MiiNePort-IDE Getting Started Guide

First Edition, March 2012

www.moxa.com/product

MiiNePort-IDE Getting Started Guide

Copyright Notice

Trademarks

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of Moxa.

Moxa provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Moxa reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Moxa assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Technical Support Contact Information

www.moxa.com/support

Moxa Americas
Toll-free: 1-888-669-2872
Tel: +1-714-528-6777
Fax: +1-714-528-6778

Moxa China (Shanghai office)
Toll-free: 800-820-5036
Tel: +86-21-5258-9955
Fax: +86-10-6872-3958

Moxa Europe
Tel: +49-89-3 70 03 99-0
Fax: +49-89-3 70 03 99-99

Moxa Asia-Pacific
Tel: +886-2-8919-1230
Fax: +886-2-8919-1231

Table of Contents

This document provides detailed instructions for the development of application-specific firmware for the MiiNePort.

1

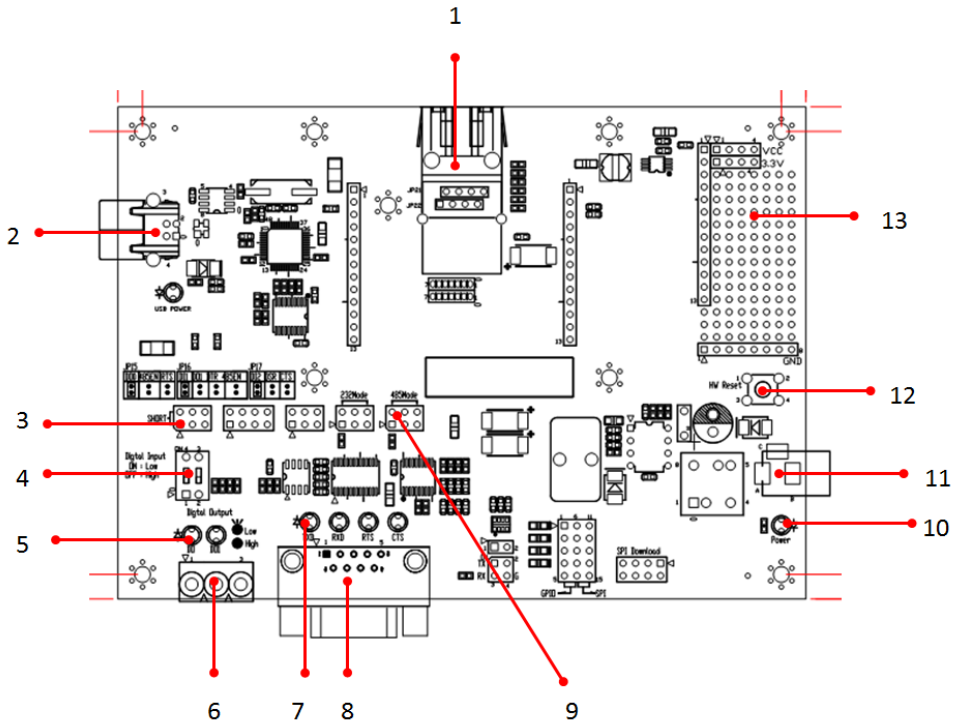
The following topics are covered in this chapter:

- **Step 1: Connect MiiNePort USB to PC**
- **Step 2: Create MiiNePort IDE Project**
- **Step 3: Start to develop**
- **Step 4: Compile the project**
- **Step 5: Upload Firmware**
- **Final Step**

Step 1: Connect MiiNePort USB to PC

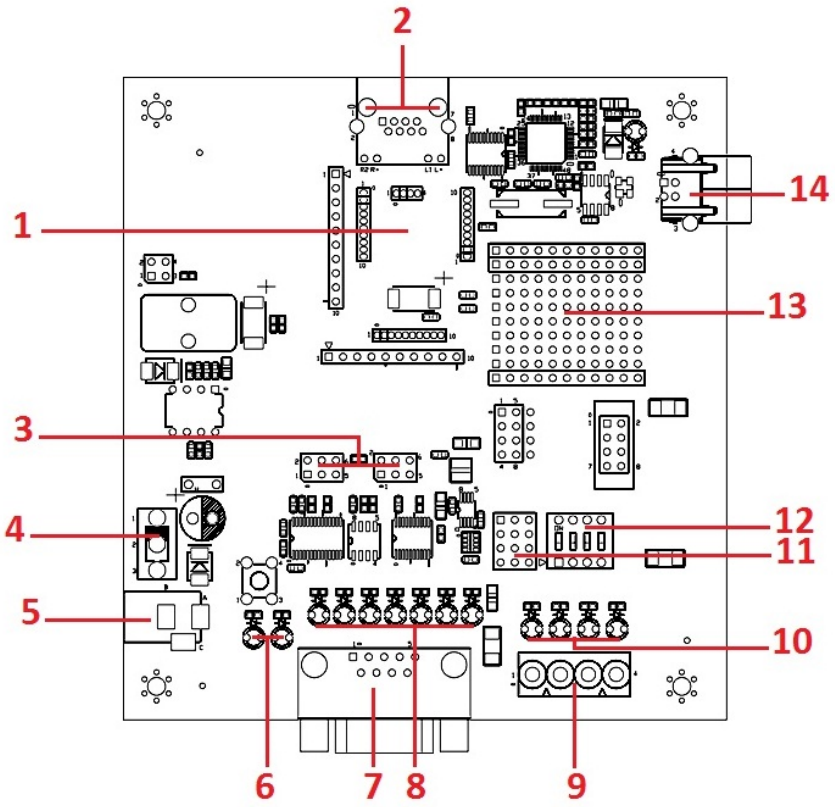
First, power on the MiiNePort evaluation board and connect the USB port of the MiiNePort to the PC. Please refer to the diagrams below for the location of the USB ports on the evaluation boards.

MiiNePort E1-SDK Evaluation Board



Number	Description
1	MiiNePort E1 Module Location
2	USB Type B Connector (Debug)
3	Configurable Pin Jumper
4	Digital Input Switch
5	Digital Output LED
6	Digital IO Terminal Block
7	Serial Port Status LED
8	DB9 Male Connector
9	Serial Interface Jumper
10	Power LED
11	Power Jack
12	Restart Button
13	Circuit Pad

MiiNePort E2-SDK Evaluation Board



Number	Description
1	MiiNePort E2 Module Location
2	Ethernet RJ45 Connector
3	Serial Interface Jumper
4	Power Switch
5	Power Jack
6	Power & Ready LED
7	DB9 Male Connector
8	Serial Port Status LED
9	Digital I/O Terminal Block
10	Digital Output LED
11	Digital Input/Output Mode
12	Digital Input Switch
13	Circuit Pad
14	USB Type B Connector (Debug)

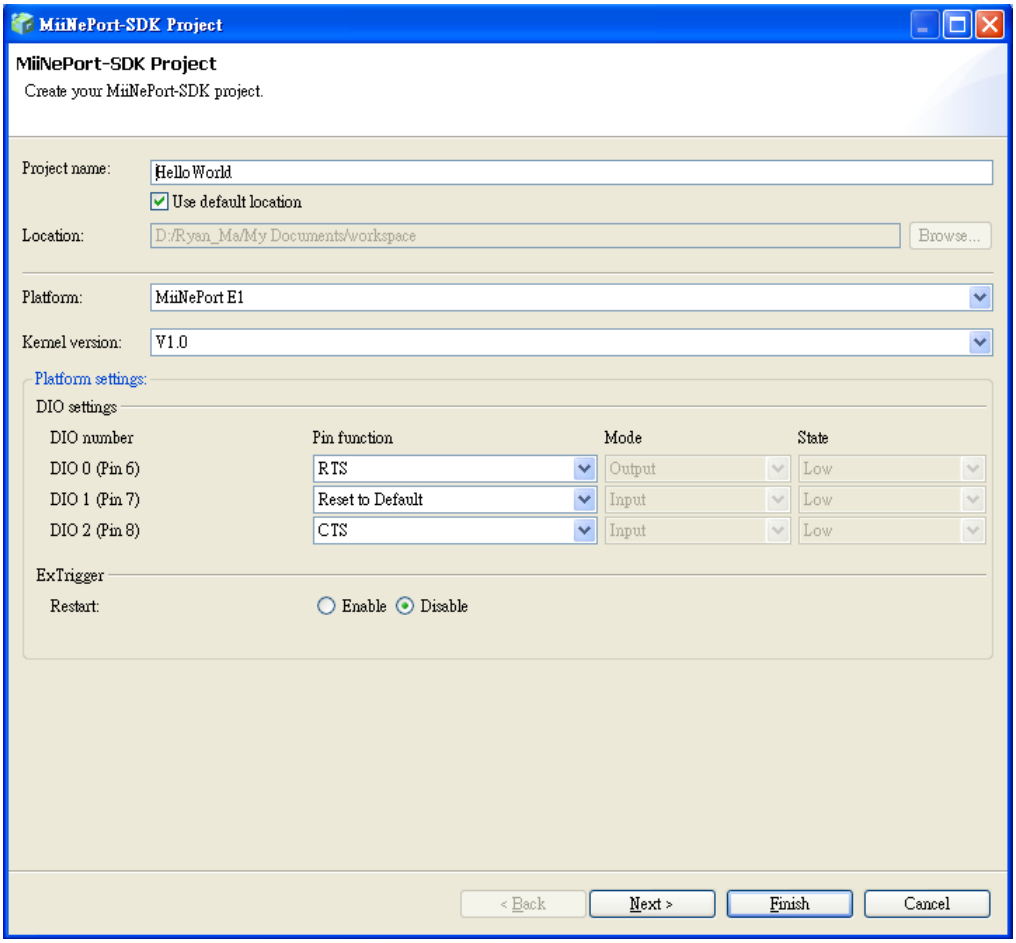
Step 2: Create Your Project

Prior to firmware development, the MiiNePort-SDK device must be connected to the unit. For developing customized firmware, MOXA provides a powerful wizard. You can compose the desired functions step by step. The following section will show you how to build a customized firmware.

Step 2-1: Create a new project by clicking File → New → MiiNePort-SDK Project.

Step 2-2: Project Name and DIO Settings

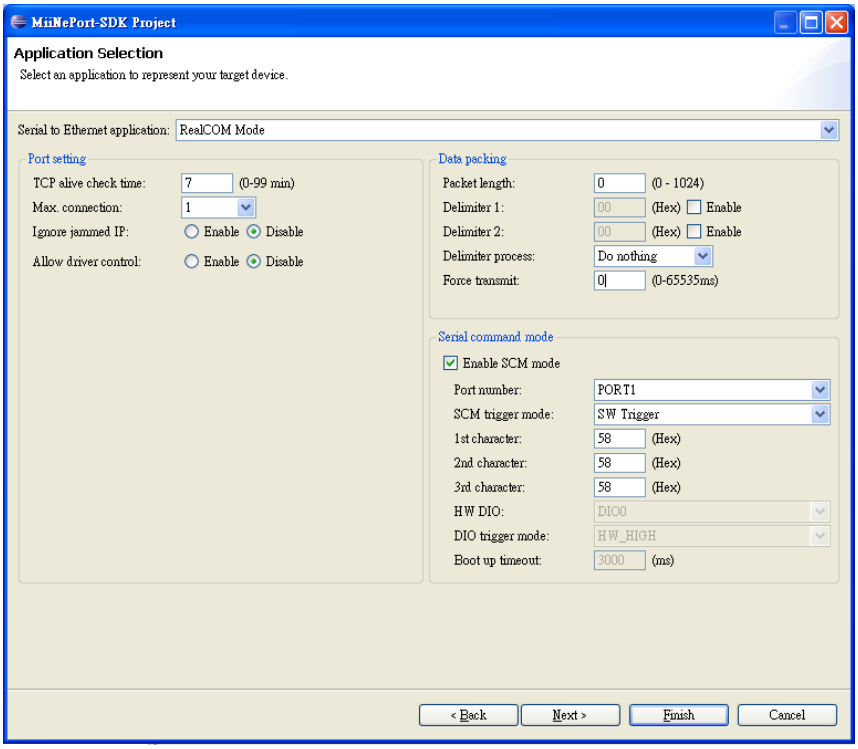
The project wizard will then begin, as shown in the figure below. Just fill in your desired value for each field and click **Next** to continue. Regarding the details of each parameter, please refer to the MiiNePort SDK user's manual.



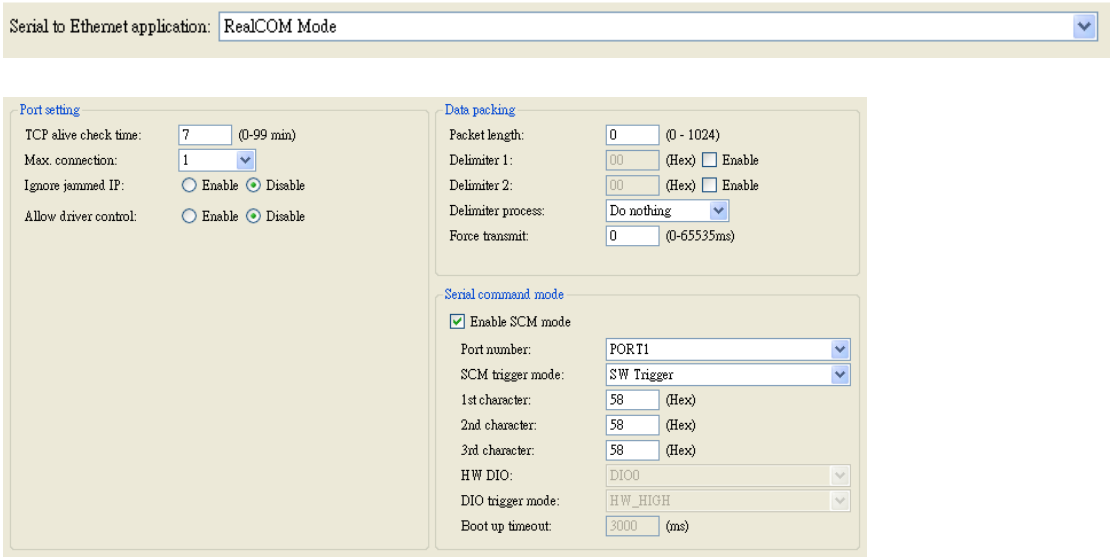
Step 2-3: Serial to Ethernet Application Settings

The next section is regarding serial to Ethernet application, data packing and serial command mode. Before reading this section, refer to Chapter 3 in the MiiNePort SDK user's manual:

Select the operation mode that best fits your device for serial to Ethernet application.



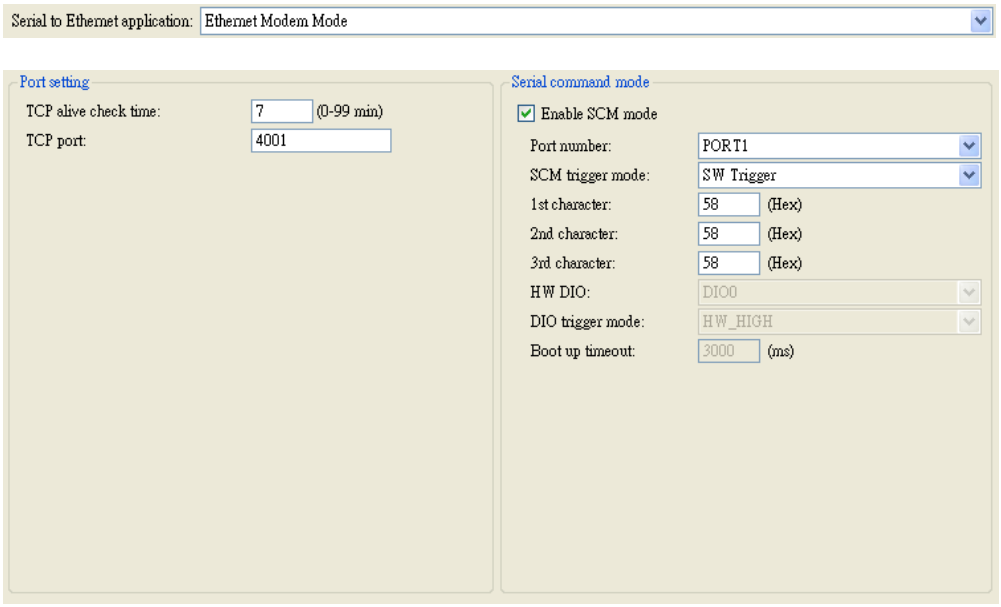
Real COM Mode



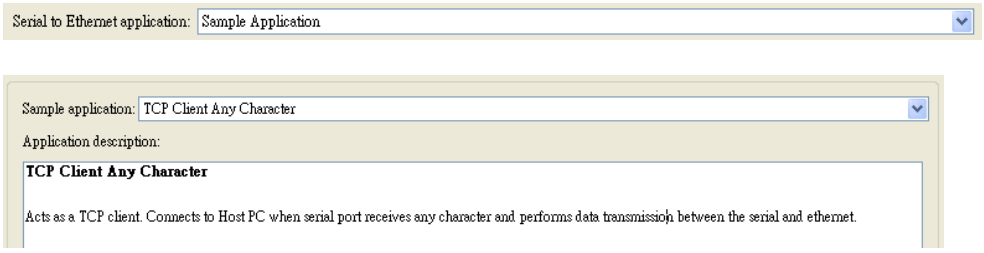
ATTENTION

To use Real COM mode, refer to Chapter 4 in MiiNePort SDK user's manual: Utility Console and Driver Installation to install the Real COM driver on Windows or Linux.

Ethernet Modem Mode

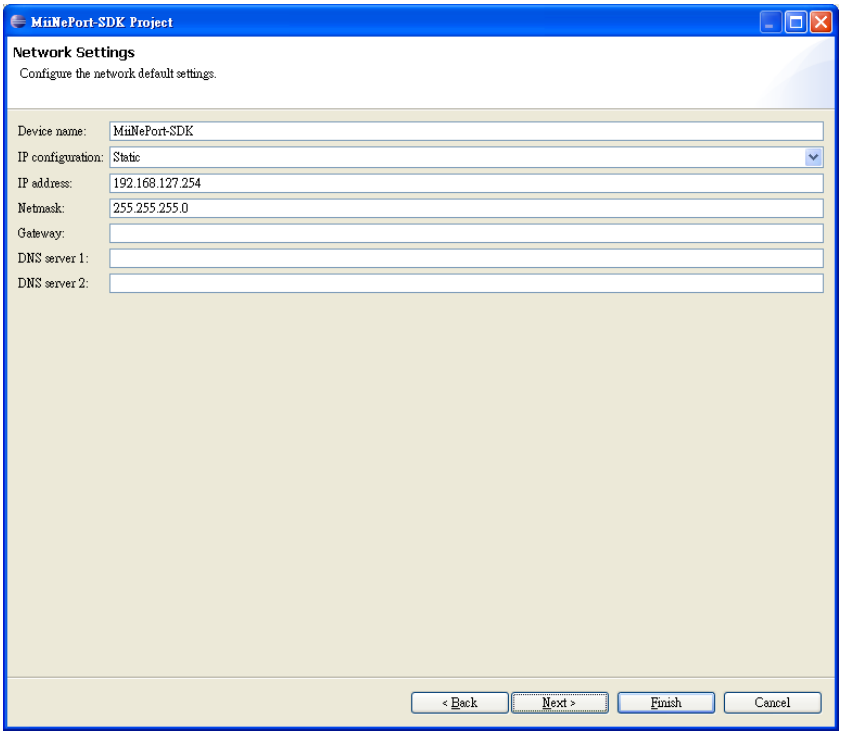


Sample Application



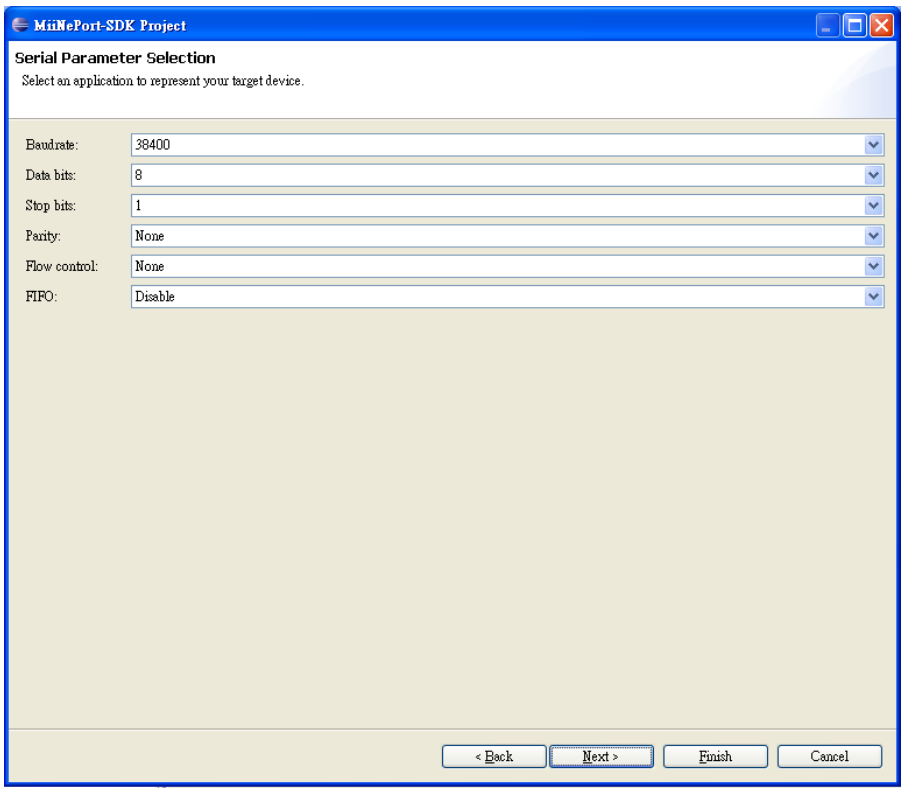
Step 2-4: Network Setting

You can assign IP configurations in this page.



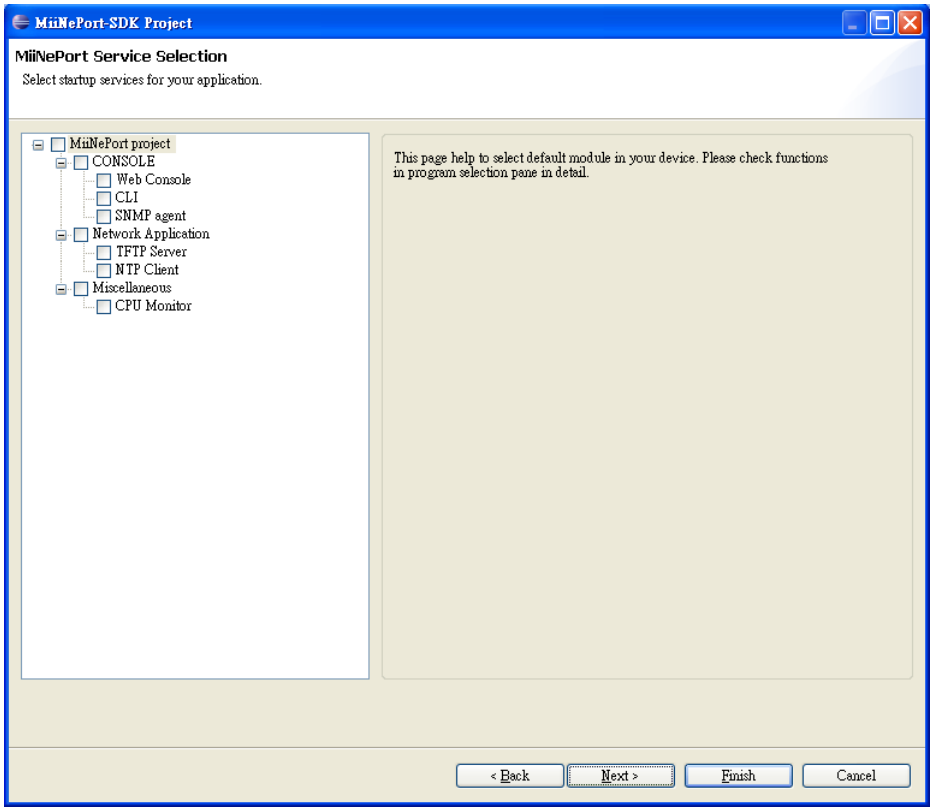
Step 2-5: Serial Parameter Settings

All serial parameters settings can be entered in this page.



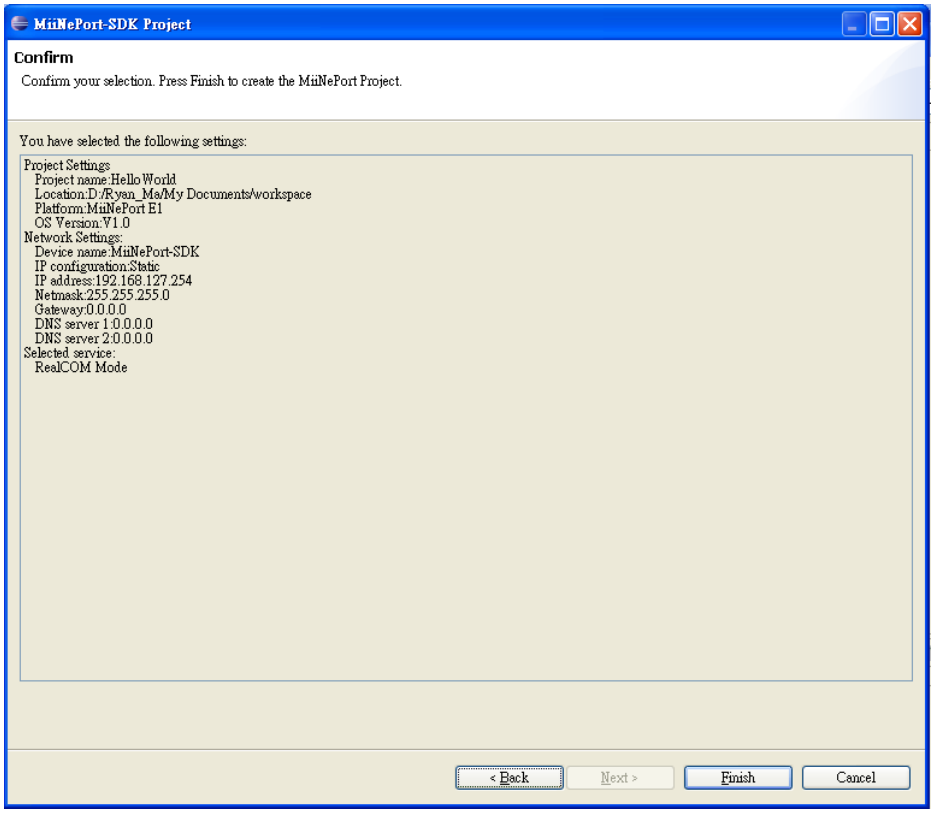
Step 2-6: MiiNePort Service Selection

MOXA provides many applications for the MiiNePort, check the desired applications and they will be applied to your customized firmware.

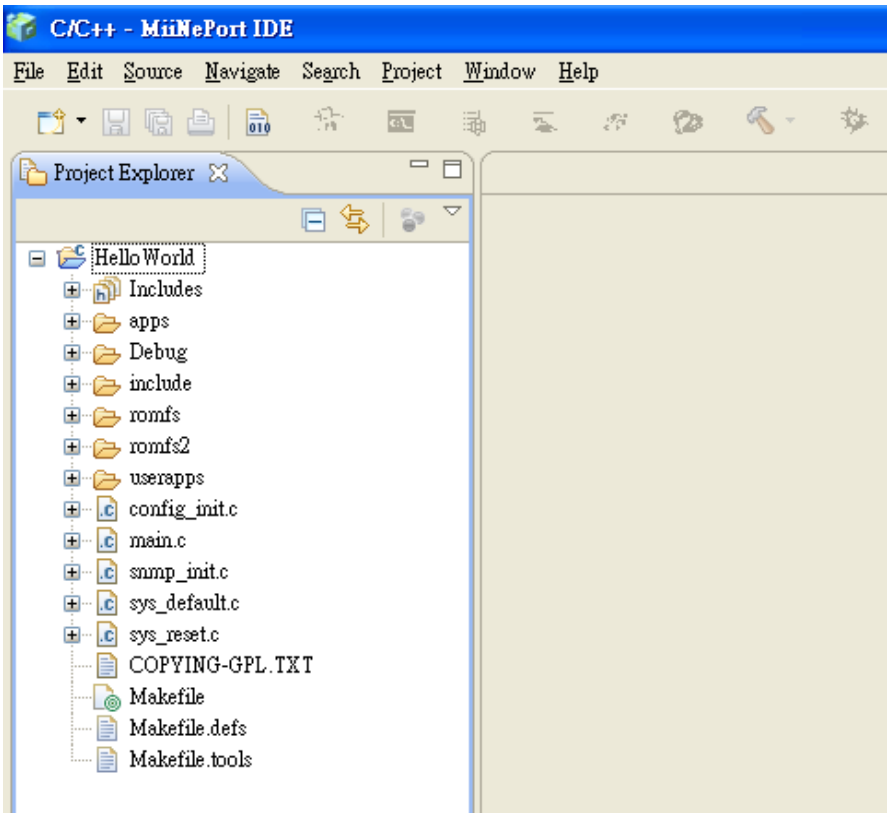


Step 2-7: Check Project Settings

This page will display the results you have selected. Click **Finish** to create this project.



Finally the project will be created on the project explorer list. Then you can modify or add source codes to develop your firmware.

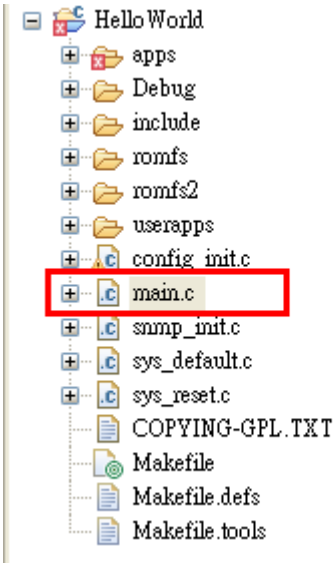


Step 3: Start to develop

After creating the project, an entry point is needed to start firmware development. The entry point is in the main function of main.c file. Of course, user can define user's AP version by this line first. For example: version 1.2.3, please modify it to AP_VER(1, 2, 3) as shown below:

```
/*
 * Define user's AP version.
 * Mandatory! Don't remove!
 *
 * Synopsis:
 *     SDK_AP_VER(main_v, sub_v, ext_v)
 * Parameters:
 *     main_v: main version 0-99.
 *     sub_v: sub version 0-99.
 *     ext_v: extension version 0-99.
 * Example:
 *     SDK_AP_VER(1, 3, 99) means version "1.3.99"
 */
AP_VER(1, 2, 3);
```

Please follow the figure below to find the main function and develop your code in main function.



```
int main(void)
{
    /*
     * Initialize platform.
     */
    platformInit();

    /*
     * Initialize system application and services.
     */
    sysAppsInit();

    /*
     * Create user task.
     */
    userAppsInit();

    return 0;
}
```

Step 4: Compile the project

After the source code is developed, the project needs to be compiled to build a firmware file.

Build project

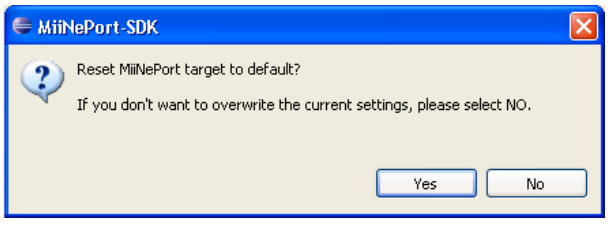
This function is used to build a customized firmware. This function can build two different types of firmware. If you select 'debug' type, the firmware will be downloaded into the MiiNePort. Then you can debug step by step. On the other hand, if you build release version firmware, it will generate a customized firmware without debug messages. Also, it won't write to the MiiNePort, you can upload it into the MiiNePort with an utility.

Step 5: Upload Firmware

Upload debug version firmware to MiiNePort

After compiling the source code, upload the debug firmware file to the MiiNePort to verify all functions with the provided source level debug tool.

First, build a debug firmware, and the MiiNePort will create a 'debug' folder. The firmware will be placed into this folder. Click the 'debug' button to upload the firmware into the MiiNePort via the USB port. Before upload, you will be prompted to confirm resetting the MiiNePort device to default configuration.



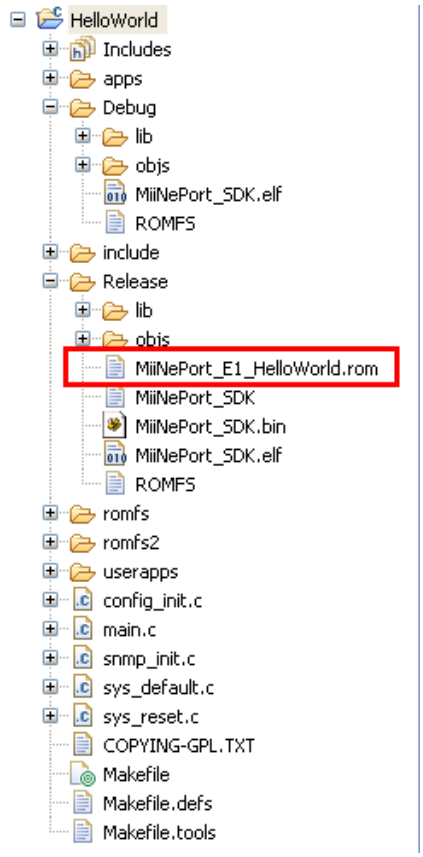
After uploading is successful, the firmware can be debugged.

Upload release version firmware to MiiNePort

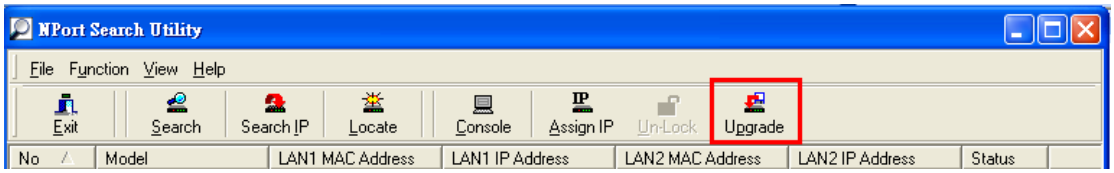
After compiling the source code, the release firmware has to be uploaded to the MiiNePort. Please remember connect the Ethernet cable to the MiiNePort first, then using the NPort Search Utility, the firmware can be uploaded.

First, build a release firmware.

Then MiiNePort will now create a folder which is called "Release" and put the firmware into this folder, users can find firmware in the sub-folder, 'objs'.



Please copy the .rom file to your desired path. The following example is copying the .rom file to C:\. Now you can upgrade the firmware with NPort Search Utility.



Select the target devices and the firmware you saved in the desired path, and click OK to start to upload the firmware to the MiiNePort devices via Ethernet.

After uploading is successful, the MiiNePort device will be updated with the new firmware.

Final Step

Congratulations! You've created, built, and debugged your own customize firmware successfully. After uploading your firmware to the MiiNePort device, you can setup your system with this firmware.