Writing Your First Firmware

For PDS-700

December 2009, Version 1.0

Table of Contents

Writing Your First Firmware

for

PDS-700

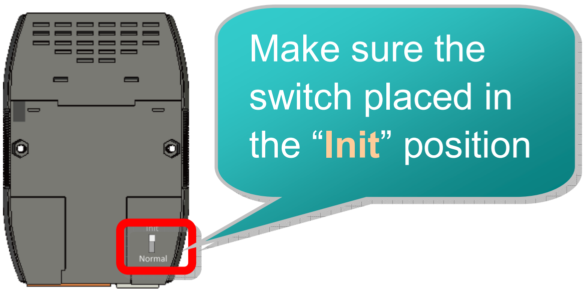
1. Compiler and Linker for C programming

To develop programs for PDS-700 series, you can use the compilers below:

1. BC++ 3.1~5.02
2. MSC
3. MSVC (before version 1.52)
4. TC 2.01
5. TC++ 1.01

From Borland's web site, you can download the free TC 2.01 compilers.
Web site: http://edn.embarcadero.com/article/20841
To download this, you must have registered as a free member.

2. Configuring Boot Mode

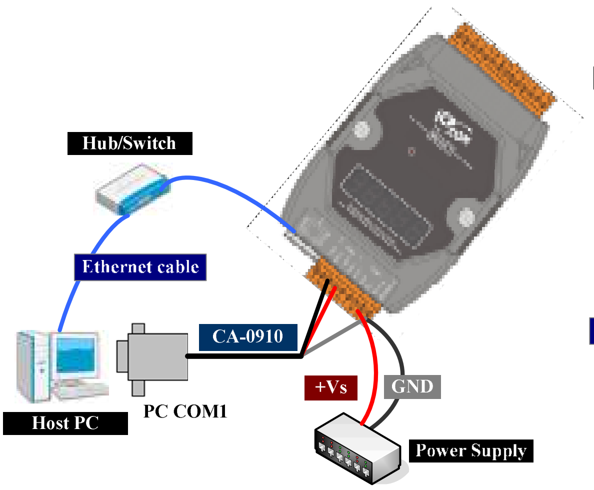


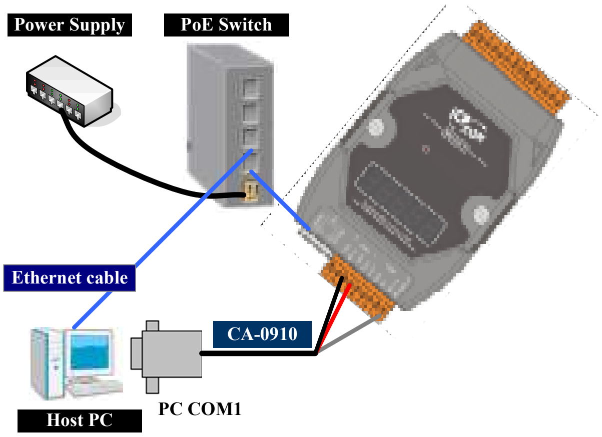
Make sure the switch placed in the “**Init**” position

3. Connecting to Network, PC and Power

The PDS-700 is equipped with an RJ-45 Ethernet port for connection to an Ethernet hub/switch and PC. You can also link directly the PDS-700 to PC with an Ethernet cable.

Non PoE： **PoE：**





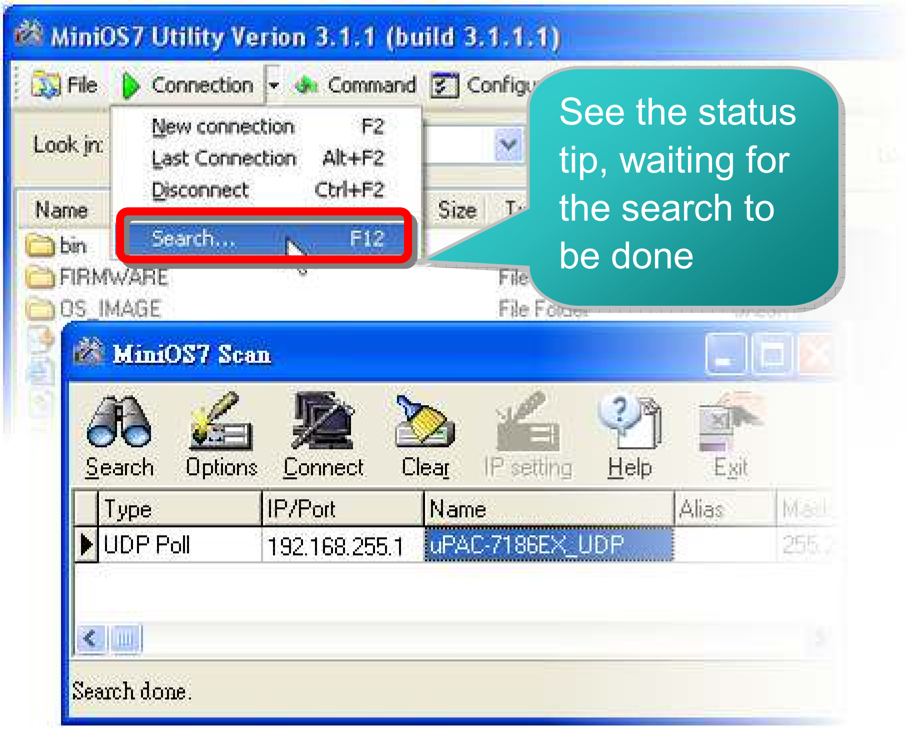
4. Using MiniOS7 Utility

The location of the MiniOS7 Studio:
CD:\Napdos\minios7\utility\minios7_utility\
ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/minios7/utility/minios7_utility/

Step 1: Run the **MiniOS7 Utility**

Double-click the MiniOS7 Utility shortcut on your desktop.

Step 2: Press "**F12**" or choose "**Search**" from the "**Connection**" menu

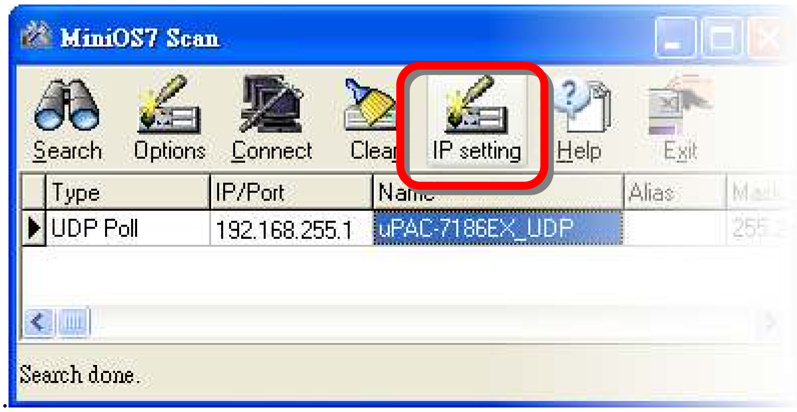


See the status tip, waiting for the search to be done

Type	IP/Port	Name	Alias	Mac
UDP Poll	192.168.255.1	uPAC-7186EX_UDP		255.2

Search done.

Step 3: Choose the module name and then choose "**IP setting**" from the toolbar



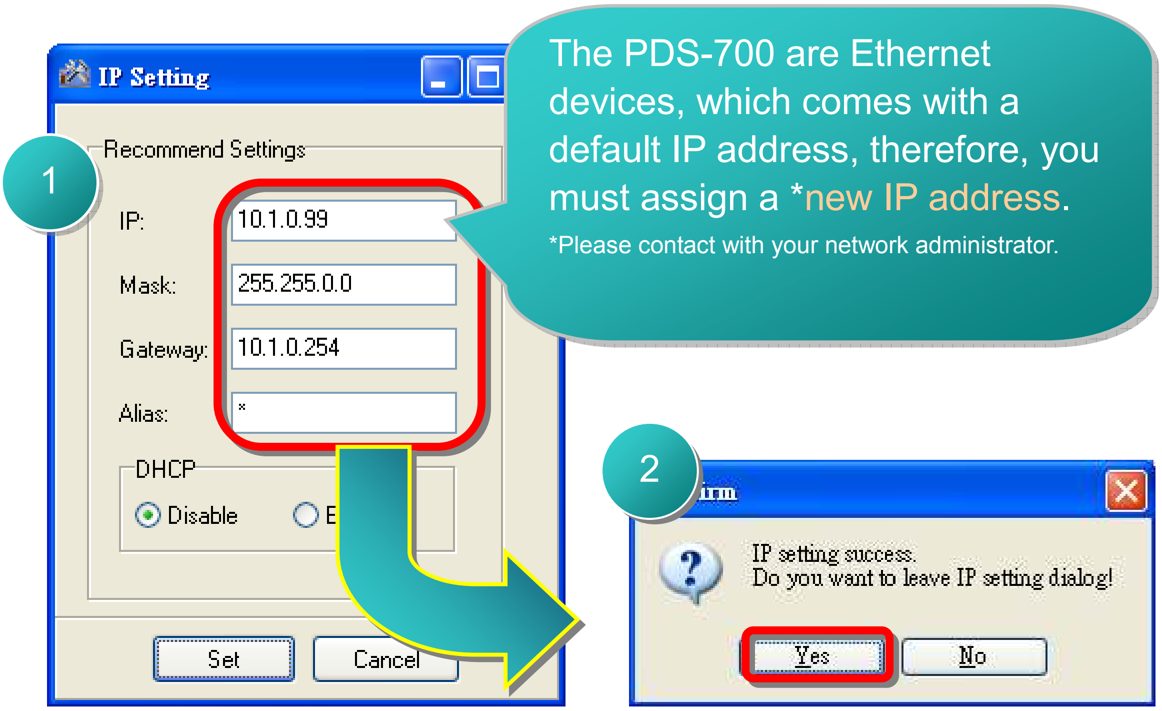
Type	IP/Port	Name	Alias	Mac
UDP Poll	192.168.255.1	uPAC-7186EX_UDP		255.2

Search done.

Step 4: Assign a new IP address and then choose “**Set**” button and press the **Yes** button

The factory default IP settings in PDS are as follows:

Item	**Default**
IP Address	192.168.255.1
Subnet Mask	255.255.0.0
Gateway	192.168.0.1



The PDS-700 are Ethernet devices, which comes with a default IP address, therefore, you must assign a *new IP address.

*Please contact with your network administrator.

1

IP Setting

Recommend Settings

IP: 10.1.0.99

Mask: 255.255.0.0

Gateway: 10.1.0.254

Alias: *

DHCP ● Disable ○ E

2

irm

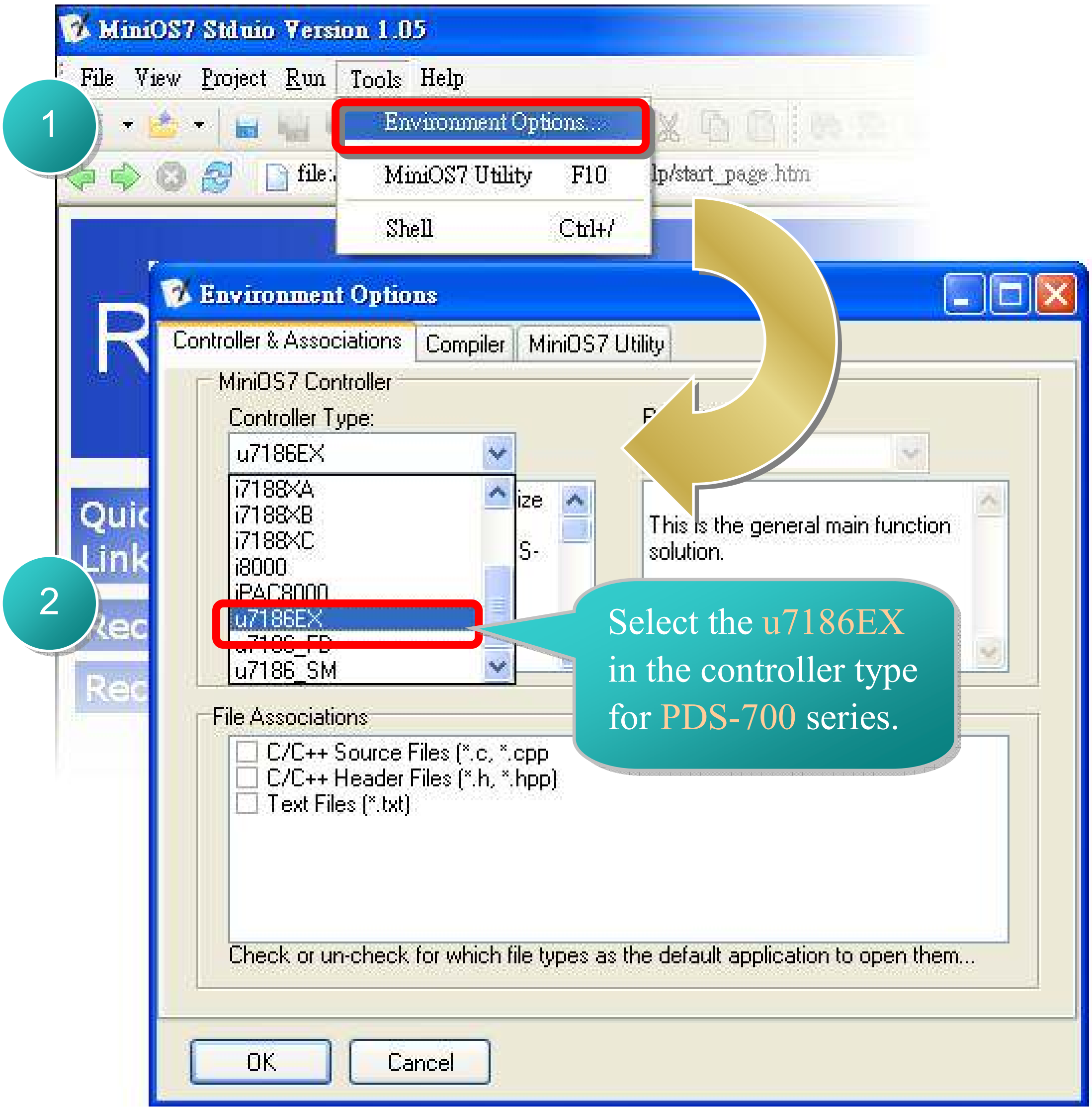
IP setting success.
Do you want to leave IP setting dialog!

5. Using MiniOS7 Studio

The location of the MiniOS7 Studio:
CD:\Napdos\minios7\Studio\
http://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/minios7/studio/

Step 1：Click the **Tools → Environment Options**

Step 2：Select the **Controller & Associations** tab and set the **Controller Type**



Controller Type: u7186EX

i7188XA
i7188XB
i7188XC
i8000
iPAC8000
u7186EX
u7186_FD
u7186_SM

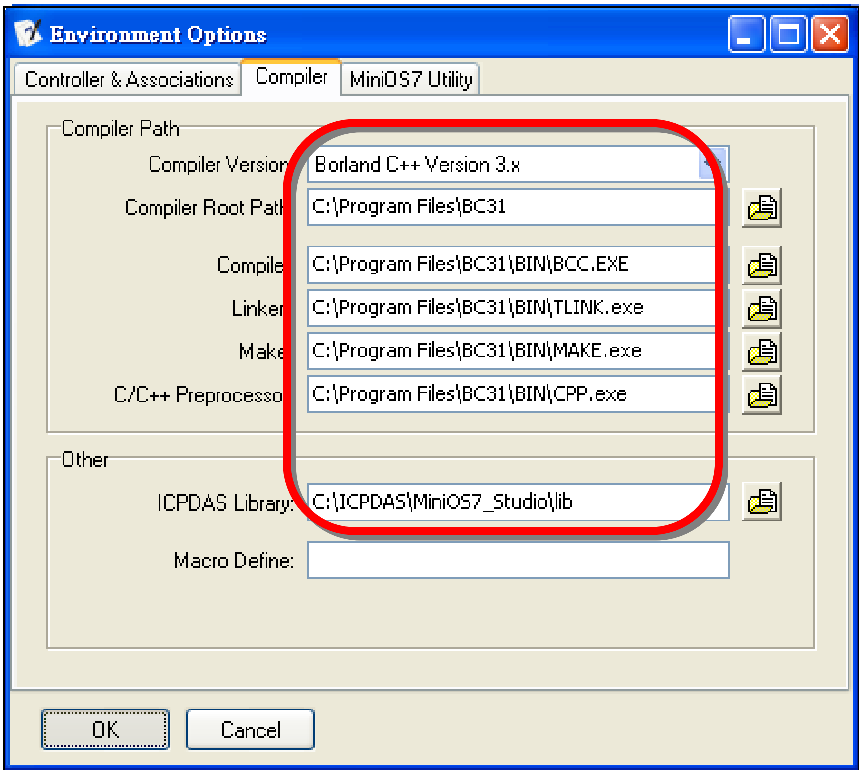
This is the general main function solution.

☐ C/C++ Source Files (*.c, *.cpp
☐ C/C++ Header Files (*.h, *.hpp)
☐ Text Files (*.txt)

Check or un-check for which file types as the default application to open them...

Select the u7186EX in the controller type for PDS-700 series.

Step 3：Select the **Compiler** tab and set the **Compiler Path**



Compiler Path

Compiler Version: Borland C++ Version 3.x

Compiler Root Path: C:\Program Files\BC31

Compiler: C:\Program Files\BC31\BIN\BCC.EXE

Linker: C:\Program Files\BC31\BIN\TLINK.exe

Make: C:\Program Files\BC31\BIN\MAKE.exe

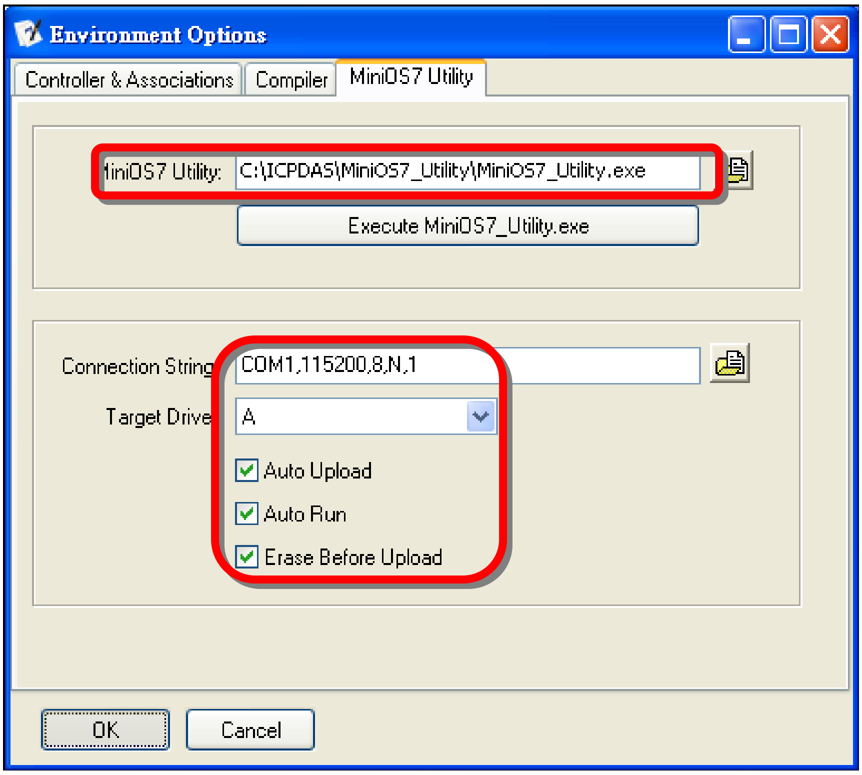
C/C++ Preprocessor: C:\Program Files\BC31\BIN\CPP.exe

Other

ICPDAS Library: C:\ICPDAS\MiniOS7_Studio\lib

Macro Define:

Step 4：Select the **MiniOS7 Utility** tab and set **MiniOS7 Utility**, **Connection String**



MiniOS7 Utility: C:\ICPDAS\MiniOS7_Utility\MiniOS7_Utility.exe

Connection String: COM1,115200,8,N,1

Target Drive: A

☑ Auto Upload

☑ Auto Run

☑ Erase Before Upload

6. Executing your first firmware

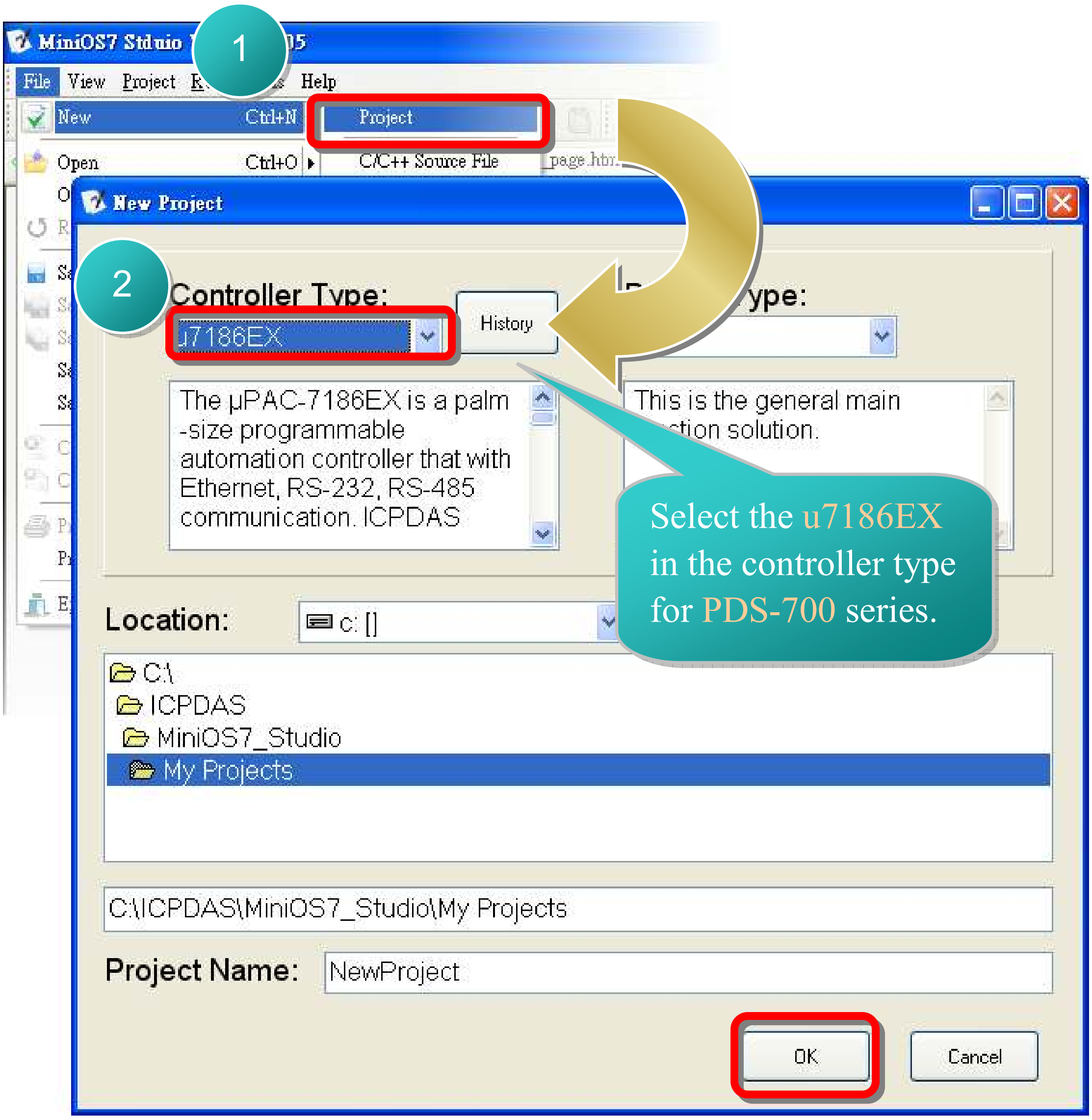
6.1：Creating the new project

Click the **File → New → Project**

The workspace window shows the new project files.

6.2：Setting the Controller Type

Choose the properly controller type for your production.



Controller Type: u7186EX

The µPAC-7186EX is a palm -size programmable automation controller that with Ethernet, RS-232, RS-485 communication. ICPDAS

ype:

This is the general main ction solution.

Select the u7186EX in the controller type for PDS-700 series.

Location: c: []

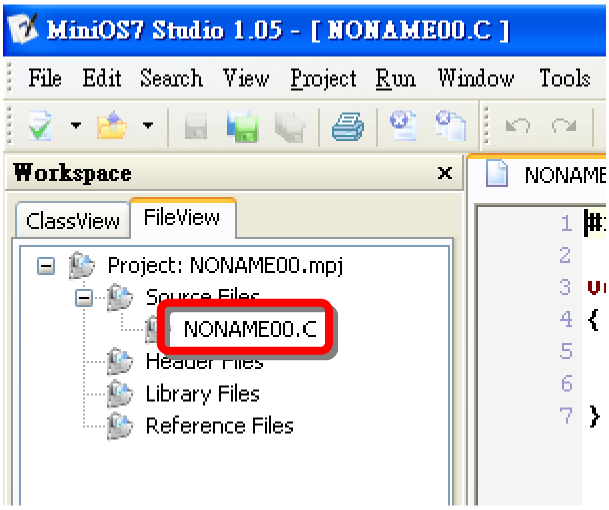
C:\
ICPDAS
MiniOS7_Studio
My Projects

C:\ICPDAS\MiniOS7_Studio\My Projects

Project Name: NewProject

6.3：Adding the source file

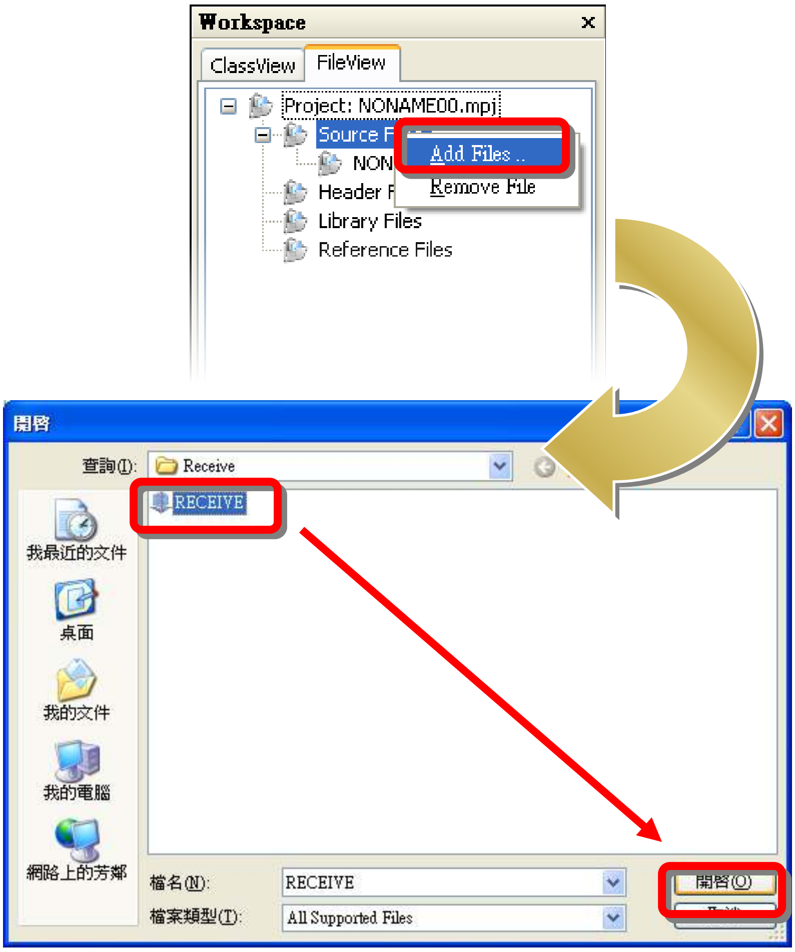
Double click the **NONAME00.C** source file for programming.



Of course you can add the existing files to the project.

Right click the **Source Files → Add Files**

Choose your file and press the **Open** button.



You can also use the demo source code below.

Please connect PDS Port 3 and Port 4.

```
#include <string.h>
#include <stdio.h>
#include "7186e.h"

// Uses COM port to receive data with 0x0D [Enter].
int Receive_Data(int iPort,unsigned char* cInBuf,unsigned long lTimeout);

void main(void)
{
        int iLength, iValue, showLED;
        int iCommandCOMPort=1, iCommandCOMPort2=3, iCommandCOMPort3=4 ;
        float fValue=0;
        unsigned char cData[100];

         InitLib();
         InstallCom(iCommandCOMPort, 115200, 8, 0, 1);
         InstallCom(iCommandCOMPort2, 115200, 8, 0, 1);
         InstallCom(iCommandCOMPort3, 115200, 8, 0, 1);
```

Port 3 send data to Port 4, and Port 4 can receive and show it to screen via Port 1.

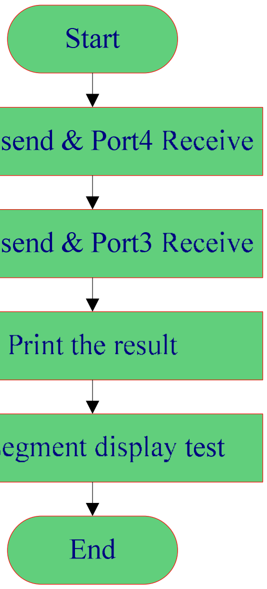
```
        ToComStr(iCommandCOMPort, "Port3 to Port4 (integer): ");
        ToComStr(iCommandCOMPort2, "12345\r");
        Delay(10);
        iLength=Receive_Data(iCommandCOMPort3, cData, 10000);
        if(iLength>0)
        {
        sscanf(cData, "%d", &iValue);
        printCom(iCommandCOMPort, "\r\nPort4 Receive Value=%d\r\n", iValue);
        }
        else
        {
        printCom(iCommandCOMPort, "Keyin timeout!\r\n");
        }
```

Port 4 send data to Port 3, and Port 3 can receive and show it to screen via Port 1.

```
         printCom(iCommandCOMPort, "\r\nPort4 Send 56.7 to Port3 (float): ");
         ToComStr(iCommandCOMPort3, "56.7 \r");
         Delay(10);
        iLength=Receive_Data(iCommandCOMPort2, cData, 10000);
        if(iLength>0)
        {
                sscanf(cData, "%f", &fValue);
                printCom(iCommandCOMPort, "\n\rPort3 Receive Value=%f\n\r", fValue);
        }
        else
        {
                printCom(iCommandCOMPort, "Keyin timeout!\n\r");
        }
          Delay(10);    // Wait for all data is transmited to COM port
          RestoreCom(iCommandCOMPort);
         RestoreCom(iCommandCOMPort2);
         RestoreCom(iCommandCOMPort3);

         Init5DigitLed();
         Show5DigitLed(1,1);
         Show5DigitLed(2,2);
         Show5DigitLed(3,3);
         Show5DigitLed(4,4);
         Show5DigitLed(5,5);
        Delay(1000);
}
```

```
int Receive_Data(int iPort, unsigned char* cInBuf, unsigned long lTimeout)
{
        unsigned char cChar;
        int iIndex=0;
        unsigned long lStartTime;

        lStartTime=GetTimeTicks();
        for(;;)
        {
                while(IsCom(iPort))         // check COM port
                {
                        cChar=ReadCom(iPort);
                        if(cChar=='\r')         // the terminal char is 0x0D
                        {
                                cInBuf[iIndex]=0;       // Add the zero end to the data
                                return iIndex;          // return data length
                                                            // (doesn't include the cTerminator)
                        }
                        else
                                cInBuf[iIndex++]=cChar;

                        lStartTime=GetTimeTicks();    // refresh data timeout
                }
                if((GetTimeTicks()-lStartTime)>=lTimeout)
                {
                        cInBuf[iIndex]=0;       // Add the zero end to the data
                        return -1;    // receive data timeout
                }

                RefreshWDT();
        }
}
```

The flowchart of this demo program

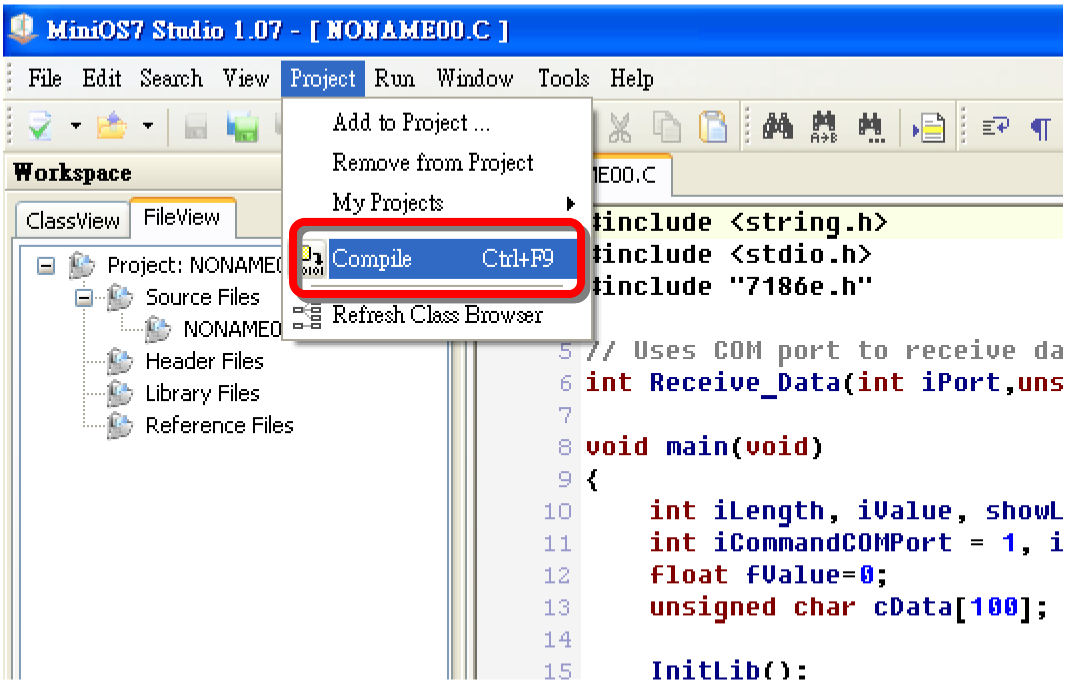


The firmware prints the result string to the Port1 of the PDS.

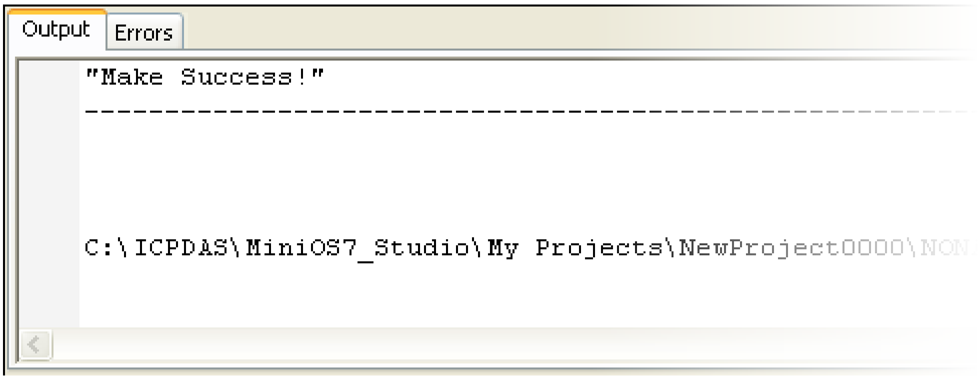
The 7188XW (PC) gets the string via the serial port and then shows it in the screen.

6.4：Compiling the Project

Click the **Project → Compile**



```
#include <string.h>
#include <stdio.h>
#include "7186e.h"

// Uses COM port to receive da
int Receive_Data(int iPort,uns

void main(void)
{
    int iLength, iValue, showL
    int iCommandCOMPort = 1, i
    float fValue=0;
    unsigned char cData[100];

    InitLib();
```

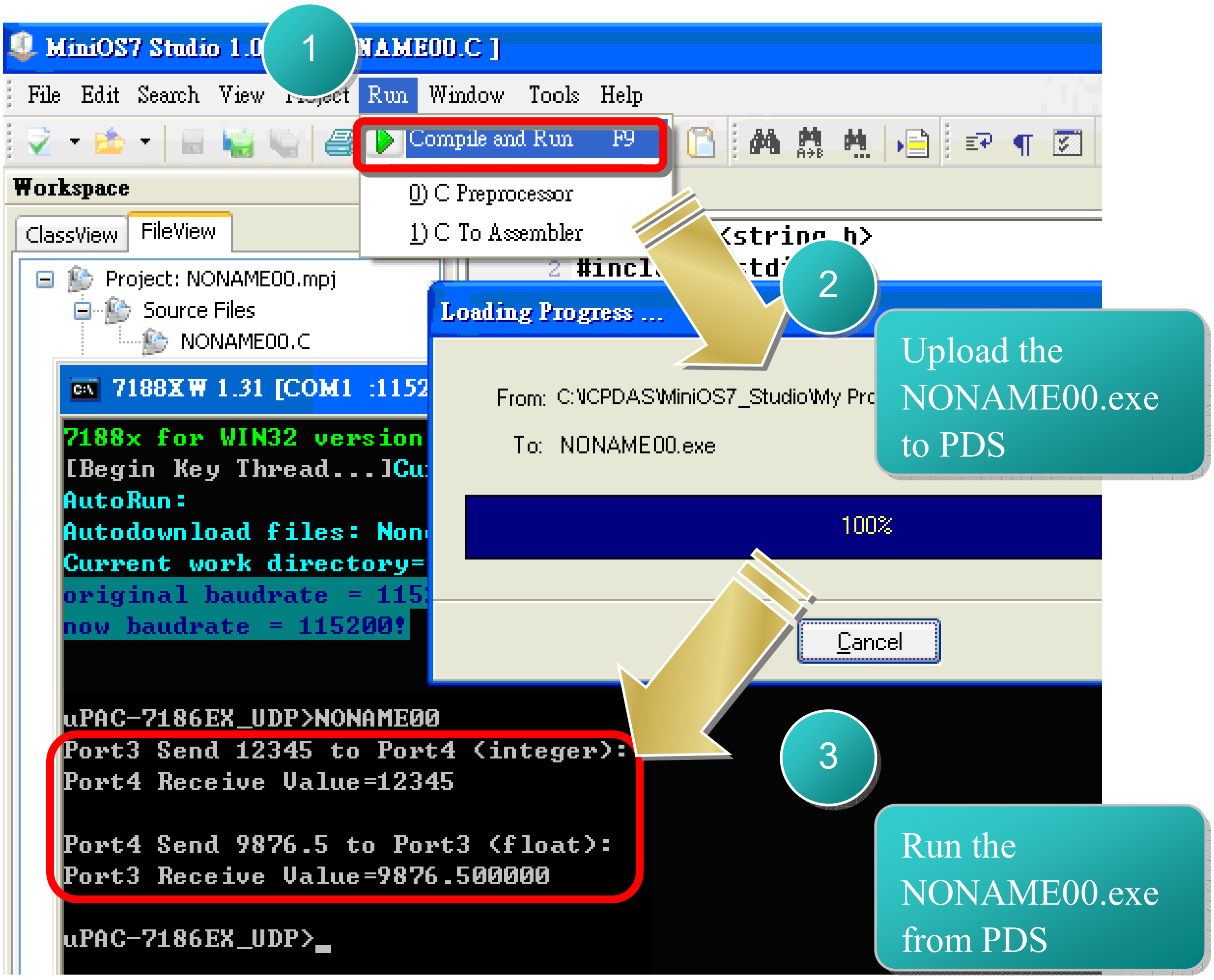


```
"Make Success!"
--------------------------------------------------------------



C:\ICPDAS\MiniOS7_Studio\My Projects\NewProject0000\NON
```

6.5：Uploading and executing

Click the **Run → Compile and Run**

Upload the exe file to PDS and execute.



From: C:\ICPDAS\MiniOS7_Studio\My Pro

To: NONAME00.exe

```
7188XW 1.31 [COM1 :1152
7188x for WIN32 version
[Begin Key Thread...]Cu
AutoRun:
Autodownload files: Non
Current work directory=
original baudrate = 115
now baudrate = 115200!

uPAC-7186EX_UDP>NONAME00
Port3 Send 12345 to Port4 (integer):
Port4 Receive Value=12345

Port4 Send 9876.5 to Port3 (float):
Port3 Receive Value=9876.500000

uPAC-7186EX_UDP>_
```

Upload the NONAME00.exe to PDS

Run the NONAME00.exe from PDS

7. Additional Information

PDS-700 Document：

CD:\Napdos\PDS/PDS-700\document\
http://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/pds/pds-700/document/

PDS-700 firmware：

CD:\ Napdos\PDS\PDS-700\VxComm\Server(PDS)\
http://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/pds/pds-700/vxcomm/serve

More demo programs：

CD: \NAPDOS\PDS\PDS-700\demo
http://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/7186e/demo/

The ICP DAS Web Site

http://www.icpdas.com

- Technical support
- Supplies and ordering information
- Information to enhance the product
- FAQ & Application story

Contact Us

Service@icpdas.com