

IPC@CHIP CLIB

How to integrate the CLIB





Table of Contents

1	INTR		. 3
2	ном	/ TO INTEGRATE THE CLIB IN A BORLAND C PROJECT	. 4
	2.1	STEP 1 - COPY THE CLIB	. 4
	2.2	STEP 2 - CREATE YOUR PROGRAM DIRECTORY	. 4
	2.3	STEP 3 - CREATE THE BORLAND C 5.02 PROJECT	. 4
	2.4	STEP 4 - ADJUST THE PROJECT SETTINGS	. 5
	2.5	STEP 5 - ADD THE CLIB TO THE PROJECT	. 6
	2.6	STEP 6 - WRITE YOUR PROGRAM	. 7
3	ном	/ TO INTEGRATE THE CLIB API DOCUMENTATION INTO THE @CHIP-RTOS API	
	DOC		. 8
	3.1	STEP 1 - COPY THE CLIB DOCUMENTATION	. 8
	3.2	STEP 2 - OPEN AND USE THE CLIB HTML DOCUMENTATION	. 8



1 Introduction

The libs provide an easy access to the IPC@CHIPs operating system using the programming language C. Beside the binary library files for the borland compiler a documentation in HTML format is provided.

The CLIB software package comes in two versions:

- CLIBVXXXBIN.ZIP includes only the binaries of the @CHIP-RTOS libraries.
- CLIBVxxxSRC.ZIP includes also the source code of the libraries.

This document describes how to integrate the:

- The CLIB into your Borland C 5.02 project and
- The CLIB documentation into the @CHIP-RTOS API documentation.

The library is available in 4 different models. One model for every comiler memory model:

- CLIBxxxC.lib Compact
- CLIBxxxL.lib-Large
- CLIBxxxM.lib-Medium
- CLIBxxxS.lib-Small

Note: The compiler memory models have NO cohesive with the available @CHIP-RTOS variants! We advise to use memory model Large.



2 How to integrate the CLIB in a Borland C project.

2.1 Step 1 - Copy the CLIB

First you have to copy the contents of the CLIB ZIP archive to your local project directory. Make a new directory and name it CLIB. Copy all header files (*.h) and all library files (*.lib) to this directory. E.g. You have now a directory C:\Projects\Clib with the CLIB contents.

2.2 Step 2 - Create your program directory

Make a directory inside of your project directory which should store your program files. E.g. You have now a directory C:\Projects\Hello which is empty.

2.3 Step 3 - Create the Borland C 5.02 project

Open the Borland C 5.02 program and create a new project.

In the field "Project Path and Name" choose the new directory, which you have created for the program followed by the project name, e.g.C:\Projects\hello\hello.ide

Make the correct project settings.

- Target Type: Application [.exe]
- Platform: DOS (Standard)
- Target Model: Large
- Frameworks: Deselect Class Library
- Math Support: None
- Libraries: No Exceptions

🚰 Borland C++				
File Edit Search View Project Script Tool Debug	Options Window Help	New Target		<u>? ×</u>
Tiel Calc Solici Wer Project School Decay New Open Close Save Ctri+K Ctri+S Save as Save all Print Print Save all Print Print Save all Print Save all Print Save all Print Save all Save all Save all Print Save all Print Save all Prin	AppExpert	Project Parlyet Project Parly and Name: C:VProjects/hello/hello.ide Target Name: hello Target Name: Parlyet Pa	Frameworks: Class Lbrary Hath Support C Floating Point C Enulation None Lbraries: No Exceptions BGI Alternate Startup Diagnostic	Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Cancel Ca
		11:37:44		

Press now the "Advanced" button and choose the *.c node type.

Advanced Options	? ×
Initial Nodes: C .cpp Node ⊙ .c Node C No Source Node	Cancel
	📍 Help

Close the "Advanced Options" dialog and the "New Target" dialog with "OK".



2.4 Step 4 - Adjust the project settings

Open the menu "Options" and select "Project...".

Select "Directories" and change the include and library path to:

```
Include: $env(bcroot)\include;..\clib<sup>1</sup>
Library: $env(bcroot}\lib
```



Now select "16-bit Compiler" and choose the 80186 instruction set for compiling.

Project Options		? ×
Topics: ◆ Directories ♦ Compiler ■ 16-bit Compiler ■ 16-bit Compiler ● Calling Convention ● Memory Model ● Segment Names Data ● Segment Names Code ● Schurkent Names Code ● 32-bit Compiler ♦ Cotent ♦ Doptimizations ♦ Messages ♦ Linker ■ Libration ♥ Resources ■ Build Attributes ● Make	Processor Instruction set:	
Select a target processor	OK DUndo Page X Cancel ?	Help

Close the dialog with the "OK" button.

¹ The expression "\$env(bcroot)" is translated to the installation directory of the Borland 5.02 compiler



2.5 Step 5 - Add the CLIB to the project

Go to the project window, use the right mouse button to click on the project node. Choose "Add node" from the popup menu.



Change the path to the CLIB directory.

Add to Project List				<u>?</u> ×	Add to Project L	ist				<u>? ×</u>
Suchen in:	🔁 hello		💽 🗕 🗲 🖬	•	Suchen in:	Projects		•	🗢 🗈 💣 📰•	
Verlauf Desktop	hello.c				Verlauf Desktop	hello				
Ligene Dateien Arbeitsplatz	2-1-1			Ölfson	Arbeitsplatz	Detrimerer				Ölfnen
Netzwerkumg	Dateityp:	C++ source (*.cpp,*.c)		Abbrechen	Netzwerkumg	Dateityp:	C++ source (*.cpp;*.c)			Abbrechen

Select from the lower listbox the file extension type *.lib or *.* and open the CLIB with the name CLIBxxxL.LIB for the Large memory model library.

Add to Project L	ist				<u>? ×</u>
Suchen in:	🔁 clib		•	🗢 🗈 💣 🎟 •	
Verlauf Verlauf Desktop Eigene Dateien	CLIB200C.LIB				
	Dateiname:	CLIB200L.LIB		-	Öffnen
Netzwerkumg	Dateityp:	Libraries (*.lib)		•	Abbrechen



2.6 Step 6 - Write your program

Include the header file CLIB.H in your C files. After this you can use all CLIB function.

🔗 Borland C++ - hello			<u>- 🗆 ×</u>
File Edit Search View Project Script Tool Debug Options Window H	elp		
er reference a state sta	V N. V.		
C:\Projects\hello.c		Project : c:\projects\hello\hello.ide	
#include <stdio.h></stdio.h>	· · •	- 🖌 🖌 hello.exe [.exe]	
#include <conio.h></conio.h>	. II.	clib.clib2001.lib [.lib]
#include "clib.n"	. II.	merio.c [.c] code siz	e-04
void main (void)			
// Set Focus to user program			
BIOS_Set_Focus (FOCUS_APPLICATION);			
printf(") pHello world");			
princi ((interio world);			
// wait for key pressed			- FI
while (!kbhit())			
		N	
// sleep for 10 Ms		4	
All steep_time(to);			
<pre>printf("\nGoodbye");</pre>			
// Set Focus to Doth (shell + user)			
BIOS_SEC_FOCUS(FOCUS_BOIN);			
	•		
	23:4	Insert 11:59	:26 //,



3 How to integrate the CLIB API documentation into the @CHIP-RTOS API documentation

To use the full documentation, the complete @CHIP-RTOS documentation is required. The CLIB API documentation is an extension to it. Copy the CLIB directory from this software package into the DOC directory of the whole @CHIP-RTOS docu.

3.1 Step 1 - Copy the CLIB documentation

There is a subdirectory called CLIB in the CLIB ZIP archive. Copy this directory and it's contents into the @CHIP RTOS API documentation directory.

E.g. You have your @CHIP-RTOS HTML documentation at this path: C:\IPC@CHIP\SC12\DOC and your CLIB HTML documentation at this location: C:\IPC@CHIP\SC12\DOC\CLIB

3.2 Step 2 - Open and use the CLIB HTML documentation

The start of the CLIB HTML documentation is in e.g. C:\IPC@CHIP\SC12\DOC\CLIB\LIBINDEX\clib_index.htm





On the @CHIP-RTOS HTML documentation index e.g. C:\IPC@CHIP\SC12\DOC\index.htm should now appear a new link to the CLIB documentation.



Each documented CLIB function has a HTML link to the invoked RTOS API software interrupt, which is described in the @CHIP-RTOS HTML documentation.

🖉 @CHIP-RTO5 C Library V2.00 - RTO5 API - Microsoft Internet Explorer	
Datei Bearbeiten Ansicht Favoriten Extras ?	1
🗢 Zurück 🔹 🔿 🗸 🔯 🖓 Suchen 🙀 Favoriten 🐲 Medien 🥨 🖏 🕁 🐲 🖅 🔹	
Adresse 🖉 🖉 Wechseln zu	Links »
RTX_Create_Task Without_Run RTX_Delete_Task RTOS API This library function invokes a RTOS software interrupt. Refer to this RTOS API function's documentation for more details. Related Topics Related Topics	-
IPC@CHIP System Tasks	
This <u>APIList</u> List of <u>C Libraries</u> @ <u>CHIP-RTOS</u> Main Index 	_
	-
🙆 📃 📃 Arbeitsplatz	//