



## Imperas Peripheral Model Guide

### Model Specific Information for [freescale.ovpworld.org](http://freescale.ovpworld.org) / KinetisCAN

#### Imperas Software Limited

Imperas Buildings, North Weston  
Thame, Oxfordshire, OX9 2HA, U.K.  
[docs@imperas.com](mailto:docs@imperas.com).



Author	Imperas Software Limited
Version	20211118.0
Filename	OVP_Peripheral_Specific_Information_KinetisCAN.pdf
Created	31 December 2021
Status	OVP Standard Release

## Copyright Notice

Copyright 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).

## Table Of Contents

<b>1.0 Model Specific Information</b>	4
1.1 Description	4
1.2 Limitations	4
1.3 Reference	4
1.4 Licensing	4
1.5 Location	4
<b>2.0 Net Ports</b>	4
<b>3.0 Bus Slave Ports</b>	4
3.1 Bus Slave Port: bport1	4
<b>4.0 Platforms that use this peripheral component</b>	6
<b>5.0 Peripheral components in the library</b>	7
<b>6.0 General Information on Peripheral Models</b>	9
6.1 Background	9
<b>7.0 Building peripherals easily with Imperas iGen</b>	9
<b>8.0 Peripheral model internals</b>	9
<b>9.0 Parts of peripheral models</b>	10
9.1 Configuring the Peripheral Instance with Parameters	10
9.2 Net Ports	10
9.3 Bus master ports	10
9.4 Bus slave ports	10
9.5 Packetnets	10
<b>10.0 More information (documentation) on peripheral models and modeling</b>	10

## 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

Model of the CAN peripheral used on the Freescale Kinetis platform

### 1.2 Limitations

Provides the base behaviour for the OVP Freescale Kinetis platforms

### 1.3 Reference

[www.freescale.com/Kinetis](http://www.freescale.com/Kinetis)

### 1.4 Licensing

Open Source Apache 2.0

### 1.5 Location

The KinetisCAN peripheral model is located in an Imperas/OVP installation at the VLNv:  
[freescale.ovpworld.org / peripheral / KinetisCAN / 1.0](http://freescale.ovpworld.org/peripheral/KinetisCAN/1.0).

## 2.0 Net Ports

This model has the following net ports:

Table 1. Net Ports

Name	Type	Must Be Connected	Description
Reset	input	F (False)	

## 3.0 Bus Slave Ports

This model has the following bus slave ports:

### 3.1 Bus Slave Port: *bport1*

Table 2. Bus Slave Port: *bport1*

Name	Size (bytes)	Must Be Connected	Description
<i>bport1</i>	0x1000	F (False)	

Table 3. Bus Slave Port: *bport1* Registers:

Name	Offset	Width (bits)	Description	R/W	is Volatile
ab_MCR	0x0	32	Module Configuration Register, offset: 0x0		

ab_CTRL1	0x4	32	Control 1 Register, offset: 0x4		
ab_TIMER	0x8	32	Free Running Timer, offset: 0x8		
ab_RXMGMASK	0x10	32	Rx Mailboxes Global Mask Register, offset: 0x10		
ab_RX14MASK	0x14	32	Rx 14 Mask Register, offset: 0x14		
ab_RX15MASK	0x18	32	Rx 15 Mask Register, offset: 0x18		
ab_ECR	0x1c	32	Error Counter, offset: 0x1C		
ab_ESR1	0x20	32	Error and Status 1 Register, offset: 0x20		
ab_IMASK2	0x24	32	Interrupt Masks 2 Register, offset: 0x24		
ab_IMASK1	0x28	32	Interrupt Masks 1 Register, offset: 0x28		
ab_IFLAG2	0x2c	32	Interrupt Flags 2 Register, offset: 0x2C		
ab_IFLAG1	0x30	32	Interrupt Flags 1 Register, offset: 0x30		
ab_CTRL2	0x34	32	Control 2 Register, offset: 0x34		
ab_ESR2	0x38	32	Error and Status 2 Register, offset: 0x38		
ab_CRCR	0x44	32	CRC Register, offset: 0x44		
ab_RXFGMASK	0x48	32	Rx FIFO Global Mask Register, offset: 0x48		
ab_RXFIR	0x4c	32	Rx FIFO Information Register, offset: 0x4C		
ab_RXIMR0	0x880	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR1	0x884	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR2	0x888	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR3	0x88c	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR4	0x890	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR5	0x894	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR6	0x898	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR7	0x89c	32	Rx Individual Mask Registers, array offset:		

			0x880, array step: 0x4		
ab_RXIMR8	0x8a0	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR9	0x8a4	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR10	0x8a8	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR11	0x8ac	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR12	0x8b0	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR13	0x8b4	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR14	0x8b8	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		
ab_RXIMR15	0x8bc	32	Rx Individual Mask Registers, array offset: 0x880, array step: 0x4		

## 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 4. Publicly available platforms using peripheral 'KinetisCAN'

Platform Name	Vendor
FreescaleKinetis60	freescale.ovpworld.org
FreescaleKinetis64	freescale.ovpworld.org

## 5.0 Peripheral components in the library

Table 5. Publicly available Imperas/OVP peripheral models (227 models)

Peripheral	Peripheral	Peripheral
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/artyoIO	sifive.ovpworld.org/DDRCTL

sifive.ovpworld.org/gpio	sifive.ovpworld.org/MSEL	sifive.ovpworld.org/PLIC
sifive.ovpworld.org/PRCI	sifive.ovpworld.org/pwm	sifive.ovpworld.org/spi
sifive.ovpworld.org/teststatus	sifive.ovpworld.org/UART	smcsc.ovpworld.org/LAN9118
smcsc.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
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	



## 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](http://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](http://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](#).

#