OVP Guide to Using Processor Models

Model Specific Information
for variant ARM_ARM966E

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

Author	Imperas Software Limited
Version	0.4
Filename	OVP_Model_Specific_Information_arm_ARM966E.pdf
Created	2 April 2013

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

Copyright Notice

Copyright © 2013 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 reader's 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.

Table of Contents

1.0 Overview.... 4
1.1 Description.... 4
1.2 Licensing.... 4
1.3 Verification.... 4
1.4 Features.... 4
2.0 Configuration.... 4
2.1 Location.... 4
2.2 GDB Path.... 4
2.3 Semi-Host Library.... 4
2.4 Processor Endian-ness.... 4
2.5 Processor ELF Code.... 4
3.0 Other Variants in this Model.... 4
4.0 Bus Ports.... 6
5.0 Net Ports.... 6
6.0 FIFO Ports.... 6
7.0 Parameters.... 6
8.0 Execution Modes.... 9
9.0 Exceptions.... 9
10.0 Hierarchy of the model.... 10
10.1 Level 1: CPU.... 10
11.0 Model Commands.... 11
11.1 Level 1: CPU.... 11
12.0 Registers.... 11
12.1 Level 1: CPU.... 11
12.1.1 Core.... 11
12.1.2 Control.... 11
12.1.3 User.... 12
12.1.4 FIQ.... 12
12.1.5 IRQ.... 12
12.1.6 Supervisor.... 12
12.1.7 Undefined.... 13
12.1.8 Abort.... 13
12.1.9 Coprocessor_32_bit.... 13
12.1.10 Integration_support.... 13

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

1.0 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

ARM Processor Model

1.2 Licensing

Imperas Modified Apache 2.0 Open Source License

1.3 Verification

Models have been extensively tested by Imperas. ARM9 models have been successfully used by customers to simulate Linux and Nucleus on ArmIntegrator virtual platforms.

1.4 Features

Thumb instructions are supported.

2.0 Configuration

2.1 Location

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

2.2 GDB Path

The default GDB for this model is found at:
    $IMPERAS_HOME/lib/$IMPERAS_ARCH/gdb/arm-elf-gdb

2.3 Semi-Host Library

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

2.4 Processor Endian-ness

This model can be set to either endian-ness (normally by a pin, or the ELF code).

2.5 Processor ELF Code

The ELF code supported by this model is: 0x28

3.0 Other Variants in this Model

Table 1.

Variant
ARMv4T
ARMv4xM
ARMv4
ARMv4TxM
ARMv5xM
ARMv5
ARMv5TxM
ARMv5T
ARMv5TExP
ARMv5TE
ARMv5TEJ
ARMv6
ARMv6K
ARMv6T2
ARMv6KZ
ARMv7
ARM7TDMI
ARM7EJ-S
ARM720T
ARM920T
ARM922T
ARM926EJ-S
ARM940T
ARM946E
ARM966E
ARM968E-S
ARM1020E
ARM1022E
ARM1026EJ-S
ARM1136J-S
ARM1156T2-S
ARM1176JZ-S
Cortex-R4
Cortex-R4F
Cortex-A5UP
Cortex-A5MPx1
Cortex-A5MPx2
Cortex-A5MPx3
Cortex-A5MPx4
Cortex-A8

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

Cortex-A9UP
Cortex-A9MPx1
Cortex-A9MPx2
Cortex-A9MPx3
Cortex-A9MPx4
Cortex-A7UP
Cortex-A7MPx1
Cortex-A7MPx2
Cortex-A7MPx3
Cortex-A7MPx4
Cortex-A15UP
Cortex-A15MPx1
Cortex-A15MPx2
Cortex-A15MPx3
Cortex-A15MPx4

4.0 Bus Ports

Table 2.

Type	Name	Bits
master (initiator)	INSTRUCTION	32
master (initiator)	DATA	32

5.0 Net Ports

Table 3.

Name	Type	Description
fiq	input	Fast external interrupt (active high)
irq	input	External interrupt (active high)
reset	input	Processor reset (active high)

6.0 FIFO Ports

No FIFO Ports in this model.

7.0 Parameters

Table 4.

Name	Type	Description
verbose	Boolean	Specify verbosity of output
showHiddenRegs	Boolean	Show hidden registers during register tracing
UAL	Boolean	Disassemble using UAL syntax
enableVFPAtReset	Boolean	Enable vector floating point (SIMD and VFP) instructions at reset. (Enables cp10/11 in CPACR and sets FPEXC.EN)

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

compatibility	Enumeration	Specify compatibility mode ISA=0 gdb=1 nopSVC=2
override_debugMask	Uns32	Specifies debug mask, enabling debug output for model components
override_numCPUs	Uns32	Specify the number of cores in a multiprocessor
override_fcsePresent	Boolean	Specifies that FCSE is present (if true)
override_fpexcDexPresent	Boolean	Specifies that the FPEXC.DEX register field is implemented (if true)
override_advSIMDPresent	Boolean	Specifies that Advanced SIMD extensions are present (if true)
override_vfpPresent	Boolean	Specifies that VFP extensions are present (if true)
override_physicalBits	Uns32	Specifies the implemented physical bus bits (defaults to connected physical bus width)
override_genericTimerScale	Uns32	Specifies generic counter rate as fraction of MIPS rate (e.g. 10 implies 10 times slower)
override_SCTLR_V	Boolean	Override SCTLR.V with the passed value (enables high vectors)
override_SCTLR_CP15BEN_Present	Boolean	Enable ARMv7 SCTLR.CP15BEN bit (CP15 barrier enable)
override_MIDR	Uns32	Override MIDR register
override_CTR	Uns32	Override CTR register
override_TLBTR	Uns32	Override TLBTR register
override_MPUIR	Uns32	Override MPUIR register
override_CLIDR	Uns32	Override CLIDR register
override_AIDR	Uns32	Override AIDR register
override_CBAR	Uns32	Override Configuration Base Address Register
override_PFR0	Uns32	Override ID_PFR0 register
override_PFR1	Uns32	Override ID_PFR1 register
override_DFR0	Uns32	Override ID_DFR0 register
override_AFR0	Uns32	Override ID_AFR0 register
override_MMFR0	Uns32	Override ID_MMFR0 register
override_MMFR1	Uns32	Override ID_MMFR1 register
override_MMFR2	Uns32	Override ID_MMFR2 register
override_MMFR3	Uns32	Override ID_MMFR3 register
override_ISAR0	Uns32	Override ID_ISAR0 register
override_ISAR1	Uns32	Override ID_ISAR1 register
override_ISAR2	Uns32	Override ID_ISAR2 register
override_ISAR3	Uns32	Override ID_ISAR3 register
override_ISAR4	Uns32	Override ID_ISAR4 register
override_ISAR5	Uns32	Override ID_ISAR5 register
override_SACTLR	Uns32	Override Cortex-R5 SACTLR register (not functionally significant in the model)

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

override_BuildOptions0	Uns32	Override Cortex-R5 BuildOptions0 register (not functionally significant in the model)
override_BuildOptions1	Uns32	Override Cortex-R5 BuildOptions1 register (not functionally significant in the model)
override_FPSID	Uns32	Override SIMD/VFP FPSID register
override_MVFR0	Uns32	Override SIMD/VFP MVFR0 register
override_MVFR1	Uns32	Override SIMD/VFP MVFR1 register
override_GICC_IIDR	Uns32	Override GICC_IIDR register
override_GICD_TYPER_ITLines	Uns32	Override ITLinesNumber field of GICD_TYPER register (aka ICDICTR)
override_GICH_VTR	Uns32	Override GICH_VTR register
override_ICCPMRBits	Uns32	Specify the number of writable bits in ICCPMR
override_minICCBPR	Uns32	Specify the minimum possible value for ICCBPR
override_ERG	Uns32	Specifies exclusive reservation granule
override_STRoffsetPC12	Boolean	Specifies that STR/STR of PC should do so with 12:byte offset from the current instruction (if true), otherwise an 8:byte offset is used
override_mpuV5ExtAP	Boolean	Specifies that ARMv5 MPU extended access permissions implemented (cp15/5/2 and cp15/5/2)
override_fcseRequiresMMU	Boolean	Specifies that FCSE is active only when MMU is enabled (if true)
override_ignoreBadCp15	Boolean	Specifies whether invalid coprocessor 15 access should be ignored (if true) or cause Invalid Instruction exceptions (if false)
override_SGIDisable	Boolean	Override whether SGI disable possible using ICDICER0
override_condUndefined	Boolean	Force undefined instructions to take Undefined Instruction exception even if they are conditional
override_deviceStrongAligned	Boolean	Force accesses to Device and Strongly Ordered regions to be aligned
override_Control_V	Boolean	Override SCTLR.V with the passed value (deprecated, use override_SCTLR_V)
override_MainId	Uns32	Override MIDR register (deprecated, use override_MIDR)
override_CacheType	Uns32	Override CTR register (deprecated, use override_CTR)
override_TLBType	Uns32	Override TLBTR register (deprecated, use override_TLBTR)
override_MPUType	Uns32	Override MPUIR register (deprecated, use override_MPUIR)
override_InstructionAttributes0	Uns32	Override ID_ISAR0 register (deprecated, use override_ISAR0)
override_InstructionAttributes1	Uns32	Override ID_ISAR1 register (deprecated, use override_ISAR1)

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

override_InstructionAttributes2	Uns32	Override ID_ISAR2 register (deprecated, use override_ISAR2)
override_InstructionAttributes3	Uns32	Override ID_ISAR3 register (deprecated, use override_ISAR3)
override_InstructionAttributes4	Uns32	Override ID_ISAR4 register (deprecated, use override_ISAR4)
override_InstructionAttributes5	Uns32	Override ID_ISAR5 register (deprecated, use override_ISAR5)

8.0 Execution Modes

Table 5.

Name	Code
User	16
FIQ	17
IRQ	18
Supervisor	19
Abort	23
Undefined	27
System	31

9.0 Exceptions

Table 6.

Name	Code
Reset	0
Undefined	1
SupervisorCall	2
PrefetchAbort	5
DataAbort	6
IRQ	8
FIQ	9

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

10.0 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: CPU

This level in the model hierarchy has 2 commands.

This level in the model hierarchy has 10 register groups:

Table 7.

Group name	Registers
Core	16
Control	3
User	7
FIQ	8
IRQ	3
Supervisor	3
Undefined	3
Abort	3
Coprocessor_32_bit	6
Integration_support	1

This level in the model hierarchy has no children.

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

11.0 Model Commands

11.1 Level 1: CPU

Table 8.

Name	Arguments
debugflags	
dumpTLB	

12.0 Registers

12.1 Level 1: CPU

12.1.1 Core

Table 9.

Name	Bits	Initial value (Hex)		Description
r0	32	0	rw	
r1	32	0	rw	
r2	32	0	rw	
r3	32	0	rw	
r4	32	0	rw	
r5	32	0	rw	
r6	32	0	rw	
r7	32	0	rw	
r8	32	0	rw	
r9	32	0	rw	
r10	32	0	rw	
r11	32	0	rw	frame pointer
r12	32	0	rw	
sp	32	0	rw	stack pointer
lr	32	0	rw	
pc	32	0	rw	program counter

12.1.2 Control

Table 10.

Name	Bits	Initial value (Hex)		Description
fps	32	0	rw	archaic FPSCR view (for gdb)
cpsr	32	d3	rw	
spsr	32	0	rw	

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

12.1.3 User

Table 11.

Name	Bits	Initial value (Hex)		Description
r8_usr	32	0	rw	
r9_usr	32	0	rw	
r10_usr	32	0	rw	
r11_usr	32	0	rw	
r12_usr	32	0	rw	
sp_usr	32	0	rw	
lr_usr	32	0	rw	

12.1.4 FIQ

Table 12.

Name	Bits	Initial value (Hex)		Description
r8_fiq	32	0	rw	
r9_fiq	32	0	rw	
r10_fiq	32	0	rw	
r11_fiq	32	0	rw	
r12_fiq	32	0	rw	
sp_fiq	32	0	rw	
lr_fiq	32	0	rw	
spsr_fiq	32	0	rw	

12.1.5 IRQ

Table 13.

Name	Bits	Initial value (Hex)		Description
sp_irq	32	0	rw	
lr_irq	32	0	rw	
spsr_irq	32	0	rw	

12.1.6 Supervisor

Table 14.

Name	Bits	Initial value (Hex)		Description

sp_svc	32	0	rw	
lr_svc	32	0	rw	
spsr_svc	32	0	rw	

12.1.7 Undefined

Table 15.

Name	Bits	Initial value (Hex)		Description
sp_undef	32	0	rw	
lr_undef	32	0	rw	
spsr_undef	32	0	rw	

12.1.8 Abort

Table 16.

Name	Bits	Initial value (Hex)		Description
sp_abt	32	0	rw	
lr_abt	32	0	rw	
spsr_abt	32	0	rw	

12.1.9 Coprocessor_32_bit

Table 17.

Name	Bits	Initial value (Hex)		Description
MIDR	32	41259661	r-	mrc p15,0,Rd,c0,c0,-- PL1:r- PL0:--
SCTLR	32	f70	rw	mcr/mrc p15,0,Rd,c1,c0,0 PL1:rw PL0:--
WaitForInterrupt	32	-	-w	mcr p15,0,Rd,c7,c0,4 PL1:-w PL0:--
DataSynchBarrier	32	-	-w	mcr p15,0,Rd,c7,c10,4 PL1:-w PL0:-w
TRACEIDR	32	0	rw	mcr/mrc p15,0,Rd,c13,c--,1 PL1:rw PL0:--
WaitForInterrupt2	32	-	-w	mcr p15,0,Rd,c15,c8,2 PL1:-w PL0:--

12.1.10 Integration_support

Table 18.

Name	Bits	Initial value (Hex)		Description
transactPL	32	1	r-	privilege level of current memory transaction

Copyright © 2013 Imperas Software Limited. www.OVPworld.org

#