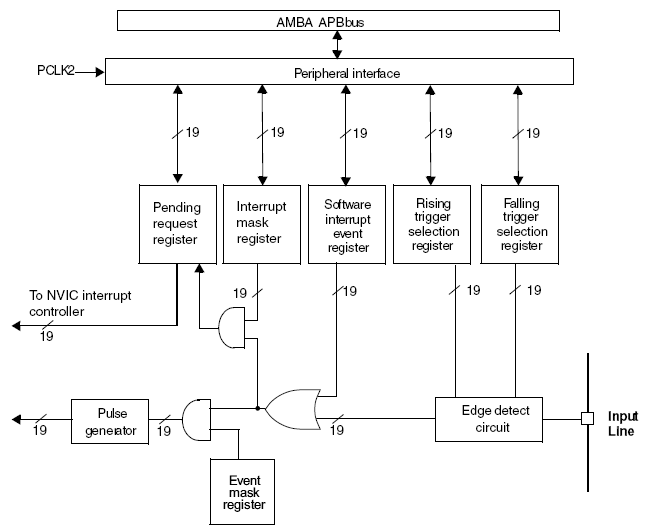
**Nested Vectored Interrupt Controller (NVIC)**

**Features**

* 68 + 16 Interrupt Lines
* 16 programmable priority levels
* Due to closely coupling between NVIC and core, low latency interrupt processing.
* All interrupts and core exceptions are managed by NVIC.

**Note:** SysTick clock should be set to 9 MHz so that SysTick calibration value should be 9000.

**External Interrupt/Event Controller (EXTI)**



**Features**

* Consist of 19-20 edge detectors for generating even/interrupt requests.
* Pending register maintains status line of the interrupt request.

**Function Description**

**Interrupt Generation**

* the interrupt line should be configured and enabled
  + by programming two trigger register with desired edge detection
  + by enabling the interrupt request by setting the interrupt mask register
* A pending request for the corresponding interrupt line is set when interrupt occurs and cleared by writing 1 to it.

**Event Generation**

* the event line should be configured and enabled
  + by programming two trigger register with desired edge detection
  + by enabling the event request by setting the event mask register
* A pending request for the corresponding interrupt line is not set.

**Hardware Interrupt Selection - configure the 20 lines as interrupt source**

* Configure mask bits of 20 interrupt line (***EXTI\_IMR***).
* Configure the trigger selection bit of the interrupt lines (***EXTI\_RTSR*** and ***EXTI\_FRST***).
* Configure the enable and mask bits that control the NVIC IRQ channel mapped to External Interrupt Controller (***EXTI***) so that output coming from one of the 20 line is correctly acknowledged.

**Hardware Event Selection - configure the 20 lines as even source**

* Configure mask bits of 20 even line (***EXTI\_EMR***).
* Configure the trigger selection bit of the Event lines (***EXTI\_RTSR*** and ***EXTI\_FRST***).

**Software Interrupt/Event Selection - configure the 20 lines as software event/interrupt source**

* Configure mask bits of 20 event/interrupt line (***EXTI\_EMR, EXTI\_EMR***).
* Set the required bit of software interrupt register(***EXIT\_SWIER***)

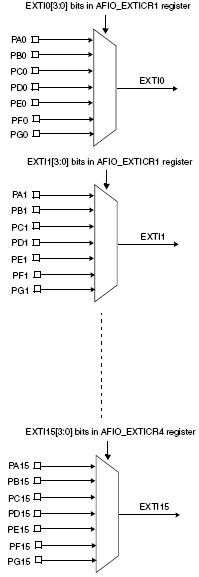
**Note:** PAx:PGx is connected to EXTIx and selected by ***AFIO\_EXTICR1.***

So, EXTI0:15 are used for Pins,

EXTI line 16 – PVD outpu

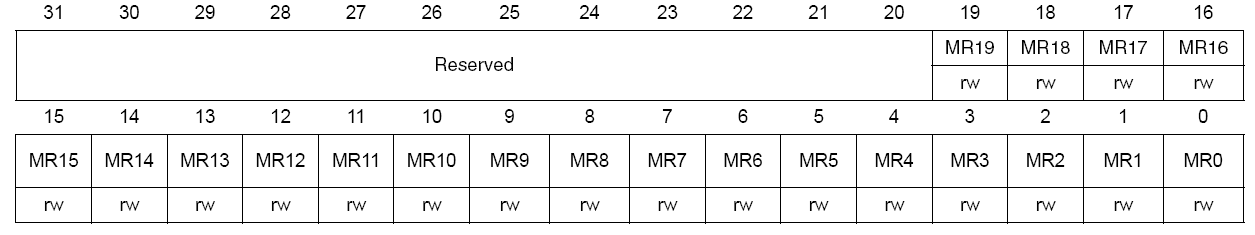
EXTI line 17 – RTC Alarm event

EXTI line 18 – USB Wakeup event

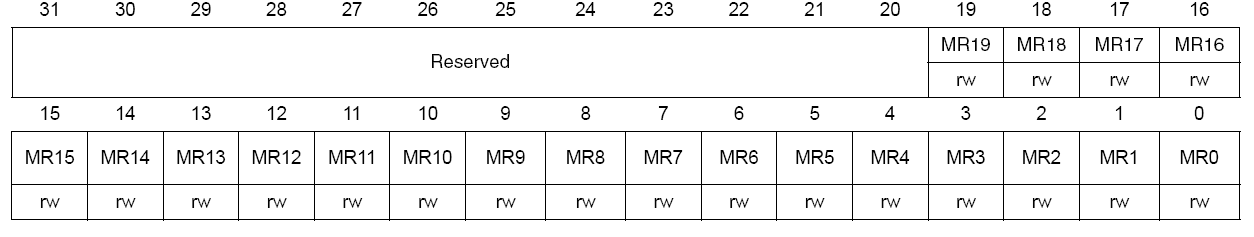


**EXTI Registers**

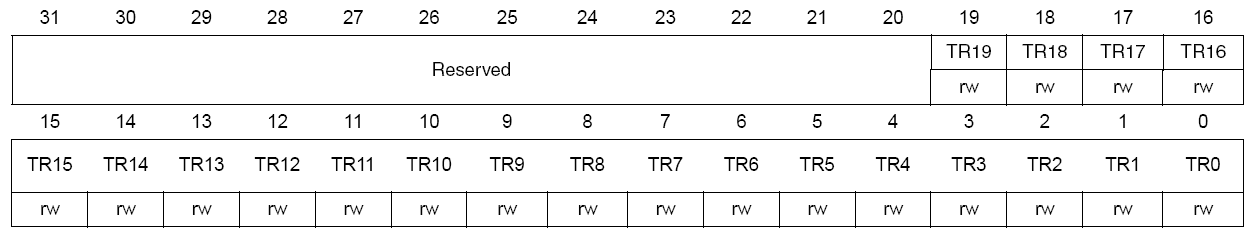
**Interrupt mask register (EXTI\_IMR)**



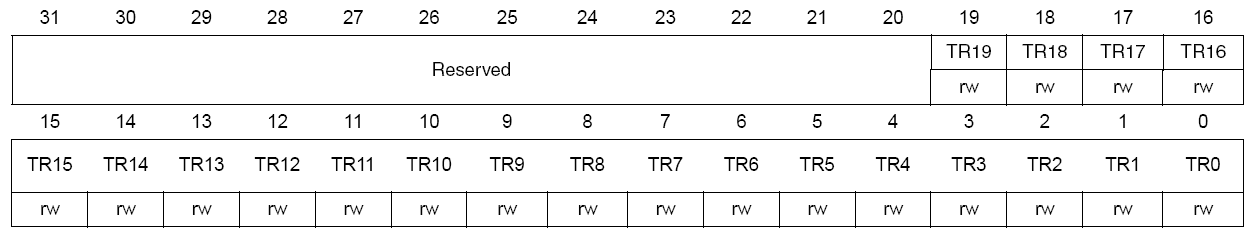
* **0 – Masked 1 – Unmasked**

**Event mask register (EXTI\_EMR)**

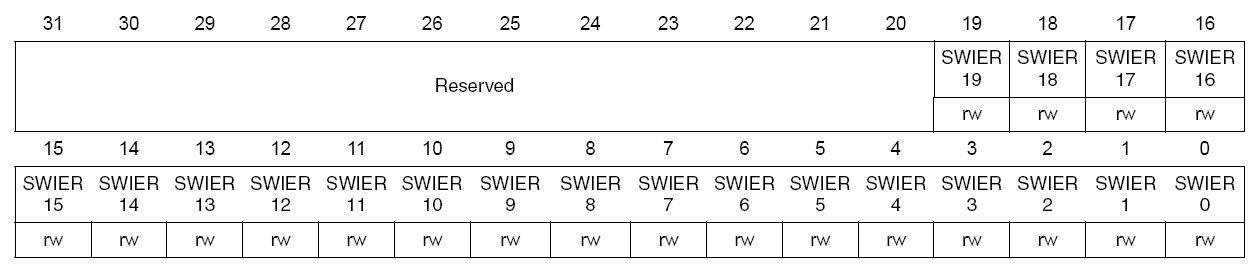
* **0 – Masked 1 – Unmasked**

**Rising trigger selection register (EXTI\_RTSR)**

* **1 –** Rising trigger enabled.

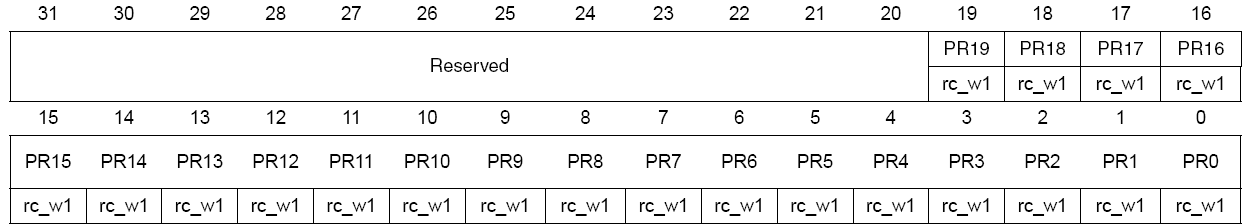
**Falling trigger selection register (EXTI\_FTSR)**

* **1 –** Falling trigger enabled

**Software interrupt event register (EXTI\_SWIER)**

* Generate software interrupt event by setting bits in this register
* cleared by writing corresponding bit in ***EXT\_PR***

**Pending register (EXTI\_PR)**



* **1 –** Trigger occurred.