointer Register |
| 0x01E2 3660  | RX0CP   | Receive Channel 0 Completion Pointer Register  |
| 0x01E2 3664  | RX1CP   | Receive Channel 1 Completion Pointer Register  |
| 0x01E2 3668  | RX2CP   | Receive Channel 2 Completion Pointer Register  |
| 0x01E2 366C  | RX3CP   | Receive Channel 3 Completion Pointer Register  |
| 0x01E2 3670  | RX4CP   | Receive Channel 4 Completion Pointer Register  |
| 0x01E2 3674  | RX5CP   | Receive Channel 5 Completion Pointer Register  |
| 0x01E2 3678  | RX6CP   | Receive Channel 6 Completion Pointer Register  |
| 0x01E2 367C  | RX7CP   | Receive Channel 7 Completion Pointer Register  |

**Table 6-32. EMAC Statistics Registers**

| BYTE ADDRESS | ACRONYM           | REGISTER DESCRIPTION                                                                                   |
|--------------|-------------------|--------------------------------------------------------------------------------------------------------|
| 0x01E2 3200  | RXGOODFRAMES      | Good Receive Frames Register                                                                           |
| 0x01E2 3204  | RXBCASTFRAMES     | Broadcast Receive Frames Register<br>(Total number of good broadcast frames received)                  |
| 0x01E2 3208  | RXMCASTFRAMES     | Multicast Receive Frames Register<br>(Total number of good multicast frames received)                  |
| 0x01E2 320C  | RXPAUSEFRAMES     | Pause Receive Frames Register                                                                          |
| 0x01E2 3210  | RXCRCERRORS       | Receive CRC Errors Register<br>(Total number of frames received with CRC errors)                       |
| 0x01E2 3214  | RXALIGNCODEERRORS | Receive Alignment/Code Errors Register<br>(Total number of frames received with alignment/code errors) |
| 0x01E2 3218  | RXOVERSIZED       | Receive Oversized Frames Register<br>(Total number of oversized frames received)                       |
| 0x01E2 321C  | RXJABBER          | Receive Jabber Frames Register<br>(Total number of jabber frames received)                             |
| 0x01E2 3220  | RXUNDERSIZED      | Receive Undersized Frames Register<br>(Total number of undersized frames received)                     |
| 0x01E2 3224  | RXFRAGMENTS       | Receive Frame Fragments Register                                                                       |
| 0x01E2 3228  | RXFILTERED        | Filtered Receive Frames Register                                                                       |
| 0x01E2 322C  | RXQOSFILTERED     | Received QOS Filtered Frames Register                                                                  |
| 0x01E2 3230  | RXOCTETS          | Receive Octet Frames Register<br>(Total number of received bytes in good frames)                       |
| 0x01E2 3234  | TXGOODFRAMES      | Good Transmit Frames Register<br>(Total number of good frames transmitted)                             |
| 0x01E2 3238  | TXBCASTFRAMES     | Broadcast Transmit Frames Register                                                                     |
| 0x01E2 323C  | TXMCASTFRAMES     | Multicast Transmit Frames Register                                                                     |
| 0x01E2 3240  | TXPAUSEFRAMES     | Pause Transmit Frames Register                                                                         |
| 0x01E2 3244  | TXDEFERRED        | Deferred Transmit Frames Register                                                                      |
| 0x01E2 3248  | TXCOLLISION       | Transmit Collision Frames Register                                                                     |
| 0x01E2 324C  | TXSINGLECOLL      | Transmit Single Collision Frames Register                                                              |
| 0x01E2 3250  | TXMULTICOLL       | Transmit Multiple Collision Frames Register                                                            |
| 0x01E2 3254  | TXEXCESSIVECOLL   | Transmit Excessive Collision Frames Register                                                           |
| 0x01E2 3258  | TXLATECOLL        | Transmit Late Collision Frames Register                                                                |
| 0x01E2 325C  | TXUNDERRUN        | Transmit Underrun Error Register                                                                       |
| 0x01E2 3260  | TXCARRIERSENSE    | Transmit Carrier Sense Errors Register                                                                 |
| 0x01E2 3264  | TXOCTETS          | Transmit Octet Frames Register                                                                         |
| 0x01E2 3268  | FRAME64           | Transmit and Receive 64 Octet Frames Register                                                          |

**Table 6-32. EMAC Statistics Registers (continued)**

| BYTE ADDRESS | ACRONYM       | REGISTER DESCRIPTION                                             |
|--------------|---------------|------------------------------------------------------------------|
| 0x01E2 326C  | FRAME65T127   | Transmit and Receive 65 to 127 Octet Frames Register             |
| 0x01E2 3270  | FRAME128T255  | Transmit and Receive 128 to 255 Octet Frames Register            |
| 0x01E2 3274  | FRAME256T511  | Transmit and Receive 256 to 511 Octet Frames Register            |
| 0x01E2 3278  | FRAME512T1023 | Transmit and Receive 512 to 1023 Octet Frames Register           |
| 0x01E2 327C  | FRAME1024TUP  | Transmit and Receive 1024 to 1518 Octet Frames Register          |
| 0x01E2 3280  | NETOCTETS     | Network Octet Frames Register                                    |
| 0x01E2 3284  | RXSOFOVERRUNS | Receive FIFO or DMA Start of Frame Overruns Register             |
| 0x01E2 3288  | RXMOFOVERRUNS | Receive FIFO or DMA Middle of Frame Overruns Register            |
| 0x01E2 328C  | RXDMAOVERRUNS | Receive DMA Start of Frame and Middle of Frame Overruns Register |

**Table 6-33. EMAC Control Module Registers**

| BYTE ADDRESS | ACRONYM        | REGISTER DESCRIPTION                                                              |
|--------------|----------------|-----------------------------------------------------------------------------------|
| 0x01E2 2000  | REV            | EMAC Control Module Revision Register                                             |
| 0x01E2 2004  | SOFTRESET      | EMAC Control Module Software Reset Register                                       |
| 0x01E2 200C  | INTCONTROL     | EMAC Control Module Interrupt Control Register                                    |
| 0x01E2 2010  | C0RXTHRESHEN   | EMAC Control Module Interrupt Core 0 Receive Threshold Interrupt Enable Register  |
| 0x01E2 2014  | C0RXEN         | EMAC Control Module Interrupt Core 0 Receive Interrupt Enable Register            |
| 0x01E2 2018  | C0TXEN         | EMAC Control Module Interrupt Core 0 Transmit Interrupt Enable Register           |
| 0x01E2 201C  | C0MISCEN       | EMAC Control Module Interrupt Core 0 Miscellaneous Interrupt Enable Register      |
| 0x01E2 2020  | C1RXTHRESHEN   | EMAC Control Module Interrupt Core 1 Receive Threshold Interrupt Enable Register  |
| 0x01E2 2024  | C1RXEN         | EMAC Control Module Interrupt Core 1 Receive Interrupt Enable Register            |
| 0x01E2 2028  | C1TXEN         | EMAC Control Module Interrupt Core 1 Transmit Interrupt Enable Register           |
| 0x01E2 202C  | C1MISCEN       | EMAC Control Module Interrupt Core 1 Miscellaneous Interrupt Enable Register      |
| 0x01E2 2030  | C2RXTHRESHEN   | EMAC Control Module Interrupt Core 2 Receive Threshold Interrupt Enable Register  |
| 0x01E2 2034  | C2RXEN         | EMAC Control Module Interrupt Core 2 Receive Interrupt Enable Register            |
| 0x01E2 2038  | C2TXEN         | EMAC Control Module Interrupt Core 2 Transmit Interrupt Enable Register           |
| 0x01E2 203C  | C2MISCEN       | EMAC Control Module Interrupt Core 2 Miscellaneous Interrupt Enable Register      |
| 0x01E2 2040  | C0RXTHRESHSTAT | EMAC Control Module Interrupt Core 0 Receive Threshold Interrupt Status Register  |
| 0x01E2 2044  | C0RXSTAT       | EMAC Control Module Interrupt Core 0 Receive Interrupt Status Register            |
| 0x01E2 2048  | C0TXSTAT       | EMAC Control Module Interrupt Core 0 Transmit Interrupt Status Register           |
| 0x01E2 204C  | C0MISCSTAT     | EMAC Control Module Interrupt Core 0 Miscellaneous Interrupt Status Register      |
| 0x01E2 2050  | C1RXTHRESHSTAT | EMAC Control Module Interrupt Core 1 Receive Threshold Interrupt Status Register  |
| 0x01E2 2054  | C1RXSTAT       | EMAC Control Module Interrupt Core 1 Receive Interrupt Status Register            |
| 0x01E2 2058  | C1TXSTAT       | EMAC Control Module Interrupt Core 1 Transmit Interrupt Status Register           |
| 0x01E2 205C  | C1MISCSTAT     | EMAC Control Module Interrupt Core 1 Miscellaneous Interrupt Status Register      |
| 0x01E2 2060  | C2RXTHRESHSTAT | EMAC Control Module Interrupt Core 2 Receive Threshold Interrupt Status Register  |
| 0x01E2 2064  | C2RXSTAT       | EMAC Control Module Interrupt Core 2 Receive Interrupt Status Register            |
| 0x01E2 2068  | C2TXSTAT       | EMAC Control Module Interrupt Core 2 Transmit Interrupt Status Register           |
| 0x01E2 206C  | C2MISCSTAT     | EMAC Control Module Interrupt Core 2 Miscellaneous Interrupt Status Register      |
| 0x01E2 2070  | C0RXIMAX       | EMAC Control Module Interrupt Core 0 Receive Interrupts Per Millisecond Register  |
| 0x01E2 2074  | C0TXIMAX       | EMAC Control Module Interrupt Core 0 Transmit Interrupts Per Millisecond Register |
| 0x01E2 2078  | C1RXIMAX       | EMAC Control Module Interrupt Core 1 Receive Interrupts Per Millisecond Register  |
| 0x01E2 207C  | C1TXIMAX       | EMAC Control Module Interrupt Core 1 Transmit Interrupts Per Millisecond Register |
| 0x01E2 2080  | C2RXIMAX       | EMAC Control Module Interrupt Core 2 Receive Interrupts Per Millisecond Register  |
| 0x01E2 2084  | C2TXIMAX       | EMAC Control Module Interrupt Core 2 Transmit Interrupts Per Millisecond Register |

**Table 6-34. EMAC Control Module RAM**

| HEX ADDRESS RANGE         |                                     |
|---------------------------|-------------------------------------|
| 0x01E2 0000 - 0x01E2 1FFF | EMAC Local Buffer Descriptor Memory |

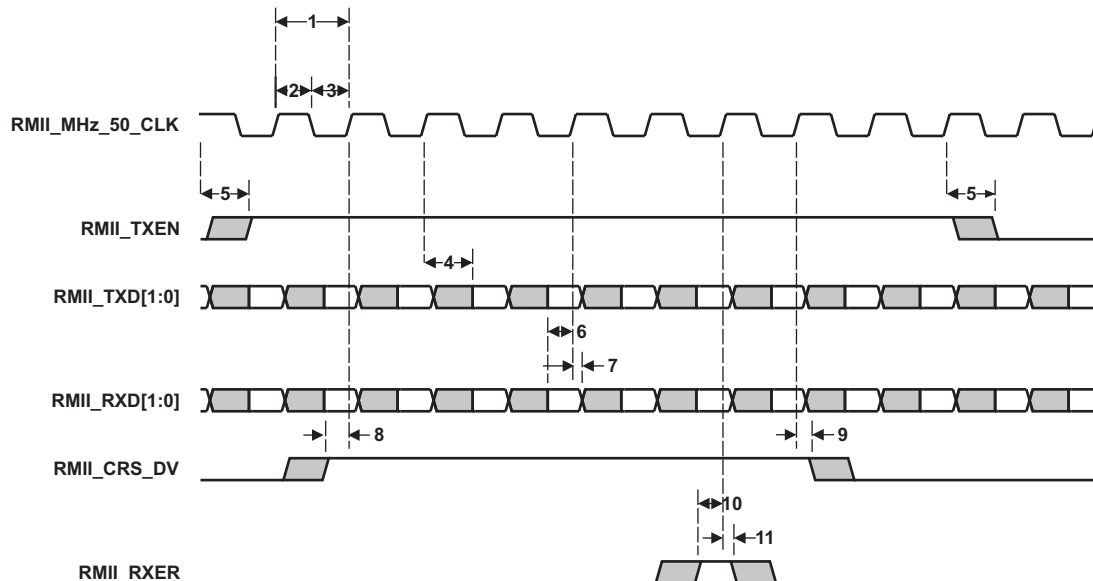
**Table 6-35. RMII Timing Requirements**

| No. | PARAMETER                                                                     | MIN | TYP | MAX | UNIT |
|-----|-------------------------------------------------------------------------------|-----|-----|-----|------|
| 1   | tc(REFCLK) Cycle Time, RMII_MHZ_50_CLK <sup>(1)</sup>                         |     | 20  |     | ns   |
| 2   | tw(REFCLKH) Pulse Width, RMII_MHZ_50_CLK High                                 | 7   |     | 13  | ns   |
| 3   | tw(REFCLKL) Pulse Width, RMII_MHZ_50_CLK Low                                  | 7   |     | 13  | ns   |
| 6   | tsu(RXD-REFCLK) Input Setup Time, RXD Valid before RMII_MHZ_50_CLK High       | 4   |     |     | ns   |
| 7   | th(REFCLK-RXD) Input Hold Time, RXD Valid after RMII_MHZ_50_CLK High          | 2   |     |     | ns   |
| 8   | tsu(CRS DV-REFCLK) Input Setup Time, CRS DV Valid before RMII_MHZ_50_CLK High | 4   |     |     | ns   |
| 9   | th(REFCLK-CRS DV) Input Hold Time, CRS DV Valid after RMII_MHZ_50_CLK High    | 2   |     |     | ns   |
| 10  | tsu(RXER-REFCLK) Input Setup Time, RXER Valid before RMII_MHZ_50_CLK High     | 4   |     |     | ns   |
| 11  | th(REFCLK-RXER) Input Hold Time, RXER Valid after RMII_MHZ_50_CLK High        | 2   |     |     | ns   |

(1) Per the RMII industry specification, the RMII reference clock (RMII\_MHZ\_50\_CLK) must have jitter tolerance of 50 ppm or less.

**Table 6-36. RMII Switching Characteristics**

| No. | PARAMETER                                                             | MIN | TYP | MAX | UNIT |
|-----|-----------------------------------------------------------------------|-----|-----|-----|------|
| 4   | td(REFCLK-TXD) Output Delay Time, RMII_MHZ_50_CLK High to TXD Valid   | 2.5 |     | 13  | ns   |
| 5   | td(REFCLK-TXEN) Output Delay Time, RMII_MHZ_50_CLK High to TXEN Valid | 2.5 |     | 13  | ns   |



**Figure 6-25. RMII Timing Diagram**

## 6.15 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.

### 6.15.1 MDIO Registers

For a list of supported MDIO registers see [Table 6-37](#) [MDIO Registers].

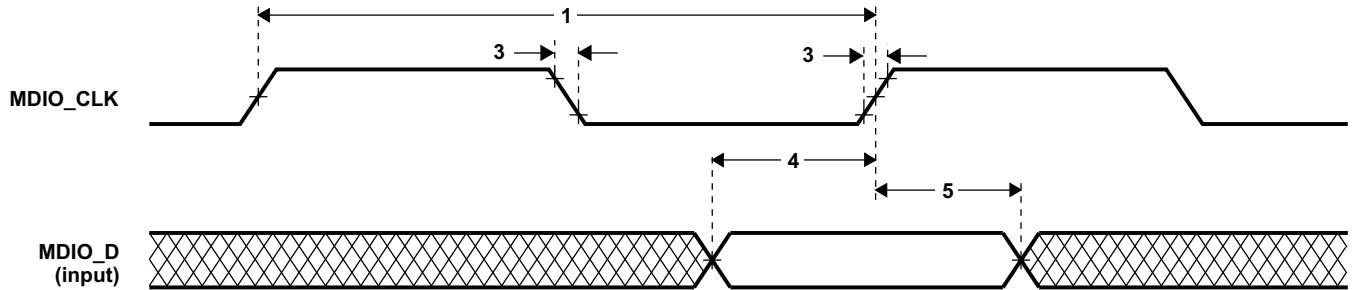
**Table 6-37. MDIO Register Memory Map**

| BYTE ADDRESS              | ACRONYM          | REGISTER DESCRIPTION                                     |
|---------------------------|------------------|----------------------------------------------------------|
| 0x01E2 4000               | REV              | Revision Identification Register                         |
| 0x01E2 4004               | CONTROL          | MDIO Control Register                                    |
| 0x01E2 4008               | ALIVE            | MDIO PHY Alive Status Register                           |
| 0x01E2 400C               | LINK             | MDIO PHY Link Status Register                            |
| 0x01E2 4010               | LINKINTRAW       | MDIO Link Status Change Interrupt (Unmasked) Register    |
| 0x01E2 4014               | LINKINTMASKED    | MDIO Link Status Change Interrupt (Masked) Register      |
| 0x01E2 4018               | –                | Reserved                                                 |
| 0x01E2 4020               | USERINTRAW       | MDIO User Command Complete Interrupt (Unmasked) Register |
| 0x01E2 4024               | USERINTMASKED    | MDIO User Command Complete Interrupt (Masked) Register   |
| 0x01E2 4028               | USERINTMASKSET   | MDIO User Command Complete Interrupt Mask Set Register   |
| 0x01E2 402C               | USERINTMASKCLEAR | MDIO User Command Complete Interrupt Mask Clear Register |
| 0x01E2 4030 - 0x01E2 407C | –                | Reserved                                                 |
| 0x01E2 4080               | USERACCESS0      | MDIO User Access Register 0                              |
| 0x01E2 4084               | USERPHYSEL0      | MDIO User PHY Select Register 0                          |
| 0x01E2 4088               | USERACCESS1      | MDIO User Access Register 1                              |
| 0x01E2 408C               | USERPHYSEL1      | MDIO User PHY Select Register 1                          |
| 0x01E2 4090 - 0x01E2 47FF | –                | Reserved                                                 |

### 6.15.2 Management Data Input/Output (MDIO) Electrical Data/Timing

**Table 6-38. Timing Requirements for MDIO Input (see Figure 6-26 and Figure 6-27)**

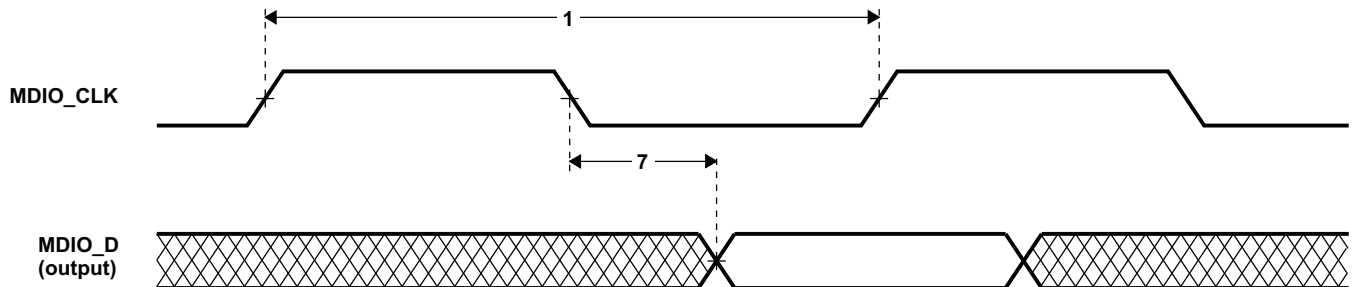
| No. | PARAMETER                        |                                                          | MIN | MAX | UNIT |
|-----|----------------------------------|----------------------------------------------------------|-----|-----|------|
| 1   | $t_c(\text{MDIO\_CLK})$          | Cycle time, MDIO_CLK                                     | 400 |     | ns   |
| 2   | $t_w(\text{MDIO\_CLK})$          | Pulse duration, MDIO_CLK high/low                        | 180 |     | ns   |
| 3   | $t_t(\text{MDIO\_CLK})$          | Transition time, MDIO_CLK                                |     | 5   | ns   |
| 4   | $t_{su}(\text{MDIO-MDIO\_CLKH})$ | Setup time, MDIO_D data input valid before MDIO_CLK high | 10  |     | ns   |
| 5   | $t_h(\text{MDIO\_CLKH-MDIO})$    | Hold time, MDIO_D data input valid after MDIO_CLK high   | 0   |     | ns   |



**Figure 6-26. MDIO Input Timing**

**Table 6-39. Switching Characteristics Over Recommended Operating Conditions for MDIO Output (see Figure 6-27)**

| No. | PARAMETER                     |                                                      | MIN | MAX | UNIT |
|-----|-------------------------------|------------------------------------------------------|-----|-----|------|
| 7   | $t_d(\text{MDIO\_CLKL-MDIO})$ | Delay time, MDIO_CLK low to MDIO_D data output valid | 0   | 100 | ns   |



**Figure 6-27. MDIO Output Timing**

## 6.16 Multichannel Audio Serial Ports (McASP0, McASP1)

The McASP serial port is specifically designed for multichannel audio applications. Its key features are:

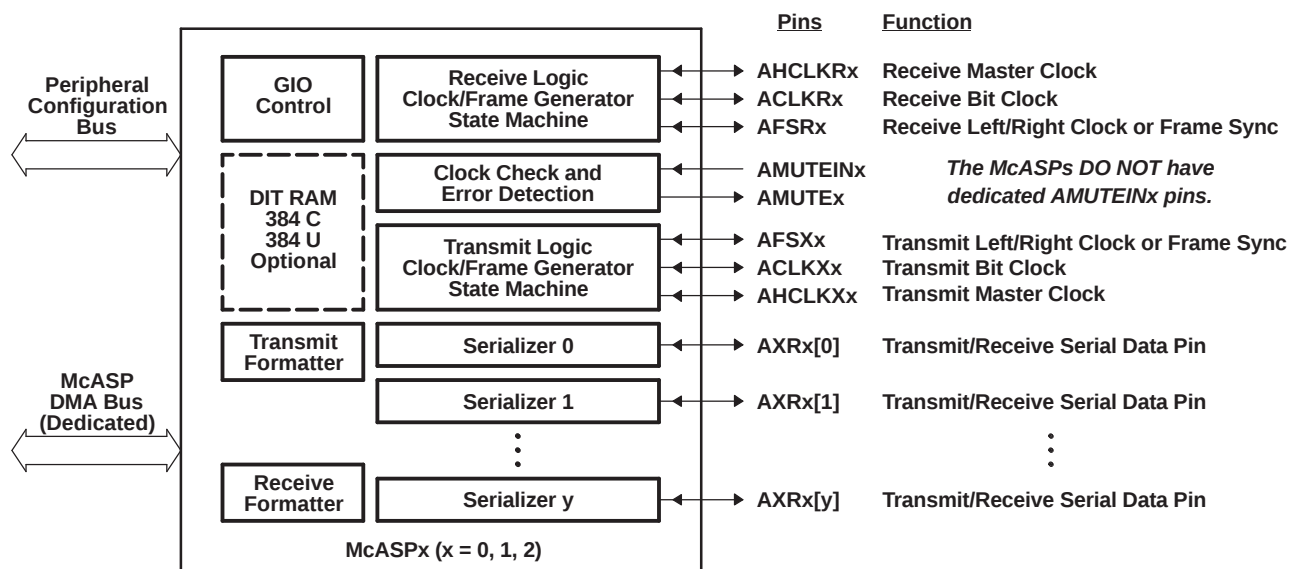
- Flexible clock and frame sync generation logic and on-chip dividers
- Up to sixteen transmit or receive data pins and serializers
- Large number of serial data format options, including:
  - TDM Frames with 2 to 32 time slots per frame (periodic) or 1 slot per frame (burst)
  - Time slots of 8, 12, 16, 20, 24, 28, and 32 bits
  - First bit delay 0, 1, or 2 clocks
  - MSB or LSB first bit order
  - Left- or right-aligned data words within time slots
- DIT Mode (optional) with 384-bit Channel Status and 384-bit User Data registers
- Extensive error checking and mute generation logic
- All unused pins GPIO-capable
- Transmit & Receive FIFO Buffers for each McASP. Allows the McASP to operate at a higher sample rate by making it more tolerant to DMA latency.
- Dynamic Adjustment of Clock Dividers
  - Clock Divider Value may be changed without resetting the McASP

The McASPs on the device are configured with the following options:

**Table 6-40. McASP Configurations<sup>(1)</sup>**

| Module | Serializers | AFIFO                    | DIT | Pins                                                                  |
|--------|-------------|--------------------------|-----|-----------------------------------------------------------------------|
| McASP0 | 16          | 64 Word RX<br>64 Word TX | N   | AXR0[13:0], AHCLKR0, ACLKR0, AFSR0, AHCLKX0, ACLKX0, AFSX0            |
| McASP1 | 12          | 64 Word RX<br>64 Word TX | N   | AXR1[11:10], AXR1[8:0], ACLKR1, AFSR1, AHCLKX1, ACLKX1, AFSX1, AMUTE1 |

(1) Pins available are the maximum number of pins that may be configured for a particular McASP; not including pin multiplexing.



**Figure 6-28. McASP Block Diagram**

### 6.16.1 McASP Peripheral Registers Description(s)

Registers for the McASP are summarized in [Table 6-41](#). The registers are accessed through the peripheral configuration port. The receive buffer registers (RBUF) and transmit buffer registers (XBUF) can also be accessed through the DMA port, as listed in [Table 6-42](#)

Registers for the McASP Audio FIFO (AFIFO) are summarized in [Table 6-43](#). Note that the AFIFO Write FIFO (WFIFO) and Read FIFO (RFIFO) have independent control and status registers. The AFIFO control registers are accessed through the peripheral configuration port.

**Table 6-41. McASP Registers Accessed Through Peripheral Configuration Port**

| McASP0<br>BYTE<br>ADDRESS | McASP1<br>BYTE<br>ADDRESS | ACRONYM   | REGISTER DESCRIPTION                                                                                                                               |
|---------------------------|---------------------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x01D0 0000               | 0x01D0 4000               | REV       | Revision identification register                                                                                                                   |
| 0x01D0 0010               | 0x01D0 4010               | PFUNC     | Pin function register                                                                                                                              |
| 0x01D0 0014               | 0x01D0 4014               | PDIR      | Pin direction register                                                                                                                             |
| 0x01D0 0018               | 0x01D0 4018               | PDOUT     | Pin data output register                                                                                                                           |
| 0x01D0 001C               | 0x01D0 401C               | PDIN      | Read returns: Pin data input register                                                                                                              |
| 0x01D0 001C               | 0x01D0 401C               | PDSET     | Writes affect: Pin data set register (alternate write address: PDOUT)                                                                              |
| 0x01D0 0020               | 0x01D0 4020               | PDCLR     | Pin data clear register (alternate write address: PDOUT)                                                                                           |
| 0x01D0 0044               | 0x01D0 4044               | GBLCTL    | Global control register                                                                                                                            |
| 0x01D0 0048               | 0x01D0 4048               | AMUTE     | Audio mute control register                                                                                                                        |
| 0x01D0 004C               | 0x01D0 404C               | DLBCTL    | Digital loopback control register                                                                                                                  |
| 0x01D0 0050               | 0x01D0 4050               | DITCTL    | DIT mode control register                                                                                                                          |
| 0x01D0 0060               | 0x01D0 4060               | RGBLCTL   | Receiver global control register: Alias of GBLCTL, only receive bits are affected - allows receiver to be reset independently from transmitter     |
| 0x01D0 0064               | 0x01D0 4064               | RMASK     | Receive format unit bit mask register                                                                                                              |
| 0x01D0 0068               | 0x01D0 4068               | RFMT      | Receive bit stream format register                                                                                                                 |
| 0x01D0 006C               | 0x01D0 406C               | AFSRCTL   | Receive frame sync control register                                                                                                                |
| 0x01D0 0070               | 0x01D0 4070               | ACLKRCTL  | Receive clock control register                                                                                                                     |
| 0x01D0 0074               | 0x01D0 4074               | AHCLKRCTL | Receive high-frequency clock control register                                                                                                      |
| 0x01D0 0078               | 0x01D0 4078               | RTDM      | Receive TDM time slot 0-31 register                                                                                                                |
| 0x01D0 007C               | 0x01D0 407C               | RINTCTL   | Receiver interrupt control register                                                                                                                |
| 0x01D0 0080               | 0x01D0 4080               | RSTAT     | Receiver status register                                                                                                                           |
| 0x01D0 0084               | 0x01D0 4084               | RSLOT     | Current receive TDM time slot register                                                                                                             |
| 0x01D0 0088               | 0x01D0 4088               | RCLKCHK   | Receive clock check control register                                                                                                               |
| 0x01D0 008C               | 0x01D0 408C               | REVTCTL   | Receiver DMA event control register                                                                                                                |
| 0x01D0 00A0               | 0x01D0 40A0               | XGBLCTL   | Transmitter global control register. Alias of GBLCTL, only transmit bits are affected - allows transmitter to be reset independently from receiver |
| 0x01D0 00A4               | 0x01D0 40A4               | XMASK     | Transmit format unit bit mask register                                                                                                             |
| 0x01D0 00A8               | 0x01D0 40A8               | XFMT      | Transmit bit stream format register                                                                                                                |
| 0x01D0 00AC               | 0x01D0 40AC               | AFSXCTL   | Transmit frame sync control register                                                                                                               |
| 0x01D0 00B0               | 0x01D0 40B0               | ACLKXCTL  | Transmit clock control register                                                                                                                    |
| 0x01D0 00B4               | 0x01D0 40B4               | AHCLKXCTL | Transmit high-frequency clock control register                                                                                                     |
| 0x01D0 00B8               | 0x01D0 40B8               | XTDM      | Transmit TDM time slot 0-31 register                                                                                                               |
| 0x01D0 00BC               | 0x01D0 40BC               | XINTCTL   | Transmitter interrupt control register                                                                                                             |
| 0x01D0 00C0               | 0x01D0 40C0               | XSTAT     | Transmitter status register                                                                                                                        |
| 0x01D0 00C4               | 0x01D0 40C4               | XSLOT     | Current transmit TDM time slot register                                                                                                            |
| 0x01D0 00C8               | 0x01D0 40C8               | XCLKCHK   | Transmit clock check control register                                                                                                              |
| 0x01D0 00CC               | 0x01D0 40CC               | XEVTCTL   | Transmitter DMA event control register                                                                                                             |
| 0x01D0 0100               | 0x01D0 4100               | DITCSRA0  | Left (even TDM time slot) channel status register (DIT mode) 0                                                                                     |

**Table 6-41. McASP Registers Accessed Through Peripheral Configuration Port (continued)**

| McASP0<br>BYTE<br>ADDRESS | McASP1<br>BYTE<br>ADDRESS | ACRONYM              | REGISTER DESCRIPTION                                              |
|---------------------------|---------------------------|----------------------|-------------------------------------------------------------------|
| 0x01D0 0104               | 0x01D0 4104               | DITCSRA1             | Left (even TDM time slot) channel status register (DIT mode) 1    |
| 0x01D0 0108               | 0x01D0 4108               | DITCSRA2             | Left (even TDM time slot) channel status register (DIT mode) 2    |
| 0x01D0 010C               | 0x01D0 410C               | DITCSRA3             | Left (even TDM time slot) channel status register (DIT mode) 3    |
| 0x01D0 0110               | 0x01D0 4110               | DITCSRA4             | Left (even TDM time slot) channel status register (DIT mode) 4    |
| 0x01D0 0114               | 0x01D0 4114               | DITCSRA5             | Left (even TDM time slot) channel status register (DIT mode) 5    |
| 0x01D0 0118               | 0x01D0 4118               | DITCSRB0             | Right (odd TDM time slot) channel status register (DIT mode) 0    |
| 0x01D0 011C               | 0x01D0 411C               | DITCSRB1             | Right (odd TDM time slot) channel status register (DIT mode) 1    |
| 0x01D0 0120               | 0x01D0 4120               | DITCSRB2             | Right (odd TDM time slot) channel status register (DIT mode) 2    |
| 0x01D0 0124               | 0x01D0 4124               | DITCSRB3             | Right (odd TDM time slot) channel status register (DIT mode) 3    |
| 0x01D0 0128               | 0x01D0 4128               | DITCSRB4             | Right (odd TDM time slot) channel status register (DIT mode) 4    |
| 0x01D0 012C               | 0x01D0 412C               | DITCSRB5             | Right (odd TDM time slot) channel status register (DIT mode) 5    |
| 0x01D0 0130               | 0x01D0 4130               | DITUDRA0             | Left (even TDM time slot) channel user data register (DIT mode) 0 |
| 0x01D0 0134               | 0x01D0 4134               | DITUDRA1             | Left (even TDM time slot) channel user data register (DIT mode) 1 |
| 0x01D0 0138               | 0x01D0 4138               | DITUDRA2             | Left (even TDM time slot) channel user data register (DIT mode) 2 |
| 0x01D0 013C               | 0x01D0 413C               | DITUDRA3             | Left (even TDM time slot) channel user data register (DIT mode) 3 |
| 0x01D0 0140               | 0x01D0 4140               | DITUDRA4             | Left (even TDM time slot) channel user data register (DIT mode) 4 |
| 0x01D0 0144               | 0x01D0 4144               | DITUDRA5             | Left (even TDM time slot) channel user data register (DIT mode) 5 |
| 0x01D0 0148               | 0x01D0 4148               | DITUDRB0             | Right (odd TDM time slot) channel user data register (DIT mode) 0 |
| 0x01D0 014C               | 0x01D0 414C               | DITUDRB1             | Right (odd TDM time slot) channel user data register (DIT mode) 1 |
| 0x01D0 0150               | 0x01D0 4150               | DITUDRB2             | Right (odd TDM time slot) channel user data register (DIT mode) 2 |
| 0x01D0 0154               | 0x01D0 4154               | DITUDRB3             | Right (odd TDM time slot) channel user data register (DIT mode) 3 |
| 0x01D0 0158               | 0x01D0 4158               | DITUDRB4             | Right (odd TDM time slot) channel user data register (DIT mode) 4 |
| 0x01D0 015C               | 0x01D0 415C               | DITUDRB5             | Right (odd TDM time slot) channel user data register (DIT mode) 5 |
| 0x01D0 0180               | 0x01D0 4180               | SRCTL0               | Serializer control register 0                                     |
| 0x01D0 0184               | 0x01D0 4184               | SRCTL1               | Serializer control register 1                                     |
| 0x01D0 0188               | 0x01D0 4188               | SRCTL2               | Serializer control register 2                                     |
| 0x01D0 018C               | 0x01D0 418C               | SRCTL3               | Serializer control register 3                                     |
| 0x01D0 0190               | 0x01D0 4190               | SRCTL4               | Serializer control register 4                                     |
| 0x01D0 0194               | 0x01D0 4194               | SRCTL5               | Serializer control register 5                                     |
| 0x01D0 0198               | 0x01D0 4198               | SRCTL6               | Serializer control register 6                                     |
| 0x01D0 019C               | 0x01D0 419C               | SRCTL7               | Serializer control register 7                                     |
| 0x01D0 01A0               | 0x01D0 41A0               | SRCTL8               | Serializer control register 8                                     |
| 0x01D0 01A4               | 0x01D0 41A4               | SRCTL9               | Serializer control register 9                                     |
| 0x01D0 01A8               | 0x01D0 41A8               | SRCTL10              | Serializer control register 10                                    |
| 0x01D0 01AC               | 0x01D0 41AC               | SRCTL11              | Serializer control register 11                                    |
| 0x01D0 01B0               | 0x01D0 41B0               | SRCTL12              | Serializer control register 12                                    |
| 0x01D0 01B4               | 0x01D0 41B4               | SRCTL13              | Serializer control register 13                                    |
| 0x01D0 01B8               | 0x01D0 41B8               | SRCTL14              | Serializer control register 14                                    |
| 0x01D0 01BC               | 0x01D0 41BC               | SRCTL15              | Serializer control register 15                                    |
| 0x01D0 0200               | 0x01D0 4200               | XBUF0 <sup>(1)</sup> | Transmit buffer register for serializer 0                         |
| 0x01D0 0204               | 0x01D0 4204               | XBUF1 <sup>(1)</sup> | Transmit buffer register for serializer 1                         |
| 0x01D0 0208               | 0x01D0 4208               | XBUF2 <sup>(1)</sup> | Transmit buffer register for serializer 2                         |
| 0x01D0 020C               | 0x01D0 420C               | XBUF3 <sup>(1)</sup> | Transmit buffer register for serializer 3                         |
| 0x01D0 0210               | 0x01D0 4210               | XBUF4 <sup>(1)</sup> | Transmit buffer register for serializer 4                         |
| 0x01D0 0214               | 0x01D0 4214               | XBUF5 <sup>(1)</sup> | Transmit buffer register for serializer 5                         |

(1) Writes to XBUF originate from peripheral configuration port only when XBUSEL = 1 in XFMT.

**Table 6-41. McASP Registers Accessed Through Peripheral Configuration Port (continued)**

| McASP0<br>BYTE<br>ADDRESS | McASP1<br>BYTE<br>ADDRESS | ACRONYM               | REGISTER DESCRIPTION                       |
|---------------------------|---------------------------|-----------------------|--------------------------------------------|
| 0x01D0 0218               | 0x01D0 4218               | XBUF6 <sup>(1)</sup>  | Transmit buffer register for serializer 6  |
| 0x01D0 021C               | 0x01D0 421C               | XBUF7                 | Transmit buffer register for serializer 7  |
| 0x01D0 0220               | 0x01D0 4220               | XBUF8 <sup>(1)</sup>  | Transmit buffer register for serializer 8  |
| 0x01D0 0224               | 0x01D0 4224               | XBUF9                 | Transmit buffer register for serializer 9  |
| 0x01D0 0228               | 0x01D0 4228               | XBUF10 <sup>(1)</sup> | Transmit buffer register for serializer 10 |
| 0x01D0 022C               | 0x01D0 422C               | XBUF11 <sup>(1)</sup> | Transmit buffer register for serializer 11 |
| 0x01D0 0230               | 0x01D0 4230               | XBUF12                | Transmit buffer register for serializer 12 |
| 0x01D0 0234               | 0x01D0 4234               | XBUF13 <sup>(1)</sup> | Transmit buffer register for serializer 13 |
| 0x01D0 0238               | 0x01D0 4238               | XBUF14 <sup>(1)</sup> | Transmit buffer register for serializer 14 |
| 0x01D0 023C               | 0x01D0 423C               | XBUF15                | Transmit buffer register for serializer 15 |
| 0x01D0 0280               | 0x01D0 4280               | RBUF0 <sup>(2)</sup>  | Receive buffer register for serializer 0   |
| 0x01D0 0284               | 0x01D0 4284               | RBUF1 <sup>(2)</sup>  | Receive buffer register for serializer 1   |
| 0x01D0 0288               | 0x01D0 4288               | RBUF2 <sup>(2)</sup>  | Receive buffer register for serializer 2   |
| 0x01D0 028C               | 0x01D0 428C               | RBUF3 <sup>(2)</sup>  | Receive buffer register for serializer 3   |
| 0x01D0 0290               | 0x01D0 4290               | RBUF4 <sup>(2)</sup>  | Receive buffer register for serializer 4   |
| 0x01D0 0294               | 0x01D0 4294               | RBUF5 <sup>(2)</sup>  | Receive buffer register for serializer 5   |
| 0x01D0 0298               | 0x01D0 4298               | RBUF6 <sup>(2)</sup>  | Receive buffer register for serializer 6   |
| 0x01D0 029C               | 0x01D0 429C               | RBUF7 <sup>(2)</sup>  | Receive buffer register for serializer 7   |
| 0x01D0 02A0               | 0x01D0 42A0               | RBUF8 <sup>(2)</sup>  | Receive buffer register for serializer 8   |
| 0x01D0 02A4               | 0x01D0 42A4               | RBUF9 <sup>(2)</sup>  | Receive buffer register for serializer 9   |
| 0x01D0 02A8               | 0x01D0 42A8               | RBUF10 <sup>(2)</sup> | Receive buffer register for serializer 10  |
| 0x01D0 02AC               | 0x01D0 42AC               | RBUF11 <sup>(2)</sup> | Receive buffer register for serializer 11  |
| 0x01D0 02B0               | 0x01D0 42B0               | RBUF12 <sup>(2)</sup> | Receive buffer register for serializer 12  |
| 0x01D0 02B4               | 0x01D0 42B4               | RBUF13 <sup>(2)</sup> | Receive buffer register for serializer 13  |
| 0x01D0 02B8               | 0x01D0 42B8               | RBUF14 <sup>(2)</sup> | Receive buffer register for serializer 14  |
| 0x01D0 02BC               | 0x01D0 42BC               | RBUF15 <sup>(2)</sup> | Receive buffer register for serializer 15  |

(2) Reads from XRBUF originate on peripheral configuration port only when RBUSEL = 1 in RFMT.

**Table 6-42. McASP Registers Accessed Through DMA Port**

|                   | McASP0<br>BYTE<br>ADDRESS | McASP1<br>BYTE<br>ADDRESS | ACRONYM | REGISTER DESCRIPTION                                                                                                                                                                                                                                  |
|-------------------|---------------------------|---------------------------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Read<br>Accesses  | 01D0 2000                 | 01D0 6000                 | RBUF    | Receive buffer DMA port address. Cycles through receive serializers, skipping over transmit serializers and inactive serializers. Starts at the lowest serializer at the beginning of each time slot. Reads from DMA port only if RBUSEL = 0 in RFMT. |
| Write<br>Accesses | 01D0 2000                 | 01D0 6000                 | XBUF    | Transmit buffer DMA port address. Cycles through transmit serializers, skipping over receive and inactive serializers. Starts at the lowest serializer at the beginning of each time slot. Writes to DMA port only if XBUSEL = 0 in XFMT.             |

**Table 6-43. McASP AFIFO Registers Accessed Through Peripheral Configuration Port**

| McASP0<br>BYTE ADDRESS | McASP1<br>BYTE ADDRESS | ACRONYM  | REGISTER DESCRIPTION                   |
|------------------------|------------------------|----------|----------------------------------------|
| 0x01D0 1000            | 0x01D0 5000            | AFIFOREV | AFIFO revision identification register |
| 0x01D0 1010            | 0x01D0 5010            | WFIFOCTL | Write FIFO control register            |
| 0x01D0 1014            | 0x01D0 5014            | WFIFOSTS | Write FIFO status register             |
| 0x01D0 1018            | 0x01D0 5018            | RFIFOCTL | Read FIFO control register             |
| 0x01D0 101C            | 0x01D0 501C            | RFIFOSTS | Read FIFO status register              |

## 6.16.2 McASP Electrical Data/Timing

### 6.16.2.1 Multichannel Audio Serial Port 0 (McASP0) Timing

Table 6-44 and Table 6-45 assume testing over recommended operating conditions (see Figure 6-29 and Figure 6-30).

**Table 6-44. McASP0 Timing Requirements<sup>(1) (2)</sup>**

| No. | PARAMETER              |                                                                      | MIN                 | MAX | UNIT |
|-----|------------------------|----------------------------------------------------------------------|---------------------|-----|------|
| 1   | $t_{c(AHCLKRX)}$       | Cycle time, AHCLKR0 external, AHCLKR0 input                          | 25                  |     | ns   |
|     |                        | Cycle time, AHCLKX0 external, AHCLKX0 input                          | 25                  |     |      |
| 2   | $t_{w(AHCLKRX)}$       | Pulse duration, AHCLKR0 external, AHCLKR0 input                      | 12.5                |     | ns   |
|     |                        | Pulse duration, AHCLKX0 external, AHCLKX0 input                      | 12.5                |     |      |
| 3   | $t_{c(ACLKRX)}$        | Cycle time, ACLKR0 external, ACLKR0 input                            | greater of 2P or 25 |     | ns   |
|     |                        | Cycle time, ACLKX0 external, ACLKX0 input                            | greater of 2P or 25 |     |      |
| 4   | $t_{w(ACLKRX)}$        | Pulse duration, ACLKR0 external, ACLKR0 input                        | 12.5                |     | ns   |
|     |                        | Pulse duration, ACLKX0 external, ACLKX0 input                        | 12.5                |     |      |
| 5   | $t_{su(AFSRX-ACLKRX)}$ | Setup time, AFSR0 input to ACLKR0 internal <sup>(3)</sup>            | 9.4                 |     | ns   |
|     |                        | Setup time, AFSX0 input to ACLKX0 internal                           | 9.4                 |     |      |
|     |                        | Setup time, AFSR0 input to ACLKR0 external input <sup>(3)</sup>      | 2.9                 |     |      |
|     |                        | Setup time, AFSX0 input to ACLKX0 external input                     | 2.9                 |     |      |
|     |                        | Setup time, AFSR0 input to ACLKR0 external output <sup>(3)</sup>     | 2.9                 |     |      |
|     |                        | Setup time, AFSX0 input to ACLKX0 external output                    | 2.9                 |     |      |
| 6   | $t_{h(ACLKRX-AFSRX)}$  | Hold time, AFSR0 input after ACLKR0 internal <sup>(3)</sup>          | -1.2                |     | ns   |
|     |                        | Hold time, AFSX0 input after ACLKX0 internal                         | -1.2                |     |      |
|     |                        | Hold time, AFSR0 input after ACLKR0 external input <sup>(3)</sup>    | 0.9                 |     |      |
|     |                        | Hold time, AFSX0 input after ACLKX0 external input                   | 0.9                 |     |      |
|     |                        | Hold time, AFSR0 input after ACLKR0 external output <sup>(3)</sup>   | 0.9                 |     |      |
|     |                        | Hold time, AFSX0 input after ACLKX0 external output                  | 0.9                 |     |      |
| 7   | $t_{su(AXR0-ACLKRX)}$  | Setup time, AXR0[n] input to ACLKR0 internal <sup>(3)</sup>          | 9.4                 |     | ns   |
|     |                        | Setup time, AXR0[n] input to ACLKX0 internal <sup>(4)</sup>          | 9.4                 |     |      |
|     |                        | Setup time, AXR0[n] input to ACLKR0 external input <sup>(3)</sup>    | 2.9                 |     |      |
|     |                        | Setup time, AXR0[n] input to ACLKX0 external input <sup>(4)</sup>    | 2.9                 |     |      |
|     |                        | Setup time, AXR0[n] input to ACLKR0 external output <sup>(3)</sup>   | 2.9                 |     |      |
|     |                        | Setup time, AXR0[n] input to ACLKX0 external output <sup>(4)</sup>   | 2.9                 |     |      |
| 8   | $t_{h(ACLKRX-AXR)}$    | Hold time, AXR0[n] input after ACLKR0 internal <sup>(3)</sup>        | -1.3                |     | ns   |
|     |                        | Hold time, AXR0[n] input after ACLKX0 internal <sup>(4)</sup>        | -1.3                |     |      |
|     |                        | Hold time, AXR0[n] input after ACLKR0 external input <sup>(3)</sup>  | 0.5                 |     |      |
|     |                        | Hold time, AXR0[n] input after ACLKX0 external input <sup>(4)</sup>  | 0.5                 |     |      |
|     |                        | Hold time, AXR0[n] input after ACLKR0 external output <sup>(3)</sup> | 0.5                 |     |      |
|     |                        | Hold time, AXR0[n] input after ACLKX0 external output <sup>(4)</sup> | 0.5                 |     |      |

- (1) ACLKX0 internal – McASP0 ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1  
 ACLKX0 external input – McASP0 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0  
 ACLKX0 external output – McASP0 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1  
 ACLKR0 internal – McASP0 ACLKRCTL.CLKRM = 1, PDIR.ACLKR = 1  
 ACLKR0 external input – McASP0 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 0  
 ACLKR0 external output – McASP0 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 1
- (2) P = SYSCLK2 period
- (3) McASP0 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR0
- (4) McASP0 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX0

**Table 6-45. McASP0 Switching Characteristics<sup>(1)</sup>**

| No. | PARAMETER                            |                                                                | MIN                                   | MAX  | UNIT |
|-----|--------------------------------------|----------------------------------------------------------------|---------------------------------------|------|------|
| 9   | $t_c(\text{AHCLKRX})$                | Cycle time, AHCLKR0 internal, AHCLKR0 output                   | 25                                    |      | ns   |
|     |                                      | Cycle time, AHCLKR0 external, AHCLKR0 output                   | 25                                    |      |      |
|     |                                      | Cycle time, AHCLKX0 internal, AHCLKX0 output                   | 25                                    |      |      |
|     |                                      | Cycle time, AHCLKX0 external, AHCLKX0 output                   | 25                                    |      |      |
| 10  | $t_w(\text{AHCLKRX})$                | Pulse duration, AHCLKR0 internal, AHCLKR0 output               | $(\text{AHR}/2) - 2.5^{(2)}$          |      | ns   |
|     |                                      | Pulse duration, AHCLKR0 external, AHCLKR0 output               | $(\text{AHR}/2) - 2.5^{(2)}$          |      |      |
|     |                                      | Pulse duration, AHCLKX0 internal, AHCLKX0 output               | $(\text{AHX}/2) - 2.5^{(3)}$          |      |      |
|     |                                      | Pulse duration, AHCLKX0 external, AHCLKX0 output               | $(\text{AHX}/2) - 2.5^{(3)}$          |      |      |
| 11  | $t_c(\text{ACLKRX})$                 | Cycle time, ACLKR0 internal, ACLKR0 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      | ns   |
|     |                                      | Cycle time, ACLKR0 external, ACLKR0 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
|     |                                      | Cycle time, ACLKX0 internal, ACLKX0 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
|     |                                      | Cycle time, ACLKX0 external, ACLKX0 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
| 12  | $t_w(\text{ACLKRX})$                 | Pulse duration, ACLKR0 internal, ACLKR0 output                 | $(\text{AR}/2) - 2.5^{(5)}$           |      | ns   |
|     |                                      | Pulse duration, ACLKR0 external, ACLKR0 output                 | $(\text{AR}/2) - 2.5^{(5)}$           |      |      |
|     |                                      | Pulse duration, ACLKX0 internal, ACLKX0 output                 | $(\text{AX}/2) - 2.5^{(6)}$           |      |      |
|     |                                      | Pulse duration, ACLKX0 external, ACLKX0 output                 | $(\text{AX}/2) - 2.5^{(6)}$           |      |      |
| 13  | $t_d(\text{ACLKRX-AFSRX})$           | Delay time, ACLKR0 internal, AFSR output <sup>(7)</sup>        | 0                                     | 5.8  | ns   |
|     |                                      | Delay time, ACLKX0 internal, AFSX output                       | 0                                     | 5.8  |      |
|     |                                      | Delay time, ACLKR0 external input, AFSR output <sup>(7)</sup>  | 2.5                                   | 11.6 |      |
|     |                                      | Delay time, ACLKX0 external input, AFSX output                 | 2.5                                   | 11.6 |      |
|     |                                      | Delay time, ACLKR0 external output, AFSR output <sup>(7)</sup> | 2.5                                   | 11.6 |      |
|     |                                      | Delay time, ACLKX0 external output, AFSX output                | 2.5                                   | 11.6 |      |
| 14  | $t_d(\text{ACLKX-AXRV})$             | Delay time, ACLKX0 internal, AXR0[n] output                    | 0                                     | 5.8  | ns   |
|     |                                      | Delay time, ACLKX0 external input, AXR0[n] output              | 2.5                                   | 11.6 |      |
|     |                                      | Delay time, ACLKX0 external output, AXR0[n] output             | 2.5                                   | 11.6 |      |
| 15  | $t_{\text{dis}}(\text{ACLKX-AXRHZ})$ | Disable time, ACLKX0 internal, AXR0[n] output                  | 0                                     | 5.8  | ns   |
|     |                                      | Disable time, ACLKX0 external input, AXR0[n] output            | 3                                     | 11.6 |      |
|     |                                      | Disable time, ACLKX0 external output, AXR0[n] output           | 3                                     | 11.6 |      |

- (1) McASP0 ACLKX0 internal – ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1  
 ACLKX0 external input – McASP0 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0  
 ACLKX0 external output – McASP0 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1  
 ACLKR0 internal – McASP0 ACLKR0CTL.CLKRM = 1, PDIR.ACLKR = 1  
 ACLKR0 external input – McASP0 ACLKR0CTL.CLKRM = 0, PDIR.ACLKR = 0  
 ACLKR0 external output – McASP0 ACLKR0CTL.CLKRM = 0, PDIR.ACLKR = 1

(2) AHR - Cycle time, AHCLKR0.

(3) AHX - Cycle time, AHCLKX0.

(4) P = SYSCLK2 period

(5) AR - ACLKR0 period.

(6) AX - ACLKX0 period.

(7) McASP0 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR0

### 6.16.2.2 Multichannel Audio Serial Port 1 (McASP1) Timing

Table 6-46 and Table 6-47 assume testing over recommended operating conditions (see Figure 6-29 and Figure 6-30).

**Table 6-46. McASP1 Timing Requirements<sup>(1) (2)</sup>**

| No. | PARAMETER              |                                                                      | MIN                 | MAX | UNIT |
|-----|------------------------|----------------------------------------------------------------------|---------------------|-----|------|
| 1   | $t_{c(AHCLKRX)}$       | Cycle time, AHCLKR1 external, AHCLKR1 input                          | 25                  |     | ns   |
|     |                        | Cycle time, AHCLKX1 external, AHCLKX1 input                          | 25                  |     |      |
| 2   | $t_{w(AHCLKRX)}$       | Pulse duration, AHCLKR1 external, AHCLKR1 input                      | 12.5                |     | ns   |
|     |                        | Pulse duration, AHCLKX1 external, AHCLKX1 input                      | 12.5                |     |      |
| 3   | $t_{c(ACLKRX)}$        | Cycle time, ACLKR1 external, ACLKR1 input                            | greater of 2P or 25 |     | ns   |
|     |                        | Cycle time, ACLKX1 external, ACLKX1 input                            | greater of 2P or 25 |     |      |
| 4   | $t_{w(ACLKRX)}$        | Pulse duration, ACLKR1 external, ACLKR1 input                        | 12.5                |     | ns   |
|     |                        | Pulse duration, ACLKX1 external, ACLKX1 input                        | 12.5                |     |      |
| 5   | $t_{su(AFSRX-ACLKRX)}$ | Setup time, AFSR1 input to ACLKR1 internal <sup>(3)</sup>            | 10.4                |     | ns   |
|     |                        | Setup time, AFSX1 input to ACLKX1 internal                           | 10.4                |     |      |
|     |                        | Setup time, AFSR1 input to ACLKR1 external input <sup>(3)</sup>      | 2.6                 |     |      |
|     |                        | Setup time, AFSX1 input to ACLKX1 external input                     | 2.6                 |     |      |
|     |                        | Setup time, AFSR1 input to ACLKR1 external output <sup>(3)</sup>     | 2.6                 |     |      |
|     |                        | Setup time, AFSX1 input to ACLKX1 external output                    | 2.6                 |     |      |
| 6   | $t_{h(ACLKRX-AFSRX)}$  | Hold time, AFSR1 input after ACLKR1 internal <sup>(3)</sup>          | -1.9                |     | ns   |
|     |                        | Hold time, AFSX1 input after ACLKX1 internal                         | -1.9                |     |      |
|     |                        | Hold time, AFSR1 input after ACLKR1 external input <sup>(3)</sup>    | 0.7                 |     |      |
|     |                        | Hold time, AFSX1 input after ACLKX1 external input                   | 0.7                 |     |      |
|     |                        | Hold time, AFSR1 input after ACLKR1 external output <sup>(3)</sup>   | 0.7                 |     |      |
|     |                        | Hold time, AFSX1 input after ACLKX1 external output                  | 0.7                 |     |      |
| 7   | $t_{su(AXR-ACLKRX)}$   | Setup time, AXR1[n] input to ACLKR1 internal <sup>(3)</sup>          | 10.4                |     | ns   |
|     |                        | Setup time, AXR1[n] input to ACLKX1 internal <sup>(4)</sup>          | 10.4                |     |      |
|     |                        | Setup time, AXR1[n] input to ACLKR1 external input <sup>(3)</sup>    | 2.6                 |     |      |
|     |                        | Setup time, AXR1[n] input to ACLKX1 external input <sup>(4)</sup>    | 2.6                 |     |      |
|     |                        | Setup time, AXR1[n] input to ACLKR1 external output <sup>(3)</sup>   | 2.6                 |     |      |
|     |                        | Setup time, AXR1[n] input to ACLKX1 external output <sup>(4)</sup>   | 2.6                 |     |      |
| 8   | $t_{h(ACLKRX-AXR)}$    | Hold time, AXR1[n] input after ACLKR1 internal <sup>(3)</sup>        | -1.8                |     | ns   |
|     |                        | Hold time, AXR1[n] input after ACLKX1 internal <sup>(4)</sup>        | -1.8                |     |      |
|     |                        | Hold time, AXR1[n] input after ACLKR1 external input <sup>(3)</sup>  | 0.5                 |     |      |
|     |                        | Hold time, AXR1[n] input after ACLKX1 external input <sup>(4)</sup>  | 0.5                 |     |      |
|     |                        | Hold time, AXR1[n] input after ACLKR1 external output <sup>(3)</sup> | 0.5                 |     |      |
|     |                        | Hold time, AXR1[n] input after ACLKX1 external output <sup>(4)</sup> | 0.5                 |     |      |

- (1) ACLKX1 internal – McASP1 ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1  
 ACLKX1 external input – McASP1 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0  
 ACLKX1 external output – McASP1 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1  
 ACLKR1 internal – McASP1 ACLKRCTL.CLKRM = 1, PDIR.ACLKR = 1  
 ACLKR1 external input – McASP1 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 0  
 ACLKR1 external output – McASP1 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 1
- (2) P = SYSCLK2 period
- (3) McASP1 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR1
- (4) McASP1 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX1

**Table 6-47. McASP1 Switching Characteristics<sup>(1)</sup>**

| No. | PARAMETER                            |                                                                | MIN                                   | MAX  | UNIT |
|-----|--------------------------------------|----------------------------------------------------------------|---------------------------------------|------|------|
| 9   | $t_c(\text{AHCLKRX})$                | Cycle time, AHCLKR1 internal, AHCLKR1 output                   | 25                                    |      | ns   |
|     |                                      | Cycle time, AHCLKR1 external, AHCLKR1 output                   | 25                                    |      |      |
|     |                                      | Cycle time, AHCLKX1 internal, AHCLKX1 output                   | 25                                    |      |      |
|     |                                      | Cycle time, AHCLKX1 external, AHCLKX1 output                   | 25                                    |      |      |
| 10  | $t_w(\text{AHCLKRX})$                | Pulse duration, AHCLKR1 internal, AHCLKR1 output               | $(\text{AHR}/2) - 2.5^{(2)}$          |      | ns   |
|     |                                      | Pulse duration, AHCLKR1 external, AHCLKR1 output               | $(\text{AHR}/2) - 2.5^{(2)}$          |      |      |
|     |                                      | Pulse duration, AHCLKX1 internal, AHCLKX1 output               | $(\text{AHX}/2) - 2.5^{(3)}$          |      |      |
|     |                                      | Pulse duration, AHCLKX1 external, AHCLKX1 output               | $(\text{AHX}/2) - 2.5^{(3)}$          |      |      |
| 11  | $t_c(\text{ACLKRX})$                 | Cycle time, ACLKR1 internal, ACLKR1 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      | ns   |
|     |                                      | Cycle time, ACLKR1 external, ACLKR1 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
|     |                                      | Cycle time, ACLKX1 internal, ACLKX1 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
|     |                                      | Cycle time, ACLKX1 external, ACLKX1 output                     | greater of 2P or 25 ns <sup>(4)</sup> |      |      |
| 12  | $t_w(\text{ACLKRX})$                 | Pulse duration, ACLKR1 internal, ACLKR1 output                 | $(\text{AR}/2) - 2.5^{(5)}$           |      | ns   |
|     |                                      | Pulse duration, ACLKR1 external, ACLKR1 output                 | $(\text{AR}/2) - 2.5^{(5)}$           |      |      |
|     |                                      | Pulse duration, ACLKX1 internal, ACLKX1 output                 | $(\text{AX}/2) - 2.5^{(6)}$           |      |      |
|     |                                      | Pulse duration, ACLKX1 external, ACLKX1 output                 | $(\text{AX}/2) - 2.5^{(6)}$           |      |      |
| 13  | $t_d(\text{ACLKRX-AFSRX})$           | Delay time, ACLKR1 internal, AFSR output <sup>(7)</sup>        | 0.5                                   | 6.7  | ns   |
|     |                                      | Delay time, ACLKX1 internal, AFSX output                       | 0.5                                   | 6.7  |      |
|     |                                      | Delay time, ACLKR1 external input, AFSR output <sup>(7)</sup>  | 3.4                                   | 13.8 |      |
|     |                                      | Delay time, ACLKX1 external input, AFSX output                 | 3.4                                   | 13.8 |      |
|     |                                      | Delay time, ACLKR1 external output, AFSR output <sup>(7)</sup> | 3.4                                   | 13.8 |      |
|     |                                      | Delay time, ACLKX1 external output, AFSX output                | 3.4                                   | 13.8 |      |
| 14  | $t_d(\text{ACLKX-AXRV})$             | Delay time, ACLKX1 internal, AXR1[n] output                    | 0.5                                   | 6.7  | ns   |
|     |                                      | Delay time, ACLKX1 external input, AXR1[n] output              | 3.4                                   | 13.8 |      |
|     |                                      | Delay time, ACLKX1 external output, AXR1[n] output             | 3.4                                   | 13.8 |      |
| 15  | $t_{\text{dis}}(\text{ACLKX-AXRHZ})$ | Disable time, ACLKX1 internal, AXR1[n] output                  | 0.5                                   | 6.7  | ns   |
|     |                                      | Disable time, ACLKX1 external input, AXR1[n] output            | 3.9                                   | 13.8 |      |
|     |                                      | Disable time, ACLKX1 external output, AXR1[n] output           | 3.9                                   | 13.8 |      |

- (1) McASP1 ACLKX1 internal – ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1  
 McASP1 ACLKX1 external input – ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0  
 McASP1 ACLKX1 external output – ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1  
 McASP1 ACLKR1 internal – ACLKR1CTL.CLKRM = 1, PDIR.ACLKR = 1  
 McASP1 ACLKR1 external input – ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 0  
 McASP1 ACLKR1 external output – ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 1

(2) AHR - Cycle time, AHCLKR1.

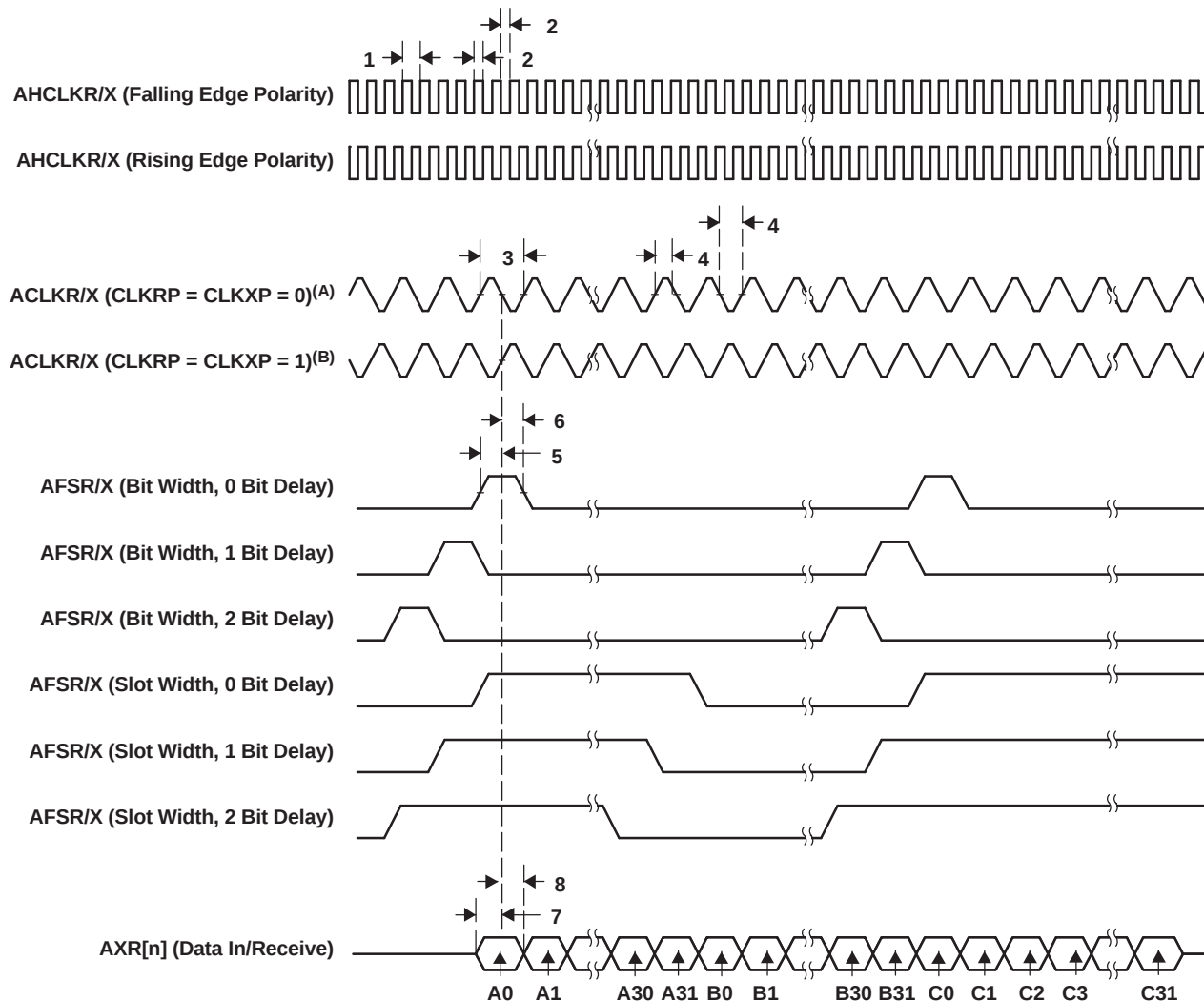
(3) AHX - Cycle time, AHCLKX1.

(4) P = SYSCLK2 period

(5) AR - ACLKR1 period.

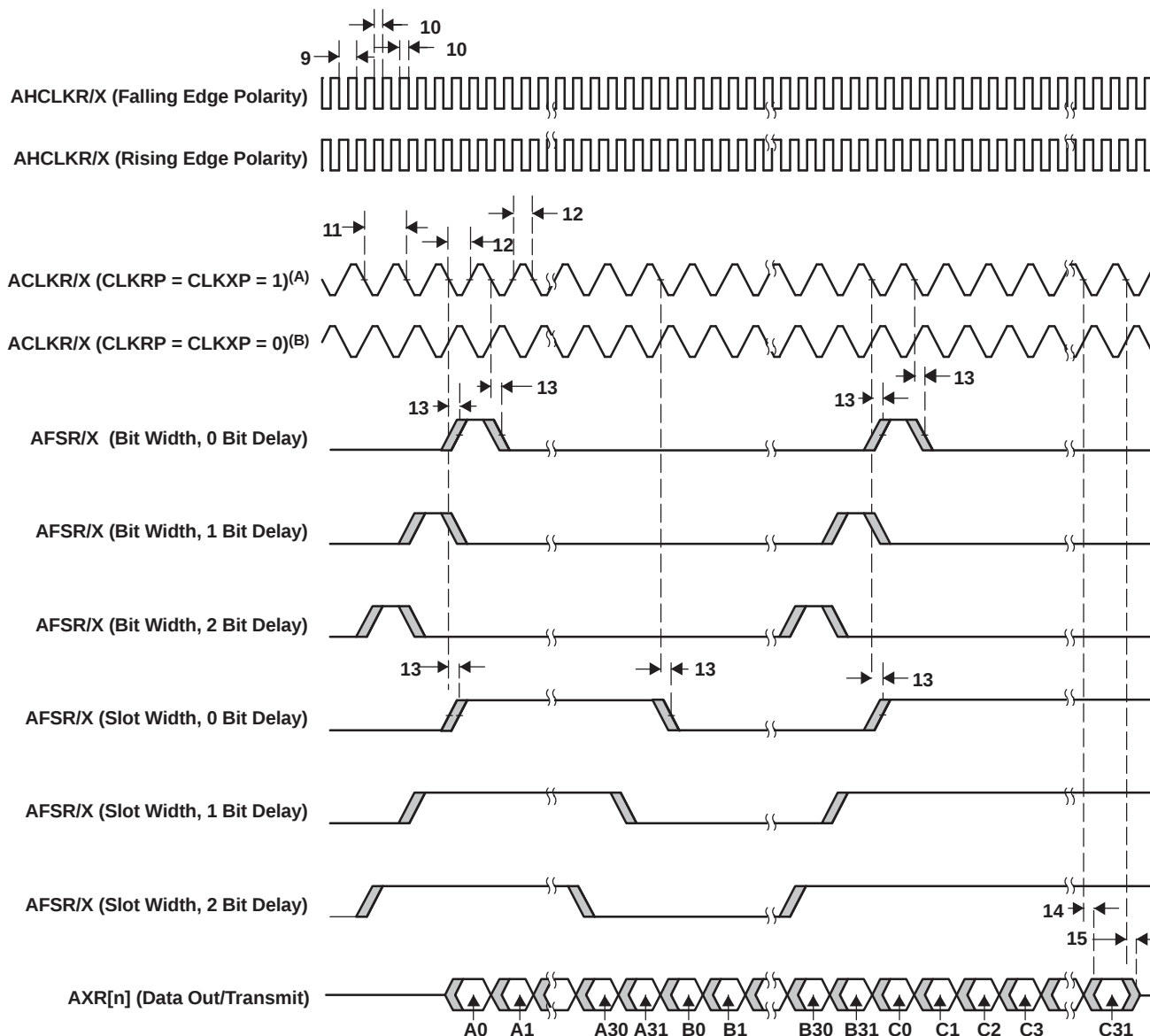
(6) AX - ACLKX1 period.

(7) McASP1 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR1



- 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 6-29. 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 6-30. McASP Output Timings**

## 6.17 Serial Peripheral Interface Ports (SPI0, SPI1)

Figure 6-31 is a block diagram of the SPI module, which is a simple shift register and buffer plus control logic. Data is written to the shift register before transmission occurs and is read from the buffer at the end of transmission. The SPI can operate either as a master, in which case, it initiates a transfer and drives the SPIx\_CLK pin, or as a slave. Four clock phase and polarity options are supported as well as many data formatting options.

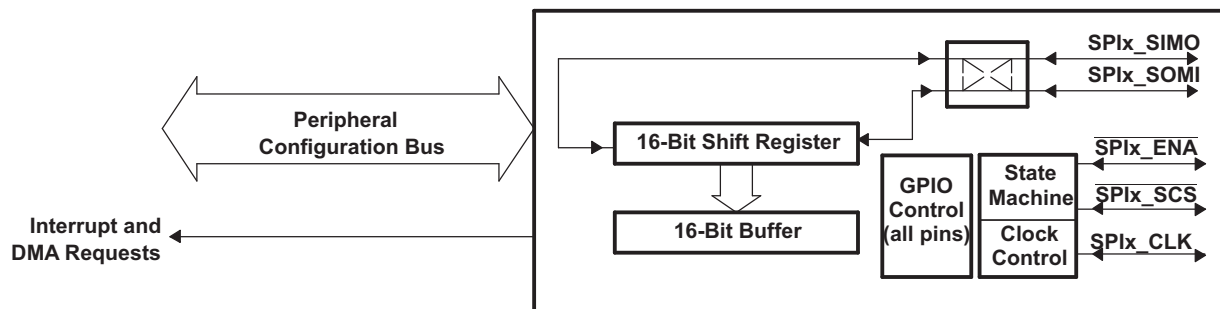


Figure 6-31. Block Diagram of SPI Module

The SPI supports 3-, 4-, and 5-pin operation with three basic pins (SPIx\_CLK, SPIx\_SIMO, and SPIx\_SOMI) and two optional pins (SPIx\_SCS, SPIx\_ENA).

The optional  $\overline{\text{SPIx\_SCS}}$  (Slave Chip Select) pin is most useful to enable in slave mode when there are other slave devices on the same SPI port. The device will only shift data and drive the SPIx\_SOMI pin when  $\overline{\text{SPIx\_SCS}}$  is held low.

In slave mode,  $\overline{\text{SPIx\_ENA}}$  is an optional output. The  $\overline{\text{SPIx\_ENA}}$  output provides the status of the internal transmit buffer (SPIDAT0/1 registers). In four-pin mode with the enable option,  $\overline{\text{SPIx\_ENA}}$  is asserted only when the transmit buffer is full, indicating that the slave is ready to begin another transfer. In five-pin mode, the  $\overline{\text{SPIx\_ENA}}$  is additionally qualified by  $\overline{\text{SPIx\_SCS}}$  being asserted. This allows a single handshake line to be shared by multiple slaves on the same SPI bus.

In master mode, the  $\overline{\text{SPIx\_ENA}}$  pin is an optional input and the master can be configured to delay the start of the next transfer until the slave asserts  $\overline{\text{SPIx\_ENA}}$ . The addition of this handshake signal simplifies SPI communications and, on average, increases SPI bus throughput since the master does not need to delay each transfer long enough to allow for the worst-case latency of the slave device. Instead, each transfer can begin as soon as both the master and slave have actually serviced the previous SPI transfer.

Although the SPI module supports two interrupt outputs, SPIx\_INT1 is the only interrupt connected on this device.

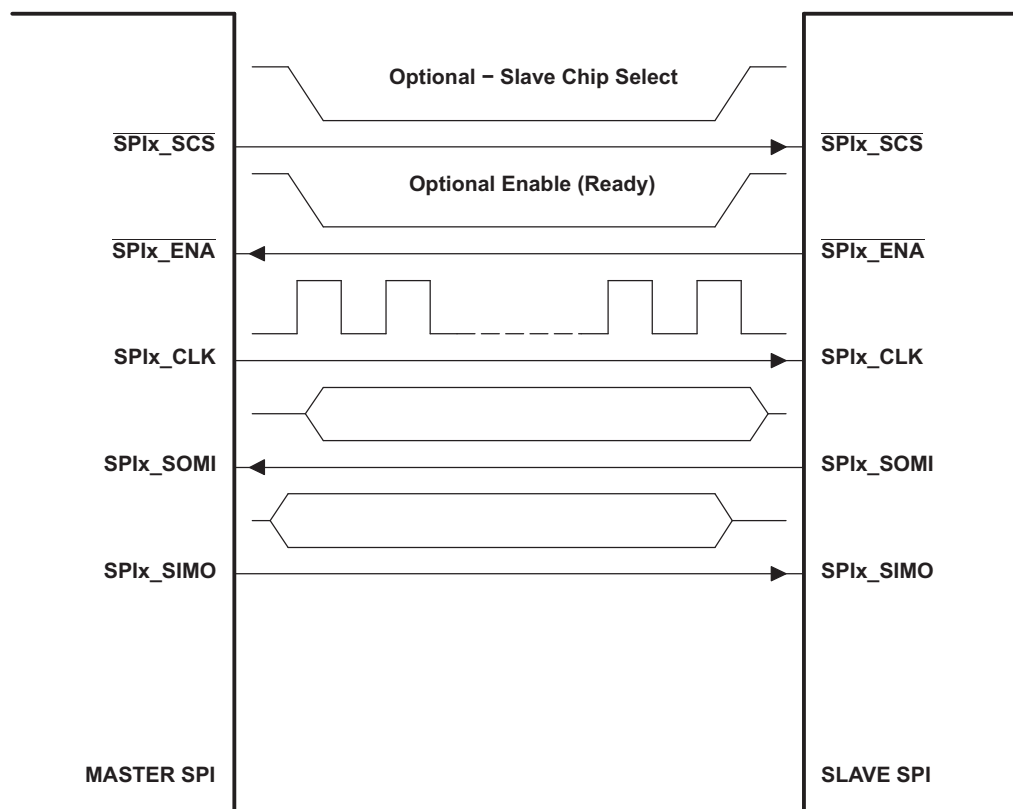


Figure 6-32. Illustration of SPI Master-to-SPI Slave Connection

### 6.17.1 SPI Peripheral Registers Description(s)

Table 6-48 is a list of the SPI registers.

**Table 6-48. SPIx Configuration Registers**

| SPI0<br>BYTE ADDRESS | SPI1<br>BYTE ADDRESS | ACRONYM  | REGISTER DESCRIPTION                     |
|----------------------|----------------------|----------|------------------------------------------|
| 0x01C4 1000          | 0x01E1 2000          | SPIGCR0  | Global Control Register 0                |
| 0x01C4 1004          | 0x01E1 2004          | SPIGCR1  | Global Control Register 1                |
| 0x01C4 1008          | 0x01E1 2008          | SPIINT0  | Interrupt Register                       |
| 0x01C4 100C          | 0x01E1 200C          | SPIVLV   | Interrupt Level Register                 |
| 0x01C4 1010          | 0x01E1 2010          | SPIFLG   | Flag Register                            |
| 0x01C4 1014          | 0x01E1 2014          | SPIPC0   | Pin Control Register 0 (Pin Function)    |
| 0x01C4 1018          | 0x01E1 2018          | SPIPC1   | Pin Control Register 1 (Pin Direction)   |
| 0x01C4 101C          | 0x01E1 201C          | SPIPC2   | Pin Control Register 2 (Pin Data In)     |
| 0x01C4 1020          | 0x01E1 2020          | SPIPC3   | Pin Control Register 3 (Pin Data Out)    |
| 0x01C4 1024          | 0x01E1 2024          | SPIPC4   | Pin Control Register 4 (Pin Data Set)    |
| 0x01C4 1028          | 0x01E1 2028          | SPIPC5   | Pin Control Register 5 (Pin Data Clear)  |
| 0x01C4 102C          | 0x01E1 202C          | Reserved | Reserved - Do not write to this register |
| 0x01C4 1030          | 0x01E1 2030          | Reserved | Reserved - Do not write to this register |
| 0x01C4 1034          | 0x01E1 2034          | Reserved | Reserved - Do not write to this register |
| 0x01C4 1038          | 0x01E1 2038          | SPIDAT0  | Shift Register 0 (without format select) |
| 0x01C4 103C          | 0x01E1 203C          | SPIDAT1  | Shift Register 1 (with format select)    |
| 0x01C4 1040          | 0x01E1 2040          | SPIBUF   | Buffer Register                          |
| 0x01C4 1044          | 0x01E1 2044          | SPIEMU   | Emulation Register                       |
| 0x01C4 1048          | 0x01E1 2048          | SPIDELAY | Delay Register                           |
| 0x01C4 104C          | 0x01E1 204C          | SPIDEF   | Default Chip Select Register             |
| 0x01C4 1050          | 0x01E1 2050          | SPIFMT0  | Format Register 0                        |
| 0x01C4 1054          | 0x01E1 2054          | SPIFMT1  | Format Register 1                        |
| 0x01C4 1058          | 0x01E1 2058          | SPIFMT2  | Format Register 2                        |
| 0x01C4 105C          | 0x01E1 205C          | SPIFMT3  | Format Register 3                        |
| 0x01C4 1060          | 0x01E1 2060          | Reserved | Reserved - Do not write to this register |
| 0x01C4 1064          | 0x01E1 2064          | INTVEC1  | Interrupt Vector for SPI INT1            |

## 6.17.2 SPI Electrical Data/Timing

### 6.17.2.1 Serial Peripheral Interface (SPI) Timing

Table 6-49 through Table 6-64 assume testing over recommended operating conditions (see Figure 6-33 through Figure 6-36).

**Table 6-49. General Timing Requirements for SPI0 Master Modes<sup>(1)</sup>**

| No. | PARAMETER            |                                                                                          | MIN                                            | MAX                   | UNIT |
|-----|----------------------|------------------------------------------------------------------------------------------|------------------------------------------------|-----------------------|------|
| 1   | $t_{c(SPC)M}$        | Cycle Time, SPI0_CLK, All Master Modes                                                   | greater of 3P or 20                            | 256P                  | ns   |
| 2   | $t_{w(SPCH)M}$       | Pulse Width High, SPI0_CLK, All Master Modes                                             | $0.5t_{c(SPC)M} - 1$                           |                       | ns   |
| 3   | $t_{w(SPCL)M}$       | Pulse Width Low, SPI0_CLK, All Master Modes                                              | $0.5t_{c(SPC)M} - 1$                           |                       | ns   |
| 4   | $t_{d(SIMO\_SPC)M}$  | Delay, initial data bit valid on SPI0_SIMO after initial edge on SPI0_CLK <sup>(2)</sup> | Polarity = 0, Phase = 0, to SPI0_CLK rising    | 5                     | ns   |
|     |                      |                                                                                          | Polarity = 0, Phase = 1, to SPI0_CLK rising    | $-0.5t_{c(SPC)M} + 5$ |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 0, to SPI0_CLK falling   | 5                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 1, to SPI0_CLK falling   | $-0.5t_{c(SPC)M} + 5$ |      |
| 5   | $t_{d(SPC\_SIMO)M}$  | Delay, subsequent bits valid on SPI0_SIMO after transmit edge of SPI0_CLK                | Polarity = 0, Phase = 0, from SPI0_CLK rising  | 5                     | ns   |
|     |                      |                                                                                          | Polarity = 0, Phase = 1, from SPI0_CLK falling | 5                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 0, from SPI0_CLK falling | 5                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 1, from SPI0_CLK rising  | 5                     |      |
| 6   | $t_{oh(SPC\_SIMO)M}$ | Output hold time, SPI0_SIMO valid after receive edge of SPI0_CLK                         | Polarity = 0, Phase = 0, from SPI0_CLK falling | $0.5t_{c(SPC)M} - 3$  | ns   |
|     |                      |                                                                                          | Polarity = 0, Phase = 1, from SPI0_CLK rising  | $0.5t_{c(SPC)M} - 3$  |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 0, from SPI0_CLK rising  | $0.5t_{c(SPC)M} - 3$  |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 1, from SPI0_CLK falling | $0.5t_{c(SPC)M} - 3$  |      |
| 7   | $t_{su(SOMI\_SPC)M}$ | Input Setup Time, SPI0_SOMI valid before receive edge of SPI0_CLK                        | Polarity = 0, Phase = 0, to SPI0_CLK falling   | 0                     | ns   |
|     |                      |                                                                                          | Polarity = 0, Phase = 1, to SPI0_CLK rising    | 0                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 0, to SPI0_CLK rising    | 0                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 1, to SPI0_CLK falling   | 0                     |      |
| 8   | $t_{ih(SPC\_SOMI)M}$ | Input Hold Time, SPI0_SOMI valid after receive edge of SPI0_CLK                          | Polarity = 0, Phase = 0, from SPI0_CLK falling | 5                     | ns   |
|     |                      |                                                                                          | Polarity = 0, Phase = 1, from SPI0_CLK rising  | 5                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 0, from SPI0_CLK rising  | 5                     |      |
|     |                      |                                                                                          | Polarity = 1, Phase = 1, from SPI0_CLK falling | 5                     |      |

(1) P = SYSCLK2 period

(2) First bit may be MSB or LSB depending upon SPI configuration. MO(0) refers to first bit and MO(n) refers to last bit output on SPI0\_SIMO. MI(0) refers to the first bit input and MI(n) refers to the last bit input on SPI0\_SOMI.

**Table 6-50. General Timing Requirements for SPI0 Slave Modes<sup>(1)</sup>**

| No. | PARAMETER            |                                                                                                    | MIN                                            | MAX                  | UNIT |
|-----|----------------------|----------------------------------------------------------------------------------------------------|------------------------------------------------|----------------------|------|
| 9   | $t_{c(SPC)}S$        | Cycle Time, SPI0_CLK, All Slave Modes                                                              | greater of 3P or 40                            |                      | ns   |
| 10  | $t_{w(SPCH)}S$       | Pulse Width High, SPI0_CLK, All Slave Modes                                                        | 18                                             |                      | ns   |
| 11  | $t_{w(SPCL)}S$       | Pulse Width Low, SPI0_CLK, All Slave Modes                                                         | 18                                             |                      | ns   |
| 12  | $t_{su(SOMI\_SPC)}S$ | Setup time, transmit data written to SPI before initial clock edge from master. <sup>(2) (3)</sup> | Polarity = 0, Phase = 0, to SPI0_CLK rising    | 2P                   | ns   |
|     |                      |                                                                                                    | Polarity = 0, Phase = 1, to SPI0_CLK rising    | 2P                   |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 0, to SPI0_CLK falling   | 2P                   |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 1, to SPI0_CLK falling   | 2P                   |      |
| 13  | $t_{d(SPC\_SOMI)}S$  | Delay, subsequent bits valid on SPI0_SOMI after transmit edge of SPI0_CLK                          | Polarity = 0, Phase = 0, from SPI0_CLK rising  | 18.5                 | ns   |
|     |                      |                                                                                                    | Polarity = 0, Phase = 1, from SPI0_CLK falling | 18.5                 |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 0, from SPI0_CLK falling | 18.5                 |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 1, from SPI0_CLK rising  | 18.5                 |      |
| 14  | $t_{oh(SPC\_SOMI)}S$ | Output hold time, SPI0_SOMI valid after receive edge of SPI0_CLK                                   | Polarity = 0, Phase = 0, from SPI0_CLK falling | $0.5t_{c(SPC)}S - 3$ | ns   |
|     |                      |                                                                                                    | Polarity = 0, Phase = 1, from SPI0_CLK rising  | $0.5t_{c(SPC)}S - 3$ |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 0, from SPI0_CLK rising  | $0.5t_{c(SPC)}S - 3$ |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 1, from SPI0_CLK falling | $0.5t_{c(SPC)}S - 3$ |      |
| 15  | $t_{su(SIMO\_SPC)}S$ | Input Setup Time, SPI0_SIMO valid before receive edge of SPI0_CLK                                  | Polarity = 0, Phase = 0, to SPI0_CLK falling   | 0                    | ns   |
|     |                      |                                                                                                    | Polarity = 0, Phase = 1, to SPI0_CLK rising    | 0                    |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 0, to SPI0_CLK rising    | 0                    |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 1, to SPI0_CLK falling   | 0                    |      |
| 16  | $t_{ih(SPC\_SIMO)}S$ | Input Hold Time, SPI0_SIMO valid after receive edge of SPI0_CLK                                    | Polarity = 0, Phase = 0, from SPI0_CLK falling | 5                    | ns   |
|     |                      |                                                                                                    | Polarity = 0, Phase = 1, from SPI0_CLK rising  | 5                    |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 0, from SPI0_CLK rising  | 5                    |      |
|     |                      |                                                                                                    | Polarity = 1, Phase = 1, from SPI0_CLK falling | 5                    |      |

(1) P = SYSCLK2 period

(2) First bit may be MSB or LSB depending upon SPI configuration. SO(0) refers to first bit and SO(n) refers to last bit output on SPI0\_SOMI. SI(0) refers to the first bit input and SI(n) refers to the last bit input on SPI0\_SIMO.

(3) Measured from the termination of the write of new data to the SPI module. In analyzing throughput requirements, additional internal bus cycles must be accounted for to allow data to be written to the SPI module by the CPU.

**Table 6-51. Additional<sup>(1)</sup> SPI0 Master Timings, 4-Pin Enable Option<sup>(2) (3)</sup>**

| No. | PARAMETER          |                                                                                                                                      | MIN                                            | MAX                         | UNIT |
|-----|--------------------|--------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|-----------------------------|------|
| 17  | $t_{d(ENA\_SPC)M}$ | Delay from slave assertion of SPI0_ENA active to first SPI0_CLK from master. <sup>(4)</sup>                                          | Polarity = 0, Phase = 0, to SPI0_CLK rising    | 3P + 3.6                    | ns   |
|     |                    |                                                                                                                                      | Polarity = 0, Phase = 1, to SPI0_CLK rising    | $0.5t_{c(SPC)M} + 3P + 3.6$ |      |
|     |                    |                                                                                                                                      | Polarity = 1, Phase = 0, to SPI0_CLK falling   | 3P + 3.6                    |      |
|     |                    |                                                                                                                                      | Polarity = 1, Phase = 1, to SPI0_CLK falling   | $0.5t_{c(SPC)M} + 3P + 3.6$ |      |
| 18  | $t_{d(SPC\_ENA)M}$ | Max delay for slave to deassert SPI0_ENA after final SPI0_CLK edge to ensure master does not begin the next transfer. <sup>(5)</sup> | Polarity = 0, Phase = 0, from SPI0_CLK falling | $0.5t_{c(SPC)M} + P + 5$    | ns   |
|     |                    |                                                                                                                                      | Polarity = 0, Phase = 1, from SPI0_CLK falling | P + 5                       |      |
|     |                    |                                                                                                                                      | Polarity = 1, Phase = 0, from SPI0_CLK rising  | $0.5t_{c(SPC)M} + P + 5$    |      |
|     |                    |                                                                                                                                      | Polarity = 1, Phase = 1, from SPI0_CLK rising  | P + 5                       |      |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-49).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before SPI0\_ENA assertion.

(5) In the case where the master SPI is ready with new data before SPI0\_ENA deassertion.

**Table 6-52. Additional<sup>(1)</sup> SPI0 Master Timings, 4-Pin Chip Select Option<sup>(2) (3)</sup>**

| No. | PARAMETER          |                                                                                  | MIN                                            | MAX                       | UNIT |
|-----|--------------------|----------------------------------------------------------------------------------|------------------------------------------------|---------------------------|------|
| 19  | $t_{d(SCS\_SPC)M}$ | Delay from SPI0_SCS active to first SPI0_CLK <sup>(4) (5)</sup>                  | Polarity = 0, Phase = 0, to SPI0_CLK rising    | 2P - 5                    | ns   |
|     |                    |                                                                                  | Polarity = 0, Phase = 1, to SPI0_CLK rising    | $0.5t_{c(SPC)M} + 2P - 5$ |      |
|     |                    |                                                                                  | Polarity = 1, Phase = 0, to SPI0_CLK falling   | 2P - 5                    |      |
|     |                    |                                                                                  | Polarity = 1, Phase = 1, to SPI0_CLK falling   | $0.5t_{c(SPC)M} + 2P - 5$ |      |
| 20  | $t_{d(SPC\_SCS)M}$ | Delay from final SPI0_CLK edge to master deasserting SPI0_SCS <sup>(6) (7)</sup> | Polarity = 0, Phase = 0, from SPI0_CLK falling | $0.5t_{c(SPC)M} + P - 3$  | ns   |
|     |                    |                                                                                  | Polarity = 0, Phase = 1, from SPI0_CLK falling | P - 3                     |      |
|     |                    |                                                                                  | Polarity = 1, Phase = 0, from SPI0_CLK rising  | $0.5t_{c(SPC)M} + P - 3$  |      |
|     |                    |                                                                                  | Polarity = 1, Phase = 1, from SPI0_CLK rising  | P - 3                     |      |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-49).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before SPI0\_SCS assertion.

(5) This delay can be increased under software control by the register bit field SPIDELAY.C2TDELAY[4:0].

(6) Except for modes when SPIDAT1.CSHOLD is enabled and there is additional data to transmit. In this case, SPI0\_SCS will remain asserted.

(7) This delay can be increased under software control by the register bit field SPIDELAY.T2CDELAY[4:0].

**Table 6-53. Additional<sup>(1)</sup> SPI0 Master Timings, 5-Pin Option<sup>(2) (3)</sup>**

| No. | PARAMETER            |                                                                                                                                                                 | MIN                                               | MAX                         | UNIT |
|-----|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------|-----------------------------|------|
| 18  | $t_{d(SPC\_ENA)M}$   | Max delay for slave to deassert $\overline{SPI0\_ENA}$ after final $SPI0\_CLK$ edge to ensure master does not begin the next transfer. <sup>(4)</sup>           | Polarity = 0, Phase = 0, from $SPI0\_CLK$ falling | $0.5t_{c(SPC)M} + P + 5$    | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1, from $SPI0\_CLK$ falling | $P + 5$                     |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0, from $SPI0\_CLK$ rising  | $0.5t_{c(SPC)M} + P + 5$    |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1, from $SPI0\_CLK$ rising  | $P + 5$                     |      |
| 20  | $t_{d(SPC\_SCS)M}$   | Delay from final $SPI0\_CLK$ edge to master deasserting $\overline{SPI0\_SCS}$ <sup>(5) (6)</sup>                                                               | Polarity = 0, Phase = 0, from $SPI0\_CLK$ falling | $0.5t_{c(SPC)M} + P - 3$    | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1, from $SPI0\_CLK$ falling | $P - 3$                     |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0, from $SPI0\_CLK$ rising  | $0.5t_{c(SPC)M} + P - 3$    |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1, from $SPI0\_CLK$ rising  | $P - 3$                     |      |
| 21  | $t_{d(SCSL\_ENAL)M}$ | Max delay for slave SPI to drive $\overline{SPI0\_ENA}$ valid after master asserts $\overline{SPI0\_SCS}$ to delay the master from beginning the next transfer, |                                                   | $C2TDELAY + P$              | ns   |
| 22  | $t_{d(SCS\_SPC)M}$   | Delay from $\overline{SPI0\_SCS}$ active to first $SPI0\_CLK$ <sup>(7) (8) (9)</sup>                                                                            | Polarity = 0, Phase = 0, to $SPI0\_CLK$ rising    | $2P - 5$                    | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1, to $SPI0\_CLK$ rising    | $0.5t_{c(SPC)M} + 2P - 5$   |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0, to $SPI0\_CLK$ falling   | $2P - 5$                    |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1, to $SPI0\_CLK$ falling   | $0.5t_{c(SPC)M} + 2P - 5$   |      |
| 23  | $t_{d(ENA\_SPC)M}$   | Delay from assertion of $\overline{SPI0\_ENA}$ low to first $SPI0\_CLK$ edge. <sup>(10)</sup>                                                                   | Polarity = 0, Phase = 0, to $SPI0\_CLK$ rising    | $3P + 3.6$                  | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1, to $SPI0\_CLK$ rising    | $0.5t_{c(SPC)M} + 3P + 3.6$ |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0, to $SPI0\_CLK$ falling   | $3P + 3.6$                  |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1, to $SPI0\_CLK$ falling   | $0.5t_{c(SPC)M} + 3P + 3.6$ |      |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-50).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before  $\overline{SPI0\_ENA}$  deassertion.

(5) Except for modes when  $SPIDAT1.CSHOLD$  is enabled and there is additional data to transmit. In this case,  $\overline{SPI0\_SCS}$  will remain asserted.

(6) This delay can be increased under software control by the register bit field  $SPIDELAY.T2CDELAY[4:0]$ .

(7) If  $\overline{SPI0\_ENA}$  is asserted immediately such that the transmission is not delayed by  $\overline{SPI0\_ENA}$ .

(8) In the case where the master SPI is ready with new data before  $\overline{SPI0\_SCS}$  assertion.

(9) This delay can be increased under software control by the register bit field  $SPIDELAY.C2TDELAY[4:0]$ .

(10) If  $\overline{SPI0\_ENA}$  was initially deasserted high and  $SPI0\_CLK$  is delayed.

**Table 6-54. Additional<sup>(1)</sup> SPI0 Slave Timings, 4-Pin Enable Option<sup>(2) (3)</sup>**

| No. | PARAMETER                 |                                                               | MIN                                                                               | MAX                         | UNIT |
|-----|---------------------------|---------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------|------|
| 24  | t <sub>d(SPC_ENAH)S</sub> | Delay from final SPI0_CLK edge to slave deasserting SPI0_ENA. | Polarity = 0, Phase = 0, from SPI0_CLK falling                                    | 1.5 P - 3      2.5 P + 18.5 | ns   |
|     |                           | Polarity = 0, Phase = 1, from SPI0_CLK falling                | − 0.5t <sub>c(SPC)M</sub> + 1.5 P - 3    − 0.5t <sub>c(SPC)M</sub> + 2.5 P + 18.5 |                             |      |
|     |                           | Polarity = 1, Phase = 0, from SPI0_CLK rising                 | 1.5 P - 3      2.5 P + 18.5                                                       |                             |      |
|     |                           | Polarity = 1, Phase = 1, from SPI0_CLK rising                 | − 0.5t <sub>c(SPC)M</sub> + 1.5 P - 3    − 0.5t <sub>c(SPC)M</sub> + 2.5 P + 18.5 |                             |      |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-50).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

**Table 6-55. Additional<sup>(1)</sup> SPI0 Slave Timings, 4-Pin Chip Select Option<sup>(2) (3)</sup>**

| No. | PARAMETER              |                                                                                 | MIN                                            | MAX                      | UNIT |
|-----|------------------------|---------------------------------------------------------------------------------|------------------------------------------------|--------------------------|------|
| 25  | $t_{d(SCSL\_SPC)S}$    | Required delay from SPI0_SCS asserted at slave to first SPI0_CLK edge at slave. |                                                | 2P                       | ns   |
| 26  | $t_{d(SPC\_SCSH)S}$    | Required delay from final SPI0_CLK edge before SPI0_SCS is deasserted.          | Polarity = 0, Phase = 0, from SPI0_CLK falling | $0.5t_{c(SPC)M} + P + 5$ | ns   |
|     |                        |                                                                                 | Polarity = 0, Phase = 1, from SPI0_CLK falling | P + 5                    |      |
|     |                        |                                                                                 | Polarity = 1, Phase = 0, from SPI0_CLK rising  | $0.5t_{c(SPC)M} + P + 5$ |      |
|     |                        |                                                                                 | Polarity = 1, Phase = 1, from SPI0_CLK rising  | P + 5                    |      |
| 27  | $t_{ena(SCSL\_SOMI)S}$ | Delay from master asserting SPI0_SCS to slave driving SPI0_SOMI valid           |                                                | P + 18.5                 | ns   |
| 28  | $t_{dis(SCSH\_SOMI)S}$ | Delay from master deasserting SPI0_SCS to slave 3-stating SPI0_SOMI             |                                                | P + 18.5                 | ns   |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-50).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

**Table 6-56. Additional<sup>(1)</sup> SPI0 Slave Timings, 5-Pin Option<sup>(2)</sup> (3)**

| No. | PARAMETER                                                                                                                               | MIN                                               | MAX                      | UNIT |
|-----|-----------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------|--------------------------|------|
| 25  | $t_{d(SCSL\_SPC)S}$ Required delay from $\overline{SPI0\_SCS}$ asserted at slave to first $SPI0\_CLK$ edge at slave.                    | 2P                                                |                          | ns   |
| 26  | $t_{d(SPC\_SCSH)S}$ Required delay from final $SPI0\_CLK$ edge before $\overline{SPI0\_SCS}$ is deasserted.                             | Polarity = 0, Phase = 0, from $SPI0\_CLK$ falling | $0.5t_{c(SPC)M} + P + 5$ | ns   |
|     |                                                                                                                                         | Polarity = 0, Phase = 1, from $SPI0\_CLK$ falling | $P + 5$                  |      |
|     |                                                                                                                                         | Polarity = 1, Phase = 0, from $SPI0\_CLK$ rising  | $0.5t_{c(SPC)M} + P + 5$ |      |
|     |                                                                                                                                         | Polarity = 1, Phase = 1, from $SPI0\_CLK$ rising  | $P + 5$                  |      |
| 27  | $t_{ena(SCSL\_SOMI)S}$ Delay from master asserting $\overline{SPI0\_SCS}$ to slave driving $SPI0\_SOMI$ valid                           |                                                   | $P + 18.5$               | ns   |
| 28  | $t_{dis(SCSH\_SOMI)S}$ Delay from master deasserting $\overline{SPI0\_SCS}$ to slave 3-stating $SPI0\_SOMI$                             |                                                   | $P + 18.5$               | ns   |
| 29  | $t_{ena(SCSL\_ENA)S}$ Delay from master deasserting $\overline{SPI0\_SCS}$ to slave driving $SPI0\_ENA$ valid                           |                                                   | 18.5                     | ns   |
| 30  | $t_{dis(SPC\_ENA)S}$ Delay from final clock receive edge on $SPI0\_CLK$ to slave 3-stating or driving high $SPI0\_ENA$ . <sup>(4)</sup> | Polarity = 0, Phase = 0, from $SPI0\_CLK$ falling | $2.5 P + 18.5$           | ns   |
|     |                                                                                                                                         | Polarity = 0, Phase = 1, from $SPI0\_CLK$ rising  | $2.5 P + 18.5$           |      |
|     |                                                                                                                                         | Polarity = 1, Phase = 0, from $SPI0\_CLK$ rising  | $2.5 P + 18.5$           |      |
|     |                                                                                                                                         | Polarity = 1, Phase = 1, from $SPI0\_CLK$ falling | $2.5 P + 18.5$           |      |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-50).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

(4)  $SPI0\_ENA$  is driven low after the transmission completes if the  $SPIINT0.ENABLE\_HIGHZ$  bit is programmed to 0. Otherwise it is 3-stated. If 3-stated, an external pullup resistor should be used to provide a valid level to the master. This option is useful when tying several SPI slave devices to a single master.

**Table 6-57. General Timing Requirements for SPI1 Master Modes<sup>(1)</sup>**

| No. | PARAMETER                                                                                                       | MIN                                               | MAX                    | UNIT |
|-----|-----------------------------------------------------------------------------------------------------------------|---------------------------------------------------|------------------------|------|
| 1   | $t_{c(SPC)M}$ Cycle Time, $SPI1\_CLK$ , All Master Modes                                                        | greater of 3P or 20                               | 256P                   | ns   |
| 2   | $t_{w(SPCH)M}$ Pulse Width High, $SPI1\_CLK$ , All Master Modes                                                 | $0.5t_{c(SPC)M} - 1$                              |                        | ns   |
| 3   | $t_{w(SPCL)M}$ Pulse Width Low, $SPI1\_CLK$ , All Master Modes                                                  | $0.5t_{c(SPC)M} - 1$                              |                        | ns   |
| 4   | $t_{d(SIMO\_SPC)M}$ Delay, initial data bit valid on $SPI1\_SIMO$ to initial edge on $SPI1\_CLK$ <sup>(2)</sup> | Polarity = 0, Phase = 0, to $SPI1\_CLK$ rising    | 5                      | ns   |
|     |                                                                                                                 | Polarity = 0, Phase = 1, to $SPI1\_CLK$ rising    | $- 0.5t_{c(SPC)M} + 5$ |      |
|     |                                                                                                                 | Polarity = 1, Phase = 0, to $SPI1\_CLK$ falling   | 5                      |      |
|     |                                                                                                                 | Polarity = 1, Phase = 1, to $SPI1\_CLK$ falling   | $- 0.5t_{c(SPC)M} + 5$ |      |
| 5   | $t_{d(SPC\_SIMO)M}$ Delay, subsequent bits valid on $SPI1\_SIMO$ after transmit edge of $SPI1\_CLK$             | Polarity = 0, Phase = 0, from $SPI1\_CLK$ rising  | 5                      | ns   |
|     |                                                                                                                 | Polarity = 0, Phase = 1, from $SPI1\_CLK$ falling | 5                      |      |
|     |                                                                                                                 | Polarity = 1, Phase = 0, from $SPI1\_CLK$ falling | 5                      |      |
|     |                                                                                                                 | Polarity = 1, Phase = 1, from $SPI1\_CLK$ rising  | 5                      |      |

(1) P = SYSCLK2 period

(2) First bit may be MSB or LSB depending upon SPI configuration.  $MO(0)$  refers to first bit and  $MO(n)$  refers to last bit output on  $SPI1\_SIMO$ .  $MI(0)$  refers to the first bit input and  $MI(n)$  refers to the last bit input on  $SPI1\_SOMI$ .

**Table 6-57. General Timing Requirements for SPI1 Master Modes<sup>(1)</sup> (continued)**

| No. | PARAMETER             |                                                                   | MIN                                            | MAX                   | UNIT |
|-----|-----------------------|-------------------------------------------------------------------|------------------------------------------------|-----------------------|------|
| 6   | $t_{oh}(SPC\_SIMO)_M$ | Output hold time, SPI1_SIMO valid after receive edge of SPI1_CLK  | Polarity = 0, Phase = 0, from SPI1_CLK falling | $0.5t_{c(SPC)_M} - 3$ | ns   |
|     |                       | Polarity = 0, Phase = 1, from SPI1_CLK rising                     | $0.5t_{c(SPC)_M} - 3$                          |                       |      |
|     |                       | Polarity = 1, Phase = 0, from SPI1_CLK rising                     | $0.5t_{c(SPC)_M} - 3$                          |                       |      |
|     |                       | Polarity = 1, Phase = 1, from SPI1_CLK falling                    | $0.5t_{c(SPC)_M} - 3$                          |                       |      |
| 7   | $t_{su}(SOMI\_SPC)_M$ | Input Setup Time, SPI1_SOMI valid before receive edge of SPI1_CLK | Polarity = 0, Phase = 0, to SPI1_CLK falling   | 0                     | ns   |
|     |                       | Polarity = 0, Phase = 1, to SPI1_CLK rising                       | 0                                              |                       |      |
|     |                       | Polarity = 1, Phase = 0, to SPI1_CLK rising                       | 0                                              |                       |      |
|     |                       | Polarity = 1, Phase = 1, to SPI1_CLK falling                      | 0                                              |                       |      |
| 8   | $t_{ih}(SPC\_SOMI)_M$ | Input Hold Time, SPI1_SOMI valid after receive edge of SPI1_CLK   | Polarity = 0, Phase = 0, from SPI1_CLK falling | 5                     | ns   |
|     |                       | Polarity = 0, Phase = 1, from SPI1_CLK rising                     | 5                                              |                       |      |
|     |                       | Polarity = 1, Phase = 0, from SPI1_CLK rising                     | 5                                              |                       |      |
|     |                       | Polarity = 1, Phase = 1, from SPI1_CLK falling                    | 5                                              |                       |      |

**Table 6-58. General Timing Requirements for SPI1 Slave Modes<sup>(1)</sup>**

| No. | PARAMETER             |                                                                                                    | MIN                                            | MAX | UNIT |
|-----|-----------------------|----------------------------------------------------------------------------------------------------|------------------------------------------------|-----|------|
| 9   | $t_c(SPC)_S$          | Cycle Time, SPI1_CLK, All Slave Modes                                                              | greater of 3P or 40                            |     | ns   |
| 10  | $t_w(SPCH)_S$         | Pulse Width High, SPI1_CLK, All Slave Modes                                                        | 18                                             |     | ns   |
| 11  | $t_w(SPCL)_S$         | Pulse Width Low, SPI1_CLK, All Slave Modes                                                         | 18                                             |     | ns   |
| 12  | $t_{su}(SOMI\_SPC)_S$ | Setup time, transmit data written to SPI before initial clock edge from master. <sup>(2) (3)</sup> | Polarity = 0, Phase = 0, to SPI1_CLK rising    | 2P  | ns   |
|     |                       |                                                                                                    | Polarity = 0, Phase = 1, to SPI1_CLK rising    | 2P  |      |
|     |                       |                                                                                                    | Polarity = 1, Phase = 0, to SPI1_CLK falling   | 2P  |      |
|     |                       |                                                                                                    | Polarity = 1, Phase = 1, to SPI1_CLK falling   | 2P  |      |
| 13  | $t_d(SPC\_SOMI)_S$    | Delay, subsequent bits valid on SPI1_SOMI after transmit edge of SPI1_CLK                          | Polarity = 0, Phase = 0, from SPI1_CLK rising  | 19  | ns   |
|     |                       |                                                                                                    | Polarity = 0, Phase = 1, from SPI1_CLK falling | 19  |      |
|     |                       |                                                                                                    | Polarity = 1, Phase = 0, from SPI1_CLK falling | 19  |      |
|     |                       |                                                                                                    | Polarity = 1, Phase = 1, from SPI1_CLK rising  | 19  |      |

(1) P = SYSCLK2 period

(2) First bit may be MSB or LSB depending upon SPI configuration. SO(0) refers to first bit and SO(n) refers to last bit output on SPI1\_SOMI. SI(0) refers to the first bit input and SI(n) refers to the last bit input on SPI1\_SIMO.

(3) Measured from the termination of the write of new data to the SPI module. In analyzing throughput requirements, additional internal bus cycles must be accounted for to allow data to be written to the SPI module by the CPU.

**Table 6-58. General Timing Requirements for SPI1 Slave Modes<sup>(1)</sup> (continued)**

| No. | PARAMETER            |                                                                   | MIN                                            | MAX                  | UNIT |
|-----|----------------------|-------------------------------------------------------------------|------------------------------------------------|----------------------|------|
| 14  | $t_{oh}(SPC\_SOMI)S$ | Output hold time, SPI1_SOMI valid after receive edge of SPI1_CLK  | Polarity = 0, Phase = 0, from SPI1_CLK falling | $0.5t_{c(SPC)}S - 3$ | ns   |
|     |                      | Polarity = 0, Phase = 1, from SPI1_CLK rising                     | $0.5t_{c(SPC)}S - 3$                           |                      |      |
|     |                      | Polarity = 1, Phase = 0, from SPI1_CLK rising                     | $0.5t_{c(SPC)}S - 3$                           |                      |      |
|     |                      | Polarity = 1, Phase = 1, from SPI1_CLK falling                    | $0.5t_{c(SPC)}S - 3$                           |                      |      |
| 15  | $t_{su}(SIMO\_SPC)S$ | Input Setup Time, SPI1_SIMO valid before receive edge of SPI1_CLK | Polarity = 0, Phase = 0, to SPI1_CLK falling   | 0                    | ns   |
|     |                      | Polarity = 0, Phase = 1, to SPI1_CLK rising                       | 0                                              |                      |      |
|     |                      | Polarity = 1, Phase = 0, to SPI1_CLK rising                       | 0                                              |                      |      |
|     |                      | Polarity = 1, Phase = 1, to SPI1_CLK falling                      | 0                                              |                      |      |
| 16  | $t_{ih}(SPC\_SIMO)S$ | Input Hold Time, SPI1_SIMO valid after receive edge of SPI1_CLK   | Polarity = 0, Phase = 0, from SPI1_CLK falling | 5                    | ns   |
|     |                      | Polarity = 0, Phase = 1, from SPI1_CLK rising                     | 5                                              |                      |      |
|     |                      | Polarity = 1, Phase = 0, from SPI1_CLK rising                     | 5                                              |                      |      |
|     |                      | Polarity = 1, Phase = 1, from SPI1_CLK falling                    | 5                                              |                      |      |

**Table 6-59. Additional<sup>(1)</sup> SPI1 Master Timings, 4-Pin Enable Option<sup>(2)</sup> <sup>(3)</sup>**

| No. | PARAMETER            |                                                                                                                                      | MIN                                            | MAX                       | UNIT |
|-----|----------------------|--------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|---------------------------|------|
| 17  | $t_{d(EN\_A\_SPC)M}$ | Delay from slave assertion of SPI1_ENA active to first SPI1_CLK from master. <sup>(4)</sup>                                          | Polarity = 0, Phase = 0, to SPI1_CLK rising    | $3P + 3$                  | ns   |
|     |                      |                                                                                                                                      | Polarity = 0, Phase = 1, to SPI1_CLK rising    | $0.5t_{c(SPC)M} + 3P + 3$ |      |
|     |                      |                                                                                                                                      | Polarity = 1, Phase = 0, to SPI1_CLK falling   | $3P + 3$                  |      |
|     |                      |                                                                                                                                      | Polarity = 1, Phase = 1, to SPI1_CLK falling   | $0.5t_{c(SPC)M} + 3P + 3$ |      |
| 18  | $t_{d(SPC\_ENA)M}$   | Max delay for slave to deassert SPI1_ENA after final SPI1_CLK edge to ensure master does not begin the next transfer. <sup>(5)</sup> | Polarity = 0, Phase = 0, from SPI1_CLK falling | $0.5t_{c(SPC)M} + P + 5$  | ns   |
|     |                      |                                                                                                                                      | Polarity = 0, Phase = 1, from SPI1_CLK falling | $P + 5$                   |      |
|     |                      |                                                                                                                                      | Polarity = 1, Phase = 0, from SPI1_CLK rising  | $0.5t_{c(SPC)M} + P + 5$  |      |
|     |                      |                                                                                                                                      | Polarity = 1, Phase = 1, from SPI1_CLK rising  | $P + 5$                   |      |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-57).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before SPI1\_ENA assertion.

(5) In the case where the master SPI is ready with new data before SPI1\_ENA deassertion.

**Table 6-60. Additional<sup>(1)</sup> SPI1 Master Timings, 4-Pin Chip Select Option<sup>(2) (3)</sup>**

| No. | PARAMETER          |                                                                                          | MIN                                               | MAX                      | UNIT |
|-----|--------------------|------------------------------------------------------------------------------------------|---------------------------------------------------|--------------------------|------|
| 19  | $t_{d(SCS\_SPC)M}$ | Delay from $\overline{SPI1\_SCS}$ active to first $SPI1\_CLK^{(4) (5)}$                  | Polarity = 0, Phase = 0, to $SPI1\_CLK$ rising    | 2P - 5                   | ns   |
|     |                    | Polarity = 0, Phase = 1, to $SPI1\_CLK$ rising                                           | $0.5t_{c(SPC)M} + 2P - 5$                         |                          |      |
|     |                    | Polarity = 1, Phase = 0, to $SPI1\_CLK$ falling                                          | 2P - 5                                            |                          |      |
|     |                    | Polarity = 1, Phase = 1, to $SPI1\_CLK$ falling                                          | $0.5t_{c(SPC)M} + 2P - 5$                         |                          |      |
| 20  | $t_{d(SPC\_SCS)M}$ | Delay from final $SPI1\_CLK$ edge to master deasserting $\overline{SPI1\_SCS}^{(6) (7)}$ | Polarity = 0, Phase = 0, from $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + P - 3$ | ns   |
|     |                    | Polarity = 0, Phase = 1, from $SPI1\_CLK$ falling                                        | P - 3                                             |                          |      |
|     |                    | Polarity = 1, Phase = 0, from $SPI1\_CLK$ rising                                         | $0.5t_{c(SPC)M} + P - 3$                          |                          |      |
|     |                    | Polarity = 1, Phase = 1, from $SPI1\_CLK$ rising                                         | P - 3                                             |                          |      |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-57).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before  $\overline{SPI1\_SCS}$  assertion.

(5) This delay can be increased under software control by the register bit field SPIDELAY.C2TDELAY[4:0].

(6) Except for modes when SPIDAT1.CSHOLD is enabled and there is additional data to transmit. In this case,  $\overline{SPI1\_SCS}$  will remain asserted.

(7) This delay can be increased under software control by the register bit field SPIDELAY.T2CDELAY[4:0].

**Table 6-61. Additional<sup>(1)</sup> SPI1 Master Timings, 5-Pin Option<sup>(2) (3)</sup>**

| No. | PARAMETER            |                                                                                                                                                                 | MIN                                                  | MAX                      | UNIT |
|-----|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------|--------------------------|------|
| 18  | $t_{d(SPC\_ENA)M}$   | Max delay for slave to deassert $\overline{SPI1\_ENA}$ after final $SPI1\_CLK$ edge to ensure master does not begin the next transfer. <sup>(4)</sup>           | Polarity = 0, Phase = 0,<br>from $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + P + 5$ | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1,<br>from $SPI1\_CLK$ falling | P + 5                    |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0,<br>from $SPI1\_CLK$ rising  | $0.5t_{c(SPC)M} + P + 5$ |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1,<br>from $SPI1\_CLK$ rising  | P + 5                    |      |
| 20  | $t_{d(SPC\_SCS)M}$   | Delay from final $SPI1\_CLK$ edge to master deasserting $\overline{SPI1\_SCS}$ <sup>(5) (6)</sup>                                                               | Polarity = 0, Phase = 0,<br>from $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + P - 3$ | ns   |
|     |                      |                                                                                                                                                                 | Polarity = 0, Phase = 1,<br>from $SPI1\_CLK$ falling | P - 3                    |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 0,<br>from $SPI1\_CLK$ rising  | $0.5t_{c(SPC)M} + P - 3$ |      |
|     |                      |                                                                                                                                                                 | Polarity = 1, Phase = 1,<br>from $SPI1\_CLK$ rising  | P - 3                    |      |
| 21  | $t_{d(SCSL\_ENAL)M}$ | Max delay for slave SPI to drive $\overline{SPI1\_ENA}$ valid after master asserts $\overline{SPI1\_SCS}$ to delay the master from beginning the next transfer. |                                                      | C2TDELAY + P             | ns   |

(1) These parameters are in addition to the general timings for SPI master modes (Table 6-58).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four master clocking modes.

(4) In the case where the master SPI is ready with new data before  $\overline{SPI1\_ENA}$  deassertion.

(5) Except for modes when SPIDAT1.CSHOLD is enabled and there is additional data to transmit. In this case,  $\overline{SPI1\_SCS}$  will remain asserted.

(6) This delay can be increased under software control by the register bit field SPIDELAY.T2CDELAY[4:0].

**Table 6-61. Additional<sup>(1)</sup> SPI1 Master Timings, 5-Pin Option<sup>(2) (3)</sup> (continued)**

| No. | PARAMETER          |                                                                                               | MIN                                             | MAX                       | UNIT |
|-----|--------------------|-----------------------------------------------------------------------------------------------|-------------------------------------------------|---------------------------|------|
| 22  | $t_{d(SCS\_SPC)M}$ | Delay from $\overline{SPI1\_SCS}$ active to first $SPI1\_CLK^{(7) (8) (9)}$                   | Polarity = 0, Phase = 0, to $SPI1\_CLK$ rising  | 2P - 5                    | ns   |
|     |                    |                                                                                               | Polarity = 0, Phase = 1, to $SPI1\_CLK$ rising  | $0.5t_{c(SPC)M} + 2P - 5$ |      |
|     |                    |                                                                                               | Polarity = 1, Phase = 0, to $SPI1\_CLK$ falling | 2P - 5                    |      |
|     |                    |                                                                                               | Polarity = 1, Phase = 1, to $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + 2P - 5$ |      |
| 23  | $t_{d(ENA\_SPC)M}$ | Delay from assertion of $\overline{SPI1\_ENA}$ low to first $SPI1\_CLK$ edge. <sup>(10)</sup> | Polarity = 0, Phase = 0, to $SPI1\_CLK$ rising  | 3P + 3                    | ns   |
|     |                    |                                                                                               | Polarity = 0, Phase = 1, to $SPI1\_CLK$ rising  | $0.5t_{c(SPC)M} + 3P + 3$ |      |
|     |                    |                                                                                               | Polarity = 1, Phase = 0, to $SPI1\_CLK$ falling | 3P + 3                    |      |
|     |                    |                                                                                               | Polarity = 1, Phase = 1, to $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + 3P + 3$ |      |

(7) If  $\overline{SPI1\_ENA}$  is asserted immediately such that the transmission is not delayed by  $\overline{SPI1\_ENA}$ .

(8) In the case where the master SPI is ready with new data before  $\overline{SPI1\_SCS}$  assertion.

(9) This delay can be increased under software control by the register bit field  $SPIDELAY.C2TDELAY[4:0]$ .

(10) If  $\overline{SPI1\_ENA}$  was initially deasserted high and  $SPI1\_CLK$  is delayed.

**Table 6-62. Additional<sup>(1)</sup> SPI1 Slave Timings, 4-Pin Enable Option<sup>(2) (3)</sup>**

| No. | PARAMETER           |                                                                                 | MIN                                               | MAX                                                          | UNIT |
|-----|---------------------|---------------------------------------------------------------------------------|---------------------------------------------------|--------------------------------------------------------------|------|
| 24  | $t_{d(SPC\_ENAH)S}$ | Delay from final $SPI1\_CLK$ edge to slave deasserting $\overline{SPI1\_ENA}$ . | Polarity = 0, Phase = 0, from $SPI1\_CLK$ falling | 1.5 P - 3      2.5 P + 19                                    | ns   |
|     |                     |                                                                                 | Polarity = 0, Phase = 1, from $SPI1\_CLK$ falling | $-0.5t_{c(SPC)M} + 1.5 P - 3$ $-0.5t_{c(SPC)M} + 2.5 P + 19$ |      |
|     |                     |                                                                                 | Polarity = 1, Phase = 0, from $SPI1\_CLK$ rising  | 1.5 P - 3      2.5 P + 19                                    |      |
|     |                     |                                                                                 | Polarity = 1, Phase = 1, from $SPI1\_CLK$ rising  | $-0.5t_{c(SPC)M} + 1.5 P - 3$ $-0.5t_{c(SPC)M} + 2.5 P + 19$ |      |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-58).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

**Table 6-63. Additional<sup>(1)</sup> SPI1 Slave Timings, 4-Pin Chip Select Option<sup>(2) (3)</sup>**

| No. | PARAMETER              |                                                                                                  | MIN                                               | MAX                      | UNIT |
|-----|------------------------|--------------------------------------------------------------------------------------------------|---------------------------------------------------|--------------------------|------|
| 25  | $t_{d(SCSL\_SPC)S}$    | Required delay from $\overline{SPI1\_SCS}$ asserted at slave to first $SPI1\_CLK$ edge at slave. | 2P                                                |                          | ns   |
| 26  | $t_{d(SPC\_SCSH)S}$    | Required delay from final $SPI1\_CLK$ edge before $\overline{SPI1\_SCS}$ is deasserted.          | Polarity = 0, Phase = 0, from $SPI1\_CLK$ falling | $0.5t_{c(SPC)M} + P + 5$ | ns   |
|     |                        |                                                                                                  | Polarity = 0, Phase = 1, from $SPI1\_CLK$ falling | P + 5                    |      |
|     |                        |                                                                                                  | Polarity = 1, Phase = 0, from $SPI1\_CLK$ rising  | $0.5t_{c(SPC)M} + P + 5$ |      |
|     |                        |                                                                                                  | Polarity = 1, Phase = 1, from $SPI1\_CLK$ rising  | P + 5                    |      |
| 27  | $t_{ena(SCSL\_SOMI)S}$ | Delay from master asserting $\overline{SPI1\_SCS}$ to slave driving $SPI1\_SOMI$ valid           |                                                   | P + 19                   | ns   |
| 28  | $t_{dis(SCSH\_SOMI)S}$ | Delay from master deasserting $\overline{SPI1\_SCS}$ to slave 3-stating $SPI1\_SOMI$             |                                                   | P + 19                   | ns   |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-58).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

**Table 6-64. Additional<sup>(1)</sup> SPI1 Slave Timings, 5-Pin Option<sup>(2) (3)</sup>**

| No. | PARAMETER              |                                                                                                             | MIN                                            | MAX                      | UNIT |
|-----|------------------------|-------------------------------------------------------------------------------------------------------------|------------------------------------------------|--------------------------|------|
| 25  | $t_{d(SCSL\_SPC)S}$    | Required delay from $\overline{SPI1\_SCS}$ asserted at slave to first SPI1_CLK edge at slave.               | 2P                                             |                          | ns   |
| 26  | $t_{d(SPC\_SCSH)S}$    | Required delay from final SPI1_CLK edge before $\overline{SPI1\_SCS}$ is deasserted.                        | Polarity = 0, Phase = 0, from SPI1_CLK falling | $0.5t_{c(SPC)M} + P + 5$ | ns   |
|     |                        |                                                                                                             | Polarity = 0, Phase = 1, from SPI1_CLK falling | P + 5                    |      |
|     |                        |                                                                                                             | Polarity = 1, Phase = 0, from SPI1_CLK rising  | $0.5t_{c(SPC)M} + P + 5$ |      |
|     |                        |                                                                                                             | Polarity = 1, Phase = 1, from SPI1_CLK rising  | P + 5                    |      |
| 27  | $t_{ena(SCSL\_SOMI)S}$ | Delay from master asserting $\overline{SPI1\_SCS}$ to slave driving SPI1_SOMI valid                         |                                                | P + 19                   | ns   |
| 28  | $t_{dis(SCSH\_SOMI)S}$ | Delay from master deasserting $\overline{SPI1\_SCS}$ to slave 3-stating SPI1_SOMI                           |                                                | P + 19                   | ns   |
| 29  | $t_{ena(SCSL\_ENA)S}$  | Delay from master deasserting $\overline{SPI1\_SCS}$ to slave driving SPI1_ENA valid                        |                                                | 19                       | ns   |
| 30  | $t_{dis(SPC\_ENA)S}$   | Delay from final clock receive edge on SPI1_CLK to slave 3-stating or driving high SPI1_ENA. <sup>(4)</sup> | Polarity = 0, Phase = 0, from SPI1_CLK falling | 2.5 P + 19               | ns   |
|     |                        |                                                                                                             | Polarity = 0, Phase = 1, from SPI1_CLK rising  | 2.5 P + 19               |      |
|     |                        |                                                                                                             | Polarity = 1, Phase = 0, from SPI1_CLK rising  | 2.5 P + 19               |      |
|     |                        |                                                                                                             | Polarity = 1, Phase = 1, from SPI1_CLK falling | 2.5 P + 19               |      |

(1) These parameters are in addition to the general timings for SPI slave modes (Table 6-58).

(2) P = SYSCLK2 period

(3) Figure shows only Polarity = 0, Phase = 0 as an example. Table gives parameters for all four slave clocking modes.

(4) SPI1\_ENA is driven low after the transmission completes if the SPIINT0.ENABLE\_HIGHZ bit is programmed to 0. Otherwise it is 3-stated. If 3-stated, an external pullup resistor should be used to provide a valid level to the master. This option is useful when tying several SPI slave devices to a single master.

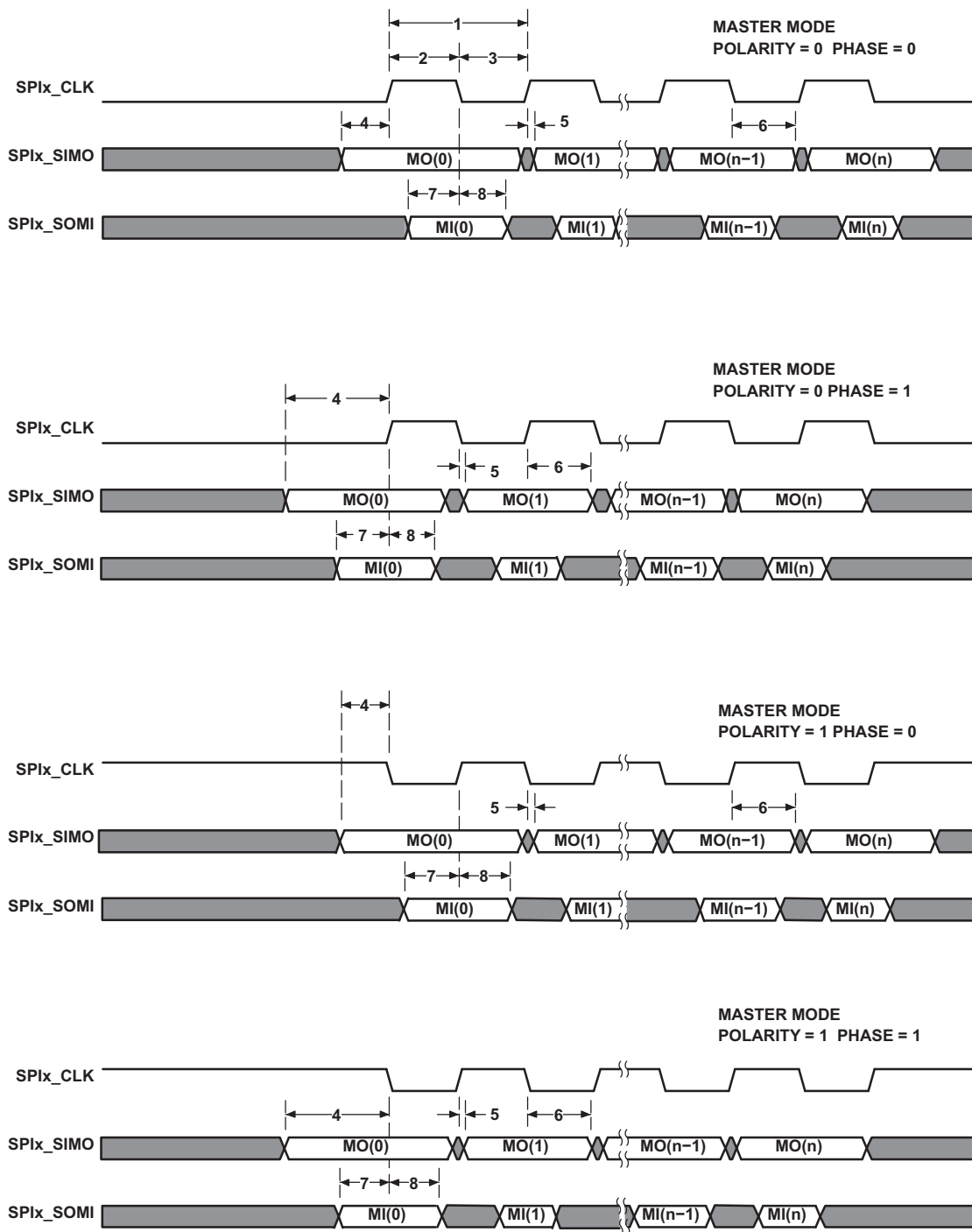


Figure 6-33. SPI Timings—Master Mode

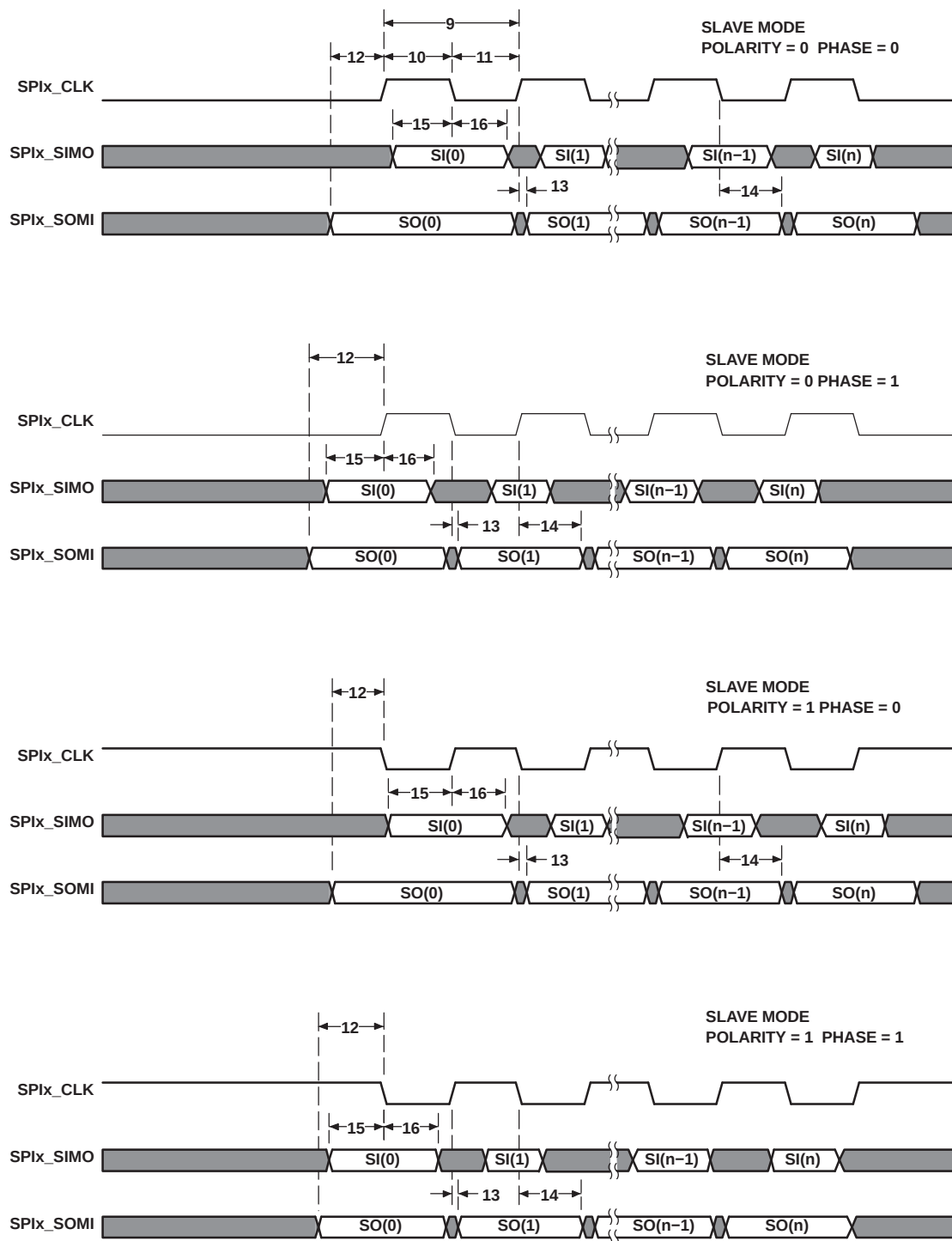
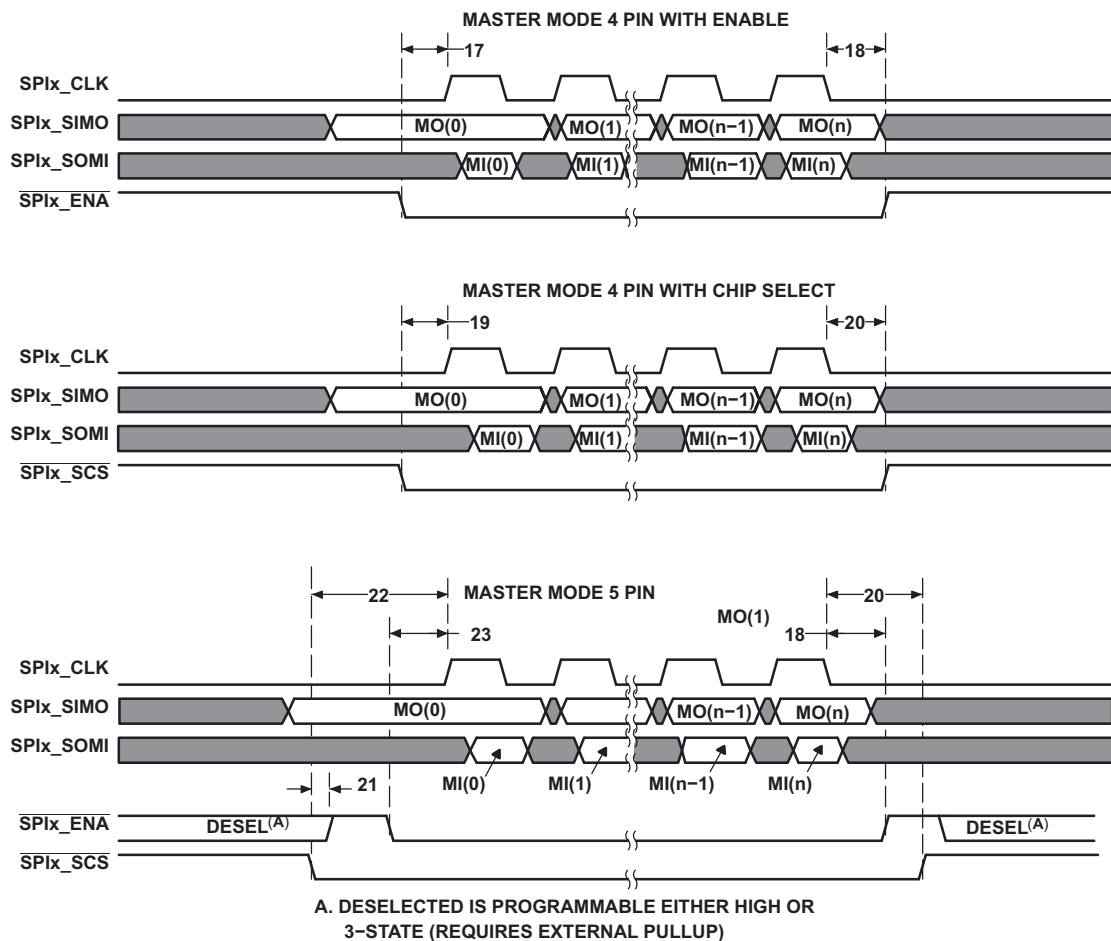
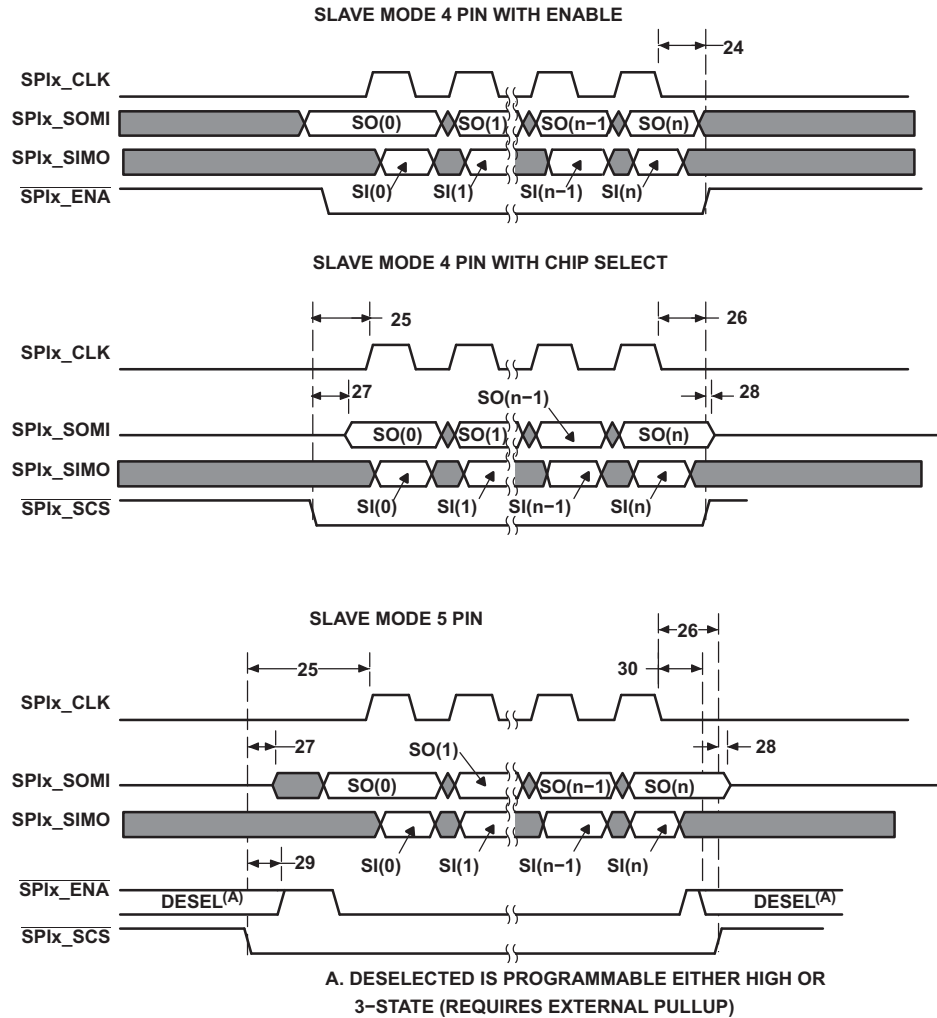


Figure 6-34. SPI Timings—Slave Mode



**Figure 6-35. SPI Timings—Master Mode (4-Pin and 5-Pin)**



**Figure 6-36. SPI Timings—Slave Mode (4-Pin and 5-Pin)**

## 6.18 Enhanced Capture (eCAP) Peripheral

The device contains up to three enhanced capture (eCAP) modules. [Figure 6-37](#) shows a functional block diagram of a module.

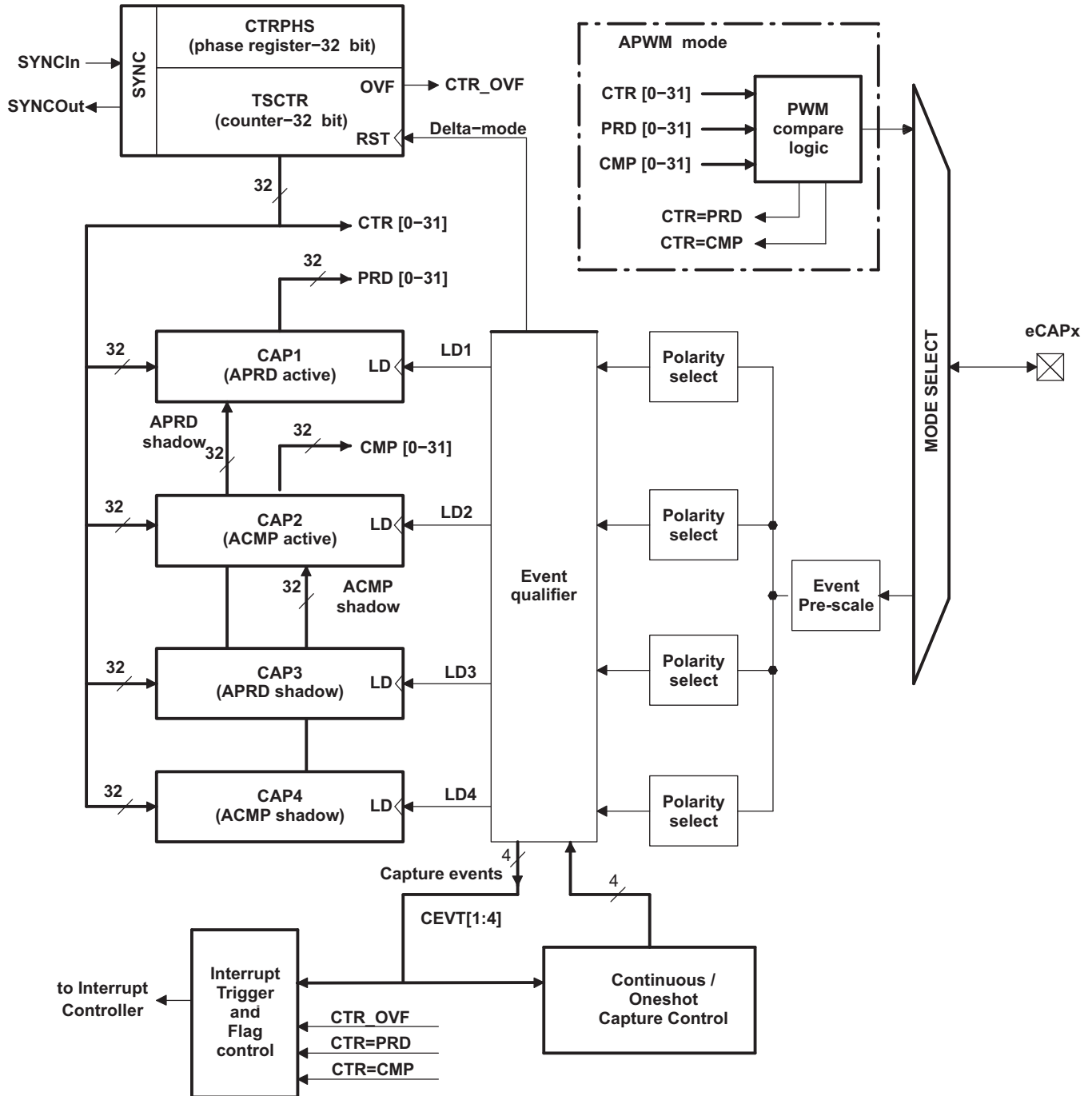
Uses for ECAP include:

- Speed measurements of rotating machinery (e.g. toothed sprockets sensed via Hall sensors)
- Elapsed time measurements between position sensor triggers
- Period and duty cycle measurements of Pulse train signals
- Decoding current or voltage amplitude derived from duty cycle encoded current/voltage sensors

The ECAP module described in this specification includes the following features:

- 32 bit time base
- 4 event time-stamp registers (each 32 bits)
- Edge polarity selection for up to 4 sequenced time-stamp capture events
- Interrupt on either of the 4 events
- Single shot capture of up to 4 event time-stamps
- Continuous mode capture of time-stamps in a 4 deep circular buffer
- Absolute time-stamp capture
- Difference mode time-stamp capture
- All the above resources are dedicated to a single input pin

The eCAP modules are clocked at the SYSCLK2 rate.



### Figure 6-37. eCAP Functional Block Diagram

Table 6-65 is the list of the ECAP registers.

**Table 6-65. ECAPx Configuration Registers**

| ECAP0<br>BYTE ADDRESS | ECAP1<br>BYTE ADDRESS | ECAP2<br>BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION                |
|-----------------------|-----------------------|-----------------------|---------|-------------------------------------|
| 0x01F0 6000           | 0x01F0 7000           | 0x01F0 8000           | TSCTR   | Time-Stamp Counter                  |
| 0x01F0 6004           | 0x01F0 7004           | 0x01F0 8004           | CTPHS   | Counter Phase Offset Value Register |
| 0x01F0 6008           | 0x01F0 7008           | 0x01F0 8008           | CAP1    | Capture 1 Register                  |
| 0x01F0 600C           | 0x01F0 700C           | 0x01F0 800C           | CAP2    | Capture 2 Register                  |
| 0x01F0 6010           | 0x01F0 7010           | 0x01F0 8010           | CAP3    | Capture 3 Register                  |
| 0x01F0 6014           | 0x01F0 7014           | 0x01F0 8014           | CAP4    | Capture 4 Register                  |
| 0x01F0 6028           | 0x01F0 7028           | 0x01F0 8028           | ECCTL1  | Capture Control Register 1          |
| 0x01F0 602A           | 0x01F0 702A           | 0x01F0 802A           | ECCTL2  | Capture Control Register 2          |
| 0x01F0 602C           | 0x01F0 702C           | 0x01F0 802C           | ECEINT  | Capture Interrupt Enable Register   |
| 0x01F0 602E           | 0x01F0 702E           | 0x01F0 802E           | ECFLG   | Capture Interrupt Flag Register     |
| 0x01F0 6030           | 0x01F0 7030           | 0x01F0 8030           | ECCLR   | Capture Interrupt Clear Register    |
| 0x01F0 6032           | 0x01F0 7032           | 0x01F0 8032           | ECFRC   | Capture Interrupt Force Register    |
| 0x01F0 605C           | 0x01F0 705C           | 0x01F0 805C           | REVID   | Revision ID                         |

Table 6-66 shows the eCAP timing requirement and Table 6-67 shows the eCAP switching characteristics.

**Table 6-66. Enhanced Capture (eCAP) Timing Requirement**

| PARAMETER                              | TEST CONDITIONS | MIN           | MAX | UNIT   |
|----------------------------------------|-----------------|---------------|-----|--------|
| $t_{w(CAP)}$ Capture input pulse width | Asynchronous    | $2t_{c(SCO)}$ |     | cycles |
|                                        | Synchronous     | $2t_{c(SCO)}$ |     | cycles |

**Table 6-67. eCAP Switching Characteristics**

| PARAMETER                                           | MIN | MAX | UNIT |
|-----------------------------------------------------|-----|-----|------|
| $t_{w(APWM)}$ Pulse duration, APWMx output high/low | 20  |     | ns   |



Table 6-68 is the list of the EQEP registers.

Table 6-69 shows the eQEP timing requirement and Table 6-70 shows the eQEP switching characteristics.

**Table 6-68. EQEP Registers**

| EQEP0<br>BYTE ADDRESS | EQEP1<br>BYTE ADDRESS | ACRONYM  | REGISTER DESCRIPTION                   |
|-----------------------|-----------------------|----------|----------------------------------------|
| 0x01F0 9000           | 0x01F0 A000           | QPOSCNT  | eQEP Position Counter                  |
| 0x01F0 9004           | 0x01F0 A004           | QPOSINIT | eQEP Initialization Position Count     |
| 0x01F0 9008           | 0x01F0 A008           | QPOSMAX  | eQEP Maximum Position Count            |
| 0x01F0 900C           | 0x01F0 A00C           | QPOSCMP  | eQEP Position-compare                  |
| 0x01F0 9010           | 0x01F0 A010           | QPOSILAT | eQEP Index Position Latch              |
| 0x01F0 9014           | 0x01F0 A014           | QPOSSLAT | eQEP Strobe Position Latch             |
| 0x01F0 9018           | 0x01F0 A018           | QPOSLAT  | eQEP Position Latch                    |
| 0x01F0 901C           | 0x01F0 A01C           | QUTMR    | eQEP Unit Timer                        |
| 0x01F0 9020           | 0x01F0 A020           | QUPRD    | eQEP Unit Period Register              |
| 0x01F0 9024           | 0x01F0 A024           | QWDTMR   | eQEP Watchdog Timer                    |
| 0x01F0 9026           | 0x01F0 A026           | QWDPRD   | eQEP Watchdog Period Register          |
| 0x01F0 9028           | 0x01F0 A028           | QDECCTL  | eQEP Decoder Control Register          |
| 0x01F0 902A           | 0x01F0 A02A           | QEPCTL   | eQEP Control Register                  |
| 0x01F0 902C           | 0x01F0 A02C           | QCAPCTL  | eQEP Capture Control Register          |
| 0x01F0 902E           | 0x01F0 A02E           | QPOSCTL  | eQEP Position-compare Control Register |
| 0x01F0 9030           | 0x01F0 A030           | QEINT    | eQEP Interrupt Enable Register         |
| 0x01F0 9032           | 0x01F0 A032           | QFLG     | eQEP Interrupt Flag Register           |
| 0x01F0 9034           | 0x01F0 A034           | QCLR     | eQEP Interrupt Clear Register          |
| 0x01F0 9036           | 0x01F0 A036           | QFRC     | eQEP Interrupt Force Register          |
| 0x01F0 9038           | 0x01F0 A038           | QEPSTS   | eQEP Status Register                   |
| 0x01F0 903A           | 0x01F0 A03A           | QCTMR    | eQEP Capture Timer                     |
| 0x01F0 903C           | 0x01F0 A03C           | QCPRD    | eQEP Capture Period Register           |
| 0x01F0 903E           | 0x01F0 A03E           | QCTMRLAT | eQEP Capture Timer Latch               |
| 0x01F0 9040           | 0x01F0 A040           | QCPRDLAT | eQEP Capture Period Latch              |
| 0x01F0 905C           | 0x01F0 A05C           | REVID    | eQEP Revision ID                       |

**Table 6-69. Enhanced Quadrature Encoder Pulse (eQEP) Timing Requirements**

| PARAMETER       | TEST CONDITIONS           | MIN                      | MAX           | UNIT   |
|-----------------|---------------------------|--------------------------|---------------|--------|
| $t_{w(QEPP)}$   | QEP input period          | Asynchronous/synchronous | $2t_{c(SCO)}$ | cycles |
| $t_{w(INDEXH)}$ | QEP Index Input High time | Asynchronous/synchronous | $2t_{c(SCO)}$ | cycles |
| $t_{w(INDEXL)}$ | QEP Index Input Low time  | Asynchronous/synchronous | $2t_{c(SCO)}$ | cycles |
| $t_{w(STROBH)}$ | QEP Strobe High time      | Asynchronous/synchronous | $2t_{c(SCO)}$ | cycles |
| $t_{w(STROBL)}$ | QEP Strobe Input Low time | Asynchronous/synchronous | $2t_{c(SCO)}$ | cycles |

**Table 6-70. eQEP Switching Characteristics**

| PARAMETER           | MIN                                                        | MAX           | UNIT   |
|---------------------|------------------------------------------------------------|---------------|--------|
| $t_{d(CNTR)xin}$    | Delay time, external clock to counter increment            | $4t_{c(SCO)}$ | cycles |
| $t_{d(PCS-OUT)QEP}$ | Delay time, QEP input edge to position compare sync output | $6t_{c(SCO)}$ | cycles |

## 6.20 Enhanced High-Resolution Pulse-Width Modulator (eHRPWM)

The device contains up to three enhanced PWM Modules (eHRPWM). Figure 6-39 shows a block diagram of multiple eHRPWM modules. Figure 4-4 shows the signal interconnections with the eHRPWM.

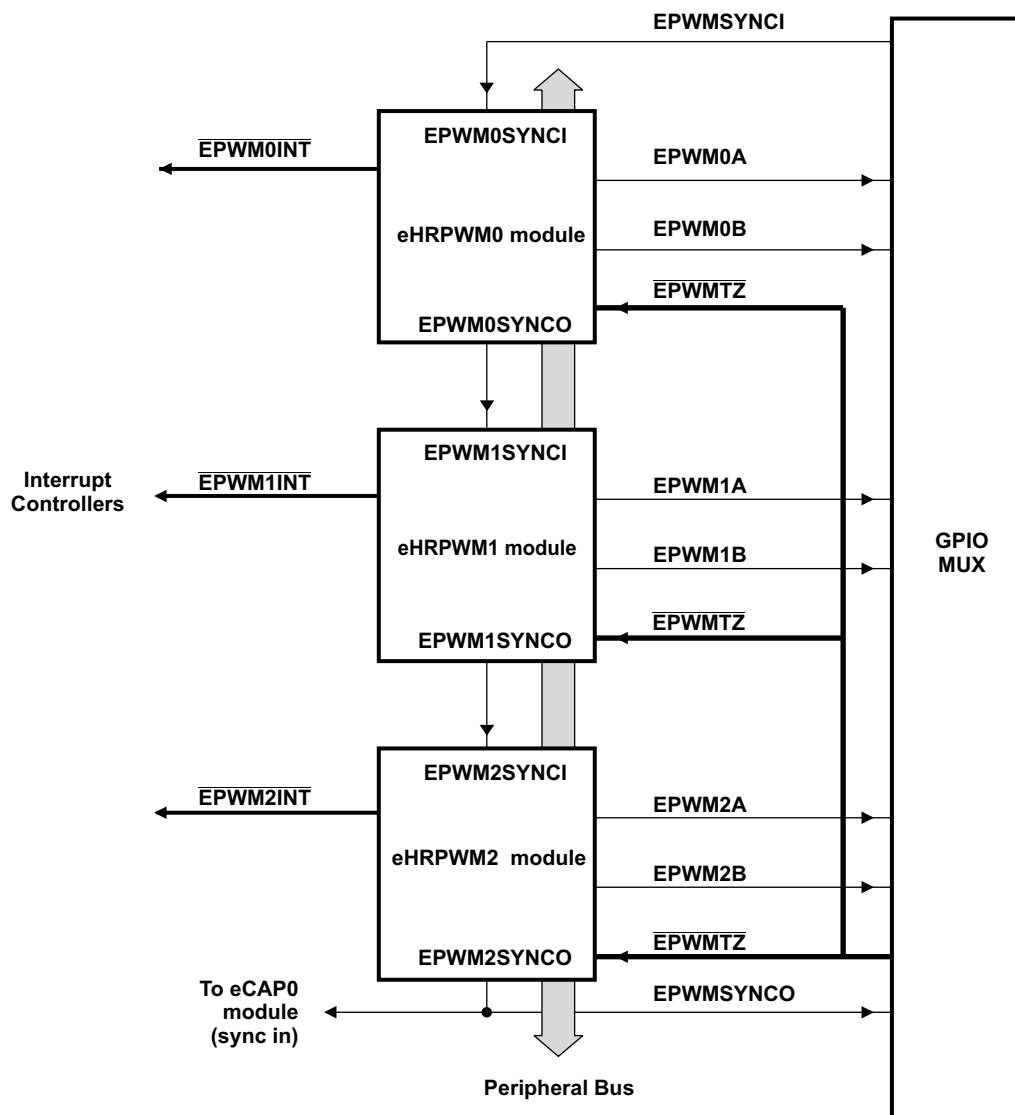


Figure 6-39. Multiple PWM Modules in the System

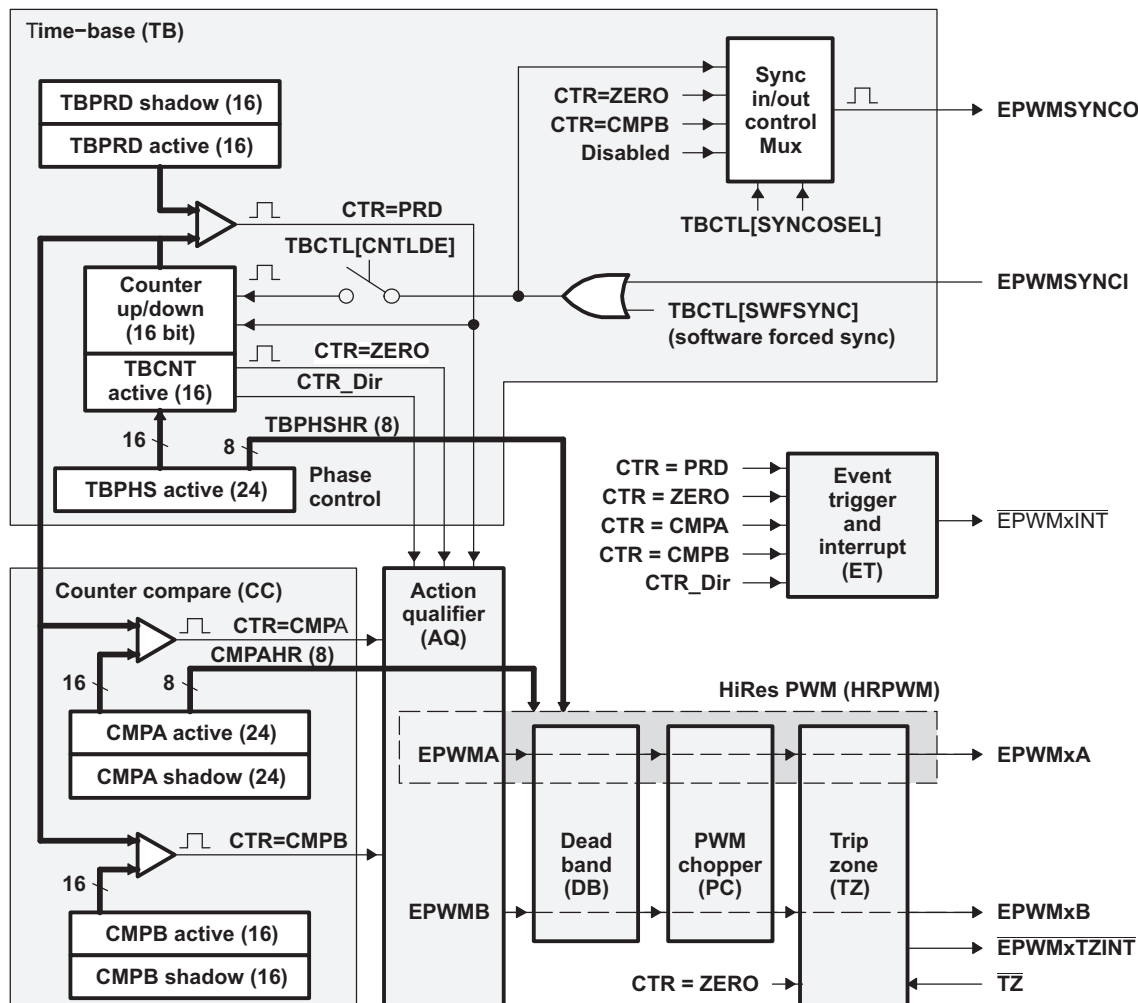


Figure 6-40. eHRPWM Sub-Modules Showing Critical Internal Signal Interconnections

Table 6-71. eHRPWM Module Control and Status Registers Grouped by Submodule

| eHRPWM0<br>BYTE<br>ADDRESS                 | eHRPWM1<br>BYTE<br>ADDRESS | eHRPWM2<br>BYTE<br>ADDRESS | ACRONYM | SIZE<br>(×16) | SHADOW | REGISTER DESCRIPTION                                          |
|--------------------------------------------|----------------------------|----------------------------|---------|---------------|--------|---------------------------------------------------------------|
| <b>TIME-BASE SUBMODULE REGISTERS</b>       |                            |                            |         |               |        |                                                               |
| 0x01F0 0000                                | 0x01F0 2000                | 0x01F0 4000                | TBCTL   | 1             | No     | Time-Base Control Register                                    |
| 0x01F0 0002                                | 0x01F0 2002                | 0x01F0 4002                | TBSTS   | 1             | No     | Time-Base Status Register                                     |
| 0x01F0 0004                                | 0x01F0 2004                | 0x01F0 4004                | TBPHSHR | 1             | No     | Extension for HRPWM Phase Register <sup>(1)</sup>             |
| 0x01F0 0006                                | 0x01F0 2006                | 0x01F0 4006                | TBPHS   | 1             | No     | Time-Base Phase Register                                      |
| 0x01F0 0008                                | 0x01F0 2008                | 0x01F0 4008                | TBCNT   | 1             | No     | Time-Base Counter Register                                    |
| 0x01F0 000A                                | 0x01F0 200A                | 0x01F0 400A                | TBPRD   | 1             | Yes    | Time-Base Period Register                                     |
| <b>COUNTER-COMPARE SUBMODULE REGISTER</b>  |                            |                            |         |               |        |                                                               |
| 0x01F0 000E                                | 0x01F0 200E                | 0x01F0 400E                | CMPCTL  | 1             | No     | Counter-Compare Control Register                              |
| 0x01F0 0010                                | 0x01F0 2010                | 0x01F0 4010                | CMPAHR  | 1             | No     | Extension for HRPWM Counter-Compare A Register <sup>(1)</sup> |
| 0x01F0 0012                                | 0x01F0 2012                | 0x01F0 4012                | CMPA    | 1             | Yes    | Counter-Compare A Register                                    |
| 0x01F0 0014                                | 0x01F0 2014                | 0x01F0 4014                | CMPB    | 1             | Yes    | Counter-Compare B Register                                    |
| <b>ACTION-QUALIFIER SUBMODULE REGISTER</b> |                            |                            |         |               |        |                                                               |

(1) These registers are only available on eHRPWM instances that include the high-resolution PWM (HRPWM) extension; otherwise, these locations are reserved.

**Table 6-71. eHRPWM Module Control and Status Registers Grouped by Submodule (continued)**

| eHRPWM0<br>BYTE<br>ADDRESS                    | eHRPWM1<br>BYTE<br>ADDRESS | eHRPWM2<br>BYTE<br>ADDRESS | ACRONYM | SIZE<br>(×16) | SHADOW | REGISTER DESCRIPTION                                      |
|-----------------------------------------------|----------------------------|----------------------------|---------|---------------|--------|-----------------------------------------------------------|
| 0x01F0 0016                                   | 0x01F0 2016                | 0x01F0 4016                | AQCTLA  | 1             | No     | Action-Qualifier Control Register for Output A (eHRPWMxA) |
| 0x01F0 0018                                   | 0x01F0 2018                | 0x01F0 4018                | AQCTLB  | 1             | No     | Action-Qualifier Control Register for Output B (eHRPWMxB) |
| 0x01F0 001A                                   | 0x01F0 201A                | 0x01F0 401A                | AQSFR   | 1             | No     | Action-Qualifier Software Force Register                  |
| 0x01F0 001C                                   | 0x01F0 201C                | 0x01F0 401C                | AQCSFR  | 1             | Yes    | Action-Qualifier Continuous S/W Force Register Set        |
| <b>DEAD-BAND GENERATOR SUBMODULE REGISTER</b> |                            |                            |         |               |        |                                                           |
| 0x01F0 001E                                   | 0x01F0 201E                | 0x01F0 401E                | DBCTL   | 1             | No     | Dead-Band Generator Control Register                      |
| 0x01F0 0020                                   | 0x01F0 2020                | 0x01F0 4020                | DBRED   | 1             | No     | Dead-Band Generator Rising Edge Delay Count Register      |
| 0x01F0 0022                                   | 0x01F0 2022                | 0x01F0 4022                | DBFED   | 1             | No     | Dead-Band Generator Falling Edge Delay Count Register     |
| <b>PWM-CHOPPER SUBMODULE REGISTER</b>         |                            |                            |         |               |        |                                                           |
| 0x01F0 003C                                   | 0x01F0 203C                | 0x01F0 403C                | PCCTL   | 1             | No     | PWM-Chopper Control Register                              |
| <b>TRIP-ZONE SUBMODULE REGISTER</b>           |                            |                            |         |               |        |                                                           |
| 0x01F0 0024                                   | 0x01F0 2024                | 0x01F0 4024                | TZSEL   | 1             | No     | Trip-Zone Select Register                                 |
| 0x01F0 0028                                   | 0x01F0 2028                | 0x01F0 4028                | TZCTL   | 1             | No     | Trip-Zone Control Register                                |
| 0x01F0 002A                                   | 0x01F0 202A                | 0x01F0 402A                | TZEINT  | 1             | No     | Trip-Zone Enable Interrupt Register                       |
| 0x01F0 002C                                   | 0x01F0 202C                | 0x01F0 402C                | TZFLG   | 1             | No     | Trip-Zone Flag Register                                   |
| 0x01F0 002E                                   | 0x01F0 202E                | 0x01F0 402E                | TZCLR   | 1             | No     | Trip-Zone Clear Register                                  |
| 0x01F0 0030                                   | 0x01F0 2030                | 0x01F0 4030                | TZFRC   | 1             | No     | Trip-Zone Force Register                                  |
| <b>EVENT-TRIGGER SUBMODULE REGISTER</b>       |                            |                            |         |               |        |                                                           |
| 0x01F0 0032                                   | 0x01F0 2032                | 0x01F0 4032                | ETSEL   | 1             | No     | Event-Trigger Selection Register                          |
| 0x01F0 0034                                   | 0x01F0 2034                | 0x01F0 4034                | ETPS    | 1             | No     | Event-Trigger Pre-Scale Register                          |
| 0x01F0 0036                                   | 0x01F0 2036                | 0x01F0 4036                | ETFLG   | 1             | No     | Event-Trigger Flag Register                               |
| 0x01F0 0038                                   | 0x01F0 2038                | 0x01F0 4038                | ETCLR   | 1             | No     | Event-Trigger Clear Register                              |
| 0x01F0 003A                                   | 0x01F0 203A                | 0x01F0 403A                | ETFRC   | 1             | No     | Event-Trigger Force Register                              |
| <b>HIGH-RESOLUTION PWM (HRPWM) SUBMODULE</b>  |                            |                            |         |               |        |                                                           |
| 0x01F0 1040                                   | 0x01F0 3040                | 0x01F0 5040                | HRCNFG  | 1             | No     | HRPWM Configuration Register <sup>(1)</sup>               |

### 6.20.1 Enhanced Pulse Width Modulator (eHRPWM) Timing

PWM refers to PWM outputs on eHRPWM1-6. Table 6-72 shows the PWM timing requirements and Table 6-73, switching characteristics.

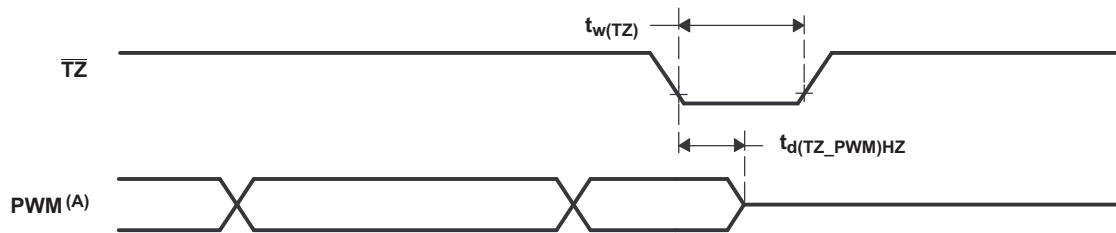
Table 6-72. eHRPWM Timing Requirements

| PARAMETER                              | TEST CONDITIONS | MIN           | MAX | UNIT   |
|----------------------------------------|-----------------|---------------|-----|--------|
| $t_{w(SYNCIN)}$ Sync input pulse width | Asynchronous    | $2t_{c(SCO)}$ |     | cycles |
|                                        | Synchronous     | $2t_{c(SCO)}$ |     | cycles |

Table 6-73. eHRPWM Switching Characteristics

| PARAMETER                                                                                                           | TEST CONDITIONS                                  | MIN           | MAX | UNIT   |
|---------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|---------------|-----|--------|
| $t_{w(PWM)}$ Pulse duration, PWMx output high/low                                                                   |                                                  | 20            |     | ns     |
| $t_{w(SYNCOUT)}$ Sync output pulse width                                                                            |                                                  | $8t_{c(SCO)}$ |     | cycles |
| $t_{d(PWM)TZA}$ Delay time, trip input active to PWM forced high<br>Delay time, trip input active to PWM forced low | no pin load;<br>no additional programmable delay |               | 25  | ns     |
| $t_{d(TZ-PWM)HZ}$ Delay time, trip input active to PWM Hi-Z                                                         | no additional programmable delay                 |               | 20  | ns     |

### 6.20.2 Trip-Zone Input Timing



- A. PWM refers to all the PWM pins in the device. The state of the PWM pins after  $\overline{TZ}$  is taken high depends on the PWM recovery software.

Figure 6-41. PWM Hi-Z Characteristics

Table 6-74. Trip-Zone input Timing Requirements

| PARAMETER                                              | TEST CONDITIONS | MIN           | MAX | UNIT   |
|--------------------------------------------------------|-----------------|---------------|-----|--------|
| $t_{w(TZ)}$ Pulse duration, $\overline{TZx}$ input low | Asynchronous    | $1t_{c(SCO)}$ |     | cycles |
|                                                        | Synchronous     | $2t_{c(SCO)}$ |     | cycles |

Table 6-75 shows the high-resolution PWM switching characteristics.

Table 6-75. High Resolution PWM Characteristics at SYSCLKOUT = (60 - 100 MHz)

| PARAMETER                                             | MIN | TYP | MAX | UNIT |
|-------------------------------------------------------|-----|-----|-----|------|
| Micro Edge Positioning (MEP) step size <sup>(1)</sup> |     | 200 |     | ps   |

- (1) MEP step size will increase with low voltage and high temperature and decrease with high voltage and cold temperature.

## 6.21 Timers

The timers support the following features:

- Configurable as single 64-bit timer or two 32-bit timers
- Period timeouts generate interrupts, DMA events or external pin events
- 8 32-bit compare registers
- Compare matches generate interrupt events
- Capture capability
- 64-bit Watchdog capability (Timer64P1 only)

[Table 6-76](#) lists the timer registers.

**Table 6-76. Timer Registers**

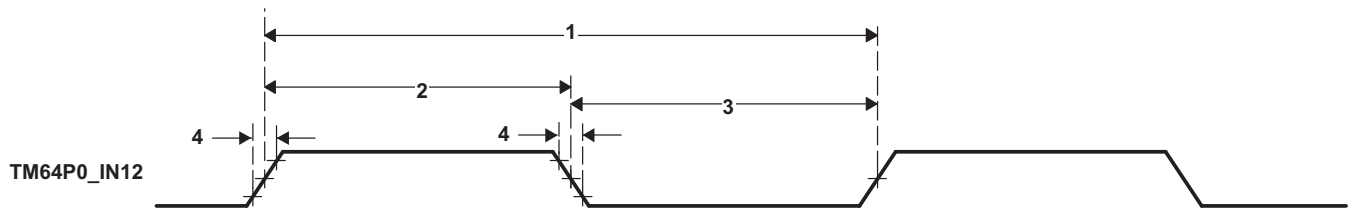
| Timer64P 0  | Timer64P 1  | ACRONYM    | REGISTER DESCRIPTION                        |
|-------------|-------------|------------|---------------------------------------------|
| 0x01C2 0000 | 0x01C2 1000 | REV        | Revision Register                           |
| 0x01C2 0004 | 0x01C2 1004 | EMUMGT     | Emulation Management Register               |
| 0x01C2 0008 | 0x01C2 1008 | GPINTGPEN  | GPIO Interrupt and GPIO Enable Register     |
| 0x01C2 000C | 0x01C2 100C | GPDATGPDIR | GPIO Data and GPIO Direction Register       |
| 0x01C2 0010 | 0x01C2 1010 | TIM12      | Timer Counter Register 12                   |
| 0x01C2 0014 | 0x01C2 1014 | TIM34      | Timer Counter Register 34                   |
| 0x01C2 0018 | 0x01C2 1018 | PRD12      | Timer Period Register 12                    |
| 0x01C2 001C | 0x01C2 101C | PRD34      | Timer Period Register 34                    |
| 0x01C2 0020 | 0x01C2 1020 | TCR        | Timer Control Register                      |
| 0x01C2 0024 | 0x01C2 1024 | TGCR       | Timer Global Control Register               |
| 0x01C2 0028 | 0x01C2 1028 | WDTCR      | Watchdog Timer Control Register             |
| 0x01C2 0034 | 0x01C2 1034 | REL12      | Timer Reload Register 12                    |
| 0x01C2 0038 | 0x01C2 1038 | REL34      | Timer Reload Register 34                    |
| 0x01C2 003C | 0x01C2 103C | CAP12      | Timer Capture Register 12                   |
| 0x01C2 0040 | 0x01C2 1040 | CAP34      | Timer Capture Register 34                   |
| 0x01C2 0044 | 0x01C2 1044 | INTCTLSTAT | Timer Interrupt Control and Status Register |
| 0x01C2 0060 | 0x01C2 1060 | CMP0       | Compare Register 0                          |
| 0x01C2 0064 | 0x01C2 1064 | CMP1       | Compare Register 1                          |
| 0x01C2 0068 | 0x01C2 1068 | CMP2       | Compare Register 2                          |
| 0x01C2 006C | 0x01C2 106C | CMP3       | Compare Register 3                          |
| 0x01C2 0070 | 0x01C2 1070 | CMP4       | Compare Register 4                          |
| 0x01C2 0074 | 0x01C2 1074 | CMP5       | Compare Register 5                          |
| 0x01C2 0078 | 0x01C2 1078 | CMP6       | Compare Register 6                          |
| 0x01C2 007C | 0x01C2 107C | CMP7       | Compare Register 7                          |

### 6.21.1 Timer Electrical Data/Timing

**Table 6-77. Timing Requirements for Timer Input<sup>(1) (2)</sup> (see Figure 6-42)**

| No. | PARAMETER                                          | MIN   | MAX                        | UNIT |
|-----|----------------------------------------------------|-------|----------------------------|------|
| 1   | $t_{c(TM64Px\_IN12)}$ Cycle time, TM64Px_IN12      | 4P    |                            | ns   |
| 2   | $t_{w(TINPH)}$ Pulse duration, TM64Px_IN12 high    | 0.45C | 0.55C                      | ns   |
| 3   | $t_{w(TINPL)}$ Pulse duration, TM64Px_IN12 low     | 0.45C | 0.55C                      | ns   |
| 4   | $t_{t(TM64Px\_IN12)}$ Transition time, TM64Px_IN12 |       | 0.25P or 10 <sup>(3)</sup> | ns   |

- (1) P = OSCIN cycle time in ns. For example, when OSCIN frequency is 27 MHz, use P = 37.037 ns.  
(2) C = TM64P0\_IN12 cycle time in ns. For example, when TM64Px\_IN12 frequency is 27 MHz, use C = 37.037 ns  
(3) Whichever is smaller. P = the period of the applied signal. Maintaining transition times as fast as possible is recommended to improve noise immunity on input signals.

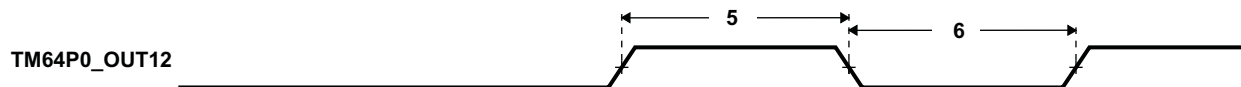


**Figure 6-42. Timer Timing**

**Table 6-78. Switching Characteristics Over Recommended Operating Conditions for Timer Output<sup>(1)</sup>**

| No. | PARAMETER                                        | MIN | MAX | UNIT |
|-----|--------------------------------------------------|-----|-----|------|
| 5   | $t_{w(TOUTH)}$ Pulse duration, TM64P0_OUT12 high | 4P  |     | ns   |
| 6   | $t_{w(TOUTL)}$ Pulse duration, TM64P0_OUT12 low  | 4P  |     | ns   |

- (1) P = OSCIN cycle time in ns. For example, when OSCIN frequency is 27 MHz, use P = 37.037 ns.



**Figure 6-43. Timer Timing**

## 6.22 Inter-Integrated Circuit Serial Ports (I2C0, I2C1)

### 6.22.1 I2C Device-Specific Information

Having two I2C modules on the device simplifies system architecture. Figure 6-44 is block diagram of the I2C Module.

Each 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
- General-Purpose I/O Capability if not used as I2C

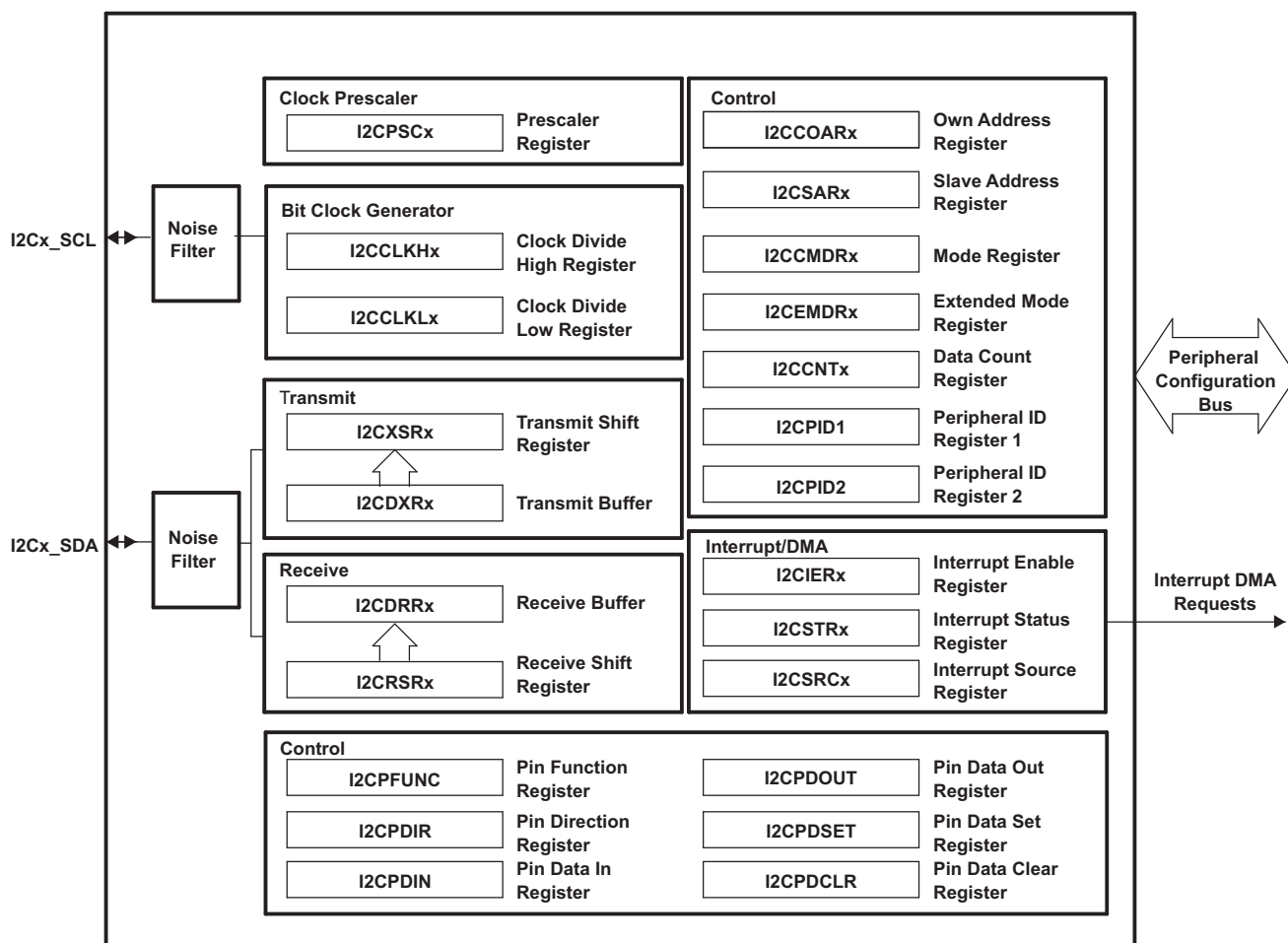


Figure 6-44. I2C Module Block Diagram

### 6.22.2 I2C Peripheral Registers Description(s)

Table 6-79 is the list of the I2C registers.

**Table 6-79. Inter-Integrated Circuit (I2C) Registers**

| I2C0<br>BYTE ADDRESS | I2C1<br>BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION                   |
|----------------------|----------------------|---------|----------------------------------------|
| 0x01C2 2000          | 0x01E2 8000          | ICOAR   | I2C Own Address Register               |
| 0x01C2 2004          | 0x01E2 8004          | ICIMR   | I2C Interrupt Mask Register            |
| 0x01C2 2008          | 0x01E2 8008          | ICSTR   | I2C Interrupt Status Register          |
| 0x01C2 200C          | 0x01E2 800C          | ICCLKL  | I2C Clock Low-Time Divider Register    |
| 0x01C2 2010          | 0x01E2 8010          | ICCLKH  | I2C Clock High-Time Divider Register   |
| 0x01C2 2014          | 0x01E2 8014          | ICCNT   | I2C Data Count Register                |
| 0x01C2 2018          | 0x01E2 8018          | ICDRR   | I2C Data Receive Register              |
| 0x01C2 201C          | 0x01E2 801C          | ICSAR   | I2C Slave Address Register             |
| 0x01C2 2020          | 0x01E2 8020          | ICDXR   | I2C Data Transmit Register             |
| 0x01C2 2024          | 0x01E2 8024          | ICMDR   | I2C Mode Register                      |
| 0x01C2 2028          | 0x01E2 8028          | ICIVR   | I2C Interrupt Vector Register          |
| 0x01C2 202C          | 0x01E2 802C          | ICEMDR  | I2C Extended Mode Register             |
| 0x01C2 2030          | 0x01E2 8030          | ICPSC   | I2C Prescaler Register                 |
| 0x01C2 2034          | 0x01E2 8034          | REVID1  | I2C Revision Identification Register 1 |
| 0x01C2 2038          | 0x01E2 8038          | REVID2  | I2C Revision Identification Register 2 |
| 0x01C2 2048          | 0x01E2 8048          | ICPFUNC | I2C Pin Function Register              |
| 0x01C2 204C          | 0x01E2 804C          | ICPDIR  | I2C Pin Direction Register             |
| 0x01C2 2050          | 0x01E2 8050          | ICPDIN  | I2C Pin Data In Register               |
| 0x01C2 2054          | 0x01E2 8054          | ICPDOUT | I2C Pin Data Out Register              |
| 0x01C2 2058          | 0x01E2 8058          | ICPDSET | I2C Pin Data Set Register              |
| 0x01C2 205C          | 0x01E2 805C          | ICPDCLR | I2C Pin Data Clear Register            |

## 6.22.3 I2C Electrical Data/Timing

### 6.22.3.1 Inter-Integrated Circuit (I2C) Timing

Table 6-80 and Table 6-81 assume testing over recommended operating conditions (see Figure 6-45 and Figure 6-46).

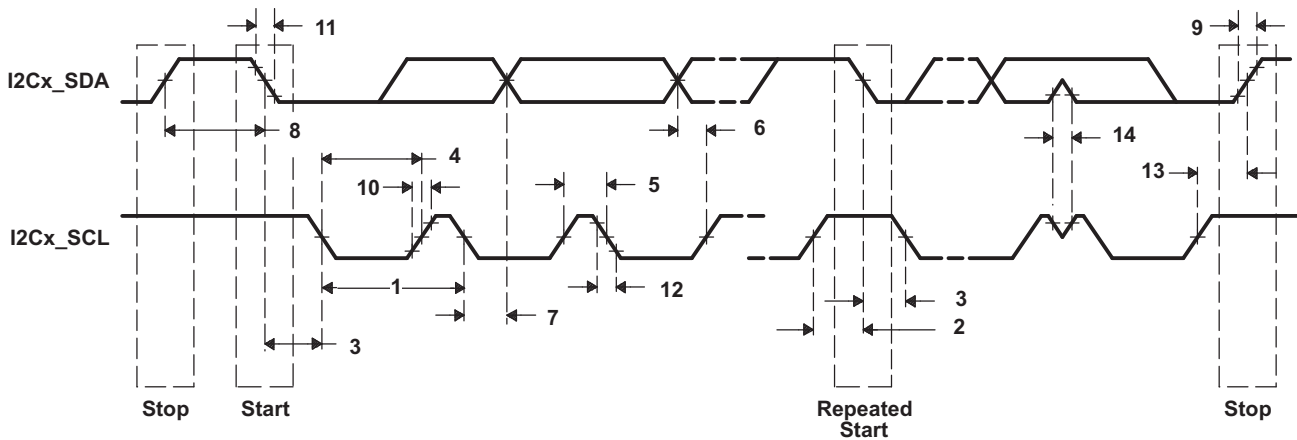
**Table 6-80. I2C Input Timing Requirements**

| No. | PARAMETER           |                                                | MIN           | MAX               | UNIT    |
|-----|---------------------|------------------------------------------------|---------------|-------------------|---------|
| 1   | $t_{c(SCL)}$        | Cycle time, I2Cx_SCL                           | Standard Mode | 10                | $\mu$ S |
|     |                     |                                                | Fast Mode     | 2.5               |         |
| 2   | $t_{su(SCLH-SDAL)}$ | Setup time, I2Cx_SCL high before I2Cx_SDA low  | Standard Mode | 4.7               | $\mu$ S |
|     |                     |                                                | Fast Mode     | 0.6               |         |
| 3   | $t_{h(SCLL-SDAL)}$  | Hold time, I2Cx_SCL low after I2Cx_SDA low     | Standard Mode | 4                 | $\mu$ S |
|     |                     |                                                | Fast Mode     | 0.6               |         |
| 4   | $t_{w(SCLL)}$       | Pulse duration, I2Cx_SCL low                   | Standard Mode | 4.7               | $\mu$ S |
|     |                     |                                                | Fast Mode     | 1.3               |         |
| 5   | $t_{w(SCLH)}$       | Pulse duration, I2Cx_SCL high                  | Standard Mode | 4                 | $\mu$ S |
|     |                     |                                                | Fast Mode     | 0.6               |         |
| 6   | $t_{su(SDA-SCLH)}$  | Setup time, I2Cx_SDA before I2Cx_SCL high      | Standard Mode | 250               | ns      |
|     |                     |                                                | Fast Mode     | 100               |         |
| 7   | $t_{h(SDA-SCLL)}$   | Hold time, I2Cx_SDA after I2Cx_SCL low         | Standard Mode | 0                 | $\mu$ S |
|     |                     |                                                | Fast Mode     | 0 0.9             |         |
| 8   | $t_{w(SDAH)}$       | Pulse duration, I2Cx_SDA high                  | Standard Mode | 4.7               | $\mu$ S |
|     |                     |                                                | Fast Mode     | 1.3               |         |
| 9   | $t_{r(SDA)}$        | Rise time, I2Cx_SDA                            | Standard Mode | 1000              | ns      |
|     |                     |                                                | Fast Mode     | $20 + 0.1C_b$ 300 |         |
| 10  | $t_{r(SCL)}$        | Rise time, I2Cx_SCL                            | Standard Mode | 1000              | ns      |
|     |                     |                                                | Fast Mode     | $20 + 0.1C_b$ 300 |         |
| 11  | $t_{f(SDA)}$        | Fall time, I2Cx_SDA                            | Standard Mode | 300               | ns      |
|     |                     |                                                | Fast Mode     | $20 + 0.1C_b$ 300 |         |
| 12  | $t_{f(SCL)}$        | Fall time, I2Cx_SCL                            | Standard Mode | 300               | ns      |
|     |                     |                                                | Fast Mode     | $20 + 0.1C_b$ 300 |         |
| 13  | $t_{su(SCLH-SDAH)}$ | Setup time, I2Cx_SCL high before I2Cx_SDA high | Standard Mode | 4                 | $\mu$ S |
|     |                     |                                                | Fast Mode     | 0.6               |         |
| 14  | $t_{w(SP)}$         | Pulse duration, spike (must be suppressed)     | Standard Mode | N/A               | ns      |
|     |                     |                                                | Fast Mode     | 0 50              |         |
| 15  | $C_b$               | Capacitive load for each bus line              | Standard Mode | 400               | pF      |
|     |                     |                                                | Fast Mode     | 400               |         |

**Table 6-81. I2C Switching Characteristics<sup>(1)</sup>**

| No. | PARAMETER           |                                                 | MIN           | MAX | UNIT    |
|-----|---------------------|-------------------------------------------------|---------------|-----|---------|
| 16  | $t_{c(SCL)}$        | Cycle time, I2Cx_SCL                            | Standard Mode | 10  | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 2.5 |         |
| 17  | $t_{su(SCLH-SDAL)}$ | Setup time, I2Cx_SCL high before I2Cx_SDA low   | Standard Mode | 4.7 | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 0.6 |         |
| 18  | $t_h(SDAL-SCLL)$    | Hold time, I2Cx_SCL low after I2Cx_SDA low      | Standard Mode | 4   | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 0.6 |         |
| 19  | $t_w(SCLL)$         | Pulse duration, I2Cx_SCL low                    | Standard Mode | 4.7 | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 1.3 |         |
| 20  | $t_w(SCLH)$         | Pulse duration, I2Cx_SCL high                   | Standard Mode | 4   | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 0.6 |         |
| 21  | $t_{su(SDAV-SCLH)}$ | Setup time, I2Cx_SDA valid before I2Cx_SCL high | Standard Mode | 250 | ns      |
|     |                     |                                                 | Fast Mode     | 100 |         |
| 22  | $t_h(SCLL-SDAV)$    | Hold time, I2Cx_SDA valid after I2Cx_SCL low    | Standard Mode | 0   | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 0   | 0.9     |
| 23  | $t_w(SDAH)$         | Pulse duration, I2Cx_SDA high                   | Standard Mode | 4.7 | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 1.3 |         |
| 28  | $t_{su(SCLH-SDAH)}$ | Setup time, I2Cx_SCL high before I2Cx_SDA high  | Standard Mode | 4   | $\mu$ S |
|     |                     |                                                 | Fast Mode     | 0.6 |         |

(1) I2C must be configured correctly to meet the timings in [Table 6-81](#).



**Figure 6-45. I2C Receive Timings**

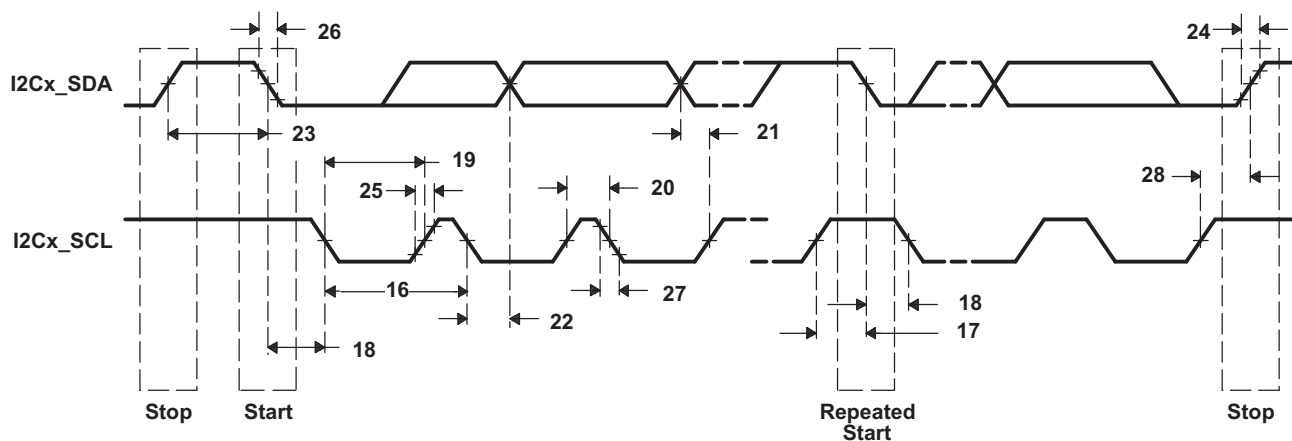


Figure 6-46. I2C Transmit Timings

## 6.23 Universal Asynchronous Receiver/Transmitter (UART)

The device has 3 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
- Autoflow control signals (CTS, RTS) on UART0 only
- DMA signaling capability for both received and transmitted data
- Programmable auto-rts and auto-cts for autoflow control
- Programmable Baud Rate up to 3MBaud
- Programmable Oversampling Options of x13 and x16
- 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

The UART registers are listed in [Section 6.23.1](#)

### 6.23.1 UART Peripheral Registers Description(s)

[Table 6-82](#) is the list of UART registers.

**Table 6-82. UART Registers**

| UART0<br>BYTE ADDRESS | UART1<br>BYTE ADDRESS | UART2<br>BYTE ADDRESS | ACRONYM     | REGISTER DESCRIPTION                          |
|-----------------------|-----------------------|-----------------------|-------------|-----------------------------------------------|
| 0x01C4 2000           | 0x01D0 C000           | 0x01D0 D000           | RBR         | Receiver Buffer Register (read only)          |
| 0x01C4 2000           | 0x01D0 C000           | 0x01D0 D000           | THR         | Transmitter Holding Register (write only)     |
| 0x01C4 2004           | 0x01D0 C004           | 0x01D0 D004           | IER         | Interrupt Enable Register                     |
| 0x01C4 2008           | 0x01D0 C008           | 0x01D0 D008           | IIR         | Interrupt Identification Register (read only) |
| 0x01C4 2008           | 0x01D0 C008           | 0x01D0 D008           | FCR         | FIFO Control Register (write only)            |
| 0x01C4 200C           | 0x01D0 C00C           | 0x01D0 D00C           | LCR         | Line Control Register                         |
| 0x01C4 2010           | 0x01D0 C010           | 0x01D0 D010           | MCR         | Modem Control Register                        |
| 0x01C4 2014           | 0x01D0 C014           | 0x01D0 D014           | LSR         | Line Status Register                          |
| 0x01C4 2018           | 0x01D0 C018           | 0x01D0 D018           | MSR         | Modem Status Register                         |
| 0x01C4 201C           | 0x01D0 C01C           | 0x01D0 D01C           | SCR         | Scratchpad Register                           |
| 0x01C4 2020           | 0x01D0 C020           | 0x01D0 D020           | DLL         | Divisor LSB Latch                             |
| 0x01C4 2024           | 0x01D0 C024           | 0x01D0 D024           | DLH         | Divisor MSB Latch                             |
| 0x01C4 2028           | 0x01D0 C028           | 0x01D0 D028           | REVID1      | Revision Identification Register 1            |
| 0x01C4 2030           | 0x01D0 C030           | 0x01D0 D030           | PWREMU_MGMT | Power and Emulation Management Register       |
| 0x01C4 2034           | 0x01D0 C034           | 0x01D0 D034           | MDR         | Mode Definition Register                      |

## 6.23.2 UART Electrical Data/Timing

**Table 6-83. Timing Requirements for UARTx Receive<sup>(1)</sup> (see Figure 6-47)**

| No. | PARAMETER                                                   | MIN   | MAX   | UNIT |
|-----|-------------------------------------------------------------|-------|-------|------|
| 4   | $t_w(\text{URXDB})$ Pulse duration, receive data bit (RXDn) | 0.96U | 1.05U | ns   |
| 5   | $t_w(\text{URXSB})$ Pulse duration, receive start bit       | 0.96U | 1.05U | ns   |

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

**Table 6-84. Switching Characteristics Over Recommended Operating Conditions for UARTx Transmit<sup>(1)</sup> (see Figure 6-47)**

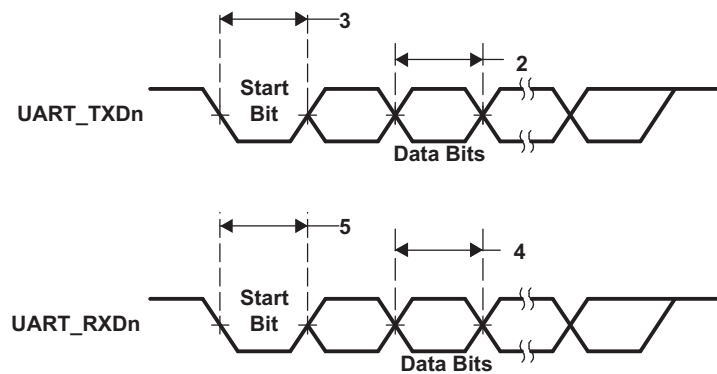
| No. | PARAMETER                                                    | MIN   | MAX                    | UNIT                 |
|-----|--------------------------------------------------------------|-------|------------------------|----------------------|
| 1   | $f_{(\text{baud})}$ Maximum programmable baud rate           |       | D/E <sup>(2) (3)</sup> | MBaud <sup>(4)</sup> |
| 2   | $t_w(\text{UTXDB})$ Pulse duration, transmit data bit (TXDn) | U - 2 | U + 2                  | ns                   |
| 3   | $t_w(\text{UTXSB})$ Pulse duration, transmit start bit       | U - 2 | U + 2                  | ns                   |

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

(2) D = UART input clock in MHz. The UART(s) input clock source is PLL0\_SYSCLK2.

(3) E = UART divisor x UART sampling rate. The UART divisor is set through the UART divisor latch registers (DLL and DLH). The UART sampling rate is set through the over-sampling mode select bit (OSM\_SEL) of the UART mode definition register (MDR).

(4) Baud rate is not indicative of data rate. Actual data rate will be limited by system factors such as EDMA loading, EMIF loading, system frequency, etc.



**Figure 6-47. UART Transmit/Receive Timing**

## 6.24 USB0 OTG (USB2.0 OTG)

The device USB2.0 peripheral supports the following features:

- USB 2.0 peripheral at full-speed (FS: 12 Mb/s)
- USB 2.0 host at speeds FS, and low speed (LS: 1.5 Mb/s)
- All transfer modes (control, bulk, interrupt, and isochronous)
- 4 Transmit (TX) and 4 Receive (RX) endpoints in addition to endpoint 0
- FIFO RAM
  - 4K endpoint
  - Programmable size
- Integrated USB 2.0 PHY
- Connects to a standard Charge Pump for VBUS 5 V generation
- RNDIS mode for accelerating RNDIS type protocols using short packet termination over USB

**Important Notice:** On the original device pinout (marked "A" in the lower right corner of the package), pins USB0\_VSSA33 (H4) and USB0\_VSSA (F3) were connected to ground outside the package. For more robust ESD performance, the USB0 ground references are now connected inside the package on packages marked "B" and the package pins are unconnected. This change will require that any external filter circuits previously referenced to ground at these pins will need to reference the board ground instead.

**Important Notice:** The USB0 controller module clock (PLL0\_SYSClk2) must be greater than 30 MHz for proper operation of the USB controller. A clock rate of 60 MHz or greater is recommended to avoid data throughput reduction.

Table 6-85 is the list of USB OTG registers.

**Table 6-85. Universal Serial Bus OTG (USB0) Registers**

| BYTE ADDRESS | ACRONYM     | REGISTER DESCRIPTION                                             |
|--------------|-------------|------------------------------------------------------------------|
| 0x01E0 0000  | REVID       | Revision Register                                                |
| 0x01E0 0004  | CTRLR       | Control Register                                                 |
| 0x01E0 0008  | STATR       | Status Register                                                  |
| 0x01E0 000C  | EMUR        | Emulation Register                                               |
| 0x01E0 0010  | MODE        | Mode Register                                                    |
| 0x01E0 0014  | AUTOREQ     | Autorequest Register                                             |
| 0x01E0 0018  | SRPFIXTIME  | SRP Fix Time Register                                            |
| 0x01E0 001C  | TEARDOWN    | Teardown Register                                                |
| 0x01E0 0020  | INTSRCR     | USB Interrupt Source Register                                    |
| 0x01E0 0024  | INTSETR     | USB Interrupt Source Set Register                                |
| 0x01E0 0028  | INTCLRR     | USB Interrupt Source Clear Register                              |
| 0x01E0 002C  | INTMSKR     | USB Interrupt Mask Register                                      |
| 0x01E0 0030  | INTMSKSETR  | USB Interrupt Mask Set Register                                  |
| 0x01E0 0034  | INTMSKCLRR  | USB Interrupt Mask Clear Register                                |
| 0x01E0 0038  | INTMASKEDR  | USB Interrupt Source Masked Register                             |
| 0x01E0 003C  | EOIR        | USB End of Interrupt Register                                    |
| 0x01E0 0040  | -           | Reserved                                                         |
| 0x01E0 0050  | GENRNDISSZ1 | Generic RNDIS Size EP1                                           |
| 0x01E0 0054  | GENRNDISSZ2 | Generic RNDIS Size EP2                                           |
| 0x01E0 0058  | GENRNDISSZ3 | Generic RNDIS Size EP3                                           |
| 0x01E0 005C  | GENRNDISSZ4 | Generic RNDIS Size EP4                                           |
| 0x01E0 0400  | FADDR       | Function Address Register                                        |
| 0x01E0 0401  | POWER       | Power Management Register                                        |
| 0x01E0 0402  | INTRTX      | Interrupt Register for Endpoint 0 plus Transmit Endpoints 1 to 4 |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                                                           | ACRONYM         | REGISTER DESCRIPTION                                                                                                                                                                     |
|------------------------------------------------------------------------|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x01E0 0404                                                            | INTRRX          | Interrupt Register for Receive Endpoints 1 to 4                                                                                                                                          |
| 0x01E0 0406                                                            | INTRTXE         | Interrupt Enable Register for INTRTX                                                                                                                                                     |
| 0x01E0 0408                                                            | INTRRXE         | Interrupt Enable Register for INTRRX                                                                                                                                                     |
| 0x01E0 040A                                                            | INTRUSB         | Interrupt Register for Common USB Interrupts                                                                                                                                             |
| 0x01E0 040B                                                            | INTRUSBE        | Interrupt Enable Register for INTRUSB                                                                                                                                                    |
| 0x01E0 040C                                                            | FRAME           | Frame Number Register                                                                                                                                                                    |
| 0x01E0 040E                                                            | INDEX           | Index Register for Selecting the Endpoint Status and Control Registers                                                                                                                   |
| 0x01E0 040F                                                            | TESTMODE        | Register to Enable the USB 2.0 Test Modes                                                                                                                                                |
| <b>INDEXED REGISTERS</b>                                               |                 |                                                                                                                                                                                          |
| These registers operate on the endpoint selected by the INDEX register |                 |                                                                                                                                                                                          |
| 0x01E0 0410                                                            | TXMAXP          | Maximum Packet Size for Peripheral/Host Transmit Endpoint (Index register set to select Endpoints 1-4 only)                                                                              |
| 0x01E0 0412                                                            | PERI_CSR0       | Control Status Register for Endpoint 0 in Peripheral Mode. (Index register set to select Endpoint 0)                                                                                     |
|                                                                        | HOST_CSR0       | Control Status Register for Endpoint 0 in Host Mode. (Index register set to select Endpoint 0)                                                                                           |
|                                                                        | PERI_TXCSR      | Control Status Register for Peripheral Transmit Endpoint. (Index register set to select Endpoints 1-4)                                                                                   |
|                                                                        | HOST_TXCSR      | Control Status Register for Host Transmit Endpoint. (Index register set to select Endpoints 1-4)                                                                                         |
| 0x01E0 0414                                                            | RXMAXP          | Maximum Packet Size for Peripheral/Host Receive Endpoint (Index register set to select Endpoints 1-4 only)                                                                               |
| 0x01E0 0416                                                            | PERI_RXCSR      | Control Status Register for Peripheral Receive Endpoint. (Index register set to select Endpoints 1-4)                                                                                    |
|                                                                        | HOST_RXCSR      | Control Status Register for Host Receive Endpoint. (Index register set to select Endpoints 1-4)                                                                                          |
| 0x01E0 0418                                                            | COUNT0          | Number of Received Bytes in Endpoint 0 FIFO. (Index register set to select Endpoint 0)                                                                                                   |
|                                                                        | RXCOUNT         | Number of Bytes in Host Receive Endpoint FIFO. (Index register set to select Endpoints 1- 4)                                                                                             |
| 0x01E0 041A                                                            | HOST_TYPE0      | Defines the speed of Endpoint 0                                                                                                                                                          |
|                                                                        | HOST_TXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Transmit endpoint. (Index register set to select Endpoints 1-4 only)                          |
| 0x01E0 041B                                                            | HOST_NAKLIMIT0  | Sets the NAK response timeout on Endpoint 0. (Index register set to select Endpoint 0)                                                                                                   |
|                                                                        | HOST_TXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Transmit endpoint. (Index register set to select Endpoints 1-4 only) |
| 0x01E0 041C                                                            | HOST_RXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Receive endpoint. (Index register set to select Endpoints 1-4 only)                           |
| 0x01E0 041D                                                            | HOST_RXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Receive endpoint. (Index register set to select Endpoints 1-4 only)  |
| 0x01E0 041F                                                            | CONFIGDATA      | Returns details of core configuration. (Index register set to select Endpoint 0)                                                                                                         |
| <b>FIFO</b>                                                            |                 |                                                                                                                                                                                          |
| 0x01E0 0420                                                            | FIFO0           | Transmit and Receive FIFO Register for Endpoint 0                                                                                                                                        |
| 0x01E0 0424                                                            | FIFO1           | Transmit and Receive FIFO Register for Endpoint 1                                                                                                                                        |
| 0x01E0 0428                                                            | FIFO2           | Transmit and Receive FIFO Register for Endpoint 2                                                                                                                                        |
| 0x01E0 042C                                                            | FIFO3           | Transmit and Receive FIFO Register for Endpoint 3                                                                                                                                        |
| 0x01E0 0430                                                            | FIFO4           | Transmit and Receive FIFO Register for Endpoint 4                                                                                                                                        |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                                                        | ACRONYM    | REGISTER DESCRIPTION                                                                                                                                                                 |
|---------------------------------------------------------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>OTG DEVICE CONTROL</b>                                           |            |                                                                                                                                                                                      |
| 0x01E0 0460                                                         | DEVCTL     | Device Control Register                                                                                                                                                              |
| <b>DYNAMIC FIFO CONTROL</b>                                         |            |                                                                                                                                                                                      |
| 0x01E0 0462                                                         | TXFIFOSZ   | Transmit Endpoint FIFO Size<br>(Index register set to select Endpoints 1-4 only)                                                                                                     |
| 0x01E0 0463                                                         | RXFIFOSZ   | Receive Endpoint FIFO Size<br>(Index register set to select Endpoints 1-4 only)                                                                                                      |
| 0x01E0 0464                                                         | TXFIFOADDR | Transmit Endpoint FIFO Address<br>(Index register set to select Endpoints 1-4 only)                                                                                                  |
| 0x01E0 0466                                                         | RXFIFOADDR | Receive Endpoint FIFO Address<br>(Index register set to select Endpoints 1-4 only)                                                                                                   |
| 0x01E0 046C                                                         | HWVERS     | Hardware Version Register                                                                                                                                                            |
| <b>TARGET ENDPOINT 0 CONTROL REGISTERS, VALID ONLY IN HOST MODE</b> |            |                                                                                                                                                                                      |
| 0x01E0 0480                                                         | TXFUNCADDR | Address of the target function that has to be accessed through the associated Transmit Endpoint.                                                                                     |
| 0x01E0 0482                                                         | TXHUBADDR  | Address of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub. |
| 0x01E0 0483                                                         | TXHUBPORT  | Port of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.    |
| 0x01E0 0484                                                         | RXFUNCADDR | Address of the target function that has to be accessed through the associated Receive Endpoint.                                                                                      |
| 0x01E0 0486                                                         | RXHUBADDR  | Address of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.  |
| 0x01E0 0487                                                         | RXHUBPORT  | Port of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.     |
| <b>TARGET ENDPOINT 1 CONTROL REGISTERS, VALID ONLY IN HOST MODE</b> |            |                                                                                                                                                                                      |
| 0x01E0 0488                                                         | TXFUNCADDR | Address of the target function that has to be accessed through the associated Transmit Endpoint.                                                                                     |
| 0x01E0 048A                                                         | TXHUBADDR  | Address of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub. |
| 0x01E0 048B                                                         | TXHUBPORT  | Port of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.    |
| 0x01E0 048C                                                         | RXFUNCADDR | Address of the target function that has to be accessed through the associated Receive Endpoint.                                                                                      |
| 0x01E0 048E                                                         | RXHUBADDR  | Address of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.  |
| 0x01E0 048F                                                         | RXHUBPORT  | Port of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.     |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                                                        | ACRONYM    | REGISTER DESCRIPTION                                                                                                                                                                 |
|---------------------------------------------------------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>TARGET ENDPOINT 2 CONTROL REGISTERS, VALID ONLY IN HOST MODE</b> |            |                                                                                                                                                                                      |
| 0x01E0 0490                                                         | TXFUNCADDR | Address of the target function that has to be accessed through the associated Transmit Endpoint.                                                                                     |
| 0x01E0 0492                                                         | TXHUBADDR  | Address of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub. |
| 0x01E0 0493                                                         | TXHUBPORT  | Port of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.    |
| 0x01E0 0494                                                         | RXFUNCADDR | Address of the target function that has to be accessed through the associated Receive Endpoint.                                                                                      |
| 0x01E0 0496                                                         | RXHUBADDR  | Address of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.  |
| 0x01E0 0497                                                         | RXHUBPORT  | Port of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.     |
| <b>TARGET ENDPOINT 3 CONTROL REGISTERS, VALID ONLY IN HOST MODE</b> |            |                                                                                                                                                                                      |
| 0x01E0 0498                                                         | TXFUNCADDR | Address of the target function that has to be accessed through the associated Transmit Endpoint.                                                                                     |
| 0x01E0 049A                                                         | TXHUBADDR  | Address of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub. |
| 0x01E0 049B                                                         | TXHUBPORT  | Port of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.    |
| 0x01E0 049C                                                         | RXFUNCADDR | Address of the target function that has to be accessed through the associated Receive Endpoint.                                                                                      |
| 0x01E0 049E                                                         | RXHUBADDR  | Address of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.  |
| 0x01E0 049F                                                         | RXHUBPORT  | Port of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.     |
| <b>TARGET ENDPOINT 4 CONTROL REGISTERS, VALID ONLY IN HOST MODE</b> |            |                                                                                                                                                                                      |
| 0x01E0 04A0                                                         | TXFUNCADDR | Address of the target function that has to be accessed through the associated Transmit Endpoint.                                                                                     |
| 0x01E0 04A2                                                         | TXHUBADDR  | Address of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub. |
| 0x01E0 04A3                                                         | TXHUBPORT  | Port of the hub that has to be accessed through the associated Transmit Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.    |
| 0x01E0 04A4                                                         | RXFUNCADDR | Address of the target function that has to be accessed through the associated Receive Endpoint.                                                                                      |
| 0x01E0 04A6                                                         | RXHUBADDR  | Address of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.  |
| 0x01E0 04A7                                                         | RXHUBPORT  | Port of the hub that has to be accessed through the associated Receive Endpoint. This is used only when full speed or low speed device is connected via a USB2.0 high-speed hub.     |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                                      | ACRONYM         | REGISTER DESCRIPTION                                                                                                                   |
|---------------------------------------------------|-----------------|----------------------------------------------------------------------------------------------------------------------------------------|
| <b>CONTROL AND STATUS REGISTER FOR ENDPOINT 0</b> |                 |                                                                                                                                        |
| 0x01E0 0502                                       | PERI_CSR0       | Control Status Register for Endpoint 0 in Peripheral Mode                                                                              |
|                                                   | HOST_CSR0       | Control Status Register for Endpoint 0 in Host Mode                                                                                    |
| 0x01E0 0508                                       | COUNT0          | Number of Received Bytes in Endpoint 0 FIFO                                                                                            |
| 0x01E0 050A                                       | HOST_TYPE0      | Defines the Speed of Endpoint 0                                                                                                        |
| 0x01E0 050B                                       | HOST_NAKLIMIT0  | Sets the NAK Response Timeout on Endpoint 0                                                                                            |
| 0x01E0 050F                                       | CONFIGDATA      | Returns details of core configuration.                                                                                                 |
| <b>CONTROL AND STATUS REGISTER FOR ENDPOINT 1</b> |                 |                                                                                                                                        |
| 0x01E0 0510                                       | TXMAXP          | Maximum Packet Size for Peripheral/Host Transmit Endpoint                                                                              |
| 0x01E0 0512                                       | PERI_TXCSR      | Control Status Register for Peripheral Transmit Endpoint (peripheral mode)                                                             |
|                                                   | HOST_TXCSR      | Control Status Register for Host Transmit Endpoint (host mode)                                                                         |
| 0x01E0 0514                                       | RXMAXP          | Maximum Packet Size for Peripheral/Host Receive Endpoint                                                                               |
| 0x01E0 0516                                       | PERI_RXCSR      | Control Status Register for Peripheral Receive Endpoint (peripheral mode)                                                              |
|                                                   | HOST_RXCSR      | Control Status Register for Host Receive Endpoint (host mode)                                                                          |
| 0x01E0 0518                                       | RXCOUNT         | Number of Bytes in Host Receive endpoint FIFO                                                                                          |
| 0x01E0 051A                                       | HOST_TXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Transmit endpoint.                          |
| 0x01E0 051B                                       | HOST_TXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Transmit endpoint. |
| 0x01E0 051C                                       | HOST_RXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Receive endpoint.                           |
| 0x01E0 051D                                       | HOST_RXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Receive endpoint.  |
| <b>CONTROL AND STATUS REGISTER FOR ENDPOINT 2</b> |                 |                                                                                                                                        |
| 0x01E0 0520                                       | TXMAXP          | Maximum Packet Size for Peripheral/Host Transmit Endpoint                                                                              |
| 0x01E0 0522                                       | PERI_TXCSR      | Control Status Register for Peripheral Transmit Endpoint (peripheral mode)                                                             |
|                                                   | HOST_TXCSR      | Control Status Register for Host Transmit Endpoint (host mode)                                                                         |
| 0x01E0 0524                                       | RXMAXP          | Maximum Packet Size for Peripheral/Host Receive Endpoint                                                                               |
| 0x01E0 0526                                       | PERI_RXCSR      | Control Status Register for Peripheral Receive Endpoint (peripheral mode)                                                              |
|                                                   | HOST_RXCSR      | Control Status Register for Host Receive Endpoint (host mode)                                                                          |
| 0x01E0 0528                                       | RXCOUNT         | Number of Bytes in Host Receive endpoint FIFO                                                                                          |
| 0x01E0 052A                                       | HOST_TXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Transmit endpoint.                          |
| 0x01E0 052B                                       | HOST_TXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Transmit endpoint. |
| 0x01E0 052C                                       | HOST_RXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Receive endpoint.                           |
| 0x01E0 052D                                       | HOST_RXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Receive endpoint.  |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                                      | ACRONYM         | REGISTER DESCRIPTION                                                                                                                   |
|---------------------------------------------------|-----------------|----------------------------------------------------------------------------------------------------------------------------------------|
| <b>CONTROL AND STATUS REGISTER FOR ENDPOINT 3</b> |                 |                                                                                                                                        |
| 0x01E0 0530                                       | TXMAXP          | Maximum Packet Size for Peripheral/Host Transmit Endpoint                                                                              |
| 0x01E0 0532                                       | PERI_TXCSR      | Control Status Register for Peripheral Transmit Endpoint (peripheral mode)                                                             |
|                                                   | HOST_TXCSR      | Control Status Register for Host Transmit Endpoint (host mode)                                                                         |
| 0x01E0 0534                                       | RXMAXP          | Maximum Packet Size for Peripheral/Host Receive Endpoint                                                                               |
| 0x01E0 0536                                       | PERI_RXCSR      | Control Status Register for Peripheral Receive Endpoint (peripheral mode)                                                              |
|                                                   | HOST_RXCSR      | Control Status Register for Host Receive Endpoint (host mode)                                                                          |
| 0x01E0 0538                                       | RXCOUNT         | Number of Bytes in Host Receive endpoint FIFO                                                                                          |
| 0x01E0 053A                                       | HOST_TXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Transmit endpoint.                          |
| 0x01E0 053B                                       | HOST_TXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Transmit endpoint. |
| 0x01E0 053C                                       | HOST_RXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Receive endpoint.                           |
| 0x01E0 053D                                       | HOST_RXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Receive endpoint.  |
| <b>CONTROL AND STATUS REGISTER FOR ENDPOINT 4</b> |                 |                                                                                                                                        |
| 0x01E0 0540                                       | TXMAXP          | Maximum Packet Size for Peripheral/Host Transmit Endpoint                                                                              |
| 0x01E0 0542                                       | PERI_TXCSR      | Control Status Register for Peripheral Transmit Endpoint (peripheral mode)                                                             |
|                                                   | HOST_TXCSR      | Control Status Register for Host Transmit Endpoint (host mode)                                                                         |
| 0x01E0 0544                                       | RXMAXP          | Maximum Packet Size for Peripheral/Host Receive Endpoint                                                                               |
| 0x01E0 0546                                       | PERI_RXCSR      | Control Status Register for Peripheral Receive Endpoint (peripheral mode)                                                              |
|                                                   | HOST_RXCSR      | Control Status Register for Host Receive Endpoint (host mode)                                                                          |
| 0x01E0 0548                                       | RXCOUNT         | Number of Bytes in Host Receive endpoint FIFO                                                                                          |
| 0x01E0 054A                                       | HOST_TXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Transmit endpoint.                          |
| 0x01E0 054B                                       | HOST_TXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Transmit endpoint. |
| 0x01E0 054C                                       | HOST_RXTYPE     | Sets the operating speed, transaction protocol and peripheral endpoint number for the host Receive endpoint.                           |
| 0x01E0 054D                                       | HOST_RXINTERVAL | Sets the polling interval for Interrupt/ISOC transactions or the NAK response timeout on Bulk transactions for host Receive endpoint.  |
| <b>DMA REGISTERS</b>                              |                 |                                                                                                                                        |
| 0x01E0 1000                                       | DMAREVID        | DMA Revision Register                                                                                                                  |
| 0x01E0 1004                                       | TDFDQ           | DMA Teardown Free Descriptor Queue Control Register                                                                                    |
| 0x01E0 1008                                       | DMAEMU          | DMA Emulation Control Register                                                                                                         |
| 0x01E0 1800                                       | TXGCR[0]        | Transmit Channel 0 Global Configuration Register                                                                                       |
| 0x01E0 1808                                       | RXGCR[0]        | Receive Channel 0 Global Configuration Register                                                                                        |
| 0x01E0 180C                                       | RXHPCRA[0]      | Receive Channel 0 Host Packet Configuration Register A                                                                                 |
| 0x01E0 1810                                       | RXHPCRB[0]      | Receive Channel 0 Host Packet Configuration Register B                                                                                 |
| 0x01E0 1820                                       | TXGCR[1]        | Transmit Channel 1 Global Configuration Register                                                                                       |
| 0x01E0 1828                                       | RXGCR[1]        | Receive Channel 1 Global Configuration Register                                                                                        |
| 0x01E0 182C                                       | RXHPCRA[1]      | Receive Channel 1 Host Packet Configuration Register A                                                                                 |
| 0x01E0 1830                                       | RXHPCRB[1]      | Receive Channel 1 Host Packet Configuration Register B                                                                                 |

**Table 6-85. Universal Serial Bus OTG (USB0) Registers (continued)**

| BYTE ADDRESS                   | ACRONYM        | REGISTER DESCRIPTION                                   |
|--------------------------------|----------------|--------------------------------------------------------|
| 0x01E0 1840                    | TXGCR[2]       | Transmit Channel 2 Global Configuration Register       |
| 0x01E0 1848                    | RXGCR[2]       | Receive Channel 2 Global Configuration Register        |
| 0x01E0 184C                    | RXHPCRA[2]     | Receive Channel 2 Host Packet Configuration Register A |
| 0x01E0 1850                    | RXHPCRB[2]     | Receive Channel 2 Host Packet Configuration Register B |
| 0x01E0 1860                    | TXGCR[3]       | Transmit Channel 3 Global Configuration Register       |
| 0x01E0 1868                    | RXGCR[3]       | Receive Channel 3 Global Configuration Register        |
| 0x01E0 186C                    | RXHPCRA[3]     | Receive Channel 3 Host Packet Configuration Register A |
| 0x01E0 1870                    | RXHPCRB[3]     | Receive Channel 3 Host Packet Configuration Register B |
| 0x01E0 2000                    | DMA_SCHED_CTRL | DMA Scheduler Control Register                         |
| 0x01E0 2800                    | WORD[0]        | DMA Scheduler Table Word 0                             |
| 0x01E0 2804                    | WORD[1]        | DMA Scheduler Table Word 1                             |
| ...                            | ...            | ...                                                    |
| 0x01E0 28FC                    | WORD[63]       | DMA Scheduler Table Word 63                            |
| <b>QUEUE MANAGER REGISTERS</b> |                |                                                        |
| 0x01E0 4000                    | QMGRREVID      | Queue Manager Revision Register                        |
| 0x01E0 4008                    | DIVERSION      | Queue Diversion Register                               |
| 0x01E0 4020                    | FDBSC0         | Free Descriptor/Buffer Starvation Count Register 0     |
| 0x01E0 4024                    | FDBSC1         | Free Descriptor/Buffer Starvation Count Register 1     |
| 0x01E0 4028                    | FDBSC2         | Free Descriptor/Buffer Starvation Count Register 2     |
| 0x01E0 402C                    | FDBSC3         | Free Descriptor/Buffer Starvation Count Register 3     |
| 0x01E0 4080                    | LRAM0BASE      | Linking RAM Region 0 Base Address Register             |
| 0x01E0 4084                    | LRAM0SIZE      | Linking RAM Region 0 Size Register                     |
| 0x01E0 4088                    | LRAM1BASE      | Linking RAM Region 1 Base Address Register             |
| 0x01E0 4090                    | PEND0          | Queue Pending Register 0                               |
| 0x01E0 4094                    | PEND1          | Queue Pending Register 1                               |
| 0x01E0 5000                    | QMEMRBASE[0]   | Memory Region 0 Base Address Register                  |
| 0x01E0 5004                    | QMEMRCTRL[0]   | Memory Region 0 Control Register                       |
| 0x01E0 5010                    | QMEMRBASE[1]   | Memory Region 1 Base Address Register                  |
| 0x01E0 5014                    | QMEMRCTRL[1]   | Memory Region 1 Control Register                       |
| ...                            | ...            | ...                                                    |
| 0x01E0 50F0                    | QMEMRBASE[15]  | Memory Region 15 Base Address Register                 |
| 0x01E0 50F4                    | QMEMRCTRL[15]  | Memory Region 15 Control Register                      |
| 0x01E0 600C                    | CTRLD[0]       | Queue Manager Queue 0 Control Register D               |
| 0x01E0 601C                    | CTRLD[1]       | Queue Manager Queue 1 Control Register D               |
| ...                            | ...            | ...                                                    |
| 0x01E0 63FC                    | CTRLD[63]      | Queue Manager Queue 63 Status Register D               |
| 0x01E0 6800                    | QSTATA[0]      | Queue Manager Queue 0 Status Register A                |
| 0x01E0 6804                    | QSTATB[0]      | Queue Manager Queue 0 Status Register B                |
| 0x01E0 6808                    | QSTATC[0]      | Queue Manager Queue 0 Status Register C                |
| 0x01E0 6810                    | QSTATA[1]      | Queue Manager Queue 1 Status Register A                |
| 0x01E0 6814                    | QSTATB[1]      | Queue Manager Queue 1 Status Register B                |
| 0x01E0 6818                    | QSTATC[1]      | Queue Manager Queue 1 Status Register C                |
| ...                            | ...            | ...                                                    |
| 0x01E0 6BF0                    | QSTATA[63]     | Queue Manager Queue 63 Status Register A               |
| 0x01E0 6BF4                    | QSTATB[63]     | Queue Manager Queue 63 Status Register B               |
| 0x01E0 6BF8                    | QSTATC[63]     | Queue Manager Queue 63 Status Register C               |

### 6.24.1 USB2.0 (USB0) Electrical Data/Timing

The USB PHY PLL can support input clock of the following frequencies: 12.0 MHz, 13.0 MHz, 19.2 MHz, 20.0 MHz, 24.0 MHz, 26.0 MHz, 38.4 MHz, 40.0 MHz or 48.0 MHz. USB\_REFCLKIN jitter tolerance is 50 ppm maximum.

**Table 6-86. Switching Characteristics Over Recommended Operating Conditions for USB2.0 [USB0] (see Figure 6-48)**

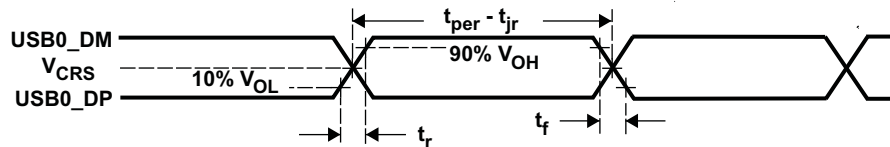
| No. | PARAMETER          |                                                               | LOW SPEED<br>1.5 Mbps |      | FULL SPEED<br>12 Mbps |      | UNIT     |
|-----|--------------------|---------------------------------------------------------------|-----------------------|------|-----------------------|------|----------|
|     |                    |                                                               | MIN                   | MAX  | MIN                   | MAX  |          |
| 1   | $t_{r(D)}$         | Rise time, USB0_DP and USB0_DM signals <sup>(1)</sup>         | 75                    | 300  | 4                     | 20   | ns       |
| 2   | $t_{f(D)}$         | Fall time, USB0_DP and USB0_DM signals <sup>(1)</sup>         | 75                    | 300  | 4                     | 20   | ns       |
| 3   | $t_{rFM}$          | Rise/Fall time, matching <sup>(2)</sup>                       | 80                    | 120  | 90                    | 111  | %        |
| 4   | $V_{CRS}$          | Output signal cross-over voltage <sup>(1)</sup>               | 1.3                   | 2    | 1.3                   | 2    | V        |
| 5   | $t_{jr(source)NT}$ | Source (Host) Driver jitter, next transition                  |                       | 2    |                       | 2    | ns       |
|     | $t_{jr(FUNC)NT}$   | Function Driver jitter, next transition                       |                       | 25   |                       | 2    | ns       |
| 6   | $t_{jr(source)PT}$ | Source (Host) Driver jitter, paired transition <sup>(3)</sup> |                       | 1    |                       | 1    | ns       |
|     | $t_{jr(FUNC)PT}$   | Function Driver jitter, paired transition                     |                       | 10   |                       | 1    | ns       |
| 7   | $t_{w(EOPT)}$      | Pulse duration, EOP transmitter <sup>(4)</sup>                | 1250                  | 1500 | 160                   | 175  | ns       |
| 8   | $t_{w(EOPR)}$      | Pulse duration, EOP receiver <sup>(4)</sup>                   | 670                   |      | 82                    |      | ns       |
| 9   | $t_{(DRATE)}$      | Data Rate                                                     |                       | 1.5  |                       | 12   | Mb/s     |
| 10  | $Z_{DRV}$          | Driver Output Resistance                                      | –                     | –    | 40.5                  | 49.5 | $\Omega$ |
| 11  | $Z_{INP}$          | Receiver Input Impedance                                      | 100k                  |      | 100k                  |      | $\Omega$ |

(1) Low Speed:  $C_L = 200$  pF, Full Speed:  $C_L = 50$  pF

(2)  $t_{RFM} = (t_r/t_f) \times 100$ . [Excluding the first transaction from the Idle state.]

(3)  $t_{jr} = t_{px(1)} - t_{px(0)}$

(4) Must accept as valid EOP



**Figure 6-48. USB0 Integrated Transceiver Interface Timing**

### 6.24.2 USB0 Unused Signal Configuration

If USB0 is unused, then the USB0 signals should be configured as shown in [Section 3.6.19](#).

## 6.25 Power and Sleep Controller (PSC)

The Power and Sleep Controllers (PSC) are responsible for managing transitions of system power on/off, clock on/off, resets (device level and module level). It is used primarily to provide granular power control for on chip modules (peripherals and CPU). A PSC module consists of a Global PSC (GPSC) and a set of Local PSCs (LPSCs). The GPSC contains memory mapped registers, PSC interrupts, a state machine for each peripheral/module it controls. An LPSC is associated with every module that is controlled by the PSC and provides clock and reset control.

The PSC includes the following features:

- Provides a software interface to:
  - Control module clock enable/disable
  - Control module reset
  - Control CPU local reset
- Supports ICEPick TAP Router power, clock and reset features. For details on ICEPick features see <http://tiexpressdsp.com/wiki/index.php?title=ICEPick>.

**Table 6-87. Power and Sleep Controller (PSC) Registers**

| PSC0<br>BYTE ADDRESS        | PSC1<br>BYTE ADDRESS        | ACRONYM          | DESCRIPTION                                                                                                  |
|-----------------------------|-----------------------------|------------------|--------------------------------------------------------------------------------------------------------------|
| 0x01C1 0000                 | 0x01E2 7000                 | REVID            | Peripheral Revision and Class Information Register                                                           |
| 0x01C1 0018                 | 0x01E2 7018                 | INTEVAL          | Interrupt Evaluation Register                                                                                |
| 0x01C1 0040                 | 0x01E2 7040                 | MERRPR0          | Module Error Pending Register 0 (module 0-15) (PSC0)<br>Module Error Pending Register 0 (module 0-31) (PSC1) |
| 0x01C1 0050                 | 0x01E2 7050                 | MERRCR0          | Module Error Clear Register 0 (module 0-15) (PSC0)<br>Module Error Clear Register 0 (module 0-31) (PSC1)     |
| 0x01C1 0060                 | 0x01E2 7060                 | PERRPR           | Power Error Pending Register                                                                                 |
| 0x01C1 0068                 | 0x01E2 7068                 | PERRCR           | Power Error Clear Register                                                                                   |
| 0x01C1 0120                 | 0x01E2 7120                 | PTCMD            | Power Domain Transition Command Register                                                                     |
| 0x01C1 0128                 | 0x01E2 7128                 | PTSTAT           | Power Domain Transition Status Register                                                                      |
| 0x01C1 0200                 | 0x01E2 7200                 | PDSTAT0          | Power Domain 0 Status Register                                                                               |
| 0x01C1 0204                 | 0x01E2 7204                 | PDSTAT1          | Power Domain 1 Status Register                                                                               |
| 0x01C1 0300                 | 0x01E2 7300                 | PDCTL0           | Power Domain 0 Control Register                                                                              |
| 0x01C1 0304                 | 0x01E2 7304                 | PDCTL1           | Power Domain 1 Control Register                                                                              |
| 0x01C1 0400                 | 0x01E2 7400                 | PDCFG0           | Power Domain 0 Configuration Register                                                                        |
| 0x01C1 0404                 | 0x01E2 7404                 | PDCFG1           | Power Domain 1 Configuration Register                                                                        |
| 0x01C1 0800-<br>0x01C1 083C | 0x01E2 7800-<br>0x01E2 787C | MDSTAT0-MDSTAT15 | Module Status <i>n</i> Register (modules 0-15) (PSC0)                                                        |
|                             |                             | MDSTAT0-MDSTAT31 | Module Status <i>n</i> Register (modules 0-31) (PSC1)                                                        |
| 0x01C1 0A00-<br>0x01C1 0A3C | 0x01E2 7A00-<br>0x01E2 7A7C | MDCTL0-MDCTL15   | Module Control <i>n</i> Register (modules 0-15) (PSC0)                                                       |
|                             |                             | MDCTL0-MDCTL31   | Module Control <i>n</i> Register (modules 0-31) (PSC1)                                                       |

### 6.25.1 Power Domain and Module Topology

The device includes two PSC modules.

Each PSC module controls clock states for several of the on chip modules, controllers and interconnect components. Table 6-88 and Table 6-89 lists the set of peripherals/modules that are controlled by the PSC, the power domain they are associated with, the LPSC assignment and the default (power-on reset) module states. See the device-specific data manual for the peripherals available on a given device. The module states and terminology are defined in Section 6.25.1.2.

**Table 6-88. PSC0 Default Module Configuration**

| LPSC Number | Module Name                   | Power Domain   | Default Module State | Auto Sleep/Wake Only |
|-------------|-------------------------------|----------------|----------------------|----------------------|
| 0           | EDMA3 Channel Controller      | AlwaysON (PD0) | SwRstDisable         | —                    |
| 1           | EDMA3 Transfer Controller 0   | AlwaysON (PD0) | SwRstDisable         | —                    |
| 2           | EDMA3 Transfer Controller 1   | AlwaysON (PD0) | SwRstDisable         | —                    |
| 3           | EMIFA (BR7)                   | AlwaysON (PD0) | SwRstDisable         | —                    |
| 4           | SPI 0                         | AlwaysON (PD0) | SwRstDisable         | —                    |
| 5           | MMC/SD 0                      | AlwaysON (PD0) | SwRstDisable         | —                    |
| 6           | ARM Interrupt Controller      | AlwaysON (PD0) | SwRstDisable         | —                    |
| 7           | ARM RAM/ROM                   | AlwaysON (PD0) | Enable               | Yes                  |
| 8           | -                             | -              | -                    | -                    |
| 9           | UART 0                        | AlwaysON (PD0) | SwRstDisable         | —                    |
| 10          | SCR0 (Br 0, Br 1, Br 2, Br 8) | AlwaysON (PD0) | Enable               | Yes                  |
| 11          | SCR1 (Br 4)                   | AlwaysON (PD0) | Enable               | Yes                  |
| 12          | SCR2 (Br 3, Br 5, Br 6)       | AlwaysON (PD0) | Enable               | Yes                  |
| 13          | PRUSS                         | AlwaysON (PD0) | SwRstDisable         | —                    |
| 14          | ARM                           | AlwaysON (PD0) | SwRstDisable         | —                    |
| 15          | -                             | -              | -                    | —                    |

**Table 6-89. PSC1 Default Module Configuration**

| LPSC Number | Module Name             | Power Domain   | Default Module State | Auto Sleep/Wake Only |
|-------------|-------------------------|----------------|----------------------|----------------------|
| 0           | Not Used                | —              | —                    | —                    |
| 1           | USB0 (USB2.0)           | AlwaysON (PD0) | SwRstDisable         | —                    |
| 2           | Not Used                | —              | —                    | —                    |
| 3           | GPIO                    | AlwaysON (PD0) | SwRstDisable         | —                    |
| 4           | Not Used                | —              | —                    | —                    |
| 5           | EMAC                    | AlwaysON (PD0) | SwRstDisable         | —                    |
| 6           | EMIFB (Br 20)           | AlwaysON (PD0) | SwRstDisable         | —                    |
| 7           | McASP0 ( + McASP0 FIFO) | AlwaysON (PD0) | SwRstDisable         | —                    |
| 8           | McASP1 ( + McASP1 FIFO) | AlwaysON (PD0) | SwRstDisable         | —                    |
| 9           | Not Used                | —              | —                    | —                    |
| 10          | SPI 1                   | AlwaysON (PD0) | SwRstDisable         | —                    |
| 11          | I2C 1                   | AlwaysON (PD0) | SwRstDisable         | —                    |
| 12          | UART 1                  | AlwaysON (PD0) | SwRstDisable         | —                    |
| 13          | UART 2                  | AlwaysON (PD0) | SwRstDisable         | —                    |
| 14-15       | Not Used                | —              | —                    | —                    |
| 16          | Not Used                | —              | —                    | —                    |
| 17          | eHRPWM0/1/2             | AlwaysON (PD0) | SwRstDisable         | —                    |
| 18-19       | Not Used                | —              | —                    | —                    |
| 20          | ECAP0/1/2               | AlwaysON (PD0) | SwRstDisable         | —                    |
| 21          | EQEP0/1                 | AlwaysON (PD0) | SwRstDisable         | —                    |
| 22-23       | Not Used                | —              | —                    | —                    |
| 24          | SCR8 (Br 15)            | AlwaysON (PD0) | Enable               | Yes                  |
| 25          | SCR7 (Br 12)            | AlwaysON (PD0) | Enable               | Yes                  |
| 26          | SCR12 (Br 18)           | AlwaysON (PD0) | Enable               | Yes                  |
| 27-30       | Not Used                | —              | —                    | —                    |
| 31          | On-chip RAM (Br 13)     | PD_SHRAM       | Enable               | Yes                  |

### 6.25.1.1 Power Domain States

A power domain can only be in one of the two states: ON or OFF, defined as follows:

- ON: power to the domain is on
- OFF: power to the domain is off

In the device, for both PSC0 and PSC1, the Always ON domain, or PD0 power domain, is always in the ON state when the chip is powered-on. This domain is not programmable to OFF state.

- On PSC1 PD1/PD\_SHRAM Domain: Controls the sleep state for the 128K Shared RAM

### 6.25.1.2 Module States

The PSC defines several possible states for a module. These states are essentially a combination of the module reset asserted or de-asserted and module clock on/enabled or off/disabled. The module states are defined in [Table 6-90](#).

**Table 6-90. Module States**

| Module State | Module Reset | Module Clock | Module State Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------|--------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Enable       | De-asserted  | On           | A module in the enable state has its module reset de-asserted and it has its clock on. This is the normal operational state for a given module                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Disable      | De-asserted  | Off          | A module in the disabled state has its module reset de-asserted and it has its module clock off. This state is typically used for disabling a module clock to save power. The device is designed in full static CMOS, so when you stop a module clock, it retains the module's state. When the clock is restarted, the module resumes operating from the stopping point.                                                                                                                                                                                                                                                                                                                                                                    |
| SyncReset    | Asserted     | On           | A module state in the SyncReset state has its module reset asserted and it has its clock on. Generally, software is not expected to initiate this state                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| SwRstDisable | Asserted     | Off          | A module in the SwResetDisable state has its module reset asserted and it has its clock disabled. After initial power-on, several modules come up in the SwRstDisable state. Generally, software is not expected to initiate this state                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Auto Sleep   | De-asserted  | Off          | A module in the Auto Sleep state also has its module reset de-asserted and its module clock disabled, similar to the Disable state. However this is a special state, once a module is configured in this state by software, it can "automatically" transition to "Enable" state whenever there is an internal read/write request made to it, and after servicing the request it will "automatically" transition into the sleep state (with module reset re de-asserted and module clock disabled), without any software intervention. The transition from sleep to enabled and back to sleep state has some cycle latency associated with it. It is not envisioned to use this mode when peripherals are fully operational and moving data. |
| Auto Wake    | De-asserted  | Off          | A module in the Auto Wake state also has its module reset de-asserted and its module clock disabled, similar to the Disable state. However this is a special state, once a module is configured in this state by software, it will "automatically" transition to "Enable" state whenever there is an internal read/write request made to it, and will remain in the "Enabled" state from then on (with module reset re de-asserted and module clock on), without any software intervention. The transition from sleep to enabled state has some cycle latency associated with it. It is not envisioned to use this mode when peripherals are fully operational and moving data.                                                             |

## 6.26 Programmable Real-Time Unit Subsystem (PRUSS)

The Programmable Real-Time Unit Subsystem (PRUSS) consists of

- Two Programmable Real-Time Units (PRU0 and PRU1) and their associated memories
- An Interrupt Controller (INTC) for handling system interrupt events. The INTC also supports posting events back to the device level host CPU.
- A Switched Central Resource (SCR) for connecting the various internal and external masters to the resources inside the PRUSS.

The two PRUs can operate completely independently or in coordination with each other. The PRUs can also work in coordination with the device level host CPU. This is determined by the nature of the program which is loaded into the PRUs instruction memory. Several different signaling mechanisms are available between the two PRUs and the device level host CPU.

The PRUs are optimized for performing embedded tasks that require manipulation of packed memory mapped data structures, handling of system events that have tight realtime constraints and interfacing with systems external to the device.

The PRUSS comprises various distinct addressable regions. Externally the subsystem presents a single 64Kbyte range of addresses. The internal interconnect bus (also called switched central resource, or SCR) of the PRUSS decodes accesses for each of the individual regions. The PRUSS memory map is documented in [Table 6-91](#) and in [Table 6-92](#). Note that these two memory maps are implemented inside the PRUSS and are local to the components of the PRUSS.

**Table 6-91. Programmable Real-Time Unit Subsystem (PRUSS) Local Instruction Space Memory Map**

| BYTE ADDRESS              | PRU0                 | PRU1                 |
|---------------------------|----------------------|----------------------|
| 0x0000 0000 - 0x0000 0FFF | PRU0 Instruction RAM | PRU1 Instruction RAM |

**Table 6-92. Programmable Real-Time Unit Subsystem (PRUSS) Local Data Space Memory Map**

| BYTE ADDRESS              | PRU0                      | PRU1                      |
|---------------------------|---------------------------|---------------------------|
| 0x0000 0000 - 0x0000 01FF | Data RAM 0 <sup>(1)</sup> | Data RAM 1 <sup>(1)</sup> |
| 0x0000 0200 - 0x0000 1FFF | Reserved                  | Reserved                  |
| 0x0000 2000 - 0x0000 21FF | Data RAM 1 <sup>(1)</sup> | Data RAM 0 <sup>(1)</sup> |
| 0x0000 2200 - 0x0000 3FFF | Reserved                  | Reserved                  |
| 0x0000 4000 - 0x0000 6FFF | INTC Registers            | INTC Registers            |
| 0x0000 7000 - 0x0000 73FF | PRU0 Control Registers    | PRU0 Control Registers    |
| 0x0000 7400 - 0x0000 77FF | Reserved                  | Reserved                  |
| 0x0000 7800 - 0x0000 7BFF | PRU1 Control Registers    | PRU1 Control Registers    |
| 0x0000 7C00 - 0xFFFF FFFF | Reserved                  | Reserved                  |

- (1) Note that PRU0 accesses Data RAM0 at address 0x0000 0000, also PRU1 accesses Data RAM1 at address 0x0000 0000. Data RAM0 is intended to be the primary data memory for PRU0 and Data RAM1 is intended to be the primary data memory for PRU1. However for passing information between PRUs, each PRU can access the data ram of the 'other' PRU through address 0x0000 2000.

The global view of the PRUSS internal memories and control ports is documented in [Table 6-93](#). The offset addresses of each region are implemented inside the PRUSS but the global device memory mapping places the PRUSS slave port in the address range 0x01C3 0000-0x01C3 FFFF. The PRU0 and PRU1 can use either the local or global addresses to access their internal memories, but using the local addresses will provide access time several cycles faster than using the global addresses. This is because when accessing via the global address the access needs to be routed through the switch fabric outside PRUSS and back in through the PRUSS slave port.

**Table 6-93. Programmable Real-Time Unit Subsystem (PRUSS) Global Memory Map**

| BYTE ADDRESS              | REGION                 |
|---------------------------|------------------------|
| 0x01C3 0000 - 0x01C3 01FF | Data RAM 0             |
| 0x01C3 0200 - 0x01C3 1FFF | Reserved               |
| 0x01C3 2000 - 0x01C3 21FF | Data RAM 1             |
| 0x01C3 2200 - 0x01C3 3FFF | Reserved               |
| 0x01C3 4000 - 0x01C3 6FFF | INTC Registers         |
| 0x01C3 7000 - 0x01C3 73FF | PRU0 Control Registers |
| 0x01C3 7400 - 0x01C3 77FF | PRU0 Debug Registers   |
| 0x01C3 7800 - 0x01C3 7BFF | PRU1 Control Registers |
| 0x01C3 7C00 - 0x01C3 7FFF | PRU1 Debug Registers   |
| 0x01C3 8000 - 0x01C3 8FFF | PRU0 Instruction RAM   |
| 0x01C3 9000 - 0x01C3 BFFF | Reserved               |
| 0x01C3 C000 - 0x01C3 CFFF | PRU1 Instruction RAM   |
| 0x01C3 D000 - 0x01C3 FFFF | Reserved               |

Each of the PRUs can access the rest of the device memory (including memory mapped peripheral and configuration registers) using the global memory space addresses.

### 6.26.1 PRUSS Register Descriptions

**Table 6-94. Programmable Real-Time Unit Subsystem (PRUSS) Control / Status Registers**

| PRU0 BYTE ADDRESS       | PRU1 BYTE ADDRESS         | ACRONYM              | REGISTER DESCRIPTION                               |
|-------------------------|---------------------------|----------------------|----------------------------------------------------|
| 0x01C3 7000             | 0x01C3 7800               | CONTROL              | PRU Control Register                               |
| 0x01C3 7004             | 0x01C3 7804               | STATUS               | PRU Status Register                                |
| 0x01C3 7008             | 0x01C3 7808               | WAKEUP               | PRU Wakeup Enable Register                         |
| 0x01C3 700C             | 0x01C3 780C               | CYCLCNT              | PRU Cycle Count                                    |
| 0x01C3 7010             | 0x01C3 7810               | STALLCNT             | PRU Stall Count                                    |
| 0x01C3 7020             | 0x01C3 7820               | CONTABBLKIDX0        | PRU Constant Table Block Index Register 0          |
| 0x01C3 7028             | 0x01C3 7828               | CONTABPROPTR0        | PRU Constant Table Programmable Pointer Register 0 |
| 0x01C3 702C             | 0x01C3 782C               | CONTABPROPTR1        | PRU Constant Table Programmable Pointer Register 1 |
| 0x01C37400 - 0x01C3747C | 0x01C3 7C00 - 0x01C3 7C7C | INTGPR0 – INTGPR31   | PRU Internal General Purpose Registers (for Debug) |
| 0x01C37480 - 0x01C374FC | 0x01C3 7C80 - 0x01C3 7CFC | INTCTER0 – INTCTER31 | PRU Internal Constants Table Registers (for Debug) |

**Table 6-95. Programmable Real-Time Unit Subsystem Interrupt Controller (PRUSS INTC) Registers**

| BYTE ADDRESS              | ACRONYM                            | REGISTER DESCRIPTION                             |
|---------------------------|------------------------------------|--------------------------------------------------|
| 0x01C3 4000               | REVID                              | Revision ID Register                             |
| 0x01C3 4004               | CONTROL                            | Control Register                                 |
| 0x01C3 4010               | GLBLEN                             | Global Enable Register                           |
| 0x01C3 401C               | GLBLNSTLVL                         | Global Nesting Level Register                    |
| 0x01C3 4020               | STATIDXSET                         | System Interrupt Status Indexed Set Register     |
| 0x01C3 4024               | STATIDXCLR                         | System Interrupt Status Indexed Clear Register   |
| 0x01C3 4028               | ENIDXSET                           | System Interrupt Enable Indexed Set Register     |
| 0x01C3 402C               | ENIDXCLR                           | System Interrupt Enable Indexed Clear Register   |
| 0x01C3 4034               | HSTINTENIDXSET                     | Host Interrupt Enable Indexed Set Register       |
| 0x01C3 4038               | HSTINTENIDXCLR                     | Host Interrupt Enable Indexed Clear Register     |
| 0x01C3 4080               | GLBLPRIIDX                         | Global Prioritized Index Register                |
| 0x01C3 4200               | STATSETINT0                        | System Interrupt Status Raw/Set Register 0       |
| 0x01C3 4204               | STATSETINT1                        | System Interrupt Status Raw/Set Register 1       |
| 0x01C3 4280               | STATCLRINT0                        | System Interrupt Status Enabled/Clear Register 0 |
| 0x01C3 4284               | STATCLRINT1                        | System Interrupt Status Enabled/Clear Register 1 |
| 0x01C3 4300               | ENABLESET0                         | System Interrupt Enable Set Register 0           |
| 0x01C3 4304               | ENABLESET1                         | System Interrupt Enable Set Register 1           |
| 0x01C3 4380               | ENABLECLR0                         | System Interrupt Enable Clear Register 0         |
| 0x01C3 4384               | ENABLECLR1                         | System Interrupt Enable Clear Register 1         |
| 0x01C3 4400 - 0x01C3 4440 | CHANMAP0 - CHANMAP15               | Channel Map Registers 0-15                       |
| 0x01C3 4800 - 0x01C3 4808 | HOSTMAP0 - HOSTMAP2                | Host Map Register 0-2                            |
| 0x01C3 4900 - 0x01C3 4928 | HOSTINTPRIIDX0 -<br>HOSTINTPRIIDX9 | Host Interrupt Prioritized Index Registers 0-9   |
| 0x01C3 4D00               | POLARITY0                          | System Interrupt Polarity Register 0             |
| 0x01C3 4D04               | POLARITY1                          | System Interrupt Polarity Register 1             |
| 0x01C3 4D80               | TYPE0                              | System Interrupt Type Register 0                 |
| 0x01C3 4D84               | TYPE1                              | System Interrupt Type Register 1                 |
| 0x01C3 5100 - 0x01C3 5128 | HOSTINTNSTLVL0-<br>HOSTINTNSTLVL9  | Host Interrupt Nesting Level Registers 0-9       |
| 0x01C3 5500               | HOSTINTEN                          | Host Interrupt Enable Register                   |

## 6.27 Emulation Logic

This section describes the steps to use a third party debugger. The debug capabilities and features for ARM are as shown below.

For TI's latest debug and emulation information see :

<http://tiexpressdsp.com/wiki/index.php?title=Category:Emulation>

### ARM:

- Basic Debug
  - Execution Control
  - System Visibility
- Advanced Debug
  - Global Start
  - Global Stop
- Advanced System Control
  - Subsystem reset via debug
  - Peripheral notification of debug events
  - Cache-coherent debug accesses
- Program Trace
  - Program flow corruption
  - Code coverage
  - Path coverage
  - Thread/interrupt synchronization problems
- Data Trace
  - Memory corruption
- Timing Trace
  - Profiling
- Analysis Actions
  - Stop program execution
  - Control trace streams
  - Generate debug interrupt
  - Benchmarking with counters
  - External trigger generation
  - Debug state machine state transition
  - Combinational and Sequential event generation
- Analysis Events
  - Program event detection
  - Data event detection
  - External trigger Detection
  - System event detection (i.e. cache miss)
  - Debug state machine state detection
- Analysis Configuration
  - Application access
  - Debugger access

**Table 6-96. ARM Debug Features**

| Category              | Hardware Feature                  | Availability                                                                                                                                                                                                                                                                                                       |
|-----------------------|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Basic Debug           | Software breakpoint               | Unlimited                                                                                                                                                                                                                                                                                                          |
|                       | Hardware breakpoint               | Up to 14 HWBPs, including:<br>2 precise <sup>(1)</sup> HWBP inside ARM core which are shared with watch points.<br>8 imprecise <sup>(1)</sup> HWBPs from ETM's address comparators, which are shared with trace function, and can be used as watch point too.<br>4 imprecise <sup>(1)</sup> HWBPs from ICECrusher. |
| Analysis              | Watch point                       | Up to 6 watch points, including:<br>2 from ARM core which is shared with HWBPs and can be associated with a data.<br>8 from ETM's address comparators, which are shared with trace function, and HWBPs.                                                                                                            |
|                       | Watch point with Data             | 2 from ARM core which is shared with HWBPs.<br>8 watch points from ETM can be associated with a data comparator, and ETM has total 4 data comparators.                                                                                                                                                             |
|                       | Counters/timers                   | 3x32-bit (1 cycle ; 2 event)                                                                                                                                                                                                                                                                                       |
|                       | External Event Trigger In         | 1                                                                                                                                                                                                                                                                                                                  |
|                       | External Event Trigger Out        | 1                                                                                                                                                                                                                                                                                                                  |
| Trace Control         | Address range for trace           | 4                                                                                                                                                                                                                                                                                                                  |
|                       | Data qualification for trace      | 2                                                                                                                                                                                                                                                                                                                  |
|                       | System events for trace control   | 20                                                                                                                                                                                                                                                                                                                 |
|                       | Counters/Timers for trace control | 2x16-bit                                                                                                                                                                                                                                                                                                           |
|                       | State Machines/Sequencers         | 1x3-State State Machine                                                                                                                                                                                                                                                                                            |
|                       | Context/Thread ID Comparator      | 1                                                                                                                                                                                                                                                                                                                  |
|                       | Independent trigger control units | 12                                                                                                                                                                                                                                                                                                                 |
| On-chip Trace Capture | Capture depth PC                  | 4k bytes ETB                                                                                                                                                                                                                                                                                                       |
|                       | Capture depth PC + Timing         | 4k bytes ETB                                                                                                                                                                                                                                                                                                       |
|                       | Application accessible            | Y                                                                                                                                                                                                                                                                                                                  |

(1) Precise hardware breakpoints will halt the processor immediately prior to the execution of the selected instruction. Imprecise breakpoints will halt the processor some number of cycles after the selected instruction depending on device conditions.

### 6.27.1 JTAG Port Description

The device target debug interface uses the five standard IEEE 1149.1(JTAG) signals ( $\overline{\text{TRST}}$ , TCK, TMS, TDI, and TDO), and a return clock (RTCK) due to the clocking requirements of the ARM926EJ-S .

TRST holds the debug and boundary scan logic in reset when pulled low (its default state). Since TRST has an internal pull-down resistor, this ensures that at power up the device functions in its normal (non-test) operation mode if TRST is not connected. Otherwise, TRST should be driven inactive by the emulator or boundary scan controller. Boundary scan test cannot be performed while the TRST pin is pulled low.

**Table 6-97. JTAG Port Description**

| PIN                      | TYPE | NAME                | DESCRIPTION                                                                                                                                                                                          |
|--------------------------|------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $\overline{\text{TRST}}$ | I    | Test Logic Reset    | When asserted (active low) causes all test and debug logic in the device to be reset along with the IEEE 1149.1 interface                                                                            |
| TCK                      | I    | Test Clock          | This is the test clock used to drive an IEEE 1149.1 TAP state machine and logic. Depending on the emulator attached to , this is a free running clock or a gated clock depending on RTCK monitoring. |
| RTCK                     | O    | Returned Test Clock | Synchronized TCK. Depending on the emulator attached to, the JTAG signals are clocked from RTCK or RTCK is monitored by the emulator to gate TCK.                                                    |
| TMS                      | I    | Test Mode Select    | Directs the next state of the IEEE 1149.1 test access port state machine                                                                                                                             |
| TDI                      | I    | Test Data Input     | Scan data input to the device                                                                                                                                                                        |
| TDO                      | O    | Test Data Output    | Scan data output of the device                                                                                                                                                                       |

### 6.27.2 Scan Chain Configuration Parameters

[Table 6-98](#) shows the TAP configuration details required to configure the router/emulator for this device.

**Table 6-98. JTAG Port Description**

| Router Port ID | Default TAP | TAP Name | Tap IR Length |
|----------------|-------------|----------|---------------|
| 17             | No          | Reserved | 38            |
| 18             | No          | ARM926   | 4             |
| 19             | No          | ETB      | 4             |

The router is ICEPick revision C and has a 6-bit IR length.

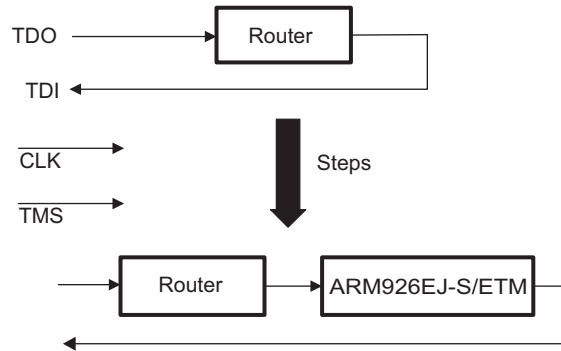
### 6.27.3 Initial Scan Chain Configuration

The first level of debug interface that sees the scan controller is the TAP router module. The debugger can configure the TAP router for serially linking up to 16 TAP controllers or individually scanning one of the TAP controllers without disrupting the IR state of the other TAPs.

#### 6.27.3.1 Adding TAPS to the Scan Chain

The TAP router must be programmed to add additional TAPs to the scan chain. The following JTAG scans must be completed to add the ARM926EJ-S to the scan chain.

A Power-On Reset (POR) or the JTAG Test-Logic Reset state configures the TAP router to contain only the router's TAP.



**Figure 6-49. Adding ARM926EJ-S to the scan chain**

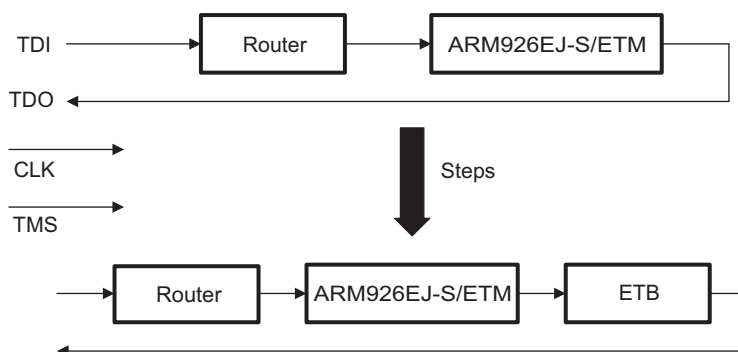
**Pre-amble:** The device whose data reaches the emulator first is listed first in the board configuration file. This device is a pre-ample for all the other devices. This device has the lowest device ID.

**Post-ample:** The device whose data reaches the emulator last is listed last in the board configuration file. This device is a post-ample for all the other devices. This device has the highest device ID.

- Function : Update the JTAG preamble and post-ample counts.
  - Parameter : The IR pre-ample count is '0'.
  - Parameter : The IR post-ample count is '0'.
  - Parameter : The DR pre-ample count is '0'.
  - Parameter : The DR post-ample count is '0'.
  - Parameter : The IR main count is '6'.
  - Parameter : The DR main count is '1'.
- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'pause-ir'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is '0x00000007'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-dr'.
  - Parameter : The JTAG destination state is 'pause-dr'.
  - Parameter : The bit length of the command is '8'.
  - Parameter : The send data value is '0x00000089'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'pause-ir'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is '0x00000002'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Embed the port address in next command.
  - Parameter : The port address field is '0x0f000000'.
  - Parameter : The port address value is '3'.

- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-dr'.
  - Parameter : The JTAG destination state is 'pause-dr'.
  - Parameter : The bit length of the command is '32'.
  - Parameter : The send data value is '0xa2002108'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Do a send-only all-ones JTAG IR/DR scan.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'run-test/idle'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is 'all-ones'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Wait for a minimum number of TCLK pulses.
  - Parameter : The count of TCLK pulses is '10'.
- Function : Update the JTAG preamble and post-amble counts.
  - Parameter : The IR pre-ample count is '0'.
  - Parameter : The IR post-ample count is '6'.
  - Parameter : The DR pre-ample count is '0'.
  - Parameter : The DR post-ample count is '1'.
  - Parameter : The IR main count is '4'.
  - Parameter : The DR main count is '1'.

The initial scan chain contains only the TAP router module. The following steps must be completed in order to add ETB TAP to the scan chain.



**Figure 6-50. Adding ETB to the scan chain**

- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'pause-ir'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is '0x00000007'.
  - Parameter : The actual receive data is 'discarded'.

- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-dr'.
  - Parameter : The JTAG destination state is 'pause-dr'.
  - Parameter : The bit length of the command is '8'.
  - Parameter : The send data value is '0x00000089'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'pause-ir'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is '0x00000002'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Embed the port address in next command.
  - Parameter : The port address field is '0x0f000000'.
  - Parameter : The port address value is '3'.
- Function : Do a send-only JTAG IR/DR scan.
  - Parameter : The route to JTAG shift state is 'shortest transition'.
  - Parameter : The JTAG shift state is 'shift-dr'.
  - Parameter : The JTAG destination state is 'pause-dr'.
  - Parameter : The bit length of the command is '32'.
  - Parameter : The send data value is '0xa3302108'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Do a send-only all-ones JTAG IR/DR scan.
  - Parameter : The JTAG shift state is 'shift-ir'.
  - Parameter : The JTAG destination state is 'run-test/idle'.
  - Parameter : The bit length of the command is '6'.
  - Parameter : The send data value is 'all-ones'.
  - Parameter : The actual receive data is 'discarded'.
- Function : Wait for a minimum number of TCLK pulses.
  - Parameter : The count of TCLK pulses is '10'.
- Function : Update the JTAG preamble and post-amble counts.
  - Parameter : The IR pre-amble count is '0'.
  - Parameter : The IR post-amble count is '6 + 4'.
  - Parameter : The DR pre-amble count is '0'.
  - Parameter : The DR post-amble count is '1 + 1'.
  - Parameter : The IR main count is '4'.
  - Parameter : The DR main count is '1'.

#### 6.27.4 JTAG 1149.1 Boundary Scan Considerations

To use boundary scan, the following sequence should be followed:

- Execute a valid reset sequence and exit reset
- Wait at least 6000 OSCIN clock cycles
- Enter boundary scan mode using the JTAG pins

If TRST is not driven by the boundary scan tool or tester, TRST should be externally pulled high during boundary scan testing.

## 6.28 IEEE 1149.1 JTAG

The JTAG <sup>(1)</sup> interface is used for BSDL testing and emulation of the device.

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

While both  $\overline{\text{TRST}}$  and  $\overline{\text{RESET}}$  need to be asserted upon power up, only  $\overline{\text{RESET}}$  needs to be released for the device to boot properly.  $\overline{\text{TRST}}$  may be asserted indefinitely for normal operation, keeping the JTAG port interface and device's emulation logic in the reset state.

$\overline{\text{TRST}}$  only needs to be released when it is necessary to use a JTAG controller to debug the 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.

$\overline{\text{RESET}}$  must be released only in order for boundary-scan JTAG to read the variant field of IDCODE correctly. Other boundary-scan instructions work correctly independent of current state of  $\overline{\text{RESET}}$ .

For maximum reliability, the device 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.

### 6.28.1 JTAG Peripheral Register Description(s) – JTAG ID Register (DEVIDR0)

**Table 6-99. DEVIDR0 Register**

| BYTE ADDRESS | ACRONYM | REGISTER DESCRIPTION         | COMMENTS                                          |
|--------------|---------|------------------------------|---------------------------------------------------|
| 0x01C1 4018  | DEVIDR0 | JTAG Identification Register | Read-only. Provides 32-bit JTAG ID of the device. |

(1) 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 device, the JTAG ID register resides at address location 0x01C1 4018. The register hex value for each silicon revision is:

- 0x8B7D F02F for silicon revision 1.1
- 0x9B7D F02F for silicon revisions 3.0, 2.1, and 2.0

For the actual register bit names and their associated bit field descriptions, see [Figure 6-51](#) and [Table 6-100](#).

**Figure 6-51. JTAG ID (DEVIDR0) Register Description - Register Value**

| 31              | 28 | 27 | 12                    | 11 | 1 | 0   |
|-----------------|----|----|-----------------------|----|---|-----|
| VARIANT (4-bit) |    |    | PART NUMBER (16-bit)  |    |   | LSB |
| R-xxxx          |    |    | R-1011 0111 1101 1111 |    |   | R-1 |

LEGEND: R = Read, W = Write, n = value at reset

**Table 6-100. JTAG ID Register Selection Bit Descriptions**

| BIT   | NAME         | DESCRIPTION                     |
|-------|--------------|---------------------------------|
| 31:28 | VARIANT      | Variant (4-Bit) value           |
| 27:12 | PART NUMBER  | Part Number (16-Bit) value      |
| 11-1  | MANUFACTURER | Manufacturer (11-Bit) value     |
| 0     | LSB          | LSB. This bit is read as a "1". |

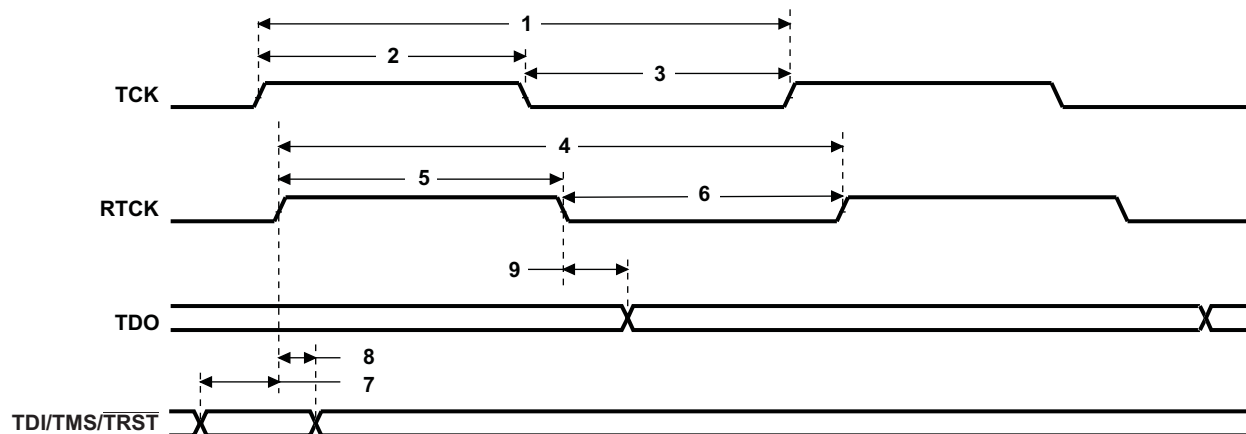
## 6.28.2 JTAG Test-Port Electrical Data/Timing

**Table 6-101. Timing Requirements for JTAG Test Port (see Figure 6-52)**

| No. | PARAMETER                                                                          | MIN | MAX | UNIT |
|-----|------------------------------------------------------------------------------------|-----|-----|------|
| 1   | $t_{c(TCK)}$ Cycle time, TCK                                                       | 40  |     | ns   |
| 2   | $t_{w(TCKH)}$ Pulse duration, TCK high                                             | 16  |     | ns   |
| 3   | $t_{w(TCKL)}$ Pulse duration, TCK low                                              | 16  |     | ns   |
| 4   | $t_{c(RTCK)}$ Cycle time, RTCK                                                     | 40  |     | ns   |
| 5   | $t_{w(RTCKH)}$ Pulse duration, RTCK high                                           | 16  |     | ns   |
| 6   | $t_{w(RTCKL)}$ Pulse duration, RTCK low                                            | 16  |     | ns   |
| 7   | $t_{su(TDIV-RTCKH)}$ Setup time, TDI/TMS/ $\overline{TRST}$ valid before RTCK high | 4   |     | ns   |
| 8   | $t_{h(RTCKH-TDIV)}$ Hold time, TDI/TMS/ $\overline{TRST}$ valid after RTCK high    | 4   |     | ns   |

**Table 6-102. Switching Characteristics Over Recommended Operating Conditions for JTAG Test Port (see Figure 6-52)**

| No. | PARAMETER                                             | MIN | MAX | UNIT |
|-----|-------------------------------------------------------|-----|-----|------|
| 9   | $t_{d(RTCKL-TDOV)}$ Delay time, RTCK low to TDO valid |     | 15  | ns   |

**Figure 6-52. JTAG Test-Port Timing**

## 7 Device and Documentation Support

TI offers an extensive line of development tools. Tools and software to evaluate the performance of the device, generate code, and develop solutions are listed below.

### 7.1 Device Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all microprocessors (MPUs) and support tools. Each device has one of three prefixes: X, P, or null (no prefix) (for example, *AM1705*). 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 (TMDX) through fully qualified production devices and tools (TMDS).

Device development evolutionary flow:

- X** Experimental device that is not necessarily representative of the final device's electrical specifications and may not use production assembly flow.
- P** Prototype device that is not necessarily the final silicon die and may not necessarily meet final electrical specifications.
- null** Production version of the silicon die that is fully qualified.

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.

X and P devices and TMDX development-support tools are shipped against the following disclaimer:

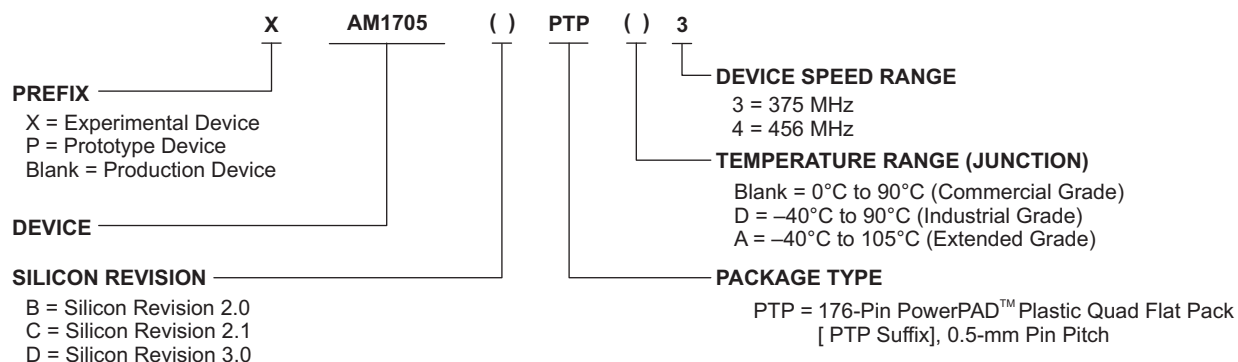
"Developmental product is intended for internal evaluation purposes."

Production 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 (X or P) 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, ZKB), the temperature range (for example, blank is the default commercial temperature range), and the device speed range, in megahertz (for example, 375). [Figure 7-1](#) provides a legend for reading the complete device name for any *AM1705* device.

For orderable part numbers of *AM1705* devices in the *your package* package types, see the Package Option Addendum of this document, [ti.com](http://ti.com), or contact your TI sales representative.



**Figure 7-1. Device Nomenclature**

## 7.2 Tools and Software

TI offers an extensive line of development tools for the device 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 the device applications:

### Software

**Code Composer Studio™ Integrated Development Environment (IDE): including Editor C/C++/Assembly Code Generation, and Debug plus additional development tools** **Development Tools**

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

## 7.3 Documentation Support

To receive notification of documentation updates, navigate to the device product folder on [ti.com](http://ti.com). In the upper right corner, click on *Alert me* to register and receive a weekly digest of any product information that has changed. For change details, review the revision history included in any revised document.

The current documentation that describes the DSP, related peripherals, and other technical collateral is listed below.

### User's Guides

**SPRUGU3** *AM1705 ARM Microprocessor System Reference Guide*

**SPRUFU0** *AM17x/AM18x ARM Microprocessor Peripherals Overview Reference Guide*

## 7.4 Community Resources

The following links connect to TI community resources. Linked contents are provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's [Terms of Use](#).

**TI E2E™ Online Community** The TI engineer-to-engineer (E2E) community was created to foster collaboration among engineers. At [e2e.ti.com](http://e2e.ti.com), you can ask questions, share knowledge, explore ideas and help solve problems with fellow engineers.

**TI Embedded Processors Wiki** Established to help developers get started with Embedded Processors from Texas Instruments and to foster innovation and growth of general knowledge about the hardware and software surrounding these devices.

## 7.5 Trademarks

E2E is a trademark of Texas Instruments.

ARM9 is a trademark of ARM.

ETM9, CoreSight are trademarks of ARM Limited.

All other trademarks are the property of their respective owners.

## 7.6 Electrostatic Discharge Caution



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

## 7.7 Export Control Notice

Recipient agrees to not knowingly export or re-export, directly or indirectly, any product or technical data (as defined by the U.S., EU, and other Export Administration Regulations) including software, or any controlled product restricted by other applicable national regulations, received from disclosing party under nondisclosure obligations (if any), or any direct product of such technology, to any destination to which such export or re-export is restricted or prohibited by U.S. or other applicable laws, without obtaining prior authorization from U.S. Department of Commerce and other competent Government authorities to the extent required by those laws.

## **7.8 Glossary**

**TI Glossary** This glossary lists and explains terms, acronyms, and definitions.

## 8 Mechanical Packaging and Orderable Information

This section describes the device orderable part numbers, packaging options, materials, thermal and mechanical parameters.

### 8.1 Thermal Data for PTP

The following table(s) show the thermal resistance characteristics for the PowerPAD™ PTP mechanical package.

**Table 8-1. Thermal Resistance Characteristics (PowerPAD™ Package) [PTP]"**

| No. |                                         | °C/W <sup>(1)</sup> | °C/W <sup>(2)</sup> | °C/W <sup>(3)</sup> | °C/W <sup>(4)</sup> | AIR FLOW<br>(m/s) <sup>(5)</sup> |
|-----|-----------------------------------------|---------------------|---------------------|---------------------|---------------------|----------------------------------|
| 1   | R $\theta_{JC}$ Junction-to-case        | 7.8                 | 9.4                 | 8.6                 | 10.1                | N/A                              |
| 2   | R $\theta_{JB}$ Junction-to-board       | 6.2                 | 9.9                 | 7.1                 | 10.6                | N/A                              |
| 3   | R $\theta_{JA}$ Junction-to-free air    | 21.3                | 27.9                | 23.2                | 30.6                | 0.00                             |
| 4   | R $\theta_{JMA}$ Junction-to-moving air | 14.3                | 20.2                |                     | 22.6                | 0.50                             |
| 5   |                                         | 13.1                | 18.6                |                     | 21.0                | 1.00                             |
| 6   |                                         | 12.1                | 17.4                |                     | 19.6                | 2.00                             |
| 7   |                                         | 11.2                | 16.2                |                     | 18.2                | 4.00                             |
| 8   | Psi $_{JT}$ Junction-to-package top     | 0.5                 | 0.7                 |                     | 0.8                 | 0.00                             |
| 9   |                                         | 0.6                 | 0.9                 |                     | 1.0                 | 0.50                             |
| 10  |                                         | 0.7                 | 1.0                 |                     | 1.1                 | 1.00                             |
| 11  |                                         | 0.8                 | 1.1                 |                     | 1.3                 | 2.00                             |
| 12  |                                         | 1.0                 | 1.3                 |                     | 1.5                 | 4.00                             |
| 13  | Psi $_{JB}$ Junction-to-board           | 6.3                 | 9.5                 |                     | 10.8                | 0.00                             |
| 14  |                                         | 5.9                 | 8.8                 |                     | 9.9                 | 0.50                             |
| 15  |                                         | 5.9                 | 8.7                 |                     | 9.8                 | 1.00                             |
| 16  |                                         | 5.8                 | 8.6                 |                     | 9.7                 | 2.00                             |
| 17  |                                         | 5.8                 | 8.5                 |                     | 9.6                 | 4.00                             |

- (1) Simulation data, using a model of a JEDEC defined 2S2P system with a 12mmx12mm copper pad on the top and bottom copper layers connected with an 8x8 thermal via array and soldered to the package thermal pad. Power dissipation of 1W assumed, 70C Ambient temp assumed. Signal layer copper coverage 20%, inner layer copper coverage 90%. Actual performance will change based on environment as well as application. For more information, see these EIA/JEDEC standards – EIA/JESD51-2, *Integrated Circuits Thermal Test Method Environment Conditions - Natural Convection (Still Air)* and JESD51-7, *High Effective Thermal Conductivity Test Board for Leadless Surface Mount Packages*.
- (2) Simulation data, using the same model but with 1oz (35um) top and bottom copper thickness and 0.5oz (18um) inner copper thickness. Power dissipation of 1W and ambient temp of 70C assumed.
- (3) Simulation data, 1S1P PCB model with 12x12mm copper pad on the top layer soldered to device thermal pad and connected to the bottom copper layer (90% copper) with an 8x8 thermal via array. Power dissipation of 1W and ambient temp of 70C assumed. Copper thickness 2oz (70um) top and bottom.
- (4) Simulation data, 1S1P PCB model with 12x12mm copper pad on the top layer soldered to device thermal pad and connected to the bottom copper layer (90% copper) with an 8x8 thermal via array. Power dissipation of 1W and ambient temp of 70C assumed. Copper thickness 1oz (35um) top and bottom.
- (5) m/s = meters per second

### 8.2 Supplementary Information About the 176-pin PTP PowerPAD™ Package

This section highlights a few important details about the 176-pin PTP PowerPAD™ package. Texas Instruments' [PowerPAD Thermally Enhanced Package Technical Brief](#) (SLMA002) should be consulted when creating a PCB footprint for this device.

### 8.2.1 Standoff Height

As illustrated in [Figure 8-1](#), the standoff height specification for this device (between 0.050 mm and 0.150 mm) is measured from the seating plane established by the three lowest package pins to the **lowest** point on the package body. Due to warpage, the lowest point on the package body is located in the center of the package at the exposed thermal pad.

Using this definition of standoff height provides the correct result for determining the correct solder paste thickness. According to TI's [PowerPAD Thermally Enhanced Package Technical Brief \(SLMA002\)](#), the recommended range of solder paste thickness for this package is between 0.152 mm and 0.178 mm.



Figure 8-1. Standoff Height Measurement on 176-pin PTP Package

### 8.2.2 PowerPAD™ PCB Footprint

In general, for proper thermal performance, the thermal pad under the package body should be as large as possible. However, the soldermask opening for the PowerPAD™ should be sized to match the pad size on the 176-pin PTP package; as illustrated in [Figure 8-2](#).

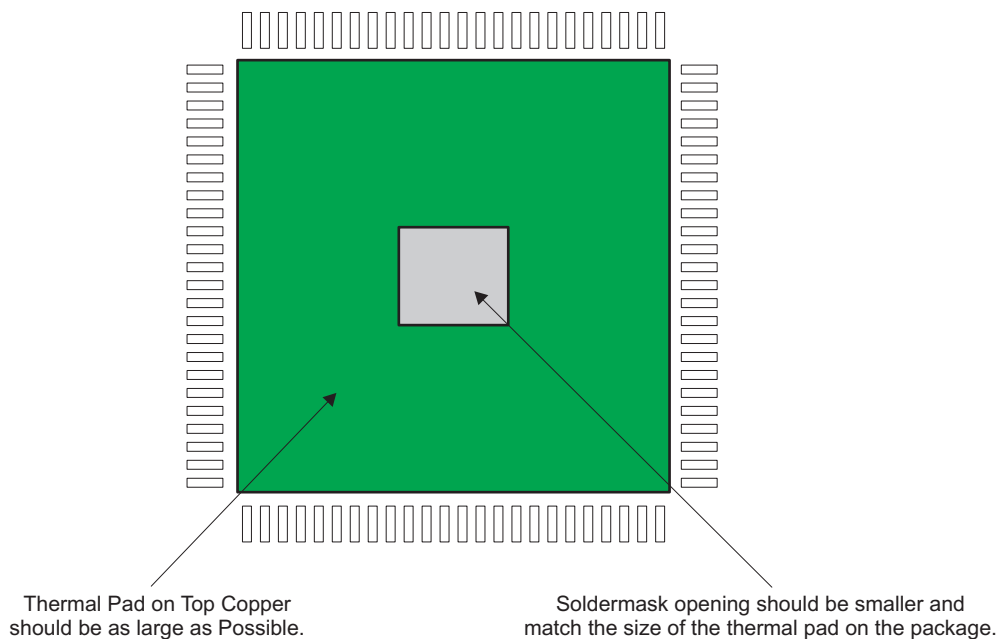


Figure 8-2. Soldermask Opening Should Match Size of Package Thermal Pad

## 8.3 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 part number        | Status<br>(1) | Material type<br>(2) | Package   Pins    | Package qty   Carrier | RoHS<br>(3) | Lead finish/<br>Ball material<br>(4) | MSL rating/<br>Peak reflow<br>(5) | Op temp (°C) | Part marking<br>(6) |
|------------------------------|---------------|----------------------|-------------------|-----------------------|-------------|--------------------------------------|-----------------------------------|--------------|---------------------|
| <a href="#">AM1705DPTP3</a>  | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | 0 to 90      | AM1705DPTP3         |
| AM1705DPTP3.B                | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | 0 to 90      | AM1705DPTP3         |
| <a href="#">AM1705DPTP4</a>  | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | 0 to 90      | AM1705DPTP4         |
| AM1705DPTP4.B                | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | 0 to 90      | AM1705DPTP4         |
| <a href="#">AM1705DPTPA3</a> | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | Call TI                              | Level-4-260C-72 HR                | -40 to 105   | AM1705DPTPA3        |
| AM1705DPTPA3.B               | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | Call TI                              | Level-4-260C-72 HR                | -40 to 105   | AM1705DPTPA3        |
| <a href="#">AM1705DPTPD4</a> | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | -40 to 90    | AM1705DPTPD4        |
| AM1705DPTPD4.B               | Active        | Production           | HLQFP (PTP)   176 | 40   JEDEC TRAY (5+1) | Yes         | NIPDAU                               | Level-4-260C-72 HR                | -40 to 90    | AM1705DPTPD4        |

<sup>(1)</sup> **Status:** For more details on status, see our [product life cycle](#).

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

<sup>(3)</sup> **RoHS values:** Yes, No, RoHS Exempt. See the [TI RoHS Statement](#) for additional information and value definition.

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

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

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

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

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

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

## GENERIC PACKAGE VIEW

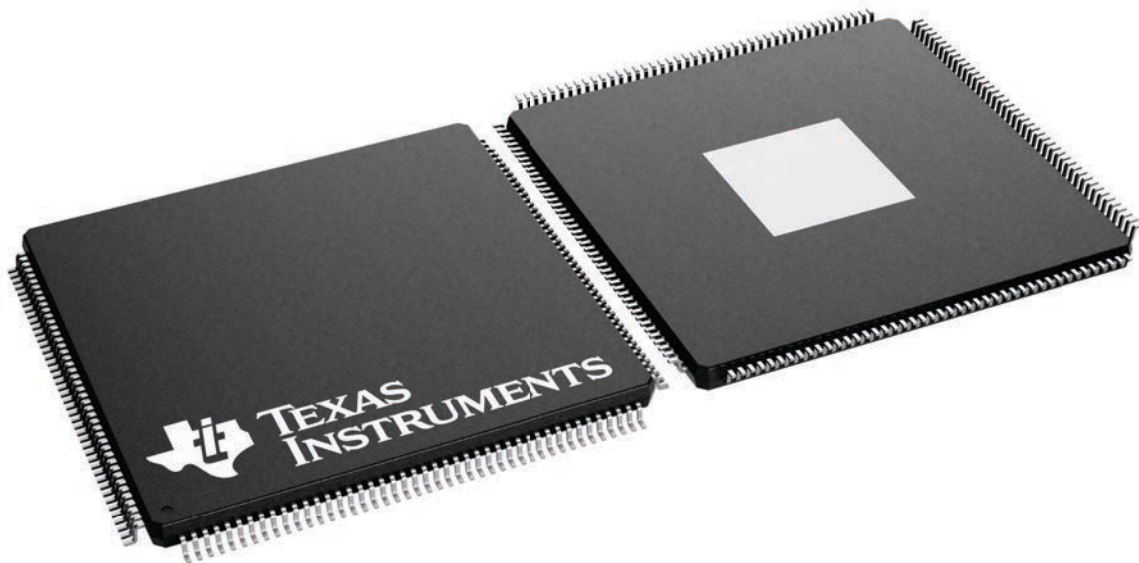
**PTP 176**

**HLQFP - 1.6 mm max height**

24 x 24, 0.5 mm pitch

PLASTIC QUAD FLATPACK

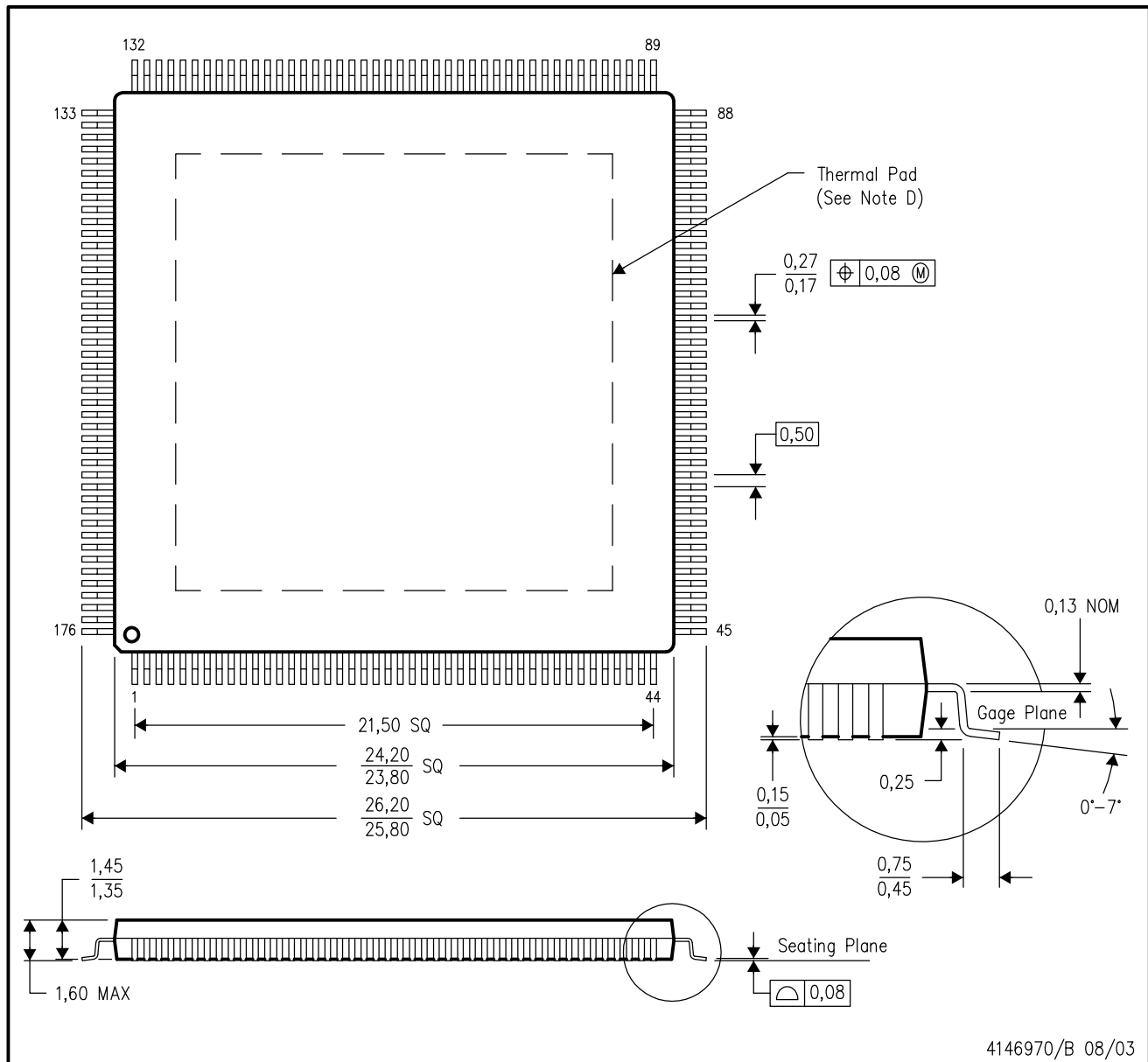
This image is a representation of the package family, actual package may vary.  
Refer to the product data sheet for package details.



4226435/A

PTP (S-PQFP-G176)

PowerPAD™ PLASTIC QUAD FLATPACK



- NOTES:
- A. All linear dimensions are in millimeters.
  - B. This drawing is subject to change without notice.
  - C. Body dimensions do not include mold flash or protrusion.
  - D. This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 for information regarding recommended board layout. This document is available at [www.ti.com](http://www.ti.com) <<http://www.ti.com>>.
  - E. Falls within JEDEC MO-026

PowerPAD is a trademark of Texas Instruments.

PTP (S-PQFP-G176)

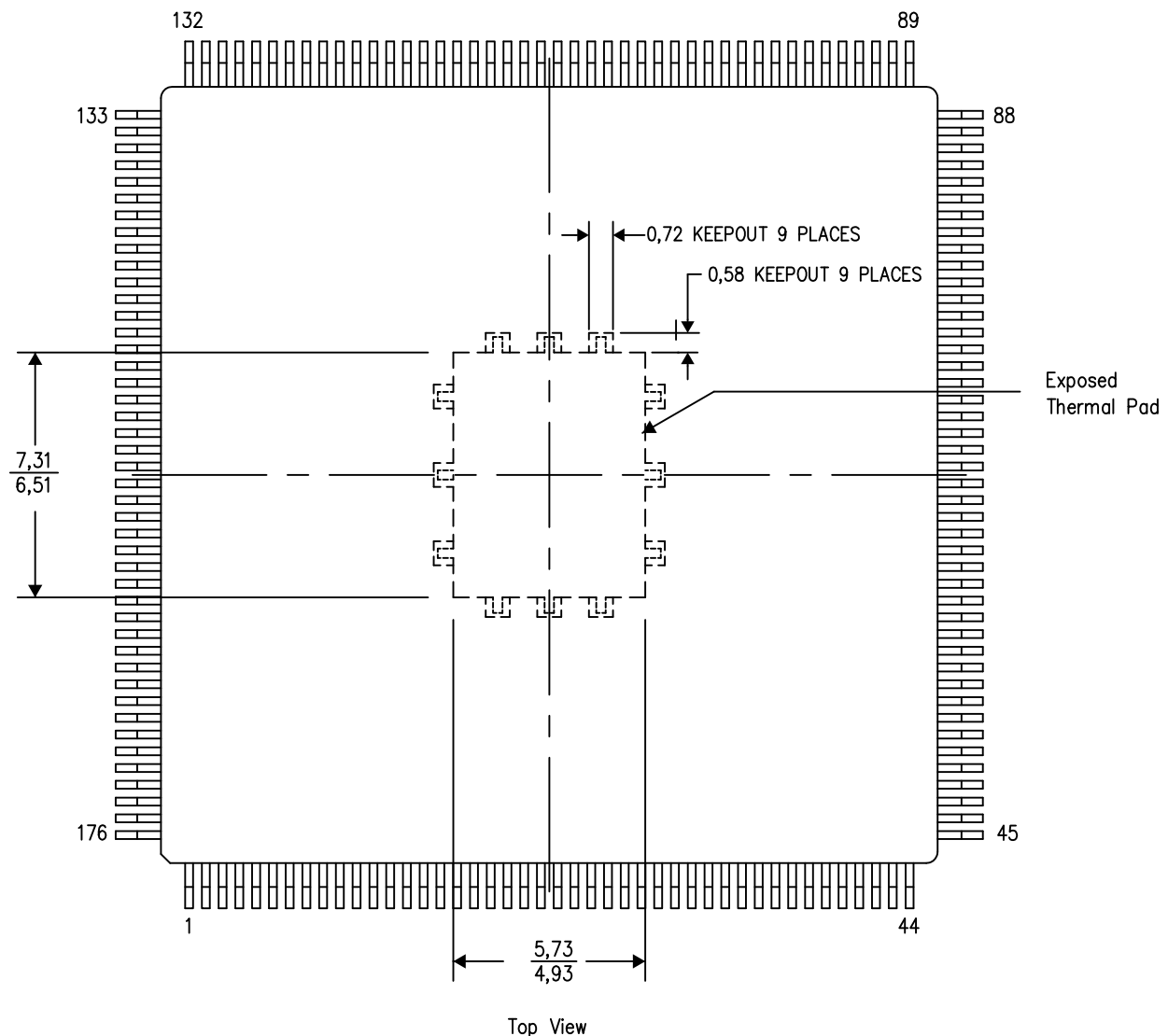
PowerPAD™ PLASTIC QUAD FLATPACK

## THERMAL INFORMATION

This PowerPAD™ package incorporates an exposed thermal pad that is designed to be attached to a printed circuit board (PCB). The thermal pad must be soldered directly to the PCB. After soldering, the PCB can be used as a heatsink. In addition, through the use of thermal vias, the thermal pad can be attached directly to the appropriate copper plane shown in the electrical schematic for the device, or alternatively, can be attached to a special heatsink structure designed into the PCB. This design optimizes the heat transfer from the integrated circuit (IC).

For additional information on the PowerPAD package and how to take advantage of its heat dissipating abilities, refer to Technical Brief, PowerPAD Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 and Application Brief, PowerPAD Made Easy, Texas Instruments Literature No. SLMA004. Both documents are available at [www.ti.com](http://www.ti.com).

The exposed thermal pad dimensions for this package are shown in the following illustration.



Exposed Thermal Pad Dimensions

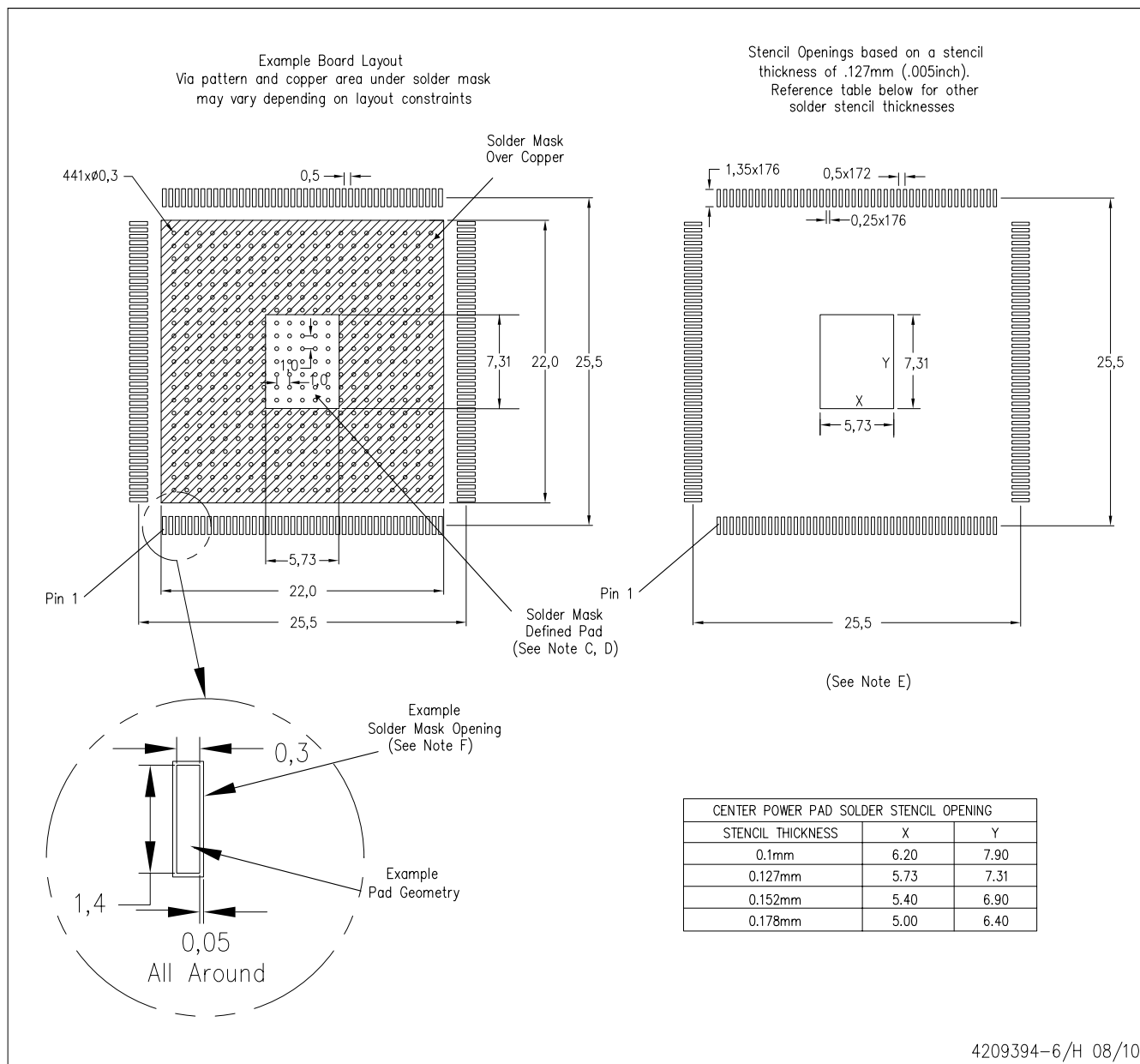
4209350-6/F 04/13

NOTE: All linear dimensions are in millimeters

PowerPAD is a trademark of Texas Instruments

PTP (S-PQFP-G176)

PowerPAD™ PLASTIC QUAD FLATPACK



- NOTES:
- All linear dimensions are in millimeters.
  - This drawing is subject to change without notice.
  - Customers should place a note on the circuit board fabrication drawing not to alter the center solder mask defined pad.
  - This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002, SLMA004, and also the Product Data Sheets for specific thermal information, via requirements, and recommended board layout. These documents are available at [www.ti.com](http://www.ti.com) <<http://www.ti.com>>. Publication IPC-7351 is recommended for alternate designs.
  - Laser cutting apertures with trapezoidal walls and also rounding corners will offer better paste release. Customers should contact their board assembly site for stencil design recommendations. Example stencil design based on a 50% volumetric metal load solder paste. Refer to IPC-7525 for other stencil recommendations.
  - Customers should contact their board fabrication site for solder mask tolerances between and around signal pads.

PowerPAD is a trademark of Texas Instruments.

## IMPORTANT NOTICE AND DISCLAIMER

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

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

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

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

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

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025