Network connection structure
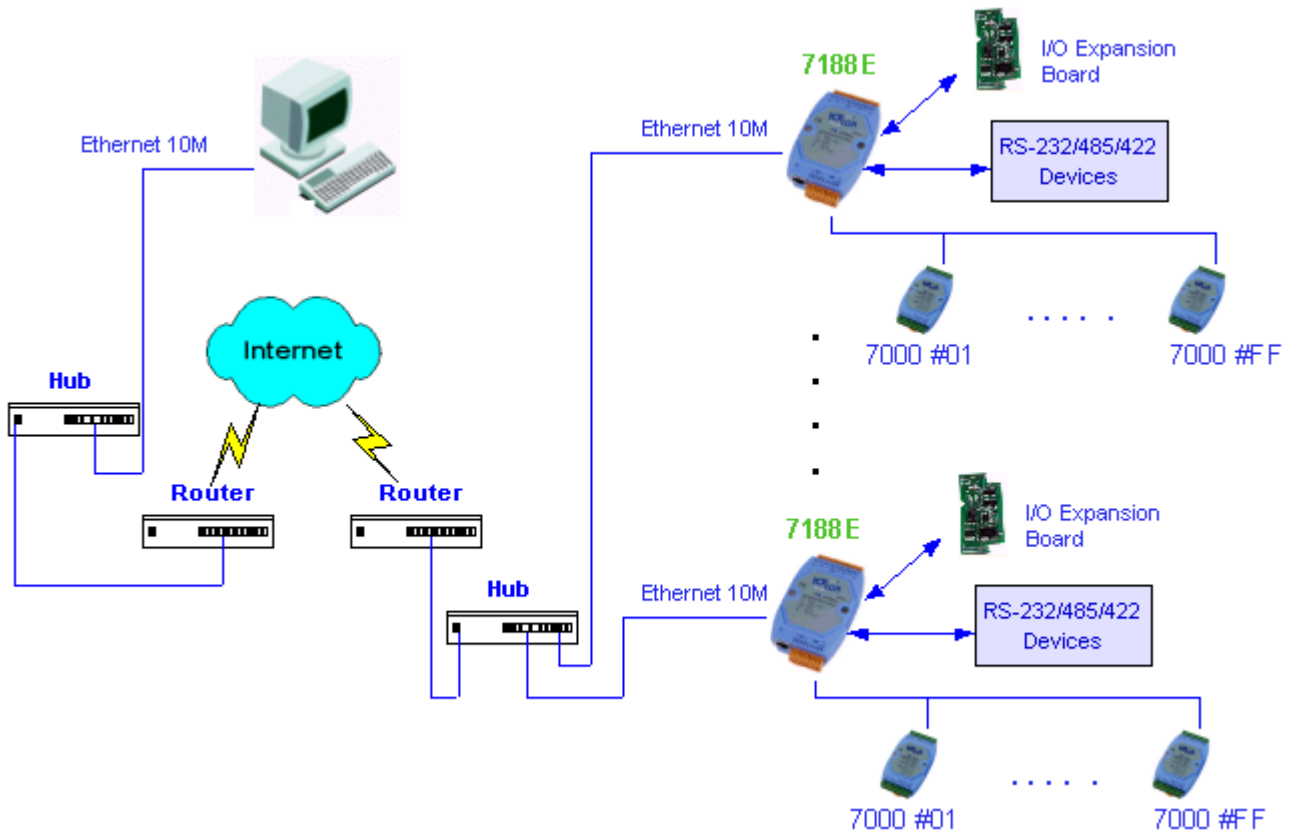

7188E series controllers can be equiped with one RJ-45 connector, one I/O expansion bus and several serial COM ports. It is an embedded controller, users can easily employ c language to develop their own programs. Futhermore, includeed with the 7188E are many demos and libraries. The 7188E can be used to access devices by Ethernet/Internet or RS-232/485/422. In addition, every 7188E controller can mount one I/O expansion board to implement various I/O functions such as D/I, D/O, A/D, D/A, Timer/Counter, Flash memory and battery backuped SRAM.

3 Typical Applications

7188E series controllers have 3 typical applications (Virtual COM, Ethernet I/O and web server implementation). These applications use different firmware and varying program styles. Users can choose which of the 3 applications they prefer.

Using the Virtual COM application, one PC can control 255 COM ports (include real COM ports). 7188E will become a RS-232 to Ethernet/Internet converter.

Using the Ethernet I/O and the Web Server applications, users can program the firmware of the 7188E (Xserver, Web server).
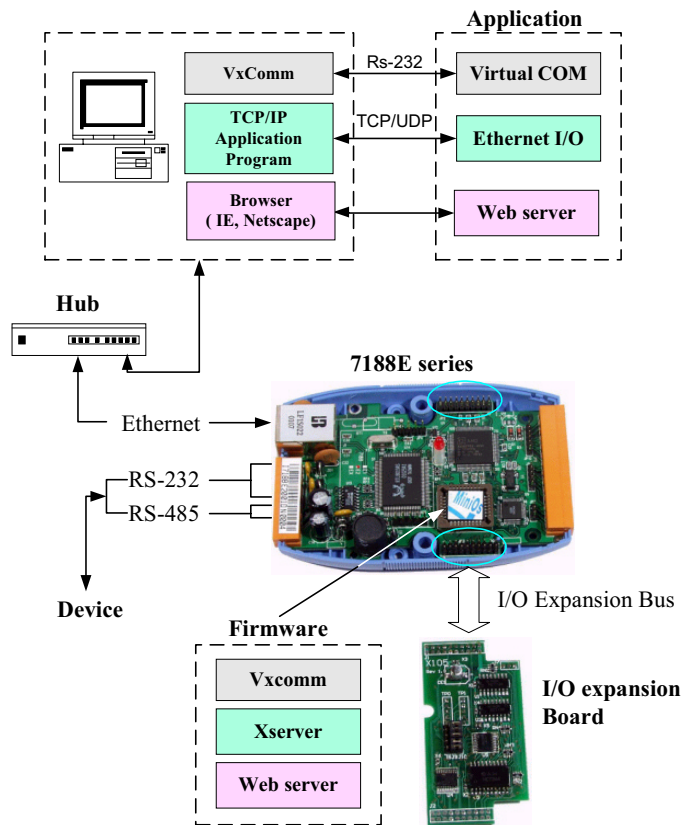
Virtual COM Application
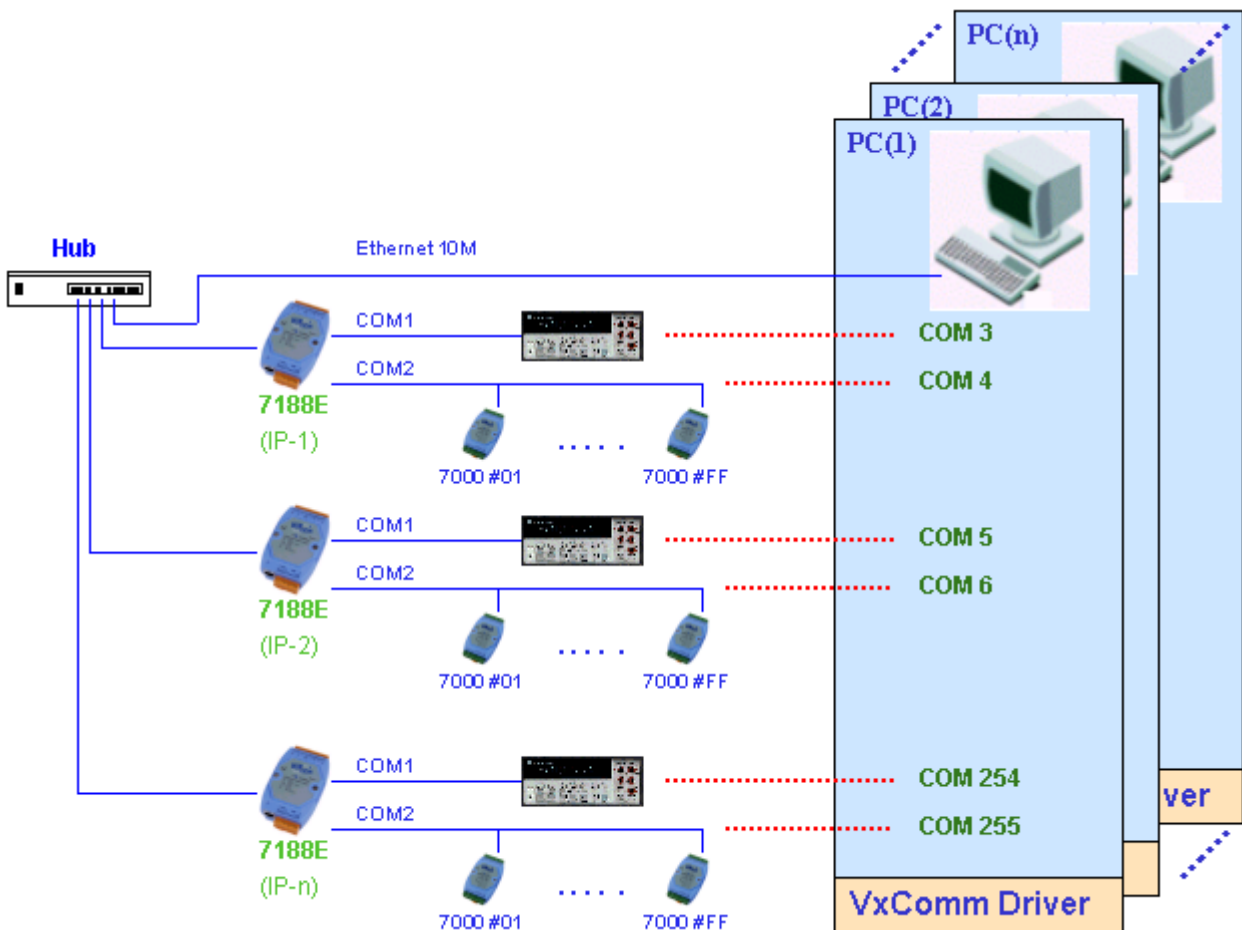
To use the Virtual COM application, users must install a VxComm driver first. After installation, the VxComm utility can be used to map remote COM ports of the 7188E. These virtual COM ports can be used by the PC to control devices directly, just like using COM1 (real COM port) to control devices. When using the Virtual COM application, one PC can use a maximum of 255 COM ports (include real COM ports).

Users need not worry about network connections. The VxComm driver will handle every Ethernet/Internet connection.

One thing users must note however is timeout. Due to possible network traffic problems, users must increase time out to about 1 second.

The advantages of VxComm:

1. Users can mount their systems to the Ethernet/Internet with increased ease as program code needs no modification.
2. The internal firmware of 7188E supprts multi-clients. One 7188E can handle a maximum of 30-N virtual COM ports simultaneously, with N being the number of the 7188E COM ports.

Ethernet I/O Application
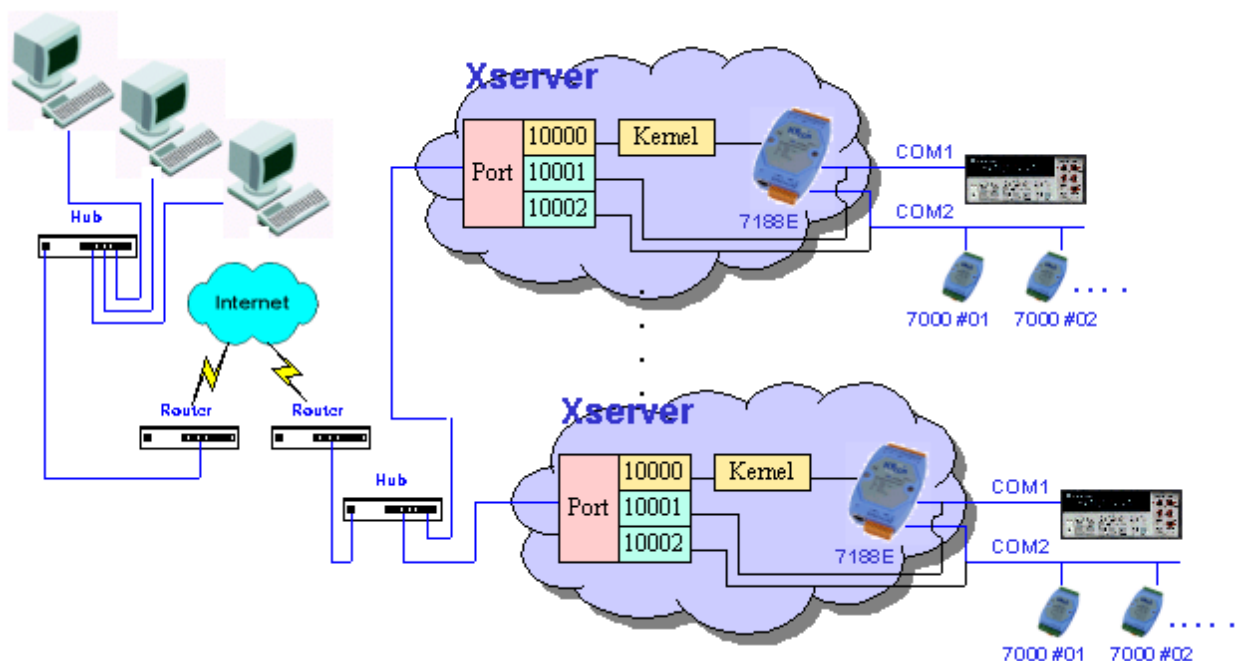
For Ethernet I/O Application, the Xserver uses ports from 10000 to 10000+N to listen to packets from the network (N being the number of 7188E COM ports). For example, in the diagram, the Xserver uses ports 10000, 10001 and 10002 to listen packets sent from the network. The client then uses one port to send packets to port 10000+N of the Xserver to request the Xserver to send packets to COM port N of the 7188E. User can control every COM port directly by sending packets to their port of choice (10001 to 10000+n).

Port 10000 of the Xserver is the most important port. The Xserver will pass packets form port 10000 directly to the Xserver's kernel, and pass packets form other ports (10000+N) to COM ports. The Xserver is a powerful program that supplies the 7188E with a range of flexible options. Users can modify the Xserver to control the whole of the 7188E's hardware, such as COM ports, I/O expansion board, 7 Seg LED and the like.

The advantages of the Xserver:

1. We design, maintain, update the Xserver for all users.
2. The Xserver can be easily modified through the use of general development tools such as TC/BC/MSC. All relative libraries are coded in C language.
3. High running speed. The original Xserver (Demo4.exe) can run about 750 scan loops per second (version 2.6.14).
4. Main program code of the Xserver is finished. The Ethernet/Internet communication and program loop control are all finished in VxComm.lib. This vastly reduces user's desiging time.
5. To modify the Xserver, users need only to modify the 5 functions below (located in user.c):
    A. UserInit(void)
    B. UserLoopFun(void)
    C. UserCount(void)
    D. UserCmd(Cmd, Response)
    E. VcomUserCmd(TCPREADDATA *p)
6. The command protocol is designed to fit most requirements of the 7188E.
7. Users can develop and extend their private command protocol very easily.
8. Supports multi-clients. The Xserver can handle a maximum of 30-N connections simultaneously, with N being the number of 7188E COM ports.
9. Auto wake up. The Xserver will check socket timeout. If software crashes, the Xserver will wake itself up automatically.
10. The Xserver demos (TC/BC/MSC) and Client demos (VB/VC) included.

Command Protocol of the Xserver

Cmd	Explain	Instruction Format	Example: Command	Example: Receive
01	Version	<01>	01	V2.6.14[10/04/2001]
04	Get system reset status	<04,Client(2)>	0414	141(First asking from client 14 after system reset) 140(Not first asking from client 14 after system reset)
05	RTS	<05,Port(1),Set(1)>	0511	OK(COM Port RTS on)
06	Baud Rate	<06,Port(1),Baud>	0619600	OK
07	Set Line Control	<07,Port(1),LineControll(3)>	0718N1	OK
08	Set IP	<08,IP(12)>	08192168255001	OK..Reconnect
10	Server Name	<10>	10	7188E2
11	Diag	<11,String(<=80)>	11Hello	Hello
12	Set Gateway	<12,GatewayIP(12)>	12192168000001	OK..Reconnect
13	Get Gateway	<13>	13	192.168.0.1
14	Set Mask	<14,Mask(12)>	14255255000000	OK..Reconnect
15	Get Mask	<15>	15	255.255.0.0
16	Get COM Status	<16,Port(1)>	161	9600,8,N,1
17	Digital Input	<17,Addr_Hex(4)>	1703f8	F8
18	Digital Output	<18,Addr_Hex(4),Data_Hex(2)>	1803f855	OK
19	**Bypass User Defined Command**	**<19,Command>**	**19(User defined)**	**(User defined)**
20	Enable 5 DigitLED	<20,Enable(1)>	201	OK (Enable 5 DititLED show information)
21	Get Mac	<21>	21	71:88:03:39:9f:e2
22	Get MiniOS Version	<22>	22	V1.2 (2000/06/17)
23	**Call VcomUserCmd in Xserver**	**<23,String>**	**23(User definded)**	**(User defined)**
24	Set feedback command No.	<24,Enable(1)>	240 241	OK 24OK

**Note 1:** The number inside ( ) of instruction format means parameter size (byte).

Note 2: Don't insert any space between parameters (except user defined command).

Note 3: All command (expect user defined command) response will be added termanial char CR (0x0d).

Note 4: Refer to **vxcomm.htm** to get more information about Xserver command protocol and parameter setting in **7188e\Tcp\Vxcomm\Doc**

The Xserver software block diagram

VxComm.lib   **User.c**
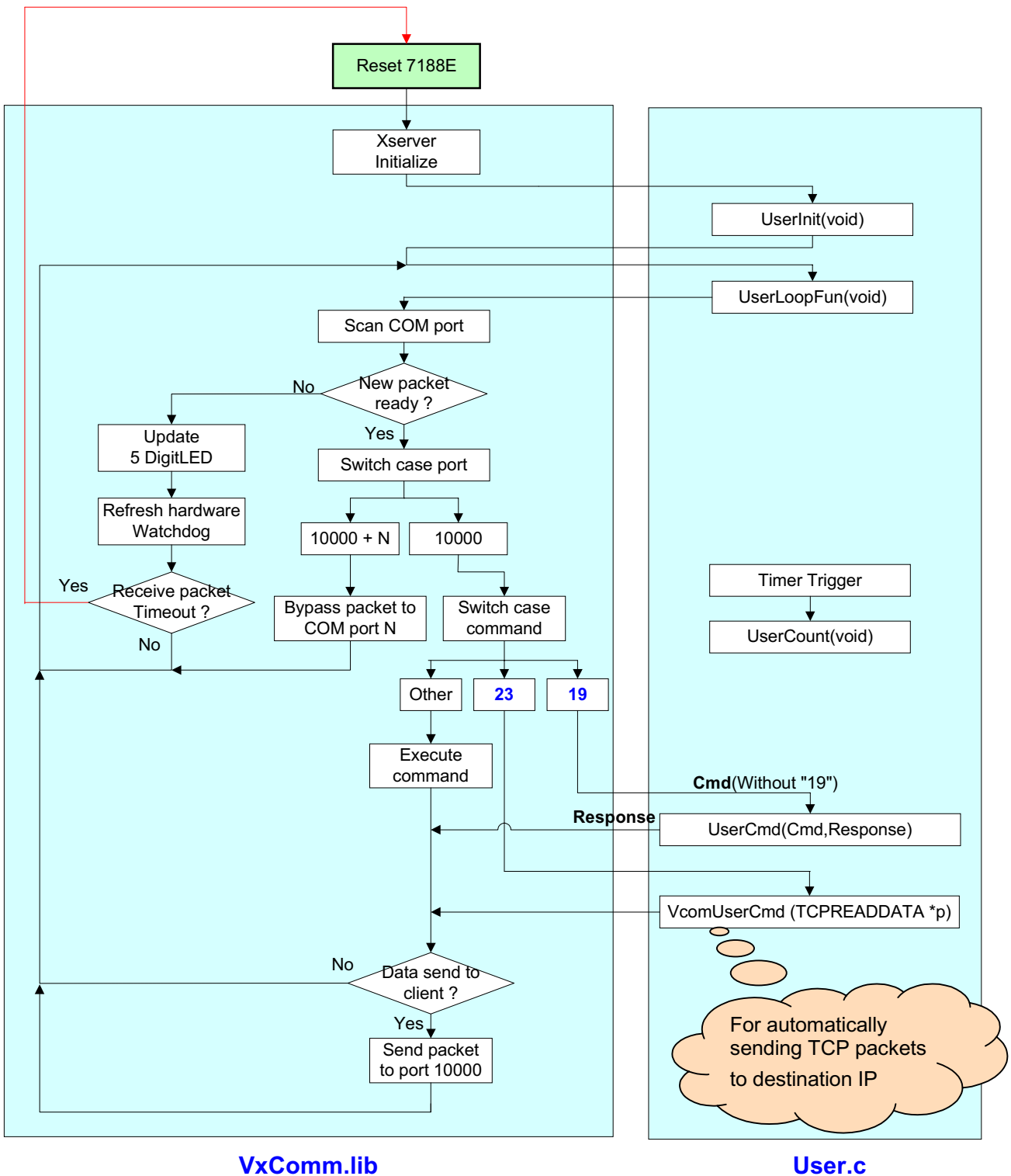
To develop an enbedded controller program with Ethernet/Internet communication is very difficult. But by using the Xserver, users can do that quickly and easily. Users need only modify 5 functions in User.c. Users build their own code in the 5 functions to make the Xserver work as they desire. The features of the 5 functions are listed below:

UserInit (void): This function will start running as soon as the 7188E is turned on. Therefore all variables of initial values or initial status must be set in this function.

UserLoopFun (void): This function will run during every scan loop. One Xserver scan loop completes in a short time, so real time work should be executed by this function. **See Demo12.**

UserCount (void): This function will be triggered when the time interval, set in AddUserTimerFunction, is up. For best result, call AddUserTimerFunction in UserInit to let the Xserver call the UserCount period. Longest time interval is 65.535 seconds (2^16–1 ms). **See Demo9.**

UserCmd (Cmd, Response): Client program sends the command "**19**" to request the Xserver execute the user's defined command. This command is defined in function UserCmd by the user. When the Xserver receives packets from **port 10000**, the Xserver checks the data. If the data is begging with "**19**", the Xserver trims "19" and passes the other data (not including "19") to be first parameter **Cmd** of function UserCmd.

User can define his own command protocol in UserCmd. For example, define <19,Read/Write(1),address(4),[value(2)]> to replace command 17 and 18, then one can send "19R03f8" to read values form address 03f8; sends "19W03f85a" to write 5a to address 03f8.

Users can decide whether or not any other characters are needed between parameters. Any command protocol format will be accepted, because it is user defined.

At the end of UserCmd, copy the results to the second parameter **Response**, then the XServer will send the string to the Client program by port 10000. **See Demo4.**

VcomUserCmd (TCPREADDATA *p): Client program sends command "**23**" to request the Xserver execute this function. This function is similar to UserCmd. When the client program sends command "23", VcomUserCmd will receive TCPREADDATA type information . The TCPREADDATA is declared as below:

Type define t_TcpReadData{

Int Comport;

int Socket;

int Length;

char* ReadUartChar;

} TCPREADDATA;

p->ReadUartChar: the buffer where command data is stored(include "23")

p->Length: the command data length (include "23")

p-Socket: the Xserver will assign a socket number to index which client sent command "23" to 7188E. So, the socket number can used to return messages to a specific client. To send message to specific client, call VcomSendSocket(int skt, char * data, int cnt). The first parameter should be the socket number. **See Demo23.**

Web server Application

With the help of Wserver, user can use standard browsers (such as IE or Netscape) to access the I/O of expansion boards or devices connected to the COM port of the 7188E.
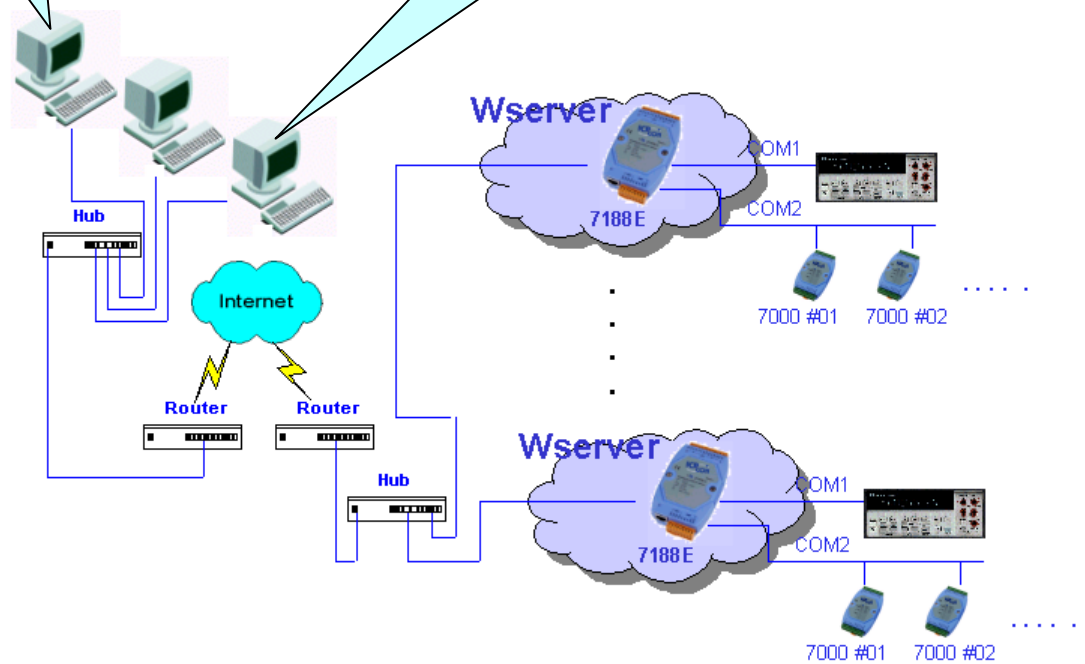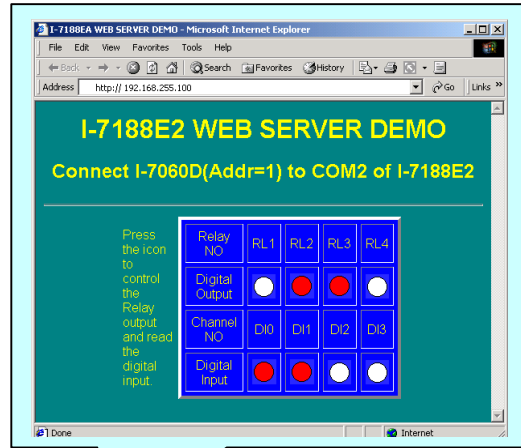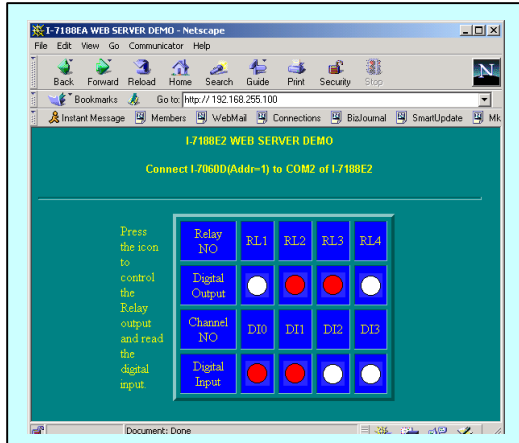
Software resources structure

To help users reduce designing time, we support many software resources, including documents, drivers, libraries, diagnostic programs, client programs and many Xserver demos. You can quickly find which resources you need by using the directory tree.

```
7188E ──┬── Document ────── Readme.htm
        │
        ├── MiniOS7 ──┬── Demo
        │             │
        │             └── Doc ────── Big5/Eng/Gb2312
        │
        ├── TCP ──┬── PCDiag ──┬── Setup ────── 1. Configure Wizard
        │         │            │                2. Send232
        │         │            │                3. Send TCP
        │         │            └── Source       4. 7188E
        │         │
        │         ├── VxComm ──┬── Doc ────── Big5/Eng/Gb2312 ────── vxcomm.htm
        │         │            │
        │         │            ├── Driver(PC) ────── NT/2K/XP
        │         │            │
        │         │            └── Server(7188e)
        │         │
        │         ├── Web-Server
        │         │
        │         ├── Xserver ──┬── 7188EA/7188E2/.../7188E8
        │         │             │
        │         │             ├── Client ────── VB/VC
        │         │             │
        │         │             ├── Demo ────── BC/MSC/TC ──┬── Demo
        │         │             │                           └── Lib
        │         │             │
        │         │             ├── Xserver.htm
        │         │             │
        │         │             └── Function.htm
        │         │
        │         └── Other
        │
        ├── UDP ──
        │
        └── Other
```
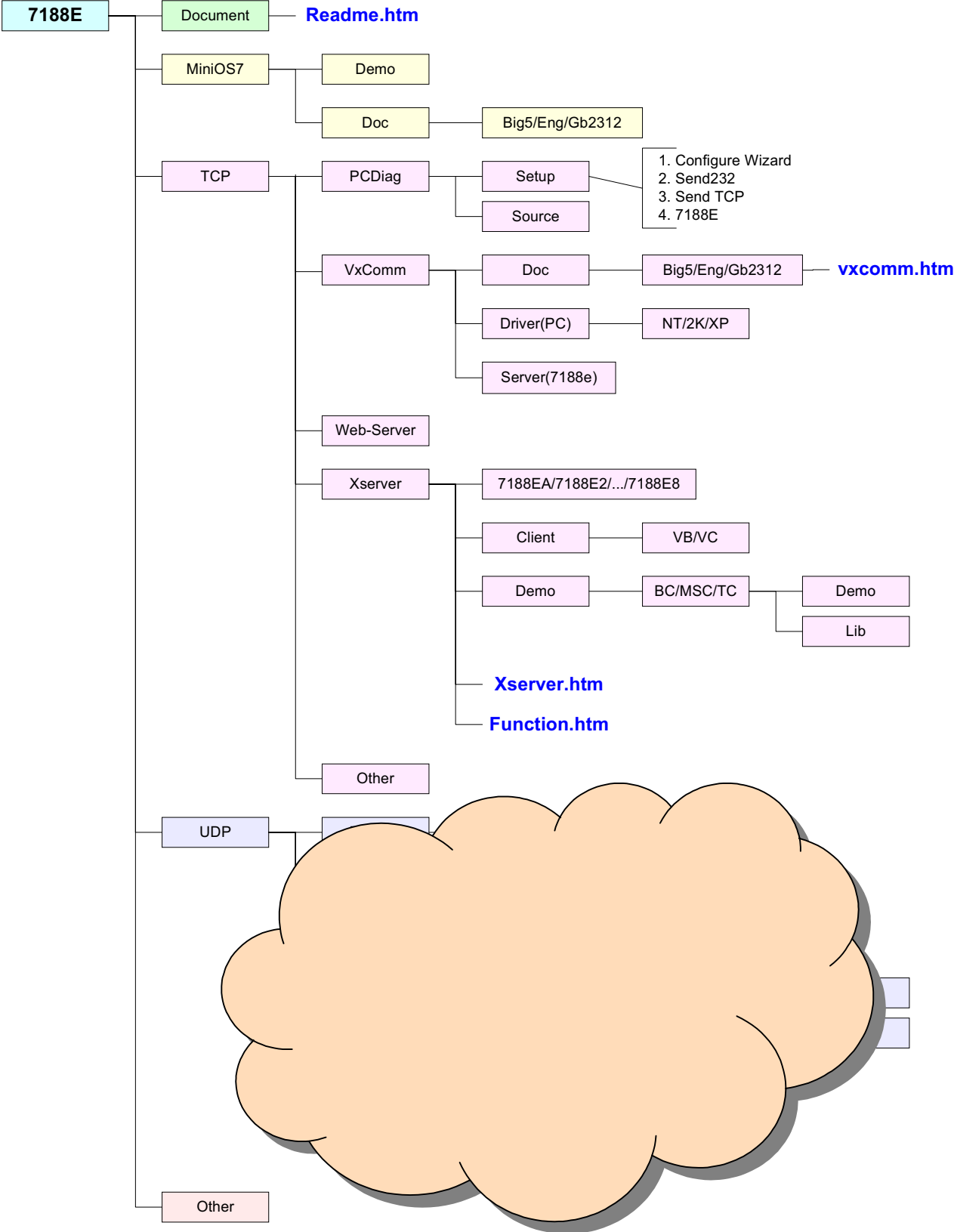
Application

In the application, there are almost 10000 analog channels that must be dealt with. A PC must acquire, average and compress all data. The system uses 14 7188E2 modules to build a distributed computing system. Every 7188E2 controls 14 I-7018 or I-7033 modules to acquire temperature data. All averaging and compressing functions are built into the Xserver. So, one PC can monitor thousands of analog signals from remote information centers by Ethernet/Internet.