Imperas Peripheral Model Guide

Model Specific Information for arm.ovpworld.org / WdtSP805

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_WdtSP805.pdf
Created	05 May 2021
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

1.0 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

ARM SP805 Watchdog Registers.

1.2 Limitations

Does NOT model watchdog functionality, just provides registers to allow code to run.

1.3 Reference

ARM Watchdog Module (SP805) Technical Reference Manual (ARM DDI 0270)

1.4 Licensing

Open Source Apache 2.0

1.5 Location

The WdtSP805 peripheral model is located in an Imperas/OVP installation at the VLNV: arm.ovpworld.org / peripheral / WdtSP805 / 1.0.

2.0 Bus Slave Ports

This model has the following bus slave ports:

2.1 Bus Slave Port: bport1

Table 1. Bus Slave Port: bport1

Name	Size (bytes)	Must Be Connected	Description
bport1	0x1000	F (False)	

Table 2. Bus Slave Port: bport1 Registers:

Name	Offset	Width (bits)	Description	R/W	is Volatile
ab_WdogLoad	0x0	32			
ab_WdogValue	0x4	32			
ab_WdogControl	0x8	32			
ab_WdogIntClr	0xc	32			
ab_WdogRIS	0x10	32			
ab_WdogMIS	0x14	32			
ab_WdogLock	0xc00	32			
ab_WdogITCR	0xf00	32			
ab_WdogITOP	0xf04	32			

ab_PeriphID0	0xfe0	32			
ab_PeriphID1	0xfe4	32			
ab_PeriphID2	0xfe8	32			
ab_PeriphID3	0xfec	32			
ab_PCellID0	0xff0	32			
ab_PCellID1	0xff4	32			
ab_PCellID2	0xff8	32			
ab_PCellID3	0xffc	32			

3.0 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 3. Publicly available platforms using peripheral 'WdtSP805'

Platform Name	Vendor
ArmVersatileExpress	arm.ovpworld.org
ArmVersatileExpress-CA15	arm.ovpworld.org
ArmVersatileExpress-CA9	arm.ovpworld.org
ArmVersatileExpress	arm.ovpworld.org
ArmVersatileExpress-CA15	arm.ovpworld.org
ArmVersatileExpress-CA9	arm.ovpworld.org

This peripheral is used in some internal Imperas virtual platforms. Please contact Imperas for more information.

4.0 Peripheral components in the library

Table 4. Publicly available Imperas/OVP peripheral models (224 models)

Peripheral	Peripheral	Peripheral
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
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	

5.0 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.

5.1 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.

6.0 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.

7.0 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.

8.0 Parts of peripheral models

8.1 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.

8.2 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.

8.3 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.

8.4 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.

8.5 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.

9.0 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.
#