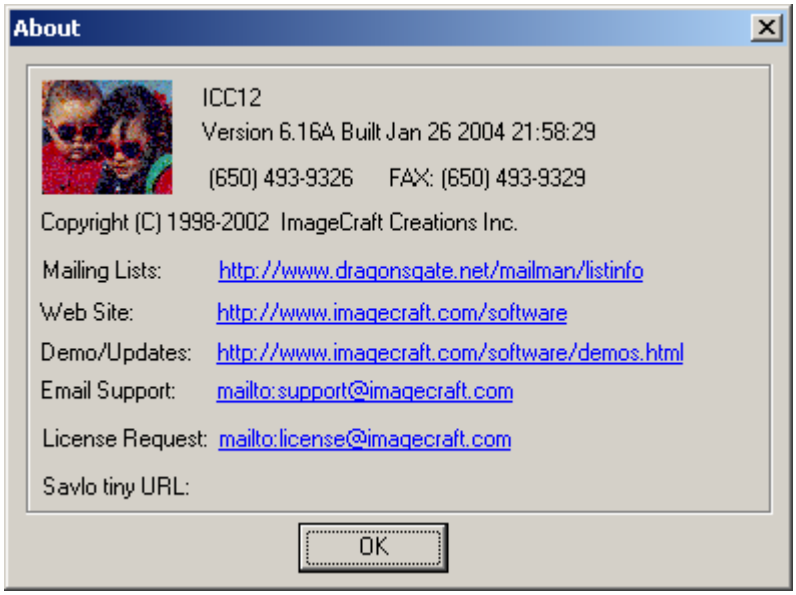
How to use ICC12 with Adapt912B32 and FLASH Loader

This document will show and demonstrate the use of ImageCraft ICC12 Latest ***Version 6*** with Technological Arts' Adapt912B32 module.

The FLASH Loader written by Gordon Doughman Arts and can be found at www.freescale.com website. The FLASHLoader will be used here to erase and program FLASH after the compilation of a test program. Other method can be used to also erase and program the FLASH but in this example it will be the FLASH Loader.

This document assumes that the user is familiar with C and so will not teach how to program C here.

ImageCraft Links:



http://www.imagecraft.com/software/
http://www.ece.utexas.edu/%7Evalvano
http://www.dragonsgate.net/FAQ/cache/20.html
http://www.imagecraft.com/software/mdevtools.html
http://www.dragonsgate.net/mailman/listinfo

Freescale Link:

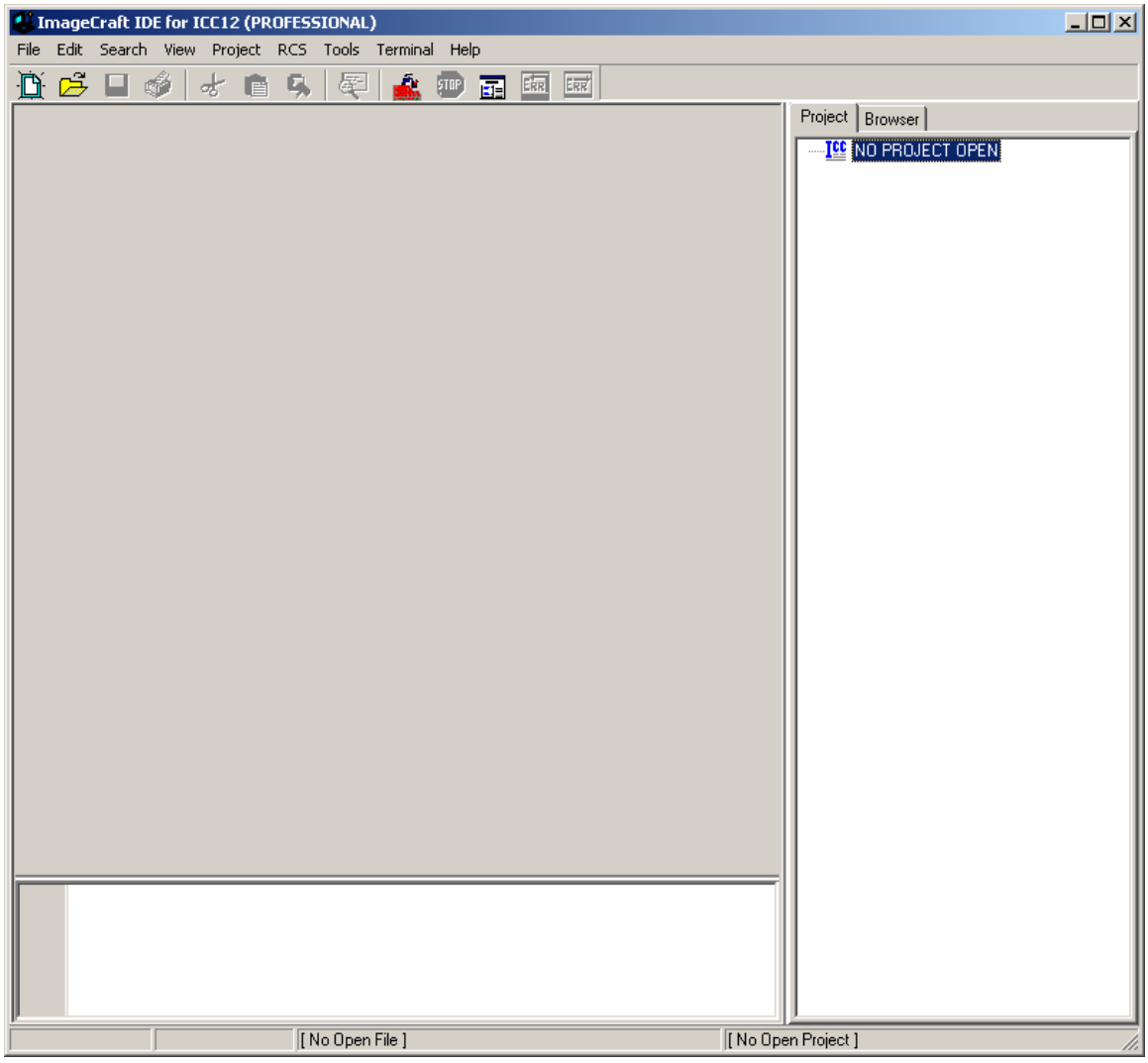
http://www.freescale.com/
http://www.freescale.com/files/soft_dev_tools/software/app_software/dbug_rom_monitors/MC68HC912B32EVBFW.zip

Getting Started:

Double click on the ICC12 icon. If a user has not read the ICC12 manual and just open the IDE one will wonder what to do next. Well wonder no more.

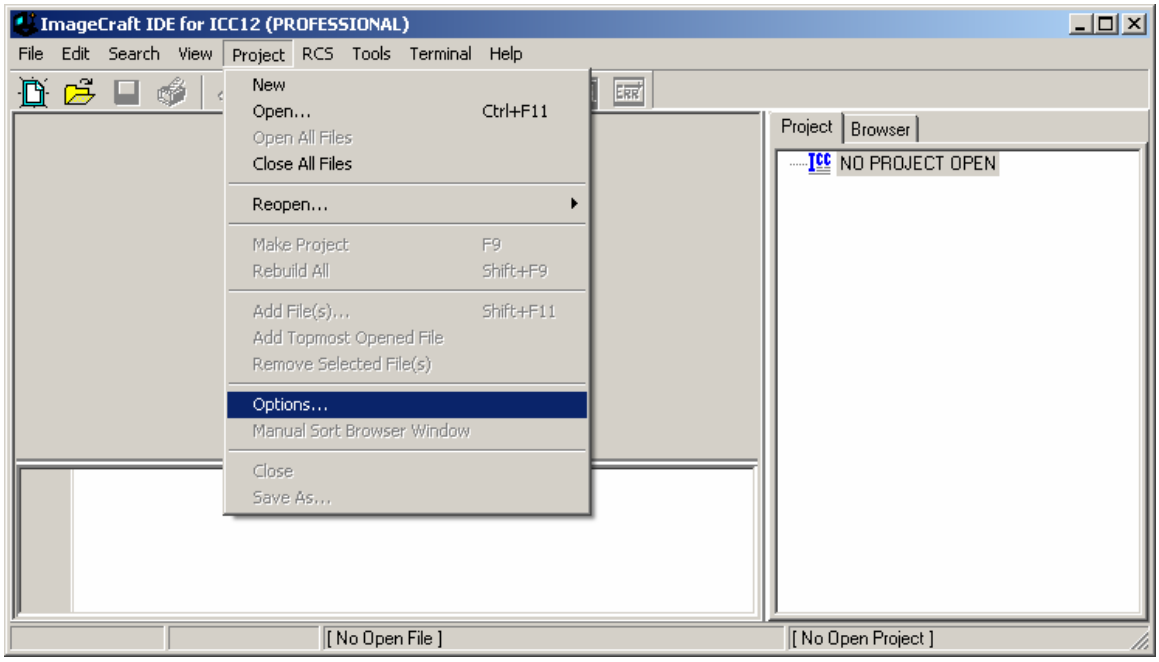
Note the 3 window panes. The top left most is greyed out and the right is the project window. The left bottom pane is where the error messages are displayed during compilation.

Before creating a new Project, the hardware target in the Compiler Options must be setup properly for the target MCU. This is to ensure that the compiler will setup the type of MCU the C program will compile for. In this example it is the Adapt912B32.

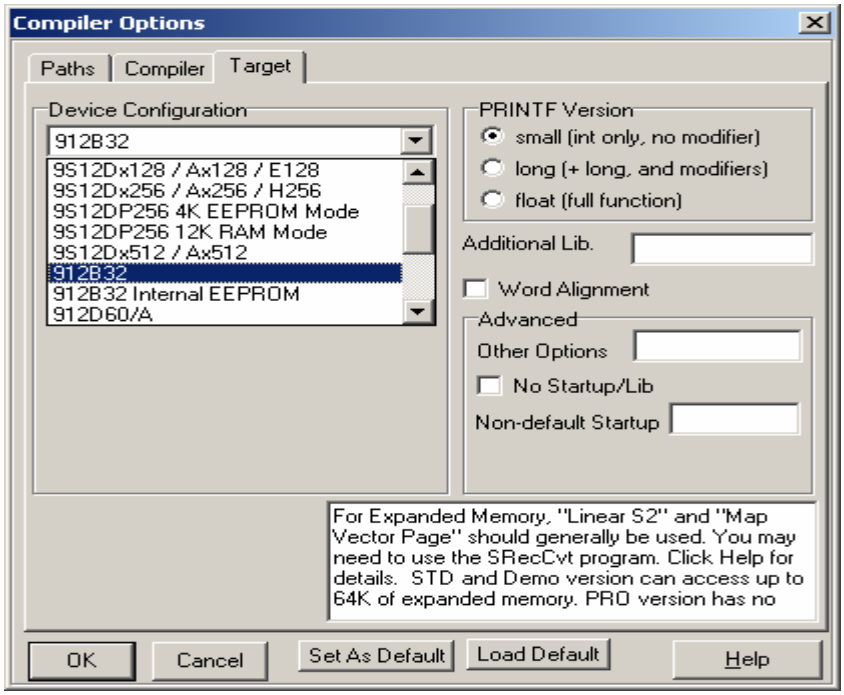


Compiler Setup:

Click on Project Menu – Options – Target Tab.

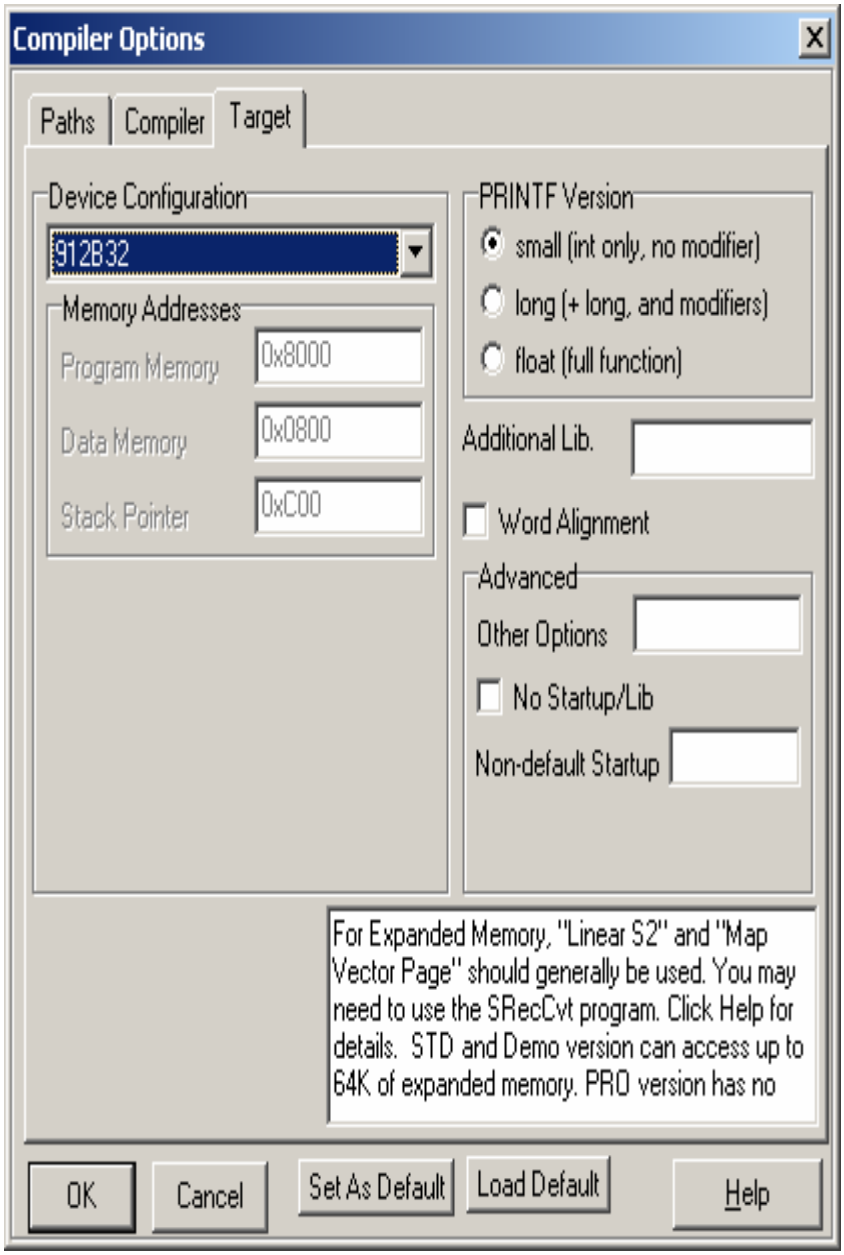


Please note the Device Configuration. Click on the pull down arrow to change the device type.



Scroll up or down to select 912B32 as shown.

Select 912B32.

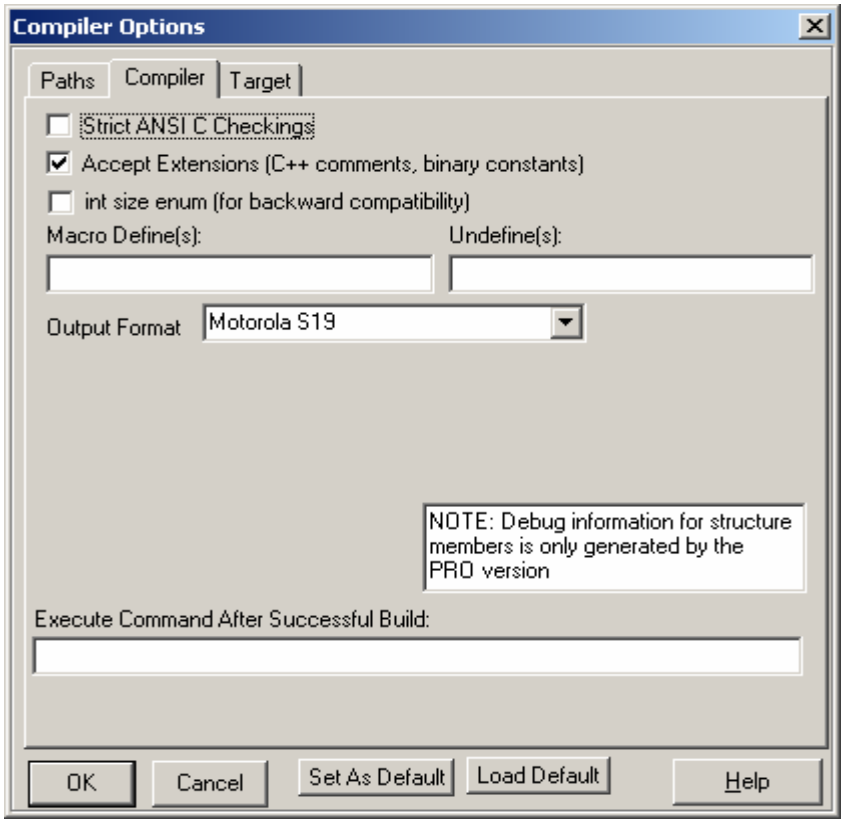


Device Configuration:

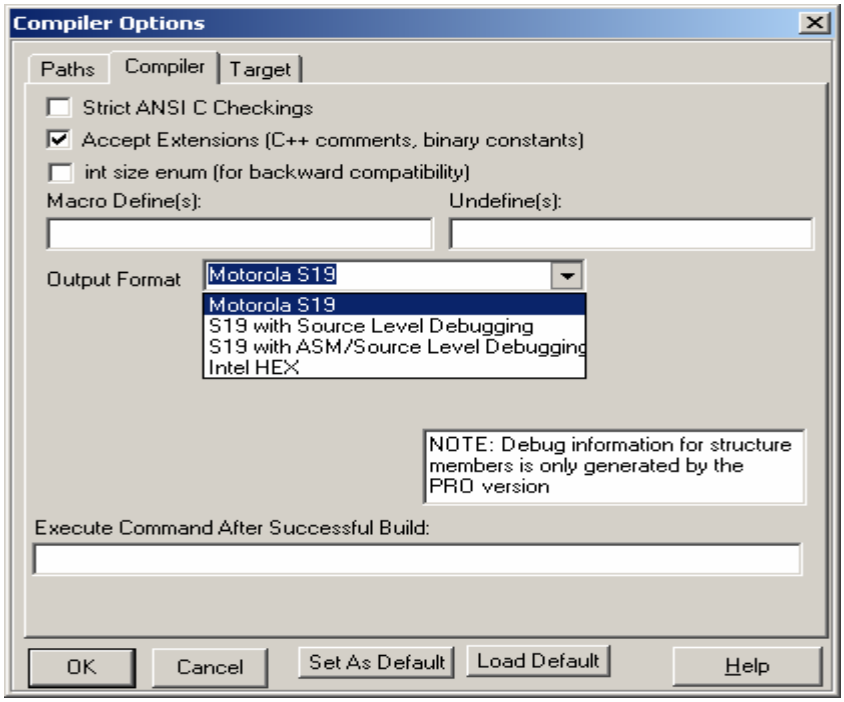
Program Memory: ***0x8000***
Data Memory: ***0x0800***
Stack Pointer: ***0x0C00***

The program code is allocated to start from 0x8000. The internal RAM is to start from 0x0800 and the stack is to start from 0x0C00 and work downward.

On the compiler tab there are several choices of S-record output as shown.

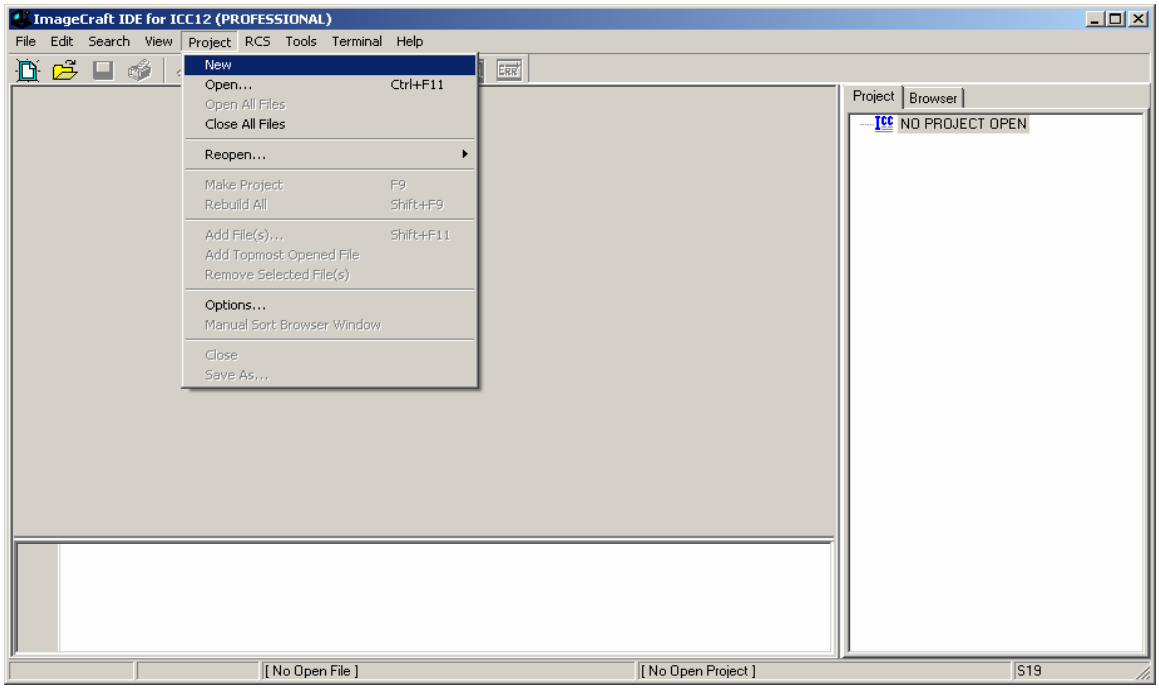


Select which one that suits you.

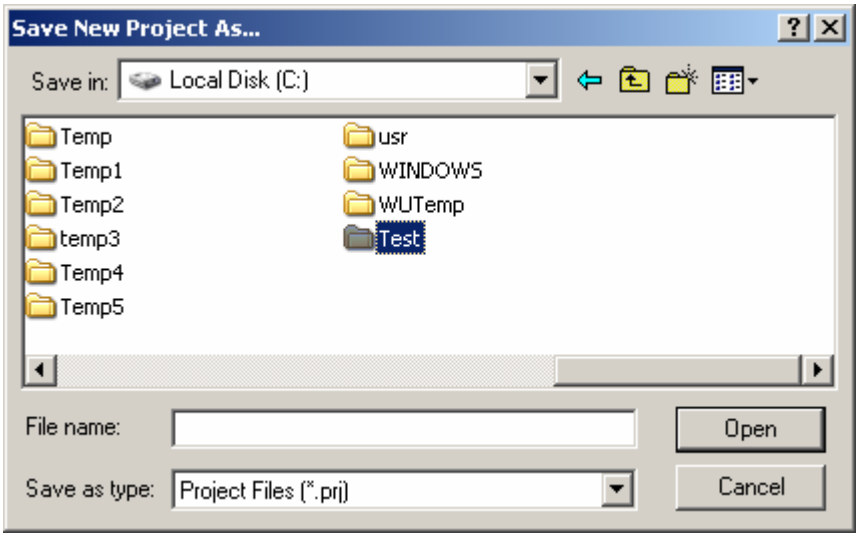


Starting a new Project:

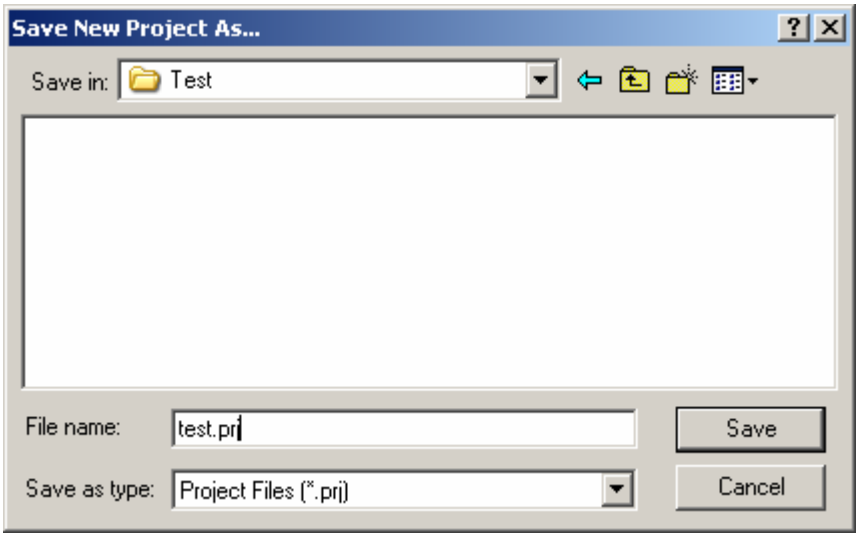
Once the compiler options are setup, a new project can be created. Click Project menu – New.



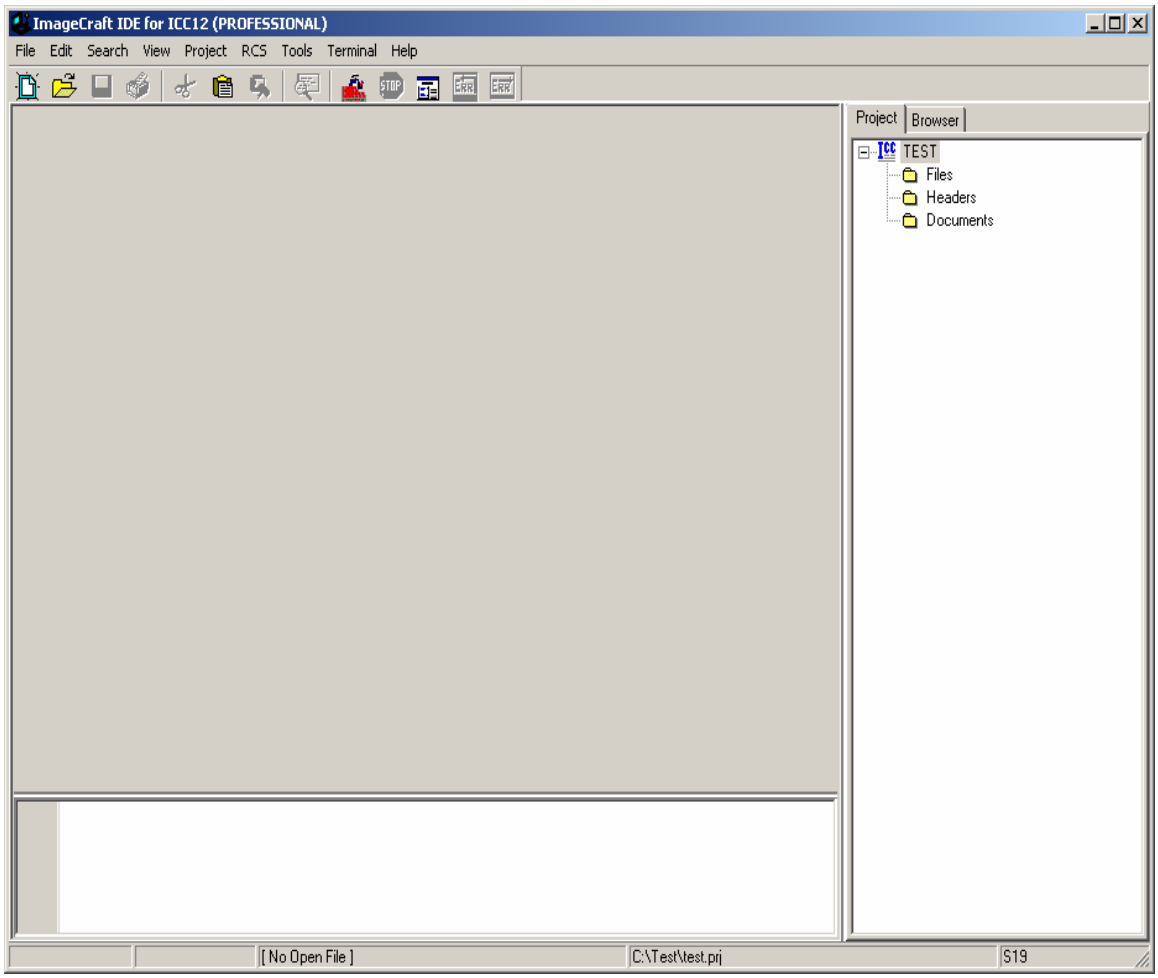
The ICC12 will prompt to save the new project. The user should decide whether to create a new directory to save the new project. In this example a new directory called ***Test*** is created and the file is saved as file ***test.prj***.



Type the filename as ***test.prj*** and click on the Save button.

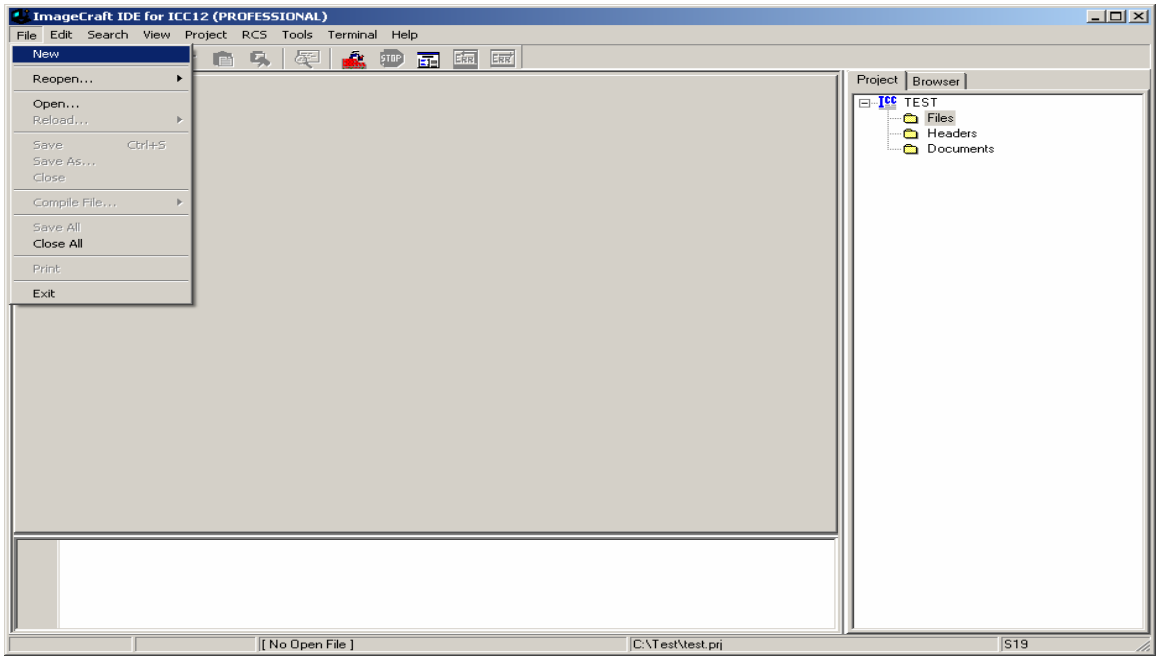


Note that the project window has changed to add Files, Headers and Documents.

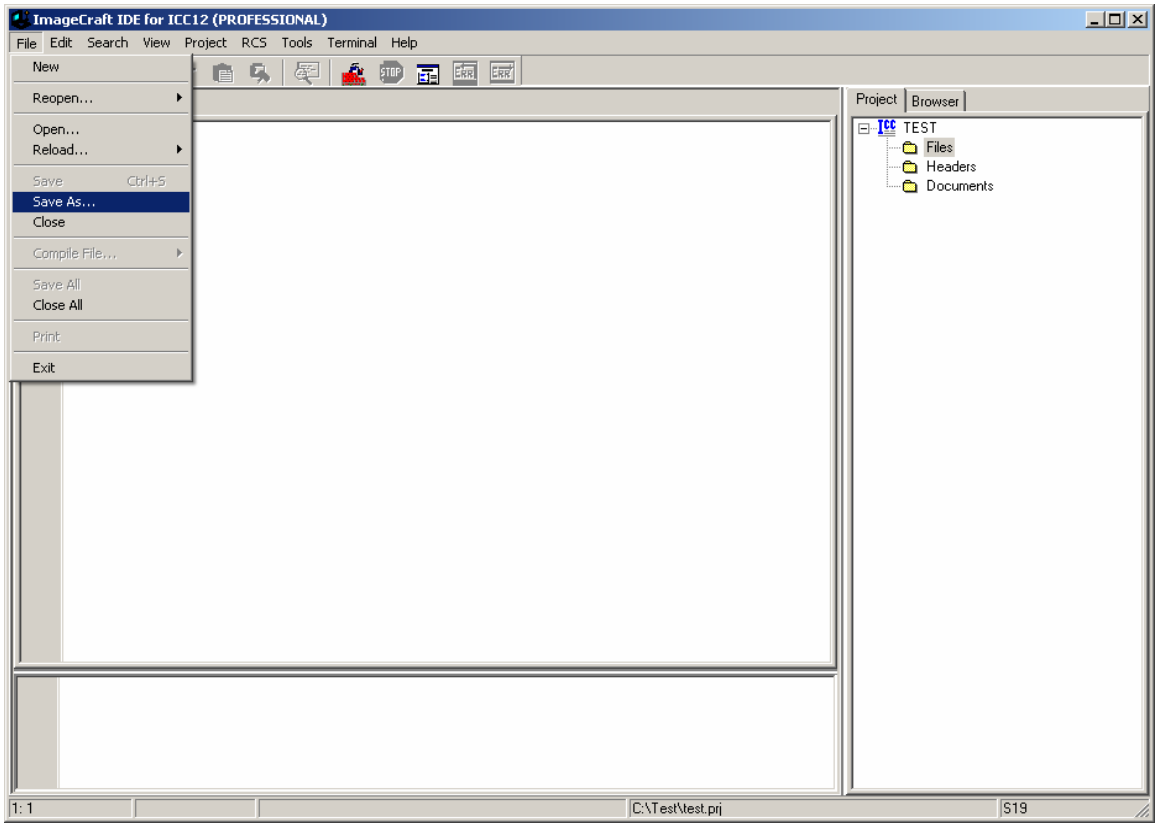


Creating a new file to the project:

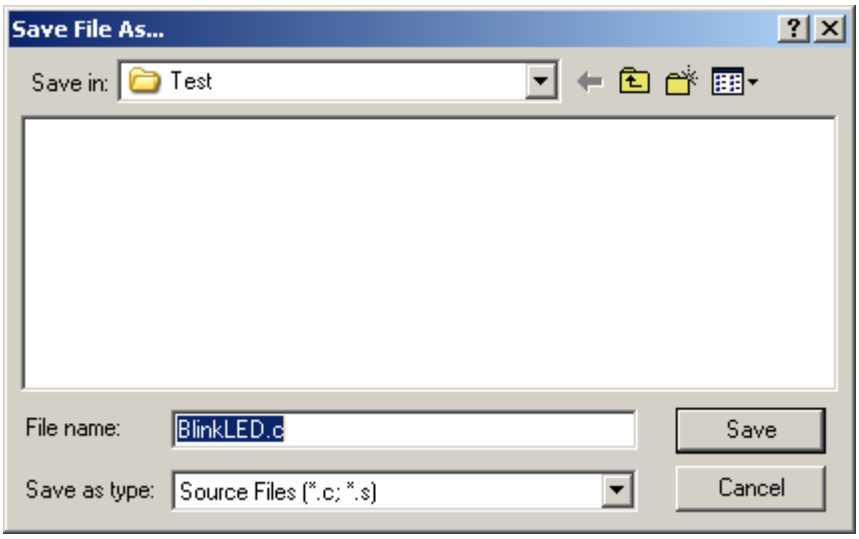
To add files to the project, click on the File menu – new as shown.



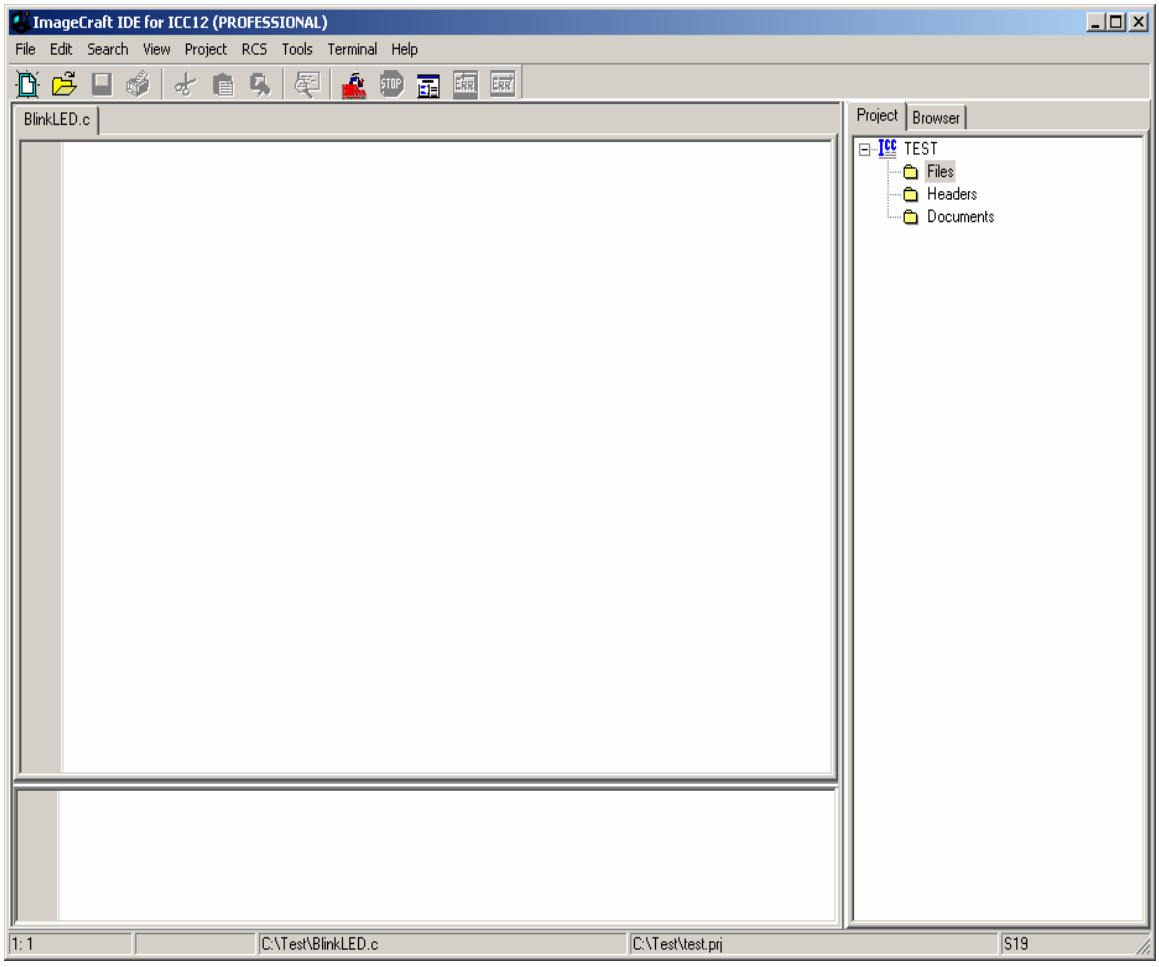
Note that ICC12 created an untitled file. Save the file as ***BlinkLED.C***.



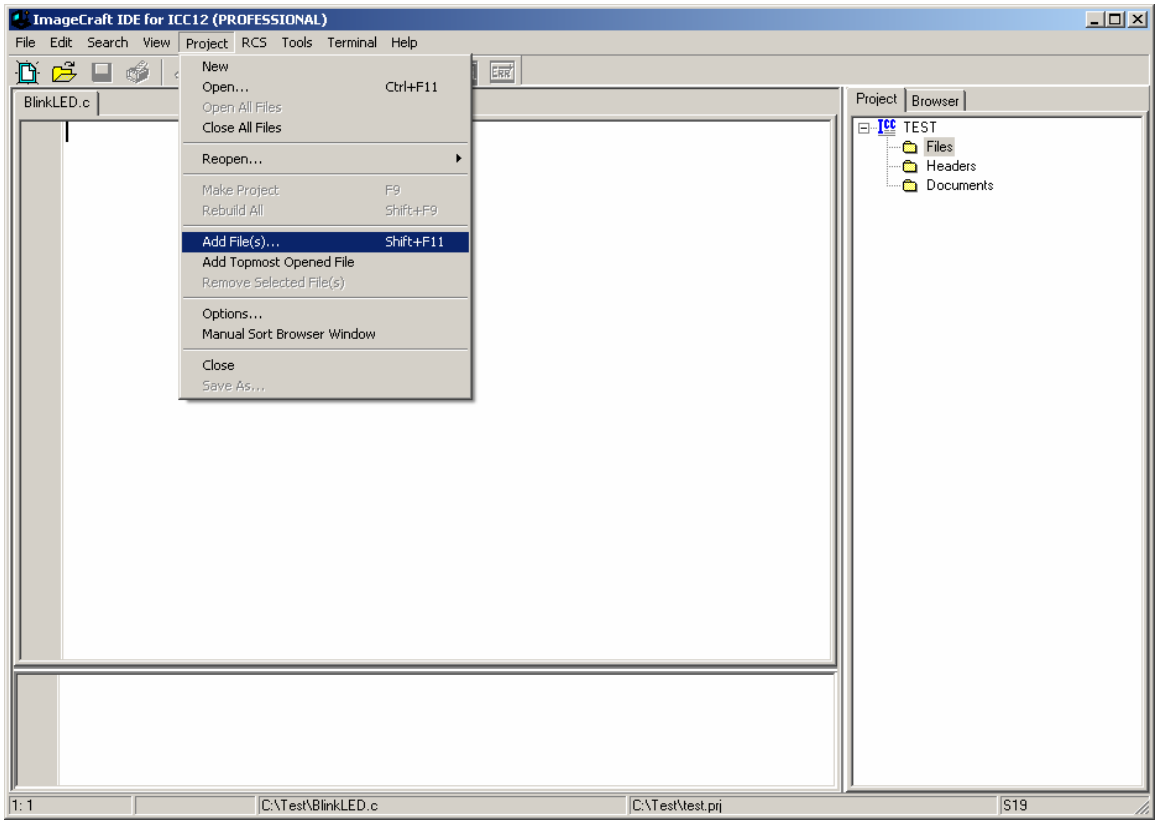
ICC12 will open an explorer window to help save the file. Type BlinkLED.c then press the save button.



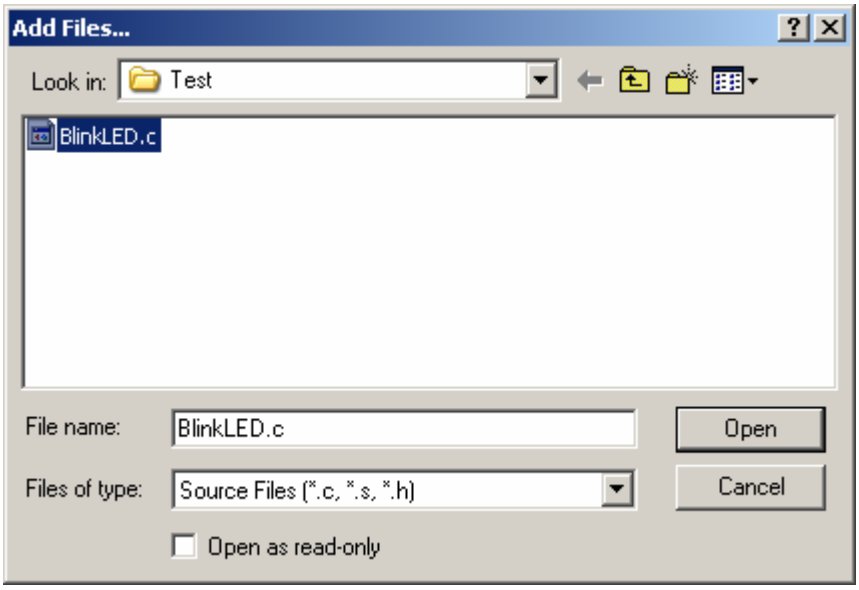
Note that ICC12 has renamed the file to BlinkLED.c.



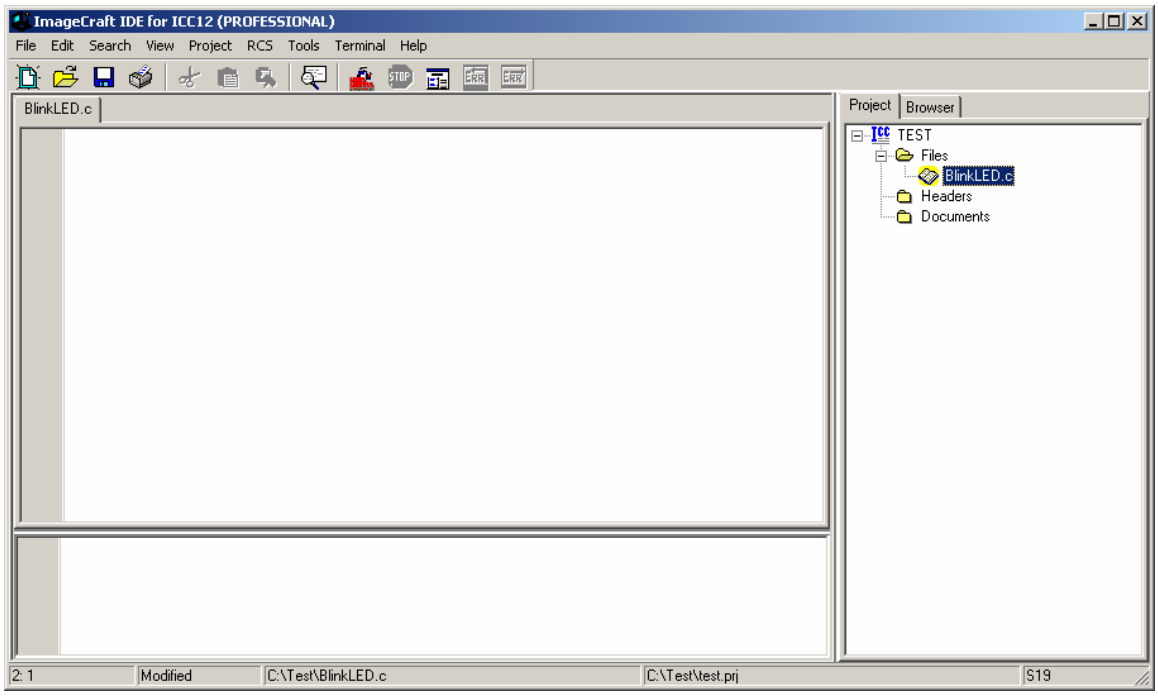
To add BlinkLED.c to the Project, click on the Project menu – Add File(s)



ICC12 will open an explorer window to help and locate the file of interest.

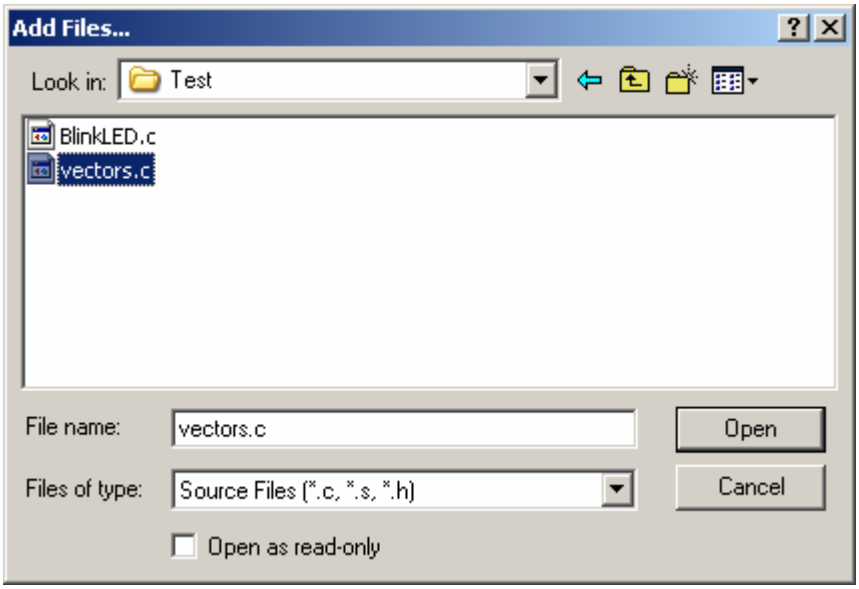


Note that the right window pane has changed to include BlinkLED.c under the Files Project.



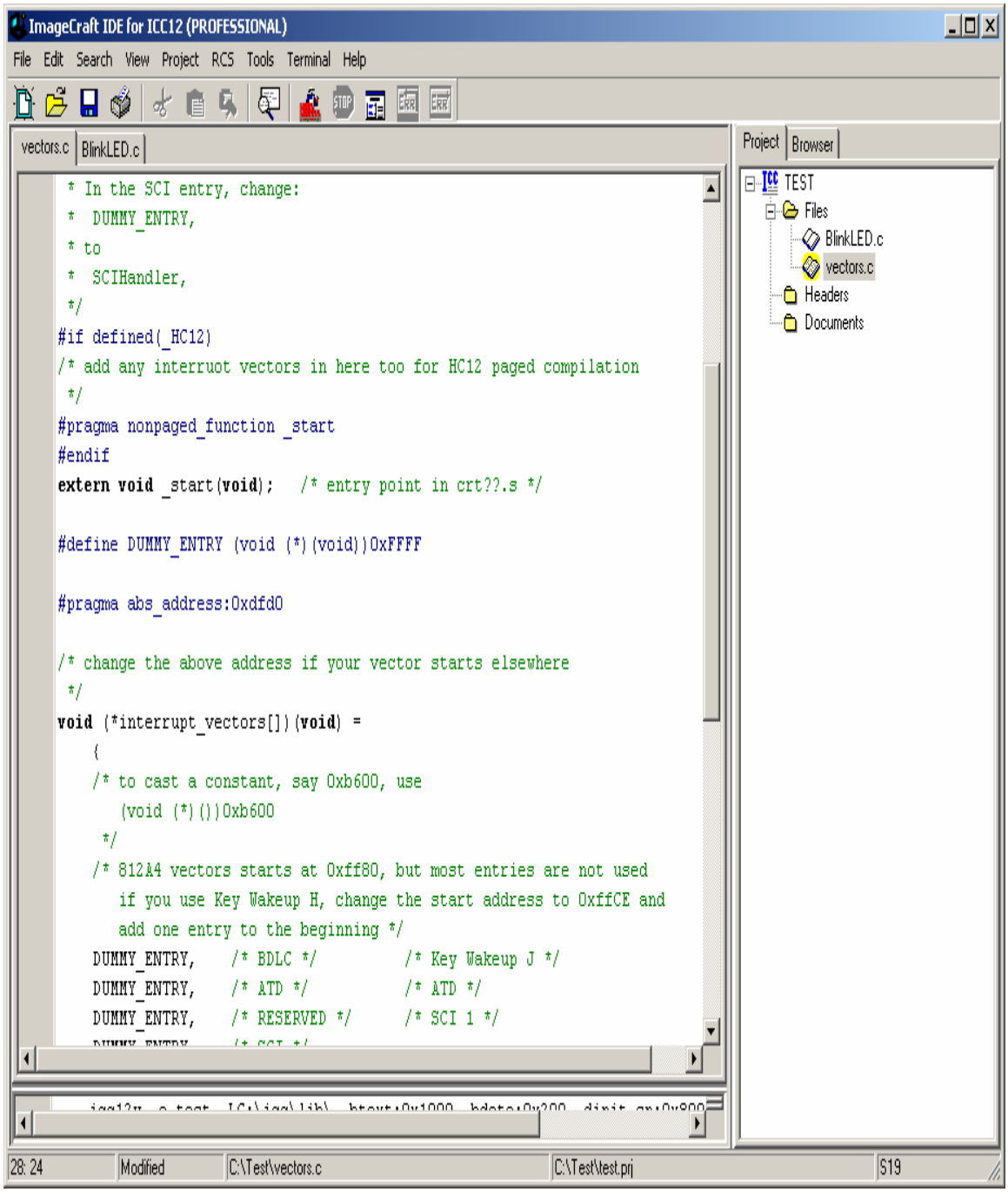
Locate ***vectors.c*** and copy file to Test directory. The major reason why this must be done is because of project to project dependency. It is not good to keep editing a single ***vectors.c*** if other projects are using this same file. It becomes a problem to keep track of the changes made to the different projects.

To add ***vectors.c*** to the Project, click on the Project menu – Add File(s)



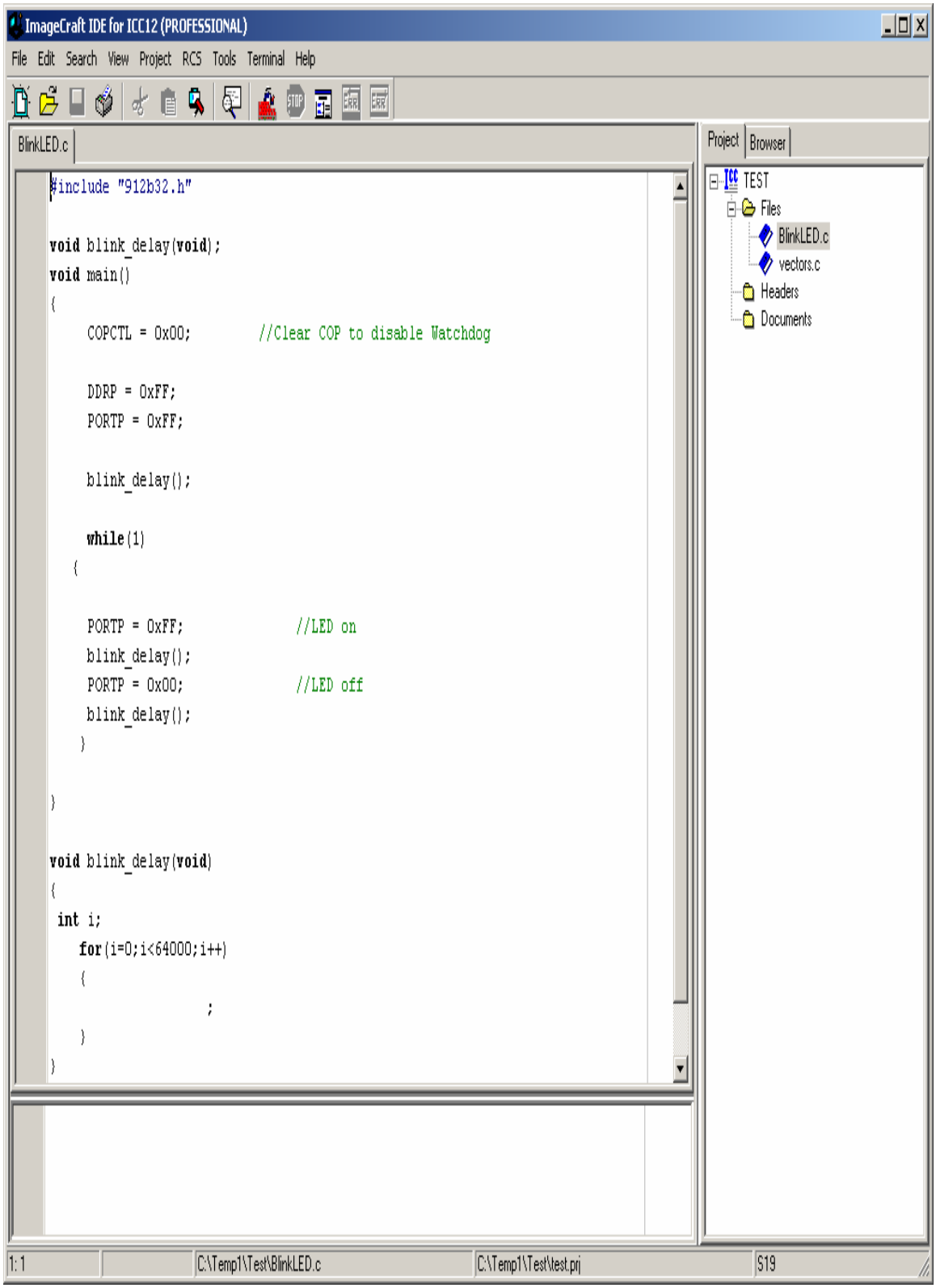
Note that ICC12 has changed to include ***vectors.c*** It is important to note that the **vectors.c** was written for the 68HC912B32 and 812A4 MCUs. This example edits the line ***#pragma abs_address:0xffd0*** to ***#pragma abs_address:0xf7d0***

The original vector address ***0xFFD0*** is changed to ***0xF7D0***. This is because the FLASH Loader resides from ***$F800*** to ***$FFFF***.



Write the codes below into BlinkLED.c file. Once it is written we can then

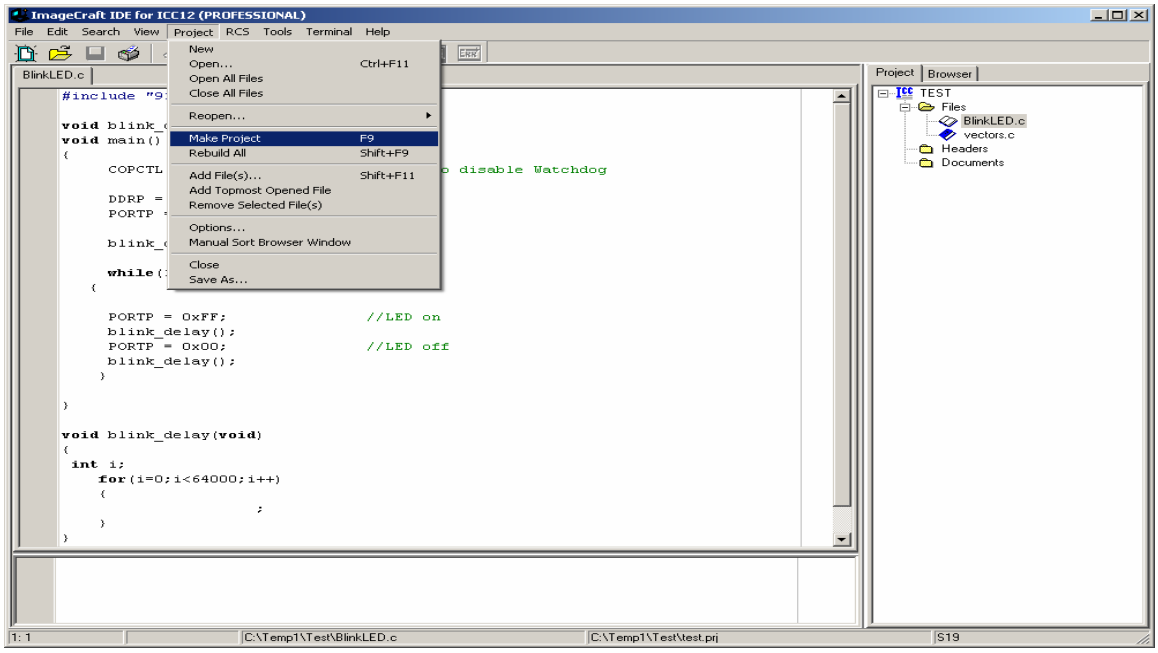
compile/make/build the code.



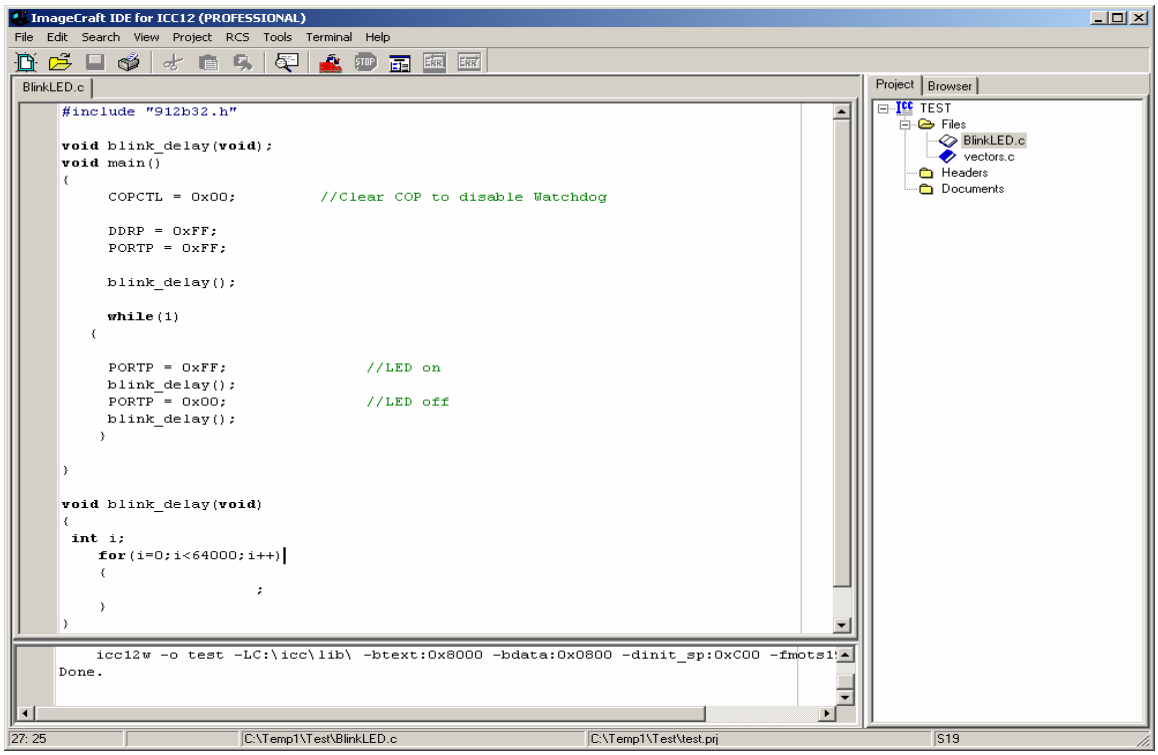
```
#include "912b32.h"

void blink_delay(void);
void main()
{
        COPCTL = 0x00;                  //Clear COP to disable Watchdog

        DDRP = 0xFF;
        PORTP = 0xFF;

        blink_delay();

        while(1)
  {

        PORTP = 0xFF;                   //LED on
        blink_delay();
        PORTP = 0x00;                   //LED off
        blink_delay();
       }

}

void blink_delay(void)
{
 int i;
        for(i=0;i<64000;i++)
        {
                                        ;
        }
}
```

Compiling/Build/Make the file:

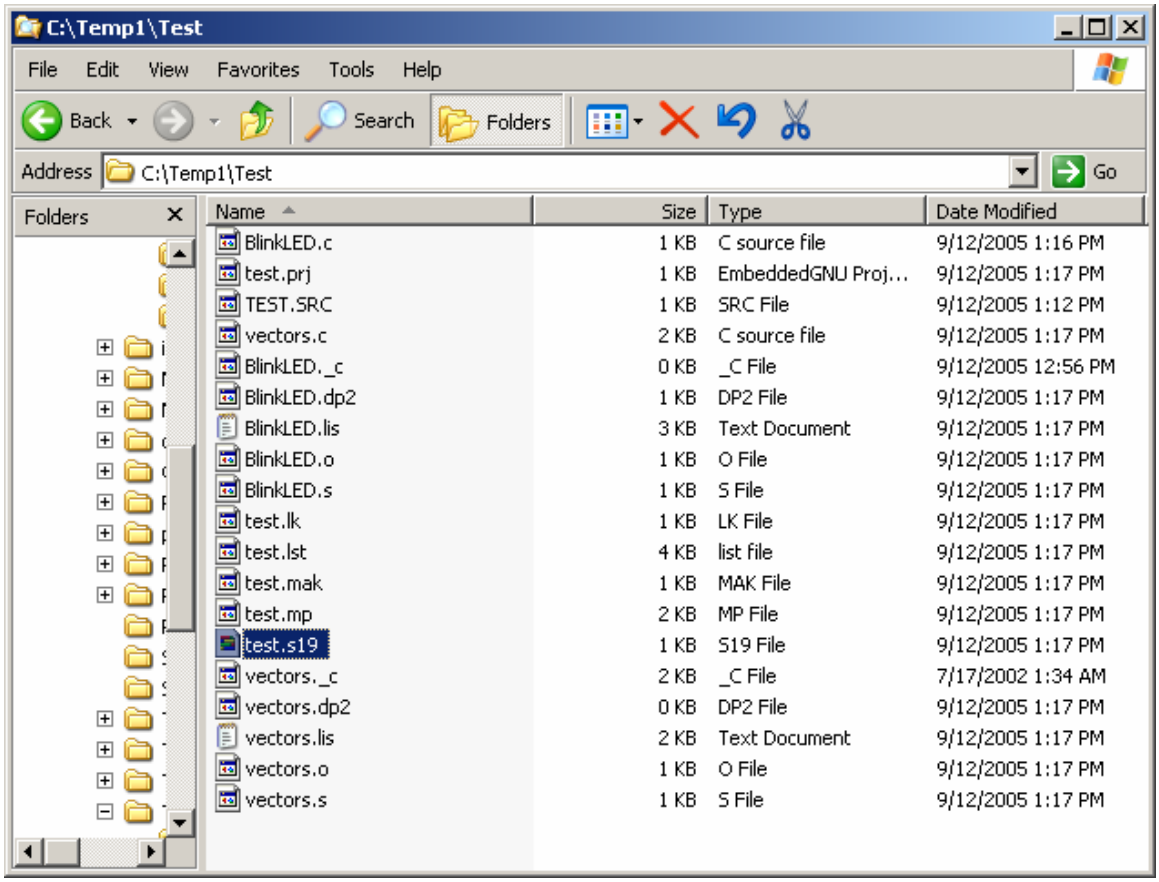
To make the file click Project menu – make project as shown.



Note the bottom window pane will show messages to display how the build progressed. Any errors, if any, are shown in this window. The build was without error so we can progress to erasing and programming the 912B32.



Note the other extraneous files are created after a make.



Using WordPad to check the content of ***test.s19*** file. Note that the S-records are of different lengths.

S10E8000CF0C0016806B87CE08008EAA
S110800B080027056A000820F6CE8070CD1D
S111801808008E80702706180A307020F516B6
S1078026802A20FE8A
S111802A790016C6FF7B0057C6FF7B00561672
S1108038804D200EC6FF7B005616804D794A
S1118045005616804D20F03D34B7751B9ECCBE
S111805300006C1E2007EC1EC300016C1EEC26
S10D80611E8CFA0025F2B757303DDB
S111F7D0FFFFFFFFFFFFFFFFFFFFFFFFFF35
S111F7DEFFFFFFFFFFFFFFFFFFFFFFFFFF27
S111F7ECFFFFFFFFFFFFFFFFFFFFFFFFFF19
S109F7FAFFFFFFFF800089
S108806B1D0016073D95
S90380007C

Examining S-record:

If one looks closely at the S-record one can see S1 to be of different lengths. This is atypical S-record generated by ICC12.

As stated previously, FLASH Loader occupies ***$F800*** to ***$FFFF*** therefore the vector address at below ***$F800***.

Note the content of the memory address at ***$F7FE:$F7FF*** is ***$8000***, the pseudo RESET vector.

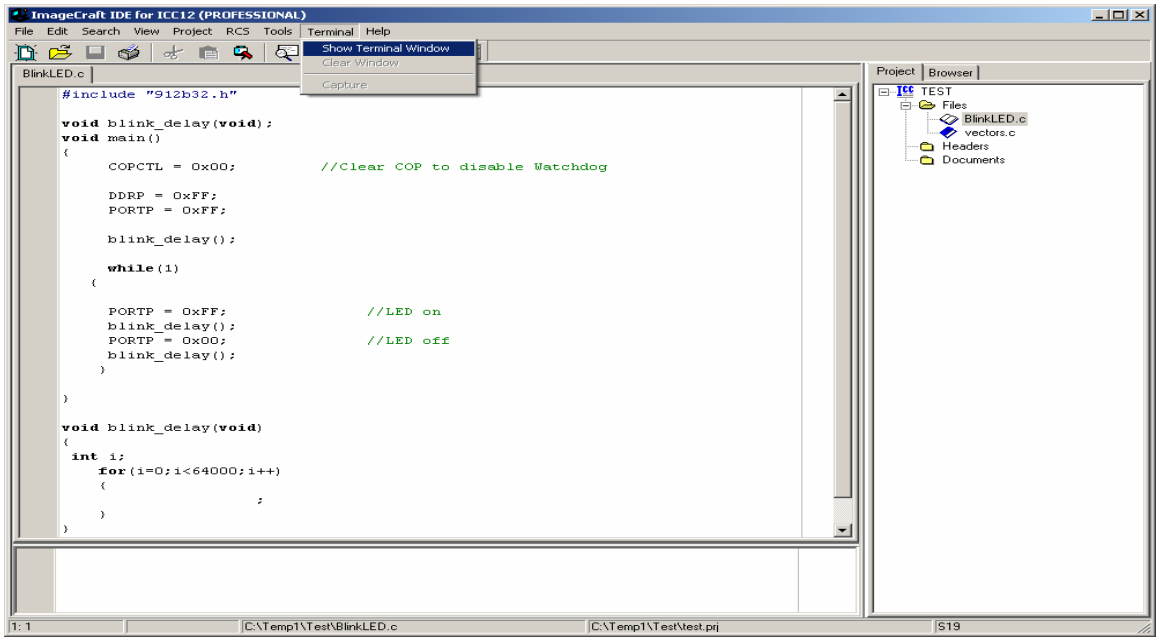
S111F7D0FFFFFFFFFFFFFFFFFFFFFFFFFFFF35
S111F7DEFFFFFFFFFFFFFFFFFFFFFFFFFFFF27
S111F7ECFFFFFFFFFFFFFFFFFFFFFFFFFFFF19
S109F7FAFFFFFFFF***8000***89

The S-record below is the start of code. The content of address beginning at $8000 to $8050

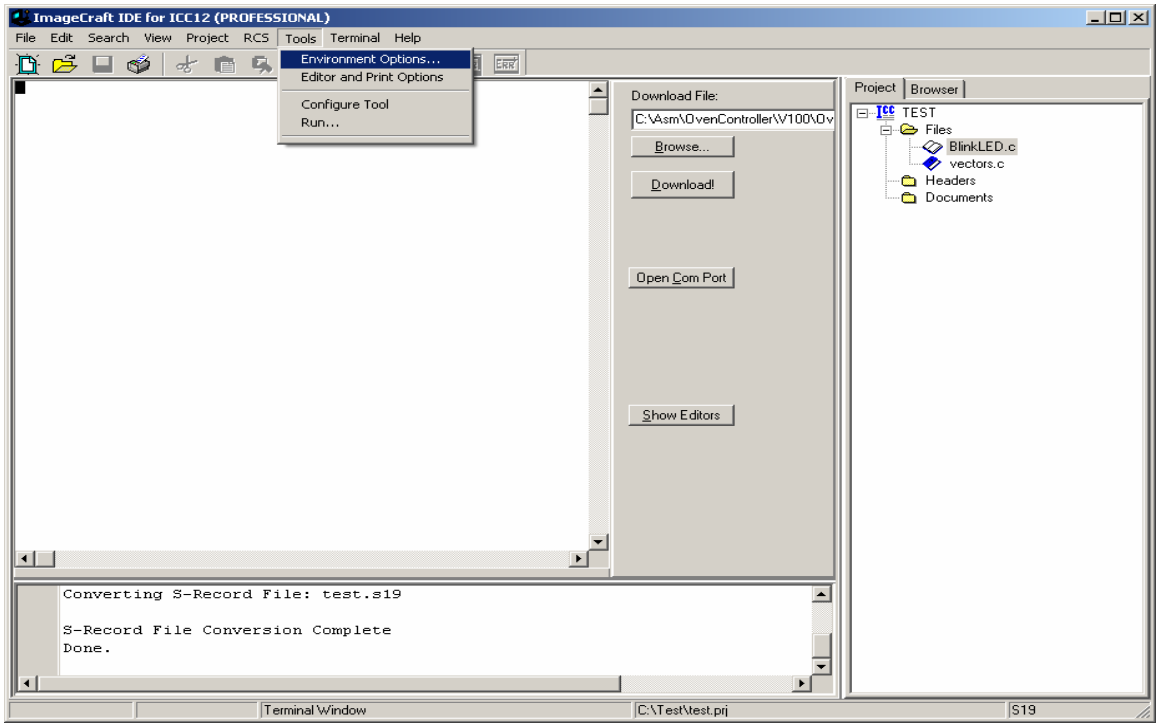
S10E***8000***CF0C0016806B87CE08008EAA
S110800B080027056A000820F6CE8070CD1D
S111801808008E80702706180A307020F516B6
S1078026802A20FE8A
S111802A790016C6FF7B0057C6FF7B00561672
S1108038804D200EC6FF7B005616804D794A
S1118045005616804D20F03D34B7751B9ECCBE
S111805300006C1E2007EC1EC300016C1EEC26
S10D80611E8CFA0025F2B757303DDB

Programming the Adapt912B32:

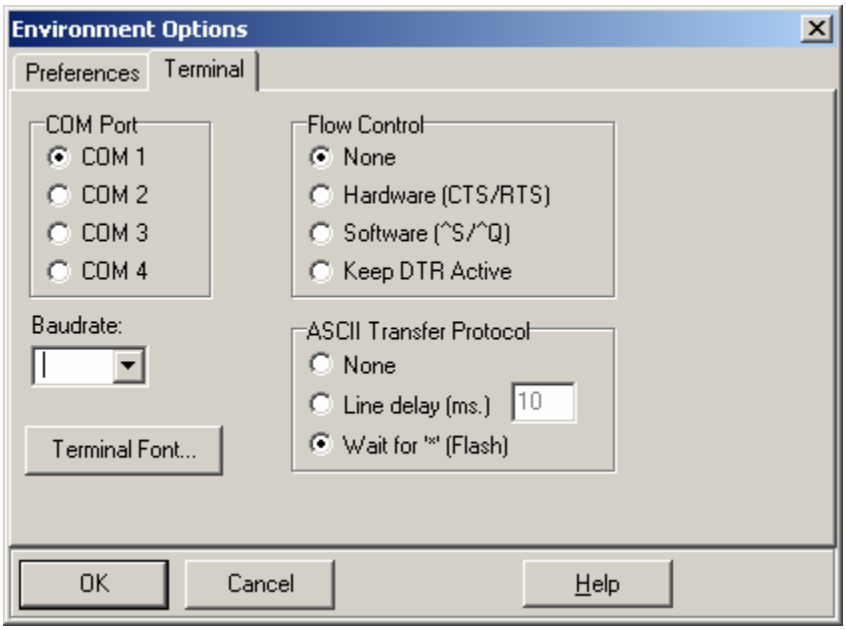
Open ICC12 terminal window by Selecting Show Terminal Window. Connect Serial cable to any available PC COM port and the other end to Adapt912B32.



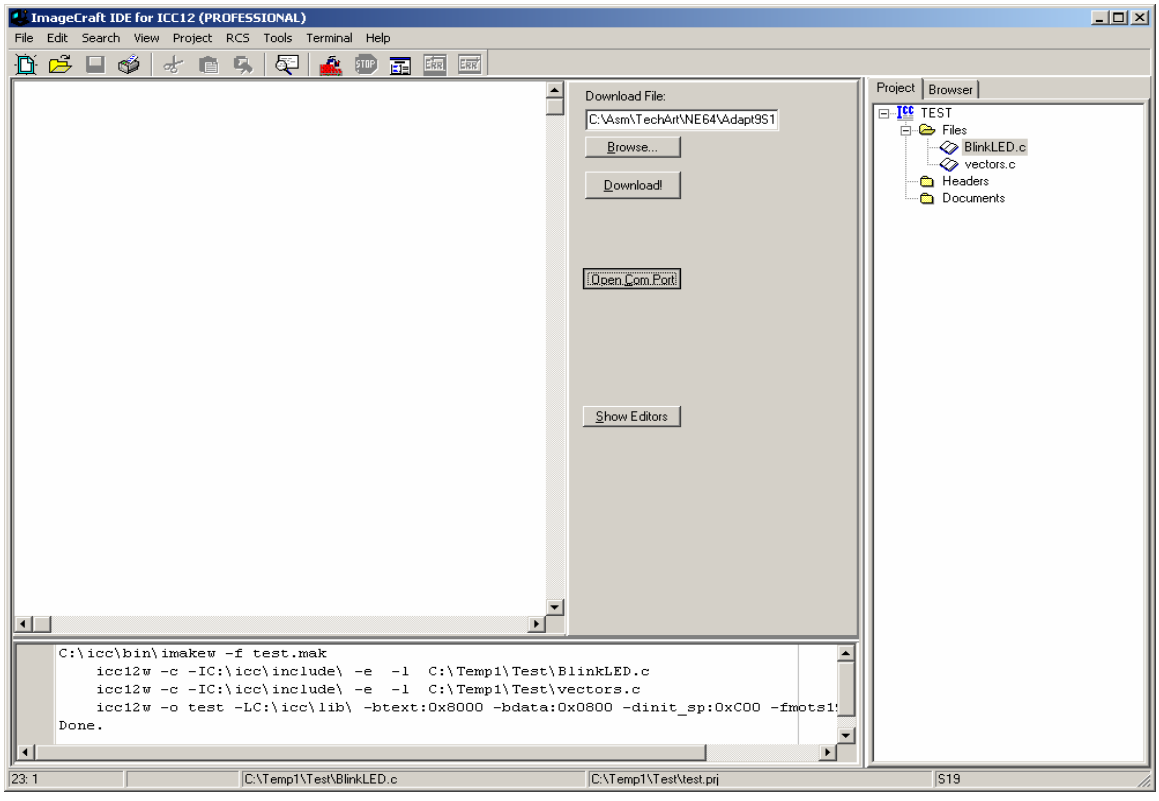
Set the COM port parameters under Tools-Environment Options and then Terminal tab.



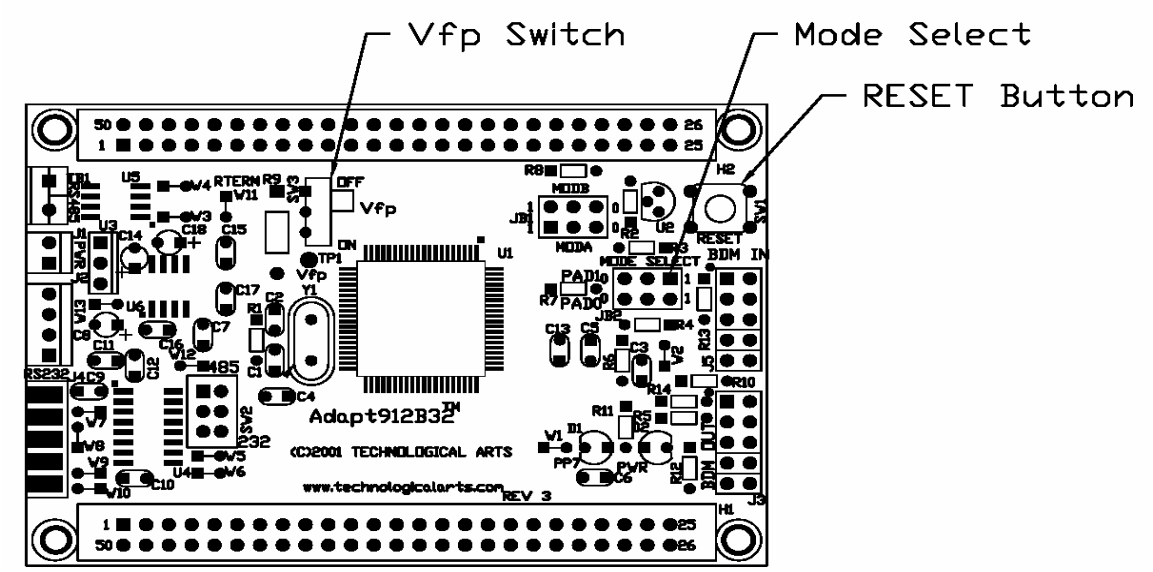
Select the COM of your choice, BAUD = 9600, Flow Control = None and ASCII Transfer Protocol to Wait for * (Flash). Press OK button to continue.

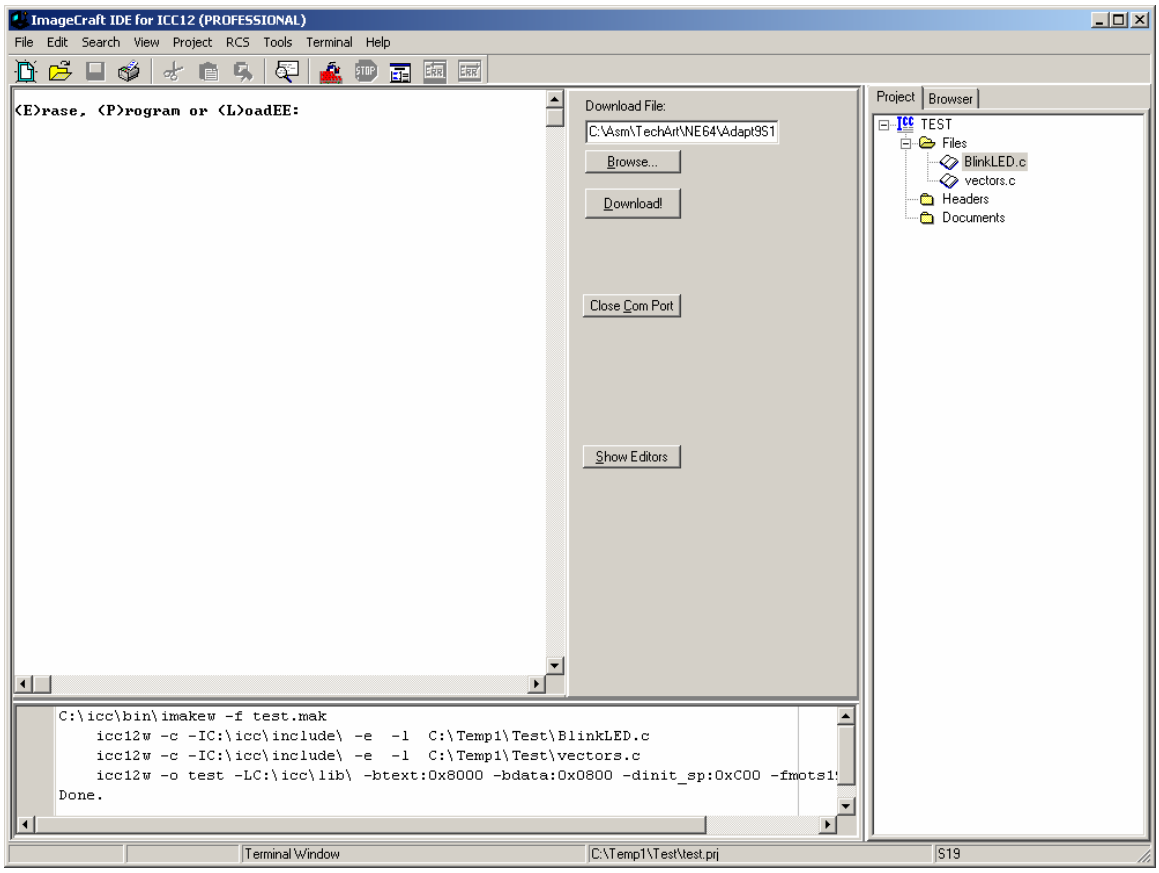


In the middle is the ***Open Com Port*** button. Double check the serial Connections between Adapt912B32 and at the back of the PC. Click to connect.



Warning!: Make sure Vfp SW3 is in OFF position before powering unit. Failure to do this will damage the MCU. Move ***Mode Select JB2 PAD0:PAD1 = 1:1***. Power up unit or press the RESET button if already powered up.

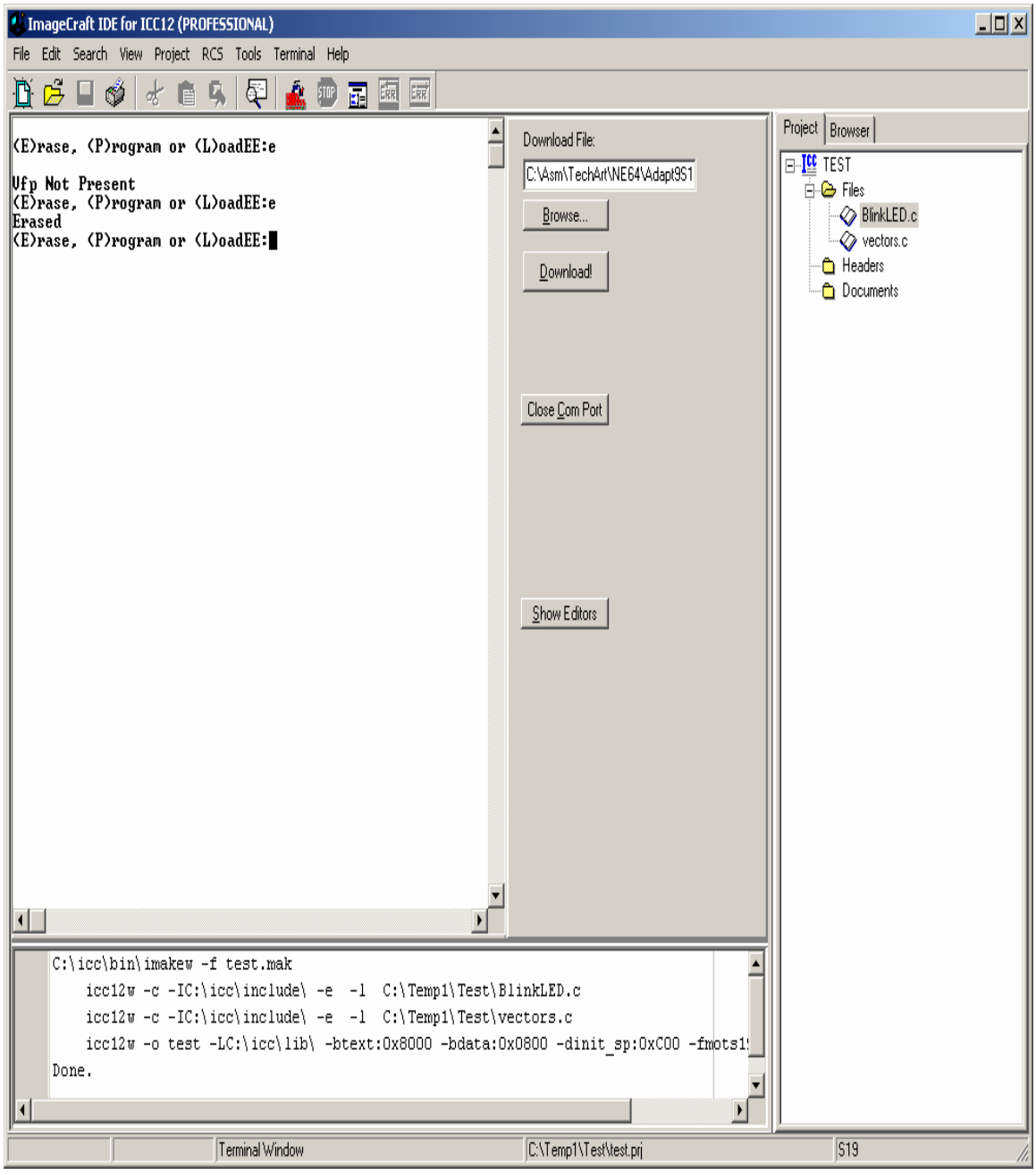




The FlashLoader allows one to Erase, Program the FLASH or EEPROM.

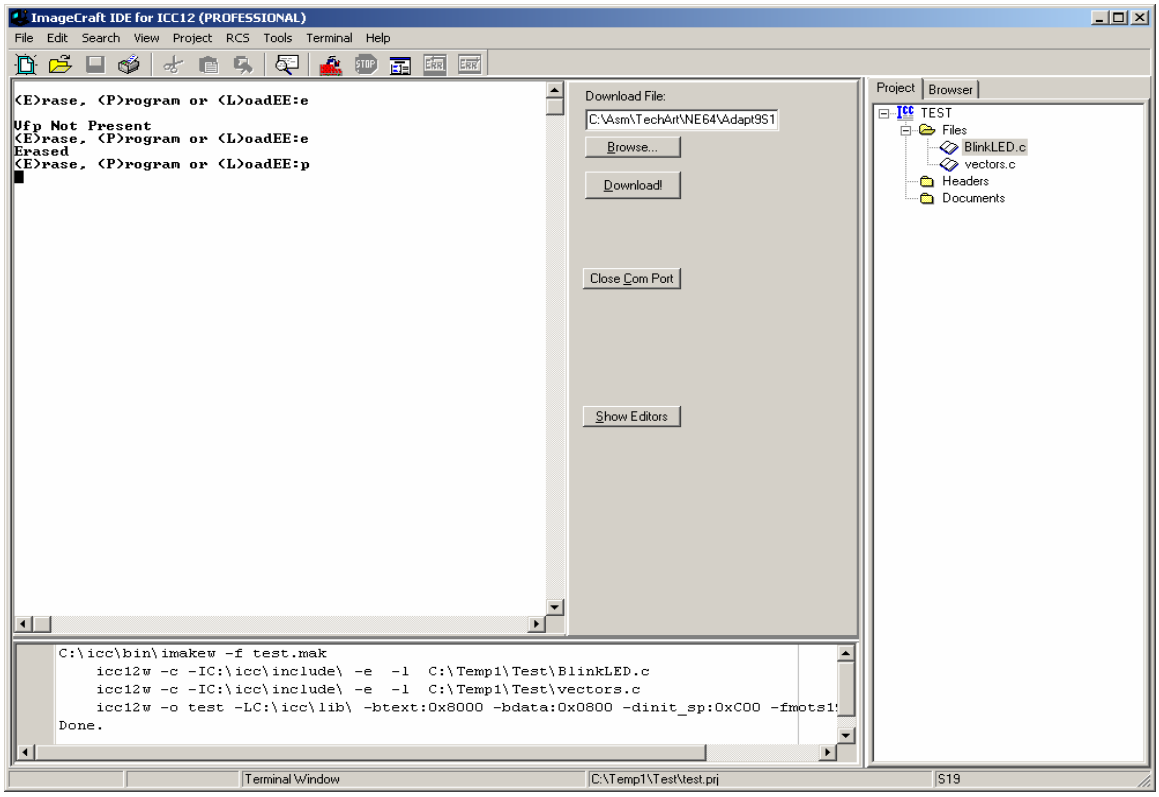
Erase command:

The command to erase FLASH is the letter ***E*** or ***e*** as shown. The error message ***Vfp Not Present*** will show up if the Vfp switch SW3 is not in the ON position. Slide the Vfp switch to ON. An ***Erased*** message will show for a successful erasure of FLASH as shown.

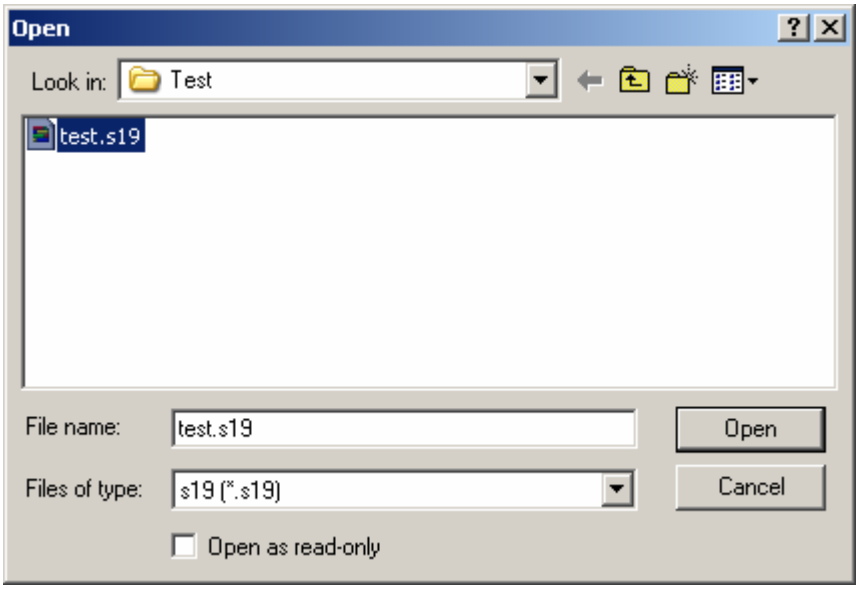


Programming:

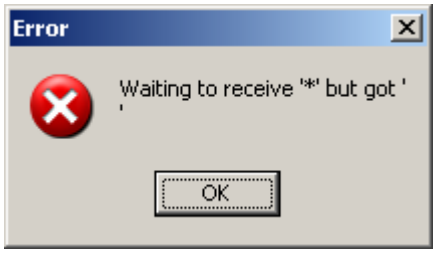
To program select the letter ***P*** or ***p*** command as shown.



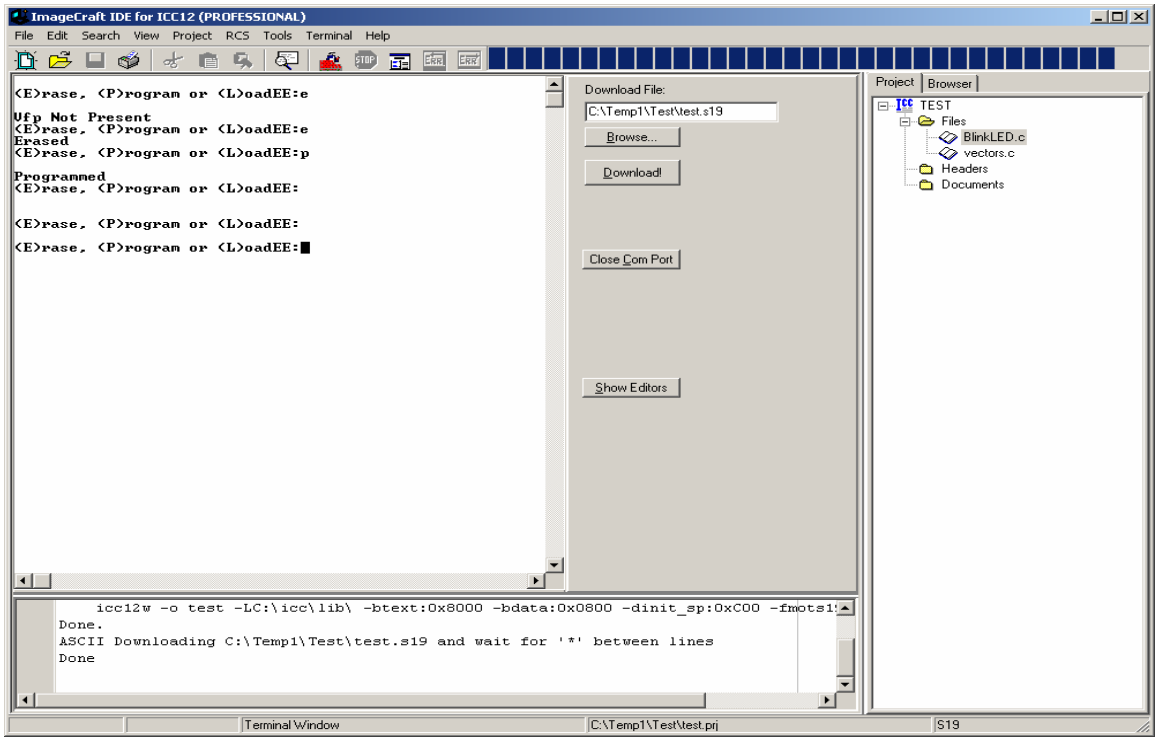
Once the programming is invoked, the MCU is now waiting to be programmed. Press the ***Browse*** button to help locate the ***test.s19*** file. Select ***test.s19*** then press the ***open*** button.



Once the correct file is selected, press the ***Download*** button to initiate upload to Adapt912B32. A progress Bar will indicate that the file is being transferred. After which an error box will pop up. Press ***OK*** button to continue.



Press the ***CR or Enter*** key to bring up the FLASH loader menu. Note the ***Programmed*** message as shown. This is to show that the FLASH was programmed successfully.



Move ***Vfp switch SW3 = OFF*** and ***JB2 PAD0:PAD1 = 0:0*** and press the RESET button. The LED on the Adapt912B32 that is connected to PP7 will begin to blink.

This concludes the use of ICC12 with Adapt912B3260 to using the FLASH Loader.