



## OVP Guide to Using Processor Models

### Model specific information for Renesas\_r8c

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



Author	Imperas Software Limited
Version	20210408.0
Filename	OVP_Model_Specific_Information_m16c_r8c.pdf
Created	9 April 2021
Status	OVP Standard Release

## Copyright Notice

Copyright (c) 2021 Imperas Software Limited. All rights reserved. This software and documentation contain information that is the property of Imperas Software Limited. The software and documentation are furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. No part of the software and documentation may be reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Imperas Software Limited, or as expressly provided by the license agreement.

## Right to Copy Documentation

The license agreement with Imperas permits licensee to make copies of the documentation for its internal use only. Each copy shall include all copyrights, trademarks, service marks, and proprietary rights notices, if any.

## Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the readers responsibility to determine the applicable regulations and to comply with them.

## Disclaimer

IMPERAS SOFTWARE LIMITED, AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

## Model Release Status

This model is released as part of OVP releases and is included in OVPworld packages. Please visit [OVPworld.org](http://OVPworld.org).

# Contents

<b>1</b>	<b>Overview</b>	<b>1</b>
1.1	Description . . . . .	1
1.2	Licensing . . . . .	1
1.3	Limitations . . . . .	1
1.4	Verification . . . . .	2
1.5	Reference . . . . .	2
<b>2</b>	<b>Configuration</b>	<b>3</b>
2.1	Location . . . . .	3
2.2	GDB Path . . . . .	3
2.3	Semi-Host Library . . . . .	3
2.4	Processor Endian-ness . . . . .	3
2.5	QuantumLeap Support . . . . .	3
2.6	Processor ELF code . . . . .	3
<b>3</b>	<b>All Variants in this model</b>	<b>4</b>
<b>4</b>	<b>Bus Master Ports</b>	<b>5</b>
<b>5</b>	<b>Bus Slave Ports</b>	<b>6</b>
<b>6</b>	<b>Net Ports</b>	<b>7</b>
<b>7</b>	<b>FIFO Ports</b>	<b>8</b>
<b>8</b>	<b>Formal Parameters</b>	<b>9</b>
<b>9</b>	<b>Execution Modes</b>	<b>10</b>
<b>10</b>	<b>Exceptions</b>	<b>11</b>
<b>11</b>	<b>Hierarchy of the model</b>	<b>12</b>
11.1	Level 1 . . . . .	12
<b>12</b>	<b>Model Commands</b>	<b>13</b>
12.1	Level 1 . . . . .	13
12.1.1	isync . . . . .	13
12.1.2	itrace . . . . .	13

<b>13 Registers</b>	<b>14</b>
13.1 Level 1 . . . . .	14

# Chapter 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

M16c Family 16Bit CISC processor model.

### 1.2 Licensing

Copyright Posedge Software, Licensed as Open Source Apache 2.0

### 1.3 Limitations

Core Instruction Set Architecture only.

Interrupt and Reset Signals are TBD.

## 1.4 Verification

Model has been validated correct by running through extensive instruction level tests

## 1.5 Reference

Renesas 16-bit Single-Chip MicroComputerM16C/60, M16C/20, M16C/Tiny Series Software Manual Rev 4.00 2004.01(rej09b0137\_m16csm.pdf)

# Chapter 2

## Configuration

### 2.1 Location

This model's VLN is `posedgesoft.ovpworld.org/processor/m16c/1.0`.

The model source is usually at:

`$IMPERAS_HOME/ImperasLib/source/posedgesoft.ovpworld.org/processor/m16c/1.0`

The model binary is usually at:

`$IMPERAS_HOME/lib/$IMPERAS_ARCH/ImperasLib/posedgesoft.ovpworld.org/processor/m16c/1.0`

### 2.2 GDB Path

The default GDB for this model is: `$IMPERAS_HOME/lib/$IMPERAS_ARCH/gdb/m32c-elf-gdb`.

### 2.3 Semi-Host Library

The default semi-host library file is `posedgesoft.ovpworld.org/semihosting/m16cNewlib/1.0`

### 2.4 Processor Endian-ness

This is a LITTLE endian model.

### 2.5 QuantumLeap Support

A simulator using this processor will not be able to use QuantumLeap.

### 2.6 Processor ELF code

ELF codes supported by this model are: `0x75` and `0x78`.

## Chapter 3

# All Variants in this model

This model has these variants

<b>Variant</b>	Description
m16c	
r8c	(described in this document)

Table 3.1: All Variants in this model



## Chapter 4

# Bus Master Ports

This model has these bus master ports.

<b>Name</b>	min	max	Connect?	Description
INSTRUCTION	32	32	mandatory	
DATA	32	32	optional	

Table 4.1: Bus Master Ports

## Chapter 5

# Bus Slave Ports

This model has no bus slave ports.

## Chapter 6

# Net Ports

This model has these net ports.

<b>Name</b>	Type	Connect?	Description
reset	input	optional	
nmi	input	optional	
int_per	input	optional	
int_ack	output	optional	

Table 6.1: Net Ports

## Chapter 7

# FIFO Ports

This model has no FIFO ports.

## Chapter 8

# Formal Parameters

Name	Type	Description
variant	Enumeration	Selects variant (either a generic ISA or a specific model)
compatibility	Enumeration	Specify compatibility mode (isa, gdb or nopbrk)
verbose	Boolean	Specify verbose output messages

Table 8.1: Parameters

## Chapter 9

# Execution Modes

This model does not have different execution modes.

## Chapter 10

# Exceptions

<b>Exception</b>	<b>Code</b>
Undefined	0
Overflow	1
BRK	2
AddressMatch	3
SingleStep	4
Watchdog	5
DBC	6
NMI	7
Reset	8
Fetch	9

Table 10.1: Exceptions implemented by this processor

## Chapter 11

# Hierarchy of the model

A CPU core may be configured 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.

### 11.1 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.



# Chapter 12

## Model Commands

A Processor model can implement one or more **Model Commands** available to be invoked from the simulator command line, from the OP API or from the Imperas Multiprocessor Debugger.

### 12.1 Level 1

#### 12.1.1 isync

specify instruction address range for synchronous execution

Argument	Type	Description
-addresshi	Uns64	end address of synchronous execution range
-addresslo	Uns64	start address of synchronous execution range

Table 12.1: isync command arguments

#### 12.1.2 itrace

enable or disable instruction tracing

Argument	Type	Description
-after	Uns64	apply after this many instructions
-enable	Boolean	enable instruction tracing
-instructioncount	Boolean	include the instruction number in each trace
-off	Boolean	disable instruction tracing
-on	Boolean	enable instruction tracing
-registerchange	Boolean	show registers changed by this instruction
-registers	Boolean	show registers after each trace

Table 12.2: itrace command arguments

# Chapter 13

## Registers

### 13.1 Level 1

Name	Bits	Initial-Hex	RW	Description
R0	16	0	rw	
R1	16	0	rw	
R2	16	0	rw	
R3	16	0	rw	
A0	16	0	rw	
A1	16	0	rw	
FB	16	0	rw	
R0B	16	0	rw	
R1B	16	0	rw	
R2B	16	0	rw	
R3B	16	0	rw	
A0B	16	0	rw	
A1B	16	0	rw	
FBB	16	0	rw	
SB	16	0	rw	
USP	16	0	rw	
ISP	16	0	rw	
INTB	20	0	rw	
PC	20	0	rw	program counter
FLG	16	0	rw	

Table 13.1: Registers at level 1