Imperas Peripheral Model Guide

Model Specific Information for xilinx.ovpworld.org / zynq_7000-dmac

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_Peripheral_Specific_Information_zynq_7000-dmac.pdf
Created	05 May 2021
Status	OVP Standard Release

Copyright Notice

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.

Table Of Contents

1.0 Model Specific Information 4
1.1 Description 4
1.2 Licensing 4
1.3 Limitations 4
1.4 Reference 4
1.5 Location 4
2.0 Net Ports 4
3.0 Bus Slave Ports 4
3.1 Bus Slave Port: bportS 4
3.2 Bus Slave Port: bportNS 8
4.0 Platforms that use this peripheral component 12
5.0 Peripheral components in the library 14
6.0 General Information on Peripheral Models 16
6.1 Background 16
7.0 Building peripherals easily with Imperas iGen 16
8.0 Peripheral model internals 16
9.0 Parts of peripheral models 17
9.1 Configuring the Peripheral Instance with Parameters 17
9.2 Net Ports 17
9.3 Bus master ports 17
9.4 Bus slave ports 17
9.5 Packetnets 17
10.0 More information (documentation) on peripheral models and modeling 17

1.0 Model Specific Information

This document provides usage information for an Imperas OVP peripheral behavioral model.

The document is split into sections providing specific information for this peripheral, including any ports for connecting into a platform, registers, other component parts, and configuration options and general information for peripheral modeling with Imperas OVP.

1.1 Description

Zynq 7000 Platform DMA Controller (DMAC)

1.2 Licensing

Open Source Apache 2.0

1.3 Limitations

This model implements the full set of registers. There is no behavior included.

1.4 Reference

Zynq-7000 TRM
(https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf)

1.5 Location

The zynq_7000-dmac peripheral model is located in an Imperas/OVP installation at the VLNV:
xilinx.ovpworld.org / peripheral / zynq_7000-dmac / 1.0.

2.0 Net Ports

This model has the following net ports:

Table 1. Net Ports

Name	Type	Must Be Connected	Description
Int0	output	F (False)	DMAC Interrupt
Int1	output	F (False)	DMAC Interrupt
Int2	output	F (False)	DMAC Interrupt
Int3	output	F (False)	DMAC Interrupt
Int4	output	F (False)	DMAC Interrupt
Int5	output	F (False)	DMAC Interrupt
Int6	output	F (False)	DMAC Interrupt
Int7	output	F (False)	DMAC Interrupt
inta	output	F (False)	Interrupt DMAC Transfer Abort

3.0 Bus Slave Ports

This model has the following bus slave ports:

3.1 Bus Slave Port: bportS

Table 2. Bus Slave Port: bportS

Name	Size (bytes)	Must Be Connected	Description
bportS	0x1000	T (True)	

Table 3. Bus Slave Port: bportS Registers:

Name	Offset	Width (bits)	Description	R/W	is Volatile
abS_DSR	0x0	32	DMA Manager Status		
abS_DPC	0x4	32	DMA Program Counter		
abS_INTEN	0x20	32	DMASEV Instruction Response Control		
abS_INT_EVENT_RIS	0x24	32	Event Interrupt Raw Status		
abS_INTMIS	0x28	32	Interrupt Status		
abS_INTCLR	0x2c	32	Interrupt Clear		
abS_FSRD	0x30	32	Fault Status DMA Manager		
abS_FSRC	0x34	32	Fault Status DMA Channel		
abS_FTRD	0x38	32	Fault Type DMA Manager		
abS_FTR0	0x40	32	Default Type DMA Channel 0		
abS_FTR1	0x44	32	Default Type DMA Channel 1		
abS_FTR2	0x48	32	Default Type DMA Channel 2		
abS_FTR3	0x4c	32	Default Type DMA Channel 3		
abS_FTR4	0x50	32	Default Type DMA Channel 4		
abS_FTR5	0x54	32	Default Type DMA Channel 5		
abS_FTR6	0x58	32	Default Type DMA Channel 6		
abS_FTR7	0x5c	32	Default Type DMA Channel 7		
abS_CSR0	0x100	32	Channel Status DMA Channel 0		
abS_CPC0	0x104	32	Channel PC for DMA Channel 0		
abS_CSR1	0x108	32	Channel Status DMA Channel 1		
abS_CPC1	0x10c	32	Channel PC for DMA Channel 1		
abS_CSR2	0x110	32	Channel Status DMA Channel 2		
abS_CPC2	0x114	32	Channel PC for DMA Channel 2		
abS_CSR3	0x118	32	Channel Status DMA Channel 3		
abS_CPC3	0x11c	32	Channel PC for DMA Channel 3		
abS_CSR4	0x120	32	Channel Status DMA Channel 4		

abS_CPC4	0x124	32	Channel PC for DMA Channel 4		
abS_CSR5	0x128	32	Channel Status DMA Channel 5		
abS_CPC5	0x12c	32	Channel PC for DMA Channel 5		
abS_CSR6	0x130	32	Channel Status DMA Channel 6		
abS_CPC6	0x134	32	Channel PC for DMA Channel 6		
abS_CSR7	0x138	32	Channel Status DMA Channel 7		
abS_CPC7	0x13c	32	Channel PC for DMA Channel 7		
abS_SAR0	0x400	32	Source Address DMA Channel 0		
abS_DAR0	0x404	32	Destination Addr DMA Channel 0		
abS_CCR0	0x408	32	Channel Control DMA Channel 0 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_0	0x40c	32	Loop Counter 0 DMA Channel 0		
abS_LC1_0	0x410	32	Loop Counter 1 DMA Channel 0		
abS_SAR1	0x420	32	Source address DMA Channel 1		
abS_DAR1	0x424	32	Destination Addr DMA Channel 1		
abS_CCR1	0x428	32	Channel Control DMA Channel 1 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_1	0x42c	32	Loop Counter 0 DMA Channel 1		
abS_LC1_1	0x430	32	Loop Counter 1 DMA Channel 1		
abS_SAR2	0x440	32	Source Address DMA Channel 2		
abS_DAR2	0x444	32	Destination Addr DMA Channel 2		
abS_CCR2	0x448	32	Channel Control DMA Channel 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_2	0x44c	32	Loop Counter 0 DMA Channel 2		
abS_LC1_2	0x450	32	Loop Counter 1 DMA Channel 2		
abS_SAR3	0x460	32	Source Address DMA Channel 3		
abS_DAR3	0x464	32	Destination Addr DMA Channel 3		
abS_CCR3	0x468	32	Channel Control DMA Channel 3 (dmac0_ns:		

			0x00000000 dmac0_s: 0x00800200)		
abS_LC0_3	0x46c	32	Loop Counter 0 DMA Channel 3		
abS_LC1_3	0x470	32	Loop Counter 1 DMA Channel 3		
abS_SAR4	0x480	32	Source Address DMA Channel 4		
abS_DAR4	0x484	32	Destination Addr DMA Channel 4		
abS_CCR4	0x488	32	Channel Control DMA Channel 4 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_4	0x48c	32	Loop Counter 0 DMA Channel 4		
abS_LC1_4	0x490	32	Loop Counter 1 DMA Channel 4		
abS_SAR5	0x4a0	32	Source Address DMA Channel 5		
abS_DAR5	0x4a4	32	Destination Addr DMA Channel 5		
abS_CCR5	0x4a8	32	Channel Control DMA Channel 5 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_5	0x4ac	32	Loop Counter 0 DMA Channel 5		
abS_LC1_5	0x4b0	32	Loop Counter 1 DMA Channel 5		
abS_SAR6	0x4c0	32	Source Address DMA Channel 6		
abS_DAR6	0x4c4	32	Destination Addr DMA Channel 6		
abS_CCR6	0x4c8	32	Channel Control DMA Channel 6 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_6	0x4cc	32	Loop Counter 0 DMA Channel 6		
abS_LC1_6	0x4d0	32	Loop Counter 1 DMA Channel 6		
abS_SAR7	0x4e0	32	Source Address DMA Channel 7		
abS_DAR7	0x4e4	32	Destination Addr DMA Channel 7		
abS_CCR7	0x4e8	32	Channel Control DMA Channel 7 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abS_LC0_7	0x4ec	32	Loop Counter 0 DMA Channel 7		
abS_LC1_7	0x4f0	32	Loop Counter 1 DMA Channel 7		
abS_DBGSTATUS	0xd00	32	DMA Manager Execution Status		

abS_DBGCMD	0xd04	32	DMA Manager Instr. command		
abS_DBGINST0	0xd08	32	DMA Manager Instruction Part A		
abS_DBGINST1	0xd0c	32	DMA Manager Instruction Part B		
abS_CR0	0xe00	32	Config. 0: Events, Peripheral Interfaces, PC, Mode (dmac0_ns: 0x00000000 dmac0_s: 0x001E3071)		
abS_CR1	0xe04	32	Config. 1: Instruction Cache (dmac0_ns: 0x00000000 dmac0_s: 0x00000074)		
abS_CR2	0xe08	32	Config. 2: DMA Mgr Boot Addr		
abS_CR3	0xe0c	32	Config. 3: Security state of IRQs		
abS_CR4	0xe10	32	Config 4, Security of Periph Interfaces		
abS_CRD	0xe14	32	DMA configuration (dmac0_ns: 0x00000000 dmac0_s: 0x07FF7F73)		
abS_WD	0xe80	32	Watchdog Timer		
abS_periph_id_0	0xfe0	32	Peripheral Identification register 0 (dmac0_ns: 0x00000000 dmac0_s: 0x00000030)		
abS_periph_id_1	0xfe4	32	Peripheral Identification register 1 (dmac0_ns: 0x00000000 dmac0_s: 0x00000013)		
abS_periph_id_2	0xfe8	32	Peripheral Identification register 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00000024)		
abS_periph_id_3	0xfec	32	Peripheral Identification register 3		
abS_pcell_id_0	0xff0	32	Component Identification register 0 (dmac0_ns: 0x00000000 dmac0_s: 0x0000000D)		
abS_pcell_id_1	0xff4	32	Component Identification register 1 (dmac0_ns: 0x00000000 dmac0_s: 0x000000F0)		
abS_pcell_id_2	0xff8	32	Component Identification register 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00000005)		
abS_pcell_id_3	0xffc	32	Component Identification register 3 (dmac0_ns: 0x00000000 dmac0_s: 0x000000B1)		

3.2 Bus Slave Port: bportNS

Table 4. Bus Slave Port: bportNS

Name	Size (bytes)	Must Be Connected	Description
bportNS	0x1000	T (True)	

Table 5. Bus Slave Port: bportNS Registers:

Name	Offset	Width (bits)	Description	R/W	is Volatile
abNS_DSR	0x0	32	DMA Manager Status		
abNS_DPC	0x4	32	DMA Program Counter		
abNS_INTEN	0x20	32	DMASEV Instruction Response Control		
abNS_INT_EVENT_RIS	0x24	32	Event Interrupt Raw Status		
abNS_INTMIS	0x28	32	Interrupt Status		
abNS_INTCLR	0x2c	32	Interrupt Clear		
abNS_FSRD	0x30	32	Fault Status DMA Manager		
abNS_FSRC	0x34	32	Fault Status DMA Channel		
abNS_FTRD	0x38	32	Fault Type DMA Manager		
abNS_FTR0	0x40	32	Default Type DMA Channel 0		
abNS_FTR1	0x44	32	Default Type DMA Channel 1		
abNS_FTR2	0x48	32	Default Type DMA Channel 2		
abNS_FTR3	0x4c	32	Default Type DMA Channel 3		
abNS_FTR4	0x50	32	Default Type DMA Channel 4		
abNS_FTR5	0x54	32	Default Type DMA Channel 5		
abNS_FTR6	0x58	32	Default Type DMA Channel 6		
abNS_FTR7	0x5c	32	Default Type DMA Channel 7		
abNS_CSR0	0x100	32	Channel Status DMA Channel 0		
abNS_CPC0	0x104	32	Channel PC for DMA Channel 0		
abNS_CSR1	0x108	32	Channel Status DMA Channel 1		
abNS_CPC1	0x10c	32	Channel PC for DMA Channel 1		
abNS_CSR2	0x110	32	Channel Status DMA Channel 2		
abNS_CPC2	0x114	32	Channel PC for DMA Channel 2		
abNS_CSR3	0x118	32	Channel Status DMA Channel 3		
abNS_CPC3	0x11c	32	Channel PC for DMA Channel 3		
abNS_CSR4	0x120	32	Channel Status DMA Channel 4		

abNS_CPC4	0x124	32	Channel PC for DMA Channel 4		
abNS_CSR5	0x128	32	Channel Status DMA Channel 5		
abNS_CPC5	0x12c	32	Channel PC for DMA Channel 5		
abNS_CSR6	0x130	32	Channel Status DMA Channel 6		
abNS_CPC6	0x134	32	Channel PC for DMA Channel 6		
abNS_CSR7	0x138	32	Channel Status DMA Channel 7		
abNS_CPC7	0x13c	32	Channel PC for DMA Channel 7		
abNS_SAR0	0x400	32	Source Address DMA Channel 0		
abNS_DAR0	0x404	32	Destination Addr DMA Channel 0		
abNS_CCR0	0x408	32	Channel Control DMA Channel 0 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_0	0x40c	32	Loop Counter 0 DMA Channel 0		
abNS_LC1_0	0x410	32	Loop Counter 1 DMA Channel 0		
abNS_SAR1	0x420	32	Source address DMA Channel 1		
abNS_DAR1	0x424	32	Destination Addr DMA Channel 1		
abNS_CCR1	0x428	32	Channel Control DMA Channel 1 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_1	0x42c	32	Loop Counter 0 DMA Channel 1		
abNS_LC1_1	0x430	32	Loop Counter 1 DMA Channel 1		
abNS_SAR2	0x440	32	Source Address DMA Channel 2		
abNS_DAR2	0x444	32	Destination Addr DMA Channel 2		
abNS_CCR2	0x448	32	Channel Control DMA Channel 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_2	0x44c	32	Loop Counter 0 DMA Channel 2		
abNS_LC1_2	0x450	32	Loop Counter 1 DMA Channel 2		
abNS_SAR3	0x460	32	Source Address DMA Channel 3		
abNS_DAR3	0x464	32	Destination Addr DMA Channel 3		
abNS_CCR3	0x468	32	Channel Control DMA Channel 3 (dmac0_ns:		

			0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_3	0x46c	32	Loop Counter 0 DMA Channel 3		
abNS_LC1_3	0x470	32	Loop Counter 1 DMA Channel 3		
abNS_SAR4	0x480	32	Source Address DMA Channel 4		
abNS_DAR4	0x484	32	Destination Addr DMA Channel 4		
abNS_CCR4	0x488	32	Channel Control DMA Channel 4 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_4	0x48c	32	Loop Counter 0 DMA Channel 4		
abNS_LC1_4	0x490	32	Loop Counter 1 DMA Channel 4		
abNS_SAR5	0x4a0	32	Source Address DMA Channel 5		
abNS_DAR5	0x4a4	32	Destination Addr DMA Channel 5		
abNS_CCR5	0x4a8	32	Channel Control DMA Channel 5 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_5	0x4ac	32	Loop Counter 0 DMA Channel 5		
abNS_LC1_5	0x4b0	32	Loop Counter 1 DMA Channel 5		
abNS_SAR6	0x4c0	32	Source Address DMA Channel 6		
abNS_DAR6	0x4c4	32	Destination Addr DMA Channel 6		
abNS_CCR6	0x4c8	32	Channel Control DMA Channel 6 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_6	0x4cc	32	Loop Counter 0 DMA Channel 6		
abNS_LC1_6	0x4d0	32	Loop Counter 1 DMA Channel 6		
abNS_SAR7	0x4e0	32	Source Address DMA Channel 7		
abNS_DAR7	0x4e4	32	Destination Addr DMA Channel 7		
abNS_CCR7	0x4e8	32	Channel Control DMA Channel 7 (dmac0_ns: 0x00000000 dmac0_s: 0x00800200)		
abNS_LC0_7	0x4ec	32	Loop Counter 0 DMA Channel 7		
abNS_LC1_7	0x4f0	32	Loop Counter 1 DMA Channel 7		
abNS_DBGSTATUS	0xd00	32	DMA Manager Execution Status		

abNS_DBGCMD	0xd04	32	DMA Manager Instr. command		
abNS_DBGINST0	0xd08	32	DMA Manager Instruction Part A		
abNS_DBGINST1	0xd0c	32	DMA Manager Instruction Part B		
abNS_CR0	0xe00	32	Config. 0: Events, Peripheral Interfaces, PC, Mode (dmac0_ns: 0x00000000 dmac0_s: 0x001E3071)		
abNS_CR1	0xe04	32	Config. 1: Instruction Cache (dmac0_ns: 0x00000000 dmac0_s: 0x00000074)		
abNS_CR2	0xe08	32	Config. 2: DMA Mgr Boot Addr		
abNS_CR3	0xe0c	32	Config. 3: Security state of IRQs		
abNS_CR4	0xe10	32	Config 4, Security of Periph Interfaces		
abNS_CRD	0xe14	32	DMA configuration (dmac0_ns: 0x00000000 dmac0_s: 0x07FF7F73)		
abNS_WD	0xe80	32	Watchdog Timer		
abNS_periph_id_0	0xfe0	32	Peripheral Identification register 0 (dmac0_ns: 0x00000000 dmac0_s: 0x00000030)		
abNS_periph_id_1	0xfe4	32	Peripheral Identification register 1 (dmac0_ns: 0x00000000 dmac0_s: 0x00000013)		
abNS_periph_id_2	0xfe8	32	Peripheral Identification register 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00000024)		
abNS_periph_id_3	0xfec	32	Peripheral Identification register 3		
abNS_pcell_id_0	0xff0	32	Component Identification register 0 (dmac0_ns: 0x00000000 dmac0_s: 0x0000000D)		
abNS_pcell_id_1	0xff4	32	Component Identification register 1 (dmac0_ns: 0x00000000 dmac0_s: 0x000000F0)		
abNS_pcell_id_2	0xff8	32	Component Identification register 2 (dmac0_ns: 0x00000000 dmac0_s: 0x00000005)		
abNS_pcell_id_3	0xffc	32	Component Identification register 3 (dmac0_ns: 0x00000000 dmac0_s: 0x000000B1)		

4.0 Platforms that use this peripheral component

Peripheral components can be used in many different platforms, including those developed by Imperas or by other users of OVP. You can use this peripheral in your own platforms.

Table 6. Publicly available platforms using peripheral 'zynq_7000-dmac'

Platform Name	Vendor
Zynq_PS	xilinx.ovpworld.org

5.0 Peripheral components in the library

Table 7. Publicly available Imperas/OVP peripheral models (224 models)

Peripheral	Peripheral	Peripheral
xilinx.ovpworld.org/zynq_7000-gpio	xilinx.ovpworld.org/zynq_7000-iic	xilinx.ovpworld.org/zynq_7000-ocm
xilinx.ovpworld.org/zynq_7000-qos301	xilinx.ovpworld.org/zynq_7000-qspi	xilinx.ovpworld.org/zynq_7000-sdio
xilinx.ovpworld.org/zynq_7000-slcr	xilinx.ovpworld.org/zynq_7000-spi	xilinx.ovpworld.org/zynq_7000-swdt
xilinx.ovpworld.org/zynq_7000-ttc	xilinx.ovpworld.org/zynq_7000-tz_GPVsecurity	xilinx.ovpworld.org/zynq_7000-tz_security
xilinx.ovpworld.org/zynq_7000-usb	altera.ovpworld.org/dw-apb-timer	altera.ovpworld.org/dw-apb-uart
altera.ovpworld.org/IntervalTimer32Core	altera.ovpworld.org/IntervalTimer64Core	altera.ovpworld.org/JtagUart
altera.ovpworld.org/PerformanceCounterCore	altera.ovpworld.org/RSTMGR	altera.ovpworld.org/SystemIDCore
altera.ovpworld.org/Uart	amd.ovpworld.org/79C970	andes.ovpworld.org/ATCUART100
andes.ovpworld.org/NCEPLIC100	andes.ovpworld.org/NCEPLMT100	arm.ovpworld.org/AaciPL041
arm.ovpworld.org/CompactFlashRegs	arm.ovpworld.org/CoreModule9x6	arm.ovpworld.org/DebugLedAndDipSwitch
arm.ovpworld.org/DMemCtrlPL341	arm.ovpworld.org/IcpControl	arm.ovpworld.org/IcpCounterTimer
arm.ovpworld.org/IntICP	arm.ovpworld.org/IntICP	arm.ovpworld.org/KbPL050
arm.ovpworld.org/L2CachePL310	arm.ovpworld.org/LcdPL110	arm.ovpworld.org/MmciPL181
arm.ovpworld.org/RtcPL031	arm.ovpworld.org/SerBusDviRegs	arm.ovpworld.org/SmartLoaderArm64Linux
arm.ovpworld.org/SmartLoaderArmLinux	arm.ovpworld.org/SMemCtrlPL354	arm.ovpworld.org/SysCtrlSP810
arm.ovpworld.org/TimerSP804	arm.ovpworld.org/TzpcBP147	arm.ovpworld.org/UartPL011
arm.ovpworld.org/VexpressSysRegs	arm.ovpworld.org/WdtSP805	atmel.ovpworld.org/AdvancedInterruptController
atmel.ovpworld.org/ParallelIOController	atmel.ovpworld.org/PowerSaving	atmel.ovpworld.org/SpecialFunction
atmel.ovpworld.org/TimerCounter	atmel.ovpworld.org/UsartInterface	atmel.ovpworld.org/WatchdogTimer
cadence.ovpworld.org/gem	cadence.ovpworld.org/uart	cirrus.ovpworld.org/GD5446
freescale.ovpworld.org/KinetisADC	freescale.ovpworld.org/KinetisAIPS	freescale.ovpworld.org/KinetisAXBS
freescale.ovpworld.org/KinetisCAN	freescale.ovpworld.org/KinetisCMP	freescale.ovpworld.org/KinetisCMT
freescale.ovpworld.org/KinetisCRC	freescale.ovpworld.org/KinetisDAC	freescale.ovpworld.org/KinetisDDR
freescale.ovpworld.org/KinetisDMA	freescale.ovpworld.org/KinetisDMAC	freescale.ovpworld.org/KinetisDMAMUX
freescale.ovpworld.org/KinetisENET	freescale.ovpworld.org/KinetisEWM	freescale.ovpworld.org/KinetisFB
freescale.ovpworld.org/KinetisFMC	freescale.ovpworld.org/KinetisFTFE	freescale.ovpworld.org/KinetisFTM
freescale.ovpworld.org/KinetisGPIO	freescale.ovpworld.org/KinetisI2C	freescale.ovpworld.org/KinetisI2S
freescale.ovpworld.org/KinetisLLWU	freescale.ovpworld.org/KinetisLPTMR	freescale.ovpworld.org/KinetisMCG
freescale.ovpworld.org/KinetisMPU	freescale.ovpworld.org/KinetisNFC	freescale.ovpworld.org/KinetisOSC
freescale.ovpworld.org/KinetisPDB	freescale.ovpworld.org/KinetisPIT	freescale.ovpworld.org/KinetisPMC
freescale.ovpworld.org/KinetisPORT	freescale.ovpworld.org/KinetisRCM	freescale.ovpworld.org/KinetisRFSYS
freescale.ovpworld.org/KinetisRFVBAT	freescale.ovpworld.org/KinetisRNG	freescale.ovpworld.org/KinetisRTC
freescale.ovpworld.org/KinetisSDHC	freescale.ovpworld.org/KinetisSIM	freescale.ovpworld.org/KinetisSMC
freescale.ovpworld.org/KinetisSPI	freescale.ovpworld.org/KinetisTSI	freescale.ovpworld.org/KinetisUART
freescale.ovpworld.org/KinetisUSB	freescale.ovpworld.org/KinetisUSBDCD	freescale.ovpworld.org/KinetisUSBHS
freescale.ovpworld.org/KinetisVREF	freescale.ovpworld.org/KinetisWDOG	freescale.ovpworld.org/Uart
freescale.ovpworld.org/VybridADC	freescale.ovpworld.org/VybridANADIG	freescale.ovpworld.org/VybridCCM
freescale.ovpworld.org/VybridDMA	freescale.ovpworld.org/VybridGPIO	freescale.ovpworld.org/VybridI2C
freescale.ovpworld.org/VybridLCD	freescale.ovpworld.org/VybridQUADSPI	freescale.ovpworld.org/VybridSDHC
freescale.ovpworld.org/VybridSPI	freescale.ovpworld.org/VybridUART	freescale.ovpworld.org/VybridUSB
imperas.ovpworld.org/frameBuffer	imperas.ovpworld.org/uart	imperas.ovpworld.org/usecCounter
intel.ovpworld.org/82077AA	intel.ovpworld.org/82371EB	intel.ovpworld.org/8253
intel.ovpworld.org/8259A	intel.ovpworld.org/NorFlash48F4400	intel.ovpworld.org/PciIDE

intel.ovpworld.org/PciPM	intel.ovpworld.org/PciUSB	intel.ovpworld.org/Ps2Control
marvell.ovpworld.org/GT6412x	maxim.ovpworld.org/max673x	microsemi.ovpworld.org/CoreUARTapb
mips.ovpworld.org/16450C	mips.ovpworld.org/MaltaFPGA	mips.ovpworld.org/SmartLoaderLinux
motorola.ovpworld.org/MC146818	national.ovpworld.org/16450	national.ovpworld.org/16550
national.ovpworld.org/16550_4bytes	nxp.ovpworld.org/iMX6_Analog	nxp.ovpworld.org/iMX6_CCM
nxp.ovpworld.org/iMX6_GPC	nxp.ovpworld.org/iMX6_GPIO	nxp.ovpworld.org/iMX6_GPT
nxp.ovpworld.org/iMX6_MMDC	nxp.ovpworld.org/iMX6_SDHC	nxp.ovpworld.org/iMX6_SRC
nxp.ovpworld.org/iMX6_UART	nxp.ovpworld.org/iMX6_WDOG	ovpworld.org/Alpha2x16Display
ovpworld.org/DynamicBridge	ovpworld.org/FlashDevice	ovpworld.org/ledRegister
ovpworld.org/SerInt	ovpworld.org/SimpleDma	ovpworld.org/switchRegister
ovpworld.org/temperatureSensor	ovpworld.org/trap	ovpworld.org/trap4K
ovpworld.org/vEthernet_Bridge	ovpworld.org/VirtioBlkMMIO	ovpworld.org/VirtioNetMMIO
philips.ovpworld.org/ISP1761	renesas.ovpworld.org/adc	renesas.ovpworld.org/bcu
renesas.ovpworld.org/brg	renesas.ovpworld.org/can	renesas.ovpworld.org/can
renesas.ovpworld.org/clkgen	renesas.ovpworld.org/crc	renesas.ovpworld.org/csib
renesas.ovpworld.org/csie	renesas.ovpworld.org/dma	renesas.ovpworld.org/intc
renesas.ovpworld.org/memc	renesas.ovpworld.org/rng	renesas.ovpworld.org/taa
renesas.ovpworld.org/tms	renesas.ovpworld.org/tmt	renesas.ovpworld.org/uartc
renesas.ovpworld.org/UPD70F3441Logic	riscv.ovpworld.org/CLINT	riscv.ovpworld.org/PLIC
riscv.ovpworld.org/SmartLoaderRV64Linux	safepower.ovpworld.org/node	safepower.ovpworld.org/NostrumNode
safepower.ovpworld.org/ring_oscillator	safepower.ovpworld.org/TTELNode	sifive.ovpworld.org/gpio
sifive.ovpworld.org/MSEL	sifive.ovpworld.org/PRCI	sifive.ovpworld.org/pwm
sifive.ovpworld.org/spi	sifive.ovpworld.org/teststatus	sifive.ovpworld.org/UART
smsc.ovpworld.org/LAN9118	smsc.ovpworld.org/LAN91C111	ti.ovpworld.org/tca6416a
ti.ovpworld.org/UartInterface	ti.ovpworld.org/ucd9012a	ti.ovpworld.org/ucd9248
vendor.com/fifo	xilinx.ovpworld.org/axi-gpio	xilinx.ovpworld.org/axi-intc
xilinx.ovpworld.org/axi-pcie	xilinx.ovpworld.org/axi-timer	xilinx.ovpworld.org/logicore-fit
xilinx.ovpworld.org/mdm	xilinx.ovpworld.org/mpmc	xilinx.ovpworld.org/xps-gpio
xilinx.ovpworld.org/xps-iic	xilinx.ovpworld.org/xps-intc	xilinx.ovpworld.org/xps-ll-temac
xilinx.ovpworld.org/xps-mch-emc	xilinx.ovpworld.org/xps-sysace	xilinx.ovpworld.org/xps-timer
xilinx.ovpworld.org/xps-uartlite	xilinx.ovpworld.org/zynq_7000-can	xilinx.ovpworld.org/zynq_7000-ddrc
xilinx.ovpworld.org/zynq_7000-devcfg	xilinx.ovpworld.org/zynq_7000-dmac	

6.0 General Information on Peripheral Models

This document provides usage information for an Imperas OVP peripheral behavioral model.

The document is split into sections providing specific information for this peripheral, including any ports for connecting into a platform, registers etc. and configuration options and general information for peripheral modeling with Imperas OVP.

6.1 Background

Imperas OVP simulation technology enables very high performance simulation, debug and analysis of platforms containing multiple processors and peripheral models. The technology is designed to be extensible: you can create new models of processors, peripherals and other platform components using interfaces and libraries defined by OVP.

The peripheral models created using the OVP APIs run on the Peripheral Simulation Engine (PSE).

The model is typically written in C and compiled into an executable for the PSE processor architecture. The model is compiled for speed of execution and to protect IP. It is dynamically loaded by the simulator at run time.

7.0 Building peripherals easily with Imperas iGen

To aid with model creation, Imperas products include iGen, a model generation tool. iGen takes the laborious and error-prone task of constructing the various hardware model and software element files required for a typical model, and automates this process. iGen creates the needed C files. iGen also creates the C++ SystemC TLM2 interface files needed to run peripheral models in SystemC simulations.

iGen takes as input a simple script specification that includes device internals such as registers and memories, port information, component descriptors, and other elements. iGen then builds the C code model files and user editable templates. These include model frameworks with registers, function calls, memory map, and other items. It ensures that all component parts of the model are well-structured using best practices, and are consistent throughout the files, thus eliminating a common source of errors.

More information on iGen can be found: imperas.com/products.

8.0 Peripheral model internals

Each instance of a peripheral model runs on its own virtual machine with an address space large enough for the model. This processor (the PSE) and its memory are separate from any processors, memories and buses in the platform being simulated; they exist only to execute the code of the peripheral model.

Interception of functions defined in the peripheral model allows the use of features of the host system in the

implementation of the behavior of a peripheral. As an example, a real platform might contain a video display device. When simulating this system, it is generally more convenient not to simulate the complete video display device but to use a video package available on the host machine, such as SDL, and to use this to render to the host display. Also models of uarts, ethernet devices and USB components can make use of the host PC resources during simulation, to allow, for example, a simulation to browse the real internet, or the simulation to connect to a real USB device.

9.0 Parts of peripheral models

9.1 Configuring the Peripheral Instance with Parameters

A peripheral can include the behaviour of several configurations. These are controlled when the peripheral is instanced in the platform by setting parameters defined on the peripheral.

9.2 Net Ports

Peripherals may be connected to other peripherals or processors with signal wires (nets). These can be used to act as interrupt signals or used to control behavior between peripherals.
The wires are created in the platform as nets and this net is connected into the peripheral using a net port.

9.3 Bus master ports

A bus master port initiates (and controls the address of) a bus cycle. Bus cycles are generated by behavioral code within the peripheral model.

9.4 Bus slave ports

A peripheral can be defined as having several bus slave ports. The bus slave ports can be split into several address blocks. Each address block be either local memory or memory mapped registers. Both of these can have associated callback functions. A memory mapped register can also be defined as specific read/write access, whether it is volatile, and also whether it is associated with a reset pin and mask. A memory mapped register can also have specific bit fields defined.

9.5 Packetnets

A peripheral can be defined as being connected to packetnet ports. A packetnet is used to model packet based communication such as Ethernet, CAN bus or GSM. A packetnet is created in a platform, then connected to packetnet ports on model instances. A packetnet can have many connections, each able to send or receive packets. A packetnet is used as an efficient method of communication within OVP models.

For more information on modeling with packetnets, please see the peripheral modeling documentation: OVP_Peripheral_Modeling_Guide.pdf, OVPsim_and_CpuManager_User_Guide.pdf and the example: $IMPERAS_HOME/Examples/Models/Peripherals/packetnet.

10.0 More information (documentation) on peripheral models and modeling

More information on modeling and APIs can be found at: OVPworld.org/technology_apis.

Specifics on modeling peripherals can be found: OVP_Peripheral_Modeling_Guide.pdf.

A full list of the currently available OVP documentation is available: OVPworld.org/documentation.
#