

ISS Wrapper

1) functional description

This component is a generic ISS wrapper. It can be used to build CABA simulation models for 32 bits RISC processors. It can wrap any Instruction Set Simulator respecting the generic ISS API defined [here](#). The wrapper itself respect the cache interface defined by the [VciXcache](#) component.

The generic ISS API is formally defined in source:trunk/soclib/lib/include/iss/iss.h

2) CABA Implementation

CABA Sources

- interface :

source:trunk/soclib/soclib/module/processor_component/iss_wrapper/caba/source/include/iss_wrapper.h

- implementation :

source:trunk/soclib/soclib/module/processor_component/iss_wrapper/caba/source/src/iss_wrapper.cpp

CABA Template parameters

IssWrapper takes the wrapped ISS as template parameter.

```
soclib::caba::IssWrapper<typename iss_t>
```

For instance if wrapping a [Mips](#), little endian form:

```
soclib::caba::IssWrapper<soclib::common::MipsElIss>
```

CABA Constructor parameters

```
IssWrapper( sc_module_name name, // instance name  
            int ident ); // processor identifier
```

Again, with a mipsel, we get:

```
soclib::caba::IssWrapper<soclib::common::MipsElIss> mips0( "mips0", 0 );
```

CABA Ports

- sc_in<bool> **p_resetn** : Global system reset
- sc_in<bool> **p_clk** : Global system clock
- sc_in<bool> **p_irq!**[N] : The interrupts. N is defined by the wrapped ISS.
- soclib::caba::IcacheProcessorPort **p_icache** : Instruction cache interface to the [VciXcache](#)
- soclib::caba::DcacheProcessorPort **p_dcach** : Data cache interface to the [VciXcache](#)