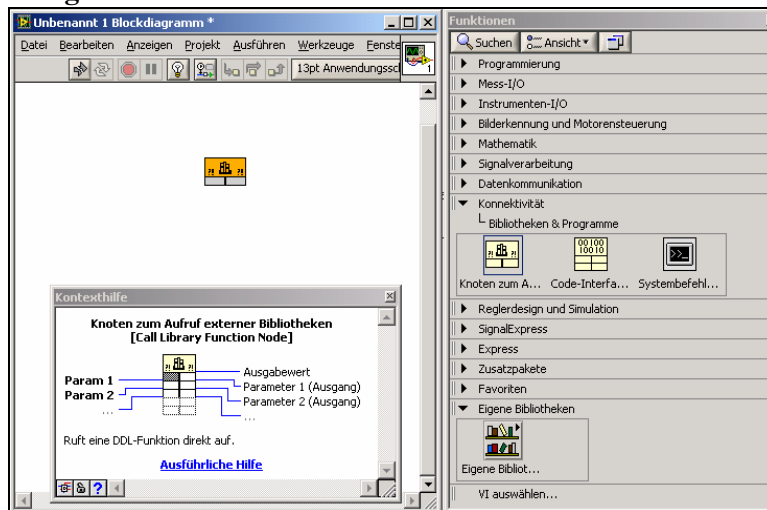


Erstellen einer DLL-Datei zur Verwendung in LabView 8.2 mittels Microsoft Visual C++ 2005 Express Edition

LabView 8.2

1. Erstellen der Prototyp-Quelldatei (*.c)

- 1.1. Einen **Knoten zum Aufruf externer Bibliotheken** in das Blockdiagramm einfügen. Den Knoten findet man im Funktionsblock unter **Konnektivität - Bibliotheken und Programme**.



- 1.2. Mit der rechten Maustaste auf den Knoten zum Aufruf externer Bibliotheken klicken und **Konfigurieren** aus dem Kontextmenü wählen, um das Dialogfeld **Aufruf externer Bibliotheken** zu öffnen.

Funktion (Pfad und Namen)

- Vorerst das Element **Bibliotheksname oder Pfad** leer lassen. Hier wird später der Pfad zur erstellten DLL angegeben, die vom Knoten aufgerufen werden soll.
- Im Feld **Funktionsname** den Namen der Funktion eintragen: **addierer**. Im Feld **Aufrufkonventionen** die Option **C** auswählen.

Parameter (Definition der Parameter)

- Im Feld **Name** den Standardnamen Rückgabety in **Fehler** ändern. Im **Typ**-Pull-down-Menü die Option **Numerisch** und im **Datentyp**-Pull-down-Menü die Option **Vorzeichenbehafteter 32-Bit-Integer** auswählen.
- Um einen weiteren Parameter (**a für Summand 1**) hinzuzufügen, auf die Schaltfläche **Einfügen (+)** rechts neben der Parameterliste klicken. Den Standardnamen arg1 im Feld **Name** durch die Angabe **a** ersetzen. Unter **Typ** die Option **Numerisch** und unter **Datentyp** die Option **Vorzeichenbehafteter 32-Bit-Integer** auswählen. Im Pull-down-Menü **Übergeben** die Option **Zeiger auf Wert** selektieren.
- Zur Definition der Parameter **b (Summand 2)** und **erg (Ergebnis)** wie unter Punkt d vorgehen.

- f Im Feld **Funktionsprototyp** müsste nun folgendes stehen:
long addierer(long *a, long *b, long *erg);
 - g Auf **OK** klicken, um die Einstellungen zu speichern und das Dialogfeld zu schließen.
- Das Symbol des Knoten zum Aufruf externer Bibliotheken wird nun entsprechend der vorgenommenen Einstellungen angezeigt.

1.3. Zum Speichern der Prototyp-Quelldatei mit der rechten Maustaste auf den Knoten klicken und **Erstellen einer *.c-Datei...** aus dem Kontextmenü wählen. Die Datei unter dem Namen **addierer.c** speichern.

Mit LabView wurde nun folgender Rumpffcode erstellt:

```
/* Call Library source file */

#include "extcode.h"

long addierer(long *a, long *b, long *erg);

long addierer(long *a, long *b, long *erg)
{
    /* Insert code here */
}
```

1.4. Speichern des Projektes und beenden von LabView.

Microsoft Visual C++ 2005 Express Edition

2. Fertigstellen der C-Quelldatei und Erstellen der DLL-Datei

2.1. Erstellen eines Bibliotheksprojekts

- a Im Menü **Datei** die Option **Neu** und anschließend **Projekt...** auswählen.
- b Im Dialogfeld **Neues Projekt – Projekttypen** unter **Visual C++** die Option **Win32** und im Bereich Vorlagen **Win32-Konsolenanwendung** auswählen. Im Feld **Name** **addierer** eintragen und als Speicherort dasselbe Verzeichnis wählen, indem sich auch die LabView Dateien befinden. Die Eingaben mit **OK** abschließen.
- c Auf der Seite **Übersicht** im **Win32-Anwendungs-Assistent** auf **Weiter** klicken. Auf der Seite **Anwendungseinstellungen** unter **Anwendungstyp** **Konsolenanwendung** auswählen und unter **Zusätzliche Optionen** die Option **Vorkompilierter Header** deaktivieren und **Leeres Projekt** aktivieren. Auf **Fertig stellen** klicken um das Projekt zu erstellen.

2.2. Einbinden und Bearbeiten des Quellcodes

Hinweis: Für das hier aufgeführte einfache Beispiel sind keine Header-Dateien erforderlich. Bei komplexeren DLLs werden Header-Dateien allerdings für alle Bibliotheken benötigt. Für nähere Informationen hierzu in der LabView-Hilfe unter Grundlagen / Aufruf textbasierten Programmcodes / Anleitung / Fertigstellen der *.c-Datei nachschlagen.

- a Im **Projektmappen-Explorer** mit der rechten Maustaste auf den Ordner **Quelldatei** klicken und aus dem Kontextmenü **Hinzufügen / Vorhandenes Element...** wählen
- b In den Ordner wechseln, in dem sich die mit LabView erstellte Prototyp-Quelldatei befindet und diese **Hinzufügen**.
- c Durch doppelklicken der Datei wird diese angezeigt.
Die Funktion der DLL muss eigens exportiert werden, damit diese in LabView zur Verfügung steht. Zum Export der Funktion wird der Befehl **__declspec (dllexport)** benötigt. **__declspec (dllexport)** ist eine Microsoft-spezifische Erweiterung der Programmiersprache C bzw. C++, näheres hierzu in der Visual C++ Hilfe.
Den Prototyp-Quellcode wie dargestellt vervollständigen:

```
/* Call Library source file */  
  
#include "extcode.h"  
  
__declspec(dllexport) long addierer(long *a, long *b, long *erg);  
  
long addierer(long *a, long *b, long *erg)  
{  
    *erg = *a + *b;  
    return 0;  
}
```

- d Über **Datei Alle Speichern** das Projekt speichern.

2.3. Konfiguration zum Erstellen einer DLL-Datei

- a Im Menü **Projekt** die Option **addierer-Eigenschaften** wählen.
- b Im sich öffnenden Fenster im linken Bereich den Knoten **Konfigurationseigenschaften** erweitern und die Option **Allgemein** auswählen.
- c Im oberen Teil des Fensters jetzt **Konfiguration** auf **Alle Konfigurationen** stellen und **Plattform** auf **Win32**.
- d Jetzt unter **Konfigurationseigenschaften / Allgemein** auf der rechten Seite unter **Projektstandards** den **Konfigurationstyp** über das Pulldown-Menü auf **Dynamische Bibliothek (.dll)** stellen.
- e Unter **Konfigurationseigenschaften** den Knoten **C/C++** erweitern, dort **Allgemein** auswählen und auf der rechten Seite unter **zusätzliche Includeverzeichnisse** den Pfad zur LabView Bibliothek „extcode.h“ eintragen. Diese Bibliothek befindet sich im LabView-Installationsverzeichnis im Ordner cintools "C:\Program Files\National Instruments\LabVIEW 8.2\cintools"
- f Die Konfiguration der Eigenschaften mit **OK** abschließen.

2.4. Kompilieren der DLL-Datei

Im Menü **Erstellen** die Option **Projektmappe erstellen** auswählen, dadurch wird die DLL-Datei erstellt, die in nun in LabView verwendet werden kann. Die DLL befindet sich im C++ - Projektordner im Verzeichnis **debug**.

LabView 8.2

3. Einbinden der DLL

- a Öffnen des unter 1. gespeicherten Projekts.
- b Mit der rechten Maustaste auf den unter 1. erstellten **Knoten zum Aufruf externer Bibliotheken** klicken und im Kontextmenü **Konfigurieren...** auswählen.
Im sich öffnenden Fenster unter **Funktion** im Feld **Bibliotheksname oder Pfad** den Pfad zur erstellten DLL-Datei eintragen.
- c Einstellungen mit **OK** abschließen.

Zum Testen:

Die Eingänge für a und b mit entsprechenden Bedienelementen verbinden und den Ausgang erg mit dem entsprechendem Anzeigeelement. Wenn alles geklappt hat wird der Pfeil für Ausführen ganz dargestellt und die Funktion kann gestartet werden.

