

WE-200M Web-based Control Application Note

Hardware Layout

ETx+	1
ETx-	2
ERx+	3
ERx-	4
Link/Act	5
PIO0	6
PIO1	7
PIO2	8
PIO3	9
Reset	10
Ready	11
SW1	12
SW2	13
GND	14
+5V	15
16	RxD
17	TxD
18	RTS
19	CTS
20	DCD
21	DSR
22	DTR
23	PIO4
24	PIO5
25	PIO6
26	PIO7
27	PIO8
28	NC
29	NC
30	GND

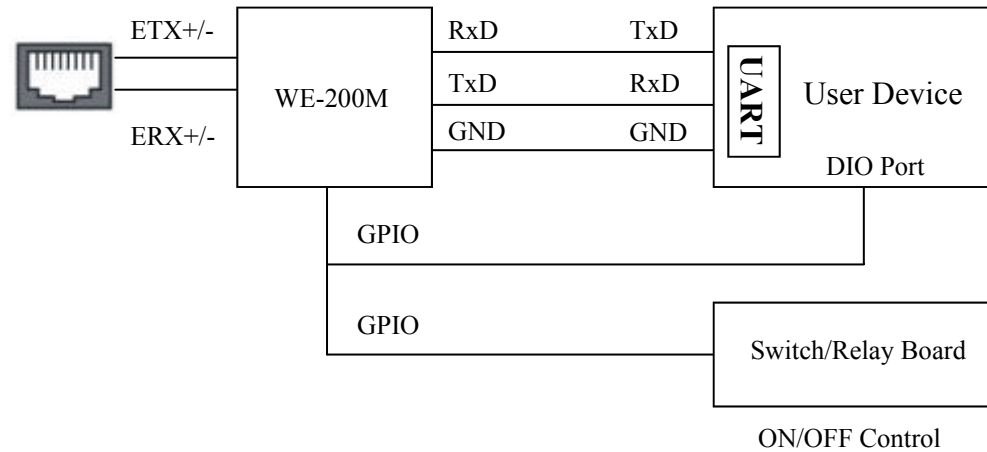


WE-200M is an Embedded Serial to Ethernet Converter module and it also comes with 9 programmable Digital I/O and 64KB Flash memory space for user to customize his Web application software.

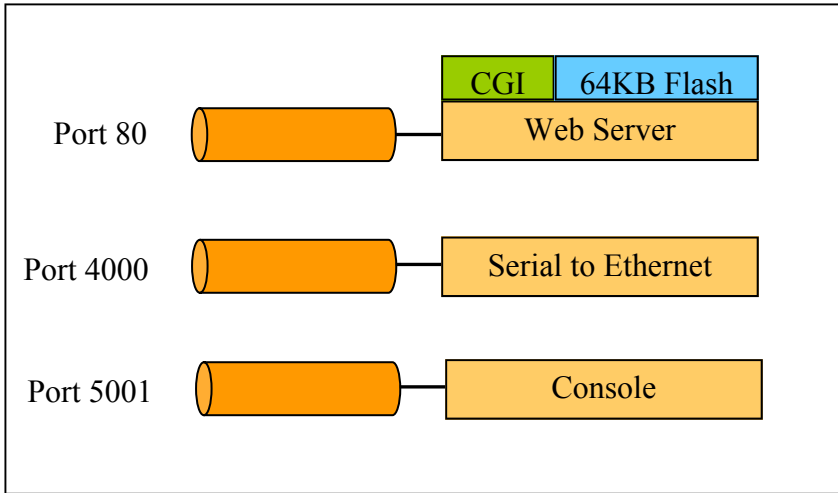
The Ethernet port of WE-200M is equipped with LAN transformer therefore user only needs to connect the ETX and ERX pins to an RJ45 connector.

The Serial Port of WE-200M is a TTL level signal. Please connect to the UART port of the device controller which you want to access via the WE-200M.

The GPIO Port of WE-200M is CMOS/TTL level programmable digital I/Os. These GPIO can be configured as digital input or output and are used to monitor the I/O status of user device or perform an ON/OFF control by using a relay module.



Firmware Architecture



WE-200M firmware includes three major functions which can be described as figure of firmware architecture.

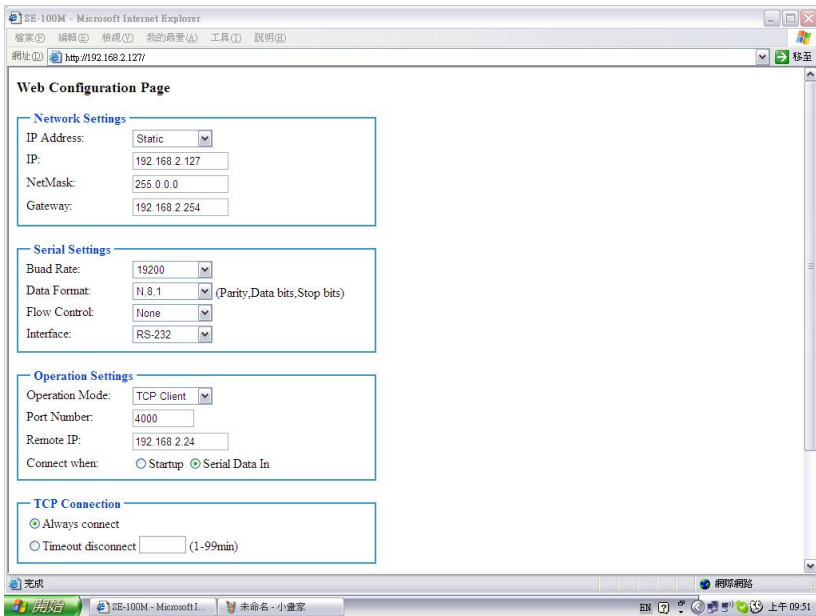
The Console module is a Telnet console port. This TCP port 5001 is used to configure the network and serial port settings of WE-200M and the GPIO controls. The console module uses Command Line Interface (CLI) to configure WE-200M. Please refer to WE-200M user manual for the detail information of the text command.

The Serial to Ethernet module is a TCP server port (Port 4000) and it will convert data to and from serial with Ethernet port. This allows user's program which includes a TCP client function to build a TCP connection and transfer data between WE-200M serial to Ethernet port.

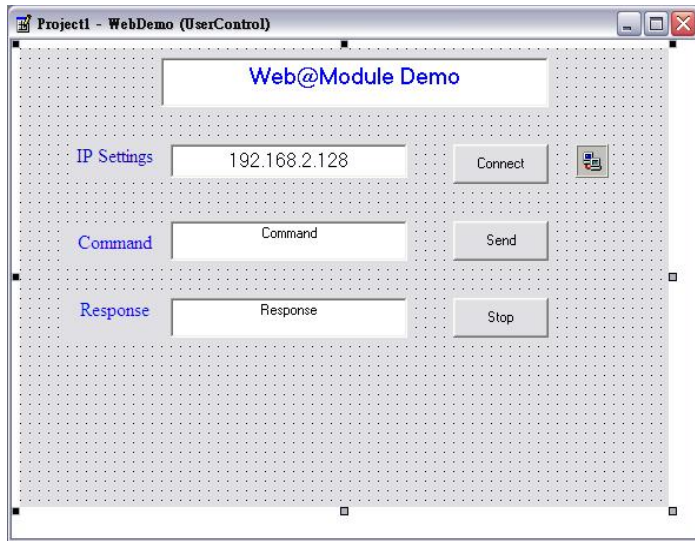
The Web Server module includes a tiny HTTP server, CGI module and 64KB flash memory space. The CGI module is a Web Console page which is used to configure the network and serial port settings of WE-200M. It is also the factory default page of the Web Server. When user web page is loaded to the Web server's 64KB space, user's page will replace CGI module as the Home page of WE-200M. Likewise, when the 64KB space is empty, Web server will load the default CGI page as home.

The user Web page can include text-based HTML and Javascript and embedded program such as Java applet, ActiveX controls etc. User can use any Web page editor to edit his text-based web page. The embedded program can be Java Applet or Visual Basic ActiveX Controls. To perform Web-based control to user device, the embedded program should contain TCP client function which will connect to Serial to Ethernet port (Port 4000). Therefore user can use web page to transfer data to and from WE-200M serial port. To perform digital I/O control, the embedded program should connect to the Telnet console port (port 5001) and use CLI to control the digital I/O port of WE-200M.

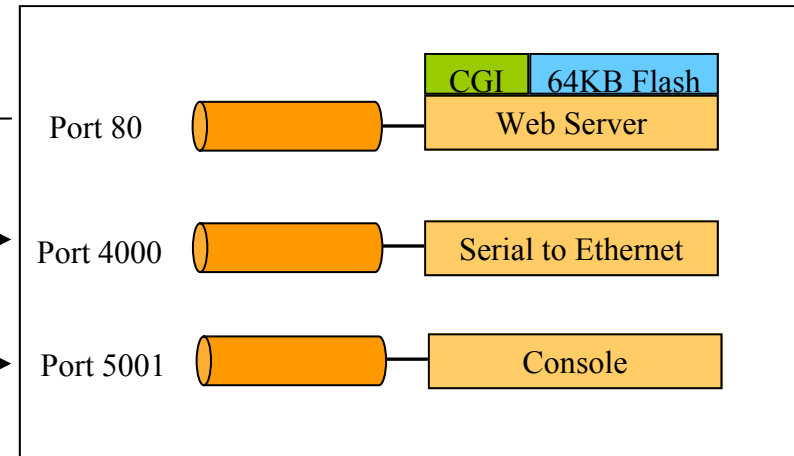
Web Console Page



Web Browser



WE-200M



When using web browser to visit WE-200M, browser will load the web page via TCP Port 80. Once web page is loaded to browser, the embedded program will be opened and build up a TCP Port 4000 connection to WE-200M. Once it is connected, user can access the serial port of WE-200M via browser. If user want to control the digital I/O of WE-200M, the embedded program need to build up a connection to console port (Port 5001).

Web@Module Visual Basic Implementation

If you are familiar with Visual Basic (VB), you will find it is very easy to implement a Web-based remote device management using Web@Module. Before you start to program your VB program, please find a good VB programming guide and practice network programming (winsock) using VB. Once you can access the serial port of Web@Module hardware through TCP server mode, you are ready to develop your web page for Web@Module. The operating principle of Web@Module is as following:

1. Develop a VB ActiveX Controls program first. This program contains a TCP client function which will build a TCP connection to Web@Module
2. Web@Module is configured as TCP server mode and the serial port of Web@Module hardware is correctly configured for the device to be controlled
3. Once your ActiveX Controls program can be operated correctly at your computer, you can pack your VB program as a web page.
4. Use the **Manager** utility program, you can convert the home page (index.html) and the related files such as winRAR (*.CAB) to a binary file (fsdata.anf).
5. Upload the fsdata.anf to Web@Module and reset the Web@Module to refresh the new web page
6. Now you can use your Web Browser (IE) to visit the web server of Web@Module
7. Remember to configure your Web Browser Security setting to allow ActiveX Controls .
8. The home page of Web@Module will download a ActiveX Controls to your web browser and the ActiveX Controls will build up a TCP connection to the Web@Module and open the serial port of Web@Module to control the device.

Following steps show you how to build a dynamic Web Page for Web@Module using VB 6.0 SP64

ATTENTION

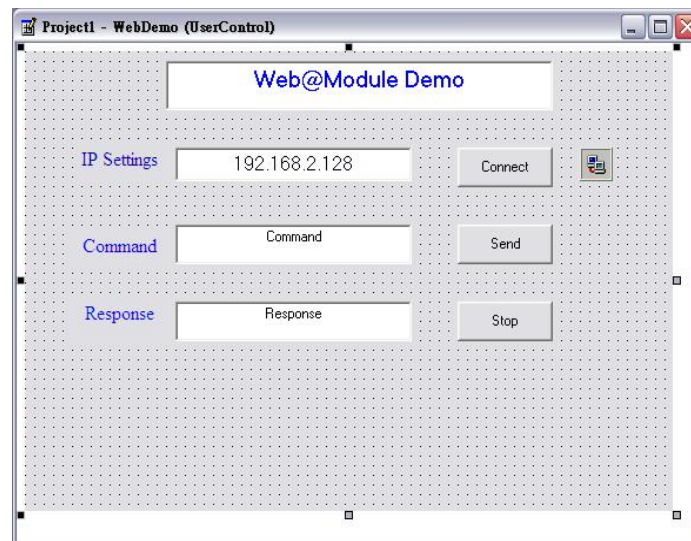


The filename of the VB ActiveX Controls must be less than 14. Otherwise the web server of Web@Module cannot recognize it.

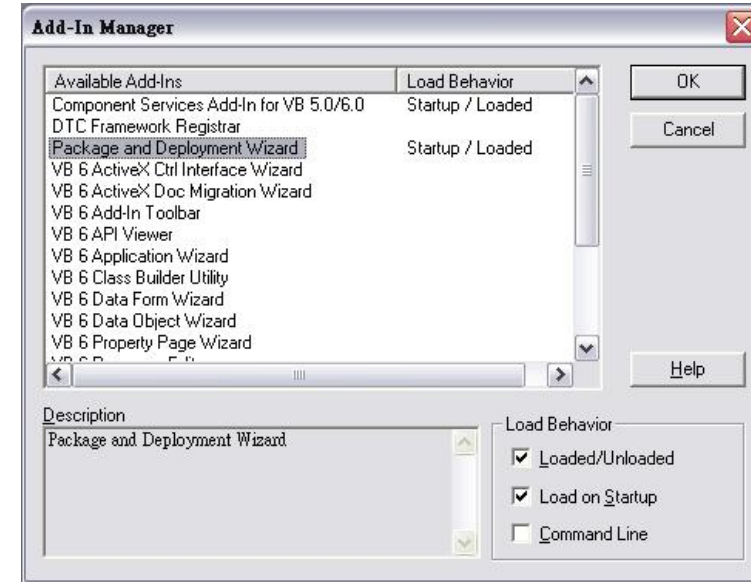
- Step 1
Open a new project and choose ActiveX Control



- Step 2
Program an ActiveX Control and which contains a TCP client function. Debug the program at your PC until it is bug free



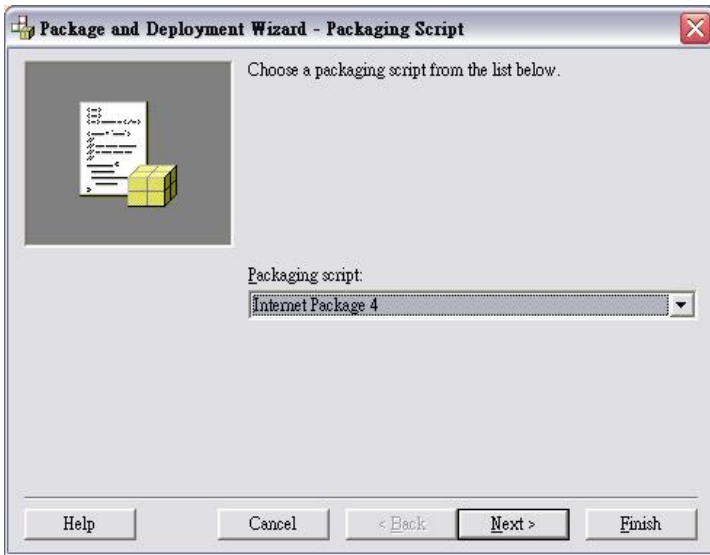
- Step 3
Under Tool menu of VB 6.0 add Package and Deployment Wizard



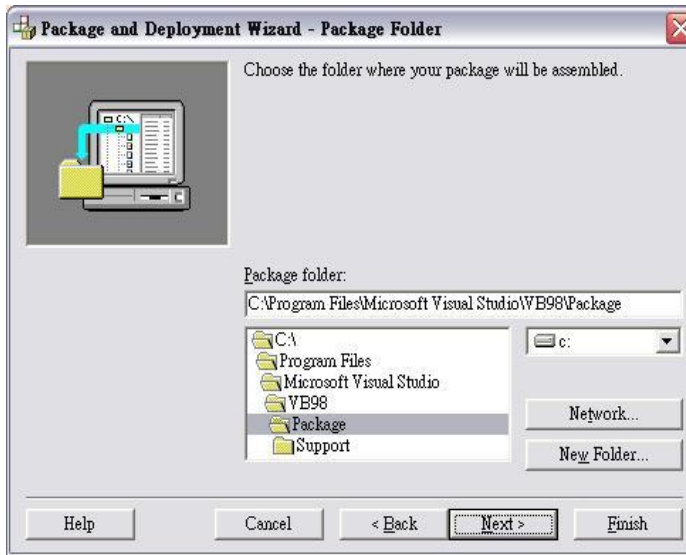
- Step 4
Use the Package and Deployment Wizard and Click **Package**



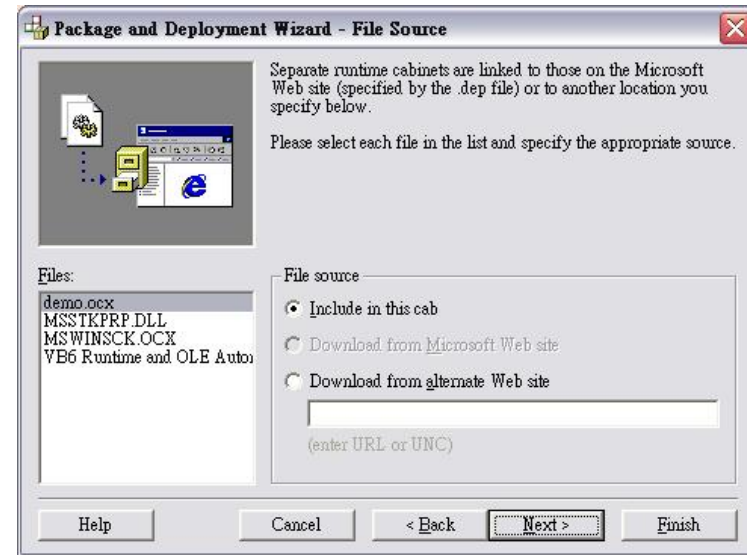
Step 5
Click Next



Step 7
Choose the folder for the web page



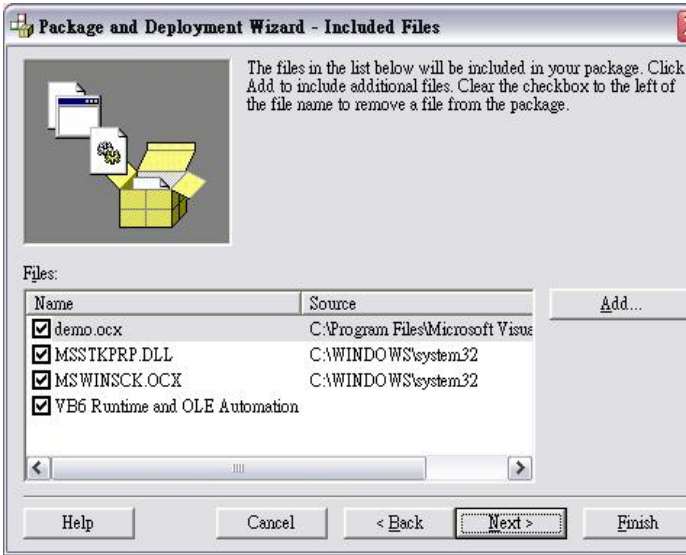
Step 9
Click Next



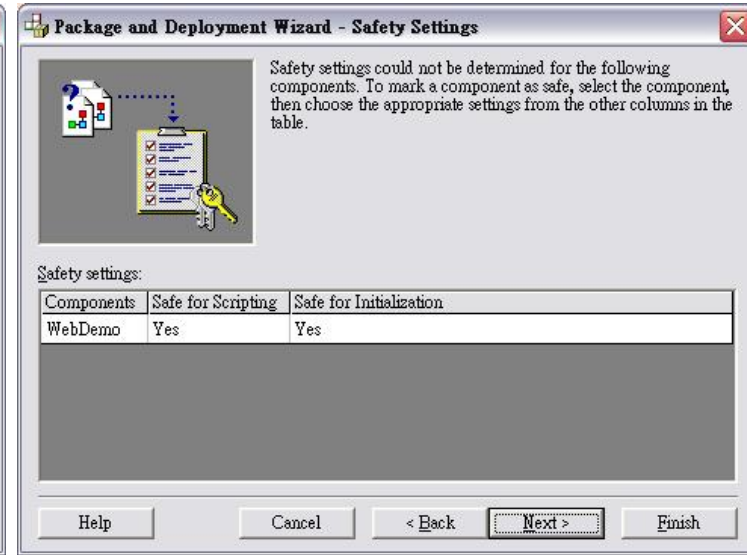
Step 6
Choose Internet Package



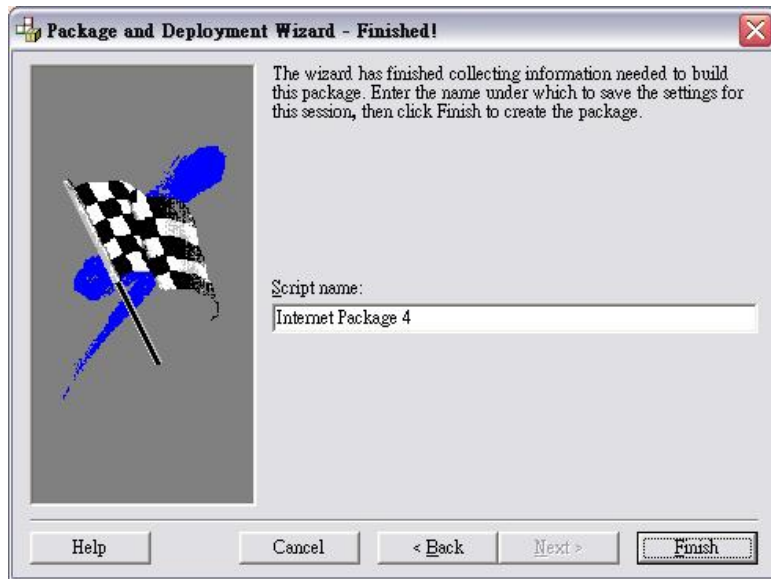
Step 8
Click Next



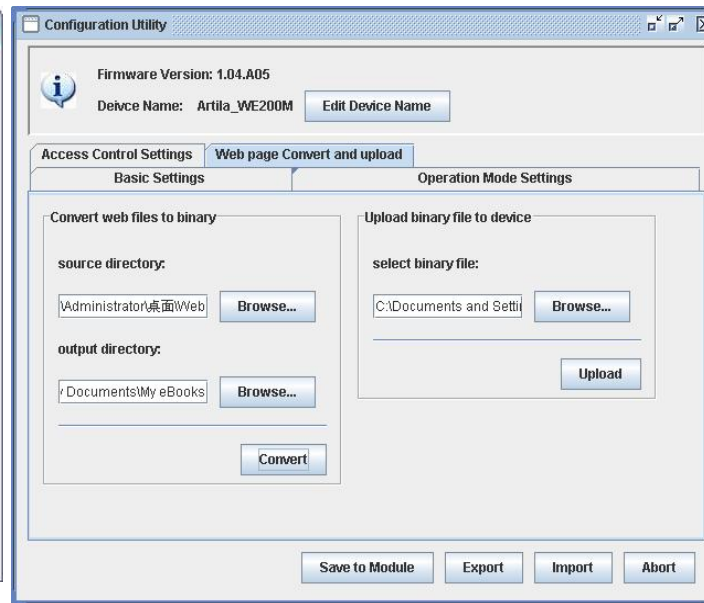
Step 10
Choose Yes in Safe for Scripting and Safe for Initialization



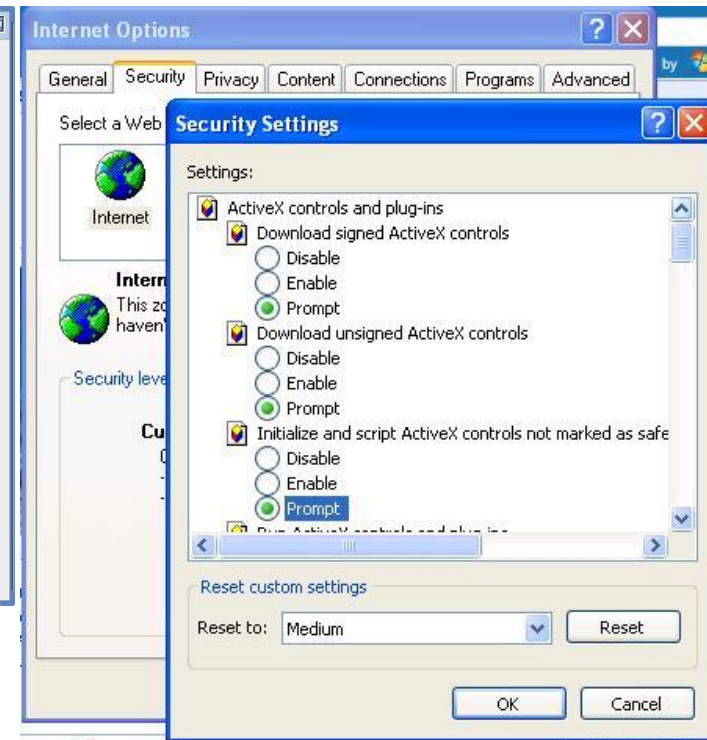
Step 11
Click Finish to Complete



Step 14
Use Manager Utility in CD to Convert the index.html and the WinRAR file (.CAB) to the output folder for fsdata.anf



Step 15
Configure Web Browser Security setting to allow ActiveX Control and disable the Internet Security Software.



Step 12
You will discover one HTML file and one WinRAR (.CAB) file are created and one Support folder in the Package folder



Step 13
Copy the WinRAR file and HTML file to a new folder and rename the HTML file as **index.html**.

- Example Program in Artila CD are located at
1. VB example program: \VB\example
 2. Index.html and WinRAR file: \VB\html
 3. Binary file fsdata.anf:\VB\upload