

# VCI - Virtual CAN Interface

Einbindung in LabWindows/CVI

---

## **IXXAT**

### **Hauptsitz**

IXXAT Automation GmbH  
Leibnizstr. 15  
D-88250 Weingarten

Tel.: +49 (0)7 51 / 5 61 46-0  
Fax: +49 (0)7 51 / 5 61 46-29  
Internet: [www.ixxat.de](http://www.ixxat.de)  
e-Mail: [info@ixxat.de](mailto:info@ixxat.de)

### **Geschäftsbereich USA**

IXXAT Inc.  
120 Bedford Center Road  
USA-Bedford, NH 03110

Phone: +1-603-471-0800  
Fax: +1-603-471-0880  
Internet: [www.ixxat.com](http://www.ixxat.com)  
e-Mail: [sales@ixxat.com](mailto:sales@ixxat.com)

## **Support**

Sollten Sie zu diesem, oder einem unserer anderen Produkte Support benötigen, wenden Sie sich bitte schriftlich an:

Fax: +49 (0)7 51 / 5 61 46-29  
e-Mail: [support@ixxat.de](mailto:support@ixxat.de)

## **Copyright**

Die Vervielfältigung (Kopie, Druck, Mikrofilm oder in anderer Form) sowie die elektronische Verbreitung dieses Dokuments ist nur mit ausdrücklicher, schriftlicher Genehmigung von IXXAT Automation erlaubt. IXXAT Automation behält sich das Recht zur Änderung technischer Daten ohne vorherige Ankündigung vor. Es gelten die allgemeinen Geschäftsbedingungen sowie die Bestimmungen des Lizenzvertrags. Alle Rechte vorbehalten.

<b>1</b>	<b>Übersicht.....</b>	<b>5</b>
<b>2</b>	<b>Systemvoraussetzungen .....</b>	<b>6</b>
<b>3</b>	<b>Die VCI unter LabWindows/CVI.....</b>	<b>7</b>
	<b>3.1 Einrichten eines ActiveX Controller für die VCI .....</b>	<b>7</b>
	3.1.1 ActiveX Controller Wizard starten.....	7
	3.1.2 Auswahl des VCI-Wrappers als ActiveX Server.....	8
	3.1.3 Abspeichern des ActiveX Controllers.....	8
	3.1.4 Erweiterte Optionen.....	9
	3.1.5 Generierung ActiveX Controller abgeschlossen .....	9
	<b>3.2 Benutzen des VCI-Wapper ActiveX Controllers.....</b>	<b>11</b>
	3.2.1 Definition einer Benutzerschnittstelle.....	11
	3.2.1.1 Neue Benutzerschnittstelle anlegen.....	11
	3.2.1.2 Steuerelement anlegen .....	12
	3.2.1.3 Steuerelement konfigurieren.....	13
	3.2.1.4 Code Generierung .....	14
	3.2.2 Einfügen von VCI-Wrapper Methodenaufrufen .....	15
	3.2.2.1 Auswahl der Typbibliothek .....	15
	3.2.2.2 Auswahl der Schnittstelle.....	15
	3.2.2.3 Auswahl Schnittstellen-Methode.....	16
	3.2.2.4 Parametrierung des Methodenaufrufs.....	17
	3.2.2.5 Nachträgliche Modifikation des Methodenaufrufs .....	18
	3.2.3 Registrieren der Rx-Event Callback-Funktion.....	19
	3.2.3.1 Auswahl der Typbibliothek .....	19
	3.2.3.2 Auswahl der Event Callback Registrierung .....	19
	3.2.3.3 Auswahl der Event Schnittstelle .....	20
	3.2.3.4 Auswahl der Registrierungsfunktion für das Event .....	20
	3.2.3.5 Parametrierung des Registrierungsaufrufs .....	21
	<b>3.3 Beispielapplikation VCIDemo.....</b>	<b>22</b>



# 1 Übersicht

Das Virtual CAN Interface (VCI) ist eine gemeinsame Programmierschnittstelle (API) für alle IXXAT CAN-Interfaces.

Die VCI-API ist eine Funktionsschnittstelle, d.h. dem Benutzer werden von der VCI-DLL exportierte Funktionen zur Verfügung gestellt.

Es ist theoretisch möglich unter LabWindows/CVI direkt auf diese VCI-API zu arbeiten. Dies gestaltet sich aber etwas umständlich und kompliziert. Deshalb empfehlen wir, den VCI-Wrapper zu installieren, der die VCI-API über eine COM-Schnittstelle als ActiveX Komponente bereitstellt. Als ActiveX kann die VCI sehr schnell und einfach in ein LabWindows/CVI Projekt eingebunden werden.

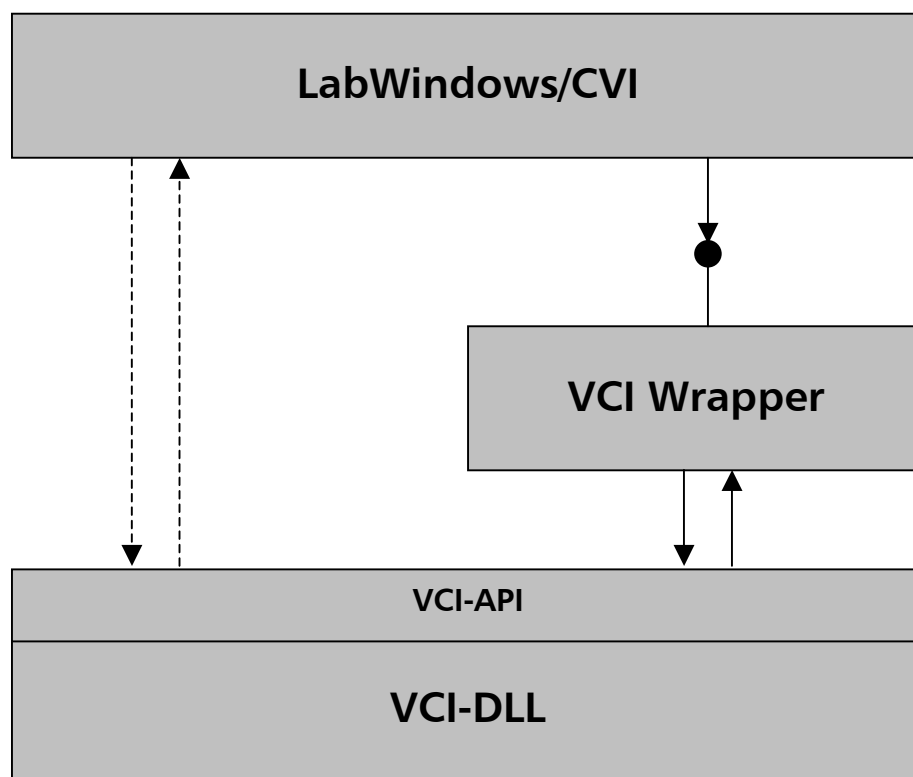


Bild 1-1: Übersicht

# 2 Systemvoraussetzungen

- VCI V2.10 (oder höher)
- VCI Wrapper V2.0

oder

- VCI V2.14 (oder höher - VCI Wrapper integriert)

## 3 Die VCI unter LabWindows/CVI

### 3.1 Einrichten eines ActiveX Controllers für die VCI

Bevor der VCI-Wrapper in einem LabWindows Projekt verwendet werden kann muß ein sogenannter ActiveX Controller als Platzhalter eingerichtet werden.

#### 3.1.1 ActiveX Controller Wizard starten

Starten Sie den ActiveX Controller Wizard über Tools | Create ActiveX Controller...

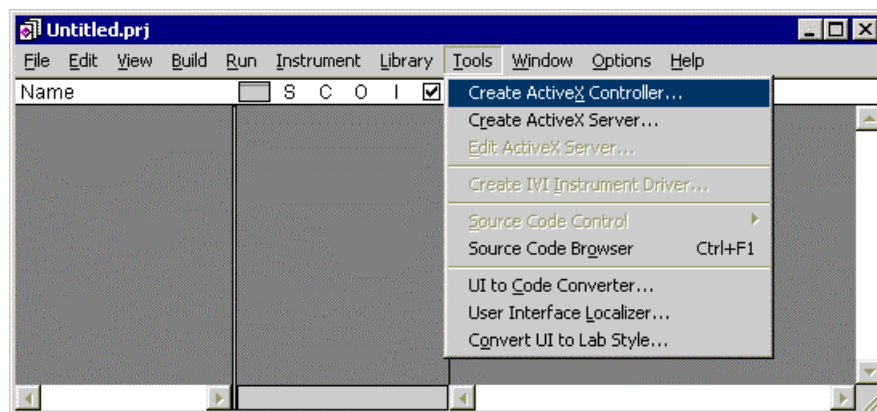


Bild 3.1-1: ActiveX Controller Wizard starten

Im folgenden Bild sehen Sie den gestarteten ActiveX Controller Wizard. Fahren Sie mit Klick auf den „Next“-Schalter fort.

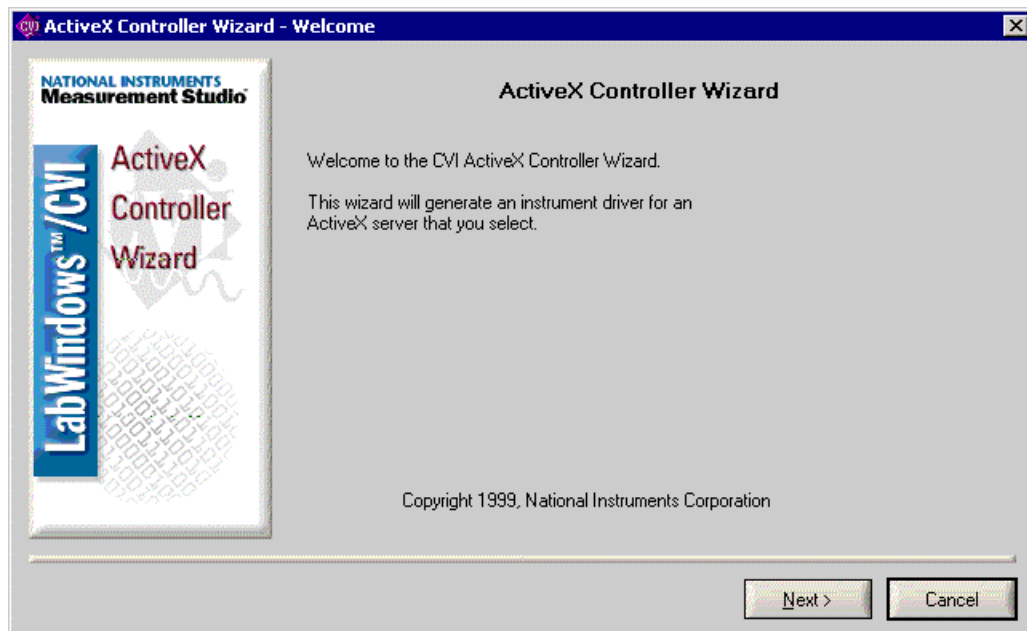


Bild 3.1-2: Startfenster ActiveX Controller Wizard

### 3.1.2 Auswahl des VCI-Wrappers als ActiveX Server

Wählen Sie den VCI-Wrapper als ActiveX Server für den ActiveX Controller aus und betätigen Sie den „Next“-Schalter.

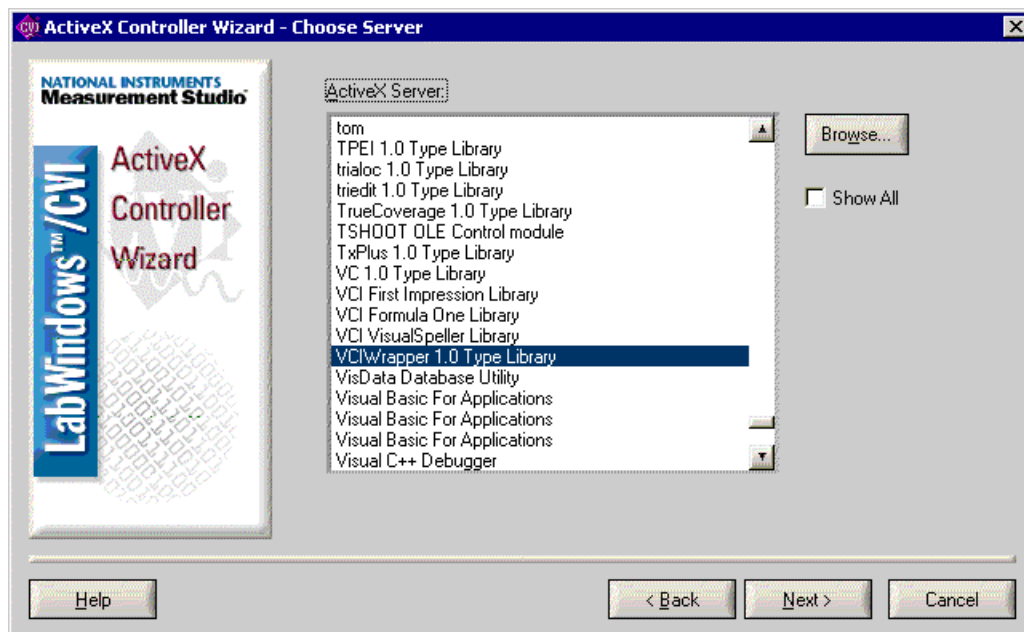


Bild 3.1-3: Auswahl des VCI-Wrappers als ActiveX Server

### 3.1.3 Abspeichern des ActiveX Controllers

Geben Sie unter „Target .fp File:“ den absoluten Pfad der Datei ein, unter der Sie den ActiveX Controller abspeichern wollen. Klicken Sie auf den „Next“-Schalter.

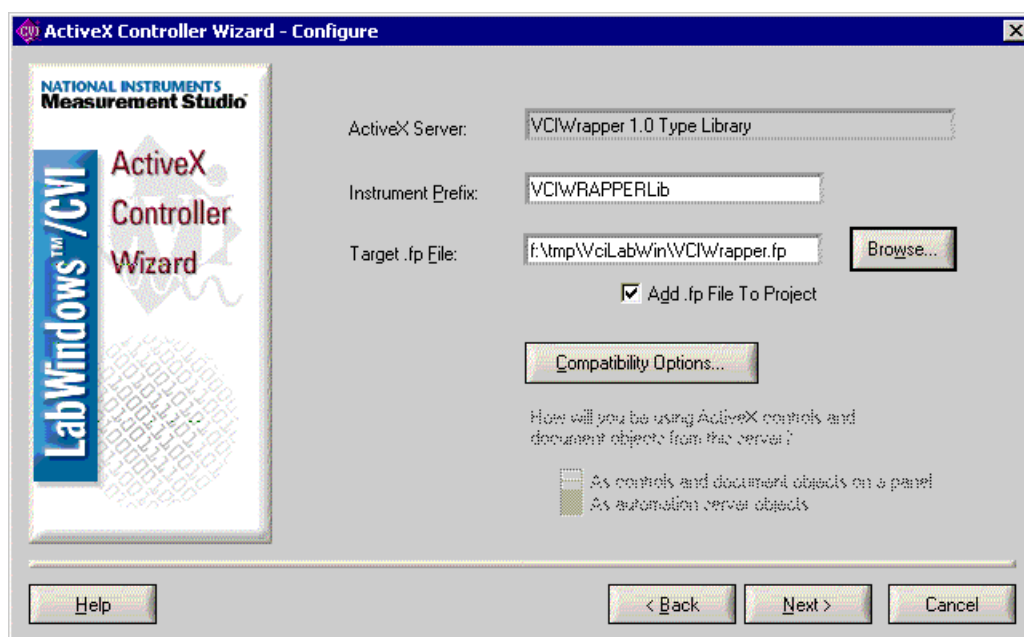


Bild 3.1-4: Abspeichern des ActiveX Controllers

## 3.1.4 Erweiterte Optionen

Den folgenden Dialog können Sie mit dem „Next“-Schalter überspringen, da die VCI-Wrapper Typbibliothek nur ein Objekt enthält.

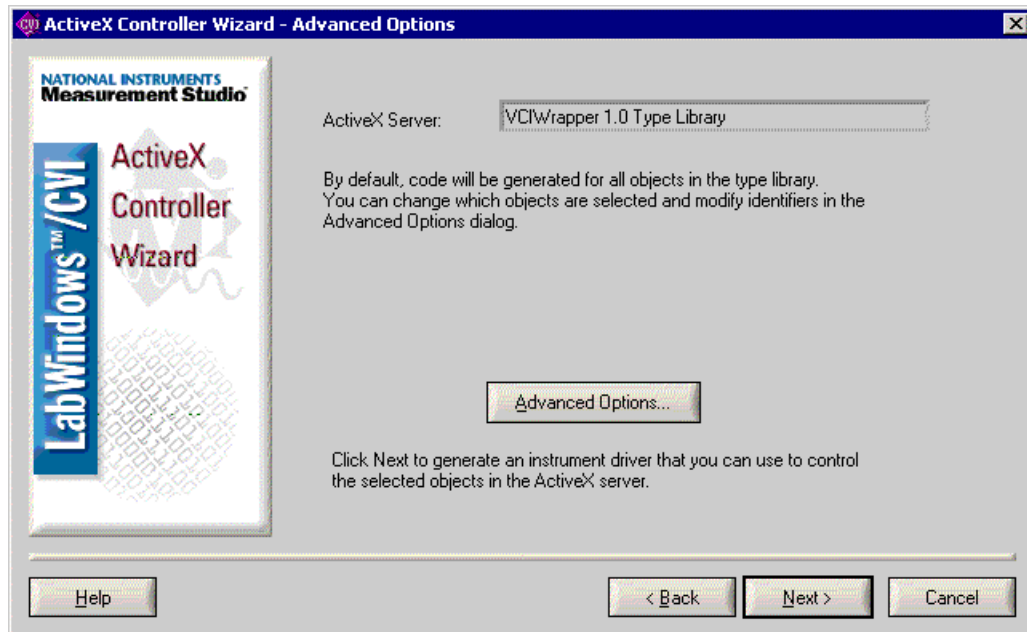


Bild 3.1-5: Erweiterte Optionen

## 3.1.5 Generierung ActiveX Controller abgeschlossen

Die Generierung des ActiveX Controllers ist nun abgeschlossen. Klicken Sie auf den „Close“-Schalter.

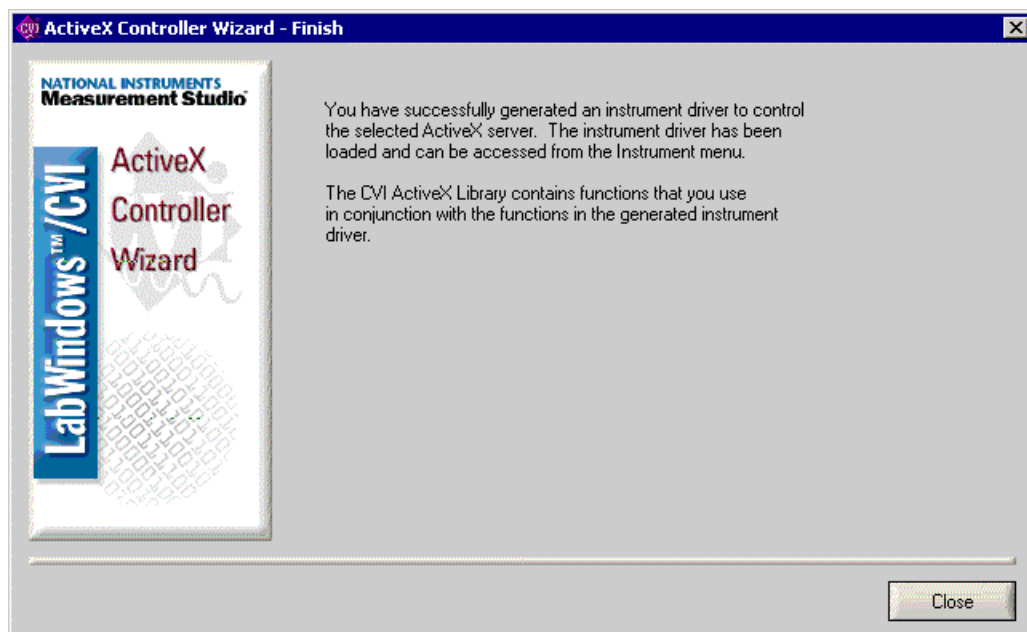
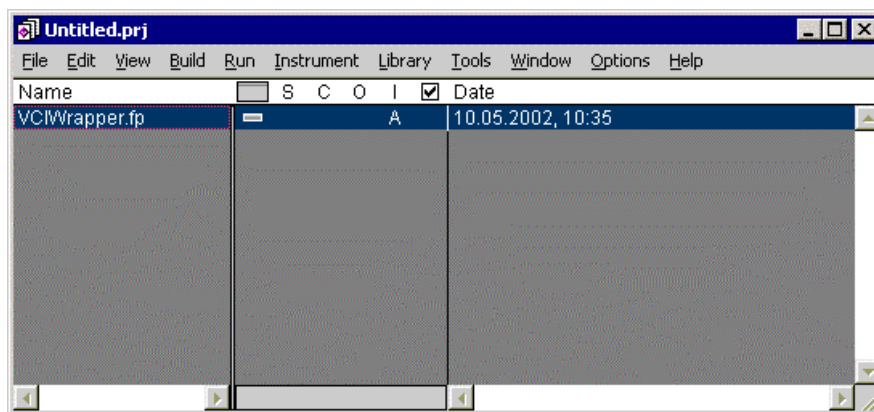


Bild 3.1-6: Generierung ActiveX Controller abgeschlossen

## Die VCI unter LabWindows/CVI

---

Der ActiveX Controller erscheint nun auch im Hauptfenster.



---

**Bild 3.1-7: ActiveX Controller im Hauptfenster**

## 3.2 Benutzen des VCI-Wapper ActiveX Controllers

### 3.2.1 Definition einer Benutzerschnittstelle

#### 3.2.1.1 Neue Benutzerschnittstelle anlegen

Legen Sie über File|New|User Interface eine neue Benutzerschnittstelle an.

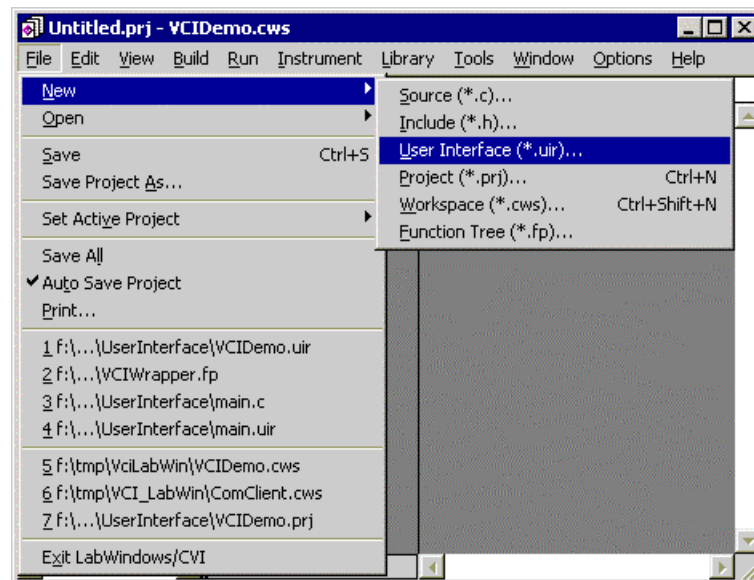


Bild 3.2-1: Neue Benutzerschnittstelle anlegen

Es erscheint daraufhin ein neues Fenster mit einer noch leeren Benutzerschnittstelle.

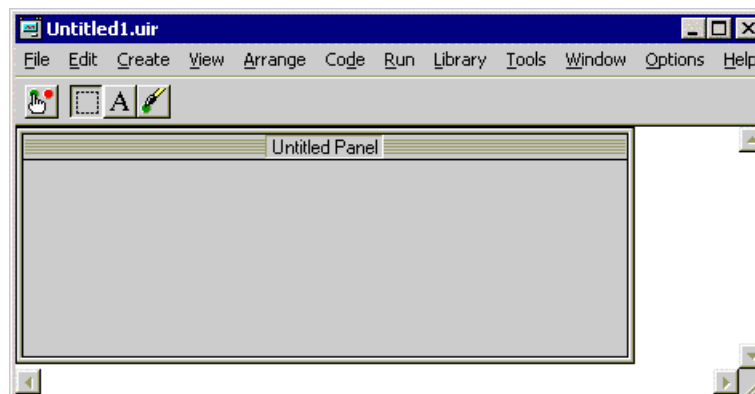
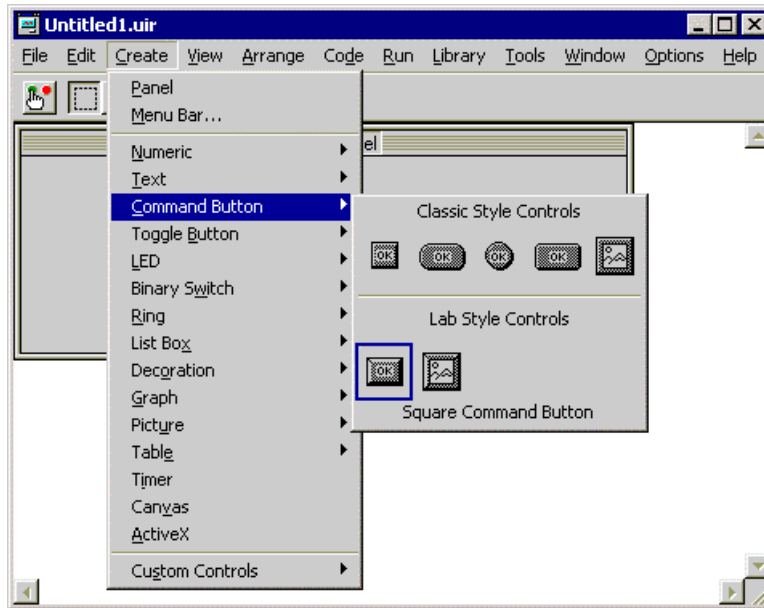


Bild 3.2-2: Die neue Benutzerschnittstelle

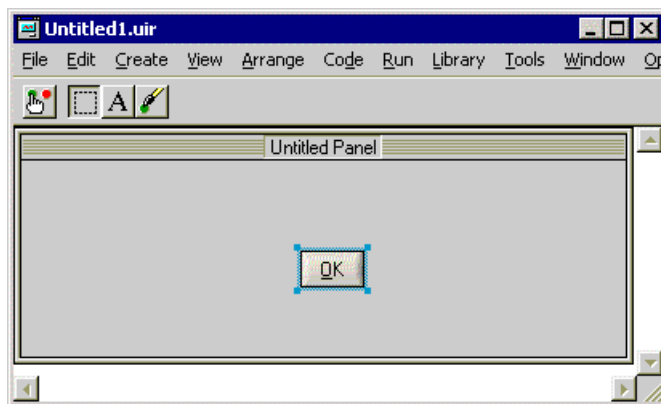
### 3.2.1.2 Steuerelement anlegen

Legen Sie über Create|Command Button z.B. einen Schalter an.



**Bild 3.2-3: Steuerelement anlegen**

LabWindows/CVI legt Ihnen einen neuen Schalter auf die Benutzerschnittstelle.

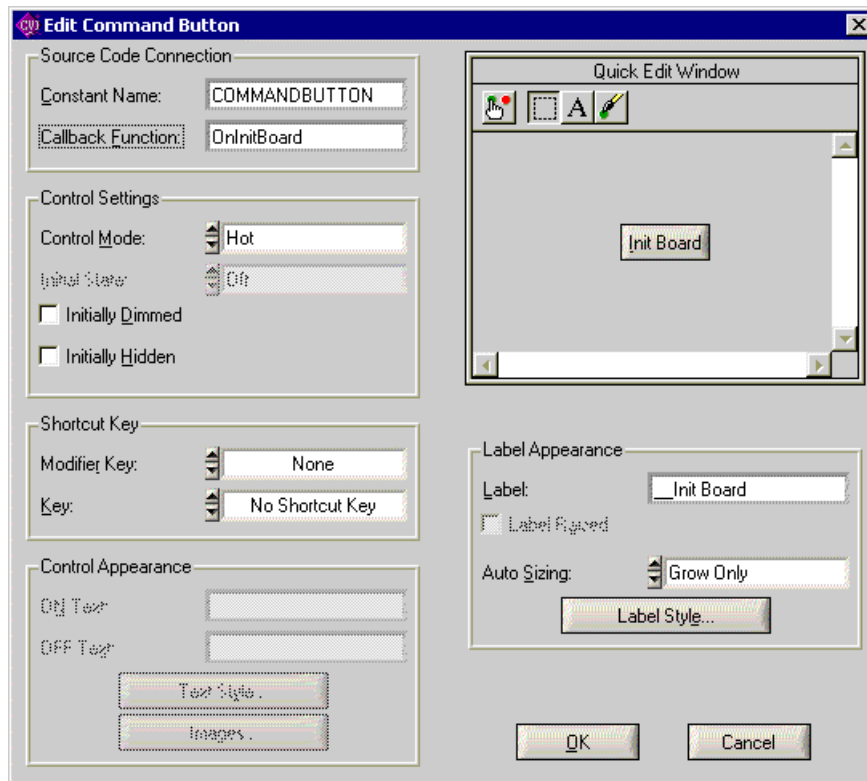


**Bild 3.2-4: Das neue Steuerelement**

### 3.2.1.3 Steuerelement konfigurieren

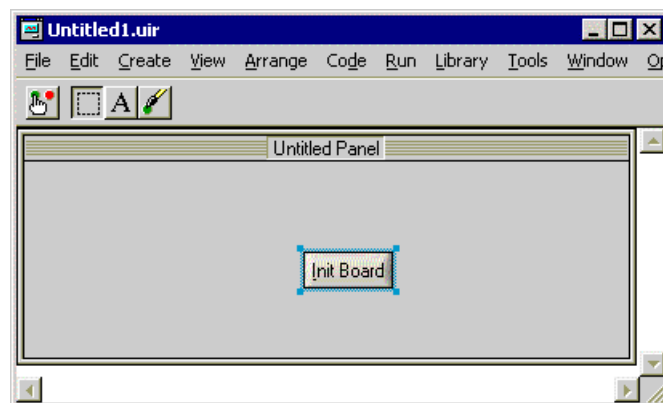
Mittels Doppelklick auf den Schalter läßt sich dieser konfigurieren. Geben Sie als Label z.B. „Init Board“ ein. Unter „Callback Function“ kann der Name einer Sourcecode-Funktion (hier „OnInitBoard“) angegeben werden, die zur Laufzeit aufgerufen wird, wenn der Benutzer den Schalter betätigt.

Die Callback Funktion wird später von LabWindows/CVI bei der Code Generierung automatisch angelegt.



**Bild 3.2-5: Steuerelement konfigurieren**

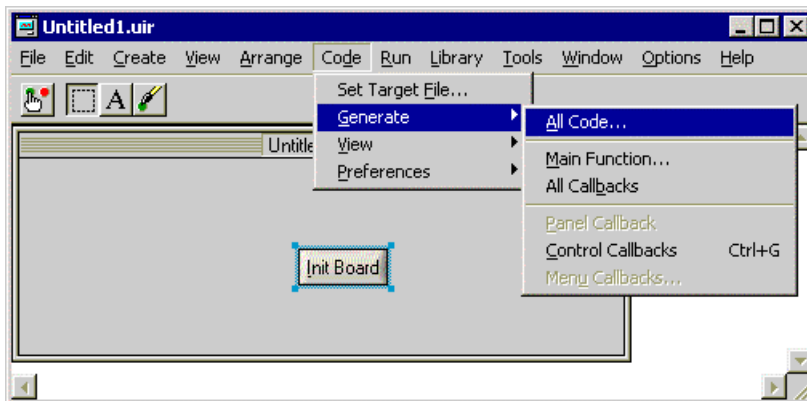
Nach bestätigen mit dem „OK“-Schalter erscheint der konfigurierte Schalter.



**Bild 3.2-6: Das konfigurierte Steuerelement**

### 3.2.1.4 Code Generierung

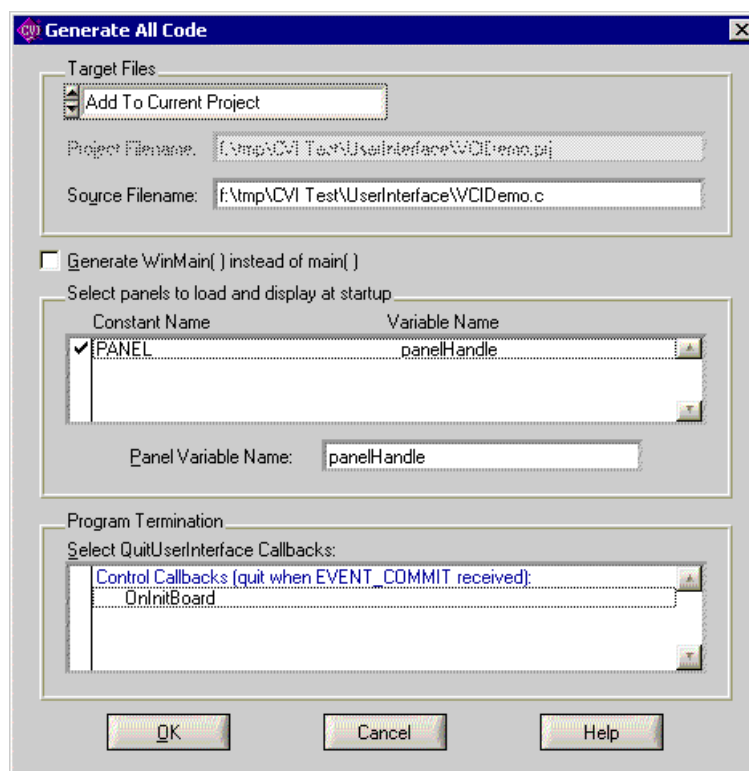
Stoßen Sie nun über Code|Generate|All Code die Generierung des Sourcecodes an.



**Bild 3.2-7: Code Generierung**

Im folgenden Dialog muß noch der Name des Sourcecode Files angegeben werden. Hierbei ist es wichtig, daß unter „Target Files“ auf „Add To Current Project“ umgestellt wird. LabWindows/CVI legt sonst ein neues Projekt an und schließt das aktuelle.

Bestätigen Sie den Dialog mit dem „OK“-Schalter.



**Bild 3.2-8: Name des Sourcefiles**

## 3.2.2 Einfügen von VCI-Wrapper Methodenaufrufen

### 3.2.2.1 Auswahl der Typbibliothek

Nach erfolgreicher Generierung des Sourcecodes erscheint nachfolgendes Sourcecode-Fenster. Hier setzen Sie den Cursor in diejenige Zeile, in der Sie den Aufruf einer VCI-Wrapper Methode einfügen wollen. Wählen Sie die VCI-Wrapper Bibliothek im Menü Instrument an.

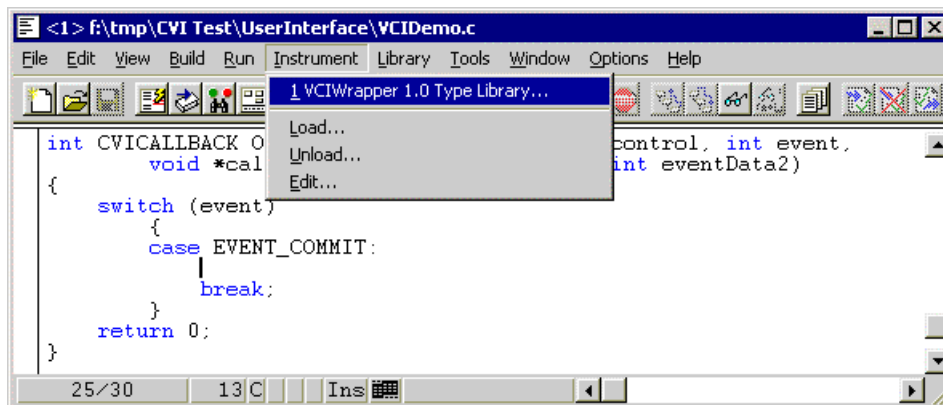


Bild 3.2-9: Auswahl der Typbibliothek

### 3.2.2.2 Auswahl der Schnittstelle

Im folgenden Dialog muß die Schnittstelle, die die gewünschte Methode enthält, ausgewählt werden. Wählen Sie entweder mit Doppelklick auf die Schnittstelle IVCIWrapper oder markieren Sie die Schnittstelle und gehen Sie über den „Select“-Schalter weiter.

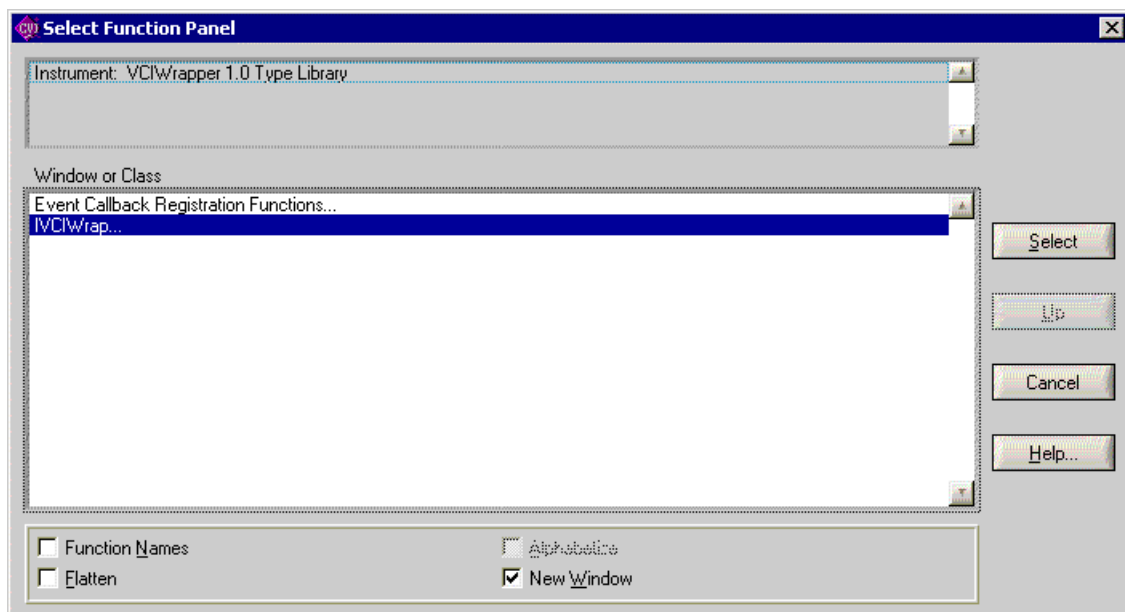


Bild 3.2-10: Auswahl der Schnittstelle

### 3.2.2.3 Auswahl Schnittstellen-Methode

Wählen Sie nun die Schnittstellenmethode, die Sie aufrufen möchten. Wählen Sie entweder mit Doppelklick auf die Methode oder markieren Sie die Methode und gehen Sie über den „Select“-Schalter weiter.

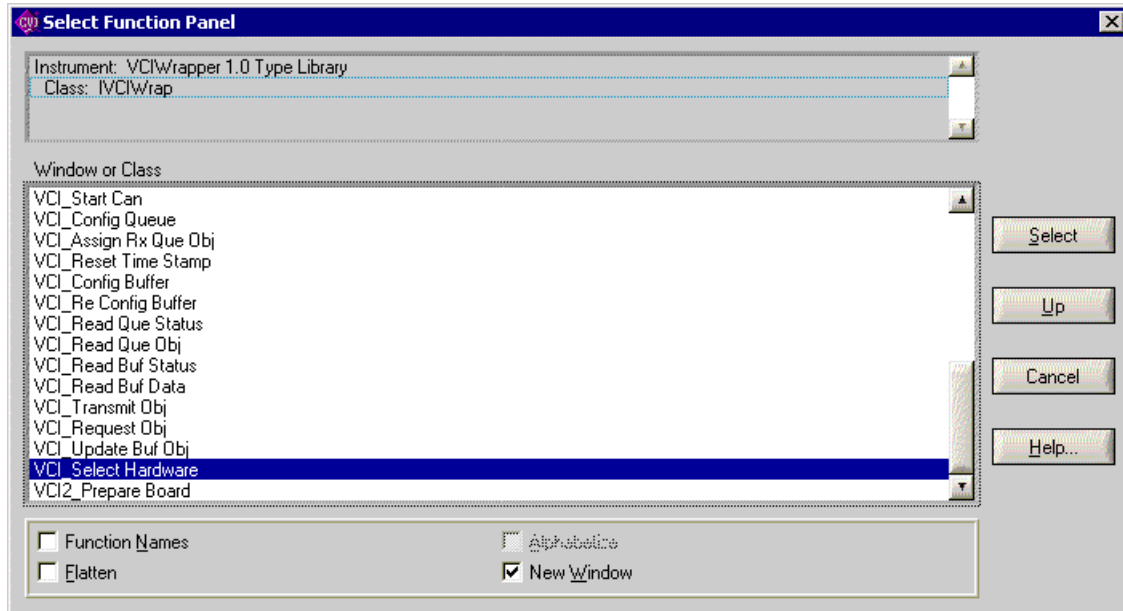
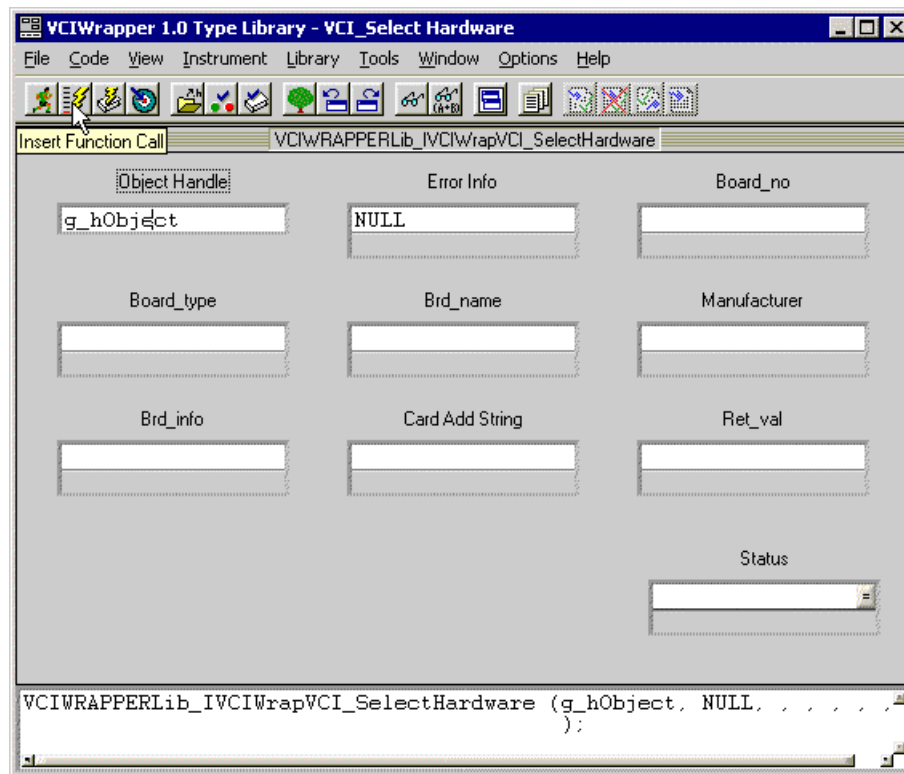


Bild 3.2-11: Auswahl Schnittstellen-Methode

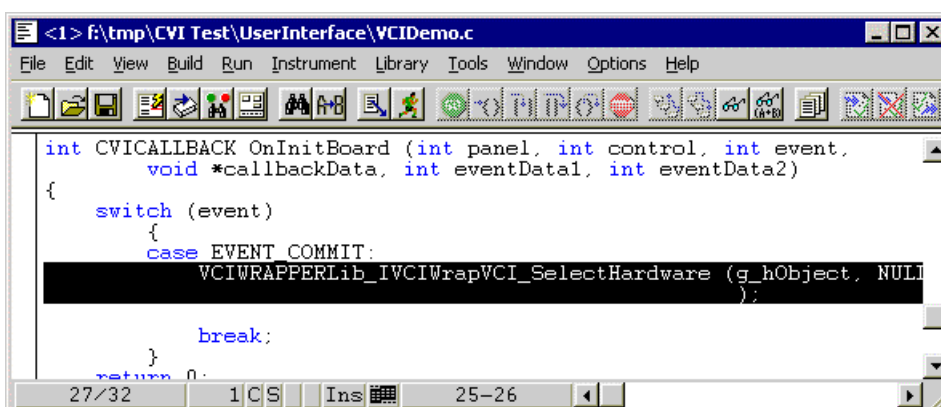
### 3.2.2.4 Parametrierung des Methodenaufrufs

Im folgenden Dialog ist es möglich den einzufügenden Methodenaufruf zu parametrieren. Sie können für jeden Parameter einen Variablennamen, ganze Argumente oder auch Konstanten eintragen. Wenn Sie diese noch nicht wissen sollten, können Sie diese Felder auch unverändert, bzw. leer lassen.



**Bild 3.2-12: Parametrierung des Methodenaufrufs**

Klicken Sie nun auf den Schalter „Insert Function Call“ in der Werkzeugleiste. Daraufhin wird der Methodenaufruf an der vorab markierten Zeile eingefügt.



**Bild 3.2-13: Der neue Methodenaufruf**

### 3.2.2.5 Nachträgliche Modifikation des Methodenaufrufs

Der Methodenaufruf kann nachträglich modifiziert werden. Es gibt die Möglichkeit, den Aufruf manuell im Sourcecode zu ändern oder über einen Klick mit der rechten Maustaste auf den Methodenaufruf im Sourcecode das „Function Panel“ zu öffnen. Dort können die Parameter einzeln in der Übersicht modifiziert werden.

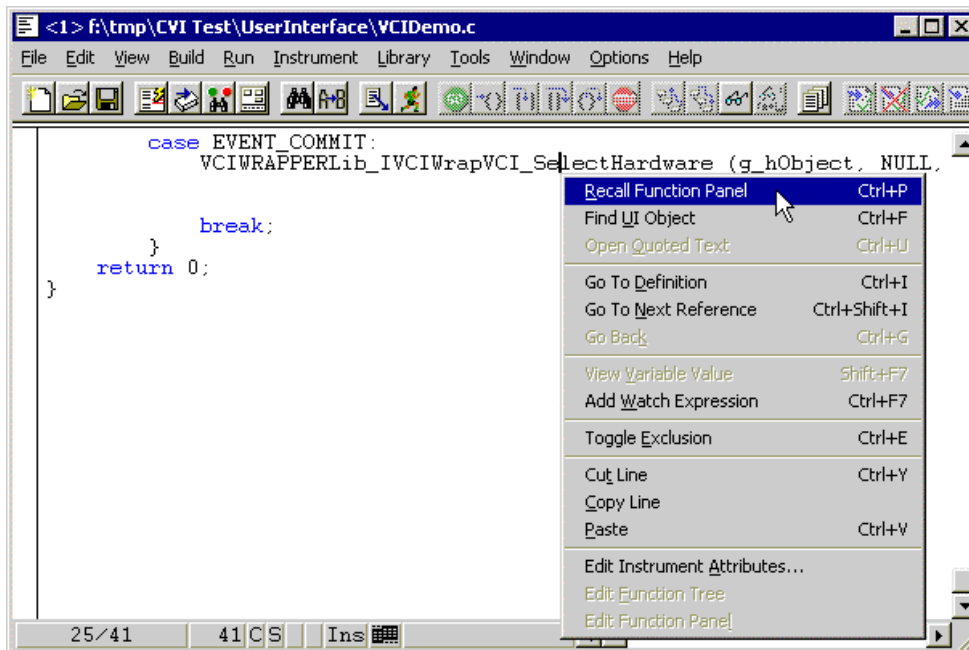


Bild 3.2-14: Modifikation des Methodenaufrufs

### 3.2.3 Registrieren der Rx-Event Callback-Funktion

Die Registrierung von Event Callback-Funktionen erfolgt über das „Function Panel“. Die Event Callback-Funktion selbst muß manuell implementiert werden.

#### 3.2.3.1 Auswahl der Typbibliothek

Setzen Sie den Cursor im Sourcecode-Fenster in diejenige Zeile, in der Sie eine Event Callback Registrierung einfügen wollen. Wählen Sie die VCI-Wrapper Bibliothek im Menü Instrument an.

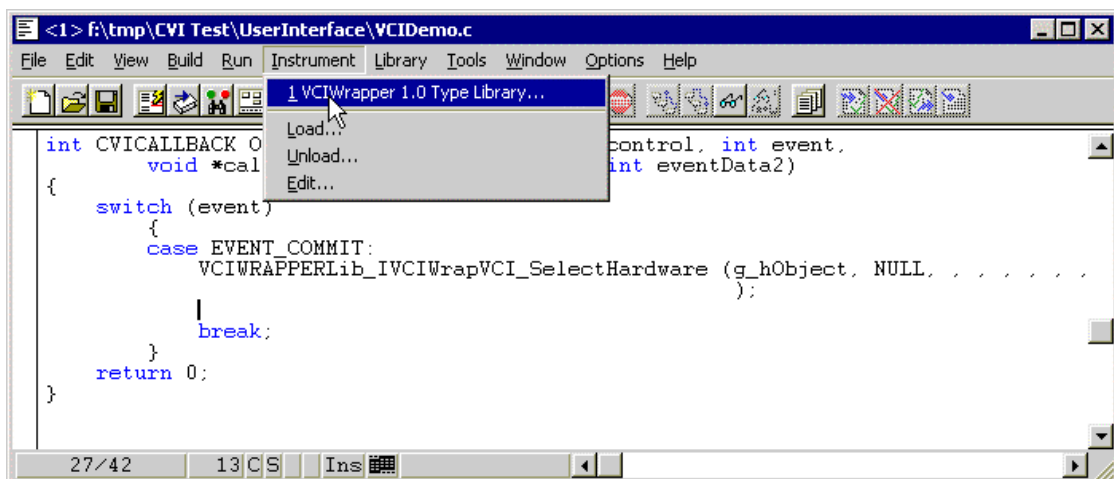


Bild 3.2-14: Auswahl der Typbibliothek

#### 3.2.3.2 Auswahl der Event Callback Registrierung

Wählen Sie im folgenden Dialog „Event Callback Registration Functions...“.

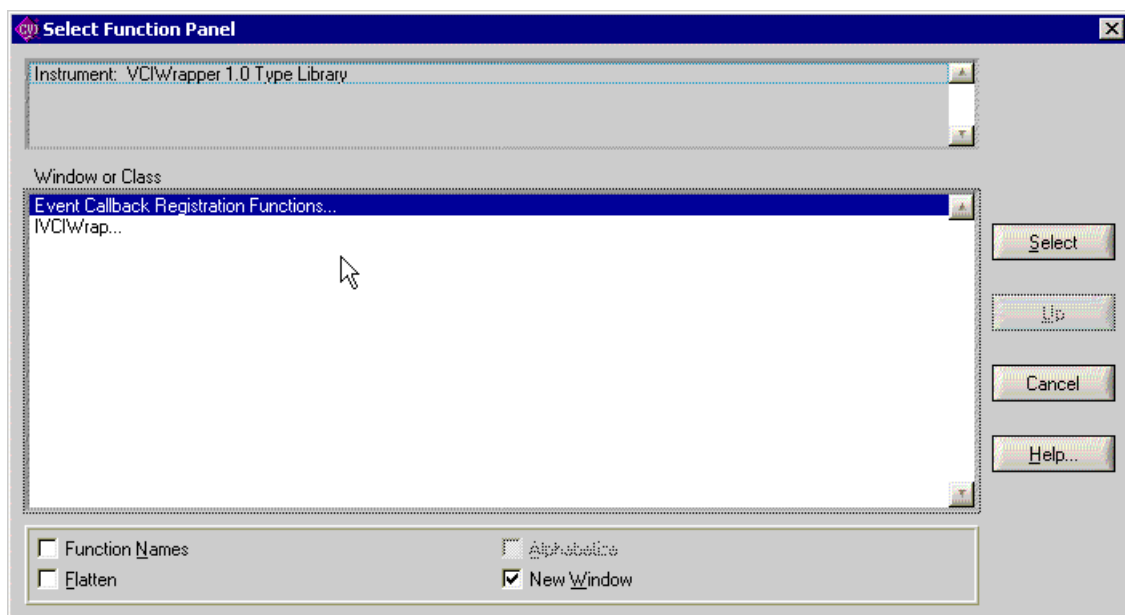


Bild 3.2-15: Auswahl der Event Callback Registrierung

### 3.2.3.3 Auswahl der Event Schnittstelle

Wählen Sie nun die Event-Schnittstelle, die das Event enthält, für das Sie eine Callback Funktion registrieren wollen. Für den VCI-Wrapper existiert nur das Event Interface IVCIWrapEvents.

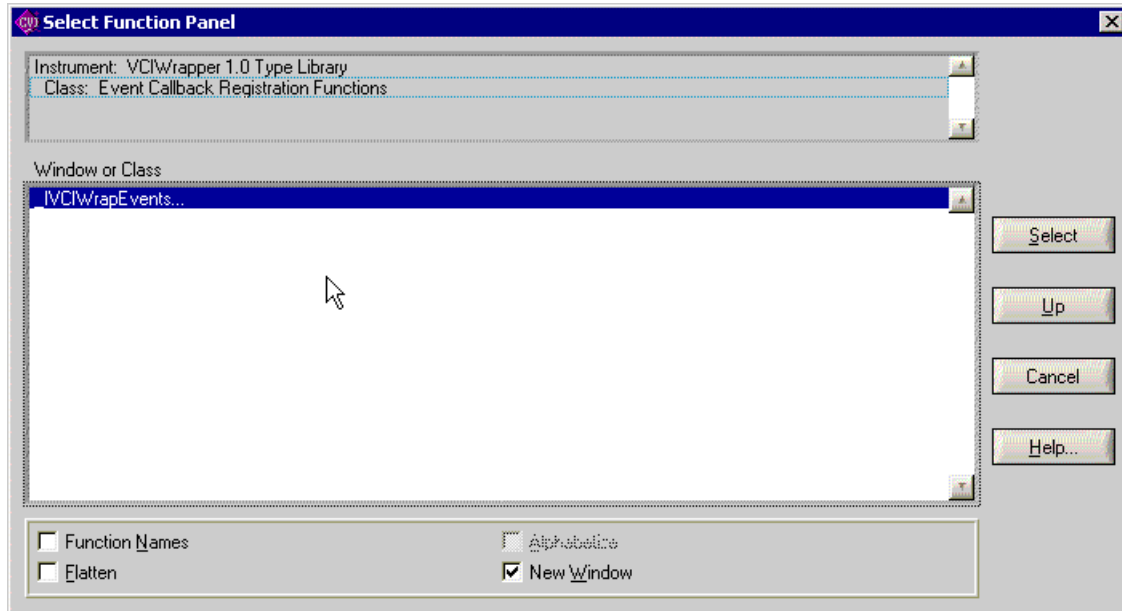


Bild 3.2-16: Auswahl der Event Schnittstelle

### 3.2.3.4 Auswahl der Registrierungsfunktion für das Event

Wählen Sie nun die Registrierungsfunktion für das entsprechende Event. Der VCI-Wrapper stellt nur ein VCIRxEvent bereit.

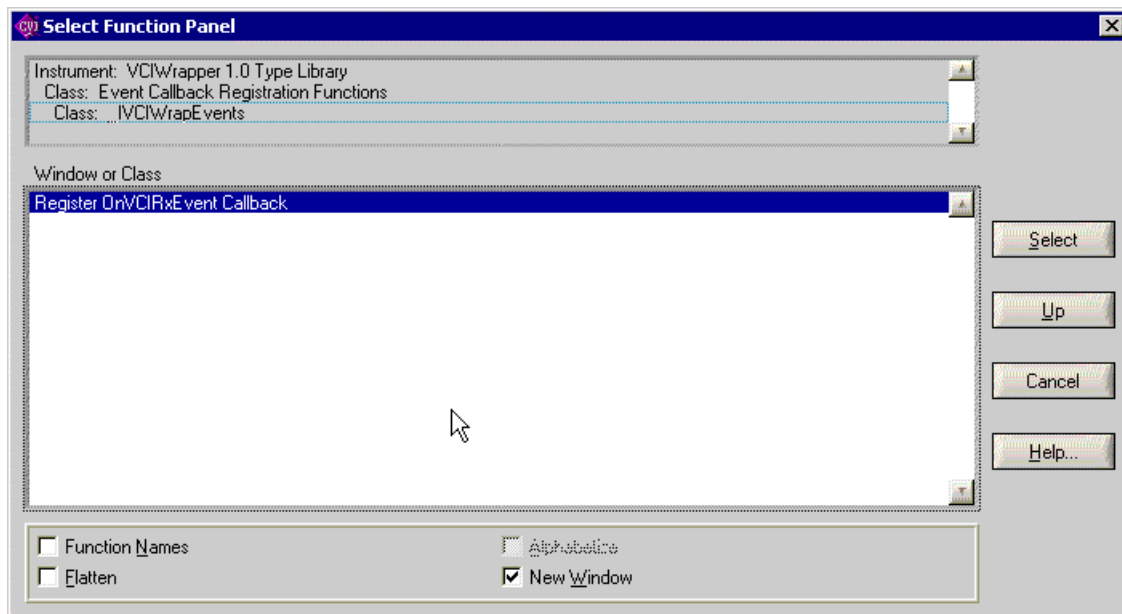


Bild 3.2-17: Auswahl der Registrierungsfunktion

### 3.2.3.5 Parametrierung des Registrierungsaufrufs

Bei der Parametrierung des Registrierungsaufrufs geben Sie nun den Namen der Event Callback Funktion an, die Sie später manuell implementieren (hier „OnRxEvent“).

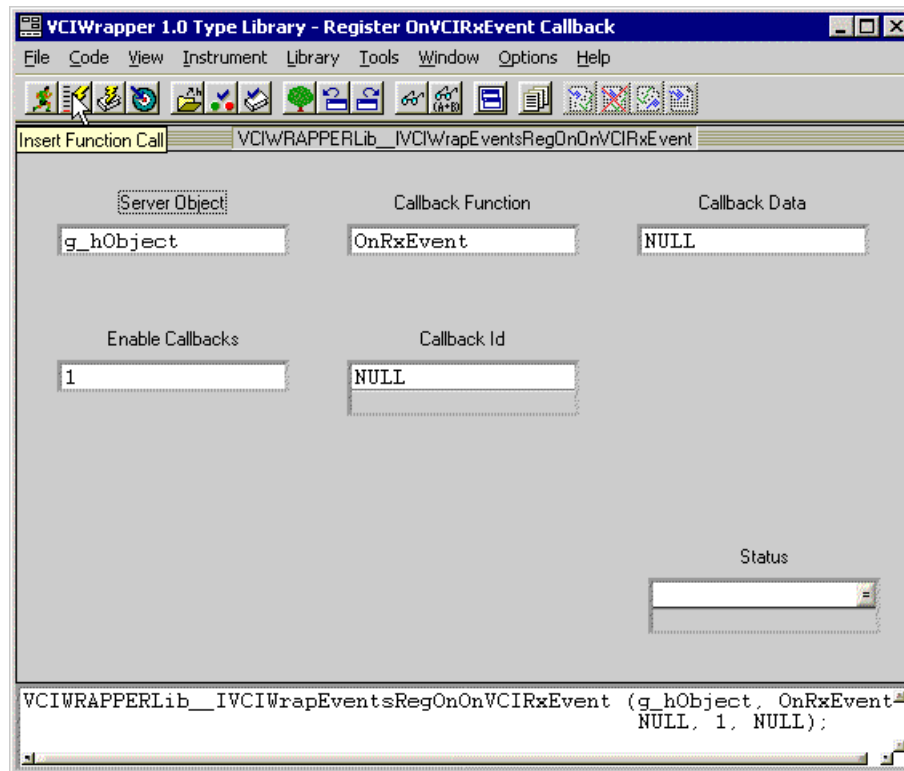


Bild 3.2-18: Parametrierung des Registrierungsaufrufs

Klicken Sie nun auf den Schalter „Insert Function Call“ in der Werkzeugleiste. Daraufhin wird der Registrierungsaufwurf an der vorab markierten Zeile eingefügt.

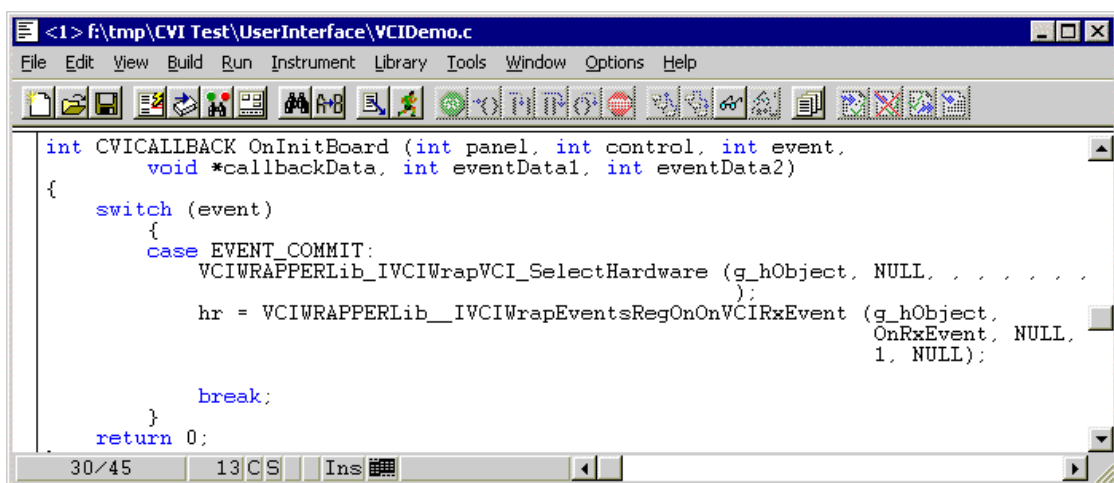


Bild 3.2-19: Der Registrierungsaufwurf

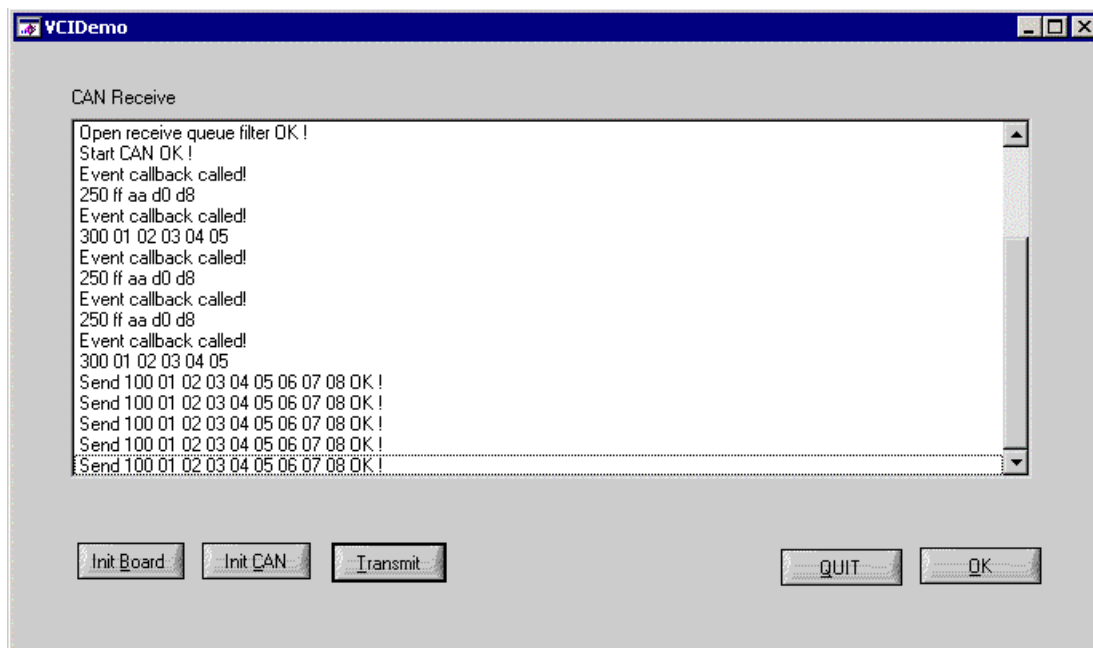
### 3.3 Beispielapplikation VCIDemo

Ein komplettes Beispiel für eine VCI Anbindung an LabWindows/CVI ist die Beispielapplikation VCI Demo.

Das Projekt VCI Demo liegt mit komplettem Sourcecode bei.

#### **ACHTUNG:**

**LabWindows/CVI arbeitet mit absoluten Pfaden. Um das Projekt mit problemlos übersetzen zu können ist es eventuell nötig, die absoluten Pfadangaben in den Dateien VCIDemo.cws und VCIDemo.prj individuell anzupassen.**



**Bild 3.3-19: Beispielapplikation VCIDemo**

VCIDemo zeigt empfangene CAN Telegramme in der Listbox an. Bei jedem Betätigen des „Transmit“-Schalters wird ein Telegramm mit ID 100 und 8 Datenbytes versendet.