SECTION 6
SYSTEM CONFIGURATION AND PROTECTION

The MPC565 / MPC566 incorporates many system functions that normally must be provided in external circuits. In addition, it is designed to provide maximum system safeguards again hardware and/or software faults. The system configuration and protection sub-module provides the following features:

- **System Configuration** — The USIU allows the configuration of the system according to the particular requirements. The functions include control of show cycle operation, pin multiplexing, and internal memory map location. System configuration also includes a register containing part and mask number constants to identify the part in software.

- **Interrupt Controller** — The interrupt controller receives interrupt requests from a number of internal and external sources and directs them on a single interrupt-request line to the RCPU.

- **General-Purpose I/O** — The USIU provides 64 pins for general-purpose I/O. The SGPIO pins are multiplexed with the address and data pins.

- **External Master Modes Support —** External master modes are special modes of operation that allow an alternate master on the external bus to access the internal modules for debugging and backup purposes.

- **Bus Monitor** — The SIU provides a bus monitor to watch internal to external accesses. It monitors the transfer acknowledge (TA) response time for internal to external transfers. A transfer error acknowledge ($\overline{\text{TEA}}$) is asserted if the TA response limit is exceeded. This function can be disabled.

- **Software Watchdog Timer (SWT)** — The SWT asserts a reset or non-maskable interrupt (as selected by the system protection control register) if the software fails to service the SWT for a designated period of time (e.g., because the software is trapped in a loop or lost). After a system reset, this function is enabled with a maximum time-out period and asserts a system reset if the time-out is reached. The SWT can be disabled or its time-out period can be changed in the SYPCR. Once the SYPCR is written, it cannot be written again until a system reset.

- **Periodic Interrupt Timer (PIT)** — The SIU provides a timer to generate periodic interrupts for use with a real-time operating system or the application software. The PIT provides a period from one µs to four seconds with a four-MHz crystal or 200 ns to 0.8 ms with a 20-MHz crystal. The PIT function can be disabled.

- **Power-PC Time Base Counter (TB)** — The TB is a 64-bit counter defined by the MPC565 / MPC566 architecture to provide a time base reference for the operating system or application software. The TB has four independent reference reg-

isters which can generate a maskable interrupt when the time-base counter reaches the value programmed in one of the four reference registers. The associated bit in the TB status register will be set for the reference register which generated the interrupt.

- **Power-PC Decrementer (DEC)** — The DEC is a 32-bit decrementing counter defined by the MPC565 / MPC566 architecture to provide a decrementer interrupt. This binary counter is clocked by the same frequency as the time base (also defined by the MPC565 / MPC566 architecture). The period for the DEC when driven by a 4-MHz oscillator is 4295 seconds, which is approximately 71.6 minutes.
- **Real-Time Clock (RTC)** — The RTC is used to provide time-of-day information to the operating system or application software. It is composed of a 45-bit counter and an alarm register. A maskable interrupt is generated when the counter reaches the value programmed in the alarm register. The RTC is clocked by the same clock as the PIT.
- **Freeze Support** — The SIU allows control of whether the SWT, PIT, TB, DEC and RTC should continue to run during the freeze mode.

Figure 6-1 shows a block diagram of the system configuration and protection logic.

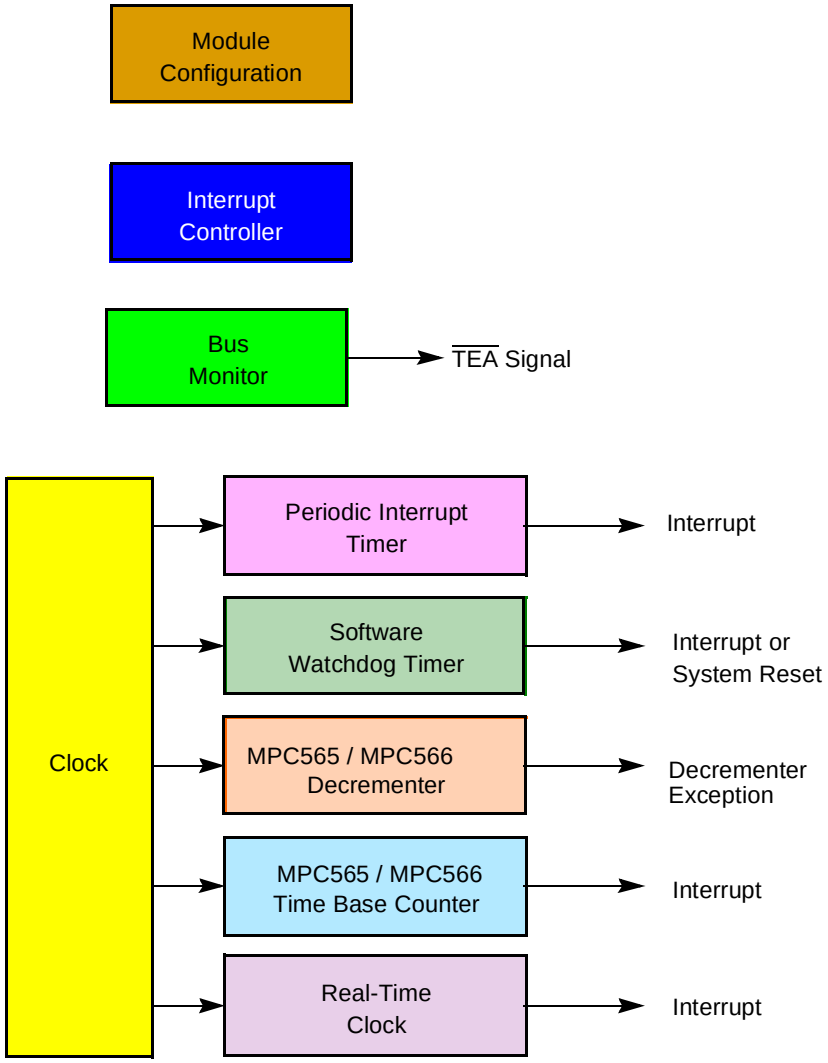


Figure 6-1 System Configuration and Protection Logic

6.1 System Configuration

The SIU allows the configuration of the system according to the particular requirements. The functions include control of show cycle operation, pin multiplexing, and internal memory map location. System configuration also includes a register containing part and mask number constants to identify the part in software.

System configuration registers include the system configuration register (SIUMCR), the internal memory mapping register (IMMR). Refer to **6.14 System Configuration and Protection Registers** for register diagrams and bit descriptions.

6.1.1 USIU Pins Multiplexing

Some of the functions defined in the various sections of the SIU (external bus interface, memory controller, and general-purpose I/O) share pins. **Table 6-1** summarizes how the pin functions of these multiplexed pins are assigned.

Table 6-1 USIU Pins Multiplexing Control

Pin Name	Multiplexing Controlled By:
IRQ0/SGPIOC0 IRQ1/RSV/SGPIOC1 IRQ2/CR/SGPIOC2/MTS IRQ3/KR/RETRY/SGPIOC3 IRQ4/AT2/SGPIOC4 IRQ5/SGPIOC5/MODCK1 IRQ6/MODCK2 IRQ7/MODCK3	At Power-On Reset: MODCK[1:3] Otherwise: Programmed in SIUMCR
SGPIOC6/FRZ/PTR SGPIOC7/IRQ_OUT/LWP0 BG/VF0/LWP1 BR/VF1/IWP2 BB/VF2/IWP3 IWP[0:1]/VFLS[0:1] BI/STS WE(0:3)/BE(0:3)/AT(0:3) TDI/DSDI TCK/DSCK TDO/DSDO	Programmed in SIUMCR and Hard Reset Configuration
DATA[0:31]/SGPIOD[0:31] ADDR[8:31]/SGPIOA[8:31]	Programmed in SIUMCR
RSTCONF/TEXP	At Power-On Reset: RSTCONF Otherwise: Programmed in SIUMCR

6.1.2 Memory Mapping

The MPC565 / MPC566 internal memory space can be assigned to one of eight locations.

The internal memory map is organized as a single four-Mbyte block. The user can assign this block to one of eight locations by programming the ISB field in the internal memory mapping register (IMMR). The eight possible locations are the first eight four-Mbyte memory blocks starting with address 0x0000 0000. (Refer to **Figure 6-2**.)

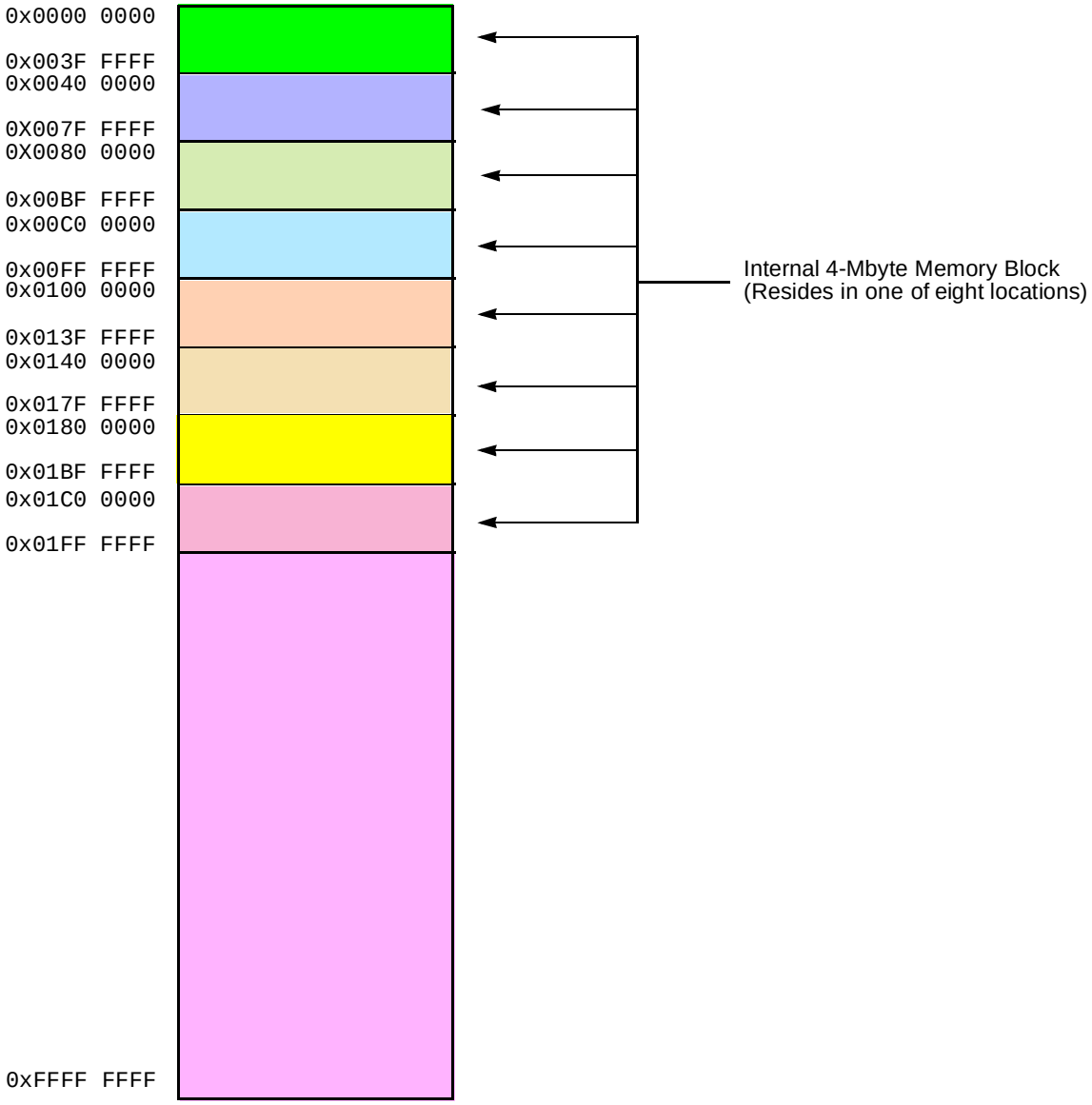


Figure 6-2 MPC565 / MPC566 Memory Map

6.1.3 Arbitration Support

Two bits in the SIUMCR control USIU bus arbitration. The external arbitration (EARB) bit determines whether arbitration is performed internally or externally. If EARB is cleared (internal arbitration), the external arbitration request priority (EARP) bit determines the priority of an external master's arbitration request. The operation of the internal arbiter is described in **9.5.6.4 Internal Bus Arbiter**.

6.2 External Master Modes

External master modes are special modes of operation that allow an alternative master on the external bus to access the internal modules for debugging and backup pur-

poses. They provide access to the internal buses (U-bus and L-bus) and to the intermodule bus (IMB3).

There are two external master modes. Peripheral mode (enabled by setting PRPM in the EMCR) uses a special slave mechanism which shuts down the RCPU and an alternative master on the external bus can perform accesses to any internal bus slave. Slave mode (enabled by setting SLVM and clearing PRPM in the EMCR) enables an external master to access any internal bus slave while the RCPU is fully operational. Both modes can be enabled and disabled by software. In addition, peripheral mode can be selected from reset.

The internal bus is not capable of providing fair priority between internal RCPU accesses and external master accesses. If the bandwidth of external master accesses is large, it is recommended that the system forces gaps between external master accesses in order to avoid suspension of internal RCPU activity.

The MPC565 / MPC566 does not support burst accesses from an external master; only single accesses of 8, 16, or 32 bits can be performed. The MPC565 / MPC566 asserts burst inhibit ($\overline{BI}$) on any attempt to initiate a burst access to internal memory.

The MPC565 / MPC566 provides memory controller services for external master accesses (single and burst) to external memories. See **SECTION 10 MEMORY CONTROLLER** for details.

6.2.1 Operation in External Master Modes

The external master modes are controlled by the EMCR register, which contains the internal bus attributes. The default attributes in the EMCR enable the external master to configure EMCR with the required attributes, and then access the internal registers. The external master must be granted external bus ownership in order to initiate the external master access. The SIU compares the address on the external bus to the allocated internal address space. If the address is within the internal space, the access is performed with the internal bus. The internal address space is determined according to ISB field (see **6.14.1.2 Internal Memory Map Register** for details). The external master access is terminated by the $\overline{TA}$, $\overline{TEA}$ or $\overline{RETRY}$ signal on the external bus.

A deadlock situation might occur if an internal-to-external access is attempted on the internal bus while an external master access is initiated on the external bus. In this case, the SIU will assert the $\overline{RETRY}$ on the external bus in order to relinquish and retry the external access until the internal access is completed. The internal bus will deny other internal accesses for the next eight clocks in order to complete the pending accesses and prevent additional internal accesses from being initiated on the internal bus. The SIU will also mask internal accesses to support consecutive external accesses if the delay between the external accesses is less than four clocks. The external master access and retry timings are described in **9.5.11 Bus Operation in External Master Modes**.

The external master may access the internal MPC565 / MPC566 special registers that are located outside the RCPU. In order to access one of these MPC565 / MPC566 registers, program the EMCR to MPC565 / MPC566 special register access (CONT = 1

and SUPU = 0 in EMCR). Next, access the register by providing the address according to the MPC565 / MPC566 address map. Only the first external master access that follows EMCR setting will be assigned to the special register map; the next accesses will be directed to the normal address map. This is done in order to enable access to the EMCR again after the required MPC565 / MPC566 special register access.

Peripheral mode does not require external bus arbitration between the external master and the internal RCPU, since the internal RCPU is disabled. The $\overline{BR}$ and $\overline{BB}$ signals should be connected to ground, and the internal bus arbitration should be selected in order to prevent the “slave” MPC565 / MPC566 from occupying the external bus. Internal bus arbitration is selected by clearing the EARB bit in the SIUMCR (see **6.14.1.1 SIU Module Configuration Register (SIUMCR)**).

6.2.2 Address Decoding for External Accesses

During an external master access, the USIU compares the external address with the internal address block to determine if MPC565 / MPC566 operation is required. Since only 24 of the 32 internal address bits are available on the external bus, the USIU assigns zeros to the most significant address bits (ADDR[0:7]).

The address compare sequence can be summarized as follows:

- Normal external access. If the CONT bit in EMCR is cleared, the address is compared to the internal address map. Refer to **6.14.1.3 External Master Control Register (EMCR)**.
- MPC565 / MPC566 special register external access. If the CONT bit in EMCR is set by the previous external master access, the address is compared to the MPC565 / MPC566 special address range. See **5.2.2 USIU Special Purpose Registers** for a list of the SPRs in the USIU.
- Memory controller external access. If the first two comparisons do not match, the internal memory controller determines whether the address matches an address assigned to one of the regions. If it finds a match, the memory controller generates the appropriate chip select and attribute accordingly

When trying to fetch an MPC565 / MPC566 special register from an external master, the address might be aliased to one of the external devices on the external bus. If this device is selected by the MPC565 / MPC566 internal memory controller, this aliasing does not occur since the chip select is disabled. If the device has its own address decoding or is being selected by external logic, this case should be resolved.

6.3 USIU General-Purpose I/O

The USIU provides 64 general-purpose I/O (SGPIO) pins. The SGPIO pins are multiplexed with the address and data pins. In single-chip mode, where communicating with external devices is not required, all 64 SGPIO pins can be used. In multiple-chip mode, only eight SGPIO pins are available. Another configuration allows the use of the address bus for instruction show cycles while the data bus is dedicated to SGPIO functionality. The functionality of these pins is assigned by the single-chip (SC) bit in the SIUMCR. (See **6.14.1.1 SIU Module Configuration Register (SIUMCR)**.)

SGPIO pins are grouped as follows:

- Six groups of eight pins each, whose direction is set uniformly for the whole group
- 16 single pins whose direction is set separately for each pin

Table 6-2 describes the SGPIO signals, and all available configurations. The SGPIO registers are described in **6.14.5 General-Purpose I/O Registers**.

Table 6-2 SGPIO Configuration

SGPIO Group Name	Individual Pin Control	Direction Control	Available When SC = 00 (32-bit Port Size Mode)	Available When SC = 01 (16-bit Port Size Mode)	Available When SC = 10 (Single-Chip Mode with Trace)	Available When SC = 11 (Single-Chip Mode)
SGPIOD[0:7]		GDDR0			X	X
SGPIOD[8:15]		GDDR1			X	X
SGPIOD[16:23]		GDDR2		X	X	X
SGPIOD[24:31]	X	SDDRD[23:31]		X	X	X
SGPIOC[0:7][1]	X	SDDRC[0:7]				
SGPIOA[8:15]		GDDR3				X
SGPIOA[16:23]		GDDR4				X
SGPIOA[24:31]		GDDR5				X

NOTES:
1. SGPIOC[0:7] is selected according to GPC and MLRC fields in SIUMCR. See **6.14.1.1 SIU Module Configuration Register (SIUMCR)**.

Figure 6-3 illustrates the functionality of the SGPIO.

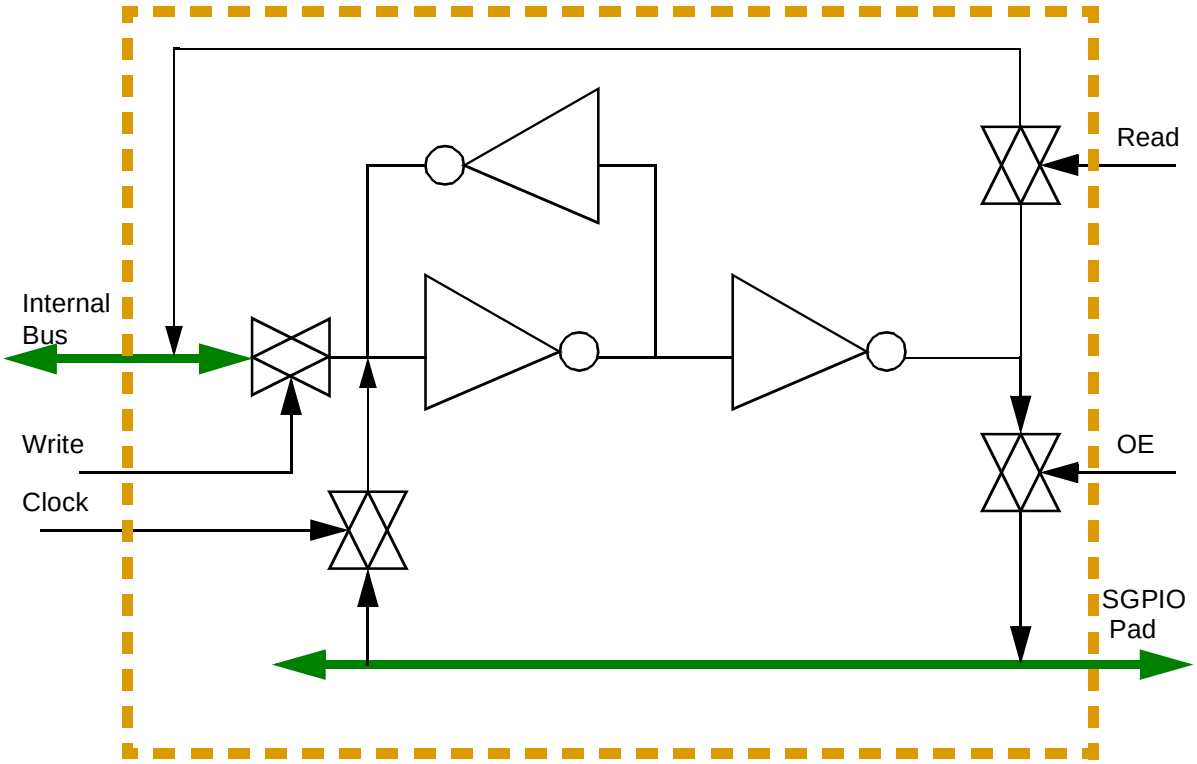


Figure 6-3 SGPIO Cell

6.4 Enhanced Interrupt Controller

6.4.1 Key Features

- Significant interrupt latency reduction
- Simplified interrupt structure
- Up to 48 different interrupt requests
- Splitting of single PowerPC™ external interrupt vector into up to 48 vectors, one for each source
- Automatic lower priority requests masking
- Full backward compatibility with MPC555/MPC556 (enhanced mode is software programmable.)

6.4.2 Interrupt Configuration

An overview of the MPC565 / MPC566 interrupt structure is shown in **Figure 6-4**. The interrupt controller receives interrupts from USIU internal sources, such as PIT, RTC, from the UIMB module (which has its own interrupt controller) or from the IMB3 bus (directly from IMB modules) and from external pins $\overline{\text{IRQ}}$[0:7].

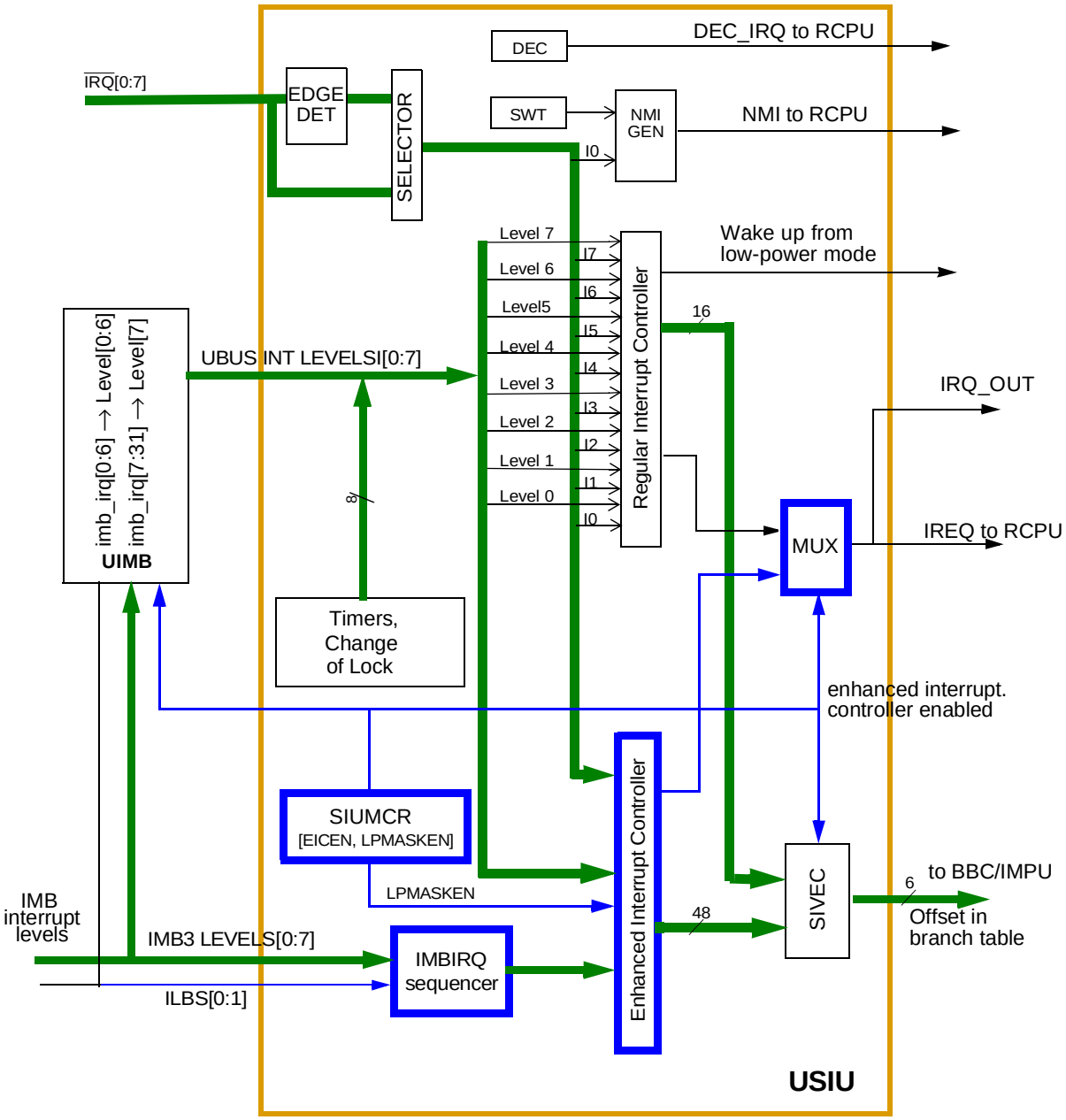


Figure 6-4 MPC565 / MPC566 Interrupt Structure

If it was programmed to generate interrupt, the SWT and external pin $\overline{\text{IRQ}}$[0] always generate an NMI, non-maskable interrupt to the RCPU.

NOTE

The RCPU takes the system reset exception when an NMI is asserted, the external interrupt exception for any other asserted interrupt request and the decrementer exception when the decrementer MSB changes from 0 to 1.

Decrementer interrupt request is not a part of interrupt controller. Each one of the external pins IRQ[1:7] has its own dedicated assigned priority level. IRQ[0] is also mapped, but it should be used only as a status bit indicating that IRQ[0] was asserted and generated NMI interrupt. There are additional eight interrupt priority levels. Each one of the SIU internal interrupt sources, or any of the peripheral module interrupt sources can be assigned by a software to any one of those eight interrupt priority levels. Thus, a very flexible interrupt scheme is implemented. The interrupt request signal generated by the interrupt controller is driven to the PPC core and to IRQ_OUT pin (optionally). This pin may be used in peripheral mode, when the RCPU is disabled, and the internal modules are accessed externally. The IMB interrupts are controlled by the UIMB. The IMB provides 32 interrupt levels, and any interrupt source could be configured to any IMB interrupt level. The UIMB contains 32-bit register that holds the IMB interrupt requests, and drives them on the USIU eight interrupt levels.

NOTE

If one interrupt level was configured to more than one interrupt source, the software should read the UIPEND register in the UIMB module, and the particular status bits in order to identify which interrupt was asserted.

The interrupt controller may be programmed to operate in two modes — a regular mode and an enhanced mode.

6.4.3 Regular Interrupt Controller Operation (MPC555/MPC556 Compatible Mode)

In regular operation mode (default setting) the interrupt controller receives interrupt requests from internal sources, such as timers, PLL lock detector and IMB modules and from external pins IRQ[0:7]. All the internal interrupt sources may be programmed to drive one or more of eight U-bus interrupt level lines while the RCPU, upon receiving an interrupt request, has to read the USIU and UIMB status register in order to determine the interrupt source.

The SIVEC register contains an 8-bit code representing the unmasked interrupt request which has the highest priority level. The priority between all interrupt sources for the regular interrupt controller operation is shown in **Table 6-3**.

Table 6-3 Priority Of interrupt Sources — Regular Operation

Number	Priority Level	Interrupt Source Description	Offset in Branch Table (Hex)	SIVEC Contents (8-ms bit)
0	Highest	$\overline{\text{EXT\_IRQ}}$[0]	0x0000	00000000
1	—	Level 0	0x0008	00000100
2	—	EXT_$\overline{\text{IRQ}}$[1]	0x0010	00001000
3	—	Level 1	0x0018	00001100
4	—	EXT_$\overline{\text{IRQ}}$[2]	0x0020	00010000
5	—	Level 2	0x0028	00010100
6	—	EXT_$\overline{\text{IRQ}}$[3]	0x0030	00011000
7	—	Level 3	0x0038	00011100
8	—	EXT_$\overline{\text{IRQ}}$[4]	0x0040	00100000
9	—	Level 4	0x0048	00100100
10	—	EXT_$\overline{\text{IRQ}}$[5]	0x0050	00101000
11	—	Level 5	0x0058	00101100
12	—	EXT_$\overline{\text{IRQ}}$[6]	0x0060	00110000
13	—	Level 6	0x0068	00110100
14	—	EXT_$\overline{\text{IRQ}}$[7]	0x0070	00111000
15	Lowest	Level 7	0x0078	00111100

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, whose influence may be neutralized by clearing an appropriate bit in SIMASK register.

6.4.4 Enhanced Interrupt Controller

6.4.4.1 General Operation

The enhanced interrupt controller operation may be turned on by setting a EICEN control bit in 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 time multiplexing scheme defined by the UIMB module. In addition to the IMB interrupt sources the external interrupts and timer interrupts are available by the same way as in the regular

scheme. In this mode, the UIMB module **does not drive** UBUS interrupt level lines. Each interrupt request will set a corresponding bit in SIPEND2 or SIPEND3 registers, whose influence may be neutralized 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 SIVEC register. See **Table 6-4**.

Table 6-4 Priority of Interrupt Sources — Enhanced Operation

Number	Priority Level	Interrupt Source Description	Offset in Branch Table (Hex)	SIVEC Contents (8-ms bit)
0	Highest	EXT_IRQ 0	0x0000	00000000
1	—	Level 0	0x0008	00000100
2	—	IMB_IRQ 0	0x0010	00001000
3	—	IMB_IRQ 1	0x0018	00001100
4	—	IMB_IRQ 2	0x0020	00010000
5	—	IMB_IRQ 3	0x0028	00010100
6	—	EXT_IRQ 1	0x0030	00011000
7	—	Level 1	0x0038	00011100
8	—	IMB_IRQ 4	0x0040	00100000
9	—	IMB_IRQ 5	0x0048	00100100
10	—	IMB_IRQ 6	0x0050	00101000
11	—	IMB_IRQ 7	0x0058	00101100
12	—	EXT_IRQ 2	0x0060	00110000
13	—	Level 2	0x0068	00110100
14	—	IMB_IRQ 8	0x0070	00111000
15	—	IMB_IRQ 9	0x0078	00111100
16	—	IMB_IRQ 10	0x0080	01000000
17	—	IMB_IRQ 11	0x0088	01000100
18	—	EXT_IRQ 3	0x0090	01001000
19	—	Level 3	0x0098	01001100
20	—	IMB_IRQ 12	0x00a0	01010000
21	—	IMB_IRQ 13	0x00a8	01010100
22	—	IMB_IRQ 14	0x00b0	01011000
23	—	IMB_IRQ 15	0x00b8	01011100
24	—	EXT_IRQ 4	0x00c0	01100000
25	—	Level 4	0x00c8	01100100
26	—	IMB_IRQ 16	0x00d0	01101000
27	—	IMB_IRQ 17	0x00d8	01101100
28	—	IMB_IRQ 18	0x00e0	01110000
29	—	IMB_IRQ 19	0x00e8	01110100
30	—	EXT_IRQ 5	0x00f0	01111000
31	—	Level 5	0x00f8	01111100
32	—	IMB_IRQ 20	0x0100	10000000

Table 6-4 Priority of Interrupt Sources — Enhanced Operation (Continued)

Number	Priority Level	Interrupt Source Description	Offset in Branch Table (Hex)	SIVEC Contents (8-ms bit)
33	—	IMB_IRQ 21	0x0108	10000100
34	—	IMB_IRQ 22	0x0110	10001000
35	—	IMB_IRQ 23	0x0118	10001100
36	—	EXT_IRQ 6	0x0120	10010000
37	—	Level 6	0x0128	10010100
38	—	IMB_IRQ 24	0x0130	10011000
39	—	IMB_IRQ 25	0x0138	10011100
40	—	IMB_IRQ 26	0x0140	10100000
41	—	IMB_IRQ 27	0x0148	10100100
42	—	EXT_IRQ 7	0x0150	10101000
43	—	Level 7	0x0158	10101100
44	—	IMB_IRQ 28	0x0160	10110000
45	—	IMB_IRQ 29	0x0168	10110100
46	—	IMB_IRQ 30	0x0170	10111000
47	Lowest	IMB_IRQ 31	0x0178	10111100

A value from SIVEC register is supplied internally to the BBC module and can be used for the external interrupt relocation feature as an offset to the branch table start address. Thus a fast way to a specific interrupt source routine is provided without software overhead. The BBCMCR (see **4.8.2.1 BBC Module Configuration Register BBCMCR**) and EIBADR (see **4.8.2.5 External Interrupt Relocation Table Base Address Register — EIBADR**) registers must be programmed to enable this feature in the BBC. Additionally, the SIPEND2 and SIPEND3 registers contain the information about all the interrupt requests that are asserted at a given time, so a software can always read them.

NOTE

In the case of enhanced interrupt controller the SIPEND and SIMASK registers are not used.

6.4.4.2 Lower Priority Request Masking

This feature (if enabled) simplifies the masking of lower priority interrupt requests when a request of certain priority is in service in applications that require interrupt nesting. **The highest (pending) request is also masked by itself.** The masking is accomplished in the following way.

Upon asserting an interrupt request the BBC generates an acknowledge signal to notify the interrupt controller that the request and the branch table offset have been latched. The interrupt controller then sets a bit in SISR (in service register), according to the asserted request and all the other requests whose priority is **lower** than or **equal** to the one that currently in service, become masked. The mask remains set until the SISR bit is cleared by a software (by interrupt handler), writing ‘1’ value to the corre-

sponding bit. The lower priority request masking diagram is presented in the **Figure 6-5**.

The lower priority request masking feature is disabled by $\overline{\text{HRESET}}$ and it may be enabled by setting the LPMASK_EN bit in the SIUMCR register.

NOTES

In regular mode of the interrupt controller the lower priority request masking feature is not possible.

The feature must be activated only together with exception table relocation in the BBC module.

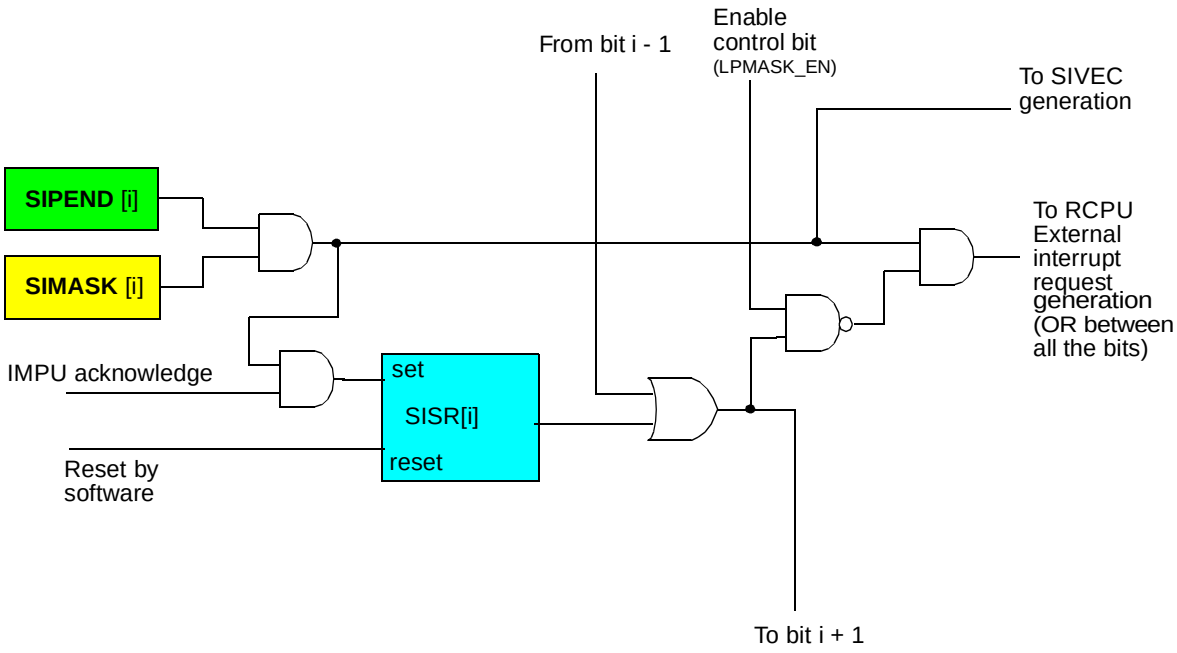


Figure 6-5 Lower Priority Request Masking — One Bit Diagram

6.4.4.3 Backward Compatibility with MPC555/MPC556

The enhanced interrupt controller is a feature that may be enabled according to a user's application using the EICEN control bit in SIUMCR register, which can be set and cleared at any time by a software. If the bit is cleared, the default interrupt controller operation is available, as described in **6.4.3 Regular Interrupt Controller Operation (MPC555/MPC556 Compatible Mode)**. The regular operation is fully compatible with the interrupt controller already implemented in MPC555 / MPC556.

Figure 6-6 illustrates the interrupt controller functionality in the MPC565 / MPC566.

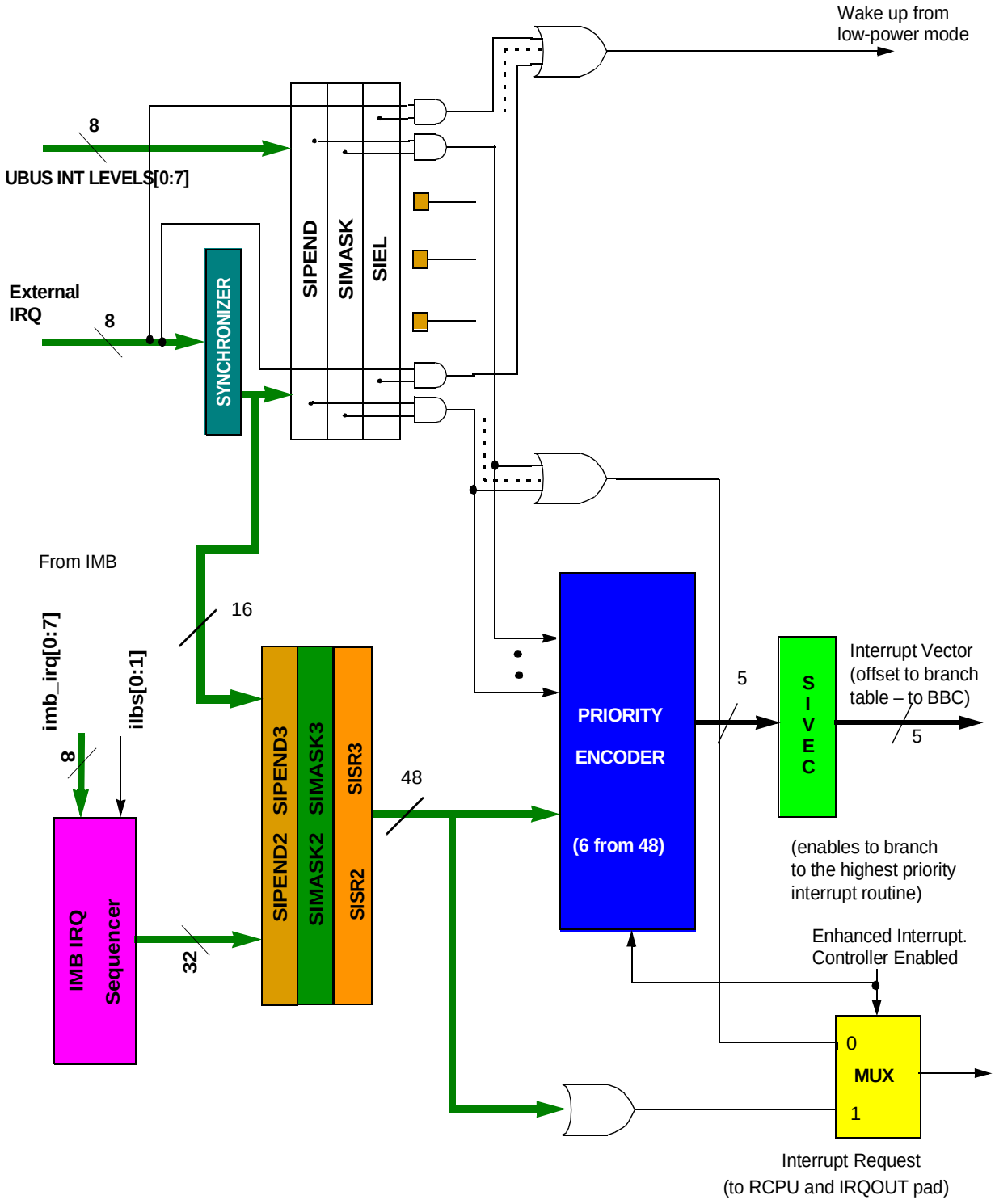


Figure 6-6 MPC565 / MPC566 Interrupt Controller Block Diagram

6.5 Interrupt Overhead Estimation for Enhanced Interrupt Controller Mode

The interrupt overhead includes two main parts:

- Store general, and special purpose registers
- Interrupt source recognitions

The interrupt overhead leads to large latency, and decreases the overall system performance. The registers saving overhead can be reduced by improving the operating system. The number of registers that should be saved, can be reduced when each interrupt event will have it's own interrupt vector. The current solution solves the interrupt source recognition overhead. **Table 6-5** below illustrates the improvements.

Only registers required for the recognition routine are considered to be saved in the calculations. Modules internal events/channels recognition is out of the scope of the calculations. See also typical interrupt handler flowchart in the **Figure 6-7**.

Table 6-5 Interrupt Latency Estimation for Three Typical Cases

	MPC555 / MPC556 Architecture without SIVEC Use	MPC565 / MPC566 Architecture with SIVEC Feature Use	MPC565 / MPC566 Architecture with Enhanced Interrupt Features
Operation Details	Interrupt propagation from request module to RCPU — **8 clocks** Store of some GPR and SPR — **10 clocks** Read SIPEND — **4 clocks** Read SIMASK — **4 clocks** SIPEND data processing — **20 clocks** (find first set, access to LUT in the flash, branches) Read UIPEND — **4 clocks** UIPEND data processing — **20 clocks** (find first set, access to LUT in the flash, branches)	Interrupt propagation from request module to RCPU — **8 clocks** Store of some GPR and SPR — **10 clocks** Read SIVEC — **4 clocks** Branch to routine — **10 clocks** Read UIPEND — **4 clocks** UIPEND data processing — **20 clocks** (find first set, access to LUT in the flash, branches)	Interrupt propagation from request module to RCPU — **6 clocks** Store of some GPR and SPR — **10 clocks** Only one branch should be executed to arrive appropriate interrupt handler belong to its own interrupt requesting device — **2 clocks**
Notes:	If there is a need to enable nesting of interrupts during source recognition procedure, at least **30 clocks** should be added to the interrupt latency estimation	To use this feature in compressed mode some tens of clocks should be added to make a branch to compressed address of the routine	—
Total:	**At Least 70-80 Clocks**	**At Least 50-60 Clocks**	**20 Clocks**

NOTE

Compiler and bus collisions overhead are not included in the calculations.

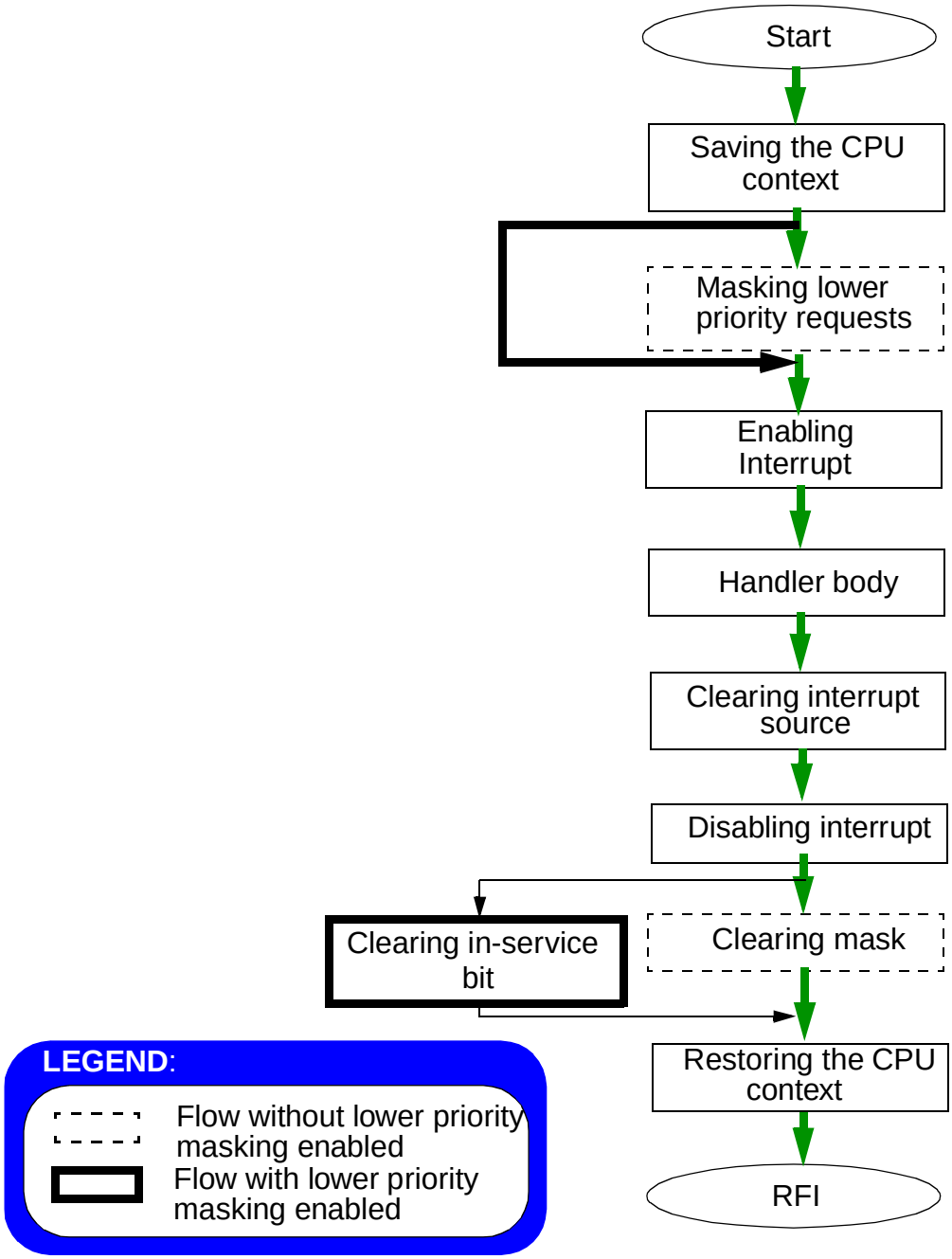


Figure 6-7 Typical Interrupt Handler Routine

6.6 Hardware Bus Monitor

The bus monitor ensures that each bus cycle is terminated within a reasonable period of time. The USIU provides a bus monitor option to monitor internal to external bus accesses on the external bus. The monitor counts from transfer start to transfer

acknowledge and from transfer acknowledge to transfer acknowledge within bursts. If the monitor times out, transfer error acknowledge ($\overline{TEA}$) is asserted internally.

The bus monitor timing bit in the system protection control register (SYPCR) defines the bus monitor time-out period. The programmability of the time-out allows for variation in system peripheral response time. The timing mechanism is clocked by the system clock divided by eight. The maximum value is 2040 system clock cycles.

The bus monitor enable (BME) bit in the SYPCR enables or disables the bus monitor. The bus monitor is always enabled, however, when freeze is asserted or when a debug mode request is pending, regardless of the state of this bit.

6.7 MPC565 / MPC566 Decrementer

The decrementer (DEC) is a 32-bit decrementing counter defined by the MPC565 / MPC566 architecture to provide a decrementer interrupt. This binary counter is clocked by the same frequency as the time base (also defined by the MPC565 / MPC566 architecture). The operation of the time base and decrementer are therefore coherent. In the MPC565 / MPC566, the DEC is clocked by the TMBCLK clock. The decrementer period is computed as follows:

$$T_{dec} = \frac{2^{32}}{F_{tmbclk}}$$

The state of the DEC is not affected by any resets and should be initialized by software. The DEC runs continuously after power-up once the time base is enabled by setting the TBE bit of the TBSCR (see **Table 6-18**) (unless the clock module is programmed to turn off the clock). The decrementer continues counting while reset is asserted.

Loading from the decrementer has no effect on the counter value. Storing to the decrementer replaces the value in the decrementer with the value in the GPR.

Whenever bit zero (the MSB) of the decrementer changes from zero to one, a decrementer exception occurs. If software alters the decrementer such that the content of bit 0 is changed to a value of 1, a decrementer exception occurs.

A decrementer exception causes a decrementer interrupt request to be pending in the RCPU. When the decrementer exception is taken, the decrementer interrupt request is automatically cleared.

Table 6-6 illustrates some of the periods available for the decrementer, assuming a 4-MHz or 20-MHz crystal, and TBS = 0 which selects **TBCLK** division to 4.

Table 6-6 Decrementer Time-Out Periods

Count Value	Time-Out @ 4 MHz	Time-Out @ 20 MHz
0	1.0 µs	0.2 µs
9	10 µs	2.0 µs
99	100 µs	20 µs
999	1.0 ms	200 µs
9999	10.0 ms	2 ms
999999	1.0 s	200 ms
9999999	10.0 s	2.0 s
99999999	100.0 s	20 s
999999999	1000. s	200 s
(hex) FFFFFFFF	4295 s	859 s

Refer to **3.9.5 Decrementer Register (DEC)** for more information.

6.8 MPC565 / MPC566 Time Base (TB)

The time base (TB) is a 64-bit free-running binary counter defined by the MPC565 / MPC566 architecture. The TB has two independent reference registers which can generate a maskable interrupt when the time base counter reaches the value programmed in one of the two reference registers. The period of the TB depends on the driving frequency. In the MPC565 / MPC566, the TB is clocked by the TMBCLK clock. The period for the TB is:

$$T_{TB} = \frac{2^{64}}{F_{tmbclk}}$$

The state of the time base is not affected by any resets and should be initialized by software. Reads and writes of the TB are restricted to special instructions. Separate special-purpose registers are defined in the MPC565 / MPC566 architecture for reading and writing the time base. For the MPC565 / MPC566 implementation, it is not possible to read or write the entire TB in a single instruction. Therefore, the **mttb** and **mftb** instructions are used to move the lower half of the time base (TBL) while the **mttbu** and **mftbu** instructions are used to move the upper half (TBU).

Two reference registers are associated with the time base: TBREF0 and TBREF1. A maskable interrupt is generated when the TB count reaches to the value programmed in one of the two reference registers. Two status bits in the time base control and status register (TBSCR) indicate which one of the two reference registers generated the interrupt.

Refer to **6.14.4 System Timer Registers** for diagrams and bit descriptions of time base registers. Refer to **3.9.4 Time Base Facility (TB) — OEA** and to ***RCPU Reference Manual* (RCPURM/AD)** for additional information regarding the MPC565 / MPC566 time base.

6.9 Real-Time Clock (RTC)

The RTC is a 32-bit counter and pre-divider used to provide a time-of-day indication to the operating system and application software. It is clocked by the *pitrtclk* clock.The counter is not affected by reset and operates in all low-power modes. It is initialized by software. The RTC can be programmed to generate a maskable interrupt when the time value matches the value programmed in its associated alarm register. It can also be programmed to generate an interrupt once a second. A control and status register is used to enable or disable the different functions and to report the interrupt source.

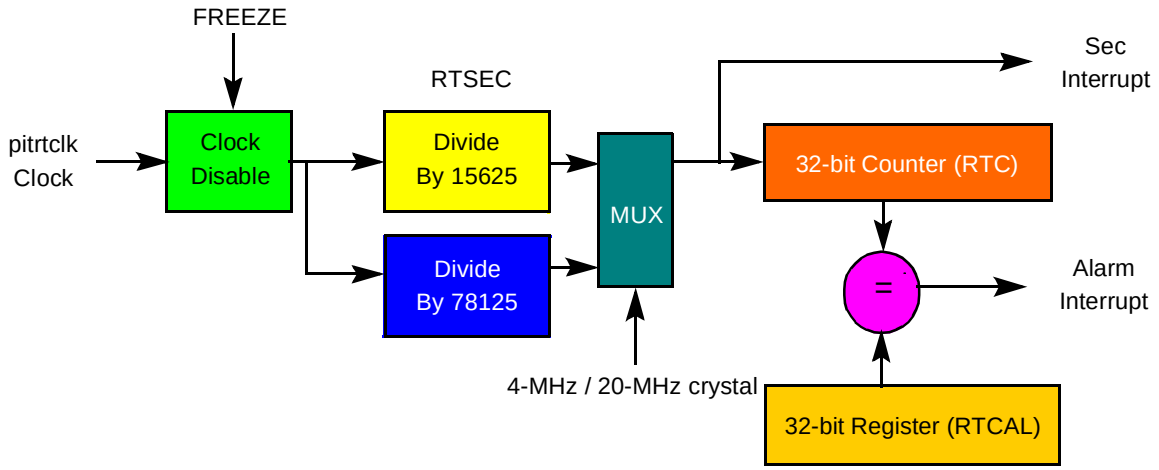


Figure 6-8 RTC Block Diagram

6.10 Periodic Interrupt Timer (PIT)

The periodic interrupt timer consists of a 16-bit counter clocked by the PITRCLK clock supplied by the clock module.

The 16-bit counter counts down to zero when loaded with a value from the PITC. After the timer reaches zero, the PS bit is set and an interrupt is generated if the PIE bit is a logic one. The software service routine should read the PS bit and then write it to zero to terminate the interrupt request. At the next input clock edge, the value in the PITC is loaded into the counter, and the process starts over again.

When a new value is loaded into the PITC, the periodic timer is updated, the divider is reset, and the counter begins counting. If the PS bit is not cleared, an interrupt request is generated. The request remains pending until PS is cleared. If the PS bit is set again prior to being cleared, the interrupt remains pending until PS is cleared.

Any write to the PITC stops the current countdown, and the count resumes with the new value in PITC. If the PTE bit is not set, the PIT is unable to count and retains the old count value. Reads of the PIT have no effect on the counter value.

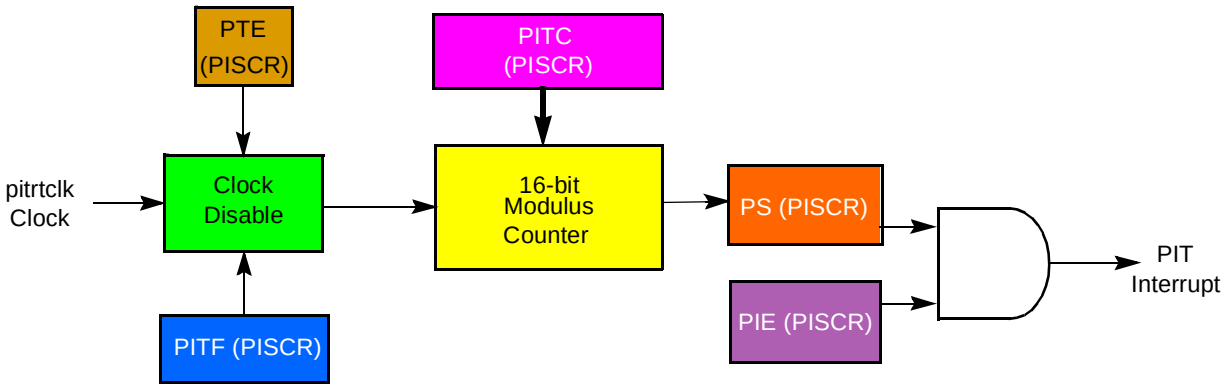


Figure 6-9 PIT Block Diagram

The timeout period is calculated as:

$$PIT_{period} = \frac{PITC + 1}{F_{pitrtclk}} = \frac{PITC + 1}{\left(\frac{ExternalClock}{4or256}\right)}$$

Solving this equation using a 4-MHz external clock and a pre-divider of 256 gives:

$$PITperiod = \frac{PITC + 1}{15625}$$

This gives a range from 64 microseconds, with a PITC of 0x0000, to 4.19 seconds, with a PITC of 0xFFFF. When a 20-MHz crystal is used with a pre-divider of 256, the range is between 12.8 microseconds to 0.84 seconds.

6.11 Software Watchdog Timer (SWT)

The software watchdog timer (SWT) prevents system lockout in case the software becomes trapped in loops with no controlled exit. The SWT is enabled after system reset to cause a system reset if it times out. it. The SWT requires a special service sequence to be executed on a periodic basis. If this periodic servicing action does not occur, the SWT times out and issues a reset or a non-maskable interrupt (NMI), depending on the value of the SWRI bit in the SYPCR.

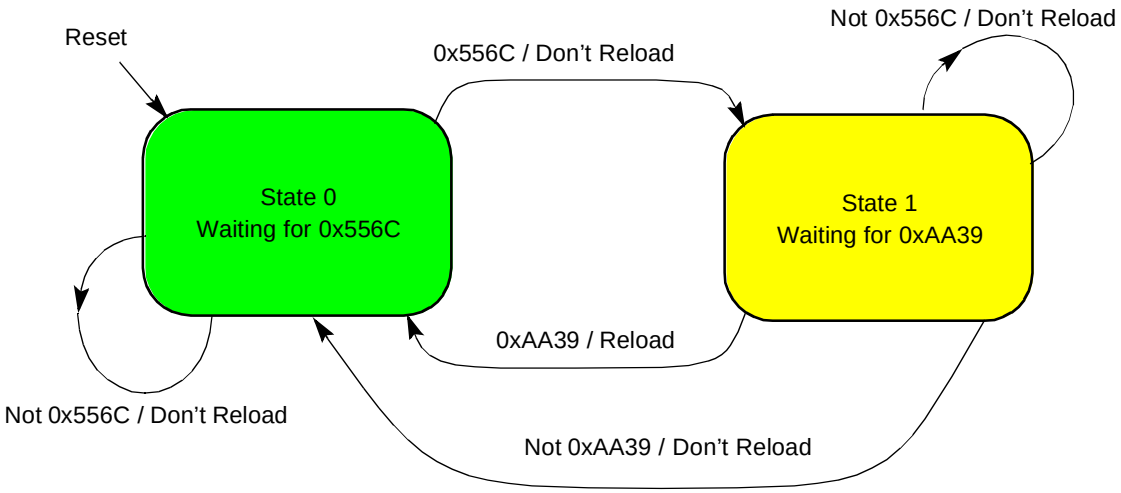
The SWT can be disabled by clearing the SWE bit in the SYPCR. Once the SYPCR is written by software, the state of the SWE bit cannot be changed.

The SWT service sequence consists of the following two steps:

1. Write 0x556C to the software service register (SWSR)
2. Write 0xAA39 to the SWSR

The service sequence clears the watchdog timer and the timing process begins again. If any value other than 0x556C or 0xAA39 is written to the SWSR, the entire sequence must start over.

Although the writes must occur in the correct order prior to time-out, any number of instructions may be executed between the writes. This allows interrupts and exceptions to occur, if necessary, between the two writes.



Although most software disciplines support the watchdog concept, different systems require different time-out periods. For this reason, the software watchdog provides a selectable range for the time-out period.

In **Figure 6-10**, the range is determined by the value SWTC field. The value held in the SWTC field is then loaded into a 16-bit decrementer clocked by the system clock. An additional divide by 2048 prescaler is used if necessary. The decrementer begins counting when loaded with a value from the software watchdog timing count field (SWTC). After the timer reaches 0x0, a software watchdog expiration request is issued to the reset or NMI control logic.

Upon reset, the value in the SWTC is set to the maximum value and is again loaded into the software watchdog register (SWR), starting the process over. When a new value is loaded into the SWTC, the software watchdog timer is not updated until the servicing sequence is written to the SWSR. If the SWE is loaded with the value zero, the modulus counter does not count.

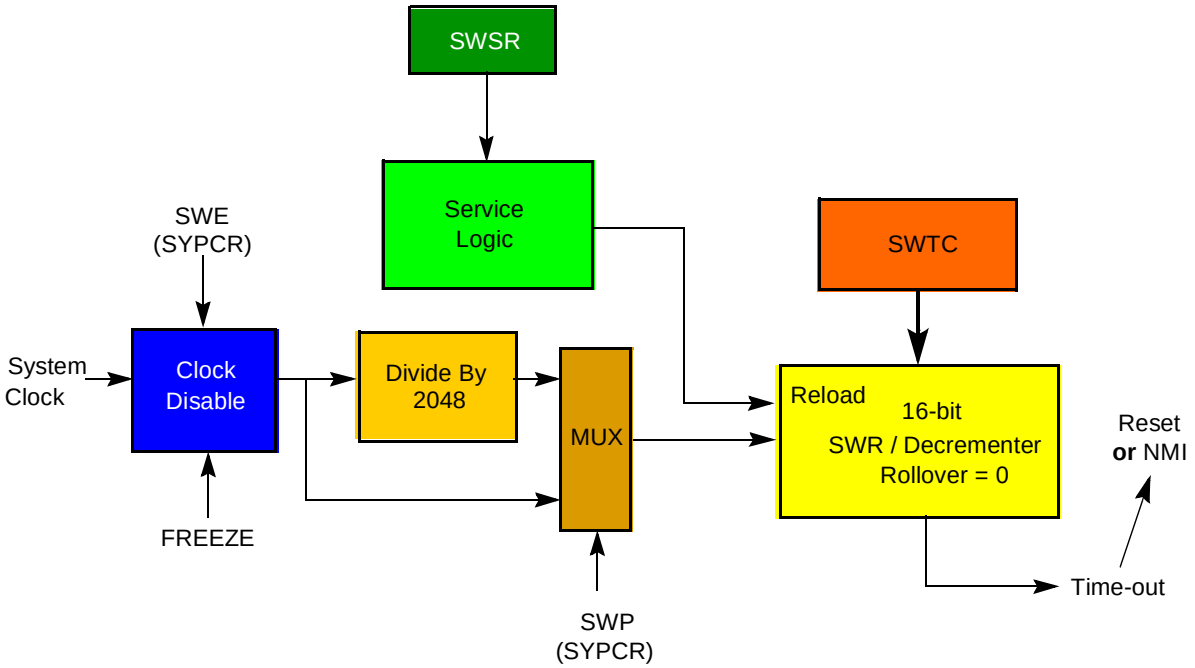


Figure 6-10 SWT Block Diagram

6.12 Freeze Operation

When the FREEZE line is asserted, the clocks to the software watchdog, the periodic interrupt timer, the real-time clock, the time base counter, and the decrementer can be disabled. This is controlled by the associated bits in the control register of each timer. If programmed to stop during FREEZE assertion, the counters maintain their values while FREEZE is asserted, unless changed by the software. The bus monitor, however, remains enabled regardless of this signal.

6.13 Low Power Stop Operation

When the processor is set in a low-power mode (doze, sleep, or deep sleep), the software watchdog timer is frozen. It remains frozen and maintain its count value until the processor exits this state and resumes executing instructions.

The periodic interrupt timer, decrementer, and time base are not affected by these low-power modes. They continue to run at their respective frequencies. These timers are capable of generating an interrupt to bring the MCU out of these low-power modes.

6.14 System Configuration and Protection Registers

This section provides diagrams and bit descriptions of the system configuration and protection registers.

6.14.1 System Configuration Registers

System configuration registers include the SIUMCR, the IMMR and the EMCR registers.

6.14.1.1 SIU Module Configuration Register (SIUMCR)

The SIUMCR contains bits which configure various features in the SIU module. The register contents are shown below.

SIUMCR — SIU Module Configuration Register **0x2F C000**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
EARB	EARP			Reserved				DSHW	DBGC		DBPC	ATWC	GPC		DLK
HRESET:															
ID(0)[1]	0	0	0	0	0	0	0	0	ID(9:10)[2]		ID(11)[3]	ID(12)[4]	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
Re-served	SC		RCTX	MLRC		Reserved		MTSC	NOSHOW	EICEN	LPMASK_EN	Reserved			
HRESET:															
0	ID(17:18)[5]		0	00		0	0	0	0	0	0	0	0	0	0

NOTES:
1. The HRESET value is a reset configuration word value, extracted from the internal data bus ID(0) line.
2. The HRESET value is a reset configuration word value, extracted from the internal data bus ID(9:10) lines.
3. The HRESET value is a reset configuration word value, extracted from the internal data bus ID(11) line.
4. The HRESET value is a reset configuration word value, extracted from the internal data bus ID(12) line.
5. The HRESET value is a reset configuration word value, extracted from the internal data bus ID(17:18) lines.

WARNING

All SIUMCR fields which are controlled by the reset configuration word should not be changed by software while the corresponding functions are active.

Table 6-7 SIUMCR Bit Descriptions

Bit(s)	Name	Description
0	EARB	External arbitration 0 = Internal arbitration is performed 1 = External arbitration is assumed
1:3	EARP	External arbitration request priority. This field defines the priority of an external master's arbitration request. This field is valid when EARB is cleared. Refer to **9.5.6.4 Internal Bus Arbiter** for details.
4:7	—	Reserved

Table 6-7 SIUMCR Bit Descriptions (Continued)

Bit(s)	Name	Description
8	DSHW	Data show cycles. This bit selects the show cycle mode to be applied to U-bus data cycles (data cycles to IMB modules and flash EEPROM). This field is locked by the DLK bit. Note that instruction show cycles are programmed in the ICTRL and L-bus data show cycles (to SRAM) are programmed in the L2UMCR. 0 = Disable show cycles for all internal data cycles 1 = Show address and data of all internal data cycles
9:10	DBGC	Debug pins configuration. Refer to **Table 6-8**.
11	DBPC	Debug port pins configuration. Refer to **Table 6-9**.
12	ATWC	Address write type enable configuration. This bit configures the pins to function as byte write enables or address types for debugging purposes. 0 = $\overline{WE}$[0:3]/$\overline{BE}$[0:3]/AT[0:3] functions as $\overline{WE}$[0:3]/$\overline{BE}$[0:3][1] 1 = $\overline{WE}$[0:3]/$\overline{BE}$[0:3]/AT[0:3] functions as AT[0:3]
13:14	GPC	This bit configures the pins as shown in **Table 6-10.**
15	DLK	Debug register lock 0 = Normal operation 1 = SIUMCR is locked and can be written only in test mode or when the internal freeze signal is asserted.
16	—	Reserved
17:18	SC	Single-chip select. This field configures the functionality of the address and data buses. Changing the SC field while external accesses are performed is not supported. Refer to **Table 6-11**.
19	RCTX	Reset configuration/timer expired. During reset the RSTCONF/TEXP pin functions as RSTCONF. After reset the pin can be configured to function as TEXP, the timer expired signal that supports the low-power modes. 0 = RSTCONF/TEXP functions as RSTCONF 1 = RSTCONF/TEXP functions as TEXP
20:21	MLRC	Multi-level reservation control. This field selects between the functionality of the reservation logic and IRQ pins, refer to **Table 6-12**.
22:23	—	Reserved
24	MTSC	Memory transfer start control. 0 = $\overline{IRQ}$[2]/$\overline{CR}$/SGPIOC[2]/$\overline{MTS}$ functions according to the MLRC bits setting 1 = $\overline{IRQ}$[2]/$\overline{CR}$/SGPIOC[2]/$\overline{MTS}$ functions as $\overline{MTS}$
25	NOSHOW	Instruction show cycles disabled — If the NOSHOW bit is set (1), then all instruction show cycles are NOT transmitted to the external bus.
26	EICEN	Enhanced interrupt controller enabled — If the EICEN bit is set (1) then the enhanced interrupt controller operation is enabled. If it is cleared (0), then the regular interrupt controller operation is enabled.
27	LPMASK_EN	Low priority request masking enabled — If the LPMASK_EN bit is set (1) then the mechanism of masking lower priority interrupt requests is enabled. If it is cleared (0), then the lower priority interrupt requests are not masked automatically.
28:31	—	Reserved

NOTES:
1. $\overline{WE}$/$\overline{BE}$ is selected per memory region by WEBS in the appropriate BR register in the memory controller.

Table 6-8 Debug Pins Configuration

DBGC	Pin Function				
	IWP[0:1]/VFLS[0:1]	$\overline{BI}$/$\overline{STS}$	$\overline{BG}$/VF0/LWP1	$\overline{BR}$/VF1/IWP2	$\overline{BB}$/VF2/IWP3
00	VFLS[0:1]	$\overline{BI}$	$\overline{BG}$	$\overline{BR}$	$\overline{BB}$
01	IWP[0:1]	$\overline{STS}$	$\overline{BG}$	$\overline{BR}$	$\overline{BB}$
10	VFLS[0:1]	$\overline{STS}$	VF[0]	VF[1]	VF[2]
11	IWP[0:1]	$\overline{STS}$	LWP[1]	IWP[2]	IWP[3]

Table 6-9 Debug Port Pins Configuration

DBPC	Pin Function		
	TCK/DSCK	TDI/DSDI	TDO/DSDO
0	DSCK	DSDI	DSDO
1	TCK	TDI	TDO

Table 6-10 General Pins Configuration

GPC	Pin Function	
	FRZ/PTR/SGPIOC6	$\overline{IRQOUT}$/LWP0/SGPIOC7
00	PTR	$\overline{LWP0}$
01	SGPIOC[6]	SGPIOC[7]
10	FRZ	LWP[0]
11	FRZ	$\overline{IRQOUT}$

Table 6-11 Single-Chip Select Field Pin Configuration

SC	Pin Function		
	DATA[0:15]/ SGPIOD[0:15]	DATA[16:31] SGPIOD[16:31]	ADDR[8:31]/ SGPIOA[8:31]
00 (multiple chip, 32-bit port size)	DATA[0:15]	DATA[16:31]	ADDR[8:31]
01 (multiple chip, 16-bit port size	DATA[0:15]	SPGIOD[16:31]	ADDR[8:31]
10 (single-chip with address show cycles for debugging)	SPGIOD[0:15]	SPGIOD[16:31]	ADDR[8:31]
11 (single-chip)	SPGIOD[0:15]	SPGIOD[16:31]	SPGIOA[8:31]

Table 6-12 Multi-Level Reservation Control Pin Configuration

MLRC	Pin Function					
	$\overline{IRQ}$[0]/ SGPIOC[0]	**$\overline{IRQ}$[1]/$\overline{RSV}$/ SGPIOC[1]**	**$\overline{IRQ}$[2]/$\overline{CR}$/ SGPIOC[2]/ $\overline{MTS}$**	**$\overline{IRQ3}$/$\overline{KR}$/$\overline{RETRY}$ /SGPIOC3**	**$\overline{IRQ4}$/AT[2]/ SGPIOC[4]**	**$\overline{IRQ}$[5]/ SGPIOC[5]/ MODCK[1][1]**
00	$\overline{IRQ}$[0]	$\overline{IRQ}$[1]	$\overline{IRQ}$[2][2]	$\overline{IRQ}$[3]	$\overline{IRQ}$[4]	$\overline{IRQ}$[5] / MODCK[1]
01	$\overline{IRQ}$[0]	$\overline{RSV}$	$\overline{CR}$[2]	$\overline{KR}$/$\overline{RETRY}$	AT[2]	$\overline{IRQ}$[5] / MODCK[1]
10	SGPIOC0	SGPIOC[1]	SGPIOC[2][2]	SGPIOC[3]	SGPIOC[4]	SGPIOC[5]/ MODCK[1]
11	$\overline{IRQ}$[0]	$\overline{IRQ}$[1]	SGPIOC[2][2]	$\overline{KR}$/$\overline{RETRY}$	AT[2]	SGPIOC[5]/ MODCK[1]

NOTES:
1. Operates as MODCK[1] during reset.
2. This holds if MTSC bit is reset to 0. Otherwise $\overline{IRQ}$[2]/$\overline{CR}$/SGPIOC[2]/$\overline{MTS}$ will function as $\overline{MTS}$.

6.14.1.2 Internal Memory Map Register

The internal memory map register (IMMR) is a special register located within the MPC565 / MPC566 special register space. The IMMR contains identification of a specific device as well as the base for the internal memory map. Based on the value read from this register, software can deduce availability and location of any on-chip system resources.

This register can be read by the **mfspr** instruction. The ISB field can be written by the **mtspr** instruction. The PARTNUM and MASKNUM fields are mask programmed and cannot be changed.

IMMR — Internal Memory Mapping Register **SPR 638**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PARTNUM								MASKNUM							
RESET:															
Read-Only Fixed Value								Read-Only Fixed Value							

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED				FLEN	RESERVED		Re-served	RESERVED				ISB			0
RESET:															
0	0	0	0	ID20*	0	0	ID23*	0	0	0	0	ID[28:30]*			0

* The reset value is a reset configuration word value extracted from the indicated bits of the internal data bus. Refer to **7.5.2 Hard Reset Configuration Word**.

Table 6-13 IMMR Bit Descriptions

Bit(s)	Name	Description
0:7	PARTNUM	This read-only field is mask programmed with a code corresponding to the part number of the part on which the SIU is located. It is intended to help factory test and user code which is sensitive to part changes. This changes when the part number changes. For example, it would change if any new module is added, if the size of any memory module is changed. It would not change if the part is changed to fix a bug in an existing module. The MPC565 / MPC566 chip has an ID of 0x33.
8:15	MASKNUM	This read-only field is mask programmed with a code corresponding to the mask number of the part. It is intended to help factory test and user code which is sensitive to part changes.
16:19	—	Reserved
20	FLEN	Flash enable. The default state of FLEN is negated, meaning that the boot is performed from external memory. This bit can be set at reset by through the reset configuration word. 0 = On-chip flash memory is disabled, and all internal cycles to the allocated flash address space are mapped to external memory 1 = On-chip flash memory is enabled
21:22	—	Reserved
23	—	Reserved. This bit should be programmed to 0 at all times.
24:27	—	Reserved
28:30	ISB	This read-write field defines the base address of the internal memory space. The initial value of this field can be configured at reset to one of eight addresses, and then can be changed to any value by software. Internal base addresses are as follows: 000 = 0x0000 0000 001 = 0x0040 0000 010 = 0x0080 0000 011 = 0x00C0 0000 100 = 0x0100 0000 101 = 0x0140 0000 110 = 0x0180 0000 111 = 0x01C0 0000
31	—	Reserved

6.14.1.3 External Master Control Register (EMCR)

The external master control register selects the external master modes and determines the internal bus attributes for external-to-internal accesses.

EMCR — External Master Control Register **0x2F C030**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RESERVED															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
PRPM	SLVM	0	SIZE		SUPU	INST	RESERVED		RESV	CONT	0	TRAC	SIZEN	RESERVED	
RESET:															
ID16*	0	0	0	1	0	1	0	0	1	1	0	1	1	0	0

* The reset value is a reset configuration word value, extracted from the indicated internal data bus line. Refer to **7.5.2 Hard Reset Configuration Word**.

Table 6-14 EMCR Bit Descriptions

Bit(s)	Name	Description
0:15	—	Reserved
16	PRPM	Peripheral mode. In this mode, the internal RCPU core is shut off and an alternative master on the external bus can access any internal slave module. The reset value of this bit is determined by the reset configuration word bit 16. The bit can also be written by software. 0 = Normal operation 1 = Peripheral mode operation
17	SLVM	Slave mode (valid only if PRPM = 0). In this mode, an alternative master on the external bus can access any internal slave module while the internal RCPU core is fully operational. If PRPM is set, the value of SLVN is a “don’t care.” 0 = Normal operation 1 = Slave mode
18	—	Reserved
19:20	SIZE	Size attribute. If SIZEN = 1, the SIZE bits controls the internal bus attributes as follows: 00 = Double word (8 bytes) 01 = Word (4 bytes) 10 = Half word (2 bytes) 11 = Byte
21	SUPU	Supervisor/user attribute. SUPU controls the supervisor/user attribute as follows: 0 = Supervisor mode access permitted to all registers 1 = User access permitted to registers designated “user access”
22	INST	Instruction attribute. INST controls the internal bus instruction attribute as follows: 0 = Instruction fetch 1 = Operand or non-CPU access
23:24	—	Reserved
25	RESV	Reservation attribute. RESV controls the internal bus reservation attribute as follows: 0 = Storage reservation cycle 1 = Not a reservation
26	CONT	Control attribute. CONT drives the internal bus control bit attribute as follows: 0 = Access to MPC565 / MPC566 control register, or control cycle access 1 = Access to global address map
27	—	Reserved
28	TRAC	Trace attribute. TRAC controls the internal bus program trace attribute as follows: 0 = Program trace 1 = Not program trace
29	SIZEN	External size enable control bit. SIZEN determines how the internal bus size attribute is driven: 0 = Drive size from external bus signals TSIZE[0:1] 1 = Drive size from SIZE0, SIZE1 in EMCR
30:31	—	Reserved

6.14.2 SIU Interrupt Controller Registers

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

SIPEND, SIPEND2 and SIPEND3 are a 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 SIPEND and SIMASK registers are applicable in the regular operation mode while the SIPEND2,3 and SIMASK2,3 registers are implemented for the enhanced operation mode.

The bits associated with the $\overline{IRQ}$ pins have a different behavior depending on the sensitivity defined for them in the SIEL register. When the $\overline{IRQ}$ is defined as a “level” interrupt the corresponding bit behaves similar to the bits associated with internal interrupt sources, (i.e., it reflects the status of the $\overline{IRQ}$ pin). This bit can not be changed by software, it will be cleared when the external signal is negated. When the $\overline{IRQ}$ 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.

6.14.2.1 SIU Interrupt Pending Register

SIPEND — SIU Interrupt Pending Register **0x2F C010**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IRQ0	LVL0	IRQ1	LVL1	IRQ2	LVL2	IRQ3	LVL3	IRQ4	LVL4	IRQ5	LVL5	IRQ6	LVL6	IRQ7	LVL7
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.14.2.2 SIU Interrupt Pending Register 2

SIPEND2 — SIU Interrupt Pending Register2 **0x2F C040**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IRQ0	LVL0	IMB IRQ0	IMB IRQ1	IMB IRQ2	IMB IRQ3	IRQ1	LVL1	IMB IRQ4	IMB IRQ5	IMB IRQ6	IMB IRQ7	IRQ2	LVL2	IMB IRQ8	IMB IRQ9
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
IMB IRQ10	IMB IRQ11	IRQ3	LVL3	IMB IRQ12	IMB IRQ13	IMB IRQ14	IMB IRQ15	IRQ4	LVL4	IMB IRQ16	IMB IRQ17	IMB IRQ18	IMB IRQ19	IRQ5	LVL5
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.14.2.3 SIU Interrupt Pending Register 3

SIPEND3 — SIU Interrupt Pending Register3 **0x2F C044**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IMB IRQ20	IMB IRQ21	IMB IRQ22	IMB IRQ23	IRQ6	LVL6	IMB IRQ24	IMB IRQ25	IMB IRQ26	IMB IRQ27	IRQ7	LVL7	IMB IRQ28	IMB IRQ29	IMB IRQ30	IMB IRQ31
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
reserved															
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SIMASK is a 32 bit read/write register. Each bit in the register corresponds to an interrupt request bit in the SIPEND register.

SIMASK2 is a 32 bit read/write register. Each bit in the register corresponds to an interrupt request bit in the SIPEND2 register.

SIMASK3 is a 32 bit read/write register. Each bit in the register corresponds to an interrupt request bit in the SIPEND3 register.

When the bit is set, it enables the generation of an interrupt request to the CPU core. SIMASK, SIMASK2, SIMASK3 are updated by software and cleared upon reset. It is the responsibility of the software to determine which of the interrupt sources are enabled at a given time.

6.14.2.4 SIU Interrupt Mask Register

SIMASK — SIU Interrupt Mask Register **0x2F C014**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IRM0*	LVM0	IRM1	LVM1	IRM2	LVM2	IRM3	LVM3	IRM4	LVM4	IRM5	LVM5	IRM6	LVM6	IRM7	LVM7
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

*$\overline{\text{IRQ}}$[0] of the SIPEND register is not affected by the setting or clearing of the IRM0 bit of the SIMASK register. $\overline{\text{IRQ}}$[0] is a non-maskable interrupt.

6.14.2.5 SIU Interrupt Mask Register 2

SIMASK2 — SIU Interrupt Mask Register2 **0x2F C048**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IRQ0	LVL0	IMB IRQ0	IMB IRQ1	IMB IRQ2	IMB IRQ3	IRQ1	LVL1	IMB IRQ4	IMB IRQ5	IMB IRQ6	IMB IRQ7	IRQ2	LVL2	IMB IRQ8	IMB IRQ9
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
IMB IRQ10	IMB IRQ11	IRQ3	LVL3	IMB IRQ12	IMB IRQ13	IMB IRQ14	IMB IRQ15	IRQ4	LVL4	IMB IRQ16	IMB IRQ17	IMB IRQ18	IMB IRQ19	IRQ5	LVL5
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

*IRQ[0] of the SIPEND register is not affected by the setting or clearing of the IRM0 bit of the SIMASK register. IRQ[00 is a non-maskable interrupt.

6.14.2.6 SIU Interrupt Mask Register 3

SIMASK3 — SIU Interrupt Mask Register3 **0x2F C04C**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IMB IRQ20	IMB IRQ21	IMB IRQ22	IMB IRQ23	IRQ6	LVL6	IMB IRQ24	IMB IRQ25	IMB IRQ26	IMB IRQ27	IRQ7	LVL7	IMB IRQ28	IMB IRQ29	IMB IRQ30	IMB IRQ31
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.14.2.7 SIU Interrupt Edge Level Register (SIEL)

The SIEL is a 32-bit read/write register. Each pair of bits corresponds to an external interrupt request. The EDx bit, if set, specifies that a falling edge in the corresponding $\overline{\text{IRQ}}$ line will be detected as an interrupt request. When the EDx bit is 0, a low logical level in the $\overline{\text{IRQ}}$ line will be detected as an interrupt request. The WMx (wake-up mask) bit, if set, indicates that an interrupt request detection in the corresponding line causes the MPC565 / MPC566 to exit low-power mode.

SIEL — SIU Interrupt Edge Level Register **0x2F C018**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ED0	WM0	ED1	WM1	ED2	WM2	ED3	WM3	ED4	WM4	ED5	WM5	ED6	WM6	ED7	WM7
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.14.2.8 SIU Interrupt Vector Register

The SIVEC is a 32-bit read-only register that contains an 8-bit code representing the unmasked interrupt source of the highest priority level. The SIVEC can be read as either a byte, half word, or word. When read as a byte, a branch table can be used in which each entry contains one instruction (branch). When read as a half-word, each entry can contain a full routine of up to 256 instructions. The interrupt code is defined such that its two least significant bits are 0, thus allowing indexing into the table. The two possible ways of the code usage are shown on **Figure 6-11**

SIVEC — SIU Interrupt Vector Register **0x2F C01C**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
INTERRUPT CODE								RESERVED							
RESET:															
0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

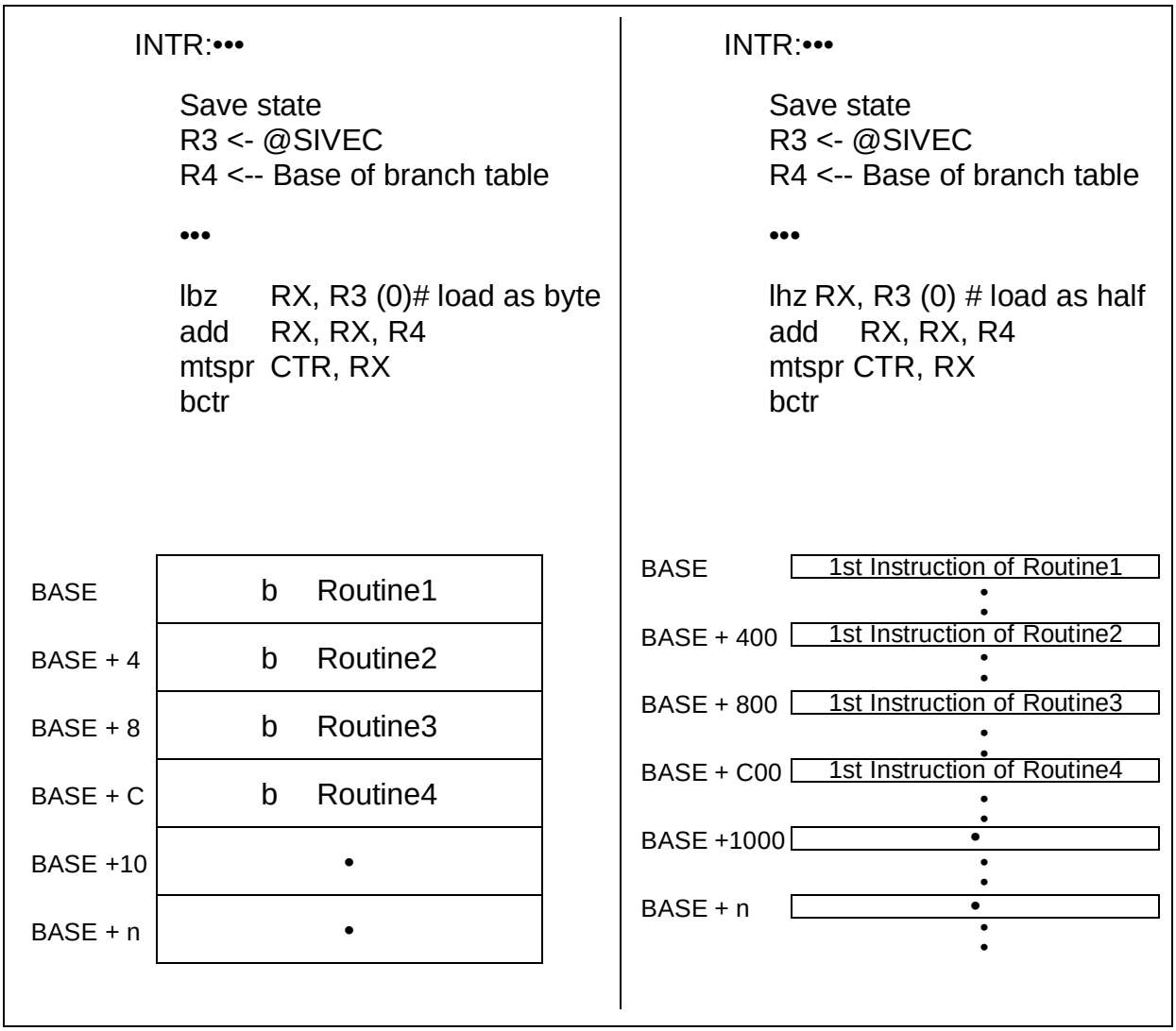


Figure 6-11 Example of SIVEC Register Usage for Interrupt Table Handling

6.14.2.9 Interrupt In-Service Registers

SISR2, SISR3 are 32 bit read/write registers. Each bit in the register corresponds to an interrupt request. The bit is set if:

- there is a pending interrupt request (SIPEND2/3), that is not masked by (SIMASK2/3), and
- the BBC/IMPU acknowledges interrupt request and latches SIVEC value.

Once a bit is set, all the requests with lower or equal priority become masked (i.e., — will not generate any interrupt request to the RCPU) until the bit is cleared by writing “1” by a software. Writing “0” has no effect.

SISR2 — Interrupt In-Service Register 2 **0x2F C050**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IRQ0	LVL0	IMB IRQ0	IMB IRQ1	IMB IRQ2	IMB IRQ3	IRQ1	LVL1	IMB IRQ4	IMB IRQ5	IMB IRQ6	IMB IRQ7	IRQ2	LVL2	IMB IRQ8	IMB IRQ9
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
IMB IRQ10	IMB IRQ11	IRQ3	LVL4	IMB IRQ12	IMB IRQ13	IMB IRQ14	IMB IRQ15	IRQ4	LVL4	IMB IRQ16	IMB IRQ17	IMB IRQ18	IMB IRQ19	IRQ5	LVL5
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SISR3 — Interrupt In-Service Register 3 **0x2F C054**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IMB IRQ20	IMB IRQ21	IMB IRQ22	IMB IRQ23	IRQ6	LVL6	IMB IRQ24	IMB IRQ25	IMB IRQ26	IMB IRQ27	IRQ7	LVL7	IMB IRQ28	IMB IRQ29	IMB IRQ30	IMB IRQ31
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
SRESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.14.3 System Protection Registers

6.14.3.1 System Protection Control Register (SYPCR)

The system protection control register (SYPCR) controls the system monitors, the software watchdog period, and the bus monitor timing. This register can be read at any time, but can be written only once after system reset.

SYPCR — System Protection Control Register **0x2F C004**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SWTC															
RESET:															
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
BMT								BME	RESERVED			SWF	SWE	SWRI	SWP
RESET:															
1	1	1	1	1	1	1	1	0	0	0	0	0	1	1	1

Table 6-15 SYPCR Bit Descriptions

Bit(s)	Name	Description
0:15	SWTC	Software watchdog timer count. This field contains the count value of the software watchdog timer.
16:23	BMT	Bus monitor timing. This field specifies the time-out period, in eight-system-clock resolution, of the bus monitor.
24	BME	Bus monitor enable 0 = Disable bus monitor 1 = Enable bus monitor
25:27	—	Reserved
28	SWF	Software watchdog freeze 0 = Software watchdog continues to run while FREEZE is asserted 1 = Software watchdog stops while FREEZE is asserted
29	SWE	Software watchdog enable. Software should clear this bit after a system reset to disable the SWT. 0 = Watchdog is disabled 1 = Watchdog is enabled
30	SWRI	Software watchdog reset/interrupt select 0 = Software watchdog time-out causes a non-maskable interrupt to the RCPU 1 = Software watchdog time-out causes a system reset
31	SWP	Software watchdog prescale 0 = Software watchdog timer is not pre-scaled 1 = Software watchdog timer is prescaled by 2048

6.14.3.2 Software Service Register (SWSR)

The SWSR is the location to which the SWT servicing sequence is written. To prevent SWT time-out, a 0x556C followed by 0xAA39 should be written to this register. The SWSR can be written at any time but returns all zeros when read.

SWSR — Software Service Register **0x2F C00E**

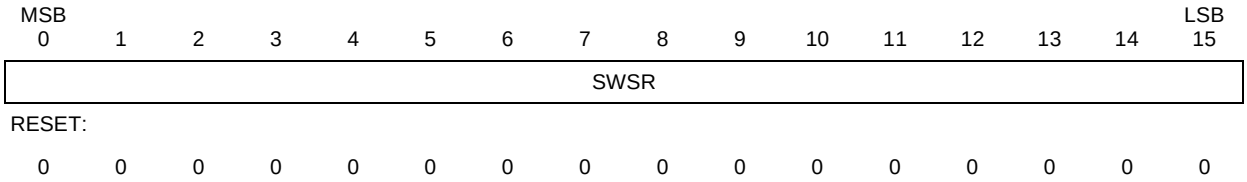


Table 6-16 SWSR Bit Descriptions

Bit(s)	Name	Description
0:15	SWSR	SWT servicing sequence is written to this register. To prevent SWT time-out, a 0x556C followed by 0xAA39 should be written to this register. The SWSR can be written at any time but returns all zeros when read.

6.14.3.3 Transfer Error Status Register (TESR)

The transfer error status register contains a bit for each exception source generated by a transfer error. A bit set to logic 1 indicates what type of transfer error exception occurred since the last time the bits were cleared by reset or by the normal software status bit-clearing mechanism.

NOTE

These bits may be set due to canceled speculative accesses which do not cause an interrupt. The register has two identical sets of bit fields; one is associated with instruction transfers and the other with data transfers.

TESR — Transfer Error Status Register **0x2F C020**

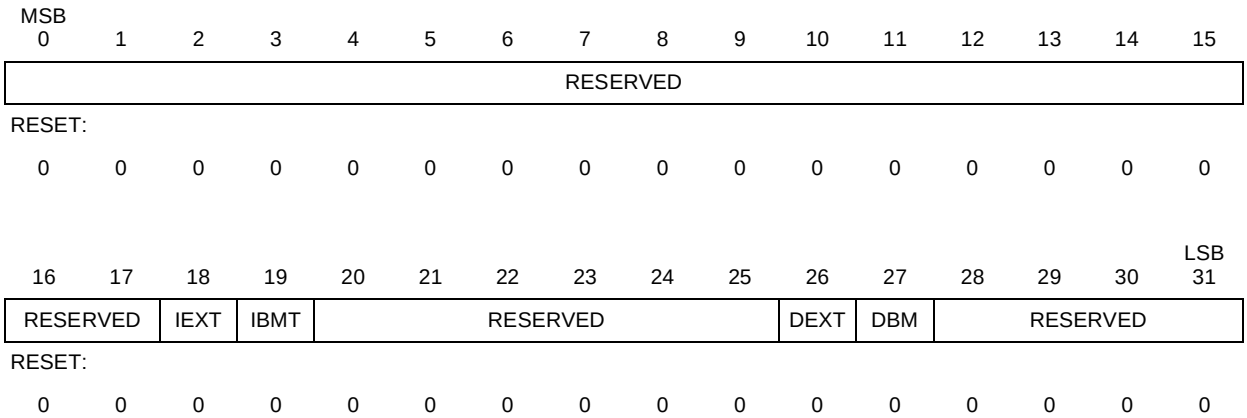


Table 6-17 TESR Bit Descriptions

Bit(s)	Name	Description
0:17	—	Reserved
18	IEXT	Instruction external transfer error acknowledge. This bit is set if the cycle was terminated by an externally generated $\overline{\text{TEA}}$ signal when an instruction fetch was initiated.

Table 6-17 TESR Bit Descriptions (Continued)

Bit(s)	Name	Description
19	IBMT	Instruction transfer monitor time out. This bit is set if the cycle was terminated by a bus monitor time-out when an instruction fetch was initiated.
20:25	—	Reserved
26	DEXT	Data external transfer error acknowledge. This bit is set if the cycle was terminated by an externally generated $\overline{\text{TEA}}$ signal when a data load or store is requested by an internal master.
27	DBM	Data transfer monitor time out. This bit is set if the cycle was terminated by a bus monitor time-out when a data load or store is requested by an internal master.
28:31	—	Reserved

6.14.4 System Timer Registers

The following sections describe registers associated with the system timers. These facilities are powered by the KAPWR and can preserve their value when the main power supply is off. Refer to **8.3.3 Pre-Divider** for details on the required actions needed in order to guarantee this data retention.

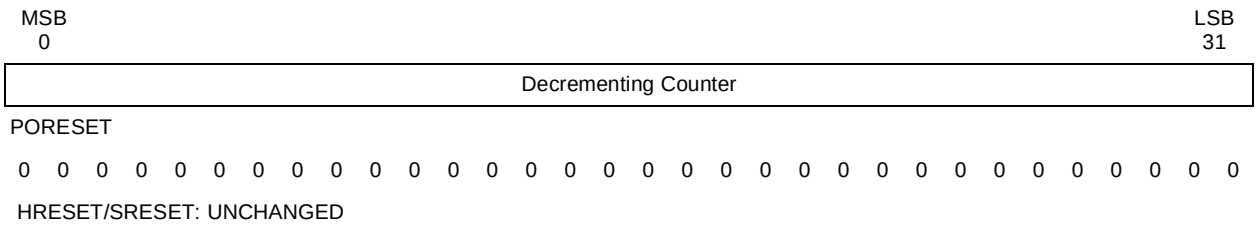
6.14.4.1 Decrementer Register

The 32-bit decrementer register is defined by the PowerPC architecture. The values stored in this register are used by a down counter to cause decrementer exceptions. The decrementer causes an exception whenever bit zero changes from a logic zero to a logic one. A read of this register always returns the current count value from the down counter.

Contents of this register can be read or written to by the **mfspr** or the **mtspr** instruction. The decrementer register is reset by $\overline{\text{PORESET}}$. $\overline{\text{HRESET}}$ and $\overline{\text{SRESET}}$ do not affect this register. The decrementer is powered by standby power and can continue to count when standby power is applied.

Decrementer counts down the time base clock and the counting is enabled by TBE bit in TBCSR register **6.14.4.4 Time Base Control and Status Register**

DEC — Decrementer Register **SPR 22**



Refer to **3.9.5 Decrementer Register (DEC)** for more information on this register.

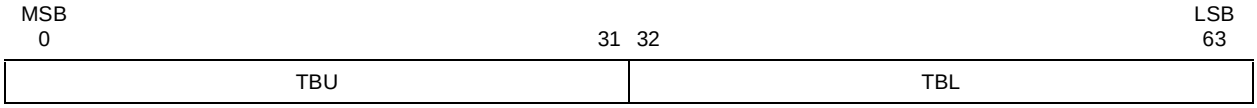
6.14.4.2 Time Base SPRs

The TB is a 64-bit register containing a 64-bit integer that is incremented periodically. There is no automatic initialization of the TB; the system software must perform this

initialization. The contents of the register may be written by the **mttbl** or the **mttbu** instructions, see **3.9.4 Time Base Facility (TB) — OEA**.

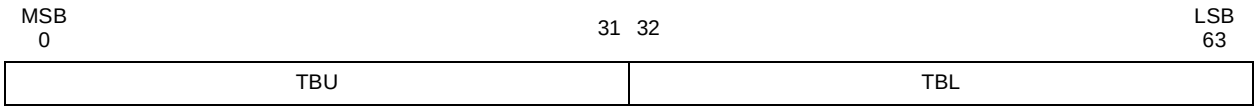
Refer to **3.8 PowerPC VEA Register Set — Time Base** and **3.9.4 Time Base Facility (TB) — OEA** for more information on reading and writing the TBU and TBL registers.

TB — Time Base (Reading) **SPR 268, 269**



RESET: UNCHANGED

TB — Time Base (Writing) **SPR 284, 285**



RESET: UNCHANGED

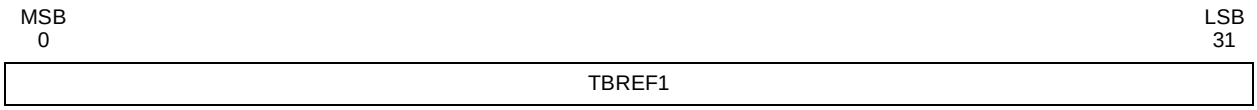
6.14.4.3 Time Base Reference Registers

Two reference registers (TBREF0 and TBREF1) are associated with the lower part of the time base (TBL). Each is a 32-bit read/write register. Upon a match between the contents of TBL and the reference register, a maskable interrupt is generated.

TBREF0 — Time Base Reference Register 0 **0x2F C204**

RESET:

TBREF1 — Time Base Reference Register 1 **0x2F C208**



RESET:

6.14.4.4 Time Base Control and Status Register

The TBSCR is 16-bit read/write register. It controls the TB, decrementer count enable, and interrupt generation and is used for reporting the source of the interrupts. The register can be read anytime. A status bit is cleared by writing a one to it. (Writing a zero has no effect.) More than one bit can be cleared at a time.

TBSCR — Time Base Control and Status Register **0x2F C200**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
TBIRQ								REFA	REFB	RESERVED		REFAE	REFBE	TBF	TBE
PORESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-18 TBSCR Bit Descriptions

Bit(s)	Name	Description
0:7	TBIRQ	Time base interrupt request. These bits determine the interrupt priority level of the time base. Refer to **6.4 Enhanced Interrupt Controller** for interrupt level encoding.
8	REFA	Reference A (TBREF0) interrupt status. 0 = No match detected 1 = TBREF0 value matches value in TBL
9	REFB	Reference B (TBREF1) interrupt status. 0 = No match detected 1 = TBREF1 value matches value in TBL
10:11	—	Reserved
12	REFAE	Reference A (TBREF0) interrupt enable. If this bit is set, the time base generates an interrupt when the REFA bit is set.
13	REFBE	Reference B (TBREF1) interrupt enable. If this bit is set, the time base generates an interrupt when the REFB bit is set.
14	TBF	Time base freeze. If this bit is set, the time base and decrementer stop while FREEZE is asserted.
15	TBE	Time base enable 0 = Time base and decrementer are disabled 1 = Time base and decrementer are enabled

6.14.4.5 Real-Time Clock Status and Control Register

The RTCSC is used to enable the different RTC functions and to report the source of the interrupts. The register can be read anytime. A status bit is cleared by writing it to a one. (Writing a zero does not affect a status bit's value.) More than one status bit can be cleared at a time. This register is locked after RESET. Unlocking is accomplished by writing 0x55CCAA33 to its associated key register. See **8.9.3.2 Keep-Alive Power Registers Lock Mechanism**.

RTCSC — Real-Time Clock Status and Control Register **0x2F C220**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
RTCIRQ								SEC	ALR	Re-served	4M	SIE	ALE	RTF	RTE
RESET:															
0	0	0	0	0	0	0	0	0	0	0	—	0	0	0	—

Table 6-19 RTCSC Bit Descriptions

Bit(s)	Name	Description
0:7	RTCIRQ	Real-time clock interrupt request. Thee bits determine the interrupt priority level of the RTC. Refer to **6.4 Enhanced Interrupt Controller** for interrupt level encoding.
8	SEC	Once per second interrupt. This status bit is set every second. It should be cleared by the software.
9	ALR	Alarm interrupt. This status bit is set when the value of the RTC equals the value programmed in the alarm register.
10	—	Reserved
11	4M	Real-time clock freeze 0 = RTC assumes that it is driven by 20 MHz to generate the seconds pulse. 1 = RTC assumes that it is driven by 4 MHz
12	SIE	Second interrupt enable. If this bit is set, the RTC generates an interrupt when the SEC bit is set.
13	ALE	Alarm interrupt enable. If this bit is set, the RTC generates an interrupt when the ALR bit is set.
14	RTF	Real-time clock freeze. If this bit is set, the RTC stops while FREEZE is asserted.
15	RTE	Real-time clock enable 0 = RTC is disabled 1 = RTC is enabled

6.14.4.6 Real-Time Clock Register (RTC)

The real-time clock register is a 32-bit read write register. It contains the current value of the real-time clock. A write to the RTC resets the seconds timer to zero. This register is locked after RESET. Unlocking is accomplished by writing 0x55CCAA33 to its associated key register. See **8.9.3.2 Keep-Alive Power Registers Lock Mechanism**.

RTC —Real-Time Clock Register **0x2F C224**

MSB 0		LSB 31
RTC		

RESET: UNCHANGED

6.14.4.7 Real-Time Clock Alarm Register (RTCAL)

The RTCAL is a 32-bit read/write register. When the value of the RTC is equal to the value programmed in the alarm register, a maskable interrupt is generated.

The alarm interrupt will be generated as soon as there is a match between the ALARM field and the corresponding bits in the RTC. The resolution of the alarm is 1 second. This register is locked after RESET. Unlocking is accomplished by writing 0x55CCAA33 to its associated key register. See **8.9.3.2 Keep-Alive Power Registers Lock Mechanism**.

RTCAL — Real-Time Clock Alarm Register **0x2F C22C**

MSB 0		LSB 31
ALARM		

RESET: UNCHANGED

6.14.4.8 Periodic Interrupt Status and Control Register (PISCR)

The PISCR contains the interrupt request level and the interrupt status bit. It also contains the controls for the 16 bits to be loaded into a modulus counter. This register can be read or written at any time.

PISCR — Periodic Interrupt Status and Control Register **0x2F C240**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB 15
PIRQ								PS	RESERVED				PIE	PITF	PTE
HARD RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-20 PISCR Bit Descriptions

Bit(s)	Name	Description
0:7	PIRQ	Periodic interrupt request. These bits determine the interrupt priority level of the PIT. Refer to **6.4 Enhanced Interrupt Controller** for interrupt level encoding.
8	PS	Periodic interrupt status. This bit is set if the PIT issues an interrupt. The PIT issues an interrupt after the modulus counter counts to zero. PS can be negated by writing a one to it. A write of zero has no affect.
9:12	—	Reserved
13	PIE	Periodic interrupt enable. If this bit is set, the time base generates an interrupt when the PS bit is set.
14	PITF	PIT freeze. If this bit is set, the PIT stops while FREEZE is asserted.
15	PTE	Periodic timer enable 0 = PIT stops counting and maintains current value 1 = PIT continues to decrement

6.14.4.9 Periodic Interrupt Timer Count Register (PITC)

The PITC register contains the 16 bits to be loaded in a modulus counter. This register is readable and writable at any time.

PITC — Periodic Interrupt Timer Count **0x2F C244**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PITC															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-21 PITC Bit Descriptions

Bit(s)	Name	Description
0:15	PITC	Periodic interrupt timing count. This field contains the 16-bit value to be loaded into the modulus counter that is loaded into the periodic timer. This register is readable and writable at any time.
16:31	—	Reserved

6.14.4.10 Periodic Interrupt Timer Register (PITR)

The periodic interrupt register is a read-only register that shows the current value in the periodic interrupt down counter. Read or writing this register does not affect the register.

PITR — Periodic Interrupt Timer Register **0x2F C248**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PIT															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
RESERVED															
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-22 PIT Bit Descriptions

Bit(s)	Name	Description
0:15	PIT	Periodic interrupt timing count — This field contains the current count remaining for the periodic timer. Writes have no effect on this field.
16:31	—	Reserved

6.14.5 General-Purpose I/O Registers

6.14.5.1 SGPIO Data Register 1 (SGPIODT1)

SGPIODT1 — SGPIO Data Register 1 **0x2F C024**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SGPIOD[0:7]								SGPIOD[8:15]							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
SGPIOD[16:23]								SGPIOD[24:31]							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-23 SGPIODT1 Bit Descriptions

Bit(s)	Name	Description
0:7	SGPIOD[0:7]	SIU general-purpose I/O Group D[0:7]. This 8-bit register controls the data of general-purpose I/O pins SGPIOD[0:7]. The direction (input or output) of this group of pins is controlled by the GDDR0 bit in the SGPIO control register.
8:15	SGPIOD[8:15]	SIU general-purpose I/O Group D[8:15]. This 8-bit register controls the data of general-purpose I/O pins SGPIOD[8:15]. The direction (input or output) of this group of pins is controlled by the GDDR1 bit in the SGPIO control register.
16:23	SGPIOD[16:23]	SIU general-purpose I/O Group D[16:23]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOD[16:23]. The direction (input or output) of this group of pins is controlled by the GDDR2 bit in the SGPIO control register
24:31	SGPIOD[24:31]	SIU general-purpose I/O Group D[24:31]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOD[24:31]. The direction of SGPIOD[24:31] is controlled by eight dedicated direction control signals SDDRD[24:31]. Each pin in this group can be configured separately as general-purpose input or output.

6.14.5.2 SGPIO Data Register 2 (SGPIODT2)

SGPIODT2 — SGPIO Data Register 2 **0x2F C028**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SGPIOC[0:7]								SGPIOA[8:15]							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
SGPIOA[16:23]								SGPIOA[24:31]							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-24 SGPIODT2 Bit Descriptions

Bit(s)	Name	Description
0:7	SGPIOC[0:7]	SIU general-purpose I/O Group C[0:7]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOC[0:7]. The direction of SGPIOC[0:7] is controlled by 8 dedicated direction control signals SDDRC[0:7] in the SGPIO control register. Each pin in this group can be configured separately as general-purpose input or output.
8:15	SGPIOA[8:15]	SIU general-purpose I/O Group A[8:15]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOA[8:15]. The GDDR3 bit in the SGPIO control register configures these pins as a group as general-purpose input or output.
16:23	SGPIOA [16:23]	SIU general-purpose I/O Group A[16:23]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOA[16:23]. The GDDR4 bit in the SGPIO control register configures these pins as a group as general-purpose input or output.
24:31	SGPIOA [24:31]]	SIU general-purpose I/O Group A[24:31]. This 8-bit register controls the data of the general-purpose I/O pins SGPIOA[24:31]. The GDDR5 bit in the SGPIO control register configures these pins as a group as general-purpose input or output.

6.14.5.3 SGPIO Control Register (SGPIOCR)

SGPIOCR — SGPIO Control Register **0x2F C02C**

MSB 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SDDRC[0:7]								RESERVED							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	LSB 31
GDDR 0	GDDR 1	GDDR 2	GDDR 3	GDDR 4	GDDR 5	RESERVED		SDDRD[24:31]							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 6-25 SGPIOCR Bit Descriptions

Bit(s)	Name	Description
0:7	SDDRC[0:7]	SGPIO data direction for SGPIOC[0:7]. Each SDDR bit zero to seven controls the direction of the corresponding SGPIOC pin zero to seven
8:15	—	Reserved
16	GDDR0	Group data direction for SGPIOD[0:7]
17	GDDR1	Group data direction for SGPIOD[8:15]

Table 6-25 SGPIOCR Bit Descriptions (Continued)

Bit(s)	Name	Description
18	GDDR2	Group data direction for SGPIOD[16:23]
19	GDDR3	Group data direction for SGPIOA[8:15]
20	GDDR4	Group data direction for SGPIOA[16:23]
21	GDDR5	Group data direction for SGPIOA[24:31]
22:23	—	Reserved
24:31	SDDRD [24:31]	SGPIO data direction for SGPIOD[24:31]. Each SDDRD bit 24:31 controls the direction of the corresponding SGPIOD pin [24:31].

Table 6-26 describes the bit values for data direction control.

Table 6-26 Data Direction Control

SDDR/GDDR	Operation
0	SGPIO configured as input
1	SGPIO configured as output