Freescale MQX RTOS Example Guide eth_to_serial demo

This document explains the eth_to_serial example, what to expect from the example and a brief introduction to the API.

The demo

The eth_to_serial example code is a simple character passing between UART console and socket. The application creates a socket on port 23 (this port is default and can be changed with demo configuration) where a TCP/IP connection can be established. This example supports both IPv4 and IPv6 protocols.

Demo configuration

All configuration options for the example are stored in file *config.h* . All configurations are done using macros. These options are:

- `ENET_IPADDR` – Board IP address for IPv4 protocol (IPv6 address id selected by auto configuration internally in RTCS). Default is 192.168.1.202.

- `ENET_IPMASK` – IP mask for IPv4. Default value is 255.255.255.0.

- `ENET_IPGATEWAY` – Gateway for IPv4. Default is 0.0.0.0.

- `DEMO_PORT` – network port on which application will be available. Default is port number 23.

- `SERIAL_DEVICE` – serial port device used for communication thru Ethernet. Default is set to BSP_DEFAULT_IO_CHANNEL.

This demo also requires some special setting in file *user_config.h.* You have to enable interrupt driven serial port (i.e. "ittyf:") and select it as default I/O channel. This can be done by adding following (example) defines to user_config.h and recompiling MQX libraries:

```
#define BSPCFG_ENABLE_ITTYF             1

#define BSP_DEFAULT_IO_CHANNEL_DEFINED

#define BSP_DEFAULT_IO_CHANNEL          "ittyf:"
```

Please consult *FSL_MQX_getting_started.pdf* for proper device name for your Freescale Tower board.

Running the demo

Connect a serial cable from the TWR-SER board to the PC. Connect an Ethernet cable from the RJ45 (Ethernet) connector from the board to the RJ45 connector in the PC.

Start HyperTerminal (or similar serial terminal) on the PC (This Start menu-> Programs-> Accessories-> Communications). Create a connection to the serial port that is connected to the board (usually will be COM1). Set it for 115200 baud, no parity, 8 bits and click OK.
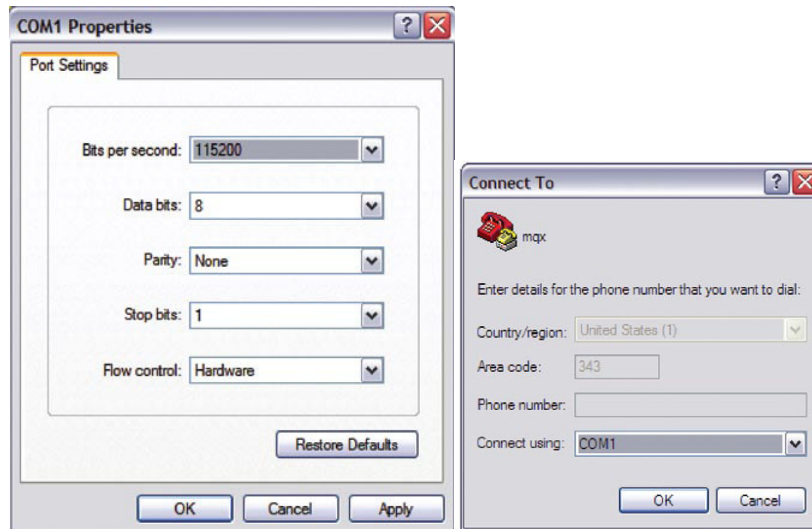
After the demo is loaded in board and run you should see following text (IPv6 address may vary):

```
Application listening on following ip addresses:
  IPv4 Address: 192.168.1.202, port:23
  IPv6 Address 0: fe80::200:5eff:fea8:1ca, port: 23

Waiting for incoming connection...
```

If you can see this message application is ready and waiting for incoming connection on port 23 on both IPv4 and IPv6 addresses. If one of addresses is missing either only IPv4 or IPv6 is enabled in RTCS and demo will use only this enabled protocol. Please note that only one connection to board can be opened at same time (you can connect to IPv4 address OR IPv6 address).

The default IP address of the board is 192.168.1.202. If you want to use IPv4 connection your PC must be on same subnet (192.168.1.x). You may configure the IP address of the computer manually. Select Start-> Settings-> Network Connections-> Local Area Connection. Note your original TCP/IP settings, and then set your IP address to 192.168.1.210 and your subnet mask to 255.255.255.0.

Now you can start a second HyperTerminal on the PC (Start menu->Programs->Accessories->Communications). Write TCP_test as connection name. In "Connecting using" parameter select "TCP/IP (Winsock)" option. Put in the IP address of the board to "Host address". In this case the 192.168.1.202 and click OK. If you are trying to connect to (link local) IPv6 address, scope ID must be added to IP in following format: `IPv6_address%scope_ID`. i.e. `fe80::200:5eff:fea8:1cb%22`.
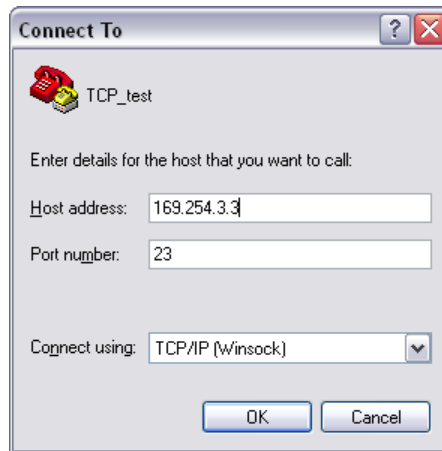
After successful connection text in serial terminal should look like this:

```
Application listening on following ip addresses:
   IPv4 Address: 192.168.1.202, port:23
   IPv6 Address 0: fe80::200:5eff:fea8:1ca, port: 23

Waiting for incoming connection...OK

Serial <-> Ethernet bridge
```

And in TCP terminal like this:

```
Ethernet <-> Serial bridge
```

The session is now established. Whatever is typed in one window will appear in the other window. Whenever you disconnect from TCP connection application is reset to wait mode and following message appears in serial terminal:

```
Client disconnected.
Waiting for incoming connection...
```

After this message you are free to reconnect again using either IPv4 or IPv6.