

Adaptation of VCI V2 applications to VCI V3 using C++

White Paper



Version: 1.1
Editor: Andreas Stobe
Date: 2007-06-26
Doc. No.: WP102-0005

Contents

1	General Informations	1
1.1	The Comparison of the Simple Sequences	2
2	Adaptation Steps	2
2.1	Included DLLs	2
2.2	Functions for Board Selection	3
2.3	Functions for Board Initialization	3
2.4	Functions for CAN-Controller Handling	3
2.5	Functions for the CAN Channel Handling	4
2.6	Reception of CAN Messages	4
2.6.1	Polling Mode	4
2.6.2	Receive-Interrupt-Handler	5
2.7	Receive Buffer	5
2.8	Sending Messages	5
2.9	Exception-Handler	5
2.10	Handler for String Output	6
2.11	Functions for VCI Support Information	6
2.12	Data Types Used	6
2.12.1	VCI-CAN-Object	6
2.12.2	VCI-Board-Information	6
2.12.3	VCI-Board-Status	7
2.12.4	VCI-CAN-Information	7
2.12.5	VCI-CAN-Status	7
2.12.6	XAT_BoardCFG	7
2.12.7	Error code	8
2.13	Functions without Equivalent	8
2.13.1	Initialization of the VCI	8
3	Further Literature	8
4	Contact Information	8

1 General Informations

The "VCI V3"- and the "VCI V2"-programming interfaces are different because of the different driver structures. You can find more information in the "VCI V2"- and "VCI V3"- programming manuals.

When you adapt your “VCI V2” application to the VCI V3 programming interface, please note that by using the VCI V3 it is possible to access a CAN interface with several applications at the same time. It is also possible to work with standard (11-Bit IDs) and extended (29-Bit IDs) mode at the same time.

1.1 The Comparison of the Simple Sequences

In the table the sequences of the demo programming examples for VCI V2 and VCI V3 are compared.

VCI V2	VCI V3	Description
XAT_SelectHardware	vciEnumDeviceOpen vciEnumDeviceNext vciEnumDeviceClose	
VCI2_PrepareBoard	vciDeviceOpen	
VCI_InitCan	canControlOpen canControlInitialize	
VCI_SetAccMask	canControlSetAccFilter	The code- and mask-values for the filtering in the VCI V2 and VCI V3 have to be calculated differently. Please find more information in the VCI programming manual.
VCI_ConfigQueue (Tx) VCI_ConfigQueue (Rx) VCI_AssignRxQueObj	canChannelOpen canChannelInitialize canChannelActivate	
VCI_StartCan	canControlStart	
VCI_TransmitObj VCI_ReadQueObj	canChannelSendMessage canChannelReadMessage	
VCI_CancelBoard	canChannelClose canControlClose vciDeviceClose	

2 Adaptation Steps

2.1 Included DLLs

The vcinpl.dll should be included instead of the vci11un6.dll and the Xat11Reg.DLL.

VCI V2	VCI V3	Description
Xat11Reg.dll Vci11un6.dll	VCINPL.DLL	

2.2 Functions for Board Selection

Please use the VCI V3 functions for accessing the device list instead of the “XAT”-registration functions. For the board selection in the VCI V3 it is possible to use the Select-Device-Dialog and it is possible use the Device Enumerator. See the § 4.2.1 Functions for accessing the device list in the VCI V3 programming manual.

VCI V2	VCI V3	Description
XAT_SelectHardware	vciSelectDeviceDlg	
XAT_GetConfig	vciEnumDeviceOpen vciEnumDeviceNext vciEnumDeviceClose	
XAT_EnumHWEntry	vciEnumDeviceOpen vciEnumDeviceNext	
XAT_FindHwEntry	vciFindDeviceByHwid vciFindDeviceByClass	
XAT_SetDefaultHwEntry	-	
XAT_GetDefaultHwEntry	-	

2.3 Functions for Board Initialization

See the § 4.2.2 Functions for accessing the bus adapter in the VCI V3 programming manual.

VCI V2	VCI V3	Description
VCI_SearchBoard	-	
VCI_SetDownloadState	-	
VCI2_PrepareBoard VCI_PrepareBoard VCI2_PrepareBoardMsg VCI_PrepareBoardMsg VCI_PrepareBoardVisBas	vciDeviceOpen	
VCI_CancelBoard	vciDeviceClose	
VCI_TestBoard	-	
VCI_ReadBoardInfo	vciDeviceGetInfo	
VCI_ReadBoardStatus	-	
VCI_ResetBoard	-	

2.4 Functions for CAN-Controller Handling

See the § 4.3.1 Functions for the CAN controller handling in the VCI V3 programming manual.

VCI V2	VCI V3	Description
--------	--------	-------------

VCI_ReadCanInfo	canControlDetectRate	
VCI_ReadCanStatus	canControlGetStatus	
VCI_InitCan	canControlInitialise	
VCI_SetAccMask	canControlSetAccFilter	The code- and mask-values for the filtering in the VCI V2 and VCI V3 have to be calculated differently. Please find more information in the VCI programming manual.
VCI_ResetCan	canControlReset	
VCI_StartCan	canControlStart	

2.5 Functions for the CAN Channel Handling

The VCI V3 does not have the Software Queues and the Software Buffers like the VCI V2. With the initialization of a VCI V3 can channel a RxFifo and a TxFifo will be created. There is no CAN channel filtering available in the VCI V3. See the § 4.3.2 Functions for the CAN channel handling in the VCI V3 programming manual.

VCI V2	VCI V3	Description
VCI_ConfigQueue	canChannelInitialize	
VCI_AssignRxQueObj	-	
VCI_ResetTimeStamp	-	
VCI_ConfigBuffer	-	
VCI_ReConfigBuffer	-	

2.6 Reception of CAN Messages

The VCI V3 does not have the Receive-Interrupt-Handlers. See the § 3.3.1 Reception of CAN messages in the VCI V3 programming manual.

2.6.1 Polling Mode

In the polling mode the VCI V2 function "VCI_ReadQueObj" should be replaced by the VCI V3 function "canChannelPeekMessage" or "canChannelReadMessage".

VCI V2	VCI V3
VCI_ReadQueObj	canChannelPeekMessage canChannelReadMessage
VCI_ReadQueStatus	canChannelGetStatus

2.6.2 Receive-Interrupt-Handler

The receive callback of the VCI V2 application should be replaced with a Receive Thread in the VCI V3 application. In the Receive Thread the function “canChannelPeekMessage” or the “canChannelWaitRxEvent” and “canChannelReadMessage” can be used.

VCI V2	VCI V3
RxCallback Handler	canChannelReadMessage or canChannelWaitRxEvent canChannelPeekMessage

2.7 Receive Buffer

The VCI V3 does not have any receive buffers.

VCI V2	VCI V3	Description
VCI_ReadBufStatus	-	
VCI_ReadBufData	-	

2.8 Sending Messages

See the §.3.3.2. Transmission of CAN messages in the VCI V3 programming manual

VCI V2	VCI V3	Description
VCI_TransmitObj	canChannelSendMessage or canChannelPostMessage	
VCI_RequestObj	canChannelSendMessage or canChannelPostMessage	
VCI_UpdateBufObj	-	

2.9 Exception-Handler

The VCI V3 does not have an Exception Handler. In the VCI V3 you have to watch the VCI return code of the called function. The VCI return code can be converted to a text which can be read by users. See the § 4.1.2 “vciFormatError” in the VCI V3 programming manual.

VCI V2	VCI V3	Description
Exception-Handlers	-	

2.10 Handler for String Output

The VCI V3 does not have a Handler for String Output. In the VCI V3 you have to watch the VCI return code of the called function. The VCI return code can be converted to a text which can be read by users. See the § 4.1.2 vciFormatError in the VCI V3 programming manual.

VCI V2	VCI V3	Description
Handler for String Output	-	

2.11 Functions for VCI Support Information

This functions were in the VCI V2 because of the compatibility with the VCI V1 only.

VCI V2	VCI V3	Description
VCI_Get_LibType	vciGetVersion vciDeviceGetInfo	
VCI_GetBrdNameByType	vciFindDeviceByClass vciFindDeviceByHwid	
VCI_GetBrdTypeByName	vciFindDeviceByClass vciFindDeviceByHwid	

2.12 Data Types Used

The segmentation of the VCI V2 and VCI V3 structures is different

2.12.1 VCI-CAN-Object

VCI V2	VCI V3	Description
VCI_CAN_OBJ ~.time_stamp ~.id ~.a_data[8]	CANMSG ~.dwTime ~.dwMsgId ~.abData[8]	
VCI_CAN_OBJ ~.len ~.rtr ~.sts	CANMSGINFO ~.Bits.dlc ~.Bits.rtr ~.Bits.ovr	

2.12.2 VCI-Board-Information

VCI V2	VCI V3	Description
VCI_BOARD_INFO ~.hw_version ~.serial_num ~.str_hw_type	VCIDEVICEINFO ~.HardwareMajorVersion ~.HardwareMinorVersion ~.UniqueHardwareId ~.DeviceClass	
VCI_BOARD_INFO	VCIDEVICECAPS	

~.can_num	~.BusCtrlCoung	
VCI_BOARD_INFO	CANCAPABILITIES	
~.time_stamp_res	~.dwTscDivisor	
VCI_BOARD_INFO	-	
~.fw_version ~.dd_version ~.sw_version ~.mem_pool_size ~.irq_num ~.board_seg		

2.12.3 VCI-Board-Status

VCI V2	VCI V3	Description
VCI_BOARD_STS	CANCHANSTATUS	
~.sts	~.fRxOverrun	
VCI_BOARD_STS	-	
~.cpu_load		

2.12.4 VCI-CAN-Information

VCI V2	VCI V3	Description
VCI_CAN_INFO	VCIDEVICECAPS	
~.can_type	~.BusCtrlTypes	
VCI_CAN_INFO	CANLINESTATUS	
~.bt0 ~.bt1	~.bBtReg0 ~.bBtReg0	
VCI_CAN_INFO	-	
~.acc_code ~.acc_mask		

2.12.5 VCI-CAN-Status

VCI V2	VCI V3	Description
VCI_CAN_STS	CANLINESTATUS	
~.sts ~.bus_load	~.dwStatus ~.bBusLoad	

2.12.6 XAT_BoardCFG

VCI V2	VCI V3	Description
XAT_BoardCFG	VCIDEVICEINFO	
~.board_no	~.VciObjectId	

~.board_type ~.sz_manufacturer ~.sz_brd_info ~.sz_CardAddString	~.DeviceClass ~.Manufacturer ~.Description ~.Description	
XAT_BoardCFG ~.sz_brd_name	VCIID ~.AsLuid	

2.12.7 Error code

The error code is defined in the vcierr.h file and can be converted with the VCI V3 function VciFormatError.

VCI V2	VCI V3	Description
HRESULT Error codes	See vcierr.h	

2.13 Functions without Equivalent

2.13.1 Initialization of the VCI

VCI V2	VCI V3	Description
VCI_Init	-	

3 Further Literature

- [1] VCI V2.16 Programming manual
- [2] VCI V3 Programming manual

4 Contact Information

IXXAT Automation GmbH	IXXAT Inc.	Others
Leibnizstr. 15 88250 Weingarten Germany Tel.: +49-(0)7 51 / 5 61 46-0 Fax: +49-(0)7 51 / 5 61 46-29 E-Mail: info@ixxat.de Internet: www.ixxat.de	120 Bedford Center Road Bedford, NH 03110 USA Phone: +1-603-471-0800 Fax: +1-603-471-0880 E-mail: sales@ixxat.com Internet: www.ixxat.com	For a list of international distributors please consult the IXXAT web page under: www.ixxat.de contact distributors