## INTERRUPT CONTROLLER SPECIAL REGISTERS

### INTERRUPT CONTROL REGISTER (INTCON)

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| INTCON | 0x01E00000 | R/W | Interrupt control Register | 0x7 |

| INTCON | Bit | Description | initial state |
|---|---|---|---|
| Reserved | [3] | 0 | 0 |
| V | [2] | This bit disables/enables vector mode for IRQ<br>0 = Vectored interrupt mode<br>1 = Non-vectored interrupt mode | 1 |
| I | [1] | This bit enables IRQ interrupt request line to CPU<br>0 = IRQ interrupt enable<br>1 = Reserved<br>**Note : Before using the IRQ interrupt this bit must be cleared.** | 1 |
| F | [0] | This bit enables FIQ interrupt request line to CPU<br>0 = FIQ interrupt enable (Not allowed vectored interrupt mode)<br>1 = Reserved<br>**Note : Before using the FIQ interrupt this bit must be cleared.** | 1 |

**NOTE:** FIQ interrupt mode does not support vectored interrupt mode.

### INTERRUPT PENDING REGISTER (INTPND)

Each of the 26 bits in the interrupt pending register, INTPND, corresponds to an interrupt source. When an interrupt request is generated, it will be set to 1. The interrupt service routine must then clear the pending condition by writing '1' to the corresponding bit of I_ISPC/F_ISPC. Although several interrupt sources generate requests simultaneously, the INTPND will indicate all interrupt sources that generate an interrupt request. Even if the interrupt source is masked by INTMSK, the corresponding pending bit can be set to 1.

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| INTPND | 0x01E00004 | R | Indicates the interrupt request status.<br>0 = The interrupt has not been requested<br>1 = The interrupt source has asserted the interrupt request | 0x0000000 |

| INTPND | Bit | Description | Initial State |
|---|---|---|---|
| EINT0 | [25] | 0 = Not requested, 1 = Requested | 0 |
| EINT1 | [24] | 0 = Not requested, 1 = Requested | 0 |
| EINT2 | [23] | 0 = Not requested, 1 = Requested | 0 |
| EINT3 | [22] | 0 = Not requested, 1 = Requested | 0 |
| EINT4/5/6/7 | [21] | 0 = Not requested, 1 = Requested | 0 |
| INT_TICK | [20] | 0 = Not requested, 1 = Requested | 0 |
| INT_ZDMA0 | [19] | 0 = Not requested, 1 = Requested | 0 |
| INT_ZDMA1 | [18] | 0 = Not requested, 1 = Requested | 0 |
| INT_BDMA0 | [17] | 0 = Not requested, 1 = Requested | 0 |
| INT_BDMA1 | [16] | 0 = Not requested, 1 = Requested | 0 |
| INT_WDT | [15] | 0 = Not requested, 1 = Requested | 0 |
| INT_UERR0/1 | [14] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER0 | [13] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER1 | [12] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER2 | [11] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER3 | [10] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER4 | [9] | 0 = Not requested, 1 = Requested | 0 |
| INT_TIMER5 | [8] | 0 = Not requested, 1 = Requested | 0 |
| INT_URXD0 | [7] | 0 = Not requested, 1 = Requested | 0 |
| INT_URXD1 | [6] | 0 = Not requested, 1 = Requested | 0 |
| INT_IIC | [5] | 0 = Not requested, 1 = Requested | 0 |
| INT_SIO | [4] | 0 = Not requested, 1 = Requested | 0 |
| INT_UTXD0 | [3] | 0 = Not requested, 1 = Requested | 0 |
| INT_UTXD1 | [2] | 0 = Not requested, 1 = Requested | 0 |
| INT_RTC | [1] | 0 = Not requested, 1 = Requested | 0 |

### INTERRUPT MODE REGISTER (INTMOD)

Each of the 26 bits in the interrupt mode register, INTMOD, corresponds to an interrupt source. When the interrupt mode bit for each source is set to 1, the interrupt is processed by the ARM7TDMI core in the FIQ (fast interrupt) mode. Otherwise, it is processed in the IRQ mode (normal interrupt). The 26 interrupt sources are summarized as follows:

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| INTMOD | 0x01E00008 | R/W | Interrupt mode Register<br>0 = IRQ mode 1 = FIQ mode | 0x0000000 |

| INTMOD | Bit | Description | initial state |
|---|---|---|---|
| EINT0 | [25] | 0 = IRQ mode 1 = FIQ mode | 0 |
| EINT1 | [24] | 0 = IRQ mode 1 = FIQ mode | 0 |
| EINT2 | [23] | 0 = IRQ mode 1 = FIQ mode | 0 |
| EINT3 | [22] | 0 = IRQ mode 1 = FIQ mode | 0 |
| EINT4/5/6/7 | [21] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TICK | [20] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_ZDMA0 | [19] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_ZDMA1 | [18] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_BDMA0 | [17] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_BDMA1 | [16] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_WDT | [15] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_UERR0/1 | [14] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER0 | [13] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER1 | [12] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER2 | [11] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER3 | [10] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER4 | [9] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_TIMER5 | [8] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_URXD0 | [7] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_URXD1 | [6] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_IIC | [5] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_SIO | [4] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_UTXD0 | [3] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_UTXD1 | [2] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_RTC | [1] | 0 = IRQ mode 1 = FIQ mode | 0 |
| INT_ADC | [0] | 0 = IRQ mode 1 = FIQ mode | 0 |

### INTERRUPT MASK REGISTER (INTMSK)

Each of the 26 bits except the global mask bit in the interrupt mask register, INTMSK, corresponds to an interrupt source. When a source interrupt mask bit is 1 and the corresponding interrupt event occurs, the interrupt is not serviced by the CPU. If the mask bit is 0, the interrupt is serviced upon a request.

If the global mask bit is set to 1, all interrupt requests are not serviced, and the INTPND register is set to 1.

If the INTMSK is changed in ISR(interrupt service routine) and the vectored interrupt is used, an INTMSK bit can not mask an interrupt event, which had been latched in INTPND before the INTMSK bit was set. To clear this problem, clear the corresponding pending bit(INTPND) after changing INTMSK.

The 26 interrupt sources and global mask bit are summarized as follows:

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| INTMSK | 0x01E0000C | R/W | Determines which interrupt source is masked. The masked interrupt source will not be serviced.<br>0 = Interrupt service is available<br>1 = Interrupt service is masked | 0x07fffffff |

**IMPORTANT NOTES**

1. INTMSK register can be masked only when it is sure that the corresponding interrupt does not be requested. If your application should mask any interrupt mask bit(INTMSK) just when the corresponding interrupt is issued, please contact our FAE (field application engineer).
2. If you need that all interrupt is masked, we recommend that I/F bits in CPSR are set using MRS, MSR instructions. The I, F bit in CPSR can be masked even when any interrupt is issued.

| INTMSK | Bit | Description | initial state |
|---|---|---|---|
| Reserved | [27] | | 0 |
| Global | [26] | 0 = Service available 1 = Masked | 1 |
| EINT0 | [25] | 0 = Service available 1 = Masked | 1 |
| EINT1 | [24] | 0 = Service available 1 = Masked | 1 |
| EINT2 | [23] | 0 = Service available 1 = Masked | 1 |
| EINT3 | [22] | 0 = Service available 1 = Masked | 1 |
| EINT4/5/6/7 | [21] | 0 = Service available 1 = Masked | 1 |
| INT_TICK | [20] | 0 = Service available 1 = Masked | 1 |
| INT_ZDMA0 | [19] | 0 = Service available 1 = Masked | 1 |
| INT_ZDMA1 | [18] | 0 = Service available 1 = Masked | 1 |
| INT_BDMA0 | [17] | 0 = Service available 1 = Masked | 1 |
| INT_BDMA1 | [16] | 0 = Service available 1 = Masked | 1 |
| INT_WDT | [15] | 0 = Service available 1 = Masked | 1 |
| INT_UERR0/1 | [14] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER0 | [13] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER1 | [12] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER2 | [11] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER3 | [10] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER4 | [9] | 0 = Service available 1 = Masked | 1 |
| INT_TIMER5 | [8] | 0 = Service available 1 = Masked | 1 |
| INT_URXD0 | [7] | 0 = Service available 1 = Masked | 1 |
| INT_URXD1 | [6] | 0 = Service available 1 = Masked | 1 |
| INT_IIC | [5] | 0 = Service available 1 = Masked | 1 |
| INT_SIO | [4] | 0 = Service available 1 = Masked | 1 |
| INT_UTXD0 | [3] | 0 = Service available 1 = Masked | 1 |
| INT_UTXD1 | [2] | 0 = Service available 1 = Masked | 1 |
| INT_RTC | [1] | 0 = Service available 1 = Masked | 1 |
| INT_ADC | [0] | 0 = Service available 1 = Masked | 1 |

### IRQ INTERRUPT SERVICE PENDING REGISTER (I_ISPR)

I_ISPR indicates the interrupt being currently serviced. Although the several interrupt pending bits are all turned on, only one bit will be turned on.

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| I_ISPR | 0x01E00020 | R | IRQ interrupt service pending register | 0x00000000 |

| I_ISPR | Bit | Description | Initial State |
|---|---|---|---|
| EINT0 | [25] | 0 = not serviced 1 = serviced now | 0 |
| EINT1 | [24] | 0 = not serviced 1 = serviced now | 0 |
| EINT2 | [23] | 0 = not serviced 1 = serviced now | 0 |
| EINT3 | [22] | 0 = not serviced 1 = serviced now | 0 |
| EINT4/5/6/7 | [21] | 0 = not serviced 1 = serviced now | 0 |
| INT_TICK | [20] | 0 = not serviced 1 = serviced now | 0 |
| INT_ZDMA0 | [19] | 0 = not serviced 1 = serviced now | 0 |
| INT_ZDMA1 | [18] | 0 = not serviced 1 = serviced now | 0 |
| INT_BDMA0 | [17] | 0 = not serviced 1 = serviced now | 0 |
| INT_BDMA1 | [16] | 0 = not serviced 1 = serviced now | 0 |
| INT_WDT | [15] | 0 = not serviced 1 = serviced now | 0 |
| INT_UERR0/1 | [14] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER0 | [13] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER1 | [12] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER2 | [11] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER3 | [10] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER4 | [9] | 0 = not serviced 1 = serviced now | 0 |
| INT_TIMER5 | [8] | 0 = not serviced 1 = serviced now | 0 |
| INT_URXD0 | [7] | 0 = not serviced 1 = serviced now | 0 |
| INT_URXD1 | [6] | 0 = not serviced 1 = serviced now | 0 |
| INT_IIC | [5] | 0 = not serviced 1 = serviced now | 0 |
| INT_SIO | [4] | 0 = not serviced 1 = serviced now | 0 |
| INT_UTXD0 | [3] | 0 = not serviced 1 = serviced now | 0 |
| INT_UTXD1 | [2] | 0 = not serviced 1 = serviced now | 0 |
| INT_RTC | [1] | 0 = not serviced 1 = serviced now | 0 |
| INT_ADC | [0] | 0 = not serviced 1 = serviced now | 0 |

### IRQ/FIQ INTERRUPT SERVICE PENDING CLEAR REGISTER (I_ISPC/F_ISPC)

I_ISPC/F_ISPC clears the interrupt pending bit (INTPND). I_ISPC/F_ISPC also informs the interrupt controller of the end of corresponding ISR (interrupt service routine). At the end of ISR(interrupt service routine), the corresponding pending bit must be cleared.

The bit of INTPND bit is cleared to zero by writing '1' on I_ISPC/F_ISPC. This feature reduces the code size to clear the INTPND. The corresponding INTPND bit is cleared automatically by I_ISPC/F_ISPC, INTPND register can not be cleared directly.

**NOTE**

To clear the I_ISPC/F_ISPC, the following two rules has to be obeyed.

1) The I_ISPC/F_ISPC registers are accessed only once in ISR(interrupt service routine).
2) The pending bit in I_ISPR/INTPND register should be cleared by writing I_ISPC register.

If these two rules are not followed, I_ISPR and INTPND register may be 0 although the interrupt has been requested.

| Register | Address | R/W | Description | Reset Value |
|---|---|---|---|---|
| I_ISPC | 0x01E00024 | W | IRQ interrupt service pending clear register | Undef. |
| F_ISPC | 0x01E0003C | W | FIQ interrupt service pending clear register | Undef. |

| I_ISPC/F_ISPC | Bit | Description | | Initial State |
|---|---|---|---|---|
| EINT0 | [25] | 0 = No change | 1 = clear the pending bit | 0 |
| EINT1 | [24] | 0 = No change | 1 = clear the pending bit | 0 |
| EINT2 | [23] | 0 = No change | 1 = clear the pending bit | 0 |
| EINT3 | [22] | 0 = No change | 1 = clear the pending bit | 0 |
| EINT4/5/6/7 | [21] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TICK | [20] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_ZDMA0 | [19] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_ZDMA1 | [18] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_BDMA0 | [17] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_BDMA1 | [16] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_WDT | [15] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_UERR0/1 | [14] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER0 | [13] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER1 | [12] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER2 | [11] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER3 | [10] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER4 | [9] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_TIMER5 | [8] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_URXD0 | [7] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_URXD1 | [6] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_IIC | [5] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_SIO | [4] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_UTXD0 | [3] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_UTXD1 | [2] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_RTC | [1] | 0 = No change | 1 = clear the pending bit | 0 |
| INT_ADC | [0] | 0 = No change | 1 = clear the pending bit | 0 |