11 Interrupts and events

11.1 Nested vectored interrupt controller (NVIC)

11.1.1 NVIC main features

- 32 maskable interrupt channels (not including the sixteen Cortex®-M0 interrupt lines)
- 4 programmable priority levels (2 bits of interrupt priority are used)
- Low-latency exception and interrupt handling
- Power management control
- Implementation of System control registers

The NVIC and the processor core interface are closely coupled, which enables low latency interrupt processing and efficient processing of late arriving interrupts.

All interrupts including the core exceptions are managed by the NVIC. For more information on exceptions and NVIC programming, refer to the PM0215 programming manual.

For code example refer to the Appendix section *A.6.1: NVIC initialization example*.

11.1.2 SysTick calibration value register

The SysTick calibration value is set to 6000, which gives a reference time base of 1 ms with the SysTick clock set to 6 MHz (max f_{HCLK} / 8).

11.1.3 Interrupt and exception vectors

Table 36 is the vector table for STM32F0xx devices. Consider peripheral availability on your device.

Table 36. Vector table

Position	Priority	Type of priority	Acronym	Description	Address
-	-	-	-	Reserved	0x0000 0000
-	-3	Fixed	Reset	Reset	0x0000 0004
-	-2	Fixed	NMI	Non maskable interrupt. The RCC clock security system (CSS) and the RAM parity check are linked to the NMI vector.	0x0000 0008
-	-1	Fixed	HardFault	All classes of fault	0x0000 000C
-	3	Settable	SVCall	System service call via SWI instruction	0x0000 002C
-	5	Settable	PendSV	Pendable request for system service	0x0000 0038
-	6	Settable	SysTick	System tick timer	0x0000 003C
0	7	Settable	WWDG	Window watchdog interrupt	0x0000 0040
1	8	Settable	PVD_VDDIO2	PVD and V_{DDIO2} supply comparator interrupt (combined EXTI lines 16 and 31)	0x0000 0044
2	9	Settable	RTC	RTC interrupts (combined EXTI lines 17, 19 and 20)	0x0000 0048

Table 36. Vector table (continued)

Position	Priority	Type of priority	Acronym	Description	Address
3	10	Settable	FLASH	Flash global interrupt	0x0000 004C
4	11	Settable	RCC_CRS	RCC and CRS global interrupts	0x0000 0050
5	12	Settable	EXTI0_1	EXTI Line[1:0] interrupts	0x0000 0054
6	13	Settable	EXTI2_3	EXTI Line[3:2] interrupts	0x0000 0058
7	14	Settable	EXTI4_15	EXTI Line[15:4] interrupts	0x0000 005C
8	15	Settable	TSC	Touch sensing interrupt	0x0000 0060
9	16	Settable	DMA_CH1	DMA channel 1 interrupt	0x0000 0064
10	17	Settable	DMA_CH2_3 DMA2_CH1_2	DMA channel 2 and 3 interrupts DMA2 channel 1 and 2 interrupts	0x0000 0068
11	18	Settable	DMA_CH4_5_6_7 DMA2_CH3_4_5	DMA channel 4, 5, 6 and 7 interrupts DMA2 channel 3, 4 and 5 interrupts	0x0000 006C
12	19	Settable	ADC_COMP	ADC and COMP interrupts (ADC interrupt combined with EXTI lines 21 and 22)	0x0000 0070
13	20	Settable	TIM1_BRK_UP_ TRG_COM	TIM1 break, update, trigger and commutation interrupt	0x0000 0074
14	21	Settable	TIM1_CC	TIM1 capture compare interrupt	0x0000 0078
15	22	Settable	TIM2	TIM2 global interrupt	0x0000 007C
16	23	Settable	TIM3	TIM3 global interrupt	0x0000 0080
17	24	Settable	TIM6_DAC	TIM6 global interrupt and DAC underrun interrupt	0x0000 0084
18	25	Settable	TIM7	TIM7 global interrupt	0x0000 0088
19	26	Settable	TIM14	TIM14 global interrupt	0x0000 008C
20	27	Settable	TIM15	TIM15 global interrupt	0x0000 0090
21	28	Settable	TIM16	TIM16 global interrupt	0x0000 0094
22	29	Settable	TIM17	TIM17 global interrupt	0x0000 0098
23	30	Settable	I2C1	I^2C1 global interrupt (combined with EXTI line 23)	0x0000 009C
24	31	Settable	I2C2	I^2C2 global interrupt	0x0000 00A0
25	32	Settable	SPI1	SPI1 global interrupt	0x0000 00A4
26	33	Settable	SPI2	SPI2 global interrupt	0x0000 00A8
27	34	Settable	USART1	USART1 global interrupt (combined with EXTI line 25)	0x0000 00AC
28	35	Settable	USART2	USART2 global interrupt (combined with EXTI line 26)	0x0000 00B0
29	36	Settable	USART3_4_5_6_7_8	USART3, USART4, USART5, USART6, USART7, USART8 global interrupts (combined with EXTI line 28)	0x0000 00B4
30	37	Settable	CEC_CAN	CEC and CAN global interrupts (combined with EXTI line 27)	0x0000 00B8
31	38	Settable	USB	USB global interrupt (combined with EXTI line 18)	0x0000 00BC

11.2 Extended interrupts and events controller (EXTI)

The extended interrupts and events controller (EXTI) manages the external and internal asynchronous events/interrupts and generates the event request to the CPU/Interrupt controller and a wake-up request to the Power manager.

The EXTI allows the management of up to 32 external/internal event line (23 external event lines and 9 internal event lines).

The active edge of each external interrupt line can be chosen independently, whilst for internal interrupt the active edge is always the rising one. An interrupt could be left pending: in case of an external one, a status register is instantiated and indicates the source of the interrupt; an event is always a simple pulse and it's used for triggering the core Wake-up (e.g. Cortex-M0 RXEV pin). For internal interrupts, the pending status is assured by the generating IP, so no need for a specific flag. Each input line can be masked independently for interrupt or event generation, in addition the internal lines are sampled only in STOP mode. This controller allows also to emulate the (only) external events by software, multiplexed with the corresponding hardware event line, by writing to a dedicated register.

11.2.1 Main features

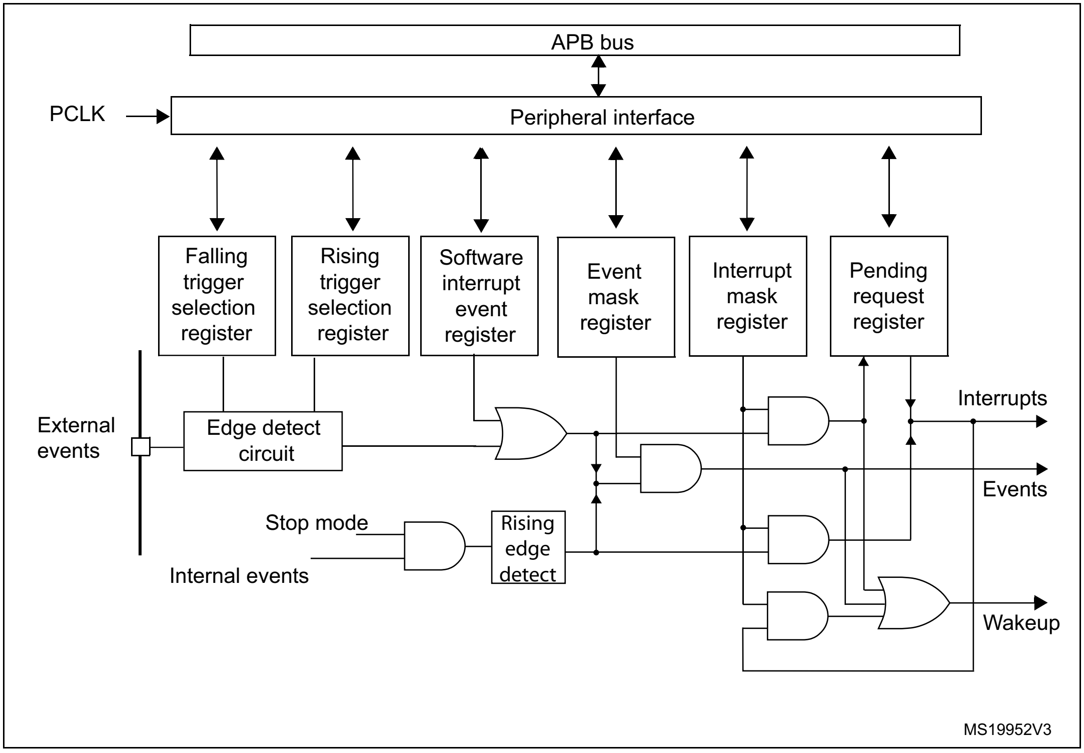
The EXTI main features are the following:

- Supports generation of up to 32 event/interrupt requests
- Independent mask on each event/interrupt line
- Automatic disable of internal lines when system is not in STOP mode
- Independent trigger for external event/interrupt line
- Dedicated status bit for external interrupt line
- Emulation for all the external event requests

11.2.2 Block diagram

The extended interrupt/event block diagram is shown in *Figure 23*.

Figure 23. Extended interrupts and events controller (EXTI) block diagram



11.2.3 Event management

The STM32F0xx is able to handle external or internal events in order to wake up the core (WFE). The wake-up event can be generated either by:

- enabling an interrupt in the peripheral control register but not in the NVIC, and enabling the SEVONPEND bit in the Cortex-M0 System control register. When the MCU resumes from WFE, the EXTI peripheral interrupt pending bit and the peripheral NVIC IRQ channel pending bit (in the NVIC interrupt clear pending register) have to be cleared.
- or by configuring an external or internal EXTI line in event mode. When the CPU resumes from WFE, it is not necessary to clear the peripheral interrupt pending bit or the NVIC IRQ channel pending bit as the pending bit corresponding to the event line is not set.

11.2.4 Functional description

For the external interrupt lines, to generate the interrupt, the interrupt line should be configured and enabled. This is done by programming the two trigger registers with the desired edge detection and by enabling the interrupt request by writing a ‘1’ to the corresponding bit in the interrupt mask register. When the selected edge occurs on the external interrupt line, an interrupt request is generated. The pending bit corresponding to the interrupt line is also set. This request is reset by writing a ‘1’ in the pending register.

For the internal interrupt lines, the active edge is always the rising edge, the interrupt is enabled by default in the interrupt mask register and there is no corresponding pending bit in the pending register.

To generate the event, the event line should be configured and enabled. This is done by programming the two trigger registers with the desired edge detection and by enabling the event request by writing a ‘1’ to the corresponding bit in the event mask register. When the selected edge occurs on the event line, an event pulse is generated. The pending bit corresponding to the event line is not set.

For the external lines, an interrupt/event request can also be generated by software by writing a ‘1’ in the software interrupt/event register.

Note: The interrupts or events associated to the internal lines can be triggered only when the system is in STOP mode. If the system is still running, no interrupt/event is generated.

For code example refer to the Appendix section *A.6.2: External interrupt selection code example*.

Hardware interrupt selection

To configure a line as interrupt source, use the following procedure:

- Configure the corresponding mask bit in the EXTI_IMR register.
- Configure the trigger selection bits of the interrupt line (EXTI_RTSR and EXTI_FTSR)
- Configure the enable and mask bits that control the NVIC IRQ channel mapped to the EXTI so that an interrupt coming from one of the EXTI line can be correctly acknowledged.

Hardware event selection

To configure a line as event source, use the following procedure:

- Configure the corresponding mask bit in the EXTI_EMR register.
- Configure the Trigger Selection bits of the Event line (EXTI_RTSR and EXTI_FTSR)

Software interrupt/event selection

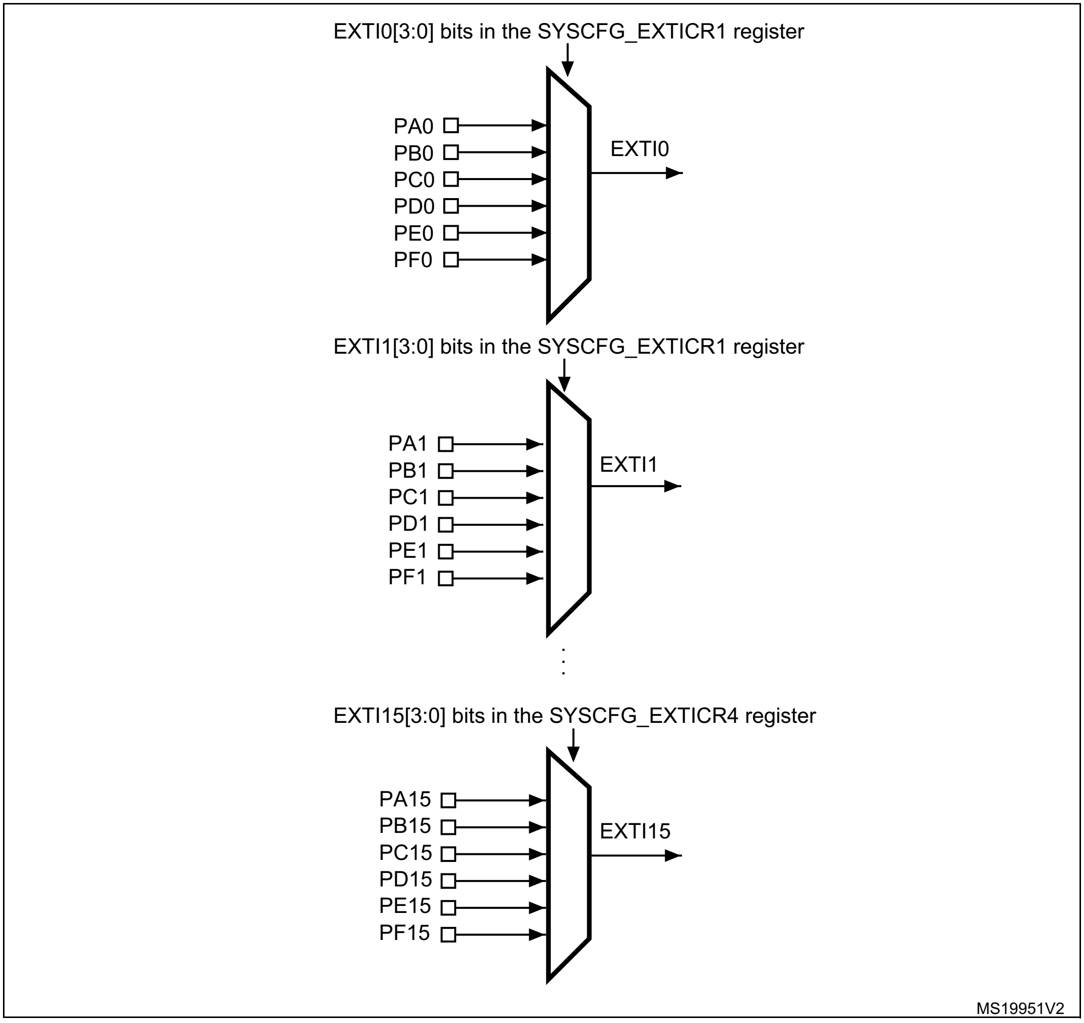
Any of the external lines can be configured as software interrupt/event lines. The following is the procedure to generate a software interrupt.

- Configure the corresponding mask bit (EXTI_IMR, EXTI_EMR)
- Set the required bit of the software interrupt register (EXTI_SWIER)

11.2.5 External and internal interrupt/event line mapping

The GPIOs are connected to the 16 external interrupt/event lines in the following manner:

Figure 24. External interrupt/event GPIO mapping


EXTI0[3:0] bits in the SYSCFG_EXTICR1 register
PA0
PB0
PC0
PD0
PE0
PF0
EXTI0
EXTI1[3:0] bits in the SYSCFG_EXTICR1 register
PA1
PB1
PC1
PD1
PE1
PF1
EXTI1
EXTI15[3:0] bits in the SYSCFG_EXTICR4 register
PA15
PB15
PC15
PD15
PE15
PF15
EXTI15
MS19951V2

The remaining lines are connected as follow:

- EXTI line 16 is connected to the PVD output
- EXTI line 17 is connected to the RTC Alarm event
- EXTI line 18 is connected to the internal USB wake-up event
- EXTI line 19 is connected to the RTC Tamper and TimeStamp events
- EXTI line 20 is connected to the RTC Wake-up event (available only on STM32F07x and STM32F09x devices)
- EXTI line 21 is connected to the Comparator 1 output
- EXTI line 22 is connected to the Comparator 2 output
- EXTI line 23 is connected to the internal I2C1 wake-up event
- EXTI line 24 is reserved (internally held low)
- EXTI line 25 is connected to the internal USART1 wake-up event
- EXTI line 26 is connected to the internal USART2 wake-up event (available only on STM32F07x and STM32F09x devices)
- EXTI line 27 is connected to the internal CEC wake-up event
- EXTI line 28 is connected to the internal USART3 wake-up event (available only on STM32F09x devices)
- EXTI line 29 is reserved (internally held low)
- EXTI line 30 is reserved (internally held low)
- EXTI line 31 is connected to the V_{DDIO2} supply comparator output (available only on STM32F04x, STM32F07x and STM32F09x devices

Note: EXTI lines which are reserved or not used on some devices are considered as internal.

11.3 EXTI registers

Refer to *Section 1.2 on page 42* for a list of abbreviations used in register descriptions.

The peripheral registers have to be accessed by words (32-bit).

11.3.1 Interrupt mask register (EXTI_IMR)

Address offset: 0x00
Reset value: 0x0FF4 0000 (STM32F03x devices)
0x7FF4 0000 (STM32F04x devices)
0x0F94 0000 (STM32F05x devices)
0x7F84 0000 (STM32F07x and STM32F09x devices)

Note: The reset value for the internal lines is set to ‘1’ in order to enable the interrupt by default.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM31	IM30	IM29	IM28	IM27	IM26	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 **IMx:** Interrupt Mask on line x (x = 31 to 0)
0: Interrupt request from Line x is masked
1: Interrupt request from Line x is not masked

11.3.2 Event mask register (EXTI_EMR)

Address offset: 0x04
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EM31	EM30	EM29	EM28	EM27	EM26	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 **EMx:** Event mask on line x (x = 31 to 0)
0: Event request from Line x is masked
1: Event request from Line x is not masked

11.3.3 Rising trigger selection register (EXTI_RTSR)

Address offset: 0x08
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RT31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT22	RT21	RT20	RT19	Res.	RT17	RT16
rw									rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **RT31:** Rising trigger event configuration bit of line 31
0: Rising trigger disabled (for Event and Interrupt) for input line
1: Rising trigger enabled (for Event and Interrupt) for input line.

Bits 30:23 Reserved, must be kept at reset value.

Bits 22:19 **RTx:** Rising trigger event configuration bit of line x (x = 22 to 19)
0: Rising trigger disabled (for Event and Interrupt) for input line
1: Rising trigger enabled (for Event and Interrupt) for input line.

Bit 18 Reserved, must be kept at reset value.

Bits 17:0 **RTx:** Rising trigger event configuration bit of line x (x = 17 to 0)
0: Rising trigger disabled (for Event and Interrupt) for input line
1: Rising trigger enabled (for Event and Interrupt) for input line.

Note: *The external wake-up lines are edge triggered. No glitches must be generated on these lines. If a rising edge on an external interrupt line occurs during a write operation to the EXTI_RTSR register, the pending bit is not set.*

Rising and falling edge triggers can be set for the same interrupt line. In this case, both generate a trigger condition.

11.3.4 Falling trigger selection register (EXTI_FTSR)

Address offset: 0x0C
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FT31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT22	FT21	FT20	FT19	Res.	FT17	FT16
rw									rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **FT31:** Falling trigger event configuration bit of line 31
0: Falling trigger disabled (for Event and Interrupt) for input line
1: Falling trigger enabled (for Event and Interrupt) for input line.

Bits 30:23 Reserved, must be kept at reset value.

Bits 22:19 **FTx:** Falling trigger event configuration bit of line x (x = 22 to 19)
0: Falling trigger disabled (for Event and Interrupt) for input line.
1: Falling trigger enabled (for Event and Interrupt) for input line.

Bit 18 Reserved, must be kept at reset value.

Bits 17:0 **FTx:** Falling trigger event configuration bit of line x (x = 17 to 0)
0: Falling trigger disabled (for Event and Interrupt) for input line.
1: Falling trigger enabled (for Event and Interrupt) for input line.

Note: *The external wake-up lines are edge triggered. No glitches must be generated on these lines. If a falling edge on an external interrupt line occurs during a write operation to the EXTI_FTSR register, the pending bit is not set.*

Rising and falling edge triggers can be set for the same interrupt line. In this case, both generate a trigger condition.

11.3.5 Software interrupt event register (EXTI_SWIER)

Address offset: 0x10
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWI31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI22	SWI21	SWI20	SWI19	Res.	SWI17	SWI16
rw									rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWI15	SWI14	SWI13	SWI12	SWI11	SWI10	SWI9	SWI8	SWI7	SWI6	SWI5	SWI4	SWI3	SWI2	SWI1	SWI0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **SWI31:** Software interrupt on line 31

If the interrupt is enabled on this line in the EXTI_IMR, writing a ‘1’ to this bit when it is at ‘0’ sets the corresponding pending bit in EXTI_PR resulting in an interrupt request generation.

This bit is cleared by clearing the corresponding bit of EXTI_PR (by writing a ‘1’ to the bit)

Bits 30:23 Reserved, must be kept at reset value.

Bits 22:19 **SWIx:** Software interrupt on line x (x = 22 to 19)

If the interrupt is enabled on this line in the EXTI_IMR, writing a ‘1’ to this bit when it is at ‘0’ sets the corresponding pending bit in EXTI_PR resulting in an interrupt request generation.

This bit is cleared by clearing the corresponding bit of EXTI_PR (by writing a ‘1’ to the bit)

Bit 18 Reserved, must be kept at reset value.

Bits 17:0 **SWIx:** Software interrupt on line x (x = 17 to 0)

If the interrupt is enabled on this line in the EXTI_IMR, writing a ‘1’ to this bit when it is at ‘0’ sets the corresponding pending bit in EXTI_PR resulting in an interrupt request generation.

This bit is cleared by clearing the corresponding bit of EXTI_PR (by writing a ‘1’ to the bit).

11.3.6 Pending register (EXTI_PR)

Address offset: 0x14

Reset value: undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PIF31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF22	PIF21	PIF20	PIF19	Res.	PIF17	PIF16
rc_w1									rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

Bit 31 **PIF31:** Pending bit on line 31

0: no trigger request occurred
1: selected trigger request occurred

This bit is set when the selected edge event arrives on the external interrupt line. This bit is cleared by writing a 1 to the bit.

Bits 30:23 Reserved, must be kept at reset value.

Bits 22:19 **PIFx:** Pending bit on line x (x = 22 to 19)

0: no trigger request occurred
1: selected trigger request occurred

This bit is set when the selected edge event arrives on the external interrupt line. This bit is cleared by writing a 1 to the bit.

Bit 18 Reserved, must be kept at reset value.

Bits 17:0 **PIFx:** Pending bit on line x (x = 17 to 0)

0: no trigger request occurred
1: selected trigger request occurred

This bit is set when the selected edge event arrives on the external interrupt line. This bit is cleared by writing a 1 to the bit.

11.3.7 EXTI register map

The following table gives the EXTI register map and the reset values.

Table 37. External interrupt/event controller register map and reset values

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**EXTI_IMR**	IM[31:0]																															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**EXTI_EMR**	EM[31:0]																															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**EXTI_RTSR**	RT31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT23	RT22	RT21	RT20	RT19	Res.	RT[17:0]																	
	Reset value	0								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**EXTI_FTSR**	FT31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT23	FT22	FT21	FT20	FT19	Res.	FT[17:0]																	
	Reset value	0								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**EXTI_SWIER**	SWI31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI23	SWI22	SWI21	SWI20	SWI19	Res.	SWI[17:0]																	
	Reset value	0								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**EXTI_PR**	PIF31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF23	PIF22	PIF21	PIF20	PIF19	Res.	PIF[17:0]																	
	Reset value	0								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Refer to *Section 2.2 on page 46* for the register boundary addresses.