

7188E Series

I-7188EF-016 Software Development Manual

Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assume no liability for any damage consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright 2005 by ICP DAS. All rights are reserved.

Trademark

The names used for identification only may be registered trademarks of their respective companies.

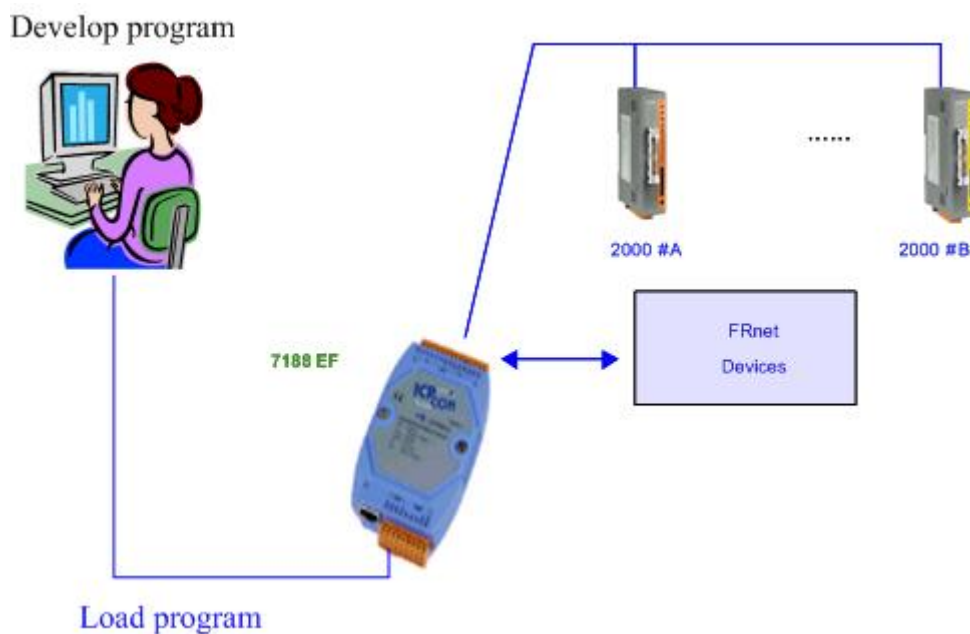
Table of Contents

1. Introduction	3
2. Developing a program?	4
2.1 LINKING TO I-7188EF-016.....	5
2.2 COMPILER AND LINKER FOR C PROGRAMMING	6
2.3 UPLOAD THE PROGRAM TO I-7188EF-016	10
3. Demo programs for the I-7188EF-016	11
3.1 USING THE COM PORT TO CONTROL FR-2000 SERIES MODULES.....	12
3.2 USING AN ETHERNET TO CONTROL FR-2000 SERIES MODULE.....	20
Appendix A: 7188xw command set and hot-key	22
Appendix B: Minios7	24

Introduction

In addition to being used as an Ethernet/Serial converter, I-7188EF-016 series modules can also be programmed to be used as an autonomous controller to control distributed FR-2000 I/O modules via FRnet.

This manual describes how to develop programs for the I-7188EF-016. I-7188EF-016 is an embedded controller which, with aid of C language, can help you develop your own programs. Included with the 7188E are many demos and libraries. I-7188EF-016 can be used to access devices via Ethernet/Internet, FRnet port or RS-232/485.



1. Developing a program?

The steps of quick start are as follows:

1. Wire the connection of I-7188/I-7188X/I-7188E series module
Users can follow steps to do hardware installation.

2. Compiler and Linker for C programming

To develop programs for 7188/7188X/7188E/8000 series, you can use the following compilers can be used:

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

The free TC 2.01 and TC++ 1.01 compilers can be downloaded from Borland's web site.

Web site: <http://community.borland.com/museum/>

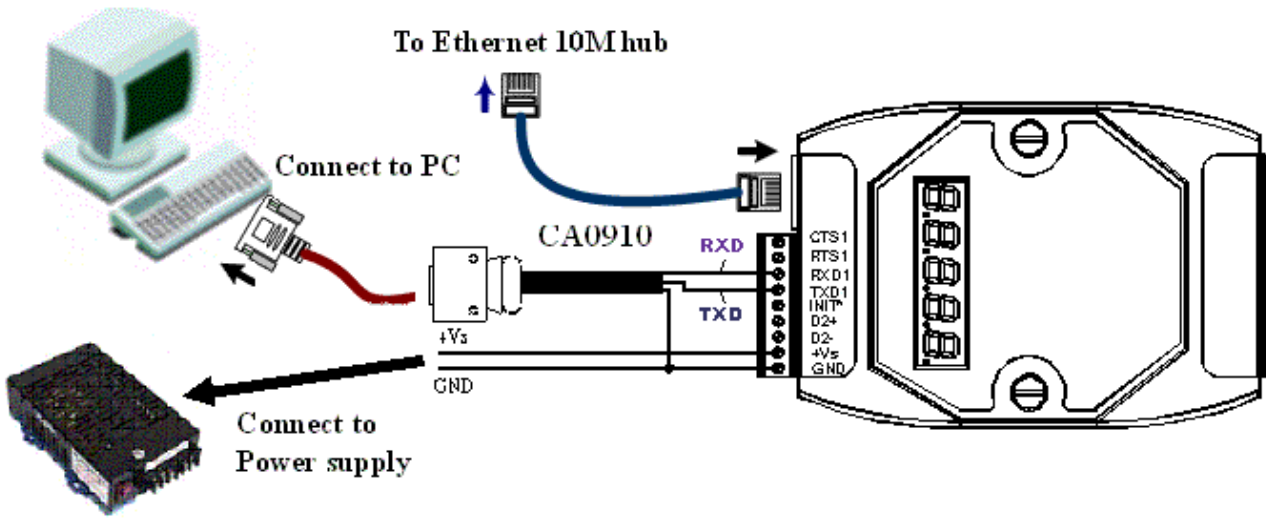
3. Load the program onto the I-7188EF-016

When you finish above-mentioned paths, you can load program to the I-7188EF-016 and run it.

1.1 Linking to I-7188EF-016

Step 1: Connect the CA0910 download-cable from the I-7188EF-016 to COM1/2 of the PC, as shown in the diagram below.

Step 2: Apply power (+Vs, GND) to the I-7188EF-016, +Vs can range from +30V to +10V DC.



Step 3: Check that the 5-digits of the 7-segment LED continuously show the following:

Hour.Minute.Second

00000->00001->00002.....

Note:

Only display versions of the I-7188EF-016 module include a 5-digit 7- segment LED.

1.2 Compiler and Linker for C programming

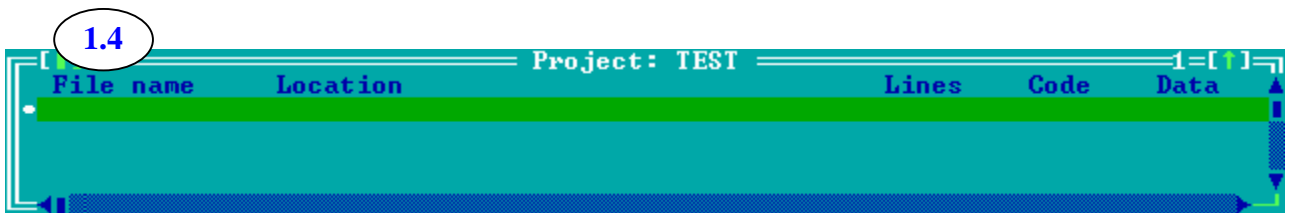
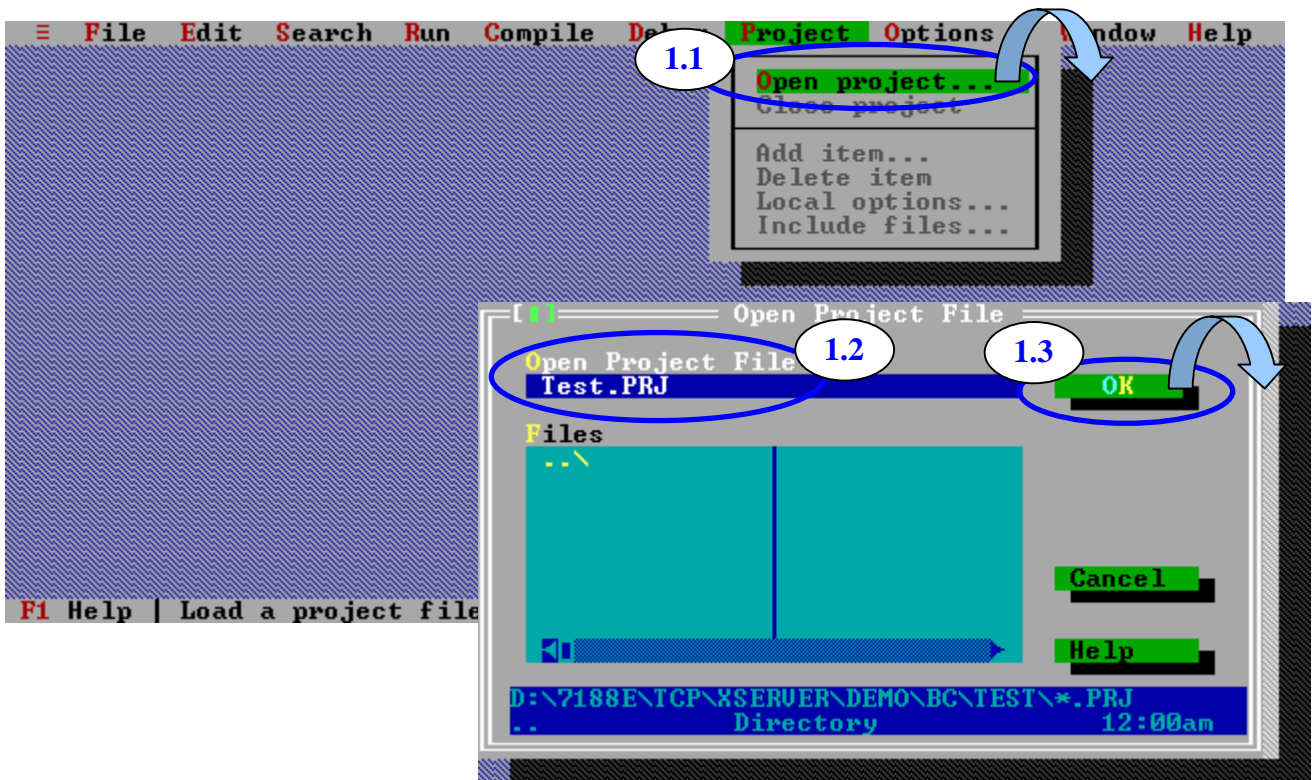
Applications must be written using the C language. The following compilers may be used to compile your program

TC, TC++, BC++, MSC or MSVC++.

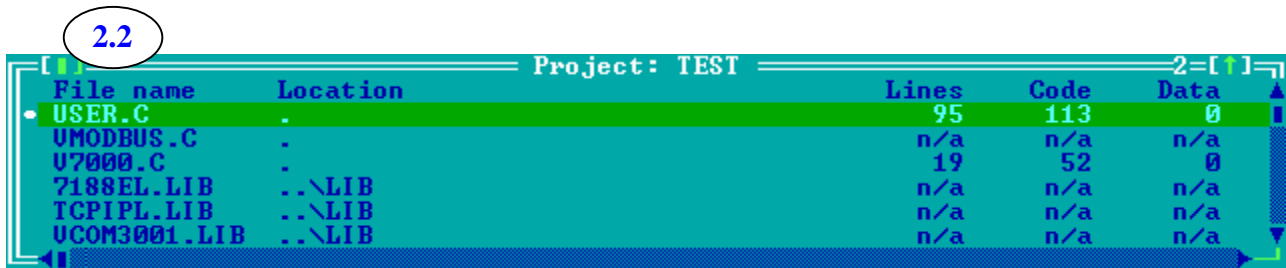
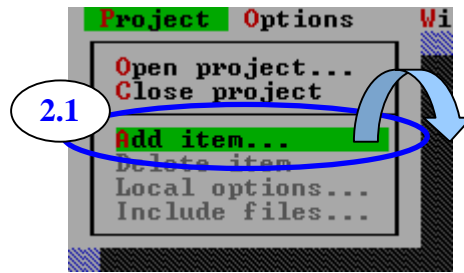
ICPDAS suggests that the BC3.1 compiler is used as we offer a demo/program library that was built using the BC3.1 compiler.

Please follow the steps below for details of how to use BC++ 3.1's IDE to compile a project.

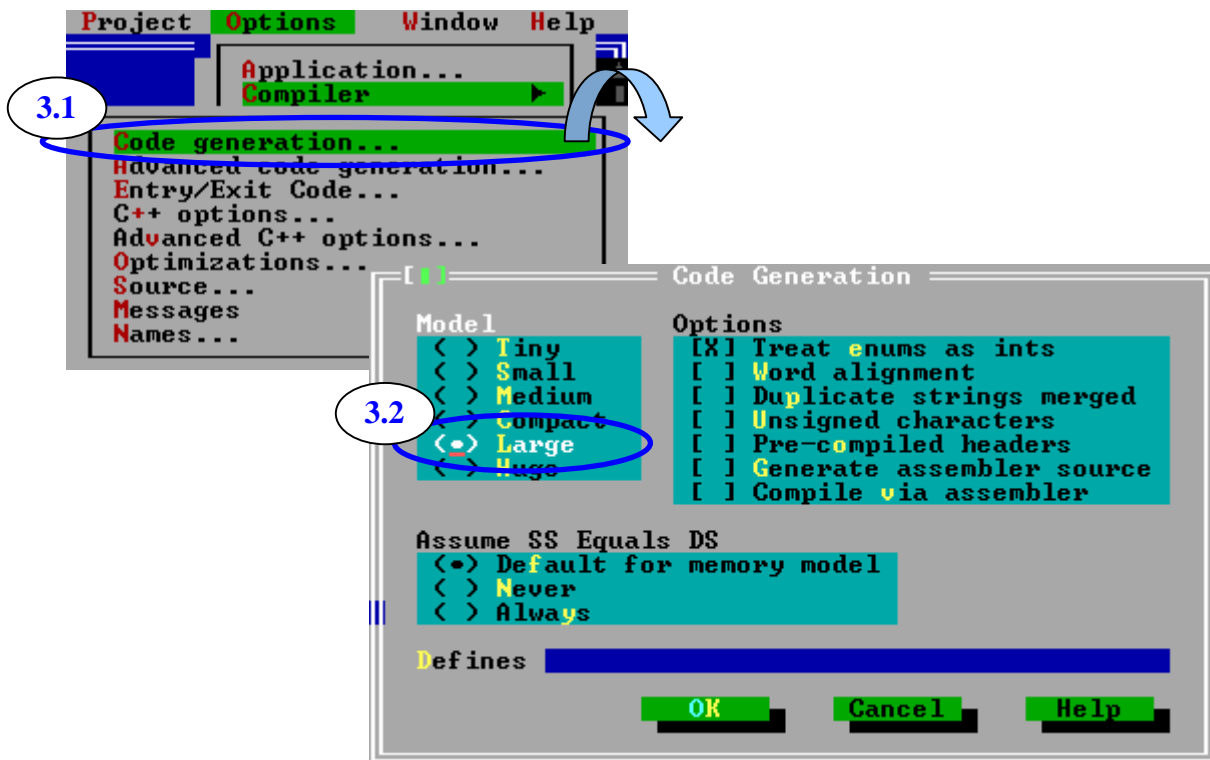
Step 1: Create a new project.



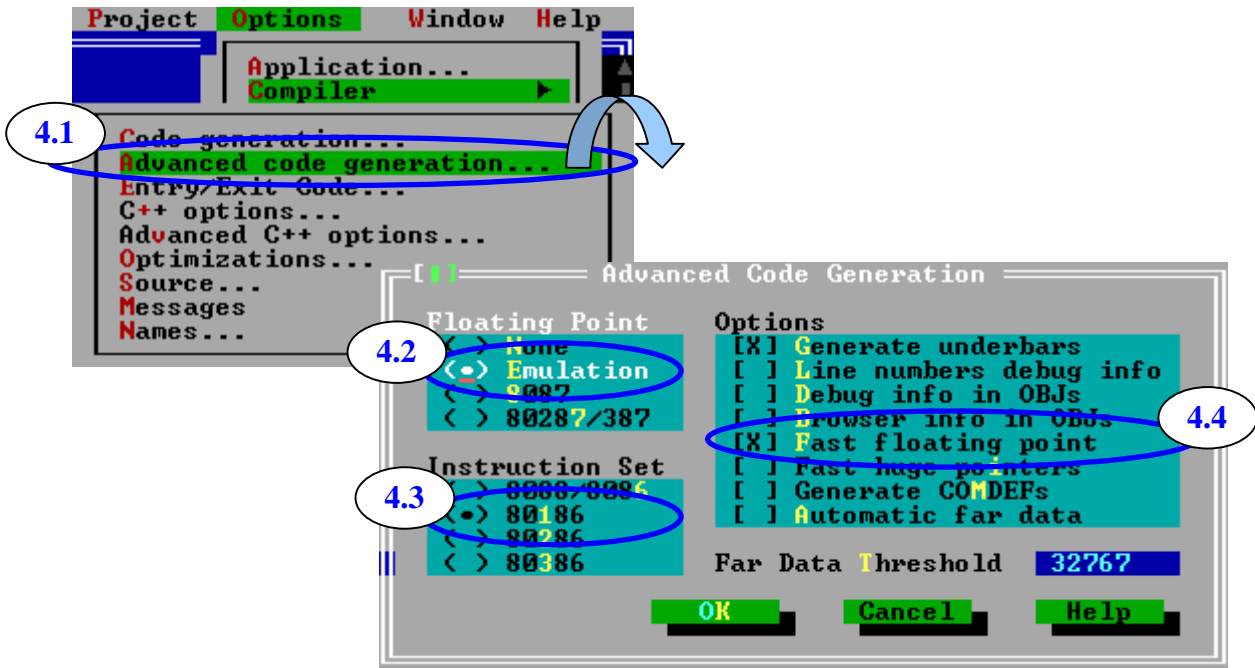
Step 2: Add all necessary files into the project.



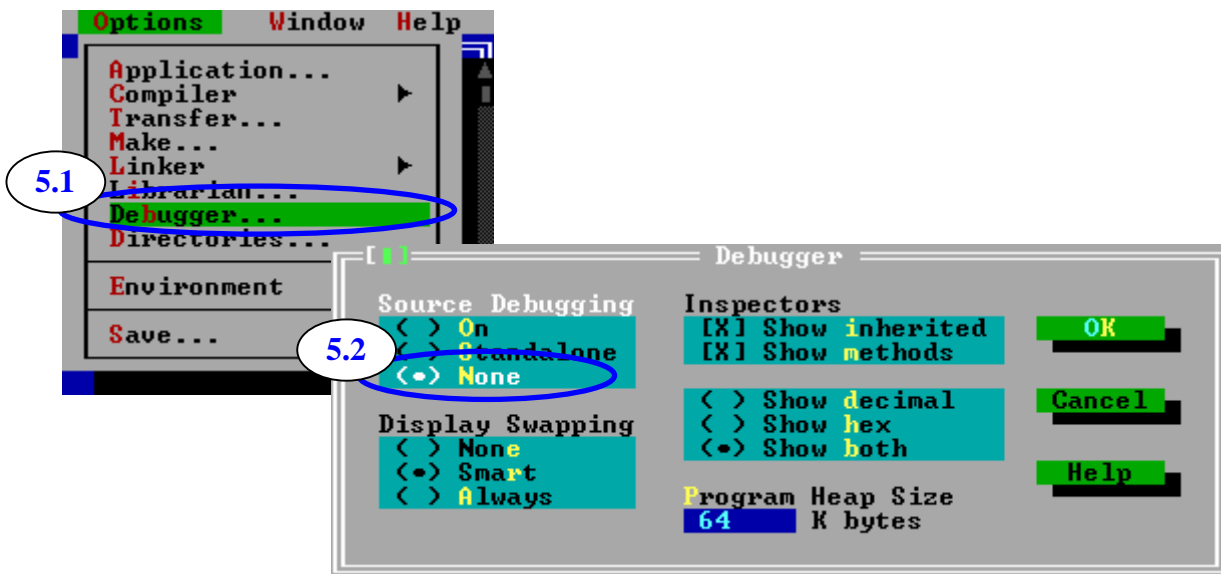
Step 3: Set the Code generation options.



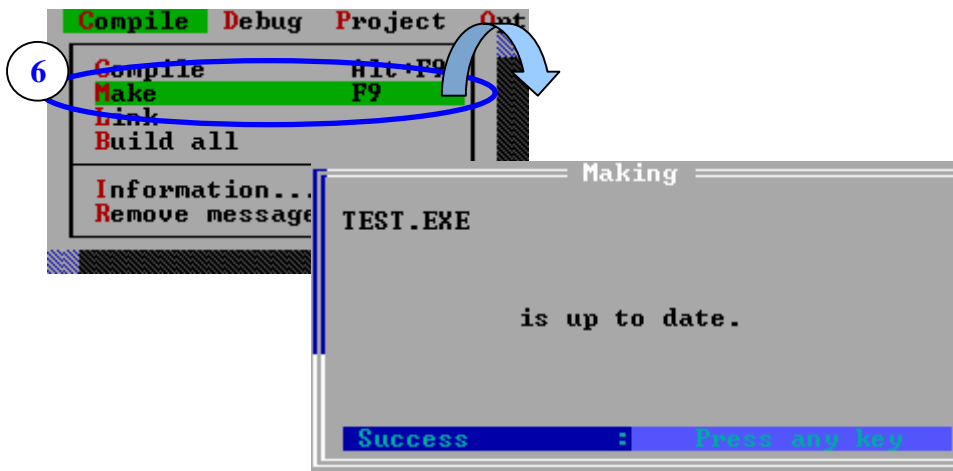
Step 4: Set the Advanced code generation options.



Step 5: Set the Debugger options.



Step 6: Make the project.



1.3 Upload the program to I-7188EF-016

7188xw.exe is a terminal utility running on a PC for I-8000 and I-7188 series controllers. It sends out the data that user key-in to COM port and show the data received from COM port on the screen of PC.

The 7188xw.exe file can be found at:

1. <ftp://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/minios7/utility/>

2. 8000 CD:\napdos\minios7\utility\

Step 1: Download the 7188xw.exe file and run it.

```
G:\FR_net\7188EF\Demo\BC\7EF_DI>7188xw /c1
7188x for WIN32 version 1.26 (10/19/2004)[By ICPDAS. Tin.]
[Begin Key Thread...][Current set: Use COM1 115200,N,8,1
AutoRun:
Autodownload files: None
Current work directory="G:\FR_net\7188EF\Demo\BC\7EF_DI"
original baudrate = 115200!
now baudrate = 115200!
7188E_UDP>
```

Note:

Please refer to Appendix B for more details regarding the 7188xw utility.

Step 2: Uploads the user programs from the host PC into the 7188, 7188X, 7188E and 8000 families. Please load “load” to upload file.

```
7188E_UDP>load
File will save to 82C1:0000
StartAddr-->8000:2C0F
Press ALT_E to download file!
```

2.1

Step 2.1: Press ALT +E to download the file.

Step 2.2: Type the filename and press the Enter key.

```
Input filename:frdi.exe
Load file:frdi.exe[crc=FEA1,0000]
Send file info. total 44 blocks
Block 44
Transfer time is: 1.922000 seconds
7188E_UDP>
```

2.2

Step 3: Type Run and press Enter.

```
7188E_UDP>run
/*****/
/*      i7188EF016 DI demo      */
/*                                  */
/*      [Feb,14,2005]           */
/*****/

0> Read DI
1> Read Single DI channel
2> Quit
Please choose(0~2):
```

Note:

The “Load” and “run” commands are part of the MiniOS7 command set. For a detailed explanation of the MiniOS7 command sets, please refer to Appendix B.

2. Demo programs for the I-7188EF-016

The source code for demo programs for both DI and DO applications is included on the enclosed CD. It is recommended that these programs be edited, modified and used as the basis for custom applications.

2.1 Using the COM port to control FR-2000 series modules

Use the addresses in the table below to read the DI value of each group DI address:

Group	Address	Reading
0	0x0	inpw(0x0)
1	0x2	inpw(0x2)
2	0x4	inpw(0x4)
3	0x6	inpw(0x6)
4	0x8	inpw(0x8)
5	0xA	inpw(0xA)
6	0xC	inpw(0xC)
7	0xE	inpw(0xE)

An example of a simple application:

```
Void main()
{
    Print("Group 0=%X", inpw(0x0));
}
```

The code shown above will retrieve the DI value from Group 0. For more details of the source code, refer to the following demo program.

Note:

The location of the demo program is:

8000 CD:\NAPDOS\FR_Net\7188EF\Demo\BC\7EF_DI

ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/fr_net/7188ef/demo/bc/7ef_di

```
#include<conio.h>
#include<stdlib.h>
#include "..\lib\7188e.h"

#define MAX_Group 8
unsigned int NumberInput(int mode)
{
/*
Return: return an integer number
*/
unsigned char sTemp[20];
unsigned int iRet;

LineInput(sTemp,20);
sscanf(sTemp,"%i",&iRet);
return iRet;
}

void main()
{
int iGroup,iRet,iChannel;
unsigned int uValue;

if(!Is7188e())
{
Print("\n This program must run on I-7188EF with MiniOS7");
return;
}

Print("/*****\n\r");
Print("/* i7188EF016 DI demo *\n\r");
Print("/* *\n\r");
Print("/* [Feb,14,2005] *\n\r");
Print("/*****\n\r");
```

```

Print("\n\r");
LedOn();    //Led on

while(1)
{
    Print(" 0) Read DI\n\r");
    Print(" 1) Read Single DI channel\n\r");
    Print(" 2) Quit\n\r");
    Print(" Please choose(0~2):");

    iRet=NumberInput(); //Input in Dec format
    Print("\r\n");

    if(iRet>2)
        Print("Wrong number!!\r\n");
    else if(iRet==2) //Quit the program
        exit(0);
    else
    {
        Print("Input FRnet DI Group number:(0~7)");
        iGroup=NumberInput();

        if(iGroup>=MAX_Group)
        {
            Print("Wrong Group number!!\r\n");
            continue;
        }
        else
        {
            uValue=inpw(iGroup*2);

            switch(iRet)
            {
                case 0: //Read Group
                    Print("FRnet DI value of Group %d = %X\n\r",iGroup,uValue);
                    break;
                case 1: //Read single channel

```

```

        Print("Input channel number:(0~15)");
        iChannel=NumberInput();

        if(iChannel>=16)
        {
            Print("Wrong Channel number\r\n");
            continue;
        }
        else
            Print("Frnet DI channel%d of Group %d =%X\r\n",iChannel,
                iGroup,(uValue>>iChannel)&01);
        break;
    }
    Print("Press any key to continue!!");
    Getch();
    Print("\r\n");
}
}
}
}

```

Use the addresses in the table below to write the DO value of each group.
DO address:

Group	Address	Write
0	0x0	outpw(0x0,Value)
1	0x2	outpw(0x2,Value)
2	0x4	outpw(0x4,Value)
3	0x6	outpw(0x6,Value)
4	0x8	outpw(0x8,Value)
5	0xA	outpw(0xA,Value)
6	0xC	outpw(0xC,Value)
7	0xE	outpw(0xE,Value)

Value => DO value (0000~FFFF)

An example of a simple application:

```
Void main()
{
    outpw(0x0, 0xffff);
}
```

The code shown above will write the address 0xffff to Group 0
For more details of the program, refer to the source code below.

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include "..\lib\7188e.h"
#define MAX_Group 8

unsigned int NumberInput(void)
{
    /*
    Return:  return an integer number
    */
    unsigned char sTemp[20];
    unsigned int iRet;

    LineInput(sTemp,20);
    sscanf(sTemp,"%i",&iRet);
    return iRet;
}

unsigned int guDOvalue;
void main()
{
    int iGroup,iRet,iChannel;
    unsigned int uValue;

    if(!Is7188e())
```



```

{
    Print("\n This program must run on I-7188e with MiniOS7");
    return;
}
Print("/*****\n\r");
Print("/*      i7188EF016 DO demo      *\n\r");
Print("/*                                  *\n\r");
Print("/*      [Feb,14,2005]      *\n\r");
Print("/*****\n\r");
Print("\n\r");
LedOn(); //Led on

while(1)
{
    Print(" 0) Write DO\n\r");
    Print(" 1) Write Single DO channel\n\r");
    Print(" 2) Quit\n\r");
    Print(" Please choose(0~2):");

    iRet=NumberInput();
    Print("\r\n");

    if(iRet>2)
        Print("Wrong number!!\r\n");
    else if(iRet==2) //Quit the program
        exit(0);
    else
    {
        Print("Input FRnet DO Group number:(0~7)");
        iGroup=NumberInput();
        if(iGroup>=MAX_Group)
        {
            Print("Wrong Group number!!\r\n");
            continue;
        }
        else
        {
            switch(iRet)

```

```

        {
            case 0: //Write Group
                Print("Input DO value:(0000~FFFF)");
                uValue=NumberInput();
                Print("Write 0x%X to FRnet DO of Group %d\n\r",uValue,
                    iGroup)
                break;
            case 1: //write single channel
                Print("Input channel number:(0~15)");
                iChannel=NumberInput();
                if(iChannel>=16)
                {
                    Print("Wrong Channel number\r\n");
                    continue;
                }
                else
                {
                    Print("Input value:(0 or 1)");
                    //Group number:Input in Dec format
                    uValue=NumberInput()&0x1;
                    Print("Write %X to DO channel%d of Group %d\n\r",uValue,
                        iChannel,iGroup);
                    if(!uValue)
                        uValue= guDOvalue & ~(1<<iChannel);
                    else
                        uValue= guDOvalue |(1<<iChannel);
                }
                break;
        }
        outpw(iGroup*2,uValue); //Output value
        guDOvalue=uValue;
        Print("Press any key to continue!!");
        Getch();
        Print("\r\n");
    }
}
}
}
}
}

```



Note:

The location of the demo program is:

8000CD\NAPDOS\FR_Net\7188EF\Demo\BC\7EF_DO

ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/fr_net/7188ef/demo/bc/7ef_do

2.2 Using an Ethernet to control FR-2000 series

module

Commands can be transmitted to the I-7188EF-016 via Ethernet, allowing control of the FR-Net I/O module.

For an example of how to control the FR-Net I/O module, refer to the demo program. Refer to Section 3.1 for details of how to set the address. A brief sample of the source code is shown below:

```
char cVal[80];
int UserCmd(unsigned char *Cmd,unsigned char *Response)
{
    // user's command interpreter
    // refer to all demo
    // Maximum length of Cmd is 1458 bytes.
    // Maximum length of Response is 1024 bytes.

    int iAddr;
    WORD wVal;

    strlwr(Cmd);
    if(Cmd[0]!='r')    //Read DI/DO readback
    {
        sscanf(Cmd+2,"%x",&iAddr);
        printCom1("string:%s, iAddr=%d\r\n",Cmd,iAddr); //Debug message
        switch(Cmd[1])
        {
            case 'i': // inp command: i 0 or IO
                wVal=inpw(iAddr*2);    //Refer to sec. 3.1
                sprintf(cVal,"%x",wVal);
                break;
            case 'o':
                sprintf(cVal,"%x",gwOutVal);
                break;
            default:
                return 0;    // return error
        }
    }
};
```

```
}
else if(Cmd[0]=='w' && Cmd[1]=='o') //write DO
{
    sscanf(Cmd+2,"%x %x",&iAddr,&wVal);
    sprintf(cVal,"%x",wVal);
    outpw(iAddr*2,wVal); //Refer to sec. 3.1
    gwOutVal=wVal;
}
else
    return 0; // return ERROR

strcpy(Response,cVal); // return I/O value
return 1;
}
```

Note:

The location of the Xserver demo program is:

8000 CD:\NAPDOS\7188e\tcp\Xserver\Demo\Bc3223\XDemo71

ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/7188e/tcp/xserver/demo/bc3223/xdemo71/

Appendix A: 7188xw command set and hot-key

7188XW.EXE Utility for Host-PC

7188xw.exe is the Win32 version of 7188x.exe.

7188xw.exe: Supports RS-232 COM ports using USB and PCMCIA interfaces.

Command line options for 7188xw.exe:

Option	Description
/c#	Uses the PC COM#
/b#	Sets the Baud Rate for the PC COM port (default is 115200)
/s#	Sets the number of rows to be displayed on the screen (default is 25, max. is 50)

Hot-key of 7188xw.exe:

Command	Description
F1	Shows help messages for 7188xw.exe
Alt_F1	Shows the Traditional Chinese (Big5) help messages for 7188xw.exe
Ctrl_F1	Shows the Simplified Chinese (GB2312) help messages for 7188xw.exe
Alt_1	Uses PC's COM1
Alt_2	Uses PC's COM2
Alt_3	Uses PC's COM3
Alt_4	Uses PC's COM4
Alt_5	Uses PC's COM5
Alt_6	Uses PC's COM6
Alt_7	Uses PC's COM7
Alt_8	Uses PC's COM8
Alt_9	Uses PC's COM9
Alt_A	Switches between normal mode and ANSI-Escape-code-support mode
Alt_C	Switches to command mode. Supports the following commands: b#: Sets a new Baud Rate for the PC COM ports. c#: Uses the PC COM#. n/e/o: Sets the parity to none/even/odd. 5/6/7/8: Sets the data bit to 5/6/7/8.

	p#: Sets the PC working directory. q: Quits the command mode.
Alt_D	Sets the date of the RTC to the current PC date.
Alt_T	Sets the time of the RTC to the current PC time
Alt_E	This command is used to download files into memory. Only use the command after the message "Press ALT_E to download file!" is displayed on the screen.
Alt_H	Switches between Hex/ASCII display mode.
Alt_L	Switches between normal/line modes. In line-mode, any characters-entered will not be sent to the COM port until the ENTER key is pressed. This command is designed for testing 7000 series.
Alt_X	Quits the 7188X.EXE program.
F2	Sets the file name to be downloaded (without download operation).
F5	Runs the program specified by F2 and the arguments set by F6.
Alt_F5	Runs the program stored in the SRAM.
F6	Sets the arguments of the execution file set by F2. (a maximum of 10 arguments. If less than 10 arguments are set, add '*' to the end).
Ctrl_F6	Clears the screen.
F8	F8=F9+F5.
F9	Downloads the file specified by F2 into the FLASH memory.
Alt_F9	Downloads all files specified by ALT_F2 into the FLASH memory.
F10	Downloads the file specified by F2 into the SRAM and execute it.
Alt_F10	Downloads the file specified by F2 into the SRAM memory.
Ctrl_B	Sends a BREAK signal to the PC COM port that is currently being used by 7188xw.exe.
... more more ...

Appendix B: Minios7

The MiniOS7 utility is an embedded operating system designed for the following ICPDAS module families:

- 7188XA/7188XB/7188XC series
- 7521/7522/7523 series
- 7188EA/7188EX/7188EX-256 series
- 7188E1/7188E2/7188E3/7188E4/7188E5/7188E8 series
- 7188EF-016
- 8000 series.
- Iview-100 series
- More new embedded controller families

The MiniOS7 utility provides more specific functions for the 7188X/7188E/7521/8000 family.

MiniOS7 features:

Function	MiniOs7
Power up time	0.1 sec
Supports I/O expansion bus	Yes
Supports ASIC Key	Yes
Supports hardware unique serial number	Yes
Supports MMI, Iview-100 series	Yes
Supports Ethernet 10M interface, 7188E & 8X3X series	Yes
Directly downloads executable programs into Flash ROM	Yes
O.S. updateable (downloadable)	Yes
Built-in hardware diagnostic functions	Yes
Directly controls 7000 series modules	Yes
Customers ODM functions	Yes
Free of charge	Yes

Note: ICPDAS reserves the right to change the specifications of the MiniOS7 utility without notice.

MiniOS7 Command Set:

Command	Description
LED5 pos value	Shows a HEX value in the specified position of the 5-digit LED.
USE NVRAM	Read/Write to the NVARM.
USE EEPROM	Read/write to the EEPROM.
USE Flash	Read/write to the Flash-ROM.
USE COM2 /option	Send/Receive to/from COM2 (RS-485).
DATE [mm/dd/yyyy]	Sets the RTC date.
TIME [hh:mm:ss]	Sets the RTC time.
MCB	Tests the current memory block.
UPLOAD	The first command for updating the MiniOS7.
BIOS1	The final command for updating the MiniOS7.
LOAD	Downloads the user program into the Flash-Memory.
DIR [/crc]	Details of all files currently in the Flash Memory.
RUN [fileno]	Executes the file where the file number=fileno, no fileno→the last file.
Name	Executes the file where the file-name=name.
DELETE (or DEL)	Deletes all files stored in the Flash Memory. Note: this command will delete all files.
RESET	Resets the CPU.
DIAG [option]	Hardware Diagnostic command.
BAUD baudrate	Sets a new Baud Rate value.
TYPE filename [/b]	Lists the contents of the file.
REP [/#] command	Repeats the execution of the same command # times.
RESERVE [n]	Reserves n Flash Memory sectors for USER programs.
LOADR	Downloads a file into the SRAM.
RUNR [param1 [param2...]]	Runs a program saved into the SRAM (downloaded using the LOADR command).
I/INP/IW/INPW port	Reads data from the hardware PORT.
O/OUTP/OW/OUTP W port value	Outputs to the hardware PORT.
... more more ...

*** Refer to the **7188E\MiniOS7\DOC\Index.htm** file on the enclosed CD for

the user manual and demo programs for the MiniOS7 ***
The following libraries are included with the 7188E module.

7188EL.LIB: CPU & I/O related library (Large model)

TCPIPL.LIB: TCP/IP related library (Large model)

VComNNNN.LIB: Xserver related library (Large model), with NNNN being
the version number of the library