

Using the Explorer 16 Development Board for Wi-Fi Applications

An Application Note

Andrew Dutton

11/16/2011

As society advances, the need for wireless communication is becoming more of a demand than a luxury. Wi-Fi can be used for many applications beyond the common internet communication. This guide will describe the abilities that the Microchip Explorer 16 Development Board contains for Wi-Fi operation, as well as how to set the Explorer 16 Development Board for Wi-Fi communication. An application is examined and outlined.

Table of Contents

• Introduction-----	2
• Getting Started-----	3
○ Hardware Options-----	3
○ Hardware Specs-----	5
○ Software-----	6
• Programming the Device-----	8
• Custom Demo Applications-----	12
○ Configure the Stack-----	12
○ Running the Demo-----	19
• Conclusion-----	20
• References-----	21

Introduction

Wireless communication has revolutionized the way people live their lives in today's society. The style of life is drastically shifting towards a remote operation of handling today's tasks. Wireless has the ability of simplifying a countless number of issues ranging from ordering movie tickets on your cell phone to remotely turning the power on and off in one's home. There are many purposes of wireless communication that the average person does not see. A number of electronic devices can be controlled via wireless communication. Wi-Fi is also known as the IEEE 802.11 standard for controlled communication at 2.4GHz

In this tutorial we will go over how to use the Microchip Explorer 16 Development Board to construct a Wi-Fi communicating device. Many factors go into developing a Wi-Fi device. Some of these factors are user application specific, and some are IEEE 802.11 standard specific. Here one will learn how about the hardware necessary to build this device, as well as the software necessary to drive this device.

Getting Started

Hardware Options & Set up

The first component that is necessary is the Microchip Explorer 16 Development Board. This development board is low cost, efficient development board to evaluate the features and performance of Microchip's new PIC24 Microcontroller, the dsPIC33 Digital Signal Controller (DSC) families, and the new 32-bit PIC32MX devices. The Explorer 16 board is an all-purpose demonstration and development board for 16-bit and 32-bit parts. It can also be expanded for TCP/IP support using the Ethernet PICtail Plus, Fast 100Mbps Ethernet PICtail Plus, or 802.11b Wi-Fi PICtail/PICtail Plus daughter board.[1] The development board comes in many configurations, and this necessary configuration is dependent on what type of transceiver module is necessary. [1]

For communication on the Wi-Fi 802.11 Media Access Control layer, the following hardware devices are necessary:[2]

- Explorer 16 Development Board (PIC24FJ128GA010 100 pin PIM)(Microchip part number DM240001)* [3]
- MRF24WB0MA Wi-Fi PICtail/PICtail Plus Daughter Board (Microchip part number AC164136-4) [4]

*The 100 pin PIM is necessary for communication to the MRF24WB0MA Wi-Fi module. For this tutorial, we will talk about using the PIC24F family, but the dsPIC33 EX-16 Dev. Board will also function correctly.



Figure 1) The MRF24WB0MA Wi-Fi PICTail module

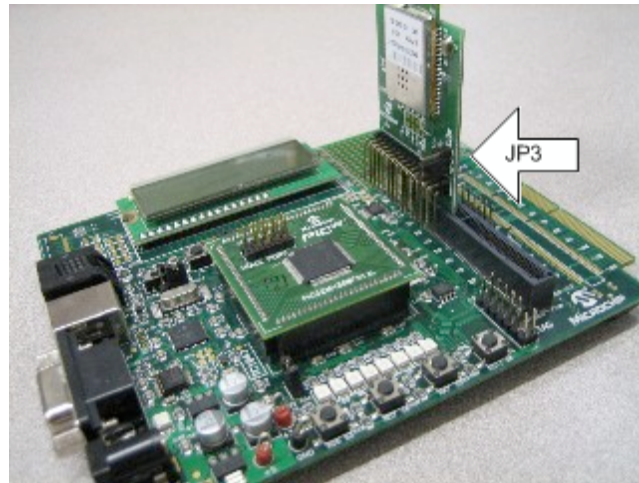


Figure 2) Explorer 16 Development Board with the 100 pin PIM, and the MRF24WB0MA Wi-Fi PICTail module

When you receive your hardware from Microchip, the Development Board will come with a CD. The CD contains user guides for the Explorer 16 Development Board, data sheets for different PIM's, and demo application code specific to the development board and the PIM itself. The Explorer 16 can be used to debug wireless functionality by connecting the PICTail as show in Figure 1, with header J2 on the PICTail inserted into the top slot of connector J5 (Explorer 16) on the demo boards. Note if jumper JP3 exists, it must be shorted between pins 1 and 2 when used on this development platform. Once your hardware is configured, you can program your board with your preferred demo project. [2]

Hardware Specifications

The CD that comes with the development board contains user guides for the Explorer 16 Development Board, data sheets for different PIM's, and demo application code specific to the development board and the PIM itself. Specification sheets for these pieces of hardware can also be found on the Microchip website (DM240001 user guide @ <http://ww1.microchip.com/downloads/en/DeviceDoc/Explorer%2016%20User%20Guide%2051589a.pdf>: MRF24WB0MA Wi-Fi PICtail/PICtail Plus Daughter Board @ <http://ww1.microchip.com/downloads/en/DeviceDoc/51912A.pdf>)

Software Options

To create a Wi-Fi device with Microchip products, we will need a few free downloadable products from Microchip. First you will need the most recent version of MPLAB Integrated Development Environment (IDE). MPLAB (IDE) is a free, integrated toolset for the development of embedded applications employing Microchip's PIC[®] and dsPIC[®] microcontrollers. MPLAB IDE runs as a 32-bit application on MS Windows[®], is easy to use and includes a host of free software components for fast application development and super-charged debugging. MPLAB IDE also serves as a single, unified graphical user interface for additional Microchip and third party software and hardware development tools. Moving between tools is a snap, and upgrading from the free software simulator to hardware debug and programming tools is done in a flash because MPLAB IDE has the same user interface for all tools.[2] Also, because we will be using the PIC24FJ128GA010 PIM, we will need to download the MPLAB[®] C Compiler for PIC24 MCUs. This is a full-featured ANSI compliant C compiler for the Microchip PIC24 family of 16-bit devices. MPLAB C is a 32-bit Windows[®] console application as well as a fully integrated component of Microchip's MPLAB Integrated Development Environment (IDE), allowing source level debugging with the MPLAB REAL ICE™ Emulator, MPLAB ICD 2 In-Circuit Debugger and MPLAB SIM Simulator.[1]

The Compiler Features are [1]:

- ANSI compliant with standard, math, memory, data conversion and math libraries
- Generates relocatable object modules for enhanced code reuse
- Optimized to generate as much as 30% less code than other 16-bit MCU compilers
- Strong support for in-line assembly when total control is absolutely necessary
- Peripheral library for quick coding using Microchip device peripherals
- Allows code and data to be located at absolute addresses
- Supports advanced code size optimizations
- Free unrestricted Evaluation Version of the C compiler

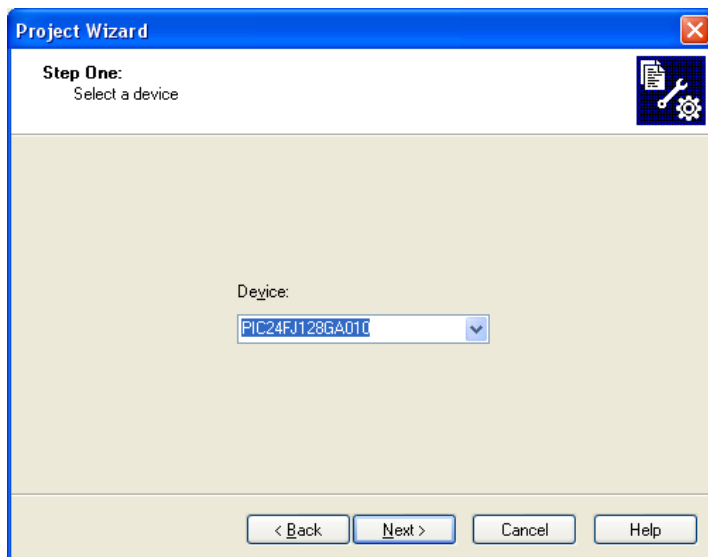
Both of these software tools can be downloaded at:

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en023073

Now that you have the necessary tools to develop your platform, there is one more piece of hardware that is necessary to download. The MRF24WB0MA transceiver module will support Microchips TCP/IP free licensed stack. The Microchip TCP/IP Stack provides a foundation for embedded network applications by handling most of the interaction required between the physical network port and your application. It includes modules for several commonly used application layers, including HTTP for serving web pages, SMTP for sending e-mails, SNMP for providing status and control, Telnet, TFTP, Serial-to-Ethernet and much more. In addition, the stack includes light-weight and high-performance implementations of the TCP and UDP transport layers, as well as other supporting modules such as IP, ICMP, DHCP, ARP, and DNS. The Microchip TCP/IP stack can be downloaded at: www.Microchip.com/MAL. When prompted where to save the file, save in C:\.

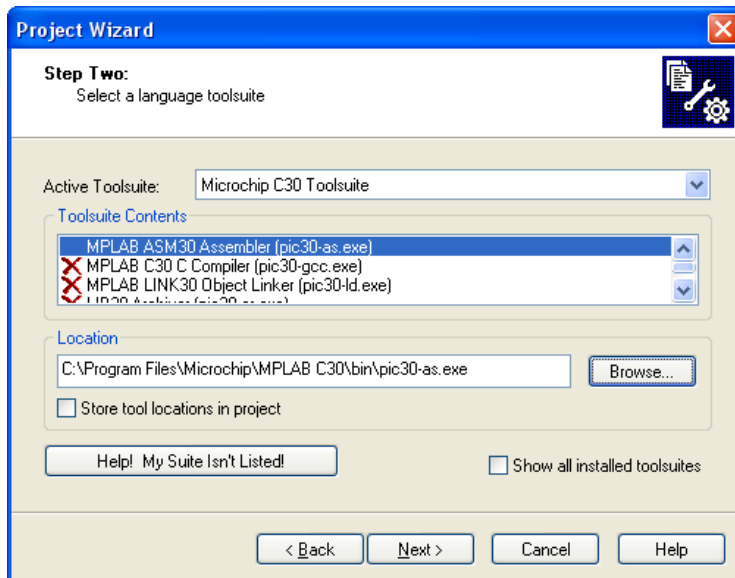
Programming the Device

To program the device, you will first need to open up a session of MPLAB. To do this either click on the desktop icon, or browse the start menu->all programs->Microchip->MPLAB IDE. Next, click on the Project tab->Project Wizard, and a window will open, click next. Now you will see the window:



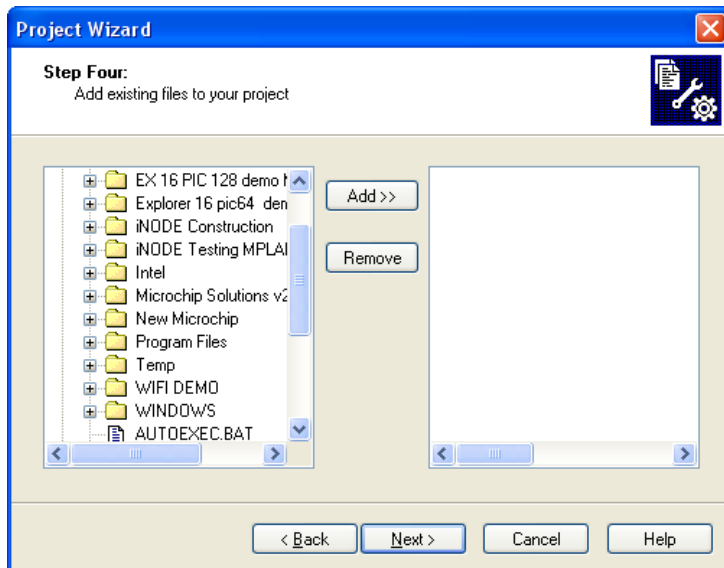
Here you will scroll to the preferred PIC Microcontroller that you will be using. For this tutorial we were using the PIC24FJ128GA010. Click Next.

Now you will see this window:



This window is very important on the way its configured because it is what will drive the previously downloaded C-30 Compile. You will click on the Toolsuite Contents, and then click browse. The browse button will already direct you to the correct path for the .exe file. For each Toolsuite contents, browse to the .exe file that is in () and click ok. Once you have the four Toolsuites directed to the correct path and you check the box “Store tool locations in project” click Next. Now browse to the path C:\Example. Here the software will create a folder containing your MPLAB project, and click Next.

This next step is the most difficult. You will now be prompted with the dialog box:



This is where you will build the project files from the TCP/IP stack download. There are files necessary for the stack to function properly, and files for the demo applications. First we will need to navigate to C:\Microchip Solutions v2011-10-18\TCPIP\Demo App\Precompiled Hex, and click on the C30-EX16_MRF24WB 24FJ128GA010.hex file. Once it is highlighted, click the Add button. You will also need to add the linker script specific to our PIC. You will navigate to C:\Program Files\Microchip\MPLAB C30\support\gld, and select the p24FJ128GA010.gld file, click the Add button.

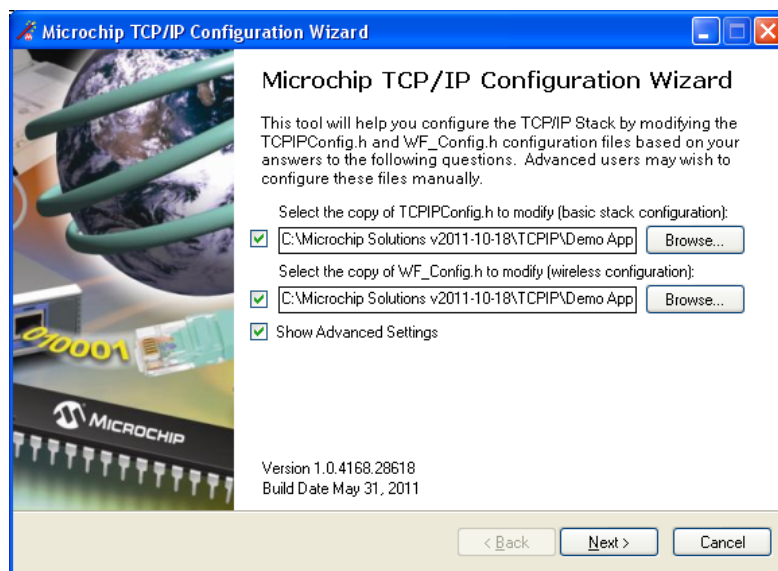
Now you will need to add the specific source files and header files for the TCP/IP stack, as well as the source files and header files for the proper demo that you want to run on your device. To obtain the necessary source files for the TCP/IP stack, navigate to C:\Microchip Solutions v2011-10-18\Microchip\TCPIP Stack. From here, click on the first file to highlight it, hold the shift key, and scroll to the bottom of the window. Now click on the last file in the window. Once all the .c files are highlighted, click the Add button. Now you will need to add all the source files for the TCP/IP Wi-Fi applications. Navigate to C:\Microchip Solutions v2011-10-18\Microchip\TCPIP Stack\WiFi and select all the .c files in the folder, and click the Add button. You now have all the proper source files necessary for TCP/IP stack and application demos.

Now we need to get all the proper header files. To achieve this navigate to C:\Microchip Solutions v2011-10-18\Microchip\Include, select all the .h files and add them to the project. Next navigate to C:\Microchip Solutions v2011-10-18\Microchip\Include\TCPIP Stack, select all of the .h files, and add them to the project. Finally you will need to add the special configure file to the project. Navigate to C:\Microchip Solutions v2011-10-18\TCPIP\Demo App\Configs and select the TCPIP MRF24WB.h file, and add this to your project. You have now successfully built your TCP/IP stack, and demo application code.

Custom Demo Application

Configuring the Stack

The first thing you will want to do is make sure the MPLAB project you previously created is opened. The following modifications will be done on the configuration files, and will automatically update your project. Before running any of the demo application code, you will first need to properly configure a couple files. The TCPIP Stack download comes with a TCPIP Configuration Wizard. Browse to the Start Menu -> All Programs -> Microchip Solutions v2011-10-18 -> TCPIP Stack -> and click on the TCPIP Configuration Wizard. We will modify the TCPIP Stack to serve webpages. You will see the following window:

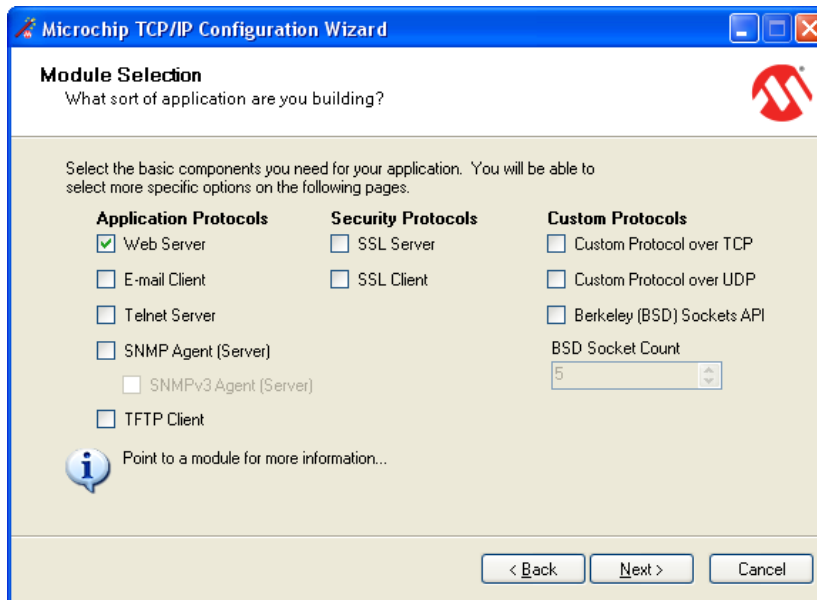


In the “Select the Copy of TCPIPConfig.h to modify” browse to C:\Microchip Solutions v2011-10-18\TCPIP\Demo App\Configs, and select the TCPIP MRF24WB.h file. In the next window “Select the Copy of WF_Config.h to Modify” browse to C:\Microchip Solutions v2011-10-18\TCPIP\WiFi

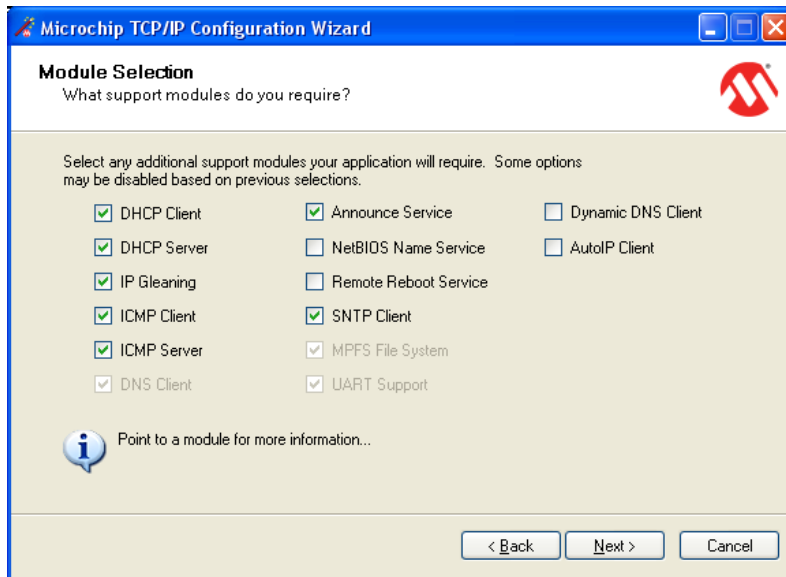
Using the Explorer 16 Development Board for Wi-Fi Applications

2011

Console and select WF_Config.h file. Once this is done select the Next button. Now you will see this window:



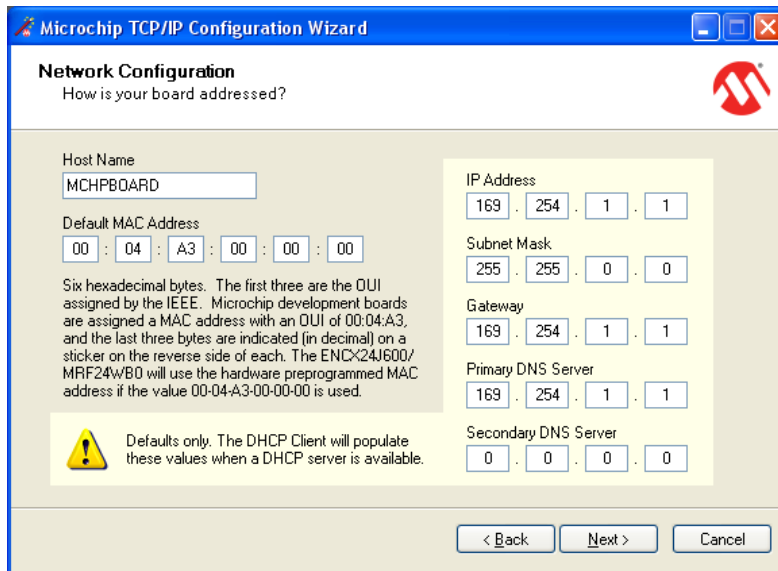
This is where you will define the configuration necessary for the specific application you are trying to build. The Web Server demo application provides an example for building a custom HTTP application using the HTTP2 server and allows several other demo features to be accessed and controlled via web interface. Select the Web Server box and select Next. For configuration purposes in the "Module Selection, What example modules would you like to include" dialog box, select all boxes except the "Serial to Ethernet Bridge" module and click Next. Now you will see the "Module Selection, What support modules do you require?" dialog box. The selected modules will be application specific.



- DHCP Client-Allows device to obtain network config from a DHCP Server
- DHCP Server-Allows device to act as a DHCP Server for a single node
- IP Gleaning-Provides a small, unorthodox replacement for DHCP for private networks
- ICMP Client-Allows device to send pings to remote nodes
- ICMP Server-Allows device to respond to ping requests
- Announce Service-Enables Ethernet tool to be found by sending a UDP announcement packet
- NetBIOS Name Service-Allows device to respond to NETBios name queries
- Remote Reboot Service-Allows device to reboot remotely by sending a special UDP Packet
- SNTP Client-ability to obtain current time from pool of global SNTP servers
- Dynamic DNS Client-enables IP updates to dynamic DNS
- AutoIP Client-Enables Auto IP Supprt.

Select the boxes in the diagram and click Next.

The next dialog box is the Network Configuration profile. The Host Name is your NETBios ID. This can be left in the default MCHPBOARD. Click next to continue forward with all the default settings.



Microchip TCP/IP Configuration Wizard

Network Configuration
How is your board addressed?

Host Name: MCHPBOARD

Default MAC Address: 00 : 04 : A3 : 00 : 00 : 00

Six hexadecimal bytes. The first three are the OUI assigned by the IEEE. Microchip development boards are assigned a MAC address with an OUI of 00:04:A3, and the last three bytes are indicated (in decimal) on a sticker on the reverse side of each. The ENC24J600/MRF24WB0 will use the hardware preprogrammed MAC address if the value 00-04-A3-00-00-00 is used.

IP Address: 169 . 254 . 1 . 1

Subnet Mask: 255 . 255 . 0 . 0

Gateway: 169 . 254 . 1 . 1

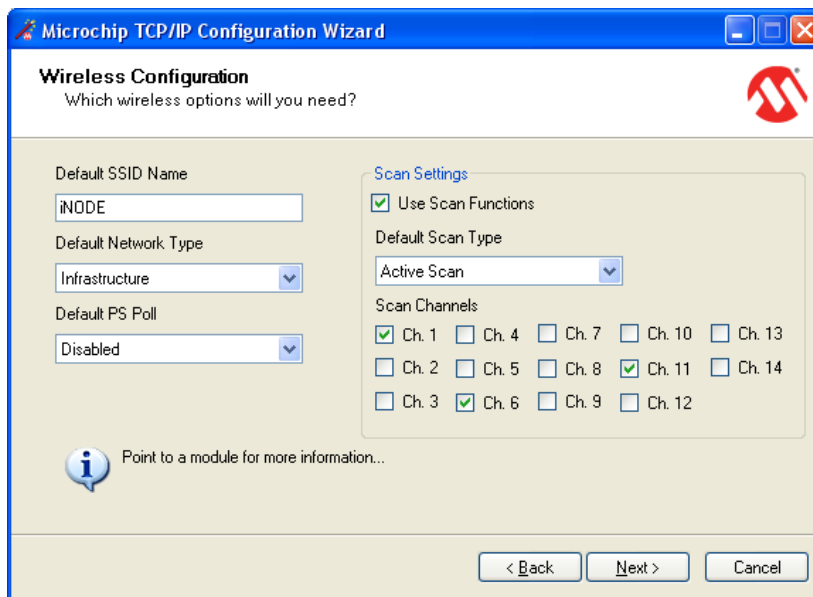
Primary DNS Server: 169 . 254 . 1 . 1

Secondary DNS Server: 0 . 0 . 0 . 0

Defaults only. The DHCP Client will populate these values when a DHCP server is available.

< Back Next > Cancel

You will now see this window. Here you can modify the host name of your device in the Default SSID Name section. This is the name of the Wi-Fi Network you intend to join. Active Scan allows the device to respond to probe requests, as well as transmit a beacon every 100ms. Channels 1, 6 and 11 the common channels commercial Wi-Fi routers operate on. Click Next.



Microchip TCP/IP Configuration Wizard

Wireless Configuration
Which wireless options will you need?

Default SSID Name: INODE

Default Network Type: Infrastructure

Default PS Poll: Disabled

Scan Settings

☒ Use Scan Functions

Default Scan Type: Active Scan

Scan Channels:

☒ Ch. 1	☐ Ch. 4	☐ Ch. 7	☐ Ch. 10	☐ Ch. 13
☐ Ch. 2	☐ Ch. 5	☐ Ch. 8	☒ Ch. 11	☐ Ch. 14
☐ Ch. 3	☒ Ch. 6	☐ Ch. 9	☐ Ch. 12	

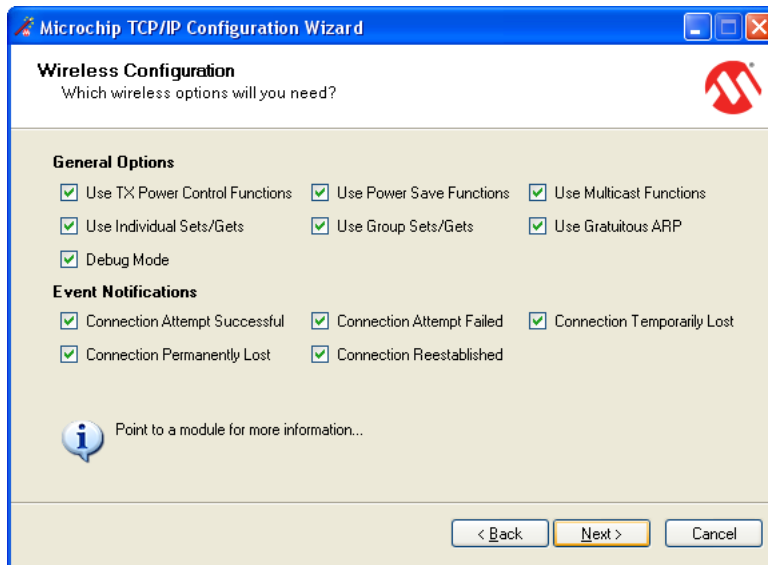
Point to a module for more information...

< Back Next > Cancel

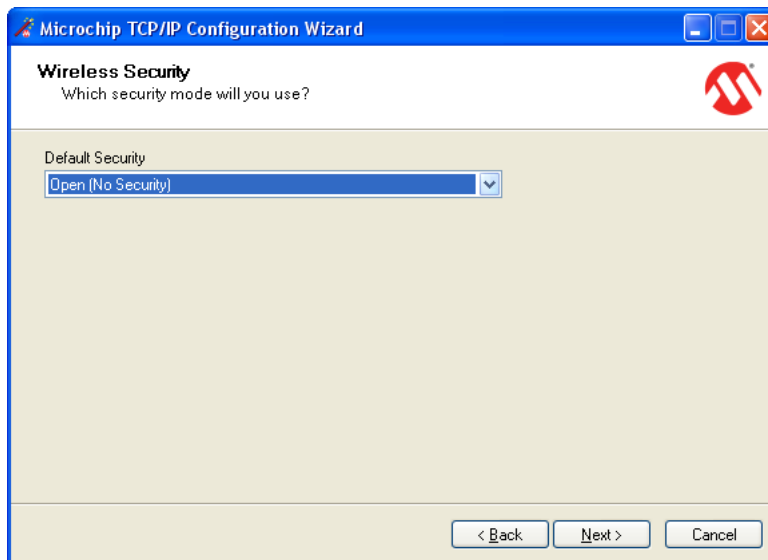
Using the Explorer 16 Development Board for Wi-Fi Applications

2011

Now you will see this dialog window. These are the options that you want to include in the Wireless configuration. You can point to a module for a definition from Microchip for these options. For this demo application, it is safe to select all options. Click next.



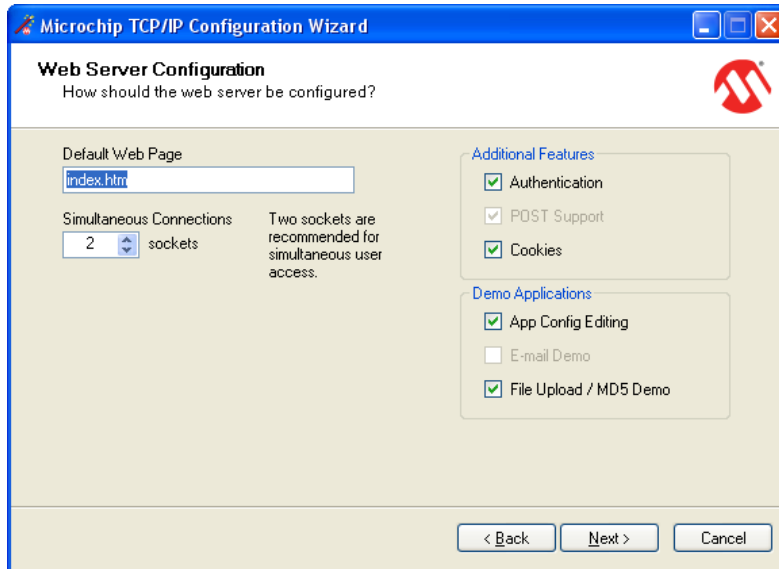
Now you will define security to your Wi-Fi device. For custom HTTP servers it is common to add a WEP or WPA security to your device. But for simplicity of this tutorial we will use the No Source security. Click Next.



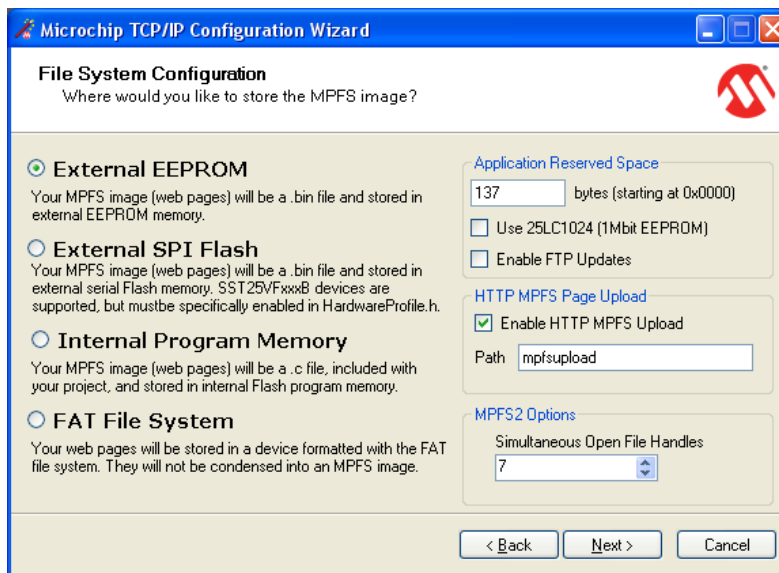
Using the Explorer 16 Development Board for Wi-Fi Applications

2011

This is where you will define the destination of the web server location. This page can be set as default values. Click Next.



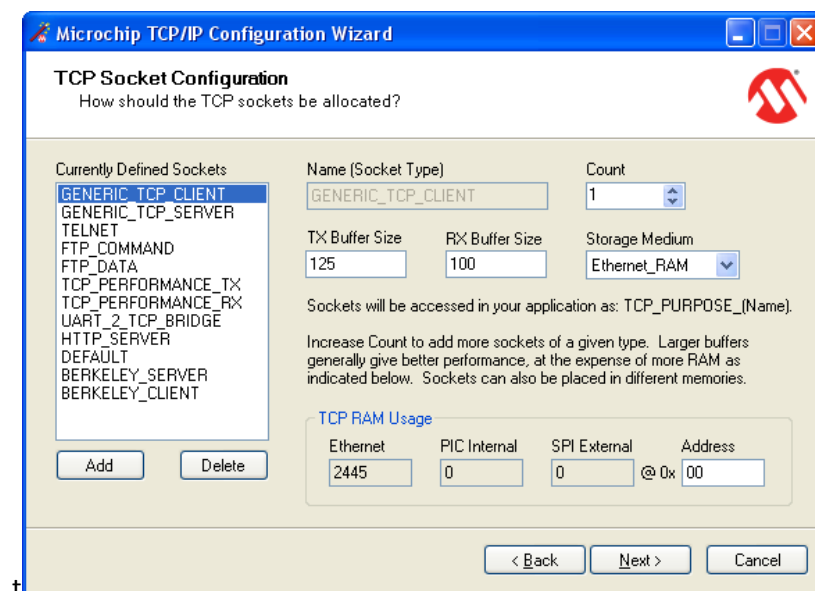
Now you will need to configure the file system. This is dependent on where you want to store your web page .bin format. You can allocate a reserved space in the different memory locations. Click Next.



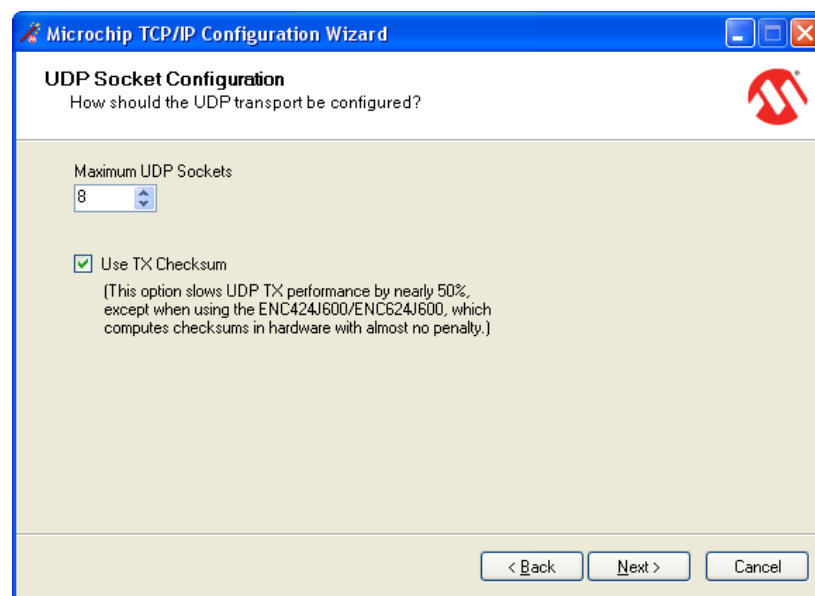
Using the Explorer 16 Development Board for Wi-Fi Applications

2011

Now you will need to configure the TCP Socket information. This will be left as its default settings. Select next.



Now configure the UDP Socket configuration by selecting the default settings. Click next. And then click finish.



Running the Demo

Once you have properly configured the MPLAB workspace, you will go to the Debugger tab at the top of MPLAB -> Select Tool -> MPLAB ICD 2. After this you will go to Project -> build all. The application should build with no errors. Then go to Debugger -> Program. This will load the program onto the PIC Microcontroller. Once it is loaded, in MPLAB go to Project -> Run. This will automatically run your module. Now you can open an internet connection and browse to index.htm, and you will see the message stored in your EEPROM.

Conclusion

In this tutorial, we walked an individual through the process of developing a Wi-Fi device using the Explorer 16 development board, and PIC microcontroller. Microchip has many development platforms. We were able to set up the Microchip demo applications to serve as a web page browser. Microchip Wi-Fi transceiver modules can be used for many other applications beyond serving web page browsers.

References

[1] www.microchip.com

[2] TCPIP Stack Help @ <http://ww1.microchip.com/downloads/en/AppNotes/00833b.pdf>

[3] DM240001 user guide @ <http://ww1.microchip.com/downloads/en/DeviceDoc/Explorer%2016%20User%20Guide%2051589a.pdf>

[4] MRF24WB0MA Wi-Fi PICtail/PICtail Plus Daughter Board @ <http://ww1.microchip.com/downloads/en/DeviceDoc/51912A.pdf>