OVP Guide to Using Processor Models

Model Specific Information for variant riscv_RV32G

Imperas Software Limited
Imperas Buildings, North Weston
Thame, Oxfordshire, OX9 2HA, UK
docs@imperas.com

Author	Imperas Software Limited
Version	0.5
Filename	OVP_Model_Specific_Information_riscv_RV32G.pdf
Created	3 October 2017
Status	OVP Standard Release

Copyright Notice

Right to Copy Documentation

Destination Control Statement

Disclaimer

Model Release Status

This model is released as part of OVP releases and is included in OVPworld packages. Please visit OVPworld.org.

Table of Contents

1 Overview

This document provides the details of an OVP Fast Processor Model variant.

OVP Fast Processor Models are written in C and provide a C API for use in C based platforms. The models also provide a native interface for use in SystemC TLM2 platforms.

The models are written using the OVP VMI API that provides a Virtual Machine Interface that defines the behavior of the processor. The VMI API makes a clear line between model and simulator allowing very good optimization and world class high speed performance.

Most models are provided as a binary shared object and also as source. This allows the download and use of the model binary or the use of the source to explore and modify the model.

The models are run through an extensive QA and regression testing process and most model families are validated using technology provided by the processor IP owners.

There is a companion document (OVP Guide to Using Processor Models) which explains the general concepts of OVP Fast Processor Models and their use. It is downloadable from the OVPworld website documentation pages.

1.1 Description

RISC-V RV32G 32-Bit Family Processor Model.

The Following Instruction Sets are supported

RV32 I - Base Integer Instruction Set

RV32 M - Standard Extension for Multiplication

RV32 A - Standard Extension for Atomic Instructions

RV32 F - Standard Extension for Single-Precision Floating-Point

RV32 F - Standard Extension for Double-Precision Floating-Point

This model only provides an ISA reference implementation, no privilege modes exist for this reference

1.2 Licensing

This Model is released under the Open Source Apache 2.0

1.3 Features

This Model is currently work in progress and has many features scheduled, but not yet implemented

1.4 Limitations

Instruction pipelines are not modeled in any way. All instructions are assumed to complete immediately. This means that instruction barrier instructions (e.g. fence.i) are treated as NOPs, with the exception of any undefined instruction behavior, which is modeled.

Caches and write buffers are not modeled in any way. All loads, fetches and stores complete immediately and in order, and are fully synchronous. Data barrier instructions (e.g. fence) are treated as NOPs, with the exception of any undefined instruction behavior, which is modeled.

Real-world timing effects are not modeled: all instructions are assumed to complete in a single cycle.

1.5 Verification

Extensive testing of supported instructions
This includes tests generated specifically for this model by Imperas
In addition to https://github.com/riscv/riscv-tests

1.6 References

The Model details are based upon the following specifications
---- RISC-V Instruction Set Manual, Volume I: User-Level ISA, Document Version 2.2
---- RISC-V Instruction Set Manual, Volume II: Privileged Architecture, Privileged Architecture Version 1.10

2 Configuration

2.1 Location

The model source and object file is found in the VLNV tree at:
riscv.ovpworld.org/processor/riscv/1.0

2.2 GDB Path

The default GDB for this model is found at:
$IMPERAS_HOME/lib/$IMPERAS_ARCH/CrossCompiler/microsemi-riscv-unknown-elf-gcc/bin/riscv64-unknown-elf-gdb

2.3 Semi-Host Library

The default semi-host library file is found in the VLNV tree at :
riscv.ovpworld.org/semihosting/riscv32Newlib/1.0

2.4 Processor Endian-ness

This is a LITTLE endian model.

2.5 QuantumLeap Support

This processor is qualified to run in a QuantumLeap enabled simulator.

2.6 Processor ELF Code

The ELF code supported by this model is: 0xf3

3 Other Variants in this Model

Table 1.

Variant
RV32I
RV32IM
RV32IMC
RV32IMAC
RV32G

RV64I
RV64IM
RV64IMC
RV64IMAC
RV64G
RISCV_UISA

4 Bus Ports

Table 2.

Type	Name	min	max
master (initiator)	INSTRUCTION	32	32
master (initiator)	DATA	32	32

5 Net Ports

Table 3.

Name	Type	Description
Reset	input	Processor Reset Port
eip	input	External Interrupt Port
lip	input	Local Interrupt Port
dip	input	

6 FIFO Ports

No FIFO Ports in this model.

7 Parameters

Table 4.

Name	Type	Description
verbose	Boolean	Specify verbose output messages
UISA_ISA_RV32I	Boolean	UISA Feature UISA_ISA_RV32I
UISA_ISA_RV32M	Boolean	UISA Feature UISA_ISA_RV32M
UISA_ISA_RV32A	Boolean	UISA Feature UISA_ISA_RV32A
UISA_ISA_RV32F	Boolean	UISA Feature UISA_ISA_RV32F
UISA_ISA_RV32D	Boolean	UISA Feature UISA_ISA_RV32D
UISA_ISA_RV32C	Boolean	UISA Feature UISA_ISA_RV32C
UISA_ISA_RV64I	Boolean	UISA Feature UISA_ISA_RV64I
UISA_ISA_RV64M	Boolean	UISA Feature UISA_ISA_RV64M
UISA_ISA_RV64A	Boolean	UISA Feature UISA_ISA_RV64A

UISA_ISA_RV64F	Boolean	UISA Feature UISA_ISA_RV64F
UISA_ISA_RV64D	Boolean	UISA Feature UISA_ISA_RV64D
UISA_ISA_RV64Q	Boolean	UISA Feature UISA_ISA_RV64Q
UISA_ISA_RV64L	Boolean	UISA Feature UISA_ISA_RV64L
UISA_ISA_RV64C	Boolean	UISA Feature UISA_ISA_RV64C
UISA_ISA_RVC	Boolean	UISA Feature UISA_ISA_RVC

8 Execution Modes

No execution modes.

9 Exceptions

Table 5.

Name	Code
Reset	0
Undefined	1
Arith	2
FPArith	3
RdAlign	4
WrAlign	5
RdAbort	6
WrAbort	7
RdPriv	8
WrPriv	9
Fetch	10

10 Hierarchy of the model

A CPU core may allow the user to configure it to instance many processors of a Symmetrical Multi Processor (SMP). A CPU core may also have sub elements within a processor, for example hardware threading blocks.
OVP processor models can be written to include SMP blocks and to have many levels of hierarchy.

Some OVP CPU models may have a fixed hierarchy, and some may be configured by settings in a configuration register. Please see the register definitions of this model.

This model documentation shows the settings and hierarchy of the default settings for this model variant.

10.1 Level 1:

This level in the model hierarchy has 2 commands.

This level in the model hierarchy has no register groups.

This level in the model hierarchy has no children.

11 Model Commands

11.1 Level 1:

Table 6.

Name	Arguments
isync	specify instruction address range for synchronous execution
itrace	enable or disable instruction tracing

12 Registers

12.1 Level 1:

Table 7.

Name	Bits	Initial value (Hex)		Description
zero	32	0	r-	
ra	32	0	rw	
sp	32	0	rw	
gp	32	0	rw	
tp	32	0	rw	
t0	32	0	rw	
t1	32	0	rw	
t2	32	0	rw	
s0	32	0	rw	
s1	32	0	rw	
a0	32	0	rw	
a1	32	0	rw	
a2	32	0	rw	
a3	32	0	rw	
a4	32	0	rw	
a5	32	0	rw	
a6	32	0	rw	
a7	32	0	rw	
s2	32	0	rw	
s3	32	0	rw	
s4	32	0	rw	
s5	32	0	rw	
s6	32	0	rw	
s7	32	0	rw	
s8	32	0	rw	
s9	32	0	rw	
s10	32	0	rw	
s11	32	0	rw	
t3	32	0	rw	

t4	32	0	rw	
t5	32	0	rw	
t6	32	0	rw	
pc	32	0	rw	program counter
ft0	32	-	--	
ft1	32	-	--	
ft2	32	-	--	
ft3	32	-	--	
ft4	32	-	--	
ft5	32	-	--	
ft6	32	-	--	
ft7	32	-	--	
ft8	32	-	--	
ft9	32	-	--	
ft10	32	-	--	
ft11	32	-	--	
ft12	32	-	--	
ft13	32	-	--	
ft14	32	-	--	
ft15	32	-	--	
ft16	32	-	--	
ft17	32	-	--	
ft18	32	-	--	
ft19	32	-	--	
ft20	32	-	--	
ft21	32	-	--	
ft22	32	-	--	
ft23	32	-	--	
ft24	32	-	--	
ft25	32	-	--	
ft26	32	-	--	
ft27	32	-	--	
ft28	32	-	--	
ft29	32	-	--	
ft30	32	-	--	
ft31	32	-	--	
ustatus	32	0	rw	
uie	32	0	rw	
utvec	32	0	rw	
uscratch	32	0	rw	
uepc	32	0	rw	
ucause	32	0	rw	

utval	32	0	rw	
uip	32	0	rw	
fflags	32	0	rw	
frm	32	0	rw	
fcsr	32	0	rw	
cycle	32	0	rw	
time	32	0	rw	
instret	32	0	rw	
hpmcounter3	32	0	rw	
hpmcounter4	32	0	rw	
hpmcounter5	32	0	rw	
hpmcounter6	32	0	rw	
hpmcounter7	32	0	rw	
hpmcounter8	32	0	rw	
hpmcounter9	32	0	rw	
hpmcounter10	32	0	rw	
hpmcounter11	32	0	rw	
hpmcounter12	32	0	rw	
hpmcounter13	32	0	rw	
hpmcounter14	32	0	rw	
hpmcounter15	32	0	rw	
hpmcounter16	32	0	rw	
hpmcounter17	32	0	rw	
hpmcounter18	32	0	rw	
hpmcounter19	32	0	rw	
hpmcounter20	32	0	rw	
hpmcounter21	32	0	rw	
hpmcounter22	32	0	rw	
hpmcounter23	32	0	rw	
hpmcounter24	32	0	rw	
hpmcounter25	32	0	rw	
hpmcounter26	32	0	rw	
hpmcounter27	32	0	rw	
hpmcounter28	32	0	rw	
hpmcounter29	32	0	rw	
hpmcounter30	32	0	rw	
hpmcounter31	32	0	rw	
cycleh	32	0	rw	
timeh	32	0	rw	
instreth	32	0	rw	
hpmcounter3h	32	0	rw	
hpmcounter4h	32	0	rw	

hpmcounter5h	32	0	rw	
hpmcounter6h	32	0	rw	
hpmcounter7h	32	0	rw	
hpmcounter8h	32	0	rw	
hpmcounter9h	32	0	rw	
hpmcounter10h	32	0	rw	
hpmcounter11h	32	0	rw	
hpmcounter12h	32	0	rw	
hpmcounter13h	32	0	rw	
hpmcounter14h	32	0	rw	
hpmcounter15h	32	0	rw	
hpmcounter16h	32	0	rw	
hpmcounter17h	32	0	rw	
hpmcounter18h	32	0	rw	
hpmcounter19h	32	0	rw	
hpmcounter20h	32	0	rw	
hpmcounter21h	32	0	rw	
hpmcounter22h	32	0	rw	
hpmcounter23h	32	0	rw	
hpmcounter24h	32	0	rw	
hpmcounter25h	32	0	rw	
hpmcounter26h	32	0	rw	
hpmcounter27h	32	0	rw	
hpmcounter28h	32	0	rw	
hpmcounter29h	32	0	rw	
hpmcounter30h	32	0	rw	
hpmcounter31h	32	0	rw	
sstatus	32	0	rw	
sedeleg	32	0	rw	
sideleg	32	0	rw	
sie	32	0	rw	
stvec	32	0	rw	
scounteren	32	0	rw	
sscratch	32	0	rw	
sepc	32	0	rw	
scause	32	0	rw	
stval	32	0	rw	
sip	32	0	rw	
satp	32	0	rw	
mvendorid	32	0	rw	
marchid	32	0	rw	
mimpid	32	0	rw	

mhartid	32	0	rw	
mstatus	32	0	rw	
misa	32	40000000	rw	
medeleg	32	0	rw	
mideleg	32	0	rw	
mie	32	0	rw	
mtvec	32	0	rw	
mcounteren	32	0	rw	
mscratch	32	0	rw	
mepc	32	0	rw	
mcause	32	0	rw	
mtval	32	0	rw	
mip	32	0	rw	
pmpcfg0	32	0	rw	
pmpcfg1	32	0	rw	
pmpcfg2	32	0	rw	
pmpcfg3	32	0	rw	
pmpaddr0	32	0	rw	
pmpaddr1	32	0	rw	
pmpaddr2	32	0	rw	
pmpaddr3	32	0	rw	
pmpaddr4	32	0	rw	
pmpaddr5	32	0	rw	
pmpaddr6	32	0	rw	
pmpaddr7	32	0	rw	
pmpaddr8	32	0	rw	
pmpaddr9	32	0	rw	
pmpaddr10	32	0	rw	
pmpaddr11	32	0	rw	
pmpaddr12	32	0	rw	
pmpaddr13	32	0	rw	
pmpaddr14	32	0	rw	
pmpaddr15	32	0	rw	
mcycle	32	0	rw	
minstret	32	0	rw	
mhpmcounter3	32	0	rw	
mhpmcounter4	32	0	rw	
mhpmcounter5	32	0	rw	
mhpmcounter6	32	0	rw	
mhpmcounter7	32	0	rw	
mhpmcounter8	32	0	rw	
mhpmcounter9	32	0	rw	

mhpmcounter10	32	0	rw	
mhpmcounter11	32	0	rw	
mhpmcounter12	32	0	rw	
mhpmcounter13	32	0	rw	
mhpmcounter14	32	0	rw	
mhpmcounter15	32	0	rw	
mhpmcounter16	32	0	rw	
mhpmcounter17	32	0	rw	
mhpmcounter18	32	0	rw	
mhpmcounter19	32	0	rw	
mhpmcounter20	32	0	rw	
mhpmcounter21	32	0	rw	
mhpmcounter22	32	0	rw	
mhpmcounter23	32	0	rw	
mhpmcounter24	32	0	rw	
mhpmcounter25	32	0	rw	
mhpmcounter26	32	0	rw	
mhpmcounter27	32	0	rw	
mhpmcounter28	32	0	rw	
mhpmcounter29	32	0	rw	
mhpmcounter30	32	0	rw	
mhpmcounter31	32	0	rw	
mcycleh	32	0	rw	
minstreth	32	0	rw	
mhpmcounter3h	32	0	rw	
mhpmcounter4h	32	0	rw	
mhpmcounter5h	32	0	rw	
mhpmcounter6h	32	0	rw	
mhpmcounter7h	32	0	rw	
mhpmcounter8h	32	0	rw	
mhpmcounter9h	32	0	rw	
mhpmcounter10h	32	0	rw	
mhpmcounter11h	32	0	rw	
mhpmcounter12h	32	0	rw	
mhpmcounter13h	32	0	rw	
mhpmcounter14h	32	0	rw	
mhpmcounter15h	32	0	rw	
mhpmcounter16h	32	0	rw	
mhpmcounter17h	32	0	rw	
mhpmcounter18h	32	0	rw	
mhpmcounter19h	32	0	rw	
mhpmcounter20h	32	0	rw	

mhpmcounter21h	32	0	rw	
mhpmcounter22h	32	0	rw	
mhpmcounter23h	32	0	rw	
mhpmcounter24h	32	0	rw	
mhpmcounter25h	32	0	rw	
mhpmcounter26h	32	0	rw	
mhpmcounter27h	32	0	rw	
mhpmcounter28h	32	0	rw	
mhpmcounter29h	32	0	rw	
mhpmcounter30h	32	0	rw	
mhpmcounter31h	32	0	rw	
mhpmevent3	32	0	rw	
mhpmevent4	32	0	rw	
mhpmevent5	32	0	rw	
mhpmevent6	32	0	rw	
mhpmevent7	32	0	rw	
mhpmevent8	32	0	rw	
mhpmevent9	32	0	rw	
mhpmevent10	32	0	rw	
mhpmevent11	32	0	rw	
mhpmevent12	32	0	rw	
mhpmevent13	32	0	rw	
mhpmevent14	32	0	rw	
mhpmevent15	32	0	rw	
mhpmevent16	32	0	rw	
mhpmevent17	32	0	rw	
mhpmevent18	32	0	rw	
mhpmevent19	32	0	rw	
mhpmevent20	32	0	rw	
mhpmevent21	32	0	rw	
mhpmevent22	32	0	rw	
mhpmevent23	32	0	rw	
mhpmevent24	32	0	rw	
mhpmevent25	32	0	rw	
mhpmevent26	32	0	rw	
mhpmevent27	32	0	rw	
mhpmevent28	32	0	rw	
mhpmevent29	32	0	rw	
mhpmevent30	32	0	rw	
mhpmevent31	32	0	rw	
tselect	32	0	rw	
tdata1	32	0	rw	

tdata2	32	0	rw	
tdata3	32	0	rw	
dcsr	32	0	rw	
dpc	32	0	rw	
dscratch	32	0	rw	

#

