**Interrupt Controller with APB Interface**

**Author :** Ciomag Andrei Daniel

**Company:** S.C. EasyIC Design S.R.L

**Specification**

Introduction

Interrupt Controller with APB Interface is a module that is used to combine several sources of interrupt into one single line, allowing priority levels to the interrupt input lines. The controller registers are configured via the AMBA 3 APB Protocol.

Features

* detects, stores and deals with interrupt pulses
* the module has four interrupt lines
* 4 bit Status Register for incoming interrupt status bits and pending interrupt status
* 4 bit configurable Mask Register for interrupt enables, this register specifies which interrupt are to be ignored and not acknowledged
* 4 bit configurable Clear Register, this register is configured to clear the Status Register and the Mask Register
* full AMBA 3 APB Interface
* asynchronous reset, active LOW
* enable signal for clock gating, to reduce dynamic power dissipation by removing the clock signal when the circuit is not in use

Interrupt Controller Interface

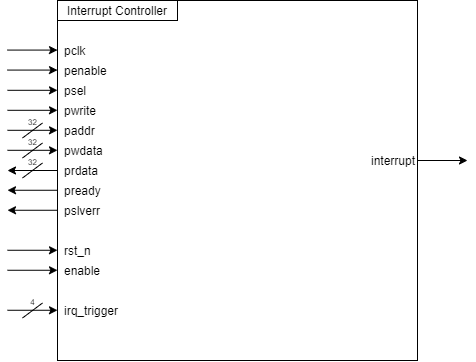


Figure Interrupt Controller Symbol

Signals Table

Table Signals Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Signal | Direction | Nr. of lines | Description | Active |
| pclk | input | 1 | clock | rising edge |
| penable | input | 1 | indicates the second cycle of an APB transfer | HIGH |
| psel | input | 1 | select to bus slave | HIGH |
| pwrite | input | 1 | write or read acces | HIGH w/LOW r |
| paddr | input | 32 | APB address bus | - |
| pwdata | input | 32 | APB write data bus | - |
| prdata | output | 32 | APB read data bus | - |
| pready | output | 1 | extend APB transfer | LOW |
| pslverr | output | 1 | indicates a transfer failure | HIGH |
| rst\_n | input | 1 | asynchronous reset | LOW |
| enable | input | 1 | clock gating enable | HIGH |
| irq\_trigger | input | 4 | interrupt requests | pulse HIGH |
| interrupt | output | 1 | interrupt | HIGH |

Implementation

Interrupt Controller has the following set of registers: Status Register, Mask Register, Clear Register. The Status Register specifies which interrupt request lines received trigger pulses, this happens through a block that detects the rising edges of the pulses. The Mask Register is a configurable register that specifies the priority of the interrupt request lines. By configuring the Clear Register, the clear function is performed through all the registers.

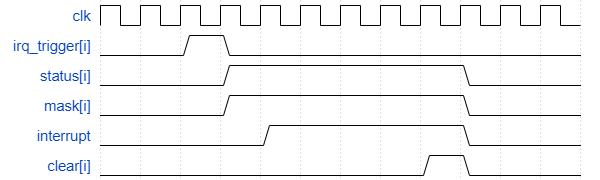


Figure Interrupt Controller registers behavior 1

Figure 2 shows that when a pulse appears on a input “i” of irq\_trigger, the flip-flop “i” of the Status Register is set to HIGH, and if the flip-flop “i” of the Mask Register is also HIGH, then the output interrupt is set HIGH.

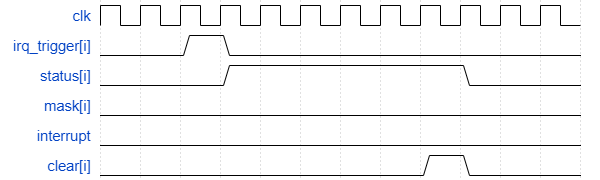
 Figure 3 shous that when the flip-flop “i” of the Mask Register is LOW, then the output interrupt is set LOW

Figure 3 Interrupt Controller registers behavior 2

The configuration of the registers takes place through the APB interface, which is used to set the priority of the interrupt request lines, and to clear the registers when the interrupt has been served. Figure 3 shows the block diagram of the Interrupt Controller.

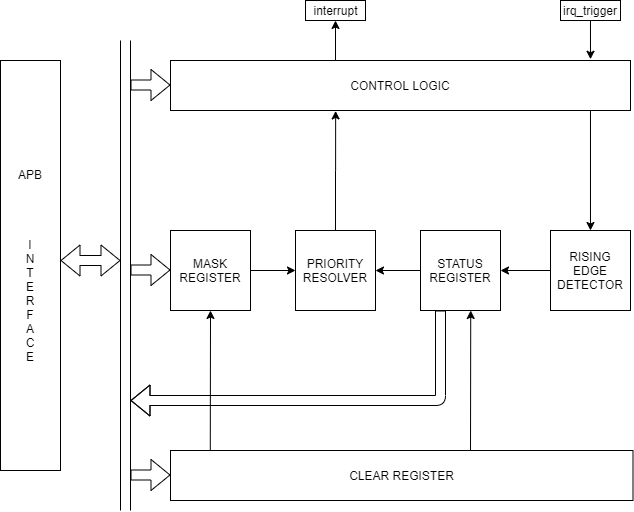


Figure Interrupt Controller block diagram

Reading and Writing of the registers

The addresses of the registers are the following:

* Status Register is 0x01
* Clear Register is 0x02
* Mask Register is 0x03

The Clear Register and the Mask Register are registers that can be written, and the Status Register can only be read.

When writing to the Mask Register only one bit can take the value 1, because there is only one interrupt output.

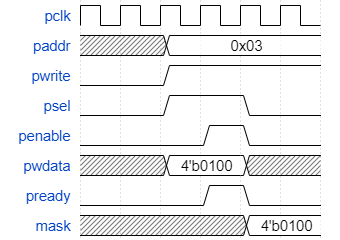


Figure 5 Write to Mask Register

In the case of the clear register, the value written in this register lasts only one clock cycle, after that, the register is set again to 0.

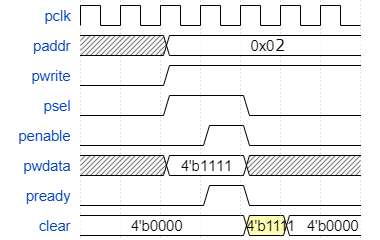


Figure 6 Write to Clear Register

When reading the Status Register it can be seen which of the irq\_trigger lines received interrupt pulses.

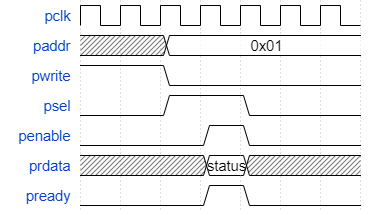


Figure 7 Read from Status Register

Priority Resolver

Interrupt requests are saved in the Status Register, but the output will take only the value of the bit that is “1” in the Mask Register and also in the Status Register. The logical operation between the two registers is AND. The output interrupt signal will take the value of the OR ON BITS operation of the result.

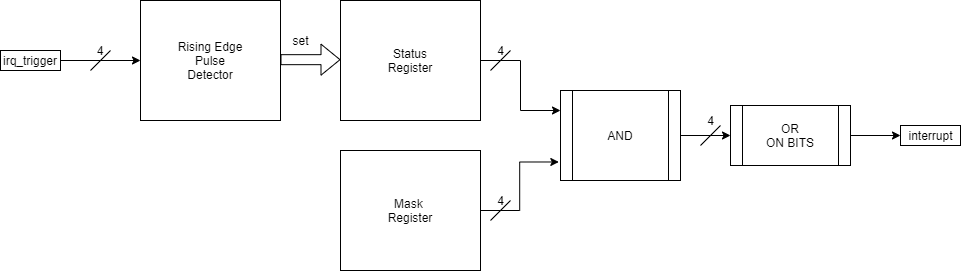


Figure 8 Output interrupt signal