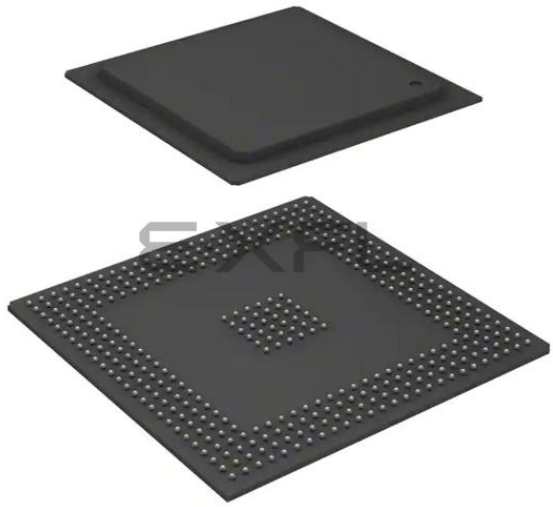

**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Not For New Designs |
| Core Processor | PowerPC |
| Core Size | 32-Bit Single-Core |
| Speed | 56MHz |
| Connectivity | CANbus, EBI/EMI, SCI, SPI, UART/USART |
| Peripherals | POR, PWM, WDT |
| Number of I/O | 56 |
| Program Memory Size | 512KB (512K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 32K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.5V ~ 2.7V |
| Data Converters | A/D 32x10b |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 388-BBGA |
| Supplier Device Package | 388-PBGA (27x27) |
| Purchase URL | https://www.e-xfl.com/product-detail/nxp-semiconductors/mpc563mzp56r2 |

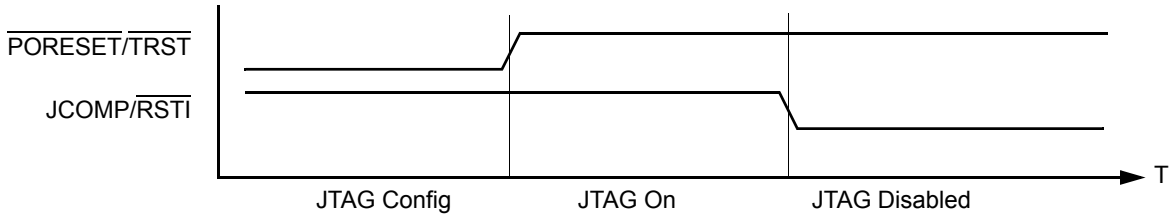


**Figure 2-4. Debug Mode Selection (JTAG)**

## 2.5.2 BDM Mode Selection

The MPC561/MPC563 has a 10 pin BDM port. See Figure 2-5 for BDM mode selection.

The BDM mode is entered by the following sequence of events:

- Hold DSCK high at reset negation (SRESET)
- Configure DSDI to select BDM clock mode, within 8 clocks of reset negation

BDM mode is exited by:

- Reset the device by asserting PORESET/TRST or HRESET

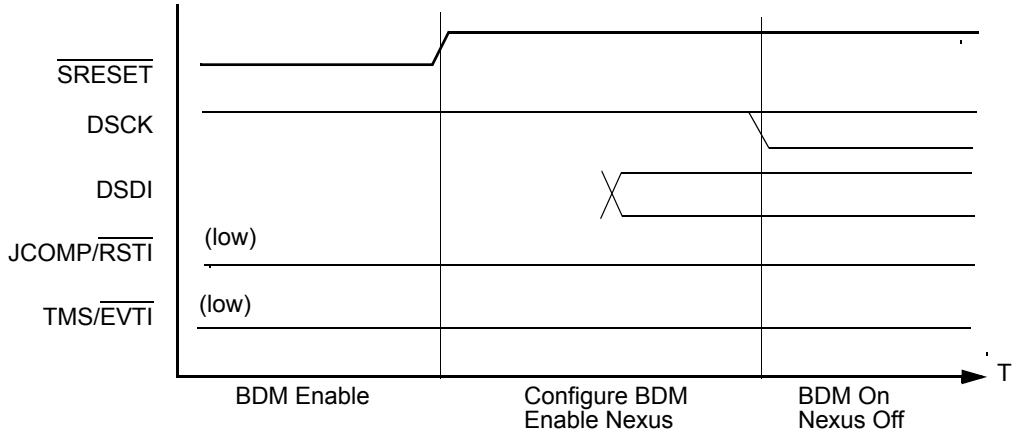


**Figure 2-5. Debug Mode Selection (BDM)**

## 2.5.3 Nexus Mode Selection

The READI module signals support the Nexus (IEEE-ISTO 5001-1999) auxiliary port interface for debug. There are two modes available: full port mode and reduced port mode. Reduced port mode allows for a 1 bit input stream and a 2 bit output stream. Full port mode allows for a 2-bit input stream and an 8-bit output stream. If MDI0 is held high when Nexus mode is enabled, full port mode will be used during Nexus debug. If MDI0 is held low when Nexus mode is enabled, reduced port mode will be used. See Figure 2-6 for Nexus mode selection.

The Nexus interface is entered by the following sequence of events:

- Hold JCOMP/RSTI low while negating PORESET/TRST
- Hold TMS/EVTI low to enable Nexus mode and configure TDI/DSDI/MDI0 for full or reduced port mode. Both of these should be done at least 4 clocks before driving JCOMP/RSTI high

# Chapter 3
# Central Processing Unit

The RISC processor (RCPU) used in the MPC500 family of microcontrollers integrates five independent execution units: an integer unit (IU), a load/store unit (LSU), a branch processing unit (BPU), a floating-point unit (FPU) and an integer multiplier divider (IMD). The RISC's use of simple instructions with rapid execution times yields high efficiency and throughput for PowerPC ISA-based systems.

Most integer instructions execute in one clock cycle. Instructions can complete out of order for increased performance; however, the processor makes execution appear sequential.

This section provides an overview of the RCPU. For a detailed description of this processor, refer to the *RCPU Reference Manual*. The following sections describe each block and sub-block.

## 3.1 RCPU Block Diagram

Figure 3-1 provides a block diagram of the RCPU.

**Table 3-5. FPSCR Bit Descriptions (continued)**

| Bits | Name | Description | |
|---|---|---|---|
| 29 | NI | Non-IEEE mode bit. | — |
| 30–31 | RN | Floating-point rounding control.<br>00 Round to nearest<br>01 Round toward zero<br>10 Round toward +infinity<br>11 Round toward -infinity | — |

Table 3-6 illustrates the floating-point result flags that correspond to FPSCR[15:19].

**Table 3-6. Floating-Point Result Flags in FPSCR**

| Result Flags (Bits 15:19) C<>=? | Result Value Class |
|---|---|
| 10001 | Quiet NaN |
| 01001 | – Infinity |
| 01000 | – Normalized number |
| 11000 | – Denormalized number |
| 10010 | – Zero |
| 00010 | + Zero |
| 10100 | + Denormalized number |
| 00100 | + Normalized number |
| 00101 | + Infinity |

## 3.7.4 Condition Register (CR)

The condition register (CR) is a 32-bit register that reflects the result of certain operations and provides a mechanism for testing and branching. The bits in the CR are grouped into eight 4-bit fields: CR0 to CR7.

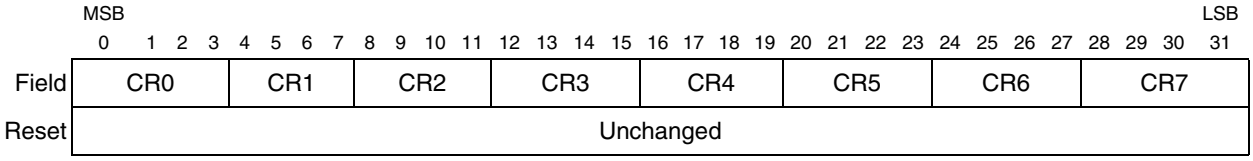


**Figure 3-7. Condition Register (CR)**

The CR fields can be set in the following ways:

- Specified fields of the CR can be set by an instruction (mtcrf) to move to the CR from a GPR.
- Specified fields of the CR can be moved from one CRx field to another with the mcrf instruction.
- A specified field of the CR can be set by an instruction (mcrxr) to move to the CR from the XER.
- Condition register logical instructions can be used to perform logical operations on specified bits in the condition register.
- CR0 can be the implicit result of an integer operation.

**Table 6-3. Priority of Interrupt Sources—Regular Operation**

| Number | Priority Level | Interrupt Source Description | Offset in Branch Table (Hex) | SIVEC Interrupt Code[1] |
|---|---|---|---|---|
| 8 | — | $\overline{\text{EXT\_IRQ4}}$ | 0x0040 | 00100000 |
| 9 | — | Level 4 | 0x0048 | 00100100 |
| 10 | — | $\overline{\text{EXT\_IRQ5}}$ | 0x0050 | 00101000 |
| 11 | — | Level 5 | 0x0058 | 00101100 |
| 12 | — | $\overline{\text{EXT\_IRQ6}}$ | 0x0060 | 00110000 |
| 13 | — | Level 6 | 0x0068 | 00110100 |
| 14 | — | $\overline{\text{EXT\_IRQ7}}$ | 0x0070 | 00111000 |
| 15 | Lowest | Level 7 | 0x0078 | 00111100 |

[1] This is the value in the 8 most significant bits of the SIVEC register (SIVEC[25:31]).

Each interrupt request from external lines and from USIU internal interrupt sources in the case of its assertion will set a corresponding bit in SIPEND register. The individual SIPEND bits may be masked by clearing an appropriate bit in SIMASK register.

### 6.1.4.4 Enhanced Interrupt Controller Operation

The enhanced interrupt controller operation may be turned on by setting the EICEN control bit in the SIUMCR register. In this mode the 32 IMB interrupt levels will be latched by USIU using eight IMB interrupt lines and two lines of ilbs via the time multiplexing scheme defined by the UIMB module. In addition to the IMB interrupt sources the external interrupts and timer interrupts are available in the same way as in the regular scheme. In this mode, the UIMB module does not drive U-bus interrupt level lines. Each interrupt request will set a corresponding bit in SIPEND2 or SIPEND3 registers. SIPEND2 an SIPEND3 may be masked by clearing an appropriate bit in SIMASK2 or SIMASK3 registers.

The priority logic is provided in order to determine the highest unmasked interrupt request, and interrupt code is generated in the SIVEC register. See Table 6-4.

**NOTE**

If the enhanced interrupt controller is enabled, a delay is required prior to re-enabling interrupts. Before clearing an interrupt related register, clear the MSR[EE] bit (EE = 0). Expect a vector offset of 0x0 if an interrupt is cleared or disabled while MSR[EE] = 1. This vector should be handled as if no interrupt has occured, that is, perform an rfi instruction. After clearing an interrupt source, sufficient time must elapse before re-enabling the MSR[EE] bit (EE = 1). This time should take longer than the time needed for a load of the same register that was just cleared. To guarantee enough time, include this load instruction before the instruction that sets MSR[EE].

**Table 6-13. EMCR Bit Descriptions (continued)**

| Bits | Name | Description |
|---|---|---|
| 26 | CONT | Control attribute. CONT drives the internal bus control bit attribute as follows:<br>0 Access to MPC561/MPC563 control register, or control cycle access<br>1 Access to global address map |
| 27 | — | Reserved |
| 28 | TRAC | Trace attribute. TRAC controls the internal bus program trace attribute as follows:<br>0 Program trace<br>1 Not program trace |
| 29 | SIZEN | External size enable control bit. SIZEN determines how the internal bus size attribute is driven:<br>0 Drive size from external bus signals TSIZE[0:1]<br>1 Drive size from SIZE0, SIZE1 in EMCR |
| 30:31 | — | Reserved |

### 6.2.2.2 SIU Interrupt Controller Registers

The SIU interrupt controller contains the following registers: SIPEND, SIPEND2 and SIPEND3 (interrupt pending registers), SIMASK, SIMASK2 and SIMASK3 (interrupt mask registers), SIEL, SIVEC, SISR2 and SISR3.

The SIPEND and SIMASK registers are used when the interrupt controller is configured for regular, MPC555/MPC556 compatible, operation. SIPEND2, SIPEND3, SIMASK2, SIMASK3, SISR2 and SISR3 registers are used only when the interrupt controller is operating in enhanced interrupt mode.

SIPEND, SIPEND2 and SIPEND3 are 32-bit registers. Each bit in the register corresponds to an interrupt request. The bits associated with internal exceptions indicate, if set, that an interrupt service is requested. These bits reflect the status of the internal requesting device, and will be cleared when the appropriate actions are initiated by software in the device itself. Writing to these bits has no effect.

The bits associated with the $\overline{\text{IRQ}}$ pins have a different behavior depending on the sensitivity defined for them in the SIEL register. When the $\overline{\text{IRQ}}$ is defined as a “level” interrupt the corresponding bit behaves in a manner similar to the bits associated with internal interrupt sources, (i.e., it reflects the status of the $\overline{\text{IRQ}}$ pin). This bit can not be changed by software, it will be cleared when the external signal is negated. When the $\overline{\text{IRQ}}$ is defined as an “edge” interrupt, if the corresponding bit is set, it indicates that a falling edge was detected on the line. The bit must be reset by software by writing a ‘1’ to it.

The following acronym definitions apply to the various bits implemented in the SIU interrupt controller registers.

**Table 6-14. SIU Interrupt Controller – Bit Acronym Definitions**

| Name | Description |
|---|---|
| IRQ*n* | Interrupt Signal *n* Request |
| LVL*n* | Interrupt Level *n* Request |
| IMBIRQ*n* | Intermodule Bus Interrupt Level *n* Request |
| IRM*n* | Interrupt Signal *n* Mask |

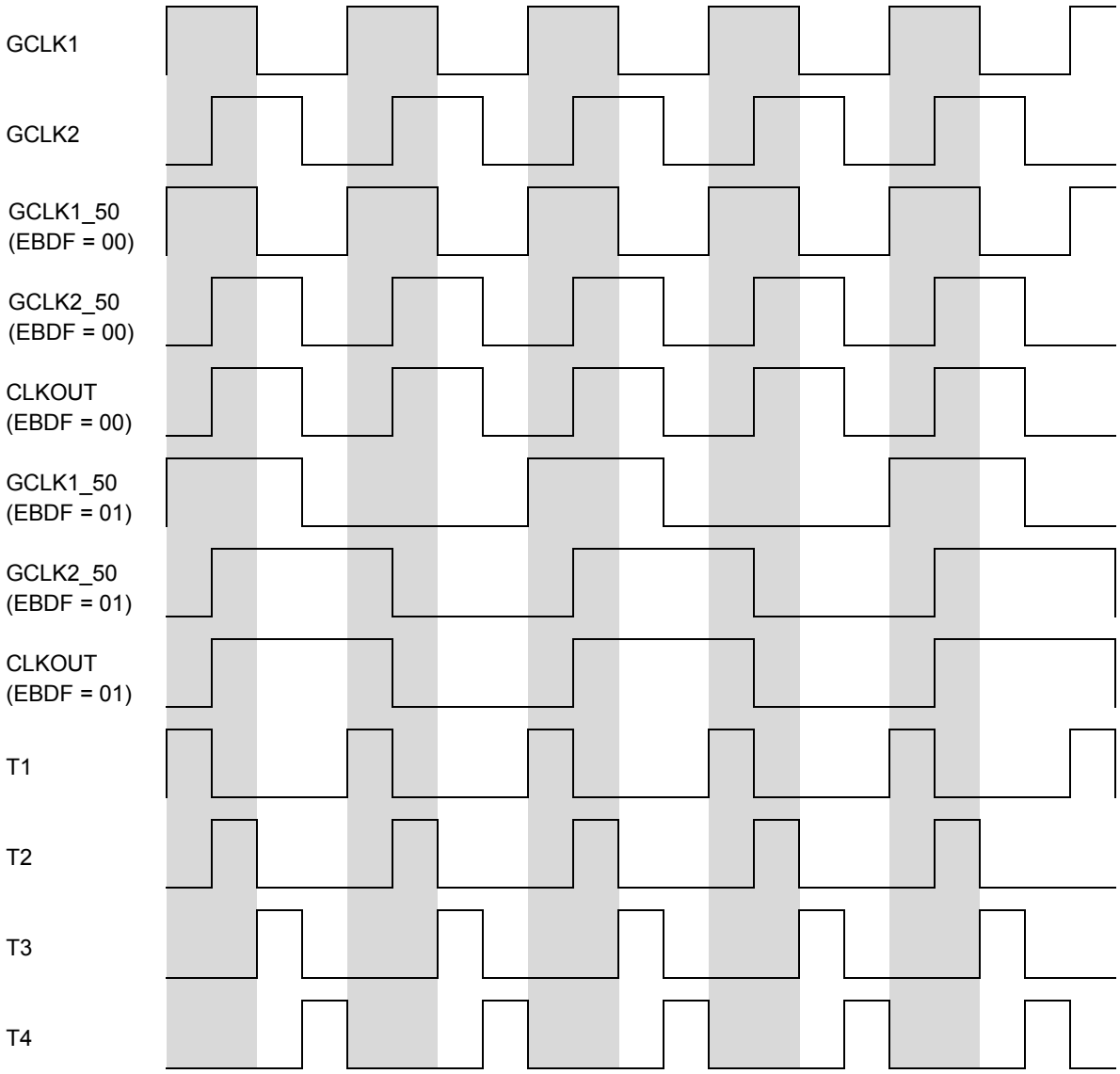


**Figure 8-4. MPC561/MPC563 Clocks**

Note that GCLK1_50, GCLK2_50, and CLKOUT can have a lower frequency than GCLK1 and GCLK2. This is to enable the external bus operation at lower frequencies (controlled by EBDF in the SCCR). GCLK2_50 always rises simultaneously with GCLK2. When DFNH = 0, GCLK2_50 has a 50% duty cycle. With other values of DFNH or DFNL, the duty cycle is less than 50%. Refer to Figure 8-7. GCLK1_50 rises simultaneously with GCLK1. When the MPC561/MPC563 is not in gear mode, the falling edge of GCLK1_50 occurs in the middle of the high phase of GCLK2_50. EBDF determines the division factor between GCLK1/GCLK2 and GCLK1_50/GCLK2_50.

During power-on reset, the MODCK1, MODCK2, and MODCK3 pins determine the clock source for the PLL and the clock drivers. These pins are latched on the positive edge of $\overline{\text{PORESET}}$. Their values must be stable as long as this line is asserted. The configuration modes are shown in Table 8-1. MODCK1 specifies

then controls the length of the cycle with the signal(s) used to terminate the cycle. A strobe signal for the address lines indicates the validity of the address.

The MPC561/MPC563 bus is synchronous with a synchronous support. The bus and control input signals must be timed to setup and hold times relative to the rising edge of the clock. Bus cycles can be completed in two clock cycles.

For all inputs, the MPC561/MPC563 latches the level of the input during a sample window around the rising edge of the clock signal. This window is illustrated in Figure 9-1, where $t_{su}$ and $t_{ho}$ are the input setup and hold times, respectively. To ensure that an input signal is recognized on a specific rising edge of the clock, that input must be stable during the sample window. If an input makes a transition during the window time period, the level recognized by the MPC561/MPC563 is not predictable; however, the MPC561/MPC563 always resolves the latched level to either a logic high or low before using it. In addition to meeting input setup and hold times for deterministic operation, all input signals must obey the protocols described in this section.

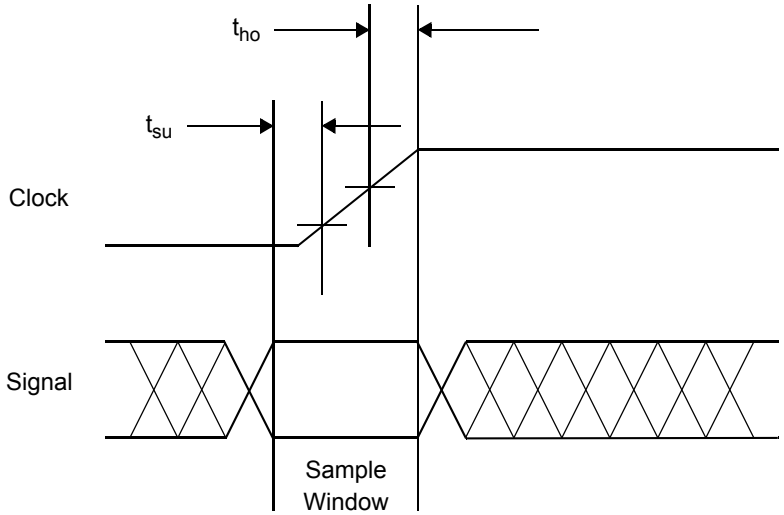


**Figure 9-1. Input Sample Window**

## 9.3 Bus Control Signals

The MPC561/MPC563 initiates a bus cycle by driving the address, size, address type, cycle type, and read/write outputs. At the beginning of a bus cycle, TSIZ[0:1] are driven with the address type signals. TSIZ0 and TSIZ1 indicate the number of bytes remaining to be transferred during an operand cycle (consisting of one or more bus cycles). These signals are valid at the rising edge of the clock in which the transfer start ($\overline{TS}$) signal is asserted.

The read/write (RD/$\overline{WR}$) signal determines the direction of the transfer during a bus cycle. Driven at the beginning of a bus cycle, RD/$\overline{WR}$ is valid at the rising edge of the clock in which $\overline{TS}$ is asserted. The logic level of RD/$\overline{WR}$ only changes when a write cycle is preceded by a read cycle or vice versa. The signal may remain low for consecutive write cycles.

[1] Cases in which both $\overline{TS}$ and $\overline{STS}$ are asserted indicate normal cycles with the show cycle attribute.

#### 9.5.8.7 Burst Data in Progress

This signal is sent from the master to the slave to indicate that there is a data beat following the current data beat. The master uses this signal to give the slave advance warning of the remaining data in the burst. $\overline{BDIP}$ can also be used to terminate the burst cycle early. Refer to Section 9.5.4, "Burst Transfer" and Section 9.5.5, "Burst Mechanism" for more information. Refer to Section 10.9.3, "Memory Controller Base Registers (BR0–BR3)" for memory controller BDIP options.

### 9.5.9 Termination Signals

The EBI uses three termination signals:

- Transfer acknowledge ($\overline{TA}$)
- Burst inhibit ($\overline{BI}$)
- Transfer error acknowledge ($\overline{TEA}$)

#### 9.5.9.1 Transfer Acknowledge

Transfer acknowledge ($\overline{TA}$) indicates normal completion of the bus transfer. During a burst cycle, the slave asserts this signal with every data beat returned or accepted.

#### 9.5.9.2 Burst Inhibit

A slave sends the $\overline{BI}$ signal to the master to indicate that the addressed device does not have burst capability. If this signal is asserted, the master must transfer in multiple cycles and increment the address for the slave to complete the burst transfer. For a system that does not use the burst mode at all, this signal can be tied low permanently. Refer to Section 10.9.3, "Memory Controller Base Registers (BR0–BR3)" for $\overline{BI}$ options.

#### 9.5.9.3 Transfer Error Acknowledge

The $\overline{TEA}$ signal terminates a bus cycle under one or more bus error conditions. The current bus cycle must be aborted. This signal overrides any other cycle termination signals, such as transfer acknowledge.

#### 9.5.9.4 Termination Signals Protocol

The transfer protocol was defined to avoid electrical contention on lines that can be driven by various sources. To this end, a slave must not drive signals associated with the data transfer until the address phase is completed and it recognizes the address as its own. The slave must disconnect from signals immediately after it has acknowledged the cycle and no later than the termination of the next address phase cycle. This means that the termination signals must be connected to power through a pull-up resistor to avoid the situation in which a master samples an undefined value in any of these signals when no real slave is addressed.

Refer to Figure 9-28 and Figure 9-29.

- The L2U does not provide show cycle for any L-bus addresses that fall in the L-bus CALRAM address space if the CALRAM protection [SP] bit is set in the L2U_MCR.

Table 11-4 summarizes the L2U show cycle support.

**Table 11-4. L2U Show Cycle Support Chart**

| Case | Destination | LB AACK | LB ABORT | Comments |
|---|---|---|---|---|
| 1 | L-bus Slave[1] | No | X[2] | Not show cycled<br>[Cycle will be retried one clock later][3] |
| 2 | L2U[4] | X | X | Not show cycled |
| 3 | U-bus/E-bus[5] | X | X | Not show cycled |
| 4 | L-bus slave[1] | Yes | No | Show cycled |
| 5 | L-bus slave[1] | Yes | Yes | Not show cycled<br>[L-bus will be released next clock] |

1 L-bus slave includes all address in the L-bus address space.

2 X indicates don't care conditions.

3 There will be a 1-clock turnaround because the L-bus retry information is not available in time to negate the L-bus arbitration.

4 L2U indicates L2U registers.

5 U-bus/E-bus refers to all destinations through the L2U interface.

## 11.8 L2U Programming Model

The L2U control registers control the L2U bus interface and the DMPU. They are accessible via the mtspr and mfspr instructions. They are also accessible by an external master when EMCR[CONT] bit is cleared. L2U control registers are accessible from both the L-bus side and the U-bus side in one clock cycle. As with all SPRs, L2U registers are accessible in supervisor mode only.

Any unimplemented bits in L2U registers return 0's on a read, and the writes to those register bits are ignored.

Table 11-5 shows L2U registers along with their SPR numbers and hexadecimal addresses that are used to access L2U registers during a peripheral mode access.

**Table 11-5. L2U (PPC) Register Decode**

| Name | SPR # | SPR[5:9] | SPR[0:4] | Address for External Master Access[1] | Access | Description |
|---|---|---|---|---|---|---|
| L2U_MCR | 568 | 10001 | 11000 | 0x0000_3110 | SUPR | L2U Module Configuration Register |
| L2U_RBA0 | 792 | 11000 | 11000 | 0x0000_3180 | SUPR | Region Base Address Register 0 |
| L2U_RBA1 | 793 | 11000 | 11001 | 0x0000_3380 | SUPR | Region Base Address Register 1 |
| L2U_RBA2 | 794 | 11000 | 11010 | 0x0000_3580 | SUPR | Region Base Address Register 2 |
| L2U_RBA3 | 795 | 11000 | 11011 | 0x0000_3780 | SUPR | Region Base Address Register 3 |
| L2U_RA0 | 824 | 11001 | 11000 | 0x0000_3190 | SUPR | Region Attribute Register 0 |
| L2U_RA1 | 825 | 11001 | 11001 | 0x0000_3390 | SUPR | Region Attribute Register 1 |

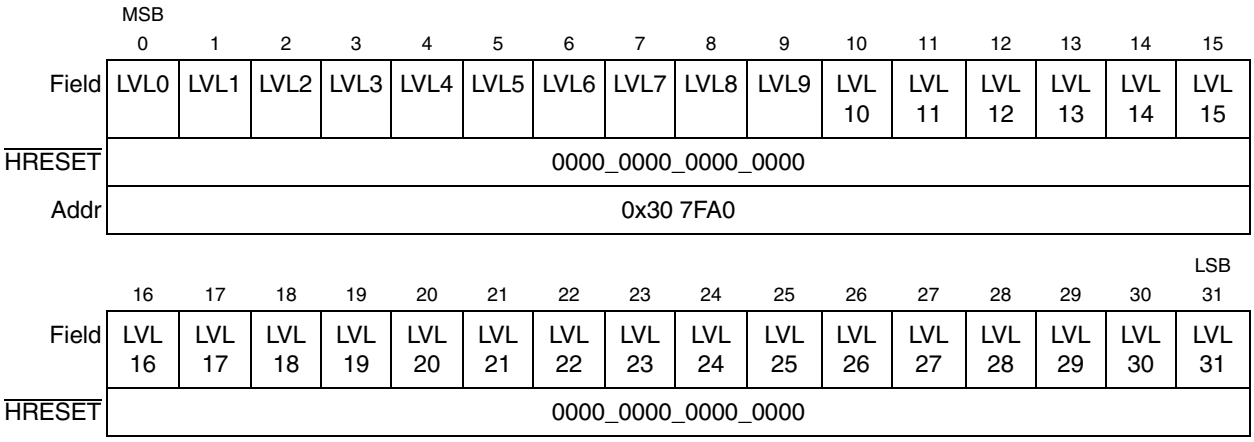


**Figure 12-8. Pending Interrupt Request Register (UIPEND)**

**Table 12-7. UIPEND Bit Descriptions**

| Bits | Name | Description |
|---|---|---|
| 0:31 | LVL*x* | Pending interrupt request level. Accessible only in supervisor mode. LVL*x* identifies the interrupt source as UIMB LVL*x*, where *x* is the interrupt number. |

**Table 17-29. MPWMSCR Bit Descriptions**

| Bits | Name | Description |
|---|---|---|
| 0 | PIN | Pin input status bit — The PIN bit reflects the state present on the MPWMSM signal. The software can thus monitor the pin state.<br>The PIN bit is a read-only bit. Writing to the PIN bit has no effect. |
| 1 | DDR | Data direction register — The DDR bit indicates the direction for the signal when the PWM function is not used (disable mode).<br>0 signal is in input.<br>1 signal is in output.<br>The DDR bit is cleared by reset.<br>Table 17-30 lists the different uses for the polarity (POL) bit, the enable (EN) bit and the data direction register (DDR) bit. |
| 2 | FREN | Freeze enable bit — This active high read/write control bit enables the MPWMSM to recognize the freeze signal on the MIOB.<br>0 MPWMSM not frozen even if the MIOB freeze line is active.<br>1 MPWMSM frozen if the MIOB freeze line is active.<br>The FREN is cleared by reset. |
| 3 | TRSP | Transparent mode — The TRSP bit indicates that the MPWMSM is in transparent mode. In transparent mode, when the software writes to either the MPWMPERR or MPWMPULR1 register the value written is immediately transferred to the counter or register MPWMPULR2 respectively.<br>0 Double-buffered mode.<br>1 Transparent mode.<br>The TRSP bit is cleared by reset. |
| 4 | POL | Output polarity control bit — The POL bit works in conjunction with the EN bit and controls whether the MPWMSM drives the signal with the direct or the inverted value of the output flip-flop.<br>Table 17-30 lists the different uses for the polarity (POL) bit, the enable (EN) bit and the data direction register (DDR) bit. |
| 5 | EN | Enable PWM signal generation — The EN bit defines whether the MPWMSM generates a PWM signal or is used as an I/O channel:<br>0 PWM generation disabled (signal can be used as I/O).<br>1 PWM generation enabled (the signal is in output mode).<br>Each time the submodule is enabled, the value of CP is loaded into the prescaler.<br>The EN bit is cleared by reset. |
| 6:7 | — | Reserved |
| 8:15 | CP | Clock prescaler — This 8-bit read/write data register stores the modulus value for loading into the built-in 8-bit clock prescaler. The value loaded defines the divide ratio for the signal that clocks the MPWMSM. The new value is loaded into the prescaler counter on the prescaler counter overflow, or upon the EN bit of the MPWMSCR being set.<br>Table 17-31 gives the clock divide ratio according to the value of CP. |

**Table 17-30. PWMSM Output Signal Polarity Selection**

| Control Bits | | | Signal Direction | Signal State | Periodic Edge | Variable Edge | Optional Interruption |
|---|---|---|---|---|---|---|---|
| POL | EN | DDR | | | | | |
| 0 | 0 | 0 | Input | INPUT | — | — | — |
| 0 | 0 | 1 | Output | Always Low | — | — | — |
| 0 | 1 | X | Output | High Pulse | Falling Edge | Rising Edge | Falling Edge |
| 1 | 0 | 0 | Input | INPUT | — | — | — |

**Table 22-2. CALRAM Control Registers**

| Address | Register |
|---|---|
| 0x38 0000 | CRAMMCR |
| 0x38 0004 | for factory test |
| 0x38 0008 | CRAM_RBA0 |
| 0x38 000C | CRAM_RBA1 |
| 0x38 0010 | CRAM_RBA2 |
| 0x38 0014 | CRAM_RBA3 |
| 0x38 0018 | CRAM_RBA4 |
| 0x38 001C | CRAM_RBA5 |
| 0x38 0020 | CRAM_RBA6 |
| 0x38 0024 | CRAM_RBA7 |
| 0x38 0028 | CRAMOVLCR |
| 0x38 002C | CRAMOTR/READI_OTR |
| 0x38 0030 | Reserved |
| 0x38 0034 | Reserved |
| 0x38 0038 | Reserved |
| 0x38 003C | Reserved |

Any unimplemented bits in CALRAM registers return 0's on a read and writes to these bits are ignored.

## 22.5.1 CALRAM Module Configuration Register (CRAMMCR)

The module configuration register (CRAMMCR) contains bits that allow the CALRAM to be configured for normal RAM accesses.

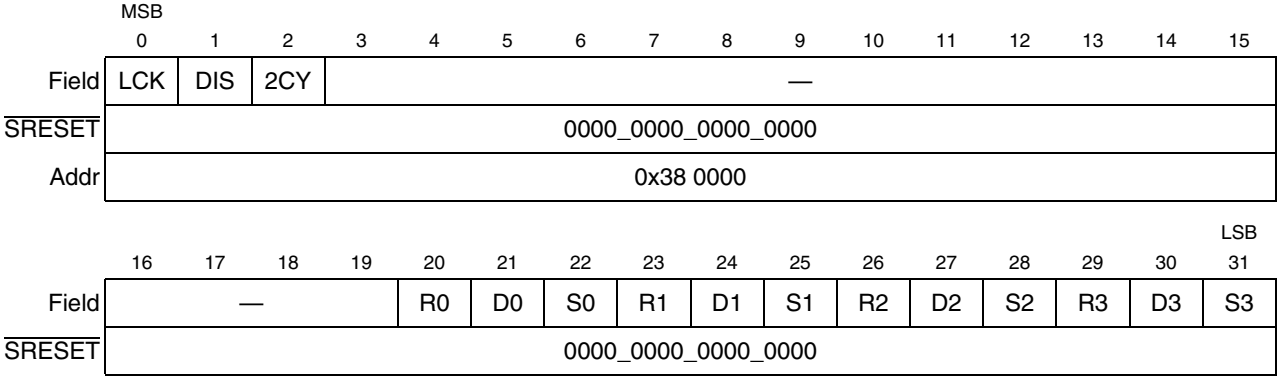


**Figure 22-9. CALRAM Module Configuration Register (CRAMMCR)**

A brief description of each bit is provided in Table 22-3

**Table 22-3. CRAMMCR Bit Descriptions (continued)**

| Bits | Name | Description |
|---|---|---|
| 22 | S0 | Supervisor-only/supervisor-user privilege (Space assignment) — If the data relocate (DR) bit is set in Machine Status Register (MSR) and S0 is also set, then any access to the array block by a user program generates an error. If DR bit is 0, both user and supervisor program can access the array block, regardless of the value programmed in S0. The CALRAM array may be placed in supervisor or unrestricted space.<br>This bit controls the highest 8-Kbyte block (lowest address) of CALRAM in the associated array. Likewise, S1, S2, and S3 control other three blocks in the same manner. See Table 22-4 for control bit address ranges.<br>S0 = 0 and DR = 0 both user and supervisor access allowed (array 8-Kbyte block)<br>S0 = 0 and DR = 1 both user and supervisor access allowed (array 8-Kbyte block)<br>S0 = 1 and DR = 0 both user and supervisor access allowed (array 8-Kbyte block)<br>S0 = 1 and DR = 1 only supervisor access allowed (array 8-Kbyte block) |
| 23 | R1 | Same as R0 except for address ranges shown on Table 22-4. |
| 24 | D1 | Same as D0 except for address ranges shown on Table 22-4. |
| 25 | S1 | Same as S0 except for address ranges shown on Table 22-4. |
| 26 | R2 | Same as R0 except for address ranges shown on Table 22-4. |
| 27 | D2 | Same as D0 except for address ranges shown on Table 22-4. |
| 28 | S2 | Same as S0 except for address ranges shown on Table 22-4. |
| 29 | R3 | Same as R0 except for address ranges shown on Table 22-4. |
| 30 | D3 | Same as D0 except for address ranges shown on Table 22-4. |
| 31 | S3 | Same as S0 except for address ranges shown on Table 22-4. |

**Table 22-4. CRAMMCR Privilege Bit Assignment for 8-Kbyte Array Blocks**

| Bit Selection | Address Block (Relative) |
|---|---|
| R0, D0, and S0 | 0xXXXX 0000 – 0xXXXX 1FFF |
| R1, D1, and S1 | 0xXXXX 2000 – 0xXXXX 3FFF |
| R2, D2, and S2 | 0xXXXX 4000 – 0xXXXX 5FFF |
| R3, D3, and S3 | 0xXXXX 6000 – 0xXXXX 7FFF |

## 22.5.2 CALRAM Region Base Address Registers (CRAM_RBAx)

The region base address register defines the base address of a region on the U-bus Flash memory space that will be overlaid by a portion of the CALRAM memory space and the region size. Because eight such regions in the Flash can be overlaid by the CALRAM, eight such registers (x = 0, 1, 2, 7) are provided.

The CRAM_RBAx[11:29] provides the base address (starting address) of the of the U-bus Flash region to be overlaid and the CRAM_RBAx[0:3] provides size corresponding to the region. See Table 22-6 for details. The RGN_SIZE[0] is reserved and should never be programmed to a one, because the MPC563 has only 512 Kbytes of Flash, and CRAM_RBAx[11] and CRAM_RBAx[12] should never be programmed to a one. Also, note that if CRAM_OVLCR[CLPS] is set, each of the eight sizes are forced

to be four bytes, regardless of the value programmed in the RGN_SIZE[0:3] field. See Section 22.5.3, “CALRAM Overlay Configuration Register (CRAM_OVLCR)” for details.

The implemented bits of CRAM_RBAx bits are unaffected by reset (hard reset). The diagram below shows one such register, CRAM_RBA0, which provides the base address of overlay region 0.

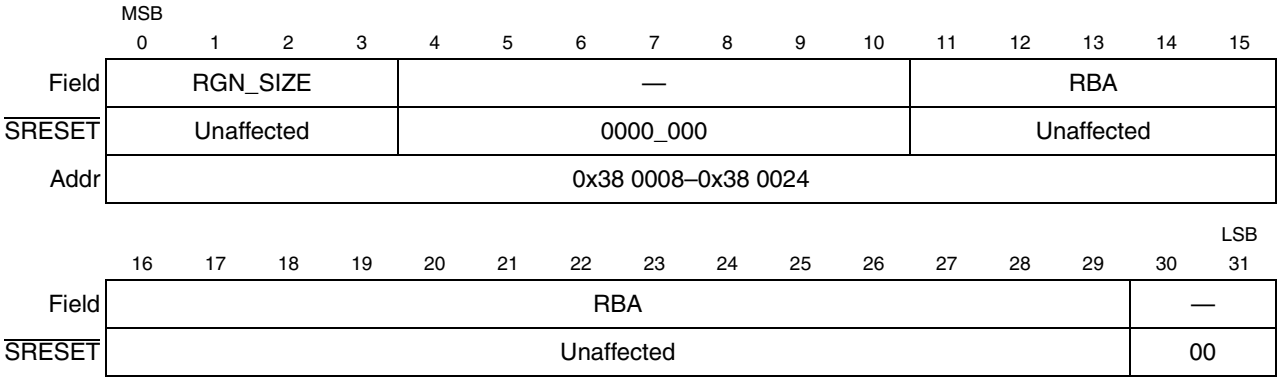


**Figure 22-10. CALRAM Region Base Address Register (CRAM_RBAx)**

**Table 22-5. CRAM_RBAx Bit Descriptions**

| Bits | Name | Description |
|---|---|---|
| 0:3 | RGN_SIZE | These bits define the size of the overlay region. See Table 22-6 for sizes. |
| 4:10 | — | Reserved |
| 11:29 | RBA | The region base address defines the starting address of the memory to be overlayed.[1] |
| 30:31 | — | Reserved |

[1] The overlay match address will include ISB in its comparison. The overlay can only be in the range of the ISB internal space.

**Table 22-6. RGN_SIZE Encoding**

| RGN_SIZE | Number of Overlay Bytes |
|---|---|
| 0000 | Overlay block disabled |
| 0001 | Overlay block is 4 bytes |
| 0010 | Overlay block is 16 bytes |
| 0011 | Overlay block is 32 bytes |
| 0100 | Overlay block is 64 bytes |
| 0101 | Overlay block is 128 bytes |
| 0110 | Overlay block is 256 bytes |
| 0111 | Overlay block is 512 bytes |
| 1xxx | Reserved |

**Note:** The overlay size of 8 bytes cannot be selected

The comparators generate match events. The match events enter the instruction AND-OR logic where the instruction watchpoints and breakpoint are generated. The instruction watchpoints, when asserted, may generate the instruction breakpoint. Two of them may decrement one of the counters. If one of the instruction watchpoints expires in a counter that is counting, the instruction breakpoint is asserted.

The instruction watchpoints and the load/store match events (address and data) enter the load/store AND-OR logic where the load/store watchpoints and breakpoint are generated. The load/store watchpoints, when asserted, may generate the load/store breakpoint or they may decrement one of the counters. When a counter that is counting one of the load/store watchpoints expires, the load/store breakpoint is asserted.

Watchpoints progress in the machine and are reported on retirement. Internal breakpoints progress in the machine until they reach the top of the history buffer when the machine branches to the breakpoint exception routine.

In order to enable the use of the breakpoint features without adding restrictions on the software, the address of the load/store cycle that generated the load/store breakpoint is not stored in the DAR (data address register), like other load/store type exceptions. In case of a load/store breakpoint, the address of the load/store cycle that generated the breakpoint is stored in an implementation-dependent register called the BAR (breakpoint address register).

Key features of internal watchpoint and breakpoint support are:

- Four I-address comparators (each supports equal, not equal, greater than, less than)
- Two L-address comparators (each supports equal, not equal, greater than, less than) including least significant bits masking according to the size of the bus cycle for the byte and half-word working modes. Refer to Section 23.2.1.2, “Byte and Half-Word Working Modes.”
- Two L-data comparators (each supports equal, not equal, greater than, less than) including byte, half-word and word operating modes and four byte mask bits for each comparator. Can be used for fix point data. Match is detected only on the valid part of the data bus (according to the cycle’s size and the two address least significant bits).
- No internal breakpoint/watchpoint matching support for unaligned words and half-words
- The L-data comparators can be programmed to treat fix point numbers as signed values or as unsigned values
- Combine comparator pairs to detect in and out of range conditions (including either signed or unsigned values on the L-data)
- A programmable AND-OR logic structure between the four instruction comparators results with five outputs, four instruction watchpoints and one instruction breakpoint
- A programmable AND-OR logic structure between the four instruction watchpoints and the four load/store comparators results with three outputs, two load/store watchpoints and one load/store breakpoint
- Five watchpoint pins, three for the instruction and two for the load/store
- Two dedicated 16-bit down counters. Each can be programmed to count either an instruction watchpoint or an load/store watchpoint. Only architecturally executed events are counted, (count up is performed in case of recovery).

**Table B-5. USIU (Unified System Interface Unit) (continued)**

| Address | Access | Symbol | Register | Size | Reset |
|---|---|---|---|---|---|
| 0x2F C010 | U | SIPEND | Interrupt Pending Register<br>See Section 6.2.2.2.1 for bit descriptions. | 32 | S |
| 0x2F C014 | U | SIMASK | Interrupt Mask Register<br>SIMASK is a 32-bit read/write register. Each bit in the register corresponds to an interrupt request bit in the SIPEND register. | 32 | S |
| 0x2F C018 | U | SIEL | Interrupt Edge Level Mask.<br>See Section 6.2.2.2.7 for bit descriptions. | 32 | H |
| 0x2F C01C | U,<br>read only | SIVEC | Interrupt Vector.<br>See Section 6.2.2.2.8 for bit descriptions. | 32 | — |
| 0x2F C020 | U | TESR | Transfer Error Status Register<br>See Table 6-17 for bit descriptions. | 32 | S |
| 0x2F C024 | U | SGPIODT1 | USIU General-Purpose I/O Data Register 1<br>See Table 6-23 for bit descriptions. | 32 | H |
| 0x2F C028 | U | SGPIODT2 | USIU General-Purpose I/O Data Register 2<br>See Table 6-24 for bit descriptions. | 32 | H |
| 0x2F C02C | U | SGPIOCR | USIU General-Purpose I/O Control Register<br>See Table 6-25 for bit descriptions. | 32 | H |
| 0x2F C030 | U | EMCR | External Master Mode Control Register<br>See Table 6-13 for bit descriptions. | 32 | H |
| 0x2F C038 | U | PDMCR2 | Pads Module Configuration Register 2<br>See Table 2-6 for bit descriptions. | 32 | H |
| 0x2F C03C | U | PDMCR | Pads Module Configuration Register<br>See Table 2-5 for bit descriptions. | 32 | H |
| 0x2F C040 — 0x2F C044 | U | SIPEND2 — SIPEND3 | Interrupt Pending Registers 2 and 3<br>See Section 6.2.2.2.1 for bit descriptions. | 32 | S |
| 0x2F C048 — 0x2F C04C | U | SIMASK2 — SIMASK3 | Interrupt Mask Register and Interrupt Mask Registers 2 and 3<br>See Section 6.2.2.2.9 for bit descriptions. | 32 | S |
| 0x2F C050 — 0x2F C054 | U | SISR2 — SISR3 | SISR2 and SISR3 Registers<br>See Section 6.2.2.2.9 for bit descriptions. | 32 | S |
| 0x2F C0FC — 0x2F C0FF | — | — | Reserved | — | — |
| **Memory Controller Registers** | | | | | |
| 0x2F C100 | U | BR0 | Base Register 0.<br>See Table 10-8 for bit descriptions. | 32 | H |
| 0x2F C104 | U | OR0 | Option Register 0.<br>See Table 10-10 for bit descriptions. | 32 | H |
| 0x2F C108 | U | BR1 | Base Register 1.<br>See Table 10-8 for bit descriptions. | 32 | H |

MF is the multiplication factor in the PLPRCR register (refer to Section 8.11.2, “PLL, Low-Power, and Reset-Control Register (PLPRCR)” for more information).

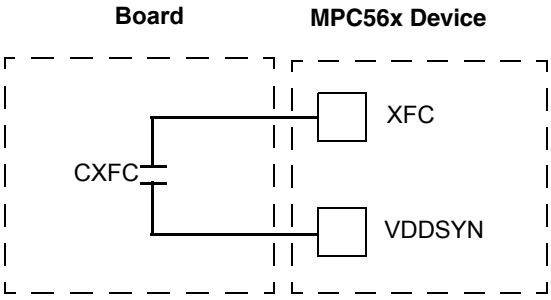


**Figure C-8. PLL Off-Chip Capacitor Example**

## C.3 PLL and Clock Oscillator External Components Layout Requirements

### C.3.1 Traces and Placement

Traces connecting capacitors, crystal, resistor should be as short as possible. Therefore, the components (crystal, resistor and capacitors) should be placed as close to the oscillator pins of the MPC561/MPC563 as possible.

The voltage to the VDDSYN pin should be well regulated and the pin should be provided with an extremely low impedance path from the VDDSYN filter to the VDDSYN pad.

The VSSSYN pin should be provided with an extremely low impedance path in the board. All the filters for the supplies should be located as close as possible to the chip package. It is recommended to design individual VSSSYN plane to improve VSSSYN quietness.

### C.3.2 Grounding/Guarding

The traces from the oscillator pins and PLL pins of the MPC561/MPC563 should be guarded from all other traces to reduce crosstalk. It can be provided by keeping other traces away from the oscillator circuit and placing a ground plane around the components and traces.

### C.3.3 IRAMSTBY Regulator Circuit

IRAMSTBY is the data retention power supply for all on-board RAM arrays (CALRAM, DPTRAM, DECRAM). It has a shunt regulator circuit to divert excess current to ground in order to regulate voltage on the IRAMSTBY power supply pin. IRAMSTBY should be connected to a positive power supply, via a resistor, and bypassed by a capacitor to ground as shown in Figure C-9.

**Table F-10. Bus Operation Timing (continued)**

**Note:** ($V_{DD}$ = 2.6 V ± 0.1 V, $V_{DDH}$ = 5.0 V ± 0.25 V, $T_A = T_L$ to $T_H$, 50 pF load unless noted otherwise)

| | Characteristic | 40 MHz | | 56 MHz[1] | | Unit |
|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | |
| 26b | $\overline{CS}$ negated to D[0:31], High Z -GPCM- write access, ACS = '00', TRLX = '0' & CSNT = '0' | 3 | | 2.25 | | ns |
| 26c | $\overline{CS}$ negated to D[0:31], High Z -GPCM- write access, TRLX = '0', CSNT = '1', ACS = '10' or ACS='11', EBDF = 0 | 8 | | 5.71 | | ns |
| 26d | $\overline{WE}$[0:3]/$\overline{BE}$[0:3] negated to D[0:31] High Z -GPCM- write access, TRLX = '1', CSNT = '1', EBDF = 0 | 28 | | 20 | | ns |
| 26e | $\overline{CS}$ negated to D[0:31] High Z -GPCM- write access, TRLX = '1', CSNT = '1', ACS = '10' or ACS='11', EBDF = 0 | 28 | | 20 | | ns |
| 26f | $\overline{WE}$[0:3]/$\overline{BE}$[0:3] negated to D[0:31] HighZ -GPCM- write access, TRLX = '0', CSNT = '1', EBDF = 1 | 5 | | 3.75 | | ns |
| 26g | $\overline{CS}$ negated to D[0:31] High Z -GPCM- write access, TRLX = '0', CSNT = '1', ACS = '10' or ACS='11', EBDF = 1 | 5 | | 3.75 | | ns |
| 26h | $\overline{WE}$[0:3]/$\overline{BE}$[0:3] negated to D[0:31] High Z -GPCM- write access, TRLX = '1', CSNT = '1', EBDF = 1 | 24 | | 17.25 | | ns |
| 26i | $\overline{CS}$ negated to D[0:31] High Z -GPCM- write access, TRLX = '1', CSNT = '1', ACS = '10' or ACS='11', EBDF = 1 | 24 | | 17.25 | | ns |
| 27 | $\overline{CS}$, $\overline{WE}$[0:3]/$\overline{BE}$[0:3] negated to ADDR[8:31] invalid -GPCM- write access[5] | 0.75 | | 1 | | ns |

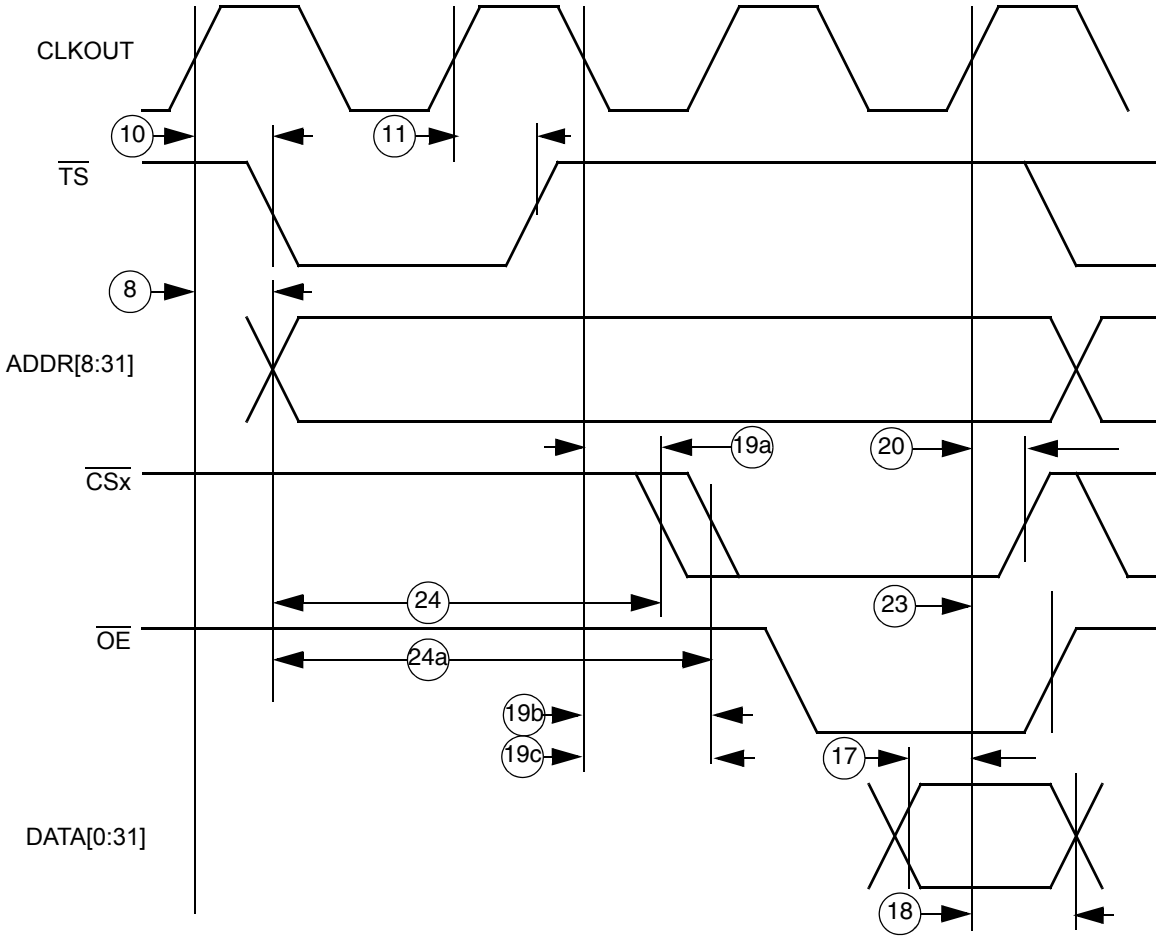


**Figure F-19. External Bus Read Timing (GPCM Controlled – TRLX = '1', ACS = '10', ACS = '11')**

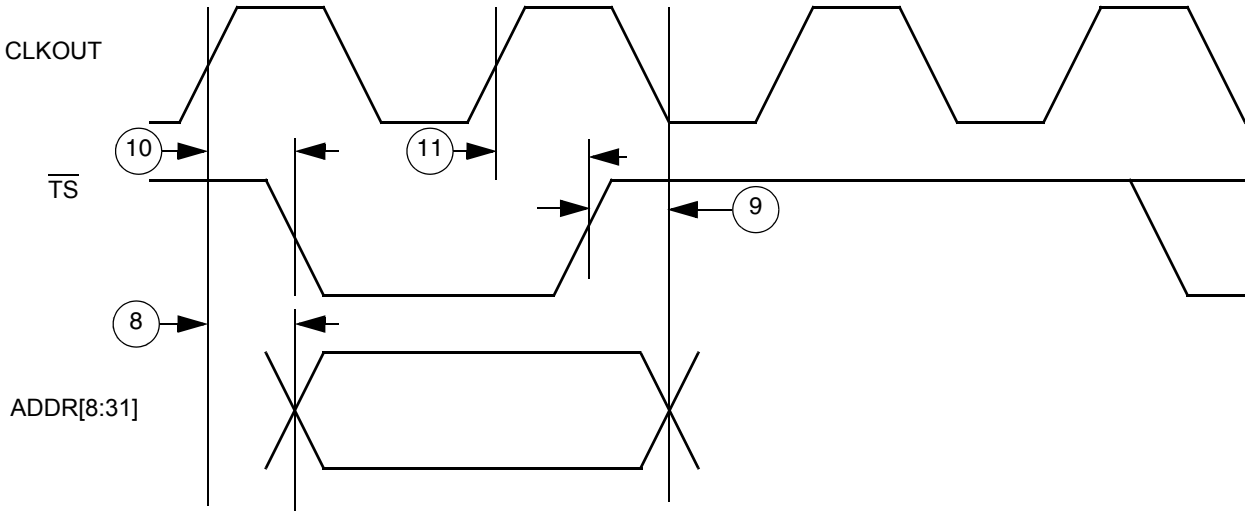


**Figure F-20. Address Show Cycle Bus Timing**

**Table F-28. MPC561/MPC563 Signal Names and Pin Names (continued)**

| Signal Name | Pin Name | Ball Assignment |
|---|---|---|
| DATA/SGPIOD[0:31] | data_sgpiod0 | AD13 |
| | data_sgpiod1 | AC12 |
| | data_sgpiod2 | AF14 |
| | data_sgpiod3 | AF13 |
| | data_sgpiod4 | AF15 |
| | data_sgpiod5 | AC13 |
| | data_sgpiod6 | AF16 |
| | data_sgpiod7 | AC14 |
| | data_sgpiod8 | AF17 |
| | data_sgpiod9 | AC16 |
| | data_sgpiod10 | AF18 |
| | data_sgpiod11 | AC17 |
| | data_sgpiod12 | AC18 |
| | data_sgpiod13 | AD18 |
| | data_sgpiod14 | AC20 |
| | data_sgpiod15 | AD19 |
| | data_sgpiod16 | AD20 |
| | data_sgpiod17 | AE20 |
| | data_sgpiod18 | AF20 |
| | data_sgpiod19 | AE19 |
| | data_sgpiod20 | AF19 |
| | data_sgpiod21 | AE18 |
| | data_sgpiod22 | AD17 |
| | data_sgpiod23 | AE17 |
| | data_sgpiod24 | AD16 |
| | data_sgpiod25 | AE16 |
| | data_sgpiod26 | AD15 |
| | data_sgpiod27 | AE15 |
| | data_sgpiod28 | AD14 |
| | data_sgpiod29 | AE14 |
| | data_sgpiod30 | AE13 |
| | data_sgpiod31 | AF12 |
| $\overline{\text{IRQ}}$0/SGPIOC0/MDO4 | irq0_b_sgpioc0_mdo4 | P3 |