

# Ppc Processor Functional Description

This hardware component is a PPC405 processor core.

This ISS uses the [ISS2 API](#) and should be wrapped with an [VciXcacheWrapper](#).

The simulation model is actually an instruction set simulator, organised as a two-stage pipeline:

- First stage: instruction fetch & execute with a possible access to the external data cache.
- Second stage: read memory access is written back to registers

The main functional specifications are the following:

- The floating point instructions are not supported
- There is no TLB, and no hardware support for virtual memory directly in the ISS. Nevertheless, MMU may be supported through the cache.

## Component definition

Available in source:`trunk/soclib/soclib/lib/metadata/ppc405.sd`

## Usage

Ppc405 has no parameters.

```
Uses( 'common:ppc405')
```

## Ppc405 Processor ISS Implementation

The implementation is in

- [source:trunk/soclib/soclib/iss/ppc405/include/iss/ppc405.h?](#)
- [source:trunk/soclib/soclib/iss/ppc405/src/iss/ppc405.cpp?](#)
- [source:trunk/soclib/soclib/iss/ppc405/src/iss/ppc405\\_instructions.cpp?](#)
- [source:trunk/soclib/soclib/iss/ppc405/src/iss/ppc405\\_jump\\_tables.cpp?](#)
- [source:trunk/soclib/soclib/iss/ppc405/src/iss/ppc405\\_instructions.cpp?](#)

## Template parameters

This component has no template parameters.

## Constructor parameters

```
Ppc405Iss(  
    const std::string &name, // Instance Name  
    int ident); // processor id
```

# **Interrupts**

Ppc405 defines two interrupt lines.

- 0: Critical interrupt
- 1: External interrupt

# **Ports**

None, it is to the wrapper to provide them.