

WbInterco

1) Functional Description

This hardware component is a generic Wishbone interconnect with a round robin arbiter.

It is a multiplexed bus with integrated address decoding.

2) CABA Implementation

CABA sources

- interface : [source/trunk/soclib/soclib/module/network_component/wb_interco/include/wb_interco.h?](#)
- implementation : [source/trunk/soclib/soclib/module/network_component/wb_interco/src/wb_interco.cpp?](#)
- Arbiter definition: [source/trunk/soclib/soclib/module/network_component/wb_interco/include/arbiter.h?](#)
- Arbiter implementation:
[source/trunk/soclib/soclib/module/network_component/wb_interco/src/arbiter.cpp?](#)

CABA Constructor parameters

```
WbInterco ( sc_module_name insname,
            const soclib::common::MappingTable &mtb,
            const size_t &nb_of_masters,
            const size_t &nb_of_slaves
        );
```

CABA Ports

- sc_in<bool> **p_resetn** : Global system reset
- sc_in<bool> **p_clk** : Global system clock
- WbMaster<wb_param> ***p_to_slave**: ports to wishbone slaves
- WbSlave <wb_param> ***p_from_master**: ports from wishbone masters

3) TLM-T implementation

There is no TLM-T implementation for this component.

4) Component definition & usage

The WbInterco uses the [mapping table](#) to identify the slaves and to decode addresses. The slaves must be connected to the **p_to_slave** ports at the position corresponding to their mapping table index.