How to Run the Point-to-Point (PPP) Example Application

Introduction

This guide describes how to set up the PPP example application which is distributed together with the *Freescale MQX Real Time Communication Suite* (MQX RTCS) and how to establish a PPP communication between the PC and Freescale evaluation boards with the MQX support. There are two options that user can select:

- PPP Server (PC serves as a guest)
- PPP Client (PC serves as a host)

PPP Server

Step One - Rebuilding

Rebuilding MQX is the first step which needs to be done. For rebuilding the MQX RTOS, refer to *Freescale MQX Real-Time Operating System User's Guide*, Chapter 4: "Rebuilding MQX". The following MQX compile-time configuration options must be set in the user_config.h to ensure the correct functionality of the PPP example.

Option	Value
RTCSCFG_ENABLE_VIRTUAL_ROUTES	1
BSPCFG_ENABLE_ITTYB*	1*

* The application requires PPP device to be defined manually and being different from the default IO channel. ITTYB is suitable for most of Freescale evaluation boards with the MQX support, however one has to check if ITTYB is not associated with another functionality.

After inserting/modifying the compile-time configuration options stated above the MQX RTOS needs to be re-compiled as described in the *Freescale MQX Real-Time Operating System User's Guide.*

Once the MQX RTOS is re-built open the shell example project located on the following path: <install_dir>/src/rtcs/examples/shell/<IDE>/shell_<evb number>.mcp.

The config.h file of the shell example project contains the following PPP-related options which has to be changed/verified:

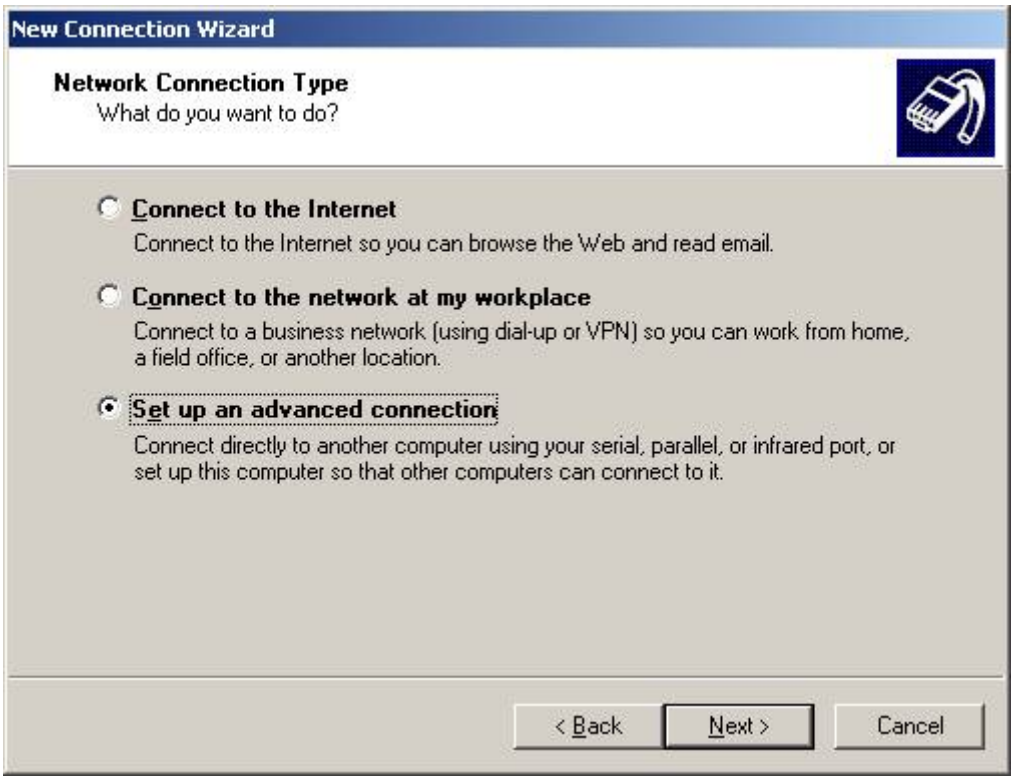
```
#define PPP_DEVICE_DUN          1 *
#define PPP_DEVICE_RAS          0
#define DEMOCFG_ENABLE_PPP      1
#define PPP_DEVICE              "ittyb:"
#define PPP_LOCADDR             IPADDR(192,168,0,216)
#define PPP_PEERADDR            IPADDR(192,168,0,217)
#define GATE_ADDR               IPADDR(192,168,0,1)
```

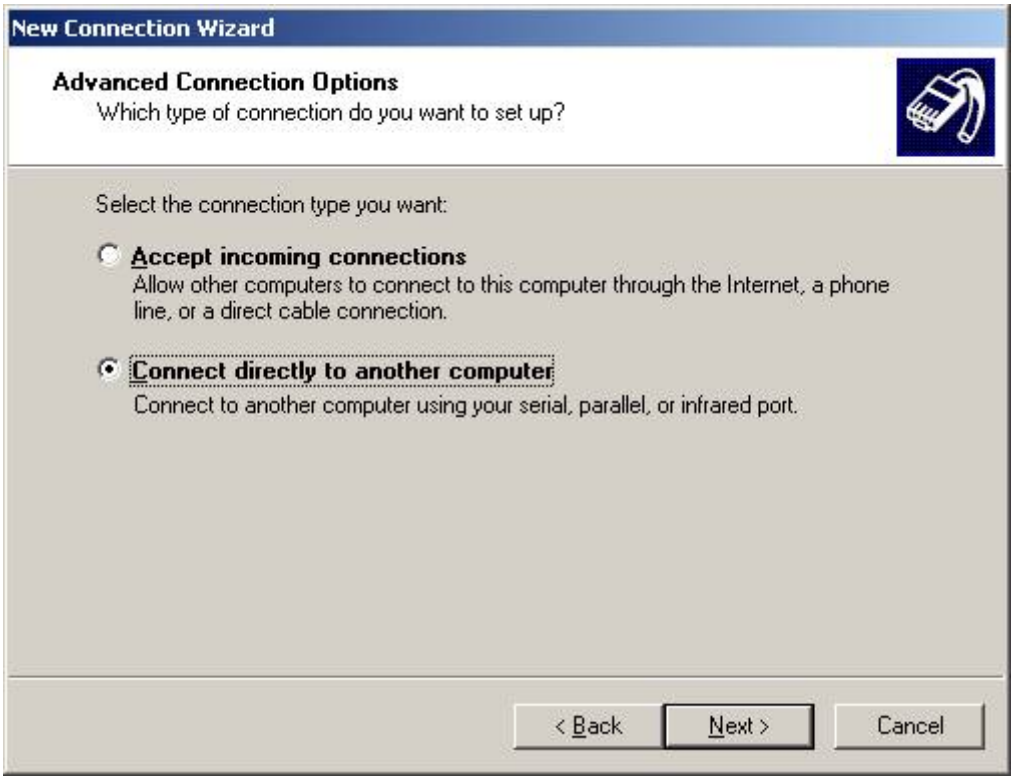
* In case of PPP server, please define PPP_DEVICE_DUN in "config.h" file.

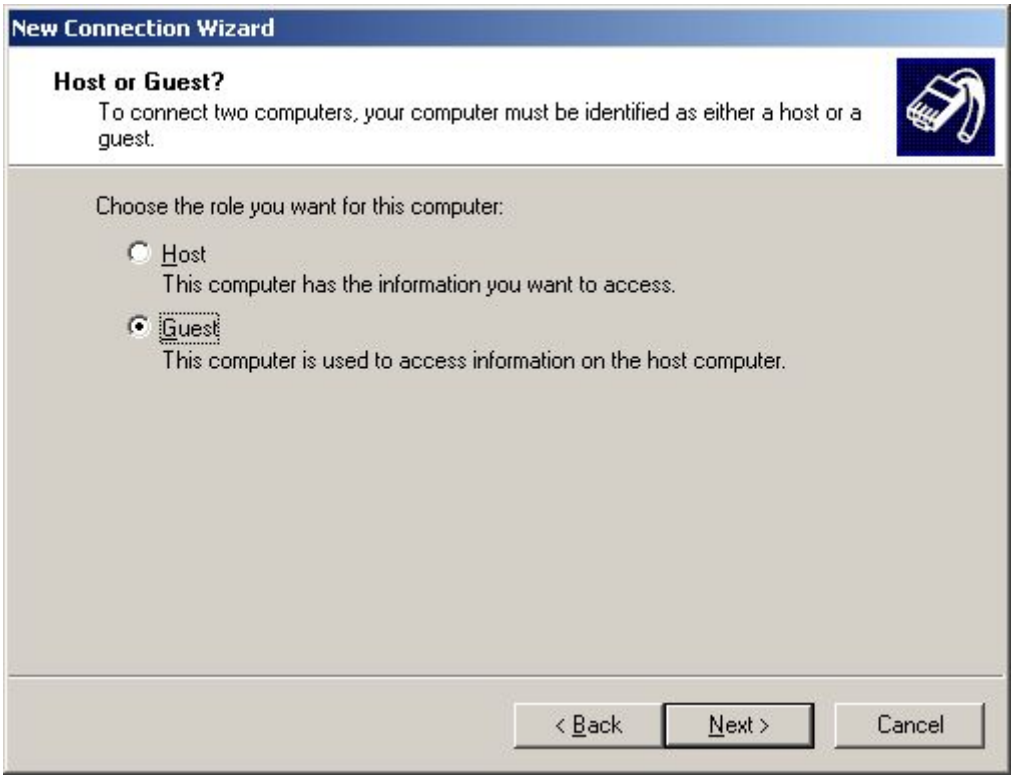
Once the config.h file is verified compile the shell example project and flash it to the evaluation board, see the *Freescale MQX Release Notes*.

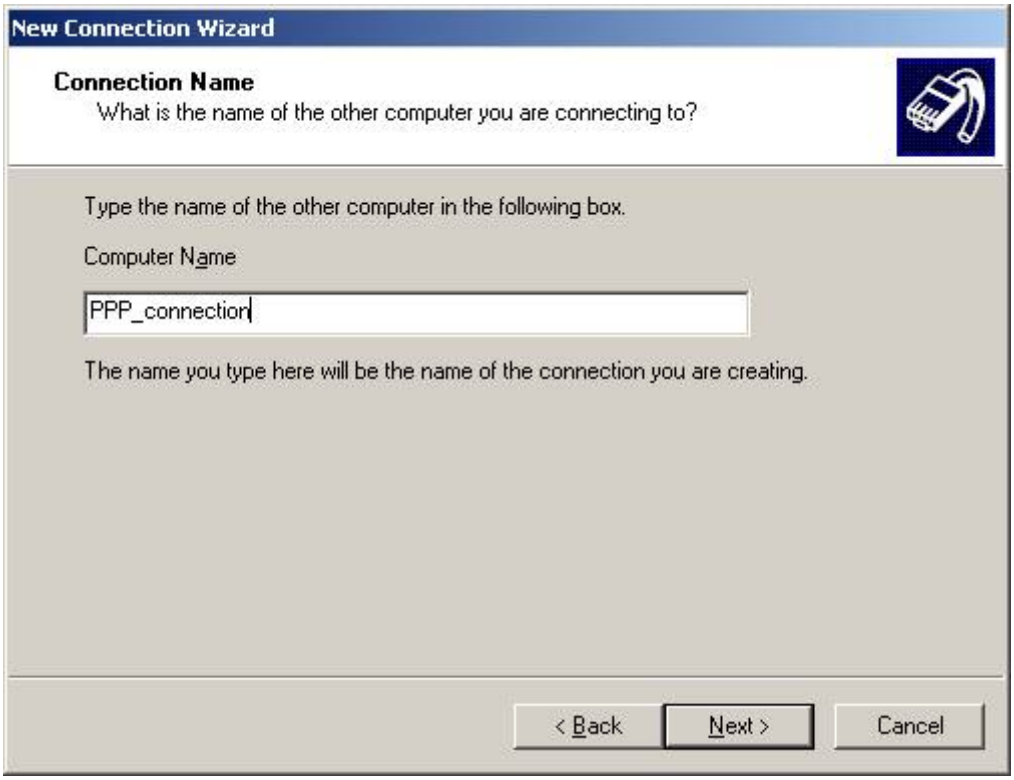
<u>Step Two</u> – Establishing the PPP Connection on the PC Side

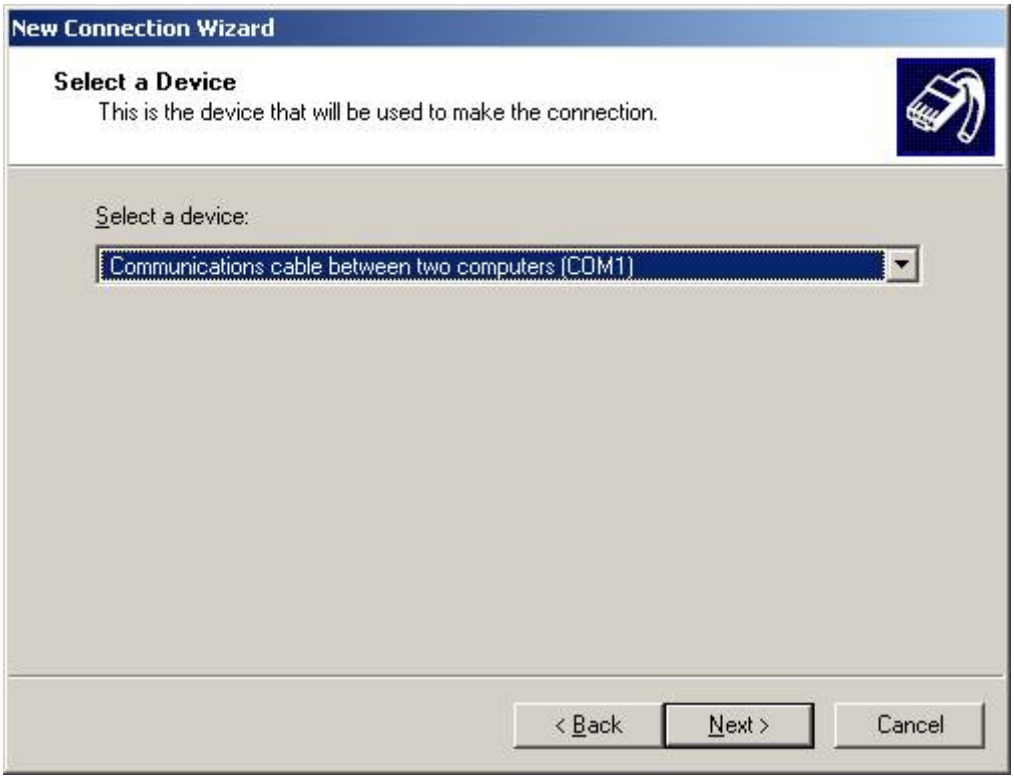
Open the MS Windows "Network Connections" dialog and start the "New Connection Wizard". Set properties of the PPP connection as depicted on the following pictures.




New Connection Wizard
Advanced Connection Options
Which type of connection do you want to set up?
Select the connection type you want:
Accept incoming connections
Allow other computers to connect to this computer through the Internet, a phone line, or a direct cable connection.
Connect directly to another computer
Connect to another computer using your serial, parallel, or infrared port.
< Back
Next >
Cancel

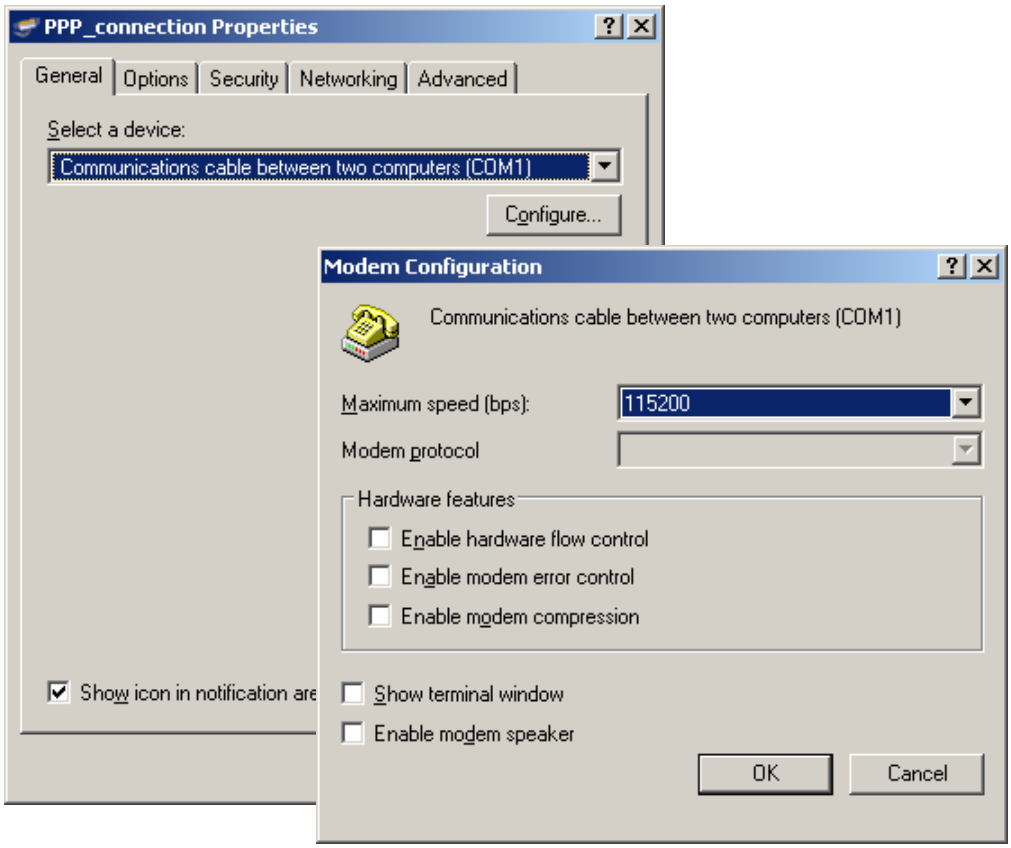

New Connection Wizard
Host or Guest?
To connect two computers, your computer must be identified as either a host or a guest.
Choose the role you want for this computer:
Host
This computer has the information you want to access.
Guest
This computer is used to access information on the host computer.
< Back
Next >
Cancel


New Connection Wizard
Connection Name
What is the name of the other computer you are connecting to?
Type the name of the other computer in the following box.
Computer Name
PPP_connection
The name you type here will be the name of the connection you are creating.
< Back
Next >
Cancel

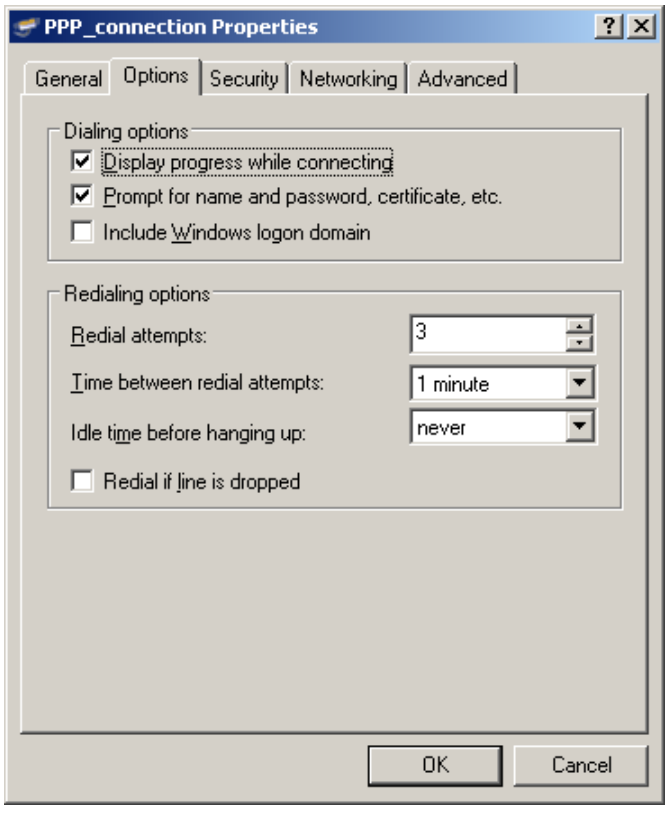

New Connection Wizard
Select a Device
This is the device that will be used to make the connection.
Select a device:
Communications cable between two computers (COM1)
< Back
Next >
Cancel

Once the new PPP connection is created set its properties as follows:

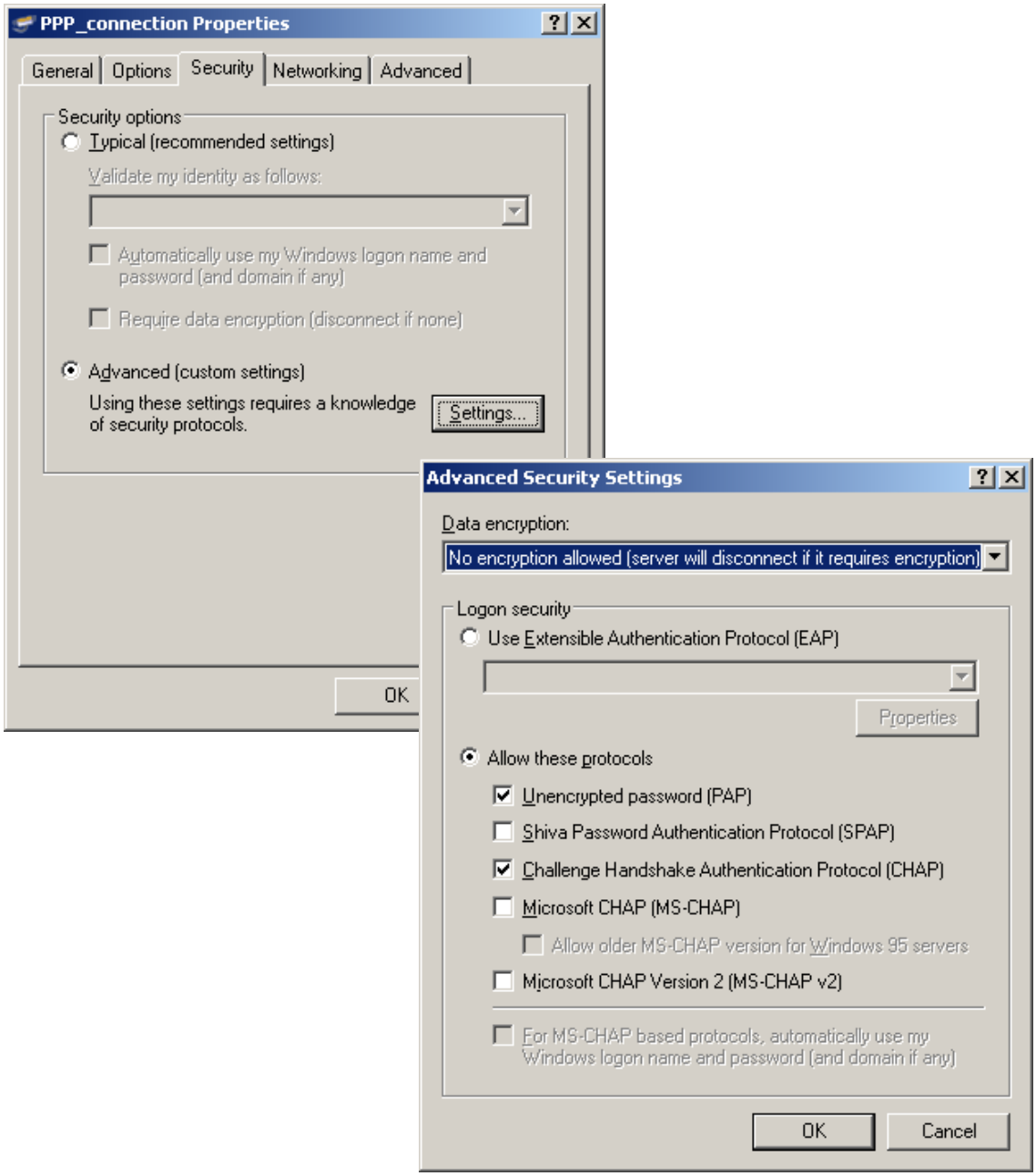
- under General, set the baud rate to match the embedded board



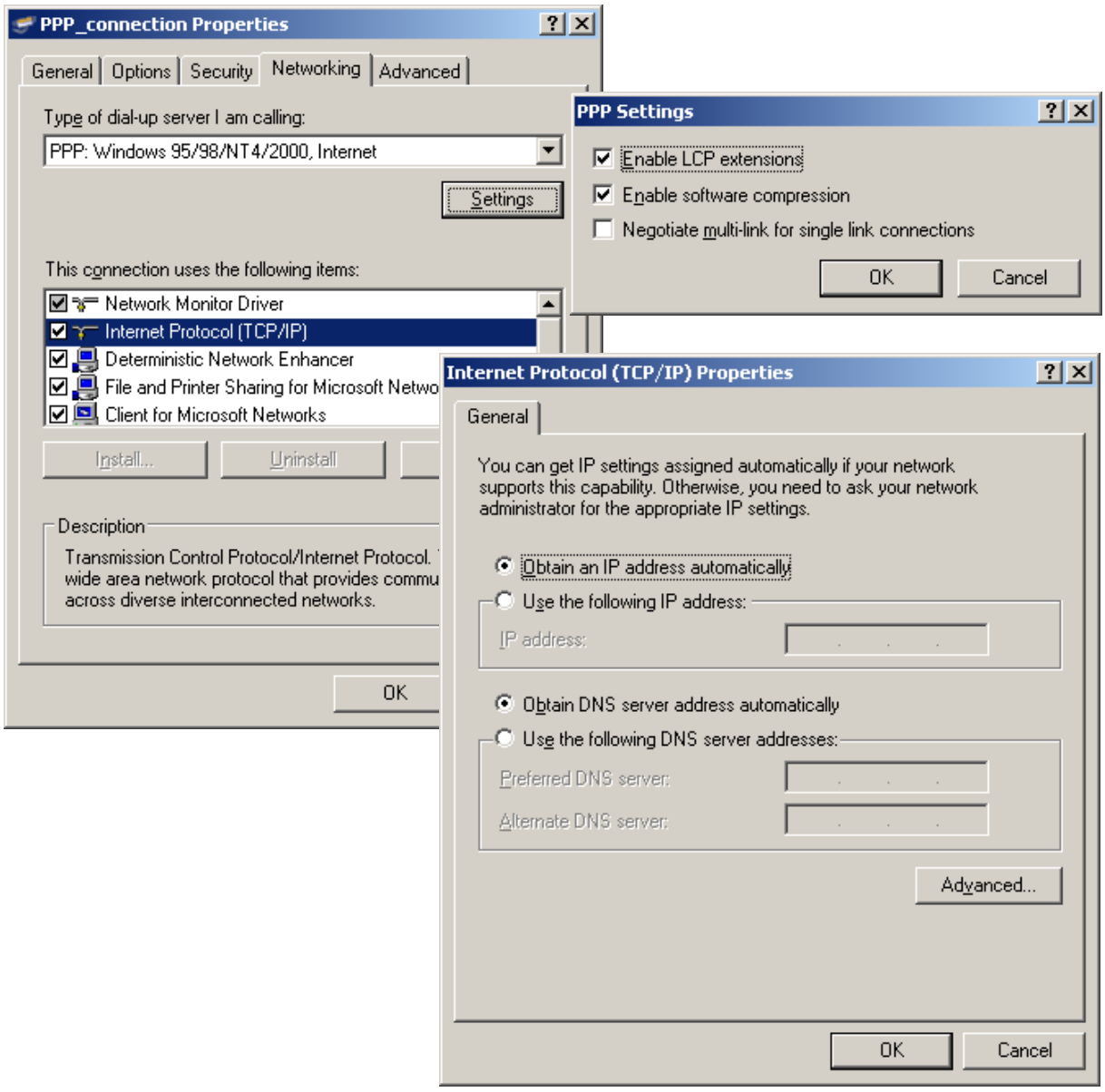
- under Options, select Display progress, and Prompt for name and password



- under Security, select Advanced, and in there allow PAP and CHAP protocols

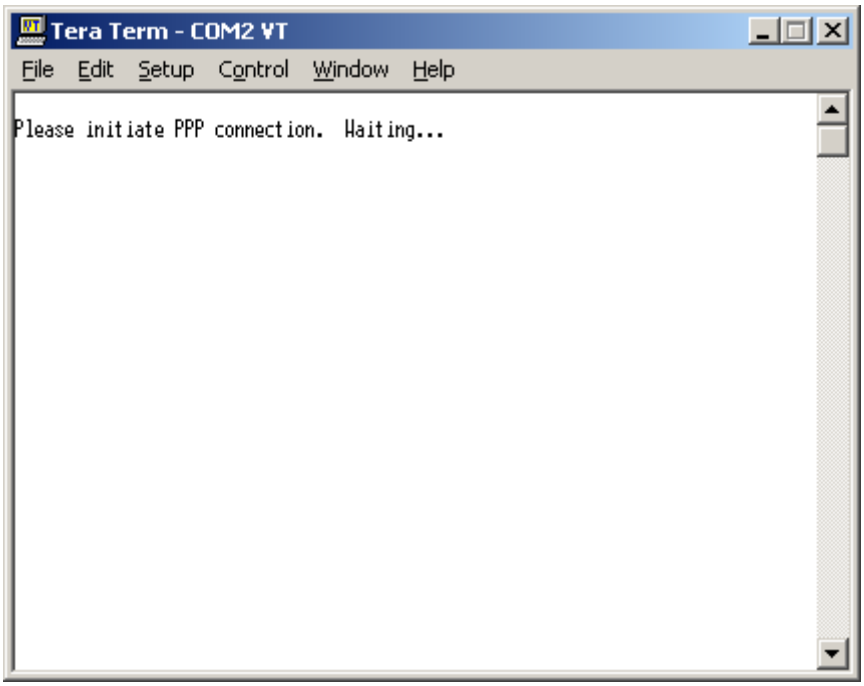


- under Networking, chose PPP, Windows 95/98/NT/2000, Internet, and the settings for that are Enable LCP, and Enable software compression, but no multilink. TCP/IP properties are set to obtain IP and DNS server address automatically.

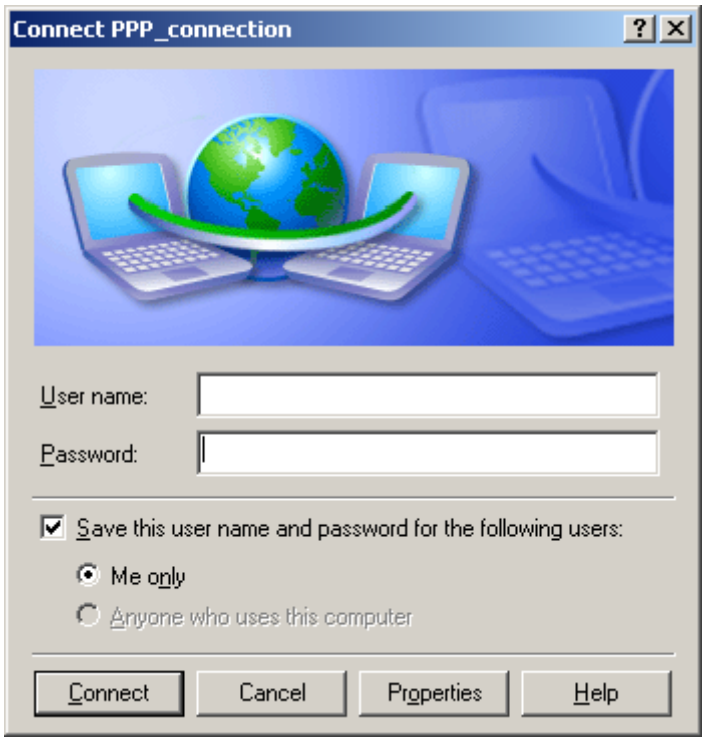


Step Three – Run the Example Application

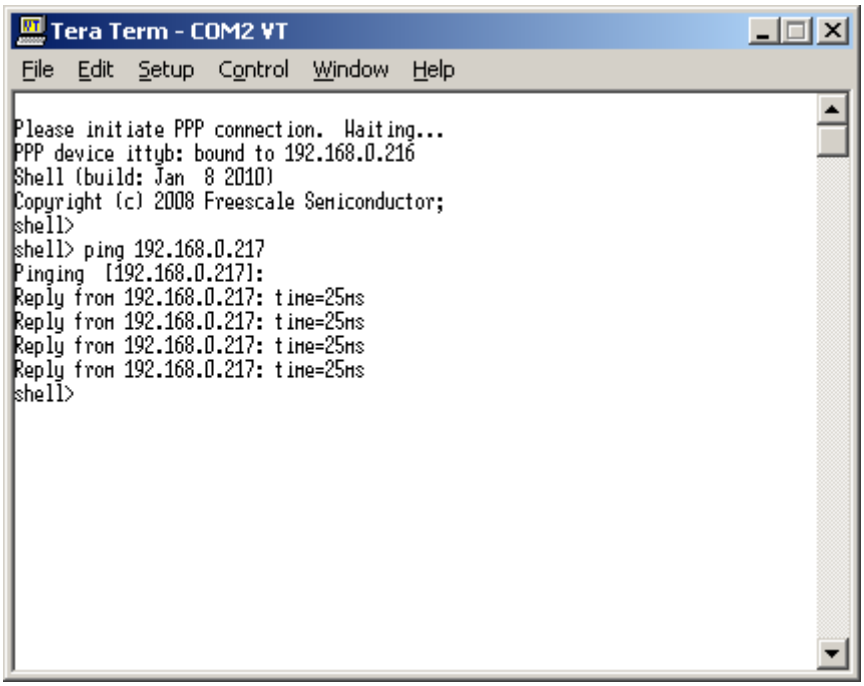
Once the example application is loaded into the Flash or MRAM memory you can start it. The following prompt occurs on the default serial console.



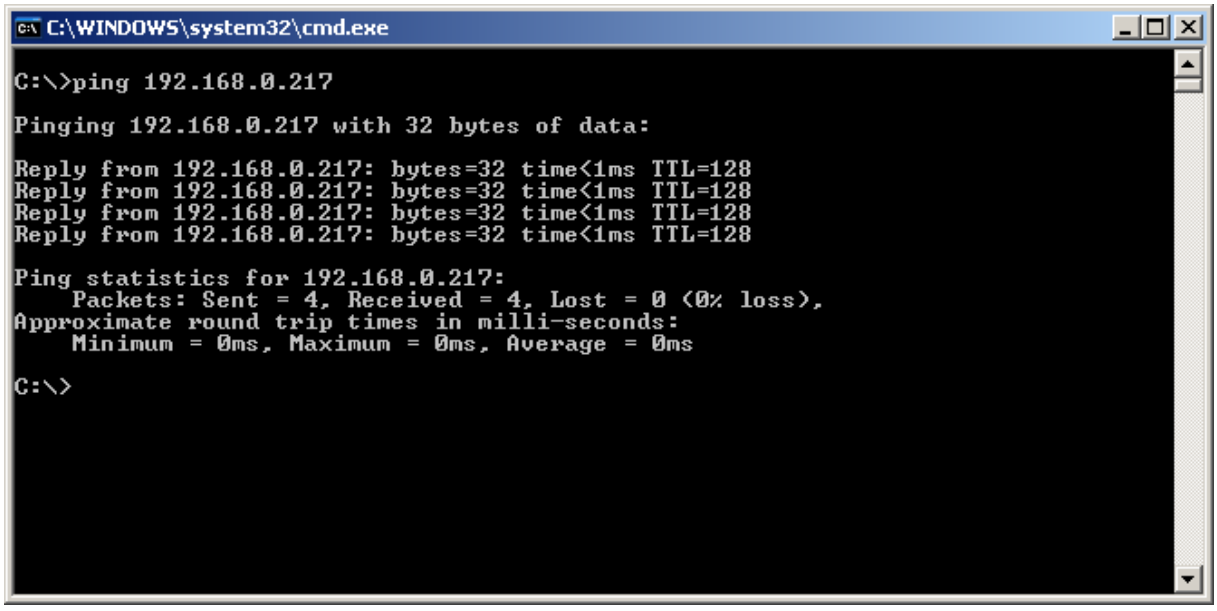
Note, that one needs two serial line connections between the PC and the evaluation board - one for the PPP communication (ittyb) and the other for the serial shell (ttya/default). In case the PC has just one RS232 you can use USB-to-Serial adapter.
If the application is started on the embedded side run the PPP connection on the PC side.



If the PPP connection was established successfully the following data is displayed on the serial console.



At this point it is possible to verify the PPP communication by pinging from both PC and embedded side, see the previous and the next picture.



PPP Client

Step One - Rebuilding

Rebuilding MQX is the first step which needs to be done. For rebuilding the MQX RTOS, refer to *Freescale MQX Real-Time Operating System User's Guide*, Chapter 4: "Rebuilding MQX". The following MQX compile-time configuration options must be set in the user_config.h to ensure the correct functionality of the PPP example.

Option	Value
RTCSCFG_ENABLE_VIRTUAL_ROUTES	1
BSPCFG_ENABLE_ITTYB*	1*

* The application requires PPP device to be defined manually and being different from the default IO channel. ITTYB is suitable for most of Freescale evaluation boards with the MQX support, however one has to check if ITTYB is not associated with another functionality.

After inserting/modifying the compile-time configuration options stated above the MQX RTOS needs to be re-compiled as described in the *Freescale MQX Real-Time Operating System User's Guide.*

Once the MQX RTOS is re-built open the shell example project located on the following path: <install_dir>/src/rtcs/examples/shell/<IDE>/shell_<evb number>.mcp.

The config.h file of the shell example project contains the following PPP-related options which has to be changed/verified:

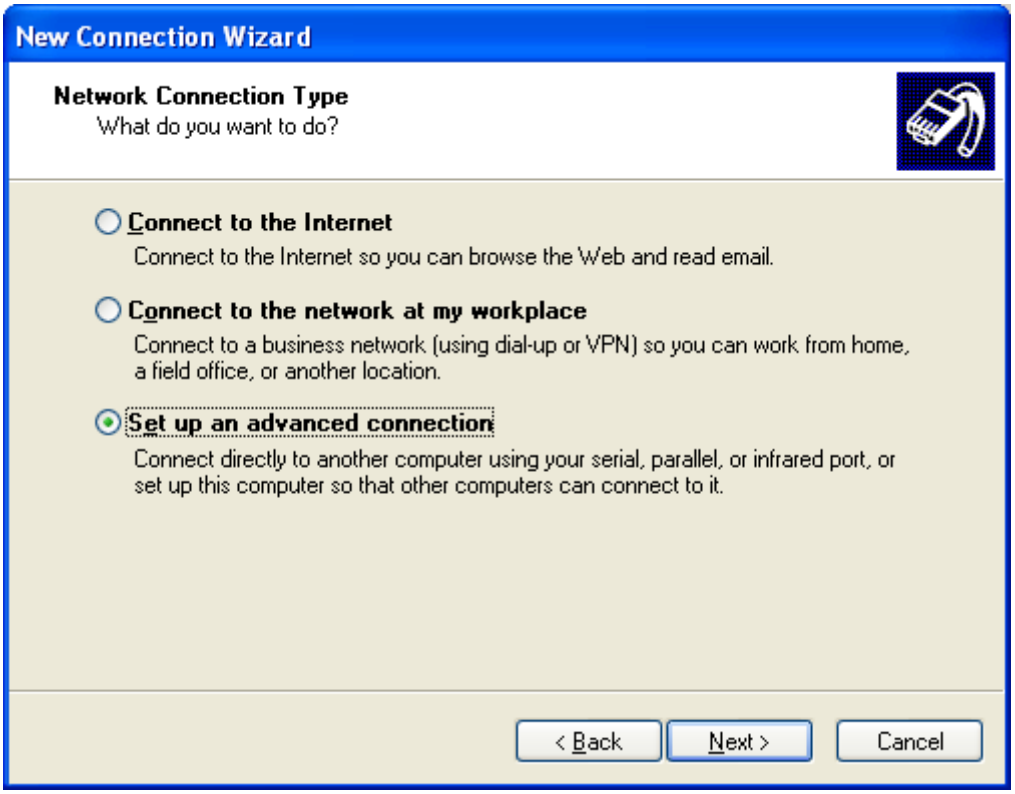
```
#define PPP_DEVICE_DUN          0
#define PPP_DEVICE_RAS          1*
#define DEMOCFG_ENABLE_PPP      1
#define PPP_DEVICE              "ittyb:"
#define PPP_LOCADDR             IPADDR(192,168,0,216)
#define PPP_PEERADDR            IPADDR(192,168,0,217)
#define GATE_ADDR               IPADDR(192,168,0,1)
```

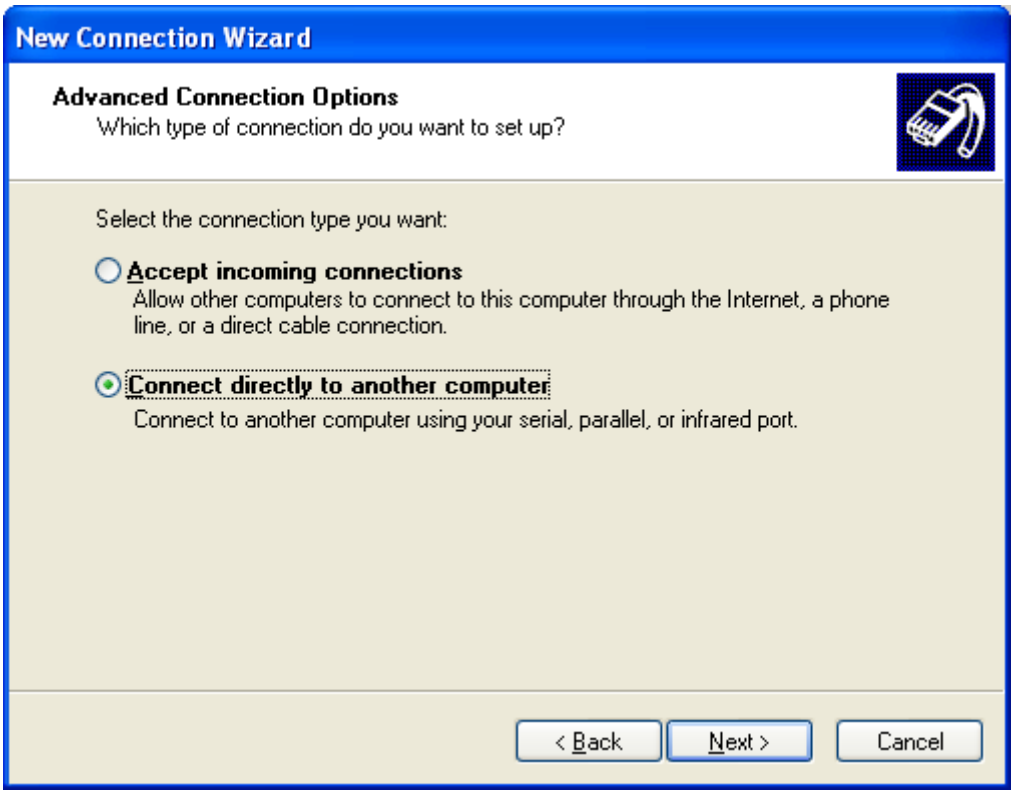
* In case of PPP client, please define PPP_DEVICE_RAS in "config.h" file.

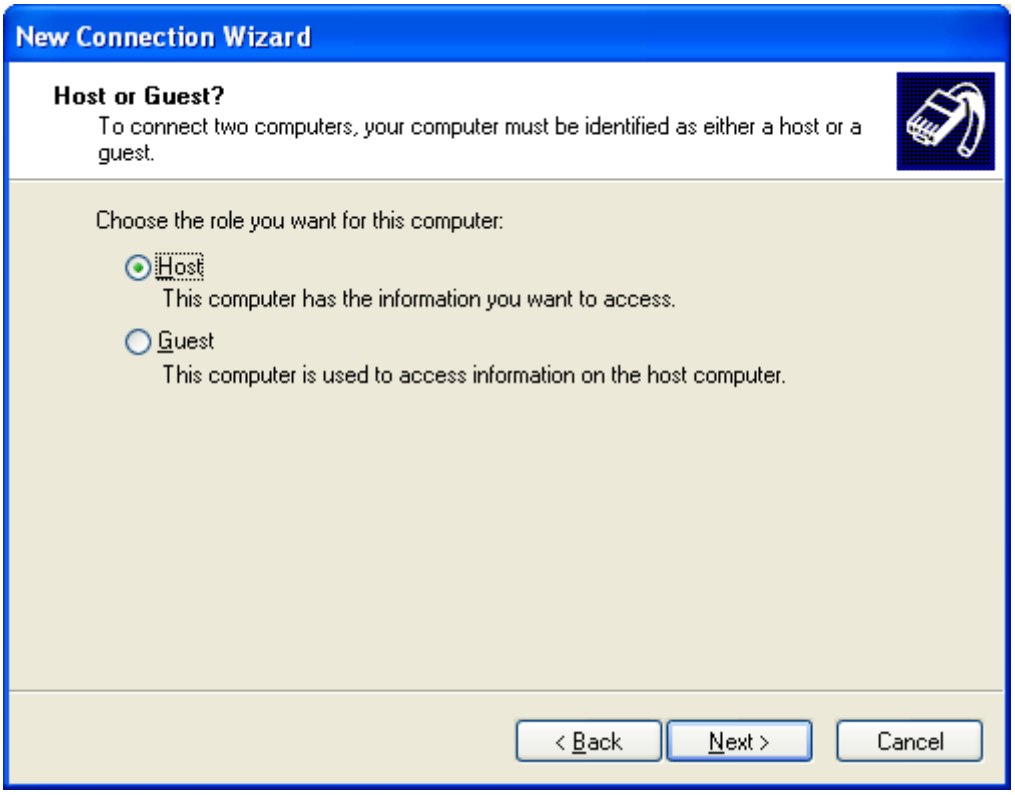
Once the config.h file is verified compile the shell example project and flash it to the evaluation board, see the *Freescale MQX Release Notes*.

<u>Step Two</u> – Establishing the PPP Connection on the PC Side

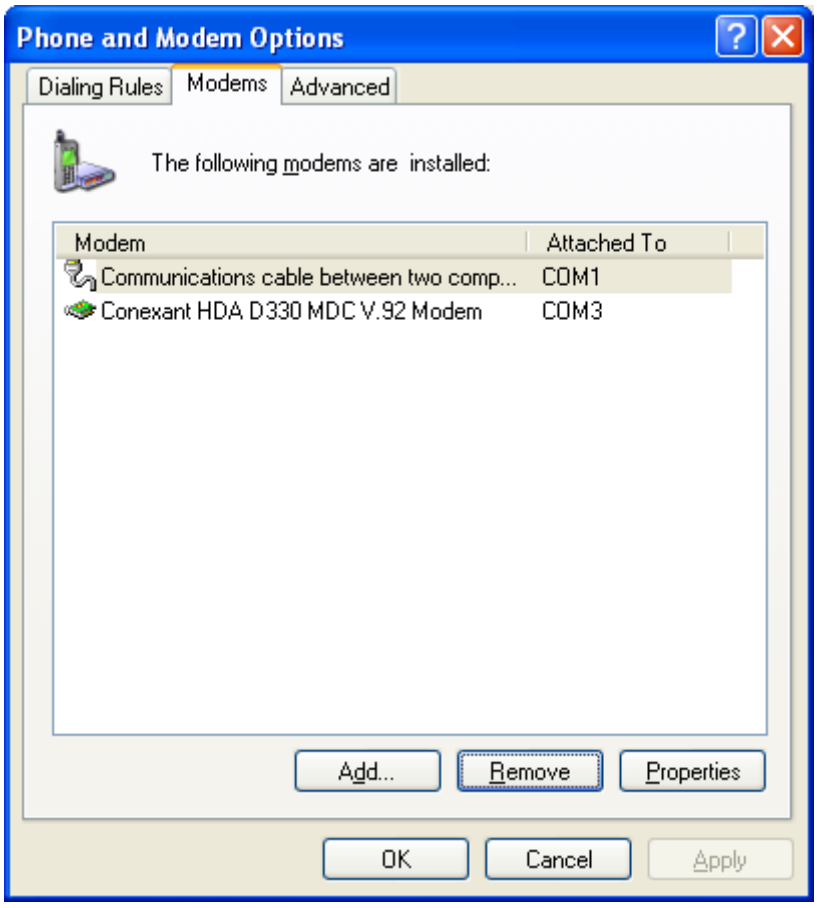
Open the MS Windows "Network Connections" dialog and start the "New Connection Wizard".



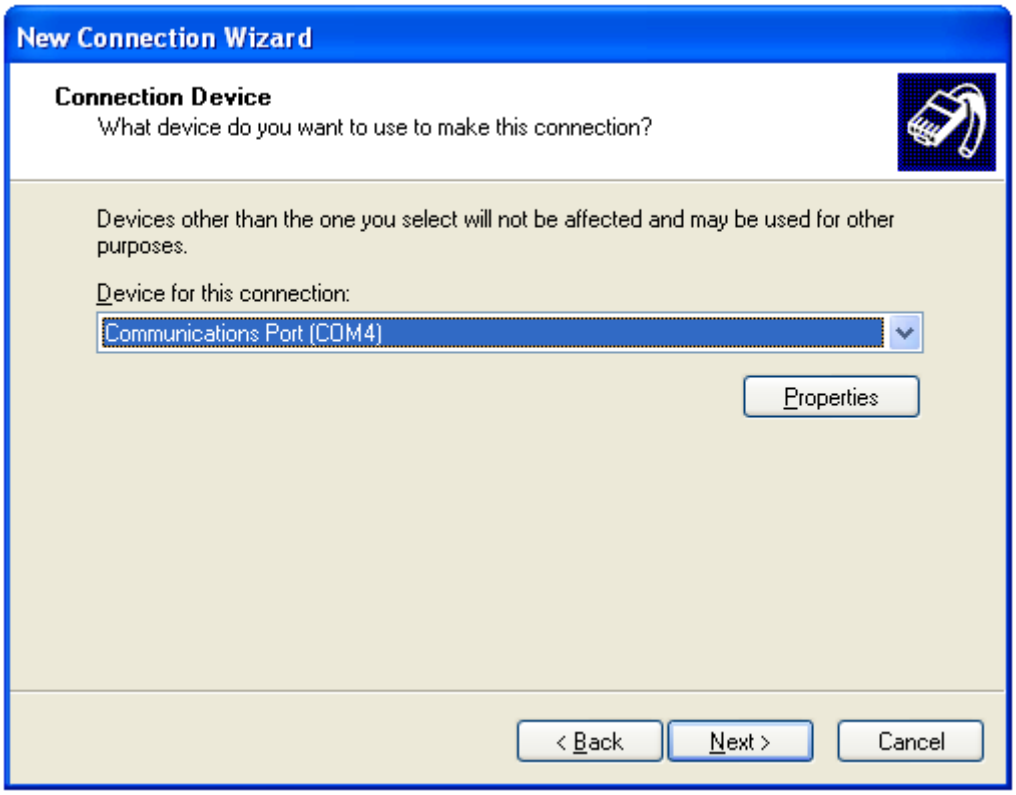

New Connection Wizard
Advanced Connection Options
Which type of connection do you want to set up?
Select the connection type you want:
Accept incoming connections
Allow other computers to connect to this computer through the Internet, a phone line, or a direct cable connection.
Connect directly to another computer
Connect to another computer using your serial, parallel, or infrared port.
< Back
Next >
Cancel


New Connection Wizard
Host or Guest?
To connect two computers, your computer must be identified as either a host or a guest.
Choose the role you want for this computer:
Host
This computer has the information you want to access.
Guest
This computer is used to access information on the host computer.
< Back
Next >
Cancel

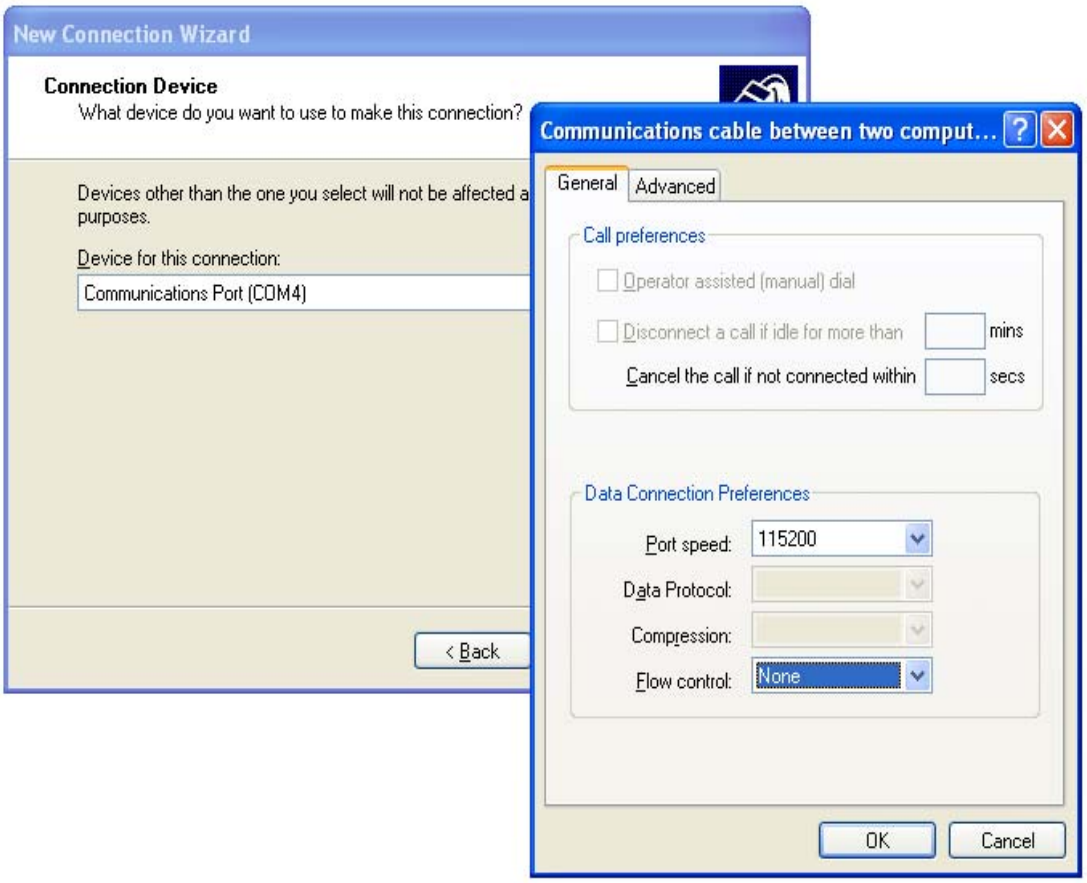
- If you face troubles with the “Incoming Connection Warning” when trying to establish the incoming connection, go into MS Windows Control Panel → Phone and Modem Options → Modems and remove all modems relating to communications cable between two computers. Then try again to create a connection using the “New Connection Wizard”.



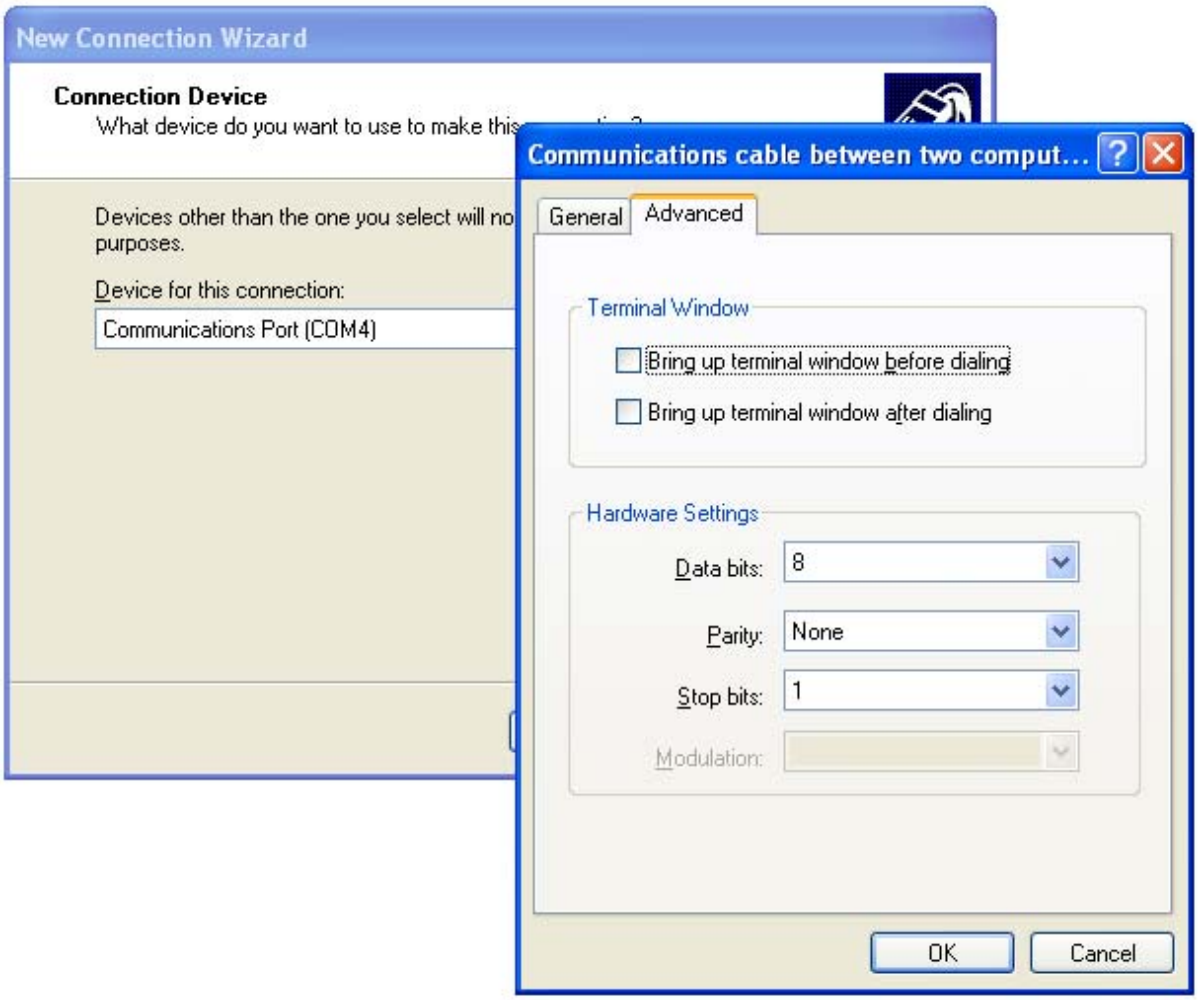
- Continue with Communication Port selection.



- Set PPP connection's properties by click "Properties" button, as depicted on the following pictures:
    - under General, set the baud rate to match the embedded board



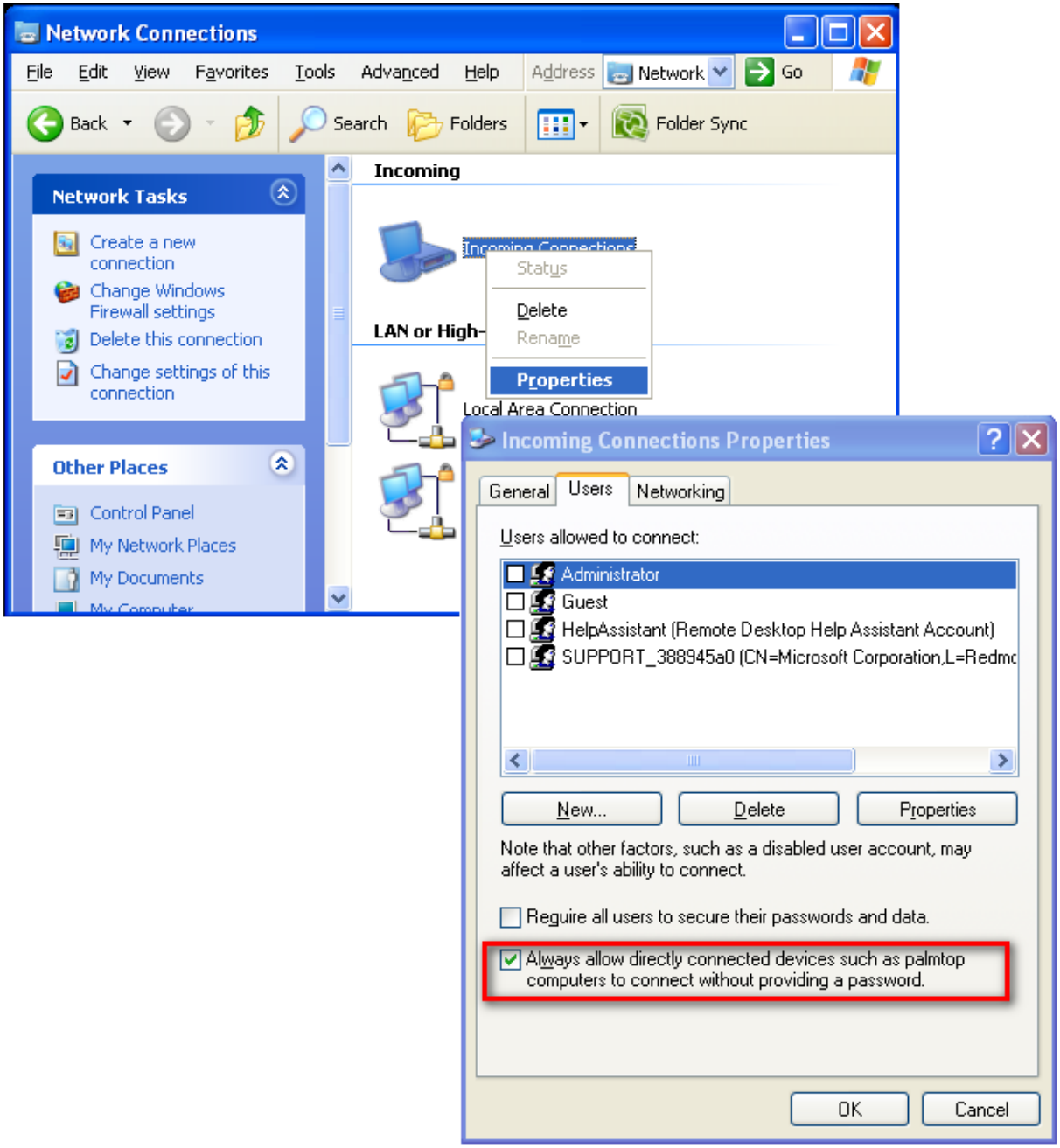
- under Advanced, set the Hardware Settings to match the embedded board

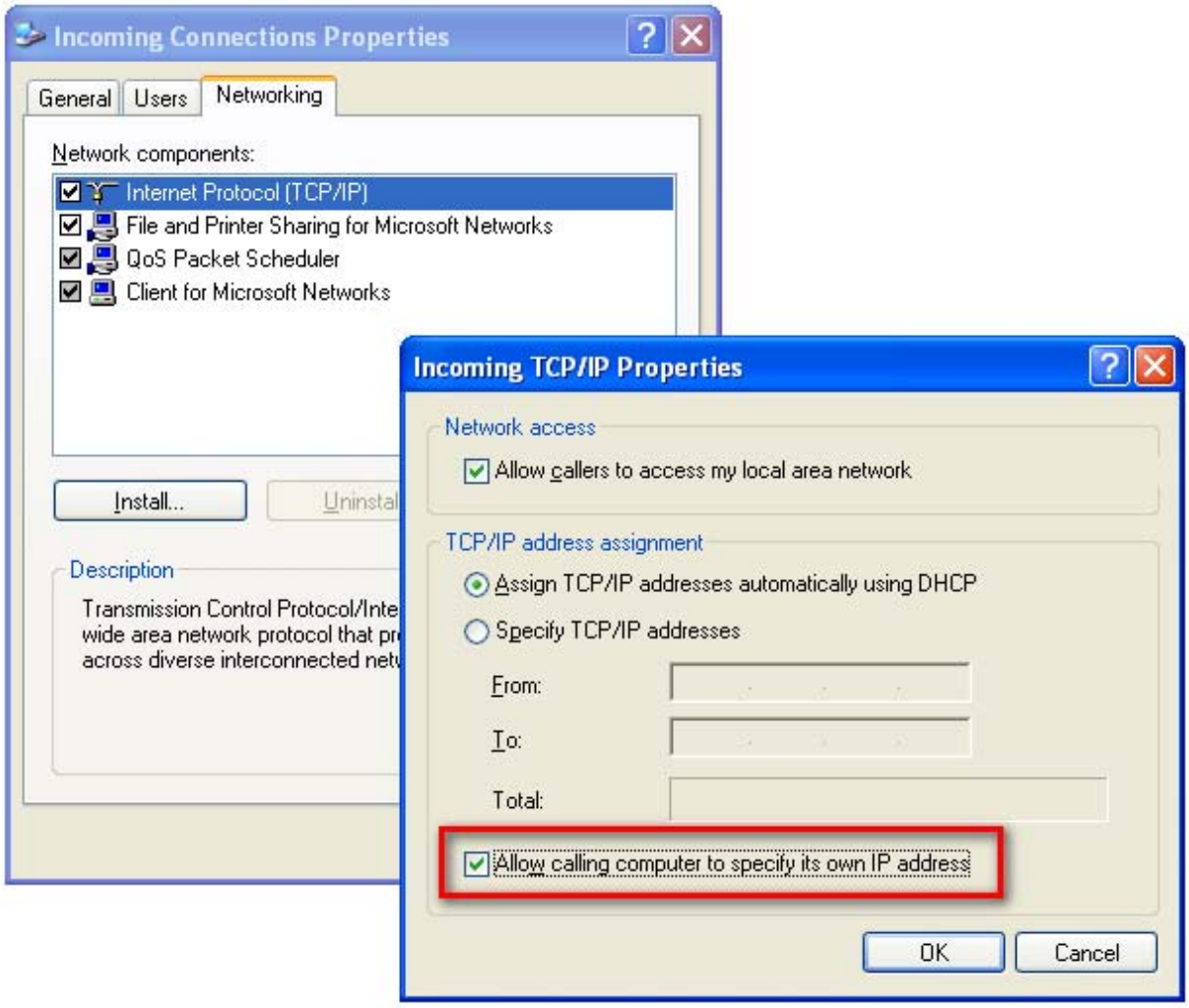


- Once PPP connection's properties are set click on "Next" button to display User Permissions dialogue. There is not necessary to make any changes here, click on "Next" button again.

- Complete the new connection by clicking on "Finish" button.

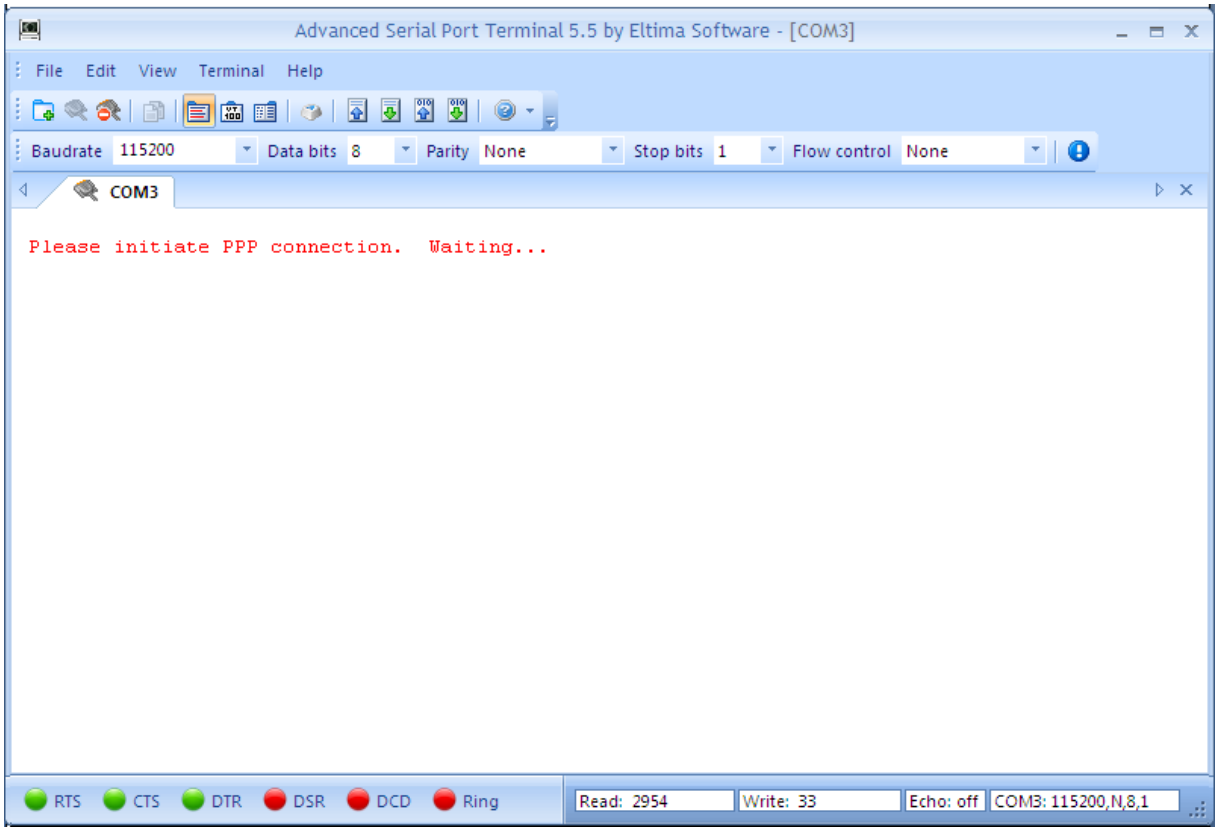
- Once the connection is created, configure it to allow client to be connect without providing a password and to specify its own IP address:




Incoming Connections Properties
General
Users
Networking
Network components:
Internet Protocol (TCP/IP)
File and Printer Sharing for Microsoft Networks
QoS Packet Scheduler
Client for Microsoft Networks
Install...
Description
Incoming TCP/IP Properties
Network access
Allow callers to access my local area network
TCP/IP address assignment
Assign TCP/IP addresses automatically using DHCP
Specify TCP/IP addresses
From:
To:
Total:
Allow calling computer to specify its own IP address
OK
Cancel

Step Three – Run the Example Application

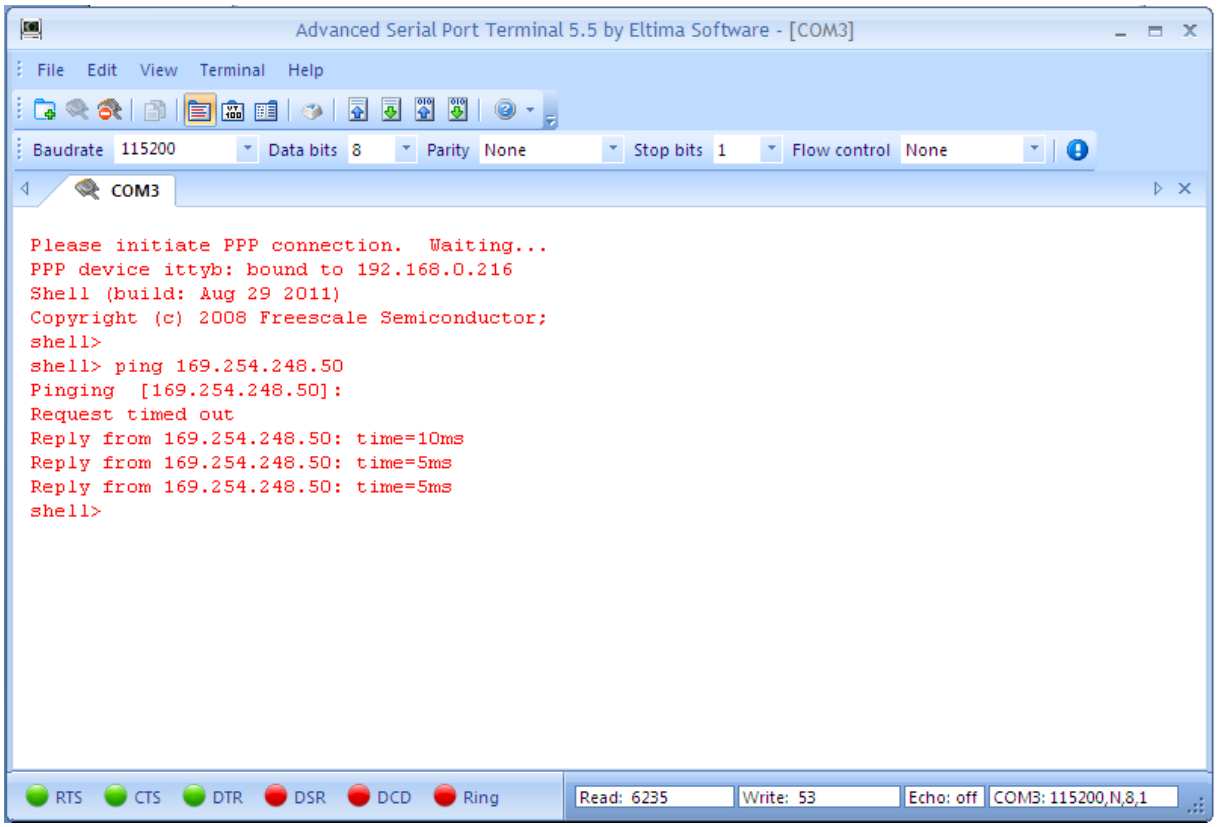
Once the example application is loaded into the Flash or MRAM memory you can start it. The following prompt occurs on the default serial console.



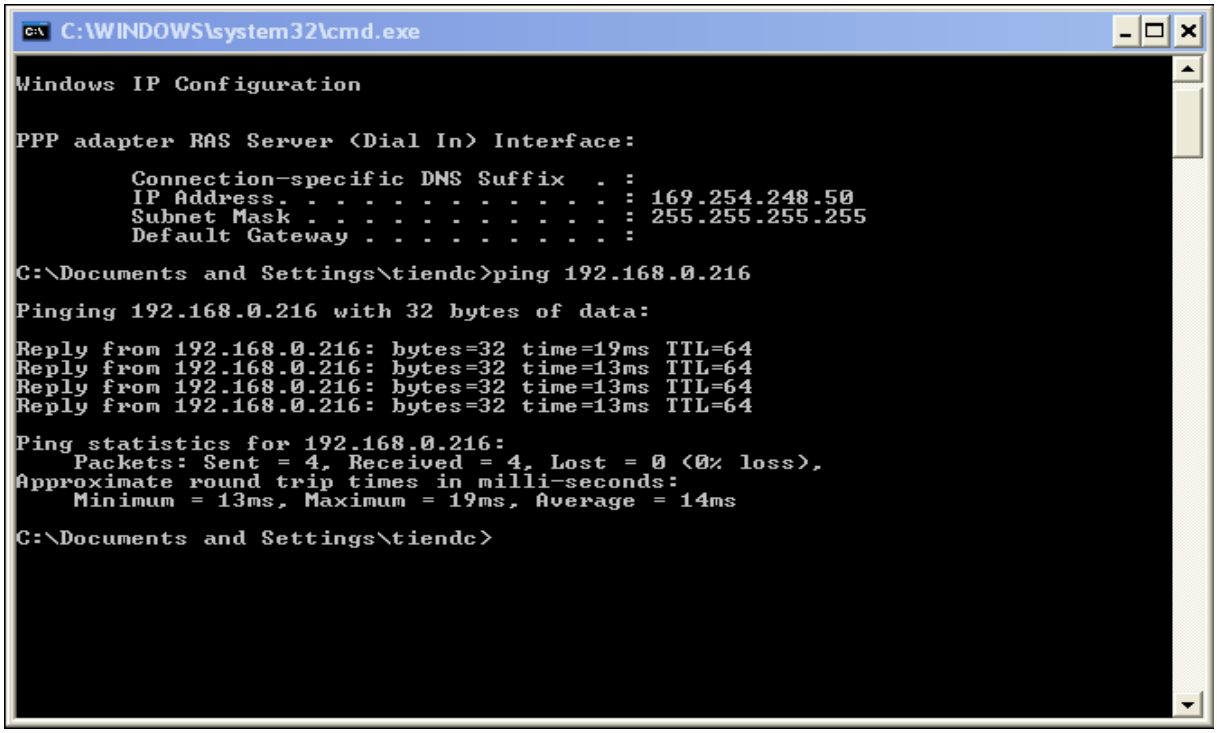
Note, that one needs two serial line connections between the PC and the evaluation board - one for the PPP communication (ittyb) and the other for the serial shell (ttya/default). In case the PC has just one RS232 you can use USB-to-Serial adapter.

Note, that the PPP connection is sometimes not successfully established when running the PPP client example from the MRAM memory due to the low speed of this memory. Please, use other available target(s) in the project to get the example running.

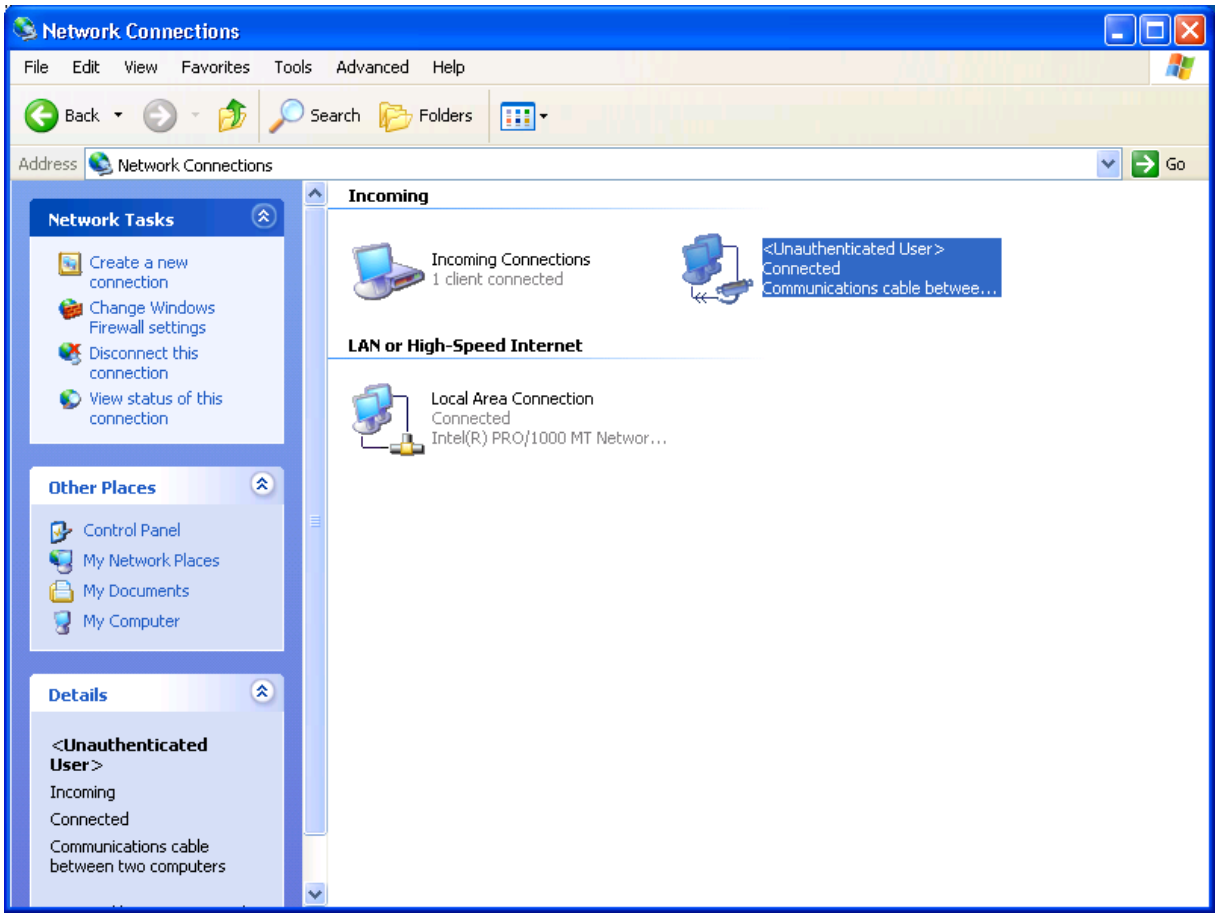
If the PPP connection was established successfully the following data is displayed on the serial console.



At this point it is possible to verify the PPP communication by pinging from both PC and embedded side, see the previous and the next picture.



Do not forget to manually disconnect the device (client) using the “Network Connections” dialog when terminating the example application on the embedded side.



ver. 1.0