

eZDSP5502net

User's Guide

Netlib trial version

Document Rev. 1.4



Email : info@dsigt.de

Phone: +49 2833 5709 77

Web : www.dsigt.de

Table of Contents

1 Software Configuration.....2

 1.1 Installation.....2

 1.2 Preparation.....3

 1.3 The first Ethernet Program.....7

 1.4 Troubleshooting.....12

 1.5 Additional Tests.....15

2 Network Configuration.....17

 2.1 Network Hardware.....17

 2.2 TCP/IP Configuration.....18

 2.2.1 Manually Assigned IP Addresses.....19

 2.2.2 DHCP-based Network.....20

3 Examples.....21

 3.1 Blocksend.....21

 3.2 Multicast.....24

 3.3 SNTP – Simple Network Time Protocol.....27

 3.4 Webserver.....28

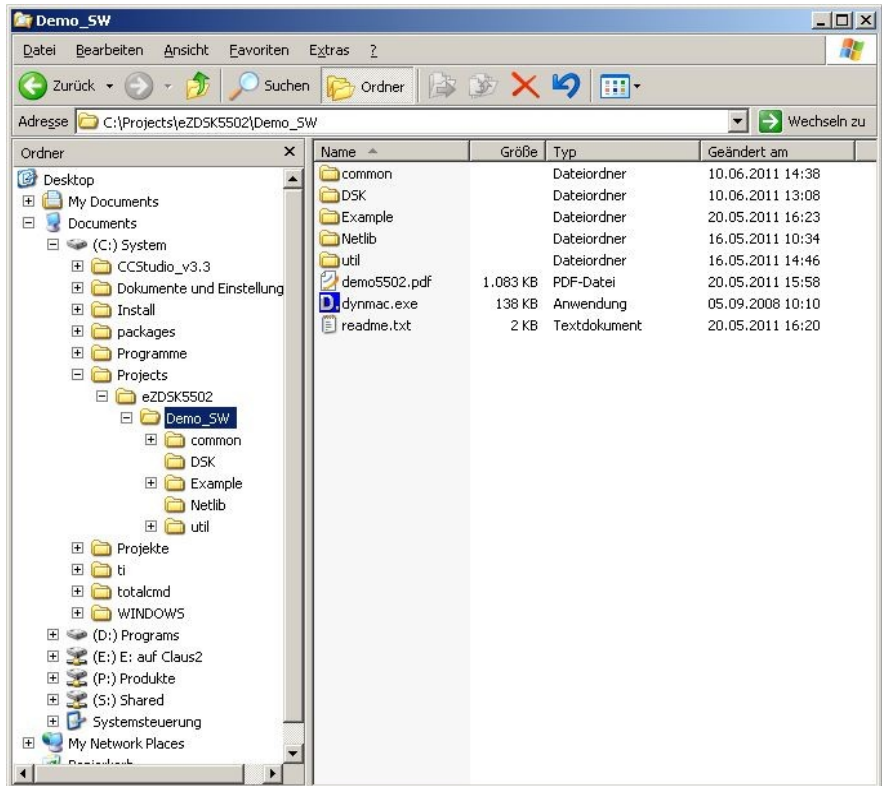
 3.5 IRadio – Internet Radio Stream.....30

 3.6 Telnet Server.....32

1 Software Configuration

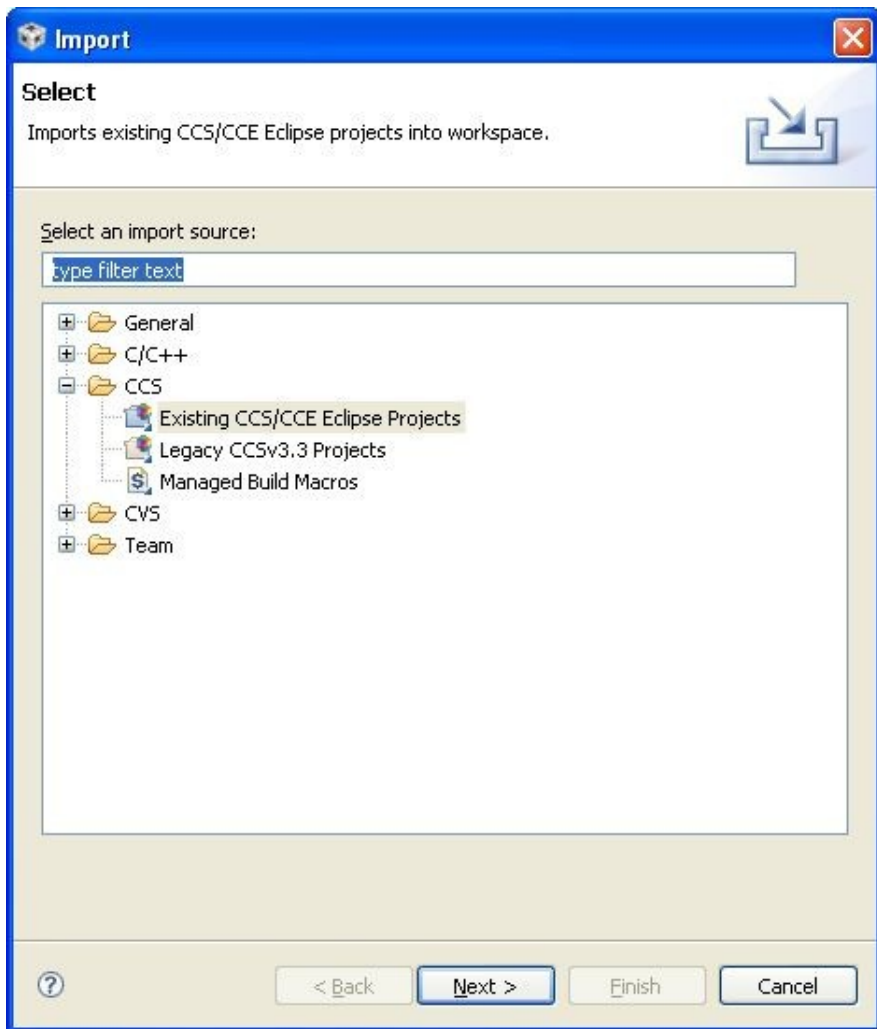
1.1 Installation

Copy the entire demo_enet path to your local harddisk. No further installation steps are necessary.



1.2 Preparation

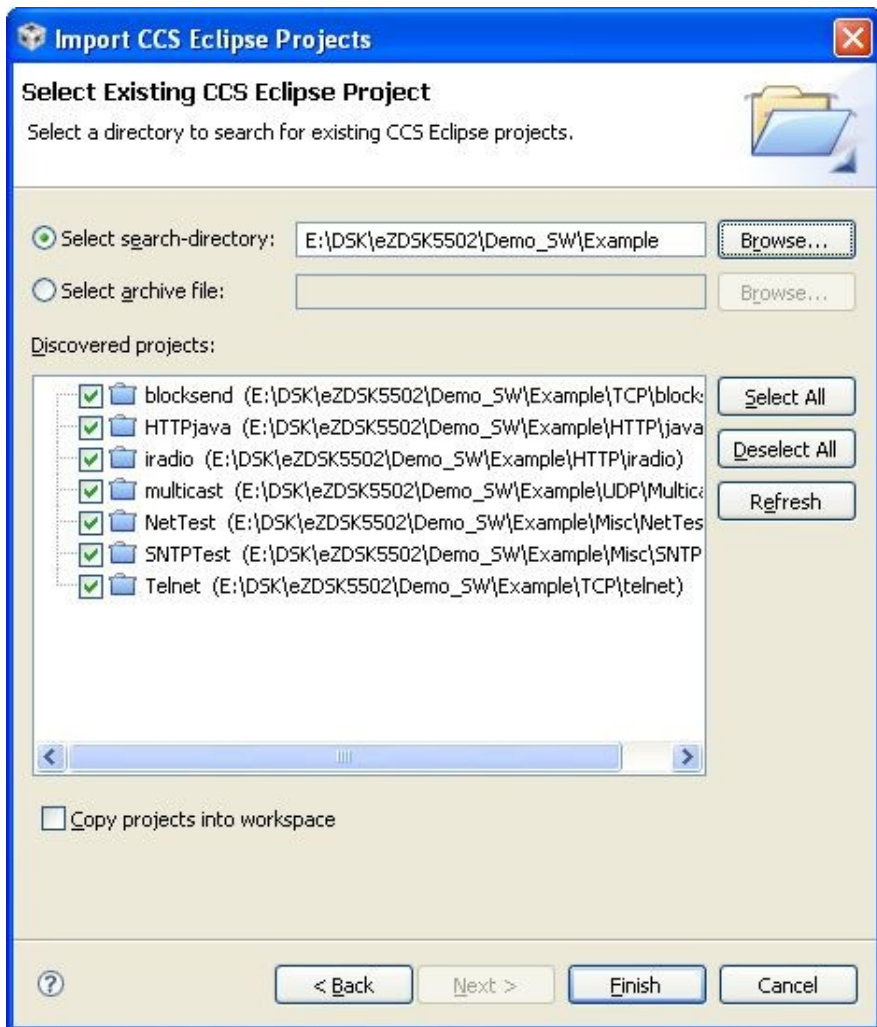
Import existing CCS/CCE Eclipse projects into workspace. Select File-Import from CCS menu.



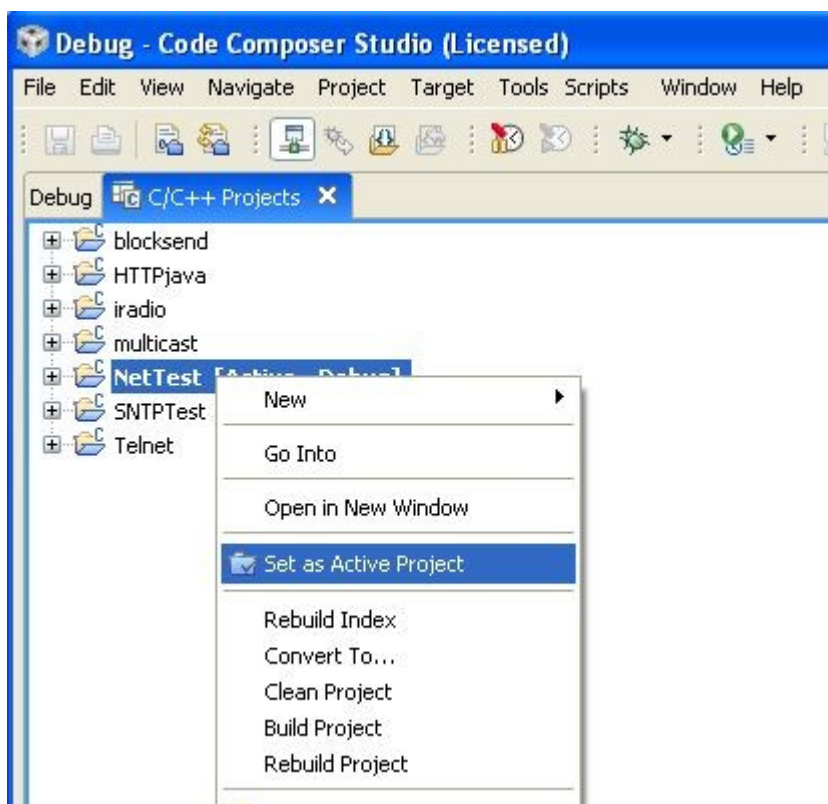
Choose the previously copied example directory:



Select all discovered projects:



Set nettest as active project



1.3 The first Ethernet Program

It is assumed that you are already familiar with the DSK/EVM, Code Generation Tools, and Code Composer Studio. The Code Composer Studio environment and the latest Code Generation Tools must be installed properly before you start with this instruction list.

1. connect the Ethernet jack on the eZDSP5502 to your network (hub or switch) using a standard 1:1 CAT-5 patch cable. You may also use a crossover cable to establish a single point-to-point connection to your PC without a network. Please note that in the latter case you will have to wait some time until the link is established. The sample programs configure the eZDSP5502 to autonegotiation mode. This will work with most networks with any kind of hub or switch, and with a point-to-point connection using a crossover cable.
2. connect the DSK / EVM USB emulator interface
3. start Code Composer Studio and load the ezdsp5502.gel File (directory \dsk). Connect to the target and load project nettest from the eZDSP5502 demo software directory \example\misc\nettest:
4. the example is configured for DHCP – dynamic assigned IP addresses. The DSP host name is mydemo. If your network does support DHCP go to step 6, if not open the source file netconfig.c, and change the following lines:

```
static char dsp_ip_addr[] = "mydemo";
```

change to

```
static char dsp_ip_addr[] = "192.168.168.200";
```

to match your network and TCP/IP configuration. Please ask your administrator for a free and unused IP address in your network.

If configured manually and if your network provides a DNS server to resolve symbolic names enter it's address (ask your administrator) in the next line

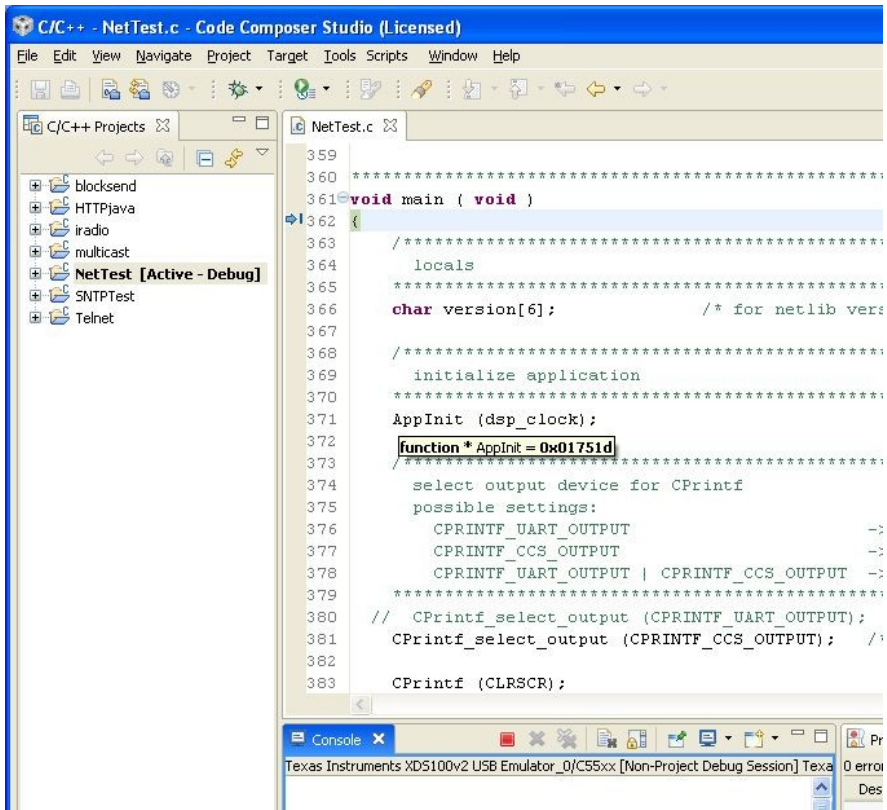
```
static char dns_server_ip[] = "192.168.168.1";
```

and activate the DNS

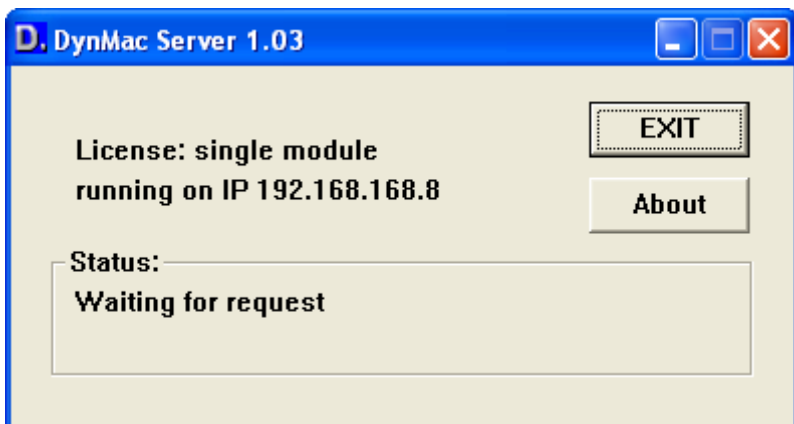
```
u_int32_t (*dns_setting)(void) = DNS_ENABLE;
```

Finally disable DHCP:

```
u_int32_t (*dhcp_setting)(int8_t *, u_int16_t) = NULL;
```

5. re-build the project and load program nettest.out to your target board.
6. before you run the DSP program make sure to start the DynMac.exe server on your PC. This server is required for the demonstration library and should run the whole time you are testing the network library. This issue is not required in the full netlib version.



7. run `nettest.out` on DSP. A starting message should be printed on CCS output window and some dots should appear during initialization:

```
Console X
Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas

-----
| D.SignT                                nettest demo |
|                                         |
| (c) 2011 D.SignT                        www.dsignt.de |
|                                         |
| netlib revision:  2.67                  |
|                                         |
-----

To verify this program, first 'ping' the DSP:
>ping mydemo

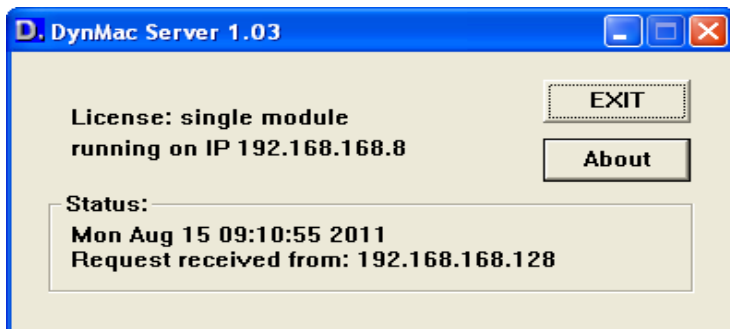
UDP test:|
>nc -u mydemo 7

TCP test
>nc mydemo 1061

initialize network...
```

When the board was initialized successfully, the LINK-LED on the eZDSP5502 should be lit.

8. During the next 5-10 seconds a new IP address is requested from your DHCP server. Afterwards the DSP MAC address and authorization key is assigned by the DynMac server.



The CodeComposer output window should now show the following printout:

```
Console x
Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas

-----
| D.SignT                                nettest demo |
|                                         ||
| (c) 2011 D.SignT                      www.dsignt.de |
|                                         |
| netlib revision:  2.67                  |
|                                         |
-----

To verify this program, first 'ping' the DSP:
>ping mydemo

UDP test:
>nc -u mydemo 7

TCP test
>nc mydemo 1061

initialize network... .....
Link status changed: Linked at 100Mbit full duplex
..... success

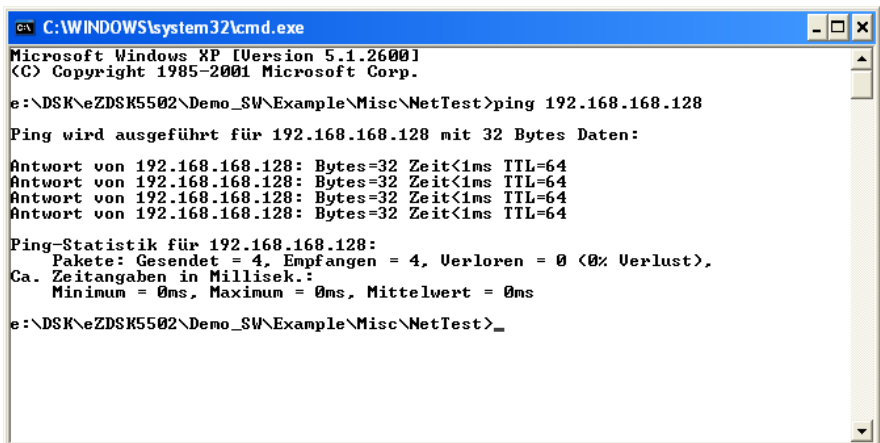
assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address:  192.168.168.128
Host name:   mydemo
Domain name: dsignt.net
DNS Server:  192.168.168.1
Gateway:     192.168.168.1
Subnetmask:  255.255.255.0
```

The assigned network configuration is printed out. In this case the DSK is connected to a 100Mbit full duplex link with assigned IP address 192.168.168.128. Gateway and DNS server IP is 192.168.168.1. The exact configuration in your environment depends on the DHCP setting of your network or on your manually assigned IP address.

9. ping the DSP from your PC by entering ping *DSP_IP_ADDR* at the command prompt, e.g. ping 192.168.168.128 or ping mydemo. Each time the PC sends a ping, the ACT_LED on the eZDSP5502 will flash, and the PC will display a message like

"Reply from 192.168.168.128: bytes=32 time=1ms TTL=64":



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

e:\DSK\eZDSP5502\Demo_SW\Example\Misc\NetTest>ping 192.168.168.128

Ping wird ausgeführt für 192.168.168.128 mit 32 Bytes Daten:

Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64

Ping-Statistik für 192.168.168.128:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0 (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 0ms, Mittelwert = 0ms

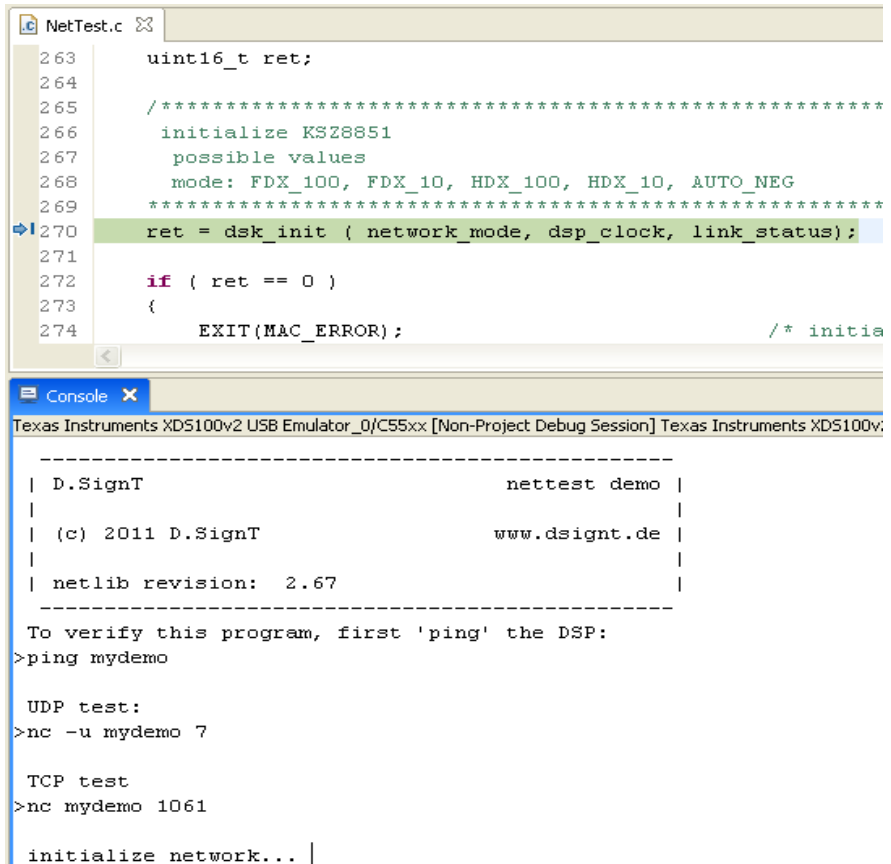
e:\DSK\eZDSP5502\Demo_SW\Example\Misc\NetTest>_
```

1.4 Troubleshooting

If the above test fails, please perform the following tests:

1. Reload nettest.out, execute "go main", and place the cursor on line

```
ret = dsk_init ( network_mode, dsp_clock, link_status);
in function InitializeNetwork().
```



```

NetTest.c
263     uint16_t ret;
264
265     /*****
266     initialize KSZ8851
267     possible values
268     mode: FDX_100, FDX_10, HDX_100, HDX_10, AUTO_NEG
269     *****/
270     ret = dsk_init ( network_mode, dsp_clock, link_status);
271
272     if ( ret == 0 )
273     {
274         EXIT(MAC_ERROR);           /* initia

Console
Texas Instruments XD5100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas Instruments XD5100v2

-----
| D.SignT                               nettest demo |
| (c) 2011 D.SignT                      www.dsignt.de |
| netlib revision:  2.67                 |
-----

To verify this program, first 'ping' the DSP:
>ping mydemo

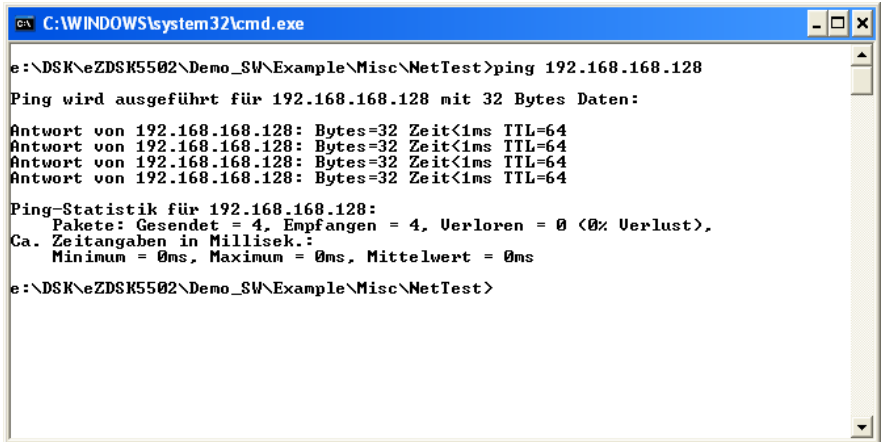
UDP test:
>nc -u mydemo 7

TCP test
>nc mydemo 1061

initialize network... |
```

2. Make sure the DynMac server is running within the same subnet
3. execute "run to line" and "step over"

4. check the return value `ret` (quick watch). It must be different from 0. If a 0 is returned, the EMIF settings may be incorrect or the daughterboard connection may be faulty. Please make sure that the appropriate project for your target board was used and the DSP board is plugged correctly. Restart the program and start again with issue 1.
5. The LINK-LED should be lit now. If not, check the network cabling and configuration. You may have connected the DSK to a hub's crossover port. If using a crossover cable and point-to-point connection to a PC it may take some time until the link is established. This time varies with different network controllers, but a link should be established at least after one or two minute.
6. If the link is established (LINK-LED is lit continuously), any traffic in your local network segment will flash the activity LED (ACT-LED). Now step-over further through the program to `net_init()`
7. while you step over `net_init()` the internal protocol stack and socket structure is initialized. Furthermore if configured for DHCP, a new IP address is requested from DHCP server. When the IP address was assigned, the DSP contacts the DynMac server running on a certain PC in your network. The DynMac server verifies the testing license and sends back the authorization key for further operation.
8. now `install_icmp_socket()` and the following `socket_open()` calls should all be successful. Run the program by pressing F8 or the "run" command.
9. Should a ping be unsuccessful though, please check the TCP/IP configuration. The DSP must be in the same subnet as the PC from which you ping the DSP. Generally this means that the IP addresses of the PC and the DSP must differ only in the last address byte. If unsure please contact your network administrator. To get your PC's IP address, run program `winipcfg.exe.` / `ipconfig.exe.` For additional information, please read chapter 2.2 TCP/IP Configuration.
10. "mydemo" is the fully qualified hostname of the DSK. You may change this name to any valid name in your network (ask your network administrator). To ping the DSP from the PC, enter "ping mydemo" at the command prompt.



```
C:\WINDOWS\system32\cmd.exe

e:\DSK\ezDSP5502\Demo_SW\Example\Misc\NetTest>ping 192.168.168.128

Ping wird ausgeführt für 192.168.168.128 mit 32 Bytes Daten:

Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.168.128: Bytes=32 Zeit<1ms TTL=64

Ping-Statistik für 192.168.168.128:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0 (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 0ms, Mittelwert = 0ms

e:\DSK\ezDSP5502\Demo_SW\Example\Misc\NetTest>
```

Some PC programs cause corrupt ISP or network settings. If you observe permanently problems with certain protocols or ports please contact your system administrator and ask for the following netsh commands:

```
C:>netsh winsock reset catalog
```

```
C:>netsh int ip reset all
```

Handle these commands with care, your winsock settings or your interface settings respectively will be resetted.

In addition sometimes the PC DNS cache is not up-to-date. Use command

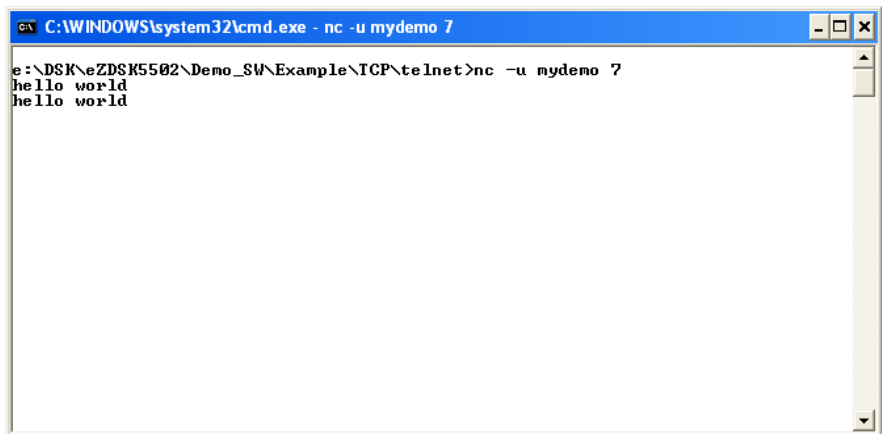
```
C:>ipconfig /flushdns
```

to flush the PC DNS cache.

1.5 Additional Tests

If pinging the DSK was successful you should proceed verifying the UDP and TCP connections using the PC program NetCat (nc.exe). NetCat is included in the DSK support software, directory \util\netcat, and is a very useful tool to send and receive data via the network using the UDP and TCP protocol. Type `nc -h` for a list of available options.

The DSK test program `nettest.out` implements a UDP echo server on port 7. Any data received on this port will be echoed back to the sender. To test this function, start NetCat by entering `nc -u mydemo 7` at the command prompt. Replace `mydemo` with your chosen DSP IP address or hostname. Type in some characters and finish input with the return (enter) button. NetCat will now transmit the data to the DSK, port 7, using the UDP protocol. Program `nettest.out` will echo the received characters back to the PC, and NetCat will display them in the next line. If everything is correct, you will see two identical lines of characters in the console window.

A screenshot of a Windows command prompt window. The title bar reads "C:\WINDOWS\system32\cmd.exe - nc -u mydemo 7". The command prompt shows the following text:

```
e:\DSK\ezDSP5502\Demo_SW\Example\TCP\telnet>nc -u mydemo 7
hello world
hello world
```

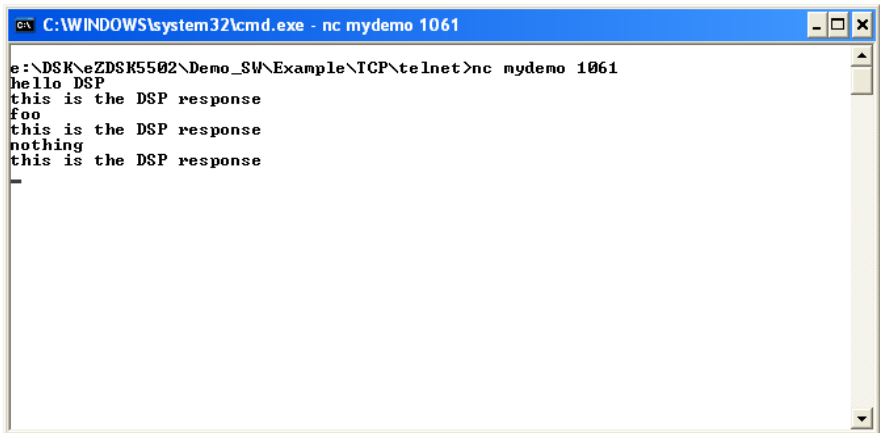
To exit NetCat type <CTRL> + C.

The maximum amount of data which can be send depends on the UDP echo buffer size specified in program `nettest.out`. Default is 1024 bytes.

To test the TCP functionality program `nettest.out` listens on port 1061. If any data is received on this port, the program will send a response to the client. Enter

```
nc mydemo 1061
```

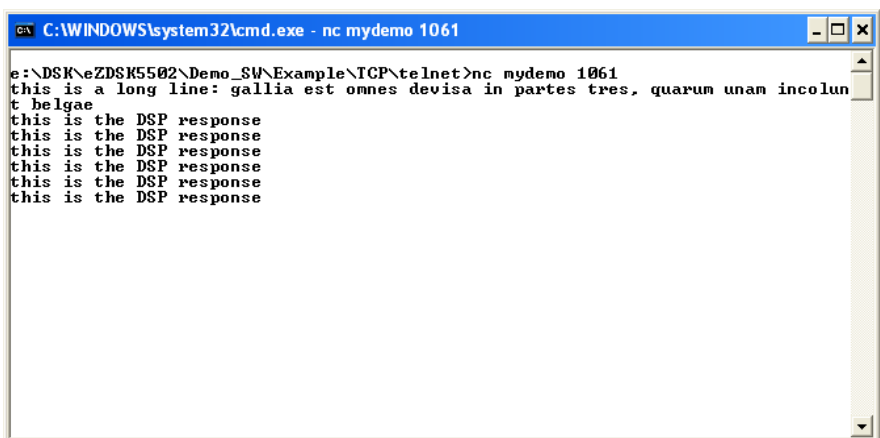

at the command prompt. Each time you hit the return (enter) button, the message "this is the DSP response" should appear in the console.



```
C:\WINDOWS\system32\cmd.exe - nc mydemo 1061

e:\DSK\ezDSP5502\Demo_SW\Example\TCP\telnet>nc mydemo 1061
hello DSP
this is the DSP response
foo
this is the DSP response
nothing
this is the DSP response
```

The buffer to receive TCP messages is set to 16 characters. This size is used as the window size published during TCP communications. If you send a message longer than 16 characters (including the return), the TCP/IP stack on the PC will partition the data stream into multiple packets. Hence, if you send a message larger than the buffer size, the DSK will receive two or more TCP messages and send two or more responses.



```
C:\WINDOWS\system32\cmd.exe - nc mydemo 1061

e:\DSK\ezDSP5502\Demo_SW\Example\TCP\telnet>nc mydemo 1061
this is a long line: gallia est omnes devisa in partes tres, quarum unam incolun
t belgae
this is the DSP response
this is the DSP response
this is the DSP response
this is the DSP response
this is the DSP response
this is the DSP response
```

2 Network Configuration

To successfully establish a network connection both the hardware and TCP/IP software configuration must match the network topology. This chapter describes typical configurations and the related initialization parameters for the eZDSP5502.

2.1 Network Hardware

Ethernet networks operate in one of the following modes:

1. 10 Mb/s/sec half duplex
2. 10 Mb/s/sec full duplex
3. 100 Mb/s/sec half duplex
4. 100 Mb/s/sec full duplex
5. Auto-Negotiation (the network hardware itself determines one of the four modes above)

The Ethernet Controller on the eZDSP5502 must be instructed to use the correct mode. The mode is determined by the parameter of function `dsk_init()`. If you are unsure about the correct mode, ask your network administrator. Generally, most low-cost single-speed hubs (sometimes also called repeater) support 10 Mbit or 100 Mbit half-duplex mode only. If parts of your network still use coaxial cable, you must limit the mode to half-duplex, even if your local hub supports full-duplex mode. The most convenient setting, auto-negotiation, is not supported by all hubs, please refer to your hub's documentation. Also some hubs do default to 100 Mbit half-duplex if auto-negotiation is not used, even if `dsk_init()` was called with full-duplex mode. If this happens, you will observe data loss: the DSK is not aware of collisions because collisions are not possible in full-duplex mode. The hub however operating in half-duplex mode may recognize collisions, but the DSK will not issue a retransmit.

If using auto-negotiation, also consider that this process may take up to 3 seconds. Any data transmitted during the auto-negotiation phase will be lost.

Network Mode	DSK Initialization
10Mbps/sec half-duplex	HDX_10
10Mbps/sec full-duplex	FDX_10
100Mbps/sec half-duplex	HDX_100
100Mbps/sec full-duplex	FDX_100
Auto-negotiation	AUTO_NEG

Table 2-1: eZDSP5502 Network Hardware Initialization

2.2 TCP/IP Configuration

A TCP/IP network uses individual IP addresses for each network member. These IP addresses may be assigned manually or automatically.

Assigning hostnames to the network members makes the TCP/IP network more convenient for users, since they do not have to remember the numerical IP addresses. A local hosts file or a DNS server provide hostname resolution.

If destinations outside the local network segment should be reached, a gateway is used to route the packets to and from other network segments. The subnet mask determines which IP addresses belong to the local network segment, and which addresses are outside.

Finally, a DHCP server greatly simplifies network maintenance: A DHCP server administrates a local network segment and assigns IP addresses to the network clients. Also provided are gateway, subnet mask, and DNS server settings. (A DHCP based network always uses symbolic hostnames).

Since local network traffic should be shielded from the outside world, some IP address ranges are reserved for local networking and are not routed to the outside, e.g. the Internet. The IANA (Internet Assigned Numbers Authority) has reserved three IP address ranges for use in local (private) internets (RFC 1918):

10.xxx.xxx.xxx, 172.16.xxx.xxx, 192.168.xxx.xxx. Addresses in this range can be used without coordination with the IANA and are the most likely IP address ranges used in any company network.

To further partition a network, e.g. into multiple divisions which are interconnected via routers and gateways, a subnet mask is used. The subnet mask for a class C network (max. 255 members) is 255.255.255.0, but smaller subnets may be used too. The TCP/IP software logically ANDs an IP address with the subnet mask to determine if the destination is within the local network subnet, or if the traffic must be routed via a gateway. A DHCP-administrated network will automatically provide the required settings. If your IP network uses manually assigned addresses, ask your network administrator for a valid IP address and subnet mask.

On the eZDSP5502 two functions are used to configure the TCP/IP network: `net_init()` and `net_set_gateway()`. Function parameters must be set according to your network configuration. The following chapters describe the possible configurations.

2.2.1 Manually Assigned IP Addresses

This is the basic configuration, all entries are made manually in `netconfig.c`. Optionally a DNS server may be present, which resolves symbolic host names. The following settings must be provided:

1. IP address of the DSK

```
static int8_t dsp_ip_addr[] = "192.168.168.200";
```

Replace 192.168.168.200 with the IP address assigned by your network administrator.

2. subnet mask and gateway if connections to machines outside the local subnet are required

3. DNS server IP address if present

```
static int8_t dns_server_ip[] = "192.168.168.1";  
u_int32_t (*dns_setting)(void) = DNS_ENABLE;
```

If DNS is not used you must initialize the DSK in `netconfig.c` with

```
static int8_t dns_server_ip[] = NULL;  
u_int32_t (*dns_setting)(void) = NULL;
```

In this configuration each network client must maintain it's own `netconfig.c` file if symbolic host names instead of numerical IP addresses should be used.

If connections to network members outside the local subnet are required, you must additionally specify the subnet mask and the gateway:

```
net_set_gateway ("192.168.168.1", "255.255.255.0");  
in sub-routine InitializeNetwork().
```

Replace 192.168.168.1 with the gateway IP address and 255.255.255.0 with the local subnet mask specified by your network administrator.

2.2.2 DHCP-based Network

A DHCP server will automatically supply all the required information. DHCP requires the use of symbolic hostnames, since the IP address of a machine may change if restarted after the DHCP lease time is expired. Hence, a DHCP-based network always requires a DNS server for name resolution. The DNS server IP address however is provided by DHCP and must not be specified manually. The `net_init` call using DHCP is:

```
net_init ("mydemo", DHCP_ENABLE, NULL, DNS_ENABLE);
```

The `netconfig.c` settings for DHCP are:

```
static int8_t dsp_ip_addr[] = "mydemo";  
static int8_t dns_server_ip[] = NULL;  
u_int32_t (*dns_setting)(void) = DNS_ENABLE;  
u_int32_t (*dhcp_setting)(int8_t *, u_int16_t) = DHCP_ENABLE;
```

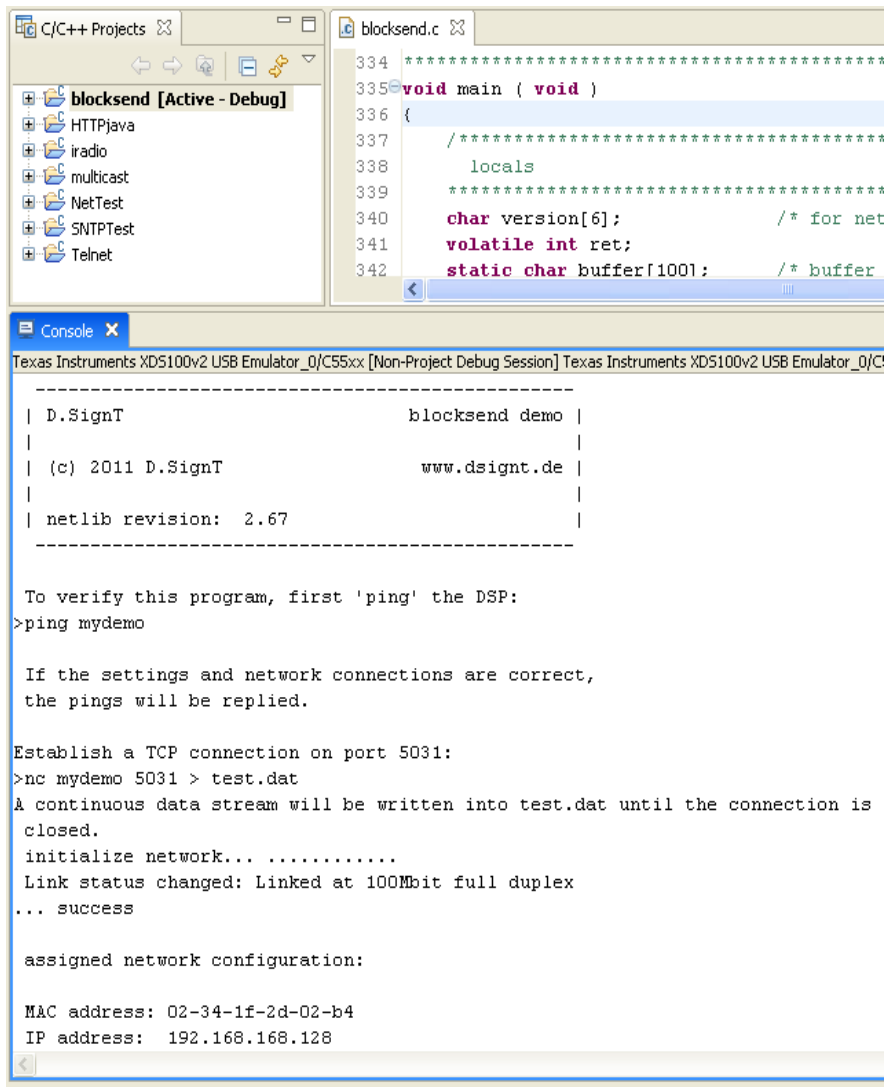
Replace `mydemo` with the desired host name for the DSK. You may specify the "fully qualified hostname", including the domain name. If you are not sure about the domain name, ask your network administrator or execute the Windows PC program `winipcfg.exe` or `ipconfig.exe`. Host Information will display the fully qualified hostname of your PC.

Since DHCP also provides the local subnet mask and the default gateway, calling `net_set_gateway` to set these parameters is obsolete and not required in this configuration.

3 Examples

3.1 Blocksend

This example is used to send a continuous data stream over TCP. Load and run the blocksend example in your CCS:



The screenshot shows the CCS IDE with the 'blocksend.c' file open. The file contains a C program that sets up a TCP connection and sends a continuous data stream. The console output shows the program's execution, including network initialization and the start of the data stream.

```
334 *****
335 void main ( void )
336 {
337     /*****
338     locals
339     *****/
340     char version[6];          /* for net
341     volatile int ret;
342     static char buffer[100]; /* buffer
```

```
-----
| D.SignT                      blocksend demo |
|                               |
| (c) 2011 D.SignT             www.dsignt.de  |
|                               |
| netlib revision: 2.67        |
|                               |
-----

To verify this program, first 'ping' the DSP:
>ping mydemo

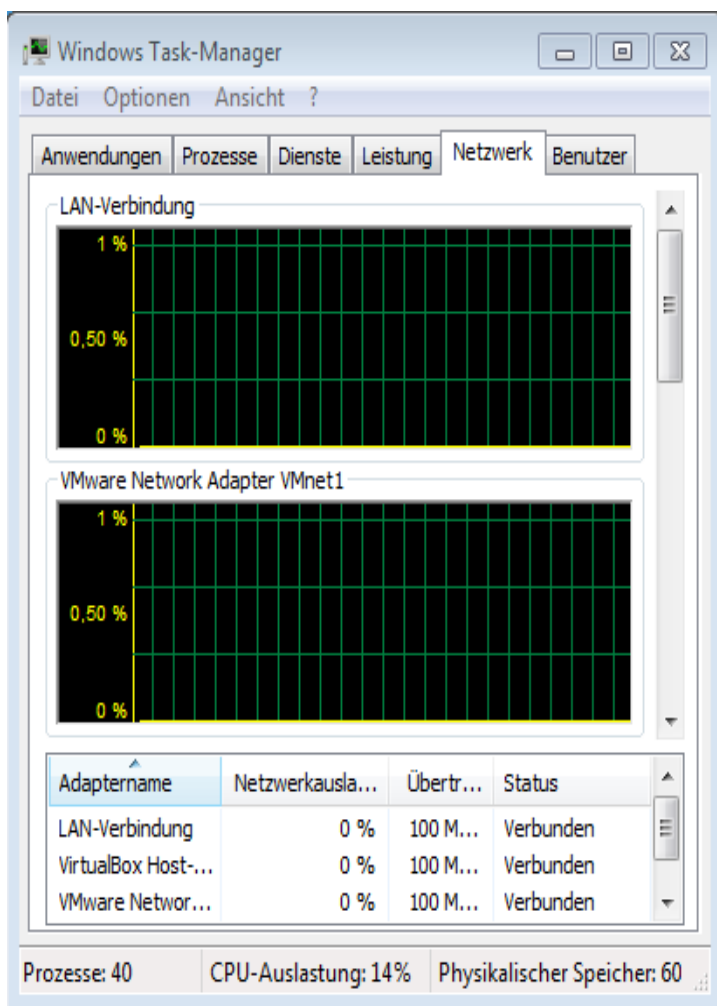
If the settings and network connections are correct,
the pings will be replied.

Establish a TCP connection on port 5031:
>nc mydemo 5031 > test.dat
A continuous data stream will be written into test.dat until the connection is
closed.
initialize network... .....
Link status changed: Linked at 100Mbit full duplex
... success

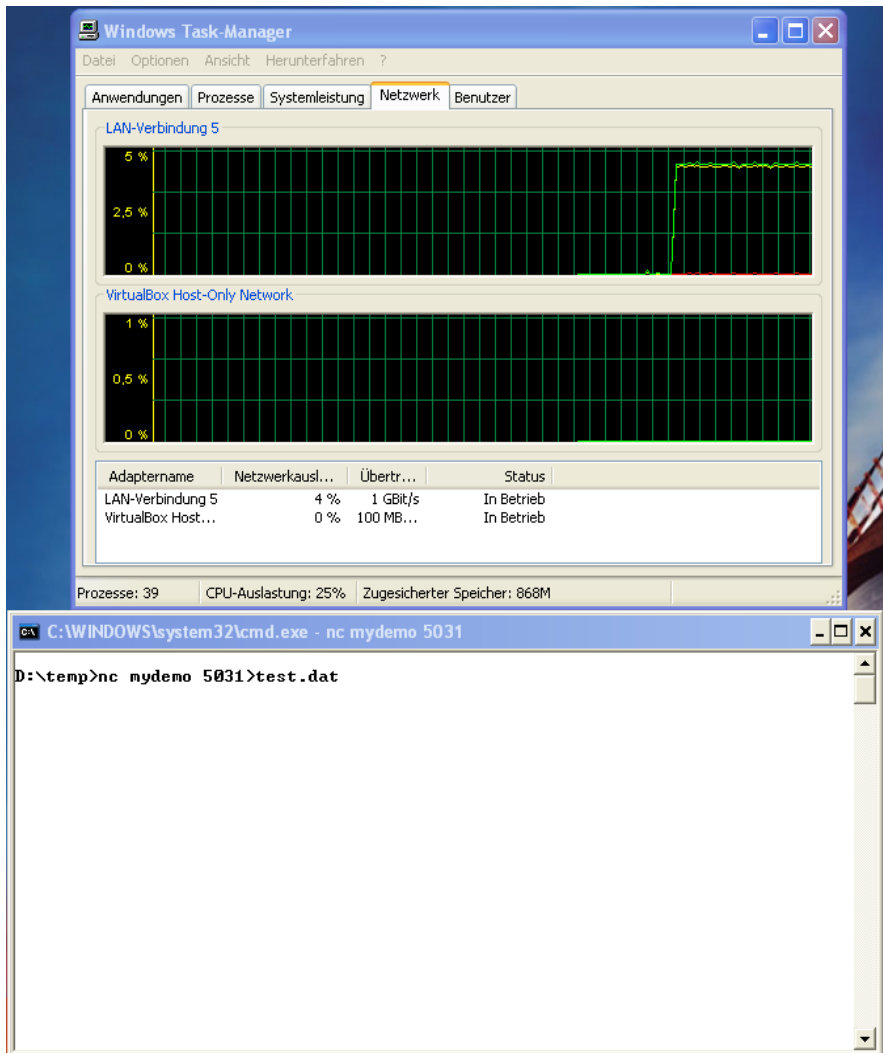
assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address: 192.168.168.128
```

On PC open the task manager and select the network tab:



Now open a netcat session and connect to the DSP:



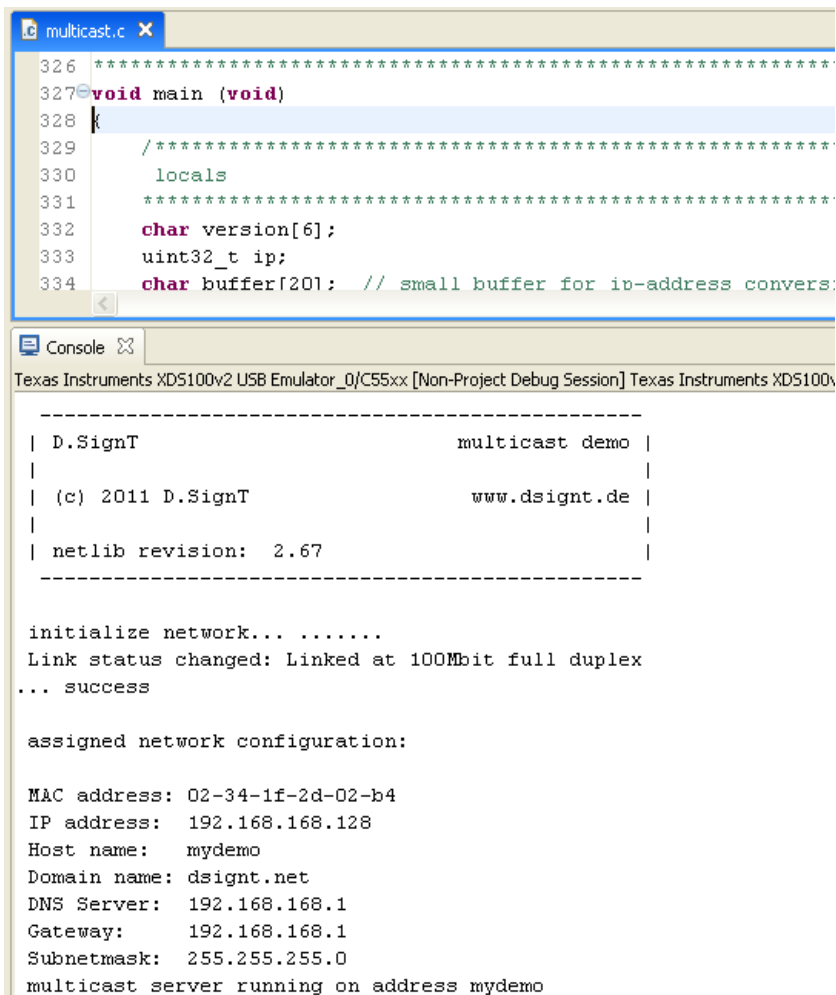
When connected, the DSP starts sending the TCP stream. All data is written into test.dat on PC.

3.2 Multicast

Multicast UDP messages are used to send a single message to many different devices simultaneously. To receive a multicast message a device must join the multicast group first:

`multicast_join_group (udp_socket, MCAST_IP_ADDR, MCAST_PORT);`

Load and run the multicast example in CCS:



```
multicast.c
326 *****
327 void main (void)
328 {
329     /*****
330     locals
331     *****/
332     char version[6];
333     uint32_t ip;
334     char buffer[20]; // small buffer for ip-address convers...
```

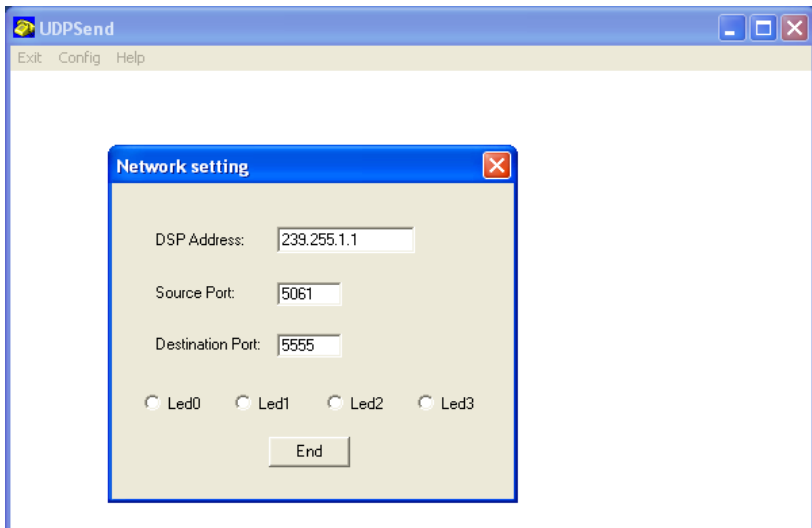
```
-----
| D.SignT                                multicast demo |
|                                         |
| (c) 2011 D.SignT                        www.dsignt.de |
|                                         |
| netlib revision:  2.67                  |
|                                         |
|-----|

initialize network... .....
Link status changed: Linked at 100Mbit full duplex
... success

assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address:  192.168.168.128
Host name:   mydemo
Domain name: dsignt.net
DNS Server:  192.168.168.1
Gateway:     192.168.168.1
Subnetmask:  255.255.255.0
multicast server running on address mydemo
```

Start the multicast UDPSend.exe on PC:



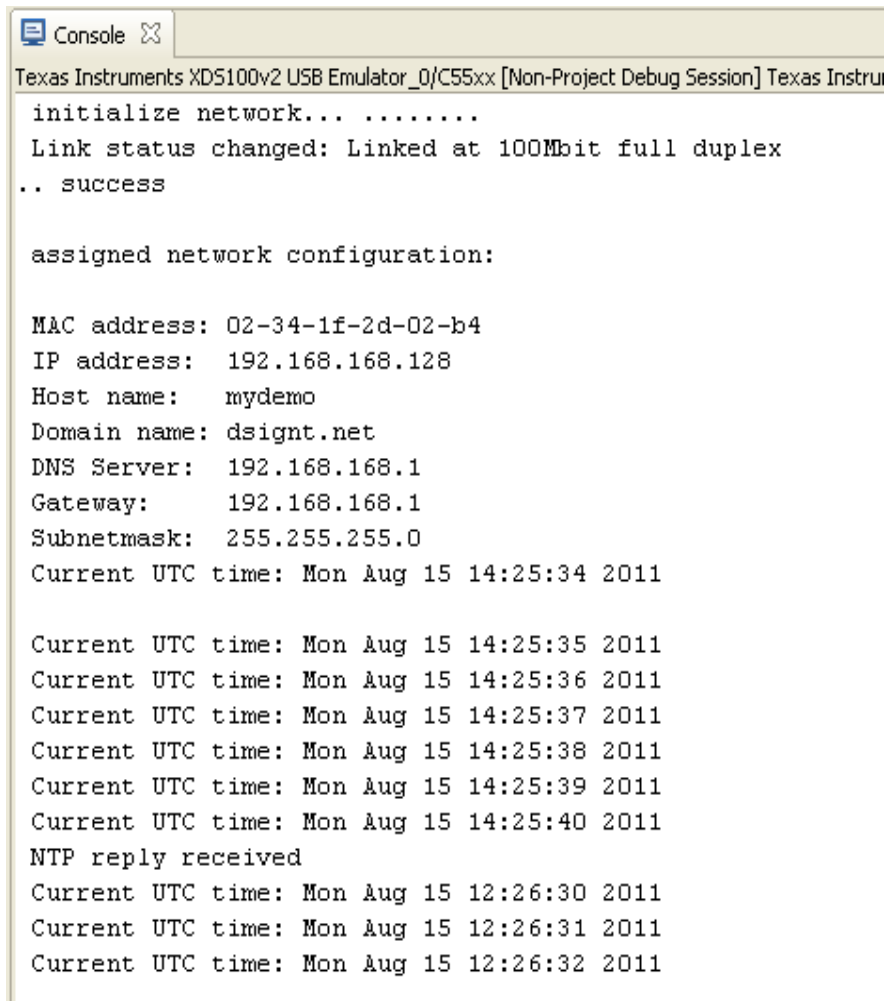
Any time you click on a LED a multicast message is sent to multicast group 239.255.1.1. The DSP program receives this message and changes the LED state:

```
Console [X]
Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas Instrumen
assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address: 192.168.168.128
Host name: mydemo
Domain name: dsight.net
DNS Server: 192.168.168.1
Gateway: 192.168.168.1
Subnetmask: 255.255.255.0
multicast server running on address mydemo
New multicast data received
Destination Address: 192.168.168.8
Multicast Group Address: 239.255.1.1
Destination Port: 5061
Source Multicast Port: 5555
New multicast data received
Destination Address: 192.168.168.8
Multicast Group Address: 239.255.1.1
Destination Port: 5061
Source Multicast Port: 5555
New multicast data received
Destination Address: 192.168.168.8
Multicast Group Address: 239.255.1.1
Destination Port: 5061
Source Multicast Port: 5555
```

3.3 SNTP – Simple Network Time Protocol

This source code implements the Simple Network Time Protocol. For proper function a SNTP server must be available in your network. Load and run SNTPTest on DSP:



```
Console X
Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas Instru
initialize network... ..
Link status changed: Linked at 100Mbit full duplex
.. success

assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address: 192.168.168.128
Host name: mydemo
Domain name: dsight.net
DNS Server: 192.168.168.1
Gateway: 192.168.168.1
Subnetmask: 255.255.255.0
Current UTC time: Mon Aug 15 14:25:34 2011

Current UTC time: Mon Aug 15 14:25:35 2011
Current UTC time: Mon Aug 15 14:25:36 2011
Current UTC time: Mon Aug 15 14:25:37 2011
Current UTC time: Mon Aug 15 14:25:38 2011
Current UTC time: Mon Aug 15 14:25:39 2011
Current UTC time: Mon Aug 15 14:25:40 2011
NTP reply received
Current UTC time: Mon Aug 15 12:26:30 2011
Current UTC time: Mon Aug 15 12:26:31 2011
Current UTC time: Mon Aug 15 12:26:32 2011
```

3.4 Webserver

The example HTTPJava is a java based webserver. The onboard LEDs are toggled via java script. In parallel a java class displays the current system time. Load and run HTTPJava:

The screenshot shows a code editor window titled 'HTTPJava.c' with the following code:

```

437 *****
438 void main ( void )
439 {
440     /*****
441         locals
442         *****/
443     char version[6];           /* for netlib version string */
444     uint32_t ip;
445     char buffer[20]; // small buffer for ip-address conversion

```

Below the code editor is a console window titled 'Console' showing the output of the program:

```

Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas Instruments XDS100v2
Current UTC time: Mon Aug 15 12:26:32 2011

-----
| D.SignT                java HTTP server demo |
|                         |                     |
| (c) 2011 D.SignT        www.dsignt.de        |
|                         |                     |
| netlib revision: 2.67   |                     |
|                         |                     |
-----

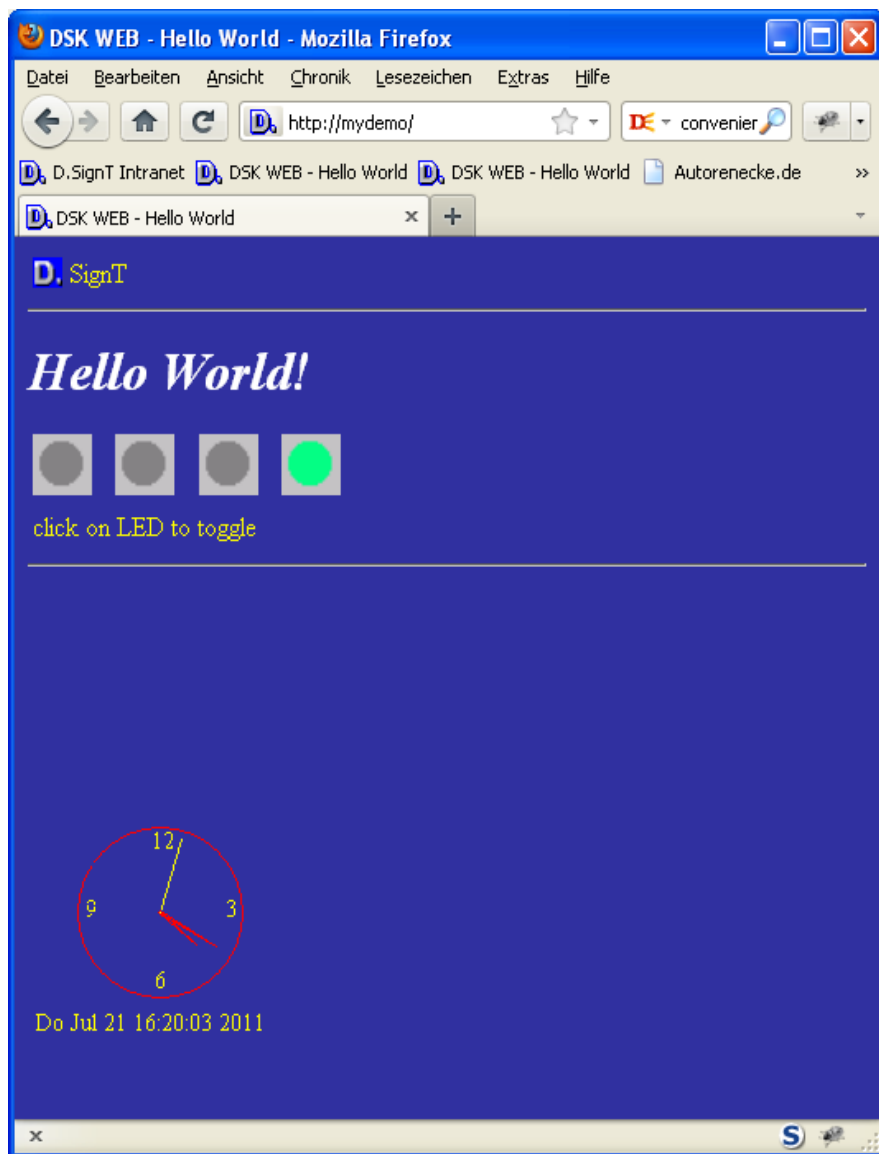
initialize network... .....
Link status changed: Linked at 100Mbit full duplex
.... success

assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address: 192.168.168.128
Host name: mydemo
Domain name: dsight.net
DNS Server: 192.168.168.1
Gateway: 192.168.168.1
Subnetmask: 255.255.255.0

```

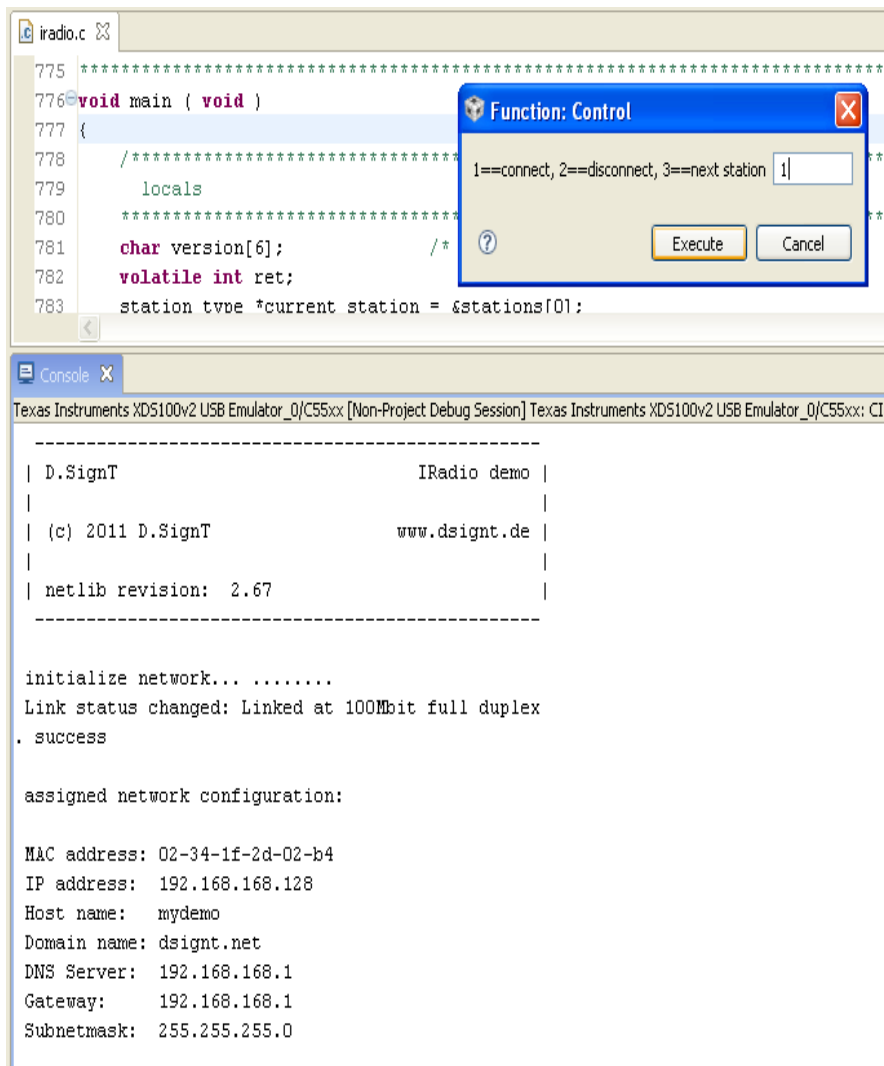
Now start your webbrowser and enter address mydemo:



Any time you click on a LED the LED is toggled.

3.5 IRadio – Internet Radio Stream

The IRadio application connects to predefined internet radio stations and receives the MP3 stream and title information. First open project IRadio and load GEL file iradiocontrol.gel. Open GEL function GEL-IRadioControl-Control



Enter 1 in Function: Control and press Execute or press key 1 on your DSP board. The DSP tries to connect to the first radio station. When the station is connected station and stream details will be requested and displayed. The incoming data stream is parsed for stream title information.

The screenshot displays the eZDSP5502net development environment. The top window shows the source code for `iradio.c`. A dialog box titled "Function: Control" is overlaid on the code, containing the text "1==connect, 2==disconnect, 3==next station" and a text input field with the value "1". The dialog has "Execute" and "Cancel" buttons. Below the code window is the "Console" window, which shows the output of the program. The console output includes copyright information, network initialization status, assigned network configuration (MAC, IP, Host, Domain, DNS, Gateway, Subnetmask), and the current station details (1LIVE, Westdeutscher Rundfunk Köln 2011) and stream title (Jason Derulo mit Dont wanna go home).

```

775  *****/
776 void main ( void )
777 {
778     /*****/
779     locals
780     *****/
781     char version[6];          /* fo
782     volatile int ret;
783     station type *current station = &stations[0];
  
```

Function: Control

1==connect, 2==disconnect, 3==next station 1

Execute Cancel

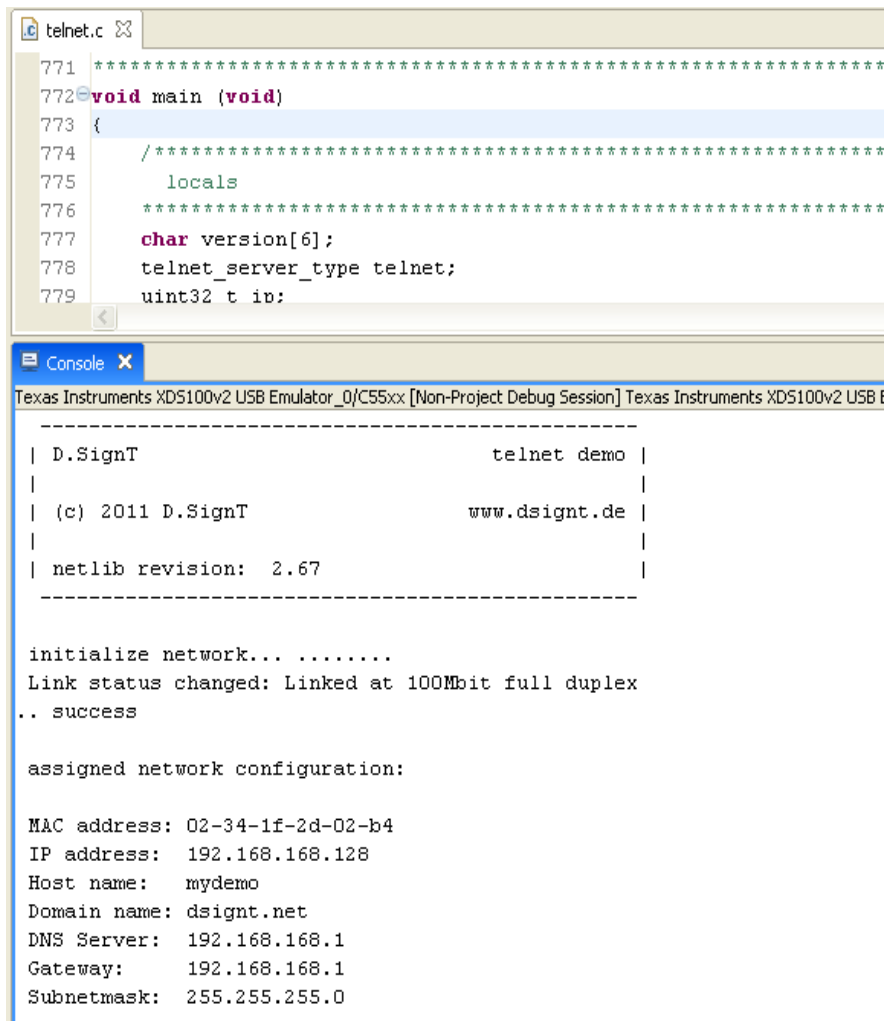
Texas Instruments XDS100v2 USB Emulator_0/C55xx [Non-Project Debug Session] Texas Instruments XDS100v2 USB Emulator_0/C55xx: CIO (

```

|
| (c) 2011 D.SignT          www.dsignt.de |
|
| netlib revision: 2.67
|-----
|
| initialize network... ..
| Link status changed: Linked at 100Mbit full duplex
| . success
|
| assigned network configuration:
|
| MAC address: 02-34-1f-2d-02-b4
| IP address: 192.168.168.128
| Host name: mydemo
| Domain name: dsignt.net
| DNS Server: 192.168.168.1
| Gateway: 192.168.168.1
| Subnetmask: 255.255.255.0
| Connecting to LIVE... connected
| Requesting station details...success
| Bitrate: 128kbit
| Station: 1live, © Westdeutscher Rundfunk Köln 2011
| Jason Derulo mit Dont wanna go home
| 1LIVE Beeck und Tobi Schlegl
  
```


3.6 Telnet Server

The telnet server is useful for obtaining statistical information about the current application and for in-field up- and downloads. Load and run the telnet server on CCS:



The screenshot shows the CCS IDE with a C file named `telnet.c` open. The code defines a `main` function that sets up a telnet server. The console window below shows the output of the program, which includes a banner, network initialization status, and assigned network configuration.

```
telnet.c
771 *****
772 void main (void)
773 {
774     /*****
775     locals
776     *****/
777     char version[6];
778     telnet_server_type telnet;
779     uint32 t_ip;
```

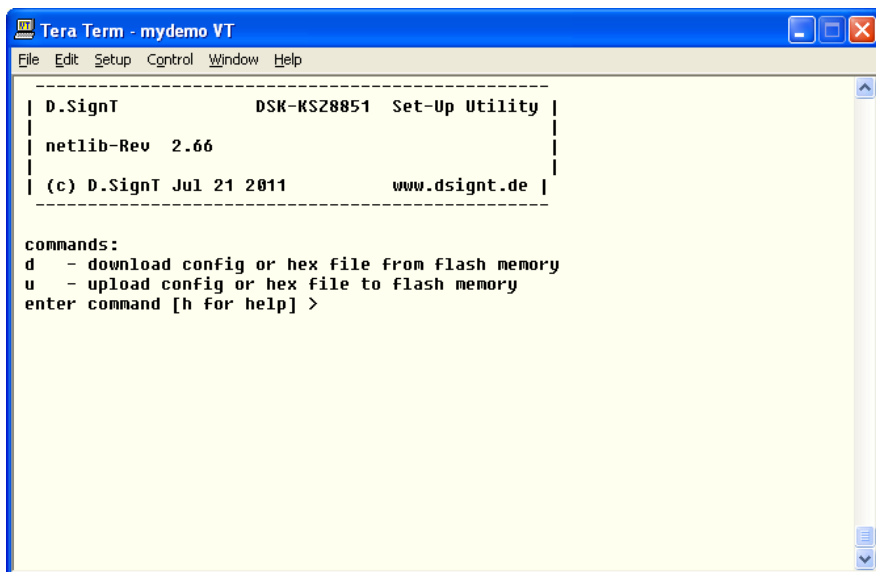
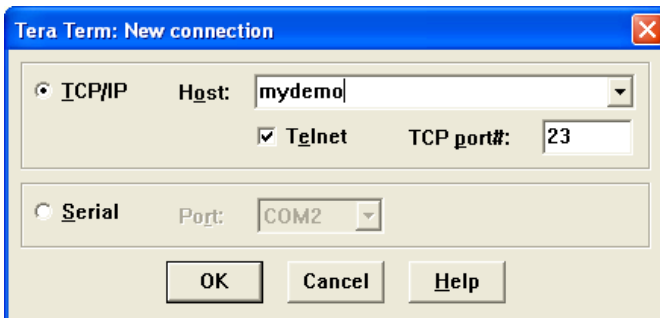
```
-----
| D.SignT                               telnet demo |
|                                         |
| (c) 2011 D.SignT                       www.dsignt.de |
|                                         |
| netlib revision:  2.67                  |
|                                         |
-----

initialize network... ..
Link status changed: Linked at 100Mbit full duplex
.. success

assigned network configuration:

MAC address: 02-34-1f-2d-02-b4
IP address:  192.168.168.128
Host name:   mydemo
Domain name: dsignt.net
DNS Server:  192.168.168.1
Gateway:     192.168.168.1
Subnetmask:  255.255.255.0
```

Now connect with a simple Telnet client to the DSP:



Two commands 'd' and 'u' are implemented. Press 'd' and the menu should change:

```

Tera Term - mydemo VT
File Edit Setup Control Window Help

-----
| D.SignT           DSK-KS28851 Set-Up Utility |
| netlib-Rev  2.66 |
| (c) D.SignT Jul 21 2011           www.dsignt.de |
-----

commands:
d  - download config or hex file from flash memory
u  - upload config or hex file to flash memory
enter command [h for help] > download <I>ntel-Hex or <C>onfig file [i|c] ? █

```

If you enter now 'I' for Intel-Hex, the DSP reads the FLASH content and sends it Intel-Hex coded to your telnet client:

```

Tera Term - demo.dsk VT
File Edit Setup Control Window Help

:20FDC000C627FEAFFCBFCDB9FFB7CDB4DC27D7A5EEAEAEAFEC68FFAFFDA6E6A59C23C4A300
:20FDE000FD3ECFACFE13EE92FDDA7D506D542CD35DC86FF0EF68FE62BF427C484E586E4A3CA
:20FE0000775FB7D9B3593359737333453373B771F76F7340374BB7D9B7D5734573F177F170
:20FE2000B759B3DDF7D937D9735F73D537D1F3513749F759F359B3D9335F33C9F3C377D9A0
:20FE4000B3FD33D9B359775977D173B9B3B9F7B9F76D3B5C731BF7E973F57329B7A1B719E5
:20FE60007719771D33DB73D13353B79DF7DB7759F7DB331F739977C9F35B37113743B7C198
:20FE80007359F77DB3793779B75F7751F75B7751B3593340F749336933473340F759F7594C
:20FEA000775D77597759B759B35F3359B759B359774FB359735BB35937353741335933494E
:20FEC0003339F37DF71BB719F3157331735137113349B37DB76BB359B35773497721331928
:20FEE000B75F771D375B731D731177543311731133493319F749B3593349B35137037709D1
:20FFF000B159735D375BB379F3753355F7F9B77173FD77E9B379F3933D777513751F351BB
:20FFF200B75D335D775977D9335D73513351F3D9775973CF33483369F75BB35D73D113C1C3
:20FFF400B37B375D3B5B3759A3D73315B759B3317B1931FDB32B73791387F3E97739B35945
:20FFF600B31B7319B31B9759375BF315F7193311F719B38D339977C9F39DF355B31B334108
:20FFF800B778B77DB37B37597379F351F75B3349334BF35937487369B373F370F35BB77913
:20FFFA00B55BF7DD335B33593355F75173513751B7497B5D7759F3597751B74CF74373516A
:20FFFC00B75D3319373BF75977157739F31917313B0B33193729374977113741F70177513D
:20FFFE000771B731DF35B3319F3153715B753B771F709775977483749B753B74D3719761148
:00000001FF
Success

commands:
d  - download config or hex file from flash memory
u  - upload config or hex file to flash memory
enter command [h for help] > █

```