1 OMAP-L137 Low-Power Applications Processor

1.1 Features

- **Applications**
  - **Industrial Control**
  - **USB, Networking**
  - **High-Speed Encoding**
  - **Professional Audio**
- **Software Support**
  - **TI DSP/BIOS™**
  - **Chip Support Library and DSP Library**
- **Dual Core SoC**
  - **300-MHz ARM926EJ-S™ RISC MPU**
  - **300-MHz C674x™ VLIW DSP**
- **ARM926EJ-S Core**
  - **32-Bit and 16-Bit (Thumb®) Instructions**
  - **DSP Instruction Extensions**
  - **Single Cycle MAC**
  - **ARM® Jazelle® Technology**
  - **EmbeddedICE-RT™ for Real-Time Debug**
- **ARM9 Memory Architecture**
- **C674x Instruction Set Features**
  - **Superset of the C67x+™ and C64x+™ ISAs**
  - **2400/1800 C674x MIPS/MFLOPS**
  - **Byte-Addressable (8-/16-/32-/64-Bit Data)**
  - **8-Bit Overflow Protection**
  - **Bit-Field Extract, Set, Clear**
  - **Normalization, Saturation, Bit-Counting**
  - **Compact 16-Bit Instructions**
- **C674x Two Level Cache Memory Architecture**
  - **32K-Byte L1P Program RAM/Cache**
  - **32K-Byte L1D Data RAM/Cache**
  - **256K-Byte L2 Unified Mapped RAM/Cache**
  - **Flexible RAM/Cache Partition (L1 and L2)**
  - **1024K-Byte L2 ROM**
- **Enhanced Direct-Memory-Access Controller 3 (EDMA3):**
  - **2 Transfer Controllers**
  - **32 Independent DMA Channels**
  - **8 Quick DMA Channels**
  - **Programmable Transfer Burst Size**
- **TMS320C674x™ Floating Point VLIW DSP Core**
  - **Load-Store Architecture With Non-Aligned Support**
  - **64 General-Purpose Registers (32 Bit)**
  - **Six ALU (32-/40-Bit) Functional Units**
    - **Supports 32-Bit Integer, SP (IEEE Single Precision/32-Bit) and DP (IEEE Double Precision/64-Bit) Floating Point**
    - **Supports up to Four SP Additions Per Clock, Four DP Additions Every 2 Clocks**
    - **Supports up to Two Floating Point (SP or DP) Approximate Reciprocal or Square Root Operations Per Cycle**
  - **Two Multiply Functional Units**
    - **Mixed-Precision IEEE Floating Point Multiply Supported up to:**
      - **2 SP x SP -> SP Per Clock**
      - **2 SP x SP -> DP Every Two Clocks**
      - **2 SP x DP -> DP Every Three Clocks**
      - **2 DP x DP -> DP Every Four Clocks**
    - **Fixed Point Multiply Supports Two 32 x 32-Bit Multiplies, Four 16 x 16-Bit Multiplies, or Eight 8 x 8-Bit Multiplies per Clock Cycle, and Complex Multiples**
  - **Instruction Packing Reduces Code Size**
  - **All Instructions Conditional**
  - **Hardware Support for Modulo Loop Operation**
  - **Protected Mode Operation**
  - **Exceptions Support for Error Detection and Program Redirection**
- **128K-Byte RAM Shared Memory**
- **Two External Memory Interfaces:**
  - **EMIFA**
    - **NOR (8-/16-Bit-Wide Data)**
    - **NAND (8-/16-Bit-Wide Data)**
    - **16-Bit SDRAM With 128MB Address Space**
  - **EMIFB**
    - **32-Bit or 16-Bit SDRAM With 256MB Address Space**
- **Three Configurable 16550 type UART Modules:**
  - **UART0 With Modem Control Signals**
  - **16-byte FIFO**
  - **16x or 13x Oversampling Option**
- **LCD Controller**
- **Two Serial Peripheral Interfaces (SPI) Each**

PRODUCT PREVIEW

With One Chip-Select

- Multimedia Card (MMC)/Secure Digital (SD) Card Interface with Secure Data I/O (SDIO)
- Two Master/Slave Inter-Integrated Circuit (I^2C Bus™)
- USB 1.1 OHCI (Host) With Integrated PHY (USB1)
- USB 2.0 OTG Port With Integrated PHY (USB0)
  - USB 2.0 High-/Full-Speed Client
  - USB 2.0 High-/Full-/Low-Speed Host
  - End Point 0 (Control)
  - End Points 1,2,3,4 (Control, Bulk, Interrupt or ISOC) Rx and Tx
- Three Multichannel Audio Serial Ports:
  - Transmit/Receive Clocks up to 50 MHz
  - Six Clock Zones and 28 Serial Data Pins
  - Supports TDM, I2S, and Similar Formats
  - DIT-Capable (McASP2)
  - FIFO buffers for Transmit and Receive
- 10/100 Mb/s Ethernet MAC (EMAC):
  - IEEE 802.3 Compliant (3.3-V I/O Only)
  - RMII Media Independent Interface
  - Management Data I/O (MDIO) Module
- One Host-Port Interface (HPI) With 16-Bit-Wide Muxed Address/Data Bus For High Bandwidth
- Real-Time Clock With 32 KHz Oscillator and Separate Power Rail
- One 64-Bit General-Purpose Timer (Configurable as Two 32-Bit Timers)
- One 64-Bit General-Purpose Timer (Watch Dog)
- Three Enhanced Pulse Width Modulators (eHRPWM):
  - 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 Modules (eCAP):
  - Configurable as 3 Capture Inputs or 3 Auxiliary Pulse Width Modulator (APWM) outputs
  - Single Shot Capture of up to Four Event Time-Stamps
- Two 32-Bit Enhanced Quadrature Encoder Pulse Modules (eQEP)
- 256-Ball Pb-Free Plastic Ball Grid Array (PBGA) [ZKB Suffix], 1.0-mm Ball Pitch
- Commercial or Extended Temperature

1.2 Trademarks

DSP/BIOS, TMS320C6000, C6000, TMS320, TMS320C62x, and TMS320C67x are trademarks of Texas Instruments.

All trademarks are the property of their respective owners.

1.3 Description

The OMAP-L137 is a Low-power applications processor based on an ARM926EJ-S™ and a C674x™ DSP core. It provides significantly lower power than other members of the TMS320C6000™ platform of DSPs.

The OMAP-L137 enables OEMs and ODMs to quickly bring to market devices featuring robust operating systems support, rich user interfaces, and high processing performance life through the maximum flexibility of a fully integrated mixed processor solution.

The dual-core architecture of the OMAP-L137 provides benefits of both DSP and Reduced Instruction Set Computer (RISC) technologies, incorporating a high-performance TMS320C674x DSP core and an ARM926EJ-S core.

The ARM926EJ-S is a 32-bit RISC processor core that performs 32-bit or 16-bit instructions and processes 32-bit, 16-bit, 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. It has separate 16K-byte instruction and 16K-byte data caches. Both are four-way associative with virtual index virtual tag (VIVT). The ARM core also has a 8KB RAM (Vector Table) and 64KB ROM.

The OMAP-L137 DSP core uses a two-level cache-based architecture. The Level 1 program cache (L1P) is a 32KB direct mapped cache and the Level 1 data cache (L1D) is a 32KB 2-way set-associative cache. The Level 2 program cache (L2P) consists of a 256KB memory space that is shared between program and data space. L2 also has a 1024KB ROM. L2 memory can be configured as mapped memory, cache, or combinations of the two. Although the DSP L2 is accessible by ARM and other hosts in the system, an additional 128KB RAM shared memory is available for use by other hosts without affecting DSP performance.

The peripheral set includes: a 10/100 Mb/s Ethernet MAC (EMAC) with a Management Data Input/Output (MDIO) module; two inter-integrated circuit (I2C) Bus interfaces; 3 multichannel audio serial port (McASP) with 16/12/4 serializers and FIFO buffers; 2 64-bit general-purpose timers each configurable (one configurable as watchdog); a configurable 16-bit host port interface (HPI); up to 8 banks of 16 pins of general-purpose input/output (GPIO) with programmable interrupt/event generation modes, multiplexed with other peripherals; 3 UART interfaces (one with $\overline{\text{RTS}}$ and $\overline{\text{CTS}}$); 3 enhanced high-resolution pulse width modulator (eHRPWM) peripherals; 3 32-bit enhanced capture (eCAP) module peripherals which can be configured as 3 capture inputs or 3 auxiliary pulse width modulator (APWM) outputs; 2 32-bit enhanced quadrature 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.

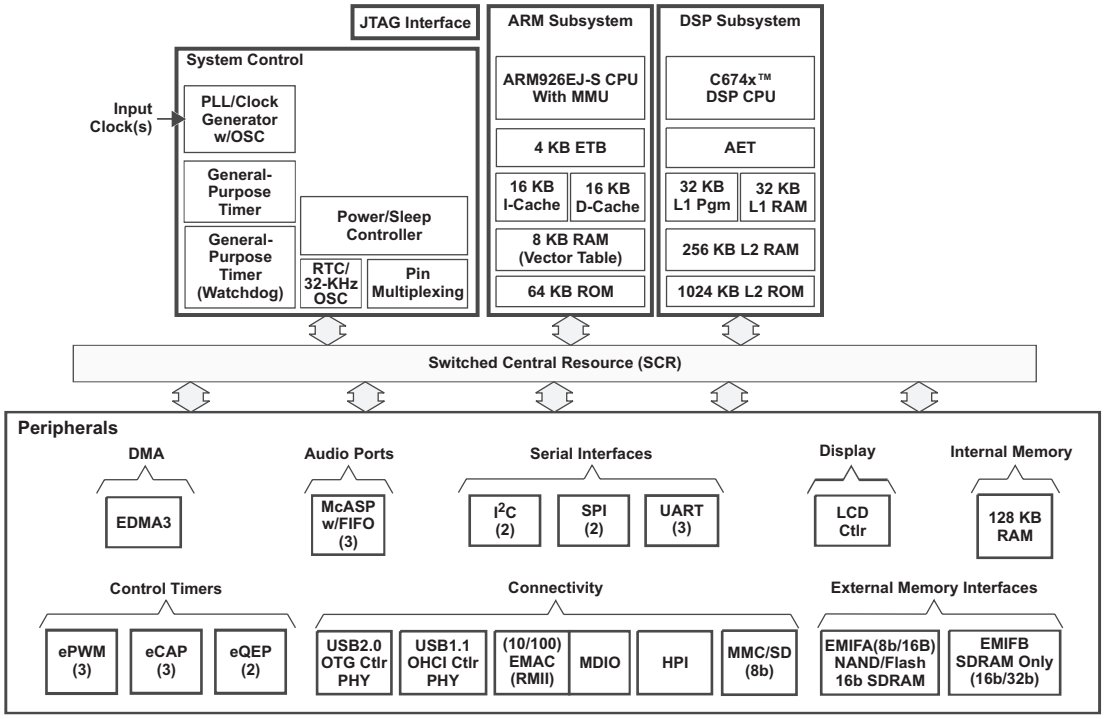
The Ethernet Media Access Controller (EMAC) provides an efficient interface between the OMAP-L137 and the network. The EMAC supports both 10Base-T and 100Base-TX, or 10 Mbits/second (Mbps) and 100 Mbps in either half- or full-duplex mode. Additionally an Management Data Input/Output (MDIO) interface is available for PHY configuration.

The HPI, I2C, SPI, USB1.1 and USB2.0 ports allow the OMAP-L137 to easily control peripheral devices and/or communicate with host processors.

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

The OMAP-L137 has a complete set of development tools for both the ARM and DSP. These include C compilers, a DSP assembly optimizer to simplify programming and scheduling, and a Windows™ debugger interface for visibility into source code execution.

1.4 Functional Block Diagram



Note: Not all peripherals are available at the same time due to multiplexing.

PRODUCT PREVIEW

PRODUCT PREVIEW

Contents

2 Revision History

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

This data manual revision history highlights the changes made to the SPRS563 device-specific data manual to make it an SPRS563A revision.

Table 2-1. Revision History

SEE	ADDITIONS/MODIFICATIONS/DELETIONS
Section 1.4, Functional Block Diagram	Updated the Functional Block Diagram.

3 Device Overview

3.1 Device Characteristics

Table 3-1 provides an overview of the OMAP-L137 Low power applications processor. 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 OMAP-L137 Processor

HARDWARE FEATURES		OMAP-L137
Peripherals Not all peripherals pins are available at the same time (for more detail, see the Device Configurations section).	EMIFB	SDRAM only, 16/32-bit bus width
	EMIFA	Asynchronous (8/16-bit bus width) RAM, Flash, 16-bit SDRAM, 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^2C	2 (both Master/Slave)
	Multichannel Audio Serial Port [McASP]	3 (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
	UHPI	1 (16-bit multiplexed address/data)
	USB 2.0 (USB0)	High-Speed OTG Controller with on-chip OTG PHY
	USB 1.1 (USB1)	Full-Speed OHCI (as host) with on-chip PHY
	General-Purpose Input/Output Port	8 banks of 16-bit
	LCD Controller	1
On-Chip Memory	Size (Bytes)	488KB RAM, 1088KB ROM
	Organization	**DSP** 32KB L1 Program (L1P)/Cache (up to 32KB) 32KB L1 Data (L1D)/Cache (up to 32KB) 256KB Unified Mapped RAM/Cache (L2) 1024KB ROM (L2) DSP Memories can be made accessible to ARM, EDMA3, and other peripherals. **ARM** 16KB I-Cache 16KB D-Cache 8KB RAM (Vector Table) 64KB ROM **ADDITIONAL SHARED MEMORY** 128KB RAM
C674x CPU ID + CPU Rev ID	Control Status Register (CSR.[31:16])	0x1400
C674x Megamodule Revision	Revision ID Register (MM_REVID[15:0])	0x0000
JTAG BSDL_ID	JTAGID Register	0x0B7D_F02F
CPU Frequency	MHz	674x DSP 300 MHz
		ARM926 300 MHz

Table 3-1. Characteristics of the OMAP-L137 Processor (continued)

HARDWARE FEATURES		OMAP-L137
Cycle Time	ns	674x DSP 3.3$\overline{3}$ ns
		ARM926 3.3$\overline{3}$ ns
Voltage	Core (V)	1.2 V
	I/O (V)	3.3 V
Package		17 mm x 17 mm, 256-Ball 1 mm pitch, PBGA (ZKB)
Product Status	Product Preview (PP), Advance Information (AI), or Production Data (PD)	PP

3.2 Device Compatibility

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

The C674x DSP core is code-compatible with the C6000™ DSP platform and supports features of both the C64x+ and C67x+ DSP families.

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

PRODUCT PREVIEW

- 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 ARM926ES-J Subsystem in the OMAP-L137 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 OMAP-L137 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.

3.3.7 ARM Memory Mapping

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

To improve security and/or robustness the OMAP-L137 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 3-3 for a detailed top level OMAP-L137 memory map that includes the ARM memory space.

3.4 DSP Subsystem

The DSP Subsystem includes the following features:

- C674x DSP CPU
- 32KB L1 Program (L1P)/Cache (up to 32KB)
- 32KB L1 Data (L1D)/Cache (up to 32KB)
- 256KB Unified Mapped RAM/Cache (L2)
- 1MB Mask-programmable ROM
- Little endian

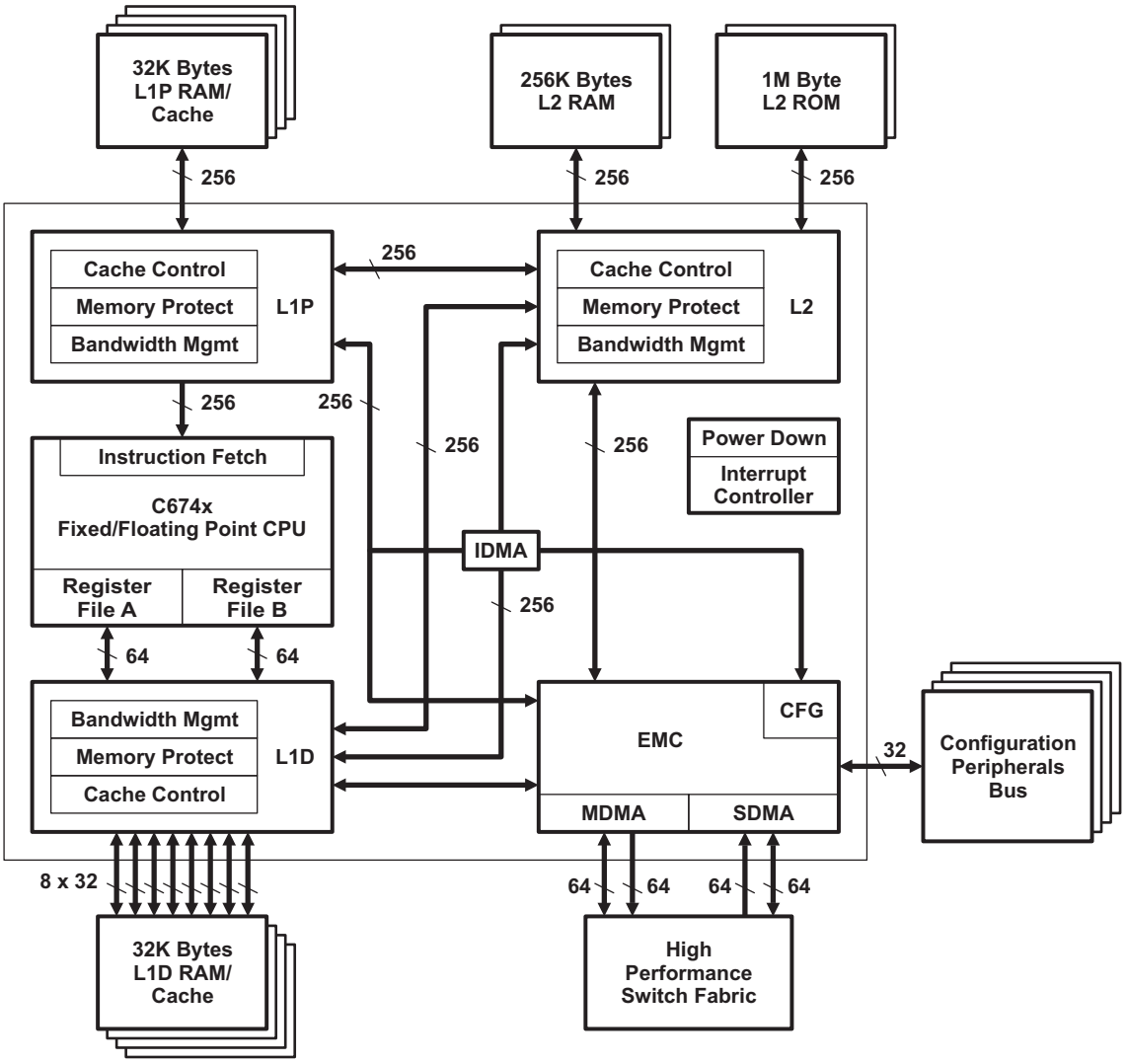


Figure 3-1. C674x Megamodule Block Diagram

3.4.1 C674x DSP CPU Description

The C674x Central Processing Unit (CPU) consists of eight functional units, two register files, and two data paths as shown in Figure 3-2. The two general-purpose register files (A and B) each contain 32 32-bit registers for a total of 64 registers. The general-purpose registers can be used for data or can be data address pointers. The data types supported include packed 8-bit data, packed 16-bit data, 32-bit data, 40-bit data, and 64-bit data. Values larger than 32 bits, such as 40-bit-long or 64-bit-long values are stored in register pairs, with the 32 LSBs of data placed in an even register and the remaining 8 or 32 MSBs in the next upper register (which is always an odd-numbered register).

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

The C674x CPU combines the performance of the C64x+ core with the floating-point capabilities of the C67x core.

Each C674x .M unit can perform one of the following each clock cycle: one 32 x 32 bit multiply, one 16 x 32 bit multiply, two 16 x 16 bit multiplies, two 16 x 32 bit multiplies, two 16 x 16 bit multiplies with add/subtract capabilities, four 8 x 8 bit multiplies, four 8 x 8 bit multiplies with add operations, and four 16 x 16 multiplies with add/subtract capabilities (including a complex multiply). There is also support for Galois field multiplication for 8-bit and 32-bit data. Many communications algorithms such as FFTs and modems require complex multiplication. The complex multiply (CMPY) instruction takes for 16-bit inputs and produces a 32-bit real and a 32-bit imaginary output. There are also complex multiplies with rounding capability that produces one 32-bit packed output that contain 16-bit real and 16-bit imaginary values. The 32 x 32 bit multiply instructions provide the extended precision necessary for high-precision algorithms on a variety of signed and unsigned 32-bit data types.

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

The C674x core enhances the .S unit in several ways. On the previous cores, dual 16-bit MIN2 and MAX2 comparisons were only available on the .L units. On the C674x core they are also available on the .S unit which increases the performance of algorithms that do searching and sorting. Finally, to increase data packing and unpacking throughput, the .S unit allows sustained high performance for the quad 8-bit/16-bit and dual 16-bit instructions. Unpack instructions prepare 8-bit data for parallel 16-bit operations. Pack instructions return parallel results to output precision including saturation support.

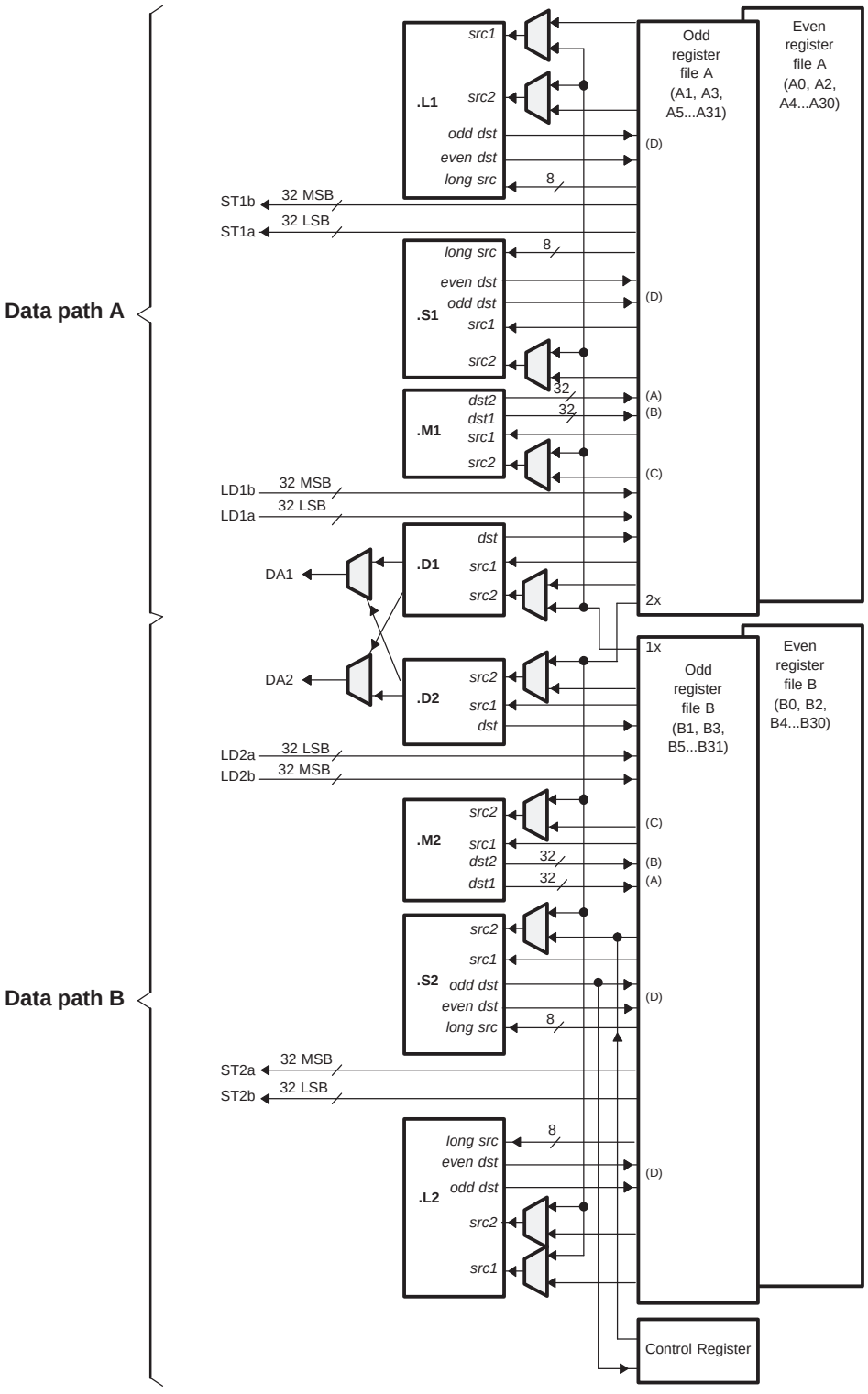
Other new features include:

- **SPLOOP** - A small instruction buffer in the CPU that aids in creation of software pipelining loops where multiple iterations of a loop are executed in parallel. The SPLOOP buffer reduces the code size associated with software pipelining. Furthermore, loops in the SPLOOP buffer are fully interruptible.
- **Compact Instructions** - The native instruction size for the C6000 devices is 32 bits. Many common instructions such as MPY, AND, OR, ADD, and SUB can be expressed as 16 bits if the C674x compiler can restrict the code to use certain registers in the register file. This compression is performed by the code generation tools.
- **Instruction Set Enhancement** - As noted above, there are new instructions such as 32-bit multiplications, complex multiplications, packing, sorting, bit manipulation, and 32-bit Galois field multiplication.
- **Exceptions Handling** - Intended to aid the programmer in isolating bugs. The C674x CPU is able to detect and respond to exceptions, both from internally detected sources (such as illegal op-codes) and from system events (such as a watchdog time expiration).
- **Privilege** - Defines user and supervisor modes of operation, allowing the operating system to give a basic level of protection to sensitive resources. Local memory is divided into multiple pages, each with read, write, and execute permissions.

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

For more details on the C674x CPU and its enhancements over the C64x architecture, see the following documents:

- *TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide* (literature number SPRU732)
- *TMS320C64x Technical Overview* (literature number SPRU395)



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

Figure 3-2. TMS320C674x™ CPU (DSP Core) Data Paths

PRODUCT PREVIEW

3.4.2 DSP Memory Mapping

The DSP memory map is shown in Section 3.5.

By default the DSP also has access to most on and off chip memory areas, with the exception of the ARM RAM, ROM, and AINTC interrupt controller. The DSP also boots first, and must release the ARM from reset before the ARM can execute any code. This allows the DSP (the secure host) to configure the memory and IO protection and ensure security first, before the ARM can even attempt to access any of the device resources.

Additionally, the DSP megamodule includes the capability to limit access to its internal memories through its SDMA port; without needing an external MPU unit.

3.4.2.1 ARM Internal Memories

The DSP does not have access to the ARM internal memory.

3.4.2.2 External Memories

The DSP has access to the following External memories:

- Asynchronous EMIF / SDRAM / NAND / NOR Flash (EMIFA)
- SDRAM (EMIFB)

3.4.2.3 DSP Internal Memories

The DSP has access to the following DSP memories:

- L2 RAM
- L1P RAM
- L1D RAM

3.4.2.4 C674x CPU

The C674x core uses a two-level cache-based architecture. The Level 1 Program cache (L1P) is 32 KB direct mapped cache and the Level 1 Data cache (L1D) is 32 KB 2-way set associated cache. The Level 2 memory/cache (L2) consists of a 256 KB memory space that is shared between program and data space. L2 memory can be configured as mapped memory, cache, or a combination of both.

Table 3-2 shows a memory map of the C674x CPU cache registers for the device.

Table 3-2. C674x Cache Registers

HEX ADDRESS RANGE	REGISTER ACRONYM	DESCRIPTION
0x0184 0000	L2CFG	L2 Cache configuration register
0x0184 0020	L1PCFG	L1P Size Cache configuration register
0x0184 0024	L1PCC	L1P Freeze Mode Cache configuration register
0x0184 0040	L1DCFG	L1D Size Cache configuration register
0x0184 0044	L1DCC	L1D Freeze Mode Cache configuration register
0x0184 0048 - 0x0184 0FFC	-	Reserved
0x0184 1000	EDMAWEIGHT	L2 EDMA access control register
0x0184 1004 - 0x0184 1FFC	-	Reserved
0x0184 2000	L2ALLOC0	L2 allocation register 0
0x0184 2004	L2ALLOC1	L2 allocation register 1
0x0184 2008	L2ALLOC2	L2 allocation register 2
0x0184 200C	L2ALLOC3	L2 allocation register 3
0x0184 2010 - 0x0184 3FFF	-	Reserved
0x0184 4000	L2WBAR	L2 writeback base address register
0x0184 4004	L2WWC	L2 writeback word count register
0x0184 4010	L2WIBAR	L2 writeback invalidate base address register
0x0184 4014	L2WIWC	L2 writeback invalidate word count register

Table 3-2. C674x Cache Registers (continued)

HEX ADDRESS RANGE	REGISTER ACRONYM	DESCRIPTION
0x0184 4018	L2IBAR	L2 invalidate base address register
0x0184 401C	L2IWC	L2 invalidate word count register
0x0184 4020	L1PIBAR	L1P invalidate base address register
0x0184 4024	L1PIWC	L1P invalidate word count register
0x0184 4030	L1DWIBAR	L1D writeback invalidate base address register
0x0184 4034	L1DWIWC	L1D writeback invalidate word count register
0x0184 4038	-	Reserved
0x0184 4040	L1DWBAR	L1D Block Writeback
0x0184 4044	L1DWWC	L1D Block Writeback
0x0184 4048	L1DIBAR	L1D invalidate base address register
0x0184 404C	L1DIWC	L1D invalidate word count register
0x0184 4050 - 0x0184 4FFF	-	Reserved
0x0184 5000	L2WB	L2 writeback all register
0x0184 5004	L2WBINV	L2 writeback invalidate all register
0x0184 5008	L2INV	L2 Global Invalidate without writeback
0x0184 500C - 0x0184 5027	-	Reserved
0x0184 5028	L1PINV	L1P Global Invalidate
0x0184 502C - 0x0184 5039	-	Reserved
0x0184 5040	L1DWB	L1D Global Writeback
0x0184 5044	L1DWBINV	L1D Global Writeback with Invalidate
0x0184 5048	L1DINV	L1D Global Invalidate without writeback
0x0184 8000 – 0x0184 80FF	MAR0 - MAR63	Reserved 0x0000 0000 – 0x3FFF FFFF
0x0184 8100 – 0x0184 817F	MAR64 – MAR95	Memory Attribute Registers for EMIFA SDRAM Data (CS0) 0x4000 0000 – 0x5FFF FFFF
0x0184 8180 – 0x0184 8187	MAR96 - MAR97	Memory Attribute Registers for EMIFA Async Data (CS2) 0x6000 0000 – 0x61FF FFFF
0x0184 8188 – 0x0184 818F	MAR98 – MAR99	Memory Attribute Registers for EMIFA Async Data (CS3) 0x6200 0000 – 0x63FF FFFF
0x0184 8190 – 0x0184 8197	MAR100 – MAR101	Memory Attribute Registers for EMIFA Async Data (CS4) 0x6400 0000 – 0x65FF FFFF
0x0184 8198 – 0x0184 819F	MAR102 – MAR103	Memory Attribute Registers for EMIFA Async Data (CS5) 0x6600 0000 – 0x67FF FFFF
0x0184 81A0 – 0x0184 81FF	MAR104 – MAR127	Reserved 0x6800 0000 – 0x7FFF FFFF
0x0184 8200	MAR128	Memory Attribute Register for Shared RAM 0x8000 0000 – 0x8001 FFFF
		Reserved 0x8002 0000 – 0x81FF FFFF
0x0184 8204 – 0x0184 82FF	MAR129 – MAR191	Reserved 0x8200 0000 – 0xBFFF FFFF
0x0184 8300 – 0x0184 837F	MAR192 – MAR223	Memory Attribute Registers for EMIFB SDRAM Data (CS2) 0xC000 0000 – 0xDFFF FFFF
0x0184 8380 – 0x0184 83FF	MAR224 – MAR255	Reserved 0xE000 0000 – 0xFFFF FFFF

See Table 3-3 for a detailed top level OMAP-L137 memory map that includes the DSP memory space.

3.5 Memory Map Summary

Table 3-3. OMAP-L137 Top Level Memory Map

Start Address	End Address	Size	ARM Mem Map	DSP Mem Map	EDMA Mem Map	Master Peripheral Mem Map	LCDC Mem Map
0x0000 0000	0x006F FFFF	6M + 1024K	-				
0x0070 0000	0x007F FFFF	1024K	-	DSP L2 ROM	-		
0x0080 0000	0x0083 FFFF	256K	-	DSP L2 RAM	-		
0x0084 0000	0x00DF FFFF	5M + 768K	-				
0x00E0 0000	0x00E0 7FFF	32K	-	DSP L1P RAM	-		
0x00E0 8000	0x00EF FFFF	992K	-				
0x00F0 0000	0x00F0 7FFF	32K	-	DSP L1D RAM	-		
0x00F0 8000	0x017F FFFF	8M + 992K	-				
0x0180 0000	0x0180 FFFF	64K	-	DSP Interrupt Controller	-		
0x0181 0000	0x0181 0FFF	4K	-	DSP Powerdown Controller	-		
0x0181 1000	0x0181 1FFF	4K	-	DSP Security ID	-		
0x0181 2000	0x0181 2FFF	4K	-	DSP Revision ID	-		
0x0181 3000	0x0181 FFFF	52K	-	-	-		
0x0182 0000	0x0182 FFFF	64K	-	DSP EMC	-		
0x0183 0000	0x0183 FFFF	64K	-	DSP Internal Reserved	-		
0x0184 0000	0x0184 FFFF	64K	-	DSP Memory System	-		
0x0185 0000	0x01BB FFFF	3M + 600K	-				
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	260K	-				
0x01C0 0000	0x01C0 7FFF	32K	EDMA3 CC				-
0x01C0 8000	0x01C0 83FF	1024	EDMA3 TC0				-
0x01C0 8400	0x01C0 87FF	1024	EDMA3 TC1				-
0x01C0 8800	0x01C0 FFFF	30K	-				
0x01C1 0000	0x01C1 0FFF	4K	PSC 0				-
0x01C1 1000	0x01C1 1FFF	4K	PLL Controller				-
0x01C1 2000	0x01C1 3FFF	8K	-				
0x01C1 4000	0x01C1 4FFF	4K	BootConfig				-
0x01C1 5000	0x01C1 5FFF	4K	-				
0x01C1 6000	0x01C1 6FFF	4K	-				-
0x01C1 7000	0x01C1 7FFF	4K	-				-
0x01C1 8000	0x01C1 FFFF	32K	-				
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	4K	RTC				-

Table 3-3. OMAP-L137 Top Level Memory Map (continued)

Start Address	End Address	Size	ARM Mem Map	DSP Mem Map	EDMA Mem Map	Master Peripheral Mem Map	LCDC Mem Map
0x01C2 4000	0x01C2 4FFF	4K	-				-
0x01C2 5000	0x01C3 FFFF	110K	-				
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	774K	-				
0x01D0 0000	0x01D0 0FFF	4K	McASP 0 Control				-
0x01D0 1000	0x01D0 1FFF	4K	McASP 0 AFIFO Ctrl				-
0x01D0 2000	0x01D0 2FFF	4K	McASP 0 Data				-
0x01D0 3000	0x01D0 3FFF	4K	-				
0x01D0 4000	0x01D0 4FFF	4K	McASP 1 Control				-
0x01D0 5000	0x01D0 5FFF	4K	McASP 1 AFIFO Ctrl				-
0x01D0 6000	0x01D0 6FFF	4K	McASP 1 Data				-
0x01D0 7000	0x01D0 7FFF	4K	-				
0x01D0 8000	0x01D0 8FFF	4K	McASP 2 Control				-
0x01D0 9000	0x01D0 9FFF	4K	McASP 2 AFIFO Ctrl				-
0x01D0 A000	0x01D0 AFFF	4K	McASP 2 Data				-
0x01D0 B000	0x01D0 BFFF	4K	-				
0x01D0 C000	0x01D0 CFFF	4K	UART 1				-
0x01D0 D000	0x01D0 DFFF	4K	UART 2				-
0x01D0 E000	0x01D0 EFFF	4K	-				-
0x01D0 F000	0x01DF FFFF	964K	-				
0x01E0 0000	0x01E0 FFFF	64K	USB0				-
0x01E1 0000	0x01E1 0FFF	4K	UHPI				-
0x01E1 1000	0x01E1 1FFF	4K	-				
0x01E1 2000	0x01E1 2FFF	4K	SPI 1				-
0x01E1 3000	0x01E1 3FFF	4K	LCD Controller				-
0x01E1 4000	0x01E1 4FFF	4K	-				-
0x01E1 5000	0x01E1 5FFF	4K	-				-
0x01E1 6000	0x01E1 FFFF	40K	-				
0x01E2 0000	0x01E2 1FFF	8K	EMAC Control Module RAM				-
0x01E2 2000	0x01E2 2FFF	4K	EMAC Control Module Registers				-
0x01E2 3000	0x01E2 3FFF	4K	EMAC Control Registers				-
0x01E2 4000	0x01E2 4FFF	4K	EMAC MDIO port				-
0x01E2 5000	0x01E2 5FFF	4K	USB1				-
0x01E2 6000	0x01E2 6FFF	4K	GPIO				-
0x01E2 7000	0x01E2 7FFF	4K	PSC 1				-
0x01E2 8000	0x01E2 8FFF	4K	I2C 1				-
0x01E2 9000	0x01E2 9FFF	4K	-				-
0x01E2 A000	0x01EF FFFF	856K	-				
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				-

PRODUCT PREVIEW

Table 3-3. OMAP-L137 Top Level Memory Map (continued)

Start Address	End Address	Size	ARM Mem Map	DSP Mem Map	EDMA Mem Map	Master Peripheral Mem Map	LCDC Mem Map
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	0x01F0 BFFF	4K	-				-
0x01F0 C000	0x116F FFFF	247M + 976K	-				
0x1170 0000	0x117F FFFF	1024K	DSP L2 ROM				-
0x1180 0000	0x1183 FFFF	256K	DSP L2 RAM				-
0x1184 0000	0x11DF FFFF	5M + 768K	-				
0x11E0 0000	0x11E0 7FFF	32K	DSP L1P RAM				-
0x11E0 8000	0x11EF FFFF	992K	-				
0x11F0 0000	0x11F0 7FFF	32K	DSP L1D RAM				-
0x11F0 8000	0x3FFF FFFF	736M + 992K	-				
0x4000 0000	0x5FFF FFFF	512M	EMIFA SDRAM data (CS0)				-
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 Regs				-
0x6800 8000	0x7FFF FFFF	383M + 992K	-				
0x8000 0000	0x8001 FFFF	128K	Shared RAM				-
0x8002 0000	0xAFFF FFFF	767M + 896K	-				
0xB000 0000	0xB000 7FFF	32K	EMIFB Control Regs				
0xB000 8000	0xBFFF FFFF	255M + 992K	-				
0xC000 0000	0xDFFF FFFF	512M	EMIFB SDRAM Data				
0xE000 0000	0xFFFC FFFF	511M + 832K	-				
0xFFFD 0000	0xFFFD FFFF	64K	ARM local ROM	-			
0xFFFE 0000	0xFFFE DFFF	56K	-				
0xFFFE E000	0xFFFE FFFF	8K	ARM Interrupt Controller	-			
0xFFFF 0000	0xFFFF 1FFF	8K	ARM local RAM	-			
0xFFFF 2000	0xFFFF FFFF	56K	-				

3.6 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.

3.6.1 Pin Map (Bottom View)

Figure 3-3 shows the pin assignments for the BGA package.Note that micro-vias are not required. Contact your TI representative for routing recommendations.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
T	V_{SS}	V_{SS}	AXR1[0]/ GP4[0]	AXR1[11]/ GP5[11]	SPI0_CLK/ EQEP1I/ GP5[2]/ BOOT[2]	SPI1_CLK/ EQEP1S/ GP5[7]/ BOOT[7]	EMA_CS[3]/ AMUTE2/ GP2[6]	EMA_CS[0]/ UHPI_HAS/ GP2[4]	EMA_A[0]/ LCD_D[7]/ GP1[0]	EMA_A[4]/ LCD_D[3]/ GP1[4]	EMA_A[8]/ LCD_PCLK/ GP1[8]	EMA_SDCKE/ GP2[0]	EMA_D[0]/ MMCSD_DAT[0]/ UHPI_HD[0]/ GP0[0]/ BOOT[12]	EMA_D[9]/ UHPI_HD[9]/ LCD_D[9]/ GP0[9]	V_{SS}	V_{SS}	T
R	DV_{DD}	AXR1[1]/ GP4[1]	UART0_RXD/ I2C0_SDA/ TM64P0_IN12/ GP5[8]/ BOOT[8]	SPI1_ENA/ UART2_RXD/ GP5[12]	SPI0_ENA/ UART0_CTS/ EQEP0A/ GP5[3]/ BOOT[3]	SPI0_SOMI[0]/ EQEP0I/ GP5[0]/ BOOT[0]	EMA_OE/ UHPI_HDS1/ AXR0[13]/ GP2[7]	EMA_BA[0]/ LCD_D[4]/ GP1[14]	EMA_A[1]/ MMCSD_CLK/ UHPI_HCNTL0/ GP1[1]	EMA_A[5]/ LCD_D[2]/ GP1[5]	EMA_A[9]/ LCD_HSYNC/ GP1[9]	EMA_CLK/ OBSCLK/ AHCLKR2/ GP1[15]	EDMA_D[2]/ MMCSD_DAT[2]/ UHPI_HD[2]/ GP0[2]	EMA_D[10]/ UHPI_HD[10]/ LCD_D[10]/ GP0[10]	EMA_D[1]/ MMCSD_DAT[1]/ UHPI_HD[1]/ GP0[1]	DV_{DD}	R
P	AXR1[3]/ EQEP1A/ GP4[3]	AXR1[2]/ GP4[2]	UART0_TXD/ I2C0_SCL/ TM64P0_OUT12/ GP5[9]/ BOOT[9]	SPI1_SCS[0]/ UART2_TXD/ GP5[13]	SPI1_SOMI[0]/ I2C1_SCL/ GP5[5]/ BOOT[5]	SPI0_SIMO[0]/ EQEP0S/ GP5[1]/ BOOT[1]	EMA_CS[2]/ UHPI_HCS/ GP2[5]/ BOOT[15]	EMA_BA[1]/ LCD_D[5]/ UHPI_HHWIL/ GP1[13]	EMA_A[2]/ MMCSD_CMD/ UHPI_HCNTL1/ GP1[2]	EMA_A[6]/ LCD_D[1]/ GP1[6]	EMA_A[11]/ LCD_AC_ENB_CS/ GP1[11]	EMA_WE_DQM[1]/ UHPI_HDS2/ AXR0[14]/ GP2[8]	EMA_D[4]/ MMCSD_DAT[4]/ UHPI_HD[4]/ GP0[4]	EMA_D[12]/ UHPI_HD[12]/ LCD_D[12]/ GP0[12]	EMA_D[3]/ MMCSD_DAT[3]/ UHPI_HD[3]/ GP0[3]	EMA_D[11]/ UHPI_HD[11]/ LCD_D[11] GP0[11]	P
N	AXR1[5]/ EPWM2B/ GP4[5]	AXR1[4]/ EQEP1B/ GP4[4]	AXR1[10]/ GP5[10]	SPI0_SCS[0]/ UART0_RTS/ EQEP0B/ GP5[4]/ BOOT[4]	SPI1_SIMO[0]/ I2C1_SDA/ GP5[6]/ BOOT[6]	EMA_WAIT[0]/ UHPI_HRDY/ GP2[10]	EMA_RAS/ EMA_CS[5]/ GP2[2]	EMA_A[10]/ LCD_VSYNC/ GP1[10]	EMA_A[3]/ LCD_D[6]/ GP1[3]	EMA_A[7]/ LCD_D[0]/ GP1[7]	EMA_A[12]/ LCD_MCLK/ GP1[12]	EMA_D[8]/ UHPI_HD[8]/ LCD_D[8]/ GP0[8]	EMA_D[6]/ MMCSD_DAT[6]/ UHPI_HD[6]/ GP0[6]	EMA_D[14]/ UHPI_HD[14]/ LCD_D[14]/ GP0[14]	EMA_D[5]/ MMCSD_DAT[5]/ UHPI_HD[5]/ GP0[5]	EMA_D[13]/ UHPI_HD[13]/ LCD_D[13]/ GP0[13]	N
M	AXR1[9]/ GP4[9]	AXR1[8]/ EPWM1A/ GP4[8]	AXR1[7]/ EPWM1B/ GP4[7]	AXR1[6]/ EPWM2A/ GP4[6]	DV_{DD}	V_{SS}	V_{SS}	DV_{DD}	DV_{DD}	V_{SS}	V_{SS}	DV_{DD}	EMA_WE/ UHPI_HRW/ AXR0[12]/ GP2[3]/ BOOT[14]]	EMA_WE_DQM[0]/ UHPI_HINT/ AXR0[15]/ GP2[9]	EMA_D[7]/ MMCSD_DAT[7]/ UHPI_HD[7]/ GP0[7]/ BOOT[13]	EMA_D[15]/ UHPI_HD[15]/ LCD_D[15]/ GP0[15]	M
L	AHCLKR1/ GP4[11]	ACLKR1/ ECAP2/ APWM2/ GP4[12]	AFSR1/ GP4[13]	AMUTE0/ RESETOUT	DV_{DD}	CV_{DD}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	DV_{DD}	DV_{DD}	EMB_CAS	EMB_D[22]	EMB_D[23]	EMA_CAS/ EMA_CS[4]/ GP2[1]	L
K	RTCK/ GP7[14]	AHCLKX1/ EPWM0B/ GP3[14]	ACLKX1/ EPWM0A/ GP3[15]	AFSX1/ EPWMSYNCI/ EPWMSYNCO/ GP4[10]	DV_{DD}	CV_{DD}	CV_{DD}	V_{SS}	V_{SS}	CV_{DD}	CV_{DD}	DV_{DD}	EMB_D[20]	EMB_WE_DQM[0]/ GP5[15]	EMB_WE	EMB_D[21]	K
J	TMS	TDI	TDO	TRST	EMU0/ GP7[15]	CV_{DD}	CV_{DD}	V_{SS}	V_{SS}	CV_{DD}	CV_{DD}	CV_{DD}	EMB_D[5]/ GP6[5]	EMB_D[19]	EMB_D[6]/ GP6[6]	EMB_D[7]/ GP6[7]	J
H	RTC_XI	RTC_XO	TCK	USB0_ VSSA33	USB0_ VDDA33	CV_{DD}	CV_{DD}	V_{SS}	V_{SS}	CV_{DD}	CV_{DD}	CV_{DD}	EMB_D[3]/ GP6[3]	EMB_D[17]	EMB_D[18]	EMB_D[4]/ GP6[4]	H
G	RTC_CV_{DD}	RTC_V_{SS}	RESET	USB0_DM	DV_{DD}	CV_{DD}	CV_{DD}	V_{SS}	V_{SS}	CV_{DD}	CV_{DD}	DV_{DD}	EMB_D[1]/ GP6[1]	EMB_D[31]	EMB_D[16]	EMB_D[2]/ GP6[2]	G
F	OSCOUT	OSCIN	USB0_VSSA	USB0_DP	DV_{DD}	CV_{DD}	RSV1	V_{SS}	V_{SS}	V_{SS}	DV_{DD}	DV_{DD}	EMB_D[15]/ GP6[15]	EMB_D[29]	EMB_D[30]	EMB_D[0]/ GP6[0]	F
E	PLL0_VSSA	OSCVSS	USB0_ VDDA18	USB0_ DRVVBUS/ GP4[15]	DV_{DD}	V_{SS}	V_{SS}	DV_{DD}	DV_{DD}	V_{SS}	V_{SS}	DV_{DD}	EMB_D[13]/ GP6[13]	EMB_D[27]	EMB_D[28]	EMB_D[14]/ GP6[14]	E
D	PLL0_VDDA	USB0_ID	USB0_VBUS	AMUTE1/ EHRPWMTZ/ GP4[14]	AFSX0/ GP2[13]/ BOOT[10]	UART1_TXD/ AXR0[10]/ GP3[10]	AXR0[6]/ RMII_RXER/ ACLKR2/ GP3[6]	AXR0[2]/ RMII_TXEN/ AXR2[3]/ GP3[2]	EMB_CS[0]	EMB_A[0]/ GP7[2]	EMB_A[4]/ GP7[6]	EMB_A[8]/ GP7[10]	EMB_D[9]/ GP6[9]	EMB_D[10]/ GP6[10]	EMB_D[11]/ GP6[11]	EMB_D[12]/ GP6[12]	D
C	USB1_ VDDA33	USB1_ VDDA18	USB0_ VDDA12	AFSR0/ GP3[12]	ACLKX0/ ECAP0/ APWM0/ GP2[12]	UART1_RXD/ AXR0[9]/ GP3[9]	AXR0[5]/ RMII_RXD[1]/ AFSX2/ GP3[5]	AXR0[1]/ RMII_TXD[1]/ ACLKX2/ GP3[1]	EMB_BA[0]/ GP7[1]	EMB_A[1]/ GP7[3]	EMB_A[5]/ GP7[7]	EMB_A[9]/ GP7[11]	EMB_SDCKE	EMB_CLK	EMB_WE_DQM[1]/ GP5[14]	EMB_D[8]/ GP6[8]	C
B	RSV2	V_{SS}	USB1_DM	ACLKR0/ ECAP1/ APWM1/ GP2[15]	AHLKX0/ AHCLKX2/ USB_ REFCLKIN/ GP2[11]	AXR0[8]/ MDIO_D/ GP3[8]	AXR0[4]/ RMII_RXD[0]/ AXR2[1]/ GP3[4]	AXR0[0]/ RMII_TXD[0]/ AFSR2/ GP3[0]	EMB_BA[1]/ GP7[0]	EMB_A[2]/ GP7[4]	EMB_A[6]/ GP7[8]	EMB_A[11]/ GP7[13]	EMB_WE_DQM[2]	EMB_D[25]	EMB_A[12]/ GP3[13]	DV_{DD}	B
A	V_{SS}	V_{SS}	USB1_DP	AHCLKR0/ RMII_MHZ_ 50_CLK/ GP2[14]/ BOOT[11]	AXR0[11]/ AXR2[0]/ GP3[11]	AXR0[7]/ MDIO_CLK/ GP3[7]	AXR0[3]/ RMII_CRS_DV/ AXR2[2]/ GP3[3]	EMB_RAS	EMB_A[10]/ GP7[12]	EMB_A[3]/ GP7[5]	EMB_A[7]/ GP7[9]	EMB_WE_DQM[3]	EMB_D[24]	EMB_D[26]	V_{SS}	V_{SS}	A
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

Figure 3-3. Pin Map (BGA)

PRODUCT PREVIEW

3.7 Terminal Functions

Table 3-4 to Table 3-24 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.

3.7.1 Device Reset and JTAG

Table 3-4. Reset and JTAG Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	DESCRIPTION
RESET				
$\overline{\text{RESET}}$	G3	I		Device reset input
AMUTE0/**$\overline{\text{RESETOUT}}$**	L4	O(3)	IPD	Reset output. Multiplexed with McASP0 mute output.
JTAG				
TMS	J1	I	IPU	JTAG test mode select
TDI	J2	I	IPU	JTAG test data input
TDO	J3	O	IPU	JTAG test data output
TCK	H3	I	IPU	JTAG test clock
$\overline{\text{TRST}}$	J4	I	IPD	JTAG test reset
EMU[0]/GP7[15]	J5	I/O	IPU	Miscellaneous emulation pin.
RSVD/GP7[14]	K1	I/O	IPU	Reserved

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor
(3) Open drain mode for $\overline{\text{RESETOUT}}$ function.

3.7.2 High-Frequency Oscillator and PLL

Table 3-5. High-Frequency Oscillator and PLL Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	DESCRIPTION
1.2-V OSCILLATOR				
OSCIN	F2	I		Oscillator input
OSCOUT	F1	O		Oscillator output
OSCVSS	E2	GND		Oscillator ground (for filter only)
1.2-V PLL				
PLL0_VDDA	D1	PWR		PLL analog V_{DD} (1.2-V filtered supply)
PLL0_VSSA	E1	GND		PLL analog V_{SS} (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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.3 Real-Time Clock and 32-kHz Oscillator

Table 3-6. Real-Time Clock (RTC) and 1.2-V, 32-kHz Oscillator Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	DESCRIPTION
	ZKB			
RTC_CVDD	G1	PWR		RTC module core power (isolated from rest of chip CV_{DD})
RTC_XI	H1	I		Low-frequency (32-kHz) oscillator receiver for real-time clock
RTC_XO	H2	O		Low-frequency (32-kHz) oscillator driver for real-time clock
RTC_V_{ss}	G2	GND		Oscillator ground (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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.4 External Memory Interface A (ASYNC, SDRAM)

Table 3-7. External Memory Interface A (EMIFA) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
EMA_D[15]/UHPI_HD[15]/LCD_D[15]/GP0[15]	M16	I/O	IPD	UHPI, LCD, GPIO	EMIFA data bus
EMA_D[14]/UHPI_HD[14]/LCD_D[14]/GP0[14]	N14	I/O	IPD		
EMA_D[13]/UHPI_HD[13]/LCD_D[13]/GP0[13]	N16	I/O	IPD		
EMA_D[12]/UHPI_HD[12]/LCD_D[12]/GP0[12]	P14	I/O	IPD		
EMA_D[11]/UHPI_HD[11]/LCD_D[11]/GP0[11]	P16	I/O	IPD		
EMA_D[10]/UHPI_HD[10]/LCD_D[10]/GP0[10]	R14	I/O	IPD		
EMA_D[9]/UHPI_HD[9]/LCD_D[9]/GP0[9]	T14	I/O	IPD		
EMA_D[8]/UHPI_HD[8]/LCD_D[8]/GP0[8]	N12	I/O	IPD		
EMA_D[7]/MMCSD_DAT[7]/UHPI_HD[7]/GP0[7]/BOOT[13]	M15	I/O	IPU	MMC/SD, UHPI, GPIO, BOOT	
EMA_D[6]/MMCSD_DAT[6]/UHPI_HD[6]/GP0[6]	N13	I/O	IPU	MMC/SD, UHPI, GPIO	
EMA_D[5]/MMCSD_DAT[5]/UHPI_HD[5]/GP0[5]	N15	I/O	IPU		
EMA_D[4]/MMCSD_DAT[4]/UHPI_HD[4]/GP0[4]	P13	I/O	IPU		
EMA_D[3]/MMCSD_DAT[3]/UHPI_HD[3]/GP0[3]	P15	I/O	IPU		
EMA_D[2]/MMCSD_DAT[2]/UHPI_HD[2]/GP0[2]	R13	I/O	IPU		
EMA_D[1]/MMCSD_DAT[1]/UHPI_HD[1]/GP0[1]	R15	I/O	IPU		
EMA_D[0]/MMCSD_DAT[0]/UHPI_HD[0]/GP0[0]/BOOT[12]	T13	I/O	IPU	MMC/SD, UHPI, GPIO, BOOT	
EMA_A[12]/LCD_MCLK/GP1[12]	N11	O	IPU	LCD, GPIO	EMIFA address bus
EMA_A[11]/$\overline{\text{LCD\_AC\_ENB\_CS}}$/GP1[11]	P11	O	IPU		
EMA_A[10]/LCD_VSYNC/GP1[10]	N8	O	IPU		
EMA_A[9]/LCD_HSYNC/GP1[9]	R11	O	IPU		
EMA_A[8]/LCD_PCLK/GP1[8]	T11	O	IPU		
EMA_A[7]/LCD_D[0]/GP1[7]	N10	O	IPD		
EMA_A[6]/LCD_D[1]/GP1[6]	P10	O	IPD		
EMA_A[5]/LCD_D[2]/GP1[5]	R10	O	IPD		
EMA_A[4]/LCD_D[3]/GP1[4]	T10	O	IPD		
EMA_A[3]/LCD_D[6]/GP1[3]	N9	O	IPD		

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

PRODUCT PREVIEW

Table 3-7. External Memory Interface A (EMIFA) Terminal Functions (continued)

SIGNAL NAME	PIN NO	TYPE[1]	PULL[2]	MUXED	DESCRIPTION
	ZKB				
EMA_A[2]/MMCSD_CMD/UHPI_HCNTL1/GP1[2]	P9	O	IPU	MMCSD, UHPI, GPIO	EMIFA address bus.
EMA_A[1]/MMCSD_CLK/UHPI_HCNTL0/GP1[1]	R9	O	IPU		
EMA_A[0]/LCD_D[7]/GP1[0]	T9	O	IPD	LCD, GPIO	
EMA_BA[1]/LCD_D[5]/UHPI_HHWIL/GP1[13]	P8	O	IPU	LCD, UHPI, GPIO	EMIFA bank address
EMA_BA[0]/LCD_D[4]/GP1[14]	R8	O	IPU	LCD, GPIO	
EMA_CLK/AHCLKR2/GP1[15]	R12	O	IPU	McASP2, GPIO	EMIFA clock.
EMA_SDCKE/GP2[0]	T12	O	IPU	GPIO	EMIFA SDRAM clock enable.
$\overline{\text{EMA\_RAS}}$/$\overline{\text{EMA\_CS[5]}}$/GP2[2]	N7	O	IPU	EMIF A chip select, GPIO	EMIFA SDRAM row address strobe.
$\overline{\text{EMA\_CAS}}$/$\overline{\text{EMA\_CS[4]}}$/GP2[1]	L16	O	IPU		EMIFA SDRAM column address strobe.
$\overline{\text{EMA\_RAS}}$/**$\overline{\text{EMA\_CS[5]}}$**/GP2[2]	N7	O	IPU	EMIF A SDRAM, GPIO	EMIFA Async Chip Select
$\overline{\text{EMA\_CAS}}$/**$\overline{\text{EMA\_CS[4]}}$**/GP2[1]	L16	O	IPU		
$\overline{\text{EMA\_CS[3]}}$/AMUTE2/GP2[6]	T7	O	IPU	McASP2, GPIO	
$\overline{\text{EMA\_CS[2]}}$/$\overline{\text{UHPI\_HCS}}$/GP2[5]/BOOT[15]	P7	O	IPU	UHPI, GPIO, BOOT	
$\overline{\text{EMA\_CS[0]}}$/$\overline{\text{UHPI\_HAS}}$/GP2[4]	T8	O	IPU	UHPI, GPIO	
$\overline{\text{EMA\_WE}}$/UHPI_HR$\overline{\text{W}}$/AXR0[12]/GP2[3]/BOOT[14]	M13	O	IPU	UHPI, MCASP0, GPIO, BOOT	EMIFA SDRAM write enable.
$\overline{\text{EMA\_WE\_DQM[1]}}$/$\overline{\text{UHPI\_HDS2}}$/AXR0[14]/GP2[8]	P12	O	IPU	UHPI, McASP, GPIO	EMIFA write enable/data mask for EMA_D[15:8]
$\overline{\text{EMA\_WE\_DQM[0]}}$/$\overline{\text{UHPI\_HINT}}$/AXR0[15]/GP2[9]	M14	O	IPU		EMIFA write enable/data mask for EMA_D[7:0].
$\overline{\text{EMA\_OE}}$/$\overline{\text{UHPI\_HDS1}}$/AXR0[13]/GP2[7]	R7	O	IPU	UHPI, McASP0, GPIO	EMIFA output enable.
EMA_WAIT[0]/$\overline{\text{UHPI\_HRDY}}$/GP2[10]	N6	I	IPU	UHPI, GPIO	EMIFA wait input/interrupt.

PRODUCT PREVIEW

3.7.5 External Memory Interface B (only SDRAM)

Table 3-8. External Memory Interface B (EMIFB) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
EMB_D[31]	G14	O	IPD		EMIFB SDRAM data bus.
EMB_D[30]	F15	O	IPD		
EMB_D[29]	F14	O	IPD		
EMB_D[28]	E15	O	IPD		
EMB_D[27]	E14	O	IPD		
EMB_D[26]	A14	O	IPD		
EMB_D[25]	B14	O	IPD		
EMB_D[24]	A13	O	IPD		
EMB_D[23]	L15	O	IPD		
EMB_D[22]	L14	O	IPD		
EMB_D[21]	K16	O	IPD		
EMB_D[20]	K13	O	IPD		
EMB_D[19]	J14	O	IPD		
EMB_D[18]	H15	O	IPD		
EMB_D[17]	H14	O	IPD		
EMB_D[16]	G15	O	IPD		
EMB_D[15]/GP6[15]	F13	I/O	IPD	GPIO	
EMB_D[14]/GP6[14]	E16	I/O	IPD		
EMB_D[13]/GP6[13]	E13	I/O	IPD		
EMB_D[12]/GP6[12]	D16	I/O	IPD		
EMB_D[11]/GP6[11]	D15	I/O	IPD		
EMB_D[10]/GP6[10]	D14	I/O	IPD		
EMB_D[9]/GP6[9]	D13	I/O	IPD		
EMB_D[8]/GP6[8]	C16	I/O	IPD		
EMB_D[7]/GP6[7]	J16	I/O	IPD		
EMB_D[6]/GP6[6]	J15	I/O	IPD		
EMB_D[5]/GP6[5]	J13	I/O	IPD		
EMB_D[4]/GP6[4]	H16	I/O	IPD		
EMB_D[3]/GP6[3]	H13	I/O	IPD		
EMB_D[2]/GP6[2]	G16	I/O	IPD		
EMB_D[1]/GP6[1]	G13	I/O	IPD		
EMB_D[0]/GP6[0]	F16	I/O	IPD		
EMB_A[12]/GP3[13]	B15	O	IPD	GPIO	EMIFB SDRAM row/column address bus.
EMB_A[11]/GP7[13]	B12	O	IPD		
EMB_A[10]/GP7[12]	A9	O	IPD		
EMB_A[9]/GP7[11]	C12	O	IPD		
EMB_A[8]/GP7[10]	D12	O	IPD		
EMB_A[7]/GP7[9]	A11	O	IPD		
EMB_A[6]/GP7[8]	B11	O	IPD		
EMB_A[5]/GP7[7]	C11	O	IPD		

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

Table 3-8. External Memory Interface B (EMIFB) Terminal Functions (continued)

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
EMB_A[4]/GP7[6]	D11	O	IPD	GPIO	EMIFB SDRAM row/column address.
EMB_A[3]/GP7[5]	A10	O	IPD		
EMB_A[2]/GP7[4]	B10	O	IPD		
EMB_A[1]/GP7[3]	C10	O	IPD		
EMB_A[0]/GP7[2]	D10	O	IPD		
EMB_BA[1]/GP7[0]	B9	O	IPU		EMIFB SDRAM bank address.
EMB_BA[0]/GP7[1]	C9	O	IPU		
EMB_CLK	C14	O	IPU		EMIF SDRAM clock.
EMB_SDCKE	C13	I/O	IPU		EMIFB SDRAM clock enable.
$\overline{\text{EMB\_WE}}$	K15	O	IPU		EMIFB write enable
$\overline{\text{EMB\_RAS}}$	A8	O	IPU		EMIFB SDRAM row address strobe.
$\overline{\text{EMB\_CAS}}$	L13	O	IPU		EMIFB column address strobe.
$\overline{\text{EMB\_CS[0]}}$	D9	O	IPU		EMIFB SDRAM chip select 0.
$\overline{\text{EMB\_WE\_DQM[3]}}$	A12	O	IPU		EMIFB write enable/data mask for EMB_D.
$\overline{\text{EMB\_WE\_DQM[2]}}$	B13	O	IPU		
$\overline{\text{EMB_WE_DQM[1]}}/GP5[14]	C15	O	IPU	GPIO	
$\overline{\text{EMB_WE_DQM[0]}}/GP5[15]	K14	O	IPU		

3.7.6 Serial Peripheral Interface Modules (SPI0, SPI1)

Table 3-9. Serial Peripheral Interface (SPI) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
SPI0					
SPI0_SCS[0]/$\overline{\text{UART0\_RTS}}$/EQEP0B/GP5[4]/BOOT[4]	N4	I/O	IPU	UART0, EQEP0B, GPIO, BOOT	SPI0 chip select.
$\overline{\text{SPI0\_ENA}}$/$\overline{\text{UART0\_CTS}}$/EQEP0A/GP5[3]/BOOT[3]	R5	I/O	IPU	UART0, EQEP0A, GPIO, BOOT	SPI0 enable.
SPI0_CLK/EQEP1I/GP5[2]/BOOT[2]	T5	I/O	IPD	eQEP1, GPIO, BOOT	SPI0 clock.
SPI0_SIMO[0]/EQEP0S/GP5[1]/BOOT[1]	P6	I/O	IPD	eQEP0, GPIO, BOOT	SPI0 data slave-in-master-out.
SPI0_SOMI[0]/EQEP0I/GP5[0]/BOOT[0]	R6	I/O	IPD		SPI0 data slave-out-master-in.
SPI1					
$\overline{\text{SPI1_SCS[0]}}/UART2_TXD/GP5[13]	P4	I/O	IPU	UART2, GPIO	SPI1 chip select.
$\overline{\text{SPI1_ENA}}/UART2_RXD/GP5[12]	R4	I/O	IPU		SPI1 enable.
SPI1_CLK/EQEP1S/GP5[7]/BOOT[7]	T6	I/O	IPD	eQEP1, GPIO, BOOT	SPI1 clock.
SPI1_SIMO[0]/I2C1_SDA/GP5[6]/BOOT[6]	N5	I/O	IPU	I2C1, GPIO, BOOT	SPI1 data slave-in-master-out.
SPI1_SOMI[0]/I2C1_SCL/GP5[5]/BOOT[5]	P5	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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

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

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

Table 3-10. Enhanced Capture Module (eCAP) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
eCAP0					
ACLKX0/**ECAP0/APWM0**/GP2[12]	C5	I/O	IPD	McASP0, GPIO	enhanced capture 0 input or auxiliary PWM 0 output.
eCAP1					
ACLKR0/**ECAP1/APWM1**/GP2[15]	B4	I/O	IPD	McASP0, GPIO	enhanced capture 1 input or auxiliary PWM 1 output.
eCAP2					
ACLKR1/**ECAP2/APWM2**/GP4[12]	L2	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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.8 Enhanced Pulse Width Modulators (eHRPWM0, eHRPWM1, eHRPWM2)

Table 3-11. Enhanced Pulse Width Modulator (eHRPWM) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
eHRPWM0					
ACLKX1/**EPWM0A**/GP3[15]	K3	I/O	IPD	McASP1, GPIO	eHRPWM0 A output (with high-resolution).
AHCLKX1/**EPWM0B**/GP3[14]	K2	I/O	IPD		eHRPWM0 B output.
AMUTE1/**EPWMTZ**/GP4[14]	D4	I/O	IPD	McASP1, eHRPWM1, GPIO, eHRPWM2	eHRPWM0 trip zone input.
AFSX1/**EPWMSYNCI**/**EPWMSYNCO**/GP4[10]	K4	I/O	IPD	McASP1, eHRPWM0, GPIO	Sync input to eHRPWM0 module or sync output to external PWM.
eHRPWM1					
AXR1[8]/**EPWM1A**/GP4[8]	M2	I/O	IPD	McASP1, GPIO	eHRPWM1 A output (with high-resolution).
AXR1[7]/**EPWM1B**/GP4[7]	M3	I/O	IPD		eHRPWM1 B output.
AMUTE1/**EPWMTZ**/GP4[14]	D4	I/O	IPD	McASP1, eHRPWM1, GPIO, eHRPWM2	eHRPWM1 trip zone input.
eHRPWM2					

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

PRODUCT PREVIEW

Table 3-11. Enhanced Pulse Width Modulator (eHRPWM) Terminal Functions (continued)

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
AXR1[6]/**EPWM2A**/GP4[6]	M4	I/O	IPD	McASP1, GPIO	eHRPWM2 A output (with high-resolution).
AXR1[5]/**EPWM2B**/GP4[5]	N1	I/O	IPD		eHRPWM2 B output.
AMUTE1/**EPWMTZ**/GP4[14]	D4	I/O	IPD	McASP1, eHRPWM1, GPIO, eHRPWM2	eHRPWM2 trip zone input.

3.7.9 Enhanced Quadrature Encoder Pulse Module (eQEP)

Table 3-12. Enhanced Quadrature Encoder Pulse Module (eQEP) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
eQEP0					
$\overline{\text{SPI0\_ENA}}$/$\overline{\text{UART0\_CTS}}$/**EQEP0A**/GP5[3]/BOOT[3]	R5	I	IPU	SPIO, UART0, GPIO, BOOT	EQEP0A quadrature input.
$\overline{\text{SPI0\_SCS[0]}}$/$\overline{\text{UART0\_RTS}}$/**EQEP0B**/GP5[4]/BOOT[4]	N4	I	IPU		EQEP0B quadrature input.
SPI0_SOMI[0]/**EQEP0I**/GP5[0]/BOOT[0]	R6	I	IPD	SPI1, GPIO, BOOT	eQEP0 index.
SPI0_SIMO[0]/**EQEP0S**/GP5[1]/BOOT[1]	P6	I	IPD		eQEP0 strobe.
eQEP1					
AXR1[3]/**EQEP1A**/GP4[3]	P1	I	IPD	McASP1, GPIO	eQEP1 quadrature input.
AXR1[4]/**EQEP1B**/GP4[4]	N2	I	IPD	McASP1, GPIO	eQEP1 quadrature input.
SPI0_CLK/**EQEP1I**/GP5[2]/BOOT[2]	T5	I	IPD	SPI1, GPIO, BOOT	eQEP1 index.
SPI1_CLK/**EQEP1S**/GP5[7]/BOOT[7]	T6	I	IPD		eQEP1 strobe.

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

PRODUCT PREVIEW

3.7.10 Boot

Table 3-13. Boot Mode Selection Terminal Functions(1)

SIGNAL NAME	PIN NO ZKB	TYPE(2)	PULL(3)	MUXED	DESCRIPTION
EMA_CS[2]/UHPI_HCS/GP2[5]/**BOOT[15]**	P7	I	IPU	EMIFA, UHPI, GPIO	Boot Mode Selection Pins
EMA_WE/UHPI_HRW/AXR0[12]/GP2[3]/**BOOT[14]**	M13	I	IPU	EMIFA, UHPI, McASP0, GPIO	
EMA_D[7]/MMCSD_DAT[7]/UHPI_HD[7]/GP0[7]/**BOOT[13]**	M15	I	IPU	EMIFA, MMC/SD, UHPI, GPIO	
EMA_D[0]/MMCSD_DAT[0]/UHPI_HD[0]/GP0[0]/**BOOT[12]**	T13	I	IPU		
AHCLKR0/RMII_MHZ_50_CLK/GP2[14]/**BOOT[11]**	A4	I	IPD	McASP0, EMAC, GPIO	
AFSX0/GP2[13]/**BOOT[10]**	D5	I	IPD	McASP0, GPIO	
UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/**BOOT[9]**	P3	I	IPU	UART0, I2C0, Timer0, GPIO	
UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/**BOOT[8]**	R3	I	IPU	UART0, I2C0, Timer0, GPIO	
SPI1_CLK/EQEP1S/GP5[7]/**BOOT[7]**	T6	I	IPD	SPI1, eQEP1, GPIO	
SPI1_SIMO[0]/I2C1_SDA/GP5[6]/**BOOT[6]**	N5	I	IPU	SPI1, I2C1, GPIO	
SPI1_SOMI[0]/I2C1_SCL/GP5[5]/**BOOT[5]**	P5	I	IPU		
SPI0_SCS[0]/UART0_RTS/EQEP0B/GP5[4]/**BOOT[4]**	N4	I	IPU	SPI0, UART0, eQEP0, GPIO	
SPI0_ENA/UART0_CTS/EQEP0A/GP5[3]/**BOOT[3]**	R5	I	IPU	SPI0, UART0, eQEP0, GPIO	
SPI0_CLK/EQEP1I/GP5[2]/**BOOT[2]**	T5	I	IPD	SPIO, eQEP1, GPIO	
SPI0_SIMO[0]/EQEP0S/GP5[1]/**BOOT[1]**	P6	I	IPD	SPI0, eQEP0, GPIO	
SPI0_SOMI[0]/EQEP0I/GP5[0]/**BOOT[0]**	R6	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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(3) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.11 Universal Asynchronous Receiver/Transmitters (UART0, UART1, UART2)

Table 3-14. Universal Asynchronous Receiver/Transmitter (UART) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
UART0					
UART0_RXD/I2C0_SDA/TM64P0_IN12/GP5[8]/BOOT[8]	R3	I	IPU	I2C0, BOOT, Timer0, GPIO,	UART0 receive data.
UART0_TXD/I2C0_SCL/TM64P0_OUT12/GP5[9]/BOOT[9]	P3	O	IPU	I2C0, Timer0, GPIO, BOOT	UART0 transmit data.
SPI0_SCS[0]/**UART0_RTS**/EQEP0B/GP5[4]/BOOT[4]	N4	O	IPU	SPIO, eQEP0, GPIO, BOOT	UART0 ready-to-send output
SPI0_ENA/**UART0_CTS**/EQEP0A/GP5[3]/BOOT[3]	R5	I	IPU		UART0 clear-to-send input

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

Table 3-14. Universal Asynchronous Receiver/Transmitter (UART) Terminal Functions (continued)

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
UART1					
UART1_RXD/AXR0[9]/GP3[9]	C6	I	IPD	McASP0, GPIO	UART1 receive data.
UART1_TXD/AXR0[10]/GP3[10]	D6	O	IPD		UART1 transmit data.
UART2					
$\overline{\text{SPI1\_ENA}}$/**UART2_RXD**/GP5[12]	R4	I	IPU	SPI1, GPIO	UART2 receive data.
$\overline{\text{SPI1\_SCS[0]}}$/**UART2_TXD**/GP5[13]	P4	O	IPU		UART2 transmit data.

3.7.12 Inter-Integrated Circuit Modules(I2C0, I2C1)

Table 3-15. Inter-Integrated Circuit (I2C) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
I2C0					
UART0_RXD/**I2C0_SDA**/TM64P0_IN12/GP5[8]/BOOT[8]	R3	I/O	IPU	UART0, Timer0, GPIO, BOOT	I2C0 serial data.
UART0_TXD/**I2C0_SCL**/TM64P0_OUT12/GP5[9]/BOOT[9]	P3	I/O	IPU	UART0, Timer0, GPIO, BOOT	I2C0 serial clock.
I2C1					
SPI1_SIMO[0]/**I2C1_SDA**/GP5[6]/BOOT[6]	N5	I/O	IPU	SPI1, GPIO, BOOT	I2C1 serial data.
SPI1_SOMI[0]/**I2C1_SCL**/GP5[5]/BOOT[5]	P5	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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.13 Timers

Table 3-16. Timers Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
TIMER0					
UART0_RXD/I2C0_SDA/**TM64P0_IN12**/GP5[8]/BOOT[8]	R3	I	IPU	UART0, I2C0, GPIO, BOOT	Timer0 lower input.
UART0_TXD/I2C0_SCL/**TM64P0_OUT12**/GP5[9]/BOOT[9]	P3	O	IPU		Timer0 lower output
TIMER1 (Watchdog)					
No external pins. The Timer1 peripheral signals 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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

PRODUCT PREVIEW

3.7.14 Universal Host-Port Interface (UHPI)

Table 3-17. Universal Host-Port Interface (UHPI) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
EMA_D[15]/**UHPI_HD[15]**/LCD_D[15]/GP0[15]	M16	I/O	IPD	EMIFA, LCD, GPIO	UHPI data bus.
EMA_D[14]/**UHPI_HD[14]**/LCD_D[14]/GP0[14]	N14	I/O	IPD		
EMA_D[13]/**UHPI_HD[13]**/LCD_D[13]/GP0[13]	N16	I/O	IPD		
EMA_D[12]/**UHPI_HD[12]**/LCD_D[12]/GP0[12]	P14	I/O	IPD		
EMA_D[11]/**UHPI_HD[11]**/LCD_D[11]/GP0[11]	P16	I/O	IPD		
EMA_D[10]/**UHPI_HD[10]**/LCD_D[10]/GP0[10]	R14	I/O	IPD		
EMA_D[9]/**UHPI_HD[9]**/LCD_D[9]/GP0[9]	T14	I/O	IPD		
EMA_D[8]/**UHPI_HD[8]**/LCD_D[8]/GP0[8]	N12	I/O	IPD		
EMA_D[7]/MMCSD_DAT[7]/**UHPI_HD[7]**/GP0[7]/ BOOT[13]	M15	I/O	IPU	EMIFA, MMC/SD, GPIO, BOOT	
EMA_D[6]/MMCSD_DAT[6]/**UHPI_HD[6]**/GP0[6]	N13	I/O	IPU	EMIFA, MMC/SD, GPIO	
EMA_D[5]/MMCSD_DAT[5]/**UHPI_HD[5]**/GP0[5]	N15	I/O	IPU		
EMA_D[4]/MMCSD_DAT[4]/**UHPI_HD[4]**/GP0[4]	P13	I/O	IPU		
EMA_D[3]/MMCSD_DAT[3]/**UHPI_HD[3]**/GP0[3]	P15	I/O	IPU		
EMA_D[2]/MMCSD_DAT[2]/**UHPI_HD[2]**/GP0[2]	R13	I/O	IPU		
EMA_D[1]/MMCSD_DAT[1]/**UHPI_HD[1]**/GP0[1]	R15	I/O	IPU		
EMA_D[0]/MMCSD_DAT[0]/**UHPI_HD[0]**/GP0[0]/ BOOT[12]	T13	I/O	IPU	EMIFA, MMC/SD, GPIO, BOOT	
EMA_A[2]/MMCSD_CMD/**UHPI_HCNTL1**/GP1[2]	P9	I/O	IPU	EMIFA, MMCSD_CMD, GPIO	UHPI access control.
EMA_A[1]/MMCSD_CLK/**UHPI_HCNTL0**/GP1[1]	R9	I/O	IPU		
EMA_BA[1]/LCD_D[5]/**UHPI_HHWIL**/GP1[13]	P8	I/O	IPU	EMIFA, LCD, GPIO	UHPI half-word identification control.
$\overline{\text{EMA\_WE}}$/**UHPI_HR$\overline{\text{W}}$**/AXR0[12]/GP2[3]/BOOT[14]	M13	I/O	IPU	EMIFA, McASP, GPIO, BOOT	UHPI read/write.
$\overline{\text{EMA\_CS[2]}}$/**$\overline{\text{UHPI\_HCS}}$**/GP2[5]/BOOT[15]	P7	I/O	IPU	EMIFA, GPIO, BOOT	UHPI chip select.
$\overline{\text{EMA\_WE\_DQM[1]}}$/**$\overline{\text{UHPI\_HDS2}}$**/AXR0[14]/GP2[8]	P12	I/O	IPU	EMIFA, McASP0, GPIO	UHPI data strobe.
$\overline{\text{EMA\_OE}}$/**$\overline{\text{UHPI\_HDS1}}$**/AXR0[13]/GP2[7]	R7	I/O	IPU		
$\overline{\text{EMA\_WE\_DQM[0]}}$/**UHPI_HINT**/AXR0[15]/GP2[9]	M14	I/O	IPU		UHPI host interrupt.
EMA_WAIT[0]/**$\overline{\text{UHPI\_HRDY}}$**/GP2[10]	N6	I/O	IPU	EMIFA, GPIO	UHPI ready.
$\overline{\text{EMA\_CS[0]}}$/**$\overline{\text{UHPI\_HAS}}$**/GP2[4]	T8	I/O	IPU		UHPI address strobe.

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.15 Multichannel Audio Serial Ports (McASP0, McASP1, McASP2)

Table 3-18. Multichannel Audio Serial Ports (McASPs) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
McASP0					
EMA_WE_DQM[0]/UHPI_HINT/**AXR0[15]**/GP2[9]	M14	I/O	IPU	EMIFA, UHPI, GPIO	McASP0 serial data.
EMA_WE_DQM[1]/UHPI_HDS2/**AXR0[14]**/GP2[8]	P12	I/O	IPU		
EMA_OE/UHPI_HDS1/**AXR0[13]**/GP2[7]	R7	I/O	IPU		
EMA_WE/UHPI_HRW/**AXR0[12]**/GP2[3]/BOOT[14]	M13	I/O	IPU	EMIFA, UHPI, GPIO, BOOT	
AXR0[11]/AXR2[0]/GP3[11]	A5	I/O	IPD	McASP2, GPIO	
AXR0[10]/GP3[10]	D6	I/O	IPD	GPIO	
AXR0[9]/GP3[9]	C6	I/O	IPD	GPIO	
AXR0[8]/MDIO_D/GP3[8]	B6	I/O	IPU	MDIO, GPIO	
AXR0[7]/MDIO_CLK/GP3[7]	A6	I/O	IPD		
AXR0[6]/RMII_RXER/ACLKR2/GP3[6]	D7	I/O	IPD	EMAC, McASP2, GPIO	
AXR0[5]/RMII_RXD[1]/AFSX2/GP3[5]	C7	I/O	IPD		
AXR0[4]/RMII_RXD[0]/AXR2[1]/GP3[4]	B7	I/O	IPD		
AXR0[3]/RMII_CRS_DV/AXR2[2]/GP3[3]	A7	I/O	IPD		
AXR0[2]/RMII_TXEN/AXR2[3]/GP3[2]	D8	I/O	IPD		
AXR0[1]/RMII_TXD[1]/ACLKX2/GP3[1]	C8	I/O	IPD		
AXR0[0]/RMII_TXD[0]/AFSR2/GP3[0]	B8	I/O	IPD		
AHCLKX0/AHCLKX2/USB_REFCLKIN/GP2[11]	B5	I/O	IPD	McASP2, USB, GPIO	McASP1 transmit master clock.
ACLKX0/ECAP0/APWM0/GP2[12]	C5	I/O	IPD	eCAP0, GPIO	McASP0 transmit bit clock.
AFSX0/GP2[13]/BOOT[10]	D5	I/O	IPD	GPIO, BOOT	McASP0 transmit frame sync.
AHCLKR0/RMII_MHZ_50_CLK/GP2[14]/BOOT[11]	A4	I/O	IPD	EMAC, GPIO, BOOT	McASP0 receive master clock.
ACLKR0/ECAP1/APWM1/GP2[15]	B4	I/O	IPD	eCAP1, GPIO	McASP0 receive bit clock.
AFSR0/GP3[12]	C4	I/O	IPD	GPIO	McASP0 receive frame sync.
AMUTE0/RESETOUT	L4	I/O	IPD	RESETOUT	McASP0 mute 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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

Table 3-18. Multichannel Audio Serial Ports (McASPs) Terminal Functions (continued)

SIGNAL NAME	PIN NO ZKB	TYPE[1]	PULL[2]	MUXED	DESCRIPTION
McASP1					
AXR1[11]/GP5[11]	T4	I/O	IPU	GPIO	McASP1 serial data.
AXR1[10]/GP5[10]	N3	I/O	IPU		
AXR1[9]/GP4[9]	M1	I/O	IPD		
AXR1[8]/EPWM1A/GP4[8]	M2	I/O	IPD	eHRPWM1 A, GPIO	
AXR1[7]/EPWM1B/GP4[7]	M3	I/O	IPD	eHRPWM1 B, GPIO	
AXR1[6]/EPWM2A/GP4[6]	M4	I/O	IPD	eHRPWM2 A, GPIO	
AXR1[5]/EPWM2B/GP4[5]	N1	I/O	IPD	eHRPWM2 B, GPIO	
AXR1[4]/EQEP1B/GP4[4]	N2	I/O	IPD	eQEP1, GPIO	
AXR1[3]/EQEP1A/GP4[3]	P1	I/O	IPD		
AXR1[2]/GP4[2]	P2	I/O	IPD	GPIO	
AXR1[1]/GP4[1]	R2	I/O	IPD		
AXR1[0]/GP4[0]	T3	I/O	IPD		
AHCLKX1/EPWM0B/GP3[14]	K2	I/O	IPD	eHRPWM0, GPIO	McASP1 transmit master clock.
ACLKX1/EPWM0A/GP3[15]	K3	I/O	IPD	eHRPWM0, GPIO	McASP1 transmit bit clock.
AFSX1/EPWMSYNCI/EPWMSYNCO/GP4[10]	K4	I/O	IPD	eHRPWM0, GPIO	McASP1 transmit frame sync.
AHCLKR1/GP4[11]	L1	I/O	IPD	GPIO	McASP1 receive master clock.
ACLKR1/ECAP2/APWM2/GP4[12]	L2	I/O	IPD	eCAP2, GPIO	McASP1 receive bit clock.
AFSR1/GP4[13]	L3	I/O	IPD	GPIO	McASP1 receive frame sync.
AMUTE1/EPWMTZ/GP4[14]	D4	I/O	IPD	eHRPWM0, eHRPWM1, GPIO, eHRPWM2	McASP1 mute output.
McASP2					
AXR0[2]/RMII_TXEN/**AXR2[3]**/GP3[2]	D8	I/O	IPD	McASP0, EMAC, GPIO	McASP2 serial data.
AXR0[3]/RMII_CRS_DV/**AXR2[2]**/GP3[3]	A7	I/O	IPD		
AXR0[4]/RMII_RXD[0]/**AXR2[1]**/GP3[4]	B7	I/O	IPD		
UART1_TXD/**AXR2[0]**AXR0[11]/GP3[11]	A5	I/O	IPD	UART1, McASP0, GPIO	
AHCLKX0/**AHCLKX2**/USB_REFCLKIN/GP2[11]	B5	I/O	IPD	McASP0, USB, GPIO	McASP2 transmit master clock.
AXR0[1]/RMII_TXD[1]/**ACLKX2**/GP3[1]	C8	I/O	IPD		McASP2 transmit bit clock.
AXR0[5]/RMII_RXD[1]/**AFSX2**/GP3[5]	C7	I/O	IPD	McASP0, EMAC, GPIO	McASP2 transmit frame sync.
EMA_CLK/OBSCLK/**AHCLKR2**/GP1[15]	R12	I/O	IPU	EMIFA, GPIO	McASP2 receive master clock.
AXR0[6]/RMII_RXER/**ACLKR2**/GP3[6]	D7	I/O	IPD	McASP0, EMAC, GPIO	McASP2 receive bit clock.
$\overline{\text{EMA\_CS[3]}}$/**AMUTE2**/GP2[6]	T7	I/O	IPU	EMIFA, GPIO	McASP2 mute output.

PRODUCT PREVIEW

3.7.16 Universal Serial Bus Modules (USB0, USB1)

Table 3-19. Universal Serial Bus (USB) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
USB0 2.0 OTG (USB0)					
USB0_DM	G4	A		NA	USB0 PHY data minus
USB0_DP	F4	A		NA	USB0 PHY data plus
USB0_VDDA33	H5	PWR		NA	USB0 PHY 3.3-V supply
USB0_VSSA33	H4	PWR		NA	USB0 PHY 3.3-V supply reference
USB0_VDDA18	E3	PWR		NA	USB0 PHY 1.8-V supply input
USB0_VDDA12	C3	PWR		NA	USB0 PHY 1.2-V LDO output for bypass cap
USB0_VSSA	F3	PWR		NA	USB0 PHY 1.8-V and 1.2-V supply reference
USB0_ID	D2	A		NA	USB0 PHY identification (mini-A or mini-B plug)
USB0_VBUS	D3	A		NA	USB0 bus voltage
USB0_DRVVBUS/GP4[15]	E4	0	IPD	GPIO	USB0 controller VBUS control output. Multiplexed with GPIO bank 4 pin 15.
AHCLKX0/AHCLKX2/**USB_REFCLKIN**/ GP2[11]	B5	I	IPD		USB_REFCLKIN. Optional clock input.
USB1 1.1 OHCI (USB1)					
USB1_DM	B3	A		NA	USB1 PHY data minus
USB1_DP	A3	A		NA	USB1 PHY data plus
USB1_VDDA33	C1	PWR		NA	USB1 PHY 3.3-V supply
USB1_VDDA18	C2	PWR		NA	USB1 PHY 1.8-V supply
AHCLKX0/AHCLKX2/**USB_REFCLKIN**/ GP2[11]	B5	I	IPD	NA	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: For multiplexed pins where functions have different types (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.17 Ethernet Media Access Controller (EMAC)

Table 3-20. Ethernet Media Access Controller (EMAC) Terminal Functions

SIGNAL NAME	PIN NO ZKB	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
RMII					
AHCLKR0/**RMII_MHZ_50_CLK**/GP2[14]/BOOT[11]	A4	I/O	IPD	McASP0, GPIO, BOOT	EMAC 50-MHz clock input or 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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

Table 3-20. Ethernet Media Access Controller (EMAC) Terminal Functions (continued)

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
AXR0[6]/**RMII_RXER**/ACLKR2/GP3[6]	D7	I	IPD	McASP0, McASP2, GPIO	EMAC RMII receiver error.
AXR0[5]/**RMII_RXD[1]**/AFSX2/GP3[5]	C7	I	IPD		EMAC RMII receive data.
AXR0[4]/**RMII_RXD[0]**/AXR2[1]/GP3[4]	B7	I	IPD		
AXR0[3]/**RMII_CRS_DV/**AXR2[2]/GP3[3]	A7	I	IPD		EMAC RMII carrier sense data valid.
AXR0[2]/**RMII_TXEN**/AXR2[3]/GP3[2]	D8	O	IPD		EMAC RMII transmit enable.
AXR0[1]/**RMII_TXD[1]/**ACLKX2/GP3[1]	C8	O	IPD		EMAC RMII trasmit data.
AXR0[0]/**RMII_TXD[0]**/AFSR2/GP3[0]	B8	O	IPD		
MDIO					
AXR0[8]/**MDIO_D/**GP3[8]	B6	I/O	IPU	McASP0, GPIO	MDIO serial data.
AXR0[7]/**MDIO_CLK**/GP3[7]	A6	O	IPD		MDIO clock

3.7.18 Multimedia Card/Secure Digital (MMC/SD)

Table 3-21. Multimedia Card/Secure Digital (MMC/SD) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
EMA_A[1]/**MMCSD_CLK**/UHPI_HCNTL0/GP1[1]	R9	O	IPU	EMIFA, UHPI, GPIO	MMCSD Clock.
EMA_A[2]/**MMCSD_CMD**/UHPI_HCNTL1/GP1[2]	P9	I/O	IPU		MMCSD Command.
EMA_D[7]/**MMCSD_DAT[7]**/UHPI_HD[7]/GP0[7]/BOOT[13]	M15	I/O	IPU	EMIFA, UHPI, GPIO, BOOT	MMC/SD data.
EMA_D[6]/**MMCSD_DAT[6]**/UHPI_HD[6]/GP0[6]	N13	I/O	IPU	EMIFA, UHPI, GPIO	
EMA_D[5]/**MMCSD_DAT[5]/**UHPI_HD[5]/GP0[5]	N15	I/O	IPU		
EMA_D[4]/**MMCSD_DAT[4]**/UHPI_HD[4]/GP0[4]	P13	I/O	IPU		
EMA_D[3]/**MMCSD_DAT[3]**/UHPI_HD[3]/GP0[3]	P15	I/O	IPU		
EMA_D[2]/**MMCSD_DAT[2]**/UHPI_HD[2]/GP0[2]	R13	I/O	IPU		
EMA_D[1]/**MMCSD_DAT[1]**/UHPI_HD[1]/GP0[1]	R15	I/O	IPU		
EMA_D[0]/**MMCSD_DAT[0]**/UHPI_HD[0]/GP0[0]/BOOT[12]	T13	I/O	IPU	EMIFA, UHPI, GPIO, BOOT	

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.19 Liquid Crystal Display Controller(LCD)

Table 3-22. Liquid Crystal Display Controller (LCD) Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	PULL(2)	MUXED	DESCRIPTION
	ZKB				
EMA_D[15]/UHPI_HD[15]/**LCD_D[15]**/GP0[15]	M16	I/O	IPD	EMIFA, UHPI, GPIO	LCD data bus.
EMA_D[14]/UHPI_HD[14]/**LCD_D[14]**/GP0[14]	N14	I/O	IPD		
EMA_D[13]/UHPI_HD[13]/**LCD_D[13]**/GP0[13]	N16	I/O	IPD		
EMA_D[12]/UHPI_HD[12]/**LCD_D[12]**/GP0[12]	P14	I/O	IPD		
EMA_D[11]/UHPI_HD[11]/**LCD_D[11]**/GP0[11]	P16	I/O	IPD		
EMA_D[10]/UHPI_HD[10]/**LCD_D[10]**/GP0[10]	R14	I/O	IPD		
EMA_D[9]/UHPI_HD[9]/**LCD_D[9]**/GP0[9]	T14	I/O	IPD		
EMA_D[8]/UHPI_HD[8]/**LCD_D[8]**/GP0[8]	N12	I/O	IPD		
EMA_A[0]/**LCD_D[7]**/GP1[0]	T9	I/O	IPD	EMIFA, GPIO	
EMA_A[3]/**LCD_D[6]**/GP1[3]	N9	I/O	IPD		
EMA_BA[1]/**LCD_D[5]**/UHPI_HHWIL/GP1[13]	P8	I/O	IPU	EMIFA, UHPI, GPIO	
EMA_BA[0]/**LCD_D[4]**/GP1[14]	R8	I/O	IPU	EMIFA, GPIO	LCD data bus.
EMA_A[4]/**LCD_D[3]**/GP1[4]	T10	I/O	IPD		
EMA_A[5]/**LCD_D[2]**/GP1[5]	R10	I/O	IPD		
EMA_A[6]/**LCD_D[1]**/GP1[6]	P10	I/O	IPD		
EMA_A[7]/**LCD_D[0]**/GP1[7]	N10	I/O	IPD		
EMA_A[8]/**LCD_PCLK**/GP1[8]	T11	O	IPU		LCD pixel clock.
EMA_A[9]/**LCD_HSYNC**/GP1[9]	R11	O	IPU		LCD horizontal sync.
EMA_A[10]/**LCD_VSYNC**/GP1[10]	N8	O	IPU		LCD vertical sync.
EMA_A[11]/**$\overline{\text{LCD\_AC\_ENB\_CS}}$**/GP1[11]	P11	O	IPU		LCD AC bias enable chip select.
EMA_A[12]/**LCD_MCLK**/GP1[12]	N11	O	IPU		LCD memory clock.

(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 (ie., input versus output), the table reflects the pin function direction for that particular peripheral.
(2) IPD = Internal Pulldown resistor, IPU = Internal Pullup resistor

3.7.20 Reserved

Table 3-23. Reserved Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	DESCRIPTION
	ZKB		
RSV1	F7	PWR	Reserved. (Leave unconnected, ***do not*** connect to power or ground.)
RSV2	B1	PWR	Reserved. For proper device operation, this pin **must** be tied directly to CV_{DD}.

(1) PWR = Supply voltage.

3.7.21 Supply and Ground

Table 3-24. Supply and Ground Terminal Functions

SIGNAL NAME	PIN NO	TYPE(1)	DESCRIPTION
	ZKB		
CVDD (Core supply)	F6,G6, G7, G10, G11,H6, H7, H10, H11, H12,J6, J7, J10, J11, J12, K6, K7, K10, K11,L6	PWR	1.2-V core supply voltage pins
DVDD (I/O supply)	B16, E5, E8, E9, E12, F5, F11, F12, G5, G12, K5, K12, L5, L11, L12, M5, M8, M9, M12, R1, R16	PWR	3.3-V I/O supply voltage pins.
VSS (Ground)	A1, A2, A15, A16, B2, E6, E7, E10, E11, F8, F9, F10, G8, G9, H8, H9, J8, J9, K8, K9, L7, L8, L9, L10, M6, M7, M10, M11, T1, T2, T15, T16	GND	Ground pins.

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

PRODUCT PREVIEW

4 Device Configuration

4.1 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
- Source of emulation suspend signal (from either ARM or DSP) of peripherals supporting this function
- Control of on-chip inter-processor interrupts for signaling between ARM and DSP

Since the SYSCFG peripheral controls global operation of the device, its registers are protected against erroneous accesses by several mechanisms:

- A special key sequence must be written to KICK0, KICK1 registers before any other registers are writeable.
  - Unlock sequence: write 0x83e70b13 to KICK0, then write 0x95A4F1E0 to KICK1
  - SYSCFG remains unlocked after the unlock sequence until locked again.
  - Any number of accesses may be performed while the module is unlocked
  - Locking the module is accomplished by writing any other value to either KICK0 or KICK1
- Additionally, many registers are accessible only by a host (ARM or DSP) 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

Offset	Acronym	Register Description	Access
0x01C1 4000	REVID	Revision Identification Register	—
0x01C14008 – 0x01C1 4014	DIEIDR0-DIEDR3	Device Identification Register 0 - 3	—
0x01C1 4020	BOOTCFG	Boot Configuration 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 4110-0x01C1 4118	MSTPRI0-MSTPRI2	Master Priority 0-2 Registers	Privileged mode

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

Offset	Acronym	Register Description	Access
0x01C1 4120-0x01C1 416C	PINMUX0-PINMUX19	Pin Multiplexing Control 0-19 Registers	Privileged mode
0x01C1 4170	SUSPSRC	Suspend Source Register	Privileged mode
0x01C1 4174	CHIPSIG	Chip Signal Register	—
0x01C1 4178	CHIPSIG_CLR	Chip Signal Clear Register	—
0x01C1 417C-0x01C1 418C	CFGCHIP0-CFGCHIP4	Chip Configuration 0-4 Registers	Privileged mode

4.2 Pin Multiplexing Control Registers

Device level pin multiplexing is controlled by registers PINMUX0 - PINMUX19 in the SYSCFG module.

For the OMAP-L13x device family, pin multiplexing can be controlled on a pin-by-pin basis. Each pin that is multiplexed with several different functions has a corresponding 4-bit field in one of the PINMUX registers.

Pin multiplexing selects which of several peripheral pin functions controls the pin's IO buffer **output** data and **output enable** values only. The default pin multiplexing control for almost every pin is to select 'none' of the peripheral functions in which case the pin's IO buffer is held tri-stated.

Note that the **input** from each pin is always routed to **all** of the peripherals that share the pin; the PINMUX registers have no effect on input from a pin. This feature allows a pin such as **AHCLKX0/AHCLKX2/USB_REFCLKIN/GP2[11]** to be used as both the McASP0 **AHCLKX0** (output) pin, and the McASP2 **AHCLKX2** master clock (output) pin simultaneously.

Section 4.2.1 through Section 4.2.20 contain the specific bit field definitions for the PINMUX registers on the OMAP-L137 devices.

4.2.1 PINMUX0 Register Definition (Address 0x01C1 4120)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX0[31:28]				PINMUX0[27:24]				PINMUX0[23:20]				PINMUX0[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX0[15:12]				PINMUX0[11:8]				PINMUX0[7:4]				PINMUX0[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-1. PINMUX0 Register Bit Layout

Table 4-2. Field Descriptions for PINMUX0

Bits	Field	ZKB Ball	Description
31:28	PINMUX0[31:28]	K15	$\overline{\text{EMB\_WE}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_WE}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
27:24	PINMUX0[27:24]	A8	$\overline{\text{EMB\_RAS}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_RAS}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
23:20	PINMUX0[23:20]	L13	$\overline{\text{EMB\_CAS}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_CAS}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
19:16	PINMUX0[19:16]	D9	$\overline{\text{EMB\_CS[0]}}$ Control 0000 = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_CS[0]}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
15:12	PINMUX0[15:12]	C14	EMB_CLK Control 0000 [Default] = Pin is tri-stated. 0001 = Reserved - Behavior is Undefined 0010 = Selects Output Function EMB_CLK 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
11:8	PINMUX0[11:8]	C13	EMB_SDCKE Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_SDCKE 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
7:4	PINMUX0[7:4]	J5	EMU[0] / GP7[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function GP7[15] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function EMU[0] other = Reserved - Behavior is Undefined.
3:0	PINMUX0[3:0]	K1	RTCK / GP7[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function GP7[14] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function RTCK other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.2 PINMUX1 Register Definition (Address 0x01C1 4124)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX1[31:28]				PINMUX1[27:24]				PINMUX1[23:20]				PINMUX1[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX1[15:12]				PINMUX1[11:8]				PINMUX1[7:4]				PINMUX1[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-2. PINMUX1 Register Bit Layout

Table 4-3. Field Descriptions for PINMUX1

Bits	Field	ZKB Ball	Description
31:28	PINMUX1[31:28]	C11	EMB_A[5] / GP7[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[5] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[7] other = Reserved - Behavior is Undefined.
27:24	PINMUX1[27:24]	D11	EMB_A[4] / GP7[6] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[4] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[6] other = Reserved - Behavior is Undefined.
23:20	PINMUX1[23:20]	A10	EMB_A[3] / GP7[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[3] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[5] other = Reserved - Behavior is Undefined.
19:16	PINMUX1[19:16]	B10	EMB_A[2] / GP7[4] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_A[2] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[4] other = Reserved - Behavior is Undefined.
15:12	PINMUX1[15:12]	C10	EMB_A[1] / GP7[3] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[1] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[3] other = Reserved - Behavior is Undefined.
11:8	PINMUX1[11:8]	D10	EMB_A[0] / GP7[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[0] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[2] other = Reserved - Behavior is Undefined.
7:4	PINMUX1[7:4]	C9	EMB_BA[0] / GP7[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_BA[0] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[1] other = Reserved - Behavior is Undefined.
3:0	PINMUX1[3:0]	B9	EMB_BA[1] / GP7[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_BA[1] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[0] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.3 PINMUX2 Register Definition (Address 0x01C1 4128)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX2[31:28]				PINMUX2[27:24]				PINMUX2[23:20]				PINMUX2[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX2[15:12]				PINMUX2[11:8]				PINMUX2[7:4]				PINMUX2[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-3. PINMUX2 Register Bit Layout

Table 4-4. Field Descriptions for PINMUX2

Bits	Field	ZKB Ball	Description
31:28	PINMUX2[31:28]	G14	EMB_D[31] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[31] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
27:24	PINMUX2[27:24]	B15	EMB_A[12] / GP3[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[12] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[13] other = Reserved - Behavior is Undefined.
23:20	PINMUX2[23:20]	B12	EMB_A[11] / GP7[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[11] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[13] other = Reserved - Behavior is Undefined.
19:16	PINMUX2[19:16]	A9	EMB_A[10]/GP7[12] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_A[10] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[12] other = Reserved - Behavior is Undefined.
15:12	PINMUX2[15:12]	C12	EMB_A[9] / GP7[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[9] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[11] other = Reserved - Behavior is Undefined.
11:8	PINMUX2[11:8]	D12	EMB_A[8] / GP7[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[8] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[10] other = Reserved - Behavior is Undefined.
7:4	PINMUX2[7:4]	A11	EMB_A[7] / GP7[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[7] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[9] other = Reserved - Behavior is Undefined.
3:0	PINMUX2[3:0]	B11	EMB_A[6] / GP7[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_A[6] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP7[8] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.4 PINMUX3 Register Definition (Address 0x01C1 412C)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX3[31:28]				PINMUX3[27:24]				PINMUX3[23:20]				PINMUX3[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX3[15:12]				PINMUX3[11:8]				PINMUX3[7:4]				PINMUX3[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-4. PINMUX3 Register Bit Layout

Table 4-5. Field Descriptions for PINMUX3

Bits	Field	ZKB Ball	Description
31:28	PINMUX3[31:28]	L15	EMB_D[23] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[23] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
27:24	PINMUX3[27:24]	A13	EMB_D[24] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[24] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
23:20	PINMUX3[23:20]	B14	EMB_D[25] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[25] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
19:16	PINMUX3[19:16]	A14	EMB_D[26] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_D[26] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
15:12	PINMUX3[15:12]	E14	EMB_D[27] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[27] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
11:8	PINMUX3[11:8]	E15	EMB_D[28] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[28] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
7:4	PINMUX3[7:4]	F14	EMB_D[29] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[29] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
3:0	PINMUX3[3:0]	F15	EMB_D[30] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[30] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.

4.2.5 PINMUX4 Register Definition (Address 0x01C1 4130)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX4[31:28]				PINMUX4[27:24]				PINMUX4[23:20]				PINMUX4[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX4[15:12]				PINMUX4[11:8]				PINMUX4[7:4]				PINMUX4[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-5. PINMUX4 Register Bit Layout

Table 4-6. Field Descriptions for PINMUX4

Bits	Field	ZKB Ball	Description
31:28	PINMUX4[31:28]	A12	$\overline{\text{EMB\_WE\_DQM[3]}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_WE\_DQM[3]}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
27:24	PINMUX4[27:24]	G15	EMB_D[16] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[16] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
23:20	PINMUX4[23:20]	H14	EMB_D[17] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[17] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
19:16	PINMUX4[19:16]	H15	EMB_D[18] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_D[18] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
15:12	PINMUX4[15:12]	J14	EMB_D[19] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[19] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
11:8	PINMUX4[11:8]	K13	EMB_D[20] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[20] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
7:4	PINMUX4[7:4]	K16	EMB_D[21] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[21] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.
3:0	PINMUX4[3:0]	L14	EMB_D[22] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[22] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.6 PINMUX5 Register Definition (Address 0x01C1 4134)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX5[31:28]				PINMUX5[27:24]				PINMUX5[23:20]				PINMUX5[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX5[15:12]				PINMUX5[11:8]				PINMUX5[7:4]				PINMUX5[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-6. PINMUX5 Register Bit Layout

Table 4-7. Field Descriptions for PINMUX5

Bits	Field	ZKB Ball	Description
31:28	PINMUX5[31:28]	J15	EMB_D[6] / GP6[6] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[6] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[6] other = Reserved - Behavior is Undefined.
27:24	PINMUX5[27:24]	J13	EMB_D[5] / GP6[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[5] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[5] other = Reserved - Behavior is Undefined.
23:20	PINMUX5[23:20]	H16	EMB_D[4] / GP6[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[4] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[4] other = Reserved - Behavior is Undefined.
19:16	PINMUX5[19:16]	H13	EMB_D[3] / GP6[3] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_D[3] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[3] other = Reserved - Behavior is Undefined.
15:12	PINMUX5[15:12]	G16	EMB_D[2] / GP6[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[2] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[2] other = Reserved - Behavior is Undefined.
11:8	PINMUX5[11:8]	G13	EMB_D[1] / GP6[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[1] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[1] other = Reserved - Behavior is Undefined.
7:4	PINMUX5[7:4]	F16	EMB_D[0] / GP6[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[0] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[0] other = Reserved - Behavior is Undefined.
3:0	PINMUX5[3:0]	B13	$\overline{\text{EMB\_WE\_DQM[2]}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_WE\_DQM[2]}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Reserved - Behavior is Undefined other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.7 PINMUX6 Register Definition (Address 0x01C1 4138)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX6[31:28]				PINMUX6[27:24]				PINMUX6[23:20]				PINMUX6[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX6[15:12]				PINMUX6[11:8]				PINMUX6[7:4]				PINMUX6[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-7. PINMUX6 Register Bit Layout

Table 4-8. Field Descriptions for PINMUX6

Bits	Field	ZKB Ball	Description
31:28	PINMUX6[31:28]	E16	EMB_D[14] / GP6[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[14] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[14] other = Reserved - Behavior is Undefined.
27:24	PINMUX6[27:24]	E13	EMB_D[13] / GP6[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[13] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[13] other = Reserved - Behavior is Undefined.
23:20	PINMUX6[23:20]	D16	EMB_D[12] / GP6[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[12] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[12] other = Reserved - Behavior is Undefined.
19:16	PINMUX6[19:16]	D15	EMB_D[11] / GP6[11] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMB_D[11] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[11] other = Reserved - Behavior is Undefined.
15:12	PINMUX6[15:12]	D14	EMB_D[10] / GP6[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[10] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[10] other = Reserved - Behavior is Undefined.
11:8	PINMUX6[11:8]	D13	EMB_D[9] / GP6[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[9] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[9] other = Reserved - Behavior is Undefined.
7:4	PINMUX6[7:4]	C16	EMB_D[8] / GP6[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[8] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[8] other = Reserved - Behavior is Undefined.
3:0	PINMUX6[3:0]	J16	EMB_D[7] / GP6[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[7] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[7] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.8 PINMUX7 Register Definition (Address 0x01C1 413C)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX7[31:28]				PINMUX7[27:24]				PINMUX7[23:20]				PINMUX7[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX7[15:12]				PINMUX7[11:8]				PINMUX7[7:4]				PINMUX7[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-8. PINMUX7 Register Bit Layout

Table 4-9. Field Descriptions for PINMUX7

Bits	Field	ZKB Ball	Description
31:28	PINMUX7[31:28]	N4	$\overline{\text{SPI0\_SCS[0]}}$ / $\overline{\text{UART0\_RTS}}$ / EQEP0B / GP5[4] / BOOT[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{SPI0\_SCS[0]}}$ 0010 = Selects Output Function $\overline{\text{UART0\_RTS}}$ 0100 = Selects Output Function EQEP0B 1000 = Selects Output Function GP5[4] other = Reserved - Behavior is Undefined.
27:24	PINMUX7[27:24]	R5	$\overline{\text{SPI0\_ENA}}$ / $\overline{\text{UART0\_CTS}}$ / EQEP0A / GP5[3] / BOOT[3] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{SPI0\_ENA}}$ 0010 = Selects Output Function $\overline{\text{UART0\_CTS}}$ 0100 = Selects Output Function EQEP0A 1000 = Selects Output Function GP5[3] other = Reserved - Behavior is Undefined.
23:20	PINMUX7[23:20]	T5	SPI0_CLK / EQEP1I / GP5[2] / BOOT[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function SPI0_CLK 0010 = Selects Output Function EQEP1I 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[2] other = Reserved - Behavior is Undefined.
19:16	PINMUX7[19:16]	P6	SPIO_SOMI[0] / EQEP0S / GP5[1] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function SPIO_SOMI[0] 0010 = Selects Output Function EQEP0S 0100 = Reserved - Behavior is Undefined. 1000 = Selects Output Function GP5[1] other = Reserved - Behavior is Undefined.
15:12	PINMUX7[15:12]	R6	SPI0_SOMI[0] / EQEP0I / GP5[0] / BOOT[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function SPI0_SOMI[0] 0010 = Selects Output Function EQEP0I 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[0] other = Reserved - Behavior is Undefined.
11:8	PINMUX7[11:8]	K14	$\overline{\text{EMB\_WE\_DQM[0]}}$ / GP5[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_WE\_DQM[0]}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[15] other = Reserved - Behavior is Undefined.
7:4	PINMUX7[7:4]	C15	$\overline{\text{EMB\_WE\_DQM[1]}}$ / GP5[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMB\_WE\_DQM[1]}}$ 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[14] other = Reserved - Behavior is Undefined.
3:0	PINMUX7[3:0]	F13	EMB_D[15] / GP6[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMB_D[15] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP6[15] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.9 PINMUX8 Register Definition (Address 0x01C1 4140)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX8[31:28]				PINMUX8[27:24]				PINMUX8[23:20]				PINMUX8[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX8[15:12]				PINMUX8[11:8]				PINMUX8[7:4]				PINMUX8[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-9. PINMUX8 Register Bit Layout

Table 4-10. Field Descriptions for PINMUX8

Bits	Field	ZKB Ball	Description
31:28	PINMUX8[31:28]	R4	$\overline{\text{SPI1\_ENA}}$ / UART2_RXD / GP5[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{SPI1\_ENA}}$ 0010 = Selects Output Function UART2_RXD 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[12] other = Reserved - Behavior is Undefined.
27:24	PINMUX8[27:24]	T4	AXR1[11] / GP5[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[11] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[11] other = Reserved - Behavior is Undefined.
23:20	PINMUX8[23:20]	N3	AXR1[10] / GP5[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[10] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[10] other = Reserved - Behavior is Undefined.
19:16	PINMUX8[19:16]	P3	UART0_TXD / I2C0_SCL / TM64P0_OUT12 / GP5[9] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function UART0_TXD 0010 = Selects Output Function I2C0_SCL 0100 = Selects Output Function TM64P0_OUT12 1000 = Selects Output Function GP5[9] other = Reserved - Behavior is Undefined.
15:12	PINMUX8[15:12]	R3	UART0_RXD / I2C0_SDA / TM64P0_IN12 / GP5[8] / BOOT[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function UART0_RXD 0010 = Selects Output Function I2C0_SDA 0100 = Selects Output Function TM64P0_IN12 1000 = Selects Output Function GP5[8] other = Reserved - Behavior is Undefined.
11:8	PINMUX8[11:8]	T6	SPI1_CLK / EQEP1S / GP5[7] / BOOT[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function SPI1_CLK 0010 = Selects Output Function EQEP1S 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[7] other = Reserved - Behavior is Undefined.
7:4	PINMUX8[7:4]	N5	SPI1_SIMO[0] / I2C1_SDA / GP5[6] / BOOT[6] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function SPI1_SIMO[0] 0010 = Selects Output Function I2C1_SDA 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[6] other = Reserved - Behavior is Undefined.
3:0	PINMUX8[3:0]	P5	SPI1_SOMI[0] / I2C1_SCL / GP5[5] / BOOT[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function SPI1_SOMI[0] 0010 = Selects Output Function I2C1_SCL 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[5] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.10 PINMUX9 Register Definition (Address 0x01C1 4144)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX9[31:28]				PINMUX9[27:24]				PINMUX9[23:20]				PINMUX9[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX9[15:12]				PINMUX9[11:8]				PINMUX9[7:4]				PINMUX9[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-10. PINMUX9 Register Bit Layout

Table 4-11. Field Descriptions for PINMUX9

Bits	Field	ZKB Ball	Description
31:28	PINMUX9[31:28]	C4	AFSR0 / GP3[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AFSR0 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[12] other = Reserved - Behavior is Undefined.
27:24	PINMUX9[27:24]	B4	ACLKR0 / ECAP1 / APWM1 / GP2[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function ACLKR0 0010 = Selects Output Function ECAP1 / APWM1 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[15] other = Reserved - Behavior is Undefined.
23:20	PINMUX9[23:20]	A4	AHCLKR0 / RMII_MHZ_50_CLK / GP2[14] / BOOT[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AHCLKR0 0010 = Selects Output Function RMII_MHZ_50_CLK 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[14] other = Reserved - Behavior is Undefined.
19:16	PINMUX9[19:16]	D5	AFSX0 / GP2[13] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function AFSX0 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[13] other = Reserved - Behavior is Undefined.
15:12	PINMUX9[15:12]	C5	ACLKX0 / ECAP0 / APWM0 / GP2[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function ACLKX0 0010 = Selects Output Function ECAP0 / APWM0 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[12] other = Reserved - Behavior is Undefined.
11:8	PINMUX9[11:8]	B5	AHCLKX0 / AHCLKX2 / USB_REFCLKIN / GP2[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AHCLKX0 0010 = Selects Output Function AHCLKX2 0100 = Selects Output Function USB_REFCLKIN 1000 = Selects Output Function GP2[11] other = Reserved - Behavior is Undefined.
7:4	PINMUX9[7:4]	E4	USB0_DRVVBUS / GP4[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function USB0_DRVVBUS 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[15] other = Reserved - Behavior is Undefined.
3:0	PINMUX9[3:0]	P4	$\overline{\text{SPI1\_SCS[0]}}$ / UART2_TXD / GP5[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{SPI1\_SCS[0]}}$ 0010 = Selects Output Function UART2_TXD 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP5[13] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.11 PINMUX10 Register Definition (Address 0x01C1 4148)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX10[31:28]				PINMUX10[27:24]				PINMUX10[23:20]				PINMUX10[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX10[15:12]				PINMUX10[11:8]				PINMUX10[7:4]				PINMUX10[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-11. PINMUX10 Register Bit Layout

Table 4-12. Field Descriptions for PINMUX10

Bits	Field	ZKB Ball	Description
31:28	PINMUX10[31:28]	D7	AXR0[6] / RMII_RXER / ACLKR2 / GP3[6] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[6] 0010 = Selects Output Function RMII_RXER 0100 = Selects Output Function ACLKR2 1000 = Selects Output Function GP3[6] other = Reserved - Behavior is Undefined.
27:24	PINMUX10[27:24]	C7	AXR0[5] / RMII_RXD[1] / AFSX2 / GP3[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[5] 0010 = Selects Output Function RMII_RXD[1] 0100 = Selects Output Function AFSX2 1000 = Selects Output Function GP3[5] other = Reserved - Behavior is Undefined.
23:20	PINMUX10[23:20]	B7	AXR0[4] / RMII_RXD[0] / AXR2[1] / GP3[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[4] 0010 = Selects Output Function RMII_RXD[0] 0100 = Selects Output Function AXR2[1] 1000 = Selects Output Function GP3[4] other = Reserved - Behavior is Undefined.
19:16	PINMUX10[19:16]	A7	AXR0[3] / RMII_CRS_DV / AXR2[2] / GP3[3] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function AXR0[3] 0010 = Selects Output Function RMII_CRS_DV 0100 = Selects Output Function AXR2[2] 1000 = Selects Output Function GP3[3] other = Reserved - Behavior is Undefined.
15:12	PINMUX10[15:12]	D8	AXR0[2] / RMII_TXEN / AXR2[3] / GP3[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[2] 0010 = Selects Output Function RMII_TXEN 0100 = Selects Output Function AXR2[3] 1000 = Selects Output Function GP3[2] other = Reserved - Behavior is Undefined.
11:8	PINMUX10[11:8]	C8	AXR0[1] / RMII_TXD[1] / ACLKX2 / GP3[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[1] 0010 = Selects Output Function RMII_TXD[1] 0100 = Selects Output Function ACLKX2 1000 = Selects Output Function GP3[1] other = Reserved - Behavior is Undefined.
7:4	PINMUX10[7:4]	B8	AXR0[0] / RMII_TXD[0] / AFSR2 / GP3[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[0] 0010 = Selects Output Function RMII_TXD[0] 0100 = Selects Output Function AFSR2 1000 = Selects Output Function GP3[0] other = Reserved - Behavior is Undefined.
3:0	PINMUX10[3:0]	L4	AMUTE0 / $\overline{\text{RESETOUT}}$ Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AMUTE0 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function $\overline{\text{RESETOUT}}$ other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.12 PINMUX11 Register Definition (Address 0x01C1 414C)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX11[31:28]				PINMUX11[27:24]				PINMUX11[23:20]				PINMUX11[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX11[15:12]				PINMUX11[11:8]				PINMUX11[7:4]				PINMUX11[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-12. PINMUX11 Register Bit Layout

Table 4-13. Field Descriptions for PINMUX11

Bits	Field	ZKB Ball	Description
31:28	PINMUX11[31:28]	K4	AFSX1 / EPWMSYNCI / EPWMSYNC0 / GP4[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AFSX1 0010 = Selects Output Function EPWMSYNCI 0100 = Selects Output Function EPWMSYNC0 1000 = Selects Output Function GP4[10] other = Reserved - Behavior is Undefined.
27:24	PINMUX11[27:24]	K3	ACLKX1 / EPWM0A / GP3[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function ACLKX1 0010 = Selects Output Function EPWM0A 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[15] other = Reserved - Behavior is Undefined.
23:20	PINMUX11[23:20]	K2	AHCLKX1 / EPWM0B / GP3[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AHCLKX1 0010 = Selects Output Function EPWM0B 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[14] other = Reserved - Behavior is Undefined.
19:16	PINMUX11[19:16]	A5	AXR0[11] / AXR2[0] / GP3[11] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function AXR0[11] 0010 = Reserved - Behavior is Undefined. 0100 = Selects Output Function AXR2[0] 1000 = Selects Output Function GP3[11] other = Reserved - Behavior is Undefined.
15:12	PINMUX11[15:12]	D6	UART1_TXD / AXR0[10] / GP3[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function UART1_TXD 0010 = Selects Output Function AXR0[10] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[10] other = Reserved - Behavior is Undefined.
11:8	PINMUX11[11:8]	C6	UART1_RXD / AXR0[9] / GP3[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function UART1_RXD 0010 = Selects Output Function AXR0[9] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[9] other = Reserved - Behavior is Undefined.
7:4	PINMUX11[7:4]	B6	AXR0[8] / MDIO_D / GP3[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[8] 0010 = Selects Output Function MDIO_D 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[8] other = Reserved - Behavior is Undefined.
3:0	PINMUX11[3:0]	A6	AXR0[7] / MDIO_CLK / GP3[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR0[7] 0010 = Selects Output Function MDIO_CLK 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP3[7] other = Reserved - Behavior is Undefined.

4.2.13 PINMUX12 Register Definition (Address 0x01C1 4150)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX12[31:28]				PINMUX12[27:24]				PINMUX12[23:20]				PINMUX12[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX12[15:12]				PINMUX12[11:8]				PINMUX12[7:4]				PINMUX12[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-13. PINMUX12 Register Bit Layout

Table 4-14. Field Descriptions for PINMUX12

Bits	Field	ZKB Ball	Description
31:28	PINMUX12[31:28]	P1	AXR1[3] / EQEP1A / GP4[3] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[3] 0010 = Selects Output Function EQEP1A 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[3] other = Reserved - Behavior is Undefined.
27:24	PINMUX12[27:24]	P2	AXR1[2] / GP4[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[2] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[2] other = Reserved - Behavior is Undefined.
23:20	PINMUX12[23:20]	R2	AXR1[1] / GP4[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[1] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[1] other = Reserved - Behavior is Undefined.
19:16	PINMUX12[19:16]	T3	AXR1[0] / GP4[0] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function AXR1[0] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined. 1000 = Selects Output Function GP4[0] other = Reserved - Behavior is Undefined.
15:12	PINMUX12[15:12]	D4	AMUTE1 / EHRPWMTZ / GP4[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AMUTE1 0010 = Selects Output Function EHRPWMTZ 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[14] other = Reserved - Behavior is Undefined.
11:8	PINMUX12[11:8]	L3	AFSR1 / GP4[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AFSR1 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[13] other = Reserved - Behavior is Undefined.
7:4	PINMUX12[7:4]	L2	ACLKR1 / ECAP2 / APWM2 / GP4[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function ACLKR1 0010 = Selects Output Function ECAP2 / APWM2 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[12] other = Reserved - Behavior is Undefined.
3:0	PINMUX12[3:0]	L1	AHCLKR1 / GP4[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AHCLKR1 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[11] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.14 PINMUX13 Register Definition (Address 0x01C1 4154)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX13[31:28]				PINMUX13[27:24]				PINMUX13[23:20]				PINMUX13[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX13[15:12]				PINMUX13[11:8]				PINMUX13[7:4]				PINMUX13[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-14. PINMUX13 Register Bit Layout

Table 4-15. Field Descriptions for PINMUX13

Bits	Field	ZKB Ball	Description
31:28	PINMUX13[31:28]	R15	EMA_D[1] / MMCSD_DAT[1] / UHPI_HD[1] / GP0[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[1] 0010 = Selects Output Function MMCSD_DAT[1] 0100 = Selects Output Function UHPI_HD[1] 1000 = Selects Output Function GP0[1] other = Reserved - Behavior is Undefined.
27:24	PINMUX13[27:24]	T13	EMA_D[0] / MMCSD_DAT[0] / UHPI_HD[0] / GP0[0] / BOOT[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[0] 0010 = Selects Output Function MMCSD_DAT[0] 0100 = Selects Output Function UHPI_HD[0] 1000 = Selects Output Function GP0[0] other = Reserved - Behavior is Undefined.
23:20	PINMUX13[23:20]	M1	AXR1[9] / GP4[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[9] 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[9] other = Reserved - Behavior is Undefined.
19:16	PINMUX13[19:16]	M2	AXR1[8] / EPWM1A / GP4[8] 0000 = Pin is tri-stated. 0001 = Selects Output Function AXR1[8] 0010 = Reserved - Behavior is Undefined. 0100 = Selects Output Function EPWM1A 1000 = Selects Output Function GP4[8] other = Reserved - Behavior is Undefined.
15:12	PINMUX13[15:12]	M3	AXR1[7] / EPWM1B / GP4[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[7] 0010 = Selects Output Function EPWM1B 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[7] other = Reserved - Behavior is Undefined.
11:8	PINMUX13[11:8]	M4	AXR1[6] / EPWM2A / GP4[6] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[6] 0010 = Selects Output Function EPWM2A 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[6] other = Reserved - Behavior is Undefined.
7:4	PINMUX13[7:4]	N1	AXR1[5] / EPWM2B / GP4[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[5] 0010 = Selects Output Function EPWM2B 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[5] other = Reserved - Behavior is Undefined.
3:0	PINMUX13[3:0]	N2	AXR1[4] / EQEP1B / GP4[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function AXR1[4] 0010 = Selects Output Function EQEP1B 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP4[4] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.15 PINMUX14 Register Definition (Address 0x01C1 4158)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX14[31:28]				PINMUX14[27:24]				PINMUX14[23:20]				PINMUX14[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX14[15:12]				PINMUX14[11:8]				PINMUX14[7:4]				PINMUX14[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-15. PINMUX14 Register Bit Layout

Table 4-16. Field Descriptions for PINMUX14

Bits	Field	ZKB Ball	Description
31:28	PINMUX14[31:28]	T14	EMA_D[9] / UHPI_HD[9] / LCD_D[9] / GP0[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[9] 0010 = Selects Output Function UHPI_HD[9] 0100 = Selects Output Function LCD_D[9] 1000 = Selects Output Function GP0[9] other = Reserved - Behavior is Undefined.
27:24	PINMUX14[27:24]	N12	EMA_D[8] / UHPI_HD[8] / LCD_D[8] / GP0[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[8] 0010 = Selects Output Function UHPI_HD[8] 0100 = Selects Output Function LCD_D[8] 1000 = Selects Output Function GP0[8] other = Reserved - Behavior is Undefined.
23:20	PINMUX14[23:20]	M15	EMA_D[7] / MMCSD_DAT[7] / UHPI_HD[7] / GP0[7] / BOOT[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[7] 0010 = Selects Output Function MMCSD_DAT[7] 0100 = Selects Output Function UHPI_HD[7] 1000 = Selects Output Function GP0[7] other = Reserved - Behavior is Undefined.
19:16	PINMUX14[19:16]	N13	EMA_D[6] / MMCSD_DAT[6] / UHPI_HD[6] / GP0[6] 0000 = Pin is tri-stated. 0001 = Selects Output Function EMA_D[6] 0010 = Selects Output Function MMCSD_DAT[6] 0100 = Selects Output Function UHPI_HD[6] 1000 = Selects Output Function GP0[6] other = Reserved - Behavior is Undefined.
15:12	PINMUX14[15:12]	N15	EMA_D[5] / MMCSD_DAT[5] / UHPI_HD[5] / GP0[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[5] 0010 = Selects Output Function MMCSD_DAT[5] 0100 = Selects Output Function UHPI_HD[5] 1000 = Selects Output Function GP0[5] other = Reserved - Behavior is Undefined.
11:8	PINMUX14[11:8]	P13	EMA_D[4] / MMCSD_DAT[4] / UHPI_HD[4] / GP0[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[4] 0010 = Selects Output Function MMCSD_DAT[4] 0100 = Selects Output Function UHPI_HD[4] 1000 = Selects Output Function GP0[4] other = Reserved - Behavior is Undefined.
7:4	PINMUX14[7:4]	P15	EMA_D[3] / MMCSD_DAT[3] / UHPI_HD[3] / GP0[3] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[3] 0010 = Selects Output Function MMCSD_DAT[3] 0100 = Selects Output Function UHPI_HD[3] 1000 = Selects Output Function GP0[3] other = Reserved - Behavior is Undefined.
3:0	PINMUX14[3:0]	R13	EMA_D[2] / MMCSD_DAT[2] / UHPI_HD[2] / GP0[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[2] 0010 = Selects Output Function MMCSD_DAT[2] 0100 = Selects Output Function UHPI_HD[2] 1000 = Selects Output Function GP0[2] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.16 PINMUX15 Register Definition (Address 0x01C1 415C)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX15[31:28]				PINMUX15[27:24]				PINMUX15[23:20]				PINMUX15[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX15[15:12]				PINMUX15[11:8]				PINMUX15[7:4]				PINMUX15[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-16. PINMUX15 Register Bit Layout

Table 4-17. Field Descriptions for PINMUX15

Bits	Field	ZKB Ball	Description
31:28	PINMUX15[31:28]	R9	EMA_A[1] / MMCSD_CLK / UHPI_HCNTL0 / GP1[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[1] 0010 = Selects Output Function MMCSD_CLK 0100 = Selects Output Function UHPI_HCNTL0 1000 = Selects Output Function GP1[1] other = Reserved - Behavior is Undefined.
27:24	PINMUX15[27:24]	T9	EMA_A[0] / LCD_D[7] / GP1[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[0] 0010 = Selects Output Function LCD_D[7] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[0] other = Reserved - Behavior is Undefined.
23:20	PINMUX15[23:20]	M16	EMA_D[15] / UHPI_HD[15] / LCD_D[15] / GP0[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[15] 0010 = Selects Output Function UHPI_HD[15] 0100 = Selects Output Function LCD_D[15] 1000 = Selects Output Function GP0[15] other = Reserved - Behavior is Undefined.
19:16	PINMUX15[19:16]	N14	EMA_D[14] / UHPI_HD[14] / LCD_D[14] / GP0[14] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMA_D[14] 0010 = Selects Output Function UHPI_HD[14] 0100 = Selects Output Function LCD_D[14] 1000 = Selects Output Function GP0[14] other = Reserved - Behavior is Undefined.
15:12	PINMUX15[15:12]	N16	EMA_D[13] / UHPI_HD[13] / LCD_D[13] / GP0[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[13] 0010 = Selects Output Function UHPI_HD[13] 0100 = Selects Output Function LCD_D[13] 1000 = Selects Output Function GP0[13] other = Reserved - Behavior is Undefined.
11:8	PINMUX15[11:8]	P14	EMA_D[12] / UHPI_HD[12] / LCD_D[12] / GP0[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[12] 0010 = Selects Output Function UHPI_HD[12] 0100 = Selects Output Function LCD_D[12] 1000 = Selects Output Function GP0[12] other = Reserved - Behavior is Undefined.
7:4	PINMUX15[7:4]	P16	EMA_D[11] / UHPI_HD[11] / LCD_D[11] / GP0[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[11] 0010 = Selects Output Function UHPI_HD[11] 0100 = Selects Output Function LCD_D[11] 1000 = Selects Output Function GP0[11] other = Reserved - Behavior is Undefined.
3:0	PINMUX15[3:0]	R14	EMA_D[10] / UHPI_HD[10] / LCD_D[10] / GP0[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_D[10] 0010 = Selects Output Function UHPI_HD[10] 0100 = Selects Output Function LCD_D[10] 1000 = Selects Output Function GP0[10] other = Reserved - Behavior is Undefined.

4.2.17 PINMUX16 Register Definition (Address 0x01C1 4160)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX16[31:28]				PINMUX16[27:24]				PINMUX16[23:20]				PINMUX16[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX16[15:12]				PINMUX16[11:8]				PINMUX16[7:4]				PINMUX16[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-17. PINMUX16 Register Bit Layout

Table 4-18. Field Descriptions for PINMUX16

Bits	Field	ZKB Ball	Description
31:28	PINMUX16[31:28]	R11	EMA_A[9] / LCD_HSYNC / GP1[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[9] 0010 = Selects Output Function LCD_HSYNC 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[9] other = Reserved - Behavior is Undefined.
27:24	PINMUX16[27:24]	T11	EMA_A[8] / LCD_PCLK / GP1[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[8] 0010 = Selects Output Function LCD_PCLK 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[8] other = Reserved - Behavior is Undefined.
23:20	PINMUX16[23:20]	N10	EMA_A[7] / LCD_D[0] / GP1[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[7] 0010 = Selects Output Function LCD_D[0] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[7] other = Reserved - Behavior is Undefined.
19:16	PINMUX16[19:16]	P10	EMA_A[6] / LCD_D[1] / GP1[6] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMA_A[6] 0010 = Selects Output Function LCD_D[1] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[6] other = Reserved - Behavior is Undefined.
15:12	PINMUX16[15:12]	R10	EMA_A[5] / LCD_D[2] / GP1[5] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[5] 0010 = Selects Output Function LCD_D[2] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[5] other = Reserved - Behavior is Undefined.
11:8	PINMUX16[11:8]	T10	EMA_A[4] / LCD_D[3] / GP1[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[4] 0010 = Selects Output Function LCD_D[3] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[4] other = Reserved - Behavior is Undefined.
7:4	PINMUX16[7:4]	N9	EMA_A[3] / LCD_D[6] / GP1[3] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[3] 0010 = Selects Output Function LCD_D[6] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[3] other = Reserved - Behavior is Undefined.
3:0	PINMUX16[3:0]	P9	EMA_A[2] / MMCSD_CMD / UHPI_HCNTL1 / GP1[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[2] 0010 = Selects Output Function MMCSD_CMD 0100 = Selects Output Function UHPI_HCNTL1 1000 = Selects Output Function GP1[2] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.18 PINMUX17 Register Definition (Address 0x01C1 4164)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX17[31:28]				PINMUX17[27:24]				PINMUX17[23:20]				PINMUX17[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX17[15:12]				PINMUX17[11:8]				PINMUX17[7:4]				PINMUX17[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-18. PINMUX17 Register Bit Layout

Table 4-19. Field Descriptions for PINMUX17

Bits	Field	ZKB Ball	Description
31:28	PINMUX17[31:28]	L16	$\overline{\text{EMA\_CAS}}$ / $\overline{\text{EMA\_CS[4]}}$ / GP2[1] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\overline{\text{EMA\_CAS}}$ 0010 = Selects Output Function $\overline{\text{EMA\_CS[4]}}$ 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[1] other = Reserved - Behavior is Undefined.
27:24	PINMUX17[27:24]	T12	EMA_SDCKE / GP2[0] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_SDCKE 0010 = Reserved - Behavior is Undefined 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[0] other = Reserved - Behavior is Undefined.
23:20	PINMUX17[23:20]	R12	EMA_CLK / OBSCLK / AHCLKR2 / GP1[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_CLK 0010 = Selects Output Function OBSCLK 0100 = Selects Output Function AHCLKR2 1000 = Selects Output Function GP1[15] other = Reserved - Behavior is Undefined.
19:16	PINMUX17[19:16]	R8	EMA_BA[0] / LCD_D[4] / GP1[14] Control 0000 = Pin is tri-stated. 0001 = Selects Output Function EMA_BA[0] 0010 = Selects Output Functio LCD_D[4] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[14] other = Reserved - Behavior is Undefined.
15:12	PINMUX17[15:12]	P8	EMA_BA[1] / LCD_D[5] / UHPI_HHWIL / GP1[13] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_BA[1] 0010 = Selects Output Function LCD_D[5] 0100 = Selects Output Function UHPI_HHWIL 1000 = Selects Output Function GP1[13] other = Reserved - Behavior is Undefined.
11:8	PINMUX17[11:8]	N11	EMA_A[12] / LCD_MCLK / GP1[12] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[12] 0010 = Selects Output Function LCD_MCLK 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[12] other = Reserved - Behavior is Undefined.
7:4	PINMUX17[7:4]	P11	EMA_A[11] / $\overline{\text{LCD\_AC\_ENB\_CS}}$ / GP1[11] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[11] 0010 = Selects Output Function $\overline{\text{LCD\_AC\_ENB\_CS}}$ 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[11] other = Reserved - Behavior is Undefined.
3:0	PINMUX17[3:0]	N8	EMA_A[10] / LCD_VSYNC / GP1[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_A[10] 0010 = Selects Output Function LCD_VSYNC 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP1[10] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.19 PINMUX18 Register Definition (Address 0x01C1 4168)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PINMUX18[31:28]				PINMUX18[27:24]				PINMUX18[23:20]				PINMUX18[19:16]			
R/W-0				R/W-0				R/W-0				R/W-0			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINMUX18[15:12]				PINMUX18[11:8]				PINMUX18[7:4]				PINMUX18[3:0]			
R/W-0				R/W-0				R/W-0				R/W-0			

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

Figure 4-19. PINMUX18 Register Bit Layout

Table 4-20. Field Descriptions for PINMUX18

Bits	Field	ZKB Ball	Description
31:28	PINMUX18[31:28]	M14	EMA_WE_DQM[0] / UHPI_HINT / AXR0[15] / GP2[9] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_WE_DQM[0] 0010 = Selects Output Function UHPI_HINT 0100 = Selects Output Function AXR0[15] 1000 = Selects Output Function GP2[9] other = Reserved - Behavior is Undefined.
27:24	PINMUX18[27:24]	P12	EMA_WE_DQM[1] / UHPI_HDS2 / AXR0[14] / GP2[8] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_WE_DQM[1] 0010 = Selects Output Function UHPI_HDS2 0100 = Selects Output Function AXR0[14] 1000 = Selects Output Function GP2[8] other = Reserved - Behavior is Undefined.
23:20	PINMUX18[23:20]	R7	EMA_OE / UHPI_HDS1 / AXR0[13] / GP2[7] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_OE 0010 = Selects Output Function UHPI_HDS1 0100 = Selects Output Function AXR0[13] 1000 = Selects Output Function GP2[7] other = Reserved - Behavior is Undefined.
19:16	PINMUX18[19:16]	T7	EMA_CS[3] / AMUTE2 / GP2[6] 0000 = Pin is tri-stated. 0001 = Selects Output Function EMA_CS[3] 0010 = Reserved - Behavior is Undefined. 0100 = Selects Output Function AMUTE2 1000 = Selects Output Function GP2[6] other = Reserved - Behavior is Undefined.
15:12	PINMUX18[15:12]	P7	EMA_CS[2] / UHPI_HCS / GP2[5] / BOOT[15] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_CS[2] 0010 = Selects Output Function UHPI_HCS 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[5] other = Reserved - Behavior is Undefined.
11:8	PINMUX18[11:8]	T8	EMA_CS[0] / UHPI_HAS / GP2[4] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_CS[0] 0010 = Selects Output Function UHPI_HAS 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[4] other = Reserved - Behavior is Undefined.
7:4	PINMUX18[7:4]	M13	EMA_WE / UHPI_HRW / AXR0[12] / GP2[3] / BOOT[14] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_WE 0010 = Selects Output Function UHPI_HRW 0100 = Selects Output Function AXR0[12] 1000 = Selects Output Function GP2[3] other = Reserved - Behavior is Undefined.
3:0	PINMUX18[3:0]	N7	EMA_RAS / EMA_CS[5] / GP2[2] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function EMA_RAS 0010 = Selects Output Function EMA_CS[5] 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[2] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.2.20 PINMUX19 Register Definition (Address 0x01C1 416C)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R/W-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												PINMUX19[3:0]			
R/W-0												R/W-0			

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

Figure 4-20. PINMUX19 Register Bit Layout

Table 4-21. Field Descriptions for PINMUX19

Bits	Field	ZKB Ball	Description
31:4	Reserved		Reserved - Write '0' to this register field when modifying this register.
3:0	PINMUX19[3:0]	N6	EMA_WAIT[0] / $\overline{\text{UHPI\_HRDY}}$ / GP2[10] Control 0000 [Default] = Pin is tri-stated. 0001 = Selects Output Function $\underline{\text{EMA\_WAIT[0]}}$ 0010 = Selects Output Function $\overline{\text{UHPI\_HRDY}}$ 0100 = Reserved - Behavior is Undefined 1000 = Selects Output Function GP2[10] other = Reserved - Behavior is Undefined.

PRODUCT PREVIEW

4.3 Bus Master Priority Configuration

The on chip switch fabric performs priority based arbitration among the various bus masters on the SOC . The priority of each master is controlled by the MSTPRI0, MSTPRI1, and MSTPRI2 registers and may be adjusted as required to suite a particular application. Section 4.3.1 through Section 4.3.3 give provide a detailed description of these registers.

4.3.1 MSTPRI0 Register Definition (0x01C1 4110)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV	RSV			RSV	RSV			RSV	RSV			RSV	RSV		
R/W-0	R/W-100			R/W-0	R/W-100			R/W-0	R/W-100			R/W-0	R/W-100		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSV	DSP_CFG			RSV	DSP_MDMA			RSV	ARM_D			RSV	ARM_I		
R/W-0	R/W-010			R/W-0	R/W-010			R/W-0	R/W-010			R/W-0	R/W-010		

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

Figure 4-21. MSTPRI0 Bit Description

Table 4-22. MSTPRI0 Field Descriptions

Bit	Field	Description
31	RSV	Reserved - Write 0 to this Field when modifying this register.
30:28	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
27	RSV	Reserved - Write 0 to this Field when modifying this register.
26:24	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
23	RSV	Reserved - Write 0 to this Field when modifying this register.
22:20	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
19	RSV	Reserved - Write 0 to this Field when modifying this register.
18:16	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
15	RSV	Reserved - Write 0 to this Field when modifying this register.
14:12	DSP_CFG	Bus Priority for Bus Master DSP - Configuration Bus - Default Value is 010 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
11	RSV	Reserved - Write 0 to this Field when modifying this register.
10:8	DSP_MDMA	Bus Priority for Bus Master DSP - DMA Bus - Default Value is 010 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
7	RSV	Reserved - Write 0 to this Field when modifying this register.
6:4	ARM_D	Bus Priority for Bus Master ARM - Data Fetch - Default Value is 010 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
3	RSV	Reserved - Write 0 to this Field when modifying this register.
2:0	ARM_I	Bus Priority for Bus Master ARM - Instruction Fetch - Default Value is 010 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)

4.3.2 MSTPRI1 Register Definition (0x01C1 4114)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV	RSV			RSV	RSV			RSV	RSV			RSV	RSV		
R/W-0	R/W-100			R/W-0	R/W-100			R/W-0	R/W-100			R/W-0	R/W-100		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSV	TC1			RSV	TC0			RSV	RSV			RSV	RSV		
R/W-0	R/W-000			R/W-0	R/W-000			R/W-0	R/W-000			R/W-0	R/W-000		

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

Figure 4-22. MSTPRI1 Bit Description

Table 4-23. MSTPRI1 Field Descriptions

Bit	Field	Description
31	RSV	Reserved - Write 0 to this Field when modifying this register.
30:28	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
27	RSV	Reserved - Write 0 to this Field when modifying this register.
26:24	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
23	RSV	Reserved - Write 0 to this Field when modifying this register.
22:20	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
19	RSV	Reserved - Write 0 to this Field when modifying this register.
18:16	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 100 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
15	RSV	Reserved - Write 0 to this Field when modifying this register.
14:12	TC1	Bus Priority for Bus Master EDMA3 TC1 - Default Value is 000 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
11	RSV	Reserved - Write 0 to this Field when modifying this register.
10:8	TC0	Bus Priority for Bus Master EDMA3 TC0 - Default Value is 000 000 = Priority 0 (Highest) 010 = Priority 2 100 = Priority 4 110 = Priority 6 001 = Priority 1 011 = Priority 3 101 = Priority 5 111 = Priority 7 (lowest)
7:0	RSV	Reserved - Write 0 to this Field when modifying this register.

4.3.3 MSTPRI2 Register Definition (0x01C1 4118)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV	LCDC			RSV	USB1			RSV	UHPI			RSV	RSV		
R/W-0	R/W-101			R/W-0	R/W-100			R/W-0	R/W-110			R/W-0	R/W-000		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSV	USB0			RSV	USB0			RSV	RSV			RSV	EMAC		
R/W-0	R/W-100			R/W-0	R/W-100			R/W-0	R/W-000			R/W-0	R/W-100		

LEGEND: R = Read, W = Write, n = value at reset. In a loaded system, the LCDC default priority value of 5 might not be a good default and may need to be changed.

Figure 4-23. MSTPRI2 Bit Description

Table 4-24. MSTPRI2 Field Descriptions

Bit	Field	Description			
31	RSV	Reserved - Write 0 to this Field when modifying this register.			
30:28	LCDC	Bus Priority for Bus Master LCDC - Default Value is 101			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
27	RSV	Reserved - Write 0 to this Field when modifying this register.			
26:24	USB1	Bus Priority for Bus Master USB1 - Default Value is 100			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
23	RSV	Reserved - Write 0 to this Field when modifying this register.			
22:20	UHPI	Bus Priority for Bus Master UHPI - Default Value is 110			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
19	RSV	Reserved - Write 0 to this Field when modifying this register.			
18:16	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 000			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
15	RSV	Reserved - Write 0 to this Field when modifying this register.			
14:12	USB0	Bus Priority for Bus Master USB0 - Default Value is 100			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
11	RSV	Reserved - Write 0 to this Field when modifying this register.			
10:8	USB0	Bus Priority for Bus Master USB0 - Default Value is 100			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
7	RSV	Reserved - Write 0 to this Field when modifying this register.			
6:4	RSV	Reserved For Future Use - Write Default Value to Maintain Compatibility - Default Value is 000			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)
3	RSV	Reserved - Write 0 to this Field when modifying this register.			
2:0	EMAC	Bus Priority for Bus Master EMAC - Default Value is 100			
		000 = Priority 0 (Highest)	010 = Priority 2	100 = Priority 4	110 = Priority 6
		001 = Priority 1	011 = Priority 3	101 = Priority 5	111 = Priority 7 (lowest)

4.4 Chip Configuration Registers (CFGCHIP and SUSPSRC)

These registers control EDMA3 default transfer burst sizes, clock muxing, McASP AMUTE and eCAP sources, UHPI enable and configuration, and USB PHY settings

4.4.1 CFGCHIP0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-n/a															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Reserved			PLLMASTERLOCK	TC1DBS		TC0DBS	
R-n/a								R/W-1		R/W-000	R/W-0	R/W-00		R/W-00	

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

Figure 4-24. CFGCHIP0 Register Bit Layout

Table 4-25. CFGCHIP0 Field Description

Bit	Field	Description
31:5	Reserved	Reserved
4	PLL_MASTER_LOCK	This bit is used to lock the PLL MMRs 0 = PLLCTRL MMR registers are freely accessible. 1 = PLLCTRL MMR registers are locked.
3:2	TC1DBS	EDMA3 TC1 Default Burst Size 00 = 16 byte 01 = 32 byte 10 = 64 byte 11 = Reserved
1:0	TC0DBS	EDMA3 TC1 Default Burst Size 00 = 16 byte 01 = 32 byte 10 = 64 byte 11 = Reserved

PRODUCT PREVIEW

4.4.2 CFGCHIP1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CAP2SRC					CAP1SRC					CAP0SRC					HPIBYTEAD
R/W-0					R/W-0					R/W-0					R/W-0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HPIENA	Rsvd		TBCLKSYNC	AMUTESEL2				AMUTESEL1				AMUTESEL0			
R/W-0	R/W-0		R/W-0	R/W-0				R/W-0				R/W-0			

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

Figure 4-25. CFGCHIP1 Register Bit Layout

Table 4-26. CFGCHIP1 Field Description

Bit	Field	Description	
31:27	CAP2SRC	eCAP2 Module Event Input Select	
26:22	CAP1SRC	eCAP1 Module Event Input Select	
21:17	CAP0SRC	eCAP0 Module Event Input Select	
		For each eCAPx (x=0,1,2): 00000 = eCAPx Pin Input 00001 = McASP0 TX DMA Event 00010 = McASP0 RX DMA Event 00011 = McASP1 TX DMA Event 00100 = McASP1 RX DMA Event 00101 = McASP2 TX DMA Event 00110 = McASP2 RX DMA Event 00111 = EMAC C0 RX Threshold Pulse Interrupt 01000 = EMAC C0 RX Pulse Interrupt 01001 = EMAC C0 TX Pulse Interrupt 01010 = EMAC C0 Misc Interrupt01011 = EMAC C1 RX Threshold Pulse Interrupt 01100 = EMAC C1 RX Pulse Interrupt 01101 = EMAC C1 TX Pulse Interrupt 01110 = EMAC C1 Misc Interrupt 01111 = EMAC C2 RX Threshold Pulse Interrupt 10000 = EMAC C2 RX Pulse Interrupt 10001 = EMAC C2 TX Pulse Interrupt 10010 = EMAC C2 Misc Interrupt 10011 - 11111 = Reserved	
16	HPIBYTEAD	HPI Module Byte / Word Address Mode	
		0 = Host Address is a word address	1 = Host Address is a byte address
15	HPIENA	HPI Enable Bit	
		0 = HPI Disabled	1 = HPI Enabled
14:13	Reserved	Reserved	
12	TBCLKSYNC	eHRPWM Module Time Base Clock Sync	
		0 (default) = The TBCLK (Time Base Clock) within each enabled eHRPWM is stopped.	1 = All enabled eHRPWM module clocks are started with the first rising edge of TBCLK aligned.
11:8	AMUTESEL2	Selects the source of the McASP2 AMUTEIN signal	
7:4	AMUTESEL1	Selects the source of the McASP1 AMUTEIN signal	
3:0	AMUTESEL0	Selects the source of the McASP0 AMUTEIN signal	

PRODUCT PREVIEW

Table 4-26. CFGCHIP1 Field Description (continued)

Bit	Field	Description
		For each McASPx (x=0,1,2) 0000 = Drive McASPx AMUTEIN Low 0001 = McASPx AMUTEIN source is GPIO Interrupt from Bank 0 0010 = McASPx AMUTEIN source is GPIO Interrupt from Bank 1 0011 = McASPx AMUTEIN source is GPIO Interrupt from Bank 2 0100 = McASPx AMUTEIN source is GPIO Interrupt from Bank 3 0101 = McASPx AMUTEIN source is GPIO Interrupt from Bank 4 0110 = McASPx AMUTEIN source is GPIO Interrupt from Bank 5 0111 =McASPx AMUTEIN source is GPIO Interrupt from Bank 6 1000 = McASPx AMUTEIN source is GPIO Interrupt from Bank 7 1001 - 1111 are reserved

4.4.3 CFGCHIP2

31	30	29	28	27	26	25	24
RESERVED							
R-n/a							

23	22	21	20	19	18	17	16
RESERVED					USB0PHYCLKGD	USB0VBUSSENSE	
R-n/a							

15	14	13	12	11	10	9	8
RESET	USB0OTGMODE		USB1PHYCLKMUX	USB0PHYCLKMUX	USB0PHYPWDN	USB0OTGPWRDN	USB0DATPOL
R/W-1	R/W-11		R/W-0	R/W-1	R/W-1	R/W-1	R/W-1

7	6	5	4	3	2	1	0
USB1SUSPENDM	USB0PHY_PLLON	USB0SESNDEN	USB0VBDTCTEN	USB0REF-FREQ[3:0]			
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0000			

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

Figure 4-26. CFGCHIP2 Register Bit Layout

Table 4-27. CFGCHIP2 Field Description

Bit	Field	Description
31:8	Reserved	Reserved
17	USB0PHYCLKGD	Indicates clock is present, power is good and phy PLL is locked.
16	USB0VBUSSENSE	Indicates status of VBUS detection.
15	Reset	When '1' drives 'phy_reset' active to put the phy UTMI+ interface in reset.
14:13	USB0OTGMODE	OTGMODE = 00. Do not override phy values. Let PHY drive signals to controller based on its comparators for the VBUS and ID pins. OTGMODE = 01. Override phy values to force USB Host Operation. Force VBUSVALID = 1, SESSVALID = 1, SESSEND = 0, IDDIG = 0 OTGMODE = 10. Override phy values to force USB Device Operation. Force VBUSVALID = 1, SESSVALID = 1, SESSEND = 0, IDDIG = 1 OTGMODE = 11. Override phy values to force USB Host Operation with VBUS low. Force VBUSVALID = 0, SESSVALID = 0, SESSEND = 1, IDDIG = 0
12	USB1PHYCLKMUX	USB1 PHY Clock Source. 1 = USB1 Phy Clock (48 MHz) is sourced by an external pin. 0 = USB1 Phy Clock (48 MHz) is sourced by the 48 MHz output of the USB0 PHY.
11	USB0PHYCLKMUX	USB0 PHY Clock Source. 1 = USB0 Phy reference clock internally generated. 0 = USB0 Phy reference clock comes from pin.
10	USB0PHYPWDN	Phy Powerdown 0=Phy is powered up, 1=Phy is powered down.
9	USB0OTGPWRDN	OTG Analog Module Powerdown 0=OTG Analog Module is powered up, 1=OTG Analog Module is powered down.
8	USB0DATPOL	USB0 Data Polarity, 0 = Reversed DP/DM polarity, 1 = Normal DP/DM polarity.
7	USB1SUSPENDM	USB1 Phy Suspend, Program to '0' if USB1 is not used, Program to '1' if USB1 is used.
6	USB0PHY_PLLON	USB0 Phy PLL On, 0 = Normal USB Behavior, 1 = Override USB SUSPEND behavior and release PLL from SUSPEND state.
5	USB0SESNDEN	Turns on session end comparator.
4	USB0VBDTCTEN	Turns on all VBUS line comparators.

Table 4-27. CFGCHIP2 Field Description (continued)

Bit	Field	Description
3:0	USB0REF-FREQ[3:0]	USB0 Phy Clock Input Select. 0000 = Reserved 0001 = 12 MHz 0010 = 24 MHz 0011 = 48 MHz 0100 = 19.2 MHz 0101 = 38.4 MHz 0110 = 13 MHz 0111 = 26 MHz 1000 = 20 MHz 1001 = 40 MHz 1010 = Reserved 1011 = Reserved 1100 = Reserved 1101 = Reserved 1110 = Reserved 1111 = Reserved

4.4.4 CFGCHIP3

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-n/a															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Reserved					DIV4P5ENA	EMA_CLKSRC	EMB_CLKSRC
R/W-1								R/W-0					R/W-0	R/W-0	R/W-0

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

Figure 4-27. CFGCHIP3 Register Bit Layout

Table 4-28. CFGCHIP3 Field Description

Bit	Field	Description
31:16	Reserved	Reserved
15:8	Reserved	Reserved
7:3	Reserved	Reserved
2	DIV4P5ENA	Fixed 4.5 divider Enable. 0 = Divide by 4.5 is Disabled. 1 = Divide by 4.5 is Enabled.
1	EMA_CLKSRC	EMIF A Memory Clock Source Select. 0 = EMIFA clock domain is driven by the PLLCTRL SYSCLK3 output. 1 = EMIFA clock domain is driven by the fixed / 4.5 PLL output.
0	EMB_CLKSRC	EMIF B Memory Clock Source Select. 0 = EMIFB SDRAM clock domain is driven by the PLLCTRL SYSCLK5 output. 1 = EMIFB SDRAM clock domain is driven by the fixed / 4.5 PLL output.

4.4.5 CFGCHIP4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-n/a															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Reserved					AMUTECLR2	AMUTECLR1	AMUTECLR0
R/W-1								R/W-0					R/W-0	R/W-0	R/W-0

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

Figure 4-28. CFGCHIP4 Register Bit Layout

Table 4-29. CFGCHIP4 Field Description

Bit	Field	Description
31:16	Reserved	Reserved
15:8	Reserved	Reserved
7:3	Reserved	Reserved
2	AMUTECLR2	Write 1 causes a single pulse that clears the 'latched' GPIO interrupt for AMUTEIN of McASP2 when '1'. Always reads back '0'.
1	AMUTECLR1	Write 1 causes a single pulse that clears the 'latched' GPIO interrupt for AMUTEIN of McASP1 when '1'. Always reads back '0'.
0	AMUTECLR0	Write 1 causes a single pulse that clears the 'latched' GPIO interrupt for AMUTEIN of McASP1 when '1'. Always reads back '0'.

4.4.6 SUSPSRC

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			TIMER64P1	TIMER64P0	GPIOSRC	ePWM2SRC	ePWM1SRC	ePWM0SRC	SPI1SRC	SPI0SRC	UART2SRC	UART1SRC	UART0SRC	I2C1SRC	I2C0SRC
R/W-1															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MMC/SD/SRC	Reserved		HPISRC	RSV	USB1SRC	USB0SRC	Reserved		RSV	EMACSRC	eQEP1SRC	eQEP0SRC	eCAP2SRC	eCAP1SRC	eCAP0SRC
R/W-1															

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

Figure 4-29. SUSPSRC Register Bit Layout

Table 4-30. SUSPSRC Field Descriptions

Bit	Field	Description
31:29	RSV	Reserved
28	TIMER64P1	TIMER64P1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
27	TIMER64P0	TIMER64P0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
26	GPIOSRC	GPIO Module Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend

PRODUCT PREVIEW

Table 4-30. SUSPSRC Field Descriptions (continued)

Bit	Field	Description
25	ePWM2SRC	ePWM2 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
24	ePWM1SRC	ePWM1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
23	ePWM0SRC	ePWM0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
22	SPI1 SRC	SPI1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
21	SPI0 SRC	SPI0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
20	UART2 SRC	UART2 SRC Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
19	UART1 SRC	UART1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
18	UART0 SRC	UART0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
17	I2C1 SRC	I2C1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
16	I2C0 SRC	I2C0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
15	MMC/SD SRC	MMC /SD Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
14:13	Reserved	Reserved
12	HPI SRC	HPI Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
11	Reserved	Reserved
10	USB1 SRC	USB1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
9	USB0 SRC	USB0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
8:6	Reserved	Reserved
5	EMACSRC	EMAC Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
4	eQEP1SRC	eQEP1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
3	eQEP0SRC	eQEP0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
2	eCAP2SRC	eCAP2 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
1	eCAP1SRC	eCAP1 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend
0	eCAP0SRC	eCAP0 Suspend Source 0 = ARM emulation suspend, 1 = DSP emulation suspend

4.5 ARM/DSP Communication Registers

4.5.1 CHIPSIG

The CHIPSIG register provides a signaling mechanism between the ARM and DSP. Writing a '1' to a bit causes the corresponding interrupt to be asserted. Writing a '0' has no effect. Reads return the value of the bit.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Rsvd															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rsvd											CHIPSIG[4:0]				
R-0											W-0				

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

Figure 4-30. CHIPSIG Register Bit Layout

Table 4-31. CHIPSIG Field Description

Bit	Field	Description
31:5	Reserved	Reserved
4	CHIPSIG[4]	Asserts DSP NMI Interrupt.
3	CHIPSIG[3]	Asserts DSP Interrupt CHIPSIG[3].
2	CHIPSIG[2]	Asserts DSP Interrupt CHIPSIG[2].
1	CHIPSIG[1]	Asserts ARM Interrupt CHIPSIG[1].
0	CHIPSIG[0]	Asserts ARM Interrupt CHIPSIG[0].

4.5.2 CHIPSIG_CLR

The CHIPSIG_CLR register clears interrupts that have been initiated using the CHIPSIG register. Writing a '1' to a bit clears the corresponding interrupt. Writing a '0' has no effect. Reads return the value of the bit.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Rsvd															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rsvd											CHIPSIG[4:0]				
R-0											W-0				

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

Figure 4-31. CHIPSIG_CLR Register Bit Layout

Table 4-32. CHIPSIG_CLR Field Description

Bit	Field	Description
31:5	Reserved	Reserved
4	CHIPSIG[4]	Clears DSP NMI Interrupt.
3	CHIPSIG[3]	Clears DSP Interrupt CHIPSIG[3].
2	CHIPSIG[2]	Clears DSP Interrupt CHIPSIG[2].
1	CHIPSIG[1]	Clears ARM Interrupt CHIPSIG[1].
0	CHIPSIG[0]	Clears ARM Interrupt CHIPSIG[0].

4.6 Device Support

4.6.1 Development Support

TI offers an extensive line of development tools for the OMAP-L13x 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 OMAP-L13x applications:

Software Development Tools:

Code Composer Studio™ Integrated Development Environment (IDE): including Editor

C/C++/Assembly Code Generation, and Debug plus additional development tools

Scalable, Real-Time Foundation Software (DSP/BIOS™), which provides the basic run-time target software needed to support any application.

Hardware Development Tools:

Extended Development System (XDS™) Emulator

For a complete listing of development-support tools for OMAP-L13x, visit the Texas Instruments web site on the Worldwide Web at http://www.ti.com uniform resource locator (URL). For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

4.6.2 Device and Development-Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all DSP devices and support tools. Each DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., TMS320C6745). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

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

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

NULL Fully-qualified production device.

Support tool development evolutionary flow:

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

TMDS Fully qualified development-support product.

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

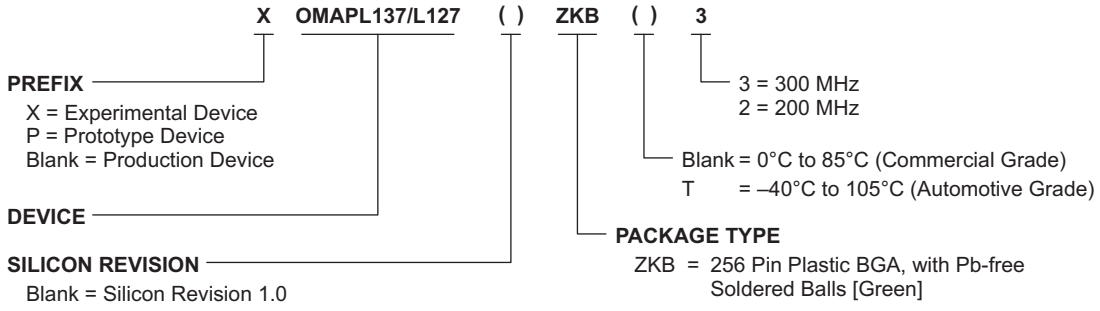
"Developmental product is intended for internal evaluation purposes."

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

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

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

Figure 4-32 provides a legend for reading the complete device name for any TMS320C674x member.



A. BGA = Ball Grid Array

B. The device speed range symbolization indicates the maximum CPU frequency when the core voltage CV_{DD} is set to 1.2 V.

Figure 4-32. Device Nomenclature

4.7 Documentation Support

4.7.1 Related Documentation From Texas Instruments

The following documents describe the OMAP-L13x Low-power Applications Processor. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

C64x+ Reference Guides

SPRU186 ***TMS320C6000 Assembly Language Tools v 6.1 User's Guide.*** Describes the assembly language tools (assembler, linker, and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for the TMS320C6000 platform of devices (including the C64x+ and C67x+ generations).

SPRU187 ***TMS320C6000 Optimizing Compiler v 6.1 User's Guide.*** Describes the TMS320C6000 C compiler and the assembly optimizer. This C compiler accepts ANSI standard C source code and produces assembly language source code for the TMS320C6000 platform of devices (including the C64x+ and C67x+ generations). The assembly optimizer helps you optimize your assembly code.

SPRU198 ***TMS320C6000 Programmer's Guide.*** Reference for programming the TMS320C6000 digital signal processors (DSPs). Before you use this manual, you should install your code generation and debugging tools. Includes a brief description of the C6000 DSP architecture and code development flow, includes C code examples and discusses optimization methods for the C code, describes the structure of assembly code and includes examples and discusses optimizations for the assembly code, and describes programming considerations for the C64x DSP.

SPRU862 ***TMS320C64x+ DSP Cache User's Guide.*** Explains the fundamentals of memory caches and describes how the two-level cache-based internal memory architecture in the TMS320C64x+ digital signal processor (DSP) of the TMS320C6000 DSP family can be efficiently used in DSP applications. Shows how to maintain coherence with external memory, how to use DMA to reduce memory latencies, and how to optimize your code to improve cache efficiency. The internal memory architecture in the C64x+ DSP is organized in a two-level hierarchy consisting of a dedicated program cache (L1P) and a dedicated data cache (L1D) on the first level. Accesses by the CPU to the these first level caches can

PRODUCT PREVIEW

complete without CPU pipeline stalls. If the data requested by the CPU is not contained in cache, it is fetched from the next lower memory level, L2 or external memory.

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

Primus DSP Reference Guides

SPRUG82 ***TMS320C674x DSP Cache User's Guide.*** Explains the fundamentals of memory caches and describes how the two-level cache-based internal memory architecture in the TMS320C674x digital signal processor (DSP) can be efficiently used in DSP applications. Shows how to maintain coherence with external memory, how to use DMA to reduce memory latencies, and how to optimize your code to improve cache efficiency. The internal memory architecture in the C674x DSP is organized in a two-level hierarchy consisting of a dedicated program cache (L1P) and a dedicated data cache (L1D) on the first level. Accesses by the CPU to the these first level caches can complete without CPU pipeline stalls. If the data requested by the CPU is not contained in cache, it is fetched from the next lower memory level, L2 or external memory.

SPRUFE8 ***TMS320C674x DSP CPU and Instruction Set Reference Guide.*** Describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C674x digital signal processors (DSPs). The C674x DSP is an enhancement of the C64x+ and C67x+ DSPs with added functionality and an expanded instruction set.

SPRUG84 ***OMAP-L137 Applications Processor System Reference Guide.*** Describes the System-on-Chip (SoC) including the ARM subsystem, DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, ARM interrupt controller (AINTC), and system configuration module.

SPRUFK5 ***TMS320C674x DSP Megamodule Reference Guide.*** Describes the TMS320C674x digital signal processor (DSP) megamodule. Included is a discussion on the internal direct memory access (IDMA) controller, the interrupt controller, the power-down controller, memory protection, bandwidth management, and the memory and cache.

SPRUGA6 ***OMAP-L137 Applications Processor Peripherals Overview Reference Guide.*** Provides an overview and briefly describes the peripherals available on the OMAP-L137 Applications Processor.

5 Device Operating Conditions

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

Supply voltage ranges	Core (CVDD, RTC_CVDD, PLL0_VDDA , USB0_VDDA12(2),) (3)	-0.5 V to 1.4 V
	I/O, 1.8V (USB0_VDDA18, USB1_VDDA18) (3)	-0.5 V to 2 V
	I/O, 3.3V (DVDD, USB0_VDDA33, USB1_VDDA33) (3)	-0.5 V to 3.8V
Input voltage ranges	V_I I/O, 1.2V (OSCIN, RTC_XI)	-0.3 V to CVDD + 0.3V
	V_I I/O, 3.3V (Steady State)	-0.3V to DVDD + 0.3V
	V_I I/O, 3.3V (Transient)	DVDD + 20% up to 20% of Signal Period
	V_I I/O, USB 5V Tolerant Pins: (USB0_DM, USB0_DP, USB0_ID, USB1_DM, USB1_DP)	5.25V(4)
	V_I I/O, USB0 VBUS	5.50V(4)
Output voltage ranges	V_O I/O, 3.3V (Steady State)	-0.5 V to DVDD + 0.3V
	V_O I/O, 3.3V (Transient)	DVDD + 20% up to 20% of 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, T_J	(default)	0°C to 105°C
	(T version)	-40°C to 125°C
Storage temperature range, T_{stg}	(default)	-55°C to 150°C

(1) Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) This pin is an internal LDO output and connected via 0.22 µF capacitor to USB0_VDDA12.
(3) All voltage values are with respect to VSS, USB0_VSSA33, USB0_VSSA, PLL0_VSSA, OSCVSS, RTC_VSS
(4) Up to a max of 24 hours.

5.2 Recommended Operating Conditions

			MIN	NOM	MAX	UNIT
CVDD	Supply voltage, Core (CVDD, RTC_CVDD, PLL0_VDDA , USB0_VDDA12[1]) [2]		1.14	1.2 or 1.26	1.32	V
DVDD	Supply voltage, I/O, 1.8V (USB0_VDDA18, USB1_VDDA18)		1.71	1.8	1.89	V
	Supply voltage, I/O, 3.3V (DVDD, USB0_VDDA33, USB1_VDDA33)		3.15	3.3	3.45	V
VSS	Supply ground (VSS, USB0_VSSA33, USB0_VSSA, PLL0_VSSA, OSCVSS[3], RTC_VSS[3])		0	0	0	V
V_{IH}	High-level input voltage, I/O, 3.3V		2			V
	High-level input voltage, OSCIN, RTC_XI		TBD			V
V_{IL}	Low-level input voltage, I/O, 3.3V				0.8	V
	Low-level input voltage, OSCIN, RTC_XI				TBD	V
t_t	Transition time, 10%-90%, All Inputs				10	ns
T_A	Operating ambient temperature range	Default	0		70	°C
		Automotive (T suffix)	-40		105	°C
$F_{SYSCLK1,6}$	DSP and ARM Operating Frequency (SYSCLK1,6)	Default	0		300	MHz
		Automotive (T suffix)	0		300	MHz

(1) This pin is an internal LDO output and connected via 0.22 µF capacitor to USB0_VDDA12.
(2) Future variants of TI SOC devices may operate at voltages ranging from 1.0 V to 1.32 V to provide a range of system power/ performance options. TI highly recommends that users design-in a supply that can handle multiple voltages within this range (i.e., 1.0 V, 1.1 V, 1.2, 1.26 V with ±5% tolerances) by implementing simple board changes such as reference resistor values or input pin configuration modifications. Not incorporating a flexible supply may limit the system's ability to easily adapt to future versions of TI SOC devices.
(3) Oscillator (OSC_VSS, RTC_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.

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

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	Low/full speed: USB0_DM and USB0_DP		2.8		USB0_VDDA33	V
	High speed: USB_DM and USB_DP		360		440	mV
	Low/full speed: USB1_DM and USB1_DP		2.8		USB1_VDDA33	V
	High-level output voltage (3.3V I/O)	DVDD = 3.15V, I_{OH} = -4 mA	2.4			V
		DVDD = 3.15V, I_{OH} = -100 µA	2.95			V
V_{OL}	Low/full speed: USB_DM and USB_DP		0.0		0.3	V
	High speed: USB_DM and USB_DP		-10		10	mV
	Low-level output voltage (3.3V I/O)	DVDD = 3.15V, I_{OL} = 4mA			0.4	V
		DVDD = 3.15V, I_{OL} = -100 µA			0.2	V
I_I (1)	Input current	V_I = VSS to DVDD without opposing internal resistor			±35	µA
		V_I = VSS to DVDD with opposing internal pullup resistor (2)	30		200	µA
		V_I = VSS to DVDD with opposing internal pulldown resistor (2)	-50		-250	µA
I_{OH}	High-level output current	All peripherals			-4	mA
I_{OL}	Low-level output current	All peripherals			4	mA

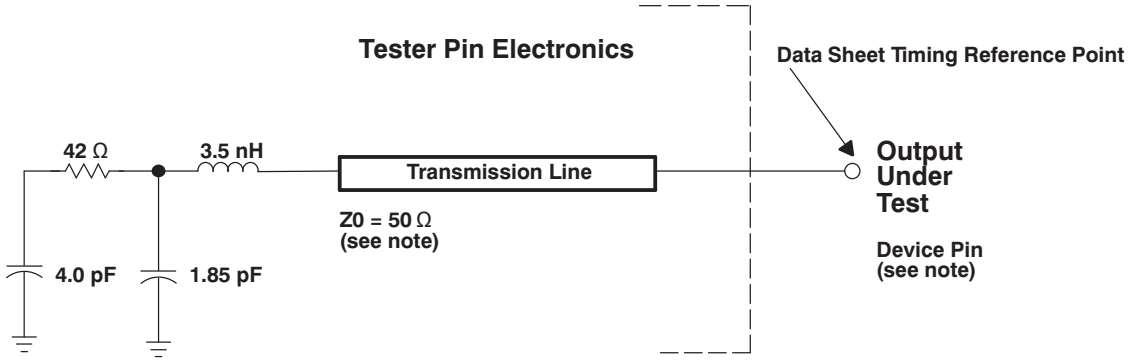
(1) I_I applies to input-only pins and bi-directional pins. For input-only pins, I_I indicates the input leakage current. For bi-directional pins, I_I indicates the input leakage current and off-state (Hi-Z) output leakage current.
(2) Applies only to pins with an internal pullup (IPU) or pulldown (IPD) resistor.

PRODUCT PREVIEW

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.

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.

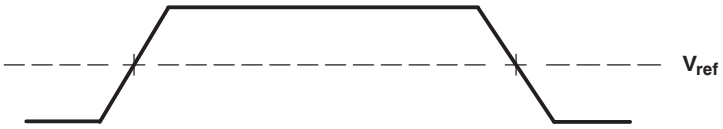


Figure 6-2. Input and Output Voltage Reference Levels for AC Timing Measurements

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.

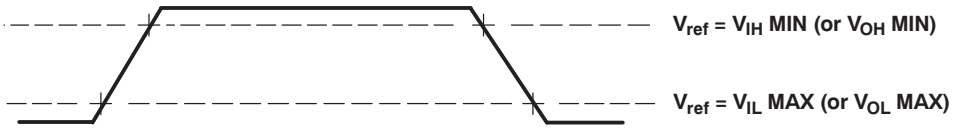


Figure 6-3. Rise and Fall Transition Time Voltage Reference Levels

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

OMAP-L13x devices include on chip logic that ensures I/O pins are tri-stated during the power on ramp, as long as the RESET\ pin is asserted. This is true even if the core voltage (CVDD) has not yet ramped.

Normally, the only requirement during the power on ramp is that both the RESET\ and TRST\ pins remain asserted (low) until after the power supply rails have fully ramped.

However, if the on chip USB modules are used; then to limit any noise on the USB0_DM, USB0_DP, USB1_DM, and USB1_DP pins to less than 200mV during the power on ramp, the sequence illustrated in Figure 6-4 must be followed. The requirement is that the core supply (CVDD) must ramp to at least 0.9V (1) before the IO supply (DVDD) reaches the 1.65V point in its ramp (2). And as is always the case, RESET\ and TRST\ must remain asserted during the power on ramp and released only after CVDD and DVDD are within their specified ranges.

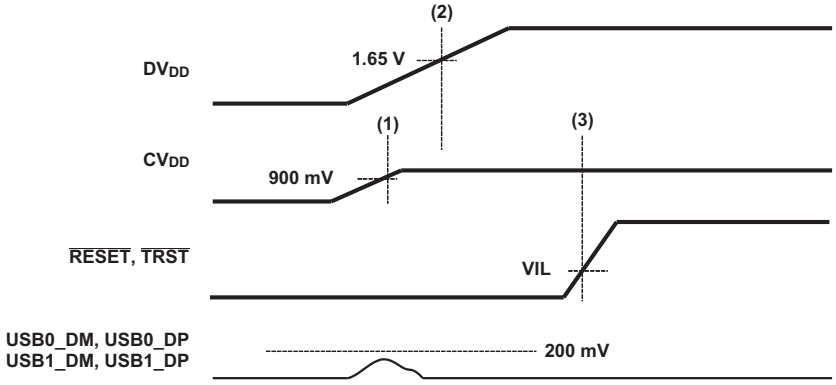


Figure 6-4. Power Sequence

6.4 Reset

TBD

6.5 Crystal Oscillator or External Clock Input

The OMAP-L137 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 6-5 and Figure 6-6.

- Figure 6-5 illustrates the option that uses on-chip 1.2V oscillator with external crystal circuit.
- Figure 6-6 illustrates the option that uses an external 1.2V clock input.

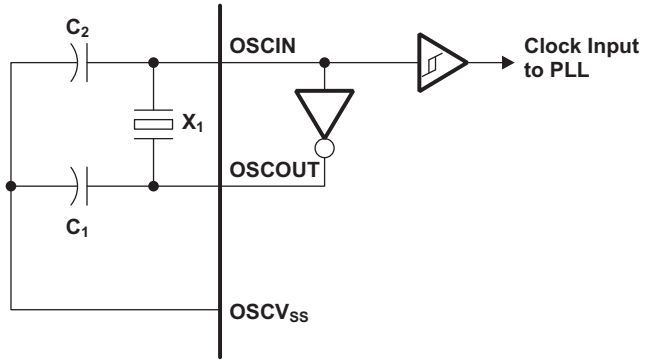


Figure 6-5. On-Chip 1.2V Oscillator

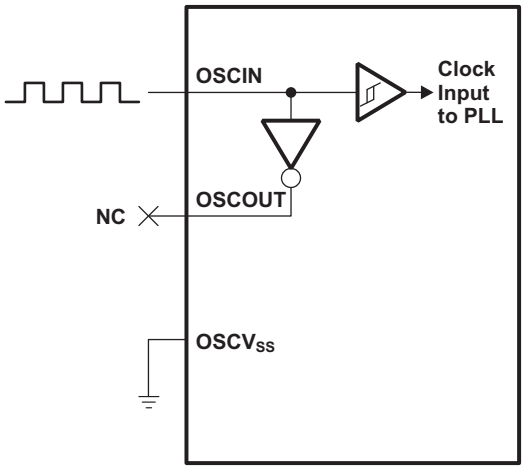


Figure 6-6. External 1.2V Clock Source

Table 6-1. CLKIN Timing Requirements

		MIN	MAX	UNIT
f_{osc}	Oscillator frequency range (OSCIN/OSCOUT)	12	30	MHz
f_{PLL}	Freuency range of PLL input , external clock source only	12	50	MHz
$t_{c(CLKIN)}$	Cycle time, external clock driven on OSCIN	20		ns
$t_{w(CLKINH)}$	Pulse width high, external clock on OSCIN	0.4 $t_{c(CLKIN)}$		ns
$t_{w(CLKINL)}$	Pulse width low, external clock on OSCIN	0.4 $t_{c(CLKIN)}$		ns
$t_{t(CLKIN)}$	Transition time, CLKIN	5		ns

PRODUCT PREVIEW

6.6 Clock PLLs

The OMAP-L137 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, , Dn

Various other controls supported are as follows:

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

6.6.1 PLL Device-Specific Information

The OMAP-L137 DSP generates the high-frequency internal clocks it requires through an on-chip PLL.

The PLL requires some external filtering components to reduce power supply noise as shown in Figure 6-7.

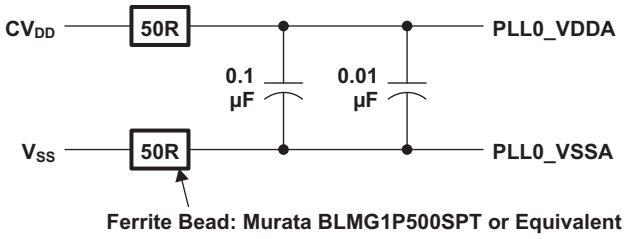


Figure 6-7. PLL External Filtering Components

The input to the PLL is either from the on-chip oscillator (OSCIN pin) or from an external clock on the CLKIN pin. The PLL outputs nine clocks that have programmable divider options. Figure 6-8 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 6-2 before enabling the DSP to run from the PLL by setting PLLEN = 1.

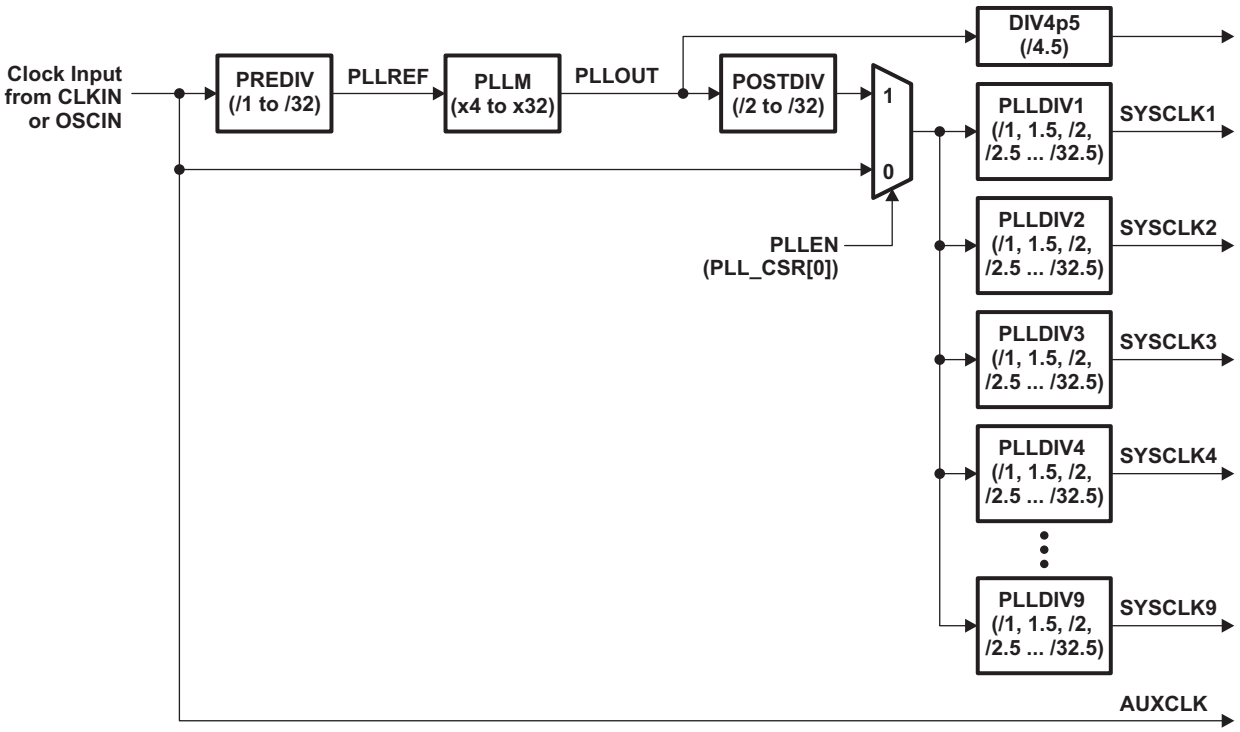


Figure 6-8. PLL Topology

Table 6-2. Allowed PLL Operating Conditions

NO	PARAMETER	MIN	MAX	UNIT
1	PLLRST: Assertion time during initialization	125	N/A	ns
2	Lock time: The time that the application has to wait for the PLL to acquire locks before setting PLLEN, after changing PREDIV, PLLM, or OSCIN	N/A	Max PLL Lock Time = $\frac{2000\,N}{\sqrt{m}}$ where N = Pre-Divider Ratio M = PLL Multiplier	ns
3	PLL input frequency (PLLREF after D0)	12	50	MHz
4	PLL multiplier values (PLLM)(1)	x4	x32	
5	PLL output frequency. (PLLOUT before dividers D1, D2, D3, ...)	400	600(2)	MHz

(1) The multiplier values must be chosen such that the PLL output frequency (at PLLOUT) is between 400 and 1000 MHz, but the frequency going into the SYSCLK dividers (after the post divider) cannot exceed 410 MHz. If the PLLOUT exceeds 410 MHz the post divider must be used to divide it down. The Post Divider and SYSCLK divider values must be chosen such that the CPU clocks do not exceed 300 MHz.

(2) PLL post divider / 2 must be used. The /4.5 clock path can be used to generate an EMIF clock from the undivided (i.e. 600 MHz) PLL output clock.

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.

PLLC0 generates several clocks from the PLL0 output clock for use by the various processors and modules. These are summarized in Table 6-3. The clock ratios between SYSCLK1, SYSCLK2, SYSCLK4 and SYSCLK6 must always be maintained as shown in the table.

Table 6-3. System PLLC0 Output Clocks

Output Clock	Used by	Default Ratio (relative to SYSCLK1)	Notes
SYSCLK1	DSP	/1	No Required Ratio
SYSCLK2	ARM RAM, ARM ROM, EDMA, DSP ports, EMIFB (ports to switch fabric), ECAP 0/1/2, EPWM 0/1/2, EQEP 0/1, Shared RAM, LCDC, McASP/FIFO 0/1/2, SPI 1, UHPI, USB2.0 (logic), UART 1/2, HRPWM 0/1/2	/2	SYSCLK1 / 2
SYSCLK3	EMIFA	/3	No Required Ratio
SYSCLK4	SYSCFG, Interrupt Controller, PLLC0, PSC 0, EMAC/MDIO, GPIO, I2C 1, PSC 1, USB1.1	/4	SYSCLK1 / 4
SYSCLK5	EMIFB	/3	No Required Ratio
SYSCLK6	ARM Subsystem	/1	SYSCLK1 / 1
SYSCLK7	RMII clock to EMAC	/6	No Required Ratio ; Should be set to 50 MHz
AUXCLK	McASP AuxClk,RTC,Timer64P0,Timer64P1	N/A	No Required Ratio
USB48	USB2.0 Phy, USB1.1 logic	N/A	No Required Ratio; Should be set to 48 MHz
USB12	USB2.0 Phy, USB1.1 logic	N/A	No Required Ratio; 12 MHz, generated by the USB1 Module by dividing USB48 by 4.
DIV4p5	133MHz clock source for EMIFB	PLL output/4.5	No Required Ratio

- The divide values in the PLL Controller 0 for SYSCLK1/SYSCLK6, SYSCLK2 and SYSCLK4 are not fixed so that user can change the divide values for power saving reasons. But users are responsible to guarantee that the divide ratios between these clock domains must be fixed to 1:2:4.
- Although the PLL is capable of running at 600 MHz, the SYSCLK dividers in the PLLC0 are not (maximum 410 MHz). For this reason, the post-divider in the PLLC0 should be configured for /2 to provide 300 MHz to each of the SYSCLK dividers.
- The DIV4p5 (/4.5) hardware clock divider is provided to generate 133 MHz from the 600 MHz PLL clock for use as clocks to the EMIFs.

6.7 Interrupts

The OMAP-L137 devices have a large number of interrupts to service the needs of its many peripherals and subsystems. Both the ARM and C674x CPUs are capable of servicing these interrupts equally. The interrupts can be selectively enabled or disabled in either of the controllers. Also, the ARM and DSP can communicate with each other through interrupts controlled by registers in the SYSCFG module.

6.7.1 ARM CPU Interrupts

The ARM9 CPU core supports 2 direct interrupts: FIQ and IRQ. The ARM Interrupt Controller on the OMAP-L13x extends the number of interrupts to 100, and provides features like programmable masking, priority, hardware nesting support, and interrupt vector generation. The OMAP-L13x ARM Interrupt controller is enhanced from previous devices like the DM6446 and DM355.

6.7.1.1 ARM Interrupt Controller (AINTC) Interrupt Signal Hierarchy

On OMAP-L13x, 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:

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

Where BASE and SIZE are programmable. The computed vector is a 32-bit address which may 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 OMAP-L137

System Interrupt assignments for the OMAP-L137 are listed in Table 6-4

Table 6-4. AINTC System Interrupt Assignments

System Interrupt	Interrupt Name	Source
0	COMMTX	ARM
1	COMMRX	ARM
2	NINT	ARM
3	-	Reserved
4	-	Reserved
5	-	Reserved
6	-	Reserved
7	-	Reserved
8	-	Reserved
9	-	Reserved
10	-	Reserved
11	EDMA3_CC0_CCINT	EDMA CC Region 0
12	EDMA3_CC0_CCERRINT	EDMA CC
13	EDMA3_TC0_TCERRINT	EDMA TC0
14	EMIFA_INT	EMIFA
15	IIC0_INT	I2C0
16	MMCSD_INT0	MMCSD
17	MMCSD_INT1	MMCSD
18	PSC0_ALLINT	PSC0
19	RTC_IRQS[1:0]	RTC
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	PROTERR	SYSCFG Protection Shared Interrupt
28	SYSCFG_CHIPINT0	SYSCFG CHIPSIG Register
29	SYSCFG_CHIPINT1	SYSCFG CHIPSIG Register
30	SYSCFG_CHIPINT2	SYSCFG CHIPSIG Register
31	SYSCFG_CHIPINT3	SYSCFG CHIPSIG Register
32	EDMA3_TC1_TCERRINT	EDMA TC1
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

Table 6-4. AINTC System Interrupt Assignments (continued)

System Interrupt	Interrupt Name	Source
47	GPIO_B5INT	GPIO Bank 5 Interrupt
48	GPIO_B6INT	GPIO Bank 6 Interrupt
49	GPIO_B7INT	GPIO Bank 7 Interrupt
50	-	Reserved
51	IIC1_INT	I2C1
52	LCDC_INT	LCD Controller
53	UART_INT1	UART1
54	MCASP_INT	McASP0, 1, 2 Combined RX / TX Interrupts
55	PSC1_ALLINT	PSC1
56	SPI1_INT	SPI1
57	UHPI_ARMINT	HPI Arm Interrupt
58	USB0_INT	USB0 Interrupt
59	USB1_HCINT	USB1 OHCI Host Controller Interrupt
60	USB1_RWAKEUP	USB1 Remote Wakeup Interrupt
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	-	Reserved
92	-	Reserved
93	-	Reserved

Table 6-4. AINTC System Interrupt Assignments (continued)

System Interrupt	Interrupt Name	Source
94	-	Reserved
95	-	Reserved
96	-	Reserved
97	-	Reserved
98	-	Reserved
99	-	Reserved
100	-	Reserved

6.7.1.5 AINTC Memory Map

Table 6-5. AINTC Memory Map

BYTE ADDRESS	REGISTER NAME	DESCRIPTION
0xFFFE E000	REV	Revision Register
0xFFFE E004	CR	Control Register
0xFFFE E008 - 0xFFFE E00F	-	Reserved
0xFFFE E010	GER	Global Enable Register
0xFFFE E014 - 0xFFFE E01B	-	Reserved
0xFFFE E01C	GNLR	Global Nesting Level Register
0xFFFE E020	SISR	System Interrupt Status Indexed Set Register
0xFFFE E024	SICR	System Interrupt Status Indexed Clear Register
0xFFFE E028	EISR	System Interrupt Enable Indexed Set Register
0xFFFE E02C	EICR	System Interrupt Enable Indexed Clear Register
0xFFFE E030	-	Reserved
0xFFFE E034	HIEISR	Host Interrupt Enable Indexed Set Register
0xFFFE E038	HIDISR	Host Interrupt Enable Indexed Clear Register
0xFFFE E03C - 0xFFFE E04F	-	Reserved
0xFFFE E050	VBR	Vector Base Register
0xFFFE E054	VSR	Vector Size Register
0xFFFE E058	VNR	Vector Null Register
0xFFFE E05C - 0xFFFE E07F	-	Reserved
0xFFFE E080	GPIR	Global Prioritized Index Register
0xFFFE E084	GPVR	Global Prioritized Vector Register
0xFFFE E088 - 0xFFFE E1FF	-	Reserved
0xFFFE E200 - 0xFFFE E20F	SRSR[0] - SRSR[3]	System Interrupt Status Raw / Set Registers
0xFFFE E210- 0xFFFE E27F	-	Reserved
0xFFFE E280 - 0xFFFE E28B	SECR[0] - SECR[3]	System Interrupt Status Enabled / Clear Registers
0xFFFE E28C - 0xFFFE E2FF	-	Reserved
0xFFFE E300 - 0xFFFE E30F	ESR[0] - ESR[3]	System Interrupt Enable Set Registers
0xFFFE E310 - 0xFFFE E37F	-	Reserved
0xFFFE E380 - 0xFFFE E38B	ECR[0] - ECR[3]	System Interrupt Enable Clear Registers
0xFFFE E38C - 0xFFFE E3FF	-	Reserved
0xFFFE E400 - 0xFFFE E45B	CMR[0] - CMR[31]	Channel Map Registers (Byte Wide Registers)
0xFFFE E45C - 0xFFFE E8FF	-	Reserved
0xFFFE E800 - 0xFFFE E81F	-	Reserved
0xFFFE E820 - 0xFFFE E8FF	-	Reserved
0xFFFE E900 - 0xFFFE E904	HIPIR[0] - HIPIR[1]	Host Interrupt Prioritized Index Registers
0xFFFE E908 - 0xFFFE EEFF	-	Reserved
0xFFFE EF00 - 0xFFFE EF04	DSR[0] - DSR[1]	Debug Select Registers
0xFFFE EF08 - 0xFFFE F0FF	-	Reserved
0xFFFE F100 - 0xFFFE F104	HINLR[0] - HINLR[1]	Host Interrupt Nesting Level Registers
0xFFFE F108 - 0xFFFE F4FF	-	Reserved
0xFFFE F500	HIER[0]	Host Interrupt Enable Register
0xFFFE F504 - 0xFFFE F5FF	-	Reserved
0xFFFE F600	HIPVR[0] - HIPVR[1]	Host Interrupt Prioritized Vector Registers
0xFFFE F608 - 0xFFFE FFFF	-	Reserved

6.7.2 DSP Interrupts

The C674x DSP interrupt controller combines device events into 12 prioritized interrupts. The source for each of the 12 CPU interrupts is user programmable and is listed in Table 6-6. Also, the interrupt controller controls the generation of the CPU exception, NMI, and emulation interrupts. Table 6-7 summarizes the C674x interrupt controller registers and memory locations.

Table 6-6. OMAP-L137 DSP Interrupts

EVT#	Interrupt Name	Source
0	EVT0	C674x Int Ctl 0
1	EVT1	C674x Int Ctl 1
2	EVT2	C674x Int Ctl 2
3	EVT3	C674x Int Ctl 3
4	T64P0_TINT12	Timer64P0 - TINT12
5	SYSCFG_CHIPINT2	SYSCFG_CHIPSIG Register
6	-	Reserved
7	EHRPWM0	HiResTimer/PWM0 Interrupt
8	TPCC0_INT1	TPCC0 Region 1 Interrupt
9	EMU-DTDMA	C674x-ECM
10	EHRPWM0TZ	HiResTimer/PWM0 Trip Zone Interrupt
11	EMU-RTDXRX	C674x-RTDX
12	EMU-RTDXTX	C674x-RTDX
13	IDMAINT0	C674x-EMC
14	IDMAINT1	C674x-EMC
15	MMCSD_INT0	MMCSD MMC/SD Interrupt
16	MMCSD_INT1	MMCSD SDIO Interrupt
17	-	Reserved
18	EHRPWM1	HiResTimer/PWM1 Interrupt
19	USB0_INT	USB0 Interrupt
20	USB1_HCINT	USB1 OHCI Host Controller Interrupt
21	USB1_RWAKEUP	USB1 Remote Wakeup Interrupt
22	-	Reserved
23	EHRPWM1TZ	HiResTimer/PWM1 Trip Zone Interrupt
24	EHRPWM2	HiResTimer/PWM2 Interrupt
25	EHRPWM2TZ	HiResTimer/PWM2 Trip Zone Interrupt
26	EMAC_C0RXTHRESH	EMAC - Core 0 Receive Threshold Interrupt
27	EMAC_C0RX	EMAC - Core 0 Receive Interrupt
28	EMAC_C0TX	EMAC - Core 0 Transmit Interrupt
29	EMAC_C0MISC	EMAC - Core 0 Miscellaneous Interrupt
30	EMAC_C1RXTHRESH	EMAC - Core 1 Receive Threshold Interrupt
31	EMAC_C1RX	EMAC - Core 1 Receive Interrupt
32	EMAC_C1TX	EMAC - Core 1 Transmit Interrupt
33	EMAC_C1MISC	EMAC - Core 1 Miscellaneous Interrupt
34	UHPI_DSPINT	UHPI DSP Interrupt
35	-	Reserved
36	IIC0_INT	I2C0
37	SP0_INT	SPI0
38	UART0_INT	UART0
39	-	Reserved
40	T64P1_TINT12	Timer64P1 Interrupt 12
41	GPIO_B1INT	GPIO Bank 1 Interrupt
42	IIC1_INT	I2C1
43	SPI1_INT	SPI1
44	-	Reserved
45	ECAP0	ECAP0
46	UART_INT1	UART1

Table 6-6. OMAP-L137 DSP Interrupts (continued)

EVT#	Interrupt Name	Source
47	ECAP1	ECAP1
48	T64P1_TINT34	Timer64P1 Interrupt 34
49	GPIO_B2INT	GPIO Bank 2 Interrupt
50	-	Reserved
51	ECAP2	ECAP2
52	GPIO_B3INT	GPIO Bank 3 Interrupt
53	EQEP1	EQEP1
54	GPIO_B4INT	GPIO Bank 4 Interrupt
55	EMIFA_INT	EMIFA
56	EDMA3_CC0_ERRINT	EDMA3 Channel Controller 0
57	EDMA3_TC0_ERRINT	EDMA3 Transfer Controller 0
58	EDMA3_TC1_ERRINT	EDMA3 Transfer Controller 1
59	GPIO_B5INT	GPIO Bank 5 Interrupt
60	EMIFB_INT	EMIFB Memory Error Interrupt
61	MCASP_INT	McASP0,1,2 Combined RX/TX Interrupts
62	GPIO_B6INT	GPIO Bank 6 Interrupt
63	RTC_IRQS	RTC Combined
64	T64P0_TINT34	Timer64P0 Interrupt 34
65	GPIO_B0INT	GPIO Bank 0 Interrupt
66	-	Reserved
67	SYSCFG_CHIPINT3	SYSCFG_CHIPSIG Register
68	EQEP0	EQEP0
69	UART2_INT	UART2
70	PSC0_ALLINT	PSC0
71	PSC1_ALLINT	PSC1
72	GPIO_B7INT	GPIO Bank 7 Interrupt
73	LCDC_INT	LDC Controller
74	PROTERR	SYSCFG Protection Shared Interrupt
75	-	Reserved
76	-	Reserved
77	-	Reserved
78	T64P0_CMPINT0	Timer64P0 - Compare 0
79	T64P0_CMPINT1	Timer64P0 - Compare 1
80	T64P0_CMPINT2	Timer64P0 - Compare 2
81	T64P0_CMPINT3	Timer64P0 - Compare 3
82	T64P0_CMPINT4	Timer64P0 - Compare 4
83	T64P0_CMPINT5	Timer64P0 - Compare 5
84	T64P0_CMPINT6	Timer64P0 - Compare 6
85	T64P0_CMPINT7	Timer64P0 - Compare 7
86	T64P1_CMPINT0	Timer64P1 - Compare 0
87	T64P1_CMPINT1	Timer64P1 - Compare 1
88	T64P1_CMPINT2	Timer64P1 - Compare 2
89	T64P1_CMPINT3	Timer64P1 - Compare 3
90	T64P1_CMPINT4	Timer64P1 - Compare 4
91	T64P1_CMPINT5	Timer64P1 - Compare 5
92	T64P1_CMPINT6	Timer64P1 - Compare 6
93	T64P1_CMPINT7	Timer64P1 - Compare 7

PRODUCT PREVIEW

Table 6-6. OMAP-L137 DSP Interrupts (continued)

EVT#	Interrupt Name	Source
94	-	Reserved
95	-	Reserved
96	INTERR	C674x-Int Ctl
97	EMC_IDMAERR	C674x-EMC
98	-	Reserved
99	-	Reserved
100	-	Reserved
101	-	Reserved
102	-	Reserved
103	-	Reserved
104	-	Reserved
105	-	Reserved
106	-	Reserved
107	-	Reserved
108	-	Reserved
109	-	Reserved
110	-	Reserved
111	-	Reserved
112	-	Reserved
113	PMC_ED	C674x-PMC
114	-	Reserved
115	-	Reserved
116	UMC_ED1	C674x-UMC
117	UMC_ED2	C674x-UMC
118	PDC_INT	C674x-PDC
119	SYS_CMPA	C674x-SYS
120	PMC_CMPA	C674x-PMC
121	PMC_CMPA	C674x-PMC
122	DMC_CMPA	C674x-DMC
123	DMC_CMPA	C674x-DMC
124	UMC_CMPA	C674x-UMC
125	UMC_CMPA	C674x-UMC
126	EMC_CMPA	C674x-EMC
127	EMC_BUSERR	C674x-EMC

Table 6-7. C674x DSP Interrupt Controller Registers

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

6.7.3 ARM/DSP Communications Interrupts

Communications Interrupts between the ARM and DSP are part of the SYSCFG module on the OMAP-L13x family of devices.(Section 4.5.1 CHIPSIG, Section 4.5.2 CHIPSIG_CLR)

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 OMAP-L137 GPIO peripheral supports the following:

- Up to 128 Pins on ZKB package configurable as GPIO
- 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
  - GPIO Banks 0, 1, 2, 3, 4, 5, 6, and 7 Interrupts assigned to DSP Events 65, 41, 49, 52, 54, 59, 62 and 72 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 anther process during GPIO programming).
- Separate Input/Output registers
- Output register in addition to set/clear so that, if preferred by firmware, some GPIO output signals can be toggled by direct write to the output register(s).
- Output register, when read, reflects output drive status. This, in addition to the input register reflecting pin status and open-drain I/O cell, allows wired logic be implemented.

The memory map for the GPIO registers is shown in Table 6-8. See the *OMAP-L137 Applications Processor DSP Peripherals Overview Reference Guide.* – Literature Number SPRUGA6 for more details.

6.8.1 GPIO Register Description(s)

Table 6-8. GPIO Registers

GPIO BYTE ADDRESS	Acronym	Register Description
0x01E2 6000	REV	Peripheral Revision Register
0x01E2 6004	RESERVED	Reserved
0x01E2 6008	BINTEN	GPIO Interrupt Per-Bank Enable Register
GPIO Banks 0 and 1		
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

Table 6-8. GPIO Registers (continued)

GPIO BYTE ADDRESS	Acronym	Register Description
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
GPIO Banks 2 and 3		
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
GPIO Banks 4 and 5		
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
GPIO Banks 6 and 7		
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
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

PRODUCT PREVIEW

6.8.2 GPIO Peripheral Input/Output Electrical Data/Timing

Table 6-9. Timing Requirements for GPIO Inputs(1) (see Figure 6-9)

NO.			MIN	MAX	UNIT
1	$t_{w(GPIH)}$	Pulse duration, GPIx high	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 OMAP-L137 recognize the GPIx changes through software polling of the GPIO register, the GPIx duration must be extended to allow OMAP-L137 enough time to access the GPIO register through the internal bus.
(2) C=SYSCLK4 period in ns. For example, when running parts at 300 MHz, C=13.33 ns

Table 6-9. Timing Requirements for GPIO Inputs (see Figure 6-9) (continued)

NO.			MIN	MAX	UNIT
2	$t_{w(GPIL)}$	Pulse duration, GPIx low	2C(1)(2)		ns

Table 6-10. Switching Characteristics Over Recommended Operating Conditions for GPIO Outputs (see Figure 6-9)

NO.	PARAMETER		MIN	MAX	UNIT
3	$t_{w(GPOH)}$	Pulse duration, GPOx high	2C(1) (2)		ns
4	$t_{w(GPOL)}$	Pulse duration, GPOx 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=SYSCLK4 period in ns. For example, when running parts at 300 MHz, C=13.33 ns

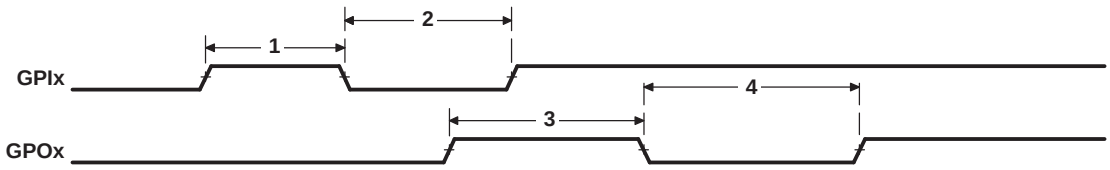


Figure 6-9. GPIO Port Timing

6.8.3 GPIO Peripheral External Interrupts Electrical Data/Timing

Table 6-11. Timing Requirements for External Interrupts(1) (see Figure 6-10)

NO.			MIN	MAX	UNIT
1	$t_{w(ILOW)}$	Width of the external interrupt pulse low	2C(1)(2)		ns
2	$t_{w(IHIGH)}$	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 OMAP-L137 recognize the GPIO changes through software polling of the GPIO register, the GPIO duration must be extended to allow OMAP-L137 enough time to access the GPIO register through the internal bus.
(2) C=SYSCLK4 period in ns. For example, when running parts at 300 MHz, C=13.33 ns

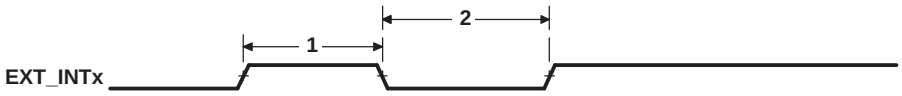


Figure 6-10. GPIO External Interrupt Timing

6.9 EDMA

Table 6-12 is the list of EDMA3 Channel Contoller 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
Global Registers		
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(1)
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
Global Channel Registers		
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
Shadow Region 0 Channel Registers		
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
Shadow Region 1 Channel Registers		
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

Offset	Transfer Controller 0 BYTE ADDRESS	Transfer Controller 1 BYTE ADDRESS	Acronym	Register Description
0h	0x01C0 8000	0x01C0 8400	PID	Peripheral Identification Register
4h	0x01C0 8004	0x01C0 8404	TCCFG	EDMA3TC Configuration Register
100h	0x01C0 8100	0x01C0 8500	TCSTAT	EDMA3TC Channel Status Register
120h	0x01C0 8120	0x01C0 8520	ERRSTAT	Error Status Register
124h	0x01C0 8124	0x01C0 8524	ERREN	Error Enable Register
128h	0x01C0 8128	0x01C0 8528	ERRCLR	Error Clear Register
12Ch	0x01C0 812C	0x01C0 852C	ERRDET	Error Details Register
130h	0x01C0 8130	0x01C0 8530	ERRCMD	Error Interrupt Command Register
140h	0x01C0 8140	0x01C0 8540	RDRATE	Read Command Rate Register
240h	0x01C0 8240	0x01C0 8640	SAOPT	Source Active Options Register
244h	0x01C0 8244	0x01C0 8644	SASRC	Source Active Source Address Register
248h	0x01C0 8248	0x01C0 8648	SACNT	Source Active Count Register
24Ch	0x01C0 824C	0x01C0 864C	SADST	Source Active Destination Address Register
250h	0x01C0 8250	0x01C0 8650	SABIDX	Source Active B-Index Register
254h	0x01C0 8254	0x01C0 8654	SAMPPRXY	Source Active Memory Protection Proxy Register
258h	0x01C0 8258	0x01C0 8658	SACNTRLD	Source Active Count Reload Register
25Ch	0x01C0 825C	0x01C0 865C	SASRCBREF	Source Active Source Address B-Reference Register
260h	0x01C0 8260	0x01C0 8660	SADSTBREF	Source Active Destination Address B-Reference Register
280h	0x01C0 8280	0x01C0 8680	DFCNTRLD	Destination FIFO Set Count Reload Register
284h	0x01C0 8284	0x01C0 8684	DFSRCBREF	Destination FIFO Set Source Address B-Reference Register
288h	0x01C0 8288	0x01C0 8688	DFDSTBREF	Destination FIFO Set Destination Address B-Reference Register
300h	0x01C0 8300	0x01C0 8700	DFOPT0	Destination FIFO Options Register 0
304h	0x01C0 8304	0x01C0 8704	DFSRC0	Destination FIFO Source Address Register 0
308h	0x01C0 8308	0x01C0 8708	DFCNT0	Destination FIFO Count Register 0
30Ch	0x01C0 830C	0x01C0 870C	DFDST0	Destination FIFO Destination Address Register 0

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

Offset	Transfer Controller 0 BYTE ADDRESS	Transfer Controller 1 BYTE ADDRESS	Acronym	Register Description
310h	0x01C0 8310	0x01C0 8710	DFBIDX0	Destination FIFO B-Index Register 0
314h	0x01C0 8314	0x01C0 8714	DFMPPRXY0	Destination FIFO Memory Protection Proxy Register 0
340h	0x01C0 8340	0x01C0 8740	DFOPT1	Destination FIFO Options Register 1
344h	0x01C0 8344	0x01C0 8744	DFSRC1	Destination FIFO Source Address Register 1
348h	0x01C0 8348	0x01C0 8748	DFCNT1	Destination FIFO Count Register 1
34Ch	0x01C0 834C	0x01C0 874C	DFDST1	Destination FIFO Destination Address Register 1
350h	0x01C0 8350	0x01C0 8750	DFBIDX1	Destination FIFO B-Index Register 1
354h	0x01C0 8354	0x01C0 8754	DFMPPRXY1	Destination FIFO Memory Protection Proxy Register 1
380h	0x01C0 8380	0x01C0 8780	DFOPT2	Destination FIFO Options Register 2
384h	0x01C0 8384	0x01C0 8784	DFSRC2	Destination FIFO Source Address Register 2
388h	0x01C0 8388	0x01C0 8788	DFCNT2	Destination FIFO Count Register 2
38Ch	0x01C0 838C	0x01C0 878C	DFDST2	Destination FIFO Destination Address Register 2
390h	0x01C0 8390	0x01C0 8790	DFBIDX2	Destination FIFO B-Index Register 2
394h	0x01C0 8394	0x01C0 8794	DFMPPRXY2	Destination FIFO Memory Protection Proxy Register 2
3C0h	0x01C0 83C0	0x01C0 87C0	DFOPT3	Destination FIFO Options Register 3
3C4h	0x01C0 83C4	0x01C0 87C4	DFSRC3	Destination FIFO Source Address Register 3
3C8h	0x01C0 83C8	0x01C0 87C8	DFCNT3	Destination FIFO Count Register 3
3CCh	0x01C0 83CC	0x01C0 87CC	DFDST3	Destination FIFO Destination Address Register 3
3D0h	0x01C0 83D0	0x01C0 87D0	DFBIDX3	Destination FIFO B-Index Register 3
3D4h	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

HEX ADDRESS RANGE	DESCRIPTION
0x01C0 4000 - 0x01C0 401F	Parameters Set 0 (8 32-bit words)
0x01C0 4020 - 0x01C0 403F	Parameters Set 1 (8 32-bit words)
0x01C0 4040 - 0x01cC0 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

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

Table 6-15. Parameter Set Entries (continued)

HEX OFFSET ADDRESS WITHIN THE PARAMETER SET	ACRONYM	PARAMETER ENTRY
0x0018	SRC_DST_CIDX	Source C Index, Destination C Index
0x001C	CCNT	C Count

Table 6-16. EDMA Events

Event	Event Name / Source	Event	Event Name / Source
0	McASP0 Receive	16	MMCSD Receive
1	McASP0 Transmit	17	MMCSD Transmit
2	McASP1 Receive	18	SPI1 Receive
3	McASP1 Transmit	19	SPI1 Transmit
4	McASP2 Receive	20	Reserved
5	McASP2 Transmit	21	Reserved
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 OMAP-L137 . It is primarily intended to support asynchronous memory types, such as NAND and NOR flash and Asynchronous SRAM. However on OMAP-L137 EMIFA also provides a secondary interface to SDRAM.

6.10.1 EMIFA Asynchronous Memory Support

EMIFA supports asynchronous:

- SRAM memories
- NAND Flash memories
- NOR Flash memories

The EMIFA data bus width is up to 16-bits on the ZKB package .The device supports up to fifteen address lines and an external wait/interrupt input. Up to four asynchronous chip selects are supported by EMIFA (EMA_CS[5:2]) . All four chip selects are available on the ZKB package.

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 Synchronous DRAM Memory Support

The OMAP-L137 ZKB package supports 16-bit SDRAM in addition to the asynchronous memories listed in Section 6.10.1. It has a single SDRAM chip select (EMA_CS[0]). SDRAM configurations that are supported are:

- One, Two, and Four Bank SDRAM devices
- Devices with Eight, Nine, Ten, and Eleven Column Address
- CAS Latency of two or three clock cycles
- Sixteen Bit Data Bus Width
- 3.3V LVCMOS Interface

Additionally, the SDRAM interface of EMIFA supports placing the SDRAM in Self Refresh and Powerdown Modes. Self Refresh mode allows the SDRAM to be put into a low power state while still retaining memory contents; since the SDRAM will continue to refresh itself even without clocks from the DSP. Powerdown mode achieves even lower power, except the DSP must periodically wake the SDRAM up and issue refreshes if data retention is required.

Finally, note that the EMIFA does not support Mobile SDRAM devices.

6.10.3 EMIFA Connection Examples

Figure 6-11 illustrates an example of how SDRAM, NOR, and NAND flash devices might be connected to EMIFA of a OMAP-L137 device simultaneously. The SDRAM chip select must be EMA_CS[0]. Note that the NOR flash is connected to EMA_CS[2] and the NAND flash is connected to EMA_CS[3] in this example. Note that any type of asynchronous memory may be connected to EMA_CS[5:2].

The on-chip bootloader makes some assumptions on which chip select the contains the boot image, and this depends on the boot mode. For NOR boot mode; the on-chip bootloader requires that the image be stored in NOR flash on EMA_CS[2]. For NAND boot mode, the bootloader requires that the boot image is stored in NAND flash on EMA_CS[3]. It is always possible to have the image span multiple chip selects, but this must be supported by second stage boot code stored in the external flash.

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.

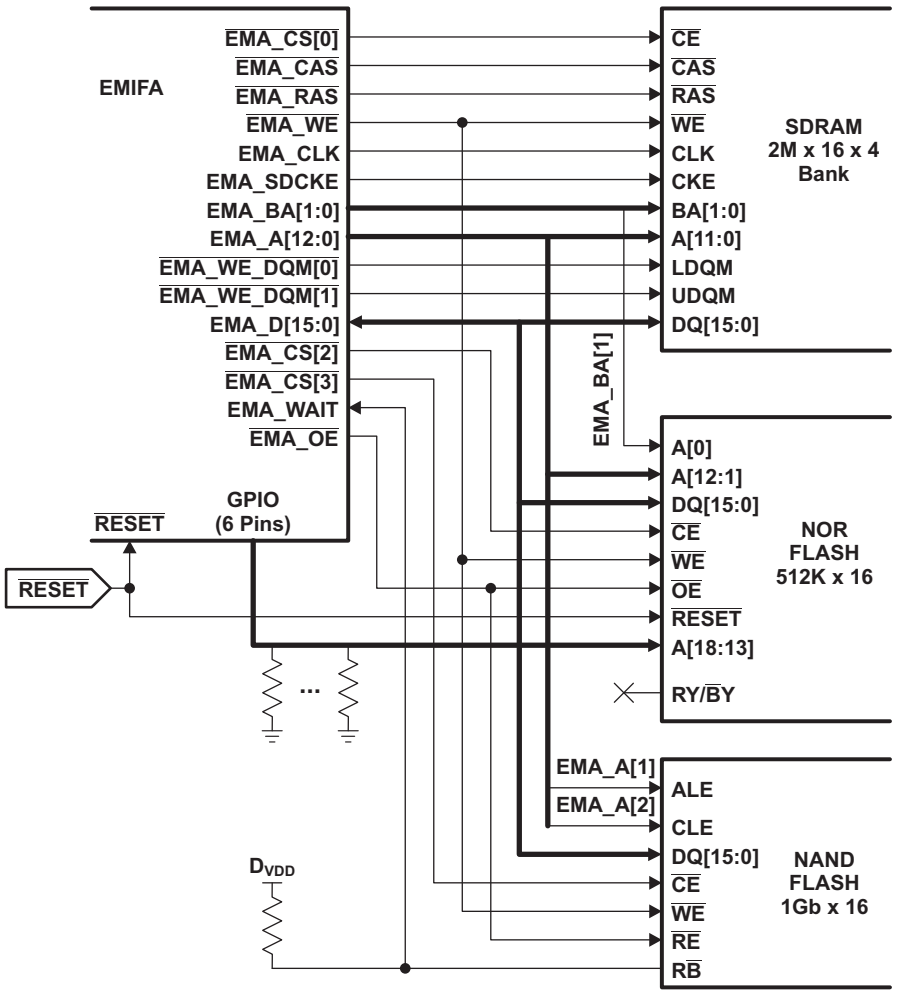


Figure 6-11. OMAP-L137 Connection Diagram: SDRAM, NOR, NAND

PRODUCT PREVIEW

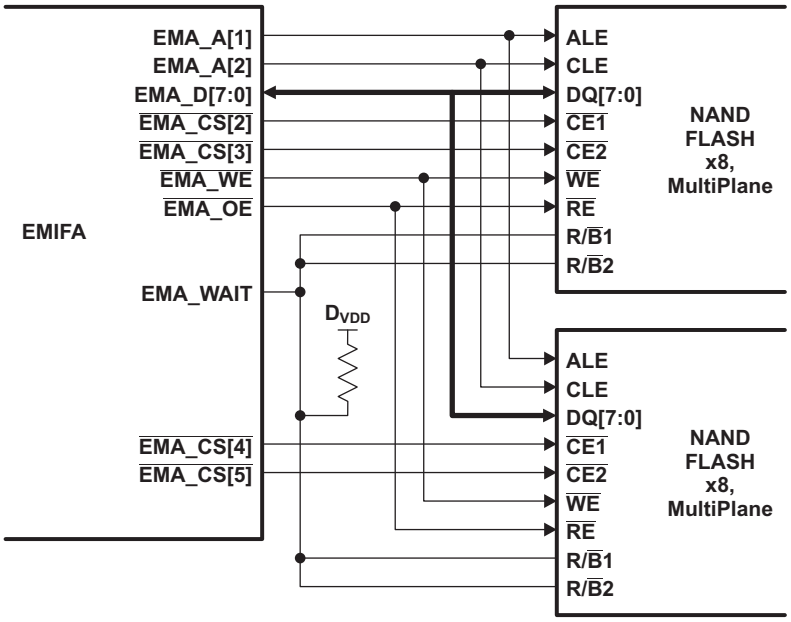


Figure 6-12. OMAP-L137 EMIFA Connection Diagram: Multiple NAND Flash Planes

6.10.4 External Memory Interface (EMIF)

Table 6-17 is a list of the EMIF registers. For more information about these registers, see the C674x DSP External Memory Interface (EMIF) User's Guide (literature number SPRU711).

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

BYTE ADDRESS	Register Name	Register Description
0x6800 0000	MIDR	Module ID Register
0x6800 0004	AWCC	Asynchronous Wait Cycle Configuration Register
0x6800 0008	SDCR	SDRAM Configuration Register
0x6800 000C	SDRCR	SDRAM Refresh Control Register
0x6800 0010	CE2CFG	Asynchronous 1 Configuration Register
0x6800 0014	CE3CFG	Asynchronous 2 Configuration Register
0x6800 0018	CE4CFG	Asynchronous 3 Configuration Register
0x6800 001C	CE5CFG	Asynchronous 4 Configuration Register
0x6800 0020	SDTIMR	SDRAM Timing Register
0x6800 003C	SDSRETR	SDRAM Self Refresh Exit Timing Register
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	NANDF3ECC	NAND Flash 3 ECC Register (CS4 Space)
0x6800 007C	NANDF4ECC	NAND Flash 4 ECC Register (CS5 Space)
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

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

BYTE ADDRESS	Register Name	Register Description
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

6.10.5 EMIFA Electrical Data/Timing

Table 6-18 through Table 6-21 assume testing over recommended operating conditions.

Table 6-18. EMIFA SDRAM Interface Timing Requirements

NO.			MIN	MAX	UNIT
19	$t_{su(EMA\_DV-EM\_CLKH)}$	Input setup time, read data valid on EMA_D[31:0] before EMA_CLK rising	1		ns
20	$t_{h(CLKH-DIV)}$	Input hold time, read data valid on EMA_D[31:0] after EMA_CLK rising	1.5		ns

Table 6-19. EMIFA SDRAM Interface Switching Characteristics

NO.	PARAMETER		MIN	MAX	UNIT
1	$t_{c(CLK)}$	Cycle time, EMIF clock EMA_CLK	10		ns
2	$t_{w(CLK)}$	Pulse width, EMIF clock EMA_CLK high or low	3		ns
3	$t_{d(CLKH-CSV)}$	Delay time, EMA_CLK rising to $\overline{EMA\_CS[0]}$ valid		7	ns
4	$t_{oh(CLKH-CSIV)}$	Output hold time, EMA_CLK rising to $\overline{EMA\_CS[0]}$ invalid	1		ns
5	$t_{d(CLKH-DQMV)}$	Delay time, EMA_CLK rising to EMA_$\overline{WE}$_DQM[1:0] valid		7	ns
6	$t_{oh(CLKH-DQMIV)}$	Output hold time, EMA_CLK rising to EMA_$\overline{WE}$_DQM[1:0] invalid	1		ns
7	$t_{d(CLKH-AV)}$	Delay time, EMA_CLK rising to EMA_A[12:0] and EMA_BA[1:0] valid		7	ns
8	$t_{oh(CLKH-AIV)}$	Output hold time, EMA_CLK rising to EMA_A[12:0] and EMA_BA[1:0] invalid	1		ns
9	$t_{d(CLKH-DV)}$	Delay time, EMA_CLK rising to EMA_D[15:0] valid		7	ns
10	$t_{oh(CLKH-DIV)}$	Output hold time, EMA_CLK rising to EMA_D[15:0] invalid	1		ns
11	$t_{d(CLKH-RASV)}$	Delay time, EMA_CLK rising to $\overline{EMA\_RAS}$ valid		7	ns
12	$t_{oh(CLKH-RASIV)}$	Output hold time, EMA_CLK rising to $\overline{EMA\_RAS}$ invalid	1		ns
13	$t_{d(CLKH-CASV)}$	Delay time, EMA_CLK rising to $\overline{EMA\_CAS}$ valid		7	ns
14	$t_{oh(CLKH-CASIV)}$	Output hold time, EMA_CLK rising to $\overline{EMA\_CAS}$ invalid	1		ns
15	$t_{d(CLKH-WEV)}$	Delay time, EMA_CLK rising to $\overline{EMA\_WE}$ valid		7	ns
16	$t_{oh(CLKH-WEIV)}$	Output hold time, EMA_CLK rising to $\overline{EMA\_WE}$ invalid	1		ns
17	$t_{dis(CLKH-DHZ)}$	Delay time, EMA_CLK rising to EMA_D[15:0] tri-stated		7	ns
18	$t_{ena(CLKH-DLZ)}$	Output hold time, EMA_CLK rising to EMA_D[15:0] driving	1		ns

Table 6-20. EMIFA Asynchronous Memory Timing Requirements(1)

NO.			OMAP-L137			UNIT
			MIN	Nom	MAX	
READS and WRITES						
2	$t_{w(EM\_WAIT)}$	Pulse duration, EM_WAIT assertion and deassertion	2E			ns
READS						
12	$t_{su(EMDV-EMOEH)}$	Setup time, EM_D[15:0] valid before $\overline{EM\_OE}$ high	3			ns
13	$t_{h(EMOEH-EMDIV)}$	Hold time, EM_D[15:0] valid after $\overline{EM\_OE}$ high	0.5			ns
14	$t_{su(EMOEL-EMWAIT)}$	Setup Time, EM_WAIT asserted before end of Strobe Phase(2)	4E+3			ns
WRITES						

(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 6-17 and Figure 6-18 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 6-20. EMIFA Asynchronous Memory Timing Requirements (continued)

NO.			OMAP-L137			UNIT
			MIN	Nom	MAX	
28	$t_{su(EMWEL-EMWAIT)}$	Setup Time, EM_WAIT asserted before end of Strobe Phase(2)	4E+3			ns

Table 6-21. EMIFA Asynchronous Memory Switching Characteristics(1)(2)(3)

NO.	PARAMETER		OMAP-L137			UNIT
			MIN	Nom	MAX	
READS and WRITES						
1	$t_{d(TURNAROUND)}$	Turn around time	(TA)*E - 3	(TA)*E	(TA)*E + 3	ns
READS						
3	$t_{c(EMRCYCLE)}$	EMIF read cycle time (EW = 0)	(RS+RST+RH)*E - 3	(RS+RST+RH)*E	(RS+RST+RH)*E + 3	ns
		EMIF read cycle time (EW = 1)	(RS+RST+RH+(EWC*16))*E - 3	(RS+RST+RH+(EWC*16))*E	(RS+RST+RH+(EWC*16))*E + 3	ns
4	$t_{su(EMCEL-EMOEL)}$	Output setup time, $\overline{EMA\_CE[5:2]}$ low to $\overline{EMA\_OE}$ low (SS = 0)	(RS)*E-3	(RS)*E	(RS)*E+3	ns
		Output setup time, $\overline{EMA\_CE[5:2]}$ low to $\overline{EMA\_OE}$ low (SS = 1)	-3	0	+3	ns
5	$t_{h(EMOEH-EMCEH)}$	Output hold time, $\overline{EMA\_OE}$ high to $\overline{EMA\_CE[5:2]}$ high (SS = 0)	(RH)*E - 3	(RH)*E	(RH)*E + 3	ns
		Output hold time, $\overline{EMA\_OE}$ high to $\overline{EMA\_CE[5:2]}$ high (SS = 1)	-3	0	+3	ns
6	$t_{su(EMBAV-EMOEL)}$	Output setup time, $\overline{EMA\_BA[1:0]}$ valid to $\overline{EMA\_OE}$ low	(RS)*E-3	(RS)*E	(RS)*E+3	ns
7	$t_{h(EMOEH-EMBAIV)}$	Output hold time, $\overline{EMA\_OE}$ high to $\overline{EMA\_BA[1:0]}$ invalid	(RH)*E-3	(RH)*E	(RH)*E+3	ns
8	$t_{su(EMBAV-EMOEL)}$	Output setup time, $\overline{EMA\_A[13:0]}$ valid to $\overline{EMA\_OE}$ low	(RS)*E-3	(RS)*E	(RS)*E+3	ns
9	$t_{h(EMOEH-EMAIV)}$	Output hold time, $\overline{EMA\_OE}$ high to $\overline{EMA\_A[13:0]}$ invalid	(RH)*E-3	(RH)*E	(RH)*E+3	ns
10	$t_{w(EMOEL)}$	$\overline{EMA\_OE}$ active low width (EW = 0)	(RST)*E-3	(RST)*E	(RST)*E+3	ns
		$\overline{EMA\_OE}$ active low width (EW = 1)	(RST+(EWC*16))*E-3	(RST+(EWC*16))*E	(RST+(EWC*16))*E+3	ns
11	$t_{d(EMWAITH-EMOEH)}$	Delay time from EMA_WAIT deasserted to $\overline{EMA\_OE}$ high	3E-3	4E	4E+3	ns
WRITES						
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*16))*E - 3	(WS+WST+WH+(EWC*16))*E	(WS+WST+WH+(EWC*16))*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

(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]. See the OMAP-L137 Asynchronous External Memory Interface (EMIF) User's Guide (SPRUED1) for more information.

(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. See the OMAP-L137 Asynchronous External Memory Interface (EMIF) User's Guide (SPRUED1) for more information.

Table 6-21. EMIFA Asynchronous Memory Switching Characteristics (continued)

NO.	PARAMETER		OMAP-L137			UNIT
			MIN	Nom	MAX	
17	$t_{h(EMWEH-EMCEH)}$	Output hold time, EMA_WE high to EMA_CE[5:2] high (SS = 0)	(WH)*E-3	(WH)*E	(WH)*E+3	ns
		Output hold time, EMA_WE high to EMA_CE[5:2] high (SS = 1)	-3	0	+3	ns
18	$t_{su(EMDQMV-EMWEL)}$	Output setup time, EMA_BA[1:0] valid to EMA_WE low	(WS)*E-3	(WS)*E	(WS)*E+3	ns
19	$t_{h(EMWEH-EMDQMIV)}$	Output hold time, EMA_WE high to EMA_BA[1:0] invalid	(WH)*E-3	(WH)*E	(WH)*E+3	ns
20	$t_{su(EMBAV-EMWEL)}$	Output setup time, EMA_BA[1:0] valid to EMA_WE low	(WS)*E-3	(WS)*E	(WS)*E+3	ns
21	$t_{h(EMWEH-EMBAIV)}$	Output hold time, EMA_WE high to EMA_BA[1:0] invalid	(WH)*E-3	(WH)*E	(WH)*E+3	ns
22	$t_{su(EMAV-EMWEL)}$	Output setup time, EMA_A[13:0] valid to EMA_WE low	(WS)*E-3	(WS)*E	(WS)*E+3	ns
23	$t_{h(EMWEH-EMAIV)}$	Output hold time, EMA_WE high to EMA_A[13:0] invalid	(WH)*E-3	(WH)*E	(WH)*E+3	ns
24	$t_{w(EMWEL)}$	EMA_WE active low width (EW = 0)	(WST)*E-3	(WST)*E	(WST)*E+3	ns
		EMA_WE active low width (EW = 1)	(WST+(EWC*16))*E-3	(WST+(EWC*16))*E	(WST+(EWC*16))*E+3	ns
25	$t_{d(EMWAITH-EMWEH)}$	Delay time from EMA_WAIT deasserted to EMA_WE high	3E-3	4E	4E+3	ns
26	$t_{su(EMDV-EMWEL)}$	Output setup time, EMA_D[15:0] valid to EMA_WE low	(WS)*E-3	(WS)*E	(WS)*E+3	ns
27	$t_{h(EMWEH-EMDIV)}$	Output hold time, EMA_WE high to EMA_D[15:0] invalid	(WH)*E-3	(WH)*E	(WH)*E+3	ns

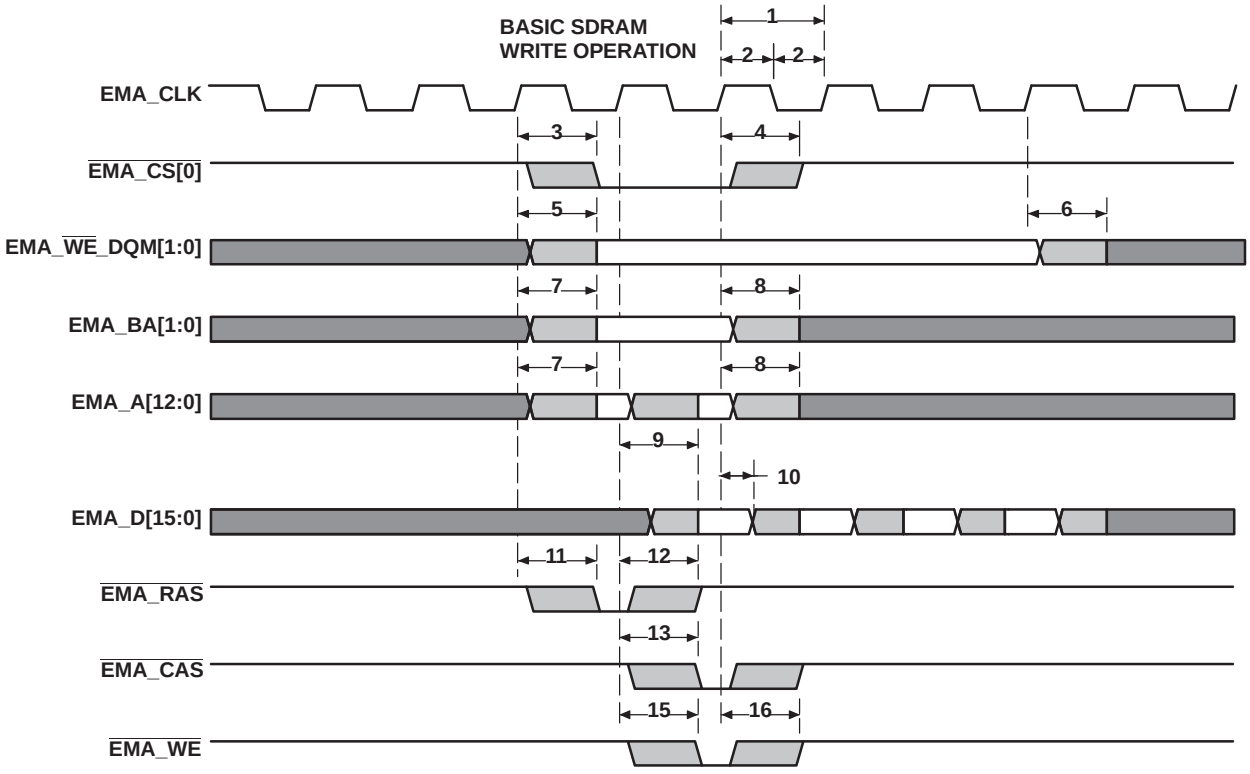


Figure 6-13. EMIFA Basic SDRAM Write Operation

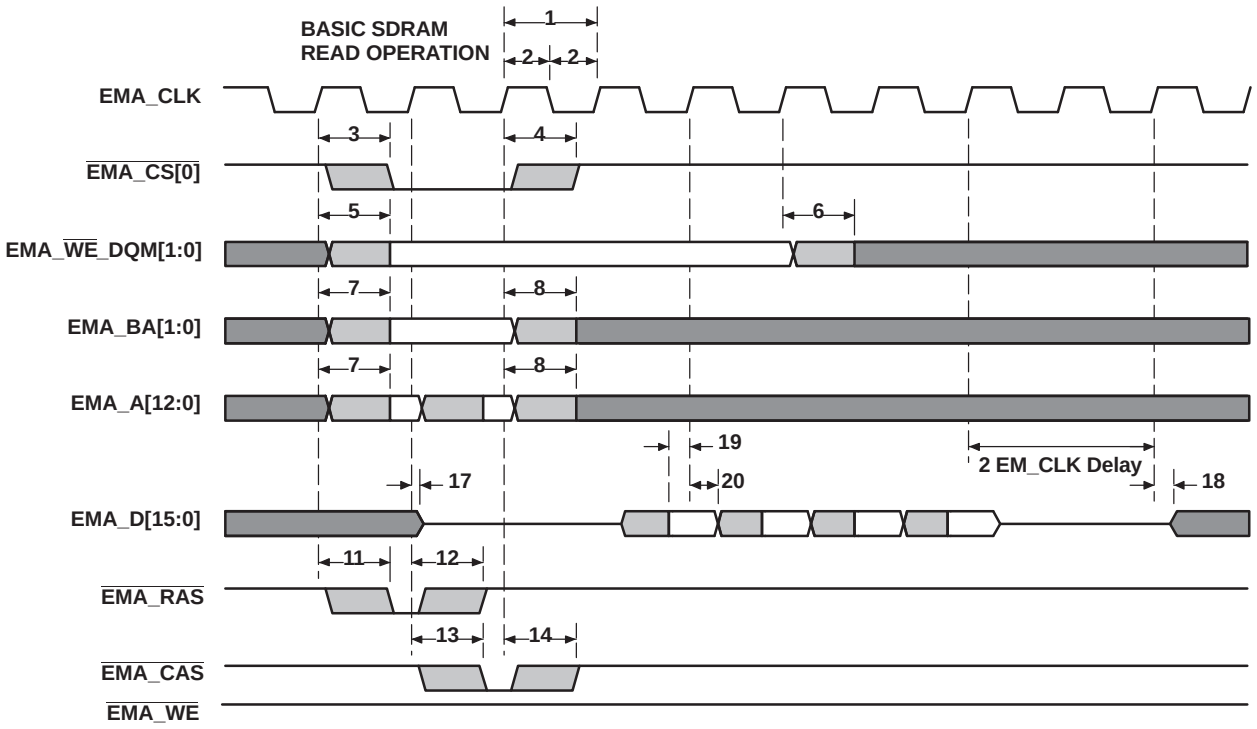


Figure 6-14. EMIFA Basic SDRAM Read Operation

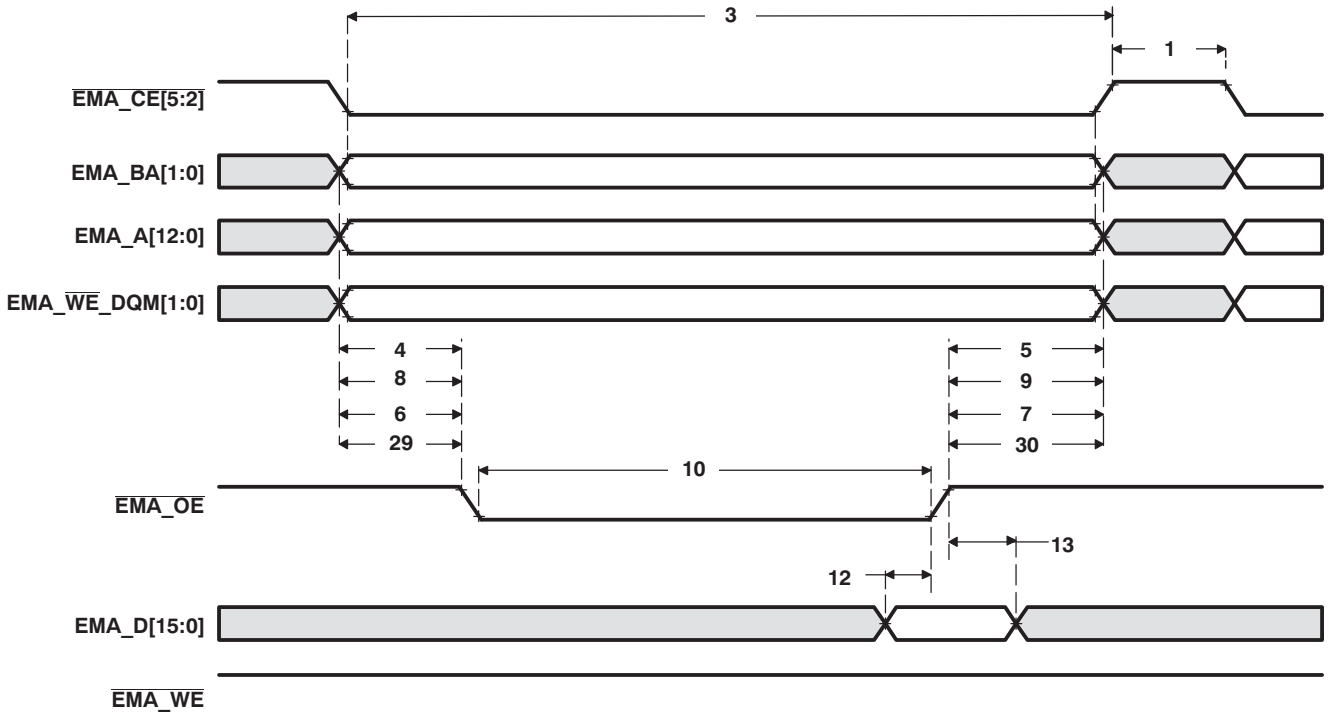


Figure 6-15. Asynchronous Memory Read Timing for EMIFA

PRODUCT PREVIEW

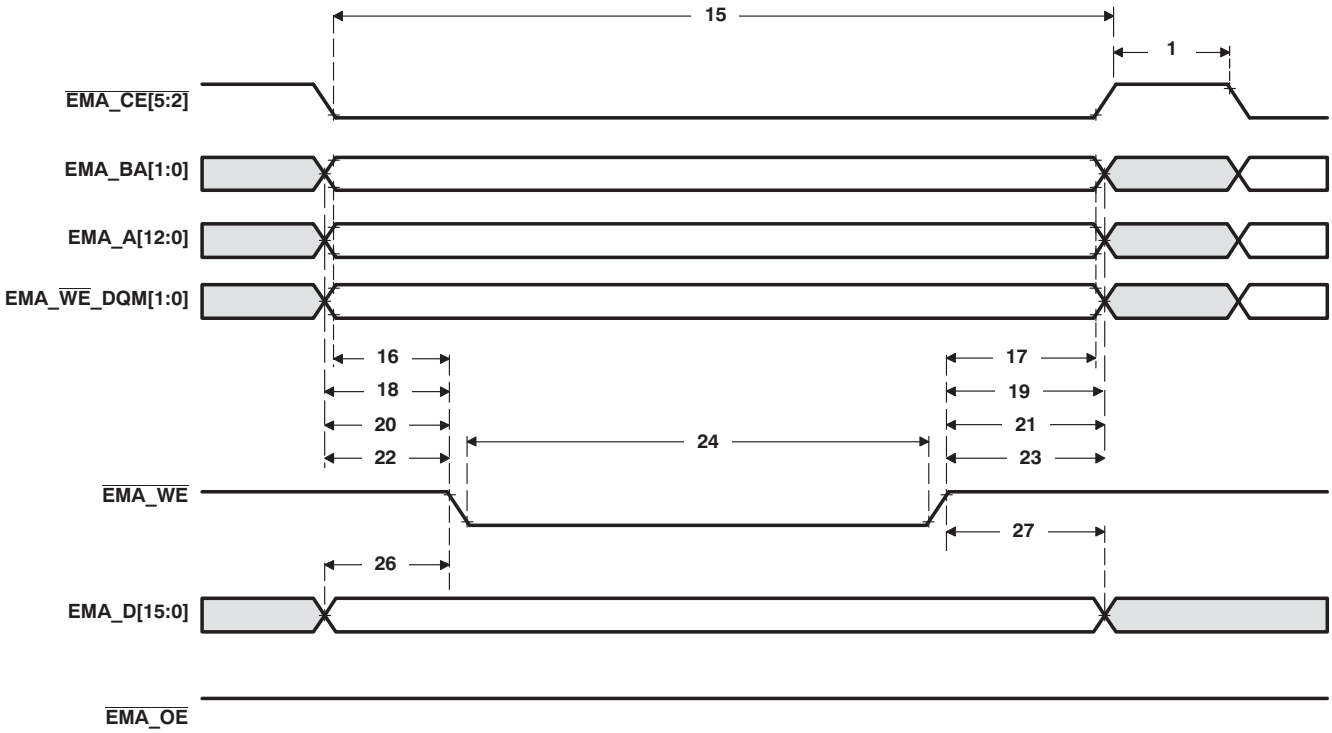


Figure 6-16. Asynchronous Memory Write Timing for EMIFA

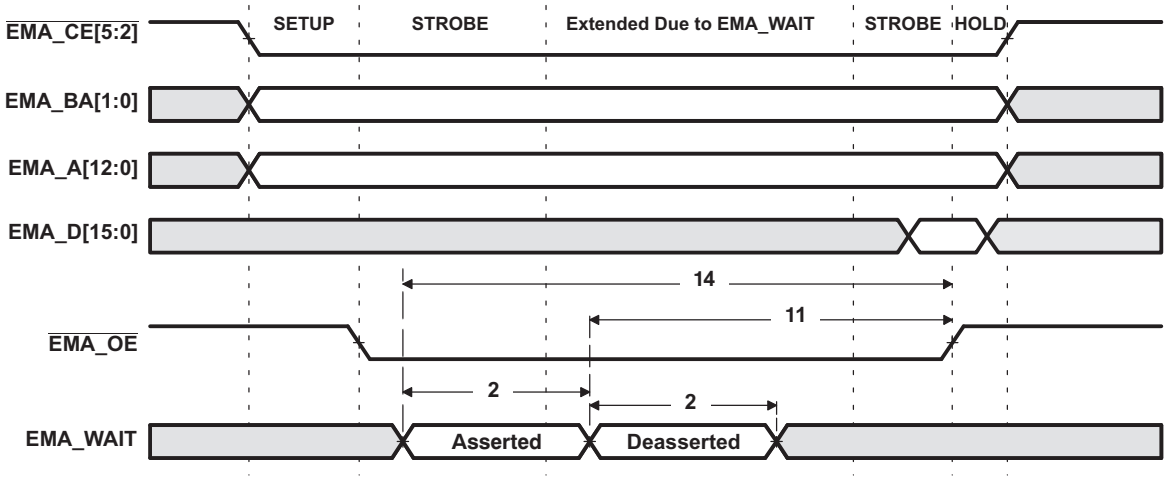


Figure 6-17. EMA_WAIT Read Timing Requirements

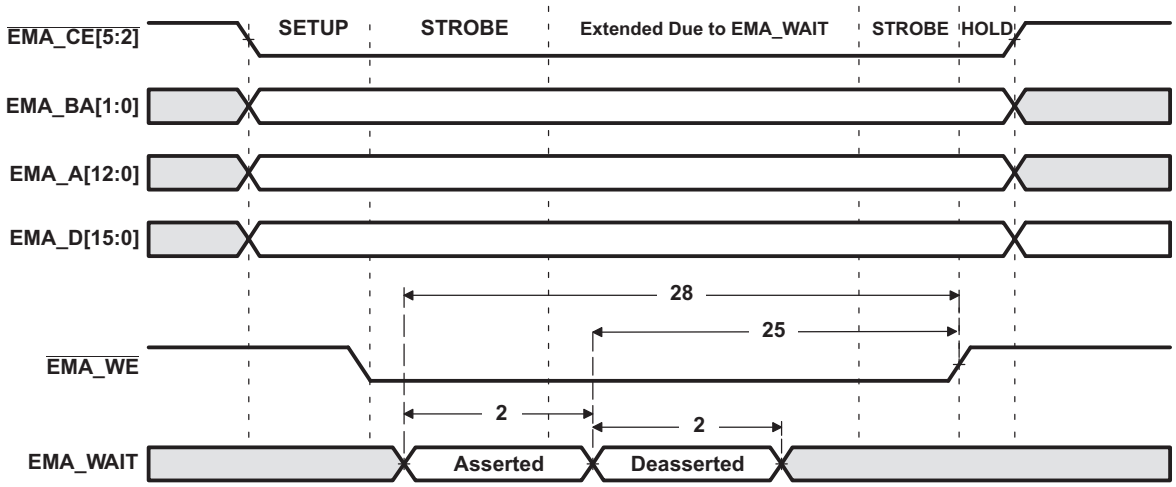


Figure 6-18. EMA_WAIT Write Timing Requirements

6.11 EMIFB Peripheral Registers Description(s)

Figure 6-19, 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 crossbar in the figure). The EMIFB implements a split transaction internal bus, allowing concurrence between reads and writes from the various requesters.

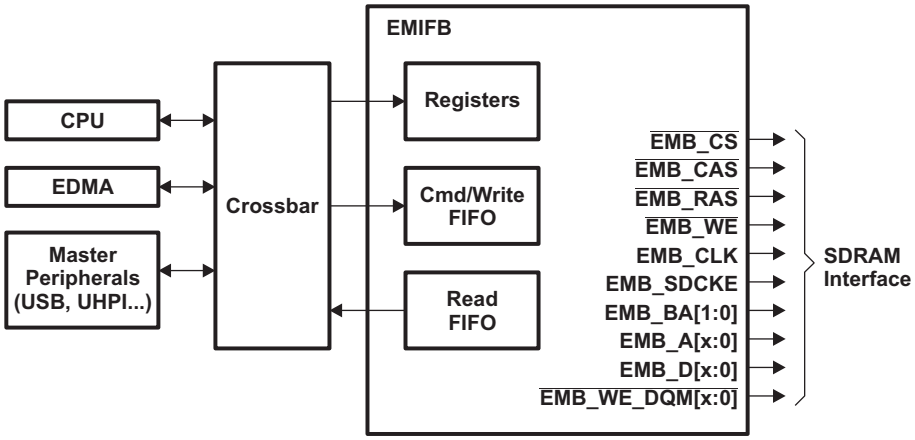


Figure 6-19. EMIFB Functional Block Diagram

6.11.1 Interfacing to SDRAM

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

- Pre-charge bit is A[10]
- The number of column address bits is 8, 9, 10 or 11
- The number of row address bits is 13 (in case of mobile SDR, number of row address bits can be 9, 10, 11, 12, or 13)
- The number of internal banks is 1, 2 or 4

Figure 6-20 shows an interface between the EMIFB and a 2M × 16 × 4 bank SDRAM device. In addition, Figure 6-21 shows an interface between the EMIFB and a 2M × 32 × 4 bank SDRAM device and Figure 6-22 shows an interface between the EMIFB and two 4M × 16 × 4 bank SDRAM devices. Refer to Table 6-22, as an example that shows additional list of commonly-supported SDRAM devices and the required connections for the address pins. Note that in Table 6-22, page size/column size (not indicated in the table) is varied to get the required addressability range.

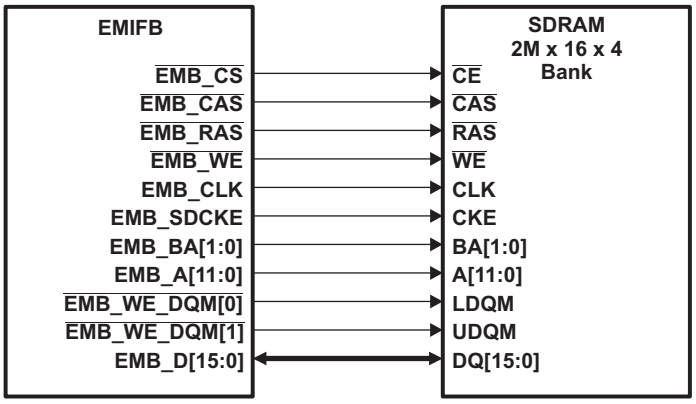


Figure 6-20. EMIFB to 2M × 16 × 4 bank SDRAM Interface

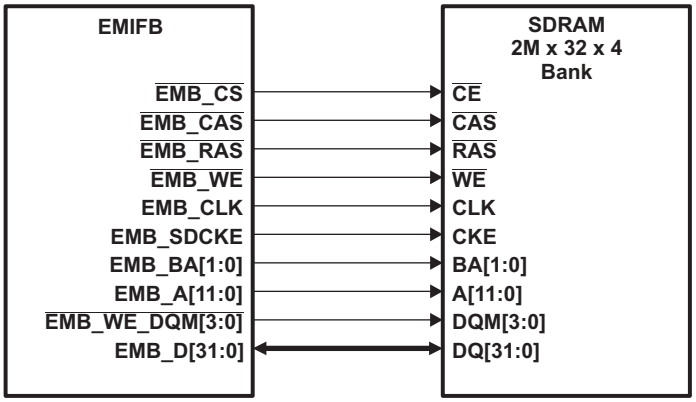


Figure 6-21. EMIFB to 2M × 32 × 4 bank SDRAM Interface

PRODUCT PREVIEW

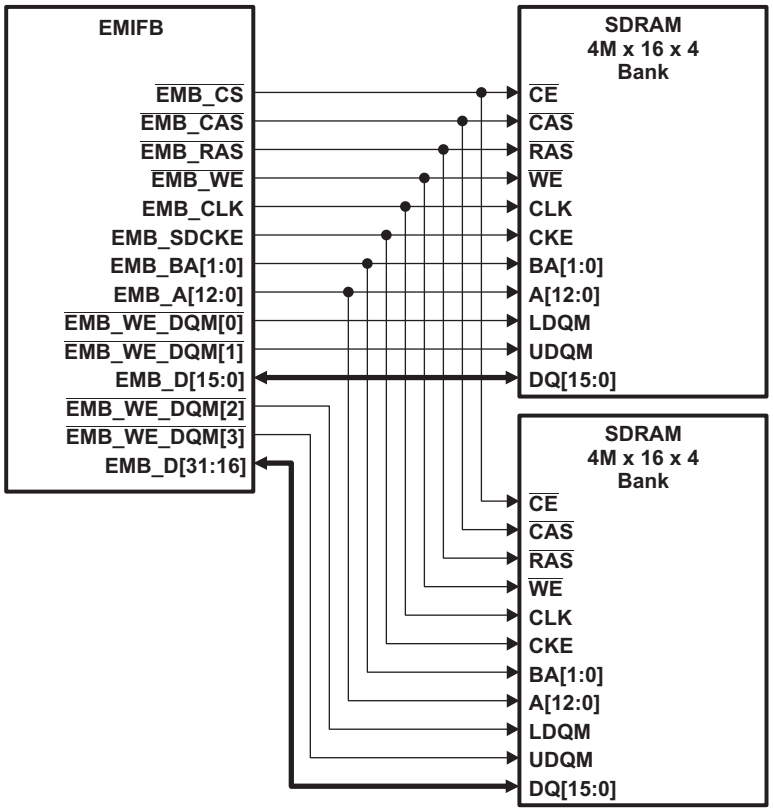


Figure 6-22. EMIFB to Dual 4M × 16 × 4 bank SDRAM Interface

Table 6-22. Example of 16/32-bit EMIFB Address Pin Connections

SDRAM Size	Width	Banks		Address Pins
64M bits	×16	4	SDRAM	A[11:0]
			EMIFB	EMB_A[11:0]
	×32	4	SDRAM	A[10:0]
			EMIFB	EMB_A[10:0]
128M bits	×16	4	SDRAM	A[11:0]
			EMIFB	EMB_A[11:0]
	×32	4	SDRAM	A[11:0]
			EMIFB	EMB_A[11:0]
256M bits	×16	4	SDRAM	A[12:0]
			EMIFB	EMB_A[12:0]
	×32	4	SDRAM	A[11:0]
			EMIFB	EMB_A[11:0]
512M bits	×16	4	SDRAM	A[12:0]
			EMIFB	EMB_A[12:0]
	×32	4	SDRAM	A[12:0]
			EMIFB	EMB_A[12:0]

Table 6-23 is a list of the EMIFB registers.

Table 6-23. EMIFB Base Controller Registers

BYTE ADDRESS	Acronym	Register
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
0xB000 00C4	IMR	Interrupt Mask Register
0xB000 00C8	IMSR	Interrupt Mask Set Register
0xB000 00CC	IMCR	Interrupt Mask Clear Register

6.11.2 EMIFB Electrical Data/Timing

Table 6-24. EMIFB SDRAM Interface Timing Requirements

NO.			MIN	MAX	UNIT
19	$t_{su(EMA\_DV-EM\_CLKH)}$	Input setup time, read data valid on EMB_D[31:0] before EMB_CLK rising	0.5		ns
20	$t_{h(CLKH-DIV)}$	Input hold time, read data valid on EMB_D[31:0] after EMB_CLK rising	1.5		ns

Table 6-25. EMIFB SDRAM Interface Switching Characteristics

NO.	PARAMETER		MIN	MAX	UNIT
1	$t_{c(CLK)}$	Cycle time, EMIF clock EMB_CLK	7.5		ns
2	$t_{w(CLK)}$	Pulse width, EMIF clock EMB_CLK high or low	3		ns
3	$t_{d(CLKH-CSV)}$	Delay time, EMB_CLK rising to $\overline{\text{EMB\_CS[0]}}$ valid		5.1	ns
4	$t_{oh(CLKH-CSIV)}$	Output hold time, EMB_CLK rising to $\overline{\text{EMB\_CS[0]}}$ invalid	0.9		ns
5	$t_{d(CLKH-DQMV)}$	Delay time, EMB_CLK rising to EMB_$\overline{\text{WE}}$_DQM[3:0] valid		5.1	ns
6	$t_{oh(CLKH-DQMIV)}$	Output hold time, EMB_CLK rising to EMB_$\overline{\text{WE}}$_DQM[3:0] invalid	0.9		ns
7	$t_{d(CLKH-AV)}$	Delay time, EMB_CLK rising to EMB_A[12:0] and EMB_BA[1:0] valid		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	0.9		ns
9	$t_{d(CLKH-DV)}$	Delay time, EMB_CLK rising to EMB_D[31:0] valid		5.1	ns
10	$t_{oh(CLKH-DIV)}$	Output hold time, EMB_CLK rising to EMB_D[31:0] invalid	0.9		ns
11	$t_{d(CLKH-RASV)}$	Delay time, EMB_CLK rising to $\overline{\text{EMB\_RAS}}$ valid		5.1	ns
12	$t_{oh(CLKH-RASIV)}$	Output hold time, EMB_CLK rising to $\overline{\text{EMB\_RAS}}$ invalid	0.9		ns
13	$t_{d(CLKH-CASV)}$	Delay time, EMB_CLK rising to $\overline{\text{EMB\_CAS}}$ valid		5.1	ns
14	$t_{oh(CLKH-CASIV)}$	Output hold time, EMB_CLK rising to $\overline{\text{EMB\_CAS}}$ invalid	0.9		ns
15	$t_{d(CLKH-WEV)}$	Delay time, EMB_CLK rising to $\overline{\text{EMB\_WE}}$ valid		5.1	ns
16	$t_{oh(CLKH-WEIV)}$	Output hold time, EMB_CLK rising to $\overline{\text{EMB\_WE}}$ invalid	0.9		ns
17	$t_{dis(CLKH-DHZ)}$	Delay time, EMB_CLK rising to EMB_D[31:0] tri-stated		5.1	ns
18	$t_{ena(CLKH-DLZ)}$	Output hold time, EMB_CLK rising to EMB_D[31:0] driving	0.9		ns

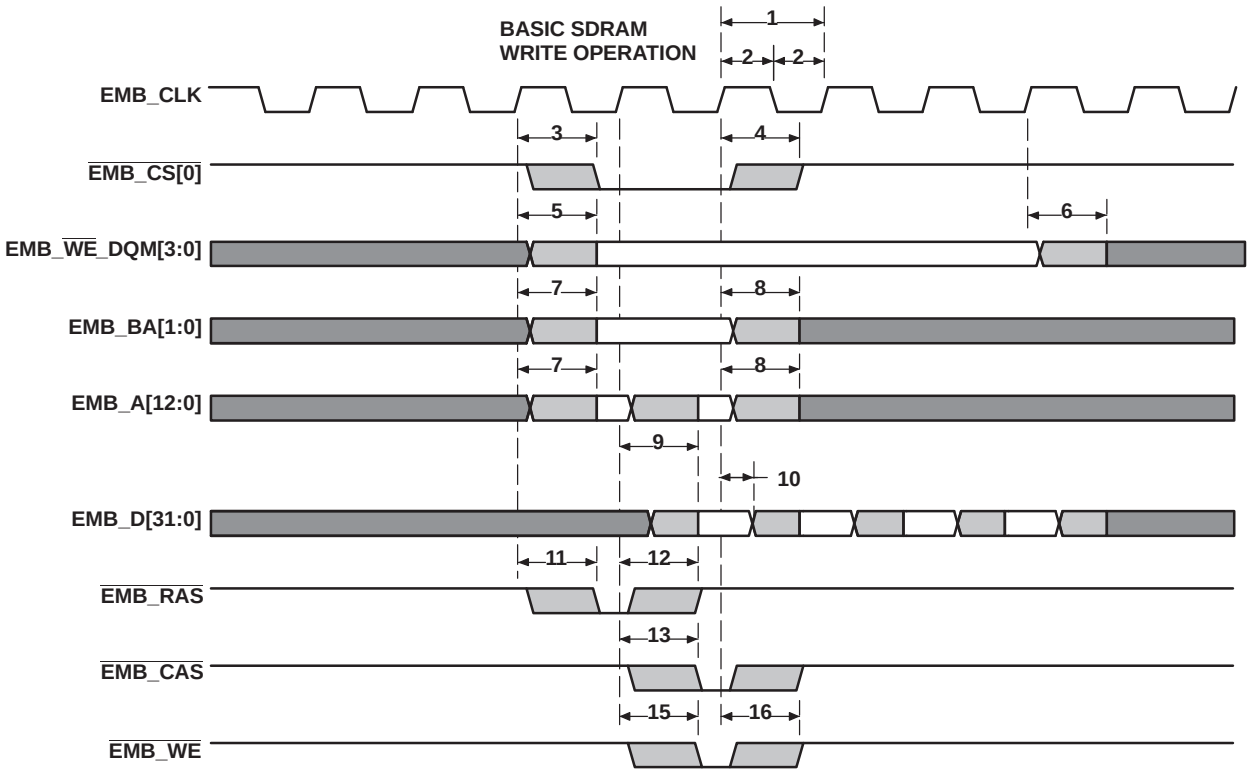


Figure 6-23. EMIFB Basic SDRAM Write Operation

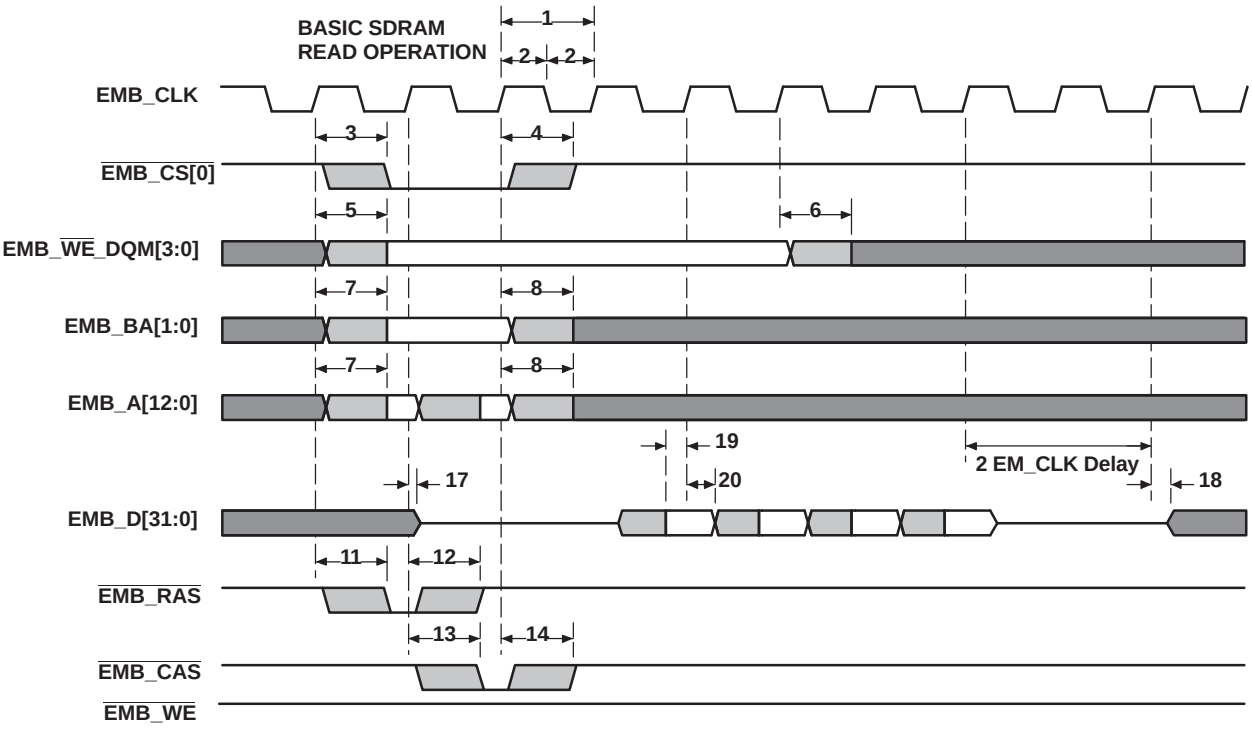


Figure 6-24. EMIFB Basic SDRAM Read Operation

6.12 MMC / SD / SDIO (MMCSD)

6.12.1 MMCSD Peripheral Description

The OMAP-L137 includes an MMCSD controller which is compliant with MMC V3.31, 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).
- Secure Digital (SD) Memory Card.
- MMC/SD protocol support.
- SDIO protocol support.
- Programmable clock frequency.
- 512 bit Read/Write FIFO to lower system overhead.
- Slave EDMA transfer capability.

The OMAP-L137 MMC/SD Controller does not support SPI mode.

6.12.2 MMCSD Peripheral Register Description(s)

Table 6-26. Multimedia Card/Secure Digital (MMC/SD) Card Controller Registers

Offset	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	MMC FIFO Control Register

6.12.3 MMC/SD Electrical Data/Timing

Table 6-27. Timing Requirements for MMC/SD Module (see Figure 6-26 and Figure 6-28)

NO.			MIN	MAX	UNIT
1	$t_{su(CMDV-CLKH)}$	Setup time, SD_CMD valid before SD_CLK high	TBD		ns
2	$t_{h(CLKH-CMDV)}$	Hold time, SD_CMD valid after SD_CLK high	TBD		ns
3	$t_{su(DATV-CLKH)}$	Setup time, SD_DATx valid before SD_CLK high	TBD		ns
4	$t_{h(CLKH-DATV)}$	Hold time, SD_DATx valid after SD_CLK high	TBD		ns

Table 6-28. Switching Characteristics Over Recommended Operating Conditions for MMC/SD Module (see Figure 6-25 through Figure 6-28)

NO.	PARAMETER		MIN	MAX	UNIT
7	$f_{(CLK)}$	Operating frequency, SD_CLK	TBD	TBD	MHz
8	$f_{(CLK\_ID)}$	Identification mode frequency, SD_CLK	TBD	TBD	KHz
9	$t_{W(CLKL)}$	Pulse width, SD_CLK low	TBD		ns
10	$t_{W(CLKH)}$	Pulse width, SD_CLK high	TBD		ns
11	$t_{r(CLK)}$	Rise time, SD_CLK		TBD	ns
12	$t_{f(CLK)}$	Fall time, SD_CLK		TBD	ns
13	$t_{d(CLKL-CMD)}$	Delay time, SD_CLK low to SD_CMD transition	TBD	TBD	ns
14	$t_{d(CLKL-DAT)}$	Delay time, SD_CLK low to SD_DATx transition	TBD	TBD	ns

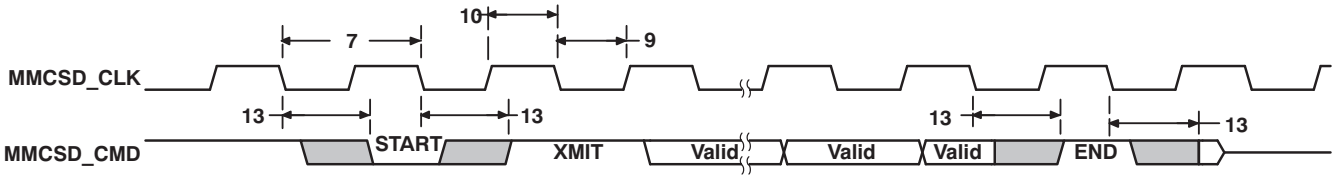


Figure 6-25. MMC/SD Host Command Timing

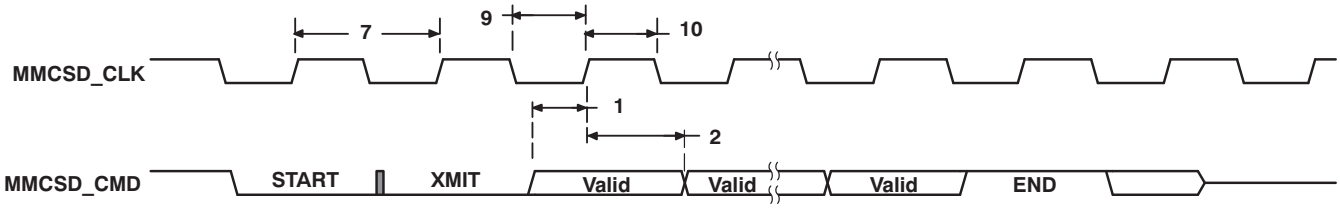


Figure 6-26. MMC/SD Card Response Timing

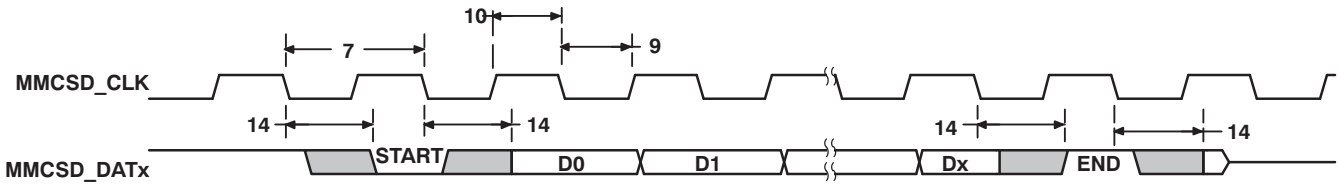


Figure 6-27. MMC/SD Host Write Timing

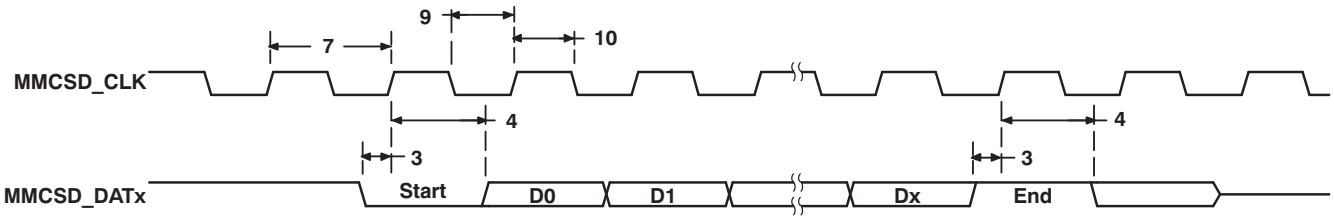


Figure 6-28. MMC/SD Host Read and Card CRC Status Timing

6.13 Ethernet Media Access Controller (EMAC)

The Ethernet Media Access Controller (EMAC) provides an efficient interface between OMAP-L137 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 OMAP-L137 device to the PHY. The MDIO module controls PHY configuration and status monitoring.

Both the EMAC and the MDIO modules interface to the OMAP-L137 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.

6.13.1 EMAC Peripheral Register Description(s)

Table 6-29. Ethernet Media Access Controller (EMAC) Registers

Offset	BYTE ADDRESS	REGISTER	Register Description
0h	0x01E2 3000	TXREV	Transmit Revision Register
4h	0x01E2 3004	TXCONTROL	Transmit Control Register
8h	0x01E2 3008	TXTEARDOWN	Transmit Teardown Register
10h	0x01E2 3010	RXREV	Receive Revision Register
14h	0x01E2 3014	RXCONTROL	Receive Control Register
18h	0x01E2 3018	RXTEARDOWN	Receive Teardown Register
80h	0x01E2 3080	TXINTSTATRAW	Transmit Interrupt Status (Unmasked) Register
84h	0x01E2 3084	TXINTSTATMASKED	Transmit Interrupt Status (Masked) Register
88h	0x01E2 3088	TXINTMASKSET	Transmit Interrupt Mask Set Register
8Ch	0x01E2 308C	TXINTMASKCLEAR	Transmit Interrupt Clear Register
90h	0x01E2 3090	MACINVECTOR	MAC Input Vector Register
94h	0x01E2 3094	MACEOIVECTOR	MAC End Of Interrupt Vector Register
A0h	0x01E2 30A0	RXINTSTATRAW	Receive Interrupt Status (Unmasked) Register
A4h	0x01E2 30A4	RXINTSTATMASKED	Receive Interrupt Status (Masked) Register
A8h	0x01E2 30A8	RXINTMASKSET	Receive Interrupt Mask Set Register
ACh	0x01E2 30AC	RXINTMASKCLEAR	Receive Interrupt Mask Clear Register
B0h	0x01E2 30B0	MACINTSTATRAW	MAC Interrupt Status (Unmasked) Register
B4h	0x01E2 30B4	MACINTSTATMASKED	MAC Interrupt Status (Masked) Register
B8h	0x01E2 30B8	MACINTMASKSET	MAC Interrupt Mask Set Register
BCh	0x01E2 30BC	MACINTMASKCLEAR	MAC Interrupt Mask Clear Register
100h	0x01E2 3100	RXMBPENABLE	Receive Multicast/Broadcast/Promiscuous Channel Enable Register
104h	0x01E2 3104	RXUNICASTSET	Receive Unicast Enable Set Register
108h	0x01E2 3108	RXUNICASTCLEAR	Receive Unicast Clear Register
10Ch	0x01E2 310C	RXMAXLEN	Receive Maximum Length Register
110h	0x01E2 3110	RXBUFFEROFFSET	Receive Buffer Offset Register
114h	0x01E2 3114	RXFILTERLOWTHRESH	Receive Filter Low Priority Frame Threshold Register
120h	0x01E2 3120	RX0FLOWTHRESH	Receive Channel 0 Flow Control Threshold Register
124h	0x01E2 3124	RX1FLOWTHRESH	Receive Channel 1 Flow Control Threshold Register
128h	0x01E2 3128	RX2FLOWTHRESH	Receive Channel 2 Flow Control Threshold Register
12Ch	0x01E2 312C	RX3FLOWTHRESH	Receive Channel 3 Flow Control Threshold Register
130h	0x01E2 3130	RX4FLOWTHRESH	Receive Channel 4 Flow Control Threshold Register
134h	0x01E2 3134	RX5FLOWTHRESH	Receive Channel 5 Flow Control Threshold Register
138h	0x01E2 3138	RX6FLOWTHRESH	Receive Channel 6 Flow Control Threshold Register
13Ch	0x01E2 313C	RX7FLOWTHRESH	Receive Channel 7 Flow Control Threshold Register

Table 6-29. Ethernet Media Access Controller (EMAC) Registers (continued)

Offset	BYTE ADDRESS	REGISTER	Register Description
140h	0x01E2 3140	RX0FREEBUFFER	Receive Channel 0 Free Buffer Count Register
144h	0x01E2 3144	RX1FREEBUFFER	Receive Channel 1 Free Buffer Count Register
148h	0x01E2 3148	RX2FREEBUFFER	Receive Channel 2 Free Buffer Count Register
14Ch	0x01E2 314C	RX3FREEBUFFER	Receive Channel 3 Free Buffer Count Register
150h	0x01E2 3150	RX4FREEBUFFER	Receive Channel 4 Free Buffer Count Register
154h	0x01E2 3154	RX5FREEBUFFER	Receive Channel 5 Free Buffer Count Register
158h	0x01E2 3158	RX6FREEBUFFER	Receive Channel 6 Free Buffer Count Register
15Ch	0x01E2 315C	RX7FREEBUFFER	Receive Channel 7 Free Buffer Count Register
160h	0x01E2 3160	MACCONTROL	MAC Control Register
164h	0x01E2 3164	MACSTATUS	MAC Status Register
168h	0x01E2 3168	EMCONTROL	Emulation Control Register
16Ch	0x01E2 316C	FIFOCONTROL	FIFO Control Register
170h	0x01E2 3170	MACCONFIG	MAC Configuration Register
174h	0x01E2 3174	SOFTRESET	Soft Reset Register
1D0h	0x01E2 31D0	MACSRCADDRLO	MAC Source Address Low Bytes Register
1D4h	0x01E2 31D4	MACSRCADDRHI	MAC Source Address High Bytes Register
1D8h	0x01E2 31D8	MACHASH1	MAC Hash Address Register 1
1DCh	0x01E2 31DC	MACHASH2	MAC Hash Address Register 2
1E0h	0x01E2 31E0	BOFFTEST	Back Off Test Register
1E4h	0x01E2 31E4	TPACETEST	Transmit Pacing Algorithm Test Register
1E8h	0x01E2 31E8	RXPAUSE	Receive Pause Timer Register
1ECh	0x01E2 31EC	TXPAUSE	Transmit Pause Timer Register
	0x01E2 3200 - 0x01E2 32FC	(see Table 6-30)	EMAC Statistics Registers
500h	0x01E2 3500	MACADDRLO	MAC Address Low Bytes Register, Used in Receive Address Matching
504h	0x01E2 3504	MACADDRHI	MAC Address High Bytes Register, Used in Receive Address Matching
508h	0x01E2 3508	MACINDEX	MAC Index Register
600h	0x01E2 3600	TX0HDP	Transmit Channel 0 DMA Head Descriptor Pointer Register
604h	0x01E2 3604	TX1HDP	Transmit Channel 1 DMA Head Descriptor Pointer Register
608h	0x01E2 3608	TX2HDP	Transmit Channel 2 DMA Head Descriptor Pointer Register
60Ch	0x01E2 360C	TX3HDP	Transmit Channel 3 DMA Head Descriptor Pointer Register
610h	0x01E2 3610	TX4HDP	Transmit Channel 4 DMA Head Descriptor Pointer Register
614h	0x01E2 3614	TX5HDP	Transmit Channel 5 DMA Head Descriptor Pointer Register
618h	0x01E2 3618	TX6HDP	Transmit Channel 6 DMA Head Descriptor Pointer Register
61Ch	0x01E2 361C	TX7HDP	Transmit Channel 7 DMA Head Descriptor Pointer Register
620h	0x01E2 3620	RX0HDP	Receive Channel 0 DMA Head Descriptor Pointer Register
624h	0x01E2 3624	RX1HDP	Receive Channel 1 DMA Head Descriptor Pointer Register
628h	0x01E2 3628	RX2HDP	Receive Channel 2 DMA Head Descriptor Pointer Register
62Ch	0x01E2 362C	RX3HDP	Receive Channel 3 DMA Head Descriptor Pointer Register
630h	0x01E2 3630	RX4HDP	Receive Channel 4 DMA Head Descriptor Pointer Register
634h	0x01E2 3634	RX5HDP	Receive Channel 5 DMA Head Descriptor Pointer Register
638h	0x01E2 3638	RX6HDP	Receive Channel 6 DMA Head Descriptor Pointer Register
63Ch	0x01E2 363C	RX7HDP	Receive Channel 7 DMA Head Descriptor Pointer Register
640h	0x01E2 3640	TX0CP	Transmit Channel 0 Completion Pointer Register
644h	0x01E2 3644	TX1CP	Transmit Channel 1 Completion Pointer Register
648h	0x01E2 3648	TX2CP	Transmit Channel 2 Completion Pointer Register
64Ch	0x01E2 364C	TX3CP	Transmit Channel 3 Completion Pointer Register

Table 6-29. Ethernet Media Access Controller (EMAC) Registers (continued)

Offset	BYTE ADDRESS	REGISTER	Register Description
650h	0x01E2 3650	TX4CP	Transmit Channel 4 Completion Pointer Register
654h	0x01E2 3654	TX5CP	Transmit Channel 5 Completion Pointer Register
658h	0x01E2 3658	TX6CP	Transmit Channel 6 Completion Pointer Register
65Ch	0x01E2 365C	TX7CP	Transmit Channel 7 Completion Pointer Register
660h	0x01E2 3660	RX0CP	Receive Channel 0 Completion Pointer Register
664h	0x01E2 3664	RX1CP	Receive Channel 1 Completion Pointer Register
668h	0x01E2 3668	RX2CP	Receive Channel 2 Completion Pointer Register
66Ch	0x01E2 366C	RX3CP	Receive Channel 3 Completion Pointer Register
670h	0x01E2 3670	RX4CP	Receive Channel 4 Completion Pointer Register
674h	0x01E2 3674	RX5CP	Receive Channel 5 Completion Pointer Register
678h	0x01E2 3678	RX6CP	Receive Channel 6 Completion Pointer Register
67Ch	0x01E2 367C	RX7CP	Receive Channel 7 Completion Pointer Register

Table 6-30. EMAC Statistics Registers

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
0x01E2 3200	RXGOODFRAMES	Good Receive Frames Register
0x01E2 3204	RXBCASTFRAMES	Broadcast Receive Frames Register (Total number of good broadcast frames received)
0x01E2 3208	RXMCASTFRAMES	Multicast Receive Frames Register (Total number of good multicast frames received)
0x01E2 320C	RXPAUSEFRAMES	Pause Receive Frames Register
0x01E2 3210	RXCRCERRORS	Receive CRC Errors Register (Total number of frames received with CRC errors)
0x01E2 3214	RXALIGNCODEERRORS	Receive Alignment/Code Errors Register (Total number of frames received with alignment/code errors)
0x01E2 3218	RXOVERSIZED	Receive Oversized Frames Register (Total number of oversized frames received)
0x01E2 321C	RXJABBER	Receive Jabber Frames Register (Total number of jabber frames received)
0x01E2 3220	RXUNDERSIZED	Receive Undersized Frames Register (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 (Total number of received bytes in good frames)
0x01E2 3234	TXGOODFRAMES	Good Transmit Frames Register (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

Table 6-30. EMAC Statistics Registers (continued)

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
0x01E2 3268	FRAME64	Transmit and Receive 64 Octet Frames Register
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 6-31. 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

Table 6-31. EMAC Control Module Registers (continued)

BYTE ADDRESS	Acronym	Register Description
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 6-32. EMAC Control Module RAM

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
0x01E2 0000 - 0x01E2 1FFF		EMAC Local Buffer Descriptor Memory

Table 6-33. RMII Timing Requirements

NO.	PARAMETER		MIN	TYP	MAX	UNIT
1	tc(REFCLK)	Cycle Time, REF_CLK		20		ns
2	tw(REFCLKH)	Pulse Width, REF_CLK High	7		13	ns
3	tw(REFCLKL)	Pulse Width, REF_CLK Low	7		13	ns
6	tsu(RXD-REFCLK)	Input Setup Time, RXD Valid before REF_CLK High	4			ns
7	th(REFCLK-RXD)	Input Hold Time, RXD Valid after REF_CLK High	2			ns
8	tsu(CRSDV-REFCLK)	Input Setup Time, CRSDV Valid before REF_CLK High	4			ns
9	th(REFCLK-CRSDV)	Input Hold Time, CRSDV Valid after REF_CLK High	2			ns
10	tsu(RXER-REFCLK)	Input Setup Time, RXER Valid before REF_CLK High	4			ns
11	th(REFCLKR-RXER)	Input Hold Time, RXER Valid after REF_CLK High	2			ns

Table 6-34. RMII Timing Requirements

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

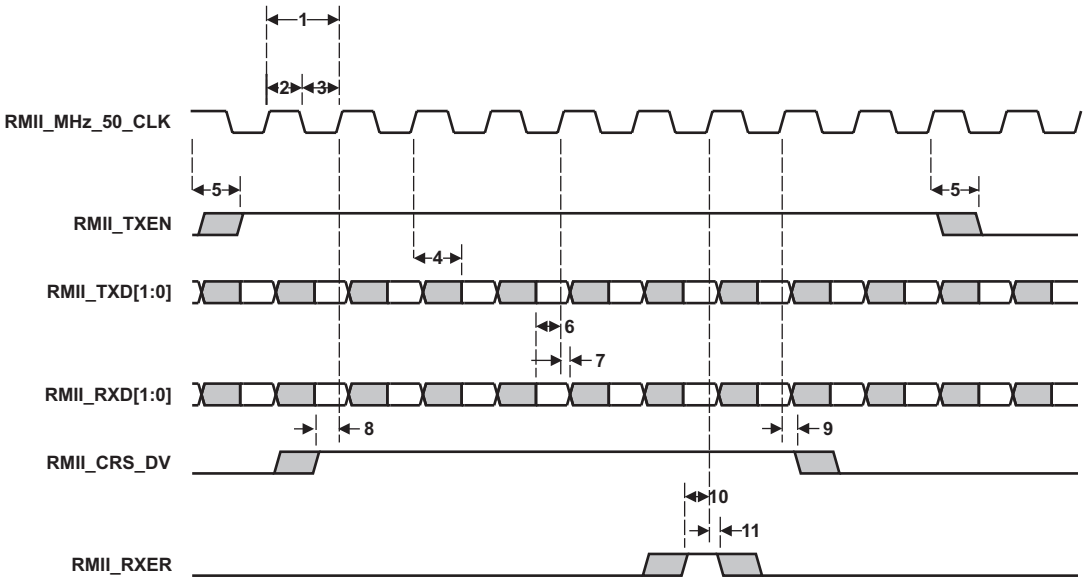


Figure 6-29. RMII Timing Diagram

6.14 Management Data Input/Output (MDIO)

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

The Management Data Input/Output (MDIO) module implements the 802.3 serial management interface to interrogate and control Ethernet PHY(s) using a shared two-wire bus. Host software uses the MDIO module to configure the auto-negotiation parameters of each PHY attached to the EMAC, retrieve the negotiation results, and configure required parameters in the EMAC module for correct operation. The module is designed to allow almost transparent operation of the MDIO interface, with very little maintenance from the core processor. Only one PHY may be connected at any given time.

For more detailed information on the MDIO peripheral, see the *OMAP-L137 Applications Processor DSP Peripherals Overview Reference Guide.* – Literature Number SPRUGA6 .

6.14.1 MDIO Registers

For a list of supported MDIO registers see Table 6-35 [MDIO Registers].

Table 6-35. MDIO Register Memory Map

HEX ADDRESS RANGE	ACRONYM	REGISTER NAME
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

6.14.2 Management Data Input/Output (MDIO) Electrical Data/Timing

Table 6-36. Timing Requirements for MDIO Input (see Figure 6-30 and Figure 6-31)

NO.			MIN	MAX	UNIT
1	$t_{c(MDCLK)}$	Cycle time, MDCLK	400		ns
2	$t_{w(MDCLK)}$	Pulse duration, MDCLK high/low	180		ns
3	$t_{t(MDCLK)}$	Transition time, MDCLK		5	ns
4	$t_{su(MDIO-MDCLKH)}$	Setup time, MDIO data input valid before MDCLK high	10		ns
5	$t_{h(MDCLKH-MDIO)}$	Hold time, MDIO data input valid after MDCLK high	10		ns

PRODUCT PREVIEW

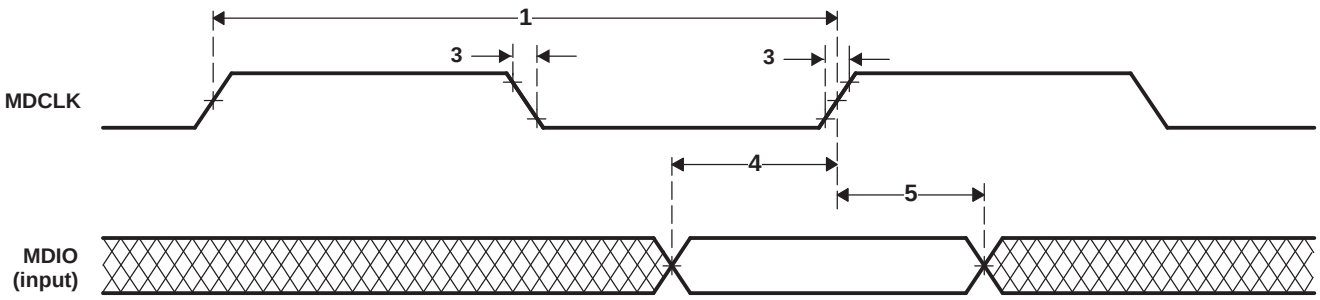


Figure 6-30. MDIO Input Timing

Table 6-37. Switching Characteristics Over Recommended Operating Conditions for MDIO Output (see Figure 6-31)

NO.			MIN	MAX	UNIT
7	$t_{d(MDCLKL-MDIO)}$	Delay time, MDCLK low to MDIO data output valid	0	100	ns

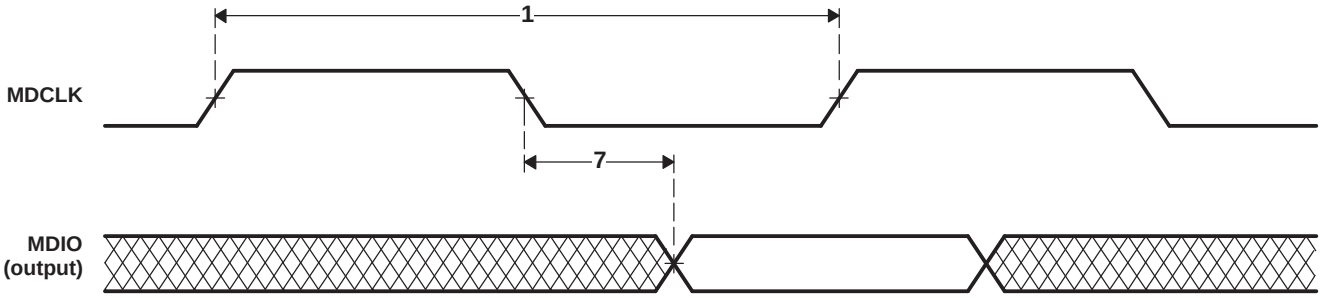


Figure 6-31. MDIO Output Timing

PRODUCT PREVIEW

6.15 Multichannel Audio Serial Ports (McASP0, McASP1, and McASP2)

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

Additionally, while the OMAP-L13x McASP modules are backward compatible with the McASP on previous devices; the OMAP-L13x McASP includes the following new features:

- 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
  - A one-shot adjustment (+/-1 Input Clock) feature has been added to enable simple input/output sample rate matching

The three McASPs on the OMAP-L137 are configured with the following options:

Table 6-38. OMAP-L137 McASP Configurations(1)

Module	Serializers	AFIFO	DIT	OMAP-L137 Pins
McASP0	16	64 Word RX 64 Word TX	N	AXR0[15:0], AHCLKR0, ACLKR0, AFSR0, AHCLKX0, ACLKX0, AFSX0, AMUTE0
McASP1	12	64 Word RX 64 Word TX	N	AXR1[11:10], AXR1[8:0], AHCLKR1, ACLKR1, AFSR1, AHCLKX1, ACLKX1, AFSX1, AMUTE1
McASP2	4	16 Word RX 16 Word TX	Y	AXR2[3:0], AHCLKR2, ACLKR2, AFSR2, AHCLKX2, ACLKX2, AFSX2, AMUTE2

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

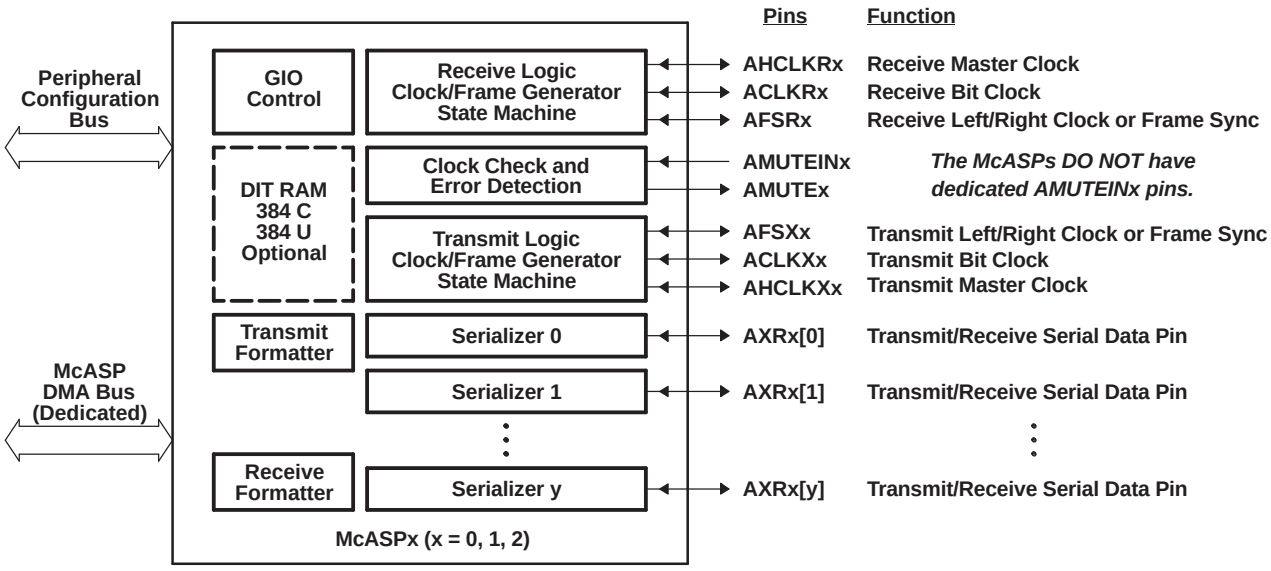


Figure 6-32. McASP Block Diagram

6.15.1 McASP Peripheral Registers Description(s)

Registers for the McASP are summarized in Table 6-39. 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 6-40

Registers for the McASP Audio FIFO (AFIFO) are summarized in Table 6-41. 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 6-39. McASP Registers Accessed Through Peripheral Configuration Port

Offset	McASP0 BYTE ADDRESS	McASP1 BYTE ADDRESS	McASP2 BYTE ADDRESS	Acronym	Register Description
0h	0x01D0 0000	0x01D0 4000	0x01D0 8000	REV	Revision identification register
10h	0x01D0 0010	0x01D0 4010	0x01D0 8010	PFUNC	Pin function register
14h	0x01D0 0014	0x01D0 4014	0x01D0 8014	PDIR	Pin direction register
18h	0x01D0 0018	0x01D0 4018	0x01D0 8018	PDOUT	Pin data output register
1Ch	0x01D0 001C	0x01D0 401C	0x01D0 801C	PDIN	Read returns: Pin data input register
1Ch	0x01D0 001C	0x01D0 401C	0x01D0 801C	PDSET	Writes affect: Pin data set register (alternate write address: PDOUT)
20h	0x01D0 0020	0x01D0 4020	0x01D0 8020	PDCLR	Pin data clear register (alternate write address: PDOUT)
44h	0x01D0 0044	0x01D0 4044	0x01D0 8044	GBLCTL	Global control register
48h	0x01D0 0048	0x01D0 4048	0x01D0 8048	AMUTE	Audio mute control register
4Ch	0x01D0 004C	0x01D0 404C	0x01D0 804C	DLBCTL	Digital loopback control register
50h	0x01D0 0050	0x01D0 4050	0x01D0 8050	DITCTL	DIT mode control register
60h	0x01D0 0060	0x01D0 4060	0x01D0 8060	RGBLCTL	Receiver global control register: Alias of GBLCTL, only receive bits are affected - allows receiver to be reset independently from transmitter
64h	0x01D0 0064	0x01D0 4064	0x01D0 8064	RMASK	Receive format unit bit mask register
68h	0x01D0 0068	0x01D0 4068	0x01D0 8068	RFMT	Receive bit stream format register
6Ch	0x01D0 006C	0x01D0 406C	0x01D0 806C	AFSRCTL	Receive frame sync control register
70h	0x01D0 0070	0x01D0 4070	0x01D0 8070	ACLKRCTL	Receive clock control register
74h	0x01D0 0074	0x01D0 4074	0x01D0 8074	AHCLKRCTL	Receive high-frequency clock control register

Table 6-39. McASP Registers Accessed Through Peripheral Configuration Port (continued)

Offset	McASP0 BYTE ADDRESS	McASP1 BYTE ADDRESS	McASP2 BYTE ADDRESS	Acronym	Register Description
78h	0x01D0 0078	0x01D0 4078	0x01D0 8078	RTDM	Receive TDM time slot 0-31 register
7Ch	0x01D0 007C	0x01D0 407C	0x01D0 807C	RINTCTL	Receiver interrupt control register
80h	0x01D0 0080	0x01D0 4080	0x01D0 8080	RSTAT	Receiver status register
84h	0x01D0 0084	0x01D0 4084	0x01D0 8084	RSLOT	Current receive TDM time slot register
88h	0x01D0 0088	0x01D0 4088	0x01D0 8088	RCLKCHK	Receive clock check control register
8Ch	0x01D0 008C	0x01D0 408C	0x01D0 808C	REVTCTL	Receiver DMA event control register
ACh	0x01D0 00A0	0x01D0 40A0	0x01D0 80A0	XGBLCTL	Transmitter global control register. Alias of GBLCTL, only transmit bits are affected - allows transmitter to be reset independently from receiver
A4h	0x01D0 00A4	0x01D0 40A4	0x01D0 80A4	XMASK	Transmit format unit bit mask register
A8h	0x01D0 00A8	0x01D0 40A8	0x01D0 80A8	XFMT	Transmit bit stream format register
ACh	0x01D0 00AC	0x01D0 40AC	0x01D0 80AC	AFSXCTL	Transmit frame sync control register
B0h	0x01D0 00B0	0x01D0 40B0	0x01D0 80B0	ACLKXCTL	Transmit clock control register
B4h	0x01D0 00B4	0x01D0 40B4	0x01D0 80B4	AHCLKXCTL	Transmit high-frequency clock control register
B8h	0x01D0 00B8	0x01D0 40B8	0x01D0 80B8	XTDM	Transmit TDM time slot 0-31 register
BCh	0x01D0 00BC	0x01D0 40BC	0x01D0 80BC	XINTCTL	Transmitter interrupt control register
C0h	0x01D0 00C0	0x01D0 40C0	0x01D0 80C0	XSTAT	Transmitter status register
C4h	0x01D0 00C4	0x01D0 40C4	0x01D0 80C4	XSLOT	Current transmit TDM time slot register
C8h	0x01D0 00C8	0x01D0 40C8	0x01D0 80C8	XCLKCHK	Transmit clock check control register
CCh	0x01D0 00CC	0x01D0 40CC	0x01D0 80CC	XEVTCTL	Transmitter DMA event control register
100h	0x01D0 0100	0x01D0 4100	0x01D0 8100	DITCSRA0	Left (even TDM time slot) channel status register (DIT mode) 0
104h	0x01D0 0104	0x01D0 4104	0x01D0 8104	DITCSRA1	Left (even TDM time slot) channel status register (DIT mode) 1
108h	0x01D0 0108	0x01D0 4108	0x01D0 8108	DITCSRA2	Left (even TDM time slot) channel status register (DIT mode) 2
10Ch	0x01D0 010C	0x01D0 410C	0x01D0 810C	DITCSRA3	Left (even TDM time slot) channel status register (DIT mode) 3
110h	0x01D0 0110	0x01D0 4110	0x01D0 8110	DITCSRA4	Left (even TDM time slot) channel status register (DIT mode) 4
114h	0x01D0 0114	0x01D0 4114	0x01D0 8114	DITCSRA5	Left (even TDM time slot) channel status register (DIT mode) 5
118h	0x01D0 0118	0x01D0 4118	0x01D0 8118	DITCSRB0	Right (odd TDM time slot) channel status register (DIT mode) 0
11Ch	0x01D0 011C	0x01D0 411C	0x01D0 811C	DITCSRB1	Right (odd TDM time slot) channel status register (DIT mode) 1
120h	0x01D0 0120	0x01D0 4120	0x01D0 8120	DITCSRB2	Right (odd TDM time slot) channel status register (DIT mode) 2
124h	0x01D0 0124	0x01D0 4124	0x01D0 8124	DITCSRB3	Right (odd TDM time slot) channel status register (DIT mode) 3
128h	0x01D0 0128	0x01D0 4128	0x01D0 8128	DITCSRB4	Right (odd TDM time slot) channel status register (DIT mode) 4
12Ch	0x01D0 012C	0x01D0 412C	0x01D0 812C	DITCSRB5	Right (odd TDM time slot) channel status register (DIT mode) 5
130h	0x01D0 0130	0x01D0 4130	0x01D0 8130	DITUDRA0	Left (even TDM time slot) channel user data register (DIT mode) 0
134h	0x01D0 0134	0x01D0 4134	0x01D0 8134	DITUDRA1	Left (even TDM time slot) channel user data register (DIT mode) 1
138h	0x01D0 0138	0x01D0 4138	0x01D0 8138	DITUDRA2	Left (even TDM time slot) channel user data register (DIT mode) 2

Table 6-39. McASP Registers Accessed Through Peripheral Configuration Port (continued)

Offset	McASP0 BYTE ADDRESS	McASP1 BYTE ADDRESS	McASP2 BYTE ADDRESS	Acronym	Register Description
13Ch	0x01D0 013C	0x01D0 413C	0x01D0 813C	DITUDRA3	Left (even TDM time slot) channel user data register (DIT mode) 3
140h	0x01D0 0140	0x01D0 4140	0x01D0 8140	DITUDRA4	Left (even TDM time slot) channel user data register (DIT mode) 4
144h	0x01D0 0144	0x01D0 4144	0x01D0 8144	DITUDRA5	Left (even TDM time slot) channel user data register (DIT mode) 5
148h	0x01D0 0148	0x01D0 4148	0x01D0 8148	DITUDRB0	Right (odd TDM time slot) channel user data register (DIT mode) 0
14Ch	0x01D0 014C	0x01D0 414C	0x01D0 814C	DITUDRB1	Right (odd TDM time slot) channel user data register (DIT mode) 1
150h	0x01D0 0150	0x01D0 4150	0x01D0 8150	DITUDRB2	Right (odd TDM time slot) channel user data register (DIT mode) 2
154h	0x01D0 0154	0x01D0 4154	0x01D0 8154	DITUDRB3	Right (odd TDM time slot) channel user data register (DIT mode) 3
158h	0x01D0 0158	0x01D0 4158	0x01D0 8158	DITUDRB4	Right (odd TDM time slot) channel user data register (DIT mode) 4
15Ch	0x01D0 015C	0x01D0 415C	0x01D0 815C	DITUDRB5	Right (odd TDM time slot) channel user data register (DIT mode) 5
180h	0x01D0 0180	0x01D0 4180	0x01D0 8180	SRCTL0	Serializer control register 0
184h	0x01D0 0184	0x01D0 4184	0x01D0 8184	SRCTL1	Serializer control register 1
188h	0x01D0 0188	0x01D0 4188	0x01D0 8188	SRCTL2	Serializer control register 2
18Ch	0x01D0 018C	0x01D0 418C	0x01D0 818C	SRCTL3	Serializer control register 3
190h	0x01D0 0190	0x01D0 4190	0x01D0 8190	SRCTL4	Serializer control register 4
194h	0x01D0 0194	0x01D0 4194	0x01D0 8194	SRCTL5	Serializer control register 5
198h	0x01D0 0198	0x01D0 4198	0x01D0 8198	SRCTL6	Serializer control register 6
19Ch	0x01D0 019C	0x01D0 419C	0x01D0 819C	SRCTL7	Serializer control register 7
1A0h	0x01D0 01A0	0x01D0 41A0	0x01D0 81A0	SRCTL8	Serializer control register 8
1A4h	0x01D0 01A4	0x01D0 41A4	0x01D0 81A4	SRCTL9	Serializer control register 9
1A8h	0x01D0 01A8	0x01D0 41A8	0x01D0 81A8	SRCTL10	Serializer control register 10
1ACh	0x01D0 01AC	0x01D0 41AC	0x01D0 81AC	SRCTL11	Serializer control register 11
1B0h	0x01D0 01B0	0x01D0 41B0	0x01D0 81B0	SRCTL12	Serializer control register 12
1B4h	0x01D0 01B4	0x01D0 41B4	0x01D0 81B4	SRCTL13	Serializer control register 13
1B8h	0x01D0 01B8	0x01D0 41B8	0x01D0 81B8	SRCTL14	Serializer control register 14
1BCh	0x01D0 01BC	0x01D0 41BC	0x01D0 81BC	SRCTL15	Serializer control register 15
200h	0x01D0 0200	0x01D0 4200	0x01D0 8200	XBUF0(1)	Transmit buffer register for serializer 0
204h	0x01D0 0204	0x01D0 4204	0x01D0 8204	XBUF1(1)	Transmit buffer register for serializer 1
208h	0x01D0 0208	0x01D0 4208	0x01D0 8208	XBUF2(1)	Transmit buffer register for serializer 2
20Ch	0x01D0 020C	0x01D0 420C	0x01D0 820C	XBUF3(1)	Transmit buffer register for serializer 3
210h	0x01D0 0210	0x01D0 4210	0x01D0 8210	XBUF4(1)	Transmit buffer register for serializer 4
214h	0x01D0 0214	0x01D0 4214	0x01D0 8214	XBUF5(1)	Transmit buffer register for serializer 5
218h	0x01D0 0218	0x01D0 4218	0x01D0 8218	XBUF6(1)	Transmit buffer register for serializer 6
21Ch	0x01D0 021C	0x01D0 421C	0x01D0 821C	XBUF7(1)	Transmit buffer register for serializer 7
220h	0x01D0 0220	0x01D0 4220	0x01D0 8220	XBUF8(1)	Transmit buffer register for serializer 8
224h	0x01D0 0224	0x01D0 4224	0x01D0 8224	XBUF9(1)	Transmit buffer register for serializer 9
228h	0x01D0 0228	0x01D0 4228	0x01D0 8228	XBUF10(1)	Transmit buffer register for serializer 10
22Ch	0x01D0 022C	0x01D0 422C	0x01D0 822C	XBUF11(1)	Transmit buffer register for serializer 11
230h	0x01D0 0230	0x01D0 4230	0x01D0 8230	XBUF12(1)	Transmit buffer register for serializer 12
234h	0x01D0 0234	0x01D0 4234	0x01D0 8234	XBUF13(1)	Transmit buffer register for serializer 13

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

Table 6-39. McASP Registers Accessed Through Peripheral Configuration Port (continued)

Offset	McASP0 BYTE ADDRESS	McASP1 BYTE ADDRESS	McASP2 BYTE ADDRESS	Acronym	Register Description
238h	0x01D0 0238	0x01D0 4238	0x01D0 8238	XBUF14(1)	Transmit buffer register for serializer 14
23Ch	0x01D0 023C	0x01D0 423C	0x01D0 823C	XBUF15(1)	Transmit buffer register for serializer 15
280h	0x01D0 0280	0x01D0 4280	0x01D0 8280	RBUF0(2)	Receive buffer register for serializer 0
284h	0x01D0 0284	0x01D0 4284	0x01D0 8284	RBUF1(2)	Receive buffer register for serializer 1
288h	0x01D0 0288	0x01D0 4288	0x01D0 8288	RBUF2(2)	Receive buffer register for serializer 2
28Ch	0x01D0 028C	0x01D0 428C	0x01D0 828C	RBUF3(2)	Receive buffer register for serializer 3
290h	0x01D0 0290	0x01D0 4290	0x01D0 8290	RBUF4(2)	Receive buffer register for serializer 4
294h	0x01D0 0294	0x01D0 4294	0x01D0 8294	RBUF5(2)	Receive buffer register for serializer 5
298h	0x01D0 0298	0x01D0 4298	0x01D0 8298	RBUF6(2)	Receive buffer register for serializer 6
29Ch	0x01D0 029C	0x01D0 429C	0x01D0 829C	RBUF7(2)	Receive buffer register for serializer 7
2A0h	0x01D0 02A0	0x01D0 42A0	0x01D0 82A0	RBUF8(2)	Receive buffer register for serializer 8
2A4h	0x01D0 02A4	0x01D0 42A4	0x01D0 82A4	RBUF9(2)	Receive buffer register for serializer 9
2A8h	0x01D0 02A8	0x01D0 42A8	0x01D0 82A8	RBUF10(2)	Receive buffer register for serializer 10
2ACh	0x01D0 02AC	0x01D0 42AC	0x01D0 82AC	RBUF11(2)	Receive buffer register for serializer 11
2B0h	0x01D0 02B0	0x01D0 42B0	0x01D0 82B0	RBUF12(2)	Receive buffer register for serializer 12
2B4h	0x01D0 02B4	0x01D0 42B4	0x01D0 82B4	RBUF13(2)	Receive buffer register for serializer 13
2B8h	0x01D0 02B8	0x01D0 42B8	0x01D0 82BB	RBUF14(2)	Receive buffer register for serializer 14
2BCh	0x01D0 02BC	0x01D0 42BC	0x01D0 82BC	RBUF15(2)	Receive buffer register for serializer 15

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

Table 6-40. McASP Registers Accessed Through DMA Port

Hex Address	Register Name	Register Description
Read Accesses	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 XBUSEL = 0 in XFMT.
Write Accesses	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 RBUSEL = 0 in RFMT.

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

McASP0 BYTE ADDRESS	McASP0 BYTE ADDRESS	McASP0 BYTE ADDRESS	Acronym	Register Description
0x01D0 1000	0x01D0 5000	0x01D0 9000	AFIFOREV	AFIFO revision identification register
0x01D0 1010	0x01D0 5010	0x01D0 9010	WFIFOCTL	Write FIFO control register
0x01D0 1014	0x01D0 5014	0x01D0 9014	WFIFOSTS	Write FIFO status register
0x01D0 1018	0x01D0 5018	0x01D0 9018	RFIFOCTL	Read FIFO control register
0x01D0 101C	0x01D0 501C	0x01D0 901C	RFIFOSTS	Read FIFO status register

6.15.2 McASP Electrical Data/Timing

6.15.2.1 Multichannel Audio Serial Port 0 (McASP0) Timing

Table 6-42 and Table 6-43 assume testing over recommended operating conditions (see Figure 6-33 and Figure 6-34).

Table 6-42. McASP0 Timing Requirements(1)(2)

NO.			MIN	MAX	UNIT
1	$t_{c(AHCLKRX)}$	Cycle time, AHCLKR0 external, AHCLKR0 input	20		ns
		Cycle time, AHCLKX0 external, AHCLKX0 input	20		
2	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR0 external, AHCLKR0 input	10		ns
		Pulse duration, AHCLKX0 external, AHCLKX0 input	10		
3	$t_{c(ACLKRX)}$	Cycle time, ACLKR0 external, ACLKR0 input	greater of 2P or 20		ns
		Cycle time, ACLKX0 external, ACLKX0 input	greater of 2P or 20		
4	$t_{w(ACLKRX)}$	Pulse duration, ACLKR0 external, ACLKR0 input	10		ns
		Pulse duration, ACLKX0 external, ACLKX0 input	10		
5	$t_{su(AFSRX-ACLKRX)}$	Setup time, AFSR0 input to ACLKR0 internal(3)	9.4		ns
		Setup time, AFSR0 input to ACLKX0 internal(4)	9.4		
		Setup time, AFSX0 input to ACLKX0 internal	9.4		
		Setup time, AFSR0 input to ACLKR0 external input(3)	2.9		
		Setup time, AFSR0 input to ACLKX0 external input(4)	2.9		
		Setup time, AFSX0 input to ACLKX0 external input	2.9		
		Setup time, AFSR0 input to ACLKR0 external output(3)	2.9		
		Setup time, AFSR0 input to ACLKX0 external output(4)	2.9		
		Setup time, AFSX0 input to ACLKX0 external output	2.9		
6	$t_{h(ACLKRX-AFSRX)}$	Hold time, AFSR0 input after ACLKR0 internal(3)	-2.1		ns
		Hold time, AFSR0 input after ACLKX0 internal(4)	-2.1		
		Hold time, AFSX0 input after ACLKX0 internal	-2.1		
		Hold time, AFSR0 input after ACLKR0 external input(3)	0.4		
		Hold time, AFSR0 input after ACLKX0 external input(4)	0.4		
		Hold time, AFSX0 input after ACLKX0 external input	0.4		
		Hold time, AFSR0 input after ACLKR0 external output(3)	0.4		
		Hold time, AFSR0 input after ACLKX0 external output(4)	0.4		
		Hold time, AFSX0 input after ACLKX0 external output	0.4		
7	$t_{su(AXR-ACLKRX)}$	Setup time, AXR0[n] input to ACLKR0 internal(3)	9.4		ns
		Setup time, AXR0[n] input to ACLKX0 internal(4)	9.4		
		Setup time, AXR0[n] input to ACLKR0 external input(3)	2.9		
		Setup time, AXR0[n] input to ACLKX0 external input(4)	2.9		
		Setup time, AXR0[n] input to ACLKR0 external output(3)	2.9		
		Setup time, AXR0[n] input to ACLKX0 external output(4)	2.9		

(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

PRODUCT PREVIEW

Table 6-42. McASP0 Timing Requirements (continued)

NO.			MIN	MAX	UNIT
8	$t_{h(ACLKRX-AXR)}$	Hold time, AXR0[n] input after ACLKR0 internal(3)	-2.1		ns
		Hold time, AXR0[n] input after ACLKX0 internal(4)	-2.1		
		Hold time, AXR0[n] input after ACLKR0 external input(3)	0.4		
		Hold time, AXR0[n] input after ACLKX0 external input(4)	0.4		
		Hold time, AXR0[n] input after ACLKR0 external output(3)	0.4		
		Hold time, AXR0[n] input after ACLKX0 external output(4)	0.4		

PRODUCT PREVIEW

Table 6-43. McASP0 Switching Characteristics(1)

NO.	PARAMETER		MIN	MAX	UNIT
9	$t_{c(AHCLKRX)}$	Cycle time, AHCLKX0 internal, AHCLKR0 output	20		ns
		Cycle time, AHCLKR0 external, AHCLKR0 output	20		
		Cycle time, AHCLKX0 internal, AHCLKX0 output	20		
		Cycle time, AHCLKX0 external, AHCLKX0 output	20		
10	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR0 internal, AHCLKR0 output	(AHR/2) – 2.5(2)		ns
		Pulse duration, AHCLKR0 external, AHCLKR0 output	(AHR/2) – 2.5(2)		
		Pulse duration, AHCLKX0 internal, AHCLKX0 output	(AHX/2) – 2.5(3)		
		Pulse duration, AHCLKX0 external, AHCLKX0 output	(AHX/2) – 2.5(3)		
11	$t_{c(ACLKRX)}$	Cycle time, ACLKR0 internal, ACLKR0 output	greater of 2P or 20 ns(4)		ns
		Cycle time, ACLKR0 external, ACLKR0 output	greater of 2P or 20 ns(4)		
		Cycle time, ACLKX0 internal, ACLKX0 output	greater of 2P or 20 ns(4)		
		Cycle time, ACLKX0 external, ACLKX0 output	greater of 2P or 20 ns(4)		
12	$t_{w(ACLKRX)}$	Pulse duration, ACLKR0 internal, ACLKR0 output	(AR/2) – 2.5(5)		ns
		Pulse duration, ACLKR0 external, ACLKR0 output	(AR/2) – 2.5(5)		
		Pulse duration, ACLKX0 internal, ACLKX0 output	(AX/2) – 2.5(6)		
		Pulse duration, ACLKX0 external, ACLKX0 output	(AX/2) – 2.5(6)		
13	$t_{d(ACLKRX-AFSRX)}$	Delay time, ACLKR0 internal, AFSR output(7)	0	5.8	ns
		Delay time, ACLKX0 internal, AFSR output(8)	0	5.8	
		Delay time, ACLKX0 internal, AFSX output	0	5.8	
		Delay time, ACLKR0 external input, AFSR output(7)	3	11.6	
		Delay time, ACLKX0 external input, AFSR output(8)	3	11.6	
		Delay time, ACLKX0 external input, AFSX output	3	11.6	
		Delay time, ACLKR0 external output, AFSR output(7)	3	11.6	
		Delay time, ACLKX0 external output, AFSR output(8)	3	11.6	
		Delay time, ACLKX0 external output, AFSX output	3	11.6	
14	$t_{d(ACLKX-AXRV)}$	Delay time, ACLKX0 internal, AXR0[n] output	0	5.8	ns
		Delay time, ACLKX0 external input, AXR0[n] output	3	11.6	
		Delay time, ACLKX0 external output, AXR0[n] output	3	11.6	
15	$t_{dis(ACLKX-AXRHZ)}$	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 – McASP0ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1
ACLKR0 internal – McASP0 ACLKR0CTL.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) 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
(8) McASP0 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX0

PRODUCT PREVIEW

6.15.2.2 Multichannel Audio Serial Port 1 (McASP1) Timing

Table 6-44 and Table 6-45 assume testing over recommended operating conditions (see Figure 6-33 and Figure 6-34).

Table 6-44. McASP1 Timing Requirements(1)(2)

NO.			MIN	MAX	UNIT
1	$t_{c(AHCLKRX)}$	Cycle time, AHCLKR1 external, AHCLKR1 input	20		ns
		Cycle time, AHCLKX1 external, AHCLKX1 input	20		
2	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR1 external, AHCLKR1 input	10		ns
		Pulse duration, AHCLKX1 external, AHCLKX1 input	10		
3	$t_{c(ACLKRX)}$	Cycle time, ACLKR1 external, ACLKR1 input	greater of 2P or 20		ns
		Cycle time, ACLKX1 external, ACLKX1 input	greater of 2P or 20		
4	$t_{w(ACLKRX)}$	Pulse duration, ACLKR1 external, ACLKR1 input	10		ns
		Pulse duration, ACLKX1 external, ACLKX1 input	10		
5	$t_{su(AFSRX-ACLKRX)}$	Setup time, AFSR1 input to ACLKR1 internal(3)	10.4		ns
		Setup time, AFSR1 input to ACLKX1 internal(4)	10.4		
		Setup time, AFSX1 input to ACLKX1 internal	10.4		
		Setup time, AFSR1 input to ACLKR1 external input(3)	2.6		
		Setup time, AFSR1 input to ACLKX1 external input(4)	2.6		
		Setup time, AFSX1 input to ACLKX1 external input	2.6		
		Setup time, AFSR1 input to ACLKR1 external output(3)	2.6		
		Setup time, AFSR1 input to ACLKX1 external output(4)	2.6		
		Setup time, AFSX1 input to ACLKX1 external output	2.6		
6	$t_{h(ACLKRX-AFSRX)}$	Hold time, AFSR1 input after ACLKR1 internal(3)	-2.6		ns
		Hold time, AFSR1 input after ACLKX1 internal(4)	-2.6		
		Hold time, AFSX1 input after ACLKX1 internal	-2.6		
		Hold time, AFSR1 input after ACLKR1 external input(3)	0.3		
		Hold time, AFSR1 input after ACLKX1 external input(4)	0.3		
		Hold time, AFSX1 input after ACLKX1 external input	0.3		
		Hold time, AFSR1 input after ACLKR1 external output(3)	0.3		
		Hold time, AFSR1 input after ACLKX1 external output(4)	0.3		
		Hold time, AFSX1 input after ACLKX1 external output	0.3		
7	$t_{su(AXR-ACLKRX)}$	Setup time, AXR1[n] input to ACLKR1 internal(3)	10.4		ns
		Setup time, AXR1[n] input to ACLKX1 internal(4)	10.4		
		Setup time, AXR1[n] input to ACLKR1 external input(3)	2.6		
		Setup time, AXR1[n] input to ACLKX1 external input(4)	2.6		
		Setup time, AXR1[n] input to ACLKR1 external output(3)	2.6		
		Setup time, AXR1[n] input to ACLKX1 external output(4)	2.6		

(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 6-44. McASP1 Timing Requirements (continued)

NO.			MIN	MAX	UNIT
8	$t_{h(ACLKRX-AXR)}$	Hold time, AXR1[n] input after ACLKR1 internal[3]	-2.6		ns
		Hold time, AXR1[n] input after ACLKX1 internal[4]	-2.6		
		Hold time, AXR1[n] input after ACLKR1 external input[3]	0.3		
		Hold time, AXR1[n] input after ACLKX1 external input[4]	0.3		
		Hold time, AXR1[n] input after ACLKR1 external output[3]	0.3		
		Hold time, AXR1[n] input after ACLKX1 external output[4]	0.3		

Table 6-45. McASP1 Switching Characteristics[1]

NO.	PARAMETER		MIN	MAX	UNIT
9	$t_{c(AHCLKRX)}$	Cycle time, AHCLKX1 internal, AHCLKR1 output	20		ns
		Cycle time, AHCLKR1 external, AHCLKR1 output	20		
		Cycle time, AHCLKX1 internal, AHCLKX1 output	20		
		Cycle time, AHCLKX1 external, AHCLKX1 output	20		
10	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR1 internal, AHCLKR1 output	(AHR/2) – 2.5[2]		ns
		Pulse duration, AHCLKR1 external, AHCLKR1 output	(AHR/2) – 2.5[2]		
		Pulse duration, AHCLKX1 internal, AHCLKX1 output	(AHX/2) – 2.5[3]		
		Pulse duration, AHCLKX1 external, AHCLKX1 output	(AHX/2) – 2.5[3]		
11	$t_{c(ACLKRX)}$	Cycle time, ACLKR1 internal, ACLKR1 output	greater of 2P or 20 ns[4]		ns
		Cycle time, ACLKR1 external, ACLKR1 output	greater of 2P or 20 ns[4]		
		Cycle time, ACLKX1 internal, ACLKX1 output	greater of 2P or 20 ns[4]		
		Cycle time, ACLKX1 external, ACLKX1 output	greater of 2P or 20 ns[4]		
12	$t_{w(ACLKRX)}$	Pulse duration, ACLKR1 internal, ACLKR1 output	(AR/2) – 2.5[5]		ns
		Pulse duration, ACLKR1 external, ACLKR1 output	(AR/2) – 2.5[5]		
		Pulse duration, ACLKX1 internal, ACLKX1 output	(AX/2) – 2.5[6]		
		Pulse duration, ACLKX1 external, ACLKX1 output	(AX/2) – 2.5[6]		
13	$t_{d(ACLKRX-AFSRX)}$	Delay time, ACLKR1 internal, AFSR output[7]	0.5	6.7	ns
		Delay time, ACLKX1 internal, AFSR output[8]	0.5	6.7	
		Delay time, ACLKX1 internal, AFSX output	0.5	6.7	
		Delay time, ACLKR1 external input, AFSR output[7]	3.9	13.8	
		Delay time, ACLKX1 external input, AFSR output[8]	3.9	13.8	
		Delay time, ACLKX1 external input, AFSX output	3.9	13.8	
		Delay time, ACLKR1 external output, AFSR output[7]	3.9	13.8	
		Delay time, ACLKX1 external output, AFSR output[8]	3.9	13.8	
		Delay time, ACLKX1 external output, AFSX output	3.9	13.8	
14	$t_{d(ACLKX-AXRV)}$	Delay time, ACLKX1 internal, AXR1[n] output	0.5	6.7	ns
		Delay time, ACLKX1 external input, AXR1[n] output	3.9	13.8	
		Delay time, ACLKX1 external output, AXR1[n] output	3.9	13.8	
15	$t_{dis(ACLKX-AXRHZ)}$	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
(8) McASP1 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX1

6.15.2.3 Multichannel Audio Serial Port 2 (McASP2) Timing

Table 6-46 and Table 6-47 assume testing over recommended operating conditions (see Figure 6-33 and Figure 6-34).

Table 6-46. McASP2 Timing Requirements(1)(2)

NO.			MIN	MAX	UNIT
1	$t_{c(AHCLKRX)}$	Cycle time, AHCLKR2 external, AHCLKR2 input	13		ns
		Cycle time, AHCLKX2 external, AHCLKX2 input	13		
2	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR2 external, AHCLKR2 input	6.5		ns
		Pulse duration, AHCLKX2 external, AHCLKX2 input	6.5		
3	$t_{c(ACLKRX)}$	Cycle time, ACLKR2 external, ACLKR2 input	greater of 2P or 13		ns
		Cycle time, ACLKX2 external, ACLKX2 input	greater of 2P or 13		
4	$t_{w(ACLKRX)}$	Pulse duration, ACLKR2 external, ACLKR2 input	6.5		ns
		Pulse duration, ACLKX2 external, ACLKX2 input	6.5		
5	$t_{su(AFSRX-ACLKRX)}$	Setup time, AFSR2 input to ACLKR2 internal(3)	10		ns
		Setup time, AFSR2 input to ACLKX2 internal(4)	10		
		Setup time, AFSX2 input to ACLKX2 internal	10		
		Setup time, AFSR2 input to ACLKR2 external input(3)	1.6		
		Setup time, AFSR2 input to ACLKX2 external input(4)	1.6		
		Setup time, AFSX2 input to ACLKX2 external input	1.6		
		Setup time, AFSR2 input to ACLKR2 external output(3)	1.6		
		Setup time, AFSR2 input to ACLKX2 external output(4)	1.6		
		Setup time, AFSX2 input to ACLKX2 external output	1.6		
6	$t_{h(ACLKRX-AFSRX)}$	Hold time, AFSR2 input after ACLKR2 internal(3)	-2.2		ns
		Hold time, AFSR2 input after ACLKX2 internal(4)	-2.2		
		Hold time, AFSX2 input after ACLKX2 internal	-2.2		
		Hold time, AFSR2 input after ACLKR2 external input(3)	1.3		
		Hold time, AFSR2 input after ACLKX2 external input(4)	1.3		
		Hold time, AFSX2 input after ACLKX2 external input	1.3		
		Hold time, AFSR2 input after ACLKR2 external output(3)	1.3		
		Hold time, AFSR2 input after ACLKX2 external output(4)	1.3		
		Hold time, AFSX2 input after ACLKX2 external output	1.3		
7	$t_{su(AXR-ACLKRX)}$	Setup time, AXR2[n] input to ACLKR2 internal(3)	10		ns
		Setup time, AXR2[n] input to ACLKX2 internal(4)	10		
		Setup time, AXR2[n] input to ACLKR2 external input(3)	1.6		
		Setup time, AXR2[n] input to ACLKX2 external input(4)	1.6		
		Setup time, AXR2[n] input to ACLKR2 external output(3)	1.6		
		Setup time, AXR2[n] input to ACLKX2 external output(4)	1.6		

(1) ACLKX2 internal – McASP2 ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1
ACLKX2 external input – McASP2 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0
ACLKX2 external output – McASP2 ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1
ACLKR2 internal – McASP2 ACLKRCTL.CLKRM = 1, PDIR.ACLKR =1
ACLKR2 external input – McASP2 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 0
ACLKR2 external output – McASP2 ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 1
(2) P = SYSCLK2 period
(3) McASP2 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR2
(4) McASP2 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX2

Table 6-46. McASP2 Timing Requirements (continued)

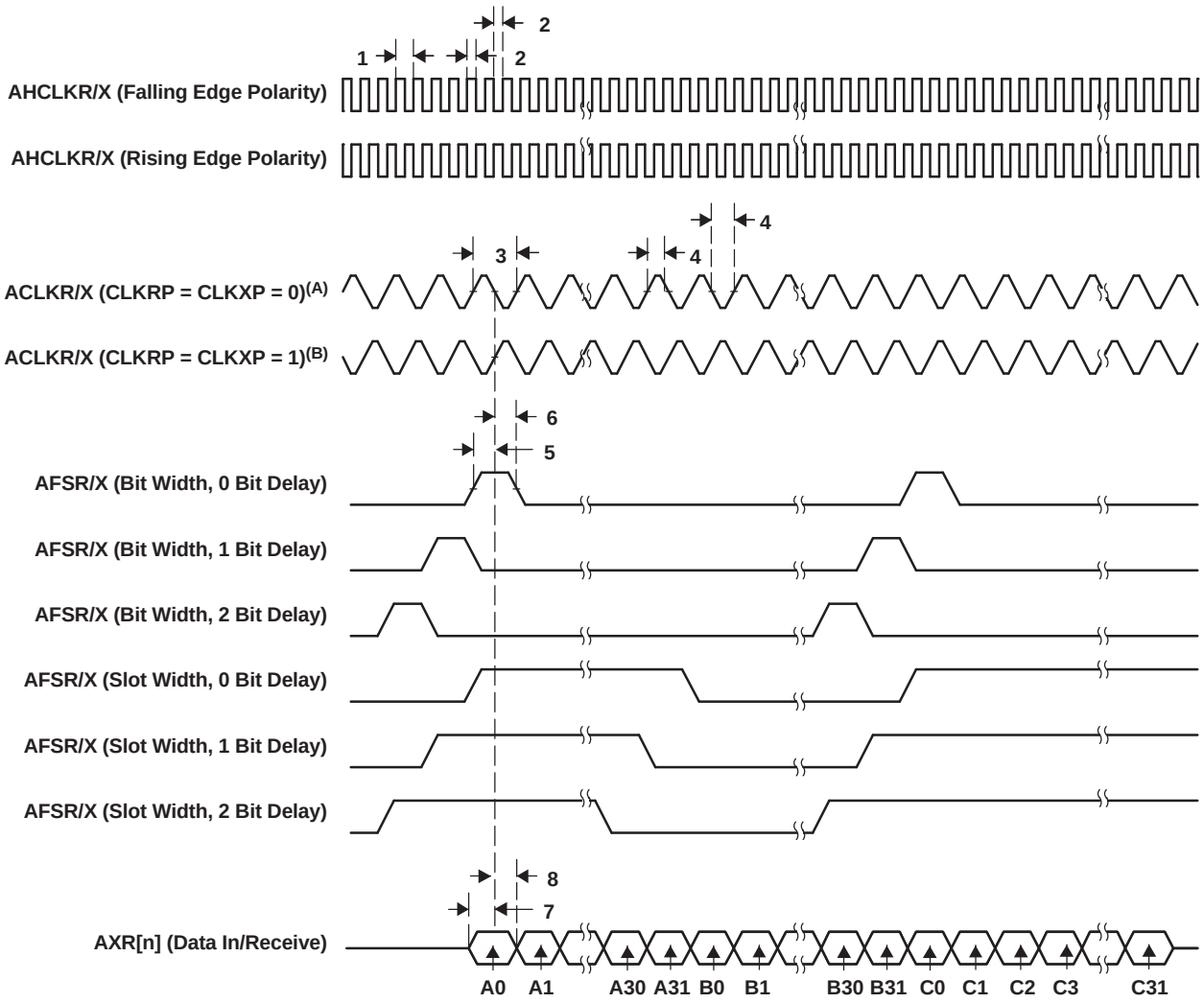
NO.			MIN	MAX	UNIT
8	$t_{h(ACLKRX-AXR)}$	Hold time, AXR2[n] input after ACLKR2 internal(3)	-2.2		ns
		Hold time, AXR2[n] input after ACLKX2 internal(4)	-2.2		
		Hold time, AXR2[n] input after ACLKR2 external input(3)	1.3		
		Hold time, AXR2[n] input after ACLKX2 external input(4)	1.3		
		Hold time, AXR2[n] input after ACLKR2 external output(3)	1.3		
		Hold time, AXR2[n] input after ACLKX2 external output(4)	1.3		

PRODUCT PREVIEW

Table 6-47. McASP2 Switching Characteristics(1)

NO.	PARAMETER		MIN	MAX	UNIT
9	$t_{c(AHCLKRX)}$	Cycle time, AHCLKX2 internal, AHCLKR2 output	13		ns
		Cycle time, AHCLKR2 external, AHCLKR2 output	13		
		Cycle time, AHCLKX2 internal, AHCLKX2 output	13		
		Cycle time, AHCLKX2 external, AHCLKX2 output	13		
10	$t_{w(AHCLKRX)}$	Pulse duration, AHCLKR2 internal, AHCLKR2 output	(AHR/2) – 2.5(2)		ns
		Pulse duration, AHCLKR2 external, AHCLKR2 output	(AHR/2) – 2.5(2)		
		Pulse duration, AHCLKX2 internal, AHCLKX2 output	(AHX/2) – 2.5(3)		
		Pulse duration, AHCLKX2 external, AHCLKX2 output	(AHX/2) – 2.5(3)		
11	$t_{c(ACLKRX)}$	Cycle time, ACLKR2 internal, ACLKR2 output	greater of 2P or 13 ns(4)		ns
		Cycle time, ACLKR2 external, ACLKR2 output	greater of 2P or 13 ns(4)		
		Cycle time, ACLKX2 internal, ACLKX2 output	greater of 2P or 13 ns(4)		
		Cycle time, ACLKX2 external, ACLKX2 output	greater of 2P or 13 ns(4)		
12	$t_{w(ACLKRX)}$	Pulse duration, ACLKR2 internal, ACLKR2 output	(AR/2) – 2.5(5)		ns
		Pulse duration, ACLKR2 external, ACLKR2 output	(AR/2) – 2.5(5)		
		Pulse duration, ACLKX2 internal, ACLKX2 output	(AX/2) – 2.5(6)		
		Pulse duration, ACLKX2 external, ACLKX2 output	(AX/2) – 2.5(6)		
13	$t_{d(ACLKRX-AFSRX)}$	Delay time, ACLKR2 internal, AFSR output(7)	-1.4	2.8	ns
		Delay time, ACLKX2 internal, AFSR output(8)	-1.4	2.8	
		Delay time, ACLKX2 internal, AFSX output	-1.4	2.8	
		Delay time, ACLKR2 external input, AFSR output(7)	2.9	10	
		Delay time, ACLKX2 external input, AFSR output(8)	2.9	10	
		Delay time, ACLKX2 external input, AFSX output	2.9	10	
		Delay time, ACLKR2 external output, AFSR output(7)	2.9	10	
		Delay time, ACLKX2 external output, AFSR output(8)	2.9	10	
		Delay time, ACLKX2 external output, AFSX output	2.9	10	
14	$t_{d(ACLKX-AXRV)}$	Delay time, ACLKX2 internal, AXR2[n] output	-1.4	2.8	ns
		Delay time, ACLKX2 external input, AXR2[n] output	2.9	10	
		Delay time, ACLKX2 external output, AXR2[n] output	2.9	10	
15	$t_{dis(ACLKX-AXRHZ)}$	Disable time, ACLKX2 internal, AXR2[n] output	-1.4	2.8	ns
		Disable time, ACLKX2 external input, AXR2[n] output	2.9	10	
		Disable time, ACLKX2 external output, AXR2[n] output	2.9	10	

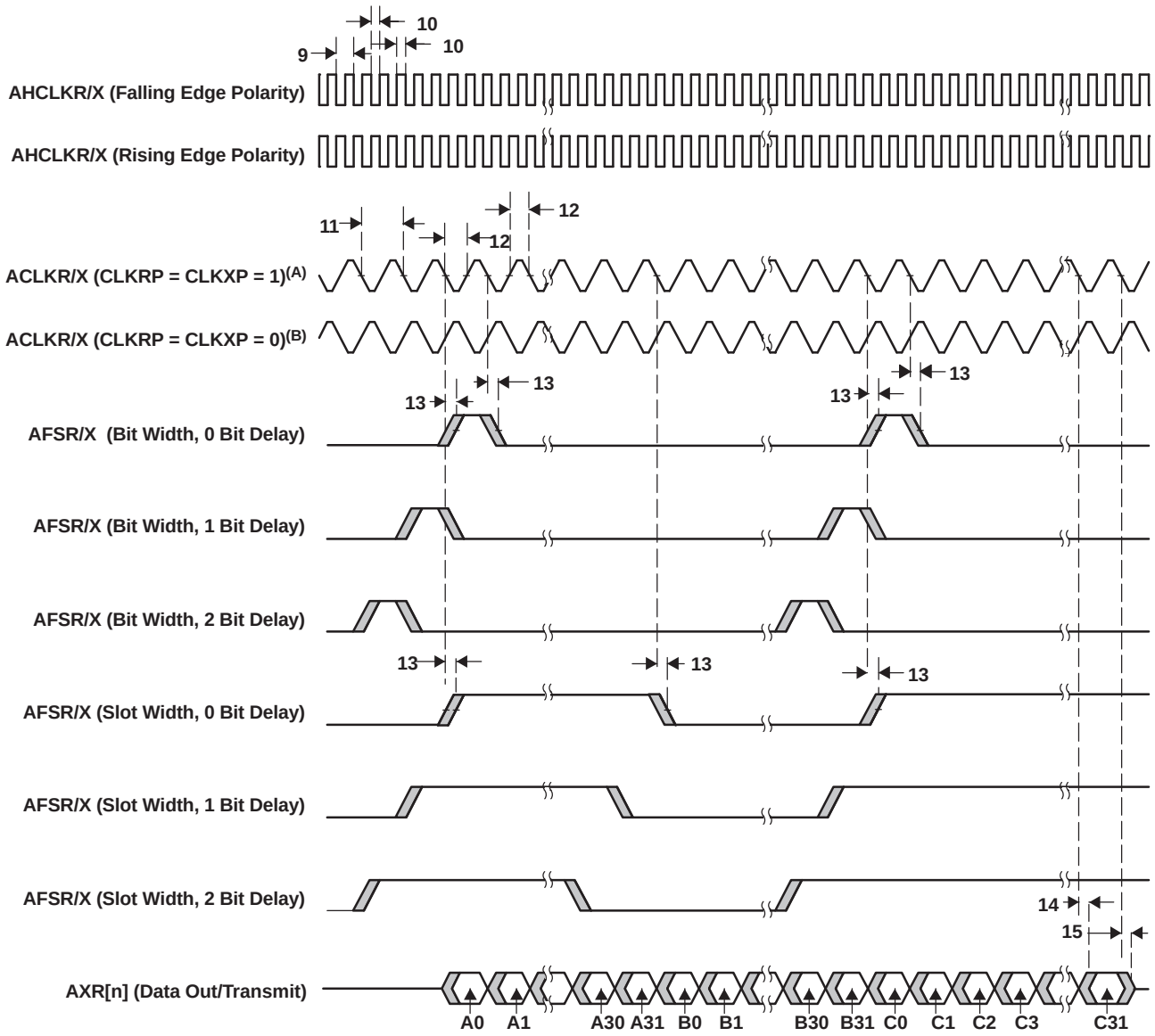
(1) McASP2 ACLKX2 internal – ACLKXCTL.CLKXM = 1, PDIR.ACLKX = 1
McASP2 ACLKX2 external input – ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 0
McASP2 ACLKX2 external output – ACLKXCTL.CLKXM = 0, PDIR.ACLKX = 1
McASP2 ACLKR2 internal – ACLKR2CTL.CLKRM = 1, PDIR.ACLKR =1
McASP2 ACLKR2 external input – ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 0
McASP2 ACLKR2 external output – ACLKRCTL.CLKRM = 0, PDIR.ACLKR = 1
(2) AHR - Cycle time, AHCLKR2.
(3) AHX - Cycle time, AHCLKX2.
(4) P = SYSCLK2 period
(5) AR - ACLKR2 period.
(6) AX - ACLKX2 period.
(7) McASP2 ACLKXCTL.ASYNC=1: Receiver is clocked by its own ACLKR2
(8) McASP2 ACLKXCTL.ASYNC=0: Receiver is clocked by transmitter's ACLKX2



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 6-33. 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 6-34. McASP Output Timings

PRODUCT PREVIEW

6.16 Serial Peripheral Interface Ports (SPI0, SPI1)

Figure 6-35 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.

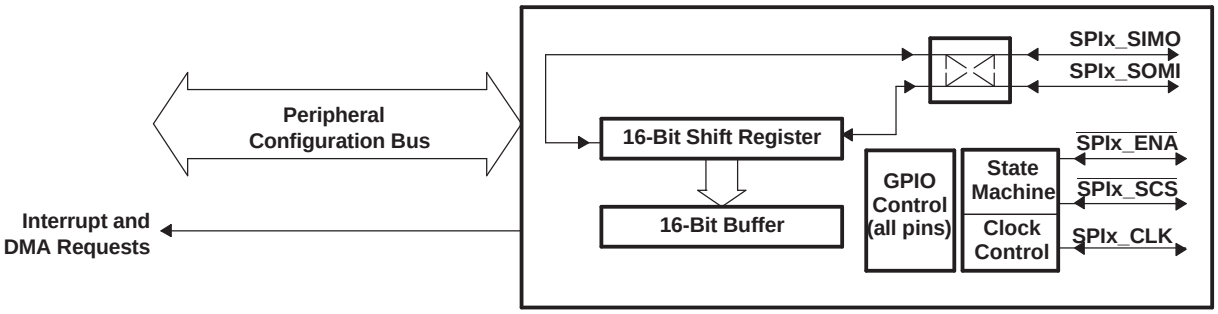


Figure 6-35. Block Diagram of SPI Module

The SPI supports 3-, 4-, and 5-pin operation with three basic pins (SPIx_CLK, SPIx_SIMO, and SPIx_SOMI) and two optional pins ($\overline{\text{SPIx\_SCS}}$, $\overline{\text{SPIx\_ENA}}$).

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 OMAP-L137 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 and can be driven in either a push-pull or open-drain manner. 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.

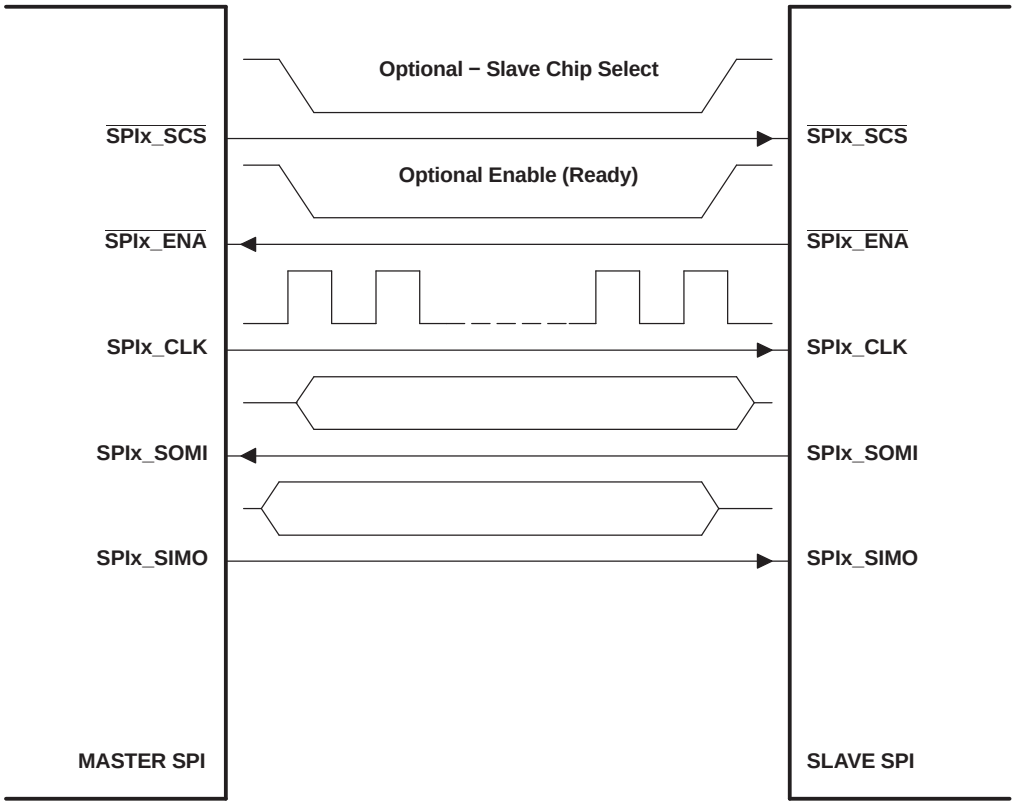


Figure 6-36. Illustration of SPI Master-to-SPI Slave Connection

6.16.1 SPI Peripheral Registers Description(s)

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

Table 6-48. SPIx Configuration Registers

SPI0 BYTE ADDRESS	SPI1 BYTE ADDRESS	REGISTER NAME	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	SPILVL	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

Table 6-48. SPIx Configuration Registers (continued)

SPI0 BYTE ADDRESS	SPI1 BYTE ADDRESS	REGISTER NAME	DESCRIPTION
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	INTVEC0	Interrupt Vector for SPI INT0
0x01C4 1064	0x01E1 2064	INTVEC1	Interrupt Vector for SPI INT1

6.16.2 SPI Electrical Data/Timing

6.16.2.1 Serial Peripheral Interface (SPI) Timing

Table 6-49 through Table 6-64 assume testing over recommended operating conditions (see Figure 6-37 through Figure 6-40).

Table 6-49. General Timing Requirements for SPI0 Master Modes(1)

NO.				MIN	MAX	UNIT
1	$t_{c(SPC)M}$	Cycle Time, SPI0_CLK, All Master Modes		greater of 2P or 20 ns	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
4,5	$t_{d(SIMO\_SPC)M}$	Delay, initial data bit valid on SPI0_SIMO to initial edge on SPI0_CLK(2)	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(1)

NO.				MIN	MAX	UNIT
9	$t_{c(SPC)S}$	Cycle Time, SPI0_CLK, All Slave Modes		greater of 2P or 20 ns	256P	ns
10	$t_{w(SPCH)S}$	Pulse Width High, SPI0_CLK, All Slave Modes		10		ns
11	$t_{w(SPCL)S}$	Pulse Width Low, SPI0_CLK, All Slave Modes		10		ns
12	$t_{su(SOMI\_SPC)S}$	Setup time, transmit data written to SPI before initial clock edge from master.(2)(3)	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		9	ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling		9	
			Polarity = 1, Phase = 0, from SPI0_CLK falling		9	
			Polarity = 1, Phase = 1, from SPI0_CLK rising		9	
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 DSP CPU.

Table 6-51. Additional[1] SPI0 Master Timings, 4-Pin Enable Option[2][3]

NO.				MIN	MAX	UNIT
17	$t_{d(ENA\_SPC)M}$	Delay from slave assertion of SPI0_ENA active to first SPI0_CLK from master.[4]	Polarity = 0, Phase = 0, to SPI0_CLK rising		3P + 5	ns
			Polarity = 0, Phase = 1, to SPI0_CLK rising		$0.5t_{c(SPC)M}$ + 3P + 5	
			Polarity = 1, Phase = 0, to SPI0_CLK falling		3P + 5	
			Polarity = 1, Phase = 1, to SPI0_CLK falling		$0.5t_{c(SPC)M}$ + 3P + 5	
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.[5]	Polarity = 0, Phase = 0, from SPI0_CLK falling		$0.5t_{c(SPC)M}$	ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling		0	
			Polarity = 1, Phase = 0, from SPI0_CLK rising		$0.5t_{c(SPC)M}$	
			Polarity = 1, Phase = 1, from SPI0_CLK rising		0	

(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_EN A deassertion.

Table 6-52. Additional[1] SPI0 Master Timings, 4-Pin Chip Select Option[2][3]

NO.				MIN	MAX	UNIT
19	$t_{d(SCS\_SPC)M}$	Delay from $\overline{SPI0\_SCS}$ active to first SPI0_CLK[4][5]	Polarity = 0, Phase = 0, to SPI0_CLK rising	2P -3		ns
			Polarity = 0, Phase = 1, to SPI0_CLK rising	$0.5t_{c(SPC)M}$ + 2P -3		
			Polarity = 1, Phase = 0, to SPI0_CLK falling	2P -3		
			Polarity = 1, Phase = 1, to SPI0_CLK falling	$0.5t_{c(SPC)M}$ + 2P -3		
20	$t_{d(SPC\_SCS)M}$	Delay from final SPI0_CLK edge to master deasserting $\overline{SPI0\_SCS}$[6][7]	Polarity = 0, Phase = 0, from SPI0_CLK falling	$0.5t_{c(SPC)M}$		ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI0_CLK rising	$0.5t_{c(SPC)M}$		
			Polarity = 1, Phase = 1, from SPI0_CLK rising	0		

(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 $\overline{SPI0\_SCS}$ 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, $\overline{SPI0\_SCS}$ 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[1] SPI0 Master Timings, 5-Pin Option[2][3]

NO.				MIN	MAX	UNIT
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.[4]	Polarity = 0, Phase = 0, from SPI0_CLK falling		P + 5	ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling		$0.5t_{c(SPC)M}$ + P + 5	
			Polarity = 1, Phase = 0, from SPI0_CLK rising		P + 5	
			Polarity = 1, Phase = 1, from SPI0_CLK rising		$0.5t_{c(SPC)M}$ + P + 5	
20	$t_{d(SPC\_SCS)M}$	Delay from final SPI0_CLK edge to master deasserting SPI0_SCS [5][6]	Polarity = 0, Phase = 0, from SPI0_CLK falling	$0.5t_{c(SPC)M}$		ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI0_CLK rising	$0.5t_{c(SPC)M}$		
			Polarity = 1, Phase = 1, from SPI0_CLK rising	0		
21	$t_{d(SCSL\_ENAL)M}$	Max delay for slave SPI to drive SPI0_ENA valid after master asserts SPI0_SCS to delay the master from beginning the next transfer,			C2TDELAY + P	ns
22	$t_{d(SCS\_SPC)M}$	Delay from SPI0_SCS active to first SPI0_CLK[7][8][9]	Polarity = 0, Phase = 0, to SPI0_CLK rising	2P -3		ns
			Polarity = 0, Phase = 1, to SPI0_CLK rising	$0.5t_{c(SPC)M}$ + 2P -3		
			Polarity = 1, Phase = 0, to SPI0_CLK falling	2P -3		
			Polarity = 1, Phase = 1, to SPI0_CLK falling	$0.5t_{c(SPC)M}$ + 2P -3		
23	$t_{d(ENA\_SPC)M}$	Delay from assertion of SPI0_ENA low to first SPI0_CLK edge.[10]	Polarity = 0, Phase = 0, to SPI0_CLK rising		3P + 5	ns
			Polarity = 0, Phase = 1, to SPI0_CLK rising		$0.5t_{c(SPC)M}$ + 3P + 5	
			Polarity = 1, Phase = 0, to SPI0_CLK falling		3P + 5	
			Polarity = 1, Phase = 1, to SPI0_CLK falling		$0.5t_{c(SPC)M}$ + 3P + 5	

(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 SPI0_ENA deassertion.
(5) Except for modes when SPIDAT1.CSHOLD is enabled and there is additional data to transmit. In this case, 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 SPI0_ENA is asserted immediately such that the transmission is not delayed by SPI0_ENA.
(8) In the case where the master SPI is ready with new data before SPI0_SCS assertion.
(9) This delay can be increased under software control by the register bit field SPIDELAY.C2TDELAY[4:0].
(10) If SPI0_ENA was initially deasserted high and SPI0_CLK is delayed.

Table 6-54. Additional[1] SPI0 Slave Timings, 4-Pin Enable Option[2][3]

NO.				MIN	MAX	UNIT
24	$t_{d(SPC\_ENAH)S}$	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 + 9	ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling	$-0.5t_{c(SPC)M}$ + 1.5 P -3	$-0.5t_{c(SPC)M}$ + 2.5 P + 9	
			Polarity = 1, Phase = 0, from SPI0_CLK rising	1.5 P -3	2.5 P + 9	
			Polarity = 1, Phase = 1, from SPI0_CLK rising	$-0.5t_{c(SPC)M}$ + 1.5 P -3	$-0.5t_{c(SPC)M}$ + 2.5 P + 9	

(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[1] SPI0 Slave Timings, 4-Pin Chip Select Option[2][3]

NO.				MIN	MAX	UNIT
25	$t_{d(SCSL\_SPC)S}$	Required delay from SPI0_SCS asserted at slave to first SPI0_CLK edge at slave.		P		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	$0.5t_{c(SPC)M}$ + 0		ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI0_CLK rising	$0.5t_{c(SPC)M}$ + 0		
			Polarity = 1, Phase = 1, from SPI0_CLK rising	0		
27	$t_{ena(SCSL\_SOMI)S}$	Delay from master asserting SPI0_SCS to slave driving SPI0_SOMI valid			P + 9	ns
28	$t_{dis(SCSH\_SOMI)S}$	Delay from master deasserting SPI0_SCS to slave 3-stating SPI0_SOMI			P + 9	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[1] SPI0 Slave Timings, 5-Pin Option[2][3]

NO.				MIN	MAX	UNIT
25	$t_{d(SCSL\_SPC)S}$	Required delay from SPI0_SCS asserted at slave to first SPI0_CLK edge at slave.		P		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	$0.5t_{c(SPC)M}$ + 0		ns
			Polarity = 0, Phase = 1, from SPI0_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI0_CLK rising	$0.5t_{c(SPC)M}$ + 0		
			Polarity = 1, Phase = 1, from SPI0_CLK rising	0		
27	$t_{ena(SCSL\_SOMI)S}$	Delay from master asserting SPI0_SCS to slave driving SPI0_SOMI valid			P + 9	ns
28	$t_{dis(SCSH\_SOMI)S}$	Delay from master deasserting SPI0_SCS to slave 3-stating SPI0_SOMI			P + 9	ns
29	$t_{ena(SCSL\_ENA)S}$	Delay from master deasserting SPI0_SCS to slave driving SPI0_ENA valid			9	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.[4]	Polarity = 0, Phase = 0, from SPI0_CLK falling		2.5 P + 9	ns
			Polarity = 0, Phase = 1, from SPI0_CLK rising		2.5 P + 9	
			Polarity = 1, Phase = 0, from SPI0_CLK rising		2.5 P + 9	
			Polarity = 1, Phase = 1, from SPI0_CLK falling		2.5 P + 9	

(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 tri-stated. If tri-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[1]

NO.				MIN	MAX	UNIT
1	$t_{c(SPC)M}$	Cycle Time, SPI1_CLK, All Master Modes		greater of 2P or 20 ns	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
4,5	$t_{d(SIMO\_SPC)M}$	Delay, initial data bit valid on SPI1_SIMO to initial edge on SPI1_CLK[2]	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 (continued)

NO.				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(1)

NO.				MIN	MAX	UNIT
9	$t_{c(SPC)S}$	Cycle Time, SPI1_CLK, All Slave Modes		greater of 2P or 20 ns	256P	ns
10	$t_{w(SPCH)S}$	Pulse Width High, SPI1_CLK, All Slave Modes		10		ns
11	$t_{w(SPCL)S}$	Pulse Width Low, SPI1_CLK, All Slave Modes		10		ns
12	$t_{su(SOMI\_SPC)S}$	Setup time, transmit data written to SPI before initial clock edge from master.(2)(3)	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		9.7	ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling		9.7	
			Polarity = 1, Phase = 0, from SPI1_CLK falling		9.7	
			Polarity = 1, Phase = 1, from SPI1_CLK rising		9.7	

PRODUCT PREVIEW

(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 DSP CPU.

Table 6-58. General Timing Requirements for SPI1 Slave Modes (continued)

NO.				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(1) SPI1 Master Timings, 4-Pin Enable Option(2)(3)

NO.				MIN	MAX	UNIT
17	$t_{d(EN\ A\_SPC)M}$	Delay from slave assertion of SPI1_ENA active to first SPI1_CLK from master.(4)	Polarity = 0, Phase = 0, to SPI1_CLK rising	3P + 5		ns
			Polarity = 0, Phase = 1, to SPI1_CLK rising	$0.5t_{c(SPC)M}$ + 3P + 5		
			Polarity = 1, Phase = 0, to SPI1_CLK falling	3P + 5		
			Polarity = 1, Phase = 1, to SPI1_CLK falling	$0.5t_{c(SPC)M}$ + 3P + 5		
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.(5)	Polarity = 0, Phase = 0, from SPI1_CLK falling	$0.5t_{c(SPC)M}$		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	$0.5t_{c(SPC)M}$		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	0		

(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[1] SPI1 Master Timings, 4-Pin Chip Select Option[2][3]

NO.				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 -3		ns
			Polarity = 0, Phase = 1, to SPI1_CLK rising	$0.5t_{c(SPC)M}$ + 2P -3		
			Polarity = 1, Phase = 0, to SPI1_CLK falling	2P -3		
			Polarity = 1, Phase = 1, to SPI1_CLK falling	$0.5t_{c(SPC)M}$ + 2P -3		
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	$0.5t_{c(SPC)M}$		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	$0.5t_{c(SPC)M}$		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	0		

(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[1] SPI1 Master Timings, 5-Pin Option[2][3]

NO.				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.[4]	Polarity = 0, Phase = 0, from SPI1_CLK falling	P + 5		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	$0.5t_{c(SPC)M}$ + P + 5		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	P + 5		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	$0.5t_{c(SPC)M}$ + P + 5		
20	$t_{d(SPC\_SCS)M}$	Delay from final SPI1_CLK edge to master deasserting $\overline{SPI1\_SCS}$[5][6]	Polarity = 0, Phase = 0, from SPI1_CLK falling	$0.5t_{c(SPC)M}$		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	$0.5t_{c(SPC)M}$		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	0		
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].

PRODUCT PREVIEW

Table 6-61. Additional SPI1 Master Timings, 5-Pin Option (continued)

NO.				MIN	MAX	UNIT
22	$t_{d(SCS\_SPC)M}$	Delay from SPI1_SCS active to first SPI1_CLK (7) (8) (9)	Polarity = 0, Phase = 0, to SPI1_CLK rising	2P -3		ns
			Polarity = 0, Phase = 1, to SPI1_CLK rising	$0.5t_{c(SPC)M}$ + 2P -3		
			Polarity = 1, Phase = 0, to SPI1_CLK falling	2P -3		
			Polarity = 1, Phase = 1, to SPI1_CLK falling	$0.5t_{c(SPC)M}$ + 2P -3		
23	$t_{d(ENA\_SPC)M}$	Delay from assertion of SPI1_ENA low to first SPI1_CLK edge. (10)	Polarity = 0, Phase = 0, to SPI1_CLK rising		3P + 5	ns
			Polarity = 0, Phase = 1, to SPI1_CLK rising		$0.5t_{c(SPC)M}$ + 3P + 5	
			Polarity = 1, Phase = 0, to SPI1_CLK falling		3P + 5	
			Polarity = 1, Phase = 1, to SPI1_CLK falling		$0.5t_{c(SPC)M}$ + 3P + 5	

(7) If SPI1_ENA is asserted immediately such that the transmission is not delayed by SPI1_ENA.
(8) In the case where the master SPI is ready with new data before SPI1_SCS assertion.
(9) This delay can be increased under software control by the register bit field SPIDELAY.C2TDELAY[4:0].
(10) If SPI1_ENA was initially deasserted high and SPI1_CLK is delayed.

Table 6-62. Additional (1) SPI1 Slave Timings, 4-Pin Enable Option (2) (3)

NO.				MIN	MAX	UNIT
24	$t_{d(SPC\_ENAH)S}$	Delay from final SPI1_CLK edge to slave deasserting SPI1_ENA.	Polarity = 0, Phase = 0, from SPI1_CLK falling	1.5 P -3	2.5 P + 9.7	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 + 9.7	
			Polarity = 1, Phase = 0, from SPI1_CLK rising	1.5 P -3	2.5 P + 9.7	
			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 + 9.7	

(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 (1) SPI1 Slave Timings, 4-Pin Chip Select Option (2) (3)

NO.				MIN	MAX	UNIT
25	$t_{d(SCSL\_SPC)S}$	Required delay from SPI1_SCS asserted at slave to first SPI1_CLK edge at slave.		P		ns
26	$t_{d(SPC\_SCSH)S}$	Required delay from final SPI1_CLK edge before SPI1_SCS is deasserted.	Polarity = 0, Phase = 0, from SPI1_CLK falling	$0.5t_{c(SPC)M}$ + 0		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	$0.5t_{c(SPC)M}$ + 0		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	0		
27	$t_{ena(SCSL\_SOMI)S}$	Delay from master asserting SPI1_SCS to slave driving SPI1_SOMI valid			P + 9.7	ns
28	$t_{dis(SCSH\_SOMI)S}$	Delay from master deasserting SPI1_SCS to slave 3-stating SPI1_SOMI			P + 9.7	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[1] SPI1 Slave Timings, 5-Pin Option[2][3]

NO.				MIN	MAX	UNIT
25	$t_{d(SCSL\_SPC)S}$	Required delay from SPI1_SCS asserted at slave to first SPI1_CLK edge at slave.		P		ns
26	$t_{d(SPC\_SCSH)S}$	Required delay from final SPI1_CLK edge before SPI1_SCS is deasserted.	Polarity = 0, Phase = 0, from SPI1_CLK falling	$0.5t_{c(SPC)M}$ + 0		ns
			Polarity = 0, Phase = 1, from SPI1_CLK falling	0		
			Polarity = 1, Phase = 0, from SPI1_CLK rising	$0.5t_{c(SPC)M}$ + 0		
			Polarity = 1, Phase = 1, from SPI1_CLK rising	0		
27	$t_{ena(SCSL\_SOMI)S}$	Delay from master asserting SPI1_SCS to slave driving SPI1_SOMI valid			P + 9.7	ns
28	$t_{dis(SCSH\_SOMI)S}$	Delay from master deasserting SPI1_SCS to slave 3-stating SPI1_SOMI			P + 9.7	ns
29	$t_{ena(SCSL\_ENA)S}$	Delay from master deasserting SPI1_SCS to slave driving SPI1_ENA valid			9.7	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.[4]	Polarity = 0, Phase = 0, from SPI1_CLK falling		2.5 P + 9.7	ns
			Polarity = 0, Phase = 1, from SPI1_CLK rising		2.5 P + 9.7	
			Polarity = 1, Phase = 0, from SPI1_CLK rising		2.5 P + 9.7	
			Polarity = 1, Phase = 1, from SPI1_CLK falling		2.5 P + 9.7	

(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 tri-stated. If tri-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.

PRODUCT PREVIEW

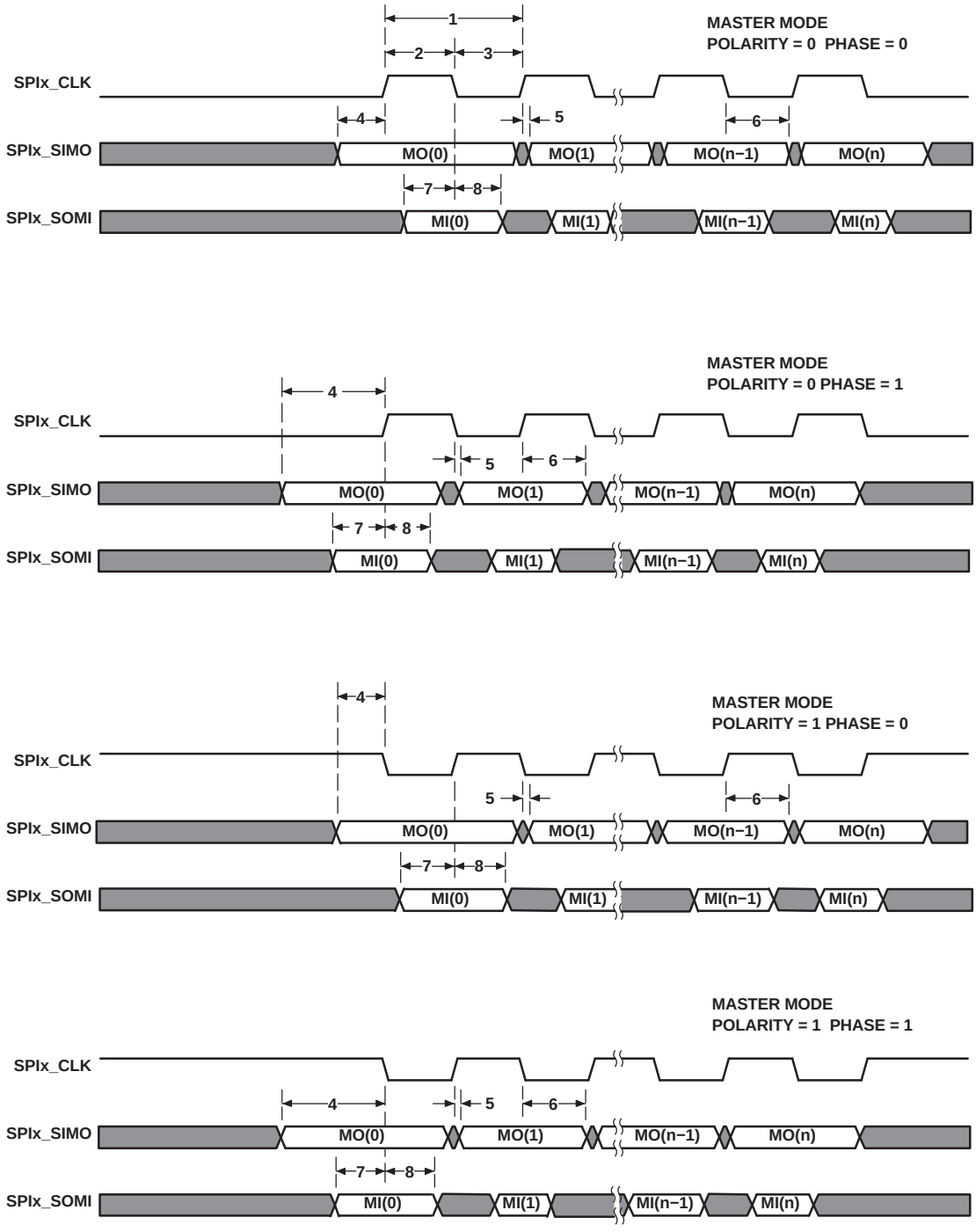


Figure 6-37. SPI Timings—Master Mode

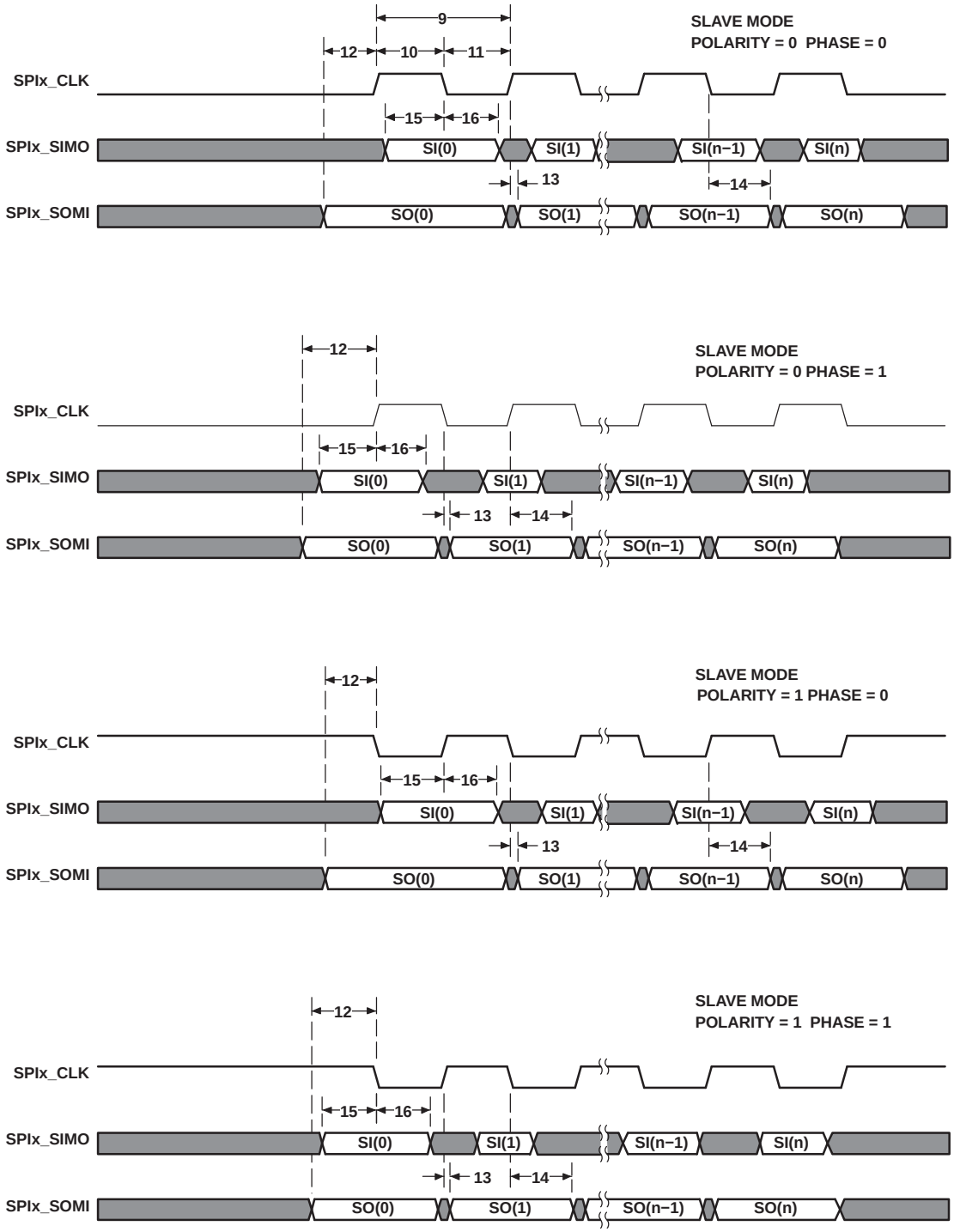


Figure 6-38. SPI Timings—Slave Mode

PRODUCT PREVIEW

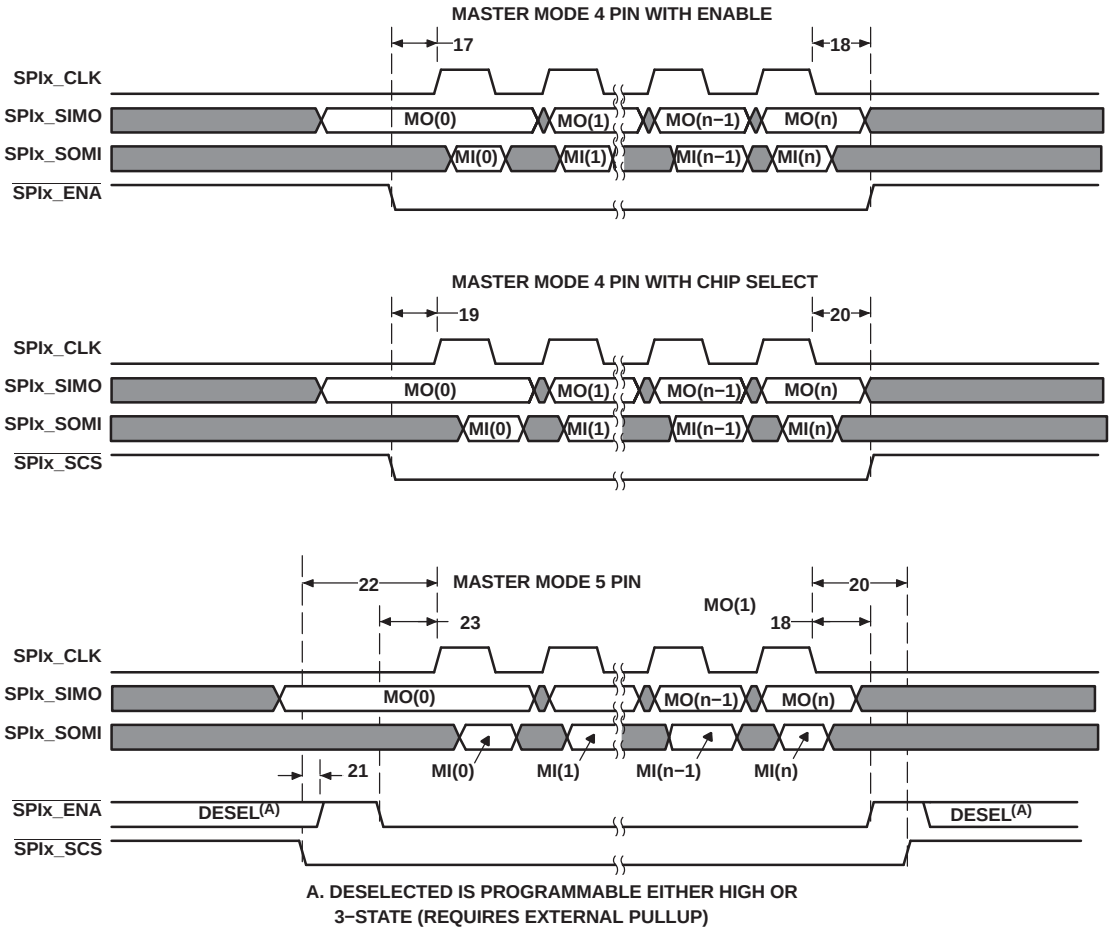


Figure 6-39. SPI Timings—Master Mode (4-Pin and 5-Pin)

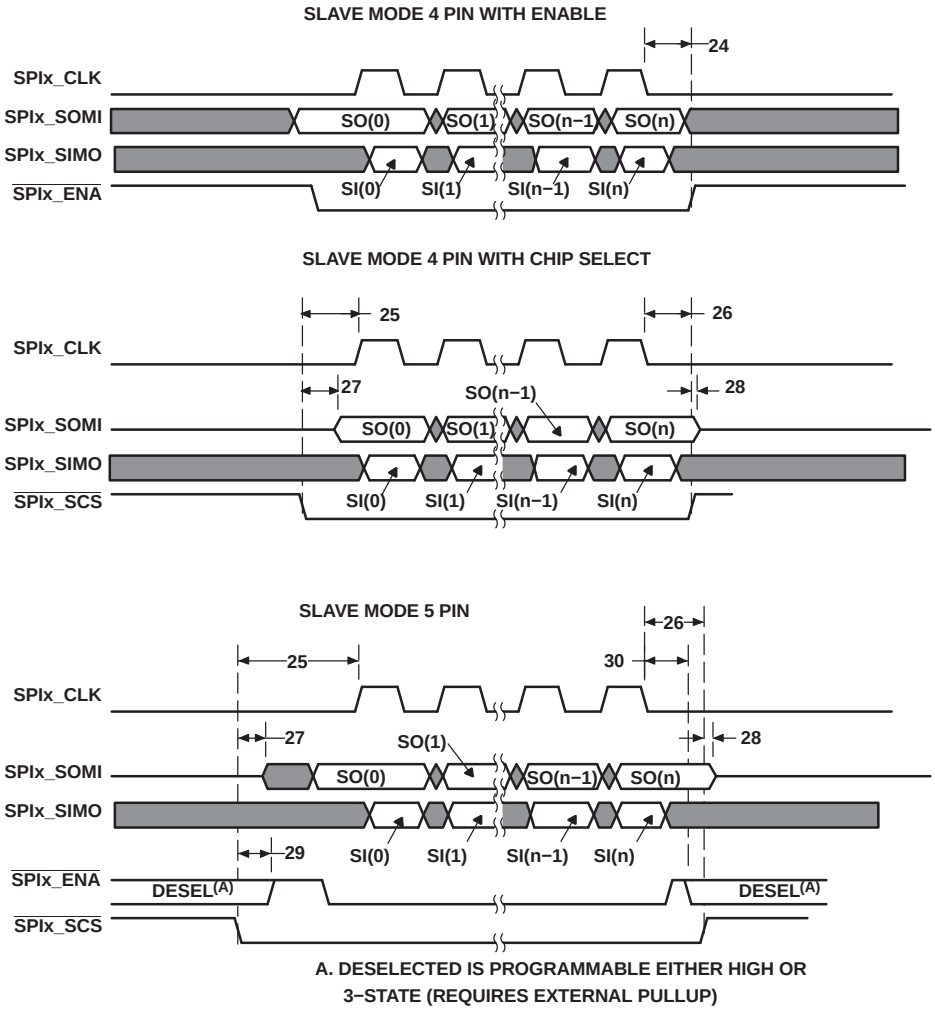


Figure 6-40. SPI Timings—Slave Mode (4-Pin and 5-Pin)

PRODUCT PREVIEW

6.17 ECAP Peripheral Registers Description(s)

The OMAP-L137 device contains up to three enhanced capture (eCAP) modules. Figure 6-41 shows a functional block diagram of a module. See the *OMAP-L137 Applications Processor DSP Peripherals Overview Reference Guide.* – Literature Number SPRUGA6 for more details.

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 cuty 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.

The clock enable bits (ECAP1/2/3/4ENCLK) in the PCLKCR1 register are used to turn off the eCAP modules individually (for low power operation). Upon reset, ECAP1ENCLK, ECAP2ENCLK, ECAP3ENCLK, and ECAP4EN CLK are set to low, indicating that the peripheral clock is off.

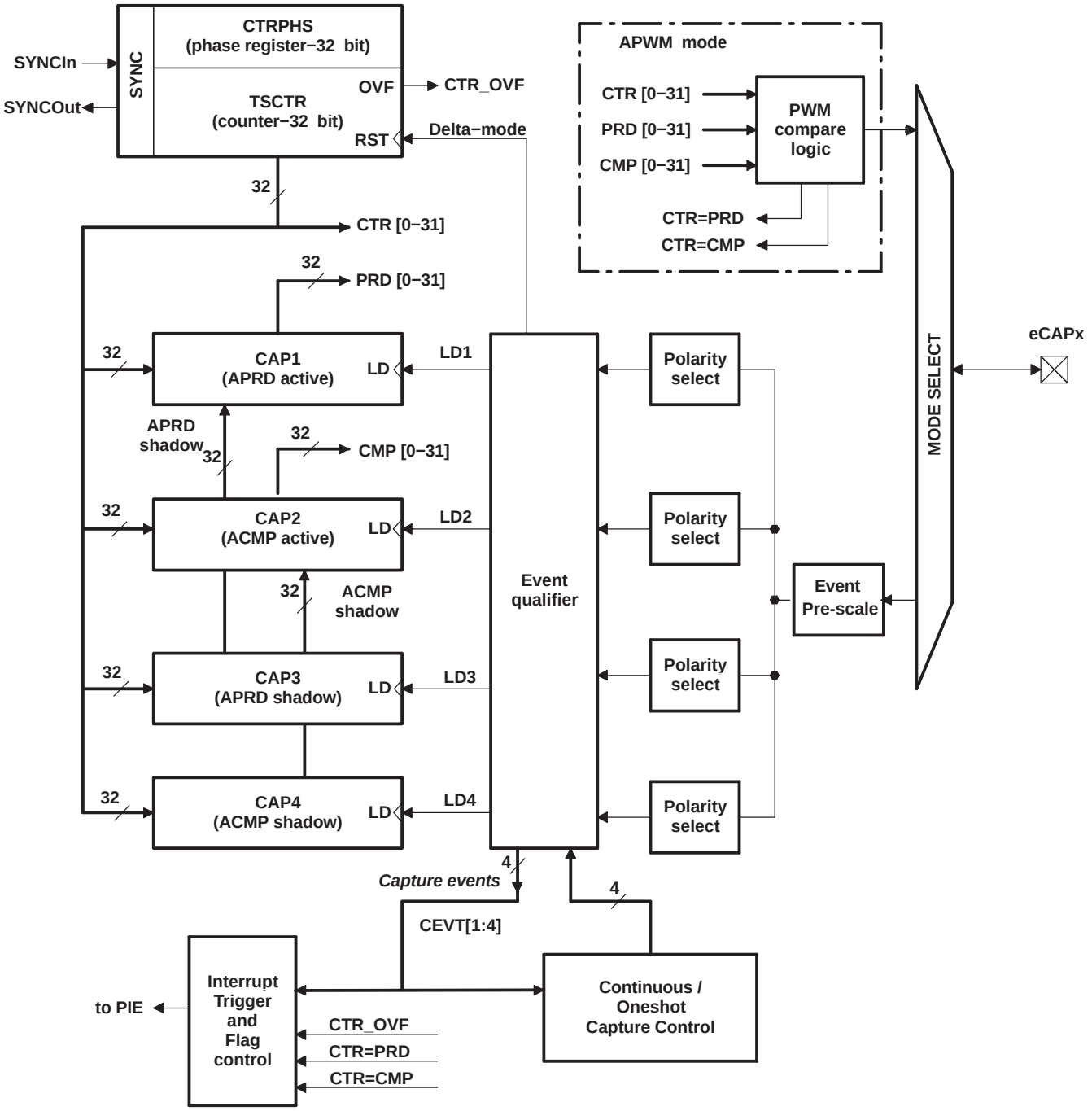


Figure 6-41. eCAP Functional Block Diagram

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

Table 6-65. ECAPx Configuration Registers

ECAP0 BYTE ADDRESS	ECAP1 BYTE ADDRESS	ECAP2 BYTE ADDRESS	REGISTER NAME	DESCRIPTION
0x01F0 6000	0x01F0 7000	0x01F0 8000	TSCTR	Time-Stamp Counter
0x01F0 6004	0x01F0 7004	0x01F0 8004	CTRPHS	Counter Phase Offset Value Register
0x01F0 6008	0x01F0 7008	0x01F0 8008	CAP1	Capture 1 Register

Table 6-65. ECAPx Configuration Registers (continued)

ECAP0 BYTE ADDRESS	ECAP1 BYTE ADDRESS	ECAP2 BYTE ADDRESS	REGISTER NAME	DESCRIPTION
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

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(CAP)}$	Capture input pulse width	Asynchronous	$2t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

Table 6-67. eCAP Switching Characteristics

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

PRODUCT PREVIEW

6.18 EQEP Peripheral Registers Description(s)

The OMAP-L137 device contains up to two enhanced quadrature encoder (eQEP) modules. See the *OMAP-L137 Applications Processor DSP Peripherals Overview Reference Guide.* – Literature Number SPRUGA6 . for more details.

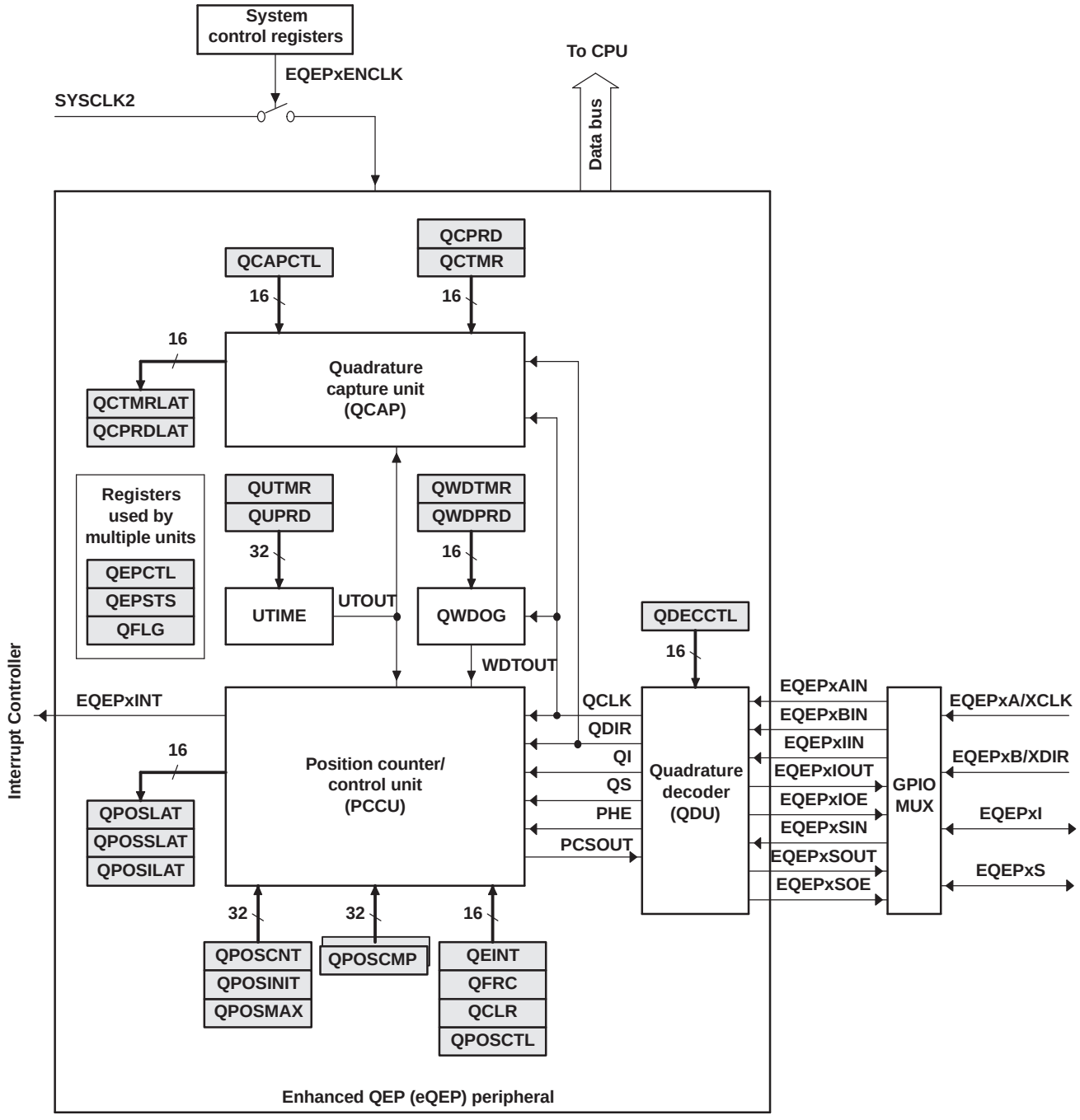


Figure 6-42. eQEP Functional Block Diagram

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 BYTE ADDRESS	EQEP1 BYTE ADDRESS	REGISTER NAME	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

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(QEPP)}$	QEP input period	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2(1t_{c(SCO)} + t_{w(IQSW)})$		cycles
$t_{w(INDEXH)}$	QEP Index Input High time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(INDEXL)}$	QEP Index Input Low time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(STROBH)}$	QEP Strobe High time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(STROBL)}$	QEP Strobe Input Low time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles

Table 6-70. eQEP Switching Characteristics

PARAMETER		TEST CONDITIONS	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

6.19 eHRPWM

The OMAP-L137 device contains up to three enhanced PWM Modules (eHRPWM). Figure 6-43 shows a block diagram of multiple eHRPWM modules. Figure 4-4 shows the signal interconnections with the eHRPWM. See the *OMAP-L137 Applications Processor DSP Peripherals Overview Reference Guide.* – Literature Number SPRUGA6 for more details.

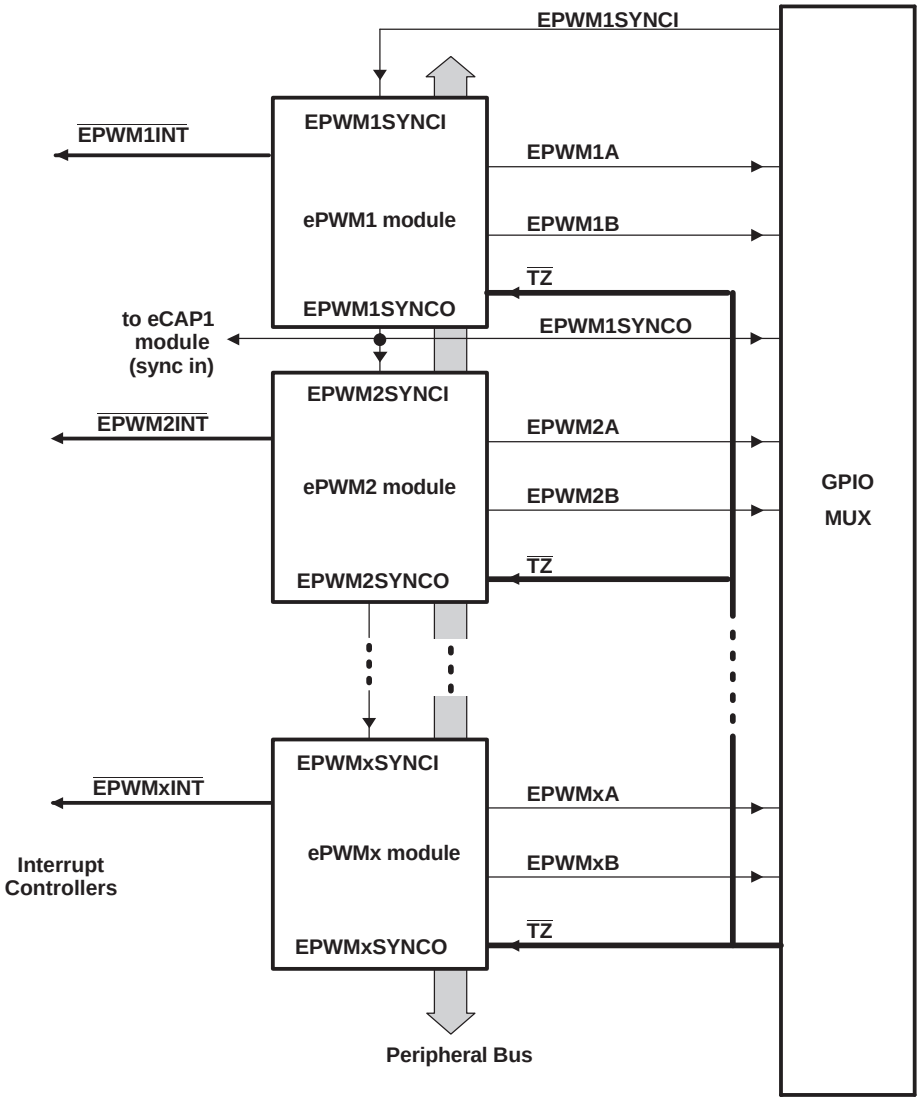


Figure 6-43. Multiple PWM Modules in a OMAP-L137 System

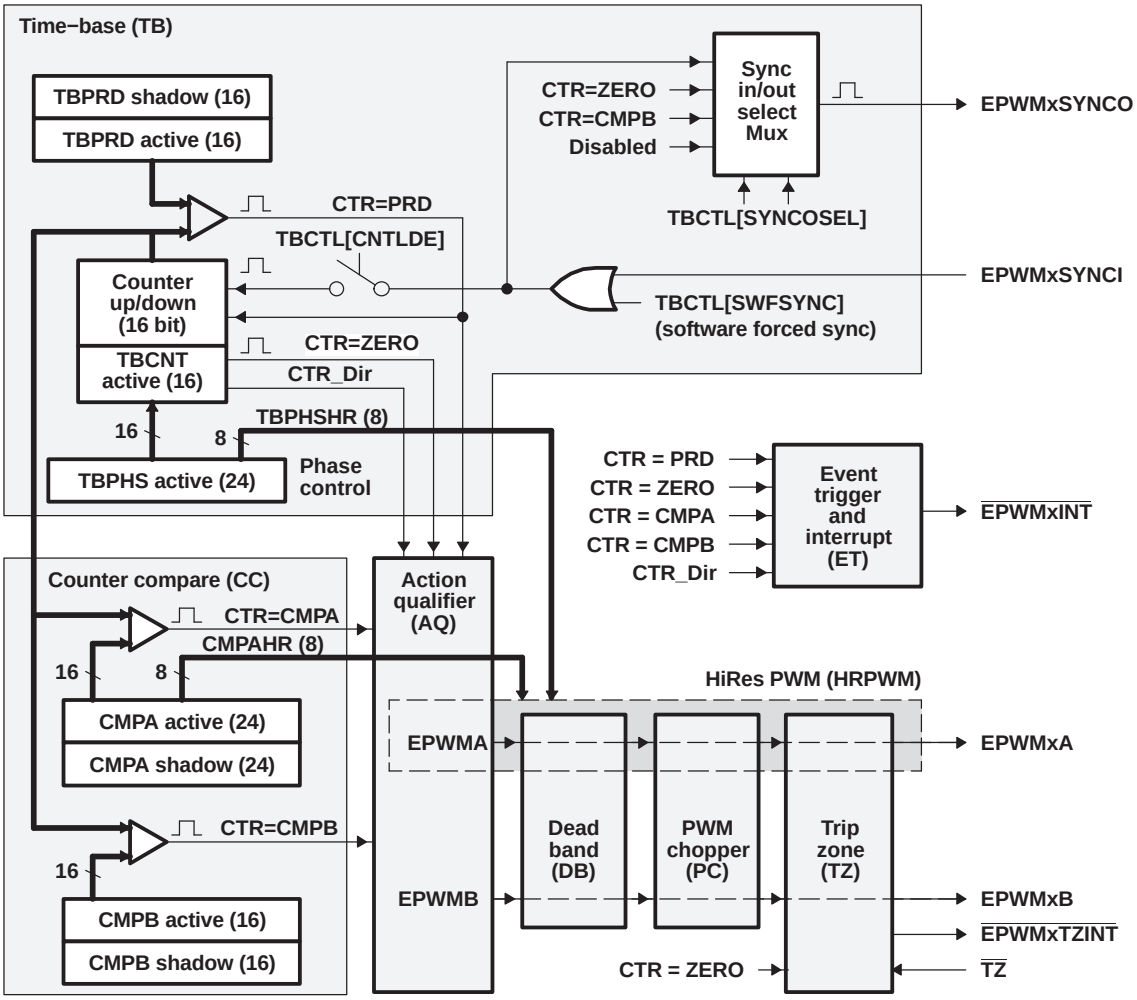


Figure 6-44. eHRPWM Sub-Modules Showing Critical Internal Signal Interconnections

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

eHRPWM1 BYTE ADDRESS	eHRPWM2 BYTE ADDRESS	eHRPWM3 BYTE ADDRESS	Acronym	Size (×16)	Shadow	Register Description
			Time-Base Submodule Registers			
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 (1)
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
			Counter-Compare Submodule Registers			
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 (1)
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

(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)

eHRPWM1 BYTE ADDRESS	eHRPWM2 BYTE ADDRESS	eHRPWM3 BYTE ADDRESS	Acronym	Size (×16)	Shadow	Register Description
			Action-Qualifier Submodule Registers			
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	AQSFRC	1	No	Action-Qualifier Software Force Register
0x01F0 001C	0x01F0 201C	0x01F0 401C	AQCSFRC	1	Yes	Action-Qualifier Continuous S/W Force Register Set
			Dead-Band Generator Submodule Registers			
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
			PWM-Chopper Submodule Registers			
0x01F0 003C	0x01F0 203C	0x01F0 403C	PCCTL	1	No	PWM-Chopper Control Register
			Trip-Zone Submodule Registers			
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
			Event-Trigger Submodule Registers			
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
			High-Resolution PWM (HRPWM) Submodule Registers			
0x01F0 1020	0x01F0 3020	0x01F0 5020	HRCNFG	1	No	HRPWM Configuration Register (1)

6.20 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

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(SYCIN)}$	Sync input pulse width	Asynchronous	$2t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

Table 6-73. eHRPWM Switching Characteristics

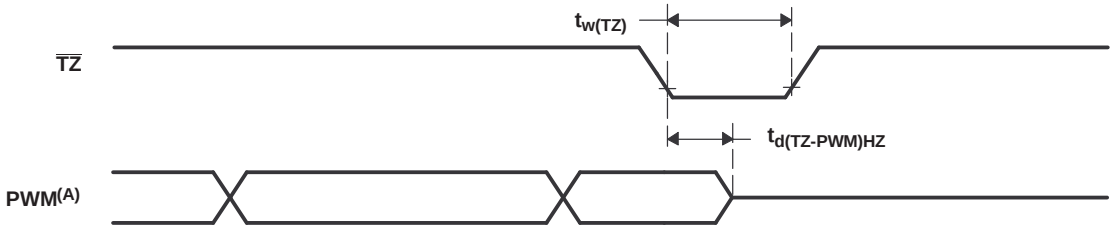
PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(PWM)}$	Pulse duration, PWMx output high/low		20		ns
$t_{w(SYNCOUT)}$	Sync output pulse width		$8t_{c(SCO)}$		cycles
$t_{d(PWM)tza}$	Delay time, trip input active to PWM forced high Delay time, trip input active to PWM forced low	no pin load		25	ns

PRODUCT PREVIEW

Table 6-73. eHRPWM Switching Characteristics (continued)

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{d(TZ-PWM)HZ}$	Delay time, trip input active to PWM Hi-Z			20	ns

6.21 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 6-45. PWM Hi-Z Characteristics

Table 6-74. Trip-Zone input Timing Requirements

			MIN	MAX	UNIT
$t_{w(TZ)}$	Pulse duration, $\overline{TZx}$ input low	Asynchronous	$1t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

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

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

	MIN	TYP	MAX	UNIT
Micro Edge Positioning (MEP) step size(1)		150	310	ps

(1) Maximum MEP step size is based on worst-case process, maximum temperature and maximum voltage. MEP step size will increase with low voltage and high temperature and decrease with voltage and cold temperature.
Applications that use the HRPWM feature should use MEP Scale Factor Optimizer (SFO) estimation software functions. See the TI software libraries for details of using SFO function in end applications. SFO functions help to estimate the number of MEP steps per SYSCLKOUT period dynamically while the HRPWM is in operation.

6.22 LCD Controller

Table 6-76 lists the LCD Controller registers

Table 6-76. LCD Controller (LCDC) Registers

Address Offset	Acronym	Register Description
0x01E1 3000	REVID	LCD Revision Identification Register
0x01E1 3004	LCD_CTRL	LCD Control Register
0x01E1 3008	LCD_STAT	LCD Status Register
0x01E1 300C	LIDD_CTRL	LCD LIDD Control Register
0x01E1 3010	LIDD_CS0_CONF	LCD LIDD CS0 Configuration Register
0x01E1 3014	LIDD_CS0_ADDR	LCD LIDD CS0 Address Read/Write Register
0x01E1 3018	LIDD_CS0_DATA	LCD LIDD CS0 Data Read/Write Register
0x01E1 301C	LIDD_CS1_CONF	LCD LIDD CS1 Configuration Register
0x01E1 3020	LIDD_CS1_ADDR	LCD LIDD CS1 Address Read/Write Register
0x01E1 3024	LIDD_CS1_DATA	LCD LIDD CS1 Data Read/Write Register
0x01E1 3028	RASTER_CTRL	LCD Raster Control Register
0x01E1 302C	RASTER_TIMING_0	LCD Raster Timing 0 Register
0x01E1 3030	RASTER_TIMING_1	LCD Raster Timing 1 Register
0x01E1 3034	RASTER_TIMING_2	LCD Raster Timing 2 Register
0x01E1 3038	RASTER_SUBPANEL	LCD Raster Subpanel Display Register
0x01E1 3040	LCDDMA_CTRL	LCD DMA Control Register
0x01E1 3044	LCDDMA_FB0_BASE	LCD DMA Frame Buffer 0 Base Address Register
0x01E1 3048	LCDDMA_FB0_CEILING	LCD DMA Frame Buffer 0 Ceiling Address Register
0x01E1 304C	LCDDMA_FB1_BASE	LCD DMA Frame Buffer 1 Base Address Register
0x01E1 3050	LCDDMA_FB1_CEILING	LCD DMA Frame Buffer 1 Ceiling Address Register

PRODUCT PREVIEW

6.22.1 LCD Interface Display Driver (LIDD Mode)

Table 6-77. LCD LIDD Mode Timing Requirements(1)

NO	PARAMETER		MIN	MAX	UNIT
16	$t_{su(LCD\_D)}$	Setup time, LCD_D[15:0] valid before LCD_MCLK ↑	7		ns
17	$t_{h(LCD\_D)}$	Hold time, LCD_D[15:0] valid after LCD_MCLK ↑	0		ns

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

Table 6-78. LCD LIDD Mode Timing Characteristics

NO	PARAMETER		MIN	MAX	UNIT
4	$t_{d(LCD\_D\_V)}$	Delay time, LCD_MCLK ↑ to LCD_D[15:0] valid (write)	0	7	ns
5	$t_{d(LCD\_D\_I)}$	Delay time, LCD_MCLK ↑ to LCD_D[15:0] invalid (write)	0	7	ns
6	$t_{d(LCD\_E\_A)}$	Delay time, LCD_MCLK ↑ to LCD_AC_ENB_CS↓	0	7	ns
7	$t_{d(LCD\_E\_I)}$	Delay time, LCD_MCLK ↑ to LCD_AC_ENB_CS↑	0	7	ns
8	$t_{d(LCD\_A\_A)}$	Delay time, LCD_MCLK ↑ to LCD_VSYNC↓	0	7	ns
9	$t_{d(LCD\_A\_I)}$	Delay time, LCD_MCLK ↑ to LCD_VSYNC↑	0	7	ns

Table 6-78. LCD LIDD Mode Timing Characteristics (continued)

NO	PARAMETER		MIN	MAX	UNIT
10	$t_{d(LCD\_W\_A)}$	Delay time, LCD_MCLK ↑ to LCD_HSYNC↓	0	7	ns
11	$t_{d(LCD\_W\_I)}$	Delay time, LCD_MCLK ↑ to LCD_HSYNC↑	0	7	ns
12	$t_{d(LCD\_STRB\_A)}$	Delay time, LCD_MCLK ↑ to LCD_PCLK↑	0	7	ns
13	$t_{d(LCD\_STRB\_I)}$	Delay time, LCD_MCLK ↑ to LCD_PCLK↓	0	7	ns
14	$t_{d(LCD\_D\_Z)}$	Delay time, LCD_MCLK ↑ to LCD_D[15:0] in 3-state	0	7	ns
15	$t_{d(Z\_LCD\_D)}$	Delay time, LCD_MCLK ↑ to 15 td(Z_LCD_D) 3-state) LCD_D[15:0] (valid from 3-state)	0	7	ns

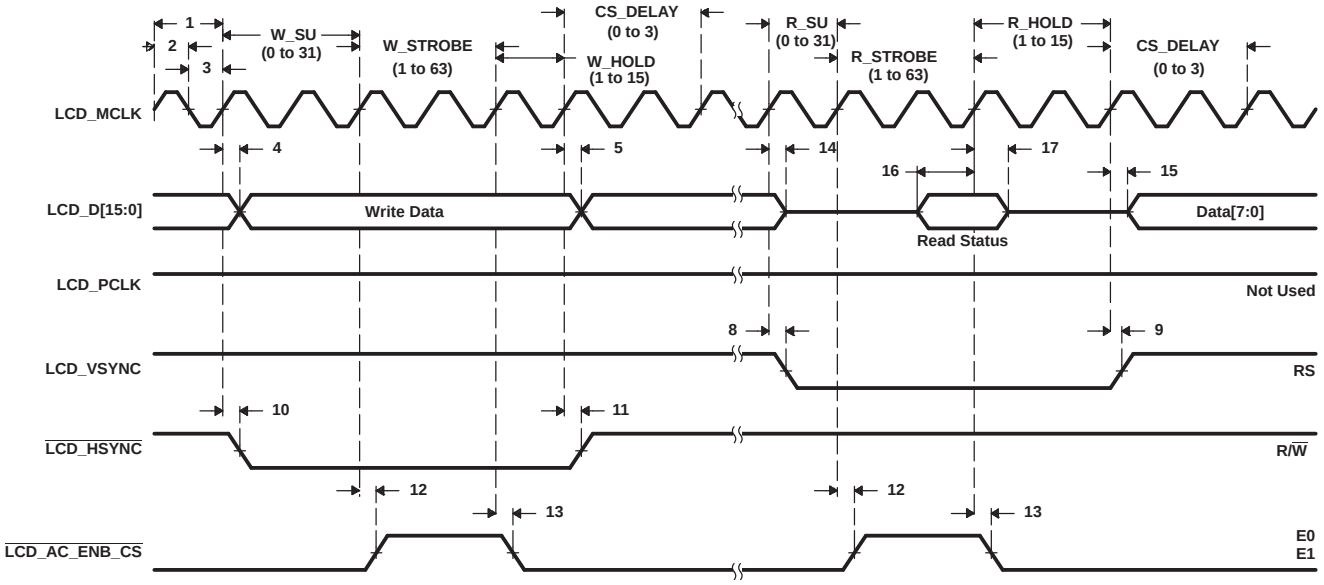


Figure 6-46. Character Display HD44780 Write

PRODUCT PREVIEW

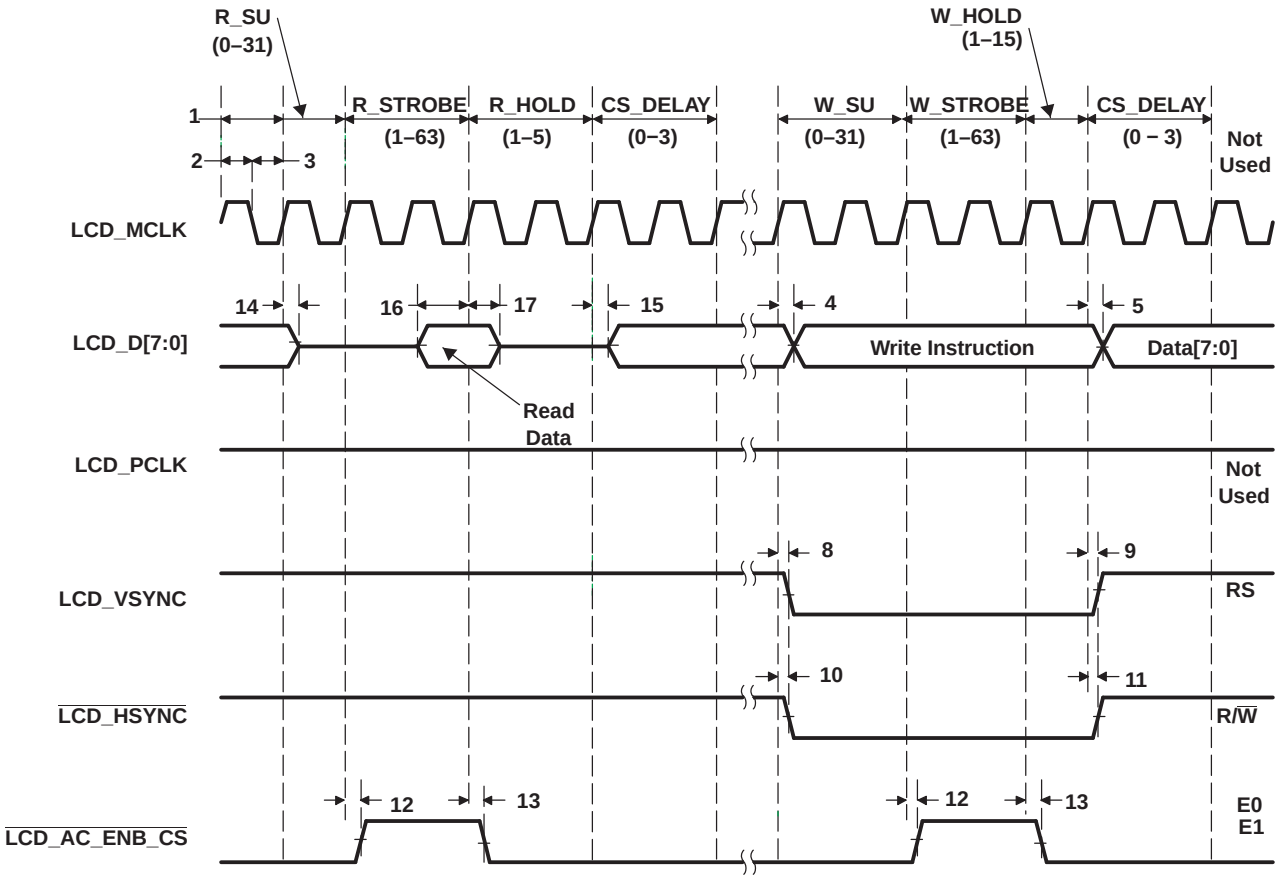


Figure 6-47. Character Display HD44780 Read

PRODUCT PREVIEW

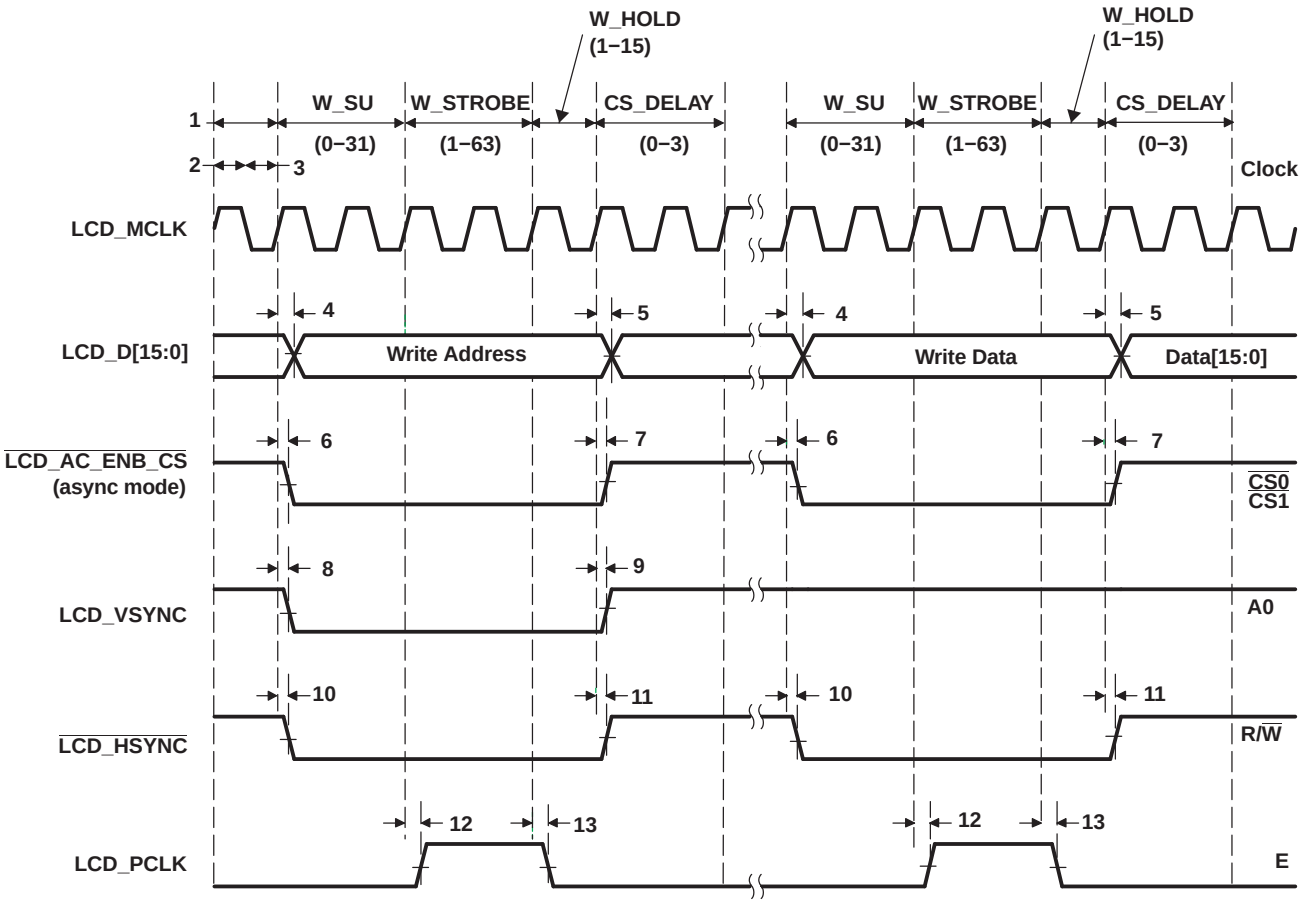


Figure 6-48. Micro-Interface Graphic Display 6800 Write

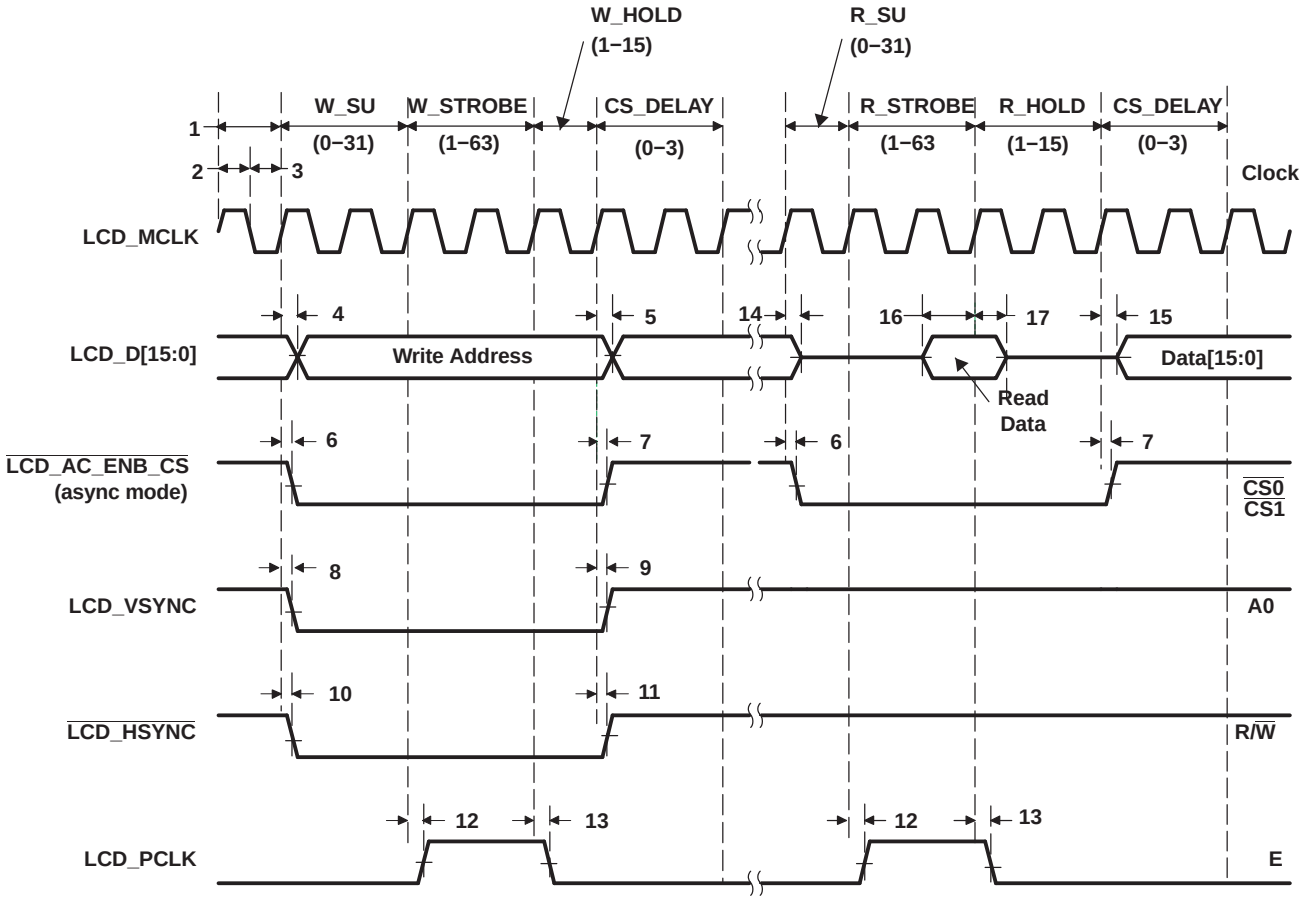


Figure 6-49. Micro-Interface Graphic Display 6800 Read

PRODUCT PREVIEW

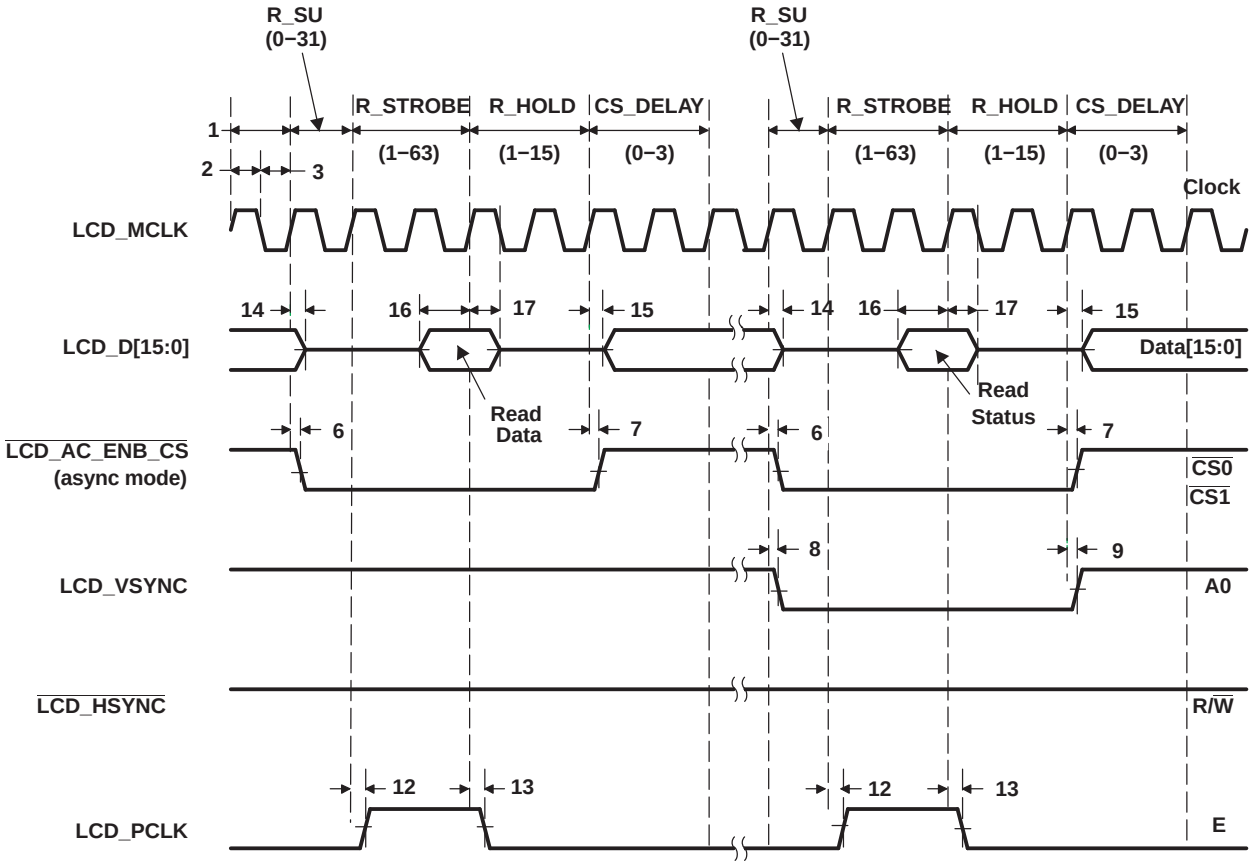


Figure 6-50. Micro-Interface Graphic Display 6800 Status

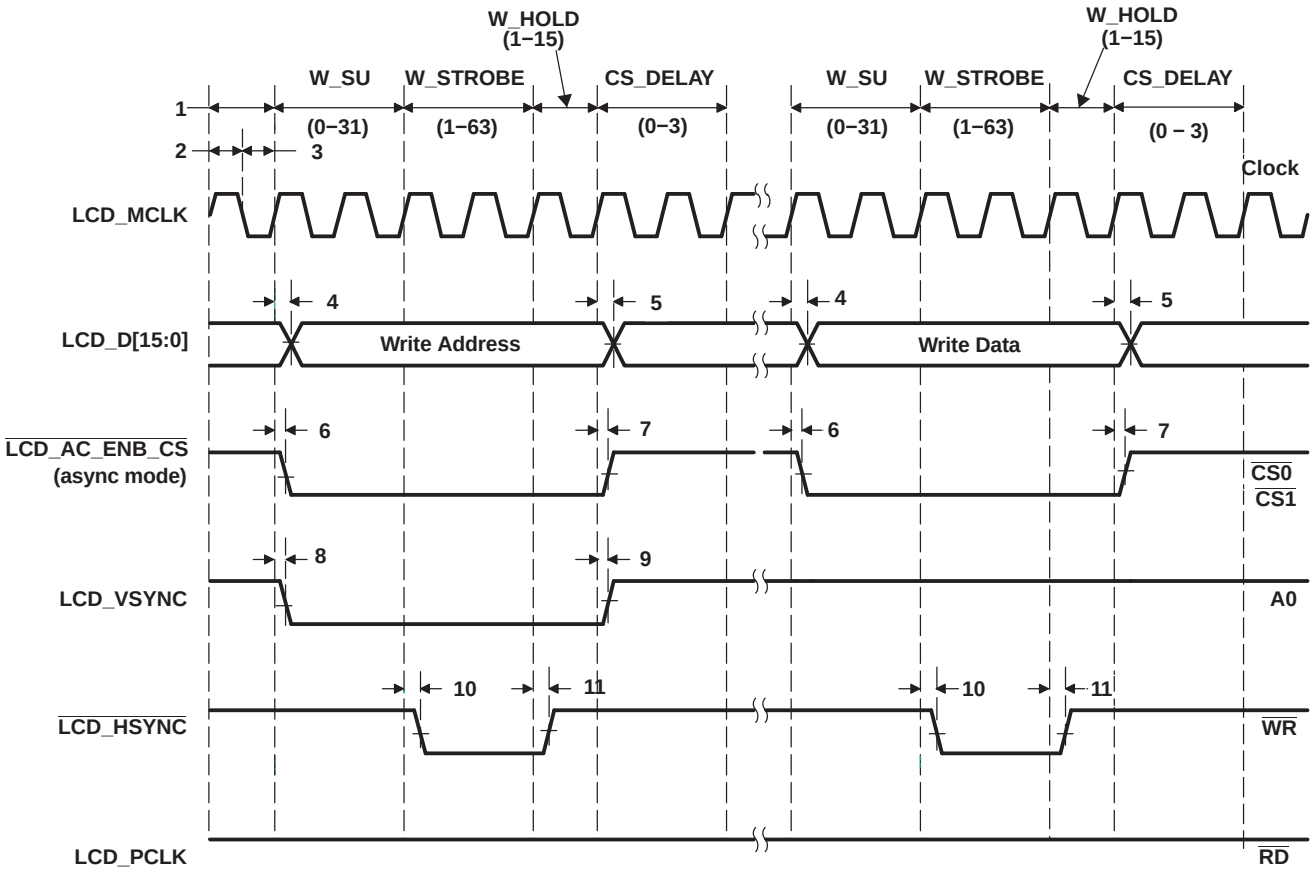


Figure 6-51. Micro-Interface Graphic Display 8080 Write

PRODUCT PREVIEW

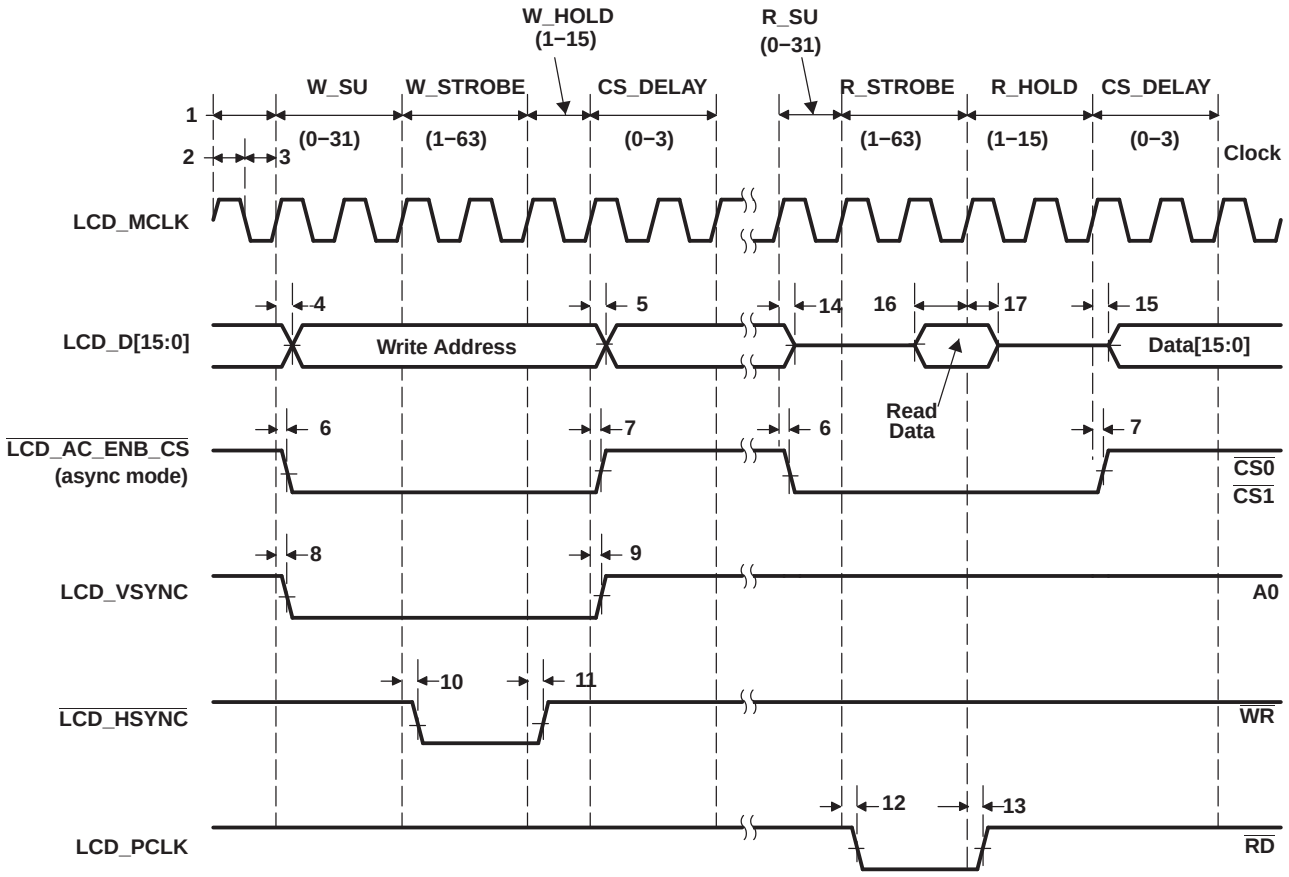


Figure 6-52. Micro-Interface Graphic Display 8080 Read

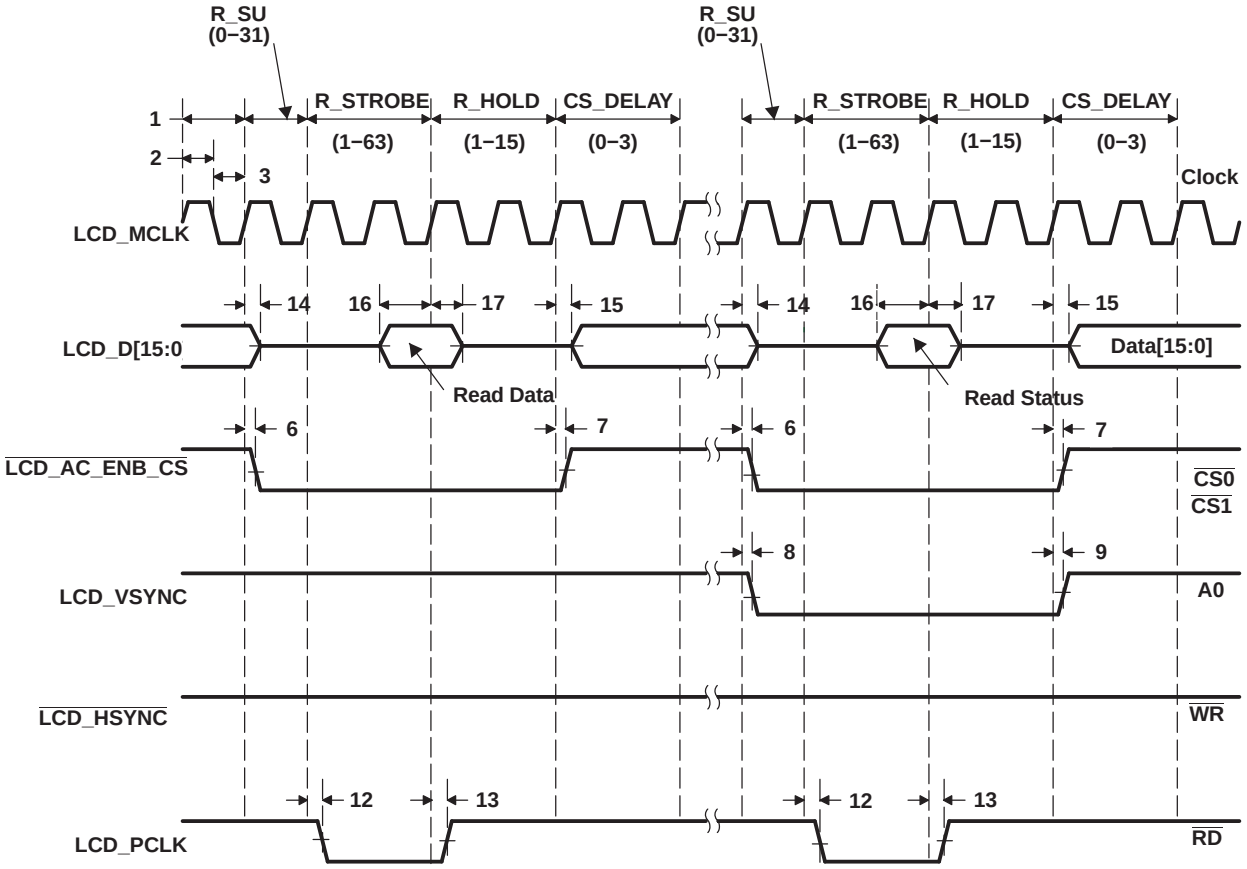


Figure 6-53. Micro-Interface Graphic Display 8080 Status

PRODUCT PREVIEW

6.22.2 LCD Raster Mode

Table 6-79. LCD Raster Mode Timing

See Figure 6-54 through Figure 6-58

NO.	PARAMETER		MIN	MAX	UNIT
	$f_{clock(PIXEL\_CLK)}$	Clock frequency, pixel clock		F/2 (1)	MHz
1	$t_{c(PIXEL\_CLK)}$	Cycle time, pixel clock	23.81		ns
2	$t_{w(PIXEL\_CLK\_H)}$	Pulse duration, pixel clock high	10		ns
3	$t_{w(PIXEL\_CLK\_L)}$	Pulse duration, pixel clock low	10		ns
4	$t_{d(LCD\_D\_V)}$	Delay time, LCD_PCLK↑ to LCD_D[15:0] valid (write)	0	12	ns
5	$t_{d(LCD\_D\_IV)}$	Delay time, LCD_PCLK↑ to LCD_D[15:0] invalid (write)	0	12	ns
6	$t_{d(\overline{LCD\_AC\_ENB\_CS}\_A)}$	Delay time, LCD_PCLK↓ to $\overline{LCD\_AC\_ENB\_CS}$↑	0	12	ns
7	$t_{d(\overline{LCD\_AC\_ENB\_CS}\_I)}$	Delay time, LCD_PCLK↓ to $\overline{LCD\_AC\_ENB\_CS}$↓	0	12	ns
8	$t_{d(LCD\_VSYNC\_A)}$	Delay time, LCD_PCLK↓ to LCD_VSYNC↑	0	12	ns
9	$t_{d(LCD\_VSYNC\_I)}$	Delay time, LCD_PCLK↓ to LCD_VSYNC↓	0	12	ns
10	$t_{d(LCD\_HSYNC\_A)}$	Delay time, LCD_PCLK↑ to LCD_HSYNC↑	0	12	ns
11	$t_{d(LCD\_HSYNC\_I)}$	Delay time, LCD_PCLK↑ to LCD_HSYNC↓	0	12	ns

(1) F = frequency of LCD_PCLK in ns

Frame-to-frame timing is derived through the following parameters in the LCD (RASTER_TIMING_1) register:

- Vertical front porch (VFP)
- Vertical sync pulse width (VSW)
- Vertical back porch (VBP)
- Lines per panel (LPP)

Line-to-line timing is derived through the following parameters in the LCD (RASTER_TIMING_0) register:

- Horizontal front porch (HFP)
- Horizontal sync pulse width (HSW)
- Horizontal back porch (HBP)
- Pixels per panel (PPL)

$\overline{LCD\_AC\_ENB\_CS}$ timing is derived through the following parameter in the LCD (RASTER_TIMING_2) register:

- AC bias frequency (ACB)

The display format produced in raster mode is shown in Figure 6-54. An entire frame is delivered one line at a time. The first line delivered starts at data pixel (1, 1) and ends at data pixel (P, 1). The last line delivered starts at data pixel (1, L) and ends at data pixel (P, L). The beginning of each new frame is denoted by the activation of I/O signal LCD_VSYNC. The beginning of each new line is denoted by the activation of I/O signal LCD_HSYNC.

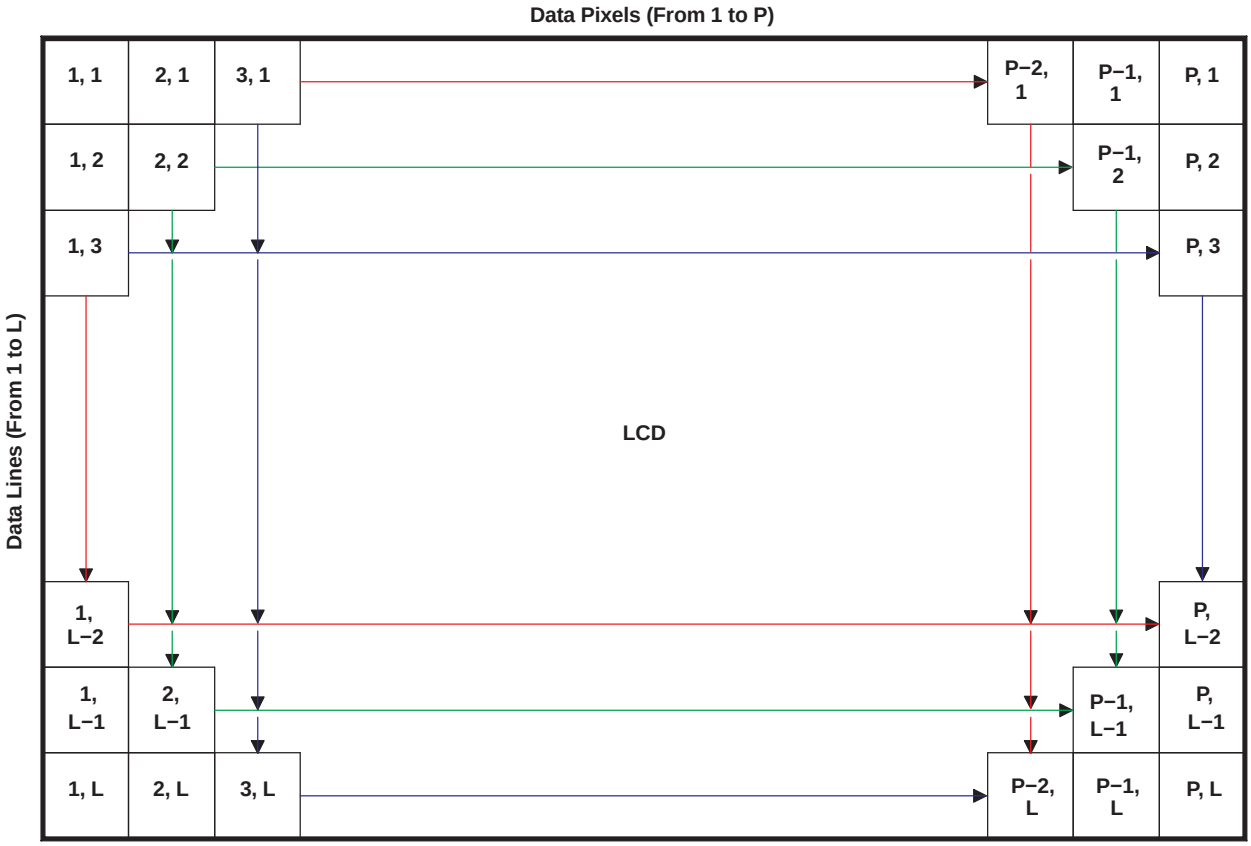


Figure 6-54. LCD Raster-Mode Display Format

PRODUCT PREVIEW

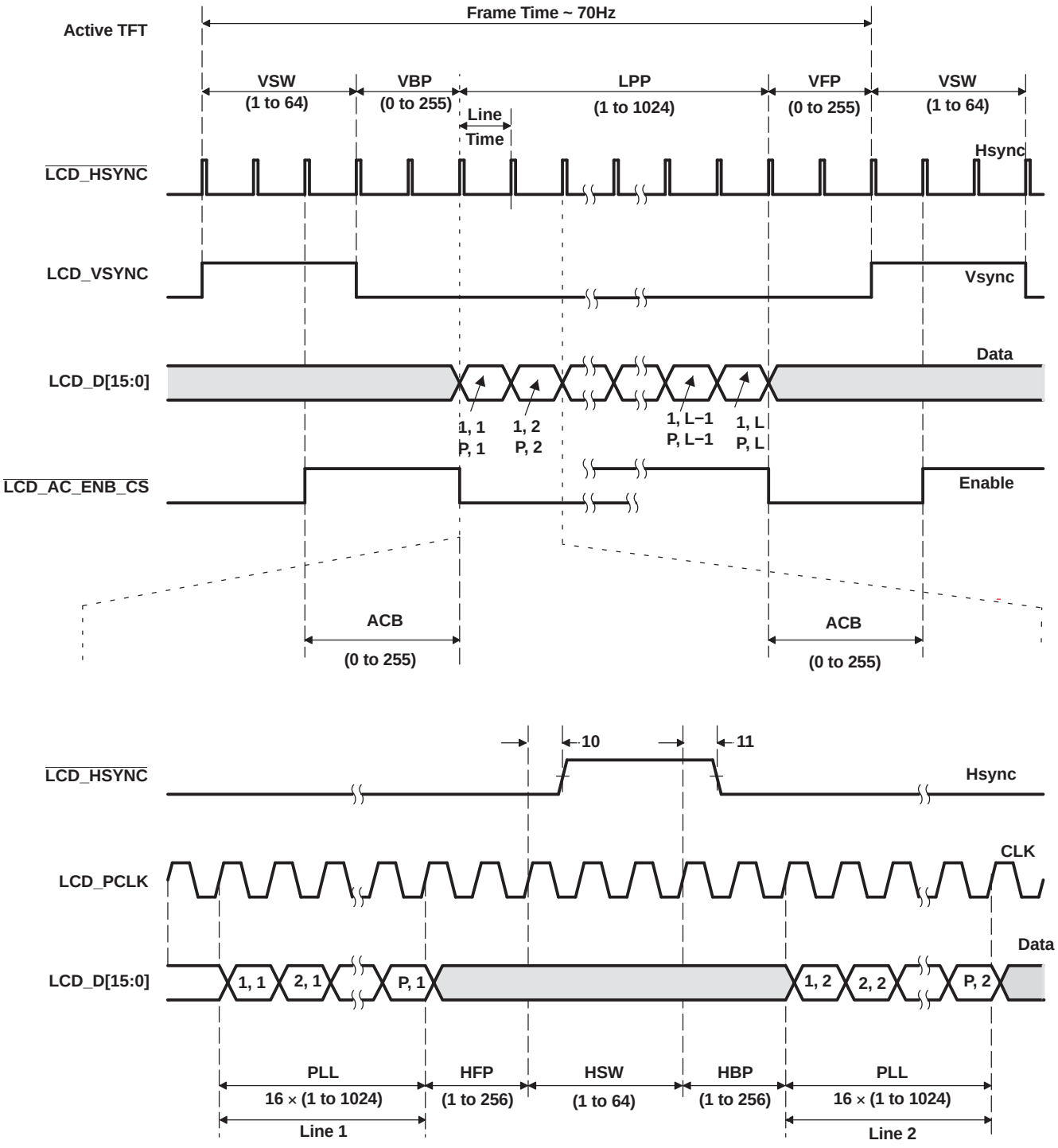


Figure 6-55. LCD Raster-Mode Active

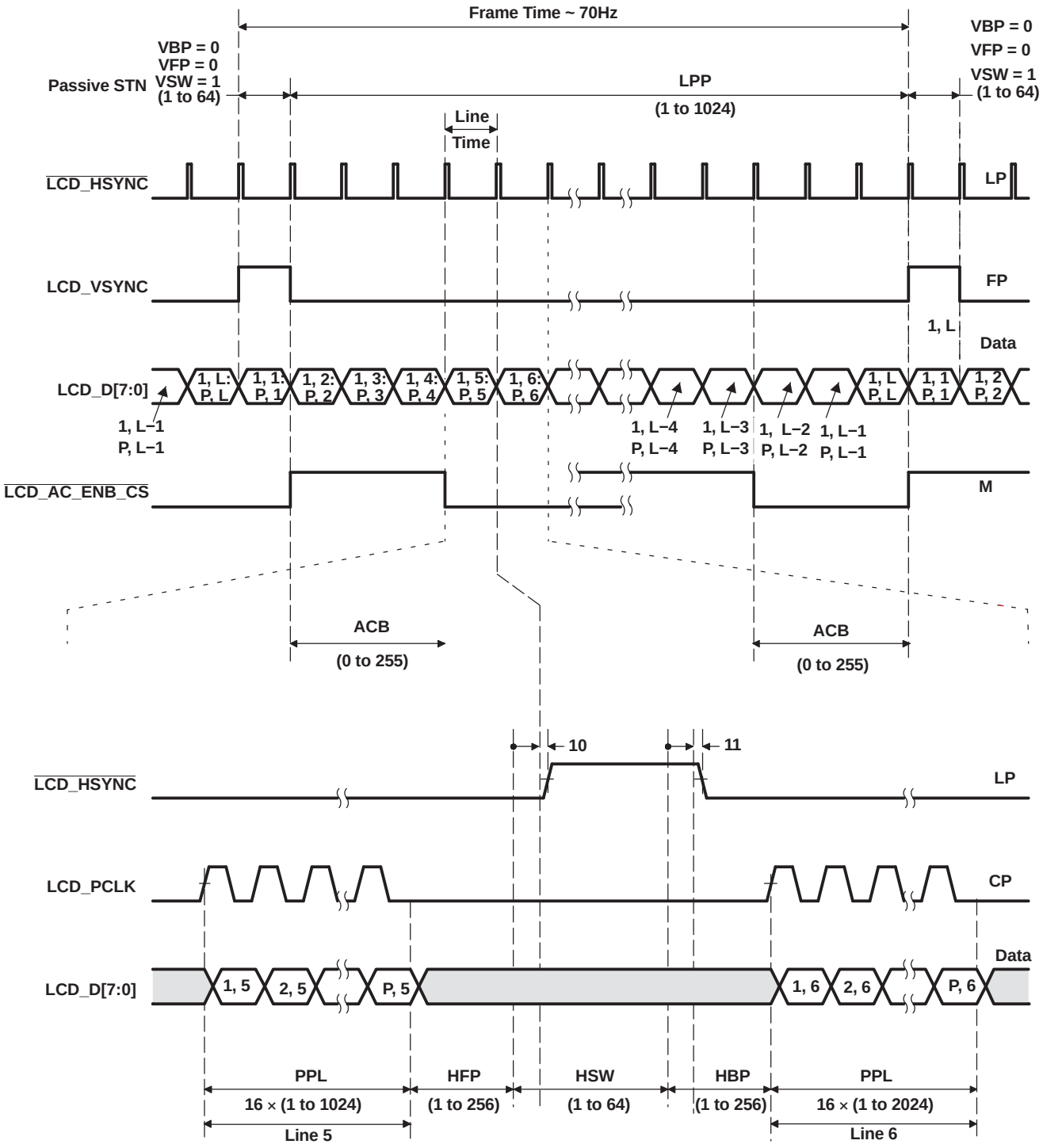


Figure 6-56. LCD Raster-Mode Passive

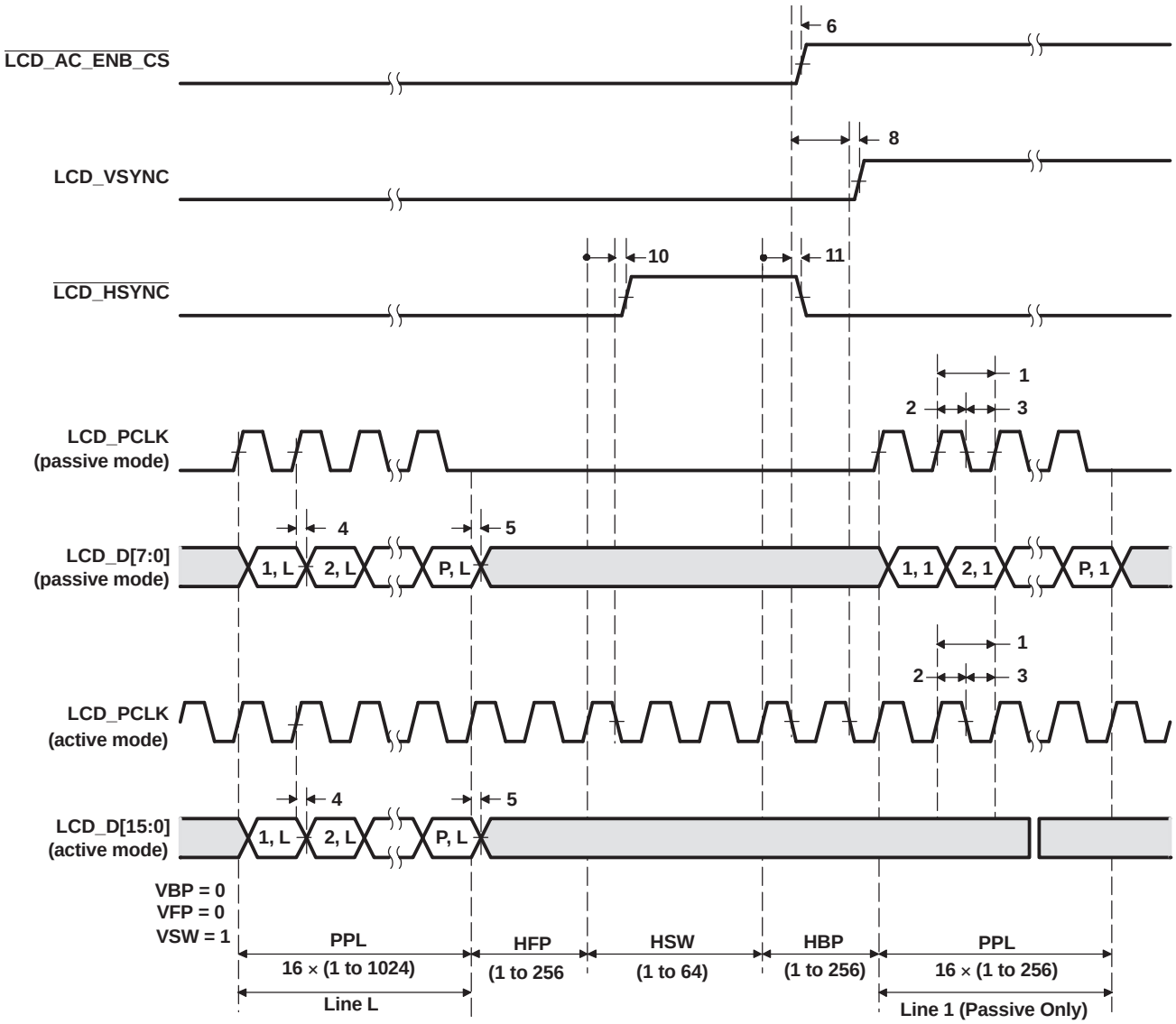


Figure 6-57. LCD Raster-Mode Control Signal Activation

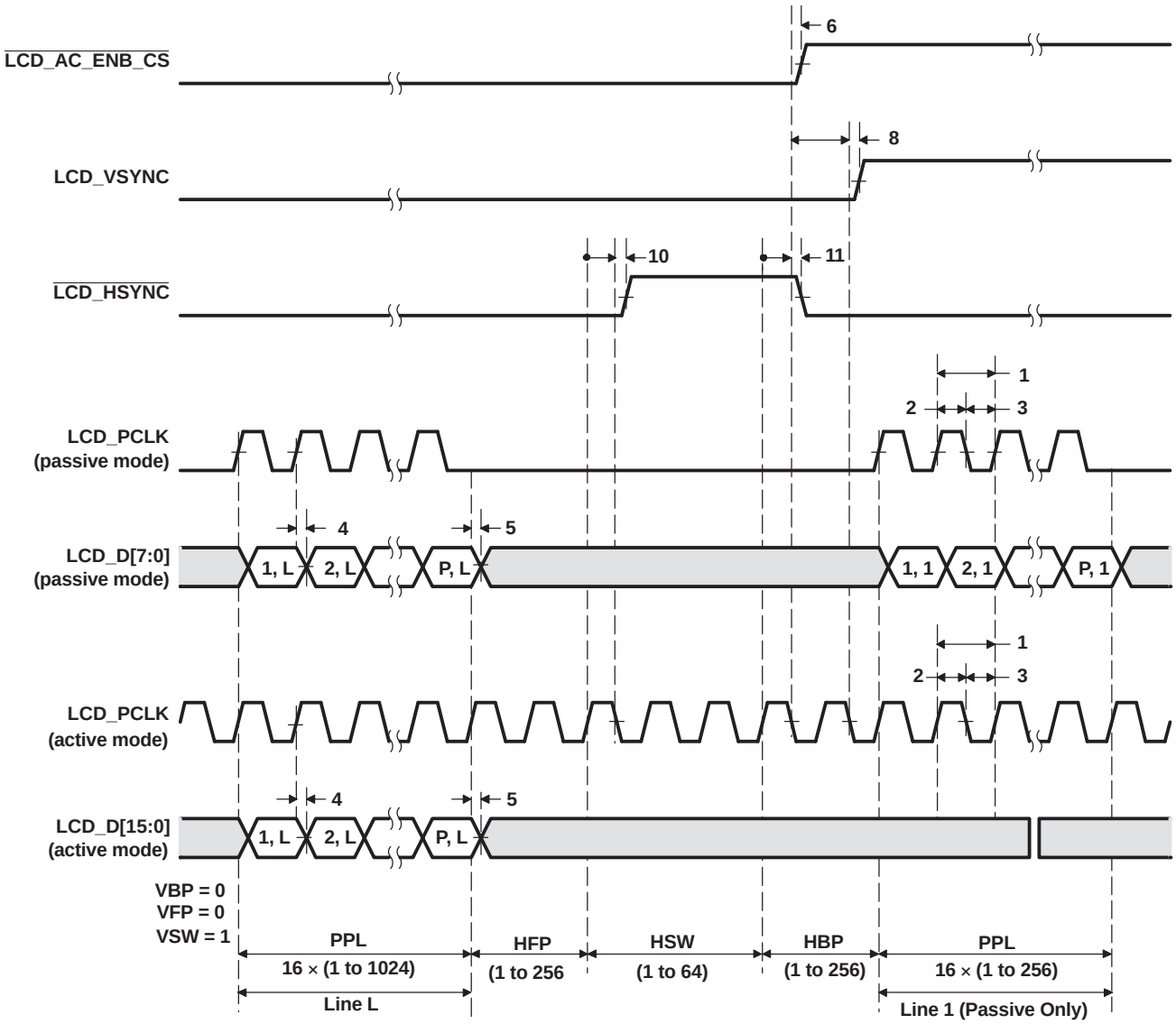


Figure 6-58. LCD Raster-Mode Control Signal Deactivation

6.23 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 6-80 lists the timer registers.

Table 6-80. 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

6.23.1 Timer Electrical Data/Timing

Table 6-81. Timing Requirements for Timer Input(1)(2) (see Figure 6-59)

NO.			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		0.05C	ns

(1) P = OSCIN cycle time in ns. For example, when OSCIN frequency is 27 MHz, use P = 37.0$\overline{37}$ ns.
(2) C = TM64P0_IN12 cycle time in ns. For example, when TM64Px_IN12 frequency is 27 MHz, use C = 37.0$\overline{37}$ ns

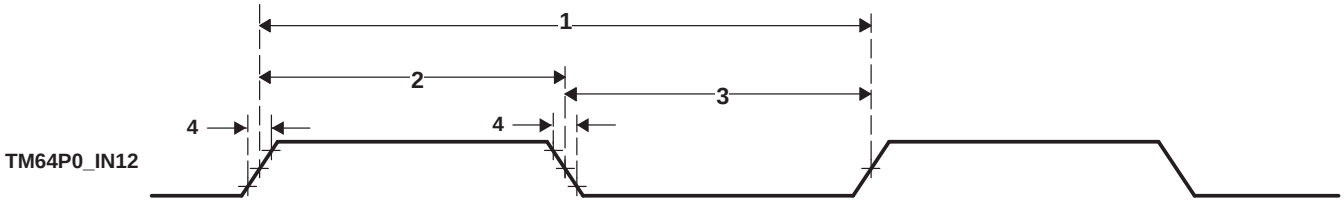


Figure 6-59. Timer Timing

Table 6-82. Switching Characteristics Over Recommended Operating Conditions for Timer Output (1)

NO.			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 = 37.0$\overline{37}$ ns.

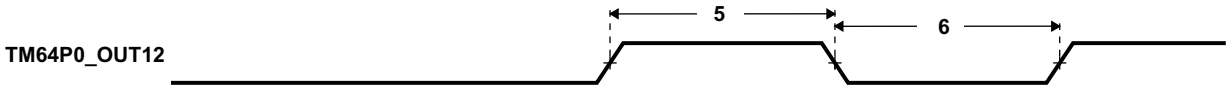


Figure 6-60. Timer Timing

PRODUCT PREVIEW

6.24 Inter-Integrated Circuit Serial Ports (I2C0, I2C1)

6.24.1 I2C Device-Specific Information

Having two I2C modules on the OMAP-L137 simplifies system architecture, since one module may be used by the DSP to control local peripherals ICs (DACs, ADCs, etc.) while the other may be used to communicate with other controllers in a system or to implement a user interface. Figure 6-61 is block diagram of the OMAP-L137 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

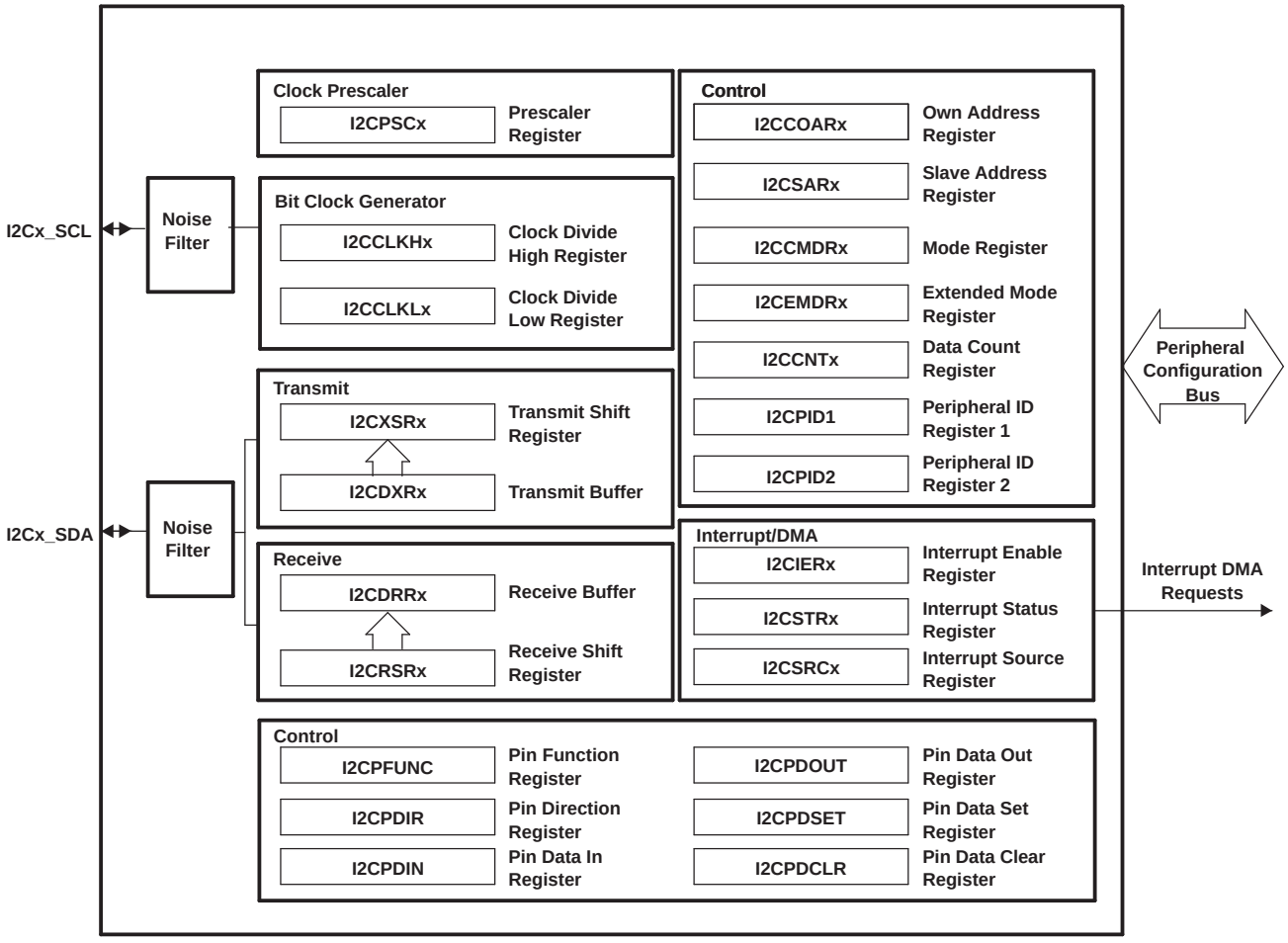


Figure 6-61. I2C Module Block Diagram

6.24.2 I2C Peripheral Registers Description(s)

Table 6-83 is the list of the I2C registers.

Table 6-83. Inter-Integrated Circuit (I2C) Registers

I2C0 BYTE ADDRESS	I2C1 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

PRODUCT PREVIEW

6.24.3 I2C Electrical Data/Timing

6.24.3.1 Inter-Integrated Circuit (I2C) Timing

Table 6-84 and Table 6-85 assume testing over recommended operating conditions (see Figure 6-62 and Figure 6-63).

Table 6-84. I2C Input Timing Requirements

NO.				MIN	MAX	UNIT
1	$t_{c(SCL)}$	Cycle time, I2Cx_SCL	Standard Mode	10		µs
			Fast Mode	2.5		
2	$t_{su(SCLH-SDAL)}$	Setup time, I2Cx_SCL high before I2Cx_SDA low	Standard Mode	4.7		µs
			Fast Mode	0.6		
3	$t_{h(SCLL-SDAL)}$	Hold time, I2Cx_SCL low after I2Cx_SDA low	Standard Mode	4		µs
			Fast Mode	0.6		
4	$t_{w(SCLL)}$	Pulse duration, I2Cx_SCL low	Standard Mode	4.7		µs
			Fast Mode	1.3		
5	$t_{w(SCLH)}$	Pulse duration, I2Cx_SCL high	Standard Mode	4		µs
			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		µs
			Fast Mode	0	0.9	
8	$t_{w(SDAH)}$	Pulse duration, I2Cx_SDA high	Standard Mode	4.7		µs
			Fast Mode	1.3		

Table 6-84. I2C Input Timing Requirements (continued)

NO.				MIN	MAX	UNIT
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		µs
			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 6-85. I2C Switching Characteristics(1)

NO.	PARAMETER			MIN	MAX	UNIT
16	$t_{c(SCL)}$	Cycle time, I2Cx_SCL	Standard Mode	10		µs
			Fast Mode	2.5		
17	$t_{su(SCLH-SDAL)}$	Setup time, I2Cx_SCL high before I2Cx_SDA low	Standard Mode	4.7		µs
			Fast Mode	0.6		
18	$t_{h(SDAL-SCLL)}$	Hold time, I2Cx_SCL low after I2Cx_SDA low	Standard Mode	4		µs
			Fast Mode	0.6		
19	$t_{w(SCLL)}$	Pulse duration, I2Cx_SCL low	Standard Mode	4.7		µs
			Fast Mode	1.3		
20	$t_{w(SCLH)}$	Pulse duration, I2Cx_SCL high	Standard Mode	4		µs
			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		µs
			Fast Mode	0	0.9	
23	$t_{w(SDAH)}$	Pulse duration, I2Cx_SDA high	Standard Mode	4.7		µs
			Fast Mode	1.3		
28	$t_{su(SCLH-SDAH)}$	Setup time, I2Cx_SCL high before I2Cx_SDA high	Standard Mode	4		µs
			Fast Mode	0.6		

(1) I2C must be configured correctly to meet the timings in Table 6-85.

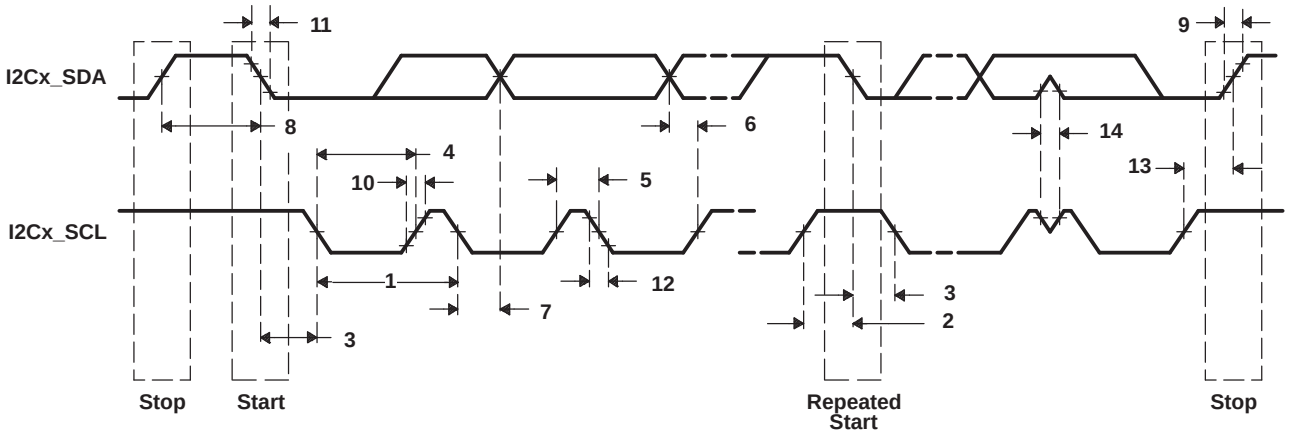


Figure 6-62. I2C Receive Timings

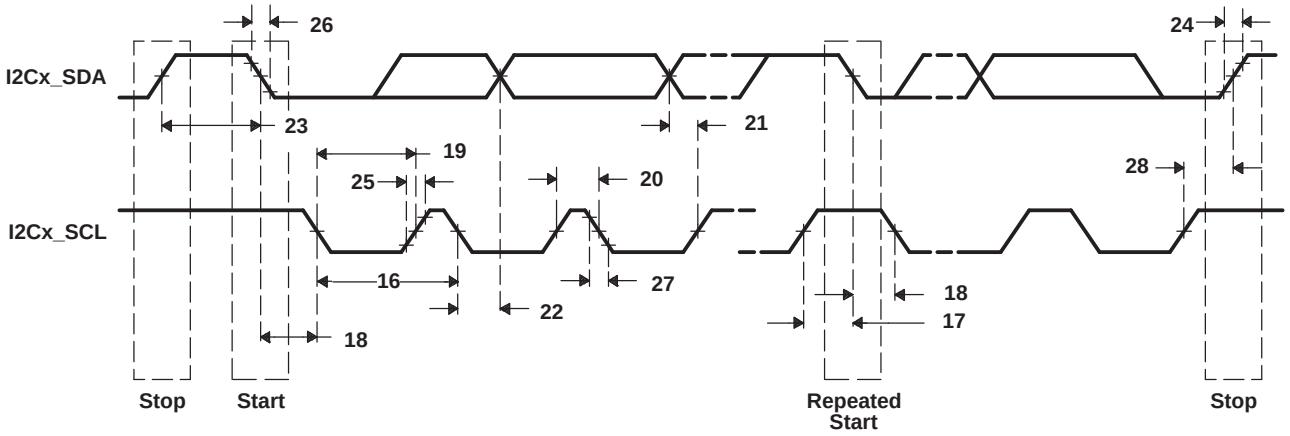


Figure 6-63. I2C Transmit Timings

PRODUCT PREVIEW

6.25 Universal Asynchronous Receiver/Transmitter (UART)

OMAP-L137 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
- 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
- Modem control functions (CTS, RTS) on ***UART0 only***.

The UART registers are listed in Section 6.25.1

6.25.1 UART Peripheral Registers Description(s)

Table 6-86 is the list of UART registers.

Table 6-86. UART Registers

UART0 BYTE ADDRESS	UART1 BYTE ADDRESS	UART2 BYTE ADDRESS	REGISTER NAME	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 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

6.25.2 UART Electrical Data/Timing

Table 6-87. Timing Requirements for UARTx Receive(1) (see Figure 6-64)

NO.			MIN	MAX	UNIT
4	$t_{w(URXDB)}$	Pulse duration, receive data bit (RXDn)	0.96U	1.05U	ns
5	$t_{w(URXSB)}$	Pulse duration, receive start bit	0.96U	1.05U	ns

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

Table 6-88. Switching Characteristics Over Recommended Operating Conditions for UARTx Transmit(1) (see Figure 6-64)

NO.	PARAMETER		MIN	MAX	UNIT
1	$f_{(baud)}$	Maximum programmable baud rate		3	MBaud
2	$t_{w(UTXDB)}$	Pulse duration, transmit data bit (TXDn)	U - 2	U + 2	ns
3	$t_{w(UTXSB)}$	Pulse duration, transmit start bit	U - 2	U + 2	ns

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

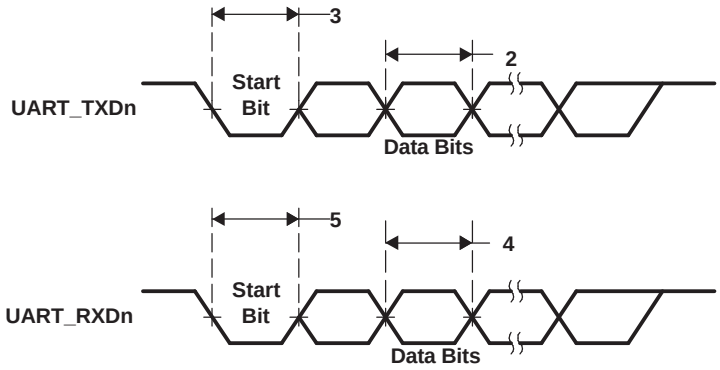


Figure 6-64. UART Transmit/Receive Timing

PRODUCT PREVIEW

6.26 USB1 Host Controller Registers (USB1.1 OHCI)

All OMAP-L137 USB interfaces are compliant with Universal Serial Bus Specifications, Revision 1.1.

Table 6-89 is the list of USB Host Controller registers.

Table 6-89. USB Host Controller Registers

USB BYTE ADDRESS	REGISTER NAME	Register Description
0x01E2 5000	HCREVISION	OHCI Revision Number Register
0x01E2 5004	HCCONTROL	HC Operating Mode Register
0x01E2 5008	HCCOMMANDSTATUS	HC Command and Status Register
0x01E2 500C	HCINTERRUPTSTATUS	HC Interrupt and Status Register
0x01E2 5010	HCINTERRUPTENABLE	HC Interrupt Enable Register
0x01E2 5014	HCINTERRUPTDISABLE	HC Interrupt Disable Register
0x01E2 5018	HCHCCA	HC HCAA Address Register(1)
0x01E2 501C	HCPERIODCURRENTED	HC Current Periodic Register(1)
0x01E2 5020	HCCONTROLHEADED	HC Head Control Register(1)
0x01E2 5024	HCCONTROLCURRENTED	HC Current Control Register(1)
0x01E2 5028	HCBULKHEADED	HC Head Bulk Register(1)
0x01E2 502C	HCBULKCURRENTED	HC Current Bulk Register(1)
0x01E2 5030	HCDONEHEAD	HC Head Done Register(1)
0x01E2 5034	HCFMINTERVAL	HC Frame Interval Register
0x01E2 5038	HCFMREMAINING	HC Frame Remaining Register
0x01E2 503C	HCFMNUMBER	HC Frame Number Register
0x01E2 5040	HCPERIODICSTART	HC Periodic Start Register
0x01E2 5044	HCLSTHRESHOLD	HC Low-Speed Threshold Register
0x01E2 5048	HCRHDESCRIPTORA	HC Root Hub A Register
0x01E2 504C	HCRHDESCRIPTORB	HC Root Hub B Register
0x01E2 5050	HCRHSTATUS	HC Root Hub Status Register
0x01E2 5054	HCRHPORTSTATUS1	HC Port 1 Status and Control Register(2)
0x01E2 5058	HCRHPORTSTATUS2	HC Port 2 Status and Control Register(3)

(1) Restrictions apply to the physical addresses used in these registers.
(2) Connected to the integrated USB1.1 phy pins (USB1_DM, USB1_DP).
(3) Although the controller implements two ports, the second port cannot be used.

Table 6-90. Switching Characteristics Over Recommended Operating Conditions for USB

NO.	PARAMETER		LOW SPEED		FULL SPEED		UNIT
			MIN	MAX	MAX	MAX	
U1	t_r	Rise time, USB.DP and USB.DM signals(1)	75(1)	300(1)	4(1)	20(1)	ns
U2	t_f	Fall time, USB.DP and USB.DM signals(1)	75(1)	300(1)	4(1)	20(1)	ns
U3	t_{RFM}	Rise/Fall time matching(2)	80(2)	120(2)	90(2)	110(2)	%
U4	V_{CRS}	Output signal cross-over voltage(1)	1.3(1)	2(1)	1.3(1)	2(1)	V
U5	t_j	Differential propagation jitter(3)	-25(3)	25(3)	-2(3)	2(3)	ns
U6	f_{op}	Operating frequency(4)		1.5		12	MHz

(1) Low Speed: C_L = 200 pF. High Speed: C_L = 50pF
(2) t_{RFM} =(t_r/t_f) x 100
(3) $t_{jr} = t_{px(1)} - t_{px(0)}$
(4) $f_{op} = 1/t_{per}$

6.27 USB0 OTG (USB2.0 OTG)

The OMAP-L137 USB2.0 peripheral supports the following features:

- USB 2.0 peripheral at speeds high speed (HS: 480 Mb/s) and full speed (FS: 12 Mb/s)
- USB 2.0 host at speeds HS, 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 High Speed 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

Table 6-91 is the list of USB OTG registers.

Table 6-91. 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	INTVECTR	USB Interrupt Vector Register
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
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

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
Indexed Registers 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)
FIFO		
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
OTG Device Control		
0x01E0 0460	DEVCTL	Device Control Register
Dynamic FIFO Control		
0x01E0 0462	TXFIFOSZ	Transmit Endpoint FIFO Size (Index register set to select Endpoints 1-4 only)
0x01E0 0463	RXFIFOSZ	Receive Endpoint FIFO Size (Index register set to select Endpoints 1-4 only)
0x01E0 0464	TXFIFOADDR	Transmit Endpoint FIFO Address (Index register set to select Endpoints 1-4 only)
0x01E0 0464	HWVERS	Hardware Version Register

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
0x01E0 0466	RXFIFOADDR	Receive Endpoint FIFO Address (Index register set to select Endpoints 1-4 only)
Target Endpoint 0 Control Registers, Valid Only in Host Mode		
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.
Target Endpoint 1 Control Registers, Valid Only in Host Mode		
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.
Target Endpoint 2 Control Registers, Valid Only in Host Mode		
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.
Target Endpoint 3 Control Registers, Valid Only in Host Mode		
0x01E0 0498	TXFUNCADDR	Address of the target function that has to be accessed through the associated Transmit Endpoint.

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
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.
Target Endpoint 4 Control Registers, Valid Only in Host Mode		
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.
Control and Status Register for Endpoint 0		
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.
Control and Status Register for Endpoint 1		
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.

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
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.
Control and Status Register for Endpoint 2		
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.
Control and Status Register for Endpoint 3		
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.
Control and Status Register for Endpoint 4		
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

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
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.
DMA Registers		
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
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 2C00	DMA_SCHED_CTRL	DMA Scheduler Control Register
0x01E0 2D00	ENTRY[0]	DMA Scheduler Table Word 0
0x01E0 2D04	ENTRY[1]	DMA Scheduler Table Word 1
. . .	. . .	. . .
0x01E0 2DFC	ENTRY[63]	DMA Scheduler Table Word 63
Queue Manager Registers		
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

Table 6-91. Universal Serial Bus OTG (USB0) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
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 5070	QMEMRBASE[7]	Memory Region 7 Base Address Register
0x01E0 5074	QMEMRCTRL[7]	Memory Region 7 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

PRODUCT PREVIEW

6.27.1 USB2.0 Electrical Data/Timing

Table 6-92. Switching Characteristics Over Recommended Operating Conditions for USB2.0 (see Figure 6-65)

NO.	PARAMETER		LOW SPEED 1.5 Mbps		FULL SPEED 12 Mbps		HIGH SPEED 480 Mbps		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
1	$t_{r(D)}$	Rise time, USB_DP and USB_DM signals(1)	75	300	4	20	0.5		ns
2	$t_{f(D)}$	Fall time, USB_DP and USB_DM signals(1)	75	300	4	20	0.5		ns
3	t_{rfM}	Rise/Fall time, matching(2)	80	120	90	111	–	–	%
4	V_{CRS}	Output signal cross-over voltage(1)	1.3	2	1.3	2	–	–	V
5	$t_{jr(source)NT}$	Source (Host) Driver jitter, next transition		2		2			(3)ns
	$t_{jr(FUNC)NT}$	Function Driver jitter, next transition		25		2		(3)	ns
6	$t_{jr(source)PT}$	Source (Host) Driver jitter, paired transition(4)		1		1		(3)	ns
	$t_{jr(FUNC)PT}$	Function Driver jitter, paired transition		10		1		(3)	ns
7	$t_{w(EOPT)}$	Pulse duration, EOP transmitter	1250	1500	160	175	–	–	ns
8	$t_{w(EOPR)}$	Pulse duration, EOP receiver	670		82		–		ns
9	$t_{(DRATE)}$	Data Rate		1.5		12		480	Mb/s
10	Z_{DRV}	Driver Output Resistance	–	–	40.5	49.5	40.5	49.5	Ω
11	Z_{INP}	Receiver Input Impedance	100k		100k		-	-	Ω

(1) Low Speed: C_L = 200 pF, Full Speed: C_L = 50 pF, High Speed: C_L = 50 pF
(2) $t_{RFM} = (t_r/t_f) \times 100$. [Excluding the first transaction from the Idle state.]
(3) For more detailed information, see the Universal Serial Bus Specification Revision 2.0, Chapter 7. Electrical.
(4) $t_{jr} = t_{px(1)} - t_{px(0)}$

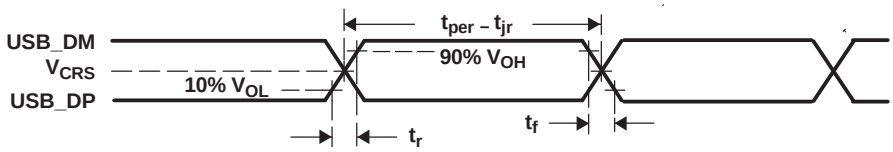


Figure 6-65. USB2.0 Integrated Transceiver Interface Timing

6.32 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 emulation features: power, clock and reset

Table 6-100. Power and Sleep Controller (PSC) Registers

PSC0	PSC1	Register	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) Module Error Pending Register 0 (module 0-31) (PSC1)
0x01C1 0050	0x01E2 7050	MERRCR0	Module Error Clear Register 0 (module 0-15) (PSC0) 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 - 0x01C1 083C	0x01E2 7800 - 0x01E2 787C	MDSTAT0-MDSTAT15 MDSTAT0-MDSTAT31	Module Status *n* Register (modules 0-15) (PSC0) Module Status *n* Register (modules 0-31) (PSC1)
0x01C1 0A00 - 0x01C1 0A3C	0x01E2 7A00 - 0x01E2 7A7C	MDCTL0-MDCTL15 MDSTAT0-MDSTAT31	Module Control *n* Register (modules 0-15) (PSC0) Module Status *n* Register (modules 0-31) (PSC1)

PRODUCT PREVIEW

6.32.1 Power Domain and Module Topology

The SoC includes two PSC modules.

Each PSC module controls clock states for several on the on chip modules, controllers and interconnect components. Table 6-101 and Table 6-102 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 6.32.1.2.

Table 6-101. 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	-	-	-	-
14	ARM	AlwaysON (PD0)	SwRstDisable	—
15	DSP	PD_DSP (PD1)	Enable	—

Table 6-102. 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	USB1 (USB1.1)	AlwaysON (PD0)	SwRstDisable	—
3	GPIO	AlwaysON (PD0)	SwRstDisable	—
4	UHPI	AlwaysON (PD0)	SwRstDisable	—
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	McASP2(+ McASP2 FIFO)	AlwaysON (PD0)	SwRstDisable	—
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	LCDC	AlwaysON (PD0)	SwRstDisable	—

Table 6-102. PSC1 Default Module Configuration (continued)

LPSC Number	Module Name	Power Domain	Default Module State	Auto Sleep/Wake Only
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	Shared RAM (Br 13)	PD_SHRAM	Enable	Yes

6.32.1.1 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 SoC , 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 PSC0 PD1/PD_DSP Domain: Controls the sleep state for DSP L1 and L2 Memories
- On PSC1 PD1/PD_SHRAM Domain: Controls the sleep state for the 128K Shared RAM

6.32.1.2 Module States

The PSC defines several possible states for a module. This 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 6-103.

Table 6-103. 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 SoC 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.

Table 6-103. Module States (continued)

Module State	Module Reset	Module Clock	Module State Definition
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.

6.34 Emulation Logic

This section describes the steps to use a third party debugger on the ARM926EJ-S within the OMAP-L137. The debug capabilities and features for DSP and ARM are as shown below.

DSP:

- Basic Debug
  - Execution Control
  - System Visibility
- Real-Time Debug
  - Interrupts serviced while halted
  - Low/non-intrusive system visibility while running
- Advanced Debug
  - Global Start
  - Global Stop
  - Specify targeted memory level(s) during memory accesses
  - HSRTDX (High Speed Real Time Data eXchange)
- Advanced System Control
  - Subsystem reset via debug
  - Peripheral notification of debug events
  - Cache-coherent debug accesses
- Security
  - Configurable levels of security and debug visibility
  - Halting on a security violation
  - Debug halts prevented during secure code execution
  - Memory accesses prevented to secure memory
- Analysis Actions
  - Stop program execution
  - 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 6-105. DSP Debug Features

Category	Hardware Feature	Availability
Basic Debug	Software breakpoint	Unlimited
	Hardware breakpoint	Up to 10 HWBPs, including: 4 precise HWBPs inside DSP core and one of them is associated with a counter. 2 imprecise HWBPs from AET. 4 imprecise HWBPs from AET which are shared for watch point.
Analysis	Watch point	Up to 4 watch points, which are shared with HWBPs, and can also be used as 2 watch points with data (32 bits)
	Watch point with Data	Up to 2, Which can also be used as 4 watch points.
	Counters/timers	1x64-bits (cycle only) + 2x32-bits (water marke counters)
	External Event Trigger In	2
	External Event Trigger Out	2

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
- Security
  - Halting on a security violation (by cross-triggering via INTC)
  - Memory accesses prevented to secure memory (this is ensured by system level security mechanism)
- 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 6-106. ARM Debug Features

Category	Hardware Feature	Availability
Basic Debug	Software breakpoint	Unlimited
	Hardware breakpoint	Up to 14 HWBPs, including: 2 precise HWBP inside ARM core which are shared with watch points. 8 imprecise HWBPs from ETM's address comparators, which are shared with trace function, and can be used as watch point too. 4 imprecise HWBPs from ICECrusher.
Analysis	Watch point	Up to 6 watch points, including: 2 from ARM core which is shared with HWBPs and can be associated with a data. 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. 8 watch points from ETM can be associated with a data comparator, and ETM of Primus has total 4 data comparators.
	Counters/timers	3x32-bit (1 cycle ; 2 event)
	External Event Trigger In	2
	External Event Trigger Out	2
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	Primus has 4k bytes ETB
	Capture depth PC + Timing	Primus has 4k bytes ETB
	Application accessible	Y

6.34.1 JTAG Port Description

The OMAP-L137 target debug interface uses the five standard IEEE 1149.1(JTAG) signals ($\overline{\text{TRST}}$, TCK, TMS, TDI, and TDO), a return clock (RTCK) due to the clocking requirements of the ARM926EJ-S and EMU0.

Table 6-107. 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 OMAP-L137 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 OMAP-L137 , 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 OMAP-L137 , 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
EMU0	I/O	Emulation 0	Channel 0 trigger + HSRTDX

6.34.2 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.

6.34.2.1 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.

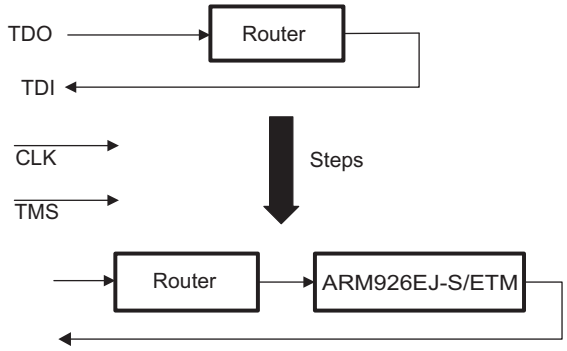


Figure 6-67. Adding ARM926EJ-S to the scan chain

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

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

- Function : Update the JTAG preamble and post-amble counts.
  - Parameter : The IR pre-amble count is '0'.
  - Parameter : The IR post-amble count is '0'.
  - Parameter : The DR pre-amble count is '0'.
  - Parameter : The DR post-amble 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 '0xa3002108'.
  - 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'.
  - Parameter : The DR pre-amble count is '0'.
  - Parameter : The DR post-amble 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.

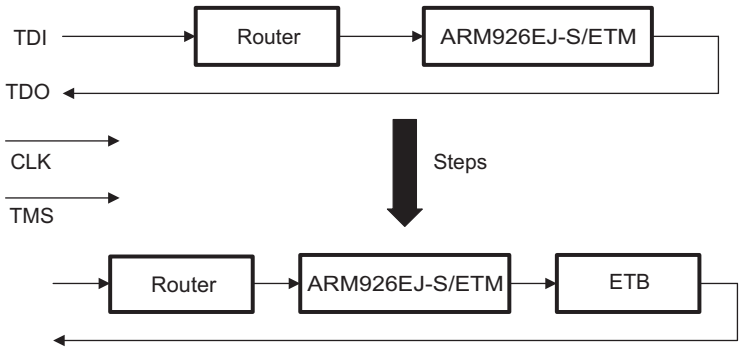


Figure 6-68. Adding ETB to the scan chain

- 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 '0xa4302108'.
  - 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'.

6.35 Real Time Clock (RTC)

The RTC provides a time reference to an application running on the device. The current date and time is tracked in a set of counter registers that update once per second. The time can be represented in 12-hour or 24-hour mode. The calendar and time registers are buffered during reads and writes so that updates do not interfere with the accuracy of the time and date.

Alarms are available to interrupt the CPU at a particular time, or at periodic time intervals, such as once per minute or once per day. In addition, the RTC can interrupt the CPU every time the calendar and time registers are updated, or at programmable periodic intervals.

The real-time clock (RTC) provides the following features:

- 100-year calendar (xx00 to xx99)
- Counts seconds, minutes, hours, day of the week, date, month, and year with leap year compensation
- Binary-coded-decimal (BCD) representation of time, calendar, and alarm
- 12-hour clock mode (with AM and PM) or 24-hour clock mode
- Alarm interrupt
- Periodic interrupt
- Single interrupt to the CPU
- Supports external 32.768-kHz crystal or external clock source of the same frequency
- Separate isolated power supply

Figure 6-69 shows a block diagram of the RTC.

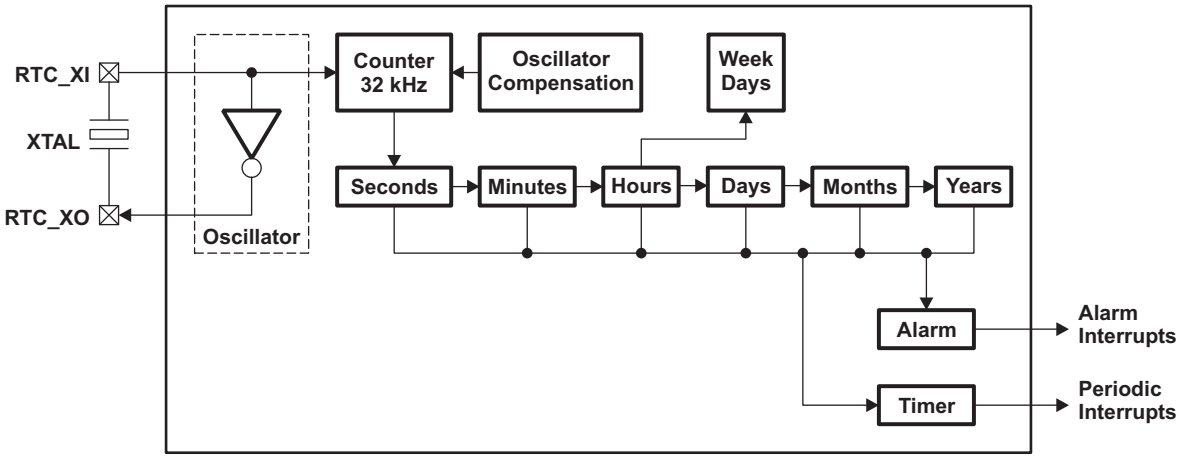


Figure 6-69. Real-Time Clock Block Diagram

6.35.1 Clock Source

The clock reference for the RTC is an external 32.768-kHz crystal or an external clock source of the same frequency. The RTC also has a separate power supply that is isolated from the rest of the system. When the CPU and other peripherals are without power, the RTC can remain powered to preserve the current time and calendar information.

The source for the RTC reference clock may be provided by a crystal or by an external clock source. The RTC has an internal oscillator buffer to support direct operation with a crystal. The crystal is connected between pins RTC_XI and RTC_XO. RTC_XI is the input to the on-chip oscillator and RTC_XO is the output from the oscillator back to the crystal.

An external 32.768-kHz clock source may be used instead of a crystal. In such a case, the clock source is connected to RTC_XI, and RTC_XO is left unconnected.

If the RTC is not used, the RTC_XI pin should be held low and RTC_XO should be left unconnected.

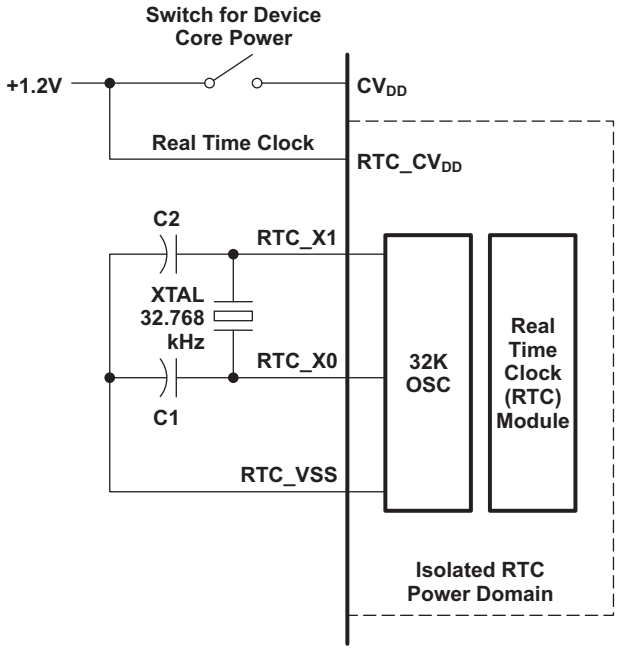


Figure 6-70. Clock Source

6.35.2 Registers

Table 6-108 lists the memory-mapped registers for the RTC. See the device-specific data manual for the memory address of these registers.

Table 6-108. Real-Time Clock (RTC) Registers

BYTE ADDRESS	Acronym	Register Description
0x01C2 3000	SECOND	Seconds Register
0x01C2 3004	MINUTE	Minutes Register
0x01C2 3008	HOUR	Hours Register
0x01C2 300C	DAY	Day of the Month Register
0x01C2 3010	MONTH	Month Register
0x01C2 3014	YEAR	Year Register
0x01C2 3018	DOTW	Day of the Week Register
0x01C2 3020	ALARMSECOND	Alarm Seconds Register

Table 6-108. Real-Time Clock (RTC) Registers (continued)

BYTE ADDRESS	Acronym	Register Description
0x01C2 3024	ALARMMINUTE	Alarm Minutes Register
0x01C2 3028	ALARMHOUR	Alarm Hours Register
0x01C2 302C	ALARMDAY	Alarm Days Register
0x01C2 3030	ALARMMONTH	Alarm Months Register
0x01C2 3034	ALARMYEAR	Alarm Years Register
0x01C2 3040	CTRL	Control Register
0x01C2 3044	STATUS	Status Register
0x01C2 3048	INTERRUPT	Interrupt Enable Register
0x01C2 304C	COMPLSB	Compensation (LSB) Register
0x01C2 3050	COMPMSB	Compensation (MSB) Register
0x01C2 3054	OSC	Oscillator Register
0x01C2 3060	SCRATCH0	Scratch 0 (General-Purpose) Register
0x01C2 3064	SCRATCH1	Scratch 1 (General-Purpose) Register
0x01C2 3068	SCRATCH2	Scratch 2 (General-Purpose) Register
0x01C2 306C	KICK0	Kick 0 (Write Protect) Register
0x01C2 3070	KICK1	Kick 1 (Write Protect) Register

7 Mechanical Packaging and Orderable Information

This section describes the OMAP-L137 orderable part numbers, packaging options, materials, thermal and mechanical parameters.

7.1 Thermal Data for ZKB

The following table(s) show the thermal resistance characteristics for the PBGA–ZKB mechanical package.

Table 7-1. Thermal Resistance Characteristics (PBGA Package) [ZKB]

NO.			°C/W(1)	°C/W(2)	AIR FLOW (m/s)(3)
1	$R\Theta_{JC}$	Junction-to-case	12.8	13.5	N/A
2	$R\Theta_{JB}$	Junction-to-board	15.1	19.7	N/A
3	$R\Theta_{JA}$	Junction-to-free air	24.5	33.8	0.00
4	$R\Theta_{JMA}$	Junction-to-moving air	21.9	30	0.50
5			21.1	28.7	1.00
6			20.4	27.4	2.00
7			19.6	26	4.00
8	Psi_{JT}	Junction-to-package top	0.6	0.8	0.00
9			0.8	1	0.50
10			0.9	1.2	1.00
11			1.1	1.4	2.00
12			1.3	1.8	4.00
13	Psi_{JB}	Junction-to-board	14.9	19.1	0.00
14			14.4	18.2	0.50
15			14.4	18	1.00
16			14.3	17.7	2.00
17			14.1	17.4	4.00

(1) These measurements were conducted in a JEDEC defined 2S2P system and 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 Leaded Surface Mount Packages*. Power dissipation of 1W and ambient temp of 70C assumed. PCB with 2oz (70um) top and bottom copper thickness and 1.5oz (50um) inner copper thickness

(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) m/s = meters per second

7.2 Mechanical Drawings

This section contains mechanical drawings for the ZKB Ball Grid Array package .

PACKAGING INFORMATION

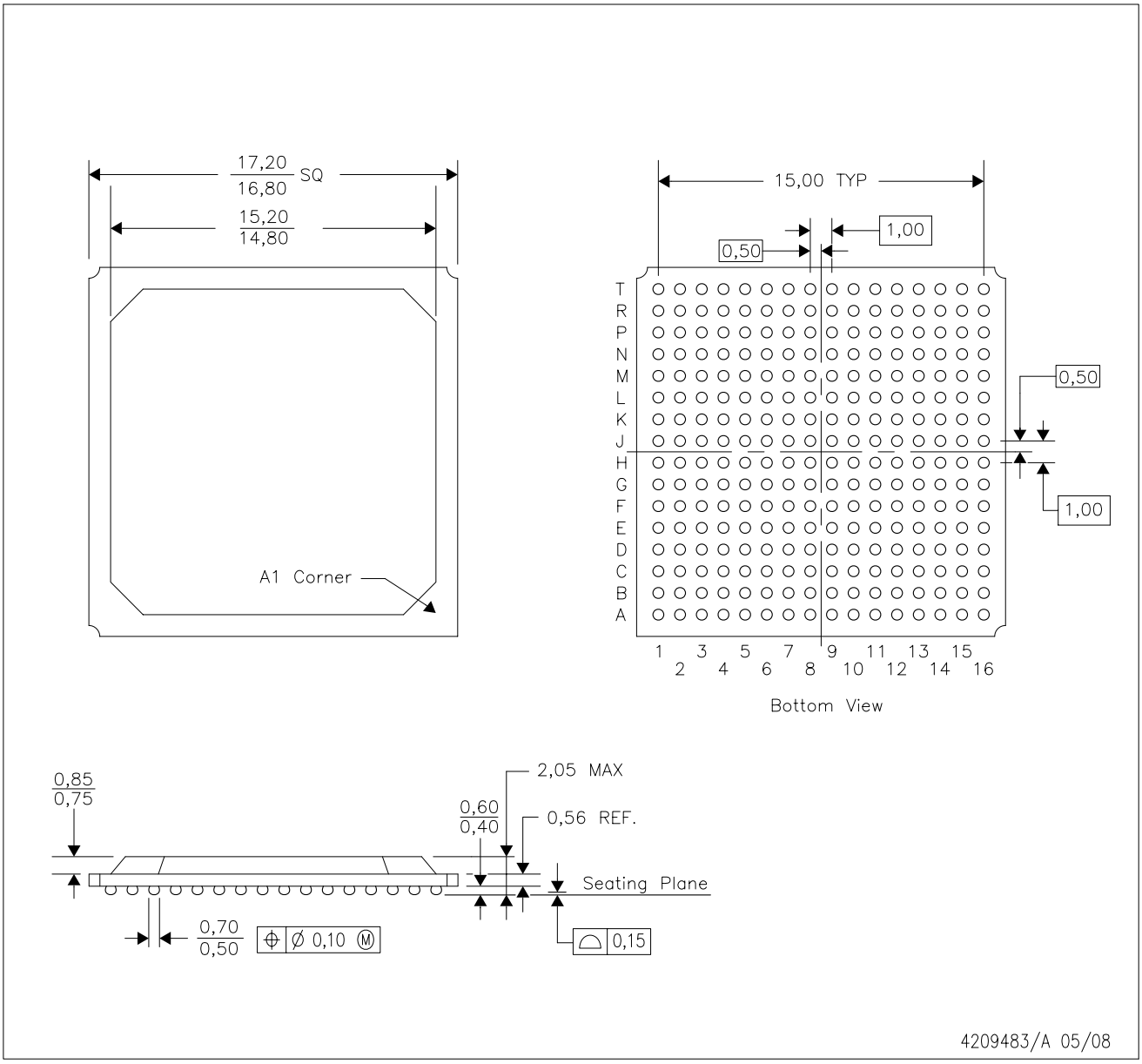
Orderable Device	Status [1]	Package Type	Package Drawing	Pins	Package Qty	Eco Plan [2]	Lead/Ball Finish	MSL Peak Temp [3]
OMAPL137ZKB3	PREVIEW	BGA	ZKB	256		TBD	Call TI	Call TI
OMAPL137ZKBT3	PREVIEW	BGA	ZKB	256		TBD	Call TI	Call TI
XOMAPL137ZKB3	ACTIVE	BGA	ZKB	256	90	TBD	Call TI	Call TI

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

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

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

ZKB (S–PBGA–N256) PLASTIC BALL GRID ARRAY



NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. This is a lead–free solder ball design.

IMPORTANT NOTICE

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

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

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

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

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

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

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

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

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

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

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

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