

direction bit is set, the GPIO is configured as an output, and the corresponding data register bit is driven out on the GPIO port.

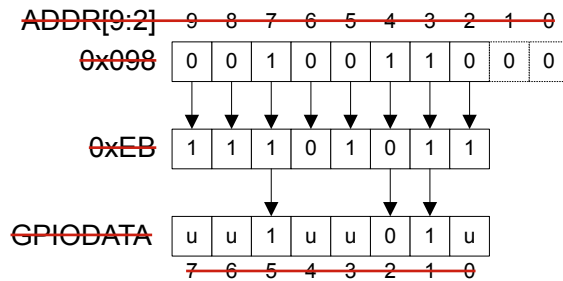
10.2.1.2 Data Register Operation

To aid in the efficiency of software, the GPIO ports allow for the modification of individual bits in the **GPIO Data (GPIODATA)** register (see page 659) by using bits [9:2] of the address bus as a mask. In this manner, software drivers can modify individual GPIO pins in a single instruction without affecting the state of the other pins. This method is more efficient than the conventional method of performing a read-modify-write operation to set or clear an individual GPIO pin. To implement this feature, the **GPIODATA** register covers 256 locations in the memory map.

During a write, if the address bit associated with that data bit is set, the value of the **GPIODATA** register is altered. If the address bit is cleared, the data bit is left unchanged.

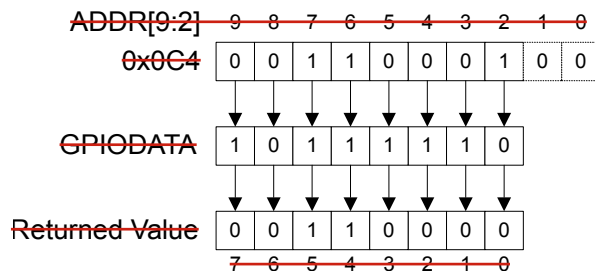
For example, writing a value of 0xEB to the address **GPIODATA + 0x098** has the results shown in Figure 10-3, where u indicates that data is unchanged by the write. This example demonstrates how **GPIODATA** bits 5, 2, and 1 are written.

Figure 10-3. GPIODATA Write Example



During a read, if the address bit associated with the data bit is set, the value is read. If the address bit associated with the data bit is cleared, the data bit is read as a zero, regardless of its actual value. For example, reading address **GPIODATA + 0x0C4** yields as shown in Figure 10-4. This example shows how to read **GPIODATA** bits 5, 4, and 0.

Figure 10-4. GPIODATA Read Example



10.2.2 Interrupt Control

The interrupt capabilities of each GPIO port are controlled by a set of seven registers. These registers are used to select the source of the interrupt, its polarity, and the edge properties. When one or more GPIO inputs cause an interrupt, a single interrupt output is sent to the interrupt controller for the entire GPIO port. For edge-triggered interrupts, software must clear the interrupt to enable any

further interrupts. For a level-sensitive interrupt, the external source must hold the level constant for the interrupt to be recognized by the controller.

Three registers define the edge or sense that causes interrupts:

- **GPIO Interrupt Sense (GPIOIS)** register (see page 661)
- **GPIO Interrupt Both Edges (GPIOIBE)** register (see page 662)
- **GPIO Interrupt Event (GPIOIEV)** register (see page 663)

Interrupts are enabled/disabled via the **GPIO Interrupt Mask (GPIOIM)** register (see page 664).

When an interrupt condition occurs, the state of the interrupt signal can be viewed in two locations: the **GPIO Raw Interrupt Status (GPIORIS)** and **GPIO Masked Interrupt Status (GPIOMIS)** registers (see page 665 and page 666). As the name implies, the **GPIOMIS** register only shows interrupt conditions that are allowed to be passed to the interrupt controller. The **GPIORIS** register indicates that a GPIO pin meets the conditions for an interrupt, but has not necessarily been sent to the interrupt controller.

For a GPIO level-detect interrupt, the interrupt signal generating the interrupt must be held until serviced. Once the input signal deasserts from the interrupt generating logical sense, the corresponding **RIS** bit in the **GPIORIS** register clears. For a GPIO edge-detect interrupt, the **RIS** bit in the **GPIORIS** register is cleared by writing a '1' to the corresponding bit in the **GPIO Interrupt Clear (GPIOICR)** register (see page 667). The corresponding **GPIOMIS** bit reflects the masked value of the **RIS** bit.

When programming the interrupt control registers (**GPIOIS**, **GPIOIBE**, or **GPIOIEV**), the interrupts should be masked (**GPIOIM** cleared). Writing any value to an interrupt control register can generate a spurious interrupt if the corresponding bits are enabled.

10.2.2.1 ~~ADC Trigger Source~~

~~Any GPIO pin can be configured to be an external trigger for the ADC using the **GPIO ADC Control (GPIOADCCTL)** register. If any GPIO is configured as a non-masked interrupt pin (the appropriate bit of **GPIOIM** is set), and an interrupt for that port is generated, a trigger signal is sent to the ADC. If the **ADC Event Multiplexer Select (ADCEMUX)** register is configured to use the external trigger, an ADC conversion is initiated. See page 830.~~

~~Note that if the Port B **GPIOADCCTL** register is cleared, PB4 can still be used as an external trigger for the ADC. This is a legacy mode which allows code written for previous devices to operate on this microcontroller.~~

10.2.2.2 ~~μDMA Trigger Source~~

~~Any GPIO pin can be configured to be an external trigger for the μDMA using the **GPIO DMA Control (GPIODMACTL)** register. If any GPIO is configured as a non-masked interrupt pin (the appropriate bit of **GPIOIM** is set), an interrupt for that port is generated and an external trigger signal is sent to the μDMA. If the μDMA is configured to start a transfer based on the GPIO signal, a transfer is initiated.~~

10.2.3 ~~Mode Control~~

~~The GPIO pins can be controlled by either software or hardware. Software control is the default for most signals and corresponds to the GPIO mode, where the **GPIODATA** register is used to read or write the corresponding pins. When hardware control is enabled via the **GPIO Alternate Function**~~

Register 3: GPIO Interrupt Sense (GPIOIS), offset 0x404

The **GPIOIS** register is the interrupt sense register. Setting a bit in the **GPIOIS** register configures the corresponding pin to detect levels, while clearing a bit configures the corresponding pin to detect edges. All bits are cleared by a reset.

GPIO Interrupt Sense (GPIOIS)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x404

Type R/W, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | IS | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-----------|---|
| 31:8 | reserved | RO | 0x0000.00 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | IS | R/W | 0x00 | GPIO Interrupt Sense |
| | | | | Value Description |
| | | | | 0 The edge on the corresponding pin is detected (edge-sensitive). |
| | | | | 1 The level on the corresponding pin is detected (level-sensitive). |

Register 4: GPIO Interrupt Both Edges (GPIOIBE), offset 0x408

The **GPIOIBE** register allows both edges to cause interrupts. When the corresponding bit in the **GPIO Interrupt Sense (GPIOIS)** register (see page 661) is set to detect edges, setting a bit in the **GPIOIBE** register configures the corresponding pin to detect both rising and falling edges, regardless of the corresponding bit in the **GPIO Interrupt Event (GPIOIEV)** register (see page 663). Clearing a bit configures the pin to be controlled by the **GPIOIEV** register. All bits are cleared by a reset.

GPIO Interrupt Both Edges (GPIOIBE)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x408
 Type R/W, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | IBE | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-------------------|--|------|-----------|---|
| 31:8 | reserved | RO | 0x0000.00 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | IBE | R/W | 0x00 | GPIO Interrupt Both Edges |
| Value Description | | | | |
| 0 | Interrupt generation is controlled by the GPIO Interrupt Event (GPIOIEV) register (see page 663). | | | |
| 1 | Both edges on the corresponding pin trigger an interrupt. | | | |

Register 5: GPIO Interrupt Event (GPIOIEV), offset 0x40C

The **GPIOIEV** register is the interrupt event register. Setting a bit in the **GPIOIEV** register configures the corresponding pin to detect rising edges or high levels, depending on the corresponding bit value in the **GPIO Interrupt Sense (GPIOIS)** register (see page 661). Clearing a bit configures the pin to detect falling edges or low levels, depending on the corresponding bit value in the **GPIOIS** register. All bits are cleared by a reset.

GPIO Interrupt Event (GPIOIEV)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x40C
 Type R/W, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | IEV | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-----------|---|
| 31:8 | reserved | RO | 0x0000.00 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | IEV | R/W | 0x00 | GPIO Interrupt Event |
| | | | | Value Description |
| | | | | 0 A falling edge or a Low level on the corresponding pin triggers an interrupt. |
| | | | | 1 A rising edge or a High level on the corresponding pin triggers an interrupt. |

Register 6: GPIO Interrupt Mask (GPIOIM), offset 0x410

The **GPIOIM** register is the interrupt mask register. Setting a bit in the **GPIOIM** register allows interrupts that are generated by the corresponding pin to be sent to the interrupt controller on the combined interrupt signal. Clearing a bit prevents an interrupt on the corresponding pin from being sent to the interrupt controller. All bits are cleared by a reset.

GPIO Interrupt Mask (GPIOIM)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x410
 Type R/W, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | IME | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-------|---|
| 31:8 | reserved | RO | 0 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | IME | R/W | 0x00 | GPIO Interrupt Mask Enable |

Value Description

| | |
|---|---|
| 0 | The interrupt from the corresponding pin is masked. |
| 1 | The interrupt from the corresponding pin is sent to the interrupt controller. |

Register 7: GPIO Raw Interrupt Status (GPIORIS), offset 0x414

The **GPIORIS** register is the raw interrupt status register. A bit in this register is set when an interrupt condition occurs on the corresponding GPIO pin. If the corresponding bit in the **GPIO Interrupt Mask (GPIOIM)** register (see page 664) is set, the interrupt is sent to the interrupt controller. Bits read as zero indicate that corresponding input pins have not initiated an interrupt. For a GPIO level-detect interrupt, the interrupt signal generating the interrupt must be held until serviced. Once the input signal deasserts from the interrupt generating logical sense, the corresponding **RIS** bit in the **GPIORIS** register clears. For a GPIO edge-detect interrupt, the **RIS** bit in the **GPIORIS** register is cleared by writing a '1' to the corresponding bit in the **GPIO Interrupt Clear (GPIOICR)** register. The corresponding **GPIOMIS** bit reflects the masked value of the **RIS** bit.

GPIO Raw Interrupt Status (GPIORIS)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x414
 Type RO, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | RIS | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-------|---|
| 31:8 | reserved | RO | 0 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | RIS | RO | 0x00 | GPIO Interrupt Raw Status |
| | | | | Value Description |
| | | | | 0 An interrupt condition has not occurred on the corresponding pin. |
| | | | | 1 An interrupt condition has occurred on the corresponding pin. |
| | | | | For edge-detect interrupts, this bit is cleared by writing a 1 to the corresponding bit in the GPIOICR register. |
| | | | | For a GPIO level-detect interrupt, the bit is cleared when the level is deasserted. |

Register 8: GPIO Masked Interrupt Status (GPIOMIS), offset 0x418

The **GPIOMIS** register is the masked interrupt status register. If a bit is set in this register, the corresponding interrupt has triggered an interrupt to the interrupt controller. If a bit is clear, either no interrupt has been generated, or the interrupt is masked.

Note that if the Port B **GPIOADCCTL** register is cleared, PB4 can still be used as an external trigger for the ADC. This is a legacy mode which allows code written for previous devices to operate on this microcontroller.

GPIOMIS is the state of the interrupt after masking.

GPIO Masked Interrupt Status (GPIOMIS)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x418
 Type RO, reset 0x0000.0000

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|-------|----------|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | MIS | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-------|---|
| 31:8 | reserved | RO | 0 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | MIS | RO | 0x00 | GPIO Masked Interrupt Status |

Value Description

| | |
|---|---|
| 0 | An interrupt condition on the corresponding pin is masked or has not occurred. |
| 1 | An interrupt condition on the corresponding pin has triggered an interrupt to the interrupt controller. |

For edge-detect interrupts, this bit is cleared by writing a 1 to the corresponding bit in the **GPIOICR** register.

For a GPIO level-detect interrupt, the bit is cleared when the level is deasserted.

Register 9: GPIO Interrupt Clear (GPIOICR), offset 0x41C

The **GPIOICR** register is the interrupt clear register. For edge-detect interrupts, writing a 1 to the **IC** bit in the **GPIOICR** register clears the corresponding bit in the **GPIOIRIS** and **GPIOMIS** registers. If the interrupt is a level-detect, the **IC** bit in this register has no effect. In addition, writing a 0 to any of the bits in the **GPIOICR** register has no effect.

GPIO Interrupt Clear (GPIOICR)

GPIO Port A (APB) base: 0x4000.4000
 GPIO Port A (AHB) base: 0x4005.8000
 GPIO Port B (APB) base: 0x4000.5000
 GPIO Port B (AHB) base: 0x4005.9000
 GPIO Port C (APB) base: 0x4000.6000
 GPIO Port C (AHB) base: 0x4005.A000
 GPIO Port D (APB) base: 0x4000.7000
 GPIO Port D (AHB) base: 0x4005.B000
 GPIO Port E (APB) base: 0x4002.4000
 GPIO Port E (AHB) base: 0x4005.C000
 GPIO Port F (APB) base: 0x4002.5000
 GPIO Port F (AHB) base: 0x4005.D000
 Offset 0x41C
 Type W1C, reset 0x0000.0000

| | | | | | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | reserved | | | | | | | | | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | reserved | | | | | | | | IC | | | | | | | |
| Type | RO | RO | RO | RO | RO | RO | RO | RO | W1C | W1C | W1C | W1C | W1C | W1C | W1C | W1C |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit/Field | Name | Type | Reset | Description |
|-----------|----------|------|-------|---|
| 31:8 | reserved | RO | 0 | Software should not rely on the value of a reserved bit. To provide compatibility with future products, the value of a reserved bit should be preserved across a read-modify-write operation. |
| 7:0 | IC | W1C | 0x00 | GPIO Interrupt Clear |
| | | | | Value Description |
| | | | | 0 The corresponding interrupt is unaffected. |
| | | | | 1 The corresponding interrupt is cleared. |