Mips Processor Functional Description

This hardware component is a Mips R3000 processor core. This is only an ISS, which should be wrapped with an IssWrapper.

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

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

The main functional specifications are the following:

- The patented LWL/LWR instructions are not implemented
- The floating point instructions are not supported
- There is no TLB, and no hardware support for virtual memory
- All Mips R3000 exceptions are handled, including the memory addressing X_IBE and X_DBE, but the write errors are not precise, due to the posted write buffer in the cache controller.
- A data cache line invalidation mechanism is supported: when a *LW* instruction is executed with the GPR[0] destination register, a cache line invalidation request is sent to the data cache.

Component definition

Available in source:trunk/soclib/desc/soclib/mips.sd

Usage

Mips has no parameters.

```
Uses( 'mips')
```

Mips Processor CABA Implementation

The caba implementation is in

- source:trunk/soclib/systemc/include/common/iss/mips.h
- source:trunk/soclib/systemc/src/common/iss/mips.cc
- source:trunk/soclib/systemc/src/common/iss/mips_instructions.cc

Template parameters

This component has no template parameters.

Constructor parameters

```
Mips(
     sc_module_name name,   //  Instance Name
     int  ident);   // processor id
```

Visible registers

The following internal registers define the processor internal state, and can be inspected:

- PC : Program counter
- IR : Instruction register
- GPR[i] : General registers (0 < i < 32)
- HI & LO : Intermediate registers for multiply / divide instructions
- CP0_REG[i] : Coprocessor 0 registers (0<=i<32). Implemented values:
    - 8: BAR : Bad address register
    - 12: SR : Status register
    - 13: CR : Cause register
    - 14: EPC : Exception PC register
    - 15: INFOS : CPU identification number on bits [9:0]

Ports

None, it is to the wrapper to provide them.