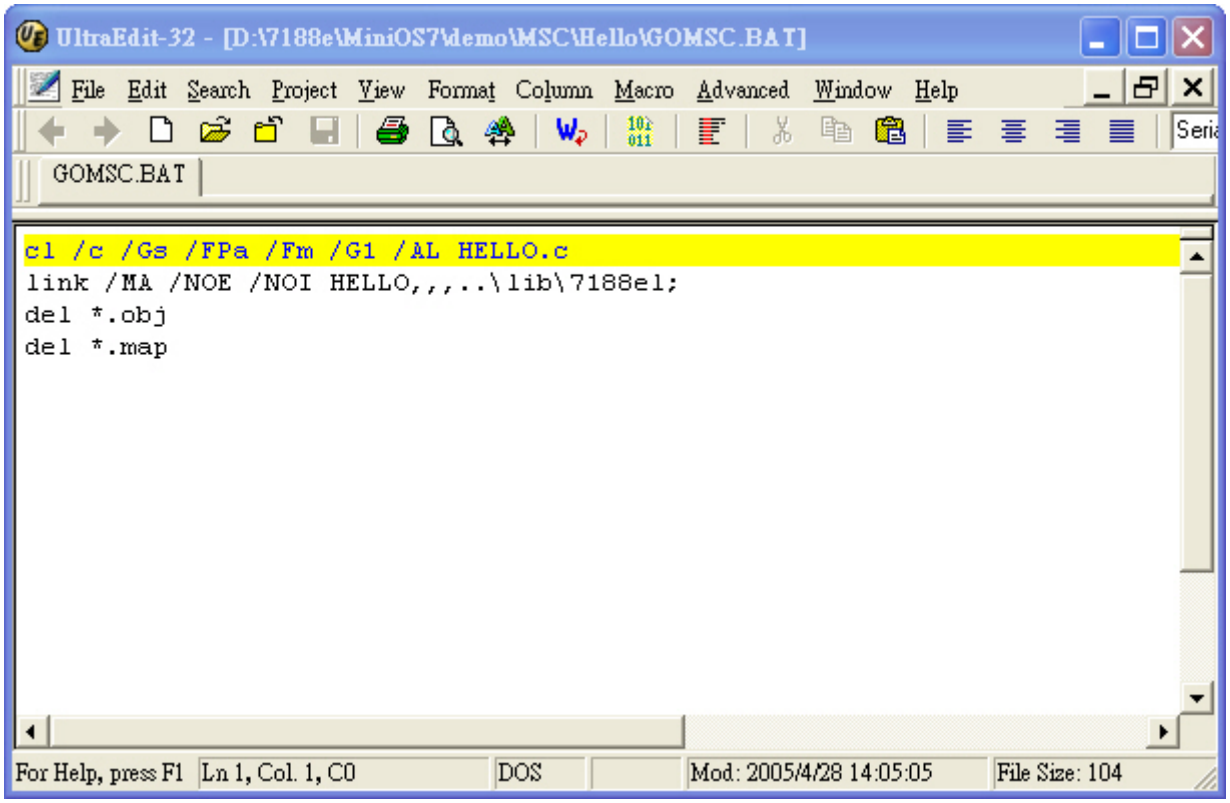
First editor: Liam
Last editor: Vic, 2014/2/25

Compiling using the MSC 6.00 Compiler

Step 1: In the source file folder, create a batch file called Gomsc.bat.



NOTE: **/C** : don't strip comments

/Gs : no stack checking

/FPa : calls with altmath

/Fm : [map file]

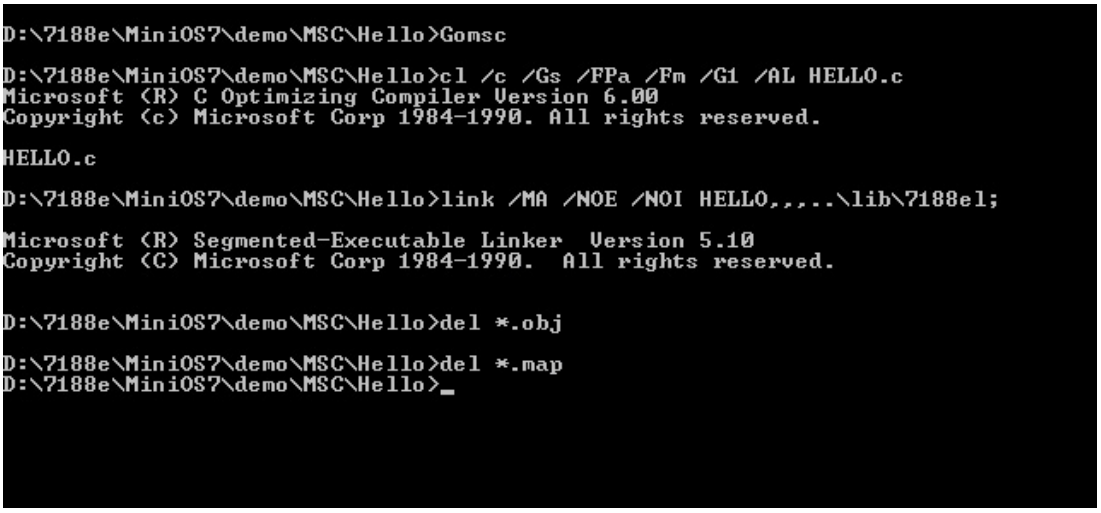
/G1 : 186 instructions

/AL : large model

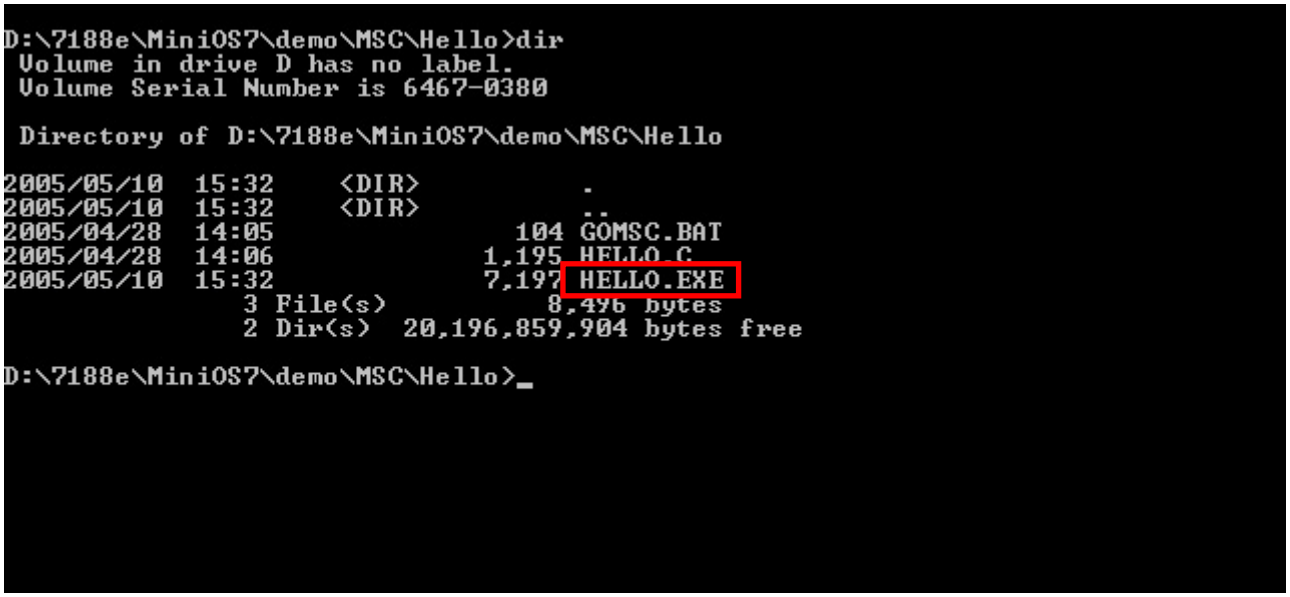
HELLO.C : the source file

..\lib\7188el : the path of the function library

Step 2: Run the Gomsc.bat file.



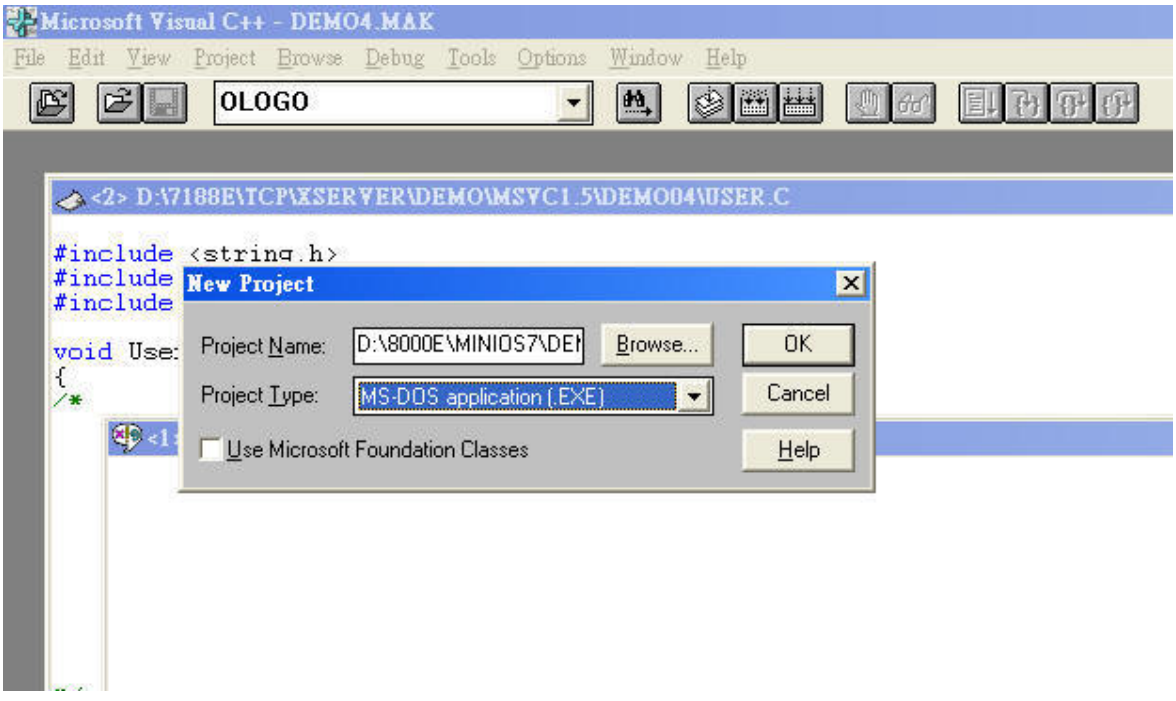
Step 3: A new executable file will be created if it is successfully compiled.



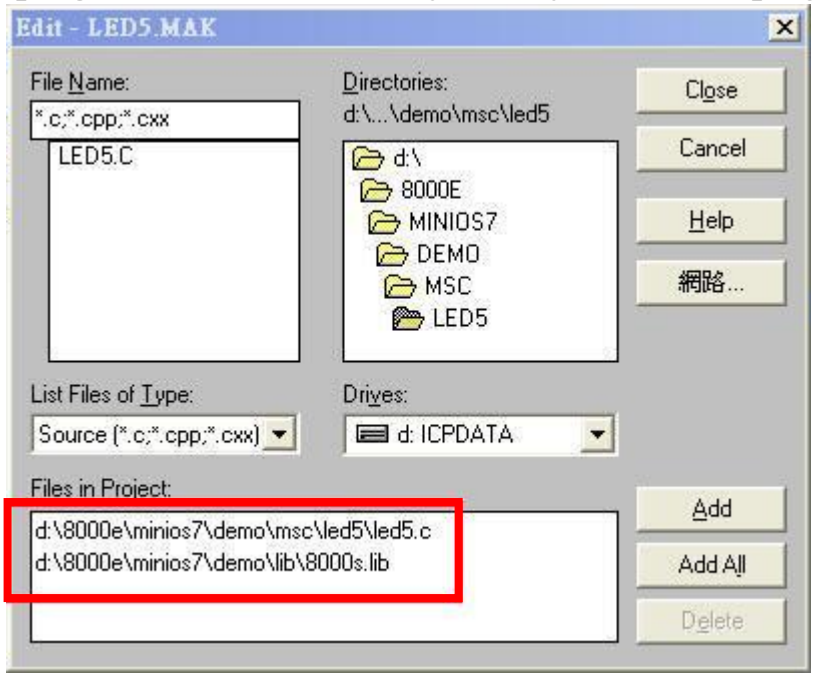
Compiling using the MSVC 1.50 Compiler

Step 1: Run MSVC.exe

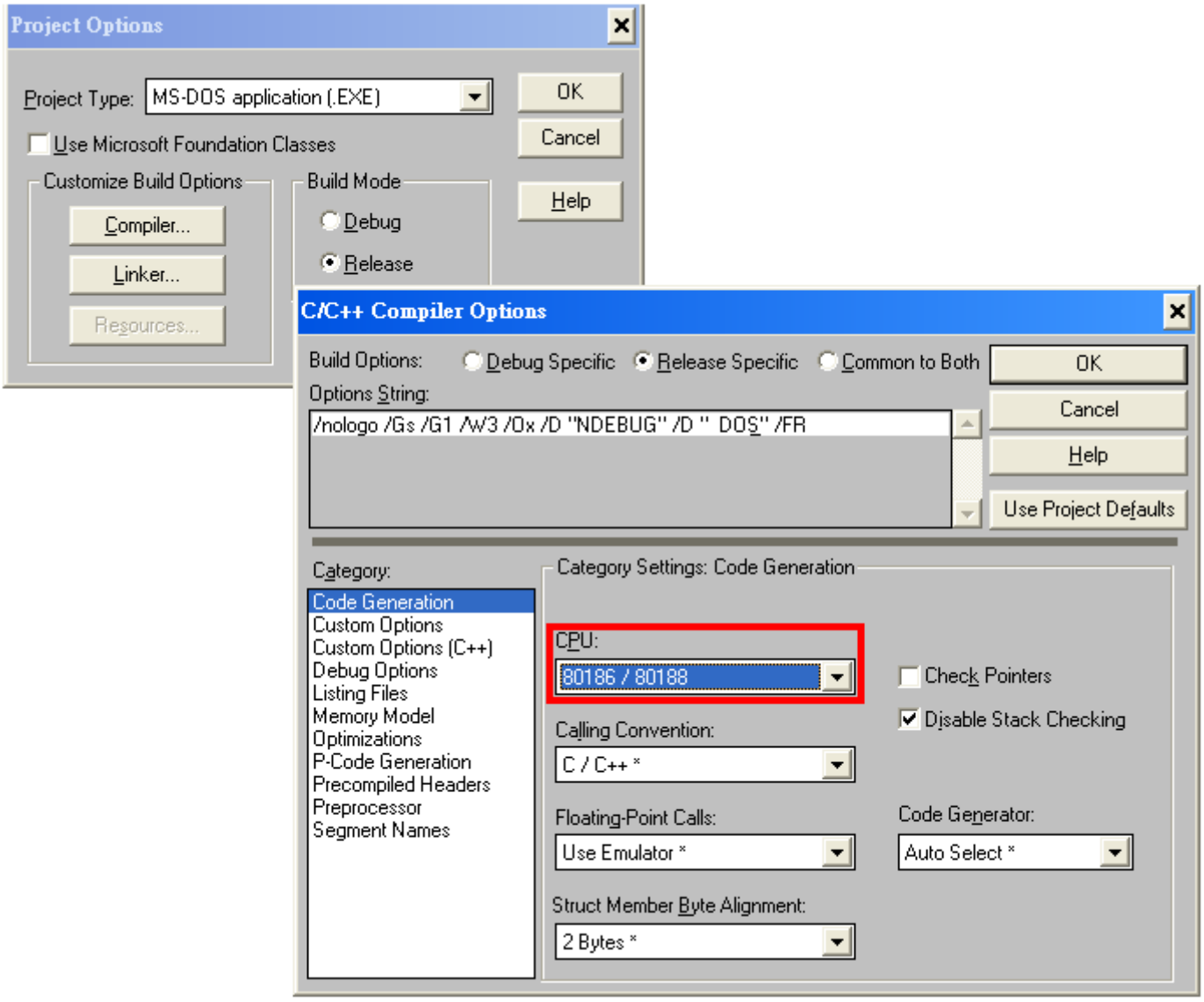
Step 2: Create a new project (*.mak) by entering the name of the project in the Project Name field and then select MS-DOS application (EXE) as the Project type.



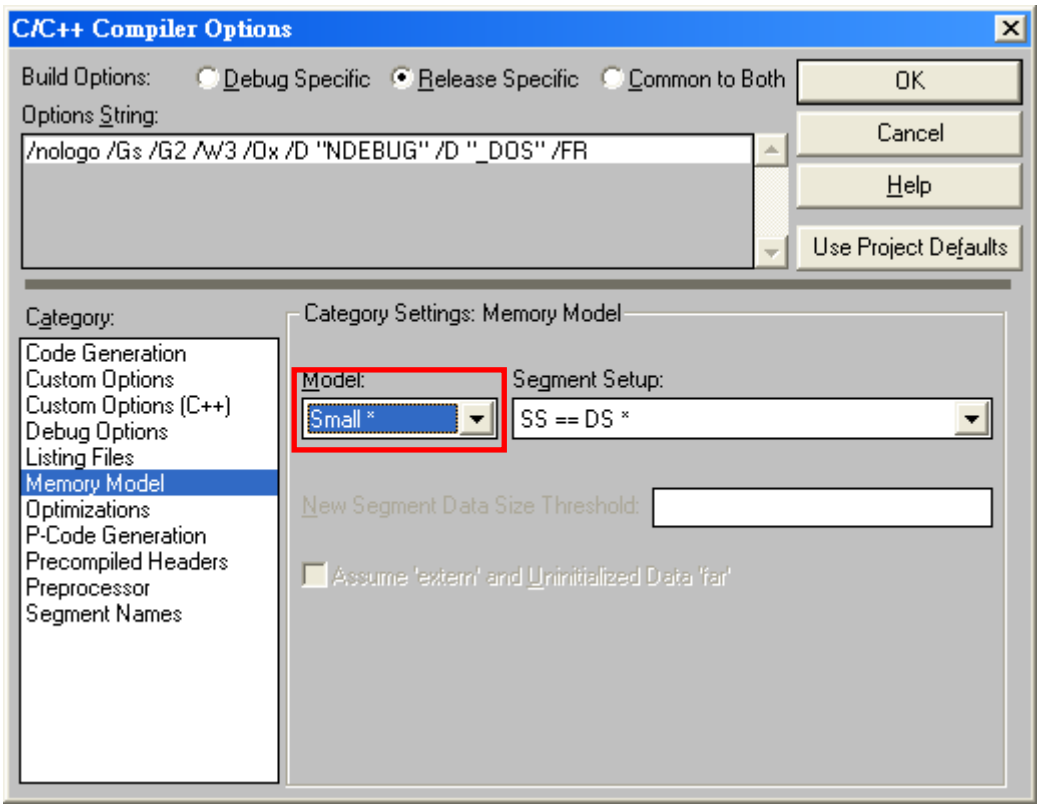
Step 3: Add the user's program and the necessary library files to the project.



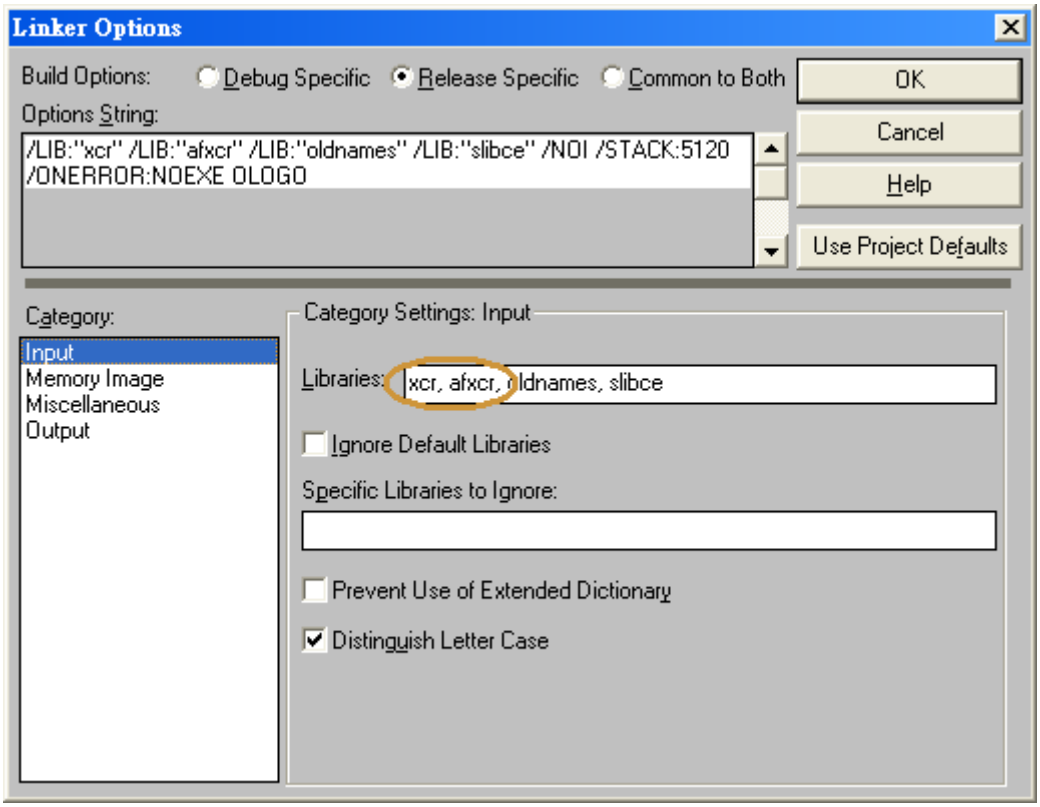
Step 4: Set the Code Generation on the Compiler.

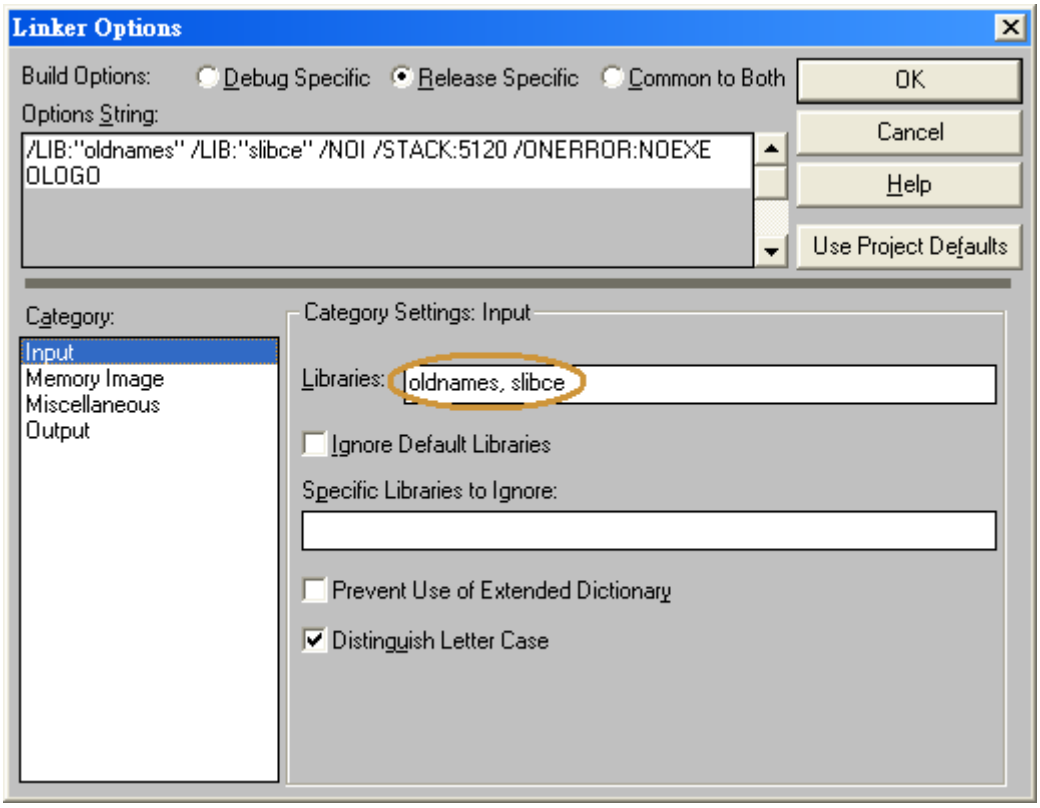


Step 5: Change the Memory model (Small for 8000s.lib\7188es.lib, large for 8000l.lib\7188el.lib).

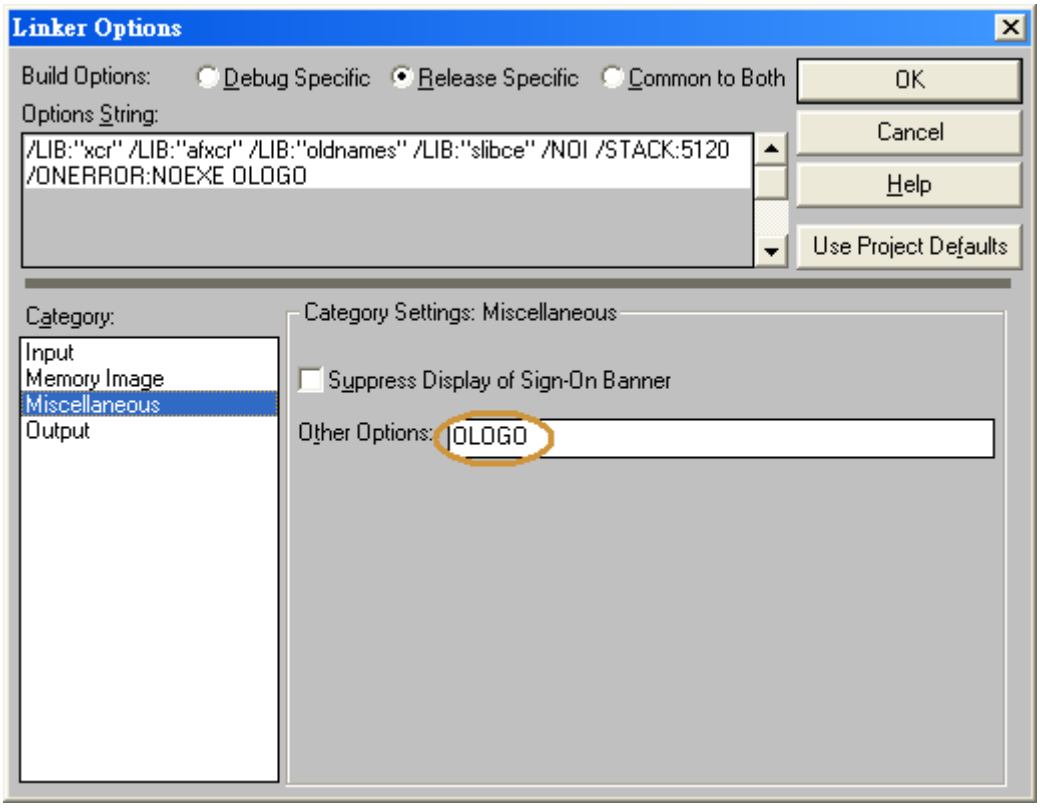


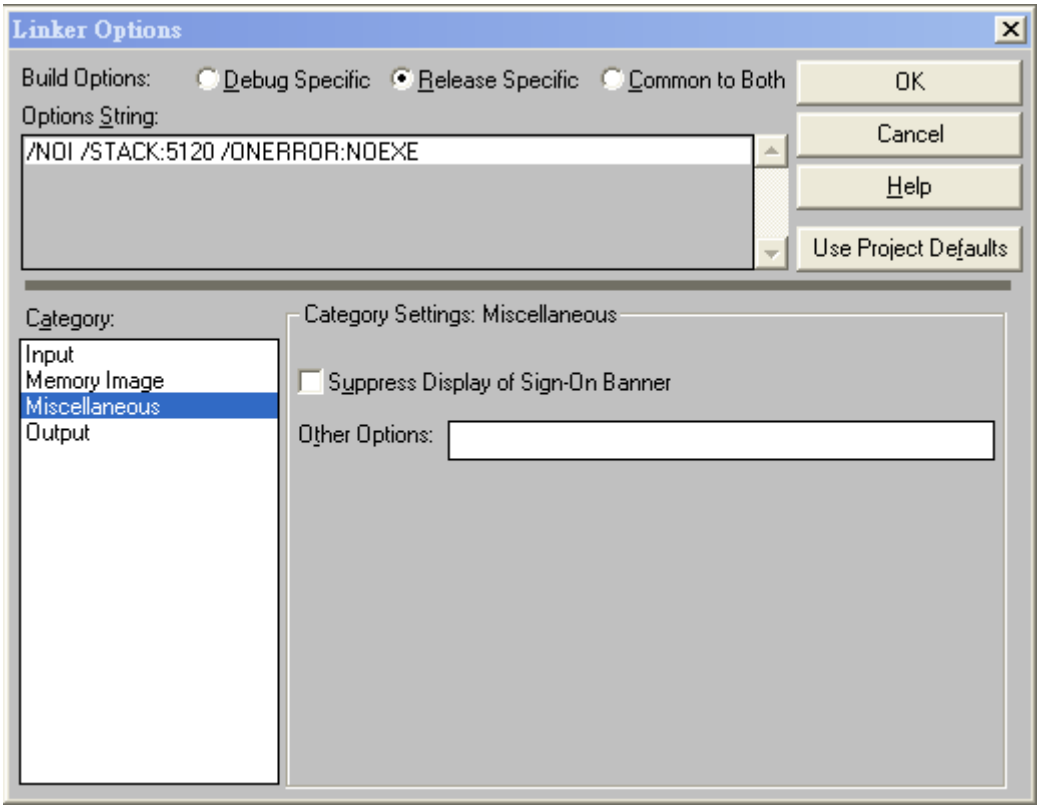
Step 6: Remove the xcr, afxcr library from the Input Category.



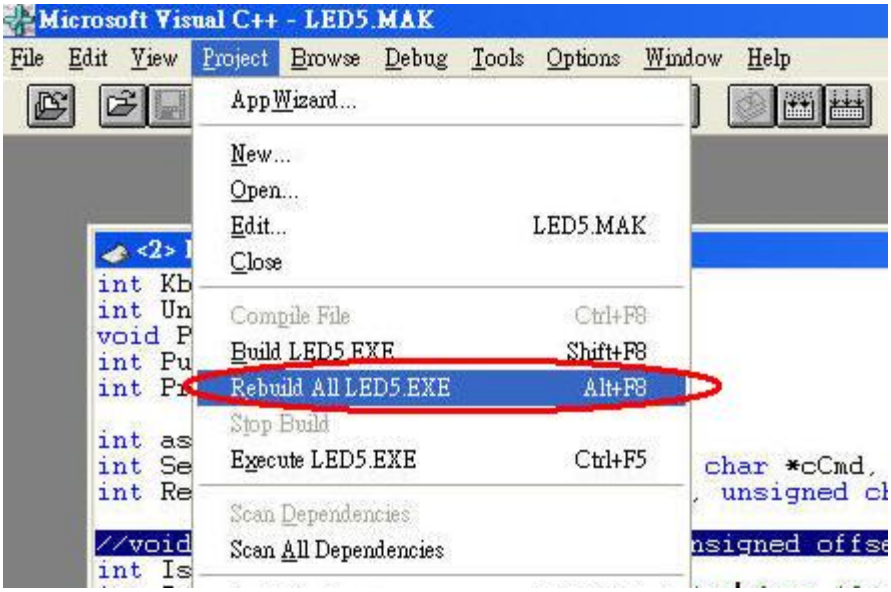


Step 7: Remove the OLOGO option from the miscellaneous Category.





Step 8: Rebuild the project.



```
<1> Output
Initializing...
Compiling...
d:\8000e\minios7\demo\msc\led5\led5.c
Linking...

Microsoft (R) Segmented Executable Linker  Version 5.60
Copyright (C) Microsoft Corp 1984-1993.  All rights reserved.

Object Modules [.obj]: /NOI /STACK:5120 /ONERROR:NOEXE   LED5.OBJ +
Object Modules [.obj]:
Run File [LED5.exe]: LED5.EXE
List File [d:LED5.map]: nul
Libraries [.lib]: c:\msvc\lib\+
Libraries [.lib]: c:\msvc\mfc\lib\+
Libraries [.lib]: ..\..\LIB\8000S.LIB+
Libraries [.lib]: oldnames+
Libraries [.lib]: slibce;
Creating browser database...
LED5.EXE - 0 error(s), 0 warning(s)
```

NOTE: If an error occurs while compiling, please commend out the line "void far *_MK_FP(unsigned segment, unsigned offset);" in the 8000.h include file.

```
<1> Output
Compiling...
d:\8000e\minios7\demo\msc\hello\hello.c
d:\8000e\minios7\demo\msc\hello\..\..\lib\8000.h(303) : error C2062: type 'void ' unexpected
 CL returned error code 2.
HELLO.C - 1 error(s), 0 warning(s)
```

<3> D:\8000E\MINIOS7\DEMO\LIB\8000.H

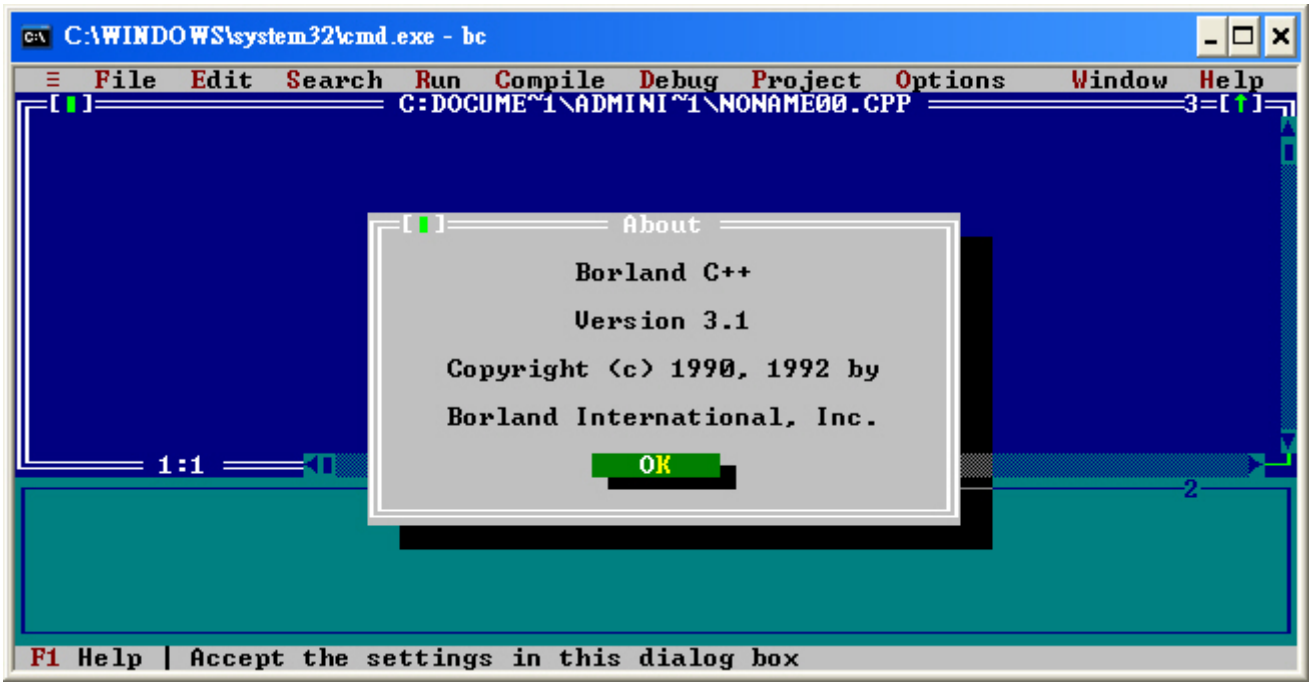
```
int Kbhit(void);
int Ungetch(int key);
void Putch(int data);
int Puts(char *str);
int Print(char *fmt,...);

int ascii_to_hex(char ascii);
int SendCmdTo7000(int iPort, unsigned char *cCmd, int iChksum);
int ReceiveResponseFrom7000(int iPort, unsigned char *cCmd, long lTimeout, int iChksum);

void far *_MK_FP(unsigned segment,unsigned offset);
int IsResetByWatchDogTimer(void);
int IsResetByPowerOff(void); /* for bios date 12/12/98 or later */

int Show5DigitLedWithDot(int pos, int data);
void Set5DigitLedTestMode(int mode);
void Set5DigitLedIntensity(int mode);
void Disable5DigitLed(void);
void Enable5DigitLed(void);
unsigned GetLibVersion(void);
```

<3> D:\8000E\MINIOS7\DEMO\LIB\8000.H

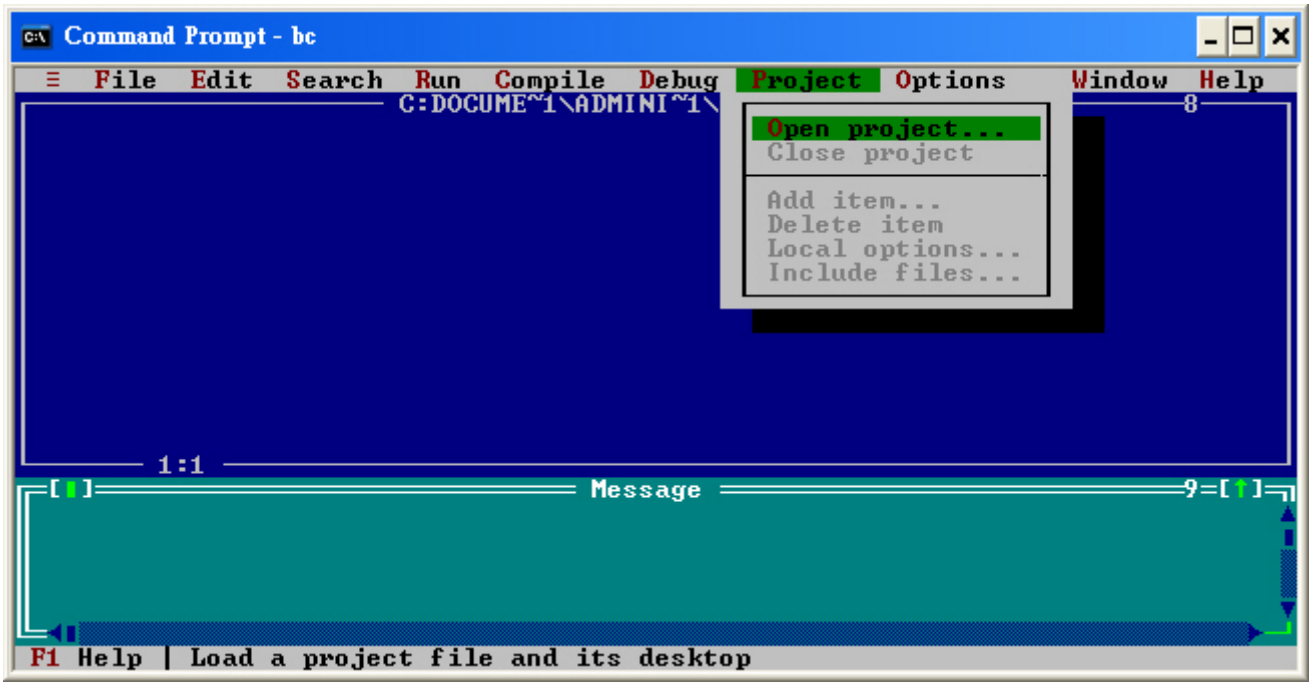
```
int Kbhit(void);
int Ungetch(int key);
void Putch(int data);
int Puts(char *str);
int Print(char *fmt,...);

int ascii_to_hex(char ascii);
int SendCmdTo7000(int iPort, unsigned char *cCmd, int iChksum);
int ReceiveResponseFrom7000(int iPort, unsigned char *cCmd, long lTimeout, int iChksum);

/*void far *_MK_FP(unsigned segment,unsigned offset);
int IsResetByWatchDogTimer(void);
int IsResetByPowerOff(void); /* for bios date 12/12/98 or later */

int Show5DigitLedWithDot(int pos, int data);
void Set5DigitLedTestMode(int mode);
void Set5DigitLedIntensity(int mode);
void Disable5DigitLed(void);
void Enable5DigitLed(void);
```

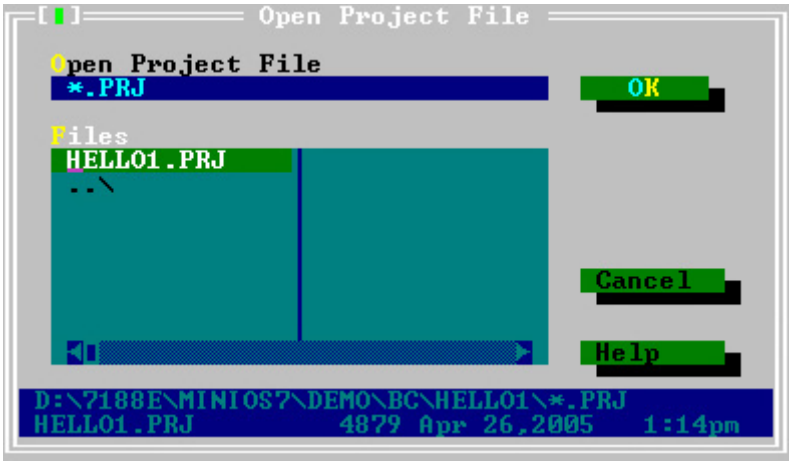
Compiling using the BC++ 3.1 IDE

Step 1: Run Borland C++ 3.1.

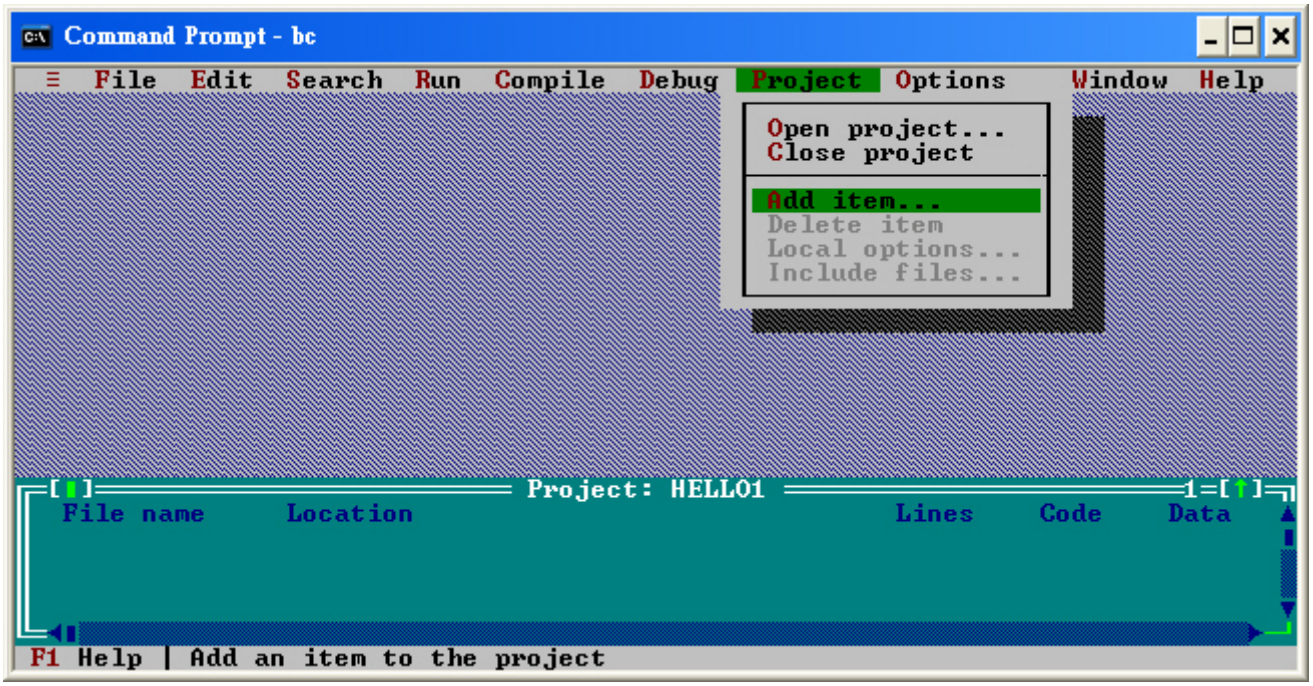


Step 2: Create a new project file (*.prj).

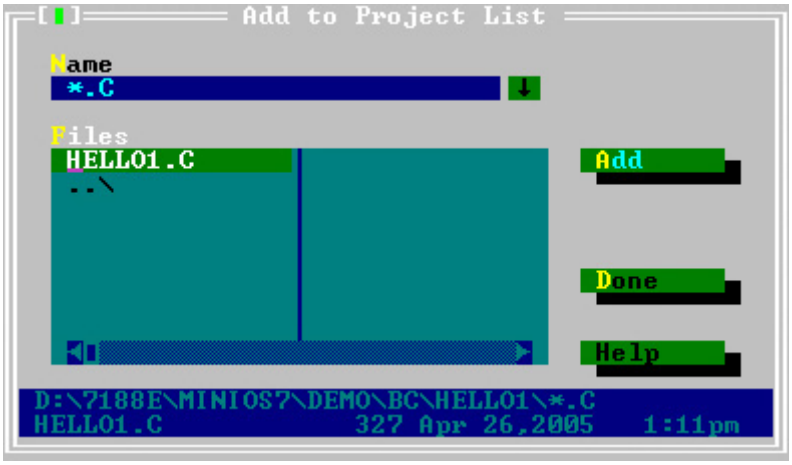




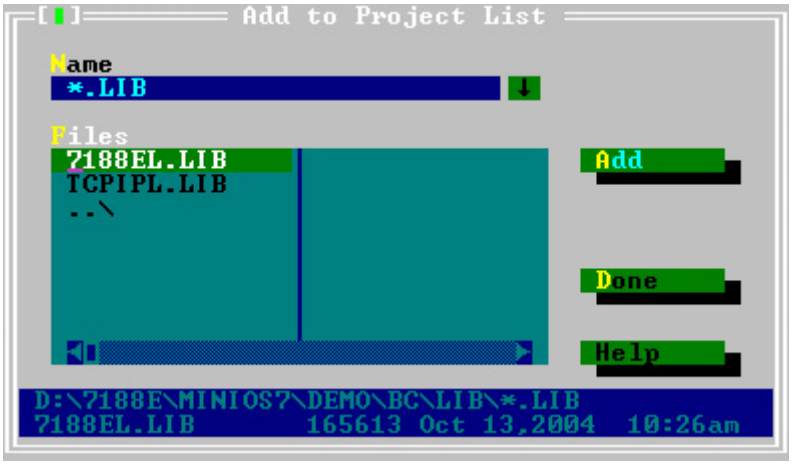
Step 3: Add all necessary files to the project.



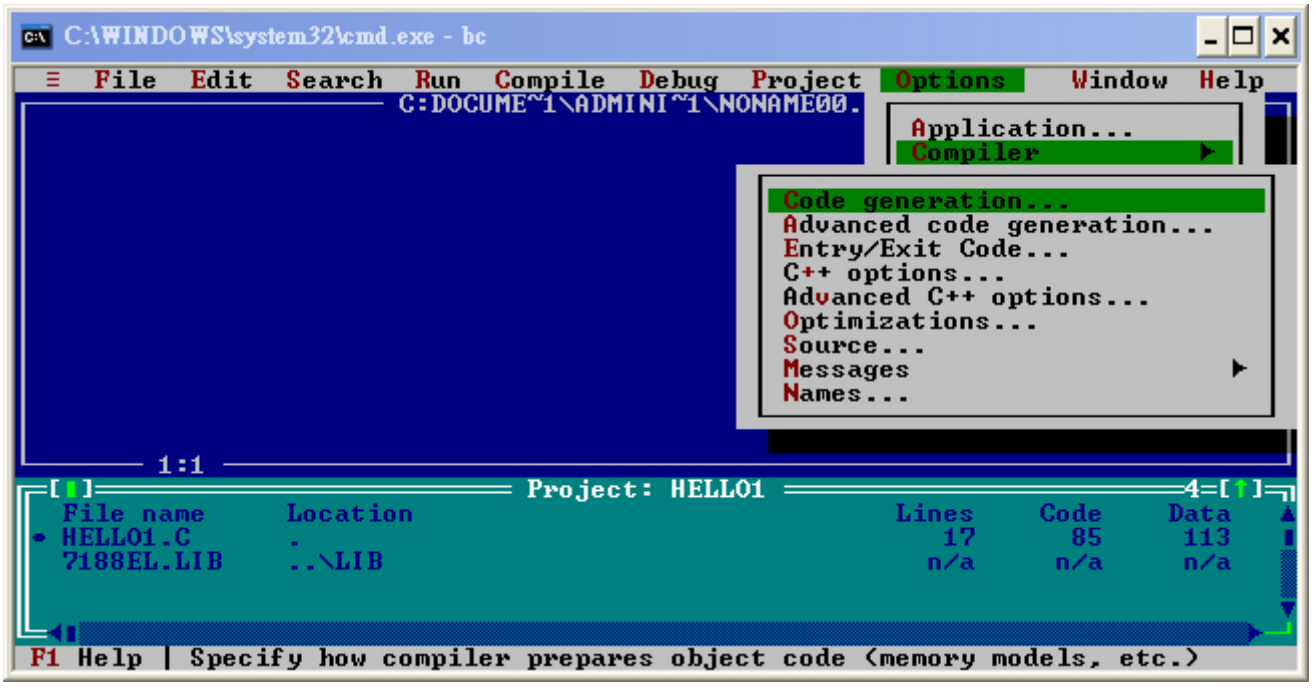
3.1 Select the source file.



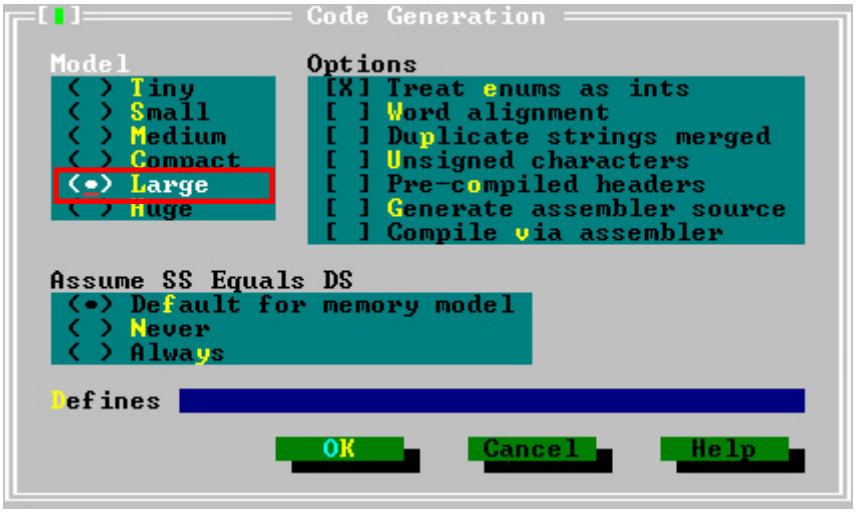
3.2 Select the function library and then click the Done button.



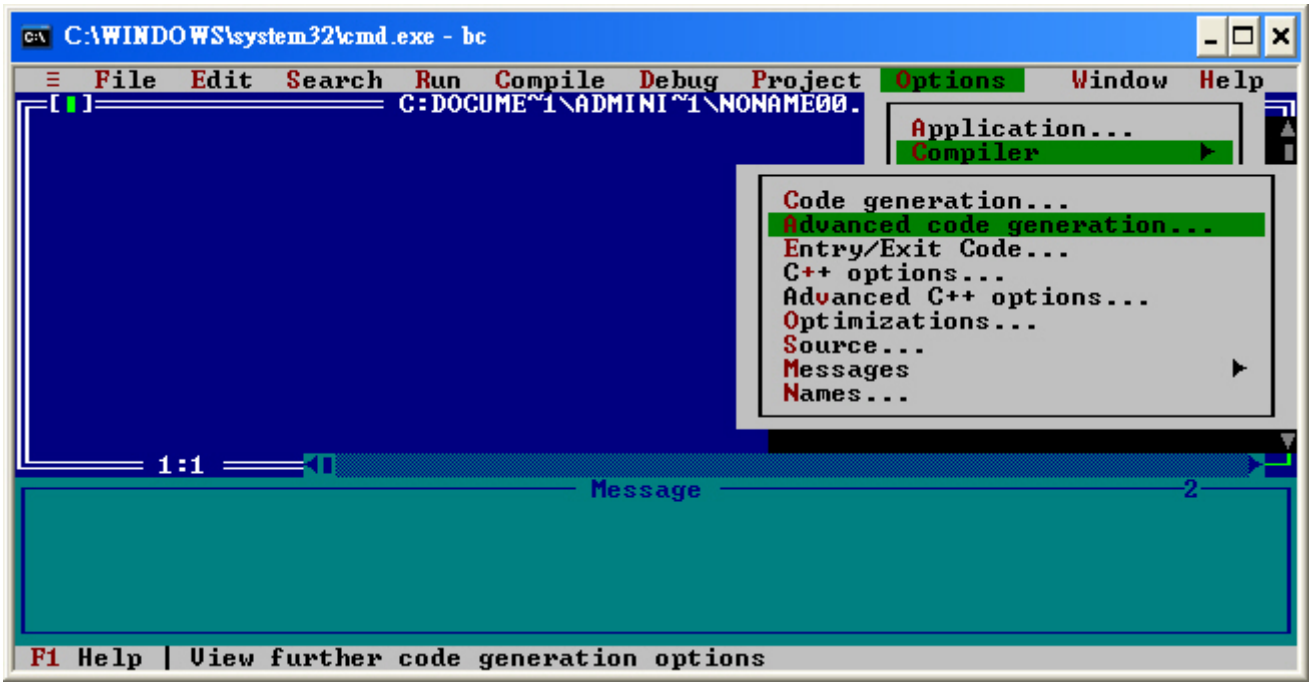
Step 4: Set the Code generation options.



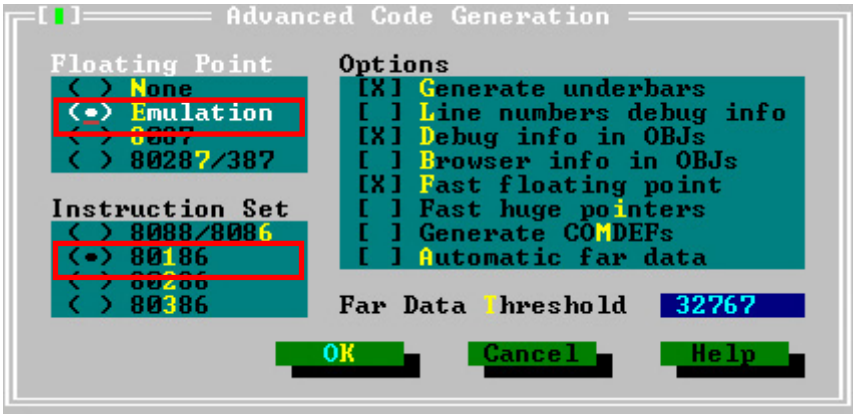
4.1 Change the Memory model (Small for 8000s.lib/7188es.lib, large for 8000l.lib/7188el.lib).



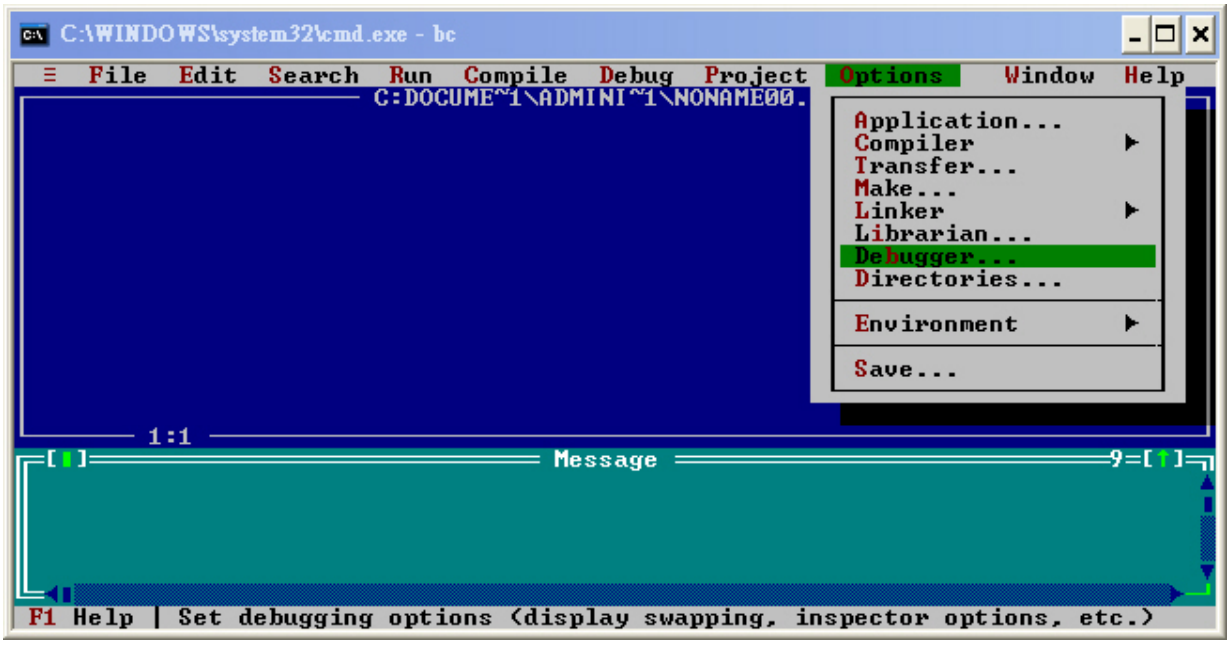
Step 5: Set the Advanced code generation options.



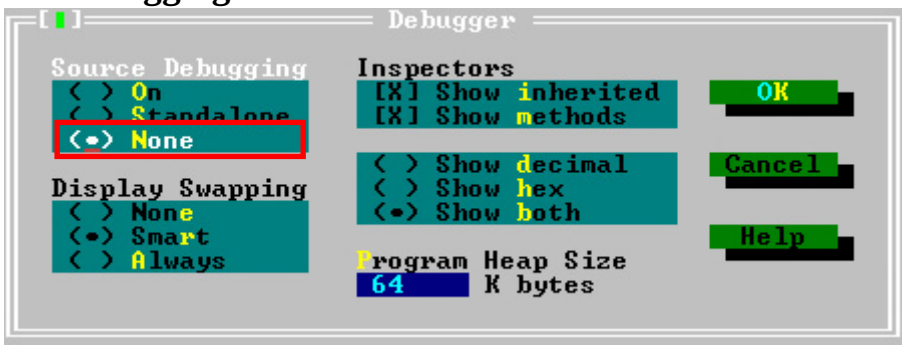
5.1 Set the Floating Point to Emulation and the Instruction Set to 80186.



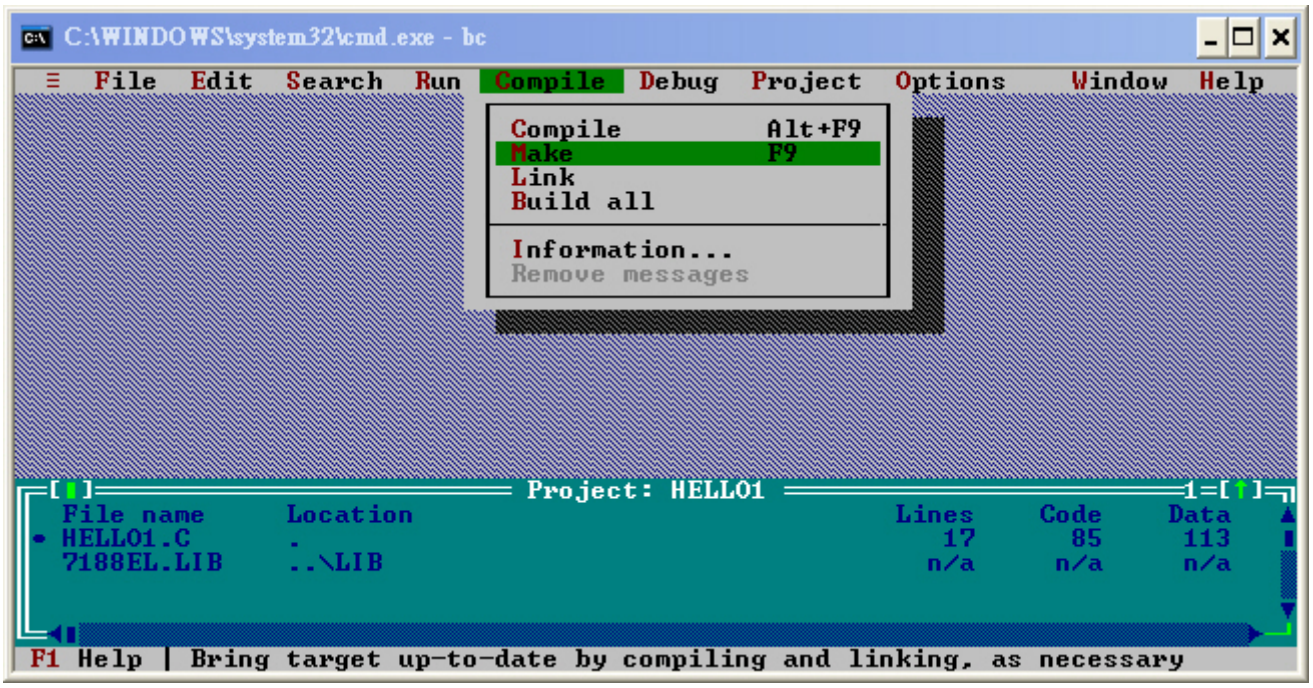
Step 6: Set the Debugger Options.

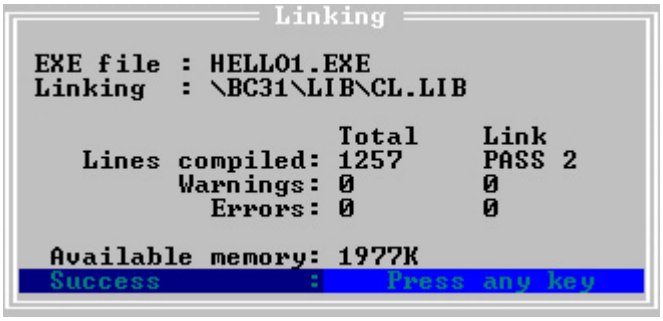


6.1 Set Source Debugging to None.



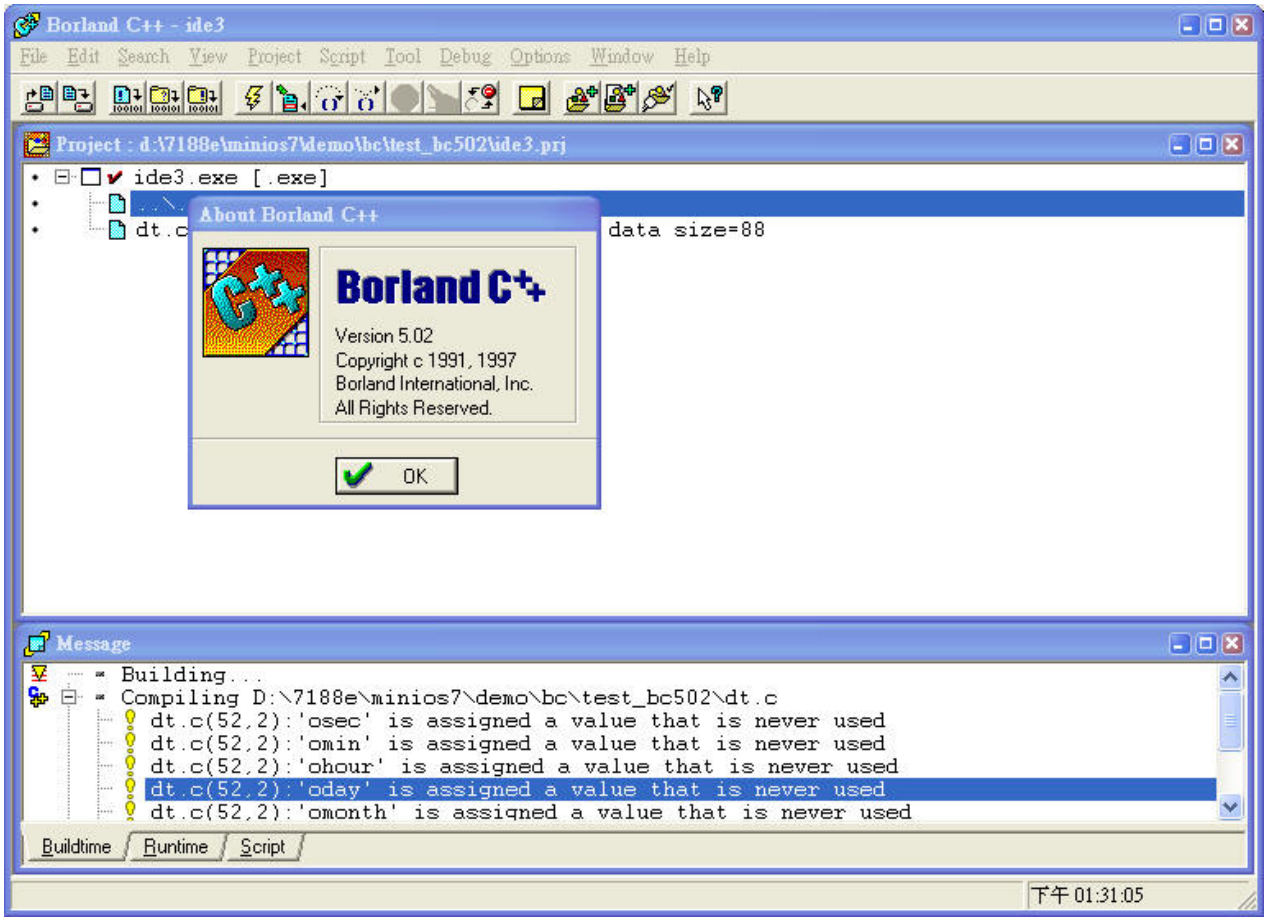
Step 7: Make the project.



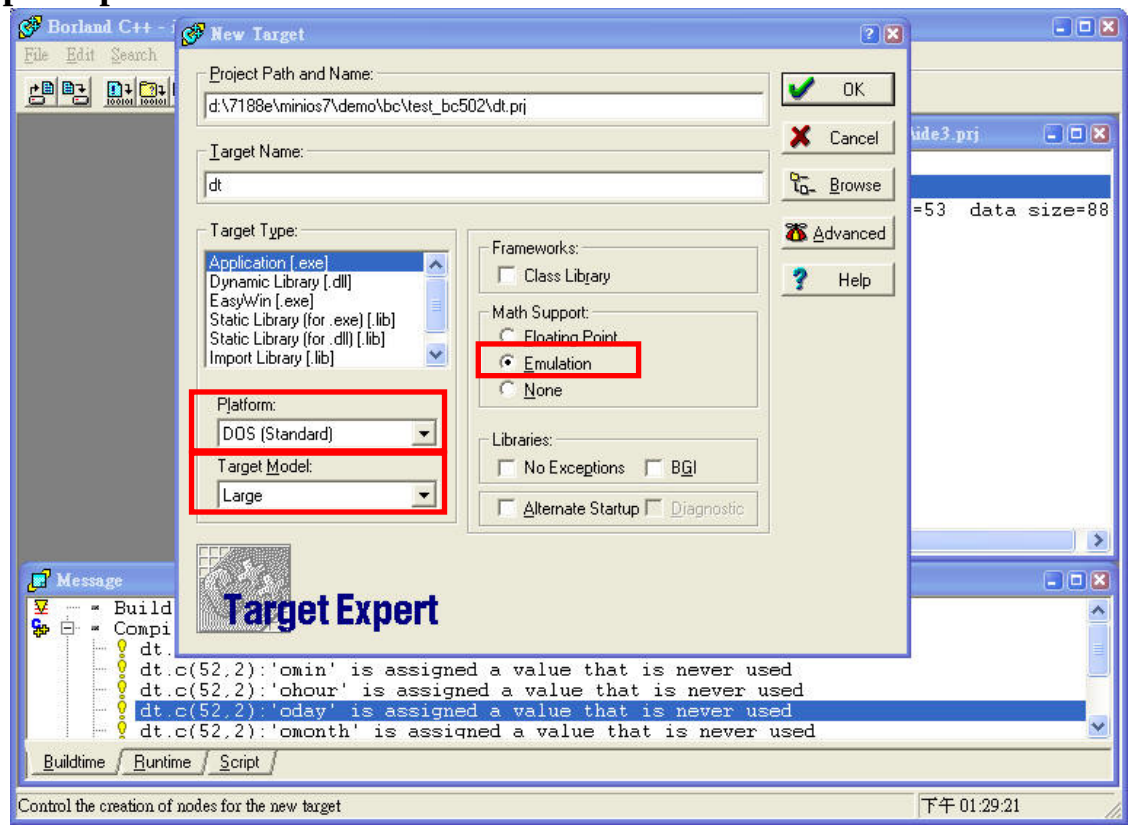


Compiling using the BC++ 5.02 IDE

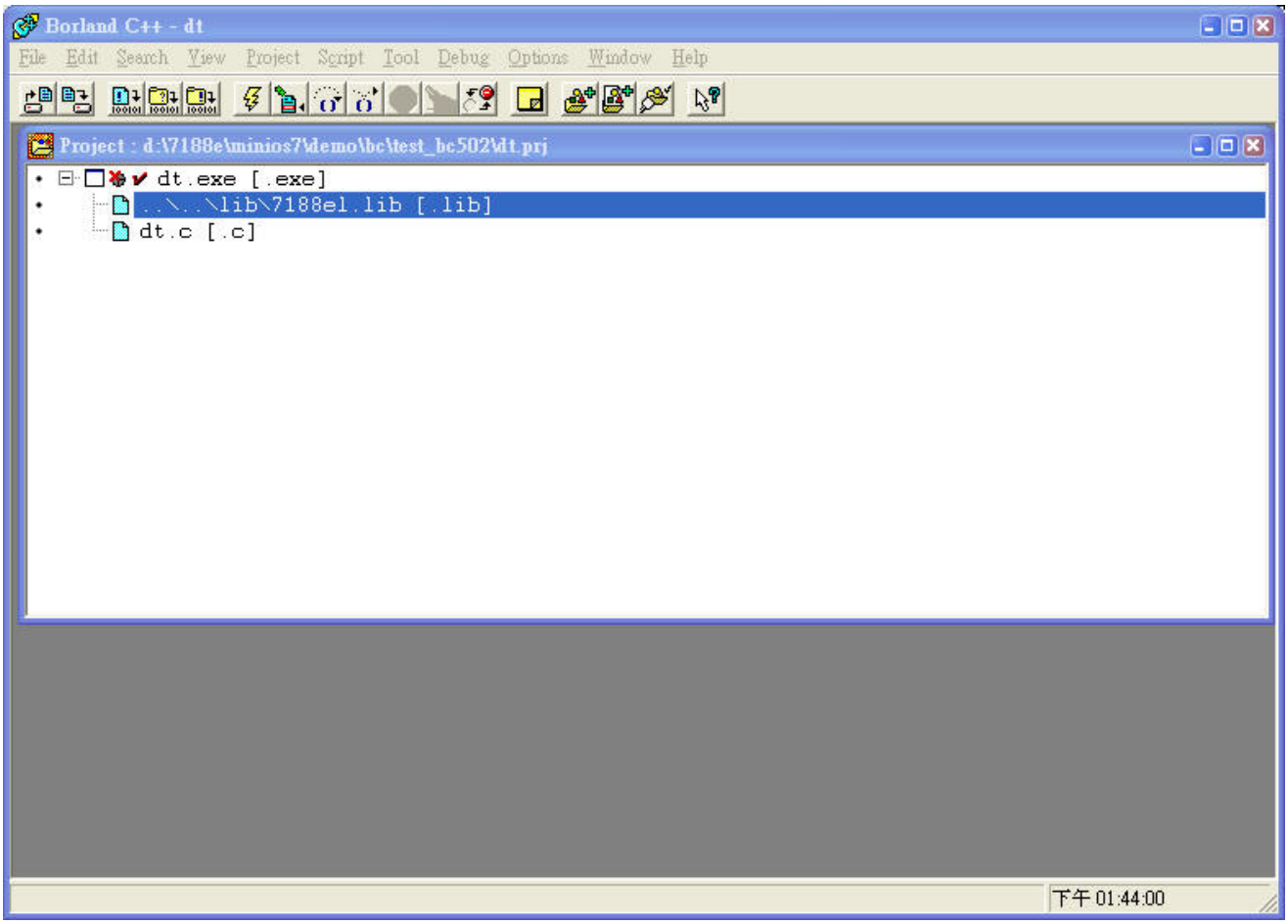
Step 1: Run BC.



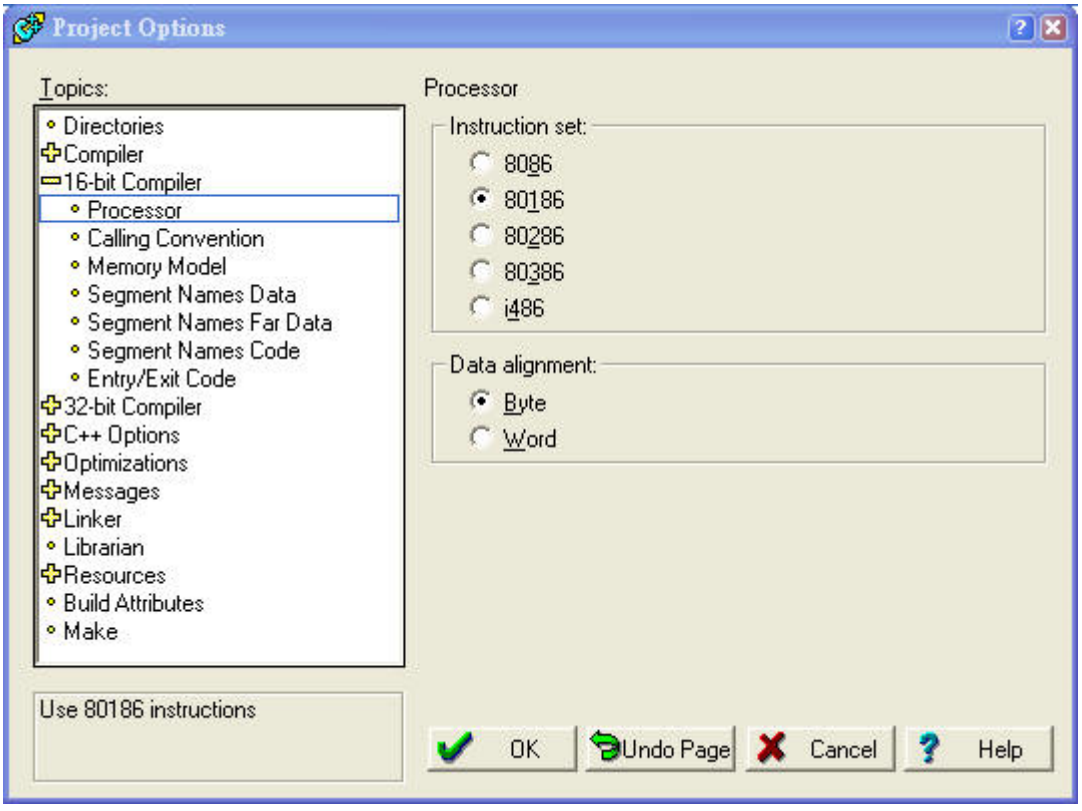
Step 2: Create a new project and set the Target Type, Platform, Target Model and Math Support options.



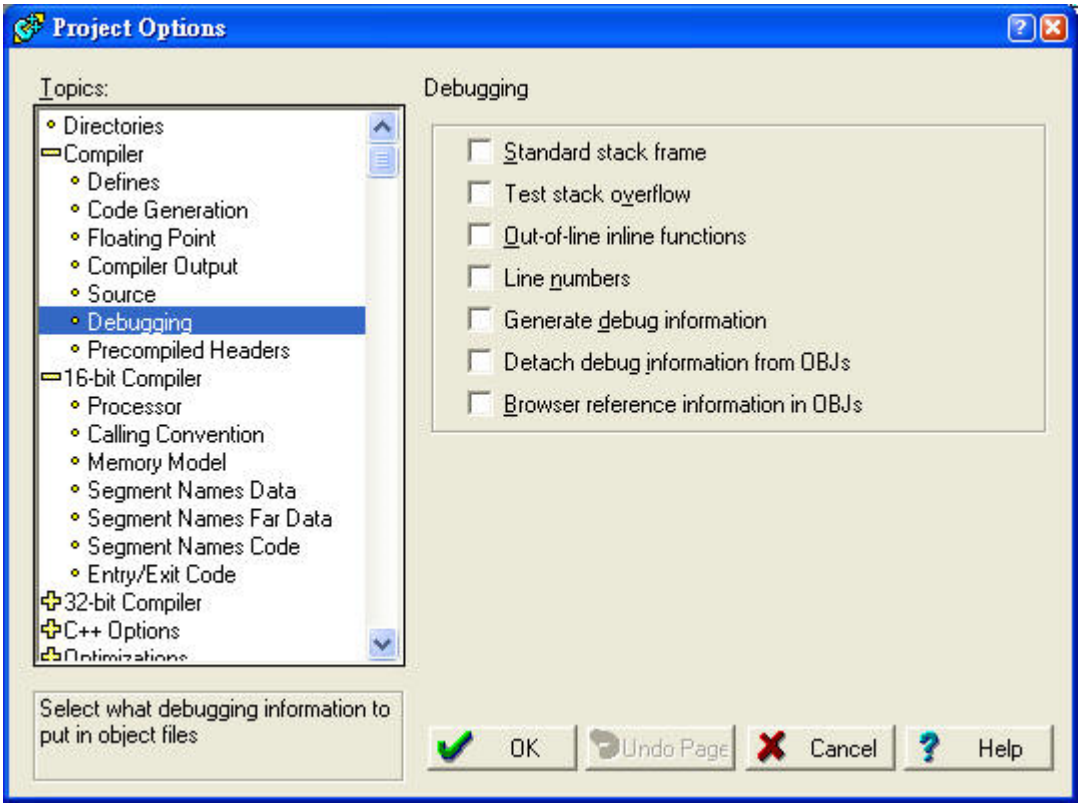
Step 3: Add all necessary files to the project.



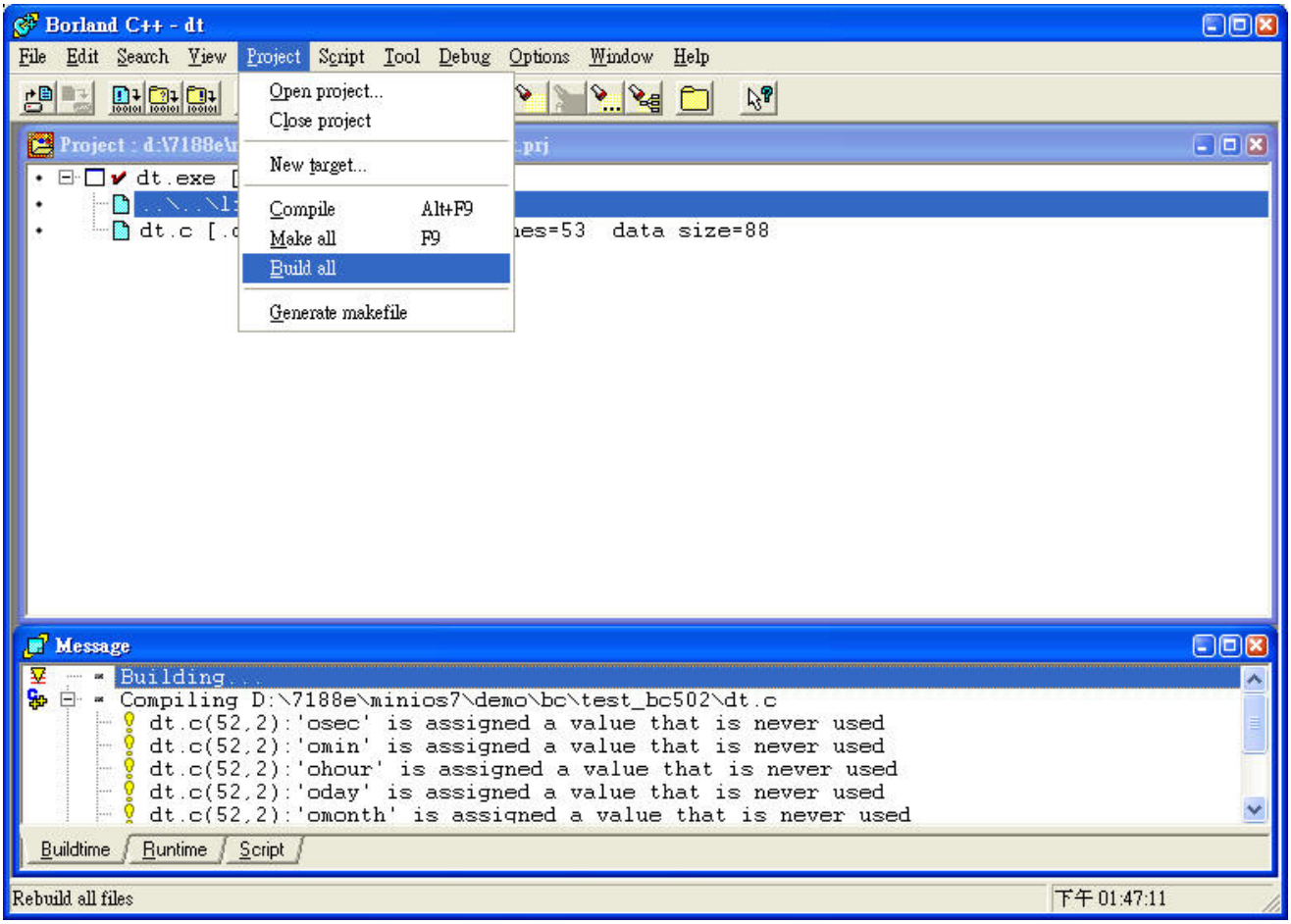
Step 4: Set the Code generation options.

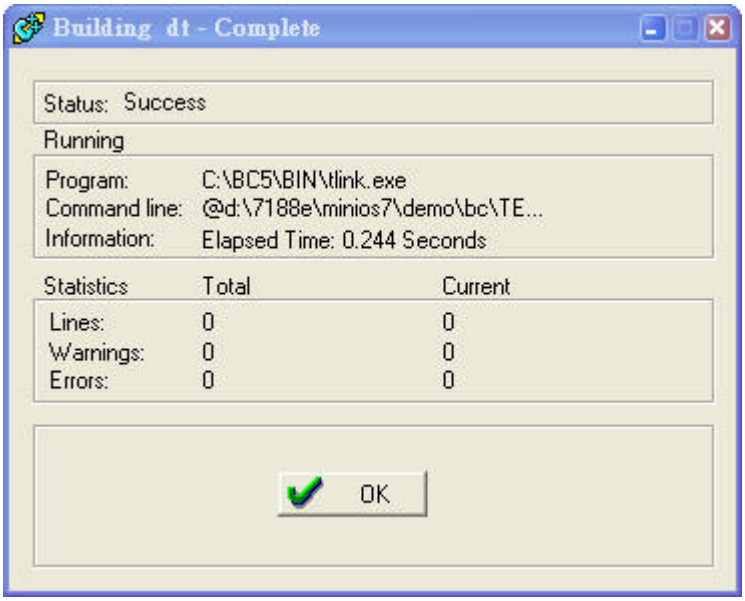


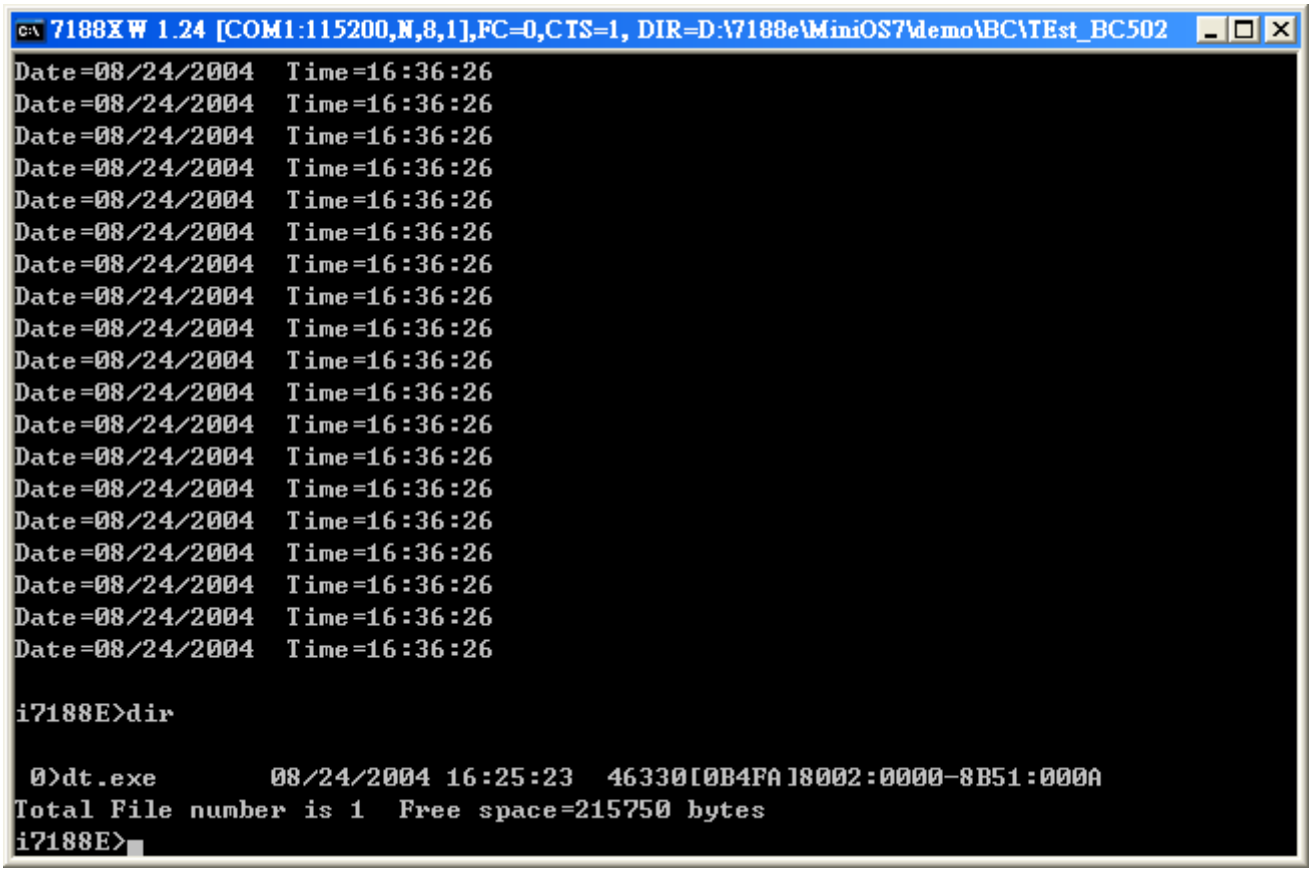
Step 5: Set the Debugger options.



Step 6: Make the project.



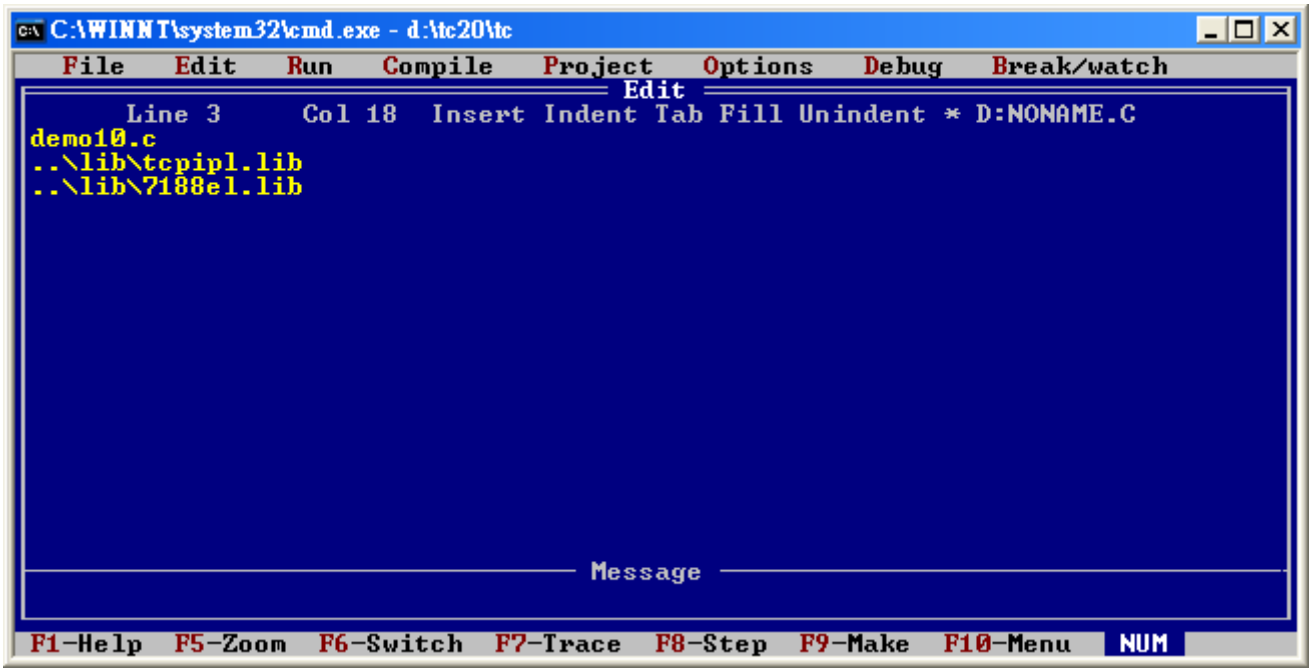

Building dt - Complete
Status: Success
Running
Program: C:\BC5\BIN\tlink.exe
Command line: @d:\7188e\minios7\demo\bc\TE...
Information: Elapsed Time: 0.244 Seconds
Statistics Total Current
Lines: 0 0
Warnings: 0 0
Errors: 0 0
OK


7188XW 1.24 [COM1:115200,N,8,1],FC=0,CTS=1, DIR=D:\7188e\MiniOS7\demo\BC\TEst_BC502
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
Date=08/24/2004 Time=16:36:26
i7188E>dir
0)dt.exe 08/24/2004 16:25:23 46330[0B4FA]8002:0000-8B51:000A
Total File number is 1 Free space=215750 bytes
i7188E>

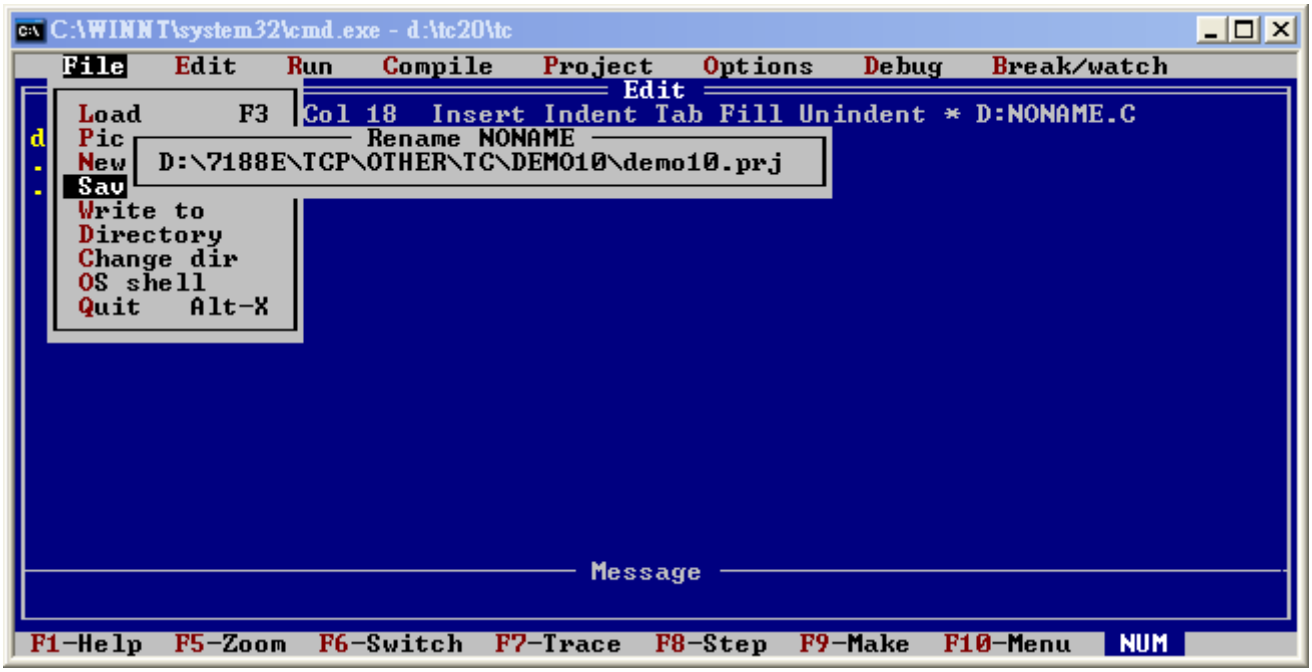
Compiling using the Turbo C 2.01 Compiler

Step 1: Execute TC.EXE to run the TC 2.01 Integrated Environment.

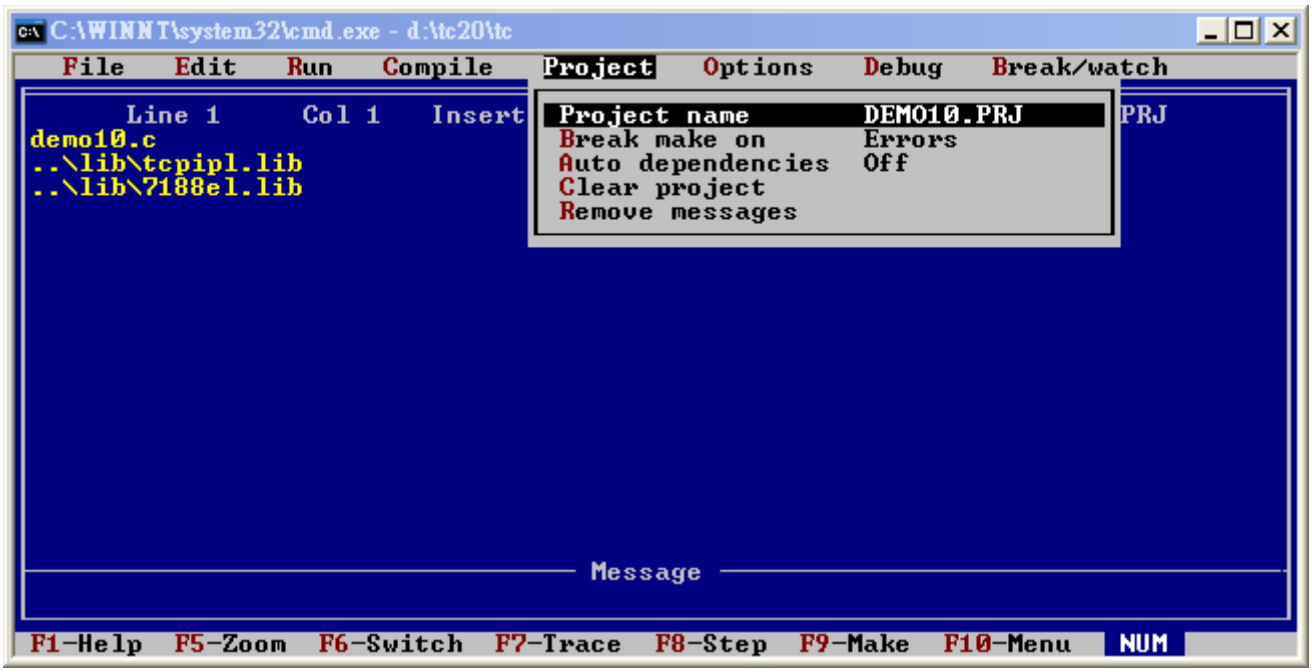
Step 2: Edit the Project file (Add the necessary libraries and files to the project).



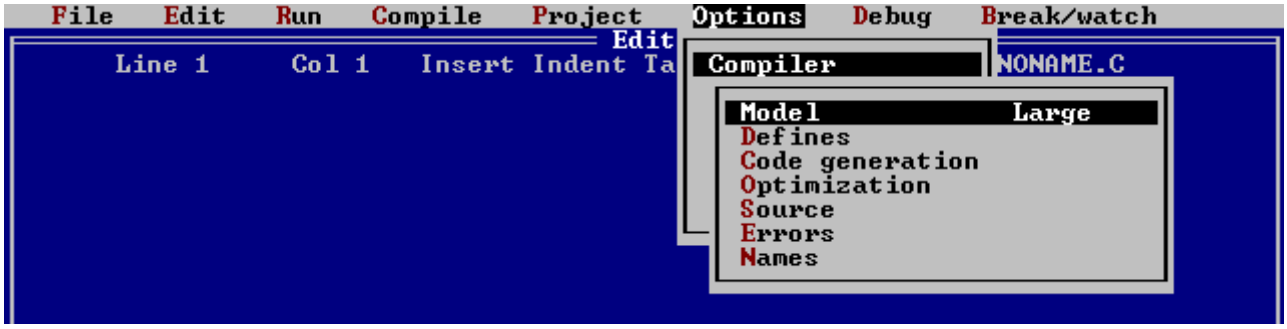
Step 3: Save as a Project file, such as demo10.prj.

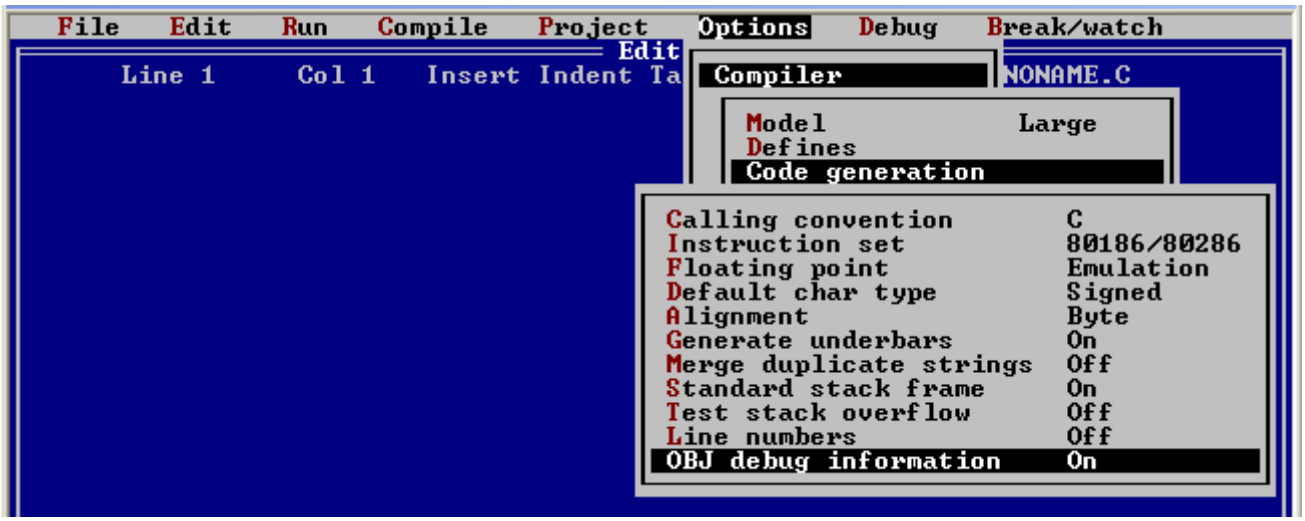


Step 4: Load the Project from the Project menu

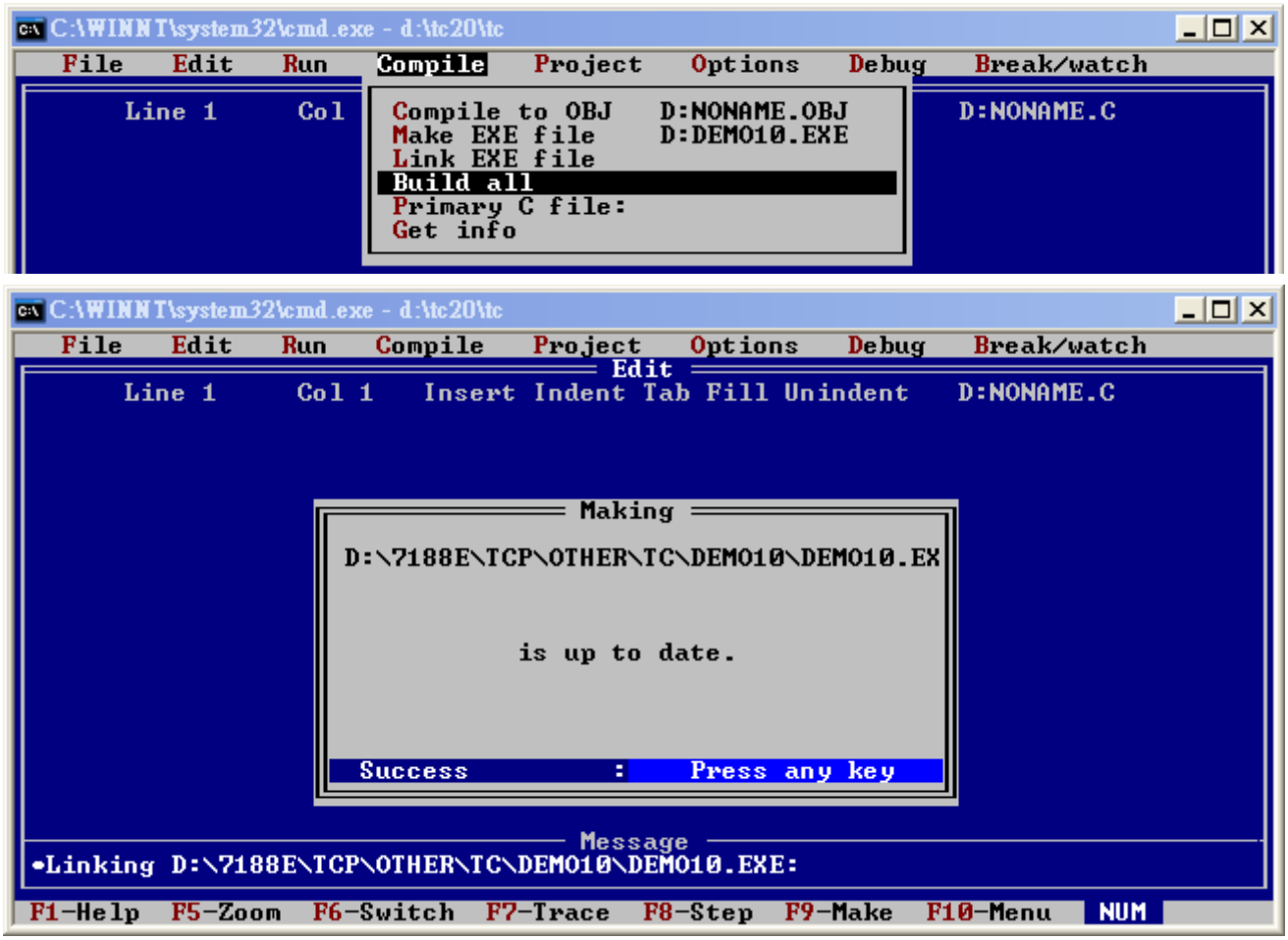


Step 5: Change the Memory model (Small for 8000s.lib/7188es.lib, large for 8000l.lib/7188el.lig) and set the Code Generation to 80186/80286 in the Compiler options.





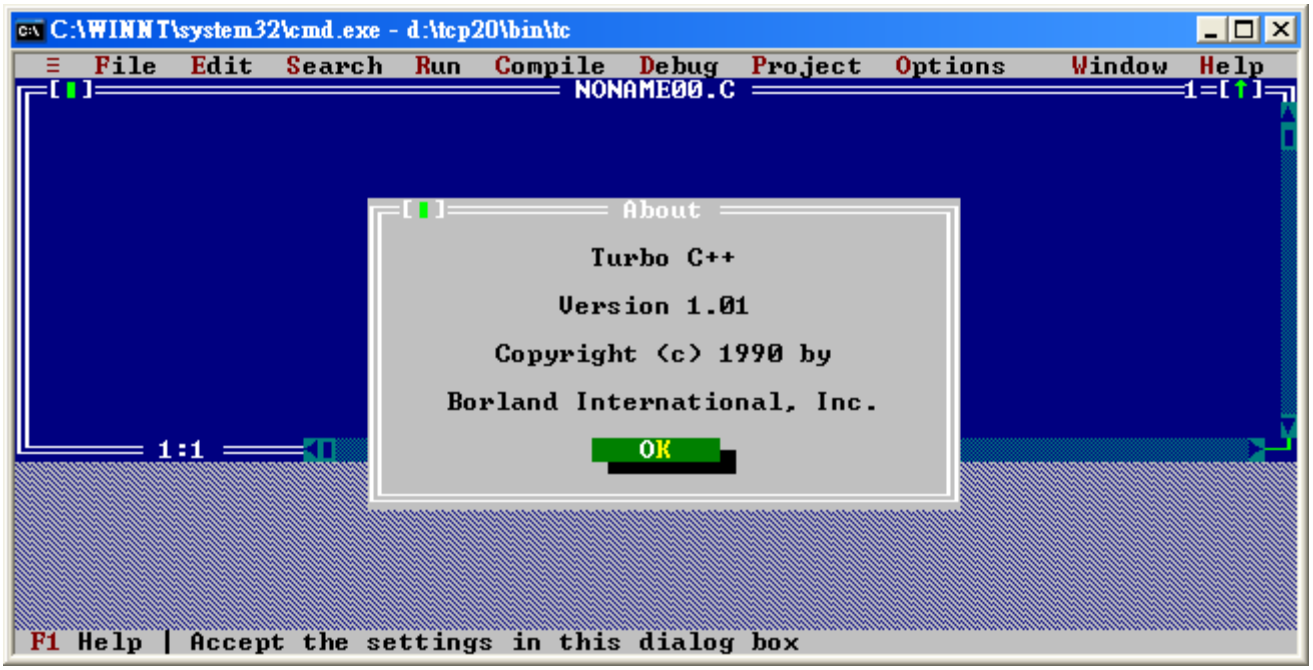
Step 6: Build the project.



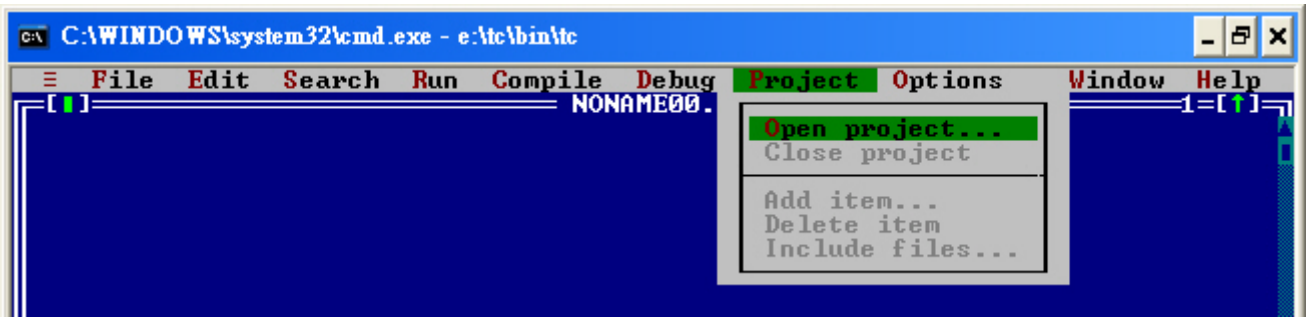
Compiler using the TC++ 1.01 Compiler

***Get free software from http://community.borland.com/museum**

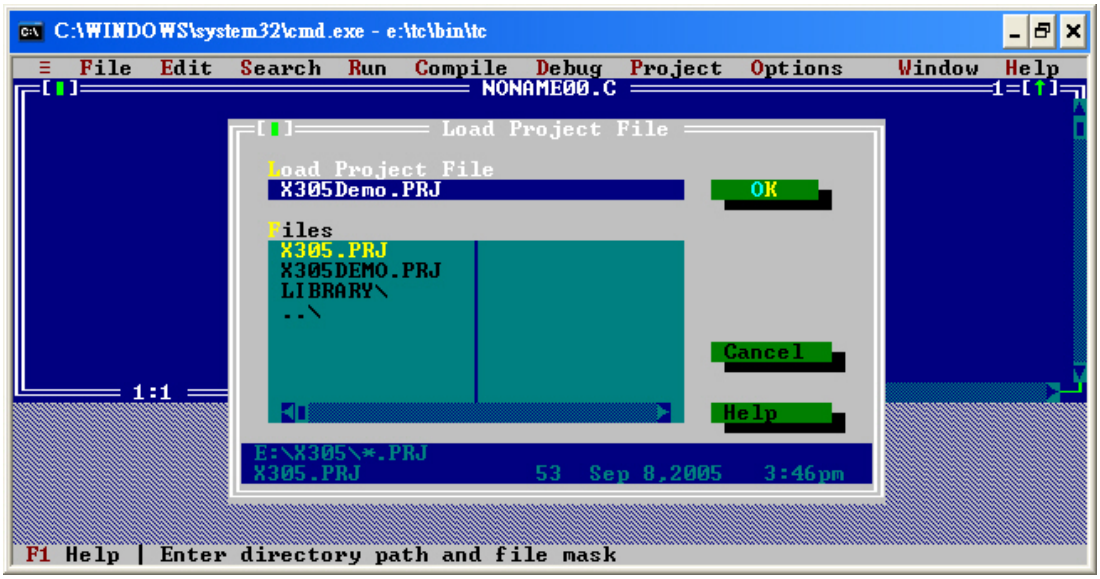
Step 1: Execute \TC\BIN\TC.EXE to run the TC++ version 1.01 Integrated Environment.



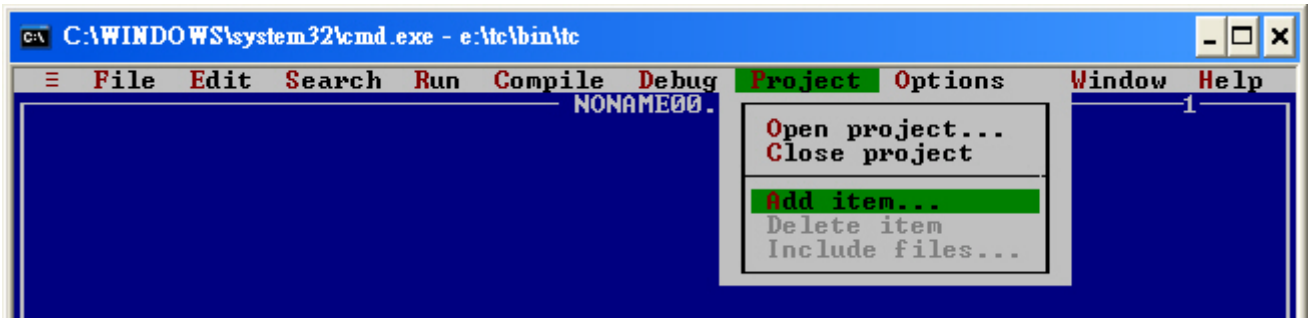
Step 2: Create a new project file (*.prj).



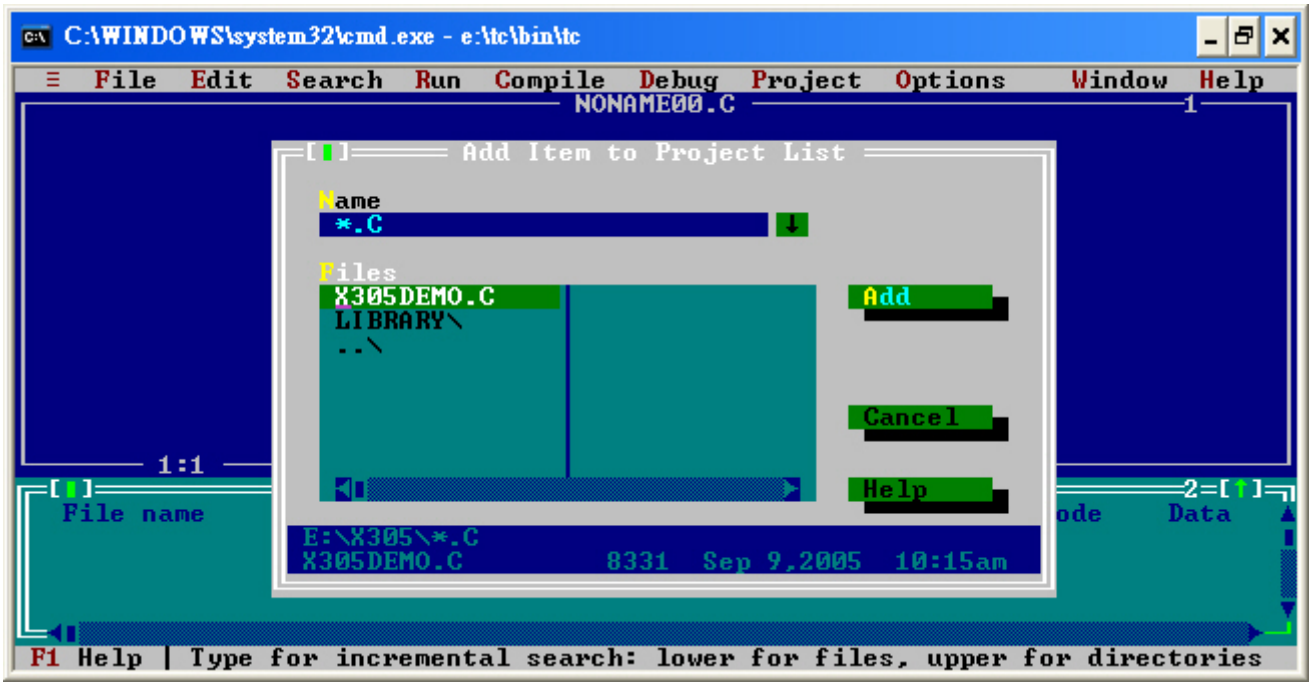
2.1 Type the name of the project file and then click the OK button.



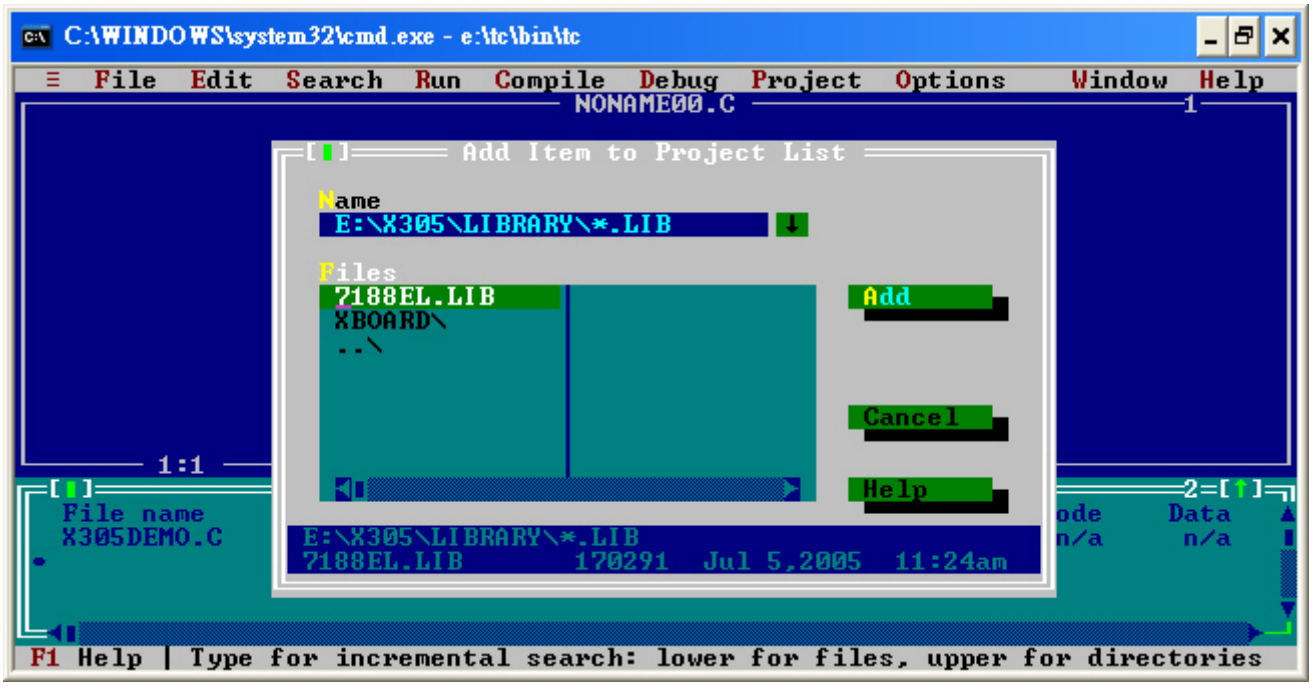
Step 3: Add all necessary files to the project.



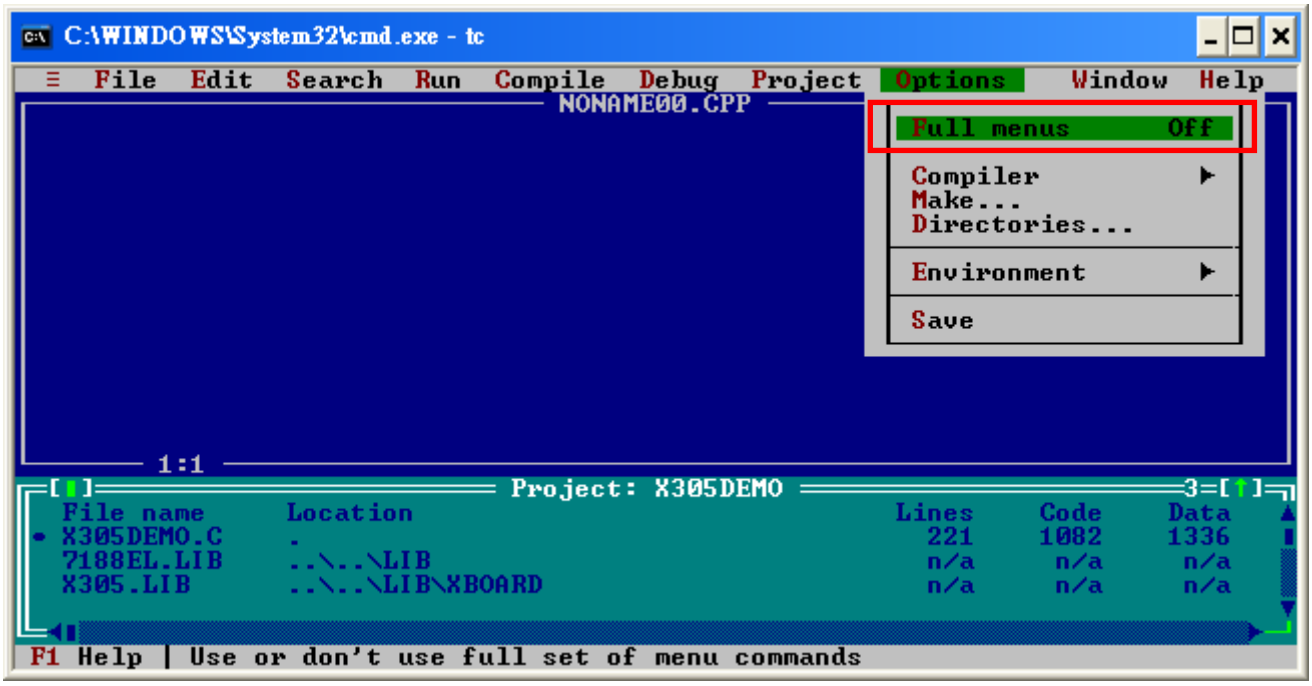
3.1 Select the source file and then click the Add button.



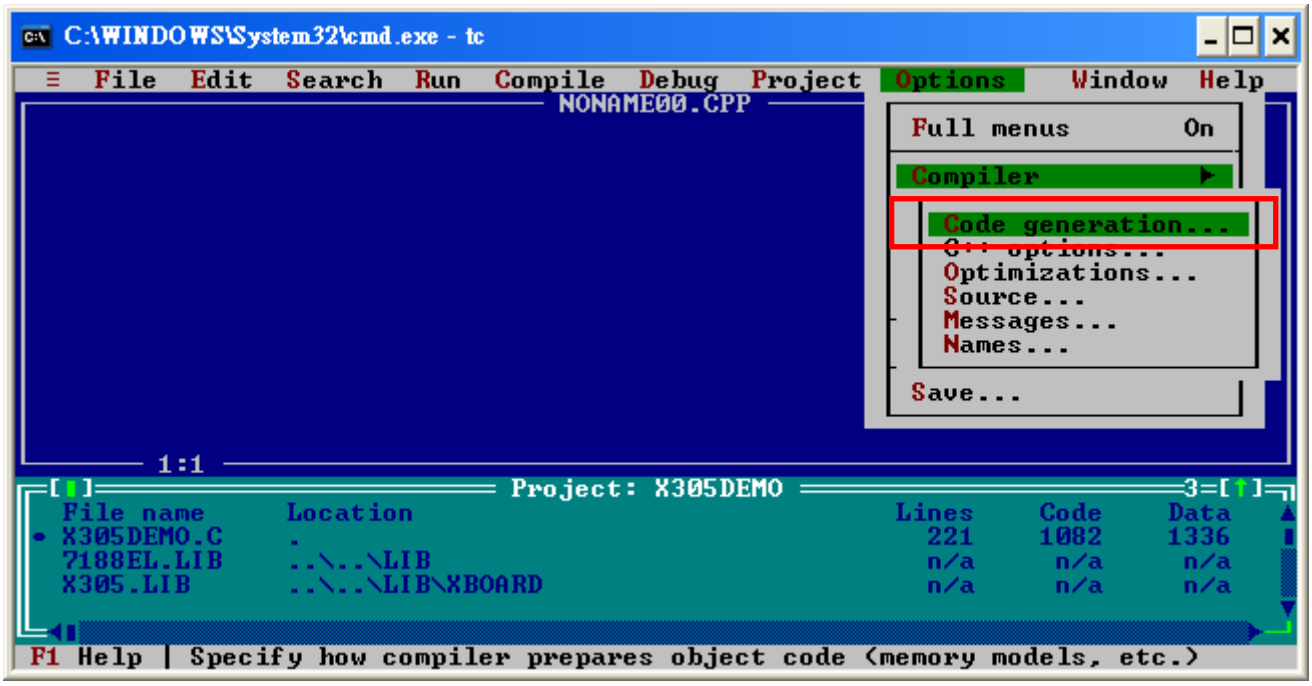
3.2 Select the function library and then click the Add button.



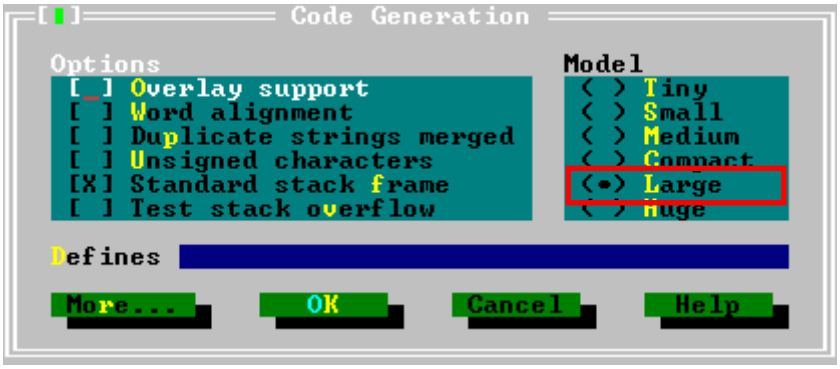
Step 4: Show full menus.



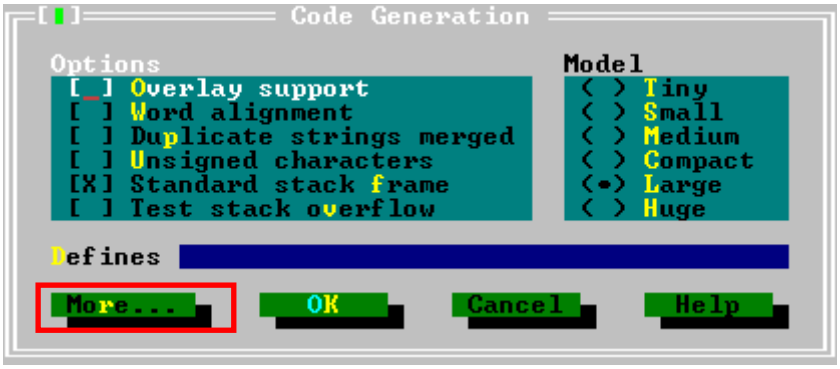
Step 5: Set Code generation options.

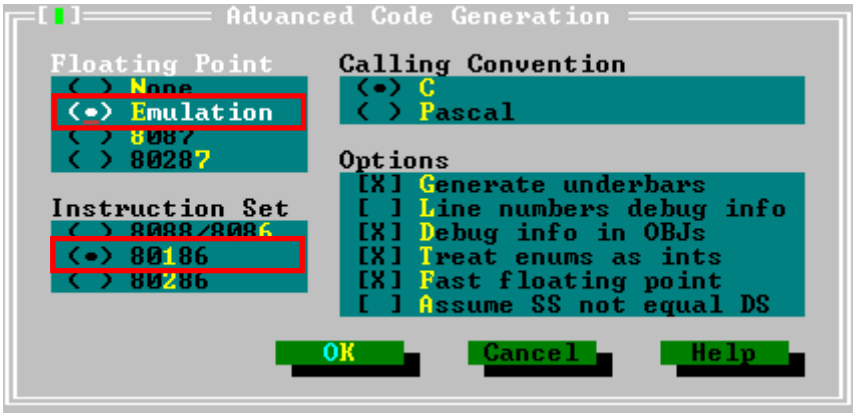


5.1 Change the Memory model (Small for 8000s.lib/7188es.lib, large for 8000l.lib/7188el.lib).

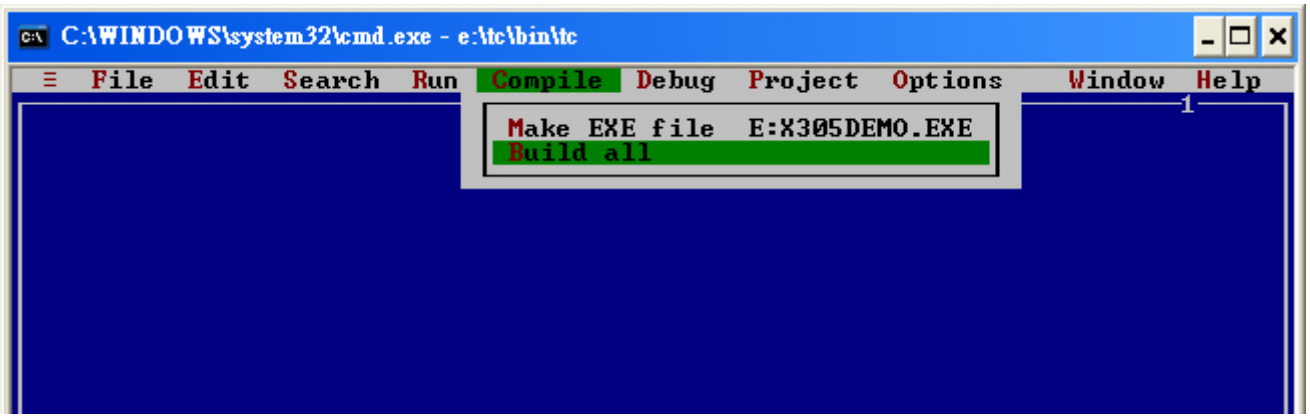


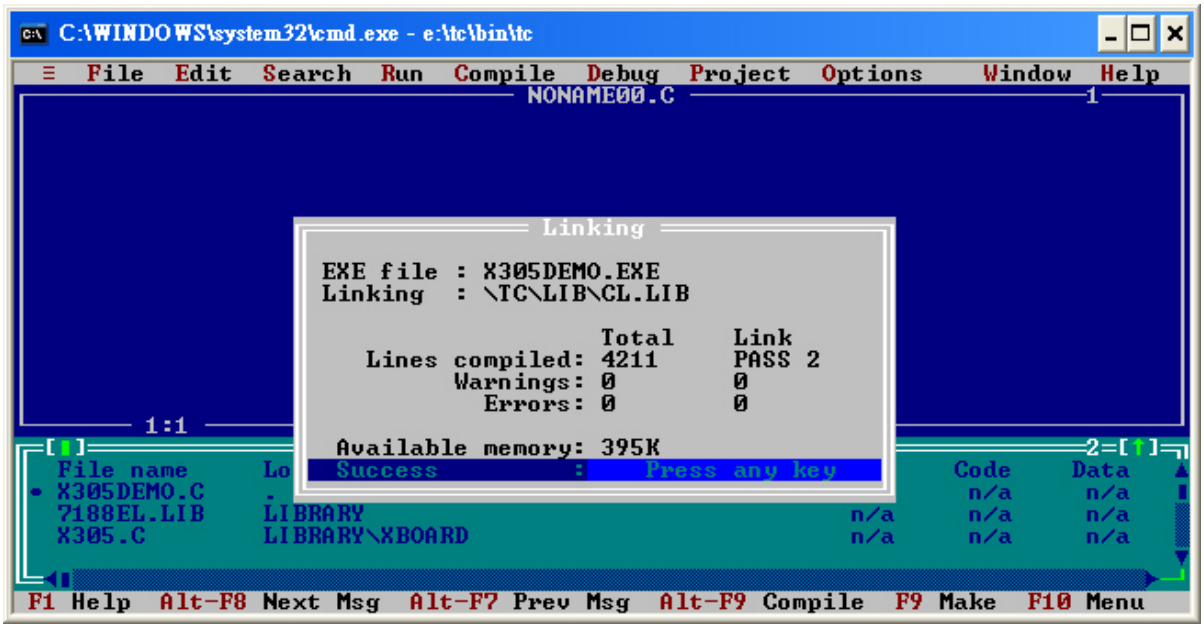
4.2 Set the Floating Point to Emulation and the Instruction Set to 80186.





Step 5: Make the project.



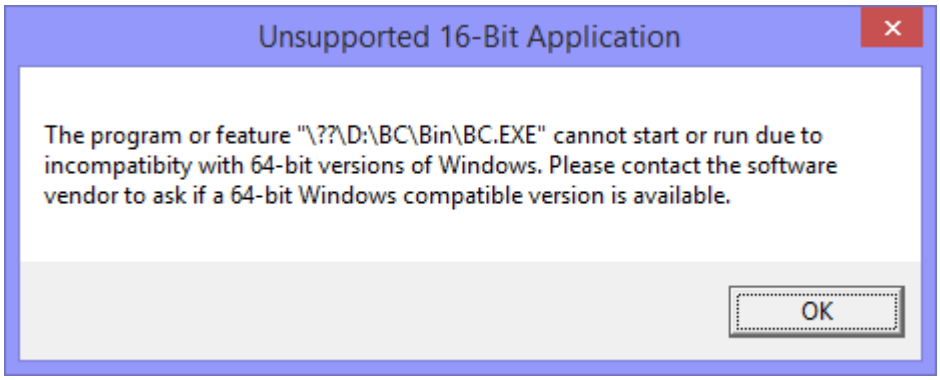


Compiling on 64-bit platform like Windows 7 or Windows 8 / 8.1

Preface:

If we try to use 16-bit compiler like BC3.1 or TC++3.0 to build MiniOS7 project on Windows 64-bit platform like Windows 7 or Windows 8, it will prompt up an error message to show compatibility problem for 64-bit platform.

"**Application can't run on your PC. Contact with your software publisher**" Or like below snap shot.

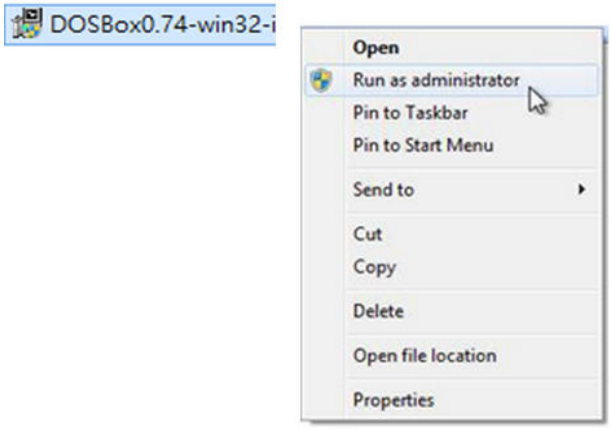


Please follow below steps to solve the problem.

Step 1: download and install DosBox for Windows. You can easily install the 32 bit version on a 64-bit OS. The download page is here:

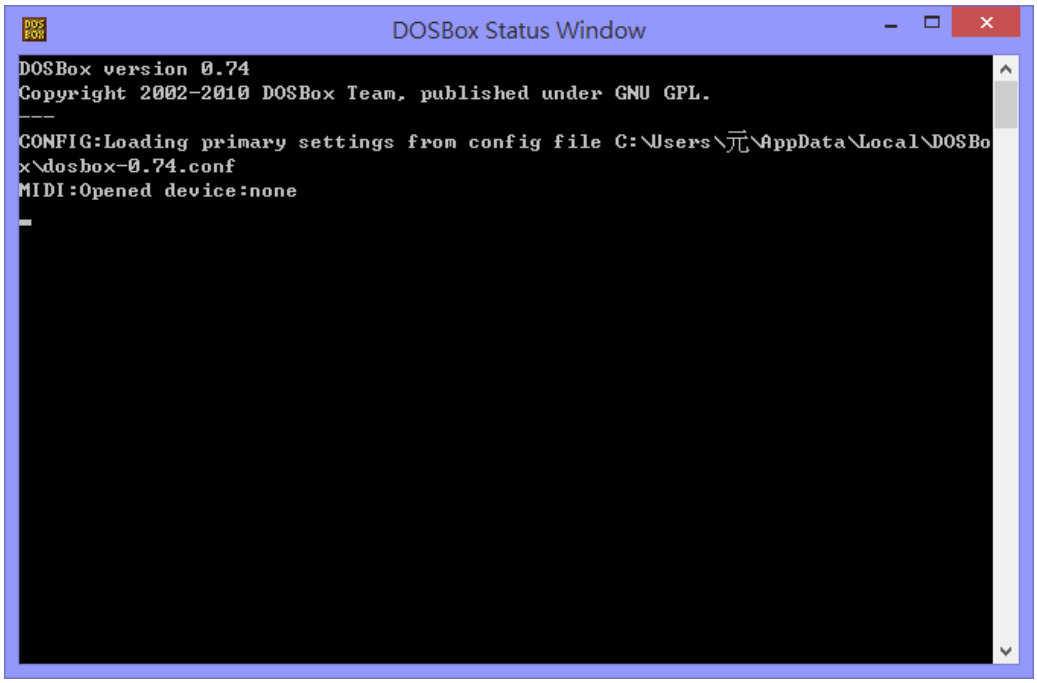
http://sourceforge.net/projects/dosbox/files/dosbox/0.74/DOSBox0.74-win32-installer.exe/download

Note: Strongly recommend installing DOSBox as administrator.

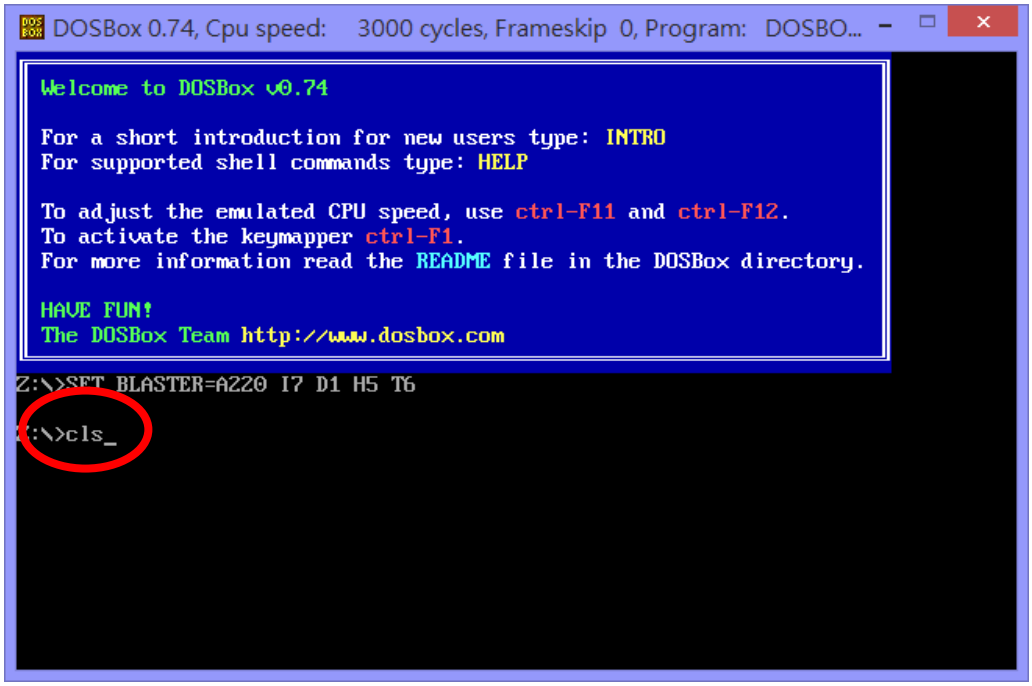


Step 2: Start up the DOSBox, click DOSBox 

, it will bring out two Windows forms, one is DOSBox Status Window and DOSBox Console as below.

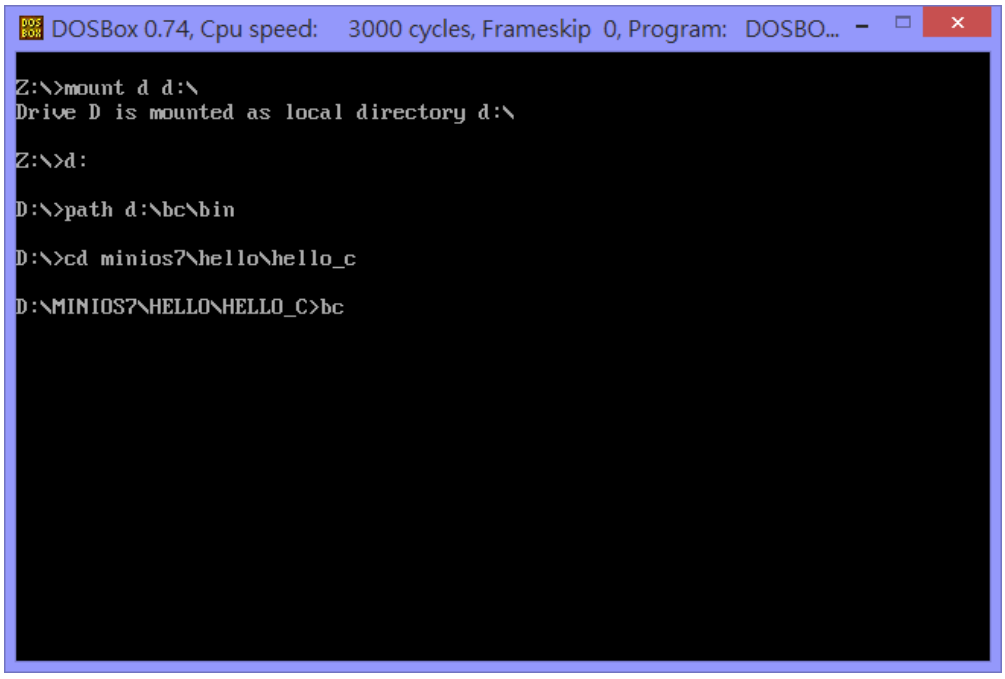


First use command “CLS” or “cls” to clear the screen.

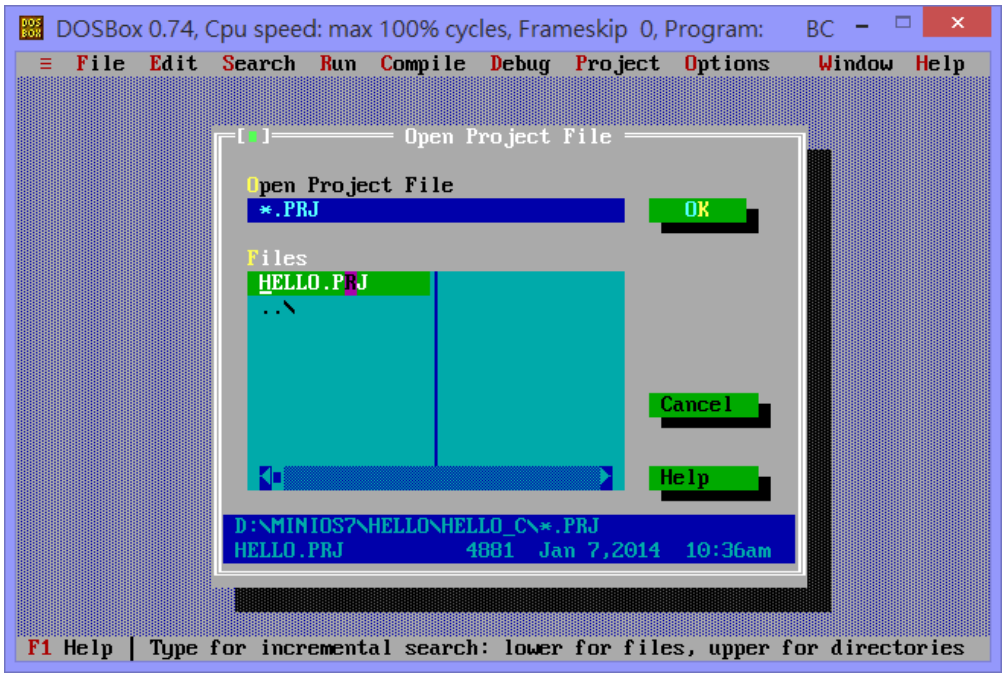


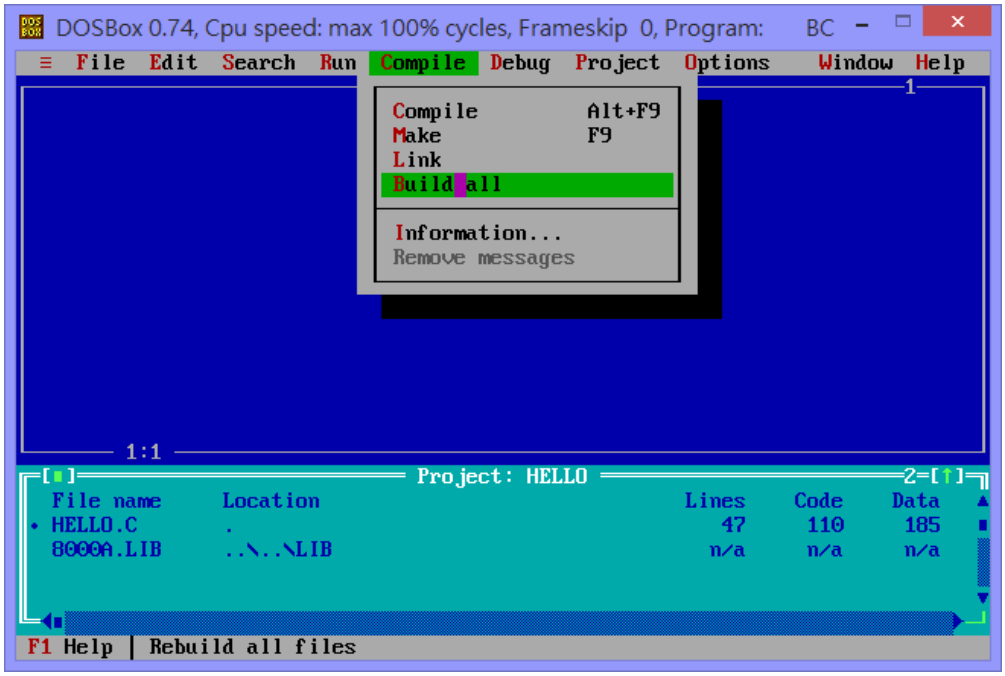
Step 3: Configure the DOSBox Environments settings.

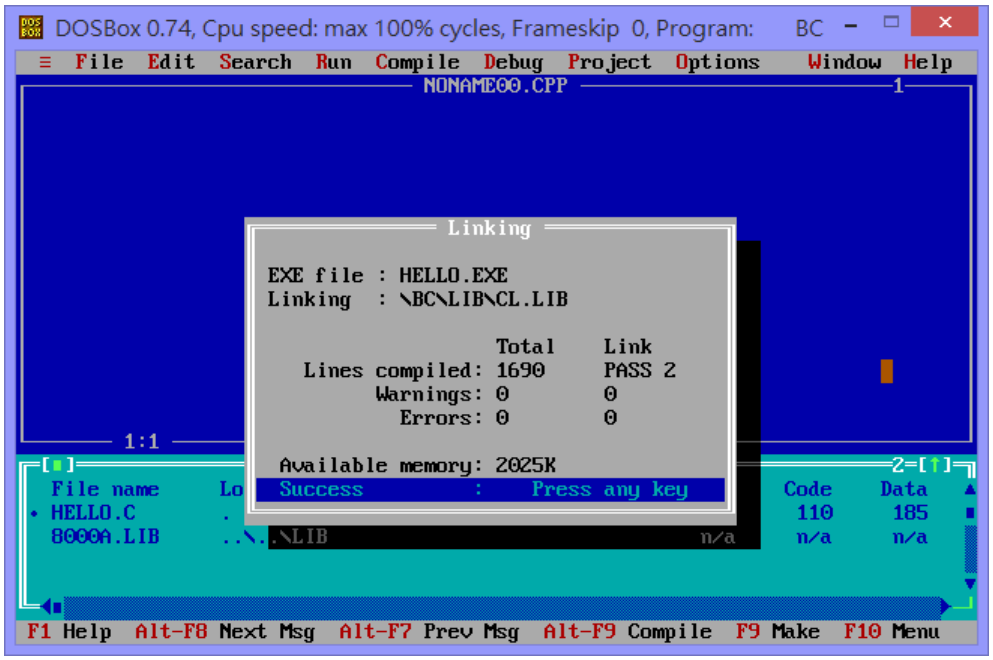
1. Make sure the demo file path, for example D:\MiniOS7\hello\hello_c
2. Make sure the BC\Bin file path, for example D:\BC\Bin
3. Use mount command to mount the disk driver for example “mount d d:\”
4. After mount disk driver D, then type “D:” to change “Z:\>” to “D:\>”.
5. Set Path to BC\Bin for example “path d:\bc\bin”
6. Change directory to demo run “bc”

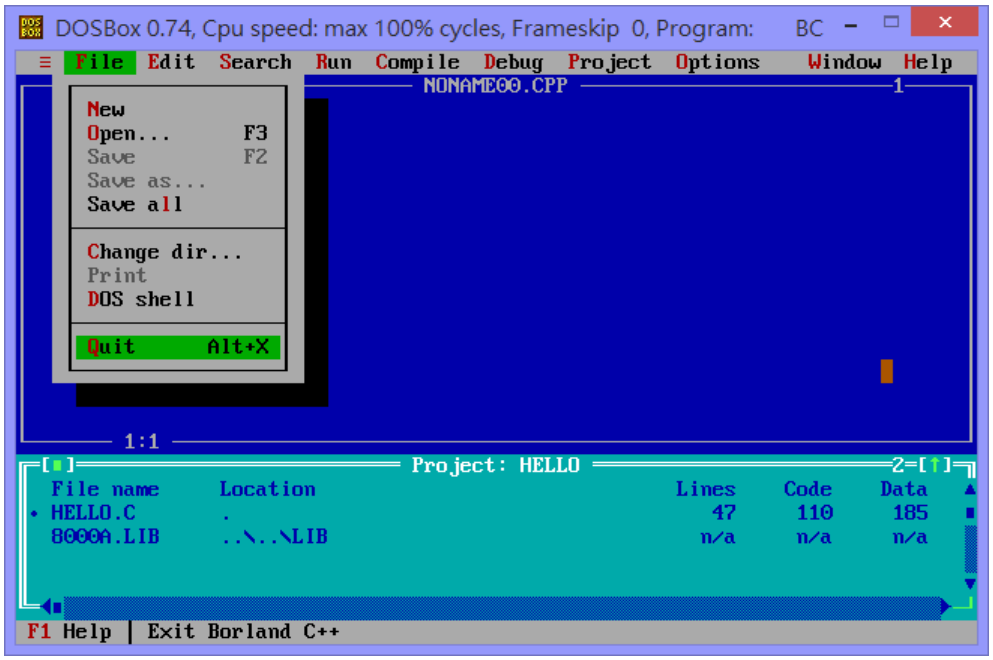


Step 4: Press “Project” to open project➔ Press “Compile” to build the project.






DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: BC
File Edit Search Run Compile Debug Project Options Window Help
NONAME00.CPP
Linking
EXE file : HELLO.EXE
Linking : \BC\LIB\CL.LIB
Total Link
Lines compiled: 1690 PASS 2
Warnings: 0 0
Errors: 0 0
Available memory: 2025K
Success : Press any key
File name Lo Code Data
HELLO.C . 110 185
8000A.LIB ..\..\LIB n/a n/a
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

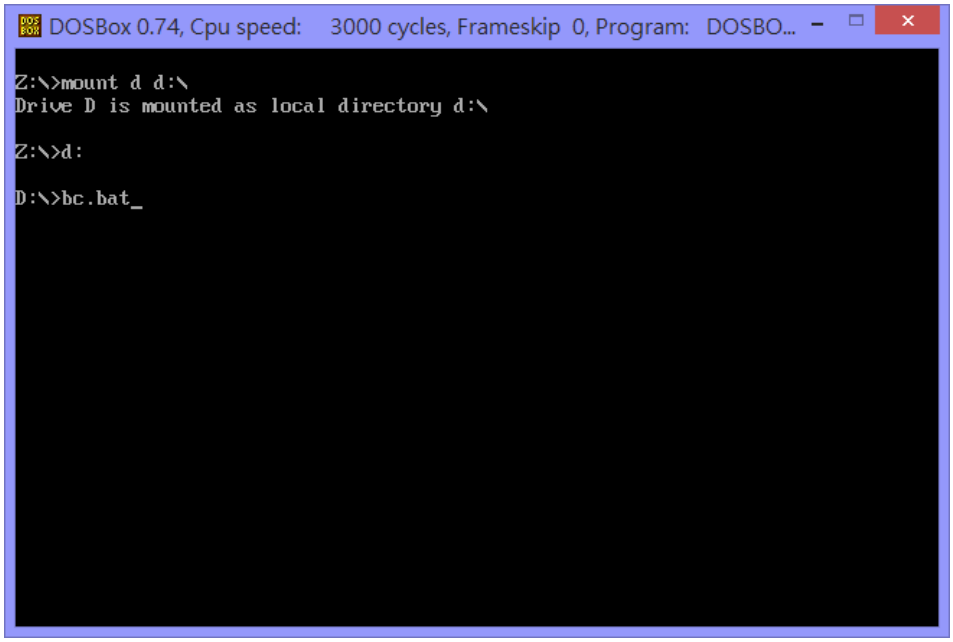

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: BC
File Edit Search Run Compile Debug Project Options Window Help
NONAME00.CPP
New
Open... F3
Save F2
Save as...
Save all
Change dir...
Print
DOS shell
Quit Alt+X
Project: HELLO
File name Location Lines Code Data
HELLO.C . 47 110 185
8000A.LIB ..\..\LIB n/a n/a n/a
F1 Help | Exit Borland C++

Tips for using batch file:

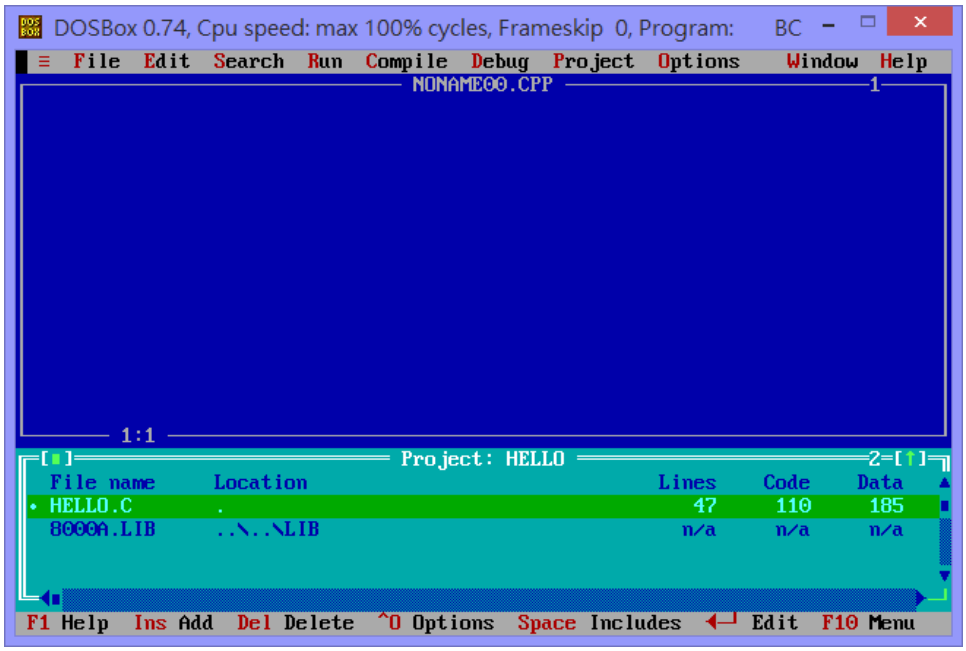
It is a little troublesome to set the path and launch the BC demo project, We can make a batch file as below for demo program,

```
path d:\bc\bin
cd minios7\hello\hello_c
bc
```

and save this batch file to path “D:\” , when Driver D is mounted as local directory and change to d: ,



After execute the bc.bat file, the demo project will be launched successfully.

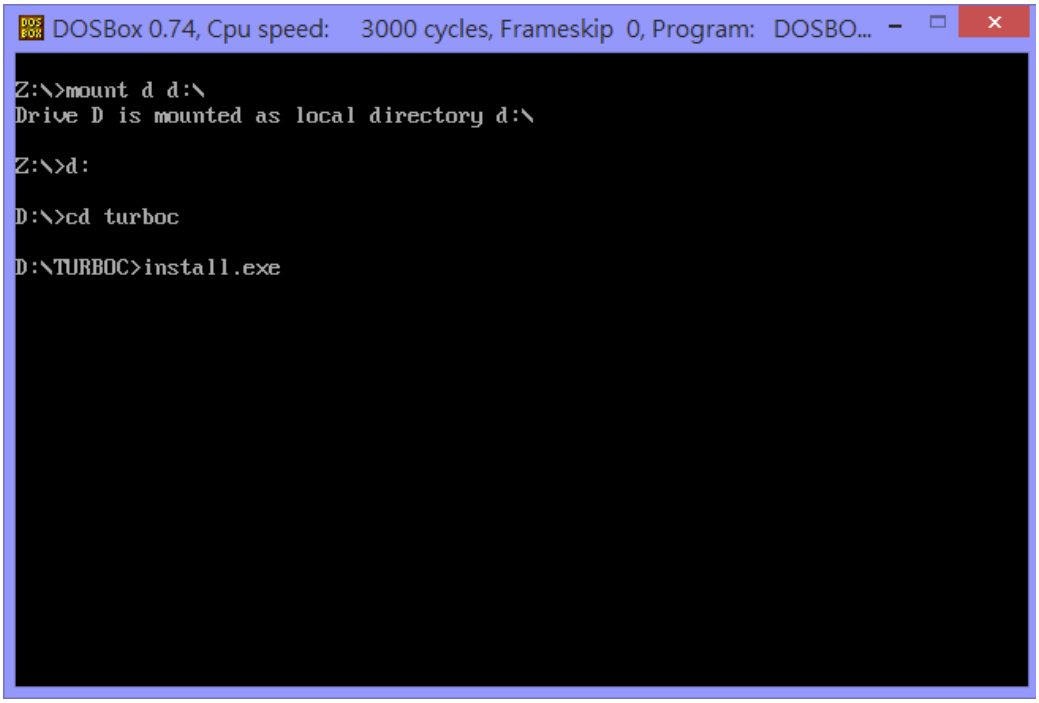


If BC3.1 compiler is not available, it can download Free TurboC++3.0 from website

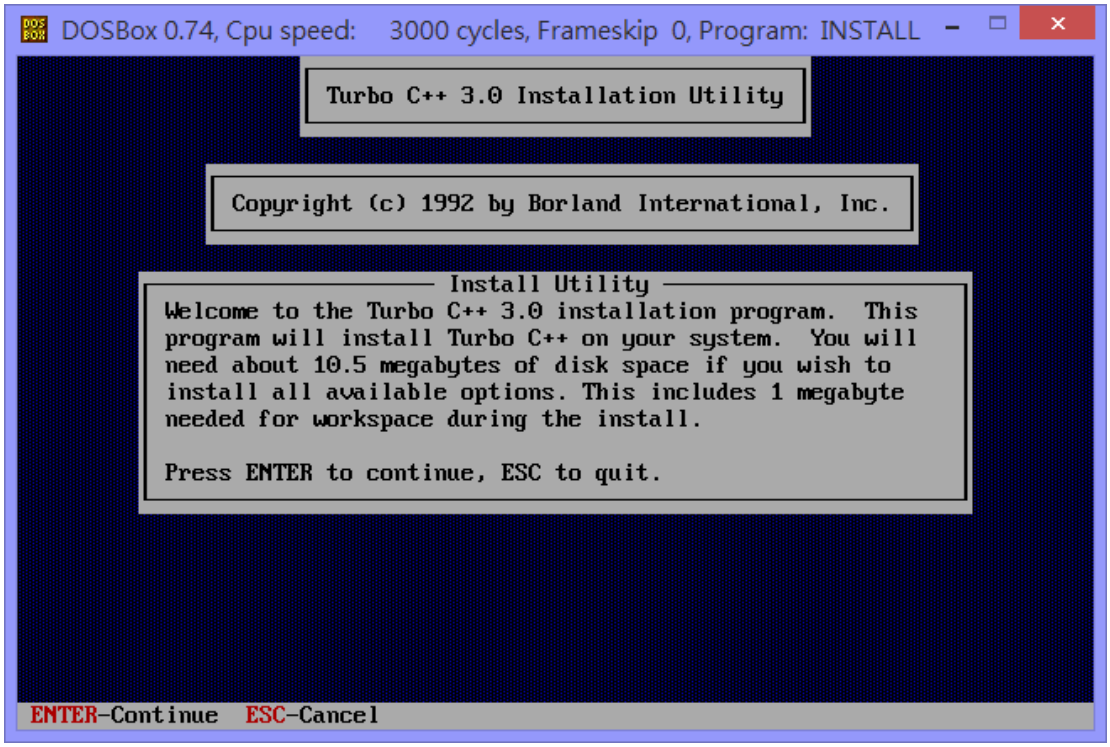
Download Free **turboc.zip** from following link

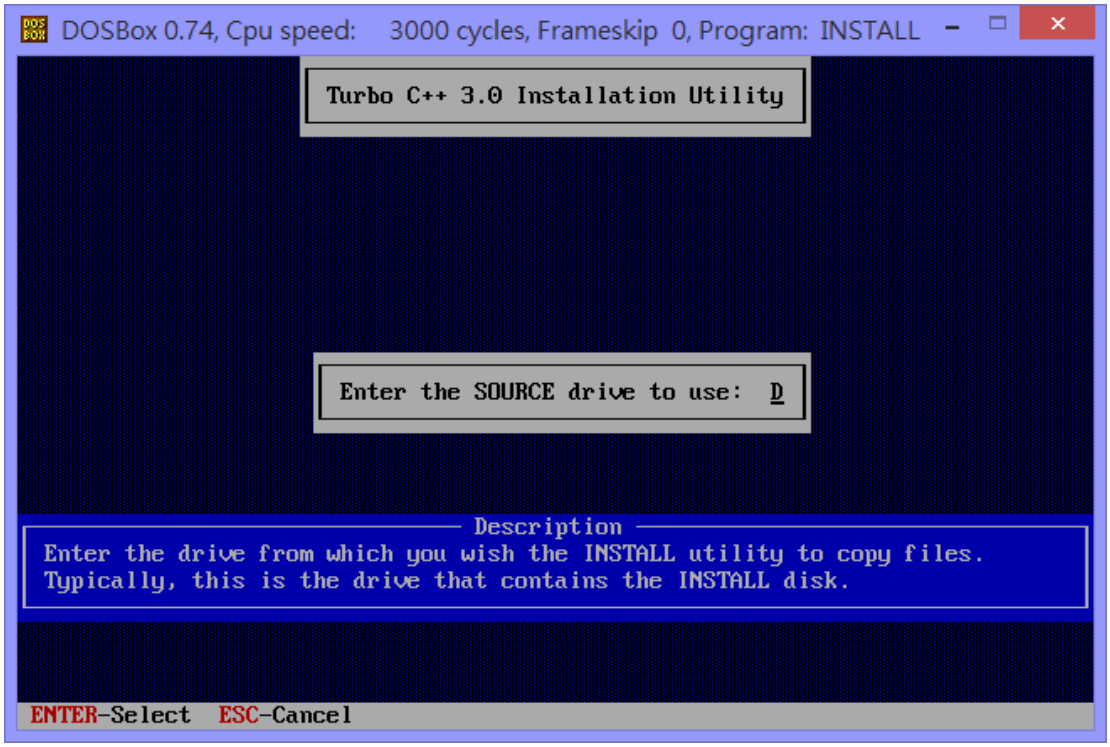
http://www.bestfreewaredownload.com/download/t-free-turbo-c--freeware-flggsdpz.html

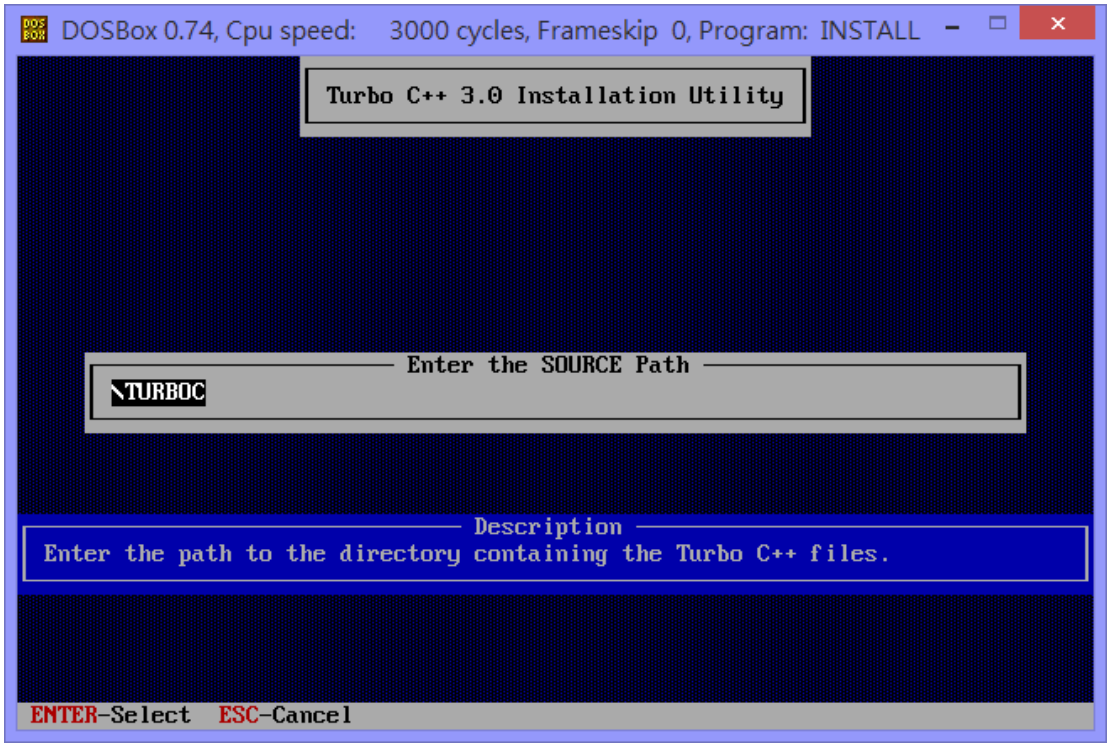
After unzip the turboc.zip and copy the turboc folder to D:\, then start up the DOSBox, and run the install.exe to install TC++3.0

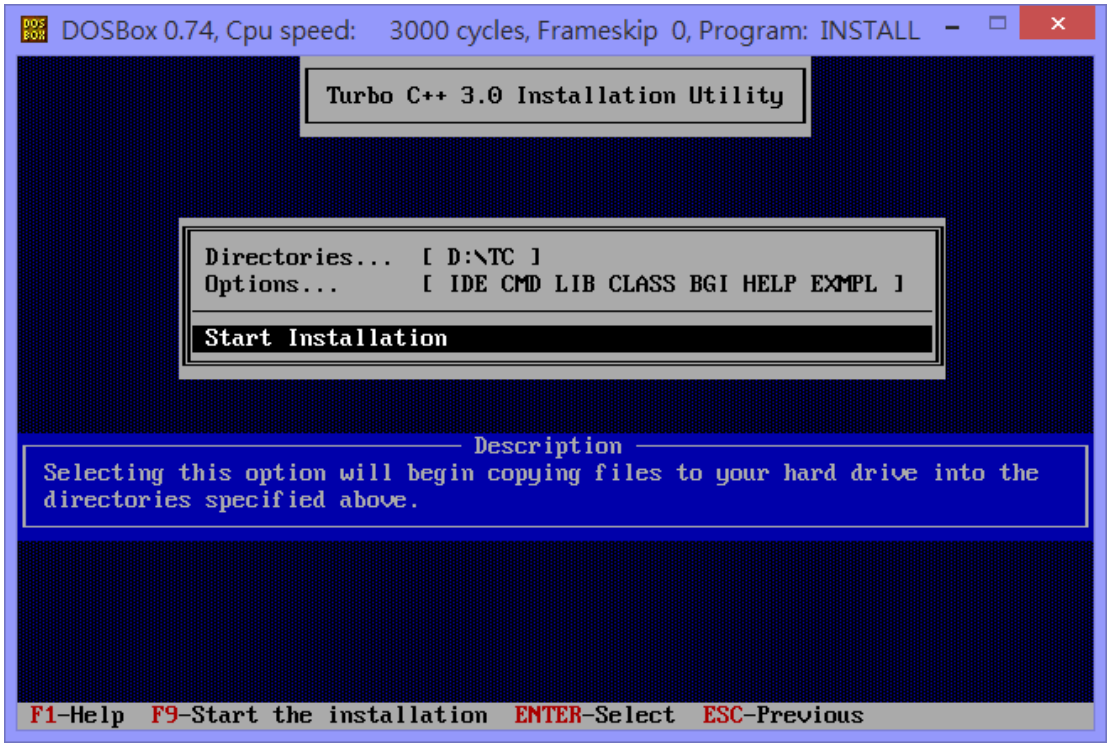


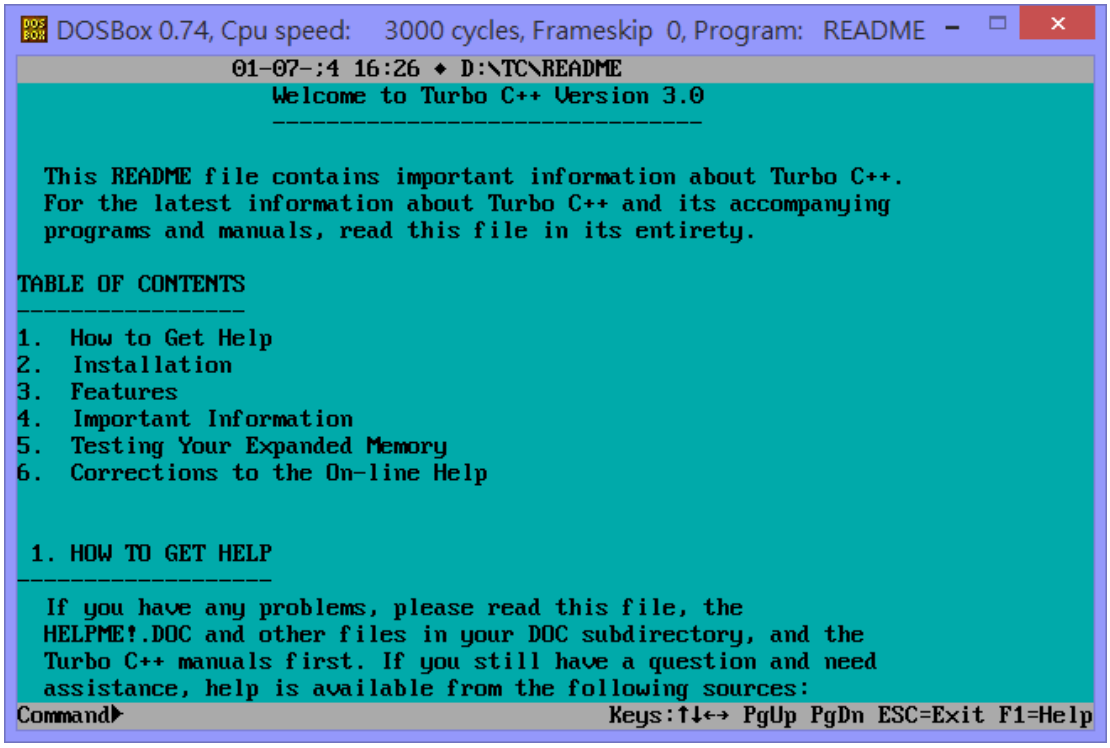
Below snap shots are steps for install TC++ 3.0



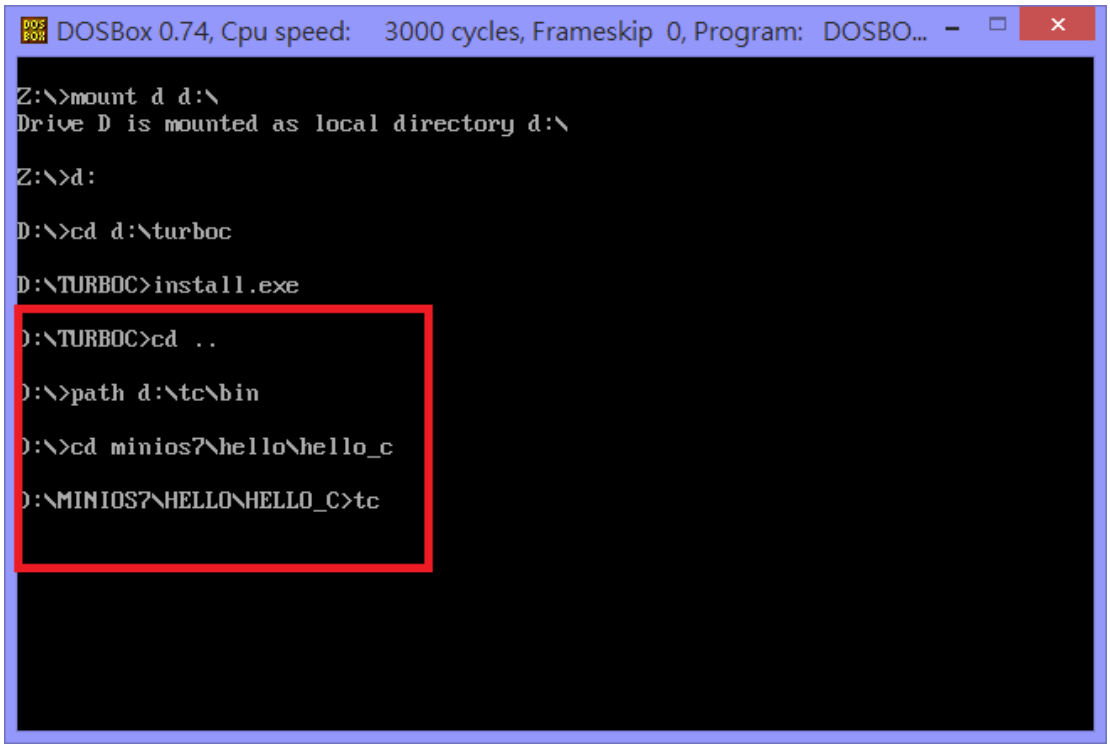


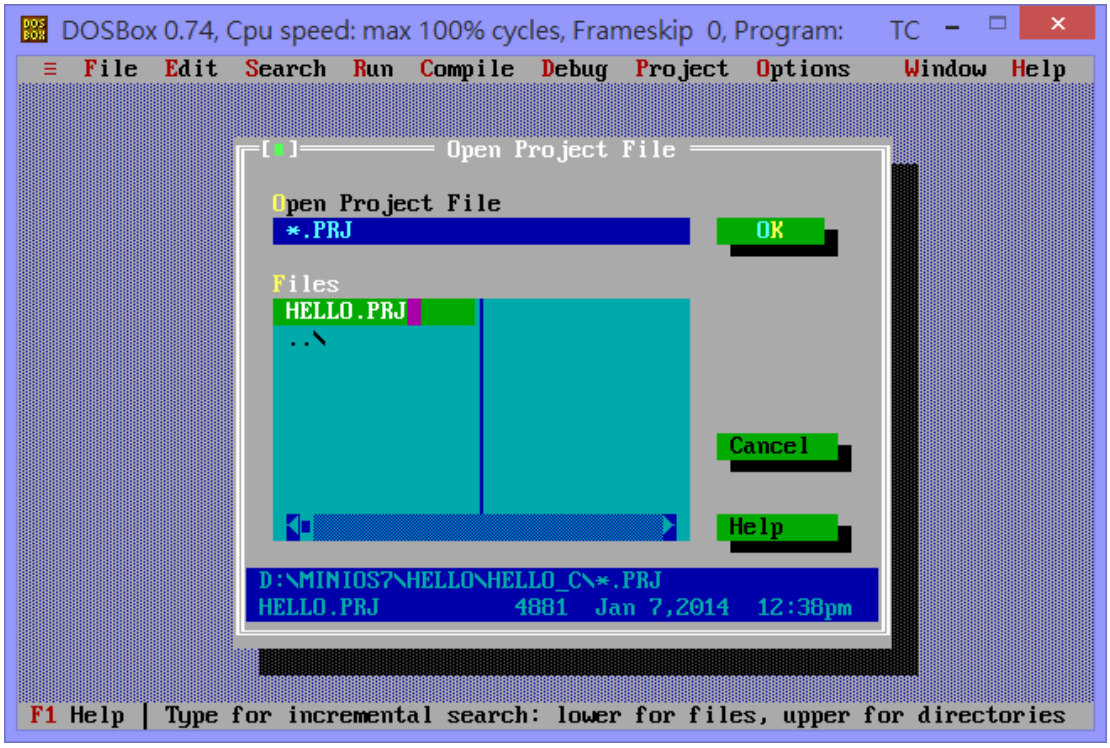

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: INSTALL
Turbo C++ 3.0 Installation Utility
Enter the SOURCE Path
\TURBOC
Description
Enter the path to the directory containing the Turbo C++ files.
ENTER-Select ESC-Cancel


DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: INSTALL
Turbo C++ 3.0 Installation Utility
Directories... [D:\TC]
Options... [IDE CMD LIB CLASS BGI HELP EXMPL]
Start Installation
Description
Selecting this option will begin copying files to your hard drive into the directories specified above.
F1-Help F9-Start the installation ENTER-Select ESC-Previous



After installed the TC++3.0, then set the Environment Path and change to demo path to run TC compiler IDE.




DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
File Edit Search Run Compile Debug Project Options Window Help
Open Project File
Open Project File
*.PRJ
OK
Files
HELLO.PRJ
..\
Cancel
Help
D:\MINIOS7\HELLO\HELLO_C*.PRJ
HELLO.PRJ 4881 Jan 7,2014 12:38pm
F1 Help | Type for incremental search: lower for files, upper for directories