

UNISIM

ppcemu Simulator Manual

Gilles Mouchard

1 Simulator technical reference (generated)

This documentation has been automatically generated from the simulator UNISIM ppcemu version 1.0beta2 on Oct 11 2011.

1.1 Introduction

UNISIM ppcemu, user level PowerPC simulator with support of ELF32 binaries and Linux system call translation.

Section 1.2 gives licensing informations about the simulator. Section 1.3 shows the set of modules and services that compose the simulator. Section 1.4 shows how to invoke the simulator at the command line prompt. Section 1.5 gives the simulator parameters. Section 1.6 gives the simulator statistic counters. Section 1.7 gives the simulator statistic formulas.

1.2 Licensing

UNISIM ppcemu 1.0beta2

Copyright (C) 2007-2010, Commissariat a l'Energie Atomique (CEA)

License: BSD (see file COPYING)

Authors: Gilles Mouchard <gilles.mouchard@cea.fr>, Daniel Gracia Pérez <daniel.gracia-perez@cea.fr>

1.3 Simulated configuration

The UNISIM ppcemu simulator is composed of the following modules and services:

- **cpu**
- **elf32-loader**: this service implements an ELF32 Loader
- **gdb-server**: this service implements the GDB server remote serial protocol over TCP/IP. Standards GDB clients (e.g. gdb, eclipse, ddd) can connect to the simulator to debug the target application that runs within the simulator.
- **host-time**: this service is an abstraction layer for the host machine time
- **inline-debugger**: this service implements a built-in debugger in the terminal console
- **linux-loader**
- **linux-os**
- **memory**: this module implements a memory
- **time**: this service is an abstraction layer for the SystemC kernel time

1.4 Using the UNISIM ppccemu simulator

The UNISIM ppccemu simulator has the following command line options:

Usage: unisim-ppccemu-1.0beta2 [<options>] [...]

Options:

- `--set <param=value>` or `-s <param=value>`: set value of parameter 'param' to 'value'
- `--config <XML file>` or `-c <XML file>`: configures the simulator with the given XML configuration file
- `--get-config <XML file>` or `-g <XML file>`: get the simulator configuration XML file (you can use it to create your own configuration. This option can be combined with `-c` to get a new configuration file with existing variables from another file)
- `--list` or `-l`: lists all available parameters, their type, and their current value
- `--warn` or `-w`: enable printing of kernel warnings
- `--doc <Latex file>` or `-d <Latex file>`: enable printing a latex documentation
- `--version` or `-v`: displays the program version information
- `--share-path <path>` or `-p <path>`: the path that should be used for the share directory (absolute path)
- `--help` or `-h`: displays this help

1.5 Configuration

Simulator configuration (see below) can be modified using command line Options `--set <param=value>` or `--config <config file>`.

| Global | |
|---|---|
| Name: enable-gdb-server Default: true Valid: true, false | Type: parameter Data type: boolean |
| Description: Enable/Disable GDB server instantiation. | |
| Name: enable-inline-debugger Default: true Valid: true, false | Type: parameter Data type: boolean |
| Description: Enable/Disable inline debugger instantiation. | |
| Name: enable-press-enter-at-exit Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Enable/Disable pressing key enter at exit. | |
| Name: estimate-power Default: false Valid: true, false | Type: parameter Data type: boolean |

| | |
|--|---|
| Description: Enable/Disable power estimators instantiation. | |
| Name: kernel_logger.file Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Keep logger output in a file. | |
| Name: kernel_logger.filename Default: logger_output.txt | Type: parameter Data type: string |
| Description: Filename to keep logger output (the option file must be activated). | |
| Name: kernel_logger.std_err Default: true Valid: true, false | Type: parameter Data type: boolean |
| Description: Show logger output through the standard error output. | |
| Name: kernel_logger.std_err_color Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Colorize logger output through the standard error output (only works if std_err is active). | |
| Name: kernel_logger.std_out Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Show logger output through the standard output. | |
| Name: kernel_logger.std_out_color Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Colorize logger output through the standard output (only works if std_out is active). | |
| Name: kernel_logger.xml_file Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Keep logger output in a file xml formatted. | |
| Name: kernel_logger.xml_file_gzipped Default: false | Type: parameter Data type: boolean |

| | |
|--|---|
| Valid: true, false | |
| Description: If the <code>xml_file</code> option is active, the output file will be compressed (a <code>.gz</code> extension will be automatically added to the <code>xml_filename</code> option). | |
| Name: <code>kernel_logger.xml_filename</code> | Type: parameter |
| Default: <code>logger_output.xml</code> | Data type: string |
| Description: Filename to keep logger xml output (the option <code>xml_file</code> must be activated). | |
| cpu | |
| Name: <code>cpu.cpu-cycle-time</code> | Type: parameter |
| Default: 3333 | Data type: unsigned 64-bit integer |
| Description: CPU cycle time in picoseconds. | |
| Name: <code>cpu.voltage</code> | Type: parameter |
| Default: 1300 | Data type: unsigned 64-bit integer |
| Description: CPU voltage in mV. | |
| Name: <code>cpu.max-inst</code> | Type: parameter |
| Default: 18446744073709551615 | Data type: unsigned 64-bit integer |
| Description: maximum number of instructions to simulate. | |
| Name: <code>cpu.verbose-all</code> | Type: parameter |
| Default: false | Data type: boolean |
| Valid: true, false | |
| Description: globally enable/disable verbosity. | |
| Name: <code>cpu.verbose-setup</code> | Type: parameter |
| Default: false | Data type: boolean |
| Valid: true, false | |
| Description: enable/disable verbosity while setup. | |
| Name: <code>cpu.verbose-step</code> | Type: parameter |
| Default: false | Data type: boolean |
| Valid: true, false | |
| Description: enable/disable verbosity when simulating an instruction. | |

| | |
|--|---|
| Name: cpu.verbose-dtlb Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when accessing data translation lookahead buffer. | |
| Name: cpu.verbose-itlb Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when accessing instruction translation lookahead buffer. | |
| Name: cpu.verbose-dl1 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when accessing L1 data cache. | |
| Name: cpu.verbose-il1 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when accessing L1 instruction cache. | |
| Name: cpu.verbose-l2 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when accessing L2 unified cache. | |
| Name: cpu.verbose-load Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when simulating a load. | |
| Name: cpu.verbose-store Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when simulating a store. | |
| Name: cpu.verbose-read-memory Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when reading memory for a debug purpose. | |

| | |
|---|---|
| Name: cpu.verbose-write-memory Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when writing memory for a debug purpose. | |
| Name: cpu.verbose-exception Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when handling exceptions. | |
| Name: cpu.verbose-set-msr Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when setting MSR. | |
| Name: cpu.verbose-set-hid0 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when setting HID0. | |
| Name: cpu.verbose-set-hid1 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when setting HID1. | |
| Name: cpu.verbose-set-hid2 Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when setting HID2. | |
| Name: cpu.verbose-set-l2cr Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity when setting L2CR. | |
| Name: cpu.trap-on-instruction-counter Default: 18446744073709551615 | Type: parameter Data type: unsigned 64-bit integer |
| Description: number of simulated instruction before trapping. | |

| | |
|--|---|
| Name: <code>cpu.bus-cycle-time</code> Default: 13333 ps | Type: parameter Data type: <code>sc_time</code> |
| Description: bus cycle time. | |
| Name: <code>cpu.nice-time</code> Default: 1 ms | Type: parameter Data type: <code>sc_time</code> |
| Description: maximum time between synchronizations. | |
| Name: <code>cpu.ipc</code> Default: 1 | Type: parameter Data type: double precision floating-point |
| Description: targeted average instructions per second. | |
| elf32-loader | |
| Name: <code>elf32-loader.filename</code> Default: | Type: parameter Data type: string |
| Description: the ELF filename to load into memory. | |
| Name: <code>elf32-loader.base-addr</code> Default: 0x00000000 | Type: parameter Data type: unsigned 32-bit integer |
| Description: if <code>force-base-addr</code> is true force base address for a unique program segment, otherwise ignored. | |
| Name: <code>elf32-loader.force-base-addr</code> Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: if true force base address for a unique program segment. | |
| Name: <code>elf32-loader.force-use-virtual- ↔address</code> Default: true Valid: true, false | Type: parameter Data type: boolean |
| Description: force use of virtual addresses instead of physical addresses. | |
| Name: <code>elf32-loader.initialize-extra- ↔segment-bytes</code> | Type: parameter |

| | |
|--|---|
| Default: true Valid: true, false | Data type: boolean |
| Description: whether to initialize extra bytes in segments (<code>p_filesz < p_memsz</code>) to zero (true for standard ELF files). | |
| Name: elf32-loader.dump-headers Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: dump headers while loading ELF file. | |
| Name: elf32-loader.verbose Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity. | |
| Name: elf32-loader.dwarf-to-html- ↔output-directory Default: | Type: parameter Data type: string |
| Description: DWARF v2/v3 to HTML output directory. | |
| Name: elf32-loader.parse-dwarf Default: true Valid: true, false | Type: parameter Data type: boolean |
| Description: Enable/Disable parsing of DWARF debugging informations. | |
| gdb-server | |
| Name: gdb-server.memory-atom-size Default: 0x00000001 | Type: parameter Data type: unsigned 32-bit integer |
| Description: size of the smallest addressable element in memory. | |
| Name: gdb-server.tcp-port Default: 0x00000000 | Type: parameter Data type: signed 32-bit integer |
| Description: TCP/IP port to listen waiting for a GDB client connection. | |
| Name: gdb-server.architecture-descriptor ↔filename Default: gdb_powerpc.xml | Type: parameter Data type: string |

Description:

filename of a XML description of the connected processor.

Name: gdb-server.verbose**Type:** parameter**Default:** false**Data type:** boolean**Valid:** true, false**Description:**

Enable/Disable verbosity.

inline-debugger**Name:** inline-debugger.memory-atom-
↔size**Type:** parameter**Default:** 0x00000001**Data type:** unsigned 32-bit integer**Description:**

size of the smallest addressable element in memory.

Name: inline-debugger.num-loaders**Type:** parameter**Default:** 1**Data type:** unsigned 32-bit integer**Description:**

number of loaders.

Name: inline-debugger.search-path**Type:** parameter**Default:****Data type:** string**Description:**

Search path for source (separated by ';').

Name: inline-debugger.init-macro**Type:** parameter**Default:****Data type:** string**Description:**

path to initial macro to run when debugger starts.

Name: inline-debugger.output**Type:** parameter**Default:****Data type:** string**Description:**

path to output file where to redirect the debugger outputs.

linux-loader**Name:** linux-loader.endianness**Type:** parameter**Default:** big-endian**Data type:** string

| | |
|--|---|
| Description: The endianness of the binary loaded. Available values are: little-endian and big-endian.. | |
| Name: linux-loader.stack-base Default: 0xc0000000 | Type: parameter Data type: unsigned 32-bit integer |
| Description: The stack base address used for the load and execution of the linux application. | |
| Name: linux-loader.max-environ Default: 16384 | Type: parameter Data type: unsigned 32-bit integer |
| Description: The maximum size of the program environment during its execution.. | |
| Name: linux-loader.argc Default: 1 | Type: parameter Data type: unsigned 32-bit integer |
| Description: Number of commands in the program execution line (usually at least one which is the name of the program executed). The different tokens can be set up with the parameters argv[<n>] where <n> can go up to argc - 1.. | |
| Name: linux-loader.envc Default: 0 | Type: parameter Data type: unsigned 32-bit integer |
| Description: Number of environment variables defined for the program execution. The different variables can be set up with the parameters envp[<n>] where <n> can go up to envc - 1.. | |
| Name: linux-loader.verbose Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: Display verbose information. | |
| Name: linux-loader.argv[0] Default: | Type: parameter Data type: string |
| Description: The '0' token in the command line.. | |
| linux-os | |
| Name: linux-os.system Default: powerpc | Type: parameter Data type: string |
| Description: Emulated system architecture available values are "arm", "arm-eabi" and "powerpc". | |

| | |
|--|---|
| Name: linux-os.endianness Default: big-endian | Type: parameter Data type: string |
| Description: The endianness of the binary loaded. Available values are: little-endian and big-endian.. | |
| Name: linux-os.memory-page-size Default: 0x00001000 | Type: parameter Data type: unsigned 32-bit integer |
| Name: linux-os.utsname-sysname Default: Linux | Type: parameter Data type: string |
| Description: The value that the uname system call should return. As this service is providing linux emulation support its value should be 'Linux', so you should not modify it.. | |
| Name: linux-os.utsname-nodename Default: localhost | Type: parameter Data type: string |
| Description: The network node hostname that the uname system call should return. Default value is localhost, but you could write whatever name you want.. | |
| Name: linux-os.utsname-release Default: 2.6.31.14 | Type: parameter Data type: string |
| Description: The kernel release information that the uname system call should return. This should usually match the linux-kernel parameter.. | |
| Name: linux-os.utsname-version Default: #UNISIM SMP Fri Mar 12 05:23:09 ↔UTC 2010 | Type: parameter Data type: string |
| Description: The kernel version information that the uname system call should return.. | |
| Name: linux-os.utsname-machine Default: powerpc | Type: parameter Data type: string |
| Description: The machine information that the uname system call should return. This should be one of the supported architectures (the system parameter, that is, arm or powerpc) or a specific model derived from it (i.e., arm926ejs).. | |

| | |
|---|---|
| Name: linux-os.utsname-domainname Default: localhost | Type: parameter Data type: string |
| Description: The domain name information that the uname system call should return.. | |
| Name: linux-os.verbose Default: false Valid: true, false | Type: parameter Data type: boolean |
| memory | |
| Name: memory.org Default: 0x00000000 | Type: parameter Data type: unsigned 32-bit integer |
| Description: memory origin/base address. | |
| Name: memory.bytesize Default: 4294967295 | Type: parameter Data type: unsigned 32-bit integer |
| Description: memory size in bytes. | |
| Name: memory.cycle-time Default: 13333 ps | Type: parameter Data type: sc_time |
| Description: memory cycle time. | |
| Name: memory.read-latency Default: 13333 ps | Type: parameter Data type: sc_time |
| Description: memory read latency. | |
| Name: memory.write-latency Default: 0 s | Type: parameter Data type: sc_time |
| Description: memory write latency. | |
| Name: memory.verbose Default: false Valid: true, false | Type: parameter Data type: boolean |
| Description: enable/disable verbosity. | |

1.6 Statistics

Simulation statistic counters are listed below:

| cpu | |
|--|---|
| Name: <code>cpu.instruction-counter</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of simulated instructions. | |
| Name: <code>cpu.bus-cycle</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of simulated bus cycles. | |
| Name: <code>cpu.num-il1-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to L1 instruction cache. | |
| Name: <code>cpu.num-il1-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to L1 instruction cache. | |
| Name: <code>cpu.num-dl1-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to L1 data cache. | |
| Name: <code>cpu.num-dl1-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to L1 data cache. | |
| Name: <code>cpu.num-l2-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to unified L2 cache. | |

| | |
|--|---|
| Name: <code>cpu.num-l2-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to unified L2 cache. | |
| Name: <code>cpu.num-ibat-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to IBATs. | |
| Name: <code>cpu.num-ibat-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to IBATs. | |
| Name: <code>cpu.num-dbat-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to DBATs. | |
| Name: <code>cpu.num-dbat-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to DBATs. | |
| Name: <code>cpu.num-itlb-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to ITLB. | |
| Name: <code>cpu.num-itlb-misses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to ITLB. | |
| Name: <code>cpu.num-dtlb-accesses</code> | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of accesses to DTLB. | |

| | |
|--|---|
| Name: cpu.num-dtlb-misses | Type: statistic Data type: unsigned 64-bit integer |
| Description: number of misses to DTLB. | |
| memory | |
| Name: memory.memory-usage | Type: statistic Data type: unsigned 32-bit integer |
| Description: host memory usage in bytes of simulated memory. | |

1.7 Formulas

Simulation statistic formulas are listed below:

| | | |
|---|--|------------------|
| cpu | | |
| Name: cpu.il1-miss-rate Formula: $\text{cpu.num-il1-misses} / \text{cpu.}\langle\rightarrow\text{num-il1-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.dl1-miss-rate Formula: $\text{cpu.num-dl1-misses} / \text{cpu.}\langle\rightarrow\text{num-dl1-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.l2-miss-rate Formula: $\text{cpu.num-l2-misses} / \text{cpu.}\langle\rightarrow\text{num-l2-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.ibat-miss-rate Formula: $\text{cpu.num-ibat-misses} / \text{cpu.}\langle\rightarrow\text{num-ibat-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.dbat-miss-rate Formula: $\text{cpu.num-dbat-misses} / \text{cpu.}\langle\rightarrow\text{num-dbat-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.itlb-miss-rate Formula: $\text{cpu.num-itlb-misses} / \text{cpu.}\langle\rightarrow\text{num-itlb-accesses}$ | Type: formula Data type: floating-point | double precision |
| Name: cpu.dtlb-miss-rate Formula: $\text{cpu.num-dtlb-misses} / \text{cpu.}\langle\rightarrow\text{num-dtlb-accesses}$ | Type: formula Data type: floating-point | double precision |