WINSOURCE ELECTRONICS

# THE DATASHEET OF

TMS320VC5501GBE300

www.win-source.net 0086-755-83957316

# *TMS320VC5501 Fixed-Point Digital Signal Processor*

## *Data Manual*

Literature Number: SPRS206K
December 2002 − Revised November 2008

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

TEXAS INSTRUMENTS

## REVISION HISTORY

This data sheet revision history highlights the technical changes made to the SPRS206J device-specific data sheet to make it an SPRS206K revision.

**Scope:** See table below.

| PAGE(S) NO. | ADDITIONS/CHANGES/DELETIONS |
|---|---|
| 21 | Table 2−4, Signal Descriptions:<br>− HD[7:0]: removed “M” from “Other” column<br>− HC0: removed “M” from “Other” column<br>− HC1: removed “M” from “Other” column<br>− HCNTL0: removed “M” from “Other” column<br>− HCNTL1: removed “M” from “Other” column<br>− HCS: removed “M” from “Other” column<br>− HR/W: removed “M” from “Other” column |
| 88 | Table 3−29, Peripheral IDLE Control Register Bit Field Description:<br>− Updated footnote |
| 161 | Figure 5−22, Reset Timings:<br>− Added footnote about the state of the DSP pins during power up |

# Contents


| Section | | | Page |
|---|---|---|---|
| **1** | **TMS320VC5501 Features** | | **15** |
| **2** | **Introduction** | | **16** |
| 2.1 | Description | | 16 |
| 2.2 | Pin Assignments | | 17 |
| | 2.2.1 | Ball Grid Array (GZZ and ZZZ) | 17 |
| | 2.2.2 | Low-Profile Quad Flatpack (PGF) | 19 |
| 2.3 | Signal Descriptions | | 21 |
| **3** | **Functional Overview** | | **38** |
| 3.1 | Memory | | 39 |
| | 3.1.1 | On-Chip ROM | 39 |
| | 3.1.2 | On-Chip Dual-Access RAM (DARAM) | 40 |
| | 3.1.3 | Instruction Cache | 40 |
| | 3.1.4 | Memory Map | 41 |
| | 3.1.5 | Boot Configuration | 42 |
| 3.2 | Peripherals | | 42 |
| 3.3 | Configurable External Ports and Signals | | 43 |
| | 3.3.1 | Parallel Port Mux | 43 |
| | 3.3.2 | Host Port Mux | 45 |
| | 3.3.3 | External Bus Selection Register (XBSR) | 46 |
| | 3.3.4 | Configuration Examples | 48 |
| 3.4 | Timers | | 50 |
| | 3.4.1 | Timer Interrupts | 51 |
| | 3.4.2 | Timer Pins | 52 |
| | 3.4.3 | Timer Signal Selection Register (TSSR) | 53 |
| 3.5 | Universal Asynchronous Receiver/Transmitter (UART) | | 54 |
| 3.6 | Inter-Integrated Circuit ($I^2C$) Module | | 56 |
| 3.7 | Host-Port Interface (HPI) | | 57 |
| 3.8 | Direct Memory Access (DMA) Controller | | 58 |
| | 3.8.1 | DMA Channel 0 Control Register (DMA_CCR0) | 58 |
| 3.9 | System Clock Generator | | 60 |
| | 3.9.1 | Input Clock Source | 61 |
| | 3.9.2 | Clock Groups | 63 |
| | 3.9.3 | EMIF Input Clock Selection | 64 |
| | 3.9.4 | Changing the Clock Group Frequencies | 64 |
| | 3.9.5 | PLL Control Registers | 66 |
| | 3.9.6 | Reset Sequence | 76 |
| 3.10 | Idle Control | | 77 |
| | 3.10.1 | Clock Domains | 77 |
| | 3.10.2 | IDLE Procedures | 77 |
| | 3.10.3 | Module Behavior at Entering IDLE State | 80 |
| | 3.10.4 | Wake-Up Procedures | 81 |
| | 3.10.5 | Auto-Wakeup/Idle Function for McBSP and DMA | 84 |

Section Page


3.10.6 Clock State of Multiplexed Modules . . . 84
3.10.7 IDLE Control and Status Registers . . . 84
3.11 General-Purpose I/O (GPIO) . . . 92
3.11.1 General-Purpose I/O Port . . . 92
3.11.2 Parallel Port General-Purpose I/O (PGPIO) . . . 94
3.12 External Bus Control Register . . . 104
3.12.1 External Bus Control Register (XBCR) . . . 105
3.13 Internal Ports and System Registers . . . 106
3.13.1 XPORT Interface . . . 106
3.13.2 DPORT Interface . . . 109
3.13.3 IPORT Interface . . . 111
3.13.4 System Configuration Register (CONFIG) . . . 112
3.13.5 Time-Out Control Register (TOCR) . . . 113
3.14 CPU Memory-Mapped Registers . . . 114
3.15 Peripheral Registers . . . 116
3.16 Interrupts . . . 129
3.16.1 IFR and IER Registers . . . 130
3.16.2 Interrupt Timing . . . 131
3.16.3 Interrupt Acknowledge . . . 131
3.17 Notice Concerning TCK . . . 132

**4 Support . . . 135**
4.1 Notices Concerning JTAG (IEEE 1149.1) Boundary Scan Test Capability . . . 135
4.1.1 Initialization Requirements for Boundary Scan Test . . . 135
4.1.2 Boundary Scan Description Language (BSDL) Model . . . 135
4.2 Documentation Support . . . 135
4.3 Device and Development-Support Tool Nomenclature . . . 136

**5 Electrical Specifications . . . 137**
5.1 Absolute Maximum Ratings . . . 137
5.2 Electrical Specifications . . . 137
5.3 Recommended Operating Conditions . . . 137
5.4 Electrical Characteristics Over Recommended Operating Case Temperature Range . . . 138
5.5 Timing Parameter Symbology . . . 139
5.6 Clock Options . . . 140
5.6.1 Internal System Oscillator With External Crystal . . . 140
5.6.2 Layout Considerations . . . 141
5.6.3 Clock Generation in Bypass Mode (APLL Synthesis Disabled) . . . 142
5.6.4 Clock Generation in Lock Mode (APLL Synthesis Enabled) . . . 143
5.6.5 EMIF Clock Options . . . 145
5.7 Memory Timings . . . 147
5.7.1 Asynchronous Memory Timings . . . 147
5.7.2 Programmable Synchronous Interface Timings . . . 150
5.7.3 Synchronous DRAM Timings . . . 154
5.8 $\overline{\text{HOLD}}$/$\overline{\text{HOLDA}}$ Timings . . . 159
5.9 Reset Timings . . . 160
5.10 External Interrupt and Interrupt Acknowledge ($\overline{\text{IACK}}$) Timings . . . 162

Section Page


5.11 XF Timings . . . . . 163
5.12 General-Purpose Input/Output (GPIOx) Timings . . . . . 164
5.13 Parallel General-Purpose Input/Output (PGPIOx) Timings . . . . . 165
5.14 TIM0/TIM1/WDTOUT Timings . . . . . 166
5.14.1 TIM0/TIM1/WDTOUT Timer Pin Timings . . . . . 166
5.14.2 TIM0/TIM1/WDTOUT General-Purpose I/O Timings . . . . . 167
5.14.3 TIM0/TIM1/WDTOUT Interrupt Timings . . . . . 168
5.15 Multichannel Buffered Serial Port (McBSP) Timings . . . . . 169
5.15.1 McBSP Transmit and Receive Timings . . . . . 169
5.15.2 McBSP General-Purpose I/O Timings . . . . . 172
5.15.3 McBSP as SPI Master or Slave Timings . . . . . 173
5.16 Host-Port Interface Timings . . . . . 179
5.16.1 HPI Read and Write Timings . . . . . 179
5.16.2 HPI General-Purpose I/O Timings . . . . . 185
5.16.3 HPI.$\overline{\text{HAS}}$ Interrupt Timings . . . . . 186
5.17 Inter-Integrated Circuit ($I^2C$) Timings . . . . . 187
5.18 Universal Asynchronous Receiver/Transmitter (UART) Timings . . . . . 189

**6 Mechanical Data . . . . . 190**
6.1 Package Thermal Resistance Characteristics . . . . . 190
6.2 Packaging Information . . . . . 192

# List of Figures


| Figure | | Page |
|---|---|---|
| 2−1 | 201-Terminal GZZ and ZZZ Ball Grid Array (Bottom View) | 17 |
| 2−2 | 176-Pin PGF Low-Profile Quad Flatpack (Top View) | 19 |
| 3−1 | TMS320VC5501 Functional Block Diagram | 38 |
| 3−2 | TMS320VC5501 Memory Map | 41 |
| 3−3 | External Bus Selection Register Layout (0x6C00) | 46 |
| 3−4 | Configuration Example A (GPIO6 = 1 at Reset) | 48 |
| 3−5 | Configuration Example B (GPIO6 = 0 at Reset) | 49 |
| 3−6 | Timer Interrupts | 51 |
| 3−7 | Timer Pins | 52 |
| 3−8 | Timer Signal Selection Register Layout (0x8000) | 53 |
| 3−9 | UART Functional Block Diagram | 55 |
| 3−10 | $I^2C$ Module Block Diagram | 56 |
| 3−11 | DMA Channel 0 Control Register Layout (0x0C01) | 58 |
| 3−12 | System Clock Generator | 60 |
| 3−13 | Internal System Oscillator With External Crystal | 61 |
| 3−14 | Clock Generator Registers | 64 |
| 3−15 | PLL Control/Status Register Layout (0x1C80) | 66 |
| 3−16 | PLL Multiplier Control Register Layout (0x1C88) | 68 |
| 3−17 | PLL Divider 0 Register Layout (0x1C8A) | 69 |
| 3−18 | PLL Divider 1 Register Layout (0x1C8C) | 70 |
| 3−19 | PLL Divider 2 Register Layout (0x1C8E) | 70 |
| 3−20 | PLL Divider 3 Register Layout (0x1C90) | 71 |
| 3−21 | Oscillator Divider1 Register Layout (0x1C92) | 72 |
| 3−22 | Oscillator Wakeup Control Register Layout (0x1C98) | 73 |
| 3−23 | CLKOUT3 Select Register Layout (0x1C82) | 74 |
| 3−24 | CLKOUT Selection Register Layout (0x8400) | 74 |
| 3−25 | Clock Mode Control Register Layout (0x8C00) | 75 |
| 3−26 | IDLE Configuration Register Layout (0x0001) | 85 |
| 3−27 | IDLE Status Register Layout (0x0002) | 87 |
| 3−28 | Peripheral IDLE Control Register Layout (0x9400) | 88 |
| 3−29 | Peripheral IDLE Status Register Layout (0x9401) | 90 |
| 3−30 | Master IDLE Control Register Layout (0x9402) | 91 |
| 3−31 | Master IDLE Status Register Layout (0x9403) | 92 |
| 3−32 | GPIO Direction Register Layout (0x3400) | 93 |
| 3−33 | GPIO Data Register Layout (0x3401) | 93 |
| 3−34 | Parallel GPIO Enable Register 0 Layout (0x4400) | 95 |
| 3−35 | Parallel GPIO Direction Register 0 Layout (0x4401) | 96 |
| 3−36 | Parallel GPIO Data Register 0 Layout (0x4402) | 97 |

| Figure | | Page |
|---|---|---|
| 3−37 | Parallel GPIO Enable Register 1 Layout (0x4403) | 98 |
| 3−38 | Parallel GPIO Direction Register 1 Layout (0x4404) | 99 |
| 3−39 | Parallel GPIO Data Register 1 Layout (0x4405) | 100 |
| 3−40 | Parallel GPIO Enable Register 2 Layout (0x4406) | 101 |
| 3−41 | Parallel GPIO Direction Register 2 Layout (0x4407) | 102 |
| 3−42 | Parallel GPIO Data Register 2 Layout (0x4408) | 103 |
| 3−43 | External Bus Control Register Layout (0x8800) | 105 |
| 3−44 | XPORT Configuration Register Layout (0x0100) | 107 |
| 3−45 | XPORT Bus Error Register Layout (0x0102) | 108 |
| 3−46 | DPORT Configuration Register Layout (0x0200) | 109 |
| 3−47 | DPORT Bus Error Register Layout (0x0202) | 110 |
| 3−48 | IPORT Bus Error Register Layout (0x0302) | 111 |
| 3−49 | System Configuration Register Layout (0x07FD) | 112 |
| 3−50 | Time-Out Control Register Layout (0x9000) | 113 |
| 3−51 | IFR0, IER0, DBIFR0, and DBIER0 Registers Layout | 130 |
| 3−52 | IFR1, IER1, DBIFR1, and DBIER1 Registers Layout | 131 |
| 3−53 | Bad TCK Transition | 133 |
| 3−54 | Good TCK Transition | 133 |
| 3−55 | Sample Noise Filtering Circuitry | 134 |
| 5−1 | 3.3-V Test Load Circuit | 139 |
| 5−2 | Internal System Oscillator With External Crystal | 140 |
| 5−3 | Bypass Mode Clock Timings | 142 |
| 5−4 | External Multiply-by-N Clock Timings | 144 |
| 5−5 | ECLKIN Timings for EMIF | 145 |
| 5−6 | ECLKOUT1 Timings for EMIF Module | 145 |
| 5−7 | ECLKOUT2 Timings for EMIF Module | 146 |
| 5−8 | Asynchronous Memory Read Timings | 148 |
| 5−9 | Asynchronous Memory Write Timings | 149 |
| 5−10 | Programmable Synchronous Interface Read Timings (With Read Latency = 2) | 151 |
| 5−11 | Programmable Synchronous Interface Write Timings (With Write Latency = 0) | 152 |
| 5−12 | Programmable Synchronous Interface Write Timings (With Write Latency = 1) | 153 |
| 5−13 | SDRAM Read Command (CAS Latency 3) | 154 |
| 5−14 | SDRAM Write Command | 155 |
| 5−15 | SDRAM ACTV Command | 155 |
| 5−16 | SDRAM DCAB Command | 156 |
| 5−17 | SDRAM DEAC Command | 156 |
| 5−18 | SDRAM REFR Command | 157 |
| 5−19 | SDRAM MRS Command | 157 |
| 5−20 | SDRAM Self-Refresh Timings | 158 |
| 5−21 | EMIF.$\overline{\text{HOLD}}$/$\overline{\text{HOLDA}}$ Timings | 159 |
| 5−22 | Reset Timings | 161 |

| Figure | | Page |
|---|---|---|
| 5−23 | External Interrupt Timings | 162 |
| 5−24 | External Interrupt Acknowledge Timings | 162 |
| 5−25 | XF Timings | 163 |
| 5−26 | General-Purpose Input/Output (GPIOx) Signal Timings | 164 |
| 5−27 | Parallel General-Purpose Input/Output (PGPIOx) Signal Timings | 165 |
| 5−28 | TIM0/TIM1/WDTOUT Timings When Configured as Timer Input Pins | 166 |
| 5−29 | TIM0/TIM1/WDTOUT Timings When Configured as Timer Output Pins | 166 |
| 5−30 | TIM0/TIM1/WDTOUT General-Purpose I/O Timings | 167 |
| 5−31 | TIM0/TIM1/WDTOUT Interrupt Timings | 168 |
| 5−32 | McBSP Receive Timings | 171 |
| 5−33 | McBSP Transmit Timings | 171 |
| 5−34 | McBSP General-Purpose I/O Timings | 172 |
| 5−35 | McBSP Timings as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0 | 174 |
| 5−36 | McBSP Timings as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0 | 175 |
| 5−37 | McBSP Timings as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1 | 177 |
| 5−38 | McBSP Timings as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1 | 178 |
| 5−39 | Multiplexed Read Timings Using HPI.$\overline{\text{HAS}}$ | 181 |
| 5−40 | Multiplexed Read Timings With HPI.$\overline{\text{HAS}}$ Held High | 182 |
| 5−41 | Multiplexed Write Timings Using HPI.$\overline{\text{HAS}}$ | 183 |
| 5−42 | Multiplexed Write Timings With HPI.$\overline{\text{HAS}}$ Held High | 184 |
| 5−43 | $\overline{\text{HINT}}$ Timings | 184 |
| 5−44 | HPI General-Purpose I/O Timings | 185 |
| 5−45 | HPI.$\overline{\text{HAS}}$ Interrupt Timings | 186 |
| 5−46 | $I^2C$ Receive Timings | 187 |
| 5−47 | $I^2C$ Transmit Timings | 188 |
| 5−48 | UART Timings | 189 |

# List of Tables


| Table | | Page |
|---|---|---|
| 2−1 | 201-Terminal GZZ and ZZZ Ball Grid Array Thermal Ball Locations | 17 |
| 2−2 | 201-Terminal GZZ and ZZZ Ball Grid Array Ball Assignments | 18 |
| 2−3 | 176-Pin PGF Low-Profile Quad Flatpack Pin Assignments | 20 |
| 2−4 | Signal Descriptions | 21 |
| 3−1 | On-Chip ROM Layout | 39 |
| 3−2 | DARAM Blocks | 40 |
| 3−3 | Boot Configuration Selection Via the BOOTM[2:0] Pins | 42 |
| 3−4 | TMS320VC5501 Routing of Parallel Port Mux Signals | 44 |
| 3−5 | TMS320VC5501 Routing of Host Port Mux Signals | 45 |
| 3−6 | External Bus Selection Register Bit Field Description | 47 |
| 3−7 | Timer Signal Selection Register Bit Field Description | 53 |
| 3−8 | Synchronization Control Function | 59 |
| 3−9 | Recommended Crystal Parameters | 62 |
| 3−10 | Internal Clocks Frequency Ranges | 63 |
| 3−11 | PLL Control Registers | 66 |
| 3−12 | PLL Control/Status Register Bit Field Description | 67 |
| 3−13 | PLL Multiplier Control Register Bit Field Description | 68 |
| 3−14 | PLL Divider 0 Register Bit Field Description | 69 |
| 3−15 | PLL Divider 1 Register Bit Field Description | 70 |
| 3−16 | PLL Divider 2 Register Bit Field Description | 71 |
| 3−17 | PLL Divider 3 Register Bit Field Description | 71 |
| 3−18 | Oscillator Divider1 Register Bit Field Description | 72 |
| 3−19 | Oscillator Wakeup Control Register Bit Field Description | 73 |
| 3−20 | CLKOUT3 Select Register Bit Field Description | 74 |
| 3−21 | CLKOUT Selection Register Bit Field Description | 75 |
| 3−22 | Clock Mode Control Register Bit Field Description | 75 |
| 3−23 | Number of Reference Clock Cycles Needed Until Program Flow Begins | 76 |
| 3−24 | Peripheral Behavior at Entering IDLE State | 80 |
| 3−25 | Wake-Up Procedures | 83 |
| 3−26 | Clock Domain Memory-Mapped Registers | 84 |
| 3−27 | IDLE Configuration Register Bit Field Description | 85 |
| 3−28 | IDLE Status Register Bit Field Description | 87 |
| 3−29 | Peripheral IDLE Control Register Bit Field Description | 88 |
| 3−30 | Peripheral IDLE Status Register Bit Field Description | 90 |
| 3−31 | Master IDLE Control Register Bit Field Description | 91 |
| 3−32 | Master IDLE Status Register Bit Field Description | 92 |
| 3−33 | GPIO Direction Register Bit Field Description | 93 |
| 3−34 | GPIO Data Register Bit Field Description | 93 |
| 3−35 | TMS320VC5501 PGPIO Cross-Reference | 94 |
| 3−36 | Parallel GPIO Enable Register 0 Bit Field Description | 95 |
| 3−37 | Parallel GPIO Direction Register 0 Bit Field Description | 96 |
| 3−38 | Parallel GPIO Data Register 0 Bit Field Description | 97 |
| 3−39 | Parallel GPIO Enable Register 1 Bit Field Description | 98 |
| 3−40 | Parallel GPIO Direction Register 1 Bit Field Description | 99 |
| 3−41 | Parallel GPIO Data Register 1 Bit Field Description | 100 |

| Table | | Page |
|---|---|---|
| 3−42 | Parallel GPIO Enable Register 2 Bit Field Description | 101 |
| 3−43 | Parallel GPIO Direction Register 2 Bit Field Description | 102 |
| 3−44 | Parallel GPIO Data Register 2 Bit Field Description | 103 |
| 3−45 | Pins With Pullups, Pulldowns, and Bus Holders | 104 |
| 3−46 | External Bus Control Register Bit Field Description | 105 |
| 3−47 | I/O Addresses Under Scope of XPORT | 106 |
| 3−48 | XPORT Configuration Register Bit Field Description | 107 |
| 3−49 | XPORT Bus Error Register Bit Field Description | 108 |
| 3−50 | DPORT Configuration Register Bit Field Description | 109 |
| 3−51 | DPORT Bus Error Register Bit Field Description | 110 |
| 3−52 | IPORT Bus Error Register Bit Field Description | 111 |
| 3−53 | System Configuration Register Bit Field Description | 112 |
| 3−54 | Time-Out Control Register Bit Field Description | 113 |
| 3−55 | CPU Memory-Mapped Registers | 114 |
| 3−56 | Peripheral Bus Controller Configuration Registers | 116 |
| 3−57 | External Memory Interface Registers | 117 |
| 3−58 | DMA Configuration Registers | 118 |
| 3−59 | Instruction Cache Registers | 121 |
| 3−60 | Trace FIFO | 121 |
| 3−61 | Timer Signal Selection Register | 121 |
| 3−62 | Timers | 121 |
| 3−63 | Multichannel Serial Port #0 | 123 |
| 3−64 | Multichannel Serial Port #1 | 124 |
| 3−65 | HPI | 125 |
| 3−66 | GPIO | 125 |
| 3−67 | Device Revision ID | 126 |
| 3−68 | $I^2C$ | 126 |
| 3−69 | UART | 127 |
| 3−70 | External Bus Selection | 127 |
| 3−71 | Clock Mode Register | 127 |
| 3−72 | CLKOUT Selector Register | 127 |
| 3−73 | Clock Controller Registers | 128 |
| 3−74 | IDLE Control Registers | 128 |
| 3−75 | Interrupt Table | 129 |
| 5−1 | Recommended Crystal Parameters | 140 |
| 5−2 | CLKIN in Bypass Mode Timing Requirements | 142 |
| 5−3 | CLKOUT in Bypass Mode Switching Characteristics | 142 |
| 5−4 | CLKIN in Lock Mode Timing Requirements | 143 |
| 5−5 | CLKOUT in Lock Mode Switching Characteristics | 143 |
| 5−6 | EMIF Timing Requirements for ECLKIN | 145 |
| 5−7 | EMIF Switching Characteristics for ECLKOUT1 | 145 |
| 5−8 | EMIF Switching Characteristics for ECLKOUT2 | 146 |
| 5−9 | Asynchronous Memory Cycle Timing Requirements for ECLKIN | 147 |
| 5−10 | Asynchronous Memory Cycle Switching Characteristics for ECLKOUT1 | 147 |
| 5−11 | Programmable Synchronous Interface Timing Requirements | 150 |
| 5−12 | Programmable Synchronous Interface Switching Characteristics | 150 |

| Table | | Page |
|---|---|---|
| 5−13 | Synchronous DRAM Cycle Timing Requirements | 154 |
| 5−14 | Synchronous DRAM Cycle Switching Characteristics | 154 |
| 5−15 | EMIF.$\overline{\text{HOLD}}$/$\overline{\text{HOLDA}}$ Timing Requirements | 159 |
| 5−16 | EMIF.$\overline{\text{HOLD}}$/$\overline{\text{HOLDA}}$ Switching Characteristics | 159 |
| 5−17 | Reset Timing Requirements | 160 |
| 5−18 | Reset Switching Characteristics | 160 |
| 5−19 | External Interrupt and Interrupt Acknowledge Timing Requirements | 162 |
| 5−20 | External Interrupt and Interrupt Acknowledge Switching Characteristics | 162 |
| 5−21 | XF Switching Characteristics | 163 |
| 5−22 | GPIO Pins Configured as Inputs Timing Requirements | 164 |
| 5−23 | GPIO Pins Configured as Outputs Switching Characteristics | 164 |
| 5−24 | PGPIO Pins Configured as Inputs Timing Requirements | 165 |
| 5−25 | PGPIO Pins Configured as Outputs Switching Characteristics | 165 |
| 5−26 | TIM0/TIM1/WDTOUT Pins Configured as Timer Input Pins Timing Requirements | 166 |
| 5−27 | TIM0/TIM1/WDTOUT Pins Configured as Timer Output Pins Switching Characteristics | 166 |
| 5−28 | TIM0/TIM1/WDTOUT General-Purpose I/O Timing Requirements | 167 |
| 5−29 | TIM0/TIM1/WDTOUT General-Purpose I/O Switching Characteristics | 167 |
| 5−30 | TIM0/TIM1/WDTOUT Interrupt Timing Requirements | 168 |
| 5−31 | McBSP Transmit and Receive Timing Requirements | 169 |
| 5−32 | McBSP Transmit and Receive Switching Characteristics | 170 |
| 5−33 | McBSP General-Purpose I/O Timing Requirements | 172 |
| 5−34 | McBSP General-Purpose I/O Switching Characteristics | 172 |
| 5−35 | McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 0) | 173 |
| 5−36 | McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 0) | 173 |
| 5−37 | McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 0) | 175 |
| 5−38 | McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 0) | 175 |
| 5−39 | McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 1) | 176 |
| 5−40 | McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 1) | 176 |
| 5−41 | McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 1) | 178 |
| 5−42 | McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 1) | 178 |
| 5−43 | HPI Read and Write Timing Requirements | 179 |
| 5−44 | HPI Read and Write Switching Characteristics | 180 |
| 5−45 | HPI General-Purpose I/O Timing Requirements | 185 |
| 5−46 | HPI General-Purpose I/O Switching Characteristics | 185 |
| 5−47 | HPI.$\overline{\text{HAS}}$ Interrupt Timing Requirements | 186 |
| 5−48 | $I^2C$ Signals (SDA and SCL) Timing Requirements | 187 |
| 5−49 | $I^2C$ Signals (SDA and SCL) Switching Characteristics | 188 |
| 5−50 | UART Timing Requirements | 189 |
| 5−51 | UART Switching Characteristics | 189 |
| 6−1 | Thermal Resistance Characteristics (Ambient) | 190 |
| 6−2 | Thermal Resistance Characteristics (Case) | 191 |

# 1 TMS320VC5501 Features

- High-Performance, Low-Power, Fixed-Point TMS320C55x™ Digital Signal Processor (DSP)
  - 3.33-ns Instruction Cycle Time for 300-MHz Clock Rate
  - 16K-Byte Instruction Cache (I-Cache)
  - One/Two Instructions Executed per Cycle
  - Dual Multipliers [Up to 600 Million Multiply-Accumulates Per Second (MMACS)]
  - Two Arithmetic/Logic Units (ALUs)
  - One Program Bus, Three Internal Data/Operand Read Buses, and Two Internal Data/Operand Write Buses
- Instruction Cache (16K Bytes)
- 16K x 16-Bit On-Chip RAM That is Composed of Four Blocks of 4K × 16-Bit Dual-Access RAM (DARAM) (32K Bytes)
- 16K × 16-Bit One-Wait-State On-Chip ROM (32K Bytes)
- 8M × 16-Bit Maximum Addressable External Memory Space
- 32-Bit External Parallel Bus Memory Supporting External Memory Interface (EMIF) With General-Purpose Input/Output (GPIO) Capabilities and Glueless Interface to:
  - Asynchronous Static RAM (SRAM)
  - Asynchronous EPROM
  - Synchronous DRAM (SDRAM)
  - Synchronous Burst RAM (SBRAM)
- Emulation/Debug Trace Capability Saves Last 16 Program Counter (PC) Discontinuities and Last 32 PC Values
- Programmable Low-Power Control of Six Device Functional Domains
- On-Chip Peripherals
  - Six-Channel Direct Memory Access (DMA) Controller
  - Two Multichannel Buffered Serial Ports (McBSPs)
  - Programmable Analog Phase-Locked Loop (APLL) Clock Generator
  - General-Purpose I/O (GPIO) Pins and a Dedicated Output Pin (XF)
  - 8-Bit Parallel Host-Port Interface (HPI)
  - Four Timers
    - Two 64-Bit General-Purpose Timers
    - 64-Bit Programmable Watchdog Timer
    - 64-Bit DSP/BIOS™ Counter
  - Inter-Integrated Circuit ($I^2C$) Interface
  - Universal Asynchronous Receiver/ Transmitter (UART)
- On-Chip Scan-Based Emulation Logic
- IEEE Std 1149.1† (JTAG) Boundary Scan Logic
- Packages:
  - 176-Terminal LQFP (Low-Profile Quad Flatpack) (PGF Suffix)
  - 201-Terminal MicroStar BGA™ (Ball Grid Array) (GZZ and ZZZ Suffixes)
- 3.3-V I/O Supply Voltage
- 1.26-V Core Supply Voltage

TMS320C55x, DSP/BIOS, and MicroStar BGA are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.
† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

# 2 Introduction

This section describes the main features of the TMS320VC5501 and gives a brief description of the device.

**NOTE:** This document is designed to be used in conjunction with the *TMS320C55x DSP CPU Reference Guide* (literature number SPRU371).

## 2.1 Description

The TMS320VC5501 (5501) fixed-point digital signal processor (DSP) is based on the TMS320C55x™ DSP generation CPU processor core. The C55x™ DSP architecture achieves high performance and low power through increased parallelism and total focus on reduction in power dissipation. The CPU supports an internal bus structure that is composed of one program bus, three data read buses, two data write buses, and additional buses dedicated to peripheral and DMA activity. These buses provide the ability to perform up to three data reads and two data writes in a single cycle. In parallel, the DMA controller can perform data transfers independent of the CPU activity.

The C55x™ CPU provides two multiply-accumulate (MAC) units, each capable of 17-bit x 17-bit multiplication in a single cycle. A central 40-bit arithmetic/logic unit (ALU) is supported by an additional 16-bit ALU. Use of the ALUs is under instruction set control, providing the ability to optimize parallel activity and power consumption. These resources are managed in the Address Unit (AU) and Data Unit (DU) of the C55x CPU.

The C55x DSP generation supports a variable byte width instruction set for improved code density. The Instruction Unit (IU) performs 32-bit program fetches from internal or external memory and queues instructions for the Program Unit (PU). The Program Unit decodes the instructions, directs tasks to AU and DU resources, and manages the fully protected pipeline. Predictive branching capability avoids pipeline flushes on execution of conditional instructions.

The 5501 peripheral set includes an external memory interface (EMIF) that provides glueless access to asynchronous memories like EPROM and SRAM, as well as to high-speed, high-density memories such as synchronous DRAM and synchronous burst RAM. Additional peripherals include UART, watchdog timer, and an I-Cache. Two full-duplex multichannel buffered serial ports (McBSPs) provide glueless interface to a variety of industry-standard serial devices, and multichannel communication with up to 128 separately enabled channels. The host-port interface (HPI) is an 8-bit parallel interface used to provide host processor access to 16K words of internal memory on the 5501. The HPI operates in multiplexed mode to provide glueless interface to a wide variety of host processors. The DMA controller provides data movement for six independent channel contexts without CPU intervention. Two general-purpose timers, eight dedicated general-purpose I/O (GPIO) pins, and analog phase-locked loop (APLL) clock generation are also included.

The 5501 is supported by the industry's award-winning eXpressDSP™, Code Composer Studio™ Integrated Development Environment (IDE), DSP/BIOS™, Texas Instruments' algorithm standard, and the industry's largest third-party network. The Code Composer Studio™ IDE features code generation tools that include a C Compiler, Visual Linker, simulator, RTDX™, XDS510™ emulation device drivers, and evaluation modules. The 5501 is also supported by the C55x™ DSP Library, which features more than 50 foundational software kernels (FIR filters, IIR filters, FFTs, and various math functions) as well as chip and board support libraries.

C55x, eXpressDSP, Code Composer Studio, RTDX, and XDS510 are trademarks of Texas Instruments.

## 2.2 Pin Assignments

### 2.2.1 Ball Grid Array (GZZ and ZZZ)

The TMS320VC5501 is offered in two 201-terminal ball grid array (BGA) packages, both of which include 25 thermal balls to improve thermal dissipation. Except for their Eco-Status (refer to Section 6.2, Packaging Information), both packages are essentially the same. Figure 2−1 illustrates the ball locations for both BGA packages. Table 2−1 lists the locations of the thermal balls and Table 2−2 lists the signal names and terminal numbers.

**NOTE:**

Some TMX samples were shipped in the GGW package. For more information on the GGW package, see the *TMS320VC5502 and TMS320VC5501 Digital Signal Processors Silicon Errata* (literature number SPRZ020D or later).

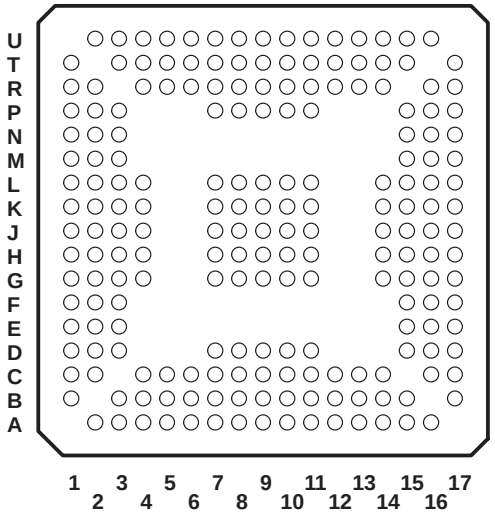



**Figure 2−1. 201-Terminal GZZ and ZZZ Ball Grid Array (Bottom View)**

**Table 2−1. 201-Terminal GZZ and ZZZ Ball Grid Array Thermal Ball Locations†**

| BALL NO. | BALL NO. | BALL NO. | BALL NO. | BALL NO. |
|---|---|---|---|---|
| G7 | G8 | G9 | G10 | G11 |
| H7 | H8 | H9 | H10 | H11 |
| J7 | J8 | J9 | J10 | J11 |
| K7 | K8 | K9 | K10 | K11 |
| L7 | L8 | L9 | L10 | L11 |

† For best device thermal performance:

- An array of 25 land pads must be added on the top layer of the PCB where the package will be mounted.
- The PCB land pads should be the same diameter as the vias in the package substrate for optimal Board Level Reliability Temperature Cycle performance.
- The land pads on the PCB should be connected together and to PCB through-holes. The PCB through-holes should in turn be connected to the ground plane for heat dissipation.
- A solid internal plane is preferred for spreading the heat.

Refer to the *MicroStar BGA™ Packaging Reference Guide* (literature number SSYZ015) for guidance on PCB design, surface mount, and reliability considerations.

**Table 2−2. 201-Terminal GZZ and ZZZ Ball Grid Array Ball Assignments†‡**

| BALL NO. | SIGNAL NAME | BALL NO. | SIGNAL NAME | BALL NO. | SIGNAL NAME | BALL NO. | SIGNAL NAME |
|---|---|---|---|---|---|---|---|
| B1 | GPIO6 | U2 | HCNTL1 | T17 | A19 | A16 | D16 |
| C2 | GPIO4 | T3 | HCNTL0 | R16 | A18 | B15 | D15 |
| C1 | GPIO2 | U3 | $V_{SS}$ | R17 | $V_{SS}$ | A15 | D14 |
| D3 | GPIO1 | R4 | HR/$\overline{W}$ | P15 | A17 | C14 | D13 |
| D2 | GPIO0 | T4 | $\overline{HDS2}$ | P16 | A16 | B14 | D12 |
| D1 | TIM1 | U4 | $CV_{DD}$ | P17 | $DV_{DD}$ | A14 | D11 |
| E3 | TIM0 | R5 | $\overline{HDS1}$ | N15 | A15 | C13 | D10 |
| E2 | $\overline{INT0}$ | T5 | HRDY | N16 | A14 | B13 | D9 |
| E1 | $CV_{DD}$ | U5 | $DV_{DD}$ | N17 | $V_{SS}$ | A13 | $DV_{DD}$ |
| F3 | $\overline{INT1}$ | R6 | CLKOUT | M15 | A13 | C12 | D8 |
| F2 | $\overline{INT2}$ | T6 | XF | M16 | A12 | B12 | D7 |
| F1 | $DV_{DD}$ | U6 | $V_{SS}$ | M17 | $CV_{DD}$ | A12 | $V_{SS}$ |
| G4 | $\overline{INT3}$ | P7 | C15 | L14 | A11 | D11 | D6 |
| G3 | $\overline{NMI}$/WDTOUT | R7 | C14 | L15 | A10 | C11 | D5 |
| G2 | $\overline{IACK}$ | T7 | $\overline{HINT}$ | L16 | A9 | B11 | D4 |
| G1 | $V_{SS}$ | U7 | $PV_{DD}$ | L17 | A8 | A11 | $CV_{DD}$ |
| H1 | CLKR0 | U8 | NC | K17 | $DV_{DD}$ | A10 | D3 |
| H4 | DR0 | P8 | X1 | K14 | A7 | D10 | D2 |
| H3 | FSR0 | R8 | X2/CLKIN | K15 | A6 | C10 | D1 |
| H2 | CLKX0 | T8 | EMIFCLKS | K16 | A5 | B10 | D0 |
| J1 | $CV_{DD}$ | U9 | $V_{SS}$ | J17 | $V_{SS}$ | A9 | $V_{SS}$ |
| J4 | DX0 | P9 | C13 | J14 | A4 | D9 | $\overline{EMU1}$/OFF |
| J3 | FSX0 | R9 | C12 | J15 | A3 | C9 | EMU0 |
| J2 | CLKR1 | T9 | C11 | J16 | A2 | B9 | TDO |
| K1 | DR1 | U10 | C10 | H17 | $CV_{DD}$ | A8 | $V_{SS}$ |
| K2 | FSR1 | T10 | C9 | H16 | D31 | B8 | TDI |
| K4 | DX1 | P10 | C8 | H14 | D30 | D8 | $\overline{TRST}$ |
| K3 | CLKX1 | R10 | C7 | H15 | D29 | C8 | TCK |
| L1 | $V_{SS}$ | U11 | $V_{SS}$ | G17 | $V_{SS}$ | A7 | TMS |
| L2 | FSX1 | T11 | ECLKIN | G16 | D28 | B7 | $\overline{RESET}$ |
| L3 | TEST§ | R11 | ECLKOUT2 | G15 | D27 | C7 | HPIENA |
| L4 | NC | P11 | ECLKOUT1 | G14 | D26 | D7 | HD7 |
| M1 | $CV_{DD}$ | U12 | $CV_{DD}$ | F17 | $CV_{DD}$ | A6 | $CV_{DD}$ |
| M2 | RX | T12 | C6 | F16 | D25 | B6 | HD6 |
| M3 | GPIO5 | R12 | C5 | F15 | D24 | C6 | HD5 |
| N1 | $DV_{DD}$ | U13 | $DV_{DD}$ | E17 | $DV_{DD}$ | A5 | $DV_{DD}$ |
| N2 | TX | T13 | C4 | E16 | D23 | B5 | HD4 |
| N3 | GPIO3 | R13 | C3 | E15 | D22 | C5 | HD3 |
| P1 | $V_{SS}$ | U14 | $V_{SS}$ | D17 | D21 | A4 | $CV_{DD}$ |
| P2 | SCL | T14 | C2 | D16 | D20 | B4 | HD2 |
| P3 | SDA | R14 | C1 | D15 | D19 | C4 | HD1 |
| R1 | HC1 | U15 | C0 | C17 | $V_{SS}$ | A3 | $V_{SS}$ |
| R2 | HC0 | T15 | A21 | C16 | D18 | B3 | HD0 |
| T1 | $\overline{HCS}$ | U16 | A20 | B17 | D17 | A2 | GPIO7¶ |

† $CV_{DD}$ is core $V_{DD}$, $DV_{DD}$ is I/O $V_{DD}$, and $PV_{DD}$ is PLL $V_{DD}$.
‡ NC indicates “no connect”.
§ The TEST pin is reserved for internal testing. It should be left unconnected.
¶ The GPIO7 pin must be low at the rising edge of the reset signal for the device to operate properly.

### 2.2.2 Low-Profile Quad Flatpack (PGF)

The TMS320VC5501 is offered in a 176-pin low-profile quad flatpack (LQFP). Figure 2−2 illustrates the pin locations for the 176-pin LQFP. Table 2−3 lists the signal names and pin numbers.

**NOTE:**

TMS320VC5501PGF has completed Temp Cycle reliability qualification testing with no failures through 1500 cycles of −55°C to 125°C following an EIA/JEDEC Moisture Sensitivity Level 4 pre-condition at 220+5/−0°C peak reflow. Exceeding this peak reflow temperature condition or storage and handling requirements may result in either immediate device failure post-reflow, due to package/die material delamination (“popcorning”), or degraded Temp cycle life performance.

Please note that Texas Instruments (TI) also provides MSL, peak reflow and floor life information on a bar-code label affixed to dry-pack shipping bags. Shelf life, temperature and humidity storage conditions and re-bake instructions are prominently displayed on a nearby screen-printed label.

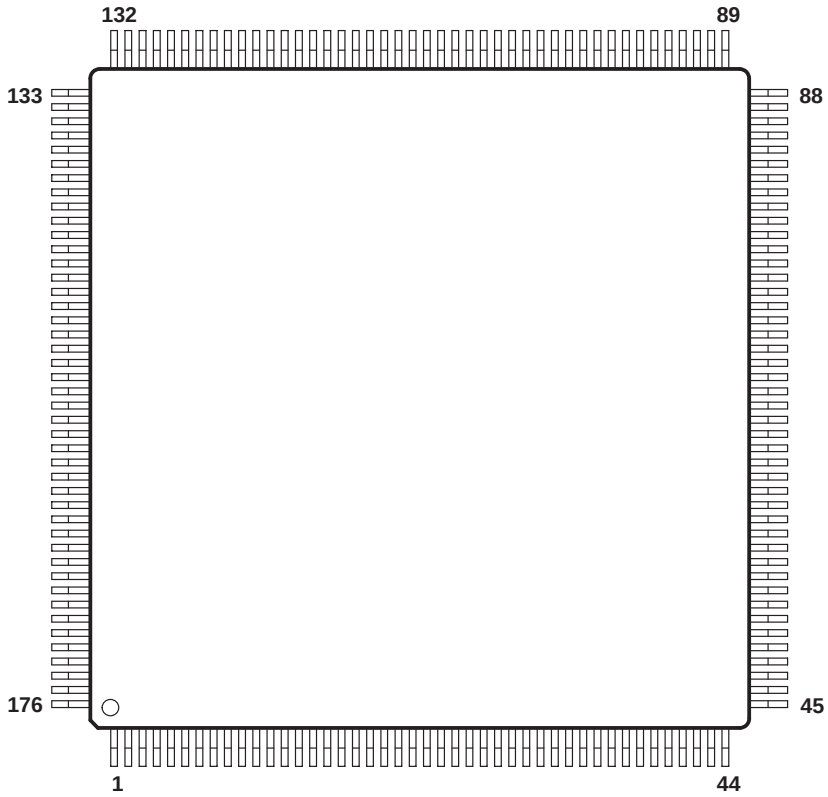



**Figure 2−2. 176-Pin PGF Low-Profile Quad Flatpack (Top View)**

**Table 2−3. 176-Pin PGF Low-Profile Quad Flatpack Pin Assignments†‡**

| PIN NO. | SIGNAL NAME | PIN NO. | SIGNAL NAME | PIN NO. | SIGNAL NAME | PIN NO. | SIGNAL NAME |
|---|---|---|---|---|---|---|---|
| 1 | GPIO6 | 45 | HCNTL1 | 89 | A19 | 133 | D16 |
| 2 | GPIO4 | 46 | HCNTL0 | 90 | A18 | 134 | D15 |
| 3 | GPIO2 | 47 | $V_{SS}$ | 91 | $V_{SS}$ | 135 | D14 |
| 4 | GPIO1 | 48 | HR/$\overline{W}$ | 92 | A17 | 136 | D13 |
| 5 | GPIO0 | 49 | $\overline{HDS2}$ | 93 | A16 | 137 | D12 |
| 6 | TIM1 | 50 | $CV_{DD}$ | 94 | $DV_{DD}$ | 138 | D11 |
| 7 | TIM0 | 51 | $\overline{HDS1}$ | 95 | A15 | 139 | D10 |
| 8 | $\overline{INT0}$ | 52 | HRDY | 96 | A14 | 140 | D9 |
| 9 | $CV_{DD}$ | 53 | $DV_{DD}$ | 97 | $V_{SS}$ | 141 | $DV_{DD}$ |
| 10 | $\overline{INT1}$ | 54 | CLKOUT | 98 | A13 | 142 | D8 |
| 11 | $\overline{INT2}$ | 55 | XF | 99 | A12 | 143 | D7 |
| 12 | $DV_{DD}$ | 56 | $V_{SS}$ | 100 | $CV_{DD}$ | 144 | $V_{SS}$ |
| 13 | $\overline{INT3}$ | 57 | C15 | 101 | A11 | 145 | D6 |
| 14 | $\overline{NMI}$/WDTOUT | 58 | C14 | 102 | A10 | 146 | D5 |
| 15 | $\overline{IACK}$ | 59 | $\overline{HINT}$ | 103 | A9 | 147 | D4 |
| 16 | $V_{SS}$ | 60 | $PV_{DD}$ | 104 | A8 | 148 | $CV_{DD}$ |
| 17 | CLKR0 | 61 | NC | 105 | $DV_{DD}$ | 149 | D3 |
| 18 | DR0 | 62 | X1 | 106 | A7 | 150 | D2 |
| 19 | FSR0 | 63 | X2/CLKIN | 107 | A6 | 151 | D1 |
| 20 | CLKX0 | 64 | EMIFCLKS | 108 | A5 | 152 | D0 |
| 21 | $CV_{DD}$ | 65 | $V_{SS}$ | 109 | $V_{SS}$ | 153 | $V_{SS}$ |
| 22 | DX0 | 66 | C13 | 110 | A4 | 154 | $\overline{EMU1/OFF}$ |
| 23 | FSX0 | 67 | C12 | 111 | A3 | 155 | EMU0 |
| 24 | CLKR1 | 68 | C11 | 112 | A2 | 156 | TDO |
| 25 | DR1 | 69 | C10 | 113 | $CV_{DD}$ | 157 | $V_{SS}$ |
| 26 | FSR1 | 70 | C9 | 114 | D31 | 158 | TDI |
| 27 | DX1 | 71 | C8 | 115 | D30 | 159 | $\overline{TRST}$ |
| 28 | CLKX1 | 72 | C7 | 116 | D29 | 160 | TCK |
| 29 | $V_{SS}$ | 73 | $V_{SS}$ | 117 | $V_{SS}$ | 161 | TMS |
| 30 | FSX1 | 74 | ECLKIN | 118 | D28 | 162 | $\overline{RESET}$ |
| 31 | TEST§ | 75 | ECLKOUT2 | 119 | D27 | 163 | HPIENA |
| 32 | NC | 76 | ECLKOUT1 | 120 | D26 | 164 | HD7 |
| 33 | $CV_{DD}$ | 77 | $CV_{DD}$ | 121 | $CV_{DD}$ | 165 | $CV_{DD}$ |
| 34 | RX | 78 | C6 | 122 | D25 | 166 | HD6 |
| 35 | GPIO5 | 79 | C5 | 123 | D24 | 167 | HD5 |
| 36 | $DV_{DD}$ | 80 | $DV_{DD}$ | 124 | $DV_{DD}$ | 168 | $DV_{DD}$ |
| 37 | TX | 81 | C4 | 125 | D23 | 169 | HD4 |
| 38 | GPIO3 | 82 | C3 | 126 | D22 | 170 | HD3 |
| 39 | $V_{SS}$ | 83 | $V_{SS}$ | 127 | D21 | 171 | $CV_{DD}$ |
| 40 | SCL | 84 | C2 | 128 | D20 | 172 | HD2 |
| 41 | SDA | 85 | C1 | 129 | D19 | 173 | HD1 |
| 42 | HC1 | 86 | C0 | 130 | $V_{SS}$ | 174 | $V_{SS}$ |
| 43 | HC0 | 87 | A21 | 131 | D18 | 175 | HD0 |
| 44 | $\overline{HCS}$ | 88 | A20 | 132 | D17 | 176 | GPIO7¶ |

† $CV_{DD}$ is core $V_{DD}$, $DV_{DD}$ is I/O $V_{DD}$, and $PV_{DD}$ is PLL $V_{DD}$.
‡ NC indicates “no connect”.
§ The TEST pin is reserved for internal testing. It should be left unconnected.
¶ The GPIO7 pin must be low at the rising edge of the reset signal for the device to operate properly.

## 2.3 Signal Descriptions

Table 2−4 lists each signal, function, and operating mode(s) grouped by function. See Section 2.2, Pin Assignments, for exact pin locations based on package type.

**Table 2−4. Signal Descriptions**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Address Bus** | | | | |
| A[21:18] | | I/O/Z | C, D, E, F, G, H, M | The A[21:18] pins of the Parallel Port serve one of two functions: parallel general-purpose input/output (PGPIO) signals PGPIO[3:0] or external memory interface (EMIF) address bus signals EMIF.A[21:18]. The function of the A[21:18] pins is determined by the state of the GPIO6 pin during reset. The A[21:18] pins are set to PGPIO[3:0] if GPIO6 is low during reset. The A[21:18] pins are set to EMIF.A[21:18] if GPIO6 is high during reset. The function of the A[21:18] pins will be set once the device is taken out of reset (RESET pin transitions from a low to high state).<br>The A[21:18] bus includes bus holders to reduce the static power dissipation caused by floating, unused pins. The bus holders also eliminate the need for external bias resistors on unused pins. When the bus goes into a high-impedance state, the bus holders keep the address bus at the logic level that was most recently driven. The bus holders are enabled at reset and can be enabled/disabled through the External Bus Control Register (XBCR). |
| | PGPIO[3:0] | I/O/Z | | **Parallel general-purpose I/O.** PGPIO[3:0] is selected if GPIO6 is low during reset. The PGPIO[3:0] signals are configured as inputs after reset. |
| | EMIF.A[21:18] | O/Z | | **EMIF address bus.** EMIF.A[21:18] is selected if GPIO6 is high during reset. The EMIF.A[21:18] signals are in a high-impedance state during reset and are configured as outputs after reset with an output value of 0. |
| A[17:2] | | I/O/Z | C, D, E, F, M | The A[17:2] pins of the Parallel Port serve one of two functions: external memory interface (EMIF) address bus signals EMIF.A[17:2] or reserved pins. The function of the A[17:2] pins is determined by the state of the GPIO6 pin during reset. The A[17:2] pins are reserved if GPIO6 is low during reset. The A[17:2] pins are set to EMIF.A[17:2] if GPIO6 is high during reset. The function of the A[17:2] pins will be set once the device is taken out of reset (RESET pin transitions from a low to high state).<br>The A[17:2] bus includes bus holders to reduce the static power dissipation caused by floating, unused pins. The bus holders also eliminate the need for external bias resistors on unused pins. When the bus goes into a high-impedance state, the bus holders keep the address bus at the logic level that was most recently driven. The bus holders are enabled at reset and can be enabled/disabled through the External Bus Control Register (XBCR). |
| | Reserved | I | | **Reserved pins.** These pins are reserved when GPIO6 is low during reset. |
| | EMIF.A[17:2] | O/Z | | **EMIF address bus.** EMIF.A[17:2] is selected when GPIO6 is high during reset. The EMIF.A[17:2] signals are in a high-impedance state during reset and are configured as outputs after reset with an output value of 0. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Data Bus** | | | | |
| D[31:16] | | I/O/Z | C, D, E, F, G, H, M | The D[31:16] pins of the Parallel Port serve one of two functions: parallel general-purpose input/output (PGPIO) signals PGPIO[19:4] or external memory interface (EMIF) data bus signals EMIF.D[31:16]. The function of the D[31:16] pins is determined by the state of the GPIO6 pin during reset. The D[31:16] pins are set to PGPIO[19:4] if GPIO6 is low during reset. The D[31:16] pins are set to EMIF.D[31:16] if GPIO6 is high during reset. The function of the D[31:16] pins will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state).<br>The D[31:16] bus includes bus holders to reduce the static power dissipation caused by floating, unused pins. The bus holders also eliminate the need for external bias resistors on unused pins. When the bus goes into a high-impedance state, the bus holders keep the data bus at the logic level that was most recently driven. The bus holders are enabled at reset and can be enabled/disabled through the External Bus Control Register (XBCR). |
| | PGPIO[19:4] | I/O/Z | | **Parallel general-purpose I/O.** PGPIO[19:4] is selected when GPIO6 is low during reset. The PGPIO[19:4] signals are configured as inputs after reset. |
| | EMIF.D[31:16] | I/O/Z | | **EMIF data bus.** EMIF.D[31:16] is selected when GPIO6 is high during reset. The EMIF.D[31:16] signals are set as inputs after reset. |
| D[15:0] | | I/O/Z | C, D, E, F, M | The D[15:0] pins of the Parallel Port serve one of two functions: external memory interface (EMIF) data bus signals EMIF.D[15:0] or reserved pins. The function of the D[15:0] pins is determined by the state of the GPIO6 pin during reset. The D[15:0] pins are reserved if GPIO6 is low during reset. The D[15:0] pins are set to EMIF.D[15:0] if GPIO6 is high during reset. The function of the D[15:0] pins will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state).<br>The D[15:0] bus includes bus holders to reduce the static power dissipation caused by floating, unused pins. The bus holders also eliminate the need for external bias resistors on unused pins. When the bus goes into a high-impedance state, the bus holders keep the data bus at the logic level that was most recently driven. The bus holders are enabled at reset and can be enabled/disabled through the External Bus Control Register (XBCR). |
| | Reserved | I/O/Z | | **Reserved pins.** These pins are reserved when GPIO6 is low during reset. |
| | EMIF.D[15:0] | I/O/Z | | **EMIF data bus.** EMIF.D[15:0] is selected when GPIO6 is high during reset. The EMIF.D[15:0] signals are configured as inputs after reset. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins** | | | | |
| C0 | | I/O/Z | C, D, E, F, G, H, M | The C0 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO20 or external memory interface control signal EMIF.ARE/SADS/SDCAS/SRE. The function of the C0 pin is determined by the state of the GPIO6 pin during reset. The C0 pin is set to PGPIO20 if GPIO6 is low during reset. The C0 pin is set to EMIF.ARE/SADS/SDCAS/SRE if GPIO6 is high during reset. The function of the C0 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO20 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO20 is selected when GPIO6 is low during reset. The PGPIO20 signal is configured as an input after reset. |
| | EMIF.ARE/SADS/SDCAS/SRE | O/Z | | **EMIF control pin.** EMIF.ARE/SADS/SDCAS/SRE is selected when GPIO6 is high during reset. The EMIF.ARE/SADS/SDCAS/SRE signal is in a high-impedance state during reset and is set to output after reset with an output value of 1.<br>The EMIF.ARE/SADS/SDCAS/SRE signal serves four different functions when used by the EMIF: asynchronous memory read-enable (EMIF.ARE), synchronous memory address strobe (EMIF.SADS), SDRAM column-address strobe (EMIF.SDCAS), and synchronous read-enable (EMIF.SRE) (selected by RENEN in the CE Secondary Control Register 1). |
| C1 | | I/O/Z | C, D, E, F, G, H, M | The C1 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO21 or external memory interface control signal EMIF.AOE/SOE/SDRAS. The function of the C1 pin is determined by the state of the GPIO6 pin during reset. The C1 pin is set to PGPIO21 if GPIO6 is low during reset. The C1 pin is set to EMIF.AOE/SOE/SDRAS if GPIO6 is high during reset. The function of the C1 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO21 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO21 is selected when GPIO6 is low during reset. The PGPIO21 signal is configured as an input after reset. |
| | EMIF.AOE/SOE/SDRAS | O/Z | | **EMIF control pin.** EMIF.AOE/SOE/SDRAS is selected when GPIO6 is high during reset. The EMIF.AOE/SOE/SDRAS signal is in a high-impedance state during reset and is set to output after reset with an output value of 1.<br>The EMIF.AOE/SOE/SDRAS signal serves three different functions when used by the EMIF: asynchronous memory output-enable (EMIF.AOE), synchronous memory output-enable (EMIF.SOE), and SDRAM row-address strobe (EMIF.SDRAS). |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins (Continued)** | | | | |
| C2 | | I/O/Z | C, D, E, F, G, H, M | The C2 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO22 or external memory interface control signal EMIF.AWE/SWE/SDWE. The function of the C2 pin is determined by the state of the GPIO6 pin during reset. The C2 pin is set to PGPIO22 if GPIO6 is low during reset. The C2 pin is set to EMIF.AWE/SWE/SDWE if GPIO6 is high during reset. The function of the C2 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO22 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO22 is selected when GPIO6 is low during reset. The PGPIO22 signal is configured as an input after reset. |
| | EMIF.AWE/ SWE/SDWE | O/Z | | **EMIF control pin.** EMIF.AWE/SWE/SDWE is selected when GPIO6 is high during reset. The EMIF.AWE/SWE/SDWE signal is in a high-impedance state during reset and is set to output after reset with an output value of 1.<br>The EMIF.AWE/SWE/SDWE signal serves three different functions when used by the EMIF: asynchronous memory write-enable (EMIF.AWE), synchronous memory write-enable (EMIF.SWE), and SDRAM write-enable (EMIF.SDWE). |
| C3 | | I/O/Z | F, G, H, J | The C3 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO23 or external memory interface control signal EMIF.ARDY. The function of the C3 pin is determined by the state of the GPIO6 pin during reset. The C3 pin is set to PGPIO23 if GPIO6 is low during reset. The C3 pin is set to EMIF.ARDY if GPIO6 is high during reset. The function of the C3 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO23 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO23 is selected when GPIO6 is low during reset. The PGPIO23 signal is configured as an input after reset. |
| | EMIF.ARDY | I | | **EMIF data ready pin.** EMIF.ARDY is selected when GPIO6 is high during reset.<br>The EMIF.ARDY signal indicates that an external device is ready for a bus transaction to be completed. If the device is not ready (EMIF.ARDY is low), the processor extends the memory access by one cycle and checks EMIF.ARDY again. An internal pullup is included to disable this feature if not used. The internal pullup can be disabled through the External Bus Control Register (XBCR). |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins (Continued)** | | | | |
| C4 | | I/O/Z | C, D, E, F, G, H, M | The C4 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO24 or external memory interface control signal EMIF.CE0. The function of the C4 pin is determined by the state of the GPIO6 pin during reset. The C4 pin is set to PGPIO24 if GPIO6 is low during reset. The C4 pin is set to EMIF.CE0 if GPIO6 is high during reset. The function of the C4 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO24 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO24 is selected when GPIO6 is low during reset. The PGPIO24 signal is configured as an input after reset. |
| | EMIF.CE0 | O/Z | | **EMIF chip-select for memory space CE0.** EMIF.CE0 is selected when GPIO6 is high during reset. The EMIF.CE0 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C5 | | I/O/Z | C, D, E, F, G, H, M | The C5 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO25 or external memory interface control signal EMIF.CE1. The function of the C5 pin is determined by the state of the GPIO6 pin during reset. The C5 pin is set to PGPIO25 if GPIO6 is low during reset. The C5 pin is set to EMIF.CE1 if GPIO6 is high during reset. The function of the C5 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO25 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO25 is selected when GPIO6 is low during reset. The PGPIO25 signal is configured as an input after reset. |
| | EMIF.CE1 | O/Z | | **EMIF chip-select for memory space CE1.** EMIF.CE1 is selected when GPIO6 is high during reset. The EMIF.CE1 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C6 | | I/O/Z | C, D, E, F, G, H, M | The C6 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO26 or external memory interface control signal EMIF.CE2. The function of the C6 pin is determined by the state of the GPIO6 pin during reset. The C6 pin is set to PGPIO26 if GPIO6 is low during reset. The C6 pin is set to EMIF.CE2 if GPIO6 is high during reset. The function of the C6 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO26 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO26 is selected when GPIO6 is low during reset. The PGPIO26 signal is configured as an input after reset. |
| | EMIF.CE2 | O/Z | | **EMIF chip-select for memory space CE2.** EMIF.CE2 is selected when GPIO6 is high during reset. The EMIF.CE2 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins (Continued)** | | | | |
| C7 | | I/O/Z | C, D, E, F, G, H, M | The C7 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO27 or external memory interface control signal EMIF.CE3. The function of the C7 pin is determined by the state of the GPIO6 pin during reset. The C7 pin is set to PGPIO27 if GPIO6 is low during reset. The C7 pin is set to EMIF.CE3 if GPIO6 is high during reset. The function of the C7 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO27 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO27 is selected when GPIO6 is low during reset. The PGPIO27 signal is configured as an input after reset. |
| | EMIF.CE3 | O/Z | | **EMIF chip-select for memory space CE3.** EMIF.CE3 is selected when GPIO6 is high during reset. The EMIF.CE3 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C8 | | I/O/Z | C, D, E, F, G, H, M | The C8 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO28 or external memory interface control signal EMIF.BE0. The function of the C8 pin is determined by the state of the GPIO6 pin during reset. The C8 pin is set to PGPIO28 if GPIO6 is low during reset. The C8 pin is set to EMIF.BE0 if GPIO6 is high during reset. The function of the C8 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO28 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO28 is selected when GPIO6 is low during reset. The PGPIO28 signal is configured as an input after reset. |
| | EMIF.BE0 | O/Z | | **EMIF byte-enable 0 control.** EMIF.BE0 is selected when GPIO6 is high during reset. The EMIF.BE0 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C9 | | I/O/Z | C, D, E, F, G, H, M | The C9 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO29 or external memory interface control signal EMIF.BE1. The function of the C9 pin is determined by the state of the GPIO6 pin during reset. The C9 pin is set to PGPIO29 if GPIO6 is low during reset. The C9 pin is set to EMIF.BE1 if GPIO6 is high during reset. The function of the C9 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO29 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO29 is selected when GPIO6 is low during reset. The PGPIO29 signal is configured as an input after reset. |
| | EMIF.BE1 | O/Z | | **EMIF byte-enable 1 control.** EMIF.BE1 is selected when GPIO6 is high during reset. The EMIF.BE1 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins (Continued)** | | | | |
| C10 | | I/O/Z | C, D, E, F, G, H, M | The C10 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO30 or external memory interface control signal EMIF.BE2. The function of the C10 pin is determined by the state of the GPIO6 pin during reset. The C10 pin is set to PGPIO30 if GPIO6 is low during reset. The C10 pin is set to EMIF.BE2 if GPIO6 is high during reset. The function of the C10 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO30 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO30 is selected when GPIO6 is low during reset. The PGPIO30 signal is configured as an input after reset. |
| | EMIF.$\overline{\text{BE2}}$ | O/Z | | **EMIF byte-enable 2 control.** EMIF.$\overline{\text{BE2}}$ is selected when GPIO6 is high during reset. The EMIF.$\overline{\text{BE2}}$ signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C11 | | I/O/Z | C, D, E, F, G, H, M | The C11 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO31 or external memory interface control signal EMIF.BE3. The function of the C11 pin is determined by the state of the GPIO6 pin during reset. The C11 pin is set to PGPIO31 if GPIO6 is low during reset. The C11 pin is set to EMIF.BE3 if GPIO6 is high during reset. The function of the C11 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO31 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO31 is selected when GPIO6 is low during reset. The PGPIO31 signal is configured as an input after reset. |
| | EMIF.$\overline{\text{BE3}}$ | O/Z | | **EMIF byte-enable 3 control.** EMIF.$\overline{\text{BE3}}$ is selected when GPIO6 is high during reset. The EMIF.$\overline{\text{BE3}}$ signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| C12 | | I/O/Z | C, D, E, F, G, H, M | The C12 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO32 or external memory interface control signal EMIF.SDCKE. The function of the C12 pin is determined by the state of the GPIO6 pin during reset. The C12 pin is set to PGPIO32 if GPIO6 is low during reset. The C12 pin is set to EMIF.SDCKE if GPIO6 is high during reset. The function of the C12 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO32 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO32 is selected when GPIO6 is low during reset. The PGPIO32 signal is configured as an input after reset. |
| | EMIF.SDCKE | O/Z | | **EMIF SDRAM clock-enable.** EMIF.SDCKE is selected when GPIO6 is high during reset. The EMIF.SDCKE signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Parallel Port − Control Pins (Continued)** | | | | |
| C13 | | I/O/Z | C, D, E, F, G, H, M | The C13 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO33 or external memory interface control signal EMIF.SOE3. The function of the C13 pin is determined by the state of the GPIO6 pin during reset. The C13 pin is set to PGPIO33 if GPIO6 is low during reset. The C13 pin is set to EMIF.SOE3 if GPIO6 is high during reset. The function of the C13 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO33 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO33 is selected when GPIO6 is low during reset.<br>The PGPIO33 signal is configured as an input after reset. |
| | EMIF.SOE3 | O/Z | | **EMIF synchronous memory output-enable for CE3.** EMIF.SOE3 is selected when GPIO6 is high during reset. The EMIF.SOE3 signal is in a high-impedance state during reset and is set to output after reset with an output value of 1.<br>The EMIF.SOE3 signal is intended for glueless FIFO interface. |
| C14 | | I/O/Z | F, G, H, J, M | The C14 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO34 or external memory interface control signal EMIF.HOLD. The function of the C14 pin is determined by the state of the GPIO6 pin during reset. The C14 pin is set to PGPIO34 if GPIO6 is low during reset. The C14 pin is set to EMIF.HOLD if GPIO6 is high during reset. The function of the C14 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO34 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO34 is selected when GPIO6 is low during reset.<br>The PGPIO34 signal is configured as an input after reset. |
| | EMIF.HOLD | I | | **EMIF hold request.** EMIF.HOLD is selected when GPIO6 is high during reset.<br>EMIF.HOLD is asserted by an external host to request control of the address, data, and control signals. |
| C15 | | I/O/Z | C, D, F, G, H, M | The C15 pin of the Parallel Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO35 or external memory interface control signal EMIF.HOLDA. The function of the C15 pin is determined by the state of the GPIO6 pin during reset. The C15 pin is set to PGPIO35 if GPIO6 is low during reset. The C15 pin is set to EMIF.HOLDA if GPIO6 is high during reset. The function of the C15 pin will be set once the device is taken out of reset (RESET pin transitions from a low to high state). |
| | PGPIO35 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO35 is selected when GPIO6 is low during reset.<br>The PGPIO35 signal is configured as an input after reset. |
| | EMIF.HOLDA | O/Z | | **EMIF hold acknowledge.** EMIF.HOLDA is selected when GPIO6 is high during reset. The EMIF.HOLDA signal is in a high-impedance state during reset and is set to output after reset with an output value of '1'.<br>EMIF.HOLDA is asserted by the DSP to indicate that the DSP is in the HOLD state and that the EMIF address, data, and control signals are in a high-impedance state, allowing the external memory interface to be accessed by other devices. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to HOLD pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode (TRST = 0, EMU0 = 1, and EMU1/OFF = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset (RESET pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **EMIF − Clock Pins** | | | | |
| ECLKIN | | I | C, L | **External EMIF input clock.** ECLKIN is selected as the input clock to the EMIF when EMIFCLKS is high. |
| ECLKOUT1 | | O/Z | E, F, M | **EMIF output clock.** ECLKOUT1 outputs the EMIF input clock by default but can be held low or set to a high-impedance state through the EMIF Global Control Register 1 (EGCR1).<br><br>The ECLKOUT1 pin is always in a high-impedance state during reset. The behavior of ECLKOUT1 immediately after reset depends on the state of GPIO6 during reset and EMIFCLKS:<br><br>**GPIO6 EMIFCLKS ECLKOUT1 Behavior**<br>0 0 Pin is in a high-impedance state.<br>0 1 Pin toggles at ECLKIN frequency.<br>1 0 Pin toggles at SYSCLK3 frequency.<br>1 1 Pin toggles at ECLKIN frequency. |
| ECLKOUT2 | | O/Z | E, F | **EMIF output clock.** ECLKOUT2 can be enabled to output the EMIF input clock divided by a factor 1, 2, or 4 through the EMIF Global Control Register 2 (EGCR2). ECLKOUT2 can also be held low or set to a high-impedance state through the EGCR2 register.<br><br>The ECLKOUT2 pin toggles with a clock frequency equal to the EMIF input clock divided by 4 during reset. The behavior of ECLKOUT2 immediately after reset depends on the state of GPIO6 during reset and EMIFCLKS:<br><br>**GPIO6 EMIFCLKS ECLKOUT2 Behavior**<br>0 0 Pin is held low.<br>0 1 Pin toggles at one-fourth of the ECLKIN frequency.<br>1 0 Pin toggles at one-fourth of the SYSCLK3 frequency.<br>1 1 Pin toggles at one-fourth of the ECLKIN frequency. |
| EMIFCLKS | | I | C, L | **EMIF input clock source select.** The clock source for the EMIF is determined by the state of the EMIFCLKS pin. The EMIF uses an internal clock (SYSCLK3) if EMIFCLKS is low. ECLKIN is used as the clock source if EMIFCLKS is high. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Host Port − Data Bus** | | | | |
| HD[7:0] | | I/O/Z | C, D, F, G, H | The HD[7:0] pins of the Host Port serve one of two functions: parallel general-purpose input/output (PGPIO) signals PGPIO[43:36] or host-port interface (HPI) data bus signals HPI.HD[7:0]. The function of the HD[7:0] pins is determined by the state of the GPIO6 pin during reset. The HD[7:0] pins are set to PGPIO[43:36] if GPIO6 is low during reset. The HD[7:0] pins are set to HPI.HD[7:0] if GPIO6 is high during reset. The function of the HD[7:0] pins will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state).<br>The HD[7:0] bus includes bus holders to reduce the static power dissipation caused by floating, unused pins. The bus holders also eliminate the need for external bias resistors on unused pins. When the bus goes into a high-impedance state, the bus holders keep the address bus at the logic level that was most recently driven. The bus holders are enabled at reset and can be enabled/disabled through the External Bus Control Register (XBCR). |
| | PGPIO[43:36] | I/O/Z | | **Parallel general-purpose I/O.** PGPIO[43:36] is selected when GPIO6 is low during reset. The PGPIO[43:36] signals are configured as inputs after reset. |
| | HPI.HD[7:0] | I/O/Z | | **Host data bus.** HPI.HD[7:0] is selected when GPIO6 is high during reset. The HPI.HD[7:0] signals are configured as inputs after reset.<br>The HPI will operate in multiplexed mode when GPIO6 is high during reset. In multiplexed mode, an 8-bit data bus (HPI.HD[7:0]) carries both address and data. Each host cycle on the bus consists of two consecutive 8-bit transfers. |
| **Host Port − Control Pins** | | | | |
| HC0 | | I/O/Z | C, F, G, H, J | The HC0 pin of the Host Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO44 or host-port interface (HPI) signal HPI.$\overline{\text{HAS}}$. The function of the HC0 pin is determined by the state of the GPIO6 pin during reset. The HC0 pin is set to PGPIO44 if GPIO6 is low during reset. The HC0 pin is set to HPI.$\overline{\text{HAS}}$ if GPIO6 is high during reset. The function of the HC0 pin will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state). |
| | PGPIO44 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO44 is selected when GPIO6 is low during reset. The PGPIO44 signal is configured as an input after reset. |
| | HPI.$\overline{\text{HAS}}$ | I | | **Host address strobe.** HPI.$\overline{\text{HAS}}$ is selected when GPIO6 is high during reset. The HPI.$\overline{\text{HAS}}$ signal is configured as an input after reset.<br>Hosts with multiplexed address and data pins may require HPI.$\overline{\text{HAS}}$ to latch the address in the HPIA register. HPI.$\overline{\text{HAS}}$ is only available when the HPI is operating in multiplexed mode. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Host Port − Control Pins (Continued)** | | | | |
| HC1 | | I/O/Z | F, G, H, K | The HC1 pin of the Host Port serves one of two functions: parallel general-purpose input/output (PGPIO) signal PGPIO45 or host-port interface (HPI) signal HPI.HBIL. The function of the HC1 pin is determined by the state of the GPIO6 pin during reset. The HC1 pin is set to PGPIO45 if GPIO6 is low during reset. The HC1 pin is set to HPI.HBIL if GPIO6 is high during reset. The function of the HC1 pin will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state). |
| | PGPIO45 | I/O/Z | | **Parallel general-purpose I/O.** PGPIO45 is selected when GPIO6 is low during reset. The PGPIO45 signal is configured as an input after reset. |
| | HPI.HBIL | I | | **Host byte identification.** HPI.HBIL is selected when GPIO6 is high during reset. The HPI.HBIL signal is configured as an input after reset.<br>In multiplexed mode, the host must use HPI.HBIL to identify the first and second bytes of the host cycle. |
| **HPI Pins** | | | | |
| HCNTL0 | | I/O/Z | F, G, H, J | **HPI access control pins.** The four binary states of the HCNTL0 and HCNTL1 pins determine which HPI register is being accessed by the host (HPIC, HPID with autoincrementing, HPIA, or HPID). The HCNTL0 and HCNTL1 pins are configured as inputs after reset. |
| HCNTL1 | | | | |
| $\overline{\text{HCS}}$ | | I/O/Z | C, F, G, H, J | **HPI chip-select.** $\overline{\text{HCS}}$ must be low for the HPI to be selected by the host. The $\overline{\text{HCS}}$ pin is configured as an input after reset.<br>A host must not initiate transfer requests until the HPI has been brought out of reset, see Section 3.7, Host-Port Interface (HPI), for more details. |
| HR/$\overline{\text{W}}$ | | I/O/Z | F, G, H, J | **Host read- or write-select.** HR/$\overline{\text{W}}$ indicates whether the current access is to be a read or write operation. The HR/$\overline{\text{W}}$ pin is configured as an input after reset. |
| $\overline{\text{HDS1}}$ | | I | C, G, H, J | **Host data strobe pins.** The $\overline{\text{HDS1}}$ and $\overline{\text{HDS2}}$ pins are used for strobing data in and out of the HPI. The $\overline{\text{HDS1}}$ and $\overline{\text{HDS2}}$ pins are configured as inputs after reset.<br>A host must not initiate transfer requests until the HPI has been brought out of reset, see Section 3.7, Host-Port Interface (HPI), for more details. |
| $\overline{\text{HDS2}}$ | | | | |
| HRDY | | O/Z | F, J, M | **Host ready (from DSP to host).** The HRDY pin informs the host when the HPI is ready for the next transfer. The HRDY pin is in a high-impedance state during reset and is set to output after reset with an output value of 1. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **HPI Pins (Continued)** | | | | |
| $\overline{\text{HINT}}$ | | O/Z | F, G, H, J, M | **Host interrupt (from DSP to host).** The $\overline{\text{HINT}}$ pin is used by the DSP to interrupt the host. The $\overline{\text{HINT}}$ signal is in a high-impedance state during reset and is set to output after reset with an output value of 1. |
| HPIENA | | I | C, L | **HPI enable.** The HPIENA pin must be dreiven high to enable the HPI for operation. If the HPIENA pin is low, the HPI will be completely disabled and all HPI output pins will be in a high-impedance state.<br>If the HPI is not needed, the HPIENA pin can be pulled low. |
| **Interrupt and Reset Pins** | | | | |
| $\overline{\text{INT}}$[3:0] | | I | C, L | **Maskable external interrupts**. $\overline{\text{INT0}}$−$\overline{\text{INT3}}$ are maskable interrupts.They are enabled through the Interrupt Enable Registers (IER0 and IER1). All maskable interrupts are globally enabled/disabled through the Interrupt Mode bit (INTM in ST1_55). $\overline{\text{INT0}}$−$\overline{\text{INT3}}$ can be polled and reset via the Interrupt Flag Registers (IFR0 and IFR1). All interrupts are prioritized as shown in Table 3−75, Interrupt Table. |
| $\overline{\text{NMI}}$/WDTOUT | | I/O/Z | C, F, J, M | **Non-maskable external interrupt or Watchdog Timer output.** The function of this pin is controlled by the Timer Signal Selection Register (TSSR). By default, the $\overline{\text{NMI}}$/WDTOUT pin has the function of the $\overline{\text{NMI}}$ signal.<br>$\overline{\text{NMI}}$ is an external interrupt that cannot be masked by the Interrupt Enable Registers (IER0 and IER1). When $\overline{\text{NMI}}$ is activated, the interrupt is always performed.<br>WDTOUT serves as an input and output pin for the Watchdog Timer. |
| $\overline{\text{IACK}}$ | | O/Z | F, M | **Interrupt acknowledge.** $\overline{\text{IACK}}$ indicates the receipt of an interrupt and that the program counter is fetching the interrupt vector location designated on the address bus. The $\overline{\text{IACK}}$ pin is set to a value of '1' during reset. |
| $\overline{\text{RESET}}$ | | I | C, L | **Device reset.** $\overline{\text{RESET}}$ causes the digital signal processor (DSP) to terminate current program execution. When $\overline{\text{RESET}}$ is brought to a high level, program execution begins by fetching the reset interrupt service vector at the reset vector address FFFF00h (IVPD:FFFFh). $\overline{\text{RESET}}$ affects various registers and status bits. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **General-Purpose I/O Pins** | | | | |
| GPIO7<br>GPIO6<br>GPIO5<br>GPIO4<br>GPIO3<br>GPIO2/BOOTM2<br>GPIO1/BOOTM1<br>GPIO0/BOOTM0 | | I/O/Z | F, G, H, M | **General-purpose configurable inputs/outputs.** GPIO[7:0] can be individually configured as inputs or outputs via the GPIO Direction Register (IODIR). Data can be read from inputs or written to outputs via the GPIO Data Register (IODATA). The GPIO pins are configured as inputs after reset.<br>**NOTE:** the GPIO7 pin must be low during the rising edge of the reset signal for the device to operate properly.<br>**Boot mode selection signals.** GPIO[2:0]/BOOTM[2:0] are sampled following reset to configure the boot mode for the DSP. After the boot is completed, these pins can be used as general-purpose inputs/outputs.<br>The GPIO4 pin is also used as an output for handshaking purposes on some of the boot modes. Although this pin is not involved in boot mode selection, users should be aware that this pin will become active as an output during the boot-load process and should design accordingly. After the boot-load process is complete, the loaded application may change the function of the GPIO4 pin.<br>**Input clock source selection.** The CLKMD0 bit of the Clock Mode Control Register (CLKMD) determines which clock, either OSCOUT or X2/CLKIN, is used as an input clock source to the DSP. If GPIO4 is low at reset, the CLKMD0 bit of the Clock Mode Control Register (CLKMD) will be set to ‘0’ and the internal oscillator and the external crystal will generate an input clock (OSCOUT) for the DSP. If GPIO4 is high, the CLKMD0 bit will be set to ‘1’ and the input clock will be taken directly from the X2/CLKIN pin.<br>An external crystal must be attached to the X1 and X2/CLKIN pins when the internal oscillator is used to generate a clock to the DSP. Otherwise, when the oscillator is not used to generate the input clock for the DSP, an externally generated 3.3-V clock must be applied to the X2/CLKIN pin and the X1 pin must be left unconnected.<br>**Function selection for multiplexed pins.** The GPIO6 pin is used to select the function of the multiplexed signals in the Parallel Port and the Host Port. The 5501 will be configured in PGPIO mode (EMIF and HPI are disabled) when the GPIO6 pin is low during reset. The 5501 will be configured in EMIF/HPI mode when the GPIO6 pin is high during reset. The function of the multiplexed signals will be set once the device is taken out of reset ($\overline{\text{RESET}}$ pin transitions from a low to high state). |
| XF | | O/Z | F | **External output** (latched software-programmable signal)**.** XF is set high by the BSET XF instruction, set low by BCLR XF instruction, or by loading ST1. XF is used for signaling other processors in multiprocessor configurations or used as a general-purpose output pin. The XF pin is set to a value of ‘1’ during reset. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

- A − Internal pullup [always enabled]
- B − Internal pulldown [always enabled]
- C − Hysteresis input
- D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
- F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
- G − Pin can be configured as a general-purpose input.
- H − PIn can be configured as a general-purpose output.
- J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
- L − Fail-safe pin
- M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Oscillator/Clock Pins** | | | | |
| CLKOUT | | O/Z | F | **Clock output.** CLKOUT can be set to reflect the clock of the Fast Peripherals Clock Group, Slow Peripherals Clock Group, and the External Memory Interface Clock Group. The CLKOUT pin is set to the internal clock SYSCLK1 during and after reset. SYSCLK1 is set equal to a divided-by-four CLKIN or OSCOUT (depending on the state of the GPIO4 pin) during and after reset. SYSCLK1 is used to clock the Fast Peripheral Clock Group. |
| X2/CLKIN | | I | | **Clock/oscillator input.** If the internal oscillator is not being used, X2/CLKIN functions as the clock input. |
| X1 | | O | | **Output pin from the internal oscillator for the crystal.** If the internal oscillator is not used, X1 should be left unconnected. |
| **Multichannel Buffered Serial Port Pins** | | | | |
| CLKR0 | | I/O/Z | C, F, G, H, M | **Receive clock input of McBSP0.** The CLKR0 pin is configured as input after reset. |
| DR0 | | I | L, G | **Serial data receive input of McBSP0** |
| FSR0 | | I/O/Z | F, G, H, M | **Frame synchronization pulse for receive input of McBSP0.** The FSR0 pin is configured as input after reset. |
| CLKX0 | | I/O/Z | C, F, G, H, M | **Transmit clock of McBSP0.** The CLKX0 pin is configured as input after reset. |
| DX0 | | O/Z | F, H, M | **Serial data transmit output of McBSP0.** The DX0 pin is in a high-impedance state during and after reset. |
| FSX0 | | I/O/Z | F, G, H, M | **Frame synchronization pulse for transmit output of McBSP0.** The FSX0 pin is configured as input after reset. |
| CLKR1 | | I/O/Z | C, G, H, M | **Receive clock input of McBSP1.** The CLKR1 pin is configured as input after reset. |
| DR1 | | I | L, G | **Serial data receive input of McBSP1** |
| FSR1 | | I/O/Z | F, G, H, M | **Frame synchronization pulse for receive input of McBSP1.** The FSR1 pin is configured as input after reset. |
| DX1 | | O/Z | F, H, M | **Serial data transmit output of McBSP1.** The DX1 pin is in a high-impedance state during and after reset. |
| CLKX1 | | I/O/Z | C, F, G, H, M | **Transmit clock of McBSP1.** The CLKX1 pin is configured as input after reset. |
| FSX1 | | I/O/Z | F, G, H, M | **Frame synchronization pulse for transmit output of McBSP1.** The FSX1 pin is configured as input after reset. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **UART Pins** | | | | |
| TX | | O | | **UART transmit data output.** The UART.TX signal outputs a value of 1 during and after reset. |
| RX | | I | | **UART receive data input** |
| **I2C Pins** | | | | |
| SCL | | I/O/Z | C, F, M | **I2C clock bidirectional port.** (Open collector I/O) |
| SDA | | I/O/Z | C, F, M | **I2C data bidirectional port.** (Open collector I/O) |
| **Timer Pins** | | | | |
| TIM0 | | I/O/Z | F, G, H, M | **Input/Output pin for Timer 0.** The TIM0 pin can be configured as an output or an input via the Timer Signal Selection Register (TSSR).When configured as an output, the TIM0 pin can signal a pulse or a change of state when the Timer 0 count matches its period. When configured as an input, the TIM0 pin can be used to provide the clock source for Timer 0 (external clock source mode) or it can be used to start/stop the timer from counting (clock gating mode). This pin can also be used as general-purpose I/O. The TIM0 pin is configured as an input after reset. |
| TIM1 | | I/O/Z | F, G, H, M | **Input/Output pin for Timer 1.** The TIM1 pin can be configured as an output or an input via the Timer Signal Selection Register (TSSR).When configured as an output, the TIM1 pin can signal a pulse or a change of state when the Timer 1 count matches its period. When configured as an input, the TIM1 pin can be used to provide the clock source for Timer 1 (external clock source mode) or it can be used to start/stop the timer from counting (clock gating mode). This pin can also be used as general-purpose I/O. The TIM1 pin is configured as an input after reset. |
| **Supply Pins** | | | | |
| $V_{SS}$ | | S | | **Digital Ground.** Dedicated ground for the device. |
| $CV_{DD}$ | | S | | **Digital Power, + $V_{DD}$.** Dedicated power supply for the core CPU. |
| $PV_{DD}$ | | S | | **Digital Power, + $V_{DD}$.** Dedicated power supply for the PLL module. |
| NC | | | | **No Connect** |
| $DV_{DD}$ | | S | | **Digital Power, + $V_{DD}$.** Dedicated power supply for the I/O pins. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{HOLD}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{TRST}$ = 0, EMU0 = 1, and EMU1/$\overline{OFF}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{RESET}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Test Pins** | | | | |
| TCK | | I | C, J | **IEEE standard 1149.1 test clock.** TCK is normally a free-running clock signal with a 50% duty cycle. The changes on test access port (TAP) of input signals TMS and TDI are clocked into the TAP controller, instruction register, or selected test data register on the rising edge of TCK. Changes at the TAP output signal (TDO) occur on the falling edge of TCK. Refer to Section 3.17, Notice Concerning TCK, for important information regarding this pin. |
| TDI | | I | J | **IEEE standard 1149.1 test data input.** Pin with internal pullup device. TDI is clocked into the selected register (instruction or data) on a rising edge of TCK. |
| TDO | | O/Z | | **IEEE standard 1149.1 test data output.** The contents of the selected register (instruction or data) are shifted out of TDO on the falling edge of TCK. TDO is in the high-impedance state except when the scanning of data is in progress. |
| TMS | | I | J | **IEEE standard 1149.1 test mode select.** Pin with internal pullup device. This serial control input is clocked into the TAP controller on the rising edge of TCK. |
| $\overline{\text{TRST}}$ | | I | C, L, K | **IEEE standard 1149.1 test reset.** $\overline{\text{TRST}}$, when high, gives the IEEE standard 1149.1 scan system control of the operations of the device. If $\overline{\text{TRST}}$ is not connected or driven low, the device operates in its functional mode, and the IEEE standard 1149.1 signals are ignored. Pin has an internal pulldown device. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

**Table 2−4. Signal Descriptions (Continued)**

| Pin Name | Multiplexed Signal Name | Pin Type† | Other‡ | Function |
|---|---|---|---|---|
| **Test Pins (Continued)** | | | | |
| EMU0 | | I/O/Z | J | **Emulator 0 pin.** When $\overline{\text{TRST}}$ is driven low, EMU0 must be high for activation of the $\overline{\text{OFF}}$ condition. When $\overline{\text{TRST}}$ is driven high, EMU0 is used as an interrupt to or from the emulator system and is defined as I/O by way of the IEEE standard 1149.1 scan system.<br>The EMU0 and EMU1/$\overline{\text{OFF}}$ pins must be pulled up when an emulator is not connected. Internal pullups have been included for this purpose. If the user chooses to disable these pins through the XBCR, external pullup resistors must be added to these two pins. |
| EMU1/$\overline{\text{OFF}}$ | | I/O/Z | J | **Emulator 1 pin/disable all outputs.** When $\overline{\text{TRST}}$ is driven high, EMU1/$\overline{\text{OFF}}$ is used as an interrupt to or from the emulator system and is defined as I/O by way of IEEE standard 1149.1 scan system. When $\overline{\text{TRST}}$ is driven low, EMU1/$\overline{\text{OFF}}$ is configured as $\overline{\text{OFF}}$. The EMU1/$\overline{\text{OFF}}$ signal, when active (low), puts all output drivers into the high-impedance state. Note that $\overline{\text{OFF}}$ is used exclusively for testing and emulation purposes (not for multiprocessing applications). Therefore, for the $\overline{\text{OFF}}$ condition, the following apply:<br>$\overline{\text{TRST}}$ = low,<br>EMU0 = high,<br>EMU1/$\overline{\text{OFF}}$ = low<br>The EMU0 and EMU1/$\overline{\text{OFF}}$ pins must be pulled up when an emulator is not connected. Internal pullups have been included for this purpose. If the user chooses to disable these pins through the XBCR, external pullup resistors must be added to these two pins. |

† I = Input, O = Output, S = Supply, Z = High impedance

‡ Other Pin Characteristics:

A − Internal pullup [always enabled]
B − Internal pulldown [always enabled]
C − Hysteresis input
D − Pin has bus holder, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
E − Pin is high impedance in HOLD mode (due to $\overline{\text{HOLD}}$ pin). The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during HOLD mode. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode.
F − Pin is high impedance in OFF mode ($\overline{\text{TRST}}$ = 0, EMU0 = 1, and EMU1/$\overline{\text{OFF}}$ = 0).
G − Pin can be configured as a general-purpose input.
H − PIn can be configured as a general-purpose output.
J − Pin has an internal pullup, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
K − Pin has an internal pulldown, it can be enabled/disabled through the External Bus Control Register (XBCR) [enabled by default].
L − Fail-safe pin
M − Pin is in high-impedance during reset ($\overline{\text{RESET}}$ pin is low)

# 3 Functional Overview

The following functional overview is based on the block diagram in Figure 3–1.

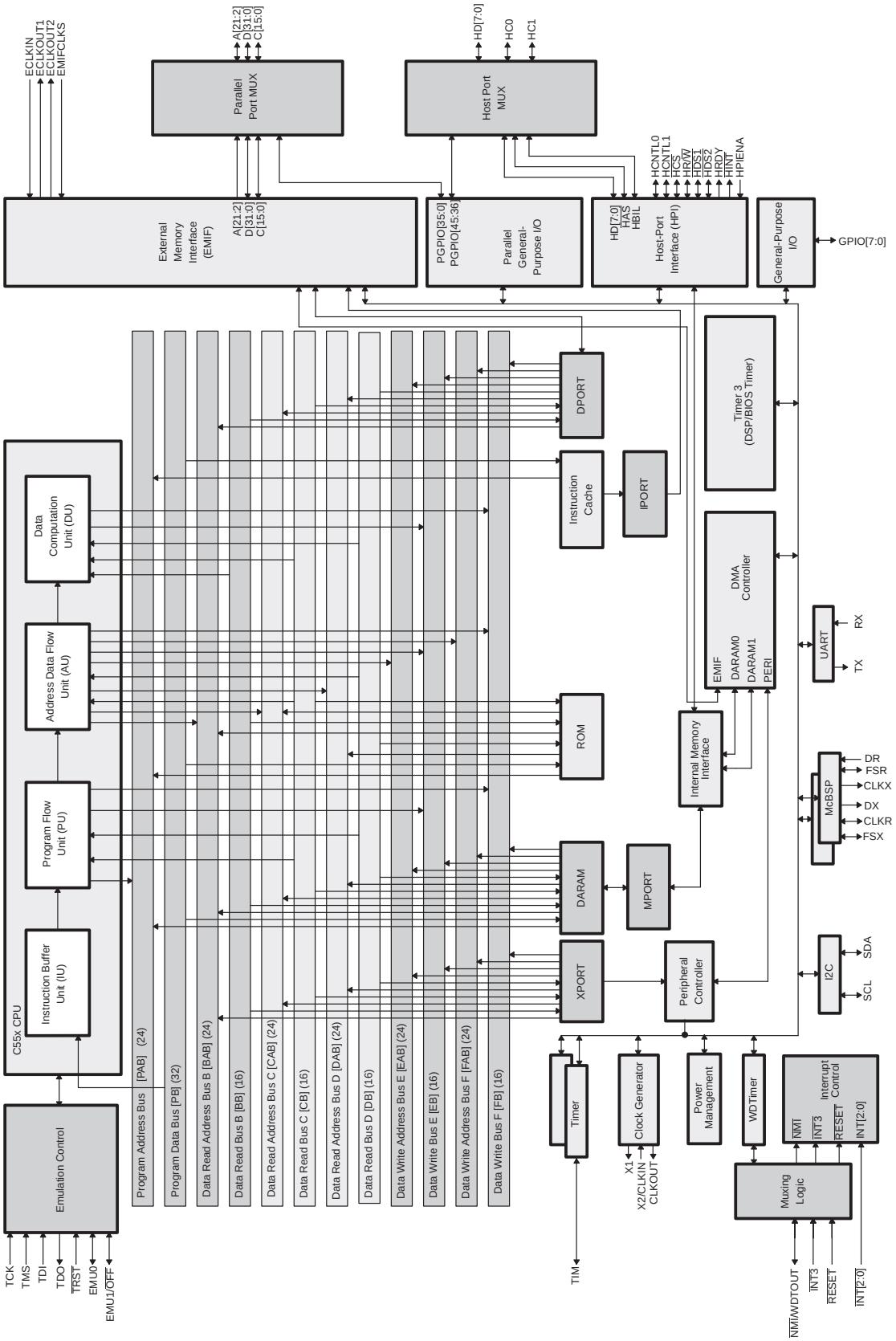



**Figure 3–1. TMS320VC5501 Functional Block Diagram**

## 3.1 Memory

The 5501 supports a unified memory map (program and data accesses are made to the same physical space). The total on-chip memory is 32K words (16K 16-bit words of RAM and 16K 16-bit words of ROM).

### *3.1.1 On-Chip ROM*

TMS320VC5501 incorporates 16K x16-bit of on-chip, one-wait-state maskable ROM that can be mapped into program memory space. The on-chip ROM is located at the byte address range FF8000h−FFFFFFh when MPNMC = 0 at reset. When MPNMC = 1 at reset, the on-chip ROM is disabled and not present in the memory map, and byte address range FF8000h−FFFFFFh is directed to external memory space. MPNMC is a bit located in the ST3 status register, and its status is determined by the logic level on the BOOTM[2:0] pins when sampled at reset. If BOOTM[2:0] are set to 00h at reset, the MPNMC bit is set to 1 and the on-chip ROM is disabled; otherwise, the MPNMC bit is cleared to 0 and the on-chip ROM is enabled. These pins are not sampled again until the next hardware reset. The software reset instruction does not affect the MPNMC bit. Software can be used to set or clear the MPNMC bit.

The ROM can be accessed by the program bus (P) and the two read data buses (C and D). The on-chip ROM is a two-cycle-per-word memory access, except for the first word access, which requires four cycles.

The standard on-chip ROM contains a bootloader which provides a variety of methods to load application code automatically after power up or a hardware reset. For more information, see Section 3.1.5, Boot Configuration. A vector table associated with the bootloader is also contained in the ROM. A boot mode branch table is included in the ROM which contains hard-coded jumps to the beginning of each boot mode code section in the bootloader.

A sine look-up table is provided containing 256 values (crossing 360 degrees) expressed in Q15 format.

The standard on-chip ROM layout is shown in Table 3−1.

**Table 3−1. On-Chip ROM Layout**

| STARTING BYTE ADDRESS | CONTENTS |
|---|---|
| FF_8000h | Bootloader Program |
| FF_ECAEh | Bootloader Revision Number |
| FF_ECB0h | Boot Mode Branch Table |
| FF_ED00h | Sine Table |
| FF_EF00h | Reserved |
| FF_FF00h | Interrupt Vector Table |

### 3.1.2 On-Chip Dual-Access RAM (DARAM)

TMS320VC5501 features 16K x 16-bit (32K bytes) of on-chip dual-access RAM. This memory enhances system performance, since the C55x CPU can access a DARAM block twice per machine cycle. The DARAM is composed of 4 blocks of 4K x 16-bit each (see Table 3−2). Each block in the DARAM can support two reads in one cycle, a read and a write in one cycle, or two writes in one cycle. The dual-access RAM is located in the (byte) address range 000000h−007FFFh, it can be accessed by the program, data and DMA buses. The HPI has NO access to the DARAM block when device is in reset.

**Table 3−2. DARAM Blocks**

| BYTE ADDRESS RANGE | MEMORY BLOCK |
|---|---|
| 000000h − 001FFFh | DARAM 0† |
| 002000h − 003FFFh | DARAM 1 |
| 004000h − 005FFFh | DARAM 2 |
| 006000h − 007FFFh | DARAM 3 |

† First 192 bytes are reserved for Memory-Mapped Registers (MMRs).

### 3.1.3 Instruction Cache

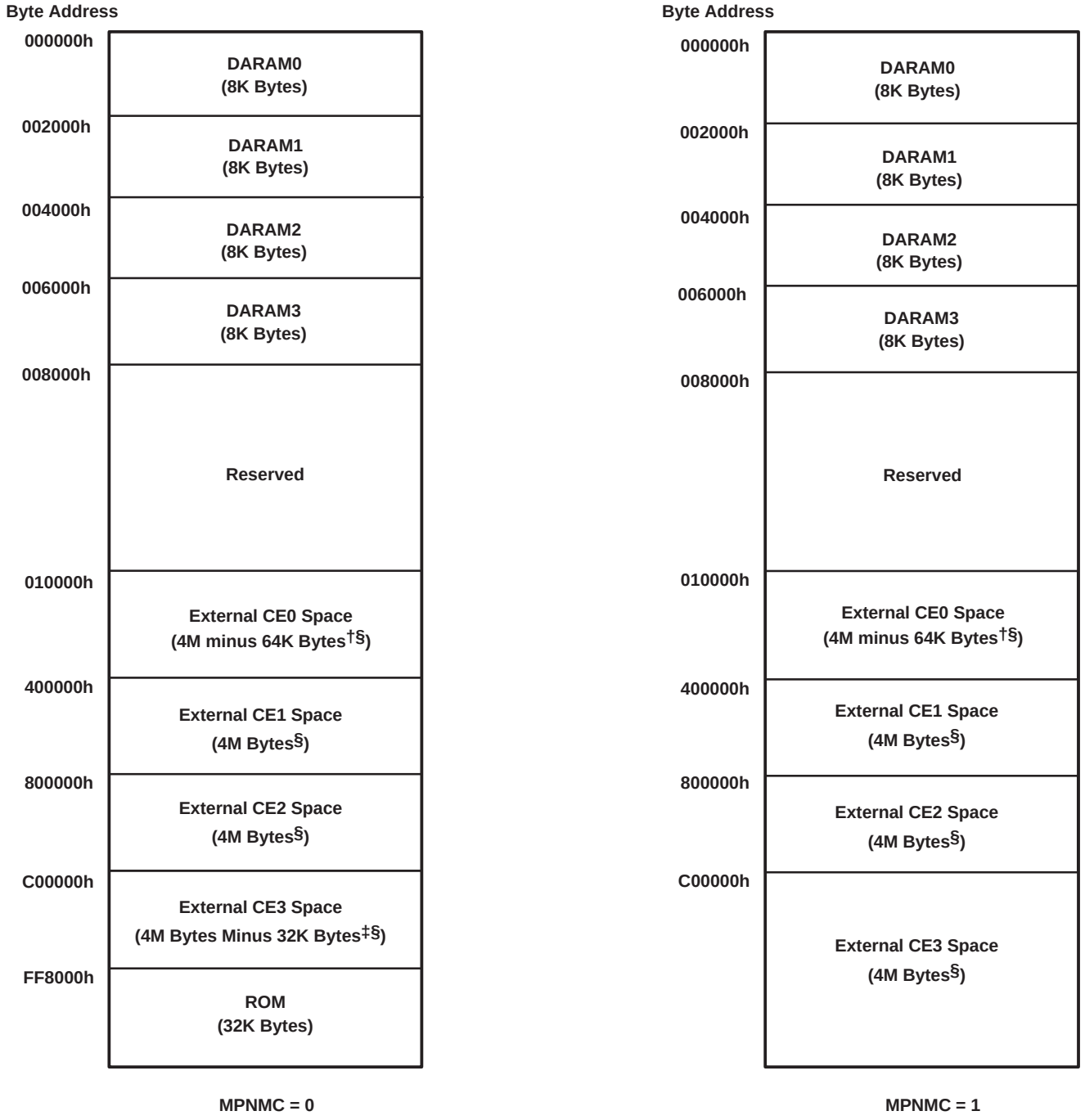
On the TMS320VC5501, instructions may reside in internal memory or external memory. When instructions reside in external memory, the I-Cache can improve the overall system performance by buffering the most recent instructions accessed by the CPU.

The 5501 includes a 16K-byte instruction cache, which consists of a single 2-way cache block. The 2-way cache uses 2-way associative mapping and holds up to 16K bytes: 512 sets, two lines per set, four 32-bit words per line. In the 2-way cache, each line is identified by a unique tag. The 2-way cache is updated based on a least-recently-used algorithm.

Control bits in the CPU status register ST3_55 provide the ability to enable, freeze, and flush the cache.

For more information on the instruction cache, see the *TMS320VC5501/5502 DSP Instruction Cache Reference Guide* (literature number SPRU630).

### 3.1.4 Memory Map




† The lower 64K bytes in CE0 Space include 32K bytes of DARAM space and 32K bytes of reserved space.
‡ The 32K bytes are for on-chip ROM block.
§ The CE space size shown in the figure represents the maximum addressable memory space for a 32-bit EMIF configuration. The maximum addressable memory space per CE is reduced when 16- or 8-bit EMIF configurations are used for asynchronous and SBSRAM memory types. For more detailed information, refer to *TMS320VC5501/5502 DSP External Memory Inteface (EMIF) Reference Guide* (literature number SPRU621).

**Figure 3−2. TMS320VC5501 Memory Map**

### 3.1.5 Boot Configuration

The on-chip bootloader provides a way to transfer application code and tables from an external source to the on-chip RAM at power up. The 5501 provides several options to download the code to accommodate varying system requirements. These options include:

- Host-port interface (HPI) boot in multiplexed mode
- External memory boot (via EMIF) from 16-bit asynchronous memory
- Serial port boot (from McBSP0) with 16-bit element length
- SPI EPROM boot (from McBSP0) supporting EPROMs with 24-bit addresses
- $I^2C$ EPROM boot (from $I^2C$) supporting EPROMs larger than 512K bits
- UART boot
- Direct execution (no boot) from 16-bit or 32-bit external asynchronous memory

The external pins BOOTM2, BOOTM1, and BOOTM0 select the boot configuration. The values of BOOTM[2:0] are latched with the rising edge of the $\overline{\text{RESET}}$ input. BOOTM2 is shared with GPIO2, BOOTM1 is shared with GPIO1, and BOOTM0 is shared with GPIO0.

The boot configurations available are summarized in Table 3−3.

**Table 3−3. Boot Configuration Selection Via the BOOTM[2:0] Pins**

| BOOTM[2:0] | BOOT PROCESS |
|---|---|
| 000 | Direct execution from 16-bit external asynchronous memory |
| 001 | SPI EPROM boot |
| 010 | Serial port boot (from McBSP0) |
| 011 | External memory boot (via EMIF) from 16-bit asynchronous memory |
| 100 | Direct execution from 32-bit external asynchronous memory |
| 101 | HPI boot |
| 110 | $I^2C$ EPROM boot |
| 111 | UART boot |

## 3.2 Peripherals

The 5501 includes the following on-chip peripherals:

- An external memory interface (EMIF)† supporting a 32-bit interface to asynchronous memory, SDRAM, and SBSRAM
- An 8-bit host-port interface (HPI)†
- A six-channel direct memory access (DMA) controller
- Two multichannel buffered serial ports (McBSPs)
- A programmable analog phase-locked loop (APLL) clock generator
- General-purpose I/O (GPIO) pins and a dedicated output pin (XF)

† The 5501 can be configured as follows:
- EMIF/HPI mode: 32-bit external memory interface with 8-bit host-port interface
- PGPIO mode: PGPIO support with no external memory interface and no host-port interface

- Four timers
  - Two 64-bit general-purpose timers
  - A programmable watchdog timer
  - A DSP/BIOS timer
- An Inter-integrated Circuit ($I^2C$) multi-master and slave interface
- A Universal Asynchronous Receiver/Transmitter (UART)

For detailed information on the C55x DSP peripherals, see the following documents:

- *TMS320VC5501/5502 DSP Instruction Cache Reference Guide* (literature number SPRU630)
- *TMS320VC5501/5502 DSP Timers Reference Guide* (literature number SPRU618)
- *TMS320VC5501/5502/5503/5507/5509 DSP Inter-Integrated Circuit (I2C) Module Reference Guide* (literature number SPRU146)
- *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)
- *TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide* (literature number SPRU613)
- *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592)
- *TMS320VC5501/5502 DSP External Memory Interface (EMIF) Reference Guide* (literature number SPRU621)
- *TMS320VC5501/5502 DSP Universal Asynchronous Receiver/Transmitter (UART) Reference Guide* (literature number SPRU597)

## 3.3 Configurable External Ports and Signals

A number of pins on the 5501 have two functions, a feature that allows system designers to choose an appropriate media interface for his/her application without the need for a large pin-count package. Two muxes are included in the 5501 to control the configuration of these dual-function pins: the Parallel Port Mux and the Host Port Mux. The state of these muxes is set at reset based on the state of the GPIO6 pin. The External Bus Selection Register (XBSR) reflects the configuration of these muxes after the 5501 comes out of reset.

### *3.3.1 Parallel Port Mux*

The Parallel Port Mux of the 5501 controls the function of 20 address signals (pins A[21:2]), 32 data signals (pins D[31:0]), and 16 control signals (pins C0 through C15). The Parallel Port Mux supports two different modes:

- **Full EMIF mode:** The EMIF is enabled and its 20 address, 32 data, and 16 control signals are routed to their corresponding pins on the Parallel Port Mux.
- **Parallel general-purpose I/O mode:** The EMIF and HPI are disabled and 16 control, 4 address, and 16 data pins of the Parallel Port Mux are set to parallel general-purpose I/O (PGPIO).

The mode of the Parallel Port Mux is determined by the state of the GPIO6 pin at reset. If GPIO6 is low, the EMIF and the HPI will be disabled: pins A[17:2] and pins D[15:0] will become reserved pins. All other pins in the Parallel Port Mux are set to parallel general-purpose I/O. The Parallel/Host Port Mux Mode bit field in the External Bus Selection Register (XBSR) will also be set to 0 to reflect the PGPIO mode of the Parallel Port Mux.

If GPIO6 is high at reset, the HPI will be enabled in multiplexed mode and the EMIF will be fully enabled: pins A[21:2] are set to EMIF.A[21:2], pins D[31:0] are set to EMIF.D[31:0], and pins C[15:0] are set to their corresponding EMIF operation. The Parallel/Host Port Mux Mode bit field in the XBSR will be set to 1 to reflect the full EMIF mode of the Parallel Port Mux. Note that in multiplexed mode, the HPI will use the HD[7:0] pins to strobe in address and data information (see Section 3.7, Host-Port Interface (HPI), for more information on the operation of the HPI in multiplexed mode).

Table 3−4 lists the individual routing of the EMIF and PGPIO signals to the external parallel address, data, and control buses.

**Table 3−4. TMS320VC5501 Routing of Parallel Port Mux Signals**

| PIN | PARALLEL/HOST PORT MUX MODE = 0 (PGPIO) | PARALLEL/HOST PORT MUX MODE = 1 (FULL EMIF) |
|---|---|---|
| **Address Bus** | | |
| A[17:2] | Reserved | EMIF.A[17:2] |
| A[21:18] | PGPIO[3:0] | EMIF.A[21:18] |
| **Data Bus** | | |
| D[31:16] | PGPIO[19:4] | EMIF.D[31:16] |
| D[15:0] | Reserved | EMIF.D[15:0] |
| **Control Bus** | | |
| C0 | PGPIO20 | EMIF.$\overline{\text{ARE}}$/$\overline{\text{SADS}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SRE}}$ |
| C1 | PGPIO21 | EMIF.$\overline{\text{AOE}}$/$\overline{\text{SOE}}$/$\overline{\text{SDRAS}}$ |
| C2 | PGPIO22 | EMIF.$\overline{\text{AWE}}$/$\overline{\text{SWE}}$/$\overline{\text{SDWE}}$ |
| C3 | PGPIO23 | EMIF.ARDY |
| C4 | PGPIO24 | EMIF.$\overline{\text{CE0}}$ |
| C5 | PGPIO25 | EMIF.$\overline{\text{CE1}}$ |
| C6 | PGPIO26 | EMIF.$\overline{\text{CE2}}$ |
| C7 | PGPIO27 | EMIF.$\overline{\text{CE3}}$ |
| C8 | PGPIO28 | EMIF.$\overline{\text{BE0}}$ |
| C9 | PGPIO29 | EMIF.$\overline{\text{BE1}}$ |
| C10 | PGPIO30 | EMIF.$\overline{\text{BE2}}$ |
| C11 | PGPIO31 | EMIF.$\overline{\text{BE3}}$ |
| C12 | PGPIO32 | EMIF.SDCKE |
| C13 | PGPIO33 | EMIF.$\overline{\text{SOE3}}$ |
| C14 | PGPIO34 | EMIF.$\overline{\text{HOLD}}$ |
| C15 | PGPIO35 | EMIF.$\overline{\text{HOLDA}}$ |

### 3.3.2 Host Port Mux

The 5501 Host Port Mux controls the function of 8 data signals (pins HD[7:0]) and 2 control signals (pins HC0 and HC1). The Host Port Mux supports two different modes:

- **8-bit multiplexed mode:** The HPI's 8 data and 2 control signals are routed to their corresponding pins on the Host Port Mux.
- **Parallel general-purpose I/O mode:** All pins on the Host Port Mux are routed to PGPIO. The HPI and EMIF are disabled.

The mode of the Host Port Mux is determined by the state of the GPIO6 pin at reset. If GPIO6 is low, the pins of the Host Port Mux will be set to PGPIO. In this mode, the EMIF and the HPI will be disabled. The Parallel/Host Port Mux Mode bit of the External Bus Control Register will be set to 0 to reflect the PGPIO mode of the Host Port Mux.

If GPIO6 is high, the HPI will be enabled in 8-bit (multiplexed) mode: pins HD[7:0] are set to HPI.HD[7:0], and HC0 and HC1 are set to HPI.$\overline{\text{HAS}}$ and HPI.HBIL, respectively. The Parallel/Host Port Mux Mode bit field in the XBSR will be set to 1 to reflect the HPI multiplexed mode of the Host Port Mux. See Section 3.7, Host-Port Interface (HPI), for more information on the operation of the HPI in multiplexed mode.

Table 3−5 lists the individual routing of the HPI and PGPIO signals to the Host Port Mux pins.

**Table 3−5. TMS320VC5501 Routing of Host Port Mux Signals**

| PIN | PARALLEL/HOST PORT MUX MODE = 0 (PGPIO) | PARALLEL/HOST PORT MUX MODE = 1 (8-BIT HPI MULTIPLEXED) |
|---|---|---|
| **Data Bus** | | |
| HD[7:0] | PGPIO[43:36] | HPI.HD[7:0] |
| **Control Bus** | | |
| HC0 | PGPIO44 | HPI.$\overline{\text{HAS}}$ |
| HC1 | PGPIO45 | HPI.HBIL |

### 3.3.3 External Bus Selection Register (XBSR)

The External Bus Selection Register controls the mode of the Parallel Port Mux and Host Port Mux. The Parallel Port Mux can be configured to support the 32-bit EMIF or to support parallel general-purpose I/O. The Host Port Mux can be configured to support the HPI in 8-bit (multiplexed) mode or parallel general-purpose I/O (PGPIO).

The XBSR configures the Parallel Port Mux and the Host Port Mux at reset based on the state of the GPIO6 pin at reset. When GPIO6 is high at reset, the Parallel Port Mux will be configured to support the 32-bit EMIF and the Host Port Mux will be configured to support the HPI in 8-bit (multiplexed) mode. When GPIO6 is low at reset, both the Parallel Port Mux and the Host Port Mux will be configured to support parallel general-purpose I/O; the EMIF and HPI will be disabled in this mode. The Paralle/Host Port Mux Mode bit of the XBSR will reflect the mode selected for the Parallel and Host Port Muxes.†

The clock to the EMIF module is disabled automatically when this module is not selected through the External Bus Selection Register. Note that any accesses to disabled modules will result in a bus error if the PERITOEN bit of the Time-Out Control Register is set to 1.

| 15 | 8 |
|---|---|
| Reserved | |
| R, 00000000 | |

| 7 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|
| Reserved | | Reserved (see NOTE) | Reserved (see NOTE) | Reserved | Parallel /Host Port Mux Mode |
| R, 0000 | | R/W, 0 | R/W, 0 | R, 0 | R/W, GPIO6 |

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

NOTE: This reserved bit must be kept as zero during any writes to XBSR.

**Figure 3−3. External Bus Selection Register Layout (0x6C00)**

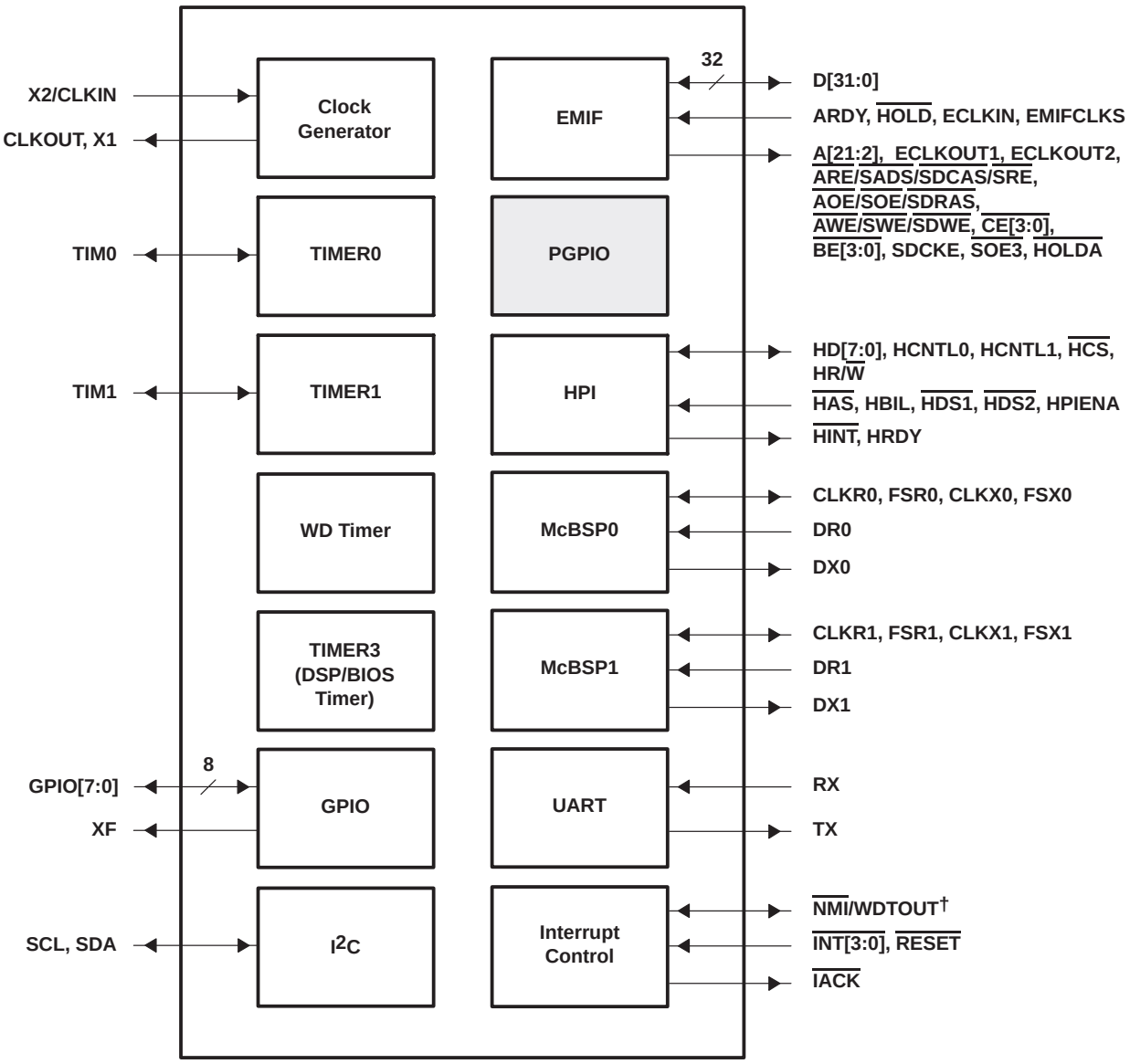
† Modifying the XBSR to change the mode of the Parallel Port Mux and Host Port Mux after the 5501 has been brought out of reset is not supported.

**Table 3−6. External Bus Selection Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−4 | R | 000000000000 | Reserved |
| Reserved | 3 | R/W | 0 | Reserved. This reserved bit must be kept as zero during any writes to XBSR. |
| Reserved | 2 | R/W | 0 | Reserved. This reserved bit must be kept as zero during any writes to XBSR. |
| Reserved | 1 | R | 0 | Reserved |
| Parallel/Host Port Mux Mode | 0 | R/W | GPIO6 | Parllel/Host Port Mux Mode bit. Determines the mode of the Parallel Port Mux and the Host Port Mux.<br>• Parallel/Host Port Mux Mode = 0:<br>The Parallel Port Mux is configured to support PGPIO. In this mode, the HPI and EMIF cannot be used.<br>The Host Port Mux is configured to support PGPIO. In this mode, the Host Port Mux pins will be routed to PGPIO.<br>• Parallel/Host Port Mux Mode = 1:<br>The Parallel Port Mux is configured to support the 32-bit EMIF. In this mode, the EMIF is enabled and its 20 address, 32 data, and 16 control signals are routed to their corresponding pins on the Parallel Port Mux.<br>The Host Port Mux is configured to support the HPI in 8-bit (multiplexed) mode. In this mode, the HPI is enabled and its eight data/address and two control signals are routed to their corresponding pins on the Host Port Mux. |

### 3.3.4 Configuration Examples

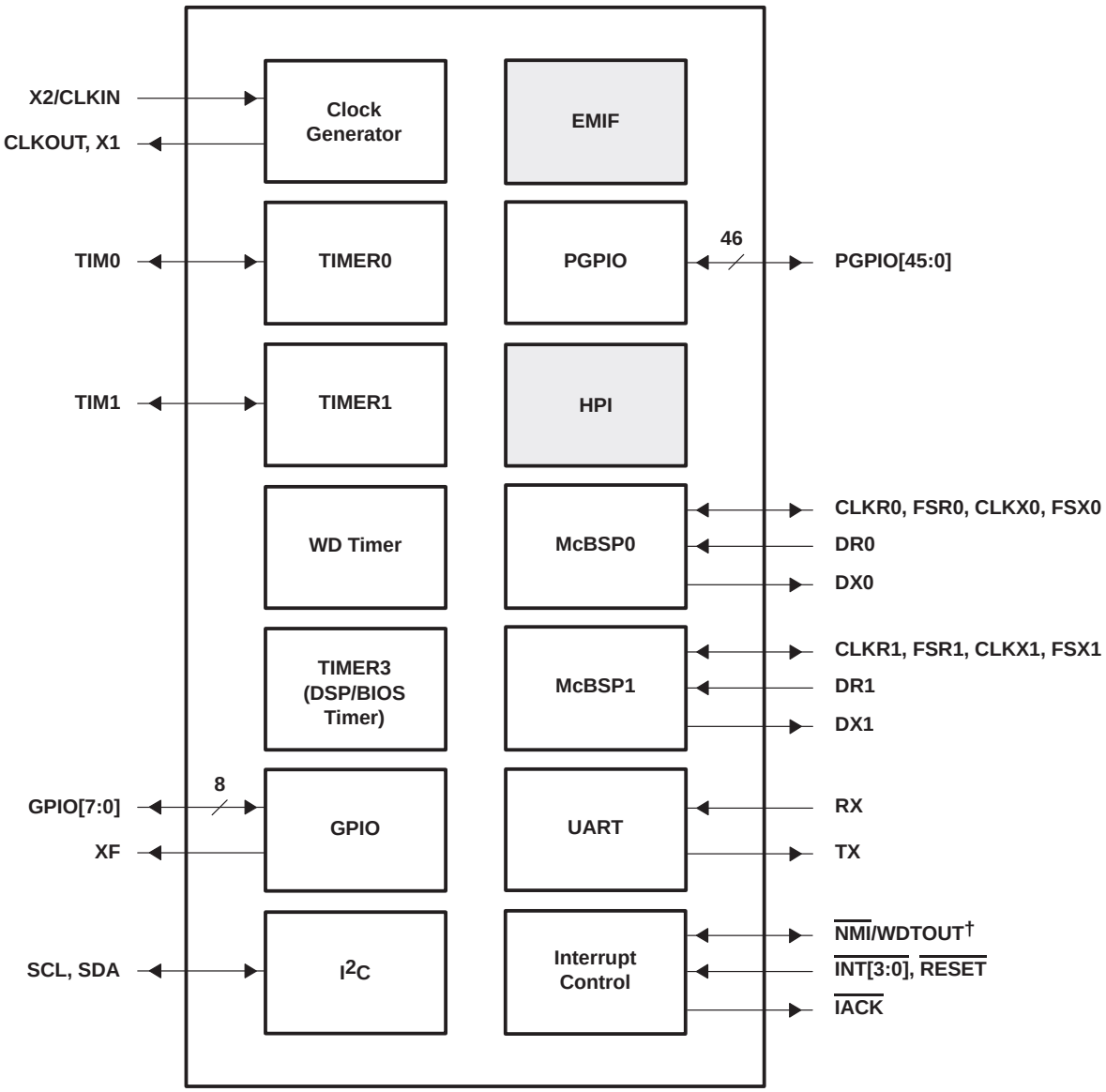
Figure 3−4 and Figure 3−5 illustrate example configurations for the 5501 based on the state of GPIO6 at reset.




☐ Shading denotes a peripheral module not available for this configuration.

† The NMI/WDTOUT pin has NMI function by default, but can be set to WDTOUT through the TSSR.

**Figure 3−4. Configuration Example A (GPIO6 = 1 at Reset)**




☐ Shading denotes a peripheral module not available for this configuration.

† The NMI/WDTOUT pin has NMI function by default, but can be set to WDTOUT through the TSSR.

**Figure 3−5. Configuration Example B (GPIO6 = 0 at Reset)**

## 3.4 Timers

The 5501 has four 64-bit timers: Timer 0, Timer 1, Watchdog Timer (WDT), and Timer 3. The first two timers, Timer 0 and Timer 1, are mainly used as general-purpose timers. The third timer, the Watchdog Timer, can be used as either a general-purpose timer or a watchdog timer. The fourth timer is reserved as a DSP/BIOS counter; users have no access to this timer.

Each timer has one input, one output, and one interrupt signal: TIN, TOUT, and TINT, respectively. Timer 0, Timer 1, and the Watchdog Timer are each assigned a pin: TIM0 pin is assigned to Timer 0, TIM1 is assigned to Timer 1, and $\overline{\text{NMI}}$/WDTOUT is used by the Watchdog Timer. The input (TIN) or output (TOUT) signal of Timer 0, Timer 1, and the Watchdog Timer can be connected to their respective pins via the Timer Signal Selection Register (TSSR).

The DSP/BIOS timer input, output, and interrupt signals are not internally connected. No interrupts are needed from this timer; therefore, the timer interrupt signal is not internally connected to the CPU interrupt logic.

The interrupt signal (TINT) of the Watchdog Timer can be internally connected to the $\overline{\text{NMI}}$, $\overline{\text{RESET}}$, and $\overline{\text{INT3}}$ signals via the TSSR.

Note that the $\overline{\text{NMI}}$/WDTOUT pin has a dual function: Watchdog Timer pin and $\overline{\text{NMI}}$ input pin. The function of the $\overline{\text{NMI}}$/WDTOUT pin can be selected through the TSSR.

For more information on the 5501 timers, see the *TMS320VC5501/5502 DSP Timers Reference Guide* (literature number SPRU618).

### 3.4.1 Timer Interrupts

As stated earlier, each timer has one input, one output, and one interrupt signal: TIN, TOUT, and TINT, respectively. The interrupt signals of Timer 0 and Timer 1 are directly connected to the interrupt logic of the DSP (see Figure 3−6). The interrupts for Timer 0 and Timer 1 are maskable and can be enabled or disabled through the TINT0 and TINT1 bits of the interrupt enable registers (IER0 and IER1); setting TINT0 of IER0 to ‘1’ enables the interrupt for Timer 0 and setting TINT1 of IER1 enables the interrupt for Timer 1.

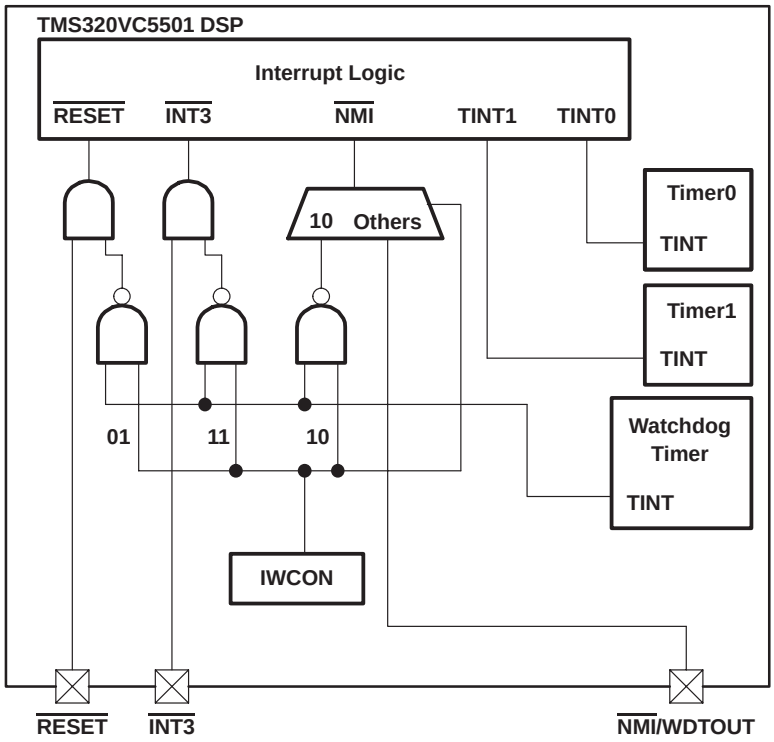



**Figure 3−6. Timer Interrupts**

The interrupt signal for the Watchdog Timer can be internally connected to the $\overline{\text{RESET}}$, $\overline{\text{INT3}}$, or $\overline{\text{NMI}}$ signals by setting the IWCON bit of the Timer Signal Selection Register (TSSR) appropriately (see Figure 3−6). The DSP will be reset once the Watchdog Timer generates an interrupt if the timer interrupt is connected to $\overline{\text{RESET}}$ (IWCON = ‘01’). A non-maskable interrupt will be generated if the timer interrupt is connected to $\overline{\text{NMI}}$ (IWCON = ‘10’). An external interrupt will be generated when the timer interrupt signal is connected to $\overline{\text{INT3}}$ (IWCON = ‘11’), but only if the INT3 bit of IER0 is set to ‘1’.

Refer to Section 3.16, Interrupts, for more information on using interrupts.

### 3.4.2 Timer Pins

The 5501 has one pin for each timer: TIM0 for Timer 0, TIM1 for Timer 1, and $\overline{\text{NMI}}$/WDTOUT for the Watchdog Timer. Either the output (TOUT) or input (TIN) signal can be connected to the timer pin (see Figure 3−7). When the timer pin is configured as an output, the TOUT signal is connected to the pin. The TIN signal is connected to the pin when the pin is configured as an input. Each pin can be configured as input or output through the Timer Signal Selection Register (TSSR) (bits TIM0_MODE, TIM1_MODE, and WDT_MODE).

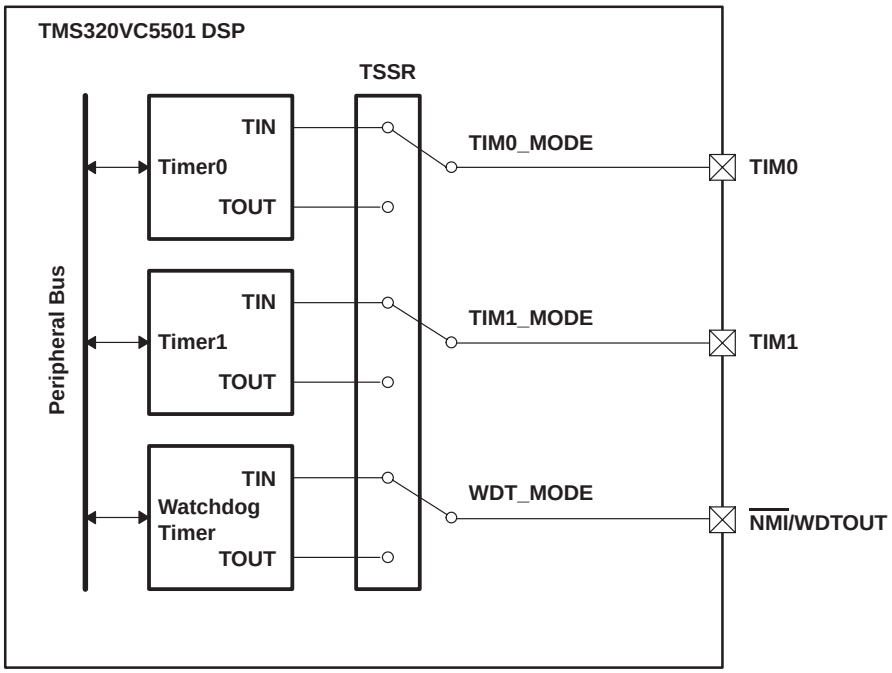



**Figure 3−7. Timer Pins**

When configured as input, the timer pin can be used to source an external clock to the timer. Also, when the timer pin is configured as input and the timer is running off an internal clock, the timer pin can be used to start or stop count of the timer (clock gating).

When the timer pin is configured as an output, the timer pin can signal a pulse (pulse mode) or a change of state (clock mode) when the timer count matches its period.

The $\overline{\text{NMI}}$/WDTOUT pin has two functions: Watchdog Timer pin or $\overline{\text{NMI}}$ pin. The NMI/WDTOUT_CFG bit of the TSSR controls the function of this pin. It is possible to configure the $\overline{\text{NMI}}$/WDTOUT pin as $\overline{\text{NMI}}$ (NMI/WDTOUT_CFG = ‘1’) and also connect the Watchdog Timer TINT signal to the $\overline{\text{NMI}}$ signal (IWCON = ‘10’). In this case, the external $\overline{\text{NMI}}$ signal will be overridden by the TINT signal of the Watchdog Timer, i.e., applying a signal to the $\overline{\text{NMI}}$/WDTOUT pin will *not* generate the non-maskable interrupt $\overline{\text{NMI}}$.

For all three timers (Timer 0, Timer 1, and the Watchdog Timer), both the TIN and TOUT signals can be used for general-purpose input/output. The timer pin must be configured for input to use the TIN signal as general-purpose input/output. The timer pin can be configured as an input by setting the pin mode bit of the Timer Signal Selection Register (TSSR) to ‘0’. The TOUT signal can be used as general-purpose input/output if the timer pin is configured for output. The timer pin can be configured as an output by setting the pin mode bit of the TSSR to ‘1’. The GPIO Enable Register (GPEN), GPIO Direction Register (GPIODIR), and the GPIO Data Register (GPDAT) of each timer can be used to control the state of the timer pins when used as general-purpose input/output.

### 3.4.3 Timer Signal Selection Register (TSSR)

The Timer Signal Selection Register (TSSR) controls several pin characteristics for Timer 0, Timer 1, and the Watchdog Timer. The TSSR can be used to specify whether the pins of Timer 0, Timer 1, and the Watchdog Timer are inputs or outputs. The TSSR also determines how the interrupt signal of the Watchdog Timer is connected internally and sets the function for the $\overline{\text{NMI}}$/WDTOUT pin of the 5501. By default, all timer pins (TIM0, TIM1, and $\overline{\text{NMI}}$/WDTOUT) are set as inputs, the interrupt signal of the Watchdog Timer is not internally connected to anything, and the $\overline{\text{NMI}}$/WDTOUT pin has the function of the $\overline{\text{NMI}}$ signal.

| 15 – 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Reserved | Reserved | | WDT_MODE | TIM1_MODE | TIM0_MODE | IWCON | | NMI/WDTOUT_CFG |
| R, 00000000 | R, 00 | | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 00 | | R/W, 1 |

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

**Figure 3−8. Timer Signal Selection Register Layout (0x8000)**

**Table 3−7. Timer Signal Selection Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−6 | R | 0000000000 | Reserved |
| WDT_MODE | 5 | R/W | 0 | WDT pin mode<br>WDT_MODE = 0: WDTOUT pin is used as the timer input pin.<br>WDT_MODE = 1: WDTOUT pin is used as the timer output pin. |
| TIM1_MODE | 4 | R/W | 0 | TIM1 pin mode<br>TIM1_MODE = 0: TIM1 pin is used as the timer input pin.<br>TIM1_MODE = 1: TIM1 pin is used as the timer output pin. |
| TIM0_MODE | 3 | R/W | 0 | TIM0 pin mode<br>TIM0_MODE = 0: TIM0 pin is used as the timer input pin.<br>TIM0_MODE = 1: TIM0 pin is used as the timer output pin. |

† If NMI/WDTOUT_CFG = 1 and IWCON = 10, only the WDTOUT signal will drive the $\overline{\text{NMI}}$ signal; the external source driving the $\overline{\text{NMI}}$/WDTOUT pin will be ignored.

**Table 3−7. Timer Signal Selection Register Bit Field Description (Continued)**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IWCON | 2:1 | R/W | 00 | Internal WDT output signal connection<br><br>IWCON = 00: Internal watchdog timer interrupt (TINT) signal has no internal connection.<br>IWCON = 01: Internal watchdog timer interrupt (TINT) signal has an internal connection to $\overline{\text{RESET}}$ pin.<br>IWCON = 10: Internal watchdog timer interrupt (TINT) signal has an internal connection to $\overline{\text{NMI}}$ pin.†<br>IWCON = 11: Internal watchdog timer interrupt (TINT) signal has an internal connection to $\overline{\text{INT3}}$ pin. |
| NMI/WDTOUT_CFG | 0 | R/W | 1 | $\overline{\text{NMI}}$/WDTOUT configuration<br><br>NMI/WDTOUT_CFG = 0: $\overline{\text{NMI}}$/WDTOUT pin is used as the $\overline{\text{WDTOUT}}$ pin.<br>NMI/WDTOUT_CFG = 1: $\overline{\text{NMI}}$/WDTOUT pin is used as the $\overline{\text{NMI}}$ input pin.† |

† If NMI/WDTOUT_CFG = 1 and IWCON = 10, only the WDTOUT signal will drive the $\overline{\text{NMI}}$ signal; the external source driving the $\overline{\text{NMI}}$/WDTOUT pin will be ignored.

## 3.5 Universal Asynchronous Receiver/Transmitter (UART)

The UART peripheral is based on the industry-standard TL16C550B asynchronous communications element, which in turn, is a functional upgrade of the TL16C450. Functionally similar to the TL16C450 on power up (character or TL16C450 mode), the UART can be placed in an alternate FIFO (TL16C550) mode. This relieves the CPU of excessive software overhead by buffering received and transmitted characters. The receiver and transmitter FIFOs store up to 16 bytes, including three additional bits of error status per byte for the receiver FIFO.

The UART performs serial-to-parallel conversions on data received from a peripheral device or modem and parallel-to-serial conversion on data received from the CPU. The CPU can read the UART status at any time. The UART includes control capability and a processor interrupt system that can be configured to minimize software management of the communications link.

The UART includes a programmable baud rate generator capable of dividing the CPU clock by divisors from 1 to 65535 and producing a 16× reference clock for the internal transmitter and receiver logic.

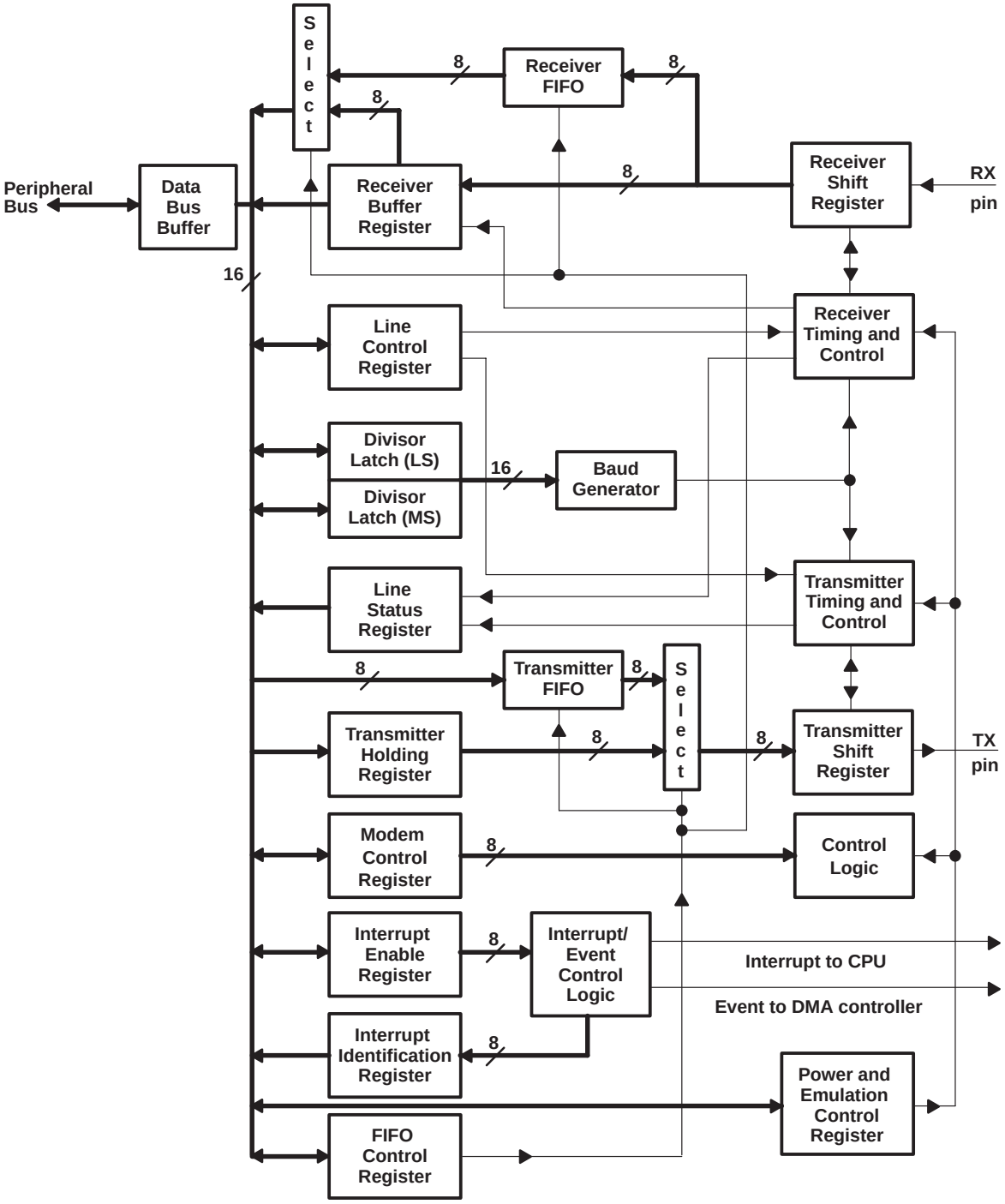



**Figure 3−9. UART Functional Block Diagram**

## 3.6 Inter-Integrated Circuit (I2C) Module

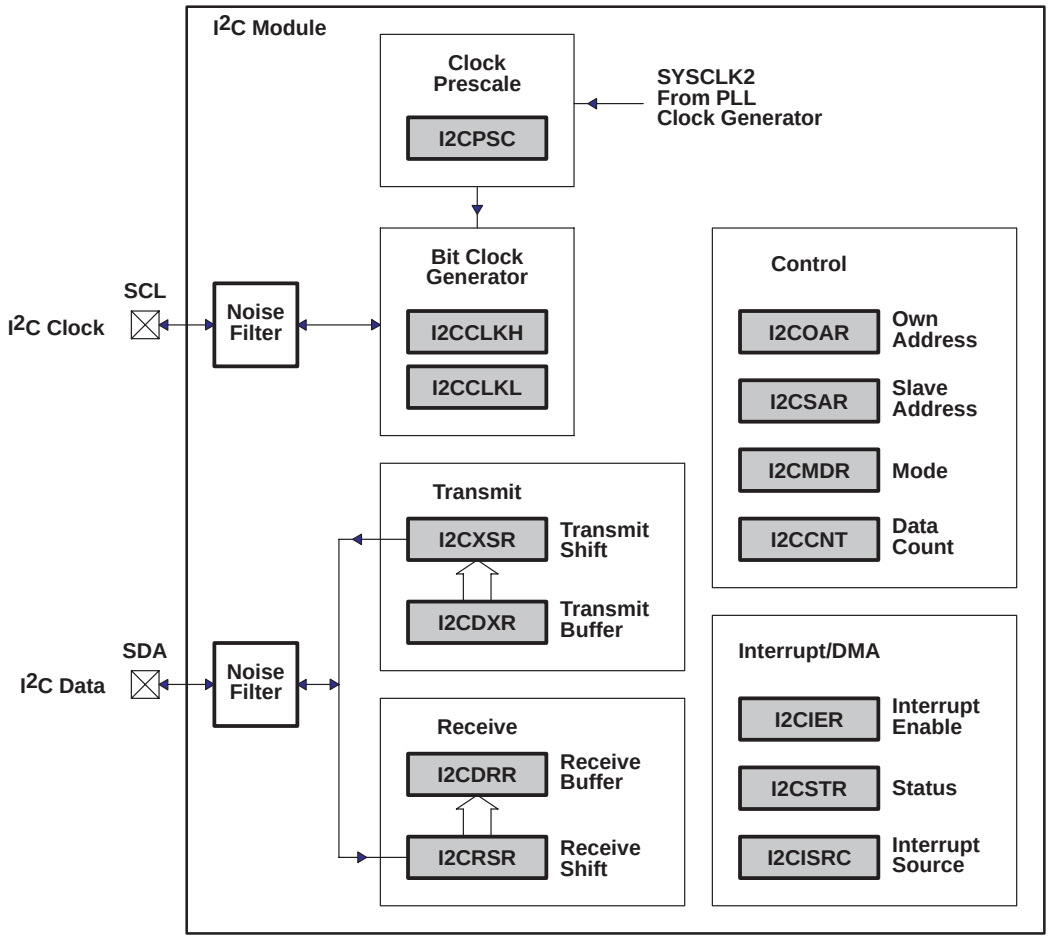
The TMS320VC5501 also includes an I2C serial port for control purposes. Features of the I2C port include:

- Compatibility with Philips' I2C-Bus Specification, Version 2.1 (January 2000)
- Fast mode up to 400 Kbps (no fail-safe I/O buffers)
- Noise filters (on the SDA and SCL pins) to suppress noise of 50 ns or less (I2C module clock *must* be in the range of 7 MHz to 12 MHz)
- 7-bit and 10-bit device addressing modes
- Master (transmit/receive) and slave (transmit/receive) functionality
- Events: DMA, interrupt, or polling
- Slew-rate limited open-drain output buffers

The I2C module clock *must* be in the range of 7 MHz to 12 MHz. This is necessary for the proper operation of the I2C module.

**NOTE:** For additional information, see the *TMS320VC5501/5502/5503/5507/5509 DSP Inter-Integrated Circuit (I2C) Module Reference Guide* (literature number SPRU146).

Figure 3−10 is a block diagram of the I2C module.




NOTE A: Shading denotes control/status registers.

**Figure 3−10. I2C Module Block Diagram**

## 3.7 Host-Port Interface (HPI)

The 5501 HPI provides an 8-bit parallel interface (multiplexed mode) to a host with the following features:

- Host access to on-chip DARAM (excluding CPU memory-mapped registers)
- 16-bit address register with autoincrement capability for faster transfers
- Multiple address/data strobes provide a glueless interface to a variety of hosts
- HRDY signal for handshaking with host

The 5501 HPI can access the entire DARAM space of the 5501 (excluding memory-mapped CPU registers); however, it does not have access to external memory of the peripheral I/O space. Furthermore, the HPI cannot access internal DARAM space when the device is in reset. Note that all accesses made through the HPI are word-addressed.

**NOTE:** No host access should occur when the HPI is placed in IDLE. The host cannot wake up the DSP through the DSP_INT bit of the HPIC register when the DSP is in IDLE mode.

The 5501 HPI only supports data transfers in multiplexed 8-bit mode. In multiplexed mode, the host can only send 8 bits of data at a time through the HD[7:0] bus; therefore, some extra steps have to be taken to read/write from the 5501's internal memory [see the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620) for more information on the 5501 HPI].

The 5501 HPI has its own register set, therefore the HINT bit of CPU register ST3_55 is not used for DSP-to-host interrrupts. The HINT bit in the Host Port Control Register (HPIC) should be used for DSP-to-host interrupts.

A host must not initiate any transfer requests from the HPI while the HPI is being brought out of reset. As described in Section 3.9.6, Reset Sequence, the C55x CPU and the peripherals are not brought out of reset immediately after the $\overline{\text{RESET}}$ pin transitions from low to high. Instead, an internal counter stretches the reset signal to allow enough time for the internal oscillator to stabilize and also to allow the reset signal to propagate through different parts of the device. The $\overline{\text{IACK}}$ pin will go low for two CPU clock cycles to indicate that this internal reset signal has been deasserted. A host must follow one of these two requirements before initiating transfer requests from the HPI:

1. Keep the HPIENA pin low until the internal reset signal has been deasserted.
2. Keep the $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$ pins inactive until the internal reset signal has been deasserted.

Note that when the HPI bootmode is used, the GPIO4 pin can also be used to determine when the internal reset signal has been deasserted as this pin is used by the HPI to signal to the host that it is ready to receive access requests.

## 3.8 Direct Memory Access (DMA) Controller

The 5501 DMA provides the following features:

- Four standard ports for the following data resources: two for DARAM, one for Peripherals, and one for External Memory
- Six channels, which allow the DMA controller to track the context of six independent DMA channels
- Programmable low/high priority for each DMA channel
- One interrupt for each DMA channel
- Event synchronization. DMA transfers in each channel can be dependent on the occurrence of selected events.
- Programmable address modification for source and destination addresses
- Idle mode that allows the DMA controller to be placed in a low-power (idle) state under software control

The 5501 has an Auto-wakeup/Idle function for McBSP to DMA to on-chip memory data transfers when the DMA and the McBSP are both set to IDLE. In the case that the McBSP is set to external clock mode and the McBSP and the DMA are set to idle, the McBSP and the DMA can wake up from IDLE state automatically if the McBSP gets a new data transfer. The McBSP and the DMA enter the idle state automatically after data transfer is complete. [The clock generator (PLL) should be active and the PLL core should not be in power-down mode for the Auto-wakeup/Idle function to work.]

The 5501 DMA controller allows transfers to be synchronized to selected events. The 5501 supports 14 separate synchronization events and each channel can be tied to separate synchronization event independent of the other channels. Synchronization events are selected by programming the SYNC field in the channel-specific DMA Channel Control Register (DMA_CCR).

The 5501 DMA can access all the internal DARAM space as well as all external memory space. The 5501 DMA also has access to the registers for the following peripheral modules: McBSP, UART, GPIO, PGPIO, and $I^2C$.

### *3.8.1 DMA Channel 0 Control Register (DMA_CCR0)*

The DMA Channel 0 Control Register (DMA_CCR0) bit layouts are shown in Figure 3−11. DMA_CCR1 to DMA_CCR5 have similar bit layouts. See the *TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide* (literature number SPRU613) for more information on the DMA Channel n Control Register (n = 0, 1, 2, 3, 4, or 5).

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| DSTAMODE | | SRCAMODE | | ENDPROG | WP | REPEAT | AUTOINIT |
| R/W, 00 | | R/W, 00 | | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |

| 7 | 6 | 5 | 4 | 0 |
|---|---|---|---|---|
| EN | PRIO | FS | SYNC | |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 00000 | |

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

**Figure 3−11. DMA Channel 0 Control Register Layout (0x0C01)**

The SYNC field (bits[4:0]) of the DMA_CCR register specifies the event that can initiate the DMA transfer for the corresponding DMA channel. The five bits allow several configurations as listed in Table 3−8. The bits are set to zero upon reset.

**Table 3−8. Synchronization Control Function**

| SYNC FIELD IN DMA_CCR | SYNCHRONIZATION MODE |
|---|---|
| 00000b | No event synchronized |
| 00001b | McBSP 0 Receive Event (REVT0) |
| 00010b | McBSP 0 Transmit Event (XEVT0) |
| 00011b | Reserved (Do not use this value) |
| 00100b | Reserved (Do not use this value) |
| 00101b | McBSP1 Receive Event (REVT1) |
| 00110b | McBSP1 Transmit Event (XEVT1) |
| 00111b | Reserved (Do not use this value) |
| 01000b | Reserved (Do not use this value) |
| 01001b | Reserved (Do not use this value) |
| 01010b | Reserved (Do not use this value) |
| 01011b | UART Receive Event (UARTREVT) |
| 01100b | UART Transmit Event (UARTXEVT) |
| 01101b | Timer 0 Event |
| 01110b | Timer 1 Event |
| 01111b | External Interrupt 0 |
| 10000b | External Interrupt 1 |
| 10001b | External Interrupt 2 |
| 10010b | External Interrupt 3 |
| 10011b | $I^2C$ Receive Event |
| 10100b | $I^2C$ Transmit Event |
| Other values | Reserved (Do not use these values) |

## 3.9 System Clock Generator

The TMS320VC5501 includes a flexible clock generator module consisting of a PLL and oscillator, with several dividers so that different clocks may be generated for different parts of the system (i.e., 55x core, Fast Peripherals, Slow Peripherals, External Memory Interface). Figure 3−12 provides an overview of the system clock generator included in the 5501.

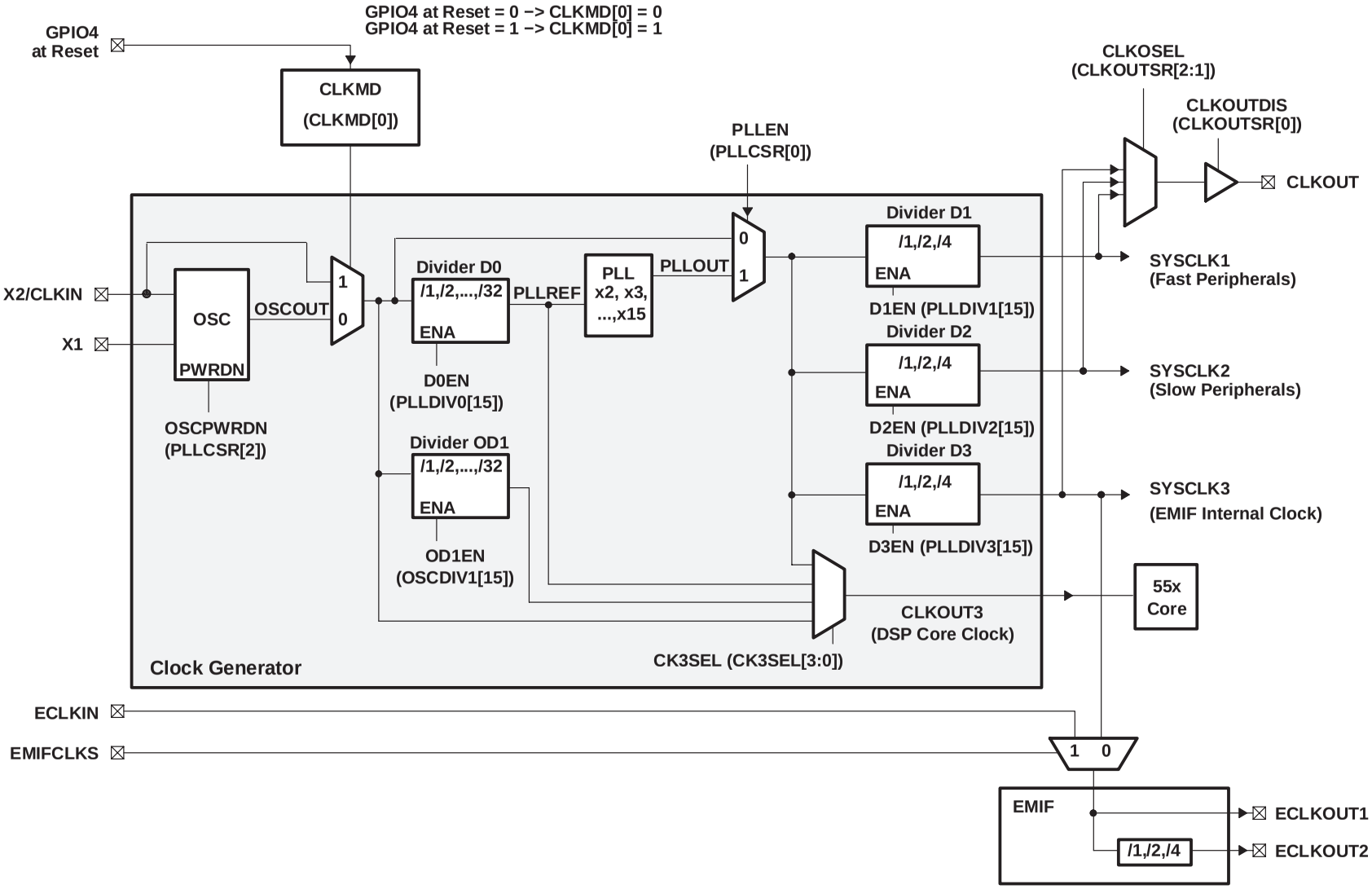



**Figure 3−12. System Clock Generator**

### 3.9.1 Input Clock Source

The clock input to the 5501 can be sourced from either an externally generated 3.3-V clock input on the X2/CLKIN pin, or from the on-chip oscillator if an external crystal circuit is attached to the device as shown in Figure 3−13. The CLKMD0 bit of the Clock Mode Control Register (CLKMD) determines which clock, either OSCOUT or X2/CLKIN, is used as an input clock source to the DSP. If GPIO4 is low at reset, the CLKMD0 bit of the Clock Mode Control Register (CLKMD) will be set to ‘0’ and the internal oscillator and the external crystal will generate the input clock to the DSP. If GPIO4 is high, the CLKMD0 bit will be set to ‘1’ and the input clock will be taken directly from the X2/CLKIN pin.

The input clock source to the DSP can be directly used to generate the clocks to other parts of the system (Bypass Mode) or it can be multiplied by a value from 2 to 15 and divided by a value from 1 to 32 to achieve a desired frequency (PLL Mode). The PLLEN bit of the PLL Control/Status Register (PLLCSR) is used to select between the PLL and bypass modes of the clock generator.

The clock generated through either the PLL Mode or the Bypass Mode can be further divided down to generate a clock source for other parts of the system, or Clock Groups. Clock groups allow for lower power and performance optimization since the frequency of groups with no high-speed requirements can be set to one-fourth or one-half the frequency of other groups. A description of the different clock groups included in the 5501 and the procedure for changing the operating frequency for those clock groups are described later in this section.

#### 3.9.1.1 Internal System Oscillator With External Crystal

The 5501 includes an internal oscillator which can be used in conjunction with an external crystal to generate the input clock to the DSP. The oscillator requires an external crystal connected across the X1 and X2/CLKIN pins. If the internal oscillator is not used, an external clock source must be applied to the X2/CLKIN pin and the X1 pin should be left unconnected. Since the internal oscillator can be used as a clock source to the PLL, the crystal oscillation frequency can be multiplied to generate the input clock to the different clock groups of the DSP.

The crystal should be in fundamental-mode operation, and parallel resonant, with a maximum effective series resistance (ESR) as specified in Table 3−9. The connection of the required circuit is shown in Figure 3−13. Under some conditions, all the components shown are not required. The capacitors, $C_1$ and $C_2$, should be chosen such that the equation below is satisfied. $C_L$ in the equation is the load specified for the crystal that is also specified in Table 3−9.

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

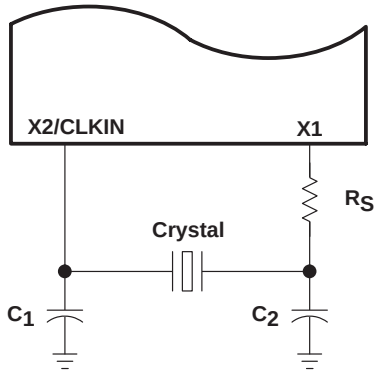



**Figure 3−13. Internal System Oscillator With External Crystal**

**Table 3−9. Recommended Crystal Parameters**

| FREQUENCY RANGE (MHz) | MAXIMUM ESR SPECIFICATIONS (Ω) | $C_{LOAD}$ (pF) | MAXIMUM $C_{SHUNT}$ (pF) | $R_S$ (kΩ) |
|---|---|---|---|---|
| 20−15 | 40 | 10 | 7 | 0 |
| 15−12 | 40 | 16 | 7 | 0 |
| 12−10 | 40 | 16 | 7 | 2.8 |
| 10−8 | 60 | 18 | 7 | 2.2 |
| 8−6 | 60 | 18 | 7 | 8.8 |
| 6−5 | 80 | 18 | 7 | 14 |

The recommended ESR is presented as a maximum, and theoretically, a crystal with a lower maximum ESR might seem to meet these specifications. However, it is recommended that crystals with actual maximum ESR specifications as shown in Table 3−9 be used since this will result in maximum crystal performance reliability.

The internal oscillator can be set to power-down mode through the use of the OSCPWRDN bit in the PLL Control/Status Register (PLLCSR). If the internal oscillator and the external crystal are generating the input clock for the DSP (CLKMD0 = 0), the internal oscillator will be set to power-down mode when the OSCPWRDN bit is set to 1 and the clock generator is set to its idle mode (CLKIS bit of the IDLE Status Register (ISTR) becomes 1). If the X2/CLKIN pin is supplying the input clock to the DSP (CLKMD0 = 1), the internal oscillator will be set to power-down immediately after the OSCPWRDN bit is set to 1.

The 5501 has internal circuitry that will count down a predetermined number of clock cycles (41,032 reference clock cycles) to allow the oscillator input to become stable after waking up from power-down state or after reset. If a reset is asserted, program flow will start after all stabilization periods have expired; this includes the oscillator stabilization period only if GPIO4 is low at reset. If the oscillator is coming out of power-down mode, program flow will start immediately after the oscillator stabilization period has completed. See Section 3.9.6, Reset Sequence, for more details on program flow after reset or after oscillator power-down. See Section 3.10, Idle Control, for more information on the oscillator power-down mode.

### 3.9.1.2 Clock Generation With PLL Disabled (Bypass Mode, Default)

After reset, the PLL multiplier (M1) and its divider (D0) will be bypassed by default and the input clock to point C in Figure 3−14 will be taken from, depending on the state of the GPIO4 pin after reset, either the internal oscillator or the X2/CLKIN pin. The PLL can be taken out of bypass mode as described in Section 3.9.4.1, C55x Subsystem Clock Group.

### 3.9.1.3 Clock Generation With PLL Enabled (PLL Mode)

When not in bypass mode, the frequency of the input clock can be divided down by a programmable divider (D0) by any factor from 1 to 32. The output clock of the divider can be multiplied by any factor from 2 to 15 through a programmable multiplier (M1). The divider factor can be set through the PLLDIV0 bit of the PLL Divider 0 Register. The multiplier factor can be set through the PLLM bits of the PLL Multiplier Control Register.

There is a specific minimum and maximum reference clock (PLLREF) and output clock (PLLOUT) for the block labeled “PLL” in Figure 3−12, as well as for the C55x Core clock (CLKOUT3), the Fast Peripherals clock (SYSCLK1), the Slow Peripherals clock (SYSCLK2), and the EMIF internal clock (SYSCLK3). The clock generator must not be configured to exceed any of these constraints (certain combinations of external clock input, internal dividers, and PLL multiply ratios might not be supported). See Table 3−10 for the PLL clock input and output frequency ranges.

#### 3.9.1.4 Frequency Ranges for Internal Clocks

There are specific minimum and maximum reference clocks for all of the internal clocks. Table 3−10 lists the minimum and maximum frequencies for the internal clocks of the TMS320VC5501.

**Table 3−10. Internal Clocks Frequency Ranges†**

| CLOCK SIGNAL | MIN | MAX | UNIT |
|---|---|---|---|
| OSCOUT (CLKMD = 0) | 5 | 20 | MHz |
| PLLREF (PLLEN = 1) | 12 | 100 | MHz |
| PLLOUT (PLLEN = 1) | 70 | 300 | MHz |
| CLKOUT3 | − | 300 | MHz |
| SYSCLK1 | − | 150 | MHz |
| SYSCLK2 | − | SYSCLK1 | MHz |
| SYSCLK3 | − | SYSCLK1‡ | MHz |

† Also see the electrical specification (timing requirements and switching characteristics parameters) in Section 5.6, Clock Options, of this data manual.

‡ When an internal clock is used for the EMIF module, the frequency for SYSCLK3 must also be less than or equal to 100 MHz. When an external clock is used, the maximum frequency of SYSCLK3 can be equal to or less than the frequency of SYSCLK1; however, the frequency of the clock signal applied to the ECLKIN pin must be less than or equal to 100 MHz.

### *3.9.2 Clock Groups*

The TMS320VC5501 has four clock groups: the C55x Subsystem Clock Group, the Fast Peripherals Clock Group, the Slow Peripherals Clock Group, and the External Memory Interface Clock Group. Clock groups allow for lower power and performance optimization since the frequency of groups with no high-speed requirements can be set to 1/4 or 1/2 the frequency of other groups.

#### 3.9.2.1 C55x Subsystem Clock Group

The C55x Subsystem Clock Group includes the C55x CPU core, internal memory (DARAM and ROM), the ICACHE, and all CPU-related modules. The input clock to this clock group is taken from the CLKOUT3 signal (as shown in Figure 3−12), the source of which can be controlled through the CLKOUT3 Select Register (CK3SEL). The different options for the CLKOUT3 signal are intended for test purposes; it is recommended that the CK3SEL bits of the CK3SEL register be kept at their default value of ‘1011b’ during normal operation. When operating the clock generator in PLL Mode, the frequency of CLKOUT3 can be set by adjusting the divider and multiplier values of D0 and M1 through the PLLDIV0 and PLLM registers, respectively.

#### 3.9.2.2 Fast Peripherals Clock Group

The Fast Peripherals Clock Group includes the DMA, HPI, and the timers. The input clock to this clock group is taken from the output of divider 1 (D1) (as shown in Figure 3−12). By default, the divider is set to divide its input clock by four, but the divide value can be changed to divide-by-1 or divide-by-2 by modifying the PLLDIV1 bits of the PLL Divider1 Register (PLLDIV1) through software.

#### 3.9.2.3 Slow Peripherals Clock Group

The Slow Peripherals Clock Group includes the McBSPs, $I^2C$, and the UART. The input clock to this clock group is taken from the output of divider 2 (D2). by default, the divider is set to divide its input clock by four, but the divide value can be changed to divide-by-1 or divide-by-2 by modifying the PLLDIV2 bits of the PLL Divider2 Register (PLLDIV2) through software. *The clock frequency of the Slow Peripherals Clock Group must be equal to or less than that of the Fast Peripherals Clock Group.*

#### 3.9.2.4 External Memory Interface Clock Group

The External Memory Interface Clock Group includes the External Memory Interface (EMIF) module and the external data bridge modules. The input clock to this clock group is taken from the output of divider 3 (D3). By default, the divider is set to divide its input clock by four, but the divide value can be changed to divide-by-1 or divide-by-2 by modifying the PLLDIV3 bits of the PLL Divider3 Register (PLLDIV3) through software. *The clock frequency of the External Memory Interface Clock Group must be equal to or less than that of the Fast Peripherals Clock Group.*

### 3.9.3 EMIF Input Clock Selection

The EMIF may be clocked from an external asynchronous clock source through the ECLKIN pin if a specific EMIF frequency is needed. The source for the EMIF clock can be specified through the EMIFCLKS pin. If EMIFCLKS is low, then the EMIF will be clocked via the same internal clock that feeds the data bridge module and performance will be optimal. If EMIFCLKS is high, then an external asynchronous clock, which can be taken up to 100 MHz, will clock the EMIF. The data throughput performance may be degraded due to synchronization issues when an external clock source is used for the EMIF.

### 3.9.4 Changing the Clock Group Frequencies

DSP software can be used to change the clock frequency of each clock group by setting adequate values in the PLL control registers. Figure 3−14 shows which PLL control registers affect the different portions of the clock generator. The following sections describe the procedures for changing the frequencies of each clock group.

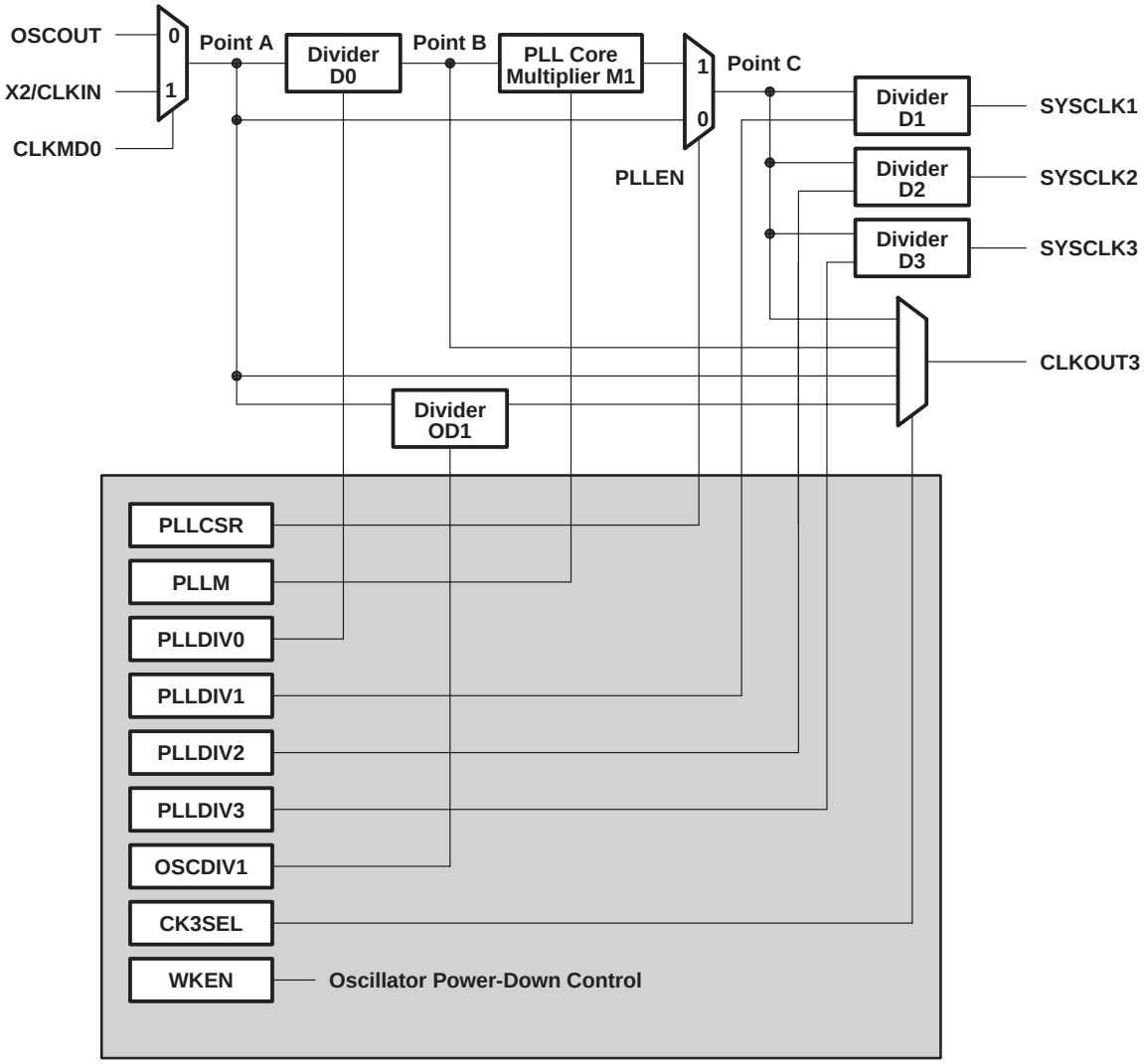



**Figure 3−14. Clock Generator Registers**

#### 3.9.4.1 C55x Subsystem Clock Group

Changes to the PLL Control Register (PLLCSR), the PLL Divider0 Register (PLLDIV0), and the PLL Multiplier Register (PLLM) affect the clock of this clock group. The following procedure must be followed to change or to set the PLL to a specific value:

1. Switch to bypass mode by setting the PLLEN bit to 0.
2. Set the PLL to its reset state by setting the PLLRST bit to 1.
3. Change the PLL setting through the PLLM and PLLDIV0 bits.
4. Wait for 1 μs.
5. Release the PLL from its reset state by setting PLLRST to 0.
6. Wait for the PLL to relock by polling the LOCK bit or by setting up a LOCK interrupt.
7. Switch back to PLL mode by setting the PLLEN bit to 1.

The frequency of the C55x Subsystem Clock Group can be up to 300 MHz.

#### 3.9.4.2 Fast Peripherals Clock Group

Changes to the clock of the C55x Subsystem Clock Group affect the clock of the Fast Peripherals Clock Group. The PLLDIV1 value of the PLL Divider1 Register (PLLDIV1) should not be set in a manner that makes the frequency for this clock group greater than 150 MHz. There must be no activity in the modules included in the Fast Peripherals Clock Group when the value of PLLDIV1 is being changed. It is recommended that the fast peripheral modules be put in IDLE mode before changing the PLLDIV1 value.

#### 3.9.4.3 Slow Peripherals Clock Group

Changes to the clock of the C55x Subsystem Clock Group affect the clock of the Slow Peripherals Clock Group. The PLLDIV2 value of the PLL Divider2 Register (PLLDIV2) should not be set in a manner that makes the frequency for this clock group greater than 150 MHz or greater than the frequency of the Fast Peripherals Clock Group. There must be no activity in the modules included in the Slow Peripherals Clock Group when the value of PLLDIV2 is being changed. It is recommended that the slow peripheral modules be put in IDLE mode before changing the PLLDIV2 value.

#### 3.9.4.4 External Memory Interface Clock Group

Changes to the clock of the C55x Subsystem Clock Group affect the clock of the External Memory Interface Clock Group. The PLLDIV3 value of the PLL Divider3 Register (PLLDIV3) should not be set in a manner that makes the frequency for this clock group greater than 100 MHz or greater than the frequency of the Fast Peripherals Clock Group, whichever is smaller. If an external clock is used, the clock on the ECLKIN pin can be up to 100 MHz and the output of divider 3 can be set equal to or lower than the frequency of the Fast Peripherals Clock Group. There must be no external memory accesses when the value of PLLDIV3 is being changed, this means that the value of PLLDIV3 cannot be changed by a program that is being executed from external memory. It is recommended that the EMIF be put in IDLE mode before changing the PLLDIV3 value.

### 3.9.5 PLL Control Registers

The 5501 PLL control registers are accessible via the I/O memory map.

**Table 3−11. PLL Control Registers**

| ADDRESS | REGISTER |
|---|---|
| 1C80h | PLLCSR |
| 1C82h | CK3SEL |
| 1C88h | PLLM |
| 1C8Ah | PLLDIV0 |
| 1C8Ch | PLLDIV1 |
| 1C8Eh | PLLDIV2 |
| 1C90h | PLLDIV3 |
| 1C92h | OSCDIV1 |
| 1C98h | WKEN |
| 8400h | CLKOUTSR |
| 8C00h | CLKMD |

#### 3.9.5.1 PLL Control / Status Register (PLLCSR)

| 15 | | | | | | | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R, 00000000 | | | | | | | |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | STABLE | LOCK | Reserved | PLLRST | OSCPWRDN | PLLPWRDN | PLLEN |
| R, 0 | R, 1 | R, 0 | R, 0 | R/W, 1 | R/W, 0 | R/W, 0 | R/W, 0 |

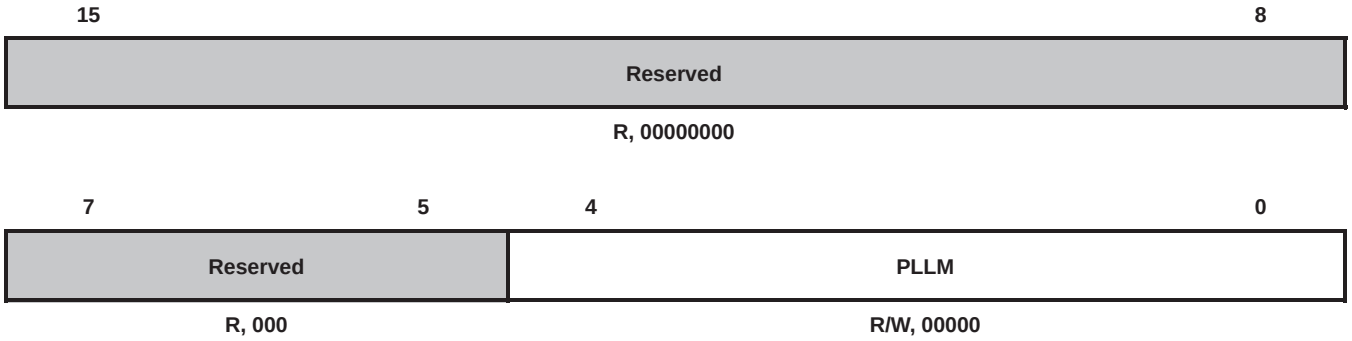
**LEGEND:** R = Read, W = Write, *n* = value at reset

**Figure 3−15. PLL Control/Status Register Layout (0x1C80)**

**Table 3−12. PLL Control/Status Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15:7 | R | 000000000 | Reserved. Reads return 0. Writes have no effect. |
| STABLE | 6 | R | 1 | Oscillator output stable. This bit indicates if the OSCOUT output has stabilized.<br><br>STABLE = 0: Oscillator output is not yet stable. Oscillator counter is not done counting 41,032 reference clock cycles.<br>STABLE = 1: Oscillator output is stable. This is true if any one of the three cases is true:<br>a) Oscillator counter has finished counting.<br>b) Oscillator counter is disabled.<br>c) Test mode. |
| LOCK | 5 | R | 0 | Lock mode indicator. This bit indicates whether the clock generator is in its lock mode.<br><br>LOCK = 0: The PLL is in the process of getting a phase lock.<br>LOCK = 1: The clock generator is in the lock mode. The PLL has a phase lock and the output clock of the PLL has the frequency determined by the PLLM register and PLLDIV0 register. |
| Reserved | 4 | R | 0 | Reserved. Reads return 0. Writes have no effect. |
| PLLRST | 3 | R/W | 1 | Asserts RESET to PLL<br><br>PLLRST = 0: PLL reset released<br>PLLRST = 1: PLL reset asserted |
| OSCPWRDN | 2 | R/W | 0 | Sets internal oscillator to power-down mode<br><br>OSCPWRDN = 0: Oscillator operational<br>OSCPWRDN = 1: Oscillator set to power-down mode based on state of CLKMD0 bit of Clock Mode Control Register (CLKMD).<br><br>When CLKMD0 = 0, the internal oscillator is set to power-down mode when the clock generator is set to its idle mode [CLKIS bit of the IDLE Status Register (ISTR) becomes 1].<br><br>When CLKMD0 = 1, the internal oscillator is set to power-down mode immediately after the OSCPWRDN bit is set to 1. |
| PLLPWRDN | 1 | R/W | 0 | Selects PLL power down<br><br>PLLPWRDN = 0: PLL operational<br>PLLPWRDN = 1: PLL placed in power-down state |
| PLLEN | 0 | R/W | 0 | PLL mode enable. This bit controls the multiplexer before dividers D1, D2, and D3.<br><br>PLLEN = 0: Bypass mode. Divider D1 and PLL are bypassed. SYSCLK1 to 3 divided down directly from input reference clock.<br>PLLEN = 1: PLL mode. Divider D1 and PLL are not bypassed. SYSCLK1 to 3 divided down from PLL output. |

### 3.9.5.2 PLL Multiplier Control Register (PLLM)




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

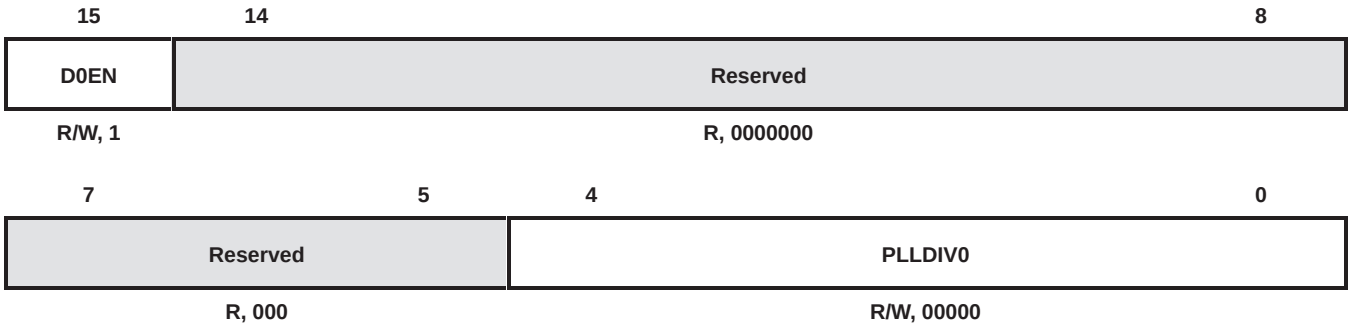
**Figure 3−16. PLL Multiplier Control Register Layout (0x1C88)**

**Table 3−13. PLL Multiplier Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15:5 | R | 00000000000 | Reserved. Reads return 0. Writes have no effect. |
| PLLM | 4:0 | R/W | 00000 | PLL multiplier-select<br><br>PLLM = 00000−00001: Reserved<br>PLLM = 00010: Times 2<br>PLLM = 00011: Times 3<br>PLLM = 00100: Times 4<br>PLLM = 00101: Times 5<br>PLLM = 00110: Times 6<br>PLLM = 00111: Times 7<br>PLLM = 01000: Times 8<br>PLLM = 01001: Times 9<br>PLLM = 01010: Times 10<br>PLLM = 01011: Times 11<br>PLLM = 01100: Times 12<br>PLLM = 01101: Times 13<br>PLLM = 01110: Times 14<br>PLLM = 01111: Times 15<br>PLLM = 10000−11111: Reserved |

### 3.9.5.3 PLL Divider 0 Register (PLLDIV0) (Prescaler)

This register controls the value of the PLL prescaler (Divider D0).




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

**Figure 3−17. PLL Divider 0 Register Layout (0x1C8A)**

**Table 3−14. PLL Divider 0 Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| D0EN | 15 | R/W | 1 | Divider D0 enable<br><br>D0EN = 0: Divider 0 disabled<br>D0EN = 1: Divider 0 enabled |
| Reserved | 14:5 | R | 0000000000 | Reserved. Reads return 0. Writes have no effect. |
| PLLDIV0 | 4:0 | R/W | 00000 | Divider D0 ratio<br><br>PLLDIV0 = 00000: Divide by 1<br>PLLDIV0 = 00001: Divide by 2<br>PLLDIV0 = 00010: Divide by 3<br>PLLDIV0 = 00011: Divide by 4<br>PLLDIV0 = 00100: Divide by 5<br>PLLDIV0 = 00101: Divide by 6<br>PLLDIV0 = 00110: Divide by 7<br>PLLDIV0 = 00111: Divide by 8<br>PLLDIV0 = 01000: Divide by 9<br>PLLDIV0 = 01001: Divide by 10<br>PLLDIV0 = 01010: Divide by 11<br>PLLDIV0 = 01011: Divide by 12<br>PLLDIV0 = 01100: Divide by 13<br>PLLDIV0 = 01101: Divide by 14<br>PLLDIV0 = 01110: Divide by 15<br>PLLDIV0 = 01111: Divide by 16<br>PLLDIV0 = 10000: Divide by 17<br>PLLDIV0 = 10001: Divide by 18<br>PLLDIV0 = 10010: Divide by 19<br>PLLDIV0 = 10011: Divide by 20<br>PLLDIV0 = 10100: Divide by 21<br>PLLDIV0 = 10101: Divide by 22<br>PLLDIV0 = 10110: Divide by 23<br>PLLDIV0 = 10111: Divide by 24<br>PLLDIV0 = 11000: Divide by 25<br>PLLDIV0 = 11001: Divide by 26<br>PLLDIV0 = 11010: Divide by 27<br>PLLDIV0 = 11011: Divide by 28<br>PLLDIV0 = 11100: Divide by 29<br>PLLDIV0 = 11101: Divide by 30<br>PLLDIV0 = 11110: Divide by 31<br>PLLDIV0 = 11111: Divide by 32 |

### 3.9.5.4 PLL Divider1 Register (PLLDIV1) for SYSCLK1

This register controls the value of the divider D1 for SYSCLK1. It is in both the BYPASS and PLL paths.

| 15 | 14 | 8 |
|---|---|---|
| D1EN | Reserved | |
| R/W, 1 | R, 0000000 | |

| 7 | 5 | 4 | 0 |
|---|---|---|---|
| Reserved | | PLLDIV1 | |
| R, 000 | | R/W, 00011 | |

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

**Figure 3−18. PLL Divider 1 Register Layout (0x1C8C)**

**Table 3−15. PLL Divider 1 Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| D1EN | 15 | R/W | 1 | Divider D1 enable<br><br>D1EN = 0: Divider 1 disabled<br>D1EN = 1: Divider 1 enabled |
| Reserved | 14:5 | R | 0000000000 | Reserved. Reads return 0. Writes have no effect. |
| PLLDIV1 | 4:0 | R/W | 00011 | Divider D1 ratio (SYSCLK1 divider)<br><br>PLLDIV1 = 00000: Divide by 1<br>PLLDIV1 = 00001: Divide by 2<br>PLLDIV1 = 00010: Reserved<br>PLLDIV1 = 00011: Divide by 4<br>PLLDIV1 = 00100−11111: Reserved |

### 3.9.5.5 PLL Divider2 Register (PLLDIV2) for SYSCLK2

This register controls the value of the divider D2 for SYSCLK2. It is in both the BYPASS and PLL paths.

| 15 | 14 | 8 |
|---|---|---|
| D2EN | Reserved | |
| R/W, 1 | R, 0000000 | |

| 7 | 5 | 4 | 0 |
|---|---|---|---|
| Reserved | | PLLDIV2 | |
| R, 000 | | R/W, 00011 | |

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

**Figure 3−19. PLL Divider 2 Register Layout (0x1C8E)**

**Table 3−16. PLL Divider 2 Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| D2EN | 15 | R/W | 1 | Divider D2 enable<br><br>D2EN = 0: Divider 2 disabled<br>D2EN = 1: Divider 2 enabled |
| Reserved | 14:5 | R | 0000000000 | Reserved. Reads return 0. Writes have no effect. |
| PLLDIV2 | 4:0 | R/W | 00011 | Divider D2 ratio (SYSCLK2 divider)<br><br>PLLDIV2 = 00000: Divide by 1<br>PLLDIV2 = 00001: Divide by 2<br>PLLDIV2 = 00010: Reserved<br>PLLDIV2 = 00011: Divide by 4<br>PLLDIV2 = 00100−11111: Reserved |

### 3.9.5.6 PLL Divider3 Register (PLLDIV3) for SYSCLK3

This register controls the value of the divider D3 for SYSCLK3. It is in both the BYPASS and PLL paths.

| 15 | 14 | 8 |
|---|---|---|
| D3EN | Reserved | |
| R/W, 1 | R, 0000000 | |

| 7 | 5 | 4 | 0 |
|---|---|---|---|
| Reserved | | PLLDIV3 | |
| R, 000 | | R/W, 00011 | |

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

**Figure 3−20. PLL Divider 3 Register Layout (0x1C90)**

**Table 3−17. PLL Divider 3 Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| D3EN | 15 | R/W | 1 | Divider D3 enable<br><br>D3EN = 0: Divider 3 disabled<br>D3EN = 1: Divider 3 enabled |
| Reserved | 14:5 | R | 0000000000 | Reserved. Reads return 0. Writes have no effect. |
| PLLDIV3 | 4:0 | R/W | 00011 | Divider D3 ratio (SYSCLK3 divider)<br><br>PLLDIV3 = 00000: Divide by 1<br>PLLDIV3 = 00001: Divide by 2<br>PLLDIV3 = 00010: Reserved<br>PLLDIV3 = 00011: Divide by 4<br>PLLDIV3 = 00100−11111: Reserved |

### 3.9.5.7 Oscillator Divider1 Register (OSCDIV1) for CLKOUT3

This register controls the value of the divider OD1 for CLKOUT3. It does not go through the PLL path.

| 15 | 14 | 8 |
|---|---|---|
| OD1EN | Reserved | |
| R/W, 0 | R, 0000000 | |

| 7 | 5 | 4 | 0 |
|---|---|---|---|
| Reserved | | OSCDIV1 | |
| R, 000 | | R/W, 00000 | |

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

**Figure 3−21. Oscillator Divider1 Register Layout (0x1C92)**

**Table 3−18. Oscillator Divider1 Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| OD1EN | 15 | R/W | 0 | Oscillator divider OD1 enable<br><br>OD1EN = 0: Oscillator divider 1 disabled<br>OD1EN = 1: Oscillator divider 1 enabled |
| Reserved | 14:5 | R | 0000000000 | Reserved. Reads return 0. Writes have no effect. |
| OSCDIV1 | 4:0 | R/W | 00000 | Divider OD1 ratio (CLKOUT3 divider)<br><br>OSCDIV1 = 00000: Divide by 1<br>OSCDIV1 = 00001: Divide by 2<br>OSCDIV1 = 00010: Divide by 3<br>OSCDIV1 = 00011: Divide by 4<br>OSCDIV1 = 00100: Divide by 5<br>OSCDIV1 = 00101: Divide by 6<br>OSCDIV1 = 00110: Divide by 7<br>OSCDIV1 = 00111: Divide by 8<br>OSCDIV1 = 01000: Divide by 9<br>OSCDIV1 = 01001: Divide by 10<br>OSCDIV1 = 01010: Divide by 11<br>OSCDIV1 = 01011: Divide by 12<br>OSCDIV1 = 01100: Divide by 13<br>OSCDIV1 = 01101: Divide by 14<br>OSCDIV1 = 01110: Divide by 15<br>OSCDIV1 = 01111: Divide by 16<br>OSCDIV1 = 10000: Divide by 17<br>OSCDIV1 = 10001: Divide by 18<br>OSCDIV1 = 10010: Divide by 19<br>OSCDIV1 = 10011: Divide by 20<br>OSCDIV1 = 10100: Divide by 21<br>OSCDIV1 = 10101: Divide by 22<br>OSCDIV1 = 10110: Divide by 23<br>OSCDIV1 = 10111: Divide by 24<br>OSCDIV1 = 11000: Divide by 25<br>OSCDIV1 = 11001: Divide by 26<br>OSCDIV1 = 11010: Divide by 27<br>OSCDIV1 = 11011: Divide by 28<br>OSCDIV1 = 11100: Divide by 29<br>OSCDIV1 = 11101: Divide by 30<br>OSCDIV1 = 11110: Divide by 31<br>OSCDIV1 = 11111: Divide by 32 |

### 3.9.5.8 Oscillator Wakeup Control Register (WKEN)

This register controls whether different events in the system are enabled to wake up the device after entering OSCPWRDN.

| 15 | 8 |
|---|---|
| Reserved | |
| R, 00000000 | |

| 7 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|
| Reserved | | WKEN4 | WKEN3 | WKEN2 | WKEN1 | WKEN0 |
| R, 000 | | R/W, 1 | R/W, 1 | R/W, 1 | R/W, 1 | R/W, 1 |

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

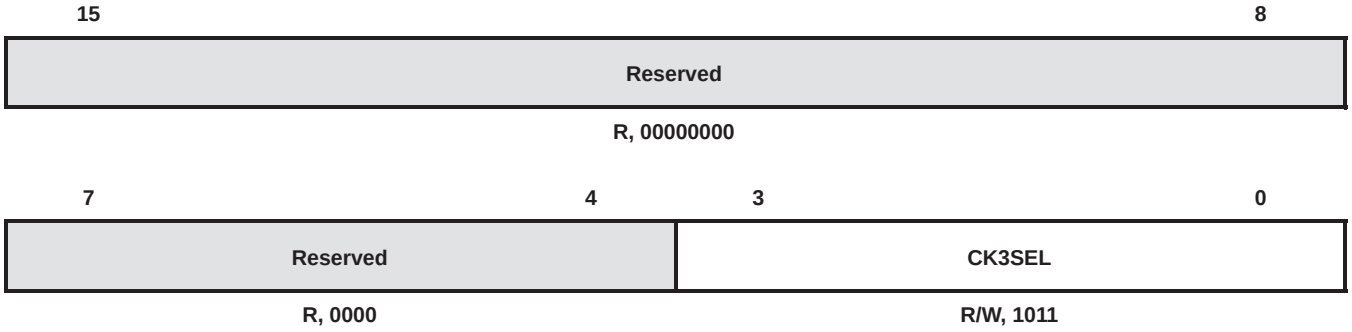
**Figure 3−22. Oscillator Wakeup Control Register Layout (0x1C98)**

**Table 3−19. Oscillator Wakeup Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15:5 | R | 00000000000 | Reserved. Reads return 0. Writes have no effect. |
| WKEN4 | 4 | R/W | 1 | Input $\overline{\text{INT3}}$ can wake up the oscillator when the OSCPWRDN bit in PLLCSR is asserted to logic 1.<br><br>WKEN4 = 0: Wake-up enabled. A low-to-high transition on $\overline{\text{INT3}}$ wakes up the oscillator and clears the OSCPWRDN bit.<br>WKEN4 = 1: Wake-up disabled. A low-to-high transition on $\overline{\text{INT3}}$ does not wake up the oscillator. |
| WKEN3 | 3 | R/W | 1 | Input $\overline{\text{INT2}}$ can wake up the oscillator when the OSCPWRDN bit in PLLCSR is asserted to logic 1.<br><br>WKEN3 = 0: Wake-up enabled. A low-to-high transition on $\overline{\text{INT2}}$ wakes up the oscillator and clears the OSCPWRDN bit.<br>WKEN3 = 1: Wake-up disabled. A low-to-high transition on $\overline{\text{INT2}}$ does not wake up the oscillator. |
| WKEN2 | 2 | R/W | 1 | Input $\overline{\text{INT1}}$ can wake up the oscillator when the OSCPWRDN bit in PLLCSR is asserted to logic 1.<br><br>WKEN2 = 0: Wake-up enabled. A low-to-high transition on $\overline{\text{INT1}}$ wakes up the oscillator and clears the OSCPWRDN bit.<br>WKEN2 = 1: Wake-up disabled. A low-to-high transition on $\overline{\text{INT1}}$ does not wake up the oscillator. |
| WKEN1 | 1 | R/W | 1 | Input $\overline{\text{INT0}}$ can wake up the oscillator when the OSCPWRDN bit in PLLCSR is asserted to logic 1.<br><br>WKEN1 = 0: Wake-up enabled. A low-to-high transition on $\overline{\text{INT0}}$ wakes up the oscillator and clears the OSCPWRDN bit.<br>WKEN1 = 1: Wake-up disabled. A low-to-high transition on $\overline{\text{INT0}}$ does not wake up the oscillator. |
| WKEN0 | 0 | R/W | 1 | Input $\overline{\text{NMI}}$ can wake up the oscillator when the OSCPWRDN bit in PLLCSR is asserted to logic 1.<br><br>WKEN0 = 0: Wake-up enabled. A low-to-high transition on $\overline{\text{NMI}}$ wakes up the oscillator and clears the OSCPWRDN bit.<br>WKEN0 = 1: Wake-up disabled. A low-to-high transition on $\overline{\text{NMI}}$ does not wake up the oscillator. |

### 3.9.5.9 CLKOUT3 Select Register (CK3SEL)

This register controls which clock is output onto the CLKOUT3 so that it may be used to test and debug the PLL (in addition to its normal function of being a direct input clock divider). Modes other than CK3SEL = 1011 are intended for debug use only and should not be used during normal operation.




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

**Figure 3−23. CLKOUT3 Select Register Layout (0x1C82)**

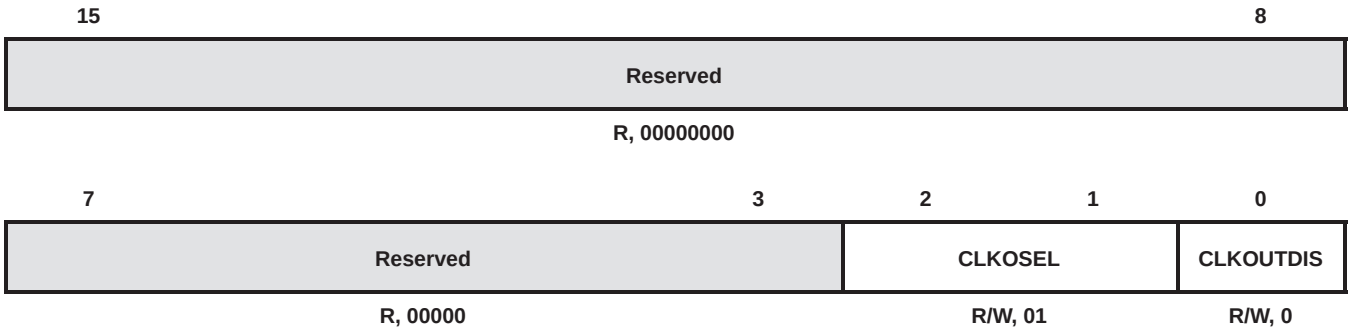
**Table 3−20. CLKOUT3 Select Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15:4 | R | 000000000000 | Reserved. Reads return 0. Writes have no effect. |
| CK3SEL | 3:0 | R/W | 1011 | Output on CLK3SEL pin†<br><br>CK3SEL = 1001 CLKOUT3 becomes point A in Figure 3−14<br>CK3SEL = 1010 CLKOUT3 becomes point B in Figure 3−14<br>CK3SEL = 0000−0111 CLKOUT3 becomes oscillator divider output in Figure 3−14<br>CK3SEL = 1011 CLKOUT3 becomes point C in Figure 3−14<br>CK3SEL = Other Not supported |

† The different options for the CLKOUT3 signal are intended for test purposes; it is recommended that the CK3SEL bits of the CK3SEL register be kept at their default value of ‘1011b’ during normal operation.

### 3.9.5.10 CLKOUT Selection Register (CLKOUTSR)

As described in Section 3.9.2, Clock Groups, the 5501 has different clock groups, each of which can be driven by a clock that is different from the CPU clock. The CLKOUT Selection Register determines which clock signal is reflected on the CLKOUT pin.



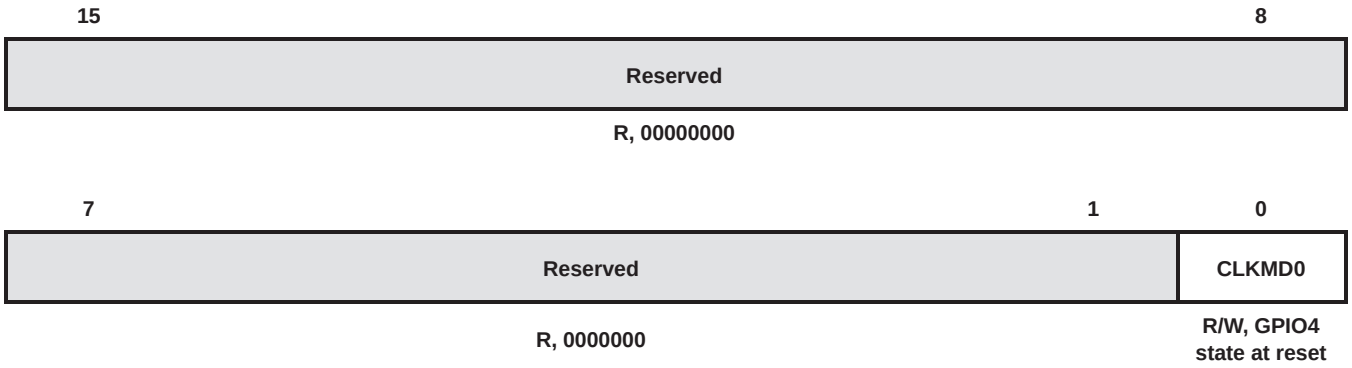

**LEGEND:** R = Read, W = Write, *n* = value at reset

**Figure 3−24. CLKOUT Selection Register Layout (0x8400)**

**Table 3−21. CLKOUT Selection Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−3 | R | 0000000000000 | Reserved |
| CLKOSEL | 2:1 | R/W | 01 | CLKOUT source-select<br><br>CLKOSEL = 00: Reserved<br>CLKOSEL = 01: CLKOUT source is SYSCLK1<br>CLKOSEL = 10: CLKOUT source is SYSCLK2<br>CLKOSEL = 11: CLKOUT source is SYSCLK3 |
| CLKOUTDIS | 0 | R/W | 0 | Disable CLKOUT<br><br>CLKOUTDIS = 0: CLKOUT enabled<br>CLKOUTDIS = 1: CLKOUT disabled (driving 0) |

### 3.9.5.11 Clock Mode Control Register (CLKMD)




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

**Figure 3−25. Clock Mode Control Register Layout (0x8C00)**

**Table 3−22. Clock Mode Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−1 | R | 000000000000000 | Reserved |
| CLKMD0 | 0 | R/W | GPIO4 state at reset | Clock output source-select<br><br>CLKMD0 = 0: OSCOUT is selected as clock input source<br>CLKMD0 = 1: X2/CLKIN is selected as clock input source |

### 3.9.6 Reset Sequence

When $\overline{\text{RESET}}$ is low, the clock generator is in bypass mode with the input clock set to CLKIN or X2/CLKIN, dependent upon the state of GPIO4. After the $\overline{\text{RESET}}$ pin transitions from low to high, the following events will occur in the order listed below.

- GPIO6 is sampled on the rising edge of the reset signal. The state of GPIO6 determines the function of the multiplexed pins of the 5501. (See Section 3.3, Configurable External Ports and Signals, for more information on pin multiplexing.) The state of GPIO6 during the rising edge of reset determines the value of the Parallel/Host Port Mux Mode bit of the External Bus Control Register (XBSR).

- GPIO4 is sampled on the rising edge of the reset signal to set the state of the CLKMD0 bit of the Clock Mode Control Register (CLKMD), which in turns, determines the clock source for the DSP. The CLKMD0 bit selects either the internal oscillator output (OSCOUT) or the X2/CLKIN pin as the input clock source for the DSP. If GPIO4 is low at reset, the CLKMD0 bit will be set to 0 and the internal oscillator and the external crystal generate the input clock for the DSP. If GPIO4 is high, the CLKMD0 bit will be set to 1 and the input clock will be taken directly from the X2/CLKIN pin.

- After the reset signal transitions from low to high, the DSP will not be taken out of reset immediately. Instead, an internal counter will count 41032 clock cycles to allow the internal oscillator to stabilize (only if GPIO4 was low). The internal counter will also add 70 reference clock cycles to allow the reset signal to propagate through different parts of the device.

- After all internal delay cycles have expired, the $\overline{\text{IACK}}$ pin will go low for two CPU clock cycles to indicate this internal reset signal has been deasserted. The BOOTM[2:0] pins will be sampled and their values will be stored in the Boot Mode Register (BOOTM_MODE). The value in the BOOTM_MODE register will be used by the bootloader to determine the boot mode of the DSP.

- Program flow will commence after all internal delay cycles have expired.

The 5501 has internal circuitry that will count down 70 reference clock cycles to allow reset signals to propagate correctly to all parts of the device after reset ($\overline{\text{RESET}}$ pin goes high). Furthermore, the 5501 also has internal circuitry that will count down 41,032 reference clock cycles to allow the oscillator input to become stable after waking up from power-down state or reset. If a reset is asserted, program flow will start after all stabilization periods have expired; this includes the oscillator stabilization period only if GPIO4 is low at reset. If the oscillator is coming out of power-down mode, program flow will start immediately after the oscillator stabilization period has completed. Table 3−23 summarizes the number of reference clock cycles needed before program flow begins.

**Table 3−23. Number of Reference Clock Cycles Needed Until Program Flow Begins**

| CONDITION | | REFERENCE CLOCK CYCLES |
|---|---|---|
| After Reset | Oscillator Not Used (GPIO4 = 1) | 70 |
| | Oscillator Used (GPIO4 = 0) | 41,102 |
| After Oscillator Power-Down | | 41,032 |

All output (O/Z) and input/output (I/O/Z) pins (except for CLKOUT, ECLKOUT2, and XF) will go into high-impedance mode during reset and will come out of high-impedance mode when the stabilization periods have expired. All output (O/Z) and input/output (I/O/Z) pins will retain their value when the device enters a power-down mode such as IDLE3 mode.

## 3.10 Idle Control

The Idle function is implemented for low power consumption. The Idle function achieves low power consumption by gating the clock to unused parts of the chip, and/or setting the clock generator (PLL) and the internal oscillator to a power-down mode.

### *3.10.1 Clock Domains*

The 5501 provides six clock domains to power-off the main clock to the portions of the device that are not being used. The six domains are:

- CPU Domain
- Master Port Domain (includes DMA and HPI modules)
- ICACHE
- Peripherals Domain
- Clock Generator Domain
- EMIF Domain

### *3.10.2 IDLE Procedures*

Before entering idle mode (executing the IDLE instruction), the user has first to determine which part of the system needs to be disabled and then program the Idle Control Register (ICR) accordingly. When the IDLE instruction is executed, the ICR will be copied into the Idle Status Register (ISTR). The different bits of the ISTR register will be propagated to disable the chosen domains. Special care has to be taken in programming the ICR as some IDLE domain combinations are not valid (for example: CPU on and clock generator off).

#### 3.10.2.1 CPU Domain Idle Procedure

The 5501 CPU can be idled by executing the following procedure.

1. Write ‘1’ to the CPUI bit (bit 0 of ICR).
2. Execute the IDLE instruction.
3. CPU will go to idle state

#### 3.10.2.2 Master Port Domain (DMA/HPI) Idle Procedure

The clock to the DMA module and/or the HPI module will be stopped when the DMA and/or the HPI bit in the MICR is set to 1 and the MPIS bit in the ISTR becomes 1. The DMA will go into idle immediately if there is no data transfer taking place. If there is a data transfer taking place, then it will finish the current transfer and then go into idle. The HPI will go into idle regardless of whether or not there is a data transfer taking place. Software must confirm that the HPI has no activity before setting it to idle.

The 5501 DMA module and the HPI module can be disabled by executing the following procedure.

1. Write ‘1’ to the DMA bit and/or the HPI bit in MICR.
2. Write ‘1’ to the MPI bit in ICR.
3. Execute the IDLE instruction.
4. DMA and/or HPI go/goes to idle.

### 3.10.2.3 Peripheral Modules Idle Procedure

The clock to the modules included in the Peripherals Domain will be stopped when their corresponding bit in the PICR is set to 1 and the PERIS bit in the ISTR becomes 1. Each module in this domain will go into idle immediately if it has no activity. If the module being set to idle has activity, it will wait until the activity completes before going into idle.

Each peripheral module can be idled by executing the following procedure.

1. Write ‘1’ to the corresponding bit in PICR for each peripheral to be idled.
2. Write ‘1’ to the PERI bit in ICR.
3. Execute the IDLE instruction.
4. Every peripheral with its corresponding PICR bit set will go to idle.

### 3.10.2.4 EMIF Module Idle Procedure

The 5501 EMIF can be idled in one of two ways: through the ICR and through the PICR. The EMIF will go into idle immediately if there is no data transfer taking place within the DMA. If there is a data transfer taking place, then the EMIF will wait until the DMA finishes the current transfer and goes into idle before going into idle itself. Please note that while the EMIF is in idle, the SDRAM refresh function of the EMIF will not be available.

The 5501 EMIF can be idled through the ICR only when the following modules are set to idle: CPU, I-Port, ICACHE, DMA, and HPI. To place the EMIF in idle using the ICR, execute the following procedure:

1. Write ‘1’ to the DMA and HPI bits in MICR.
2. Write ‘1’ to the CPUI, MPI, ICACHEI, EMIFI, and IPORTI bits in ICR.
3. Execute the IDLE instruction.
4. EMIF and all modules listed in Step 2 will go to idle.

The 5501 EMIF can also be idled through the PICR. To place the EMIF in idle using the PICR, execute the following procedure:

1. Write a ‘1’ to the EMIF bit in PICR.
2. Write a ‘1’ to the PERI bit in ICR.
3. Execute IDLE instruction.
4. EMIF will go to IDLE.

### 3.10.2.5 IDLE2 Mode

In IDLE2 mode, all modules except the CLOCK module are set to idle state. To place the 5501 in IDLE2 mode, perform the following steps.

1. Write a ‘1’ to all peripheral module bits in the PICR.
2. Write a ‘1’ to the HPI and DMA bits in MICR.
3. Write a ‘1’ to all domain bits in the ICR except the CLOCK domain bit (CLKI).
4. Execute the IDLE instruction.
5. All internal clocks will be disabled, the CLOCK module will remain active.

### 3.10.2.6 IDLE3 Mode

In IDLE3 mode, all modules (including the CLOCK module) are set to idle state. To place the 5501 in IDLE3 mode, perform the following steps:

1. Clear (i.e., set to ‘0’) the PLLEN bit in PLLCSR to place the PLL in bypass mode.
2. Set the PLLPWRDN and PLLRST bits in PLLCSR to ‘1’.
3. Write a ‘1’ to all peripheral module bits in PICR (write 0x3FFF to PICR).
4. Write a ‘1’ to the HPI and DMA bits in MICR (write 0x0003 to MICR).
5. Write a ‘1’ to all domain bits and bit 9 in the ICR (write 0x03FF to ICR).
6. Execute the IDLE instruction.
7. PLL core is set to power-down mode and all internal clocks are disabled.

### 3.10.2.7 IDLE3 Mode With Internal Oscillator Disabled

In this state, all modules (including the CLOCK module) are set to the idle state and the internal oscillator is set to the power-down mode. This is the lowest power-consuming state that 5501 can be placed under.

1. Clear (i.e., set to ‘0’) the PLLEN bit in PLLCSR to place the PLL in bypass mode.
2. Set the PLLPWRDN, PLLRST, and OSCPWRDN bits in PLLCSR to ‘1’.
3. Set the WKEN register to specify which event will wake up internal oscillator [e.g., set bit 1 to have interrupt 0 (INT0) wake up the oscillator].†
4. Write a ‘1’ to all peripheral module bits in PICR (write 0x3FFF to PICR).
5. Write a ‘1’ to the HPI and DMA bits in MICR (write 0x0003 to MICR).
6. Write a ‘1’ to all domain bits and bit 9 in the ICR (write 0x03FF to ICR).
7. Execute the IDLE instruction.
8. Internal oscillator is set to power-down mode, PLL core is set to power-down mode, and all internal clocks are disabled.

Note that the internal oscillator can be awakened through an NMI or external interrupt as long as that event is specified in the Oscillator Wakeup Control Register and, in the case of an external interrupt, the interrupt is enabled in the CPU’s Interrupt Enable Register.

† Maskable external interrupts must be enabled through IER prior to setting the 5501 to IDLE.

### 3.10.3 Module Behavior at Entering IDLE State

All transactions must be completed before entering the IDLE state. Table 3−24 lists the behavior of each module before entering the IDLE state.

**Table 3−24. Peripheral Behavior at Entering IDLE State**

| CLOCK DOMAIN | MODULES | MODULE BEHAVIOR AT ENTERING IDLE STATE (ASSUMING THE IDLE CONTROL IS SET) |
|---|---|---|
| CPU | CPU | Enter IDLE after CPU stops pipeline. |
| | Interrupt Controller | Enter IDLE after CPU stops. |
| | IDLE Controller | Enter IDLE after CPU stops. |
| | PLL Controller | Enter IDLE after CPU stops. |
| Master Port | DMA | Enter IDLE state after current DMA transfer to internal memory, EMIF, or peripheral, or enter IDLE state immediately if no transfer exists.<br><br>DMA has function of Auto-wakeup/Idle with McBSP data transfer during IDLE. |
| | HPI | Enter IDLE state immediately. Software has to take care of HPI activity. |
| ICACHE | ICACHE | Enter IDLE state after current data transfer from EMIF or program fetch from CPU finishes, or enter IDLE state immediately if no transfer and no access exist. |
| Peripheral | External Bus Selection Register | Enter IDLE after CPU stops. |
| | Timer Signal Selection Register | Enter IDLE after CPU stops. |
| | CLKOUT Selection Register | Enter IDLE after CPU stops. |
| | External Bus Control Register | Enter IDLE after CPU stops. |
| | Clock Mode Control Register | Enter IDLE after CPU stops. |
| | Timer0/1 and WDT | Enter IDLE state immediately |
| | DSP/BIOS Timer | Enter IDLE state immediately |
| | McBSP0/1 | **External Clock and Frame:**<br>Enter IDLE state after current McBSP activity is finished or enter IDLE state immediately if no activity exists. McBSP has function of Auto-wakeup/Idle with DMA transfer during IDLE.<br>**Internal Clock and Frame:**<br>Enter IDLE state immediately if both transmitter and receiver are in reset (XRST = 0 and RRST = 0). IDLE state not entered otherwise. |
| | GPIO | Enter IDLE state immediately. |
| | I2C | Enter IDLE state after current $I^2C$ activity is finished or enter IDLE state immediately if no activity exists. |
| | UART | Enter IDLE state after current UART activity is finished or enter IDLE state immediately if no activity exists. |
| | Parallel GPIO | Enter IDLE state immediately. |
| Clock Generator | PLL divider | Enter IDLE state immediately. |
| | PLL core | Power-down state if set by software before IDLE |
| | Oscillator | Power-down state if set by software before IDLE |
| EMIF | EMIF | Enter IDLE mode after current DMA transfer or enter IDLE mode immediately if no activity exists. |

### 3.10.4 Wake-Up Procedures

It is the user's responsibility to ensure that there exists a valid wake-up procedure before entering idle mode. Keep in mind that a hardware reset will restore all modules to their active state. All wake-up procedures are described in the next sections.

#### 3.10.4.1 CPU Domain Wake-up Procedure

The CPU domain can be taken out of idle though an enabled external interrupt or an $\overline{\text{NMI}}$ signal. External interrupts can be enabled through the use of the IER0 and IER1 registers. Other modules, such as the EMIF module, will be taken out of idle automatically when the CPU wakes up. Please see the wake-up procedures for other modules for more information.

#### 3.10.4.2 Master Port Domain (DMA/HPI) Wake-up Procedure

The 5501 DMA module and the HPI module can be taken out of idle simultaneously by executing the following procedure.

1. Write '0' to the MPI bit in ICR.
2. Execute the IDLE instruction.
3. DMA **and** HPI wake up.

It is also possible to wake up the DMA and HPI modules individually through the use of the Master Idle Control Register. To wake up only the DMA or the HPI module, perform the following steps:

1. Write '0' to the DMA bit **or** the HPI bit in MICR.
2. Selected module wakes up.

#### 3.10.4.3 Peripheral Modules Wake-up Procedure

All 5501 peripherals can be taken out of idle simultaneously by executing the following procedure.

1. Write '0' to the PERI bit in ICR.
2. Execute the IDLE instruction.
3. All idled peripherals wake up.

It is also possible to wake up individual peripherals through the use of the Peripheral Idle Control Register by executing the following procedure.

1. Write '0' to the idle control bit of peripheral(s) in PICR.
2. Idled peripherals with '0' in PICR wake up.

### 3.10.4.4 EMIF Module Wake-up Procedure

If both the CPU and the EMIF are in idle, then the EMIF will come out of idle when the CPU is taken out of idle. The CPU can be taken out of idle through the use of an NMI or an enabled external interrupt. External interrupts can be enabled through the IER0 and IER1 registers.

If the CPU is not in idle, then the EMIF can be taken out of idle through either of the following two procedures:

1. Write ‘0’ to the PERI bit in ICR.
2. Execute the IDLE instruction.
3. All idled peripherals, including the EMIF, wake up.

Or:

1. Write ‘0’ to the EMIF bit in PICR.
2. The EMIF module will wake up.

### 3.10.4.5 IDLE2 Mode Wake-up Procedure

The 5501 can be taken completely out of IDLE2 mode by executing the following procedure.

1. CPU wakes up from idle through NMI or enabled external interrupt.
2. Write ‘0’ to all bits in the ICR.
3. Execute the IDLE instruction.
4. All internal clocks are enabled and all modules come out of idle.

### 3.10.4.6 IDLE3 Mode Wake-up Procedure

The 5501 can be taken completely out of IDLE3 mode by executing the following procedure.

1. CPU wakes up from idle through NMI or enabled external interrupt.
2. Write ‘0’ to all bits in the ICR.
3. Execute the IDLE instruction.
4. All internal clocks are enabled and all modules come out of idle.
5. Write ‘0’ to the PLLPWRDN and PLLRST bits in PLLCSR.
6. Wait for the PLL to relock by polling the LOCK bit or by setting up a LOCK interrupt.
7. Set the PLLEN bit in PLLCSR to ‘1’.
8. All internal clocks will now come from the PLL core.

   **NOTE:** Step 3 can be modified to only wake up certain modules, see previous sections for more information on the wake-up procedures for the 5501 modules.

### 3.10.4.7 IDLE3 Mode With Internal Oscillator Disabled Wake-up Procedure

The internal oscillator of the 5501 will be woken up along with the CLOCK module through an NMI or an enabled external interrupt. The source (INT0, INT1, INT2, INT3, or NMI) for the wake-up signal can be selected through the use of the WKEN register. The maskable external interrupts must be enabled through IER0 and IER1 prior to setting the 5501 to Idle 3 mode.

The 5501 has internal circuitry that will count down a predetermined number of clock cycles (41,032 reference clock cycles) to allow the oscillator input to become stable after waking up from power-down state or reset. When waking up from idle mode, program flow will start after the stabilization period of the oscillator has expired (41032 reference clock cycles).

To take the 5501 (including the internal oscillator) out of the idle 3 state, execute the following procedure:

1. External interrupt or NMI occurs (as specified in the WKEN register) and program flow begins after 41,032 reference clock cycles.
2. CPU wakes up.
3. Write ‘0’ to all bits in the ICR.
4. Execute the IDLE instruction.
5. All internal clocks are enabled and all modules come out of idle.
6. Write ‘0’ to the PLLPWRDN, PLLRST, and OSCPWRDN bits in PLLCSR.
7. Wait for the PLL to relock by polling the LOCK bit or by setting up a LOCK interrupt.
8. Set the PLLEN bit in PLLCSR to ‘1’.
9. All internal clocks will now come from the PLL core.

**NOTE:** Step 2 can be modified to only wake up certain modules, see previous sections for more information on the wake-up procedures for the 5501 modules.

### 3.10.4.8 Summary of Wake-up Procedures

Table 3−25 summarizes the wake-up procedures.

**Table 3−25. Wake-Up Procedures**

| ISTR VALUE | CLOCK DOMAIN STATUS | EXIT FROM IDLE | ICR AFTER WAKE-UP | ISTR AFTER WAKE-UP |
|---|---|---|---|---|
| xxx0xxx0 | CPU − ON<br>Clock Generator − ON<br>Other − ON/OFF | 1. DSP software modifies ICR and executes “IDLE” instruction<br>2. Reset | 1. Modified value<br>2. All “0” | 1. Updated to ICR modified value after “IDLE” instruction<br>2. All “0” |
| xxx0xxx1 | CPU − OFF<br>Clock Generator − ON<br>Other − ON/OFF | 1. Unmasked interrupt from external or on-chip module<br>2. Reset | 1. Not modified<br>2. All “0” | 1. CPUIS, CLKIS, and EMIFIS/XPORTIS/IPORTIS are set to “0”<br>2. All “0” |
| xxx11111 | CPU − OFF<br>Clock Generator − OFF<br>Other − OFF | 1. Unmasked interrupt from external<br>2. Reset | 1. Not modified<br>2. All “0” | 1. CPUIS, CLKIS, and EMIFIS/XPORTIS/IPORTIS are set to “0”<br>2. All “0” |

### 3.10.5 Auto-Wakeup/Idle Function for McBSP and DMA

The 5501 has an Auto-wakeup/Idle function for McBSP to DMA to on-chip memory data transfers when the DMA and the McBSP are both set to IDLE. In the case that the McBSP is set to external clock mode and the McBSP and the DMA are set to idle, the McBSP and the DMA can wake up from IDLE state automatically if the McBSP gets a new data transfer. The McBSP and the DMA enter the idle state automatically after data transfer is complete. [The clock generator (PLL) should be active and the PLL core should not be in power-down mode for the Auto-wakeup/Idle function to work.]

### 3.10.6 Clock State of Multiplexed Modules

The clock to the EMIF module is disabled automatically when this module is not selected through the External Bus Selection Register (XBSR). Note that any accesses to disabled modules will result in a bus error.

### 3.10.7 IDLE Control and Status Registers

The clock domains are controlled by the IDLE Configuration Register (ICR) that allows the user to place different parts of the device in Idle mode. The IDLE Status Register (ISTR) reflects the portion of the device that remains active. The peripheral domain is controlled by the Peripheral IDLE Control Register (PICR). The Peripheral IDLE Status Register (PISTR) reflects the portion of the peripherals that are in the IDLE state. The Master IDLE Control Register (MICR) is used to place the HPI and DMA in Idle mode. The IDLE state of the HPI and DMA is reflected by the Master IDLE Status Register (MISR). The PLL Control/Status Register (PLLCSR) is used to power down the PLL core when the IDLE instruction is executed.

The settings in the ICR, PICR, and MICR take effect after the IDLE instruction is executed. For example, writing xxx000001b into the ICR does not indicate that the CPU domain is in IDLE mode; rather, it indicates that after the IDLE instruction, the CPU domain will be in IDLE mode. Procedures for placing portions of the device in Idle mode and taking them out of Idle mode are described in Section 3.10.2 (IDLE Procedures) and Section 3.10.4 (Wake-Up Procedures), respectively.

**Table 3−26. Clock Domain Memory-Mapped Registers**

| ADDRESS | REGISTER NAME |
|---|---|
| 0x0001 | IDLE Configuration Register (ICR) |
| 0x0002 | IDLE Status Register (ISTR) |
| 0x9400 | Peripheral IDLE Control Register (PICR) |
| 0x9401 | Peripheral IDLE Status Register (PISTR) |
| 0x9402 | Master IDLE Control Register (MICR) |
| 0x9403 | Master IDLE Status Register (MISR) |

### 3.10.7.1 IDLE Configuration Register (ICR)

| 15 | | | | | | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | CLKEI† | IPORTI |
| R, 000000 | | | | | | | R/W, 0 | R/W, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| MPORTI | XPORTI | EMIFI | CLKI | PERI | ICACHEI | MPI | CPUI |
| R/W, 0 | R/W, 0 | 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

† This bit must be set to '1' when placing the clock generator in idle; otherwise, a bus error interrupt will be generated.

**Figure 3−26. IDLE Configuration Register Layout (0x0001)**

**Table 3−27. IDLE Configuration Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−10 | R | 000000 | Reserved |
| CLKEI | 9 | R/W | 0 | Extended device clock generator idle control bit. The CLKEI bit must be set to 1 along with the CLKI bit in order to properly place the device clock generator in idle.<br><br>**CLKI CLKEI**<br>0 0 Device clock generator module remains active after execution of an IDLE instruction.<br>1 1 Device clock generator is disabled after execution of an IDLE instruction.<br><br>Any other combination of CLKI and CLKEI is not valid. Setting CLKI to 1 and executing the IDLE instruction will generate a bus error interrupt if CLKEI is not set to 1.<br><br>Disabling the clock generator provides the lowest level of power reduction by stopping the system clock. Whenever the clock generator is idled, the CLKEI, CPUI, MPI, ICACHEI, EMIFI, XPORTI, MPORTI, and IPORTI bits must be set to 1 in order to ensure a proper power-down mode. A bus error interrupt will be generated if the idle instruction is executed when CLKI = 1 and any of these bits are not set to 1. |
| IPORTI | 8 | R/W | 0 | IPORT idle control bit**.** The IPORT is used for all ICACHE transactions.<br><br>IPORTI = 0: IPORT remains active after execution of an IDLE instruction<br>IPORTI = 1: IPORT is disabled after execution of an IDLE instruction |
| MPORTI | 7 | R/W | 0 | MPORT idle control bit. The MPORT is used for all DMA and HPI transactions.<br><br>MPORTI = 0: MPORT remains active after execution of an IDLE instruction<br>MPORTI = 1: MPORT is disabled after execution of an IDLE instruction |
| XPORTI | 6 | R/W | 0 | XPORT idle control bit. The XPORT is used for all I/O memory transactions.<br><br>XPORTI = 0: XPORT remains active after execution of an IDLE instruction<br>XPORTI = 1: XPORT is disabled after execution of an IDLE instruction |
| EMIFI | 5 | R/W | 0 | External Memory Interface (EMIF) idle control bit<br><br>EMIFI = 0: EMIF module remains active after execution of an IDLE instruction<br>EMIFI = 1: EMIF module is disabled after execution of an IDLE instruction |

**Table 3−27. IDLE Configuration Register Bit Field Description (Continued)**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| CLKI | 4 | R/W | 0 | Device clock generator idle control bit. The CLKEI bit must be set to 1 along with the CLKI bit in order to properly place the device clock generator in idle.<br><br>**CLKI CLKEI**<br>0 0 Device clock generator module remains active after execution of an IDLE instruction.<br>1 1 Device clock generator is disabled after execution of an IDLE instruction.<br><br>Any other combination of CLKI and CLKEI is not valid. Setting CLKI to 1 and executing the IDLE instruction will generate a bus error interrupt if CLKEI is not set to 1.<br><br>Disabling the clock generator provides the lowest level of power reduction by stopping the system clock. Whenever the clock generator is idled, the CLKEI, CPUI, MPI, ICACHEI, EMIFI, XPORTI, MPORTI, and IPORTI bits must be set to 1 in order to ensure a proper power-down mode. A bus error interrupt will be generated if the idle instruction is executed when CLKI = 1 and any of these bits are not set to 1. |
| PERI | 3 | R/W | 0 | Peripheral Idle control bit<br><br>PERI = 0: All peripheral modules become/remain active after execution of an IDLE instruction<br>PERI = 1: All peripheral modules with 1 in PICR are disabled after execution of an IDLE instruction |
| ICACHEI | 2 | R/W | 0 | ICACHE idle control bit<br><br>ICACHEI = 0: ICACHE module remains active after execution of an IDLE instruction<br>ICACHEI = 1: ICACHE module is disabled after execution of an IDLE instruction |
| MPI | 1 | R/W | 0 | Master peripheral (DMA and HPI) idle control bit<br><br>MPI = 0: DMA and HPI modules remain active after execution of an IDLE instruction<br>MPI = 1: DMA and HPI modules are disabled after execution of an IDLE instruction |
| CPUI | 0 | R/W | 0 | CPU idle control bit<br><br>CPUI = 0: CPU module remains active after execution of an IDLE instruction<br>CPUI = 1: CPU module is disabled after execution of an IDLE instruction |

### 3.10.7.2 IDLE Status Register (ISTR)

| 15 | 9 | 8 |
|---|---|---|
| Reserved | | IPORTIS |
| R, 0000000 | | R, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| MPORTIS | XPORTIS | EMIFIS | CLKIS | PERIS | ICACHEIS | MPIS | CPUIS |
| R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 |

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

**Figure 3−27. IDLE Status Register Layout (0x0002)**

**Table 3−28. IDLE Status Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−9 | R | 0000000 | Reserved |
| IPORTIS | 8 | R | 0 | IPORT idle status bit**.** The IPORT is used for all ICACHE transactions.<br><br>IPORTIS = 0: IPORT is active<br>IPORTIS = 1: IPORT is disabled |
| MPORTIS | 7 | R | 0 | MPORT idle status bit**.** The MPORT is used for all DMA and HPI transactions.<br><br>MPORTIS = 0: MPORT is active<br>MPORTIS = 1: MPORT is disabled |
| XPORTIS | 6 | R | 0 | XPORT idle status bit**.** The XPORT is used for all I/O memory transactions.<br><br>XPORTIS = 0: XPORT is active<br>XPORTIS = 1: XPORT is disabled |
| EMIFIS | 5 | R | 0 | External Memory Interface (EMIF) idle status bit<br><br>EMIFIS = 0: EMIF module is active<br>EMIFIS = 1: EMIF module is disabled |
| CLKIS | 4 | R | 0 | Device clock generator idle status bit<br><br>CLKIS = 0: Device clock generator module is active<br>CLKIS = 1: Device clock generator is disabled |
| PERIS | 3 | R | 0 | Peripheral idle status bit<br><br>PERIS = 0: All peripheral modules are active<br>PERIS = 1: All peripheral modules are disabled |
| ICACHEIS | 2 | R | 0 | ICACHE idle status bit<br><br>ICACHEIS = 0: ICACHE module is active<br>ICACHEIS = 1: ICACHE module is disabled |
| MPIS | 1 | R | 0 | DMA and HPI idle status bit<br><br>MPIS = 0: DMA and HPI modules are active<br>MPIS = 1: DMA and HPI modules are disabled |
| CPUIS | 0 | R | 0 | CPU idle status bit<br><br>CPUIS = 0: CPU module is active<br>CPUIS = 1: CPU module is disabled |

### 3.10.7.3 Peripheral IDLE Control Register (PICR)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | MISC | EMIF | BIOST | WDT | PIO | URT |
| R, 00 | | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| I2C | ID | IO | Reserved | SP1 | SP0 | TIM1 | TIM0 |
| R/W, 0 | R/W, 0 | 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 3−28. Peripheral IDLE Control Register Layout (0x9400)**

**Table 3−29. Peripheral IDLE Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−14 | R | 00 | Reserved |
| MISC | 13† | R/W | 0 | MISC bit<br><br>MISC = 0: Miscellaneous modules remain active when ISTR.PERIS = 1 and IDLE instruction is executed.<br>MISC = 1: MIscellaneous module is disabled when ISTR.PERIS = 1 and IDLE instruction is executed.<br><br>Miscellaneous modules include the XBSR, TIMEOUT Error Register, XBCR, Timer Signal Selection Register, CLKOUT Select Register, and Clock Mode Control Register. |
| EMIF | 12† | R/W | 0 | EMIF bit<br><br>EMIF = 0: EMIF module remains active when ISTR.PERIS = 1 and IDLE instruction is executed.<br>EMIF = 1: EMIF module is disabled when ISTR.PERIS = 1 and IDLE instruction is executed. |
| BIOST | 11† | R/W | 0 | BIOS timer bit<br><br>BIOST = 0: DSP/BIOS timer remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>BIOST = 1: DSP/BIOS timer is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| WDT | 10† | R/W | 0 | Watchdog timer bit<br><br>WDT = 0: WDT remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>WDT = 1: WDT is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| PIO | 9† | R/W | 0 | Parallel GPIO bit<br><br>PIO = 0: Parallel GPIO remains active when ISTR.PERIS = 1 (ISTR.[3]) and the IDLE instruction is executed.<br>PIO = 1: Parallel GPIO is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |

† If the peripheral is already in IDLE, setting PERI (bit 3 of ICR) to 0 and executing the IDLE instruction will wake up all peripherals, and PICR bit settings will be ignored. If PERIS (bit 3 of ISTR) = 1, executing the IDLE instruction will wake up the peripheral if its PICR bit is 0.

**Table 3−29. Peripheral IDLE Control Register Bit Field Description (Continued)**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| URT | 8† | R/W | 0 | UART bit<br>URT = 0: UART remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>URT = 1: UART is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| I2C | 7† | R/W | 0 | I2C bit<br>I2C = 0: $I^2C$ remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>I2C = 1: $I^2C$ is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| ID | 6† | R/W | 0 | ID bit<br>ID = 0: ID remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>ID = 1: ID is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| IO | 5† | R/W | 0 | IO bit<br>IO = 0: GPIO remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>IO = 1: GPIO is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| Reserved | 4 | R/W | 0 | Reserved |
| SP1 | 3† | R/W | 0 | McBSP1 bit<br>SP1 = 0: McBSP1 remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>SP1 = 1: McBSP1 is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| SP0 | 2† | R/W | 0 | McBSP0 bit<br>SP0 = 0: McBSP0 remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>SP0 = 1: McBSP0 is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| TIM1 | 1† | R/W | 0 | TIMER1 bit<br>TIM1 = 0: TIMER1 remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>TIM1 = 1: TIMER1 is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |
| TIM0 | 0† | R/W | 0 | TIMER0 bit<br>TIM0 = 0: TIMER0 remains active when ISTR.PERIS = 1 and the IDLE instruction is executed.<br>TIM0 = 1: TIMER0 is disabled when ISTR.PERIS = 1 and the IDLE instruction is executed. |

† If the peripheral is already in IDLE, setting PERI (bit 3 of ICR) to 0 and executing the IDLE instruction will wake up all peripherals, and PICR bit settings will be ignored. If PERIS (bit 3 of ISTR) = 1, executing the IDLE instruction will wake up the peripheral if its PICR bit is 0.

### 3.10.7.4 Peripheral IDLE Status Register (PISTR)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | MISC | EMIF | BIOST | WDT | PIO | URT |
| R, 00 | | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| I2C | ID | IO | Reserved | SP1 | SP0 | TIM1 | TIM0 |
| R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 | R, 0 |

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

**Figure 3−29. Peripheral IDLE Status Register Layout (0x9401)**

**Table 3−30. Peripheral IDLE Status Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−14 | R | 00 | Reserved |
| MISC | 13 | R | 0 | MISC bit<br><br>MISC = 0: Miscellaneous modules are active<br>MISC = 1: Miscellaneous modules are disabled<br><br>Miscellaneous modules include the XBSR, TIMEOUT Error Register, XBCR, Timer Signal Selection Register, CLKOUT Select Register, and Clock Mode Control Register. |
| EMIF | 12 | R | 0 | EMIF bit<br><br>EMIF = 0: EMIF module is active<br>EMIF = 1: EMIF module is disabled |
| BIOST | 11 | R | 0 | BIOS timer bit<br><br>BIOST = 0: DSP/BIOS timer is active<br>BIOST = 1: DSP/BIOS timer is disabled |
| WDT | 10 | R | 0 | Watchdog timer bit<br><br>WDT = 0: WDT is active<br>WDT = 1: WDT is disabled |
| PIO | 9 | R | 0 | Parallel GPIO bit<br><br>PIO = 0: Parallel GPIO is active<br>PIO = 1: Parallel GPIO is disabled |
| URT | 8 | R | 0 | UART bit<br><br>URT = 0: UART is active<br>URT = 1: UART is disabled |
| I2C | 7 | R | 0 | I2C bit<br><br>I2C = 0: $I^2C$ is active<br>I2C = 1: $I^2C$ is disabled |
| ID | 6 | R | 0 | ID bit<br><br>ID = 0: ID is active<br>ID = 1: ID is disabled |
| IO | 5 | R | 0 | IO bit<br><br>IO = 0: GPIO is active<br>IO = 1: GPIO is disabled |

**Table 3−30. Peripheral IDLE Status Register Bit Field Description (Continued)**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 4 | R | 0 | Reserved |
| SP1 | 3 | R | 0 | McBSP1 bit<br><br>SP1 = 0: McBSP1 is active<br>SP1 = 1: McBSP1 is disabled |
| SP0 | 2 | R | 0 | McBSP0 bit<br><br>SP0 = 0: McBSP0 is active<br>SP0 = 1: McBSP0 is disabled |
| TIM1 | 1 | R | 0 | TIMER1 bit<br><br>TIM1 = 0: TIMER1 is active<br>TIM1 = 1: TIMER1 is disabled |
| TIM0 | 0 | R | 0 | TIMER0 bit<br><br>TIM0 = 0: TIMER0 is active<br>TIM0 = 1: TIMER0 is disabled |

### 3.10.7.5 Master IDLE Control Register (MICR)

| 15 | 8 |
|---|---|
| Reserved | |
| R, 00000000 | |

| 7 | 2 | 1 | 0 |
|---|---|---|---|
| Reserved | | HPI | DMA |
| R, 000000 | | R/W, 0 | R/W, 0 |

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

**Figure 3−30. Master IDLE Control Register Layout (0x9402)**

**Table 3−31. Master IDLE Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−2 | R | 00000000000000 | Reserved |
| HPI | 1 | R/W | 0 | HPI bit<br><br>HPI = 0: HPI remains active when ISTR.MPIS becomes 1<br>HPI = 1: HPI is disabled when ISTR.MPIS becomes 1 |
| DMA | 0 | R/W | 0 | DMA bit<br><br>DMA = 0: DMA remains active when ISTR.MPIS becomes 1<br>DMA = 1: DMA is disabled when ISTR.MPIS becomes 1 |

### 3.10.7.6 Master IDLE Status Register (MISR)

| 15 | | | | | | | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R, 00000000 | | | | | | | |

| 7 | | | | | | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | HPI | DMA |
| R, 000000 | | | | | | | R, 0 | R, 0 |

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

**Figure 3−31. Master IDLE Status Register Layout (0x9403)**

**Table 3−32. Master IDLE Status Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−2 | R | 00000000000000 | Reserved |
| HPI | 1 | R | 0 | HPI bit<br><br>HPI = 0: HPI is active<br>HPI = 1: HPI is in IDLE status |
| DMA | 0 | R | 0 | DMA bit<br><br>DMA = 0: DMA is active<br>DMA = 1: DMA is in IDLE status |

## 3.11 General-Purpose I/O (GPIO)

The 5501 includes an 8-bit I/O port solely for general-purpose input and output. Several dual-purpose (multiplexed) pins complement the dedicated GPIO pins. The following sections describe the 8-bit GPIO port as well as the dual GPIO functions of the Parallel Port Mux and Host Port Mux pins.

### *3.11.1 General-Purpose I/O Port*

The general-purpose I/O port consists of eight individually bit-selectable I/O pins GPIO0 (LSB) through GPIO7 (MSB). The I/O port is controlled using two registers—IODIR and IODATA—that can be accessed by the CPU or by the DMA, via the peripheral bus controller. The General-Purpose I/O Direction Register (IODIR) is mapped at address 0x3400, and the General-Purpose I/O Data Register (IODATA) is mapped at address 0x3401.

Figure 3−32 and Figure 3−33 show the bit layout of IODIR and IODATA, respectively. Table 3−33 and Table 3−34 describe the bit fields of these registers.

### 3.11.1.1 General-Purpose I/O Direction Register (IODIR)

| 15 | | | | | | | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R, 00000000 | | | | | | | |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO7DIR | IO6DIR | IO5DIR | IO4DIR | IO3DIR | IO2DIR | IO1DIR | IO0DIR |
| R/W, 0 | R/W, 0 | 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 3−32. GPIO Direction Register Layout (0x3400)**

**Table 3−33. GPIO Direction Register Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−8 | R | 00000000 | Reserved |
| IOxDIR | 7−0 | R/W | 00000000 | Data direction bits that configure the GPIO pins as inputs or outputs.<br><br>IOxDIR = 0: Configure corresponding GPIO pin as an input<br>IOxDIR = 1: Configure corresponding GPIO pin as an output |

† x = value from 0 to 7

### 3.11.1.2 General-Purpose I/O Data Register (IODATA)

| 15 | | | | | | | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R, 00000000 | | | | | | | |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO7D | IO6D | IO5D | IO4D | IO3D | IO2D | IO1D | IO0D |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |

**LEGEND:** R = Read, W = Write, *n* = value at reset, *pin* = the reset value depends on the signal level on the corresponding I/O pin.

**Figure 3−33. GPIO Data Register Layout (0x3401)**

**Table 3−34. GPIO Data Register Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−8 | R | 00000000 | Reserved |
| IOxD | 7−0 | R/W | Depends on the signal level on the corresponding I/O pin | Data bits that are used to control the level of the I/O pins configured as outputs and to monitor the level of the I/O pins configured as inputs.<br><br>If IOxDIR = 0, then:<br>IOxD = 0: Corresponding GPIO pin is read as a low<br>IOxD = 1: Corresponding GPIO pin is read as a high<br><br>If IOxDIR = 1, then:<br>IOxD = 0: Set corresponding GPIO pin to low<br>IOxD = 1: Set corresponding GPIO pin to high |

† x = value from 0 to 7

### 3.11.2 Parallel Port General-Purpose I/O (PGPIO)

Four address pins (A[21:18]), 16 data pins (D[31:16]), 16 control signals (C[15:0]), 8 host data pins (HD[7:0]), and 2 HPI control pins (HC0, HC1) can be individually enabled as PGPIO when the Parallel/Host Port Mux Mode bit field of the External Bus Selection Register (XBSR) is cleared for PGPIO mode (see Table 3−35). These pins are controlled by three sets of registers: the PGPIO enable registers, the PGPIO direction registers, and the PGPIO data registers.

- The PGPIO enable registers PGPIOEN0−PGPIOEN2 (see Figure 3−34, Figure 3−37, and Figure 3−40) determine if the output function of the PGPIO pins is enabled or disabled.
- The PGPIO direction registers PGPIODIR0−PGPIODIR2 (see Figure 3−35, Figure 3−38, and Figure 3−41) determine if corresponding bits in the PGPIO data registers specify an output value or an input value.
- The PGPIO data registers PGPIODAT0−PGPIODAT2 (see Figure 3−36, Figure 3−39, and Figure 3−42) store the value read or written externally.

To use a PGPIO pin as an output, its corresponding bit must be set to 1 in both the enable and direction registers. The state of the pin is then controlled through its bit in the data register. Conversely, to use a PGPIO pin as an input, its corresponding bit must be cleared to 0 in both the enable and the direction registers. The state of the pin can then be read from its bit in the data register.

**NOTE:** The enable registers PGPIOENn cannot override the External Bus Selection Register (XBSR) setting.

**Table 3−35. TMS320VC5501 PGPIO Cross-Reference**

| PIN | PARALLEL/HOST PORT MUX MODE = 0 (PGPIO) | PARALLEL/HOST PORT MUX MODE = 1 (FULL EMIF) |
|---|---|---|
| **EMIF Address Bus** | | |
| A[21:18] | PGPIO[3:0] | EMIF.A[21:18] |
| **EMIF Data Bus** | | |
| D[31:16] | PGPIO[19:4] | EMIF.D[31:16] |
| **EMIF Control Bus** | | |
| C0 | PGPIO20 | EMIF.$\overline{\text{ARE}}$/$\overline{\text{SADS}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SRE}}$ |
| C1 | PGPIO21 | EMIF.$\overline{\text{AOE}}$/$\overline{\text{SOE}}$/$\overline{\text{SDRAS}}$ |
| C2 | PGPIO22 | EMIF.$\overline{\text{AWE}}$/$\overline{\text{SWE}}$/$\overline{\text{SDWE}}$ |
| C3 | PGPIO23 | EMIF.ARDY |
| C4 | PGPIO24 | EMIF.$\overline{\text{CE0}}$ |
| C5 | PGPIO25 | EMIF.$\overline{\text{CE1}}$ |
| C6 | PGPIO26 | EMIF.$\overline{\text{CE2}}$ |
| C7 | PGPIO27 | EMIF.$\overline{\text{CE3}}$ |
| C8 | PGPIO28 | EMIF.$\overline{\text{BE0}}$ |
| C9 | PGPIO29 | EMIF.$\overline{\text{BE1}}$ |
| C10 | PGPIO30 | EMIF.$\overline{\text{BE2}}$ |
| C11 | PGPIO31 | EMIF.$\overline{\text{BE3}}$ |
| C12 | PGPIO32 | EMIF.SDCKE |
| C13 | PGPIO33 | EMIF.$\overline{\text{SOE3}}$ |
| C14 | PGPIO34 | EMIF.$\overline{\text{HOLD}}$ |
| C15 | PGPIO35 | EMIF.$\overline{\text{HOLDA}}$ |

**Table 3−35. TMS320VC5501 PGPIO Cross-Reference (Continued)**

| PIN | PARALLEL/HOST PORT MUX MODE = 0 (PGPIO) | PARALLEL/HOST PORT MUX MODE = 1 (FULL EMIF) |
|---|---|---|
| **HPI Data Bus** | | |
| HD[7:0] | PGPIO[43:36] | HPI.HD[7:0] |
| **HPI Control Bus** | | |
| HC0 | PGPIO44 | HPI.$\overline{\text{HAS}}$ |
| HC1 | PGPIO45 | HPI.HBIL |

### 3.11.2.1 Parallel GPIO Enable Register 0 (PGPIOEN0)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO15EN | IO14EN | IO13EN | IO12EN | IO11EN | IO10EN | IO9EN | IO8EN |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO7EN | IO6EN | IO5EN | IO4EN | IO3EN | IO2EN | IO1EN | IO0EN |
| R/W, 0 | R/W, 0 | 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 3−34. Parallel GPIO Enable Register 0 Layout (0x4400)**

**Table 3−36. Parallel GPIO Enable Register 0 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxEN | 15−0 | R/W | 0000000000000000 | Enable or disable output function of the corresponding I/O pins. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxEN = 0: Output function of the PGPIOx pin is disabled—i.e., the pin cannot drive an output signal; it can only be used as an input. When IOxEN = 0, IOxDIR must also be cleared to 0.<br>IOxEN = 1: Output function of the PGPIOx pin is enabled—i.e., the pin is used to drive an output signal. When IOxEN = 0, IOxDIR must also be set to 1; otherwise, the output value is undefined. |

† x = value from 0 to 15

### 3.11.2.2 Parallel GPIO Direction Register 0 (PGPIODIR0)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO15DIR | IO14DIR | IO13DIR | IO12DIR | IO11DIR | IO10DIR | IO9DIR | IO8DIR |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IO7DIR | IO6DIR | IO5DIR | IO4DIR | IO3DIR | IO2DIR | IO1DIR | IO0DIR |
| R/W, 0 | R/W, 0 | 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 3−35. Parallel GPIO Direction Register 0 Layout (0x4401)**

**Table 3−37. Parallel GPIO Direction Register 0 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxDIR | 15−0 | R/W | 0000000000000000 | Data direction bits specify if corresponding bits in the data registers specify an output value or an input value. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxDIR = 0: Corresponding bit in the data register specifies the value read on the PGPIOx pin (input). When IOxDIR = 0, IOxEN must also be cleared to 0.<br>IOxDIR = 1: Corresponding bit in the data register specifies the value driven on the PGPIOx pin (output). When IOxDIR = 1, IOxEN must also be set to 1. |

† x = value from 0 to 15

### 3.11.2.3 Parallel GPIO Data Register 0 (PGPIODAT0)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO15DAT | IO14DAT | IO13DAT | IO12DAT | IO11DAT | IO10DAT | IO9DAT | IO8DAT |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IO7DAT | IO6DAT | IO5DAT | IO4DAT | IO3DAT | IO2DAT | IO1DAT | IO0DAT |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |

**LEGEND:** R = Read, W = Write, *n* = value at reset, *pin* = the reset value depends on the signal level on the corresponding I/O pin.

**Figure 3−36. Parallel GPIO Data Register 0 Layout (0x4402)**

**Table 3−38. Parallel GPIO Data Register 0 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxDAT | 15−0 | R/W | Depends on the signal level on the corresponding I/O pin | Data bits that are used to either control the level of the corresponding I/O pins configured as output pins or to monitor the level of the corresponding I/O pins configured as input pins. The function of the data register bits is determined by the setting of the direction register bits. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br><br>If IOxEN = 0 and IOxDIR = 0, then IOxDAT is used to read the value of the PGPIOx pin:<br><br>IOxDAT = 0: PGPIOx pin is read as a low<br>IOxDAT = 1: PGPIOx pin is read as a high<br><br>If IOxEN = 1 and IOxDIR = 1, then IOxDAT is used to set the value of the PGPIOx pin:<br><br>IOxDAT = 0: Set PGPIOx pin to low<br>IOxDAT = 1: Set PGPIOx pin to high<br><br>Note that other combinations of IOxEN and IOxDIR are not supported—i.e., IOxEN and IOxDIR must always be set to the same value. |

† x = value from 0 to 15

### 3.11.2.4 Parallel GPIO Enable Register 1 (PGPIOEN1)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO31EN | IO30EN | IO29EN | IO28EN | IO27EN | IO26EN | IO25EN | IO24EN |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IO23EN | IO22EN | IO21EN | IO20EN | IO19EN | IO18EN | IO17EN | IO16EN |
| R/W, 0 | R/W, 0 | 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 3−37. Parallel GPIO Enable Register 1 Layout (0x4403)**

**Table 3−39. Parallel GPIO Enable Register 1 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxEN | 15−0 | R/W | 0000000000000000 | Enable or disable output function of the corresponding I/O pins. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxEN = 0: Output function of the PGPIOx pin is disabled—i.e., the pin cannot drive an output signal; it can only be used as an input. When IOxEN = 0, IOxDIR must also be cleared to 0.<br>IOxEN = 1: Output function of the PGPIOx pin is enabled—i.e., the pin is used to drive an output signal. When IOxEN = 0, IOxDIR must also be set to 1; otherwise, the output value is undefined. |

† x = value from 16 to 31

### 3.11.2.5 Parallel GPIO Direction Register 1 (PGPIODIR1)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO31DIR | IO30DIR | IO29DIR | IO28DIR | IO27DIR | IO26DIR | IO25DIR | IO24DIR |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IO23DIR | IO22DIR | IO21DIR | IO20DIR | IO19DIR | IO18DIR | IO17DIR | IO16DIR |
| R/W, 0 | R/W, 0 | 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 3−38. Parallel GPIO Direction Register 1 Layout (0x4404)**

**Table 3−40. Parallel GPIO Direction Register 1 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxDIR | 15−0 | R/W | 0000000000000000 | Data direction bits specify if corresponding bits in the data registers specify an output value or an input value. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxDIR = 0: Corresponding bit in the data register specifies the value read on the PGPIOx pin (input). When IOxDIR = 0, IOxEN must also be cleared to 0.<br>IOxDIR = 1: Corresponding bit in the data register specifies the value driven on the PGPIOx pin (output). When IOxDIR = 1, IOxEN must also be set to 1. |

† x = value from 16 to 31

### 3.11.2.6 Parallel GPIO Data Register 1 (PGPIODAT1)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| IO31DAT | IO30DAT | IO29DAT | IO28DAT | IO27DAT | IO26DAT | IO25DAT | IO24DAT |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IO23DAT | IO22DAT | IO21DAT | IO20DAT | IO19DAT | IO18DAT | IO17DAT | IO16DAT |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |

**LEGEND:** R = Read, W = Write, *n* = value at reset, *pin* = the reset value depends on the signal level on the corresponding I/O pin.

**Figure 3−39. Parallel GPIO Data Register 1 Layout (0x4405)**

**Table 3−41. Parallel GPIO Data Register 1 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| IOxDAT | 15−0 | R/W | Depends on the signal level on the corresponding I/O pin | Data bits that are used to either control the level of the corresponding I/O pins configured as output pins or to monitor the level of the corresponding I/O pins configured as input pins. The function of the data register bits is determined by the setting of the direction register bits. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br><br>If IOxEN = 0 and IOxDIR = 0, then IOxDAT is used to read the value of the PGPIOx pin:<br><br>IOxDAT = 0: PGPIOx pin is read as a low<br>IOxDAT = 1: PGPIOx pin is read as a high<br><br>If IOxEN = 1 and IOxDIR = 1, then IOxDAT is used to set the value of the PGPIOx pin:<br><br>IOxDAT = 0: Set PGPIOx pin to low<br>IOxDAT = 1: Set PGPIOx pin to high<br><br>Note that other combinations of IOxEN and IOxDIR are not supported—i.e., IOxEN and IOxDIR must always be set to the same value. |

† x = value from 16 to 31

### 3.11.2.7 Parallel GPIO Enable Register 2 (PGPIOEN2)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | IO45EN | IO44EN | IO43EN | IO42EN | IO41EN | IO40EN |
| R/W, 00 | | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO39EN | IO38EN | IO37EN | IO36EN | IO35EN | IO34EN | IO33EN | IO32EN |
| R/W, 0 | R/W, 0 | 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 3−40. Parallel GPIO Enable Register 2 Layout (0x4406)**

**Table 3−42. Parallel GPIO Enable Register 2 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−14 | R/W | 00 | Reserved |
| IOxEN | 13−0 | R/W | 00000000000000 | Enable or disable output function of the corresponding I/O pins. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxEN = 0: Output function of the PGPIOx pin is disabled—i.e., the pin cannot drive an output signal; it can only be used as an input. When IOxEN = 0, IOxDIR must also be cleared to 0.<br>IOxEN = 1: Output function of the PGPIOx pin is enabled—i.e., the pin is used to drive an output signal. When IOxEN = 0, IOxDIR must also be set to 1; otherwise, the output value is undefined. |

† x = value from 32 to 45

### 3.11.2.8 Parallel GPIO Direction Register 2 (PGPIODIR2)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | IO45DIR | IO44DIR | IO43DIR | IO42DIR | IO41DIR | IO40DIR |
| R/W, 00 | | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO39DIR | IO38DIR | IO37DIR | IO36DIR | IO35DIR | IO34DIR | IO33DIR | IO32DIR |
| R/W, 0 | R/W, 0 | 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 3−41. Parallel GPIO Direction Register 2 Layout (0x4407)**

**Table 3−43. Parallel GPIO Direction Register 2 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−14 | R/W | 00 | Reserved |
| IOxDIR | 13−0 | R/W | 00000000000000 | Data direction bits specify if corresponding bits in the data registers specify an output value or an input value. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br>IOxDIR = 0: Corresponding bit in the data register specifies the value read on the PGPIOx pin (input). When IOxDIR = 0, IOxEN must also be cleared to 0.<br>IOxDIR = 1: Corresponding bit in the data register specifies the value driven on the PGPIOx pin (output). When IOxDIR = 1, IOxEN must also be set to 1. |

† x = value from 32 to 45

### 3.11.2.9 Parallel GPIO Data Register 2 (PGPIODAT2)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | IO45DAT | IO44DAT | IO43DAT | IO42DAT | IO41DAT | IO40DAT |
| R/W, 00 | | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |
| **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| IO39DAT | IO38DAT | IO37DAT | IO36DAT | IO35DAT | IO34DAT | IO33DAT | IO32DAT |
| R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin | R/W, pin |

**LEGEND:** R = Read, W = Write, *n* = value at reset, *pin* = the reset value depends on the signal level on the corresponding I/O pin.

**Figure 3−42. Parallel GPIO Data Register 2 Layout (0x4408)**

**Table 3−44. Parallel GPIO Data Register 2 Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−14 | R/W | 00 | Reserved |
| IOxDAT | 13−0 | R/W | Depends on the signal level on the corresponding I/O pin | Data bits that are used to either control the level of the corresponding I/O pins configured as output pins or to monitor the level of the corresponding I/O pins configured as input pins. The function of the data register bits is determined by the setting of the direction register bits. See Table 3−35, TMS320VC5501 PGPIO Cross-Reference to determine which device pins correspond to the PGPIO pins.<br><br>If IOxEN = 0 and IOxDIR = 0, then IOxDAT is used to read the value of the PGPIOx pin:<br><br>IOxDAT = 0: PGPIOx pin is read as a low<br>IOxDAT = 1: PGPIOx pin is read as a high<br><br>If IOxEN = 1 and IOxDIR = 1, then IOxDAT is used to set the value of the PGPIOx pin:<br><br>IOxDAT = 0: Set PGPIOx pin to low<br>IOxDAT = 1: Set PGPIOx pin to high<br><br>Note that other combinations of IOxEN and IOxDIR are not supported—i.e., IOxEN and IOxDIR must always be set to the same value. |

† x = value from 32 to 45

## 3.12 External Bus Control Register

The External Bus Control Register is used to disable/enable the bus pullups, pulldowns, and bus holders of the 5501 pins. Table 3−45 lists which 5501 pins have pullups, pulldowns, and bus holders, and which bit on the XBCR enables/disables that feature. Please note that for pins with dual functionality (e.g., HC0, HC1, C0, etc.), the bus holder, pullup, and pulldown feature of each pin can be enabled or disabled regardless of the function of the pin at the time.

**Table 3−45. Pins With Pullups, Pulldowns, and Bus Holders**

| XBCR CONTROL BIT | PIN | FEATURE |
|---|---|---|
| TEST | TCK | Pullup |
| | TDI | Pullup |
| | TMS | Pullup |
| | $\overline{\text{TRST}}$ | Pulldown |
| EMU | EMU1/$\overline{\text{OFF}}$ | Pullup |
| | EMU0 | Pullup |
| WDT | $\overline{\text{NMI}}$/WDTOUT | Pullup |
| HC | HC0 | Pullup |
| | HC1 | Pulldown |
| | HCNTL0 | Pullup |
| | HCNTL1 | Pullup |
| | $\overline{\text{HCS}}$ | Pullup |
| | HR/$\overline{\text{W}}$ | Pullup |
| | $\overline{\text{HDS1}}$ | Pullup |
| | $\overline{\text{HDS2}}$ | Pullup |
| | HRDY | Pullup |
| | $\overline{\text{HINT}}$ | Pullup |
| HD | HD[7:0] | Bus Holder |
| PC | C0 | Bus Holder |
| | C1 | Bus Holder |
| | C2 | Bus Holder |
| | C3 | Pullup |
| | C4 | Bus Holder |
| | C5 | Bus Holder |
| | C6 | Bus Holder |
| | C7 | Bus Holder |
| | C8 | Bus Holder |
| | C9 | Bus Holder |
| | C10 | Bus Holder |
| | C11 | Bus Holder |
| | C12 | Bus Holder |
| | C13 | Bus Holder |
| | C14 | Pullup |
| | C15 | Bus Holder |
| PD | D[31:0] | Bus Holder |
| PA | A[21:2] | Bus Holder |

### 3.12.1 External Bus Control Register (XBCR)

| 15 | | | | | | | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| R, 00000000 | | | | | | | |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EMU | TEST | WDT | HC | HD | PC | PD | PA |
| R/W, 0 | R/W, 0 | 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 3−43. External Bus Control Register Layout (0x8800)**

**Table 3−46. External Bus Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−8 | R | 00000000 | Reserved |
| EMU | 7 | R/W | 0 | EMU bit<br><br>EMU = 0: Pullups on EMU1 and EMU0 pins are enabled.<br>EMU = 1: Pullups on EMU1 and EMU0 pins are disabled. |
| TEST | 6 | R/W | 0 | TEST bit<br><br>TEST = 0: Pullups/pulldowns on test pins are enabled (does not include EMU1 and EMU0 pins)<br>TEST = 1: Pullups/pulldowns on test pins are disabled (does not include EMU1 and EMU0 pins) |
| WDT | 5 | R/W | 0 | WDT bit<br><br>WDT = 0: Pullup on $\overline{\text{NMI}}$/WDTOUT pin is enabled<br>WDT = 1: Pullup on NMI/WDTOUT pin is disabled |
| HC | 4 | R/W | 0 | HPI control signal bit<br><br>HC = 0: Pullups/pulldowns on HPI control pins (HC0 and HC1) are enabled<br>HC = 1: Pullups/pulldowns on HPI control pins (HC0 and HC1) are disabled |
| HD | 3 | R/W | 0 | HPI data bus bit<br><br>HD = 0: Bus holders on HPI data bus (pins HD[7:0]) are enabled<br>HD = 1: Bus holders on HPI data bus (pins HD[7:0]) are disabled |
| PC | 2 | R/W | 0 | EMIF control signals<br><br>PC = 0: Bus holders and pullups on EMIF control pins are enabled<br>PC = 1: Bus holders and pullups on EMIF control pins are disabled |
| PD | 1 | R/W | 0 | EMIF data bus signals<br><br>PD = 0: Bus holders on EMIF data bus (pins D[31:0]) are enabled<br>PD = 1: Bus holders on EMIF data bus (pins D[31:0]) are disabled |
| PA | 0 | R/W | 0 | EMIF address bus signals<br><br>PA = 0: Bus holders on EMIF address bus (pins A[21:2]) are enabled<br>PA = 1: Bus holders on EMIF address bus (pins A[21:2]) are disabled |

## 3.13 Internal Ports and System Registers

The 5501 includes three internal ports that interface the CPU core with the peripheral modules. Although these ports cannot be directly controlled by user code, the registers associated with each port can be used to monitor a number of error conditions that could be generated through illegal operation of the 5501. The port registers are described in the following sections.

The 5501 also includes two registers that can be used to monitor and control several aspects of the interface between the CPU and the system-level peripherals, these registers are also described in the following sections.

### *3.13.1 XPORT Interface*

The XPORT interfaces the CPU core to all peripheral modules. The XPORT will generate bus errors for invalid accesses to any registers that fall under the ranges shown in Table 3−47. The INTERREN bit of the XPORT Configuration Register (XCR) controls the bus error feature of the XPORT. The INTERR bit of the XPORT Bus Error Register (XERR) is set to “1” when an error occurs during an access to a register listed in Table 3−47. The EBUS and DBUS bits can be used to distinguish whether the error occurred during a write or read access.

**Table 3−47. I/O Addresses Under Scope of XPORT**

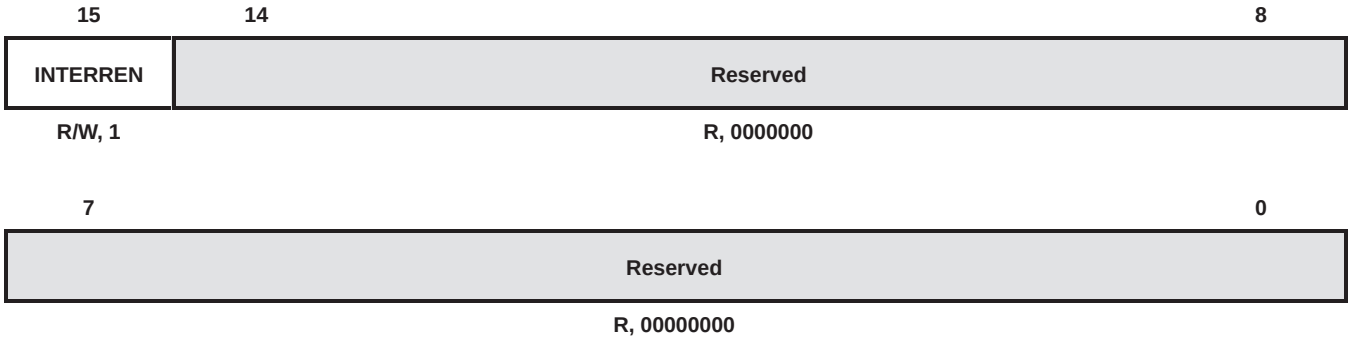
| I/O ADDRESS RANGE |
|---|
| 0x0000−0x03FF |
| 0x1400−0x17FF |
| 0x2000−0x23FF |

The PERITO bit of the XERR is used to indicate that a CPU, DMA, or HPI access to a disabled/idled peripheral module has generated a time-out error. The time-out error feature is enabled through the PERITOEN bit of the Time-Out Control Register (TOCR). A time-out error is generated when 512 clock cycles pass without a response from the peripheral register.

The XPORT can be placed into idle by setting the XPORTI bit of the Idle Control Register (ICR) and executing the IDLE instruction. When the XPORT is in idle, it will stop accepting new peripheral module requests and it will also not check for internal I/O bus errors. If there is a request from the CPU core or a peripheral module, the XPORT will not respond and hang. The ICR register will generate a bus error if the XPORT is idled without the CPU or Master Port domains being in idle mode.

### 3.13.1.1 XPORT Configuration Register (XCR)

The XPORT Configuration Register bit layout is shown in Figure 3−44 and the bits are described in Table 3−48.




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

**Figure 3−44. XPORT Configuration Register Layout (0x0100)**

**Table 3−48. XPORT Configuration Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| INTERREN | 15 | R/W | 1 | INTERREN bit<br><br>INTERREN = 0: The XPORT will not generate a bus error for invalid accesses to registers listed in Table 3−47. Note that any invalid accesses to these registers will hang the pipeline.<br>INTERREN = 1: The XPORT will generate a bus error for invalid accesses to registers listed in Table 3−47.† Note that when a bus error occurs, any data returned by the read instruction will not be valid. |
| Reserved | 14−0 | R | 000000000000000 | Reserved |

† This feature will not work if the XPORT is placed in idle through the ICR. However, a bus error will be generated if the XPORT is placed in idle without the CPU being in idle.

### 3.13.1.2 XPORT Bus Error Register (XERR)

The XPORT Bus Error Register bit layout is shown in Figure 3−45 and the bits are described in Table 3−49.

| 15 | 14 | 13 | 12 | 11 | 8 |
|---|---|---|---|---|---|
| INTERR | Reserved | | PERITO | Reserved | |
| R, 0 | R, 00 | | R, 0 | R, 0000 | |

| 7 | 5 | 4 | 3 | 2 | 0 |
|---|---|---|---|---|---|
| Reserved | | EBUS | DBUS | Reserved | |
| R, 000 | | R, 0 | R, 0 | R, 000 | |

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

**Figure 3−45. XPORT Bus Error Register Layout (0x0102)**

**Table 3−49. XPORT Bus Error Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| INTERR | 15 | R | 0 | INTERR bit<br><br>INTERR = 0: No error<br>INTERR = 1: An error occurred during an access to one of the registers listed in Table 3−47. |
| Reserved | 14−13 | R | 00 | Reserved |
| PERITO | 12 | R | 0 | PERITO bit<br><br>PERITO = 0: No error<br>PERITO = 1: A time-out error occurred during an access to a peripheral register. |
| Reserved | 11−5 | R | 0000000 | Reserved |
| EBUS | 4 | R | 0 | EBUS error bit†<br><br>EBUS = 0: No error<br>EBUS = 1: An error occurred during an EBUS access (write) to one of the registers listed in Table 3−47. |
| DBUS | 3 | R | 0 | DBUS error bit†<br><br>DBUS = 0: No error<br>DBUS = 1: An error occurred during a DBUS access (read) to one of the registers listed in Table 3−47. |
| Reserved | 2−0 | R | 000 | Reserved |

† See the *TMS320C55x DSP CPU Reference Guide* (literature number SPRU371) for more information on the D-bus and E-bus.

### 3.13.2 DPORT Interface

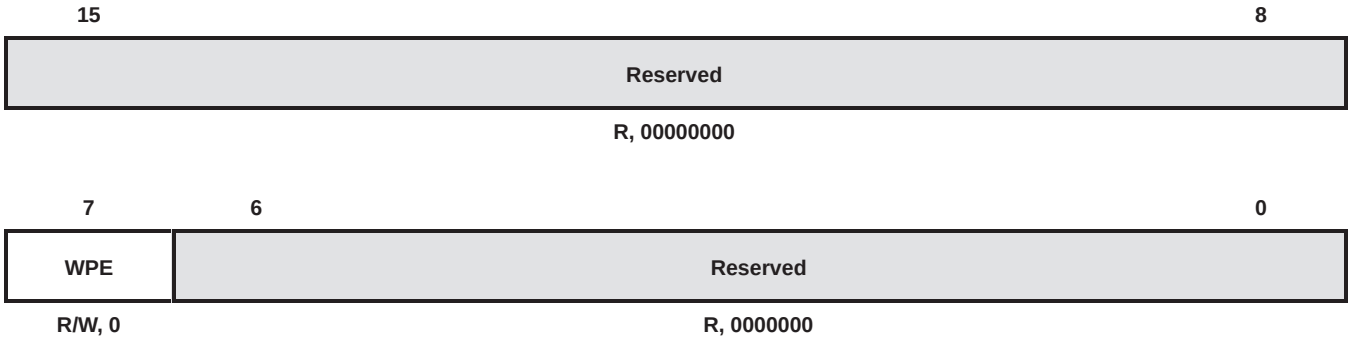
The DPORT interfaces the CPU to the EMIF module. The DPORT is capable of enabling write posting on the EMIF module. Write posting prevents stalls to the CPU during external memory writes. Two write posting registers, which are freely associated with E and F bus writes, exist within the DPORT and are used to store the write address and data so that writes can be zero wait state for the CPU. External memory writes will not generate stalls to the CPU unless the two write posting registers are filled. Write posting is enabled by setting the WPE bit of the DCR to 1.

The EMIFTO bit of the DERR is used to indicate that a CPU, DMA, HPI, or IPORT access to external memory has generated a time-out error. The time-out error feature is enabled through the EMIFTOEN bit of the Time-Out Control Register (TOCR). This function is not recommended during normal operation of the 5501.

The DPORT can be placed into idle through the EMIFI bit of the Idle Control Register (ICR) and executing the IDLE instruction. When the DPORT is in idle, it will stop accepting new EMIF requests. If there is a request from the CPU or the EMIF, the DPORT will not respond and hang. The ICR register will generate a bus error if the DPORT is idled without the CPU or Master Port domains being in idle.

#### 3.13.2.1 DPORT Configuration Register (DCR)

The DPORT Configuration Register bit layout is shown in Figure 3−46 and the bits are described in Table 3−50.




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

**Figure 3−46. DPORT Configuration Register Layout (0x0200)**

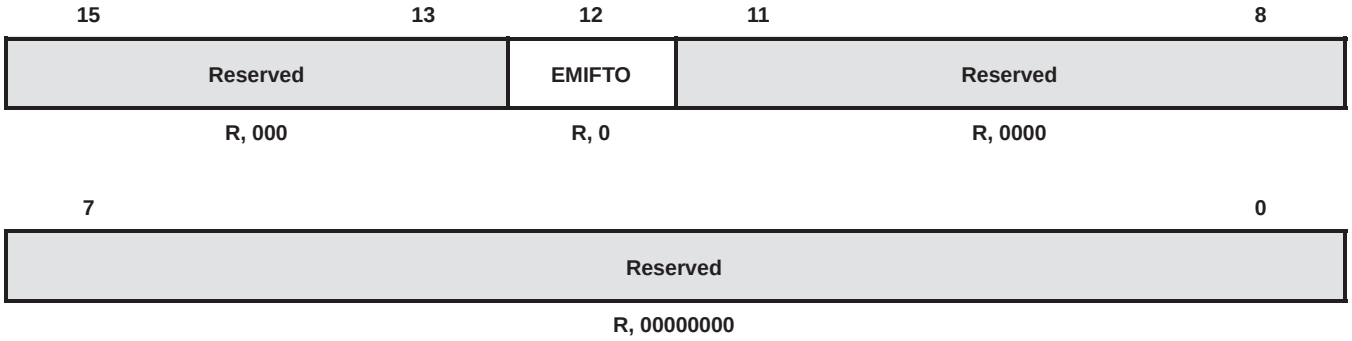
**Table 3−50. DPORT Configuration Register Bit Field Description†**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−8 | R | 00000000 | Reserved |
| WPE | 7 | R/W | 0 | Write Posting Enable bit†<br><br>WPE = 0: Write posting disabled<br>WPE = 1: Write posting enabled |
| Reserved | 6−0 | R | 0000000 | Reserved |

† Write posting should not be enabled or disabled while the EMIF is conducting a transaction with external memory.

### 3.13.2.2 DPORT Bus Error Register (DERR)

The DPORT Bus Error Register bit layout is shown in Figure 3−47 and the bits are described in Table 3−51.




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

**Figure 3−47. DPORT Bus Error Register Layout (0x0202)**

**Table 3−51. DPORT Bus Error Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−13 | R | 000 | Reserved |
| EMIFTO | 12 | R | 0 | EMIFTO bit<br><br>EMIFTO = 0: No error<br>EMIFTO = 1: Error 1 error |
| Reserved | 11−0 | R | 000000000000 | Reserved |

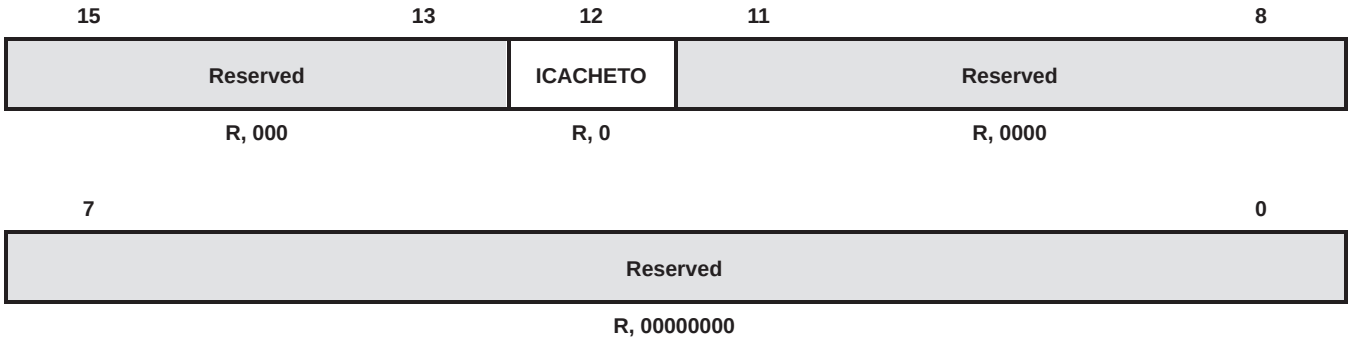
### 3.13.3 IPORT Interface

The IPORT interfaces the I-Cache to the EMIF module. The ICACHETO bit of the IPORT Bus Error Register (IERR) can be used to determine if a time-out error has occurred during an ICACHE access to external memory. The time-out feature is enabled through the EMIFTOEN bit of the Time-Out Control Register (TOCR).

The IPORT can be placed into idle through the IPORTI bit of the Idle Control Register (ICR) and executing the IDLE instruction. The IPORT will go into idle when there are no new requests from the ICACHE. When the IPORT is in idle, it will stop accepting new requests from the CPU, it is important that the program flow not use external memory in this case. If there are requests from the CPU, the IPORT will not respond and hang. The ICR register will generate a bus error if the IPORT is idled without the CPU domain being in idle.

#### 3.13.3.1 IPORT Bus Error Register (IERR)

The IPORT Bus Error Register bit layout is shown in Figure 3−48 and the bits are described in Table 3−52.




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

**Figure 3−48. IPORT Bus Error Register Layout (0x0302)**

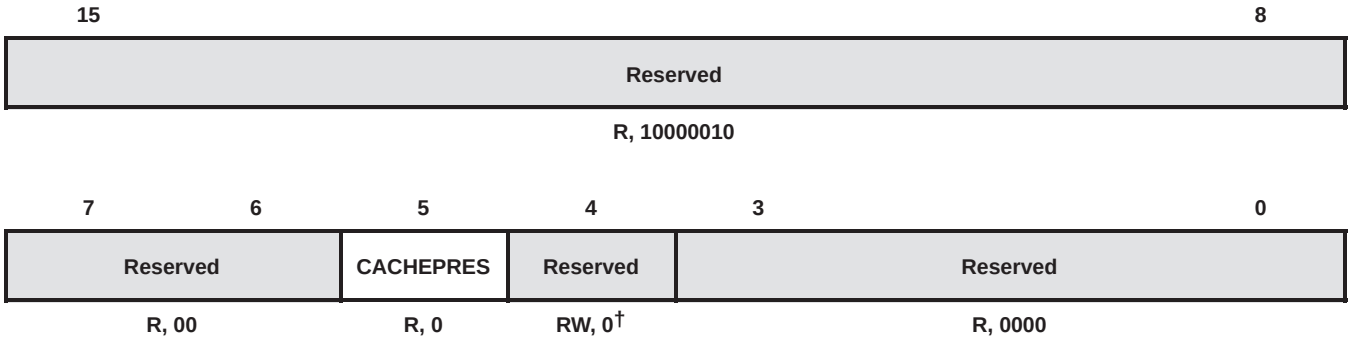
**Table 3−52. IPORT Bus Error Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−13 | R | 000 | Reserved |
| ICACHETO | 12 | R | 0 | ICACHETO bit<br><br>ICACHETO = 0: No error<br>ICACHETO = 1: A time-out error occurred during an ICACHE access to external memory. |
| Reserved | 11−0 | R | 000000000000 | Reserved |

### 3.13.4 System Configuration Register (CONFIG)

The System Configuration Register can be used to determine the operational state of the ICACHE. If the ICACHE is not functioning, the CACHEPRES bit of the CONFIG register will be cleared. If the ICACHE is functioning normally, this bit will be set.

The System Configuration Register bit layout is shown in Figure 3−49 and the bits are described in Table 3−53.




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

† This Reserved bit *must* be kept as zero during any writes to CONFIG.

**Figure 3−49. System Configuration Register Layout (0x07FD)**

**Table 3−53. System Configuration Register Bit Field Description**

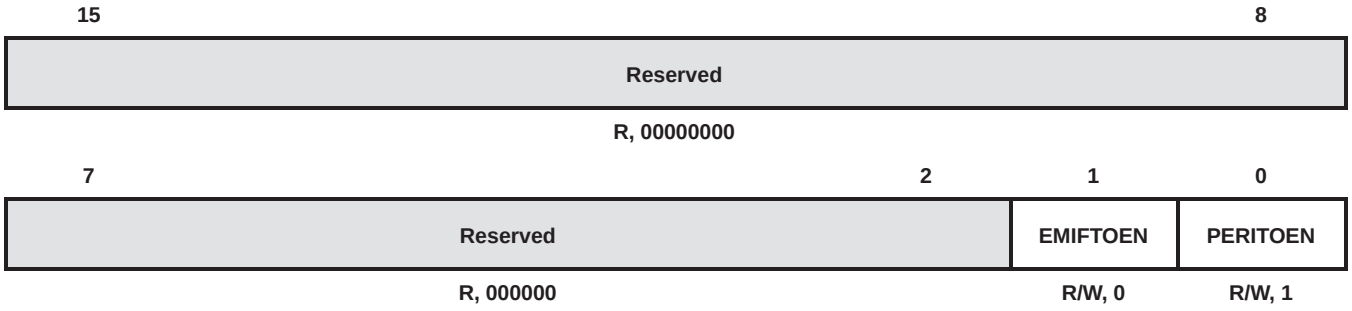
| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−6 | R | 1000001000 | Reserved |
| CACHEPRES | 5 | R | 0 | ICACHE present<br><br>CACHEPRES = 0: ICACHE is not functioning<br>CACHEPRES = 1: ICACHE is enabled and working |
| Reserved | 4 | R/W | 0† | Reserved |
| Reserved | 3−0 | R | 0000 | Reserved |

† This Reserved bit *must* be kept as zero during any writes to CONFIG.

### 3.13.5 Time-Out Control Register (TOCR)

The Time-Out Control Register can be used to select whether or not a time-out error is generated when an access to a disabled/idled peripheral module occurs. If the CPU or DMA access a disabled/idle peripheral module and 512 CPU clock cycles pass without an acknowledgement from the peripheral module, then a time-out error will be sent to the corresponding module if bit 1 in the Time-Out Control Register is set. A time-out error will generate a CPU bus error that can be serviced through software by using the bus error interrupt (BERR) (see Section 3.16, Interrupts, for more information on interrupts). If the DMA gets a time-out error, it will set the TIMEOUT bit in the DMA Status Register (DMACSR) and generate a time-out error that can be serviced through software by the CPU [see the *TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide* (literature number SPRU613) for more information on using this feature of the DMA].

The Time-Out Control Register can also be used to select whether or not a time-out error is generated when a memory access through the EMIF module stalls for more than 512 CPU clock cycles. It is recommended that this feature not be used for it can cause unexpected results.




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

**Figure 3−50. Time-Out Control Register Layout (0x9000)**

**Table 3−54. Time-Out Control Register Bit Field Description**

| BIT NAME | BIT NO. | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| Reserved | 15−2 | R | 00000000000000 | Reserved |
| EMIFTOEN | 1 | R/W | 0 | EMIF time-out control bit<br>EMIFTOEN = 0: A time-out error is not generated when an EMIF access stalls for more than 512 CPU clock cycles.<br>EMIFTOEN = 1: A time-out error is generated when an EMIF access stalls for more than 512 CPU clock cycles. |
| PERITOEN | 0 | R/W | 1 | Peripheral module time-out control bit<br>PERITOEN = 0: A time-out error is not generated when a CPU access to a disabled/idle peripheral module stalls for more than 512 CPU clock cycles.<br>PERITOEN = 1: A time-out error is generated when a CPU access to a disabled/idle peripheral module stalls for more than 512 CPU clock cycles. |

## 3.14 CPU Memory-Mapped Registers

The 5501 has 78 memory-mapped CPU registers that are mapped in data memory space address 0h to 4Fh. Table 3−55 provides a list of the CPU memory-mapped registers (MMRs) available. The corresponding TMS320C54x™ (C54x™) CPU registers are also indicated where applicable.

**Table 3−55. CPU Memory-Mapped Registers**

| C54X REGISTER | C55X REGISTER | WORD ADDRESS (HEX) | C55X REGISTER DESCRIPTION | BIT FIELD |
|---|---|---|---|---|
| IER | IER0 | 00 | Interrupt Enable Register 0 | [15−0] |
| IFR | IFR0 | 01 | Interrupt Flag Register 0 | [15−0] |
| − | ST0_55 | 02 | Status Register 0 | [15−0] |
| − | ST1_55 | 03 | Status Register 1 | [15−0] |
| − | ST3_55 | 04 | Status Register 3 | [15−0] |
| − | − | 05 | Reserved | [15−0] |
| ST0 | ST0 | 06 | Status Register 0 (protected address for C54x code) | [15−0] |
| ST1 | ST1 | 07 | Status Register 1 (protected address for C54x code) | [15−0] |
| AL | AC0L | 08 | Accumulator 0 | [15−0] |
| AH | AC0H | 09 | | [31−16] |
| AG | AC0G | 0A | | [39−32] |
| BL | AC1L | 0B | Accumulator 1 | [15−0] |
| BH | AC1H | 0C | | [31−16] |
| BG | AC1G | 0D | | [39−32] |
| TREG | T3 | 0E | Temporary Register 3 | [15−0] |
| TRN | TRN0 | 0F | Transition Register 0 | [15−0] |
| AR0 | AR0 | 10 | Auxiliary Register 0 | [15−0] |
| AR1 | AR1 | 11 | Auxiliary Register 1 | [15−0] |
| AR2 | AR2 | 12 | Auxiliary Register 2 | [15−0] |
| AR3 | AR3 | 13 | Auxiliary Register 3 | [15−0] |
| AR4 | AR4 | 14 | Auxiliary Register 4 | [15−0] |
| AR5 | AR5 | 15 | Auxiliary Register 5 | [15−0] |
| AR6 | AR6 | 16 | Auxiliary Register 6 | [15−0] |
| AR7 | AR7 | 17 | Auxiliary Register 7 | [15−0] |
| SP | SP | 18 | Data Stack Pointer | [15−0] |
| BK | BK03 | 19 | Circular Buffer Size Register for AR[0−3] | [15−0] |
| BRC | BRC0 | 1A | Block Repeat Counter 0 | [15−0] |
| RSA | RSA0L | 1B | Low Part of Block Repeat Start Address Register 0 | [15−0] |
| REA | REA0L | 1C | Low Part of Block Repeat End Address Register 0 | [15−0] |
| PMST | PMST | 1D | Status Register 3 (protected address for C54x code) | [15−0] |
| XPC | XPC | 1E | Program Counter Extension Register for C54x code | [7−0] |
| − | − | 1F | Reserved | [15−0] |
| − | T0 | 20 | Temporary Register 0 | [15−0] |
| − | T1 | 21 | Temporary Register 1 | [15−0] |
| − | T2 | 22 | Temporary Register 2 | [15−0] |
| − | T3 | 23 | Temporary Register 3 | [15−0] |
| − | AC2L | 24 | Accumulator 2 | [15−0] |
| − | AC2H | 25 | | [31−16] |
| − | AC2G | 26 | | [39−32] |

TMS320C54x and C54x are trademarks of Texas Instruments.

**Table 3−55. CPU Memory-Mapped Registers (Continued)**

| C54X REGISTER | C55X REGISTER | WORD ADDRESS (HEX) | C55X REGISTER DESCRIPTION | BIT FIELD |
|---|---|---|---|---|
| − | CDP | 27 | Coefficient Data Pointer | [15−0] |
| − | AC3L | 28 | Accumulator 3 | [15−0] |
| − | AC3H | 29 | | [31−16] |
| − | AC3G | 2A | | [39−32] |
| − | DPH | 2B | High Part of the Extended Data Page Register (XDP = DPH:DP) | [6−0] |
| − | − | 2C | Reserved | [6−0] |
| − | − | 2D | Reserved | [6−0] |
| − | DP | 2E | Data Page Register | [15−0] |
| − | PDP | 2F | Peripheral Data Page Register | [8−0] |
| − | BK47 | 30 | Circular Buffer Size Register for AR[4−7] | [15−0] |
| − | BKC | 31 | Circular Buffer Size Register for CDP | [15−0] |
| − | BSA01 | 32 | Circular Buffer Start Address Register for AR[0−1] | [15−0] |
| − | BSA23 | 33 | Circular Buffer Start Address Register for AR[2−3] | [15−0] |
| − | BSA45 | 34 | Circular Buffer Start Address Register for AR[4−5] | [15−0] |
| − | BSA67 | 35 | Circular Buffer Start Address Register for AR[6−7] | [15−0] |
| − | BSAC | 36 | Circular Buffer Start Address Register for CDP | [15−0] |
| − | BIOS | 37 | Data Page Pointer Storage Location for 128-word Data Table | [15−0] |
| − | TRN1 | 38 | Transition Register 1 | [15−0] |
| − | BRC1 | 39 | Block Repeat Counter 1 | [15−0] |
| − | BRS1 | 3A | BRC1 Save Register | [15−0] |
| − | CSR | 3B | Computed Single Repeat Register | [15−0] |
| − | RSA0H | 3C | Block Repeat Start Address Register 0 | [23−16] |
| − | RSA0L | 3D | | [15−0] |
| − | REA0H | 3E | Block Repeat End Address Register 0 | [23−16] |
| − | REA0L | 3F | | [15−0] |
| − | RSA1H | 40 | Block Repeat Start Address Register 1 | [23−16] |
| − | RSA1L | 41 | | [15−0] |
| − | REA1H | 42 | Block Repeat End Address Register 1 | [23−16] |
| − | REA1L | 43 | | [15−0] |
| − | RPTC | 44 | Single Repeat Counter | [15−0] |
| − | IER1 | 45 | Interrupt Enable Register 1 | [15−0] |
| − | IFR1 | 46 | Interrupt Flag Register 1 | [15−0] |
| − | DBIER0 | 47 | Debug Interrupt Enable Register 0 | [15−0] |
| − | DBIER1 | 48 | Debug Interrupt Enable Register 0 | [15−0] |
| − | IVPD | 49 | Interrupt Vector Pointer | [15−0] |
| − | IVPH | 4A | Interrupt Vector Pointer | [15−0] |
| − | ST2_55 | 4B | Status Register 2 | [15−0] |
| − | SSP | 4C | System Stack Pointer | [15−0] |
| − | SP | 4D | Data Stack Pointer | [15−0] |
| − | SPH | 4E | High Part of the Extended Stack Pointers (XSP = SPH:SP, XSSP = SPH:SSP) | [6−0] |
| − | CDPH | 4F | High Part of the Extended Coefficient Data Pointer (XCDP = CDPH:CDP) | [6−0] |

## 3.15 Peripheral Registers

Each 5501 device has a set of peripheral memory-mapped registers as listed in Table 3−56 through Table 3−74. Peripheral registers are accessed using the port qualifier. For more information on the use of the port qualifier, see the *TMS320C55x Assembly Language Tools User's Guide* (literature number SPRU280). Some registers use less than 16 bits. When reading these registers, unused bits are always read as 0.

The user guides for each peripheral contain detailed information on the operation and the functions of each of the peripheral registers (see Section 4.2, Documentation Support, for a list of documents supporting each peripheral).

**Table 3−56. Peripheral Bus Controller Configuration Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x0000 | Reserved | | |
| 0x0001 | ICR | Idle Configuration Register | 0000 0000 0000 0000 |
| 0x0002 | ISTR | Idle Status Register | 0000 0000 0000 0000 |
| 0x0003 to 0x000D | Reserved | | |
| 0x000E | Reserved | | |
| 0x000F | BOOT_MOD | Boot Mode Register (read only) | Value of GPIO[2:0] at reset |
| 0x0010 | Reserved | | |
| 0x0011 | Reserved | | |
| 0x0100 | XCR | XPORT Configuration Register | 1000 0000 0000 0000 |
| 0x0102 | XERR | XPORT Bus Error Register | 0000 0000 0000 0000 |
| 0x0200 | DCR | DPORT Configuration Register | 0000 0000 0000 0000 |
| 0x0202 | DERR | DPORT Bus Error Register | 0000 0000 0000 0000 |
| 0x0302 | IERR | IPORT Bus Error Register | 0000 0000 0000 0000 |
| 0x07FD | CONFIG | System Configuration Register | 1000 0010 0000 0000 |
| 0x9000 | TOCR | Time-Out Control Register | 0000 0000 0000 0001 |

† x denotes a "don't care."

**Table 3−57. External Memory Interface Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x0800 | EGCR1 | EMIF Global Control Register 1 | 0010 0111 0111 1100 |
| 0x0801 | EGCR2 | EMIF Global Control Register 2 | 0000 0000 0000 1001 |
| 0x0802 | CE1_1 | EMIF CE1 Space Control Register 1 | 1111 1111 0001 1111 |
| 0x0803 | CE1_2 | EMIF CE1 Space Control Register 2 | 1111 1111 1111 1111 |
| 0x0804 | CE0_1 | EMIF CE0 Space Control Register 1 | 1111 1111 0000 0011 |
| 0x0805 | CE0_2 | EMIF CE0 Space Control Register 2 | 1111 1111 1111 1111 |
| 0x0806 | − | Reserved | |
| 0x0807 | − | Reserved | |
| 0x0808 | CE2_1 | EMIF CE2 Space Control Register 1 | 1111 1111 1111 0011 |
| 0x0809 | CE2_2 | EMIF CE2 Space Control Register 2 | 1111 1111 1111 1111 |
| 0x080A | CE3_1 | EMIF CE3 Space Control Register 1 | 1111 1111 1111 0011 |
| 0x080B | CE3_2 | EMIF CE3 Space Control Register 2 | 1111 1111 1111 1111 |
| 0x080C | SDC1 | EMIF SDRAM Control Register 1 | 1111 0000 0000 0000 |
| 0x080D | SDC2 | EMIF SDRAM Control Register 2 | 0000 0011 0100 1000 |
| 0x080E | SDRC1 | EMIF SDRAM Refresh Control Register 1 | 1100 0101 1101 1100 |
| 0x080F | SDRC2 | EMIF SDRAM Refresh Control Register 2 | 0000 0000 0101 1101 |
| 0x0810 | SDX1 | EMIF SDRAM Extension Register 1 | 0101 1111 1101 1111 |
| 0x0811 | SDX2 | EMIF SDRAM Extension Register 2 | 0000 0000 0001 0111 |
| 0x0812 to 0x0821 | − | Reserved | |
| 0x0822 | CE1_SC1 | EMIF CE1 Secondary Control Register 1 | 0000 0000 0000 0010 |
| 0x0823 | CE1_SC2 | EMIF CE1 Secondary Control Register 2 | 0000 0000 0000 0000 |
| 0x0824 | CE0_SC1 | EMIF CE0 Secondary Control Register 1 | 0000 0000 0000 0010 |
| 0x0825 | CE0_SC2 | EMIF CE0 Secondary Control Register 2 | 0000 0000 0000 0000 |
| 0x0826 | − | Reserved | |
| 0x0827 | − | Reserved | |
| 0x0828 | CE2_SC1 | EMIF CE2 Secondary Control Register 1 | 0000 0000 0000 0010 |
| 0x0829 | CE2_SC2 | EMIF CE2 Secondary Control Register 2 | 0000 0000 0000 0000 |
| 0x082A | CE3_SC1 | EMIF CE3 Secondary Control Register 1 | 0000 0000 0000 0010 |
| 0x082B | CE3_SC2 | EMIF CE3 Secondary Control Register 2 | 0000 0000 0000 0000 |
| 0x082C to 0x0839 | − | Reserved | |
| 0x0840 | CESCR1 | EMIF CE Size Control Register 1 | 0000 0000 0000 0000 |
| 0x0841 | CESCR2 | EMIF CE Size Control Register 2 | 0000 0000 0000 0000 |

† x denotes a “don’t care.”

**Table 3−58. DMA Configuration Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| **GLOBAL REGISTER** | | | |
| 0x0E00 | DMA_GCR(2:0) | DMA Global Control Register | 000 |
| 0x0E01 | DMA_GTCR(3:0) | DMA Global Timeout Control Register | 0000 |
| **CHANNEL #0 REGISTERS** | | | |
| 0x0C00 | DMA_CSDP0 | DMA Channel 0 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0C01 | DMA_CCR0(15:0) | DMA Channel 0 Control Register | 0000 0000 0000 0000 |
| 0x0C02 | DMA_CICR0(5:0) | DMA Channel 0 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0C03 | DMA_CSR0(6:0) | DMA Channel 0 Status register | 00 0000 |
| 0x0C04 | DMA_CSSA_L0 | DMA Channel 0 Source Start Address, lower bits, register | Undefined |
| 0x0C05 | DMA_CSSA_U0 | DMA Channel 0 Source Start Address, upper bits, register | Undefined |
| 0x0C06 | DMA_CDSA_L0 | DMA Channel 0 Source Destination Address, lower bits, register | Undefined |
| 0x0C07 | DMA_CDSA_U0 | DMA Channel 0 Source Destination Address, upper bits, register | Undefined |
| 0x0C08 | DMA_CEN0 | DMA Channel 0 Element Number register | Undefined |
| 0x0C09 | DMA_CFN0 | DMA Channel 0 Frame Number register | Undefined |
| 0x0C0A | DMA_CSFI0 | DMA Channel 0 Source Frame Index register | Undefined |
| 0x0C0B | DMA_CSEI0 | DMA Channel 0 Source Element Index register | Undefined |
| 0x0C0C | DMA_CSAC0 | DMA Channel 0 Source Address Counter register | Undefined |
| 0x0C0D | DMA_CDAC0 | DMA Channel 0 Destination Address Counter register | Undefined |
| 0x0C0E | DMA_CDEI0 | DMA Channel 0 Destination Element Index register | Undefined |
| 0x0C0F | DMA_CDFI0 | DMA Channel 0 Destination Frame Index register | Undefined |
| **CHANNEL #1 REGISTERS** | | | |
| 0x0C20 | DMA_CSDP1 | DMA Channel 1 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0C21 | DMA_CCR1(15:0) | DMA Channel 1 Control Register | 0000 0000 0000 0000 |
| 0x0C22 | DMA_CICR1(5:0) | DMA Channel 1 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0C23 | DMA_CSR1(6:0) | DMA Channel 1 Status register | 00 0000 |
| 0x0C24 | DMA_CSSA_L1 | DMA Channel 1 Source Start Address, lower bits, register | Undefined |
| 0x0C25 | DMA_CSSA_U1 | DMA Channel 1 Source Start Address, upper bits, register | Undefined |
| 0x0C26 | DMA_CDSA_L1 | DMA Channel 1 Source Destination Address, lower bits, register | Undefined |
| 0x0C27 | DMA_CDSA_U1 | DMA Channel 1 Source Destination Address, upper bits, register | Undefined |
| 0x0C28 | DMA_CEN1 | DMA Channel 1 Element Number register | Undefined |
| 0x0C29 | DMA_CFN1 | DMA Channel 1 Frame Number register | Undefined |
| 0x0C2A | DMA_CSFI1 | DMA Channel 1 Source Frame Index register | Undefined |
| 0x0C2B | DMA_CSEI1 | DMA Channel 1 Source Element Index register | Undefined |
| 0x0C2C | DMA_CSAC1 | DMA Channel 1 Source Address Counter register | Undefined |
| 0x0C2D | DMA_CDAC1 | DMA Channel 1 Destination Address Counter register | Undefined |
| 0x0C2E | DMA_CDEI1 | DMA Channel 1 Destination Element Index register | Undefined |
| 0x0C2F | DMA_CDFI1 | DMA Channel 1 Destination Frame Index register | Undefined |

**Table 3−58. DMA Configuration Registers (Continued)**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| **CHANNEL #2 REGISTERS** | | | |
| 0x0C40 | DMA_CSDP2 | DMA Channel 2 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0C41 | DMA_CCR2(15:0) | DMA Channel 2 Control Register | 0000 0000 0000 0000 |
| 0x0C42 | DMA_CICR2(5:0) | DMA Channel 2 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0C43 | DMA_CSR2(6:0) | DMA Channel 2 Status register | 00 0000 |
| 0x0C44 | DMA_CSSA_L2 | DMA Channel 2 Source Start Address, lower bits, register | Undefined |
| 0x0C45 | DMA_CSSA_U2 | DMA Channel 2 Source Start Address, upper bits, register | Undefined |
| 0x0C46 | DMA_CDSA_L2 | DMA Channel 2 Source Destination Address, lower bits, register | Undefined |
| 0x0C47 | DMA_CDSA_U2 | DMA Channel 2 Source Destination Address, upper bits, register | Undefined |
| 0x0C48 | DMA_CEN2 | DMA Channel 2 Element Number register | Undefined |
| 0x0C49 | DMA_CFN2 | DMA Channel 2 Frame Number register | Undefined |
| 0x0C4A | DMA_CSFI2 | DMA Channel 2 Source Frame Index register | Undefined |
| 0x0C4B | DMA_CSEI2 | DMA Channel 2 Source Element Index register | Undefined |
| 0x0C4C | DMA_CSAC2 | DMA Channel 2 Source Address Counter register | Undefined |
| 0x0C4D | DMA_CDAC2 | DMA Channel 2 Destination Address Counter register | Undefined |
| 0x0C4E | DMA_CDEI2 | DMA Channel 2 Destination Element Index register | Undefined |
| 0x0C4F | DMA_CDFI2 | DMA Channel 2 Destination Frame Index register | Undefined |
| **CHANNEL #3 REGISTERS** | | | |
| 0x0C60 | DMA_CSDP3 | DMA Channel 3 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0C61 | DMA_CCR3(15:0) | DMA Channel 3 Control Register | 0000 0000 0000 0000 |
| 0x0C62 | DMA_CICR3(5:0) | DMA Channel 3 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0C63 | DMA_CSR3(6:0) | DMA Channel 3 Status register | 00 0000 |
| 0x0C64 | DMA_CSSA_L3 | DMA Channel 3 Source Start Address, lower bits, register | Undefined |
| 0x0C65 | DMA_CSSA_U3 | DMA Channel 3 Source Start Address, upper bits, register | Undefined |
| 0x0C66 | DMA_CDSA_L3 | DMA Channel 3 Source Destination Address, lower bits, register | Undefined |
| 0x0C67 | DMA_CDSA_U3 | DMA Channel 3 Source Destination Address, upper bits, register | Undefined |
| 0x0C68 | DMA_CEN3 | DMA Channel 3 Element Number register | Undefined |
| 0x0C69 | DMA_CFN3 | DMA Channel 3 Frame Number register | Undefined |
| 0x0C6A | DMA_CSFI3 | DMA Channel 3 Source Frame Index register | Undefined |
| 0x0C6B | DMA_CSEI3 | DMA Channel 3 Source Element Index register | Undefined |
| 0x0C6C | DMA_CSAC3 | DMA Channel 3 Source Address Counter register | Undefined |
| 0x0C6D | DMA_CDAC3 | DMA Channel 3 Destination Address Counter register | Undefined |
| 0x0C6E | DMA_CDEI3 | DMA Channel 3 Destination Element Index register | Undefined |
| 0x0C6F | DMA_CDFI3 | DMA Channel 3 Destination Frame Index register | Undefined |

**Table 3−58. DMA Configuration Registers (Continued)**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| **CHANNEL #4 REGISTERS** | | | |
| 0x0C80 | DMA_CSDP4 | DMA Channel 4 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0C81 | DMA_CCR4(15:0) | DMA Channel 4 Control Register | 0000 0000 0000 0000 |
| 0x0C82 | DMA_CICR4(5:0) | DMA Channel 4 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0C83 | DMA_CSR4(6:0) | DMA Channel 4 Status register | 00 0000 |
| 0x0C84 | DMA_CSSA_L4 | DMA Channel 4 Source Start Address, lower bits, register | Undefined |
| 0x0C85 | DMA_CSSA_U4 | DMA Channel 4 Source Start Address, upper bits, register | Undefined |
| 0x0C86 | DMA_CDSA_L4 | DMA Channel 4 Source Destination Address, lower bits, register | Undefined |
| 0x0C87 | DMA_CDSA_U4 | DMA Channel 4 Source Destination Address, upper bits, register | Undefined |
| 0x0C88 | DMA_CEN4 | DMA Channel 4 Element Number register | Undefined |
| 0x0C89 | DMA_CFN4 | DMA Channel 4 Frame Number register | Undefined |
| 0x0C8A | DMA_CSFI4 | DMA Channel 4 Source Frame Index register | Undefined |
| 0x0C8B | DMA_CSEI4 | DMA Channel 4 Source Element Index register | Undefined |
| 0x0C8C | DMA_CSAC4 | DMA Channel 4 Source Address Counter register | Undefined |
| 0x0C8D | DMA_CDAC4 | DMA Channel 4 destination Address Counter register | Undefined |
| 0x0C8E | DMA_CDEI4 | DMA Channel 4 Destination Element Index register | Undefined |
| 0x0C8F | DMA_CDFI4 | DMA Channel 4 Destination Frame Index register | Undefined |
| **CHANNEL #5 REGISTERS** | | | |
| 0x0CA0 | DMA_CSDP5 | DMA Channel 5 Source Destination Parameters Register | 0000 0000 0000 0000 |
| 0x0CA1 | DMA_CCR5(15:0) | DMA Channel 5 Control Register | 0000 0000 0000 0000 |
| 0x0CA2 | DMA_CICR5(5:0) | DMA Channel 5 Interrupt Control register | 0000 0001 1000 0011 |
| 0x0CA3 | DMA_CSR5(6:0) | DMA Channel 5 Status register | 00 0000 |
| 0x0CA4 | DMA_CSSA_L5 | DMA Channel 5 Source Start Address, lower bits, register | Undefined |
| 0x0CA5 | DMA_CSSA_U5 | DMA Channel 5 Source Start Address, upper bits, register | Undefined |
| 0x0CA6 | DMA_CDSA_L5 | DMA Channel 5 Source Destination Address, lower bits, register | Undefined |
| 0x0CA7 | DMA_CDSA_U5 | DMA Channel 5 Source Destination Address, upper bits, register | Undefined |
| 0x0CA8 | DMA_CEN5 | DMA Channel 5 Element Number register | Undefined |
| 0x0CA9 | DMA_CFN5 | DMA Channel 5 Frame Number register | Undefined |
| 0x0CAA | DMA_CSFI5 | DMA Channel 5 Source Frame Index register | Undefined |
| 0x0CAB | DMA_CSEI5 | DMA Channel 5 Source Element Index register | Undefined |
| 0x0CAC | DMA_CSAC5 | DMA Channel 5 Source Address Counter register | Undefined |
| 0x0CAD | DMA_CDAC5 | DMA Channel 5 Destination Address Counter register | Undefined |
| 0x0CAE | DMA_CDEI5 | DMA Channel 5 Destination Element Index register | Undefined |
| 0x0CAF | DMA_CDFI5 | DMA Channel 5 Destination Frame Index register | Undefined |

**Table 3−59. Instruction Cache Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x1400 | ICGC | ICache Global Control Register |
| 0x1401 | ICFLARL | ICache Flush Line Address Register Low Part |
| 0x1402 | ICFLARH | ICache Flush Line Address Register High Part |
| 0x1409 | ICWMC | ICache Way Miss-Counter Register |

**Table 3−60. Trace FIFO†**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION |
|---|---|---|
| 0x2000 − 0x203F | TRC00 − TRC63 | Trace Register Discontinuity Section |
| 0x2040 − 0x204F | TRC64 − TRC79 | Trace Register Last PC Section |
| 0x2050 | TRC_LPCOFFSET1 | Trace LPC Offset Register 1 |
| 0x2051 | TRC_LPCOFFSET2 | Trace LPC Offset Register 2 |
| 0x2052 | TRC_PTR | Trace Pointer Register |
| 0x2053 | TRC_CNTL | Trace Control Register |
| 0x2054 | TRC_ID | Trace ID Register |

† The Trace FIFO registers are used by the emulator only and do not require any intervention from the user.

**Table 3−61. Timer Signal Selection Register**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x8000 | TSSR | Timer Signal Selection Register | 0000 0000 0000 0000 |

**Table 3−62. Timers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x1000 | GPTPID1_0 | Peripheral ID register 1, Timer #0 | 0000 0111 0000 0001 |
| 0x1001 | GPTPID2_0 | Peripheral ID register 2, Timer #0 | 0000 0000 0000 0001 |
| 0x1002 | GPTEMU_0 | Emulation Management Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1003 | GPTCLK_0 | Timer Clock Speed Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1004 | GPTGPINT_0 | GPIO Interrupt Control Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1005 | GPTGPEN_0 | GPIO Enable Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1006 | GPTGPDAT_0 | GPIO Data Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1007 | GPTGPDIR_0 | GPIO Direction Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1008 | GPTCNT1_0 | Timer Counter 1 Register, Timer #0 | 0000 0000 0000 0000 |
| 0x1009 | GPTCNT2_0 | Timer Counter 2 Register, Timer #0 | 0000 0000 0000 0000 |
| 0x100A | GPTCNT3_0 | Timer Counter 3 Register, Timer #0 | 0000 0000 0000 0000 |
| 0x100B | GPTCNT4_0 | Timer Counter 4 Register, Timer #0 | 0000 0000 0000 0000 |
| 0x100C | GPTPRD1_0 | Period Register 1, Timer #0 | 0000 0000 0000 0000 |
| 0x100D | GPTPRD2_0 | Period Register 2, Timer #0 | 0000 0000 0000 0000 |
| 0x100E | GPTPRD3_0 | Period Register 3, Timer #0 | 0000 0000 0000 0000 |
| 0x100F | GPTPRD4_0 | Period Register 4, Timer #0 | 0000 0000 0000 0000 |
| 0x1010 | GPTCTL1_0 | Timer Control Register 1, Timer #0 | 0000 0000 0000 0000 |
| 0x1011 | GPTCTL2_0 | Timer Control Register 2, Timer #0 | 0000 0000 0000 0000 |
| 0x1012 | GPTGCTL1_0 | Global Timer Control Register 1, Timer #0 | 0000 0000 0000 0000 |
| 0x1013 | | Reserved | |
| 0x2400 | GPTPID1_1 | Peripheral ID register 1, Timer #1 | 0000 0111 0000 0001 |
| 0x2401 | GPTPID2_1 | Peripheral ID register 2, Timer #1 | 0000 0000 0000 0001 |
| 0x2402 | GPTEMU_1 | Emulation Management Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2403 | GPTCLK_1 | Timer Clock Speed Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2404 | GPTGPINT_1 | GPIO Interrupt Control Register, Timer #1 | 0000 0000 0000 0000 |

**Table 3−62. Timers (Continued)**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x2405 | GPTGPEN_1 | GPIO Enable Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2406 | GPTGPDAT_1 | GPIO Data Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2407 | GPTGPDIR_1 | GPIO Direction Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2408 | GPTCNT1_1 | Timer Counter 1 Register, Timer #1 | 0000 0000 0000 0000 |
| 0x2409 | GPTCNT2_1 | Timer Counter 2 Register, Timer #1 | 0000 0000 0000 0000 |
| 0x240A | GPTCNT3_1 | Timer Counter 3 Register, Timer #1 | 0000 0000 0000 0000 |
| 0x240B | GPTCNT4_1 | Timer Counter 4 Register, Timer #1 | 0000 0000 0000 0000 |
| 0x240C | GPTPRD1_1 | Period Register 1, Timer #1 | 0000 0000 0000 0000 |
| 0x240D | GPTPRD2_1 | Period Register 2, Timer #1 | 0000 0000 0000 0000 |
| 0x240E | GPTPRD3_1 | Period Register 3, Timer #1 | 0000 0000 0000 0000 |
| 0x240F | GPTPRD4_1 | Period Register 4, Timer #1 | 0000 0000 0000 0000 |
| 0x2410 | GPTCTL1_1 | Timer Control Register 1, Timer #1 | 0000 0000 0000 0000 |
| 0x2411 | GPTCTL2_1 | Timer Control Register 2, Timer #1 | 0000 0000 0000 0000 |
| 0x2412 | GPTGCTL1_1 | Global Timer Control Register 1, Timer #1 | 0000 0000 0000 0000 |
| 0x2413 | | Reserved | |
| 0x4000 | WDTPID1 | Peripheral ID register 1, Watchdog Timer | 0000 0111 0000 0001 |
| 0x4001 | WDTPID2 | Peripheral ID register 2, Watchdog Timer | 0000 0000 0000 0001 |
| 0x4002 | WDTEMU | Emulation Management Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4003 | WDTCLK | Timer Clock Speed Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4004 | WDTGPINT | GPIO Interrupt Control Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4005 | WDTGPEN | GPIO Enable Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4006 | WDTGPDAT | GPIO Data Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4007 | WDTGPDIR | GPIO Direction Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4008 | WDTCNT1 | Timer Counter 1 Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4009 | WDTCNT2 | Timer Counter 2 Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400A | WDTCNT3 | Timer Counter 3 Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400B | WDTCNT4 | Timer Counter 4 Register, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400C | WDTPRD1 | Period Register 1, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400D | WDTPRD2 | Period Register 2, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400E | WDTPRD3 | Period Register 3, Watchdog Timer | 0000 0000 0000 0000 |
| 0x400F | WDTPRD4 | Period Register 4, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4010 | WDTCTL1 | Timer Control Register 1, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4011 | WDTCTL2 | Timer Control Register 2, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4012 | WDTGCTL1 | Global Timer Control Register 1, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4013 | | Reserved | |
| 0x4014 | WDTWCTL1 | WD Timer Control Register 1, Watchdog Timer | 0000 0000 0000 0000 |
| 0x4015 | WDTWCTL2 | WD Timer Control Register 2, Watchdog Timer | 0000 0000 0000 0000 |

**Table 3−63. Multichannel Serial Port #0**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x2800 | DRR1_0 | Data Receive Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x2801 | DRR2_0 | Data Receive Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x2802 | DXR1_0 | Data Transmit Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x2803 | DXR2_0 | Data Transmit Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x2804 | SPCR1_0 | Serial Port Control Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x2805 | SPCR2_0 | Serial Port Control Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x2806 | RCR1_0 | Receive Control Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x2807 | RCR2_0 | Receive Control Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x2808 | XCR1_0 | Transmit Control Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x2809 | XCR2_0 | Transmit Control Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x280A | SRGR1_0 | Sample Rate Generator Register 1, McBSP #0 | 0000 0000 0000 0001 |
| 0x280B | SRGR2_0 | Sample Rate Generator Register 2, McBSP #0 | 0010 0000 0000 0000 |
| 0x280C | MCR1_0 | Multichannel Control Register 1, McBSP #0 | 0000 0000 0000 0000 |
| 0x280D | MCR2_0 | Multichannel Control Register 2, McBSP #0 | 0000 0000 0000 0000 |
| 0x280E | RCERA_0 | Receive Channel Enable Register Partition A, McBSP #0 | 0000 0000 0000 0000 |
| 0x280F | RCERB_0 | Receive Channel Enable Register Partition B, McBSP #0 | 0000 0000 0000 0000 |
| 0x2810 | XCERA_0 | Transmit Channel Enable Register Partition A, McBSP #0 | 0000 0000 0000 0000 |
| 0x2811 | XCERB_0 | Transmit Channel Enable Register Partition B, McBSP #0 | 0000 0000 0000 0000 |
| 0x2812 | PCR0 | Pin Control Register, McBSP #0 | 0000 0000 0000 0000 |
| 0x2813 | | Reserved | |
| 0x2814 | RCERC_0 | Receive Channel Enable Register Partition C, McBSP #0 | 0000 0000 0000 0000 |
| 0x2815 | RCERD_0 | Receive Channel Enable Register Partition D, McBSP #0 | 0000 0000 0000 0000 |
| 0x2816 | XCERC_0 | Transmit Channel Enable Register Partition C, McBSP #0 | 0000 0000 0000 0000 |
| 0x2817 | XCERD_0 | Transmit Channel Enable Register Partition D, McBSP #0 | 0000 0000 0000 0000 |
| 0x2818 | RCERE_0 | Receive Channel Enable Register Partition E, McBSP #0 | 0000 0000 0000 0000 |
| 0x2819 | RCERF_0 | Receive Channel Enable Register Partition F, McBSP #0 | 0000 0000 0000 0000 |
| 0x281A | XCERE_0 | Transmit Channel Enable Register Partition E, McBSP #0 | 0000 0000 0000 0000 |
| 0x281B | XCERF_0 | Transmit Channel Enable Register Partition F, McBSP #0 | 0000 0000 0000 0000 |
| 0x281C | RCERG_0 | Receive Channel Enable Register Partition G, McBSP #0 | 0000 0000 0000 0000 |
| 0x281D | RCERH_0 | Receive Channel Enable Register Partition H, McBSP #0 | 0000 0000 0000 0000 |
| 0x281E | XCERG_0 | Transmit Channel Enable Register Partition G, McBSP #0 | 0000 0000 0000 0000 |
| 0x281F | XCERH_0 | Transmit Channel Enable Register Partition H, McBSP #0 | 0000 0000 0000 0000 |

**Table 3−64. Multichannel Serial Port #1**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x2C00 | DRR1_1 | Data Receive Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C01 | DRR2_1 | Data Receive Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C02 | DXR1_1 | Data Transmit Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C03 | DXR2_1 | Data Transmit Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C04 | SPCR1_1 | Serial Port Control Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C05 | SPCR2_1 | Serial Port Control Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C06 | RCR1_1 | Receive Control Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C07 | RCR2_1 | Receive Control Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C08 | XCR1_1 | Transmit Control Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C09 | XCR2_1 | Transmit Control Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C0A | SRGR1_1 | Sample Rate Generator Register 1, McBSP #1 | 0000 0000 0000 0001 |
| 0x2C0B | SRGR2_1 | Sample Rate Generator Register 2, McBSP #1 | 0010 0000 0000 0000 |
| 0x2C0C | MCR1_1 | Multichannel Control Register 1, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C0D | MCR2_1 | Multichannel Control Register 2, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C0E | RCERA_1 | Receive Channel Enable Register Partition A, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C0F | RCERB_1 | Receive Channel Enable Register Partition B, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C10 | XCERA_1 | Transmit Channel Enable Register Partition A, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C11 | XCERB_1 | Transmit Channel Enable Register Partition B, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C12 | PCR1 | Pin Control Register, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C13 | | Reserved | |
| 0x2C14 | RCERC_1 | Receive Channel Enable Register Partition C, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C15 | RCERD_1 | Receive Channel Enable Register Partition D, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C16 | XCERC_1 | Transmit Channel Enable Register Partition C, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C17 | XCERD_1 | Transmit Channel Enable Register Partition D, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C18 | RCERE_1 | Receive Channel Enable Register Partition E, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C19 | RCERF_1 | Receive Channel Enable Register Partition F, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1A | XCERE_1 | Transmit Channel Enable Register Partition E, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1B | XCERF_1 | Transmit Channel Enable Register Partition F, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1C | RCERG_1 | Receive Channel Enable Register Partition G, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1D | RCERH_1 | Receive Channel Enable Register Partition H, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1E | XCERG_1 | Transmit Channel Enable Register Partition G, McBSP #1 | 0000 0000 0000 0000 |
| 0x2C1F | XCERH_1 | Transmit Channel Enable Register Partition H, McBSP #1 | 0000 0000 0000 0000 |

**Table 3−65. HPI**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0xA000 | PID LSW | PID [15:0] | — |
| 0xA001 | PID MSW | PID [31:16] | — |
| 0xA002 | HPWREMU | Power and Emulation Management Register | 0000 0000 0000 0000 |
| 0xA003 | | Reserved | |
| 0xA004 | HGPIOINT1 | General-Purpose I/O Interrupt Control Register 1 | 0000 0000 0000 0000 |
| 0xA005 | HGPIOINT2 | General-Purpose I/O Interrupt Control Register 2 | 0000 0000 0000 0000 |
| 0xA006 | HGPIOEN | General-Purpose I/O Enable Register | 0000 0000 0000 0000 |
| 0xA007 | | Reserved | |
| 0xA008 | HGPIODIR1 | General-Purpose I/O Direction Register 1 | 0000 0000 0000 0000 |
| 0xA009 | | Reserved | |
| 0xA00A | HGPIODAT1 | General-Purpose I/O Data Register 1 | xxxx xxxx xxxx xxxx |
| 0xA00B | | Reserved | |
| 0xA00C | HGPIODIR2 | General-Purpose I/O Direction Register 2 | 0000 0000 0000 0000 |
| 0xA00D | | Reserved | |
| 0xA00E | HGPIODAT2 | General-Purpose I/O Data Register 2 | xxxx xxxx xxxx xxxx |
| 0xA00F − 0xA017 | | Reserved | |
| 0xA018 | HPIC | Host Port Control Register | 0000 0000 0000 1000 |
| 0xA019 | | Reserved | |
| 0xA01A | HPIAW | Host Port Write Address Register | xxxx xxxx xxxx xxxx |
| 0xA01B | | Reserved | |
| 0xA01C | HPIAR | Host Port Read Address Register | xxxx xxxx xxxx xxxx |
| 0xA01D − 0xA020 | | Reserved | |

† x denotes a “don’t care.”

**Table 3−66. GPIO**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x3400 | IODIR | General-purpose I/O Direction Register | 0000 0000 0000 0000 |
| 0x3401 | IODATA | General-purpose I/O Data Register | 0000 0000 xxxx xxxx |
| 0x4400 | PGPIOEN0 | Parallel GPIO Enable Register 0 | 0000 0000 0000 0000 |
| 0x4401 | PGPIODIR0 | Parallel GPIO Direction Register 0 | 0000 0000 0000 0000 |
| 0x4402 | PGPIODAT0 | Parallel GPIO Data Register 0 | 0000 0000 0000 0000 |
| 0x4403 | PGPIOEN1 | Parallel GPIO Enable Register 1 | 0000 0000 0000 0000 |
| 0x4404 | PGPIODIR1 | Parallel GPIO Direction Register 1 | 0000 0000 0000 0000 |
| 0x4405 | PGPIODAT1 | Parallel GPIO Data Register 1 | 0000 0000 0000 0000 |
| 0x4406 | PGPIOEN2 | Parallel GPIO Enable Register 2 | 0000 0000 0000 0000 |
| 0x4407 | PGPIODIR2 | Parallel GPIO Direction Register 2 | 0000 0000 0000 0000 |
| 0x4408 | PGPIODAT2 | Parallel GPIO Data Register 2 | 0000 0000 0000 0000 |

† x denotes a “don’t care.”

**Table 3−67. Device Revision ID**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x3800 − 0x3803 | Die ID | Die ID | |
| 0x3804 | Chip ID (LSW) | Defines F# 3LS digits and PG rev | 1001 0100 0110 xxxx |
| 0x3805 | Chip ID (MSW) | Defines F# 3MS digits | 0000 0111 0101 0001 |
| 0x3806 | Sub ID | Defines subsytem ID | 0000 0000 0000 0000‡ |

† x denotes a “don’t care.”
‡ Denotes single core

**Table 3−68. $I^2C$**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x3C00 | I2COAR§ | $I^2C$ Own Address Register | 0000 0000 0000 0000 |
| 0x3C01 | I2CIER | $I^2C$ Interrupt Enable Register | 0000 0000 0000 0000 |
| 0x3C02 | I2CSTR | $I^2C$ Status Register | 0000 0100 0001 0000 |
| 0x3C03 | I2CCLKL | $I^2C$ Clock Low-Time Divider Register | 0000 0000 0000 0000 |
| 0x3C04 | I2CCLKH | $I^2C$ Clock High-Time Divider Register | 0000 0000 0000 0000 |
| 0x3C05 | I2CCNT | $I^2C$ Data Count | 0000 0000 0000 0000 |
| 0x3C06 | I2CDRR | $I^2C$ Data Receive Register | 0000 0000 0000 0000 |
| 0x3C07 | I2CSAR | $I^2C$ Slave Address Register | 0000 0011 1111 1111 |
| 0x3C08 | I2CDXR | $I^2C$ Data Transmit Register | 0000 0000 0000 0000 |
| 0x3C09 | I2CMDR | $I^2C$ Mode Register | 0000 0000 0000 0000 |
| 0x3C0A | I2CISRC | $I^2C$ Interrupt Source Register | 0000 0000 0000 0000 |
| 0x3C0B | I2CGPIO | $I^2C$ General-Purpose Register (Not supported) | xxxx xxxx xxxx xxxx |
| 0x3C0C | I2CPSC | $I^2C$ Prescaler Register | 0000 0000 0000 0000 |
| 0x3C0D | PID1 | $I^2C$ Peripheral ID Register 1 | − |
| 0x3C0E | PID2 | $I^2C$ Peripheral ID Register 2 | − |
| − | I2CXSR | $I^2C$ Transmit Shift Register | − |
| − | I2CRSR | $I^2C$ Receive Shift Register | − |

† x denotes a “don’t care.”
§ Specifies a unique 5501 $I^2C$ address. This register is fully programmable in both 7-bit and 10-bit modes and must be set by the programmer. When this device is used in conjunction with another $I^2C$ device, it must be programmed to the $I^2C$ slave address (01011A2A1A0) allocated by Philips Semiconductor for the 5501 (allocation number: 1946). A2, A1, and A0 are programmable address bits.

**Table 3−69. UART**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE† |
|---|---|---|---|
| 0x9C00 | URRBR/<br>URTHR/<br>URDLL‡ | Receive Buffer Register<br>Transmit Holding Register<br>Divisor Latch LSB Register | xxxx xxxx |
| 0x9C01 | URIER/<br>URDLM§ | Interrupt Enable Register<br>Divisor Latch MSB Register | 0000 0000 |
| 0x9C02 | URIIR/<br>URFCR¶ | Interrupt Identification Register<br>FIFO Control Register | 0000 0001<br>0000 0000 |
| 0x9C03 | URLCR | Line Control Register | 0000 0000 |
| 0x9C04 | URMCR | Modem Control Register | 0000 0000 |
| 0x9C05 | URLSR | Line Status Register | 0110 0000 |
| 0x9C07 | URSCR | Scratch Register | xxxx xxxx |
| 0x9C08 | URDLL‡ | Divisor Latch LSB Register | − |
| 0x9C09 | URDLM§ | Divisor Latch MSB Register | − |
| 0x9C0A | URPID1 | Peripheral ID Register (LSW) | − |
| 0x9C0B | URPID2 | Peripheral ID Register (MSW) | − |
| 0x9C0C | URPECR | Power and Emulation Control Register | 0000 0000 0000 0000 |

† x denotes a “don’t care.”

‡ The registers URRBR, URTHR, and URDLL share one address. URDLL also has a dedicated address. When using the dedicated address, the DLAB bit can be kept cleared, so that URRBR and URTHR are always selected at the shared address.

If DLAB = 0 : Read Only: URRBR
Write Only: URTHR

If DLAB = 1: Read/Write: URDLL

§ The registers URIER and URDLM share one address. URDLM also has a dedicated address. When using the dedicated address, the DLAB bit can be kept cleared, so that URIER is always selected at the shared address.

If DLAB = 0: Read/WRite: URIER

If DLAB = 1: Read/Write: URDLM

¶ The registers URIIR and URFCR share one address.

Read Only: URIIR

Write Only: URFCR

**Table 3−70. External Bus Selection**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x6C00 | XBSR | External Bus Selection Register | 0000 0000 0000 0000 |
| 0x8800 | XBCR | External Bus Control Register | 0000 0000 0000 0000 |

**Table 3−71. Clock Mode Register**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x8C00 | CLKMD | Clock Mode Control Register | 0000 0000 0000 0000 |

**Table 3−72. CLKOUT Selector Register**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x8400 | CLKOUTSR | CLKOUT Selection Register | 0000 0000 0000 0010 |

**Table 3−73. Clock Controller Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x1C80 | PLLCSR | PLL Control Status Register | 0000 0000 0000 0000 |
| 0x1C82 | CK3SEL | CLKOUT3 Select Register | 0000 0000 0000 1011 |
| 0x1C88 | PLLM | PLL Multiplier Control Register | 0000 0000 0000 0000 |
| 0x1C8A | PLLDIV0 | PLL Divider 0 Register | 1000 0000 0000 0000 |
| 0x1C8C | PLLDIV1 | PLL Divider 1 Register | 1000 0000 0000 0011 |
| 0x1C8E | PLLDIV2 | PLL Divider 2 Register | 1000 0000 0000 0011 |
| 0x1C90 | PLLDIV3 | PLL Divider 3 Register | 1000 0000 0000 0011 |
| 0x1C92 | OSCDIV1 | Oscillator Divider 1 Register | 0000 0000 0000 0000 |
| 0x1C98 | WKEN | Oscillator Wakeup Control Register | 0000 0000 0000 0000 |

**Table 3−74. IDLE Control Registers**

| WORD ADDRESS | REGISTER NAME | DESCRIPTION | RESET VALUE |
|---|---|---|---|
| 0x9400 | PICR | Peripheral IDLE Control Register | 0000 0000 0000 0000 |
| 0x9401 | PISTR | Peripheral IDLE Status Register | 0000 0000 0000 0000 |
| 0x9402 | MICR | Master IDLE Control Register | 0000 0000 0000 0000 |
| 0x9403 | MISR | Master IDLE Status Register | 0000 0000 0000 0000 |

## 3.16 Interrupts

Vector-relative locations and priorities for all internal and external interrupts are shown in Table 3−75. For more information on setting up and using interrupts, please refer to the *TMS320C55x DSP CPU Reference Guide* (literature number SPRU371).

**Table 3−75. Interrupt Table**

| NAME | SOFTWARE (TRAP) EQUIVALENT | LOCATION (HEX BYTES) | PRIORITY | FUNCTION |
|---|---|---|---|---|
| RESET | SINT0 | 0 | 0 | Reset (hardware and software) |
| NMI | SINT1 | 8 | 1 | Nonmaskable interrupt |
| INT0 | SINT2 | 10 | 3 | External interrupt #0 |
| INT2 | SINT3 | 18 | 5 | External interrupt #2 |
| TINT0 | SINT4 | 20 | 6 | Timer #0 interrupt |
| RINT0 | SINT5 | 28 | 7 | McBSP #0 receive interrupt |
| RINT1 | SINT6 | 30 | 9 | McBSP #1 receive interrupt |
| XINT1 | SINT7 | 38 | 10 | McBSP #1 transmit interrupt |
| LCKINT | SINT8 | 40 | 11 | PLL lock interrupt |
| DMAC1 | SINT9 | 48 | 13 | DMA Channel #1 interrupt |
| DSPINT | SINT10 | 50 | 14 | Interrupt from host |
| INT3/WDTINT† | SINT11 | 58 | 15 | External interrupt #3 or Watchdog timer interrupt |
| UART | SINT12 | 60 | 17 | UART interrupt |
| – | SINT13 | 68 | 18 | Software interrupt #13 |
| DMAC4 | SINT14 | 70 | 21 | DMA Channel #4 interrupt |
| DMAC5 | SINT15 | 78 | 22 | DMA Channel #5 interrupt |
| INT1 | SINT16 | 80 | 4 | External interrupt #1 |
| XINT0 | SINT17 | 88 | 8 | McBSP #0 transmit interrupt |
| DMAC0 | SINT18 | 90 | 12 | DMA Channel #0 interrupt |
| – | SINT19 | 98 | 16 | Software interrupt #19 |
| DMAC2 | SINT20 | A0 | 19 | DMA Channel #2 interrupt |
| DMAC3 | SINT21 | A8 | 20 | DMA Channel #3 interrupt |
| TINT1 | SINT22 | B0 | 23 | Timer #1 interrupt |
| IIC | SINT23 | B8 | 24 | $I^2C$ interrupt |
| BERR | SINT24 | C0 | 2 | Bus Error interrupt |
| DLOG | SINT25 | C8 | 25 | Data Log interrupt |
| RTOS | SINT26 | D0 | 26 | Real-time Operating System interrupt |
| – | SINT27 | D8 | 27 | Software interrupt #27 |
| – | SINT28 | E0 | 28 | Software interrupt #28 |
| – | SINT29 | E8 | 29 | Software interrupt #29 |
| – | SINT30 | F0 | 30 | Software interrupt #30 |
| | SINT31 | F8 | 31 | Software interrupt #31 |

† WDTINT is generated only when the WDT interrupt pin is connected to INT3 through the TSSR.

### 3.16.1 IFR and IER Registers

The Interrupt Enable Registers (IER0 and IER1) control which interrupts will be masked or enabled during normal operation. The Interrupt Flag Registers (IFR0 and IFR1) contain flags that indicate interrupts that are currently pending.

The Debug Interrupt Enable Registers (DBIER0 and DBIER1) are used only when the CPU is *halted* in the real-time emulation mode. If the CPU is *running* in real-time mode, the standard interrupt processing (IER0/1) is used and DBIER0/1 are ignored.

A maskable interrupt enabled in DBIER0/1 is defined as a time-critical interrupt. When the CPU is halted in the real-time mode, the only interrupts that are serviced are time-critical interrupts that are also enabled in an interrupt enable register (IER0 or IER1).

Write the DBIER0/1 to enable or disable time-critical interrupts. To enable an interrupt, set its corresponding bit. To disable an interrupt, clear its corresponding bit. Initialize these registers before using the real-time emulation mode.

A DSP hardware reset clears IFR0/1, IER0/1, and DBIER0/1 to 0. A software reset instruction clears IFR0/1 to 0 but does not affect IER0/1 and DBIER0/1.

The bit layouts of these registers for each interrupt are shown in Figure 3−51 and Figure 3−52. For more information on the IER, IFR, and DBIER registers, refer to the *TMS320C55x DSP CPU Reference Guide* (literature number SPRU371).

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| DMAC5 | DMAC4 | Reserved | UART | INT3/ WDTINT‡ | DSPINT | DMAC1 | Reserved |
| R/W, 0 | R/W, 0 | R/W, 0† | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| XINT1 | RINT1 | RINT0 | TINT0 | INT2 | INT0 | Reserved | |
| R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R/W, 0 | R, 0 | |

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

† This bit must be kept zero when writing to IER0.
‡ WDTINT is generated only when the WDT interrupt pin is connected to INT3 through the TSSR.

**Figure 3−51. IFR0, IER0, DBIFR0, and DBIER0 Registers Layout**

| 15 | | | | | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Reserved | | | | | | RTOS | DLOG | BERR |
| R, 0 | | | | | | R/W, 0 | R/W, 0 | R/W, 0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| $I^2C$ | TINT1 | DMAC3 | DMAC2 | INT4 | DMAC0 | XINT0 | INT1 |
| R/W, 0 | R/W, 0 | 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 3−52. IFR1, IER1, DBIFR1, and DBIER1 Registers Layout**

### 3.16.2 Interrupt Timing

The external interrupts ($\overline{\text{NMI}}$ and $\overline{\text{INT}}$) are synchronized to the CPU by way of a two-flip-flop synchronizer. The interrupt inputs are sampled on falling edges of the CPU clock. A sequence on the interrupt pin of 1–0–0–0 on consecutive cycles is required for an interrupt to be detected. Therefore, the minimum low pulse duration on the external interrupts on the 5501 is three CPU clock periods.

TIM0, TIM1, WDTOUT, and HPI.$\overline{\text{HAS}}$ can be configured to generate interrupts to the CPU. When they are used for this function, these pins will generate the interrupt associated with that module, i.e., TIM0 will generate TINT0, HPI.$\overline{\text{HAS}}$ will generate DSPINT, etc. Three SYSCLK1 clock cycles must be allowed to pass between consecutive interrupts generated using the HPI.$\overline{\text{HAS}}$ signal; otherwise, the last interrupt will be ignored (i.e., a sequence of 0−1−1−1−0 on consecutive cycles is required for consecutive interrupts). For more information on configuring TIM0, TIM1, WDTOUT, and HPI.$\overline{\text{HAS}}$ as interrupt pins, please refer to the *TMS320VC5501/5502 DSP Timers Reference Guide* (literature number SPRU618) for the timer pins and to the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620) for the HPI pin.

### 3.16.3 Interrupt Acknowledge

The $\overline{\text{IACK}}$ pin is used to indicate the receipt of an interrupt and that the program counter is fetching the interrupt vector location designated on the address bus. As the CPU fetches the first word or the software vector, it generates the $\overline{\text{IACK}}$ signal, which clears the appropriate interrupt flag bit. The $\overline{\text{IACK}}$ signal will go low for a total of one CPU clock pulse and then go high again. For maskable interrupts, note that the CPU will not jump to an interrupt service routine if the appropriate interrupt enable bit is not set; consequently, the $\overline{\text{IACK}}$ pin will not go low when the interrupt is generated.

## 3.17 Notice Concerning TCK

Under certain conditions, the emulation hardware may corrupt the emulation control state machine or may cause it to lose synchronization with the emulator software. When emulation commands fail as a result of the problem, Code Composer Studio™ Integrated Development Environment (IDE) may be unable to start or it may report errors when interacting with the TMS320C55x™ DSP (for example, when halting the CPU, reaching a breakpoint, etc.).

This phenomenon is observed when an erroneous clock edge is generated from the TCK signal inside the C55x™ DSP. This can be caused by several factors, acting independently or cumulatively:

- TCK transition times (as measured between 2.4 V and 0.8 V) in excess of 3 ns.
- Operating the C55x DSP in a socket, which can aggravate noise or glitches on the TCK input.
- Poor signal integrity on the TCK line from reflections or other layout issues.

A TCK edge that can cause this problem might look similar to the one shown in Figure 3−53. A TCK edge that does not cause the problem looks similar to the one shown in Figure 3−54. The key difference between the two figures is that Figure 3−54 has a clean and sharp transition whereas Figure 3−53 has a “knee” in the transition zone. Problematic TCK signals may not have a knee that is as pronounced as the one in Figure 3−53. Due to the TCK signal amplification inside the chip, any perturbation of the signal can create erroneous clock edges.

As a result of the faster edge transition, there is increased ringing in Figure 3−54. As long as the ringing does not cross logic input thresholds (0.8 V for falling edges, and 2.4 V for rising edges), this ringing is acceptable.

When examining a TCK signal for this issue, either in board simulation or on an actual board, it is very important to probe the TCK line as close to the DSP input pin as possible. In simulation, it should not be difficult to probe right at the DSP input. For most physical boards, this means using the via for the TCK pad on the back side of the board. Similarly, ground for the probe should come from one of the nearby ground pad vias to minimize EMI noise picked up by the probe.

Code Composer Studio, TMS320C55x, and C55x are trademarks of Texas Instruments.

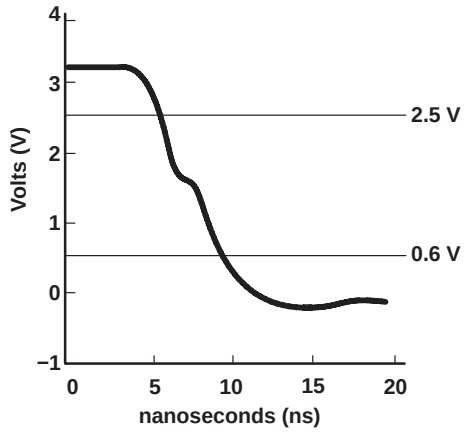



**Figure 3−53. Bad TCK Transition**

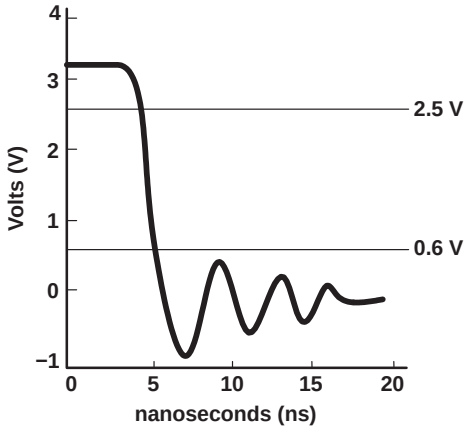



**Figure 3−54. Good TCK Transition**

As the problem may be caused by one or more of the above factors, one or more of the steps outlined below may be necessary to fix it:

- Avoid using a socket
- Ensure the board design achieves rise times and fall times of less than 3 ns with clean monotonic edges for the TCK signal.
- For designs where TCK is supplied by the emulation pod, implement noise filtering circuitry on the target board. A sample circuit is shown in Figure 3−55.

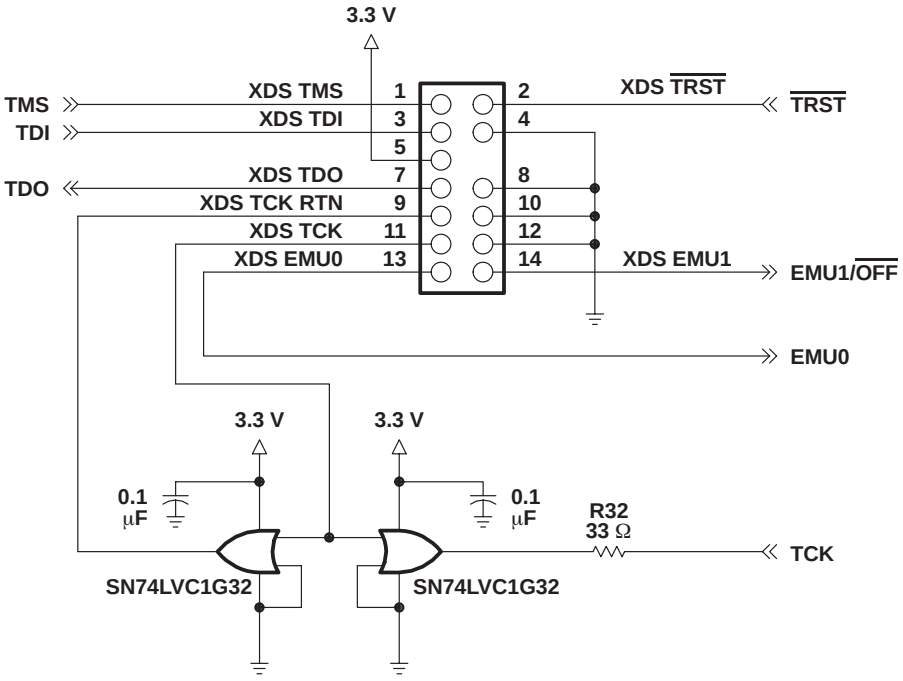



**Figure 3−55. Sample Noise Filtering Circuitry**

# 4 Support

## 4.1 Notices Concerning JTAG (IEEE 1149.1) Boundary Scan Test Capability

### *4.1.1 Initialization Requirements for Boundary Scan Test*

The TMS320VC5501 uses the JTAG port for boundary scan tests, emulation capability and factory test purposes. To use boundary scan test, the EMU0 and EMU1/$\overline{\text{OFF}}$ pins must be held HIGH through a rising edge of the $\overline{\text{TRST}}$ signal prior to the first scan. This operation selects the appropriate TAP control for boundary scan. If at any time during a boundary scan test a rising edge of $\overline{\text{TRST}}$ occurs when EMU0 or EMU1/$\overline{\text{OFF}}$ are not high, a factory test mode may be selected preventing boundary scan test from being completed. For this reason, it is recommended that EMU0 and EMU1/$\overline{\text{OFF}}$ be pulled or driven high at all times during boundary scan test.

### *4.1.2 Boundary Scan Description Language (BSDL) Model*

BSDL models are available on the web in the TMS320VC5501 product folder under the “simulation models” section.

## 4.2 Documentation Support

Extensive documentation supports all TMS320™ DSP family of devices from product announcement through applications development. The following types of documentation are available to support the design and use of the TMS320C5000™ platform of DSPs:

- Device-specific data sheets
- Complete user’s guides
- Development support tools
- Hardware and software application reports
- *MicroStar BGA ™ Packaging Reference Guide* (literature number SSYZ015)

TMS320C55x reference documentation that includes, but is not limited to, the following:

- *TMS320C55x DSP CPU Reference Guide* (literature number SPRU371)
- *TMS320C55x DSP Mnemonic Instruction Set Reference Guide* (literature number SPRU374)
- *TMS320C55x DSP Algebraic Instruction Set Reference Guide* (literature number SPRU375)
- *TMS320C55x DSP Programmer’s Guide* (literature number SPRU376)
- *TMS320C55x Assembly Language Tools User’s Guide* (literature number SPRU280)
- *TMS320VC5501/5502 DSP Instruction Cache Reference Guide* (literature number SPRU630)
- *TMS320VC5501/5502 DSP Timers Reference Guide* (literature number SPRU618)
- *TMS320VC5501/5502/5503/5507/5509 DSP Inter-Integrated Circuit (I2C) Module Reference Guide* (literature number SPRU146)
- *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)
- *TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide* (literature number SPRU613)
- *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592)
- *TMS320VC5501/5502 DSP External Memory Interface (EMIF) Reference Guide* (literature number SPRU621)
- *TMS320VC5501/5502 DSP Universal Asynchronous Receiver/Transmitter (UART) Reference Guide* (literature number SPRU597)
- *TMS320VC5502 and TMS320VC5501 Digital Signal Processors Silicon Errata* (literature number SPRZ020D or later)
- *TMS320VC5501/02 Power Consumption Summary* Application Report (literature number SPRA993)

TMS320 and TMS320C5000 are trademarks of Texas Instruments.

The reference guides describe in detail the TMS320C55x™ DSP products currently available and the hardware and software applications, including algorithms, for fixed-point TMS320 DSP family of devices.

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support digital signal processing research and education. The TMS320 DSP newsletter, *Details on Signal Processing*, is published quarterly and distributed to update TMS320 DSP customers on product information.

Information regarding TI DSP products is also available on the Worldwide Web at *http://www.ti.com* uniform resource locator (URL).

## 4.3 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., **TMS**320VC5501). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

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

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

**TMS** Fully qualified production device

Support tool development evolutionary flow:

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

**TMDS** Fully qualified development-support product

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

"Developmental product is intended for internal evaluation purposes."

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

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

# 5 Electrical Specifications

This section provides the absolute maximum ratings and the recommended operating conditions for the TMS320VC5501 DSP.

All electrical and switching characteristics in this data manual are valid over the recommended operating conditions unless otherwise specified.

## 5.1 Absolute Maximum Ratings

The list of absolute maximum ratings are specified over operating case temperature. Stresses beyond those listed under Section 5.2, Electrical Specifications, 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 Section 5.3, Recommended Operating Conditions, is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability. All supply voltage values (core and I/O) are with respect to $V_{SS}$. Figure 5−1 provides the test load circuit values for a 3.3-V device. Measured timing information contained in this data manual is based on the test load setup and conditions shown in Figure 5−1.

## 5.2 Electrical Specifications

This section provides the absolute maximum ratings for the TMS320VC5501 DSP.

Supply voltage I/O range, $DV_{DD}$ . . . . . − 0.3 V to 4.0 V
Supply voltage core range, $CV_{DD}$ . . . . . − 0.3 V to 2.0 V
Input voltage range, $V_I$ . . . . . − 0.3 V to 4.5 V
Output voltage range, $V_o$ . . . . . − 0.3 V to 4.5 V
Operating case temperature range, $T_C$ . . . . . −40°C to 85°C
Storage temperature range $T_{stg}$ . . . . . − 55°C to 150°C

## 5.3 Recommended Operating Conditions

This section provides the recommended operating conditions for the TMS320VC5501 DSP.

| | | | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|---|
| $DV_{DD}$ | Device supply voltage, I/O | | 3.0 | 3.3 | 3.6 | V |
| $CV_{DD}$ | Device supply voltage, core | | 1.20 | 1.26 | 1.32 | V |
| $PV_{DD}$ | Device supply voltage, PLL | | 3.0 | 3.3 | 3.6 | V |
| $V_{SS}$ | Supply voltage, GND | | | 0 | | V |
| $V_{IH}$ | High-level input voltage, I/O | Hysteresis inputs $DV_{DD}$ = 3.0 − 3.6 V | 2.2 | | $DV_{DD}$ + 0.3 | V |
| | | All other inputs $DV_{DD}$ = 3.0 − 3.6 V | 2 | | $DV_{DD}$ + 0.3 | |
| $V_{IL}$ | Low-level input voltage, I/O | Hysteresis inputs $DV_{DD}$ = 3.0 − 3.6 V | −0.3 | | 0.8 | V |
| | | All other inputs $DV_{DD}$ = 3.0 − 3.6 V | −0.3 | | 0.8 | |
| $I_{OH}$ | High-level output current | All outputs | | | − 300 | µA |
| $I_{OL}$ | Low-level output current | All outputs | | | 1.5 | mA |
| $T_C$ | Operating case temperature | | −40 | | 85 | °C |

## 5.4 Electrical Characteristics Over Recommended Operating Case Temperature Range (Unless Otherwise Noted)

| PARAMETER | | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{OH}$ | High-level output voltage | | $DV_{DD}$ = 3.3 ± 0.3 V, $I_{OH}$ = MAX | 2.4 | | | V |
| $V_{OL}$ | Low-level output voltage | | $I_{OL}$ = MAX | | | 0.4 | V |
| $I_{IZ}$ | Input current for outputs in high impedance | Output-only or input/output pins with bus holders | Bus holders enabled<br>$DV_{DD}$ = MAX, $V_O$ = $V_{SS}$ to $DV_{DD}$ | − 300 | | 300 | µA |
| | | All other output-only or input/output pins | $DV_{DD}$ = MAX, $V_I$ = $V_{SS}$ to $DV_{DD}$ | − 5 | | 5 | |
| $I_I$ | Input current | Input pins with internal pulldown | $DV_{DD}$ = MAX, $V_I$ = $V_{SS}$ to $DV_{DD}$ | − 5 | | 300 | µA |
| | | X2/CLKIN | $DV_{DD}$ = MAX, $V_I$ = $V_{SS}$ to $DV_{DD}$ | − 50 | | 50 | |
| | | Input pins with internal pullup | Pullup enabled<br>$DV_{DD}$ = MAX, $V_I$ = $V_{SS}$ to $DV_{DD}$ | − 300 | | 5 | |
| | | All other input-only pins | $DV_{DD}$ = MAX, $V_I$ = $V_{SS}$ to $DV_{DD}$ | − 5 | | 5 | |
| $I_{DDC}$ | $CV_{DD}$ supply current† | | $CV_{DD}$ = Nominal<br>CPU clock = 300 MHz<br>$T_C$ = 25°C | | | 239 | mA |
| $I_{DDD}$ | $DV_{DD}$ supply current† | | $DV_{DD}$ = Nominal<br>CPU clock = 300 MHz<br>$T_C$ = 25°C | | | 39 | mA |
| $I_{DDP}$ | $PV_{DD}$ supply current† | | $PV_{DD}$ = Nominal<br>20-MHz clock input,<br>APLL mode = x15 | | | 11 | mA |
| $C_i$ | Input capacitance | | | | 3 | | pF |
| $C_o$ | Output capacitance | | | | 3 | | pF |

† Current draw is highly application-dependent. The power numbers quoted here are for the sample application described in the *TMS320VC5501/02 Power Consumption Summary* application report (literature number SPRA993). The spreadsheet provided with the application report can be used to estimate the power consumption for a particular application. The spreadsheet also contains the current consumption that can be expected when running the DSP in its idle configurations.

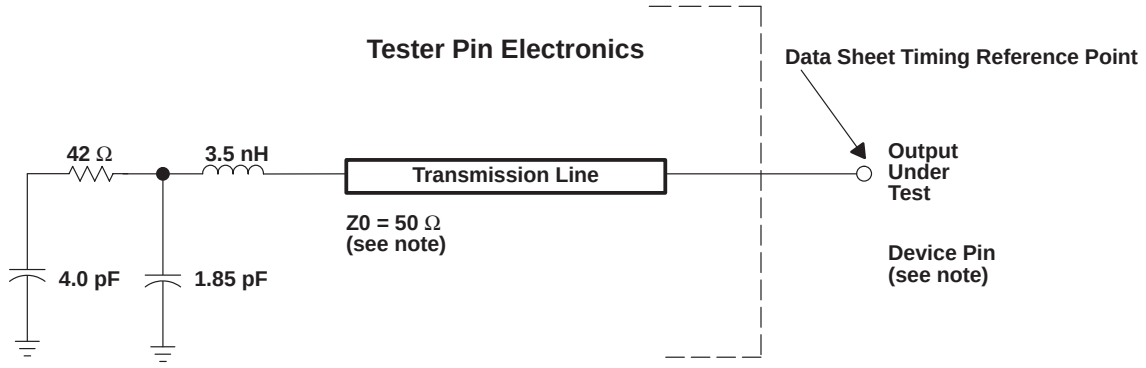
The sample application can be summarized as follows:

Case temperature: 25°C
APLL: 300 MHz
CPU: 85% utilization
- Instruction cache enabled
- CLKOUT off

EMIF: 75 MHz, 118% utilization, 100% writes, 32 bits, 100% switching
- ECLKOUT1 and ECLKOUT2: Off

HPI: 5Mwords/second, 100% utilization, 100% writes, 100% switching
DMA:
- Channel 0: 35% utilization, 32-bit elements, 100% switching (for internal memory to external memory transfers)
- Channel 1: 1.56% utilization, 32-bit elements, 100% switching (for internal memory to McBSP0 transfers)
- Channel 2: 1.56% utilization, 32-bit elements, 100% switching (for McBSP1 to internal memory transfers)
- Channels 3 and 4: 0% utilization (reserved for UART transfers)
- Channel 5: 60% utilization (for internal memory to internal memory transfers using Watchdog Timer event)

McBSP0: 25 MHz, 100% utilization, 100% switching
Timer0: 5 MHz, 100% utilization, 100% switching
Timer1: 10 MHz, 100% utilization, 100% switching
WD Timer: 30 MHz, 100% utilization, 100% switching
UART: 9600 baud, 100% utilization
All other peripherals use 0 MHz, 0% utilization




NOTE: The data sheet provides timing at the device pin. For output timing analysis, the tester pin electronics and its transmission line effects must be taken into account. A transmission line with a delay of 2 ns 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 5−1. 3.3-V Test Load Circuit**

## 5.5 Timing Parameter Symbology

Timing parameter symbols used in the timing requirements and switching characteristics tables are created in accordance with JEDEC Standard 100. To shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

Lowercase subscripts and their meanings:

| | |
|---|---|
| a | access time |
| c | cycle time (period) |
| d | delay time |
| dis | disable time |
| en | enable time |
| f | fall time |
| h | hold time |
| r | rise time |
| su | setup time |
| t | transition time |
| v | valid time |
| w | pulse duration (width) |
| X | Unknown, changing, or don't care level |

Letters and symbols and their meanings:

| | |
|---|---|
| H | High |
| L | Low |
| V | Valid |
| Z | High impedance |

## 5.6 Clock Options

This section provides the timing requirements and switching characteristics for the various clock options available on the 5501.

### *5.6.1 Internal System Oscillator With External Crystal*

The 5501 includes an internal oscillator which can be used in conjunction with an external crystal to generate the input clock to the DSP. The oscillator requires an external crystal connected across the X1 and X2/CLKIN pins. If the internal oscillator is not used, an external clock source must be applied to the X2/CLKIN pin and the X1 pin should be left unconnected. Since the internal oscillator can be used as a clock source to the PLL, the crystal oscillation frequency can be multiplied to generate the input clock to the different clock groups of the DSP.

GPIO4 is sampled on the rising edge of the reset signal to set the state of the CLKMD0 bit of the Clock Mode Control Register (CLKMD), which in turns, determines the clock source for the DSP. The CLKMD0 bit selects either the internal oscillator output (OSCOUT) or the X2/CLKIN pin as the input clock source for the DSP. If GPIO4 is low at reset, the CLKMD0 bit will be set to ‘0’ and the internal oscillator and the external crystal generate the input clock for the DSP. If GPIO4 is high, the CLKMD0 bit will be set to ‘1’ and the input clock will be taken directly from the X2/CLKIN pin.

The crystal should be in fundamental-mode operation, and parallel resonant, with a maximum effective series resistance (ESR) as specified in Table 5−1. The connection of the required circuit is shown in Figure 5−2. Under some conditions, all the components shown are not required. The capacitors, $C_1$ and $C_2$, should be chosen such that the equation below is satisfied. $C_L$ in the equation is the load specified for the crystal that is also specified in Table 5−1.

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

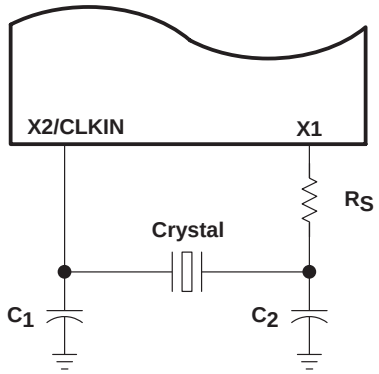



**Figure 5−2. Internal System Oscillator With External Crystal**

**Table 5−1. Recommended Crystal Parameters**

| FREQUENCY RANGE (MHz) | MAXIMUM ESR SPECIFICATIONS (Ω) | $C_{LOAD}$ (pF) | MAXIMUM $C_{SHUNT}$ (pF) | $R_S$ (kΩ) |
|---|---|---|---|---|
| 20−15 | 40 | 10 | 7 | 0 |
| 15−12 | 40 | 16 | 7 | 0 |
| 12−10 | 40 | 16 | 7 | 2.8 |
| 10−8 | 60 | 18 | 7 | 2.2 |
| 8−6 | 60 | 18 | 7 | 8.8 |
| 6−5 | 80 | 18 | 7 | 14 |

The recommended ESR is presented as a maximum, and theoretically, a crystal with a lower maximum ESR might seem to meet these specifications. However, it is recommended that crystals with actual maximum ESR specifications as shown in Table 5−1 be used since this will result in maximum crystal performance reliability.

### 5.6.2 Layout Considerations

Since parasitic capacitance, inductance, and resistance can be significant in this and any circuit, good PC board layout practices should always be observed when planning trace routing to the discrete components used in this oscillator circuit. Specifically, the crystal and the associated discrete components should be located as close to the DSP as physically possible. Also, X1 and X2/CLKIN traces should be separated as soon as possible after routing away from the DSP to minimize parasitic capacitance between them, and a ground trace should be run between these two signal lines. This also helps to minimize stray capacitance between these two signals.

### 5.6.3 Clock Generation in Bypass Mode (APLL Synthesis Disabled)

Table 5−2 and Table 5−3 assume testing over recommended operating conditions (see Figure 5−3).

**Table 5−2. CLKIN in Bypass Mode Timing Requirements**

| NO. | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|
| C7 | $t_{c(CI)}$ | Cycle time, CLKIN† | APLL Synthesis Disabled | 20 | ‡ | ns |
| C8 | $t_{f(CI)}$ | Fall time, CLKIN | | | 10 | ns |
| C9 | $t_{r(CI)}$ | Rise time, CLKIN | | | 10 | ns |
| C10 | $t_{w(CIL)}$ | Pulse duration, CLKIN low | | 0.4 * $t_{c(CI)}$ | | ns |
| C11 | $t_{w(CIH)}$ | Pulse duration, CLKIN high | | 0.4 * $t_{c(CI)}$ | | ns |

† If an external crystal is used, the X2/CLKIN cycle time is limited by the crystal frequency range listed in Table 5−1.

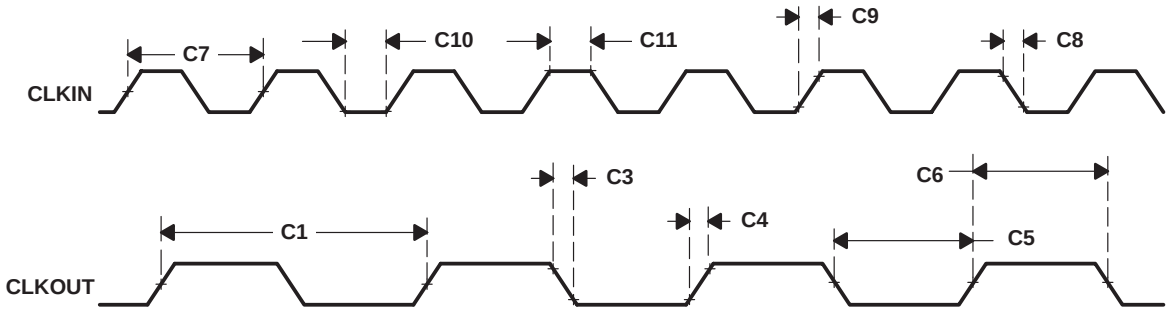
‡ This device utilizes a fully static design and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz.

**Table 5−3. CLKOUT in Bypass Mode Switching Characteristics**

| NO. | PARAMETER | | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| C1 | $t_{c(CO)}$ | Cycle time, CLKOUT | 20 | K * $t_{c(CI)}$§ | ¶ | ns |
| C3 | $t_{f(CO)}$ | Fall time, CLKOUT | | | 3 | ns |
| C4 | $t_{r(CO)}$ | Rise time, CLKOUT | | | 3 | ns |
| C5 | $t_{w(COL)}$ | Pulse duration, CLKOUT low | K * $t_{c(CI)}$/2 − 1 | | K * $t_{c(CI)}$/2 + 1 | ns |
| C6 | $t_{w(COH)}$ | Pulse duration, CLKOUT high | K * $t_{c(CI)}$/2 − 1 | | K * $t_{c(CI)}$/2 + 1 | ns |

§ K = divider ratio between CPU clock and system clock selected as CLKOUT. For example, when SYSCLK1 is selected as CLKOUT and SYSCLK1 is set to the CPU clock divided by four, use K = 4.

¶ This device utilizes a fully static design and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz.




NOTE: The relationship of CLKIN to CLKOUT depends on the system clock selected to drive CLKOUT. The waveform relationship shown in Figure 5−3 is intended to illustrate the timing parameters only and may differ based on configuration.

**Figure 5−3. Bypass Mode Clock Timings**

### 5.6.4 Clock Generation in Lock Mode (APLL Synthesis Enabled)

The frequency of the reference clock provided at the CLKIN pin can be multiplied by a synthesis factor of N to generate the internal CPU clock cycle. The synthesis factor is determined by:

$$N = \frac{M}{D_0}$$

where: M = the multiply factor set in the PLLM field of the PLL Multiplier Control Register (PLLM)
$D_0$ = the divide factor set in the PLLDIV0 field of the PLL Divider 0 Register (PLLDIV0)

Valid values for M are (multiply by) 2 to 15. Valid values for $D_0$ are (divide by) 1 to 32.

For detailed information on clock generation configuration, see Section 3.9, System Clock Generator.

Table 5−4 and Table 5−5 assume testing over recommended operating conditions (see Figure 5−4).

**Table 5−4. CLKIN in Lock Mode Timing Requirements**

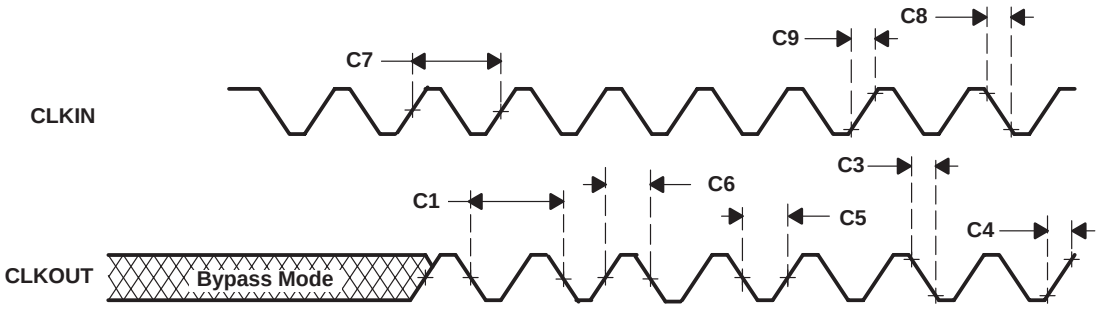
| NO. | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|
| C7 | $t_{c(CI)}$ | Cycle time, CLKIN† | APLL synthesis enabled | 10‡ | 83.3 | ns |
| C8 | $t_{f(CI)}$ | Fall time, CLKIN | | | 10 | ns |
| C9 | $t_{r(CI)}$ | Rise time, CLKIN | | | 10 | ns |

† If an external crystal is used, the X2/CLKIN cycle time is limited by the crystal frequency range listed in Table 5−1.
‡ The clock frequency synthesis factor and minimum CLKIN cycle time should be chosen such that the resulting CLKOUT cycle time is within the specified range [$t_{c(CO)}$].

**Table 5−5. CLKOUT in Lock Mode Switching Characteristics**

| NO. | PARAMETER | | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| C1 | $t_{c(CO)}$ | Cycle time, CLKOUT | 6.66 | K * $t_{c(CI)}$/N§ | 14.29 | ns |
| C3 | $t_{f(CO)}$ | Fall time, CLKOUT | | | 3 | ns |
| C4 | $t_{r(CO)}$ | Rise time, CLKOUT | | | 3 | ns |
| C5 | $t_{w(COL)}$ | Pulse duration, CLKOUT low | K * $t_{c(CI)}$/2N − 1 | | K * $t_{c(CI)}$/2N + 1 | ns |
| C6 | $t_{w(COH)}$ | Pulse duration, CLKOUT high | K * $t_{c(CI)}$/2N − 1 | | K * $t_{c(CI)}$/2N + 1 | ns |

§ N = Clock frequency synthesis factor. K = divider ratio between CPU clock and system clock selected as CLKOUT. For example, when SYSCLK1 is selected as CLKOUT and SYSCLK1 is set to the CPU clock divided by four, use K = 4.




NOTE: The waveform relationship of CLKIN to CLKOUT depends on the multiply and divide factors chosen for the APLL synthesis and on the system clock selected to drive CLKOUT. The waveform relationship shown in Figure 5−4 is intended to illustrate the timing parameters only and may differ based on configuration.

**Figure 5−4. External Multiply-by-N Clock Timings**

### 5.6.5 EMIF Clock Options

Table 5−6 through Table 5−8 assume testing over recommended operating conditions (see Figure 5−5 through Figure 5−7).

**Table 5−6. EMIF Timing Requirements for ECLKIN†‡**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| E7 | $t_{c(EKI)}$ | Cycle time, ECLKIN | 10 | 16P | ns |
| E8 | $t_{w(EKIH)}$ | Pulse duration, ECLKIN high | 0.4 * $t_{c(EKI)}$ | | ns |
| E9 | $t_{w(EKIL)}$ | Pulse duration, ECLKIN low | 0.4 * $t_{c(EKI)}$ | | ns |
| E10 | $t_{t(EKI)}$ | Transition time, ECLKIN | | 2 | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.
‡ The reference points for the rise and fall transitions are measured at $V_{IL}$ MAX and $V_{IH}$ MIN.

**Table 5−7. EMIF Switching Characteristics for ECLKOUT1§¶#**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| E1 | $t_{c(EKO1)}$ | Cycle time, ECLKOUT1 | E − 1 | E + 1 | ns |
| E2 | $t_{w(EKO1H)}$ | Pulse duration, ECLKOUT1 high | EH − 1 | EH + 1 | ns |
| E3 | $t_{w(EKO1L)}$ | Pulse duration, ECLKOUT1 low | EL − 1 | EL + 1 | ns |
| E4 | $t_{t(EKO1)}$ | Transition time, ECLKOUT1 | | 1 | ns |
| E5 | $t_{d(EKIH-EKO1H)}$ | Delay time, ECLKIN high to ECLKOUT1 high | 3 | 13 | ns |
| E6 | $t_{d(EKIL-EKO1L)}$ | Delay time, ECLKIN low to ECLKOUT1 low | 3 | 13 | ns |

§ The reference points for the rise and fall transitions are measured at $V_{OL}$ MAX and $V_{OH}$ MIN.
¶ E = the EMIF input clock (CPU clock, CPU/2 clock, or CPU/4 clock) period in ns for EMIF.
# EH is the high period of E (EMIF input clock period) in ns and EL is the low period of E (EMIF input clock period) in ns for EMIF.

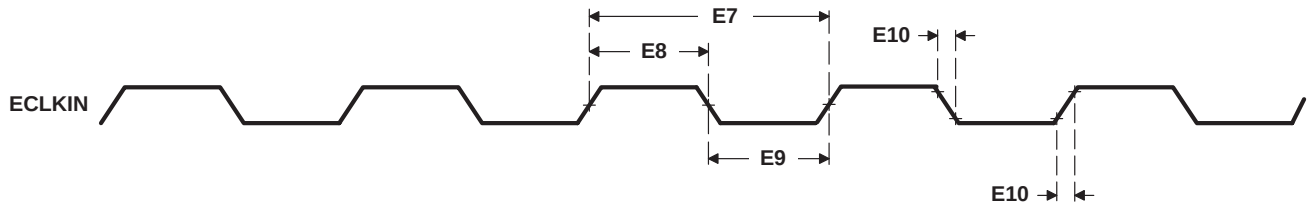



**Figure 5−5. ECLKIN Timings for EMIF**

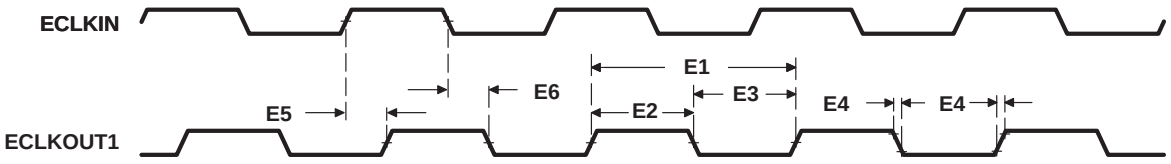



**Figure 5−6. ECLKOUT1 Timings for EMIF Module**

**Table 5−8. EMIF Switching Characteristics for ECLKOUT2†‡**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| **E11** | $t_{c(EKO2)}$ | Cycle time, ECLKOUT2 | NE − 1 | NE + 1 | ns |
| **E12** | $t_{w(EKO2H)}$ | Pulse duration, ECLKOUT2 high | 0.5NE − 1 | 0.5NE + 1 | ns |
| **E13** | $t_{w(EKO2L)}$ | Pulse duration, ECLKOUT2 low | 0.5NE − 1 | 0.5NE + 1 | ns |
| **E14** | $t_{t(EKO2)}$ | Transition time, ECLKOUT2 | | 1 | ns |
| **E15** | $t_{d(EKIH-EKO2H)}$ | Delay time, ECLKIN high to ECLKOUT2 high | 3 | 13 | ns |
| **E16** | $t_{d(EKIH-EKO2L)}$ | Delay time, ECLKIN high to ECLKOUT2 low | 3 | 13 | ns |

† The reference points for the rise and fall transitions are measured at $V_{OL}$ MAX and $V_{OH}$ MIN.
‡ E = the EMIF input clock (CPU clock, CPU/2 clock, or CPU/4 clock) period in ns for EMIF.
N = the EMIF input clock divider; N = 1, 2, or 4.

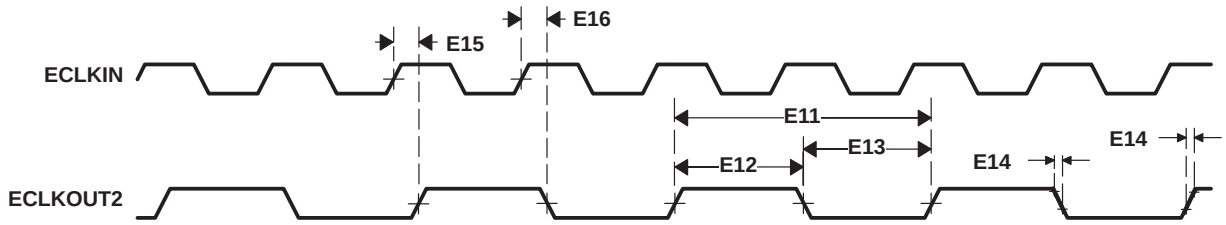



**Figure 5−7. ECLKOUT2 Timings for EMIF Module**

## 5.7 Memory Timings

### 5.7.1 Asynchronous Memory Timings

Table 5−9 and Table 5−10 assume testing over recommended operating conditions (see Figure 5−8 and Figure 5−9).

**Table 5−9. Asynchronous Memory Cycle Timing Requirements for ECLKIN†‡**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| A3 | $t_{su(EDV-AREH)}$ | Setup time, EMIF.Dx valid before EMIF.$\overline{ARE}$ high | 6 | | ns |
| A4 | $t_{h(AREH-EDV)}$ | Hold time, EMIF.Dx valid after EMIF.$\overline{ARE}$ high | 1 | | ns |
| A6 | $t_{su(ARDY-EKO1H)}$ | Setup time, EMIF.ARDY valid before ECLKOUT1 high | 3.5 | | ns |
| A7 | $t_{h(EKO1H-ARDY)}$ | Hold time, EMIF.ARDY valid after ECLKOUT1 high | 1 | | ns |

† To ensure data setup time, simply program the strobe width wide enough. EMIF.ARDY is internally synchronized. The EMIF.ARDY signal is recognized in the cycle for which the setup and hold time is met. To use EMIF.ARDY as an asynchronous input, the pulse width of the EMIF.ARDY signal should be wide enough (e.g., pulse width = 2E) to ensure setup and hold time is met.

‡ RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold. These parameters are programmed via the EMIF CE space control registers.

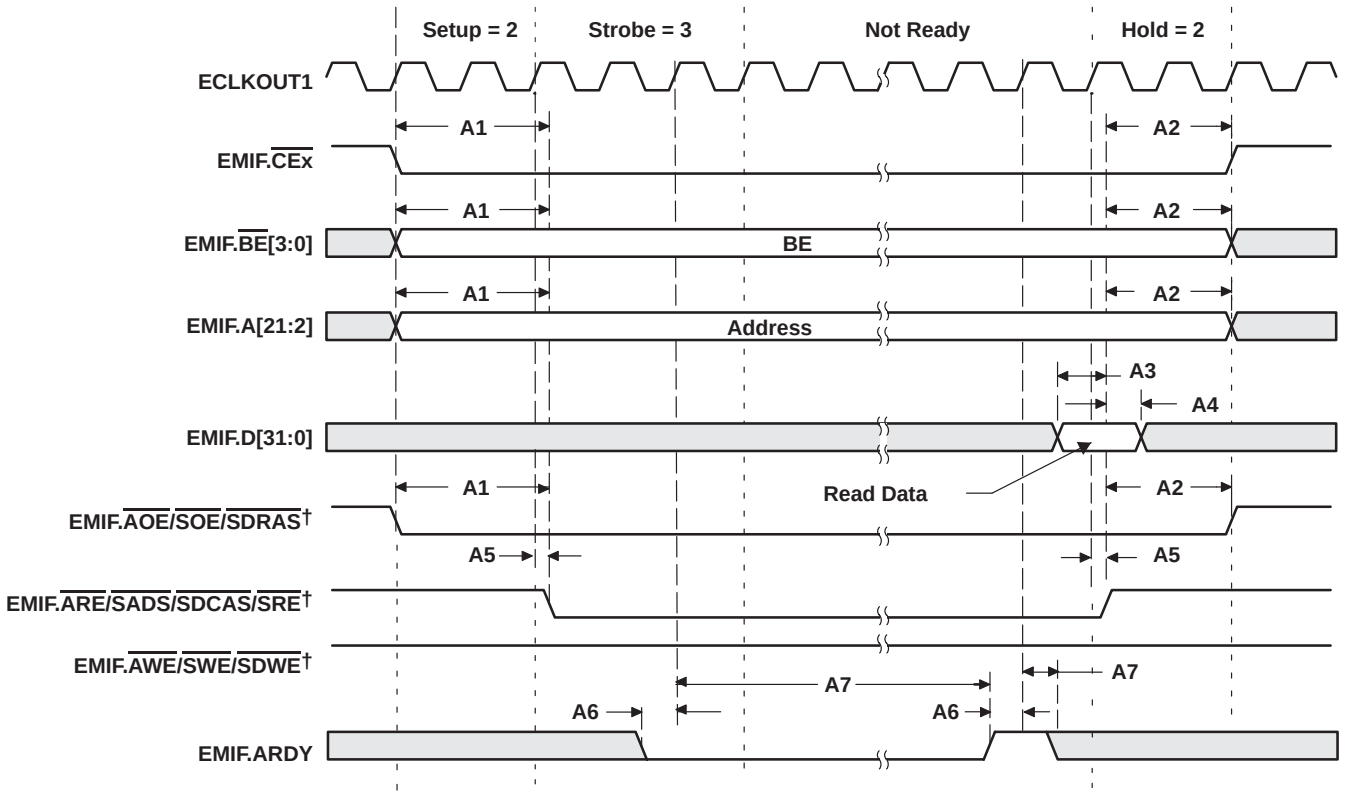
**Table 5−10. Asynchronous Memory Cycle Switching Characteristics for ECLKOUT1‡§¶**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| A1 | $t_{osu(SELV-AREL)}$ | Output setup time, select signals valid to EMIF.$\overline{ARE}$ low | RS * E − 1.5 | | ns |
| A2 | $t_{oh(AREH-SELIV)}$ | Output hold time, EMIF.$\overline{ARE}$ high to select signals invalid | RH * E − 1.5 | | ns |
| A5 | $t_{d(EKO1H-AREV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{ARE}$ valid | 1.5 | 5 | ns |
| A8 | $t_{osu(SELV-AWEL)}$ | Output setup time, select signals valid to EMIF.$\overline{AWE}$ low | WS * E − 1.5 | | ns |
| A9 | $t_{oh(AWEH-SELIV)}$ | Output hold time, EMIF.$\overline{AWE}$ high to select signals invalid | WH * E − 1.5 | | ns |
| A10 | $t_{d(EKO1H-AWEV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{AWE}$ valid | 1.5 | 5 | ns |

‡ RS = Read setup, RST = Read strobe, RH = Read hold, WS = Write setup, WST = Write strobe, WH = Write hold. These parameters are programmed via the EMIF CE space control registers.

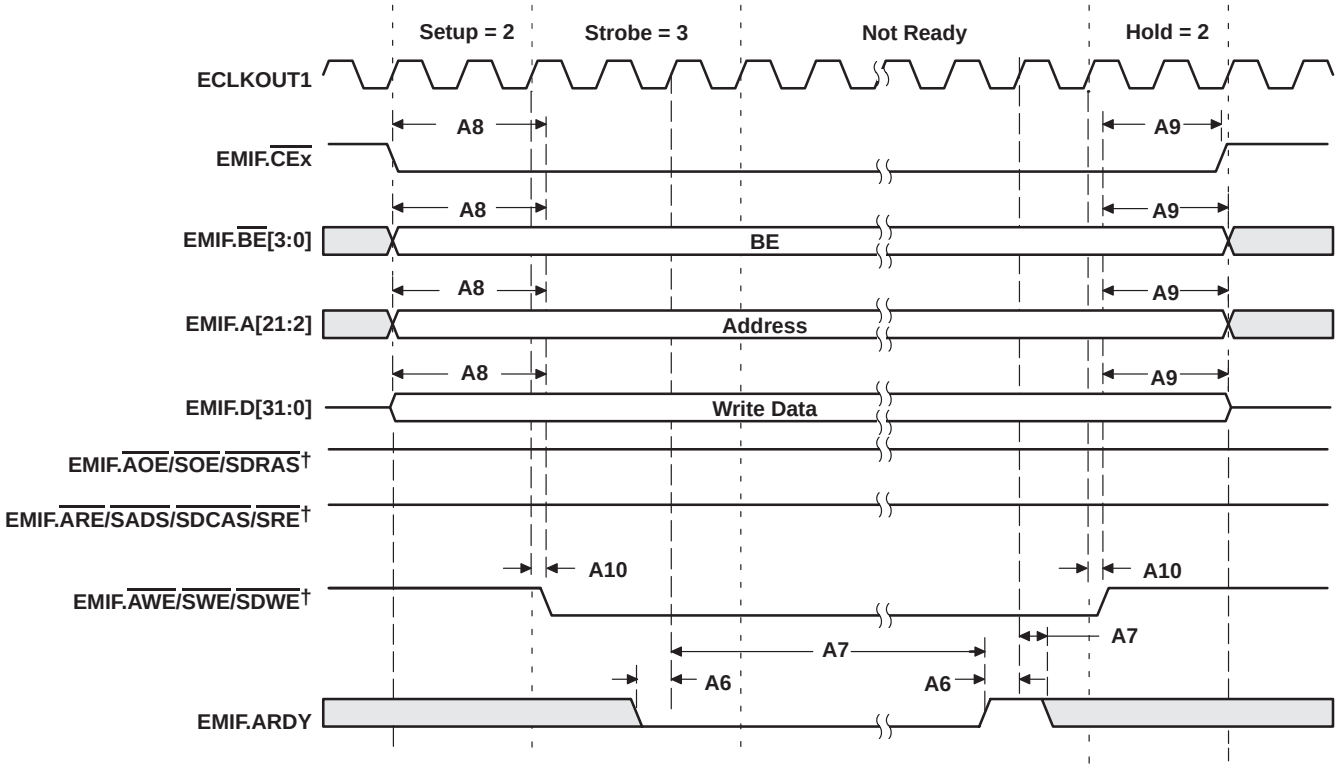
§ E = ECLKOUT1 period in ns for EMIF.

¶ Select signals for EMIF include: EMIF.$\overline{CE}$x, EMIF.$\overline{BE}$[3:0], EMIF.A[21:2], and EMIF.$\overline{AOE}$; and for EMIF writes, include EMIF.D[31:0].




† EMIF.AOE/SOE/SDRAS, EMIF.ARE/SADS/SDCAS/SRE, and EMIF.AWE/SWE/SDWE operate as EMIF.AOE (identified under select signals), EMIF.ARE, and EMIF.AWE, respectively, during asynchronous memory accesses.

**Figure 5−8. Asynchronous Memory Read Timings†**




† EMIF.AOE/SOE/SDRAS, EMIF.ARE/SADS/SDCAS/SRE, and EMIF.AWE/SWE/SDWE operate as EMIF.AOE (identified under select signals), EMIF.ARE, and EMIF.AWE, respectively, during asynchronous memory accesses.

**Figure 5−9. Asynchronous Memory Write Timings†**

### 5.7.2 Programmable Synchronous Interface Timings

Table 5−11 and Table 5−12 assume testing over recommended operating conditions (see Figure 5−10 through Figure 5−12).

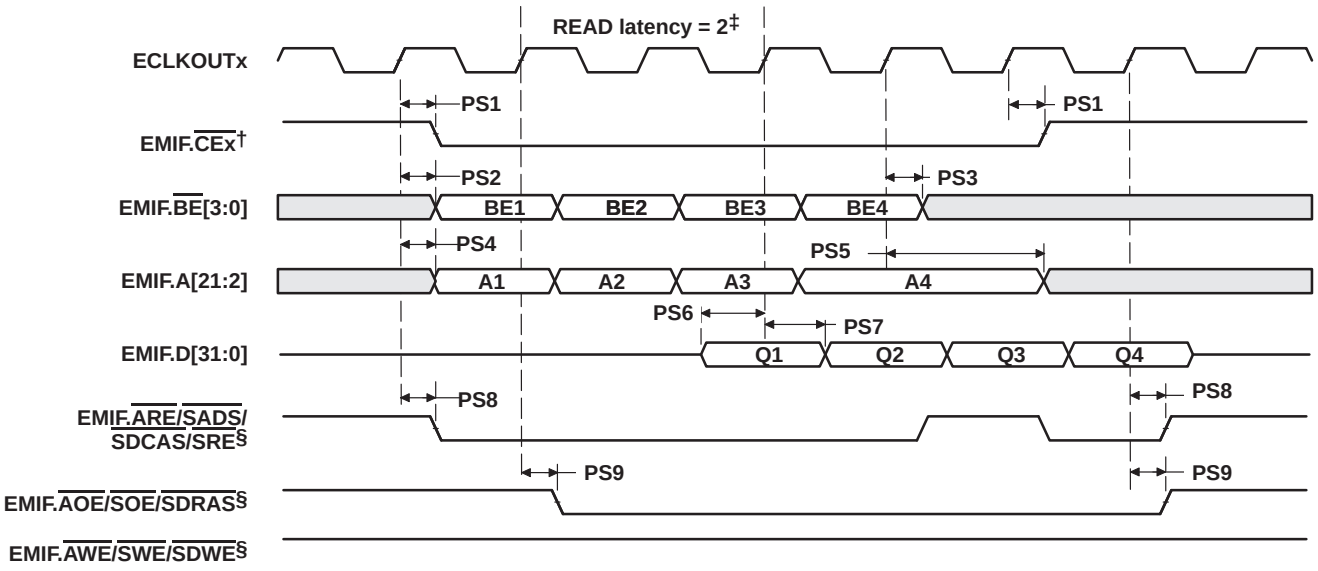
**Table 5−11. Programmable Synchronous Interface Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| PS6 | $t_{su(EDV-EKOxH)}$ | Setup time, read EMIF.Dx valid before ECLKOUTx high | 2 | | ns |
| PS7 | $t_{h(EKOxH-EDV)}$ | Hold time, read EMIF.Dx valid after ECLKOUTx high | 1.5 | | ns |

**Table 5−12. Programmable Synchronous Interface Switching Characteristics†**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| PS1 | $t_{d(EKOxH-CEV)}$ | Delay time, ECLKOUTx high to EMIF.$\overline{CEx}$ valid | 0.8 | 7 | ns |
| PS2 | $t_{d(EKOxH-BEV)}$ | Delay time, ECLKOUTx high to EMIF.$\overline{BEx}$ valid | | 7 | ns |
| PS3 | $t_{d(EKOxH-BEIV)}$ | Delay time, ECLKOUTx high to EMIF.$\overline{BEx}$ invalid | 0.8 | | ns |
| PS4 | $t_{d(EKOxH-EAV)}$ | Delay time, ECLKOUTx high to EMIF.Ax valid | | 7 | ns |
| PS5 | $t_{d(EKOxH-EAIV)}$ | Delay time, ECLKOUTx high to EMIF.Ax invalid | 0.8 | | ns |
| PS8 | $t_{d(EKOxH-ADSV)}$ | Delay time, ECLKOUTx high to EMIF.$\overline{SADS}$/$\overline{SRE}$ valid | 0.8 | 7 | ns |
| PS9 | $t_{d(EKOxH-OEV)}$ | Delay time, ECLKOUTx high to, EMIF.$\overline{SOE}$ valid | 0.8 | 7 | ns |
| PS10 | $t_{d(EKOxH-EDV)}$ | Delay time, ECLKOUTx high to EMIF.Dx valid | | 7 | ns |
| PS11 | $t_{d(EKOxH-EDIV)}$ | Delay time, ECLKOUTx high to EMIF.Dx invalid | 0.8 | | ns |
| PS12 | $t_{d(EKOxH-WEV)}$ | Delay time, ECLKOUTx high to EMIF.$\overline{SWE}$ valid | 0.8 | 7 | ns |

† The following parameters are programmable via the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2):
- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- EMIF.$\overline{CEx}$ assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, EMIF.$\overline{CEx}$ goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, EMIF.CEx is active when EMIF.$\overline{SOE}$ is active (CEEXT = 1).
- Function of EMIF.$\overline{SADS}$/$\overline{SRE}$ (RENEN): For standard SBSRAM or ZBT SRAM interface, EMIF.$\overline{SADS}$/$\overline{SRE}$ acts as EMIF.$\overline{SADS}$ with deselect cycles (RENEN = 0). For FIFO interface, EMIF.$\overline{SADS}$/$\overline{SRE}$ acts as EMIF.$\overline{SRE}$ with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2



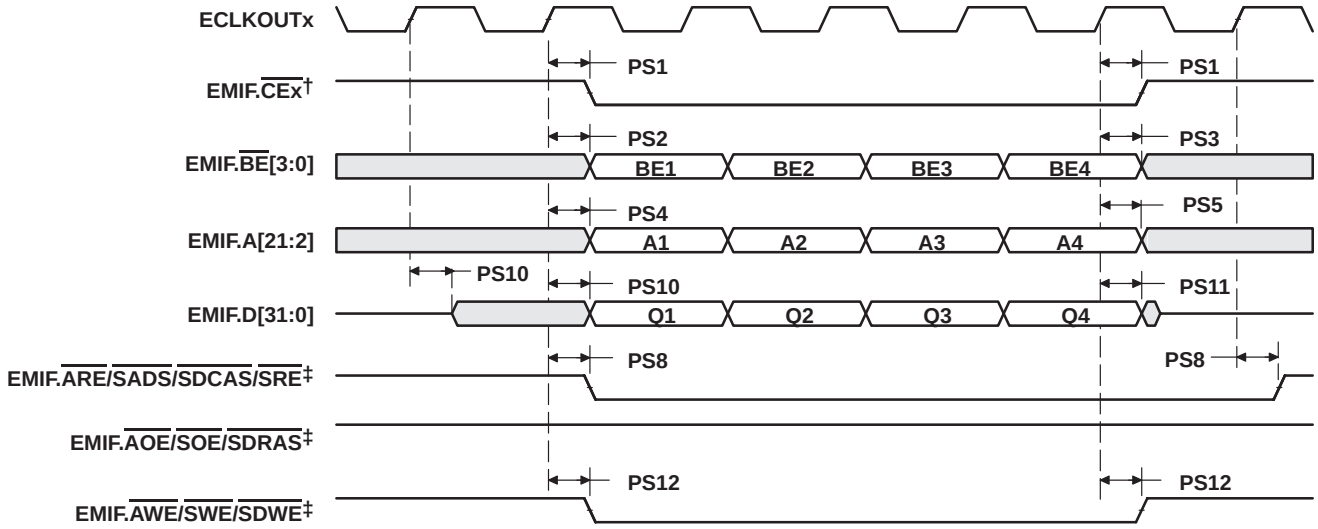

† The read latency and the length of EMIF.CEx assertion are programmable via the SYNCRL and CEEXT fields, respectively, in the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2). In the figure, SYNCRL = 2 and CEEXT = 0.

‡ The following parameters are programmable via the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2):

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- EMIF.CEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, EMIF.CEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, EMIF.CEx is active when EMIF.SOE is active (CEEXT = 1).
- Function of EMIF.SADS/SRE (RENEN): For standard SBSRAM or ZBT SRAM interface, EMIF.SADS/SRE acts as EMIF.SADS with deselect cycles (RENEN = 0). For FIFO interface, EMIF.SADS/SRE acts as EMIF.SRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

§ EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, and EMIF.AWE/SWE/SDWE operate as EMIF.SADS/SRE, EMIF.SOE, and EMIF.SWE, respectively, during programmable synchronous interface accesses.

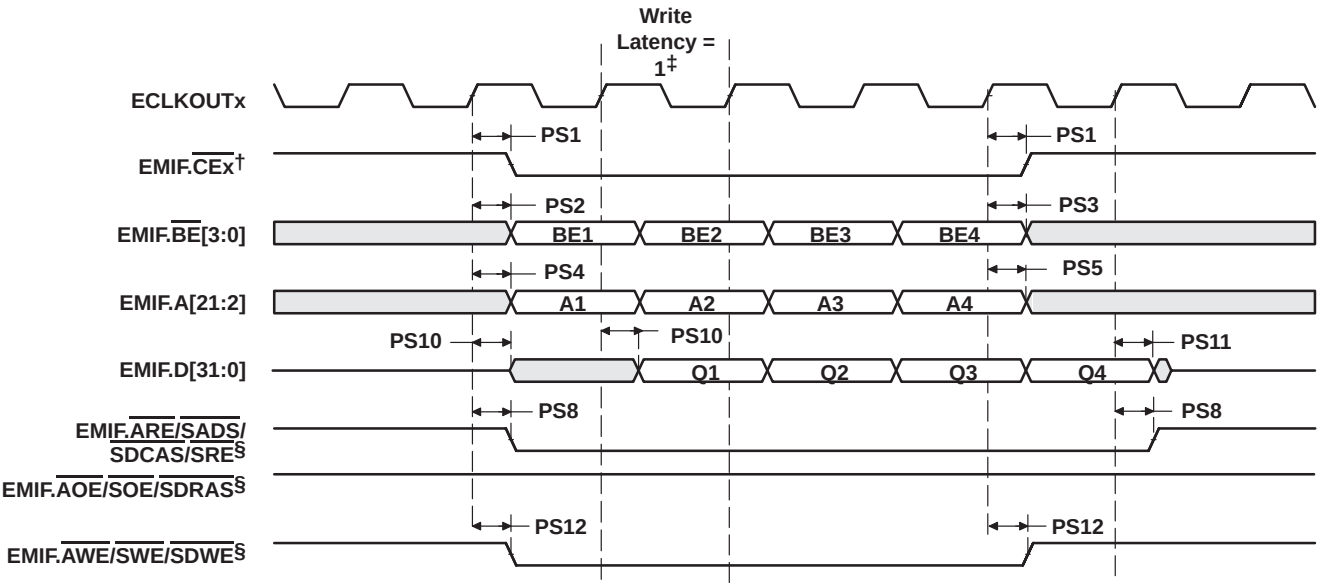
**Figure 5−10. Programmable Synchronous Interface Read Timings (With Read Latency = 2)**




† The write latency and the length of EMIF.CEx assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2). In this figure, SYNCWL = 0 and CEEXT = 0.

‡ EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, and EMIF.AWE/SWE/SDWE operate as EMIF.SADS/SRE, EMIF.SOE, and EMIF.SWE, respectively, during programmable synchronous interface accesses.

**Figure 5−11. Programmable Synchronous Interface Write Timings (With Write Latency = 0)**




† The write latency and the length of EMIF.CEx assertion are programmable via the SYNCWL and CEEXT fields, respectively, in the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2). In this figure, SYNCWL = 1 and CEEXT = 0.

‡ The following parameters are programmable via the EMIF CE Secondary Control Registers (CEx_SC1, CEx_SC2):

- Read latency (SYNCRL): 0-, 1-, 2-, or 3-cycle read latency
- Write latency (SYNCWL): 0-, 1-, 2-, or 3-cycle write latency
- EMIF.CEx assertion length (CEEXT): For standard SBSRAM or ZBT SRAM interface, EMIF.CEx goes inactive after the final command has been issued (CEEXT = 0). For synchronous FIFO interface with glue, EMIF.CEx is active when EMIF.SOE is active (CEEXT = 1).
- Function of EMIF.SADS/SRE (RENEN): For standard SBSRAM or ZBT SRAM interface, EMIF.SADS/SRE acts as EMIF.SADS with deselect cycles (RENEN = 0). For FIFO interface, EMIF.SADS/SRE acts as EMIF.SRE with NO deselect cycles (RENEN = 1).
- Synchronization clock (SNCCLK): Synchronized to ECLKOUT1 or ECLKOUT2

§ EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, and EMIF.AWE/SWE/SDWE operate as EMIF.SADS/SRE, EMIF.SOE, and EMIF.SWE, respectively, during programmable synchronous interface accesses.

**Figure 5−12. Programmable Synchronous Interface Write Timings (With Write Latency = 1)**

### 5.7.3 Synchronous DRAM Timings

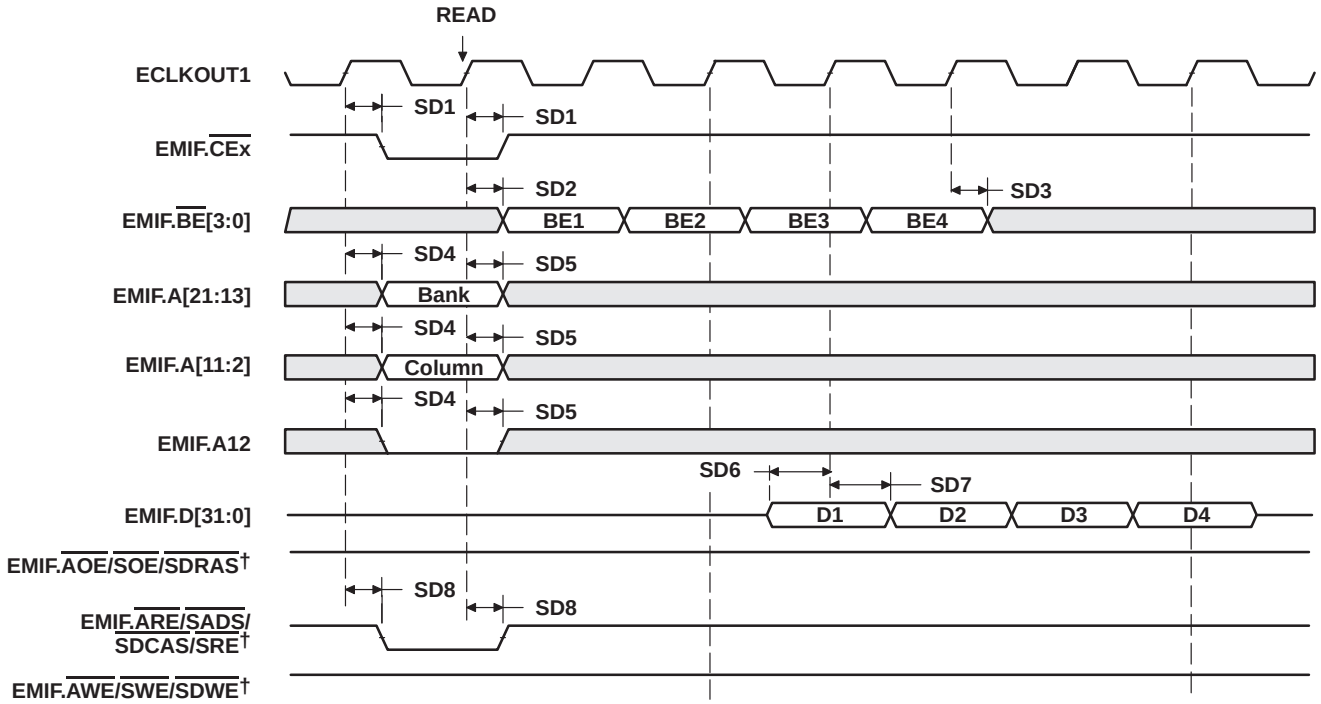
Table 5−13 and Table 5−14 assume testing over recommended operating conditions (see Figure 5−13 through Figure 5−20).

**Table 5−13. Synchronous DRAM Cycle Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| SD6 | $t_{su(EDV-EKO1H)}$ | Setup time, read EMIF.Dx valid before ECLKOUT1 high | 2 | | ns |
| SD7 | $t_{h(EKO1H-EDV)}$ | Hold time, read EMIF.Dx valid after ECLKOUT1 high | 2 | | ns |

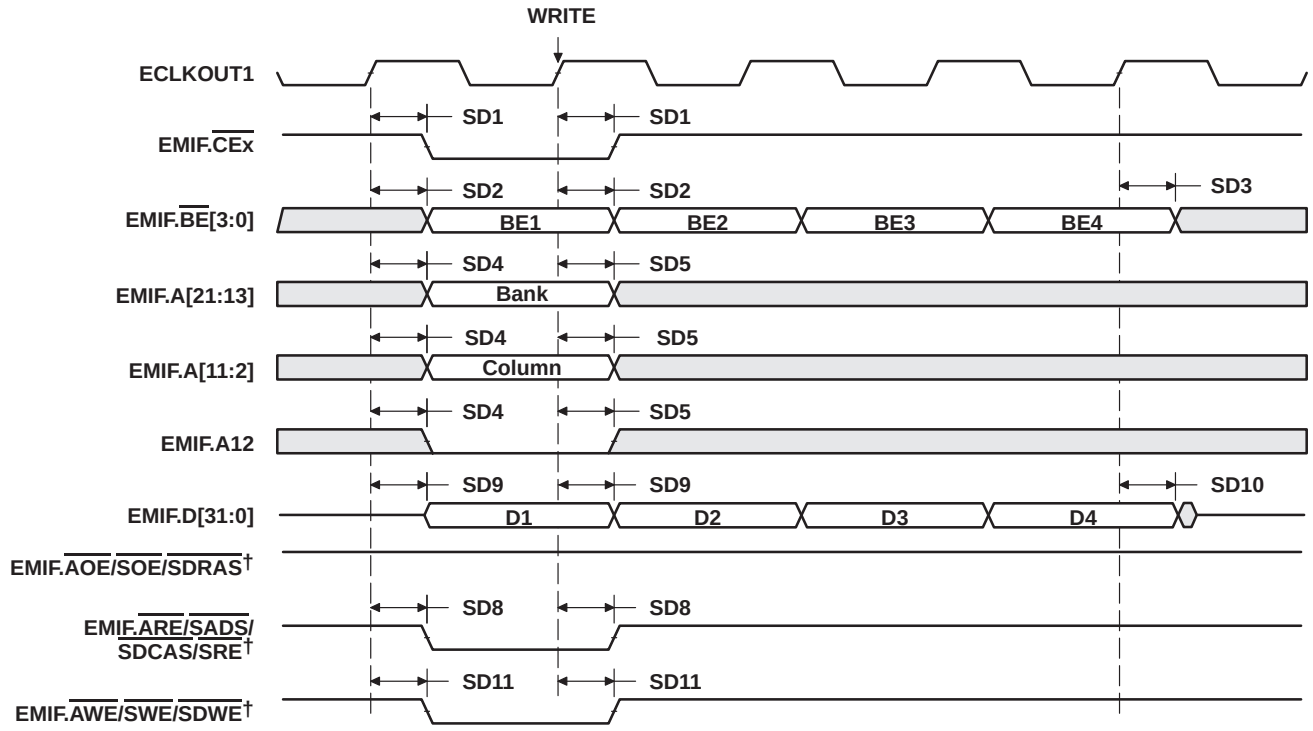
**Table 5−14. Synchronous DRAM Cycle Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| SD1 | $t_{d(EKO1H-CEV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{CEx}$ valid/invalid | 0.8 | 7 | ns |
| SD2 | $t_{d(EKO1H-BEV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{BEx}$ valid | | 7 | ns |
| SD3 | $t_{d(EKO1H-BEIV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{BEx}$ invalid | 0.8 | | ns |
| SD4 | $t_{d(EKO1H-EAV)}$ | Delay time, ECLKOUT1 high to EMIF.Ax valid | | 7 | ns |
| SD5 | $t_{d(EKO1H-EAIV)}$ | Delay time, ECLKOUT1 high to EMIF.Ax invalid | 0.8 | | ns |
| SD8 | $t_{d(EKO1H-CASV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{SDCAS}$ valid | 0.8 | 7 | ns |
| SD9 | $t_{d(EKO1H-EDV)}$ | Delay time, ECLKOUT1 high to EMIF.Dx valid | | 7 | ns |
| SD10 | $t_{d(EKO1H-EDIV)}$ | Delay time, ECLKOUT1 high to EMIF.Dx invalid | 0.8 | | ns |
| SD11 | $t_{d(EKO1H-WEV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{SDWE}$ valid | 0.8 | 7 | ns |
| SD12 | $t_{d(EKO1H-RASV)}$ | Delay time, ECLKOUT1 high to EMIF.$\overline{SDRAS}$ valid | 0.8 | 7 | ns |
| SD13 | $t_{d(EKO1H-CKEV)}$ | Delay time, ECLKOUT1 high to EMIF.SDCKE valid | 0.8 | 7 | ns |



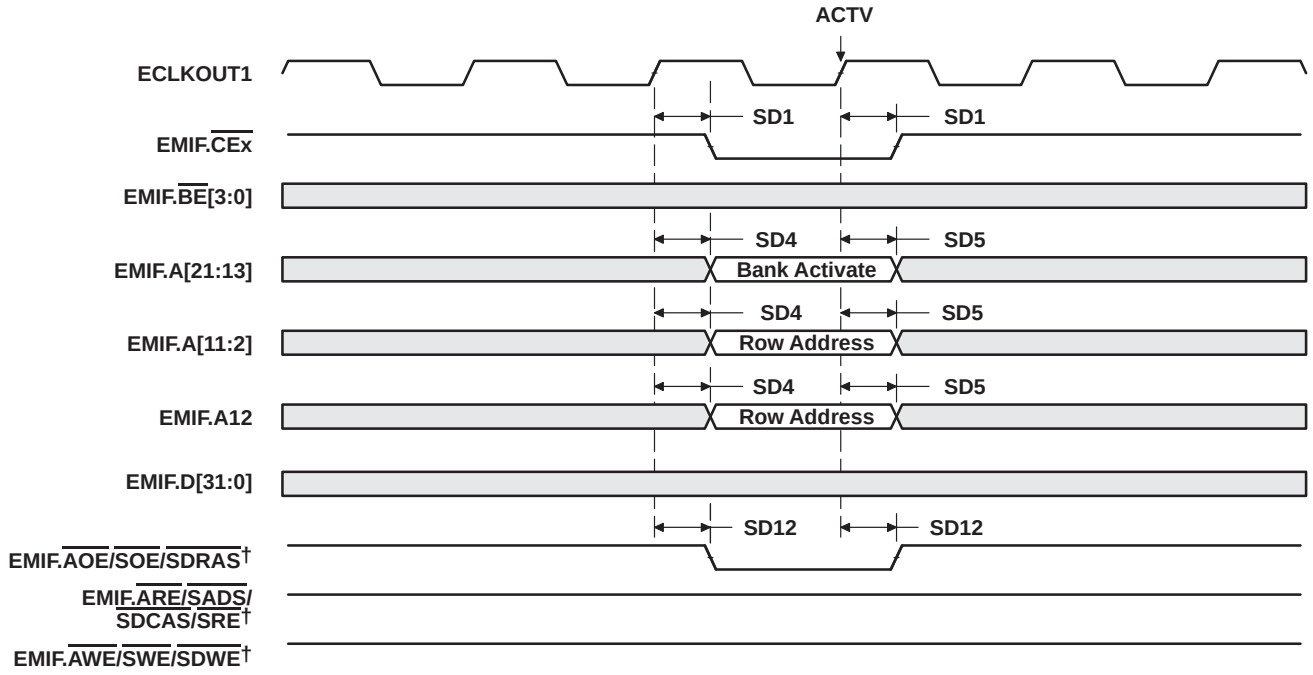

† EMIF.$\overline{ARE}$/$\overline{SADS}$/$\overline{SDCAS}$/$\overline{SRE}$, EMIF.$\overline{AWE}$/$\overline{SWE}$/$\overline{SDWE}$, and EMIF.$\overline{AOE}$/$\overline{SOE}$/$\overline{SDRAS}$ operate as EMIF.$\overline{SDCAS}$, EMIF.$\overline{SDWE}$, and EMIF.$\overline{SDRAS}$, respectively,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

**Figure 5−13. SDRAM Read Command (CAS Latency 3)**



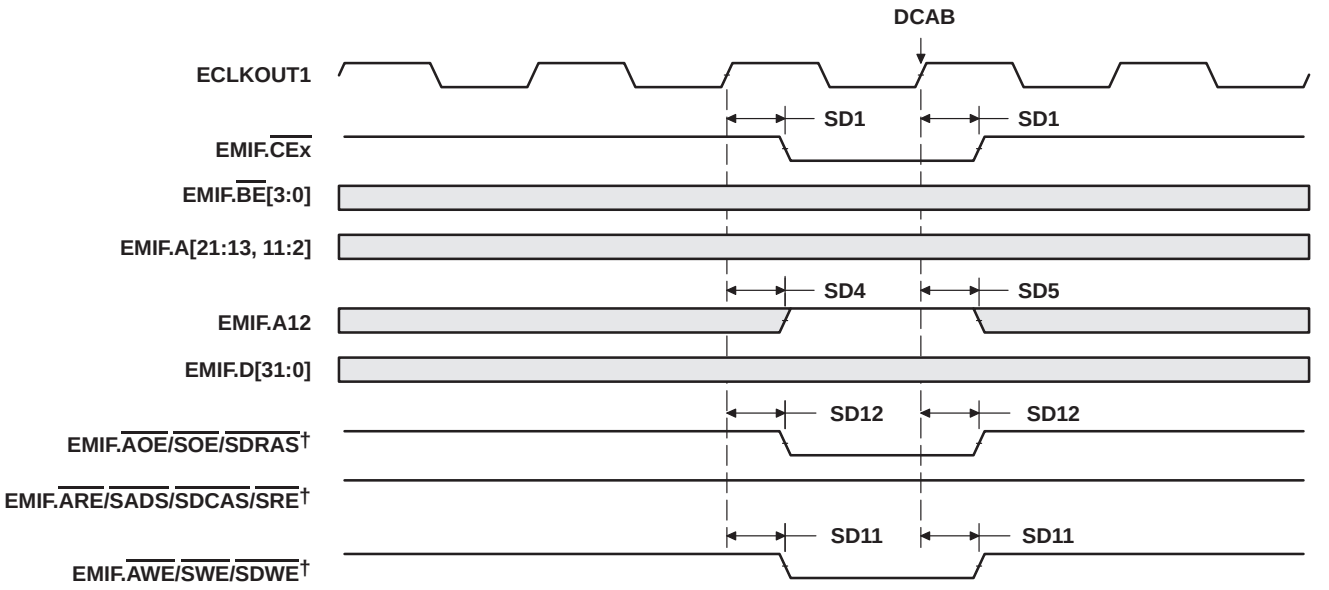

† EMIF.ARE/SADS/SDCAS/SRE, EMIF.AWE/SWE/SDWE, and EMIF.AOE/SOE/SDRAS operate as EMIF.SDCAS, EMIF.SDWE, and EMIF.SDRAS, respectively, during SDRAM accesses.

**Figure 5−14. SDRAM Write Command**



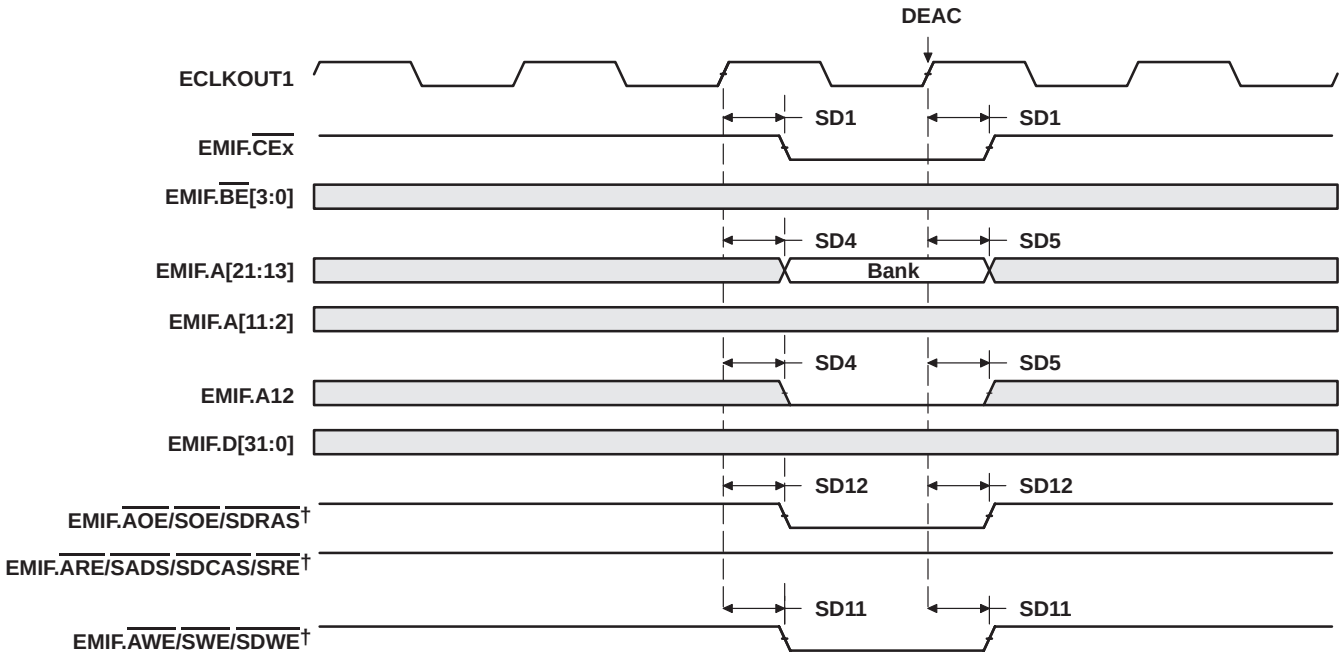

† EMIF.ARE/SADS/SDCAS/SRE, EMIF.AWE/SWE/SDWE, and EMIF.AOE/SOE/SDRAS operate as EMIF.SDCAS, EMIF.SDWE, and EMIF.SDRAS, respectively, during SDRAM accesses.

**Figure 5−15. SDRAM ACTV Command**



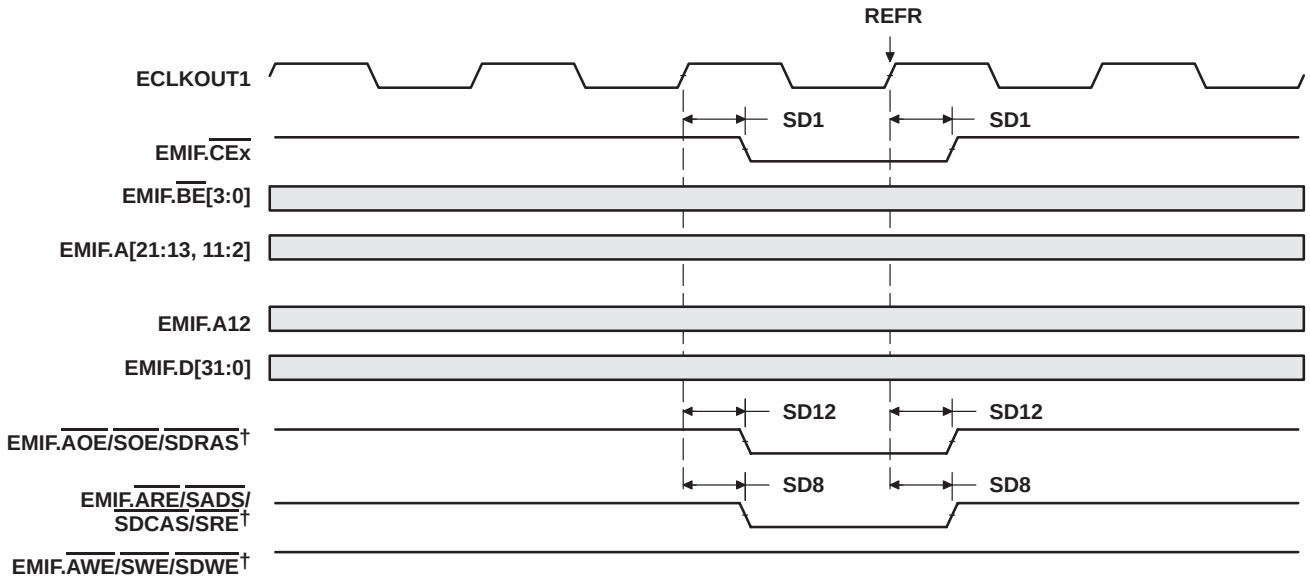

† EMIF.ARE/SADS/SDCAS/SRE, EMIF.AWE/SWE/SDWE, and EMIF.AOE/SOE/SDRAS operate as EMIF.SDCAS, EMIF.SDWE, and EMIF.SDRAS, respectively, during SDRAM accesses.

**Figure 5−16. SDRAM DCAB Command**



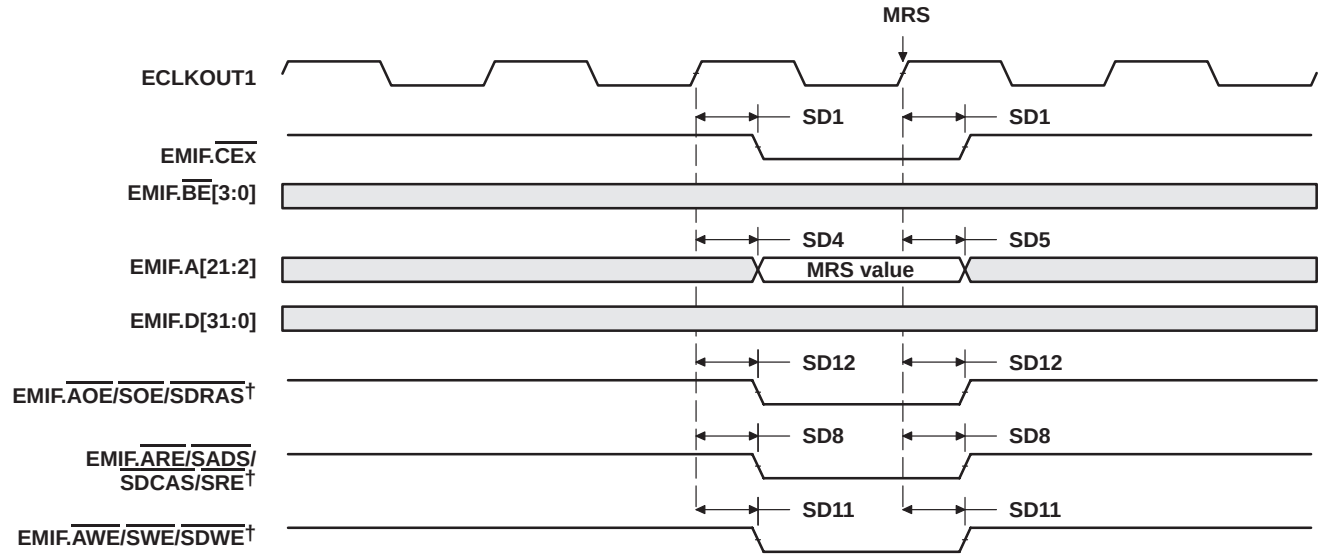

† EMIF.ARE/SADS/SDCAS/SRE, EMIF.AWE/SWE/SDWE, and EMIF.AOE/SOE/SDRAS operate as EMIF.SDCAS, EMIF.SDWE, and EMIF.SDRAS, respectively, during SDRAM accesses.

**Figure 5−17. SDRAM DEAC Command**



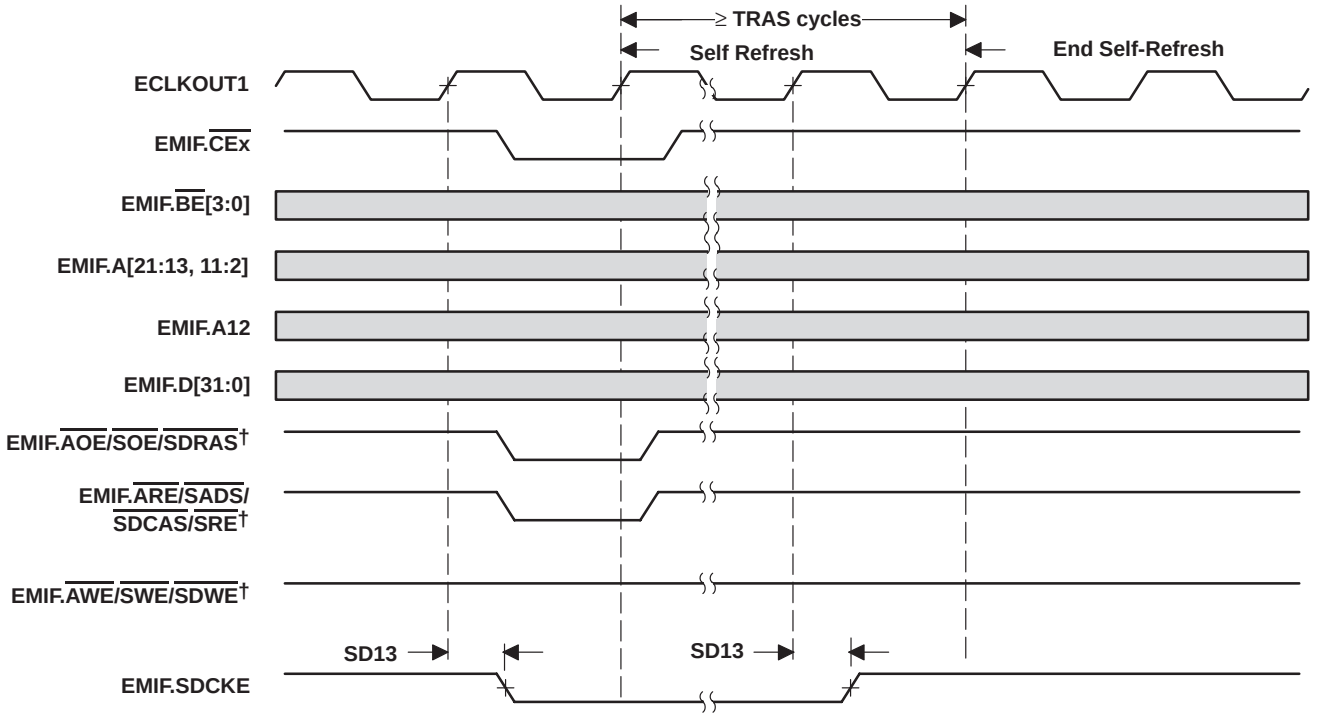

† EMIF.ARE/SADS/SDCAS/SRE, EMIF.AWE/SWE/SDWE, and EMIF.AOE/SOE/SDRAS operate as EMIF.SDCAS, EMIF.SDWE, and EMIF.SDRAS, respectively, during SDRAM accesses.

**Figure 5−18. SDRAM REFR Command**




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

**Figure 5−19. SDRAM MRS Command**




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

**Figure 5−20. SDRAM Self-Refresh Timings**

## 5.8 HOLD/HOLDA Timings

Table 5−15 and Table 5−16 assume testing over recommended operating conditions (see Figure 5−21).

**Table 5−15. EMIF.HOLD/HOLDA Timing Requirements†**

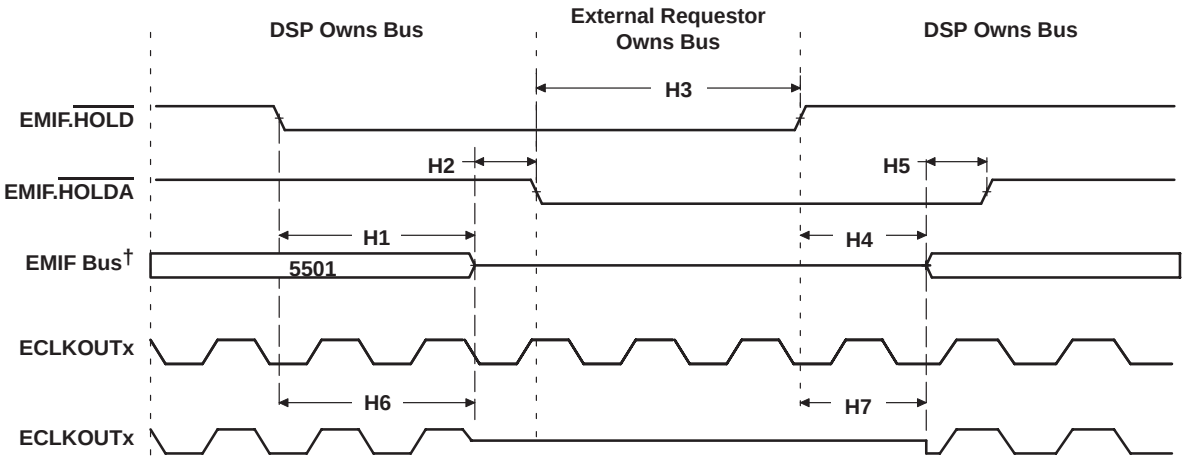
| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| H3 | $t_{oh(HOLDAL-HOLDL)}$ | Hold time, EMIF.HOLD low after EMIF.HOLDA low | E | | ns |

† E = the EMIF input clock (ECLKIN, CPU/1 clock, CPU1/2 clock, or CPU1/4 clock) period in ns for EMIF.

**Table 5−16. EMIF.HOLD/HOLDA Switching Characteristics†‡§**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| H1 | $t_{d(HOLDL-EMHZ)}$ | Delay time, EMIF.HOLD low to EMIF Bus high impedance | 4E | ¶ | ns |
| H2 | $t_{d(EMHZ-HOLDAL)}$ | Delay time, EMIF Bus high impedance to EMIF.HOLDA low | 0 | 2E | ns |
| H4 | $t_{d(HOLDH-EMLZ)}$ | Delay time, EMIF.HOLD high to EMIF Bus low impedance | 2E | 7E | ns |
| H5 | $t_{d(EMLZ-HOLDAH)}$ | Delay time, EMIF Bus low impedance to EMIF.HOLDA high | 0 | 2E | ns |
| H6 | $t_{d(HOLDL-EKOHZ)}$ | Delay time, EMIF.HOLD low to ECLKOUTx high impedance | 4E | ¶ | ns |
| H7 | $t_{d(HOLDH-EKOLZ)}$ | Delay time, EMIF.HOLD high to ECLKOUTx low impedance | 2E | 7E | ns |

† E = the EMIF input clock (ECLKIN, CPU/1 clock, CPU1/2 clock, or CPU1/4 clock) period in ns for EMIF.
‡ EMIF Bus consists of: EMIF.CE[3:0], EMIF.BE[3:0], EMIF.D[31:0], EMIF.A[21:2], EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, and EMIF.AWE/SWE/SDWE, EMIF.SDCKE, and EMIF.SOE3.
§ The EKxHZ bits in the EMIF Global Control Registers (EGCR1, EGCR2) determine the state of the ECLKOUTx signals during EMIF.HOLDA. If EKxHZ = 0, ECLKOUTx continues clocking during HOLD mode. If EKxHZ = 1, ECLKOUTx goes to high impedance during HOLD mode, as shown in Figure 5−21.
¶ All pending EMIF transactions are allowed to complete before EMIF.HOLDA is asserted. If no bus transactions are occurring, then the minimum delay time can be achieved. Also, bus hold can be indefinitely delayed by setting NOHOLD = 1.




† EMIF Bus consists of: EMIF.CE[3:0], EMIF.BE[3:0], EMIF.D[31:0], EMIF.A[21:2], EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, and EMIF.AWE/SWE/SDWE, EMIF.SDCKE, and EMIF.SOE3.

**Figure 5−21. EMIF.HOLD/HOLDA Timings**

## 5.9 Reset Timings

Table 5−17 and Table 5−18 assume testing over recommended operating conditions (see Figure 5−22).

**Table 5−17. Reset Timing Requirements †**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| R1 | $t_{w(RSL)}$ | Pulse width, $\overline{\text{RESET}}$ low | 2P + 5 | | ns |

† P = the period of the clock on the X2/CLKIN pin in ns. For example, when using 20 MHz as the input clock, use P = 50 ns.

**Table 5−18. Reset Switching Characteristics †**

| NO. | PARAMETER | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|
| R2 | $t_{d(RSL-EMIFHZ)}$ | Delay time, $\overline{\text{RESET}}$ low to EMIF group high impedance‡ | | | 12 | ns |
| R3 | $t_{d(RSH-EMIFV)}$ | Delay time, $\overline{\text{RESET}}$ high to EMIF group valid‡ | GPIO4 = 0 (CLKMOD = 0) | | 41115P + 21 | ns |
| | | | GPIO4 = 1 (CLKMOD = 1) | | 148P + 22 | |
| R4 | $t_{d(RSL-HIGHIV)}$ | Delay time, $\overline{\text{RESET}}$ low to high group invalid§ | | | 12 | ns |
| R5 | $t_{d(RSH-HIGHV)}$ | Delay time, $\overline{\text{RESET}}$ high to high group valid§ | GPIO4 = 0 (CLKMOD = 0) | | 41044P + 17 | ns |
| | | | GPIO4 = 1 (CLKMOD = 1) | | 77P + 18 | |
| R6 | $t_{d(RSL-ZHZ)}$ | Delay time, $\overline{\text{RESET}}$ low to Z group high impedance¶ | | | 10 | ns |
| R7 | $t_{d(RSH-ZV)}$ | Delay time, $\overline{\text{RESET}}$ high to Z group invalid¶ | GPIO4 = 0 (CLKMOD = 0) | | 41044P + 18 | ns |
| | | | GPIO4 = 1 (CLKMOD = 1) | | 77P + 19 | |
| R8 | $t_{d(RSL-IOIM)}$ | Delay time, $\overline{\text{RESET}}$ low to Input/Output group switch to input mode# | | | 13 | ns |
| R9 | $t_{d(RSL-TGLD)}$ | Delay time, $\overline{\text{RESET}}$ low to Toggle group switch to default toggle frequency‖ | | | 11 + 14P | ns |

† P = the period of the clock on the X2/CLKIN pin in ns. For example, when using 20 MHz as the input clock, use P = 50 ns.

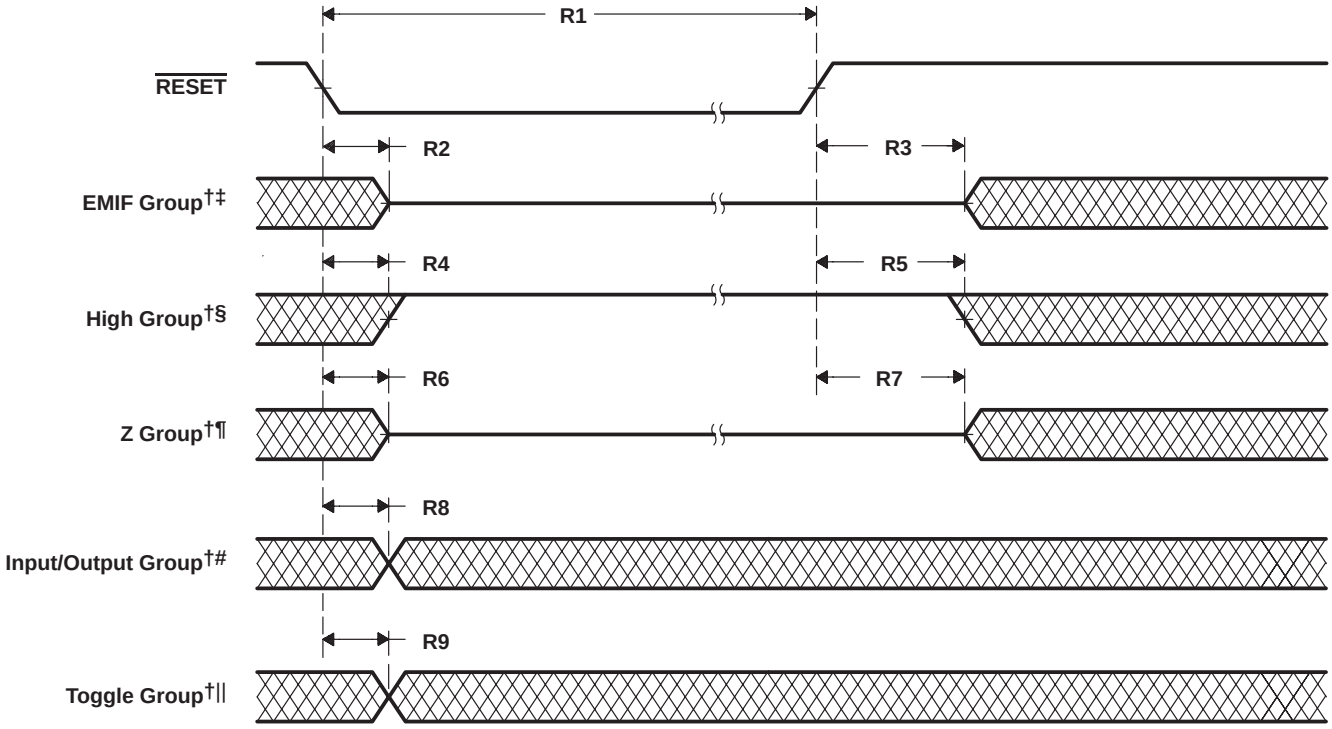
‡ EMIF group: EMIF.A[21:2], EMIF.$\overline{\text{ARE}}$/$\overline{\text{SADS}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SRE}}$, EMIF.$\overline{\text{AOE}}$/$\overline{\text{SOE}}$/$\overline{\text{SDRAS}}$, EMIF.$\overline{\text{AWE}}$/$\overline{\text{SWE}}$/$\overline{\text{SDWE}}$, EMIF.$\overline{\text{ARDY}}$, EMIF.$\overline{\text{CE0}}$, EMIF.$\overline{\text{CE1}}$, EMIF.$\overline{\text{CE2}}$, EMIF.$\overline{\text{CE3}}$, EMIF.$\overline{\text{BE0}}$, EMIF.$\overline{\text{BE1}}$, EMIF.$\overline{\text{BE2}}$, EMIF.$\overline{\text{BE3}}$, EMIF.SDCKE, EMIF.$\overline{\text{SOE3}}$, EMIF.$\overline{\text{HOLD}}$, EMIF.$\overline{\text{HOLDA}}$, ECLKOUT1.
EMIF.ARDY and EMIF.$\overline{\text{HOLDA}}$ do not go to a high-impedance state during reset since they are input-only signals; they are included here simply for completeness.

§ High group: $\overline{\text{IACK}}$, $\overline{\text{XF}}$, SCL (assumes external pullup on pin), SDA (assumes external pullup on pin), UART.TX, TDO.

¶ Z group: HRDY, $\overline{\text{HINT}}$, DX1, DX0

# Input/Output group: PGPIO[45:0], HPI.HD[7:0], EMIF.D[31:0], HPI.$\overline{\text{HAS}}$, HPI.HBIL, HCNTL1, HCNTL0, $\overline{\text{HCS}}$, HR/$\overline{\text{W}}$, $\overline{\text{HDS1}}$, $\overline{\text{HDS2}}$, NMI/WDTOUT, GPIO[7:0], TIM0, TIM1, CLKR0, CLKX0, FSR0, FSX0, CLKR1, CLKX1, FSR1, FSX1, EMU0, EMU1/$\overline{\text{OFF}}$. Signals in this group switch to input mode with reset.

‖ Toggle group: ECLKOUT2, CLKOUT. Pins in this group toggle with a default frequency during reset.




† The state of the DSP pins during power up is undefined until RESET is asserted. It is recommended that the RESET pin be kept low during power up.

‡ EMIF group: EMIF.A[21:2], EMIF.ARE/SADS/SDCAS/SRE, EMIF.AOE/SOE/SDRAS, EMIF.AWE/SWE/SDWE, EMIF.ARDY, EMIF.CE0, EMIF.CE1, EMIF.CE2, EMIF.CE3, EMIF.BE0, EMIF.BE1, EMIF.BE2, EMIF.BE3, EMIF.SDCKE, EMIF.SOE3, EMIF.HOLD, EMIF.HOLDA, ECLKOUT1.
EMIF.ARDY and EMIF.HOLDA do not go to a high-impedance state during reset since they are input-only signals; they are included here simply for completeness.

§ High group: IACK, XF, SCL (assumes external pullup on pin), SDA (assumes external pullup on pin), UART.TX, TDO.

¶ Z group: HRDY, HINT, DX1, DX0

# Input/Output group: PGPIO[45:0], HPI.HD[7:0], EMIF.D[31:0], HPI.HAS, HPI.HBIL, HCNTL1, HCNTL0, HCS, HR/W, HDS1, HDS2, NMI/WDTOUT, GPIO[7:0], TIM0, TIM1, CLKR0, CLKX0, FSR0, FSX0, CLKR1, CLKX1, FSR1, FSX1, EMU0, EMU1/OFF. Signals in this group switch to input mode with reset.

|| Toggle group: ECLKOUT2, CLKOUT. Pins in this group toggle with a default frequency during reset.

**Figure 5−22. Reset Timings**

## 5.10 External Interrupt and Interrupt Acknowledge ($\overline{\text{IACK}}$) Timings

Table 5−19 and Table 5−20 assume testing over recommended operating conditions (see Figure 5−23 and Figure 5−24).

**Table 5−19. External Interrupt and Interrupt Acknowledge Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| I1 | $t_{w(INTL)A}$ | Pulse width, interrupt low, CPU active | 3P† | | ns |
| I2 | $t_{w(INTH)A}$ | Pulse width, interrupt high, CPU active | 1P† | | ns |

† P = 1/CPU clock frequency in ns. For example, when running parts at 300 MHz, use P = 3.33 ns.

**Table 5−20. External Interrupt and Interrupt Acknowledge Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| I3 | $t_{d(COH-IACKV)}$ | Delay time, CLKOUT high to $\overline{\text{IACK}}$ valid‡ | 0 | 8 | ns |

‡ In this case, CLKOUT refers to the CPU clock. Since CLKOUT cannot be programmed to reflect the CPU clock, there might be an extra delay of a certain number of CPU clocks based on the ratio between the system clock shown on CLKOUT and the CPU clock. For example, if SYSCLK2 is shown on CLKOUT and SYSCLK2 is programmed to be half the CPU clock, there might be an extra delay of one CPU clock period between the transition of CLKOUT and the specified timing. If system clock is programmed to be one-fourth of the CPU clock, there might be an extra delay of 1, 2, or 3 CPU clocks between the transition of CLKOUT and the specified timing. The extra delay must be taken into account when considering the MAX value for the timing under question. Note that if the CPU clock and the system clock shown on CLKOUT are operating at the same frequency, there will be no extra delay in the specified timing.

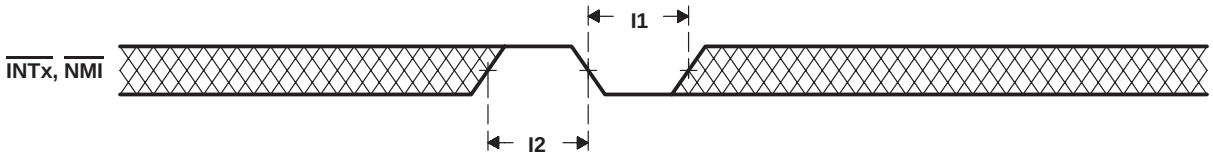
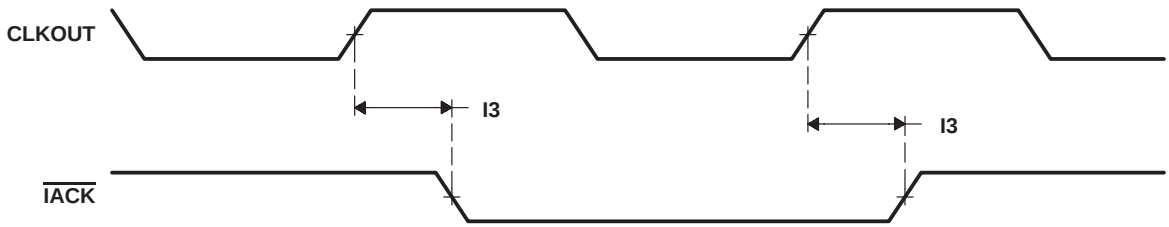



**Figure 5−23. External Interrupt Timings**




NOTE: The figure shows the case in which CLKOUT is programmed to show a system clock that is operating at the same frequency as the CPU clock.

**Figure 5−24. External Interrupt Acknowledge Timings**

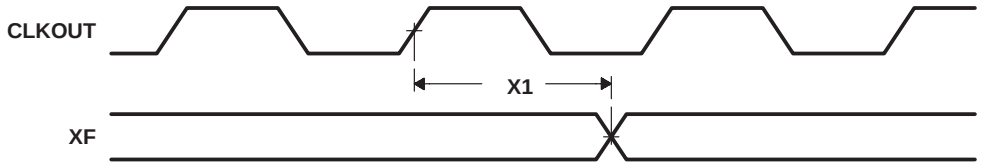
## 5.11 XF Timings

Table 5−21 assumes testing over recommended operating conditions (see Figure 5−25).

**Table 5−21. XF Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| X1 | $t_{d(XF)}$ | Delay time, CLKOUT high to XF high† | 0 | 5 | ns |
| | | Delay time, CLKOUT high to XF low† | 0 | 6 | |

† In this case, CLKOUT refers to the CPU clock. Since CLKOUT cannot be programmed to reflect the CPU clock, there might be an extra delay of a certain number of CPU clocks based on the ratio between the system clock shown on CLKOUT and the CPU clock. For example, if SYSCLK2 is shown on CLKOUT and SYSCLK2 is programmed to be half the CPU clock, there might be an extra delay of one CPU clock period between the transition of CLKOUT and the specified timing. If system clock is programmed to be one-fourth of the CPU clock, there might be an extra delay of 1, 2, or 3 CPU clocks between the transition of CLKOUT and the specified timing. The extra delay must be taken into account when considering the MAX value for the timing under question. Note that if the CPU clock and the system clock shown on CLKOUT are operating at the same frequency, there will be no extra delay in the specified timing.




NOTE: The figure shows the case in which CLKOUT is programmed to show a system clock that is operating at the same frequency as the CPU clock.

**Figure 5−25. XF Timings**

## 5.12 General-Purpose Input/Output (GPIOx) Timings

Table 5−22 and Table 5−23 assume testing over recommended operating conditions (see Figure 5−26).

**Table 5−22. GPIO Pins Configured as Inputs Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| G2 | $t_{su(GPIO–COH)}$ | Setup time, GPIOx input valid before CLKOUT high† | 5 | | ns |
| G3 | $t_{h(COH–GPIO)}$ | Hold time, GPIOx input valid after CLKOUT high† | 0 | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

**Table 5−23. GPIO Pins Configured as Outputs Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| G1 | $t_{d(COH–GPIO)}$ | Delay time, CLKOUT high to GPIOx output change† | 0 | 8 | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

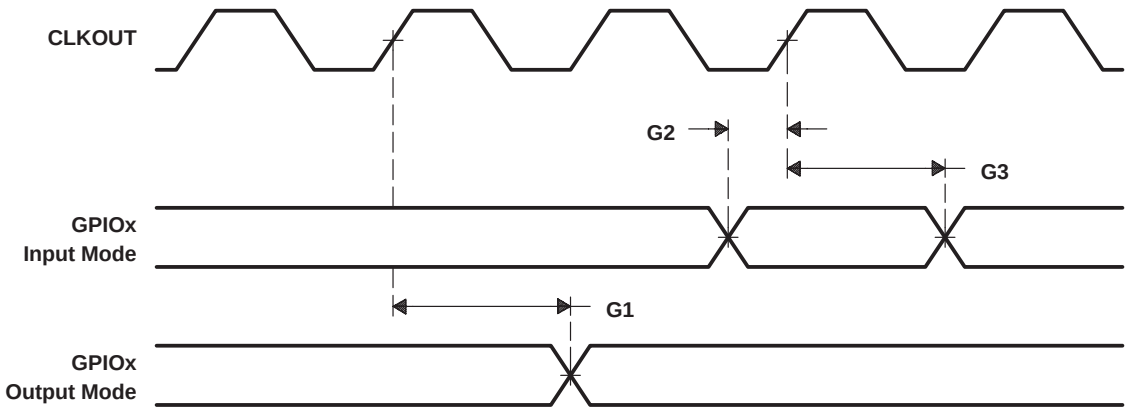



**Figure 5−26. General-Purpose Input/Output (GPIOx) Signal Timings**

## 5.13 Parallel General-Purpose Input/Output (PGPIOx) Timings

Table 5−24 and Table 5−25 assume testing over recommended operating conditions (see Figure 5−27).

**Table 5−24. PGPIO Pins Configured as Inputs Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| PG2 | $t_{su(PGPIO–COH)}$ | Setup time, PGPIOx input valid before CLKOUT high† | 6 | | ns |
| PG3 | $t_{h(COH–PGPIO)}$ | Hold time, PGPIOx input valid after CLKOUT high† | 0 | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

**Table 5−25. PGPIO Pins Configured as Outputs Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| PG1 | $t_{d(COH–PGPIO)}$ | Delay time, CLKOUT high to PGPIOx output change† | 0 | 10 | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

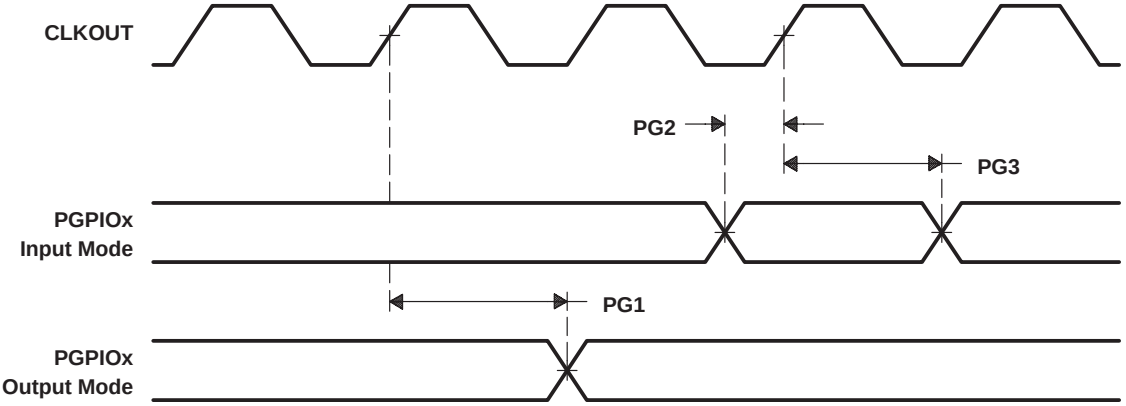



**Figure 5−27. Parallel General-Purpose Input/Output (PGPIOx) Signal Timings**

## 5.14 TIM0/TIM1/WDTOUT Timings

### *5.14.1 TIM0/TIM1/WDTOUT Timer Pin Timings*

Table 5−26 and Table 5−27 assume testing over recommended operating conditions (see Figure 5−28 and Figure 5−29).

**Table 5−26. TIM0/TIM1/WDTOUT Pins Configured as Timer Input Pins Timing Requirements†**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| T4 | $t_{w(TIML)}$ | Pulse width, TIM0/TIM1/WDTOUT low | 4P | | ns |
| T5 | $t_{w(TIMH)}$ | Pulse width, TIM0/TIM1/WDTOUT high | 4P | | ns |

† P = (Divider1 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the fast peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

**Table 5−27. TIM0/TIM1/WDTOUT Pins Configured as Timer Output Pins Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| T1 | $t_{d(COH–TIMH)}$ | Delay time, CLKOUT high to TIM0/TIM1/WDTOUT high‡ | 0 | 6 | ns |
| T2 | $t_{d(COH–TIML)}$ | Delay time, CLKOUT high to TIM0/TIM1/WDTOUT low‡ | 0 | 7 | ns |
| T3 | $t_{w(TIM)}$ | Pulse duration, TIM0/TIM1/WDTOUT | P† | | ns |

† P = (Divider1 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the fast peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

‡ In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

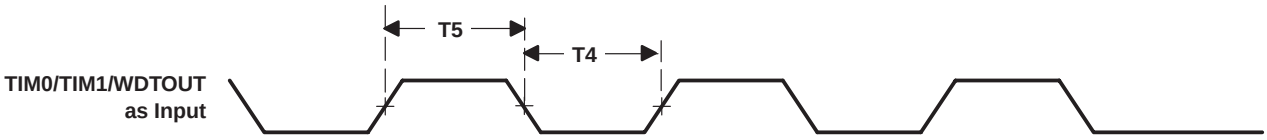



**Figure 5−28. TIM0/TIM1/WDTOUT Timings When Configured as Timer Input Pins**

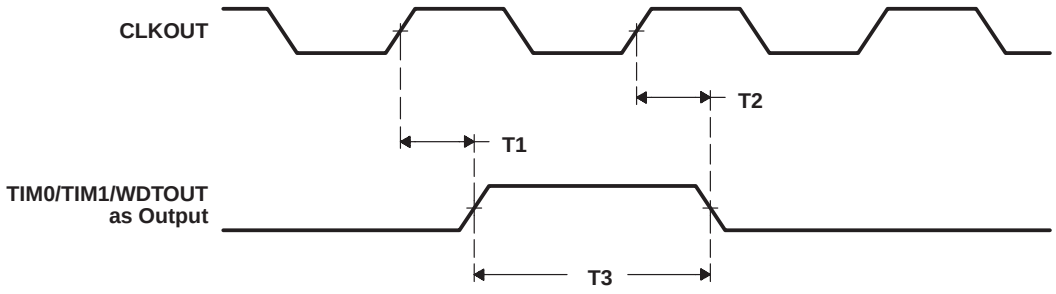



**Figure 5−29. TIM0/TIM1/WDTOUT Timings When Configured as Timer Output Pins**

### 5.14.2 TIM0/TIM1/WDTOUT General-Purpose I/O Timings

Table 5−28 and Table 5−29 assume testing over recommended operating conditions (see Figure 5−30).

**Table 5−28. TIM0/TIM1/WDTOUT General-Purpose I/O Timing Requirements†**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| T9 | $t_{su(TIM0GPIO-COH)}$ | Setup time, TIM0-GPIO input mode before CLKOUT high | 5 | | ns |
| T10 | $t_{h(COH-TIM0GPIO)}$ | Hold time, TIM0-GPIO input mode after CLKOUT high | 0 | | ns |
| T11 | $t_{su(TIM1GPIO-COH)}$ | Setup time, TIM1-GPIO input mode before CLKOUT high | 5 | | ns |
| T12 | $t_{h(COH-TIM1GPIO)}$ | Hold time, TIM1-GPIO input mode after CLKOUT high | 0 | | ns |
| T13 | $t_{su(WDTGPIO-COH)}$ | Setup time, WDTOUT-GPIO input mode before CLKOUT high | 5 | | ns |
| T14 | $t_{h(COH-WDTGPIO)}$ | Hold time, WDTOUT-GPIO input mode after CLKOUT high | 0 | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

**Table 5−29. TIM0/TIM1/WDTOUT General-Purpose I/O Switching Characteristics†**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| T6 | $t_{d(COH-TIM0GPIO)}$ | Delay time, CLKOUT high to TIM0-GPIO output mode | | 10 | ns |
| T7 | $t_{d(COH-TIM1GPIO)}$ | Delay time, CLKOUT high to TIM1-GPIO output mode | | 10 | ns |
| T8 | $t_{d(COH-WDTGPIO)}$ | Delay time, CLKOUT high to WDTOUT-GPIO output mode | | 10 | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.

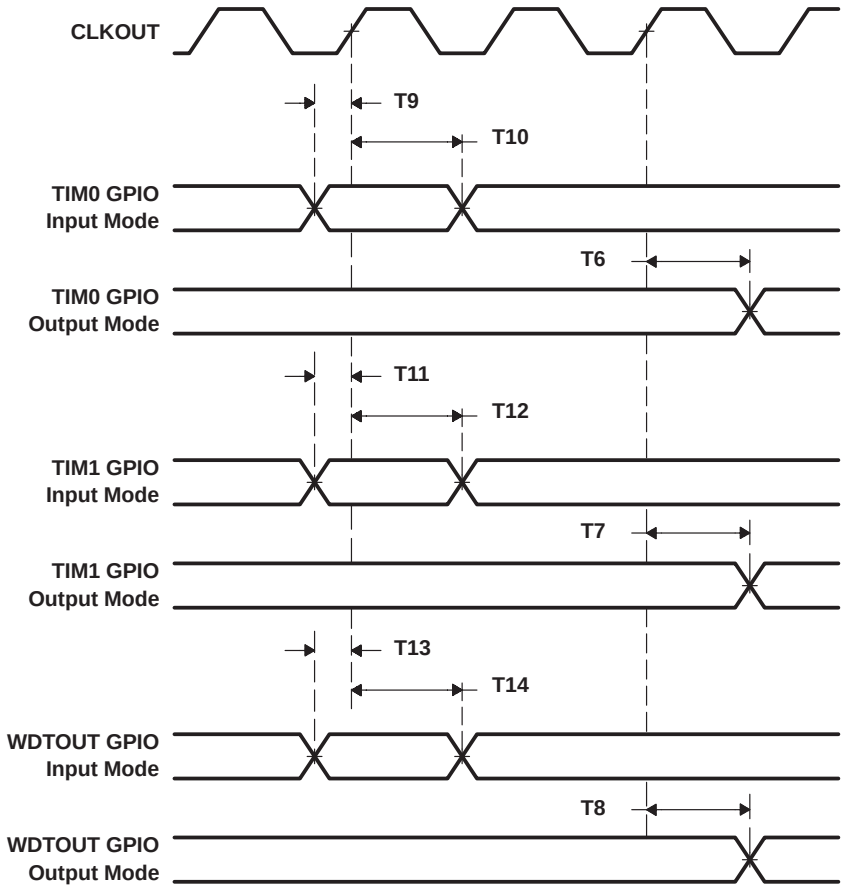



**Figure 5−30. TIM0/TIM1/WDTOUT General-Purpose I/O Timings**

### 5.14.3 TIM0/TIM1/WDTOUT Interrupt Timings

Table 5−30 assumes testing over recommended operating conditions (see Figure 5−31).

**Table 5−30. TIM0/TIM1/WDTOUT Interrupt Timing Requirements†‡**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| T15 | $t_{su(TIM0L-COH)}$ | Setup time, TIM0 low§ before CLKOUT rising edge | 5 | | ns |
| T16 | $t_{h(COH-TIM0L)}$ | Hold time, TIM0 low§ after CLKOUT rising edge | 0 | | ns |
| T17 | $t_{w(TIM0L)}$ | Pulse width, TIM0 low§ | P | | ns |
| T18 | $t_{su(TIM1L-COH)}$ | Setup time, TIM1 low§ before CLKOUT rising edge | 5 | | ns |
| T19 | $t_{h(COH-TIM1L)}$ | Hold time, TIM1 low§ after CLKOUT rising edge | 0 | | ns |
| T20 | $t_{w(TIM1L)}$ | Pulse width, TIM1 low§ | P | | ns |
| T21 | $t_{su(WDTL-COH)}$ | Setup time, WDTOUT low§ before CLKOUT rising edge | 5 | | ns |
| T22 | $t_{h(COH-WDTL)}$ | Hold time, WDTOUT low§ after CLKOUT rising edge | 0 | | ns |
| T23 | $t_{w(WDTL)}$ | Pulse width, WDTOUT low§ | P | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.
‡ P = (Divider1 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the fast peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.
§ An interrupt can be triggered by setting the timer pins high or low, depending on the setting of the TIN1INV bit in the GPIO Interrupt Control Register (GPINT). Refer to the *TMS320VC5501/5502 DSP Timers Reference Guide* (literature number SPRU618) for more information on the interrupt capability of the timer pins.

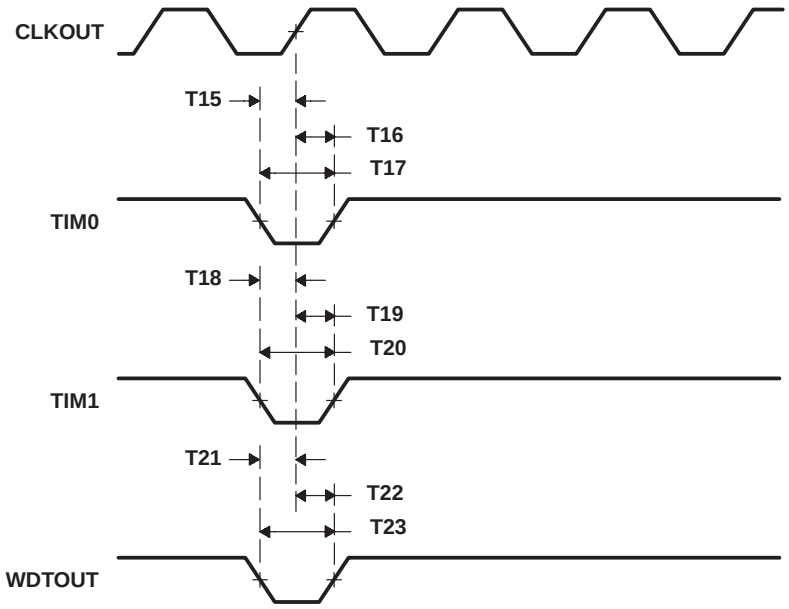



**Figure 5−31. TIM0/TIM1/WDTOUT Interrupt Timings**

## 5.15 Multichannel Buffered Serial Port (McBSP) Timings

### 5.15.1 McBSP Transmit and Receive Timings

Table 5−31 and Table 5−32 assume testing over recommended operating conditions (see Figure 5−32 and Figure 5−33).

**Table 5−31. McBSP Transmit and Receive Timing Requirements†‡**

| NO. | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|
| **M11** | $t_{c(CKRX)}$ | Cycle time, CLKR/X | CLKR/X ext | 2P | | ns |
| **M12** | $t_{w(CKRX)}$ | Pulse duration, CLKR/X high or CLKR/X low | CLKR/X ext | P – 2 | | ns |
| **M13** | $t_{r(CKRX)}$ | Rise time, CLKR/X | CLKR/X ext | | 5 | ns |
| **M14** | $t_{f(CKRX)}$ | Fall time, CLKR/X | CLKR/X ext | | 5 | ns |
| **M15** | $t_{su(FRH–CKRL)}$ | Setup time, external FSR high before CLKR low | CLKR int | 5 | | ns |
| | | | CLKR ext | 1 | | |
| **M16** | $t_{h(CKRL–FRH)}$ | Hold time, external FSR high after CLKR low | CLKR int | 1 | | ns |
| | | | CLKR ext | 6 | | |
| **M17** | $t_{su(DRV–CKRL)}$ | Setup time, DR valid before CLKR low | CLKR int | 3 | | ns |
| | | | CLKR ext | 1 | | |
| **M18** | $t_{h(CKRL–DRV)}$ | Hold time, DR valid after CLKR low | CLKR int | 1 | | ns |
| | | | CLKR ext | 6 | | |
| **M19** | $t_{su(FXH–CKXL)}$ | Setup time, external FSX high before CLKX low | CLKX int | 5 | | ns |
| | | | CLKX ext | 1 | | |
| **M20** | $t_{h(CKXL–FXH)}$ | Hold time, external FSX high after CLKX low | CLKX int | 1 | | ns |
| | | | CLKX ext | 6 | | |

† Polarity bits CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

## Table 5−32. McBSP Transmit and Receive Switching Characteristics†‡

| NO. | PARAMETER | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| M1 | $t_{c(CKRX)}$ | Cycle time, CLKR/X | | CLKR/X int | 2P | | ns |
| M2 | $t_{w(CKRXH)}$ | Pulse duration, CLKR/X high | | CLKR/X int | D−1§ | D+1§ | ns |
| M3 | $t_{w(CKRXL)}$ | Pulse duration, CLKR/X low | | CLKR/X int | C−1§ | C+1§ | ns |
| M4 | $t_{d(CKRH–FRV)}$ | Delay time, CLKR high to internal FSR valid | | CLKR int | −2 | 6 | ns |
| | | | | CLKR ext | 4 | 16 | |
| M5 | $t_{d(CKXH–FXV)}$ | Delay time, CLKX high to internal FSX valid | | CLKX int | 0 | 6 | ns |
| | | | | CLKX ext | 4 | 16 | |
| M6 | $t_{dis(CKXH–DXHZ)}$ | Disable time, CLKX high to DX high impedance following last data bit | | CLKX int | −5 | 5 | ns |
| | | | | CLKX ext | 1 | 11 | |
| M7 | $t_{d(CKXH–DXV)}$ | Delay time, CLKX high to DX valid. This applies to all bits except the first bit transmitted. | | CLKX int | | 6 | ns |
| | | | | CLKX ext | | 16 | |
| | | Delay time, CLKX high to DX valid¶ Only applies to first bit transmitted when in Data Delay 1 or 2 (XDATDLY=01b or 10b) modes | DXENA = 0 | CLKX int | | 6 | |
| | | | | CLKX ext | | 16 | |
| | | | DXENA = 1 | CLKX int | | 2P+2 | |
| | | | | CLKX ext | | 2P+8 | |
| M8 | $t_{en(CKXH–DX)}$ | Enable time, CLKX high to DX driven¶ Only applies to first bit transmitted when in Data Delay 1 or 2 (XDATDLY=01b or 10b) modes | DXENA = 0 | CLKX int | 0 | | ns |
| | | | | CLKX ext | 6 | | |
| | | | DXENA = 1 | CLKX int | 2P | | |
| | | | | CLKX ext | 2P+6 | | |
| M9 | $t_{d(FXH–DXV)}$ | Delay time, FSX high to DX valid¶ Only applies to first bit transmitted when in Data Delay 0 (XDATDLY=00b) mode. | DXENA = 0 | FSX int | | 2 | ns |
| | | | | FSX ext | | 7 | |
| | | | DXENA = 1 | FSX int | | 2P+2 | |
| | | | | FSX ext | | 2P+7 | |
| M10 | $t_{en(FXH–DX)}$ | Enable time, FSX high to DX driven¶ Only applies to first bit transmitted when in Data Delay 0 (XDATDLY=00b) mode | DXENA = 0 | FSX int | 0 | | ns |
| | | | | FSX ext | 6 | | |
| | | | DXENA = 1 | FSX int | 2P | | |
| | | | | FSX ext | P+6 | | |

† Polarity bits CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ T=CLKRX period = (1 + CLKGDV) * P
C=CLKRX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * P when CLKGDV is even
D=CLKRX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * P when CLKGDV is even

¶ See the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592) for a description of the DX enable (DXENA) and data delay features of the McBSP.

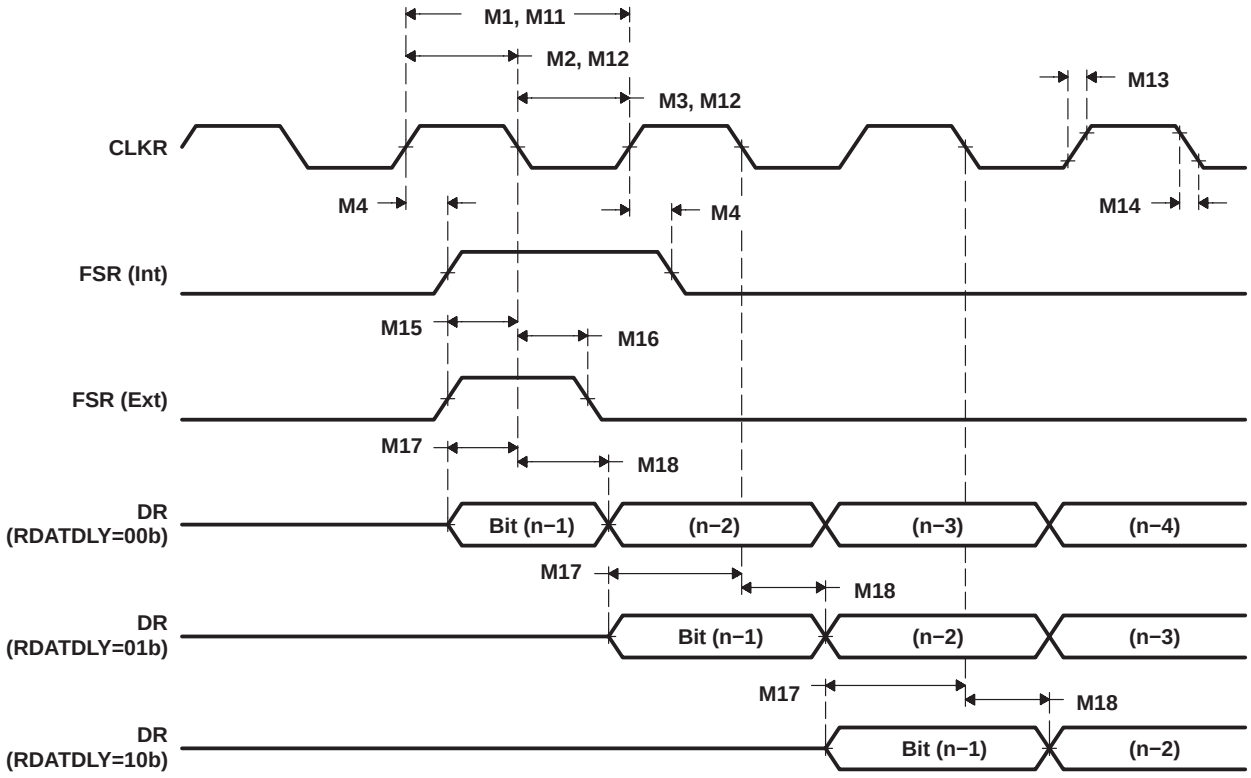
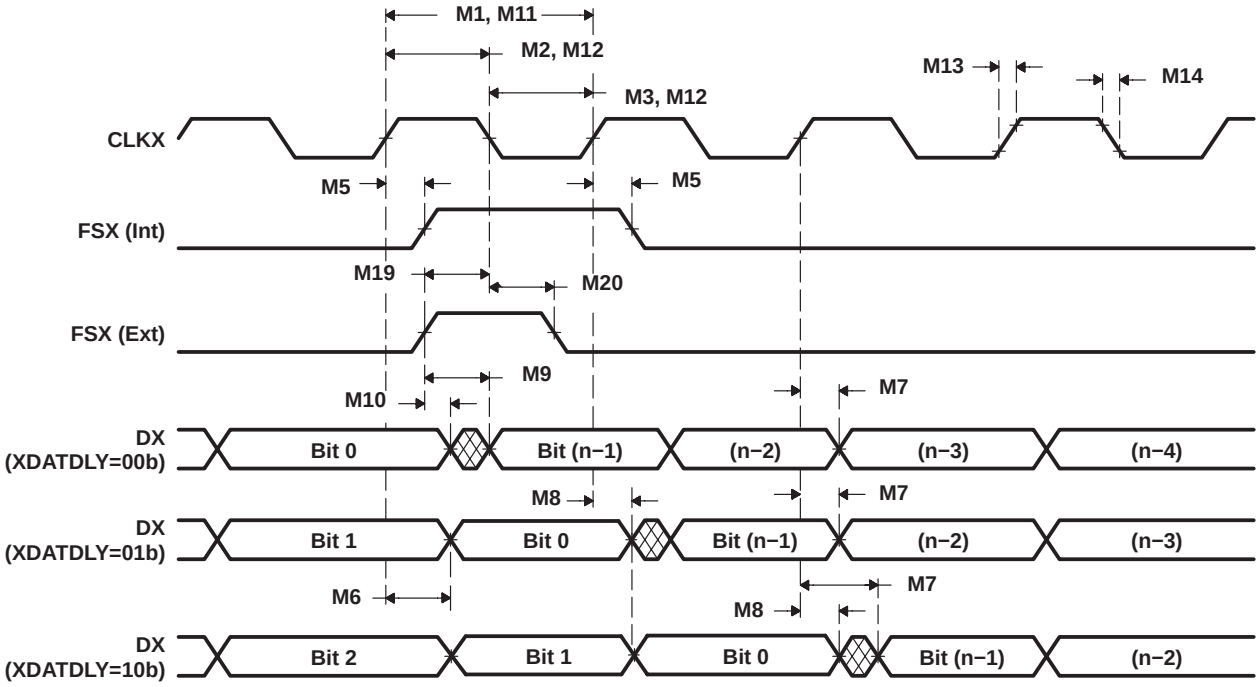



**Figure 5−32. McBSP Receive Timings**




NOTE A: This figure does not include first or last frames. For first frame, no data will be present before frame synchronization. For last frame, no data will be present after frame synchronization.

**Figure 5−33. McBSP Transmit Timings**

### 5.15.2 McBSP General-Purpose I/O Timings

Table 5−33 and Table 5−34 assume testing over recommended operating conditions (see Figure 5−34).

**Table 5−33. McBSP General-Purpose I/O Timing Requirements**

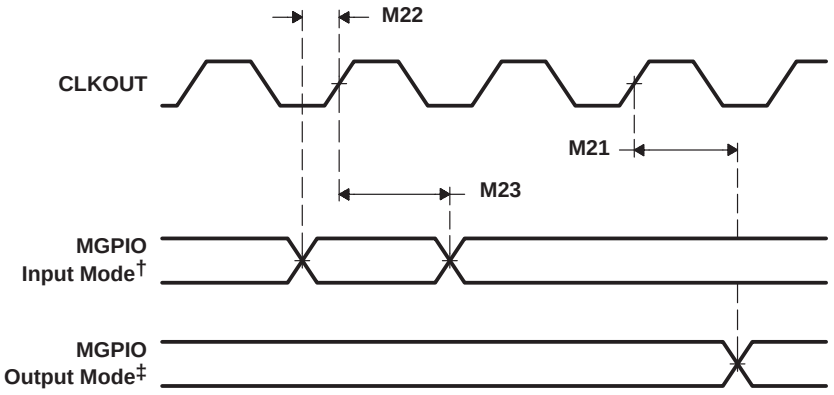
| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| M22 | $t_{su(MGPIO–COH)}$ | Setup time, MGPIOx input mode before CLKOUT high†‡ | 4 | | ns |
| M23 | $t_{h(COH–MGPIO)}$ | Hold time, MGPIOx input mode after CLKOUT high†‡ | 0 | | ns |

† MGPIOx refers to CLKRx, FSRx, DRx, CLKXx, or FSXx when configured as a general-purpose input.
‡ In this case, CLKOUT reflects SYSCLK2. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK2 as CLKOUT.

**Table 5−34. McBSP General-Purpose I/O Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| M21 | $t_{d(COH–MGPIO)}$ | Delay time, CLKOUT high to MGPIOx output mode‡§ | 0 | 6 | ns |

‡ In this case, CLKOUT reflects SYSCLK2. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK2 as CLKOUT.
§ MGPIOx refers to CLKRx, FSRx, CLKXx, FSXx, or DXx when configured as a general-purpose output.




† MGPIOx refers to CLKRx, FSRx, DRx, CLKXx, or FSXx when configured as a general-purpose input.
‡ MGPIOx refers to CLKRx, FSRx, CLKXx, FSXx, or DXx when configured as a general-purpose output.

**Figure 5−34. McBSP General-Purpose I/O Timings**

### 5.15.3 McBSP as SPI Master or Slave Timings

Table 5−35 to Table 5−42 assume testing over recommended operating conditions (see Figure 5−35 through Figure 5−38).

**Table 5−35. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 0)†‡§**

| NO. | | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M30 | $t_{su(DRV–CKXL)}$ | Setup time, DR valid before CLKX low | 13 | | 0 − 5P | | ns |
| M31 | $t_{h(CKXL–DRV)}$ | Hold time, DR valid after CLKX low | 1 | | 9 + 6P | | ns |
| M32 | $t_{su(FXL–CKXH)}$ | Setup time, FSX low before CLKX high | | | 10 | | ns |
| M33 | $t_{c(CKX)}$ | Cycle time, CLKX | 2P | | 16P | | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.
§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

**Table 5−36. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 0)†‡§¶**

| NO. | PARAMETER | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M24 | $t_{d(CKXL–FXL)}$ | Delay time, CLKX low to FSX low# | T − 2 | T + 6 | | | ns |
| M25 | $t_{d(FXL–CKXH)}$ | Delay time, FSX low to CLKX high‖ | C − 6 | C + 4 | | | ns |
| M26 | $t_{d(CKXH–DXV)}$ | Delay time, CLKX high to DX valid | −4 | 6 | 4P | 6P | ns |
| M27 | $t_{dis(CKXL–DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX low | C − 2 | C +10 | | | ns |
| M28 | $t_{dis(FXH–DXHZ)}$ | Disable time, DX high impedance following last data bit from FSX high | | | 2P+ 4 | 4P + 10 | ns |
| M29 | $t_{d(FXL–DXV)}$ | Delay time, FSX low to DX valid | | | 2P + 4 | 4P + 10 | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.
§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).
¶ T = BCLKX period = (1 + CLKGDV) * 2P
C = BCLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * 2P when CLKGDV is even
# FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP
‖ FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

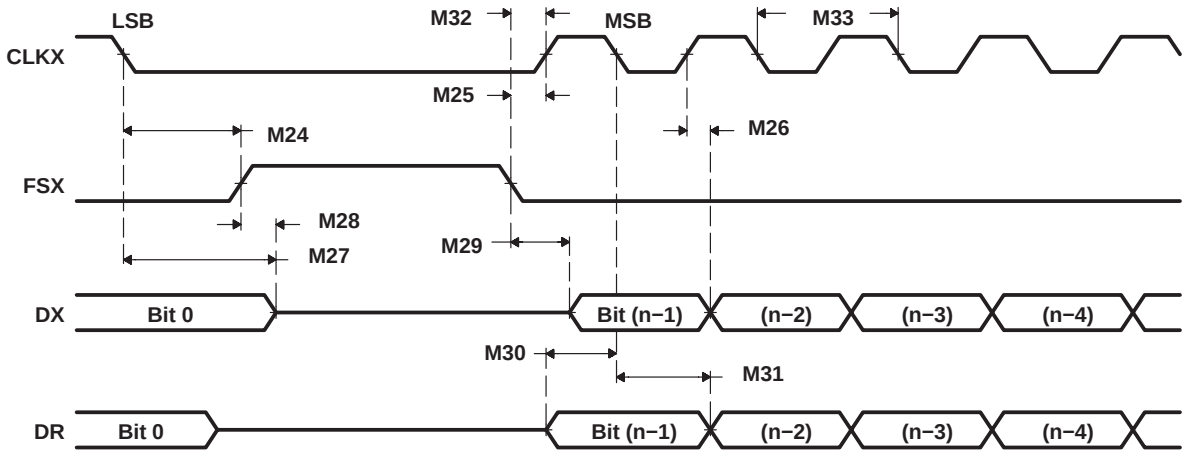



**Figure 5−35. McBSP Timings as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0**

**Table 5−37. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 0)†‡§**

| NO. | | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M39 | $t_{su(DRV–CKXH)}$ | Setup time, DR valid before CLKX high | 13 | | 0 − 5P | | ns |
| M40 | $t_{h(CKXH–DRV)}$ | Hold time, DR valid after CLKX high | 1 | | 9 + 6P | | ns |
| M41 | $t_{su(FXL–CKXH)}$ | Setup time, FSX low before CLKX high | | | 10 | | ns |
| M42 | $t_{c(CKX)}$ | Cycle time, CLKX | 2P | | 16P | | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

**Table 5−38. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 0)†‡§¶**

| NO. | PARAMETER | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M34 | $t_{d(CKXL–FXL)}$ | Delay time, CLKX low to FSX low# | C − 2 | C + 6 | | | ns |
| M35 | $t_{d(FXL–CKXH)}$ | Delay time, FSX low to CLKX high‖ | T − 6 | T + 4 | | | ns |
| M36 | $t_{d(CKXL–DXV)}$ | Delay time, CLKX low to DX valid | −4 | 6 | 4P | 6P | ns |
| M37 | $t_{dis(CKXL–DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX low | −2 | 10 | 3P + 4 | 4P + 18 | ns |
| M38 | $t_{d(FXL–DXV)}$ | Delay time, FSX low to DX valid | D − 2 | D +10 | 2P − 4 | 4P + 10 | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

¶ T = CLKX period = (1 + CLKGDV) * P
C = CLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * P when CLKGDV is even
D = CLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * P when CLKGDV is even

# FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

‖ FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

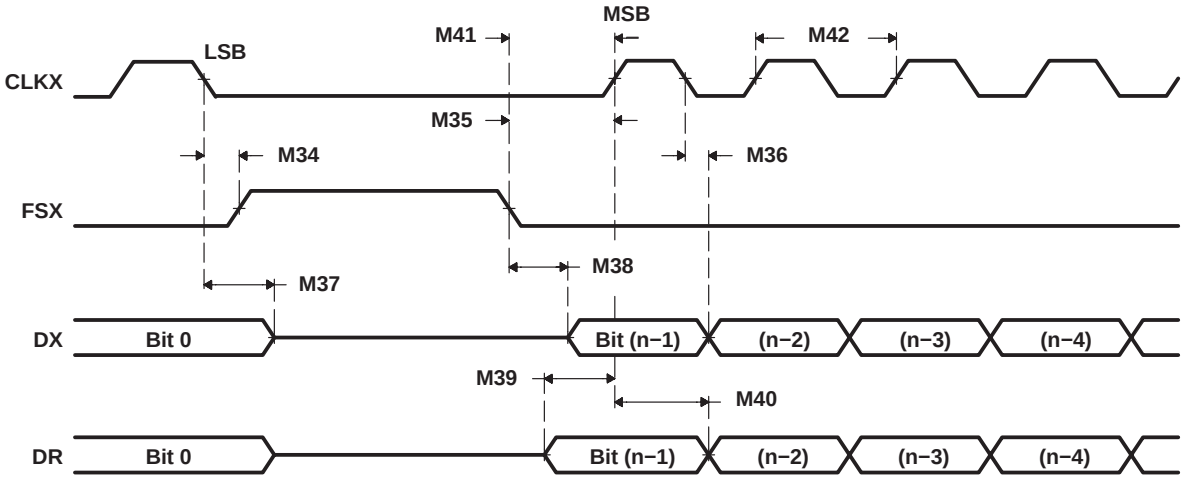



**Figure 5−36. McBSP Timings as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0**

### Table 5−39. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 1)†‡§

| NO. | | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M49 | $t_{su(DRV–CKXH)}$ | Setup time, DR valid before CLKX high | 13 | | 0 − 5P | | ns |
| M50 | $t_{h(CKXH–DRV)}$ | Hold time, DR valid after CLKX high | 1 | | 9 + 6P | | ns |
| M51 | $t_{su(FXL–CKXL)}$ | Setup time, FSX low before CLKX low | | | 10 | | ns |
| M52 | $t_{c(CKX)}$ | Cycle time, CLKX | 2P | | 16P | | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

### Table 5−40. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 1)†‡§¶

| NO. | PARAMETER | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M43 | $t_{d(CKXH–FXL)}$ | Delay time, CLKX high to FSX low# | T − 2 | T + 6 | | | ns |
| M44 | $t_{d(FXL–CKXL)}$ | Delay time, FSX low to CLKX low‖ | D − 6 | D + 4 | | | ns |
| M45 | $t_{d(CKXL–DXV)}$ | Delay time, CLKX low to DX valid | −4 | 6 | 4P | 6P | ns |
| M46 | $t_{dis(CKXH–DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX high | D − 2 | D +10 | | | ns |
| M47 | $t_{dis(FXH–DXHZ)}$ | Disable time, DX high impedance following last data bit from FSX high | | | 2P + 4 | 4P + 10 | ns |
| M48 | $t_{d(FXL–DXV)}$ | Delay time, FSX low to DX valid | | | 2P − 4 | 4P + 10 | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

¶ T = CLKX period = (1 + CLKGDV) * P
D = CLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * P when CLKGDV is even

# FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

‖ FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

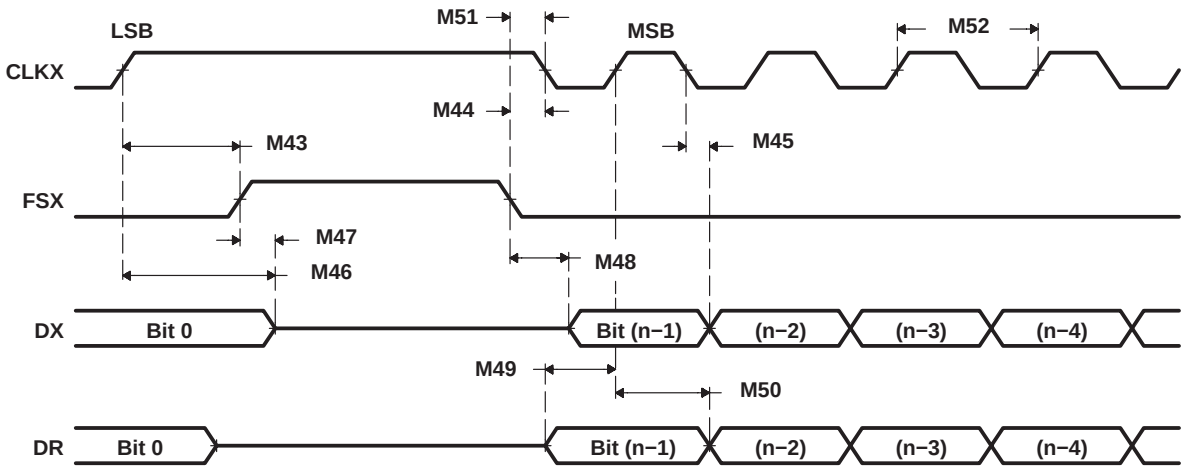



**Figure 5−37. McBSP Timings as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1**

**Table 5−41. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 1)†‡§**

| NO. | | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M58 | $t_{su(DRV–CKXL)}$ | Setup time, DR valid before CLKX low | 13 | | 0 − 5P | | ns |
| M59 | $t_{h(CKXL–DRV)}$ | Hold time, DR valid after CLKX low | 1 | | 9 + 6P | | ns |
| M60 | $t_{su(FXL–CKXL)}$ | Setup time, FSX low before CLKX low | | | 10 | | ns |
| M61 | $t_{c(CKX)}$ | Cycle time, CLKX | 2P | | 16P | | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

**Table 5−42. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 1)†‡§¶**

| NO. | PARAMETER | | MASTER | | SLAVE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| M53 | $t_{d(CKXH–FXL)}$ | Delay time, CLKX high to FSX low# | D − 2 | D + 6 | | | ns |
| M54 | $t_{d(FXL–CKXL)}$ | Delay time, FSX low to CLKX low‖ | T − 6 | T + 4 | | | ns |
| M55 | $t_{d(CKXH–DXV)}$ | Delay time, CLKX high to DX valid | −4 | 6 | 4P | 6P | ns |
| M56 | $t_{dis(CKXH–DXHZ)}$ | Disable time, DX high impedance following last data bit from CLKX high | −2 | 10 | 3P + 4 | 4P + 18 | ns |
| M57 | $t_{d(FXL–DXV)}$ | Delay time, FSX low to DX valid | C − 2 | C +10 | 2P − 4 | 4P + 10 | ns |

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ P = (Divider2 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the slow peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

§ McBSP register values required to configure the McBSP as an SPI master and as an SPI slave are listed in the *TMS320VC5501/5502/5503/5507/5509/5510 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide* (literature number SPRU592).

¶ T = CLKX period = (1 + CLKGDV) * P
C = CLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * P when CLKGDV is even
D = CLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * P when CLKGDV is even

# FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

‖ FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

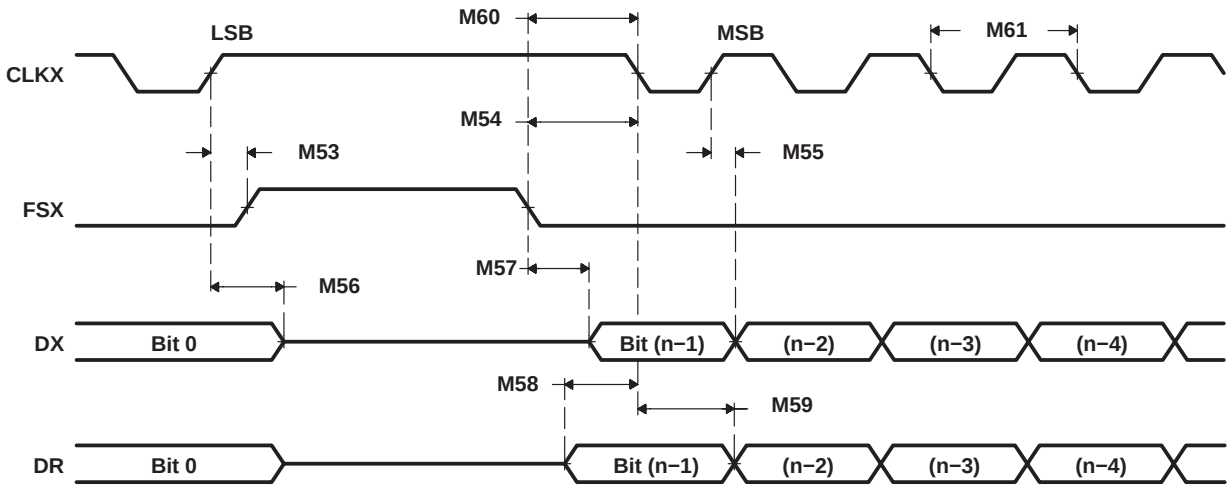



**Figure 5−38. McBSP Timings as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1**

## 5.16 Host-Port Interface Timings

### *5.16.1 HPI Read and Write Timings*

Table 5−43 and Table 5−44 assume testing over recommended operating conditions (see Figure 5−39 through Figure 5−43).

**Table 5−43. HPI Read and Write Timing Requirements†‡§**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| **H9** | $t_{su(HASL-DSL)}$ | Setup time, HPI.$\overline{HAS}$ low before DS falling edge | 5 | | ns |
| **H10** | $t_{h(DSL-HASL)}$ | Hold time, HPI.$\overline{HAS}$ low after DS falling edge | 2 | | ns |
| **H11** | $t_{su(HAD-HASL)}$ | Setup time, HAD valid before HPI.$\overline{HAS}$ falling edge | 5 | | ns |
| **H12** | $t_{h(HASL-HAD)}$ | Hold time, HAD valid after HPI.$\overline{HAS}$ falling edge | 5 | | ns |
| **H13** | $t_{w(DSL)}$ | Pulse duration, DS low | 15 | | ns |
| **H14** | $t_{w(DSH)}$ | Pulse duration, DS high | 2P | | ns |
| **H15** | $t_{su(HAD-DSL)}$ | Setup time, HAD valid before DS falling edge | 5 | | ns |
| **H16** | $t_{h(DSL-HAD)}$ | Hold time, HAD valid after DS falling edge | 5 | | ns |
| **H17** | $t_{su(HD-DSH)}$ | Setup time, HD valid before DS rising edge | 5 | | ns |
| **H18** | $t_{h(DSH-HD)}$ | Hold time, HD valid after DS rising edge | 0 | | ns |
| **H37** | $t_{su(HCSL-DSL)}$ | Setup time, $\overline{HCS}$ low before DS falling edge | 0 | | ns |
| **H38** | $t_{h(HRDYH-DSL)}$ | Hold time, DS low after HRDY rising edge | 0 | | ns |

† P = (Divider1 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the fast peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

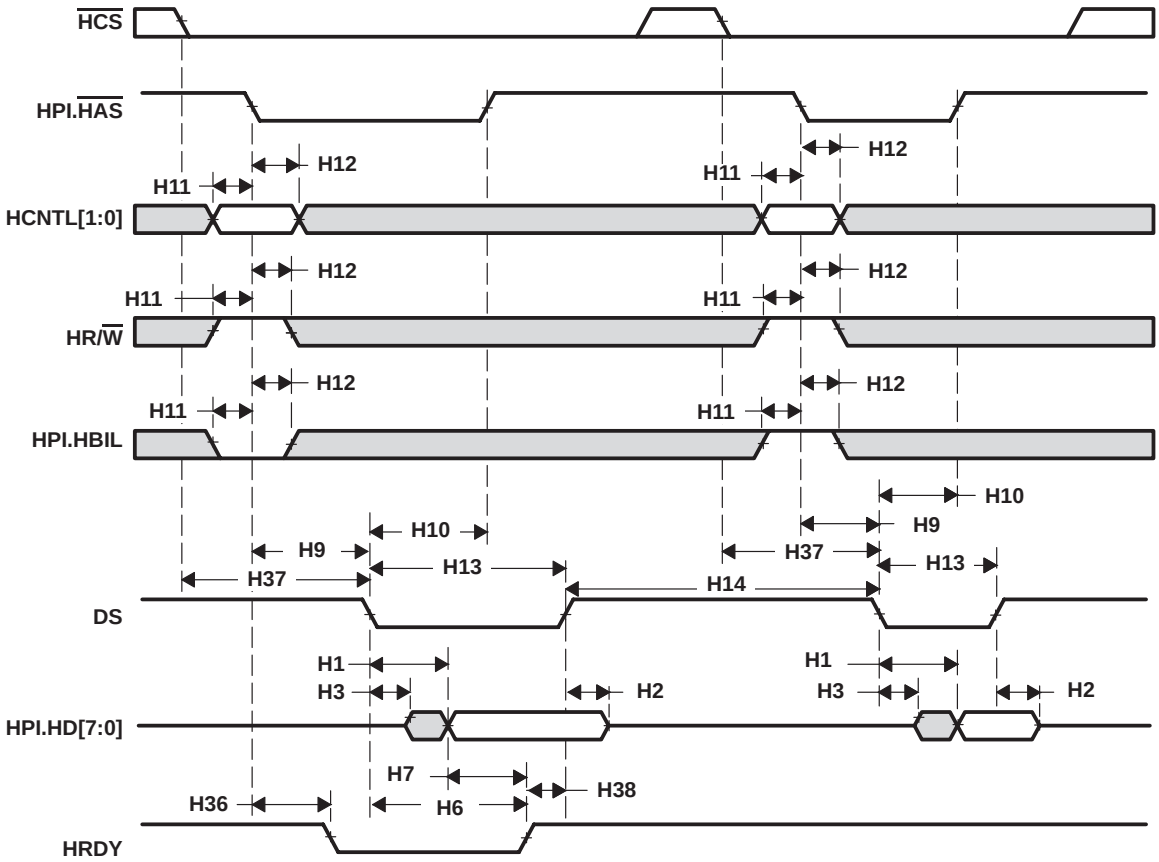
‡ DS refers to logical OR of $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$. HD refers to HPI Data Bus. HDS refers to $\overline{HDS1}$ or $\overline{HDS2}$. HAD refers to HCNTL0, HCNTL1, HPI.HBIL, and HR/$\overline{W}$.

§ A host must not initiate transfer requests until the HPI has been brought out of reset, see Section 3.7, Host-Port Interface (HPI), for more details.

**Table 5−44. HPI Read and Write Switching Characteristics†‡§**

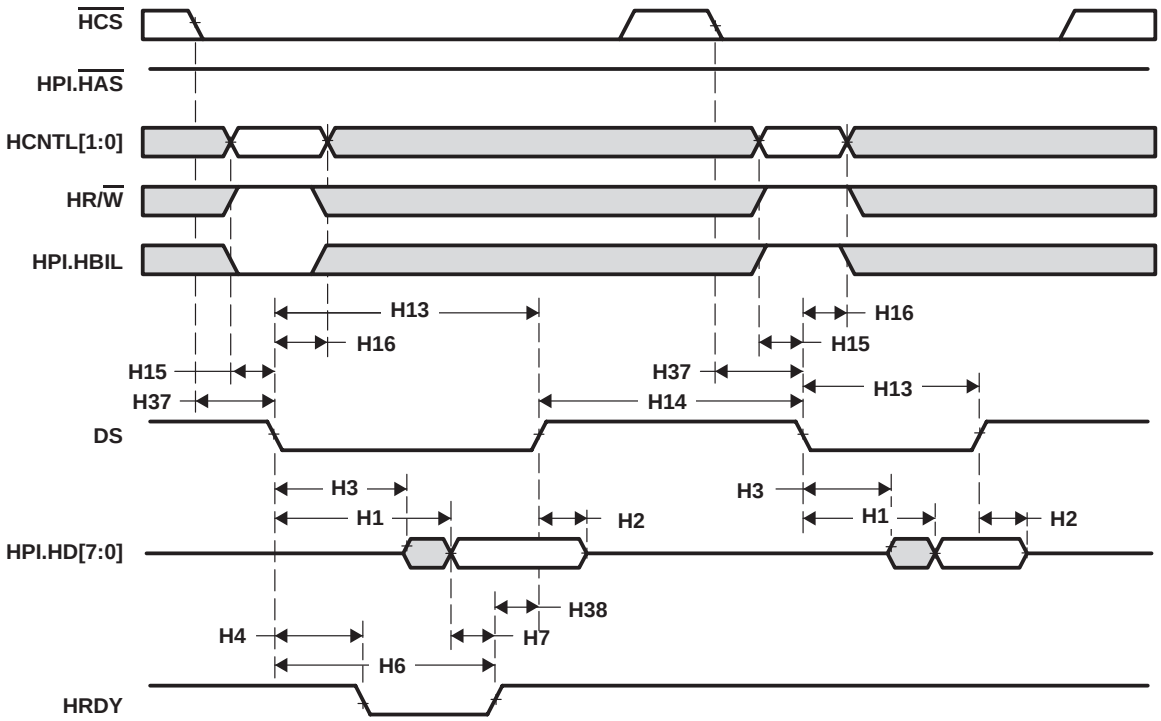
| NO. | PARAMETER | | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| H1 | $t_{d(DSL-HDV)}$ | Delay time, DS low to HD valid | *Case 1.* HPIC or HPIA read | | 5 | 15 | ns |
| | | | *Case 2.* HPID read with no auto-increment¶ | K = 1‖ | | 9 * 2H + 20 | |
| | | | | K = 2‖ | | 10 * 2H + 20 | |
| | | | | K = 4‖ | | 11 * 2H + 20 | |
| | | | *Case 3.* HPID read with auto-increment and read FIFO initially empty¶ | K = 1‖ | | 9 * 2H + 20 | |
| | | | | K = 2‖ | | 10 * 2H + 20 | |
| | | | | K = 4‖ | | 11 * 2H + 20 | |
| | | | *Case 4.* HPID read with auto-increment and data previously prefetched into the read FIFO | | 5 | 15 | |
| H2 | $t_{dis(DSH-HDV)}$ | Disable time, HD high-impedance from DS high | | | 1 | 4 | ns |
| H3 | $t_{en(DSL-HDD)}$ | Enable time, HD driven from DS low | | | 3 | 15 | ns |
| H4 | $t_{d(DSL-HRDYL)}$ | Delay time, DS low to HRDY low | | | | 12 | ns |
| H5 | $t_{d(DSH-HRDYL)}$ | Delay time, DS high to HRDY low | | | | 12 | ns |
| H6 | $t_{d(DSL-HRDYH)}$ | Delay time, DS low to HRDY high | *Case 1.* HPID read with no auto-increment¶ | K = 1‖ | | 10 * 2H + 20 | ns |
| | | | | K = 2‖ | | 11 * 2H + 20 | |
| | | | | K = 4‖ | | 12 * 2H + 20 | |
| | | | *Case 2.* HPID read with auto-increment and read FIFO initially empty¶ | K = 1‖ | | 10 * 2H + 20 | ns |
| | | | | K = 2‖ | | 11 * 2H + 20 | |
| | | | | K = 4‖ | | 12 * 2H + 20 | |
| H7 | $t_{d(HDV-HRDYH)}$ | Delay time, HD valid to HRDY high | | | 0 | | ns |
| H8 | $t_{d(COH-HINT)}$ | Delay time, CLKOUT high to $\overline{HINT}$ change# | | | | 8 | ns |
| H34 | $t_{d(DSH-HRDYH)}$ | Delay time, DS high to HRDY high | *Case 1.* HPIA write¶ | K = 1, 2, 4‖ | | 5 * 2H + 20 | ns |
| | | | *Case 2.* HPID write with no auto-increment¶ | K = 1‖ | | 5 * 2H + 20 | |
| | | | | K = 2‖ | | 5 * 2H + 20 | |
| | | | | K = 4‖ | | 6 * 2H + 20 | |
| H35 | $t_{d(DSL-HRDYH)}$ | Delay time, DS low to HRDY high for HPIA write and FIFO not empty¶ | | K = 1‖ | | 40 * 2H + 20 | ns |
| | | | | K = 2‖ | | 40 * 2H + 20 | |
| | | | | K = 4‖ | | 24 * 2H + 20 | |
| H36 | $t_{d(HASL-HRDYL)}$ | Delay time, HPI.$\overline{HAS}$ low to HRDY low | | | | 12 | ns |

† DS refers to logical OR of $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$. HD refers to HPI Data Bus. HDS refers to $\overline{HDS1}$ or $\overline{HDS2}$. HAD refers to HCNTL0, HCNTL1, HPI.HBIL, and HR/$\overline{W}$.
‡ H is half the SYSCLK1 clock cycle.
§ A host must not initiate transfer requests until the HPI has been brought out of reset, see Section 3.7, Host-Port Interface (HPI), for more details.
¶ Assumes no other DMA or CPU memory activity.
# In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.
‖ K = divider ratio between CPU clock and SYSCLK1. For example, when SYSCLK1 is set to the CPU clock divided by four, use K = 4.



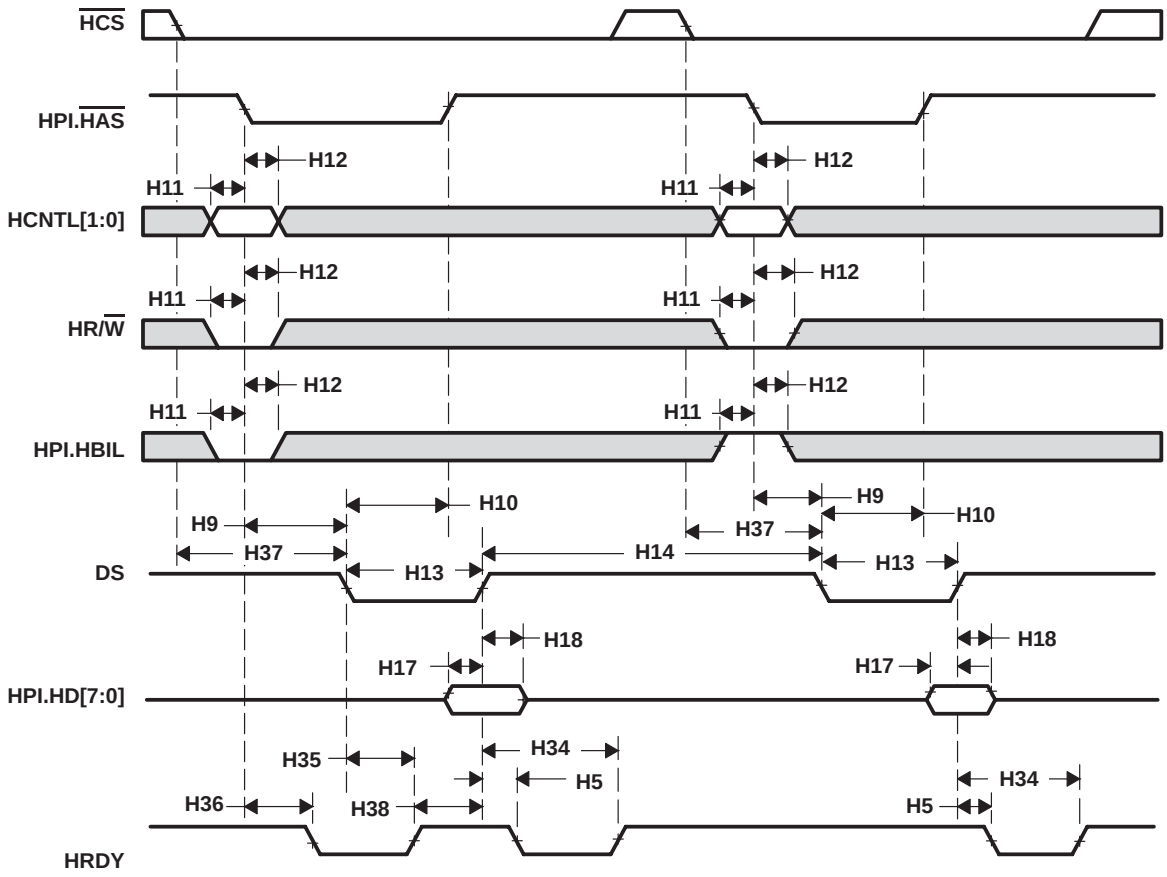

NOTE: Depending on the type of write or read operation (HPID without auto-incrementing, HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur [see the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)].

**Figure 5−39. Multiplexed Read Timings Using HPI.HAS**



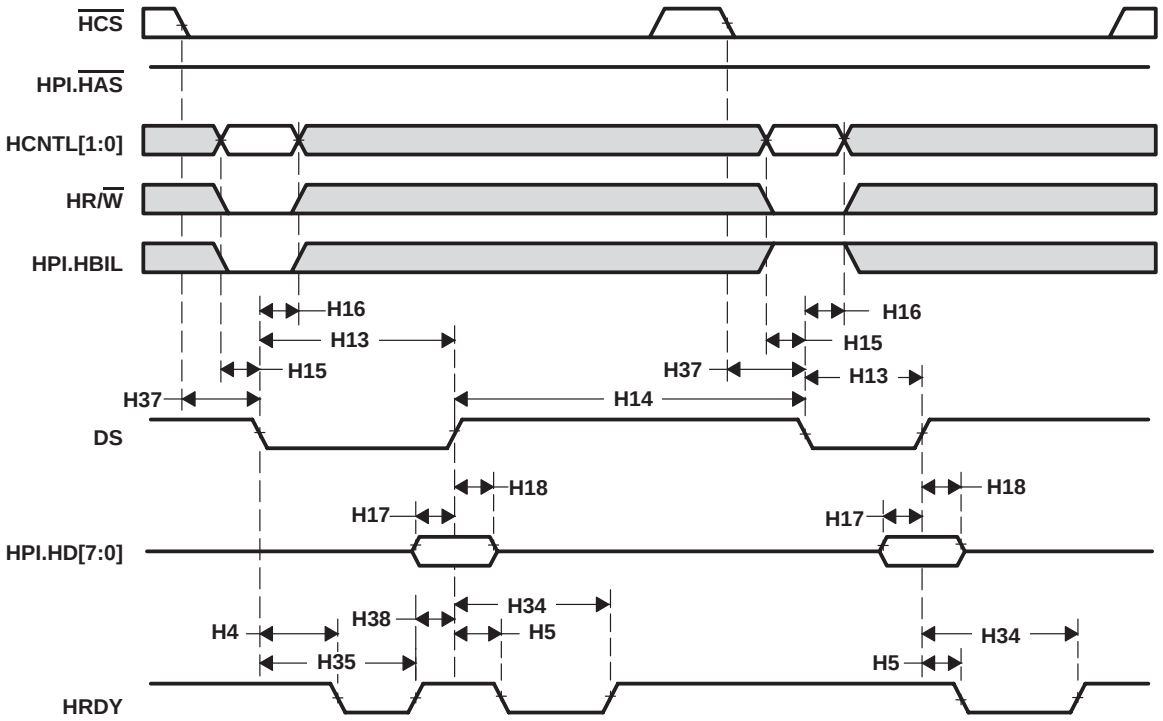

NOTE: Depending on the type of write or read operation (HPID without auto-incrementing, HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur [see the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)].

**Figure 5−40. Multiplexed Read Timings With HPI.HAS Held High**




NOTE: Depending on the type of write or read operation (HPID without auto-incrementing, HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur [see the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)].

**Figure 5−41. Multiplexed Write Timings Using HPI.HAS**




NOTE: Depending on the type of write or read operation (HPID without auto-incrementing, HPIA, HPIC, or HPID with auto-incrementing) and the state of the FIFO, transitions on HRDY may or may not occur [see the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620)].

**Figure 5−42. Multiplexed Write Timings With HPI.HAS Held High**

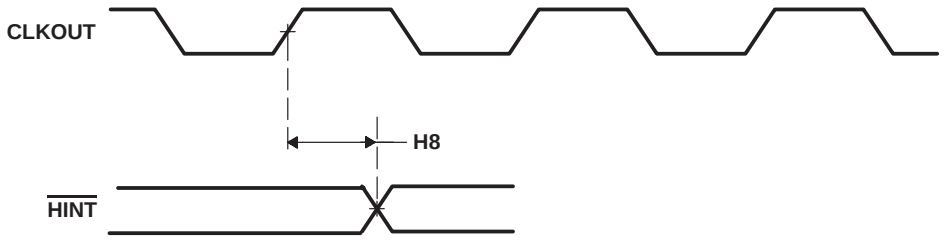



**Figure 5−43. HINT Timings**

### 5.16.2 HPI General-Purpose I/O Timings

Table 5−45 and Table 5−46 assume testing over recommended operating conditions (see Figure 5−44).

**Table 5−45. HPI General-Purpose I/O Timing Requirements†**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| H27 | $t_{su(HDGPIO-COH)}$ | Setup time, HDGPIO input mode before CLKOUT high‡ | 5 | | ns |
| H28 | $t_{h(COH-HDGPIO)}$ | Hold time, HDGPIO input mode after CLKOUT high‡ | 0 | | ns |
| H29 | $t_{su(HCGPIO-COH)}$ | Setup time, HCGPIO input mode before CLKOUT high§ | 5 | | ns |
| H30 | $t_{h(COH-HCGPIO)}$ | Hold time, HCGPIO input mode after CLKOUT high§ | 0 | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.
‡ HDGPIO refers to HPI.HD[7:0] configured as general-purpose input.
§ HCGPIO refers to HPI.$\overline{HAS}$, HPI.HBIL, HCNTL0, HCNTL1, $\overline{HCS}$, HR/$\overline{W}$, $\overline{HDS1}$, $\overline{HDS2}$, HRDY, and $\overline{HINT}$ configured as general-purpose input.

**Table 5−46. HPI General-Purpose I/O Switching Characteristics†**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| H21 | $t_{d(COH-HDGPIO)}$ | Delay time, CLKOUT high to HDGPIO output mode¶ | | 10 | ns |
| H22 | $t_{d(COH-HCGPIO)}$ | Delay time, CLKOUT high to HCGPIO output mode# | | 10 | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.
¶ HDGPIO refers to HPI.HD[7:0] configured as general-purpose output.
# HCGPIO refers to HPI.$\overline{HAS}$, HPI.HBIL, HCNTL0, HCNTL1, $\overline{HCS}$, HR/$\overline{W}$, $\overline{HDS1}$, $\overline{HDS2}$, HRDY, and $\overline{HINT}$ configured as general-purpose output.

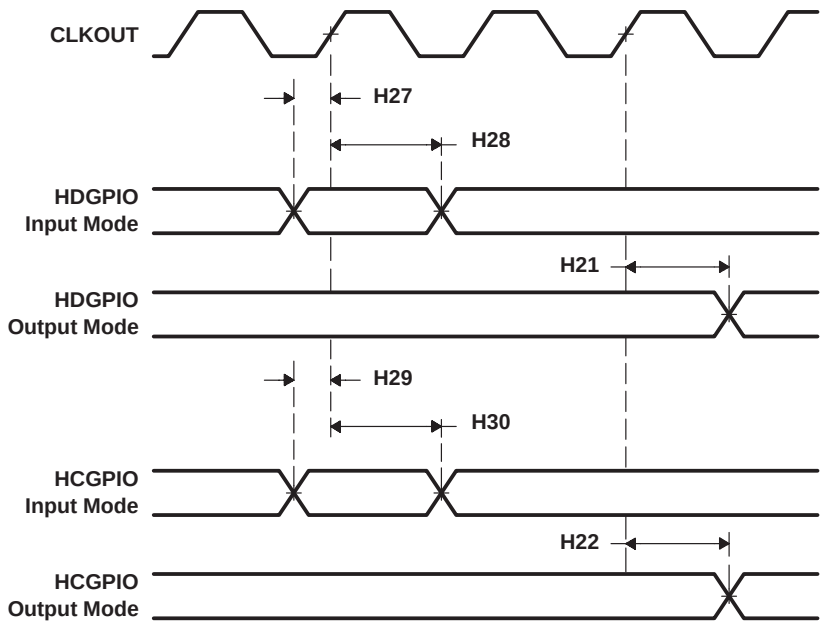



**Figure 5−44. HPI General-Purpose I/O Timings**

### 5.16.3 HPI.$\overline{\text{HAS}}$ Interrupt Timings

Table 5−47 assumes testing over recommended operating conditions (see Figure 5−45).

**Table 5−47. HPI.$\overline{\text{HAS}}$ Interrupt Timing Requirements†**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| H31 | $t_{su(HASL-COH)}$ | Setup time, HPI.$\overline{\text{HAS}}$ low‡ before CLKOUT rising edge | 5 | | ns |
| H32 | $t_{h(COH-HASL)}$ | Hold time, HPI.$\overline{\text{HAS}}$ low‡ after CLKOUT rising edge | 0 | | ns |
| H33 | $t_{w(HASL)}$ | Pulse width, HPI.$\overline{\text{HAS}}$ low‡ | P§ | | ns |

† In this case, CLKOUT reflects SYSCLK1. The CLKOUT Selection Register (CLKOUTSR) can be programmed to select SYSCLK1 as CLKOUT.
‡ An interrupt can be triggered by setting the HPI.$\overline{\text{HAS}}$ signal high or low, depending on the setting of the HAS bit in the General-Purpose I/O Interrupt Control Register 2 (HPGPIOINT2). Refer to the *TMS320VC5501/5502 DSP Host Port Interface (HPI) Reference Guide* (literature number SPRU620) for more information on the interrupt capability of the HPI.$\overline{\text{HAS}}$ signal.
§ P = (Divider1 Ratio)/(CPU Clock Frequency) in ns. For example, when running parts at 300 MHz with the fast peripheral domain at 1/2 the CPU clock frequency, use P = 2/300 MHz = 6.66 ns.

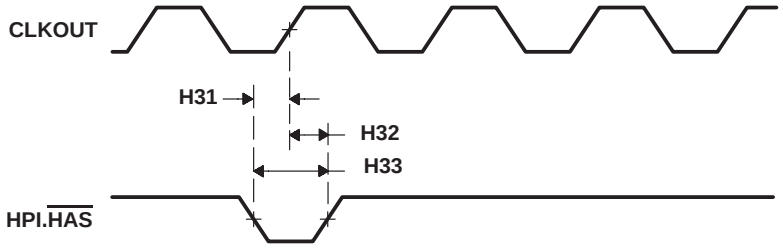



**Figure 5−45. HPI.$\overline{\text{HAS}}$ Interrupt Timings**

## 5.17 Inter-Integrated Circuit ($I^2C$) Timings

Table 5−48 and Table 5−49 assume testing over recommended operating conditions (see Figure 5−46 and Figure 5−47).

**Table 5−48. $I^2C$ Signals (SDA and SCL) Timing Requirements**

| NO. | | | STANDARD MODE | | FAST MODE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| IC1 | $t_{c(SCL)}$ | Cycle time, SCL | 10 | | 2.5 | | µs |
| IC2 | $t_{su(SCLH-SDAL)}$ | Setup time, SCL high before SDA low for a repeated START condition | 4.7 | | 0.6 | | µs |
| IC3 | $t_{h(SCLL-SDAL)}$ | Hold time, SCL low after SDA low for a START and a repeated START condition | 4 | | 0.6 | | µs |
| IC4 | $t_{w(SCLL)}$ | Pulse duration, SCL low | 4.7 | | 1.3 | | µs |
| IC5 | $t_{w(SCLH)}$ | Pulse duration, SCL high | 4 | | 0.6 | | µs |
| IC6 | $t_{su(SDA-SCLH)}$ | Setup time, SDA valid before SCL high | 250 | | 100† | | ns |
| IC7 | $t_{h(SDA-SCLL)}$ | Hold time, SDA valid after SCL low \| For $I^2C$ bus™ devices | 0‡ | | 0‡ | 0.9§ | µs |
| IC8 | $t_{w(SDAH)}$ | Pulse duration, SDA high between STOP and START conditions | 4.7 | | 1.3 | | µs |
| IC9 | $t_{r(SDA)}$ | Rise time, SDA | | 1000 | 20 + 0.1$C_b$¶ | 300 | ns |
| IC10 | $t_{r(SCL)}$ | Rise time, SCL | | 1000 | 20 + 0.1$C_b$¶ | 300 | ns |
| IC11 | $t_{f(SDA)}$ | Fall time, SDA | | 300 | 20 + 0.1$C_b$¶ | 300 | ns |
| IC12 | $t_{f(SCL)}$ | Fall time, SCL | | 300 | 20 + 0.1$C_b$¶ | 300 | ns |
| IC13 | $t_{su(SCLH-SDAH)}$ | Setup time, SCL high before SDA high (for STOP condition) | 4.0 | | 0.6 | | µs |
| IC14 | $t_{w(SP)}$ | Pulse duration, spike (must be suppressed) | | | 0 | 50 | ns |
| IC15 | $C_b$¶ | Capacitive load for each bus line | | 400 | | 400 | pF |

† A Fast-mode $I^2C$-bus device can be used in a Standard-mode $I^2C$-bus system, but the requirement $t_{su(SDA-SCLH)} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_r$ max + $t_{su(SDA-SCLH)}$ = 1000 + 250 = 1250 ns (according to the Standard-mode $I^2C$-Bus Specification) before the SCL line is released.

‡ A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the $V_{IHmin}$ of the SCL signal) to bridge the undefined region of the falling edge of SCL.

§ The maximum $t_{h(SDA-SCLL)}$ has only to be met if the 5501 $I^2C$ operates in master-receiver mode and the slave device does not stretch the LOW period [$t_{w(SCLL)}$] of the SCL signal.

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

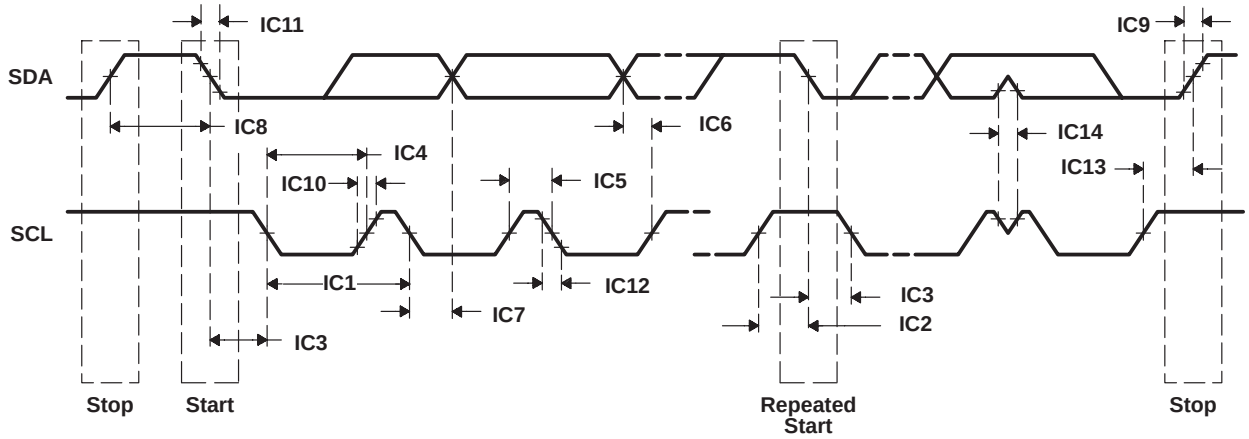



**Figure 5−46. $I^2C$ Receive Timings**

$I^2C$ Bus is a trademark of Koninklijke Philips Electronics N.V.

**Table 5−49. I²C Signals (SDA and SCL) Switching Characteristics**

| NO. | PARAMETER | | STANDARD MODE | | FAST MODE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| IC16 | $t_{c(SCL)}$ | Cycle time, SCL | 10 | | 2.5 | | µs |
| IC17 | $t_{d(SCLH-SDAL)}$ | Delay time, SCL high to SDA low for a repeated START condition | 4.7 | | 0.6 | | µs |
| IC18 | $t_{d(SDAL-SCLL)}$ | Delay time, SDA low to SCL low for a START and a repeated START condition | 4 | | 0.6 | | µs |
| IC19 | $t_{w(SCLL)}$ | Pulse duration, SCL low | 4.7 | | 1.3 | | µs |
| IC20 | $t_{w(SCLH)}$ | Pulse duration, SCL high | 4 | | 0.6 | | µs |
| IC21 | $t_{d(SDA-SCLH)}$ | Delay time, SDA valid to SCL high | 250 | | 100 | | ns |
| IC22 | $t_{v(SCLL-SDAV)}$ | Valid time, SDA valid after SCL low \| For I²C bus devices | 0 | | 0 | | µs |
| IC23 | $t_{w(SDAH)}$ | Pulse duration, SDA high between STOP and START conditions | 4.7 | | 1.3 | | µs |
| IC24 | $t_{r(SDA)}$ | Rise time, SDA | | 1000 | $20 + 0.1C_b$† | 300 | ns |
| IC25 | $t_{r(SCL)}$ | Rise time, SCL | | 1000 | $20 + 0.1C_b$† | 300 | ns |
| IC26 | $t_{f(SDA)}$ | Fall time, SDA | | 300 | $20 + 0.1C_b$† | 300 | ns |
| IC27 | $t_{f(SCL)}$ | Fall time, SCL | | 300 | $20 + 0.1C_b$† | 300 | ns |
| IC28 | $t_{d(SCLH-SDAH)}$ | Delay time, SCL high to SDA high for a STOP condition | 4 | | 0.6 | | µs |
| IC29 | $C_p$ | Capacitance for each I²C pin | | 10 | | 10 | pF |

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

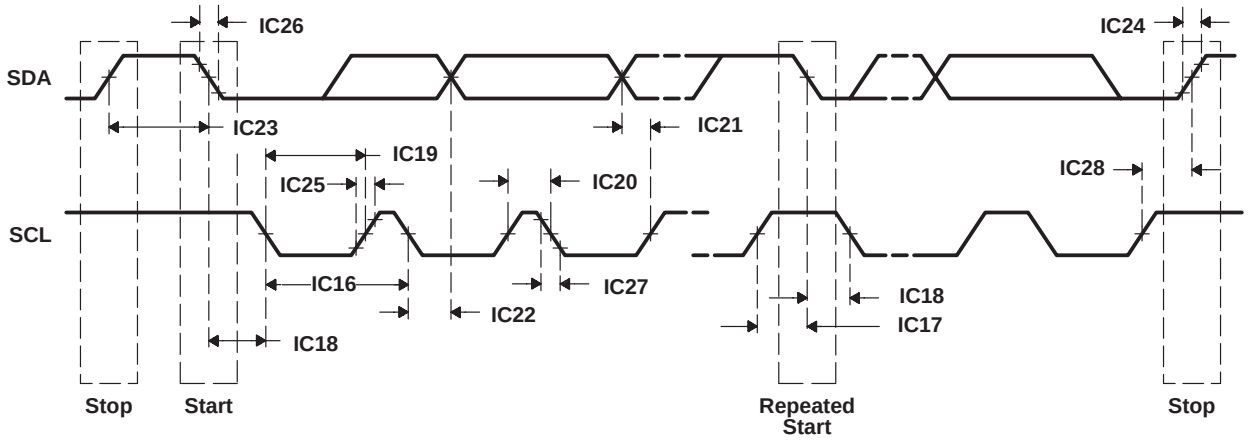



**Figure 5−47. I²C Transmit Timings**

## 5.18 Universal Asynchronous Receiver/Transmitter (UART) Timings

Table 5−50 to Table 5−51 assume testing over recommended operating conditions (see Figure 5−48).

**Table 5−50. UART Timing Requirements**

| NO. | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| U4 | $t_{w(UDB)R}$ | Pulse width, receive data bit | 0.96U† | 1.05U† | ns |
| U5 | $t_{w(USB)R}$ | Pulse width, receive start bit | 0.96U† | 1.05U† | ns |

† U = UART baud time = 1/programmed baud rate

**Table 5−51. UART Switching Characteristics**

| NO. | PARAMETER | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| U1 | $f_{baud}$ | Maximum programmable baud rate | | 5 | MHz |
| U2 | $t_{w(UDB)X}$ | Pulse width, transmit data bit | U − 2† | U + 2† | ns |
| U3 | $t_{w(USB)X}$ | Pulse width, transmit start bit | U − 2† | U + 2† | ns |

† U = UART baud time = 1/programmed baud rate

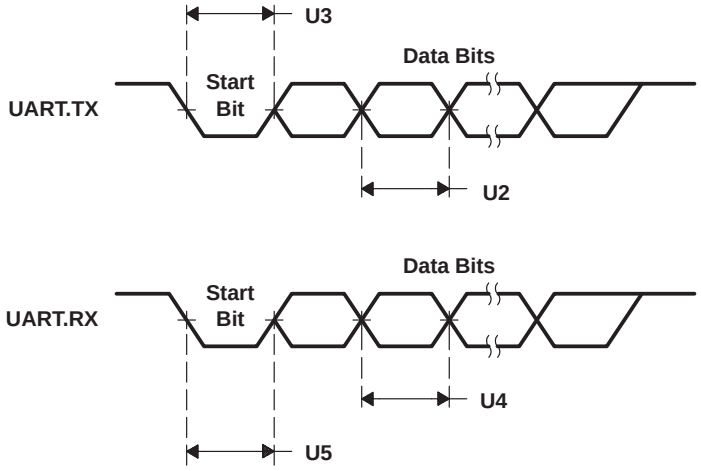



**Figure 5−48. UART Timings**

# 6 Mechanical Data

Some TMX samples were shipped in the GGW package. For more information on the GGW package, see the *TMS320VC5502 and TMS320VC5501 Digital Signal Processors Silicon Errata* (literature number SPRZ020D or later).

TMS320VC5501PGF has completed Temp Cycle reliability qualification testing with no failures through 1500 cycles of −55°C to 125°C following an EIA/JEDEC Moisture Sensitivity Level 4 pre-condition at 220+5/−0°C peak reflow. Exceeding this peak reflow temperature condition or storage and handling requirements may result in either immediate device failure post-reflow, due to package/die material delamination (“popcorning”), or degraded Temp cycle life performance.

Please note that Texas Instruments (TI) also provides MSL, peak reflow and floor life information on a bar-code label affixed to dry-pack shipping bags. Shelf life, temperature and humidity storage conditions and re-bake instructions are prominently displayed on a nearby screen-printed label.

## 6.1 Package Thermal Resistance Characteristics

Table 6−1 and Table 6−2 provide the thermal resistance characteristics for the recommended package types used on the TMS320VC5501 DSP.

**NOTE:**
Some TMX samples were shipped in the GGW package. For more information on the GGW package, see the *TMS320VC5502 and TMS320VC5501 Digital Signal Processors Silicon Errata* (literature number SPRZ020D or later).

**Table 6−1. Thermal Resistance Characteristics (Ambient)**

| PACKAGE | | $R_{\theta JA}$ (°C/W) | BOARD TYPE† | AIRFLOW (LFM) |
|---|---|---|---|---|
| GZZ, ZZZ | (Without thermal vias) | 94 | High-K | 0 |
| | | 93 | High-K | 150 |
| | | 91 | High-K | 250 |
| | | 87 | High-K | 500 |
| | | 117 | Low-K | 0 |
| | | 114 | Low-K | 150 |
| | | 109 | Low-K | 250 |
| | | 101 | Low-K | 500 |
| | (With thermal vias)‡ | 39 | High-K | 0 |
| | | 37 | High-K | 150 |
| | | 36 | High-K | 250 |
| | | 34 | High-K | 500 |

† Board types are as defined by JEDEC. Reference JEDEC Standard JESD51−9, Test Boards for Area Array Surface Mount Package Thermal Measurements.

‡ Adding thermal vias will significantly improve the thermal performance of the device. To use the thermal balls on the GZZ and ZZZ packages:
- An array of 25 land pads must be added on the top layer of the PCB where the package will be mounted.
- The PCB land pads should be the same diameter as the vias in the package substrate for optimal Board Level Reliability Temperature Cycle performance.
- The land pads on the PCB should be connected together and to PCB through-holes. The PCB through-holes should in turn be connected to the ground plane for heat dissipation.
- A solid internal plane is preferred for spreading the heat.

Refer to the *MicroStar BGA ™ Packaging Reference Guide* (literature number SSYZ015) for guidance on PCB design, surface mount, and reliability considerations.

**Table 6−1. Thermal Resistance Characteristics (Ambient) (Continued)**

| PACKAGE | $R_{\theta JA}$ (°C/W) | BOARD TYPE† | AIRFLOW (LFM) |
|---|---|---|---|
| PGF | 60 | High-K | 0 |
| | 52 | High-K | 150 |
| | 49 | High-K | 250 |
| | 45 | High-K | 500 |
| | 104 | Low-K | 0 |
| | 81 | Low-K | 150 |
| | 73 | Low-K | 250 |
| | 64 | Low-K | 500 |

† Board types are as defined by JEDEC. Reference JEDEC Standard JESD51−9, Test Boards for Area Array Surface Mount Package Thermal Measurements.

‡ Adding thermal vias will significantly improve the thermal performance of the device. To use the thermal balls on the GZZ and ZZZ packages:

- An array of 25 land pads must be added on the top layer of the PCB where the package will be mounted.
- The PCB land pads should be the same diameter as the vias in the package substrate for optimal Board Level Reliability Temperature Cycle performance.
- The land pads on the PCB should be connected together and to PCB through-holes. The PCB through-holes should in turn be connected to the ground plane for heat dissipation.
- A solid internal plane is preferred for spreading the heat.

Refer to the *MicroStar BGA ™ Packaging Reference Guide* (literature number SSYZ015) for guidance on PCB design, surface mount, and reliability considerations.

**Table 6−2. Thermal Resistance Characteristics (Case)**

| PACKAGE | $R_{\theta JC}$ (°C/W) | BOARD TYPE† |
|---|---|---|
| GZZ, ZZZ | 22 | 2s JEDEC Test Card |
| PGF | 13.2 | 2s JEDEC Test Card |

† Board types are as defined by JEDEC. Reference JEDEC Standard JESD51−9, Test Boards for Area Array Surface Mount Package Thermal Measurements.

## 6.2 Packaging Information

The following packaging information reflects the most current released data available for the designated device(s). This data is subject to change without notice and without revision of this document.

## PACKAGING INFORMATION

| Orderable part number | Status (1) | Material type (2) | Package \| Pins | Package qty \| Carrier | RoHS (3) | Lead finish/ Ball material (4) | MSL rating/ Peak reflow (5) | Op temp (°C) | Part marking (6) |
|---|---|---|---|---|---|---|---|---|---|
| C55EQUAL | Active | Production | LQFP (PGF) \| 176 | 40 \| JEDEC TRAY (5+1) | Yes | NIPDAU | Level-4-260C-72 HR | -40 to 85 | 320VC5501PGF TMS 300 |
| TMS320VC5501GBE300 | Active | Production | NFBGA (GBE) \| 201 | 126 \| JEDEC TRAY (5+1) | No | SNPB | Level-3-220C-168 HR | -40 to 85 | TMS 320VC5501GBE A 300 |
| TMS320VC5501GBE300.A | Active | Production | NFBGA (GBE) \| 201 | 126 \| JEDEC TRAY (5+1) | No | SNPB | Level-3-220C-168 HR | -40 to 85 | TMS 320VC5501GBE A 300 |
| TMS320VC5501GZZ300 | Obsolete | Production | BGA MICROSTAR (GZZ) \| 201 | - | - | Call TI | Call TI | -40 to 85 | 320VC5501GZZ TMS 300 |
| TMS320VC5501PGF300 | Active | Production | LQFP (PGF) \| 176 | 40 \| JEDEC TRAY (5+1) | Yes | NIPDAU | Level-4-260C-72 HR | -40 to 85 | 320VC5501PGF TMS 300 |
| TMS320VC5501PGF300.A | Active | Production | LQFP (PGF) \| 176 | 40 \| JEDEC TRAY (5+1) | Yes | NIPDAU | Level-4-260C-72 HR | -40 to 85 | 320VC5501PGF TMS 300 |
| TMS320VC5501ZAV300 | Active | Production | NFBGA (ZAV) \| 201 | 126 \| JEDEC TRAY (5+1) | Yes | SNAGCU | Level-3-260C-168 HR | -40 to 85 | TMS 320VC5501ZAV A 300 |
| TMS320VC5501ZAV300.A | Active | Production | NFBGA (ZAV) \| 201 | 126 \| JEDEC TRAY (5+1) | Yes | SNAGCU | Level-3-260C-168 HR | -40 to 85 | TMS 320VC5501ZAV A 300 |
| TMS320VC5501ZZZ300 | Obsolete | Production | BGA MICROSTAR (ZZZ) \| 201 | - | - | Call TI | Call TI | -40 to 85 | 320VC5501ZZZ TMS 300 |
| TNETV2501INPGF | Active | Production | LQFP (PGF) \| 176 | 40 \| JEDEC TRAY (5+1) | Yes | NIPDAU | Level-4-260C-72 HR | -40 to 85 | 320VC5501PGF TMS 300 |

| Orderable part number | Status (1) | Material type (2) | Package \| Pins | Package qty \| Carrier | RoHS (3) | Lead finish/ Ball material (4) | MSL rating/ Peak reflow (5) | Op temp (°C) | Part marking (6) |
|---|---|---|---|---|---|---|---|---|---|
| VC5501LMPGF300 | Active | Production | LQFP (PGF) \| 176 | 40 \| JEDEC TRAY (5+1) | Yes | NIPDAU | Level-4-260C-72 HR | -40 to 85 | 320VC5501PGF TMS 300 |

(1) **Status:** For more details on status, see our product life cycle.

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

(3) **RoHS values:** Yes, No, RoHS Exempt. See the TI RoHS Statement for additional information and value definition.

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

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

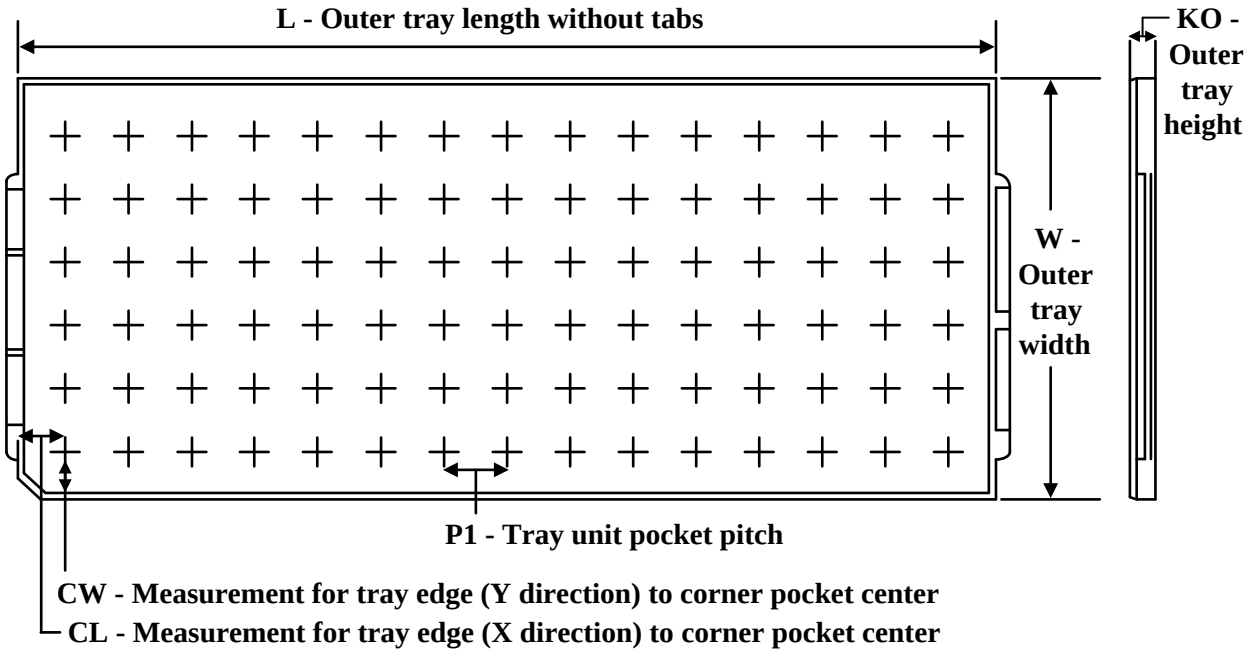
(6) **Part marking:** There may be an additional marking, which relates to the logo, the lot trace code information, or the environmental category of the part.

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

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

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

## TRAY

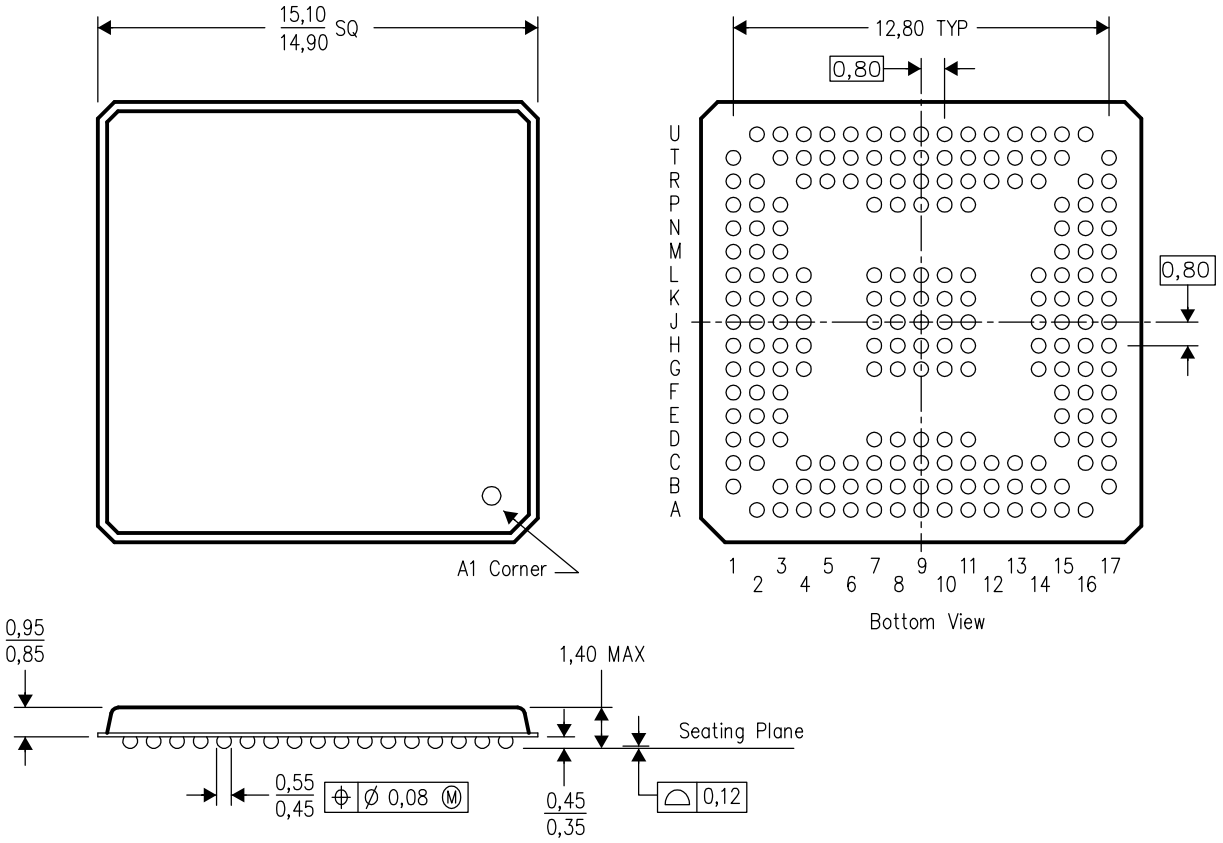



Chamfer on Tray corner indicates Pin 1 orientation of packed units.

*All dimensions are nominal

| Device | Package Name | Package Type | Pins | SPQ | Unit array matrix | Max temperature (°C) | L (mm) | W (mm) | K0 (µm) | P1 (mm) | CL (mm) | CW (mm) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| C55EQUAL | PGF | LQFP | 176 | 40 | 4x10 | 150 | 315 | 135.9 | 7620 | 20.7 | 30.4 | 20.7 |
| TMS320VC5501GBE300 | GBE | NFBGA | 201 | 126 | 7 X 18 | 150 | 315 | 135.9 | 7620 | 17.2 | 11.3 | 16.35 |
| TMS320VC5501GBE300.A | GBE | NFBGA | 201 | 126 | 7 X 18 | 150 | 315 | 135.9 | 7620 | 17.2 | 11.3 | 16.35 |
| TMS320VC5501PGF300 | PGF | LQFP | 176 | 40 | 4x10 | 150 | 315 | 135.9 | 7620 | 20.7 | 30.4 | 20.7 |
| TMS320VC5501PGF300.A | PGF | LQFP | 176 | 40 | 4x10 | 150 | 315 | 135.9 | 7620 | 20.7 | 30.4 | 20.7 |
| TMS320VC5501ZAV300 | ZAV | NFBGA | 201 | 126 | 7 X 18 | 150 | 315 | 135.9 | 7620 | 17.2 | 11.3 | 16.35 |
| TMS320VC5501ZAV300.A | ZAV | NFBGA | 201 | 126 | 7 X 18 | 150 | 315 | 135.9 | 7620 | 17.2 | 11.3 | 16.35 |
| TNETV2501INPGF | PGF | LQFP | 176 | 40 | 4x10 | 150 | 315 | 135.9 | 7620 | 20.7 | 30.4 | 20.7 |
| VC5501LMPGF300 | PGF | LQFP | 176 | 40 | 4x10 | 150 | 315 | 135.9 | 7620 | 20.7 | 30.4 | 20.7 |

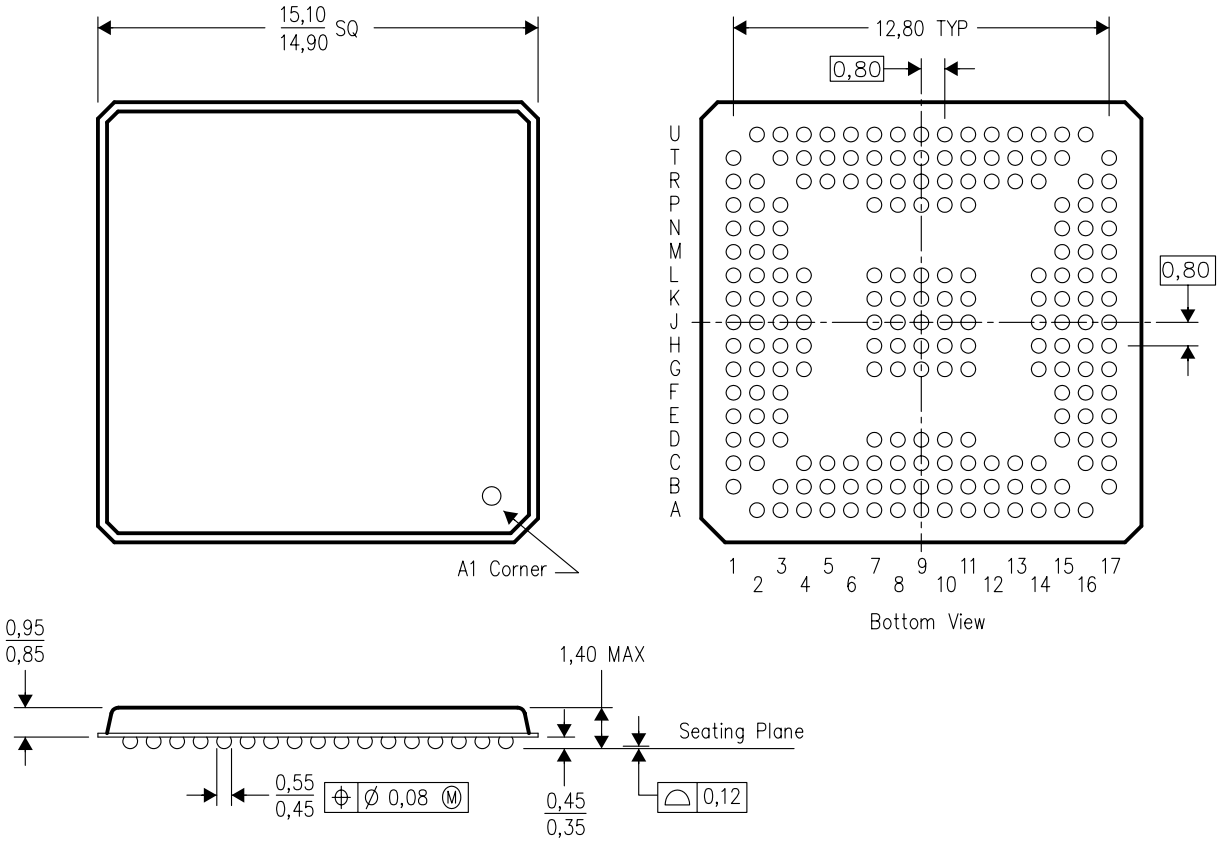
## GZZ (S–PBGA–N201) PLASTIC BALL GRID ARRAY




NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. MicroStar BGA™ configuration

MicroStar BGA is a trademark of Texas Instruments.

## ZZZ (S–PBGA–N201) PLASTIC BALL GRID ARRAY




4206500/A 10/04

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

# IMPORTANT NOTICE AND DISCLAIMER

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

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

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

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

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

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View TMS320VC5501GBE300 on WIN SOURCE
- Texas Instruments Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

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