OVP Guide to Using Processor Models

Model specific information for Renesas_RH850G3M

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

Author	Imperas Software Limited
Version	20220722.0
Filename	OVP_Model_Specific_Information_rh850_RH850G3M.pdf
Created	27 July 2022
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.

Contents

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

RH850 Family Processor Model.

Reference Document : RH850G3M Rev 1.00, Aug. 2014

1.2 Licensing

Open Source Apache 2.0

1.3 Limitations

No FPU Exceptions

1.4 Verification

Models have been extensively tested by Imperas, In addition Verification suites have been supplied by Renesas for Feature Set validation

1.5 Features

RH850 Support for MPU

RH850 Vector based Exceptions.

RH850 Integer Instructions.

RH850 Floating Point Instructions.

Supervisor & User Execution Modes

Chapter 2

Configuration

2.1 Location

This model's VLNV is renesas.ovpworld.org/processor/rh850/1.0.
The model source is usually at:
$IMPERAS_HOME/ImperasLib/source/renesas.ovpworld.org/processor/rh850/1.0
The model binary is usually at:
$IMPERAS_HOME/lib/$IMPERAS_ARCH/ImperasLib/renesas.ovpworld.org/processor/rh850/1.0

2.2 GDB Path

The default GDB for this model is: $IMPERAS_HOME/lib/$IMPERAS_ARCH/gdb/v850-elf-gdb.

2.3 Semi-Host Library

The default semi-host library file is renesas.ovpworld.org/semihosting/v850Newlib/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

ELF codes supported by this model are:0x57, 0x24, 0x70f1, 0x70ff and 0x747b.

Chapter 3

All Variants in this model

This model has these variants

Variant	Description
RH850G3M	(described in this document)

Table 3.1: All Variants in this model

Chapter 4

Bus Master Ports

This model has these bus master ports.

Name	min	max	Connect?	Description
INSTRUCTION	28	32	mandatory	
DATA	28	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.

Name	Type	Connect?	Description
intp	input	optional	Interrupt Port
nmi0	input	optional	Non-Maskable Interrupt Port
nmi1	input	optional	Non-Maskable Interrupt Port
nmi2	input	optional	Non-Maskable Interrupt Port
reset	input	optional	Reset Port
mireti	output	optional	Return from Interrupt Port
intack	output	optional	Interrupt Acknowledge Port

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)
verbose	Boolean	Specify verbose output messages
GDBSIMMODE	Boolean	GDB Simulator Compatibility Mode
nofpu	Boolean	Disable Processor Internal FPU
RBASE	Uns32	RBASE register Reset vector Address
ucbank	Boolean	Enable the User Compatible Bank Registers (eg, VFOREST)
PEID	Uns32	Processor element number
SPID	Uns32	System Protection Number

Table 8.1: Parameters

8.1 Parameter values

These are the current parameter values.

Name	Value
(Others)	
variant	RH850G3M
verbose	T
GDBSIMMODE	F
nofpu	F
RBASE	0
ucbank	F
PEID	1
SPID	0

Table 8.2: Parameter values

Chapter 9

Execution Modes

Mode	Code	Description
SUPERVISOR	0	Supervisor mode
USER	1	User mode
SUPERVISOR_MPU	2	Supervisor mode MPU
USER_MPU	3	User mode MPU

Table 9.1: Modes implemented in this processor

Chapter 10

Exceptions

Exception	Code	Description
reset	0	Reset Signal Exception
syserr	16	System Error
fetrap	48	FE Level Trap
trap0	64	EI Level Trap 0
trap1	80	EI Level Trap 1
rie	96	Reserved Instruction Exception
fpp	113	FPU Exception (precise)
fpi	114	FPU Exception (imprecise)
ucpop	128	Coprocessor Unusable Exception
mip	144	Memory Protection Exception (execution privilege)
mdp	145	Memory Protection Exception (access privilege)
pie	160	Privileged Instruction Exception
debug	176	Debug Exception
mae	192	Misalignment Exception
fenmi	224	FENMI Interrupt
feint	240	FEINT Interrupt
eiint	256	EIINT Interrupt
syscall	32768	System Call

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 3 register groups:

Group name	Registers
User	32
System	100
Integration_support	8

Table 11.1: 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
-memory	String	show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system)
-mode	Boolean	show processor mode changes
-off	Boolean	disable instruction tracing
-on	Boolean	enable instruction tracing
-processorname	Boolean	Include processor name in all trace lines
-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

13.1.1 User

Registers at level:1, group:User

Name	Bits	Initial-Hex	RW	Description
R0	32	0	r-	Zero Register
R1	32	0	rw	Assembler-reserved register
R2	32	0	rw	Address/data variable register (when the real-time OS to be used is not using r2)
R3	32	0	rw	Stack pointer (SP)
R4	32	0	rw	Global pointer (GP)
R5	32	0	rw	Test pointer (TP)
R6	32	0	rw	Address/data variable registers
R7	32	0	rw	Address/data variable registers
R8	32	0	rw	Address/data variable registers
R9	32	0	rw	Address/data variable registers
R10	32	0	rw	Address/data variable registers
R11	32	0	rw	Address/data variable registers
R12	32	0	rw	Address/data variable registers
R13	32	0	rw	Address/data variable registers
R14	32	0	rw	Address/data variable registers
R15	32	0	rw	Address/data variable registers
R16	32	0	rw	Address/data variable registers
R17	32	0	rw	Address/data variable registers
R18	32	0	rw	Address/data variable registers
R19	32	0	rw	Address/data variable registers
R20	32	0	rw	Address/data variable registers
R21	32	0	rw	Address/data variable registers
R22	32	0	rw	Address/data variable registers
R23	32	0	rw	Address/data variable registers
R24	32	0	rw	Address/data variable registers
R25	32	0	rw	Address/data variable registers
R26	32	0	rw	Address/data variable registers
R27	32	0	rw	Address/data variable registers
R28	32	0	rw	Address/data variable registers
R29	32	0	rw	Address/data variable registers
R30	32	0	rw	Element pointer (EP)
R31	32	0	rw	Link pointer (LP)

Table 13.1: Registers at level 1, group:User

13.1.2 System

Registers at level:1, group:System

Name	Bits	Initial-Hex	RW	Description
EIPC	32	0	rw	Interrupt status-saving register PC
EIPSW	32	20	rw	Interrupt status-saving register PSW
FEPC	32	0	rw	NMI status-saving register PC
FEPSW	32	20	rw	NMI status-saving register PSW
PSW	32	20	rw	Program status word
EIIC	32	0	rw	EI level Cause Register
FEIC	32	0	rw	FE level Cause Register
CTPC	32	0	rw	CALLT status-saving register PC
CTPSW	32	0	rw	CALLT status-saving register PSW
CTBP	32	0	rw	CALLT base pointer
EIWR	32	0	rw	EI level exception working register
FEWR	32	0	rw	FE level exception working register
BSEL	32	0	rw	Bank Select Register (Backward compatibility)
PC	32	0	rw	Program Counter
FPSR	32	20000	rw	Floating-point configuration/status
FPEPC	32	0	rw	Floating-point exception program counter
FPST	32	0	rw	Floating-point status
FPCC	32	0	rw	Floating-point comparison result
FPCFG	32	0	rw	Floating-point configuration
FPEC	32	0	rw	Floating-point exception control
MCFG0	32	4	r-	Machine Configuration
RBASE	32	0	r-	Reset vector base address
EBASE	32	0	r-	Exception handler vector address
INTBP	32	0	r-	Base address of the interrupt handler table
MCTL	32	80000000	r-	CPU control
PID	32	0	r-	Processor ID
FPIPR	32	80000000	r-	FPI exception interrupt priority setting
SCCFG	32	0	r-	SYSCALL operation setting
SCBP	32	0	r-	SYSCALL base pointer
HTCFG0	32	18000	r-	Thread configuration
MEA	32	0	r-	Memory error address
ASID	32	0	r-	Address space ID
MEI	32	0	r-	Memory error information
ISPR	32	0	r-	Priority of interrupt being serviced
PMR	32	0	r-	Interrupt priority masking
ICSR	32	0	r-	Interrupt control status
INTCFG	32	0	r-	Interrupt function setting
ICTAGL	32	0	rw	Instruction cache tag Lo access
ICTAGH	32	0	rw	Instruction cache tag Hi access
ICDATL	32	0	rw	Instruction cache data Lo access
ICDATH	32	0	rw	Instruction cache data Hi access
ICCTRL	32	10003	rw	Instruction cache control
ICCFG	32	10000	rw	Instruction cache configuration
ICERR	32	0	rw	Instruction cache error
MPM	32	0	rw	Memory protection operation mode setting
MPRC	32	0	rw	MPU region control
MPBRGN	32	0	rw	MPU base region number
MPTRGN	32	10	rw	MPU end region number

MCA	32	0	rw	Memory protection setting check address
MCS	32	0	rw	Memory protection setting check size
MCC	32	0	rw	Memory protection setting check command
MCR	32	0	rw	Memory protection setting check result
MPLA0	32	0	rw	Protection area minimum address
MPUA0	32	0	rw	Protection area maximum address
MPAT0	32	0	rw	Protection area attribute
MPLA1	32	0	rw	Protection area minimum address
MPUA1	32	0	rw	Protection area maximum address
MPAT1	32	0	rw	Protection area attribute
MPLA2	32	0	rw	Protection area minimum address
MPUA2	32	0	rw	Protection area maximum address
MPAT2	32	0	rw	Protection area attribute
MPLA3	32	0	rw	Protection area minimum address
MPUA3	32	0	rw	Protection area maximum address
MPAT3	32	0	rw	Protection area attribute
MPLA4	32	0	rw	Protection area minimum address
MPUA4	32	0	rw	Protection area maximum address
MPAT4	32	0	rw	Protection area attribute
MPLA5	32	0	rw	Protection area minimum address
MPUA5	32	0	rw	Protection area maximum address
MPAT5	32	0	rw	Protection area attribute
MPLA6	32	0	rw	Protection area minimum address
MPUA6	32	0	rw	Protection area maximum address
MPAT6	32	0	rw	Protection area attribute
MPLA7	32	0	rw	Protection area minimum address
MPUA7	32	0	rw	Protection area maximum address
MPAT7	32	0	rw	Protection area attribute
MPLA8	32	0	rw	Protection area minimum address
MPUA8	32	0	rw	Protection area maximum address
MPAT8	32	0	rw	Protection area attribute
MPLA9	32	0	rw	Protection area minimum address
MPUA9	32	0	rw	Protection area maximum address
MPAT9	32	0	rw	Protection area attribute
MPLA10	32	0	rw	Protection area minimum address
MPUA10	32	0	rw	Protection area maximum address
MPAT10	32	0	rw	Protection area attribute
MPLA11	32	0	rw	Protection area minimum address
MPUA11	32	0	rw	Protection area maximum address
MPAT11	32	0	rw	Protection area attribute
MPLA12	32	0	rw	Protection area minimum address
MPUA12	32	0	rw	Protection area maximum address
MPAT12	32	0	rw	Protection area attribute
MPLA13	32	0	rw	Protection area minimum address
MPUA13	32	0	rw	Protection area maximum address
MPAT13	32	0	rw	Protection area attribute
MPLA14	32	0	rw	Protection area minimum address
MPUA14	32	0	rw	Protection area maximum address
MPAT14	32	0	rw	Protection area attribute
MPLA15	32	0	rw	Protection area minimum address
MPUA15	32	0	rw	Protection area maximum address
MPAT15	32	0	rw	Protection area attribute

Table 13.2: Registers at level 1, group:System

13.1.3 Integration_support

Registers at level:1, group:Integration_support

Name	Bits	Initial-Hex	RW	Description
stop	32	0	rw	Support Register for Stopping Processor
ResultHi	32	0	rw	External ResultHi Register
ResultLo	32	0	rw	External ResultLo Register
ResultSz	8	0	rw	External ResultSz Register
ResultId	8	0	rw	External ResultId Register
ResultFlg	32	0	rw	External ResultFlg Register
ResultCC	32	0	rw	External ResultCC Register
FLG_LL	8	0	rw	Load/Store Exclusive FLG_LL

Table 13.3: Registers at level 1, group:Integration_support