

Vcilopic

1) Functional Description

This component is a multi-channels, programmable, Hardware Interrupt to Software Interrupt translator. It can be used to translate a variable number of hardware interrupt lines (HWI) to the same number of write triggered interrupt (SWI) that can be handled by a VciXicu component. It acts as a VCI initiator, to send a single flit VCI packet to the target interrupt controller, when a rising edge is detected on a given HWI input. It acts also as a memory mapped VCI target, as the addresses of the WTI associated to a given HWI must be configured by the software.

For each HWI channel, there is three 32 bits addressable registers:

IOPIC_ADDRESS This READ/WRITE register contains the 32 LSB bits of the physical WTI address associated to the HWI channel.

IOPIC_EXTEND This READ/WRITE register contains the 32 MSB bits of the physical WTI address associated to the HWI channel.

IOPIC_STATUS This READ-ONLY register register contains the HWI channel status. Only the two LSB bits are significant:

- Bit 0 : HWI line current value.
- Bit 1 : Error reported in a WTI transaction when tis bit is set.

Any read access to the IOPIC status register reset the ERROR bit.

2) Component definition & usage

source:trunk/soclib/module/infrastructure_component/interrupt_infrastructure/vci_iopic/caba/metadata/vci_iopic.sd

```
Uses( 'vci_iopic' )
```

3) CABA Implementation

CABA sources

- interface :
[source:trunk/soclib/soclib/module/infrastructure_component/interrupt_infrastructure/vci_iopic/caba/source/include/vci_iopic.h](#)
- implementation :
[source:trunk/soclib/soclib/module/infrastructure_component/interrupt_infrastructure/vci_iopic/caba/source/src/vci_iopic.c](#)

CABA Constructor parameters

```
VciIopic(  
    sc_module_name name, // Component Name  
    const soclib::common::MappingTable &mt, // Mapping Table  
    const soclib::common::IntTab &srcid, // Initiator index  
    const soclib::common::IntTab &tgtid, // Target index  
    const size_t channels ); // Number of HWI channels (inputs)
```

CABA Ports

- sc_in<bool> **p_clk** : Global system clock
- sc_in<bool> **p_resetn** : Global system reset
- soclib::caba::VciInitiator<vci_param> **p_vci_initiator** : VCI initiator port
- soclib::caba::VciTarget<vci_param> **p_vci_target** : VCI target port
- sc_in<bool> ***p_hwi** : Input interrupts ports array