Getting Started

IPC@CHIP Embedded Web Controller Family

Development Kit DK41 Order No. 20041011
Development Kit DK51 Order No. 538823



© 2005 Beck IPC GmbH

Table of Contents

1 INTRODUCTION 3

SYSTEM START-UP 4

2.1 DK40 DEVELOPMENT BOARD START-UP (SC12) 4
2.1.1 What do we need to get started? 4
2.1.2 Step 1 – Switching on the DK40 Development Board 5
2.1.3 Step 2 – Setting up the serial communication 6
2.1.4 Step 3 – Setting up the Ethernet communication 7
2.1.5 Step 4 – Downloading the DK40 Online demo program 11
2.1.6 Step 5 – Operating the Demo program 14
2.2 DK50 DEVELOPMENT BOARD START-UP (SC13) 15
2.2.1 What do we need to get started? 15
2.2.2 Step 1 – Switching on the DK50 Development Board 16
2.2.3 Step 2 – Setting up the serial communication 17
2.2.4 Step 3 – Setting up the Ethernet communication 18
2.2.5 Step 4 – Downloading the DK50 Online demo program 22
2.2.6 Step 5 – Operating the Demo program 25
2.2.7 Step 6 – Configuring the CF Card 26
2.3 DK50 DEVELOPMENT BOARD START-UP (SC11) 29
2.3.1 What do we need to get started? 29
2.3.2 Step 1 – Switching on the DK50 Development Board 30
2.3.3 Step 2 – Setting up the serial communication 31
2.3.4 Step 3 – Setting up the PPP communication 32
2.3.5 Step 4 – Downloading the DK50 Online demo program 41
2.3.6 Step 5 – Operating the Demo program 44
2.3.7 Step 6 – Configuring the CF Card 45

3 INFORMATION FOR ADVANCED USERS 48

3.1 MANUAL IP CONFIGURATION VIA THE TERMINAL 48

3.2 IPC@CHIP @CHIP-RTOS 49
3.2.1 Updates 49
3.2.2 Downloading RTOS updates 50
3.2.3 CHIP.INI configuration file 51
3.2.4 The command shell 52
3.2.5 The PPP Server/Client 52
3.2.6 The HTTP Web Server 53
3.3 WRITING EXECUTION PROGRAMS FOR THE IPC@CHIP 54
3.3.1 Introduction 54
3.3.2 IPC@CHIP API documentation 54
3.3.3 IPC@CHIP example programs 54
3.3.4 Program debugging 57
3.3.5 Stripping debug information from executables prior to download 59

4 APPENDIX 60

4.1 BASICS OF IP NETWORKING 60

4.2 WHERE TO FIND OTHER INFORMATION ON THE IPC@CHIP 61

4.3 WHERE TO GET SUPPORT 61

4.4 USEFUL WEB LINKS 61

4.5 ORDER NUMBERS 61

© 2005 Beck IPC GmbH

1 Introduction

Congratulations on purchasing your IPC@CHIP Development Kit. The Development Kits are complete ready-to-use development systems for building applications based on the Embedded Web Controller IPC@CHIP SC12 and IPC@CHIP SC13. Both products run with the @CHIP-RTOS operating system, which includes features like a real-time kernel that is capable of running common DOS applications concurrently, a full TCP/IP stack with a UDP/TCP socket interface, DHCP, FTP, PPP, Telnet, Web server etc.
The DK41 is the Development Kit for the Embedded Web Controller SC12. The Kit contains: a DK40 Evaluation Module, an Embedded Web Controller SC12 and an SM14 programming cable. Available as an option: a PSE4 power supply unit and the Borland C++ Software Development Environment:



DK41 Development Kit

The DK51 is the Development Kit for the new Embedded Web Controller SC13. The Kit contains within a systainer case a DK50 Development Board, an SC13 Embedded Controller, the Borland C++ Software Development Environment, a serial cable and an AC power supply unit with interchangeable plugs:



DK51 Development Kit

The Development Kits contain both hardware and software components required for a basic development. This document provides a step-by-step guide on how to put the Embedded Web Controllers SC12 and SC13 into operation, and concentrates on a system level start-up. For more information about hardware and software functionality please also read the IPC@CHIP API Documentation and IPC@CHIP Hardware Manual.

© 2005 Beck IPC GmbH

2 System Start-up

2.1 DK40 Development Board Start-up (SC12)



2.1.1 What do we need to get started?

- The DK41 Development Kit consisting of:
  - DK40 Development Board
  - Embedded Web Controller SC12
  - Null-modem cable
  - SM14 RS232 to TTL converter
- A PC running Windows 95/98/NT/2000/XP, with an Ethernet network interface
- Serial port on your PC
- Ethernet network with an available twisted pair connection for the DK40 Development Board
- A DC power supply (24 VDC will do fine)
- Windows HyperTerminal program

- Zip-files from the Beck Getting Started web site (http://www.beck-ipc.com/ipc/gettingstarted):
  - FTP client TotalCommander as shareware
  - CHIPTOOL Software
  - Application Visualisation Demo for DK40/DK41

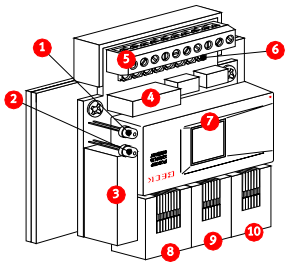
The Getting Started web site contains all the tools and programs required to put the Development Kit into operation. The FTP client is a shareware program. For more details please read the online documentation of this tool. You can use any FTP program which provides file transfer support.

ATTENTION	Be aware, that the Embedded Controller SC12 and DK40 Development Board are Electrostatic Sensitive Devices. Observe general precautions for handling.

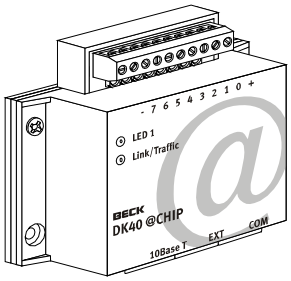
© 2005 Beck IPC GmbH

2.1.2 Step 1 – Switching on the DK40 Development Board

The DK40 Development Board is an evaluation platform for the Embedded Controller SC12. It provides two serial ports, an Ethernet interface, eight digital I/O, a 'link/traffic' LED, a free programmable LED and a switching DC/DC converter (15 ... 30V -> 5 V).



1. Free programmable user LED
2. Combined link/traffic/reset LED
3. Network transformer for twisted pair
4. Interference capacitor
5. Socket strip with screw terminal for voltage supply and combined inputs/outputs
6. Control LEDs for combined inputs/outputs
7. IPC @CHIP SC10/SC11/SC12
8. Twisted pair network connection for 10Base T
9. EXT interface
10. COM interface



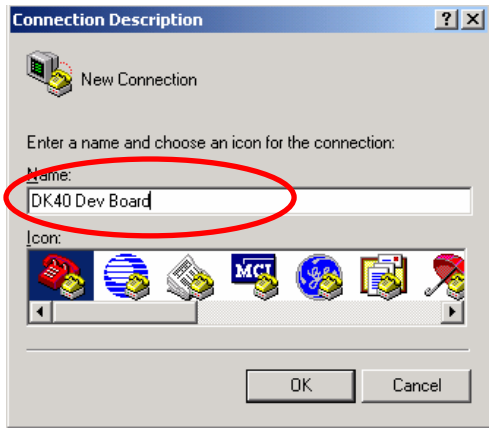
First of all, you have to place the SC12 into the socket inside the DK40. The small dot on the top of the SC12 diagonally opposite to the BECK logo indicates pin 1. The notch in the socket indicates the position of pin 1 in the socket. Please also refer to the drawing of the DK40 and make sure you insert the chip with the correct orientation.
Connect the power supply (24 VDC) to the DK40 socket strip. GND must be connected to the pin marked "-". +24 VDC must be connected to the pin marked '+'.

© 2005 Beck IPC GmbH

2.1.3 Step 2 – Setting up the serial communication

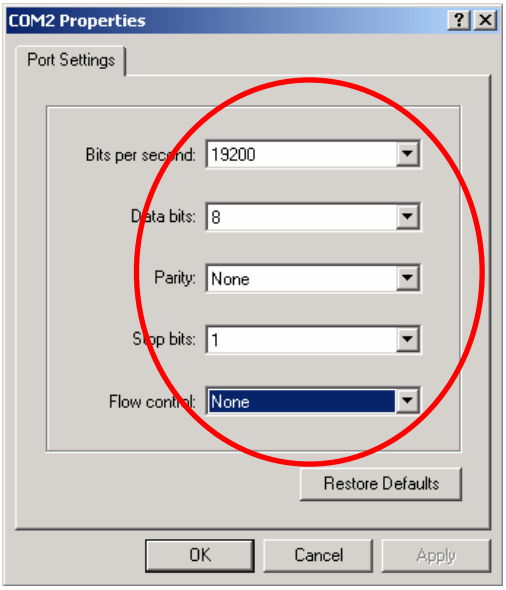
Please connect the null modem cable and SM14 converter from the DK41 Development Kit to your PC's COM port and to the COM port of the DK40 (10).

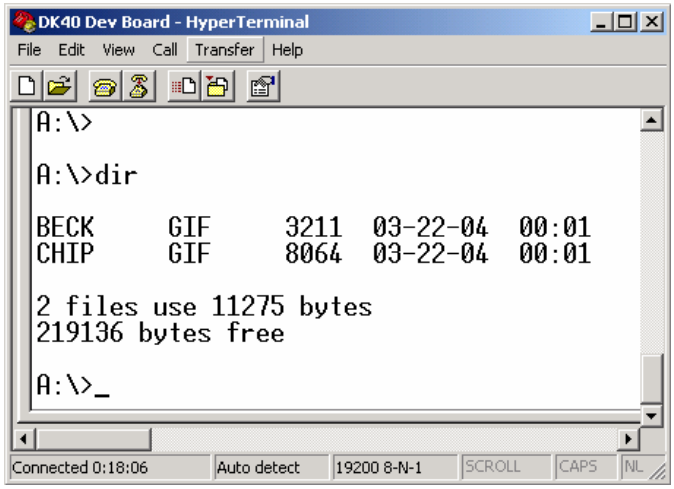
You then need to launch the Windows HyperTerminal program. If it's not part of your current Windows installation, you should install the application from your Windows CD.
Start the HyperTerminal program. Enter a name for the new connection (e.g. DK40 Dev Board). Choose the connection via the COM port of your PC.





Set the communication parameters to 19200 baud, 8 data bits, no parity, one stop bit, no handshake. The HyperTerminal program now connects the PC with the DK40. Press Enter when the terminal window is opened to see the 'A:>' prompt for the drive A: on the Embedded Controller. Enter the command 'dir' and you will see the root directory of the Embedded Controller.





The Embedded Controller is factory shipped with the two graphics files "BECK.GIF" and "CHIP.GIF" on the drive.

© 2005 Beck IPC GmbH

2.1.4 Step 3 – Setting up the Ethernet communication

The following chapter describes the set-up of the Ethernet communication. Before moving on, you must decide if you want to use either DHCP[1] or fixed addressing. With DHCP, the IPC@CHIP retrieves a valid IP configuration from a DHCP server. With fixed addressing, you define the IP address, network mask and an optional gateway[2].

The default configuration for the IPC@CHIP is DHCP since this will ensure that the IPC@CHIP adapts to an existing network if this network has a DHCP server. If you do not have a DHCP server, the IPC@CHIP will not disturb the operation of your network. We cannot ship with a fixed IP configuration since we have no knowledge of your network configuration.

a) DHCP IP configuration: (default setting)
In a network supporting DHCP the DK40 retrieves a valid IP configuration from a DHCP server automatically.

Connect the DK40 to your Ethernet network hub or switch with a standard twisted pair cable. Reset your DK40 with power off and power on. On the terminal window (HyperTerminal) you will see the output from the start-up procedure. Here you will see the serial number, @CHIP-RTOS version, build date and other start-up information. If an IP configuration could be successfully received from the DHCP server, this IP configuration will be displayed at the end of the start-up output messages. In the event of an error, the message “DHCP request failed” will be output after a timeout of several seconds.

```
@CHIP-RTOS
Copyright (c) BECK IPC GmbH       1999-2004
Copyright (c) Elmic Systems, Inc. 1999-2004
Target=SC12 Serialnumber=020C2
Bootstrap version V2.24
@CHIP-RTOS version SC12 V1.10 LARGE
Build: Mar 22 2004
Serial EXT Port using DMA receive, IRQ send
Serial COM Port using DMA receive, IRQ send
Starting TCPIP
Init network
Ethernet packet interface
Config server
FTP server
Web server
Telnet server
Ext. disk interface.
TCPIP device driver API
Software SPI API
@CHIP-RTOS boot successfully finished

Leased IP 172.30.11.228
Mask      255.255.240.0
Gateway   172.30.0.2
Leasing time: 3600 secs
```

[1] DHCP: "Dynamic Host Configuration Protocol“.

[2] Gateway: "Computers that interconnect two networks and pass packets from one to the other are called internet gateways or internet routers."

© 2005 Beck IPC GmbH

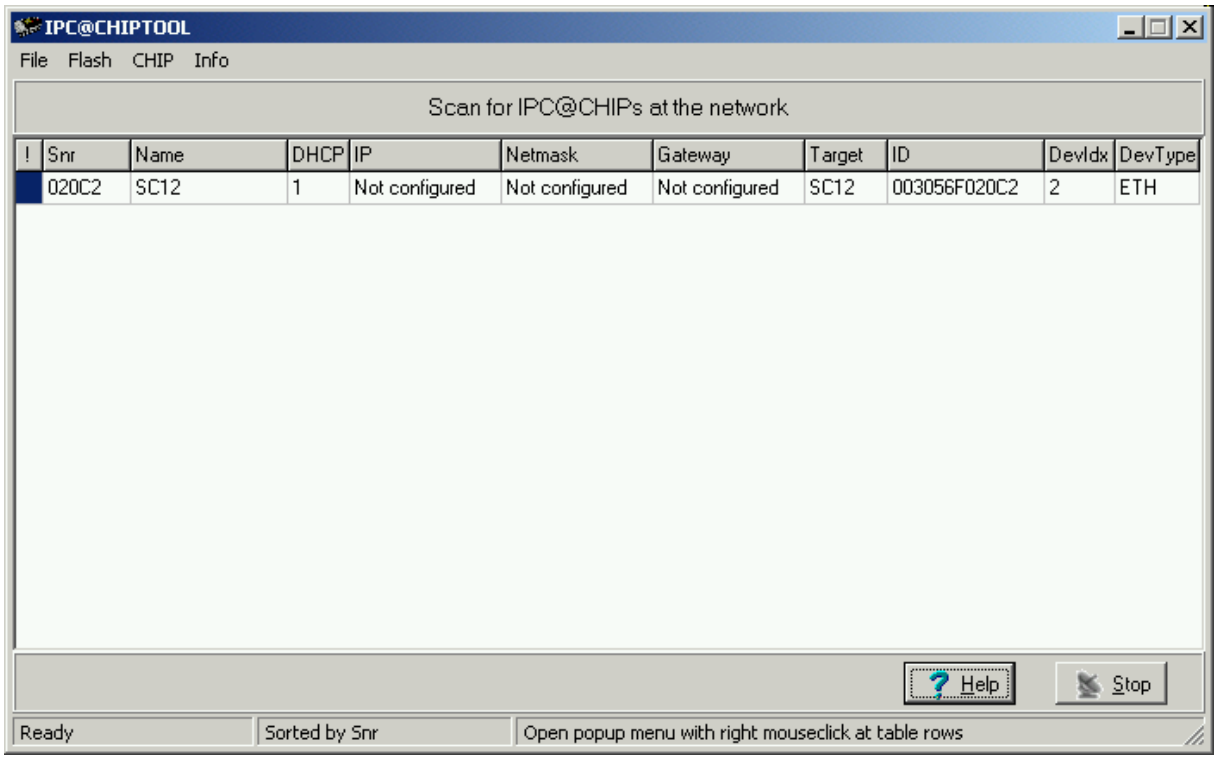
b) Manual IP configuration:
If your local network does not include a DHCP server, you must assign an IP address manually. Please contact your network administrator for details about your local network. Your administrator can give you a valid IP address setting for your network. A brief introduction to IP basics in TCP/IP networks is given in the Appendix.

For Ethernet connection you can use two alternatives:
1. You have a network with multiple clients. The network uses hubs and/or switches. In this case, you need a regular twisted pair Ethernet cable running from one of your hubs to the DK40.
2. You have a network consisting of only two devices (your PC and the DK40) without a hub or a 'switch'. In this case you need a so-called 'cross over' twisted pair Ethernet cable.

To review or change the IP configuration, you can either access the IPC@CHIP using a terminal program connected to the COM port of the IPC@CHIP (see chapter "Manual IP Configuration via the Terminal"), or use the 'CHIPTOOL' software that communicates with the IPC@CHIP over the Ethernet. The method using the CHIPTOOL software is described below.

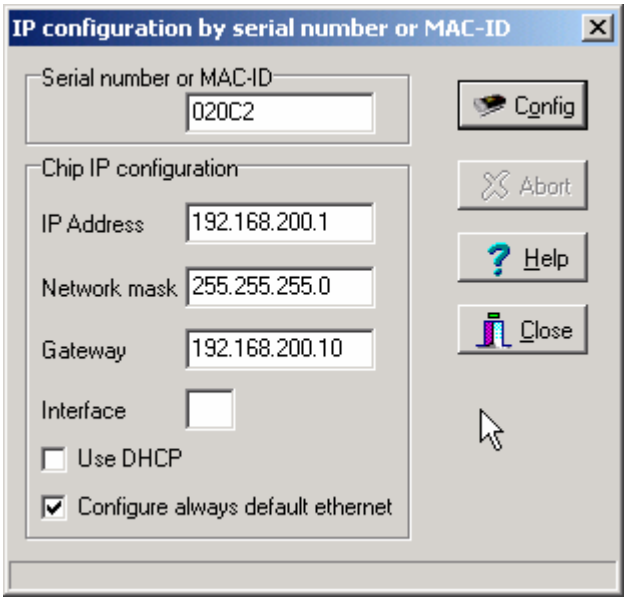
The Windows application CHIPTOOL can be used to manage IPC@CHIP products. CHIPTOOL provides network configuration and software updates of IPC@CHIPs during the development and production process of IPC@CHIP-based products. It is also useful for third-party customers to manage their IPC@CHIP-based products. This tool can be downloaded from the IPC@CHIP home page: http://www.beck-ipc.com/ipc in the download area. For more details about CHIPTOOL take a look at the integrated help documentation of the application.

Launch the CHIPTOOL software. To verify the current IP configuration, take a look at the main window of the CHIPTOOL application. This table will show all IPC@CHIP targets it finds on this section of the local network. The scan is repeated continuously. New devices that are switched on will be found within seconds. The IPC@CHIP will be displayed as "not configured" if a DHCP server is not available.



© 2005 Beck IPC GmbH

Select the menu option ‘Chip’ and ‘IP Config’. Fill in the dialog as illustrated below:



The serial number data shown here is just an example. At ‘Serial number or MAC-ID’ you enter the serial number of the IPC@CHIP that you want to configure.

The serial number is a 5 digit hexadecimal number (valid hexadecimal digits: 0..9 and A..F) reported to the terminal at start-up. The serial number is also displayed in the main window of CHIPTOOL next to every IPC@CHIP detected and is also printed on a label on every IPC@CHIP.

Since SC12@CHIP-RTOS version 1.10Beta, it is also possible to enter the MAC-address (we use this address also as a unique ID[3] to identify IPC@CHIPs) of the internal Ethernet controller. This ID is shown at the column ID of the CHIPTOOL’s network scan window.

In the area ‘Chip IP configuration’ you enter the IP address, network mask and optionally the gateway address.

Now click the ‘Config’ button. This will send a UDP broadcast datagram over the network, which includes the serial number of the IPC@CHIP and the new IP configuration.

Check in the main window of CHIPTOOL that the IP configuration you have entered is correctly displayed.

The IP configuration information is now stored in the file ‘CHIP.INI’ on the flash disk.

[3] This address is unique because Ethernet controllers must have a world-wide unique 6 byte address.

© 2005 Beck IPC GmbH

In the terminal window (HyperTerminal), press the ‘Enter’ key once until you see the ‘A:>’ prompt. Enter the command ‘ipcfg’ followed by the ‘Enter’ key. The output will show the current IP configuration of all the supported device interfaces. For example:

```
A:\>ipcfg

Device  = NE2000
Type    = ETH
Idx     = 2
IP      = 192.168.200.1
Netmask = 255.255.255.0
DHCP    = 0
Mac     = 00 30 56 F0 20 C2

Device  = LOOPBACK
Type    = LPK
Idx     = 1
IP      = 127.0.0.1
Netmask = 255.255.255.255

Gateway = 192.168.200.10
```

The figure above shows the IP configuration of the Ethernet device and of the “Internal loopback device”, which is a TCP/IP-internal device that can be used for connections to the same host.
Please note down the IP address of the Ethernet device for you will need it later.

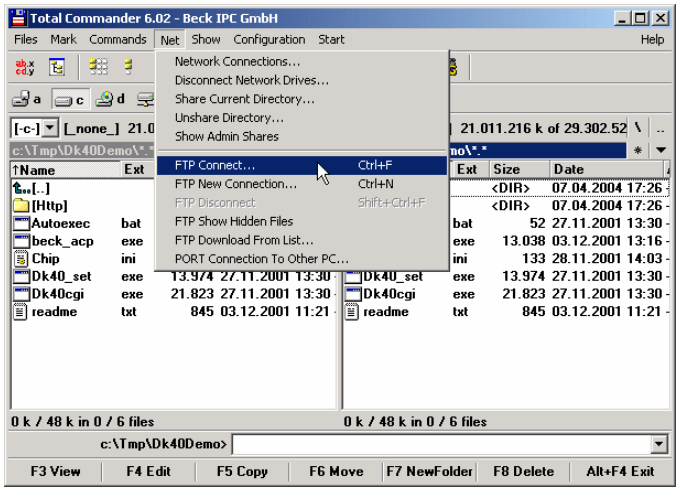
© 2005 Beck IPC GmbH

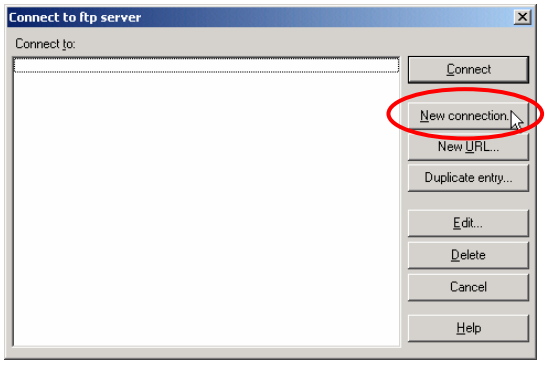
2.1.5 Step 4 – Downloading the DK40 Online demo program

The following describes how to download files to the IPC@CHIP and put the DK40 online demo into operation on your DK40.

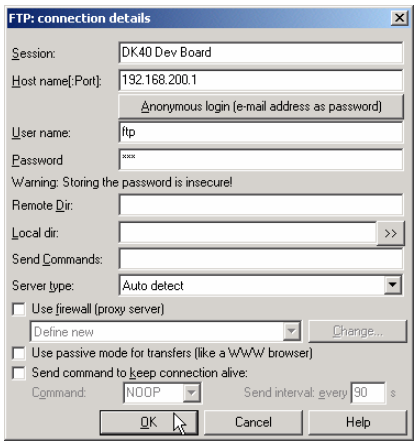
The RTOS of the IPC@CHIP features an FTP server for transferring files. FTP stands for File Transfer Protocol and has become a standard for transferring files over TCP/IP networks. All you need to transfer a file from or to the IPC@CHIP is an FTP client. Your Windows system only provides an FTP client with a command line interface. Clearly, a graphical user interface, such as with the WS-FTP or TotalCommander, is much more user-friendly.
The following explains the data transfer procedure using the TotalCommander shareware program.

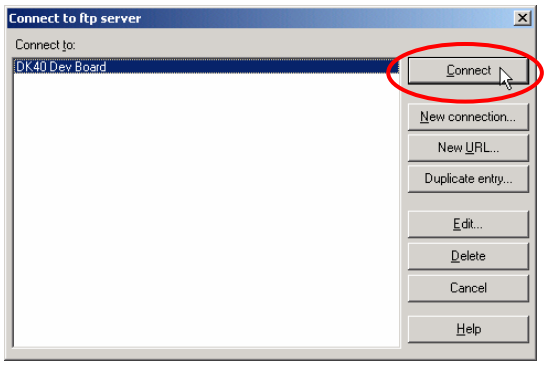
Install and start TotalCommander. In the left-hand pane select the path where the DK40 online demo files are located. Switch to the right-hand pane and choose FTP Connect from the Net menu. Click “New connection” in the dialog that opens.





Under "Session" enter a name for the connection, such as “DK40 Dev Board”. The IP address of the IPC@CHIP that you should have noted down before must be specified in the “Host name” field. To connect to the FTP server, you need also a user name and a password. For the IPC@CHIP you use ‘ftp’ as user name and also ‘ftp’ as password. These are default settings that can be changed at a later time.
Click OK to confirm and close the dialog. In the previous dialog select the new connection and click “Connect".





© 2005 Beck IPC GmbH

You should now be connected to the FTP server of the IPC@CHIP and the drive content of the IPC@CHIP should now be visible in the right-hand pane of TotalCommander. Note that the FTP connection will timeout if you don't issue any actions for some time. You will then have to reconnect. The procedure now depends on whether you have configured the IP address of the IPC@CHIP by hand or with DHCP.

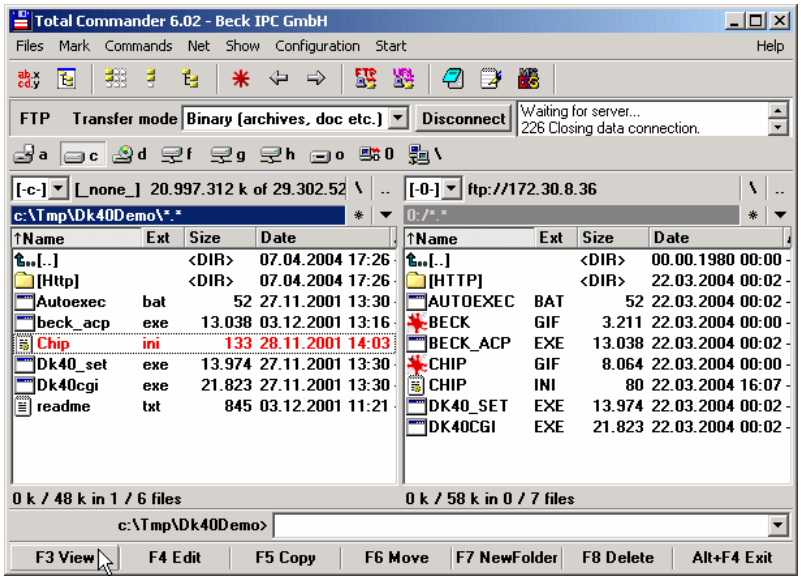
2.1.5.1 IP address was configured using DHCP

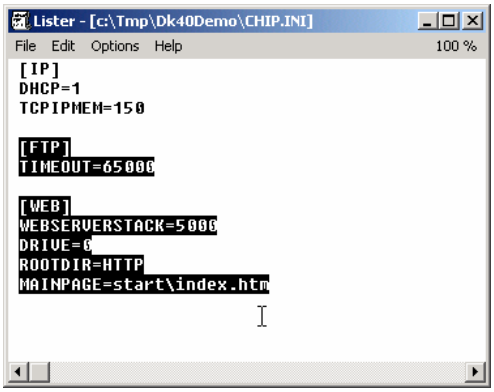
Select all files and subfolders in the path of the DK40 Online Demo. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.

The DK40 must be restarted in order to activate the demo. To do this, switch the power off and on.

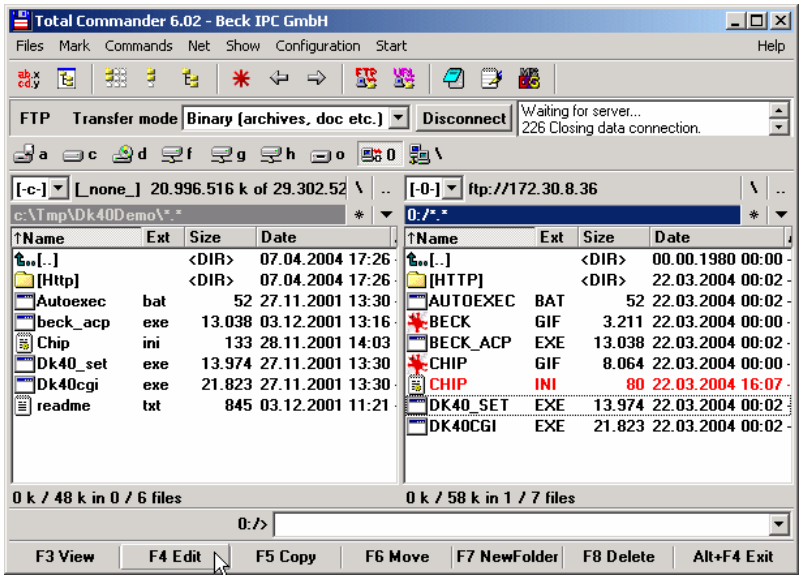
2.1.5.2 IP address was configured by hand

Select all files and subfolders in the path of the DK40 Online Demo apart from the CHIP.INI file. This will ensure that you do not overwrite the IP configuration you have entered beforehand. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.
Now select the CHIP.INI file on the PC and click the "F3 View" button to view the content of the file. Select all the text apart from the section [IP] and copy it to the Clipboard with CTRL+C. Then close the window.

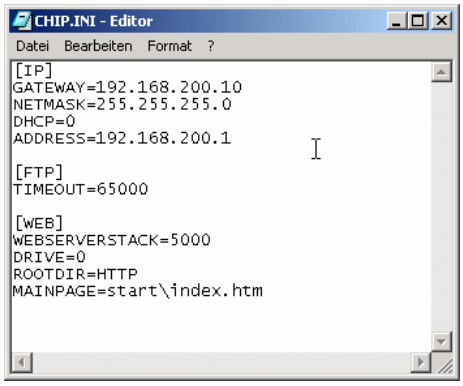




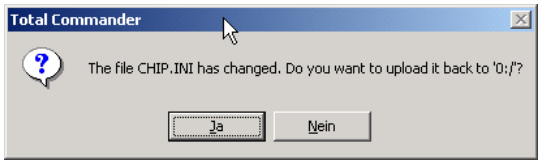
Now choose the CHIP.INI file on the IPC@CHIP and click “F4 Edit” to edit the file.



Paste the content of the Clipboard into the file after the IP section with CTRL+V.



Save the file and close the editor. In the dialog prompt that follows, you must then confirm that you wish to upload the modified CHIP.INI file onto the IPC@CHIP. You then have to confirm that you wish to overwrite the existing file.





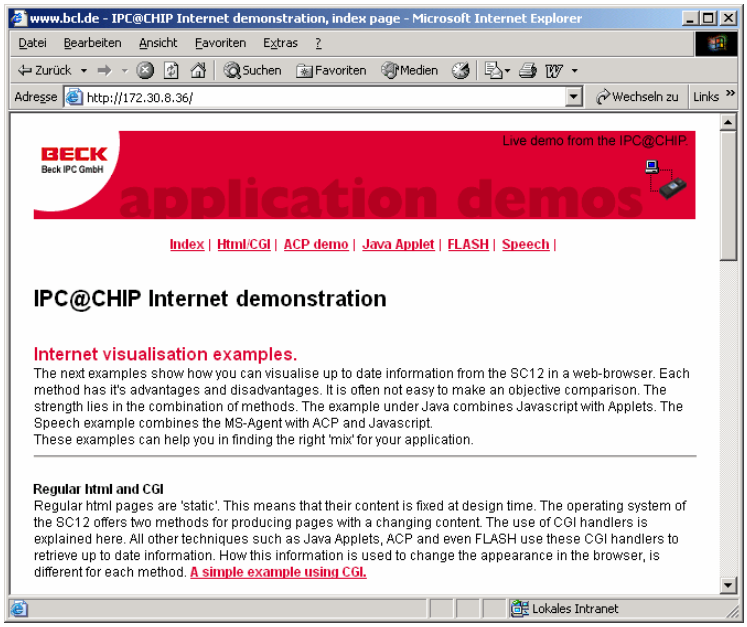
You can close the FTP connection by clicking the “Disconnect” button once all files have been transferred.

The DK40 must be restarted in order to activate the demo. To do this, switch the power off and on.

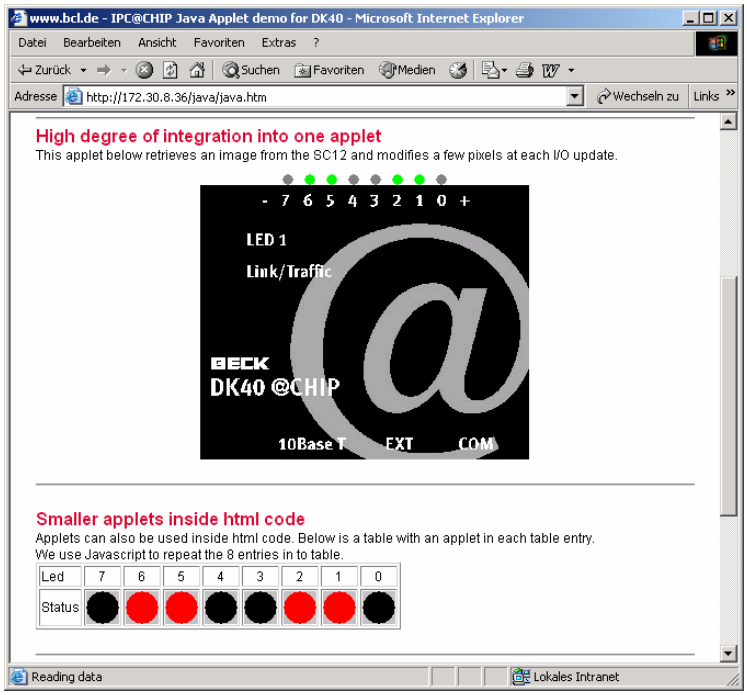
© 2005 Beck IPC GmbH

2.1.6 Step 5 – Operating the Demo program

Start your Internet browser and enter the IP address of your IPC@CHIP (the one that you have noted down) in the address bar. The start page of the DK40 Online Demo will appear. Please choose the Java Applet example from the start page. The security settings of your browser must allow the use of Java applets.



A Java applet is then loaded that visually represents the digital I/Os of the DK40. Click the fields marked 7 to 0 in order to switch the outputs of the DK40 on and off.



2.2 DK50 Development Board Start-up (SC13)



2.2.1 What do we need to get started?

- The DK51 Development Kit consisting of:
  - DK50 Development Board
  - Embedded Web Controller SC13 (on the DK50)
  - Power supply unit
  - Null-modem cable
  - Borland C++ Software Development Environment
- A PC running Windows 95/98/NT/2000/XP, with an Ethernet network interface
- Serial port on your PC
- Ethernet network with an available twisted pair connection for the DK50 Development Board
- Windows HyperTerminal program
- Optional: Compact Flash Card, if you want to configure the CF Interface (not needed for start-up)

- Zip-files from the Beck Getting Started web site (http://www.beck-ipc.com/ipc/gettingstarted):
  - FTP client TotalCommander as shareware
  - CHIPTOOL Software
  - Application Visualisation Demo for DK50/DK51
  - IDE Driver package (Containing EXTIDE.EXE)

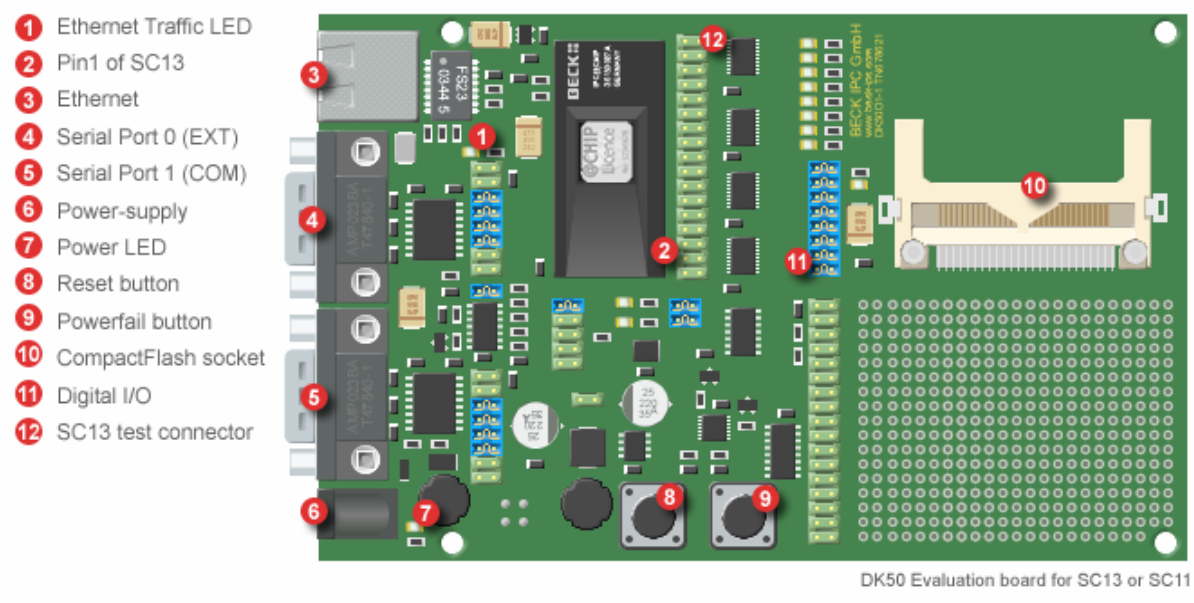
The Getting Started web site contains all the tools and programs required to put the Development Kit into operation. The FTP client is a shareware program. For more details please read the online documentation of this tool. You can use any FTP program which provides file transfer support.

	Be aware, that the Embedded Controller SC13 and DK50 Development Board are Electrostatic Sensitive Devices. Observe general precautions for handling.

© 2005 Beck IPC GmbH

2.2.2 Step 1 – Switching on the DK50 Development Board

The DK50 Development Board is an evaluation platform for the Embedded Controller SC13 and SC11. Among other functions, it provides two serial ports, an Ethernet interface, eight digital I/O, a 'traffic' LED[4], CompactFlash Socket and a switching DC/DC converter.



DK50 Evaluation board for SC13 or SC11

The AC power supply is delivered with country specific AC adapters. Please choose the right adapter for your country. Then connect the AC power supply to your main supply and to the DK50 Development Board. The Power LED (7) will light up and Traffic LED will light up for less than half a second.

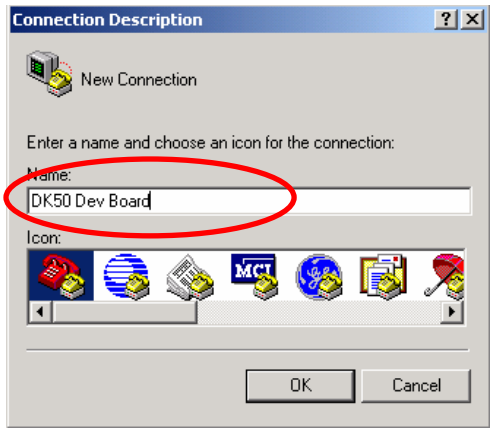
[4] Please note: The SC12 has a combined Ethernet Link/Traffic function. The SC13 has only Ethernet Traffic function.

© 2005 Beck IPC GmbH

2.2.3 Step 2 – Setting up the serial communication

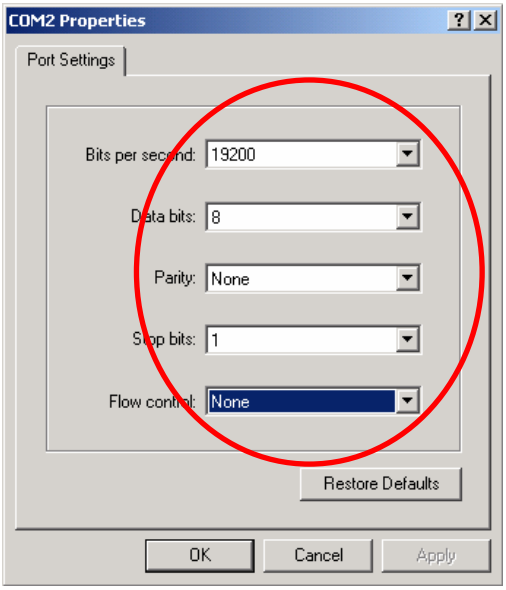
Please connect the null modem cable from the DK51 Development Kit to your PC's COM port and to the COM port of the DK50 (5).

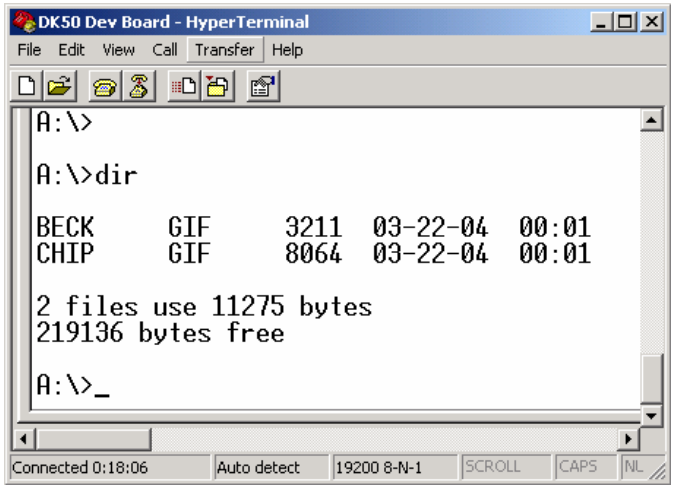
Next, you need the Windows HyperTerminal program. If it's not part of your current Windows installation, you should install the application from your Windows CD.
Start the HyperTerminal program. Enter a name for the new connection (e.g. DK50 Dev Board). Choose the connection via the COM port of your PC.





Set the communication parameters to 19200 baud, 8 data bits, no parity, one stop bit, no handshake. The HyperTerminal program now connects the PC with the DK50. Press Enter when the terminal window is opened to see the 'A:>' prompt for the drive A: on the Embedded Controller. Enter the command 'dir' and you will see the root directory of the Embedded Controller.





The Embedded Controller is factory shipped with the two graphics files "BECK.GIF" and "CHIP.GIF" on the drive.

© 2005 Beck IPC GmbH

2.2.4 Step 3 – Setting up the Ethernet communication

The following chapter describes the set-up of the Ethernet communication. Before moving on, you must decide if you want to use either DHCP[5] or fixed addressing. With DHCP, the IPC@CHIP retrieves a valid IP configuration from a DHCP server. With fixed addressing, you define the IP address, network mask and an optional gateway[6].

The default configuration for the IPC@CHIP is DHCP since this will ensure that the IPC@CHIP adapts to an existing network if this network has a DHCP server. If you do not have a DHCP server, the IPC@CHIP will not disturb the operation of your network. We cannot ship with a fixed IP configuration since we have no knowledge of your network configuration.

a) DHCP IP configuration: (default setting)
In a network supporting DHCP the DK50 retrieves a valid IP configuration from a DHCP server automatically.

Connect the DK50 to your Ethernet network hub or switch with a standard twisted pair cable. Press the Reset button (8) of your DK50. On the terminal window (HyperTerminal) you will see the output from the start-up procedure. Here you will see the serial number, @CHIP-RTOS version, build date and other start-up information. If an IP configuration could be successfully received from the DHCP server, this IP configuration will be displayed at the end of the start-up output messages. In the event of an error, the message “DHCP request failed” will be output after a timeout of several seconds.

```
@CHIP-RTOS
Copyright (c) BECK IPC GmbH       1999-2004
Copyright (c) Elmic Systems, Inc. 1999-2004
Target=SC13 Serialnumber=00017
Bootstrap version V2.25
@CHIP-RTOS version SC13 V1.00 LARGE
Build: Mar 22 2004
Serial EXT Port using DMA receive, IRQ send
Serial COM Port using DMA receive, IRQ send
Starting TCPIP
Init network
Ethernet packet interface
Config server
FTP server
Web server
Telnet server
Ext. disk interface.
TCPIP device driver API
Software SPI API
@CHIP-RTOS boot successfully finished

Leased IP 172.30.11.228
Mask      255.255.240.0
Gateway   172.30.0.2
Leasing time: 3600 secs
```

[5] DHCP: "Dynamic Host Configuration Protocol“.

[6] Gateway: "Computers that interconnect two networks and pass packets from one to the other are called internet gateways or internet routers."

© 2005 Beck IPC GmbH

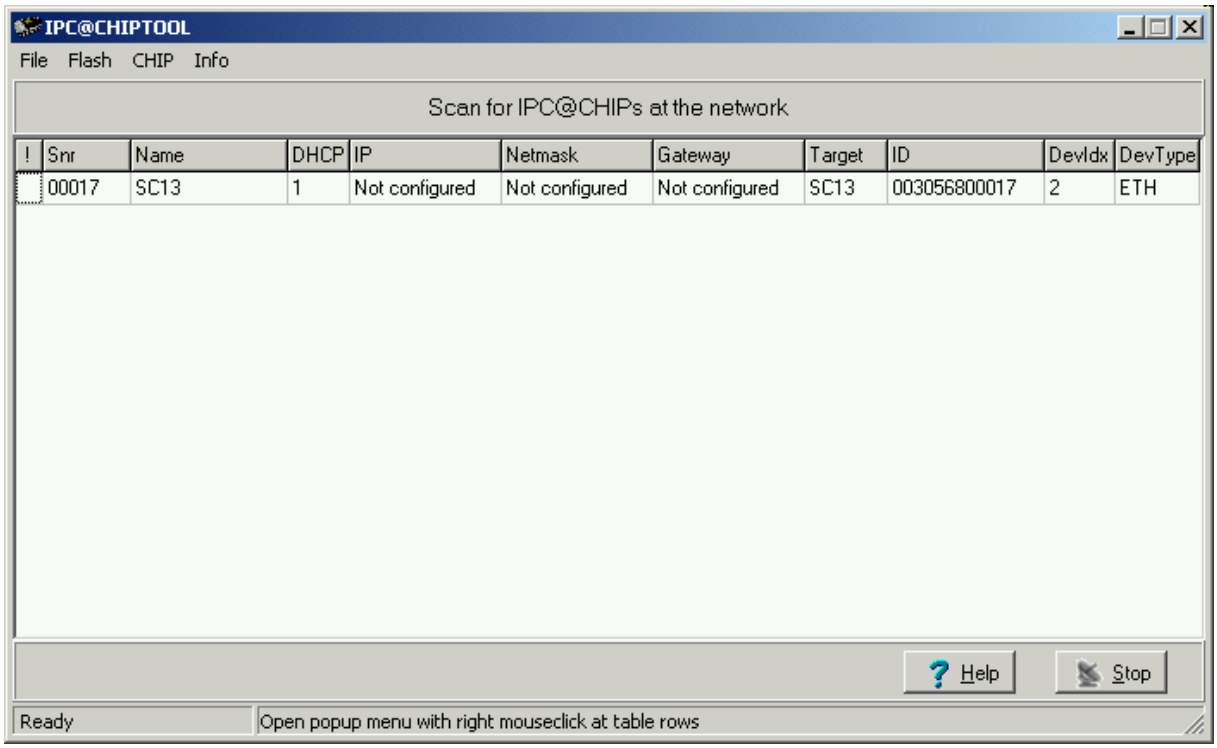
b) Manual IP configuration:
If your local network does not include a DHCP server, you must assign an IP address manually. Please contact your network administrator for details about your local network. Your administrator can give you a valid IP address setting for your network. A brief introduction to IP basics in TCP/IP networks is given in the Appendix.

For Ethernet connection you can use two alternatives:
1. You have a network with multiple clients. The network uses hubs and/or switches. In this case, you need a regular twisted pair Ethernet cable running from one of your hubs to the DK50.
2. You have a network consisting of only two devices (your PC and the DK50) without a hub or a 'switch'. In this case you need a so-called 'cross over' twisted pair Ethernet cable.

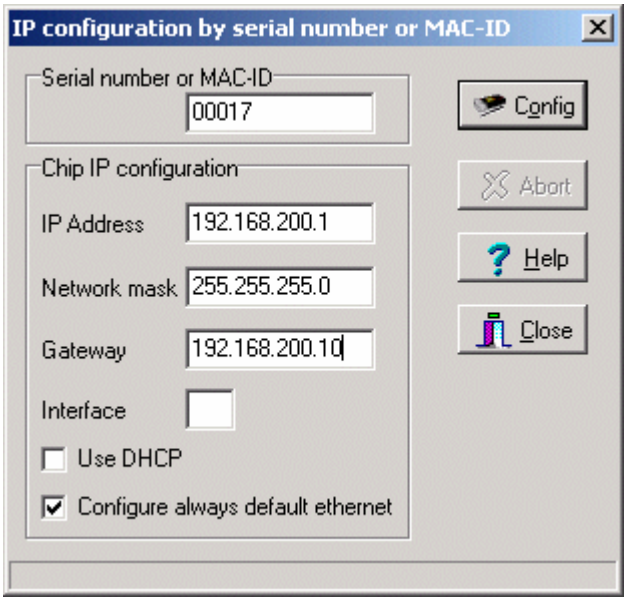
To review or change the IP configuration, you can either access the IPC@CHIP using a terminal program connected to the COM port of the IPC@CHIP (see chapter "Manual IP Configuration via the Terminal"), or use the 'CHIPTOOL' software that communicates with the IPC@CHIP over the Ethernet. The method using the CHIPTOOL software is described below.

The Windows application CHIPTOOL can be used to manage IPC@CHIP products. CHIPTOOL provides network configuration and software updates of IPC@CHIPs during the development and production process of IPC@CHIP-based products. It is also useful for third-party customers to manage their IPC@CHIP-based products. This tool can be downloaded from the IPC@CHIP home page: http://www.beck-ipc.com/ipc in the download area. For more details about CHIPTOOL take a look at the integrated help documentation of the application.

Launch the CHIPTOOL software. To verify the current IP configuration, take a look at the main window of the CHIPTOOL application. This table will show all IPC@CHIP targets it finds on this section of the local network. The scan is repeated continuously. New devices that are switched on will be found within seconds. The IPC@CHIP will be displayed as "not configured" if a DHCP server is not available.



Select the menu option ‘Chip’ and ‘IP Config’. Fill in the dialog as illustrated below :



The serial number data shown here is just an example. At ‘Serial number or MAC-ID’ you enter the serial number of the IPC@CHIP that you want to configure.

The serial number is a 5 digit hexadecimal number (valid hexadecimal digits: 0..9 and A..F) reported to the terminal at start-up. The serial number is also displayed in the main window of CHIPTOOL next to every IPC@CHIP detected and is also printed on a label on every IPC@CHIP.

Since SC13@CHIP-RTOS version 0.91Beta, it is also possible to enter the MAC-address (we use this address also as a unique ID[7] to identify IPC@CHIPs) of the internal Ethernet controller. This ID is shown at the column ID of the CHIPTOOL’s network scan window.

In the area ‘Chip IP configuration’ you enter the IP address, network mask and optionally the gateway address.

Now click the ‘Config’ button. This will send a UDP broadcast datagram over the network, which includes the serial number of the IPC@CHIP and the new IP configuration.

Check in the main window of CHIPTOOL that the IP configuration you have entered is correctly displayed.

The IP configuration information is now stored in the file ‘CHIP.INI’ on the flash disk.

[7] This address is unique, because Ethernet controllers must have a world-wide unique 6 byte address.

© 2005 Beck IPC GmbH

In the terminal window (HyperTerminal), press the ‘Enter’ key once until you see the ‘A:>’ prompt. Enter the command ‘ipcfg’ followed by the ‘Enter’ key. The output will show the current IP configuration of all the supported device interfaces. For example:

```
A:\>ipcfg

Device  = 91C111
Type    = ETH
Idx     = 2
IP      = 192.168.200.1
Netmask = 255.255.255.0
DHCP    = 0
Mac     = 00 30 56 80 00 17
Status  = Link ok, 10 MBit/s, Halfdplx

Device  = LOOPBACK
Type    = LPK
Idx     = 1
IP      = 127.0.0.1
Netmask = 255.255.255.255

Gateway = 192.168.200.10
```

The figure above shows the IP configuration of the Ethernet device and of the “Internal loopback device”, which is a TCP/IP-internal device that can be used for connections to the same host.
Please note down the IP address of the Ethernet device for you will need it later.

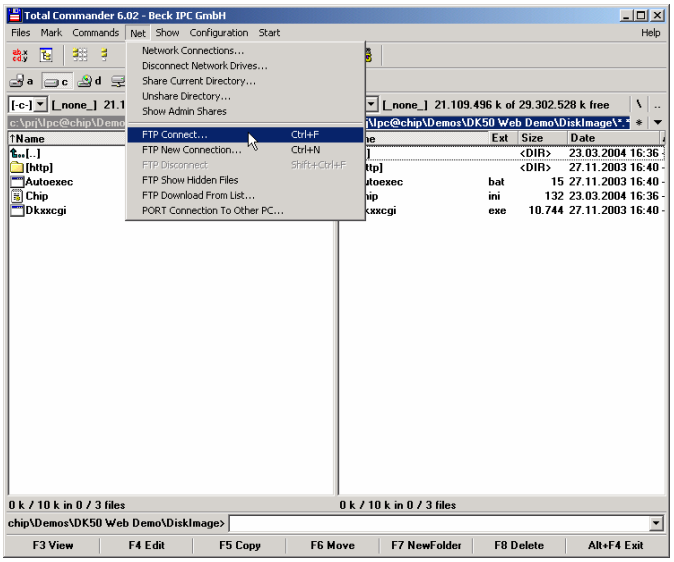
© 2005 Beck IPC GmbH

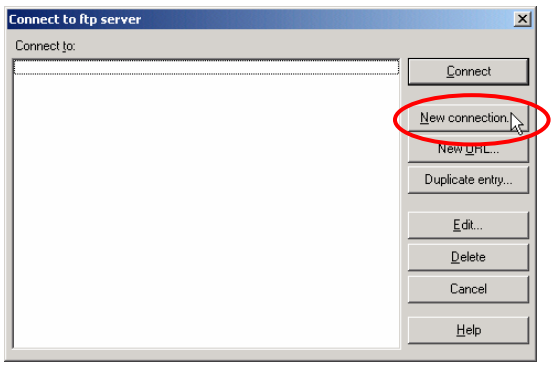
2.2.5 Step 4 – Downloading the DK50 Online demo program

The following describes how to download files to the IPC@CHIP and put the DK50 online demo into operation on your DK50.

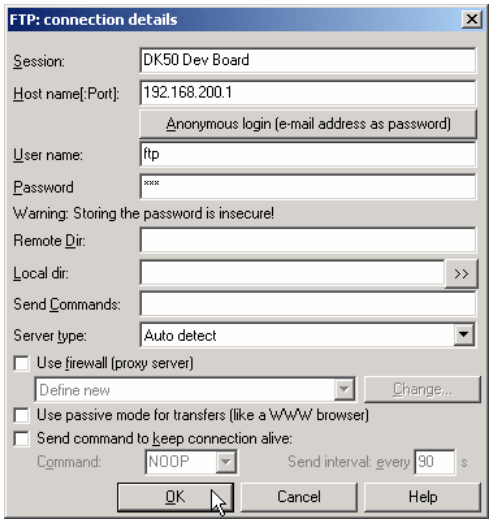
The RTOS of the IPC@CHIP features an FTP server for transferring files. FTP stands for File Transfer Protocol and has become a standard for transferring files over TCP/IP networks. All you need to transfer a file from or to the IPC@CHIP is an FTP client. Your Windows system only provides an FTP client with a command line interface. Clearly, a graphical user interface, such as with the WS-FTP or TotalCommander, is much more user-friendly.
The following explains the data transfer procedure using the TotalCommander shareware program.

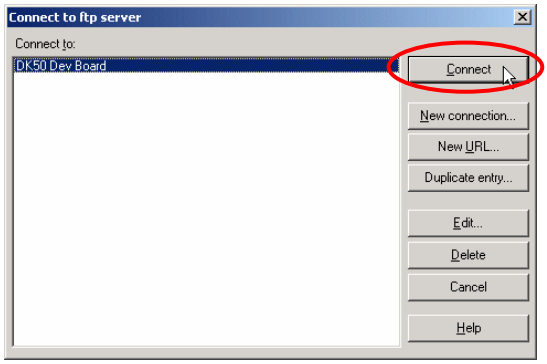
Install and start TotalCommander. In the left-hand pane select the path where the DK50 online demo files are located. Switch to the right-hand pane and choose FTP Connect from the Net menu. Click “New connection” in the dialog that opens.





Under "Session" enter a name for the connection, such as “DK50 Dev Board”. The IP address of the IPC@CHIP that you should have noted down before must be specified in the “Host name” field. To connect to the FTP server, you need also a user name and a password. For the IPC@CHIP you use ‘ftp’ as user name and also ‘ftp’ as password. These are default settings that can be changed at a later time.
Click OK to confirm and close the dialog. In the previous dialog select the new connection and click “Connect".





© 2005 Beck IPC GmbH

You should now be connected to the FTP server of the IPC@CHIP and the drive content of the IPC@CHIP should now be visible in the right-hand pane of TotalCommander. Note that the FTP connection will timeout if you don't issue any actions for some time. You will then have to reconnect. The procedure now depends on whether you have configured the IP address of the IPC@CHIP by hand or with DHCP.

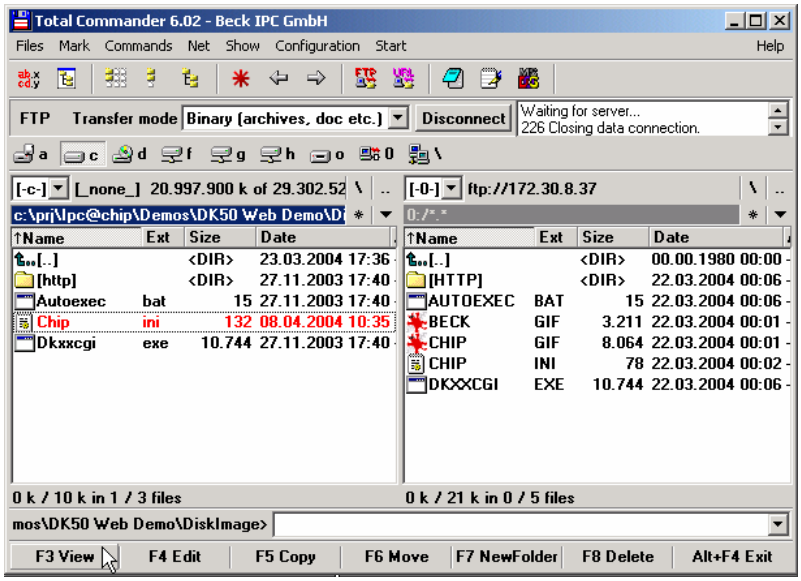
2.2.5.1 IP address was configured using DHCP

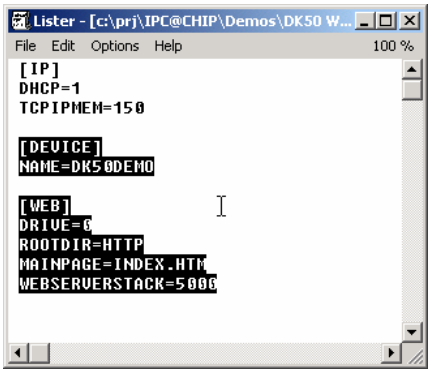
Select all files and subfolders in the path of the DK50 Online Demo. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.

The DK50 must be restarted in order to activate the demo. To do this, press the Reset button.

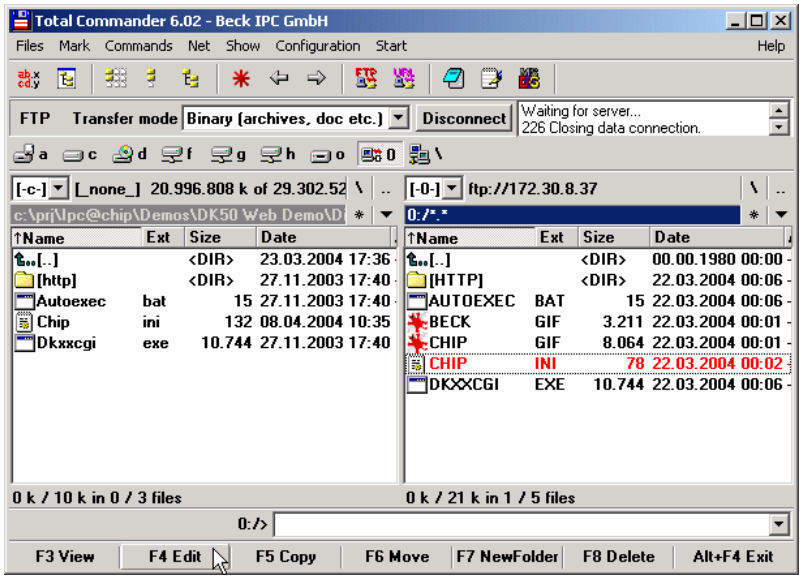
2.2.5.2 IP address was configured by hand

Select all files and subfolders in the path of the DK50 Online Demo apart from the CHIP.INI file. This will ensure that you do not overwrite the IP configuration you have entered beforehand. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.
Now select the CHIP.INI file on the PC and click the "F3 View" button to view the content of the file. Select all the text apart from the section [IP] and copy it to the Clipboard with CTRL+C. Then close the window.

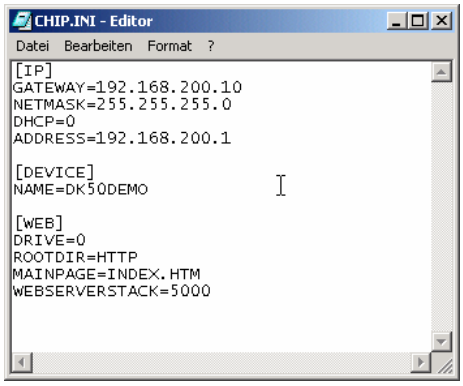




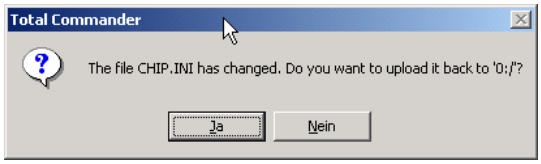
Now choose the CHIP.INI file on the IPC@CHIP and click “F4 Edit” to edit the file.



Paste the content of the Clipboard into the file after the IP section with CTRL+V.



Save the file and close the editor. In the dialog prompt that follows, you must then confirm that you wish to upload the modified CHIP.INI file onto the IPC@CHIP. You then have to confirm that you wish to overwrite the existing file.





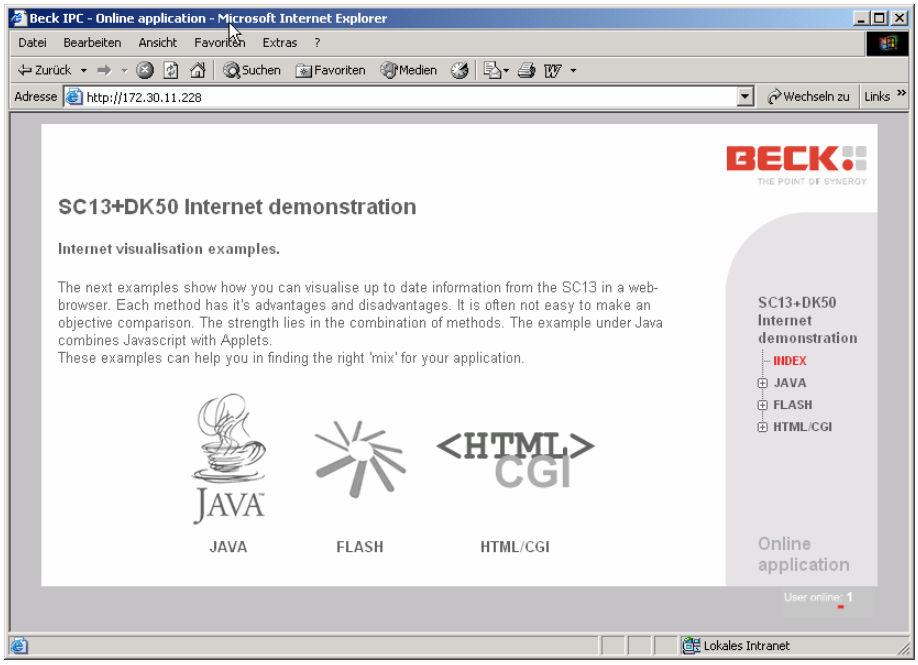
You can close the FTP connection by clicking the “Disconnect” button once all files have been transferred.

The DK50 must be restarted in order to activate the demo. To do this, press the Reset button.

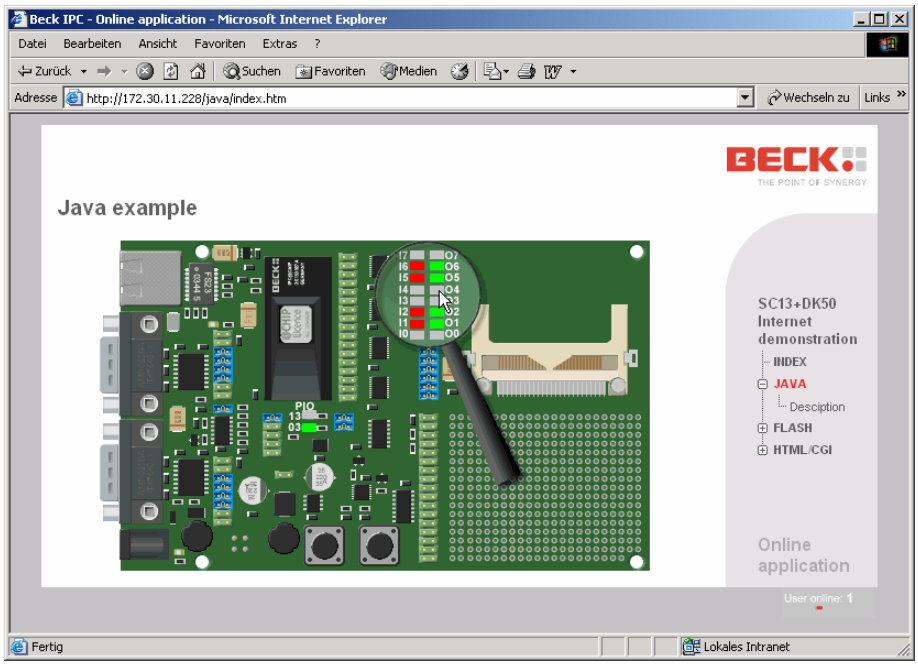
© 2005 Beck IPC GmbH

2.2.6 Step 5 – Operating the Demo program

Start your Internet browser and enter the IP address of your IPC@CHIP (the one that you have noted down) in the address bar. The start page of the DK50 Online Demo will appear. Please choose the Java example from the start page. The security settings of your browser must allow the use of Java applets.



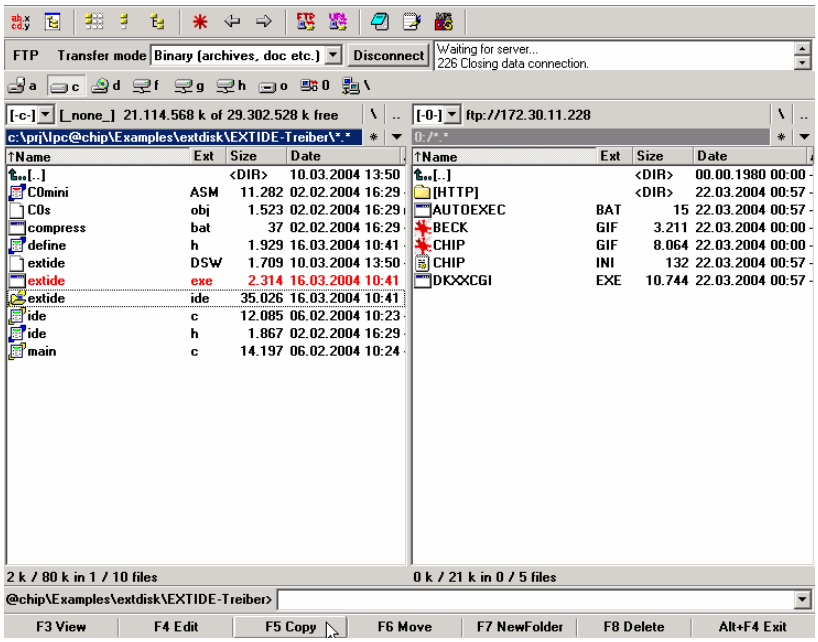
A Java applet is then loaded that visually represents the digital I/Os of the DK50. Click the fields marked O7 to O0 in order to switch the outputs of the DK50 on and off. The LEDs on PIOs 13 and 3 can also be switched on and off in this way.



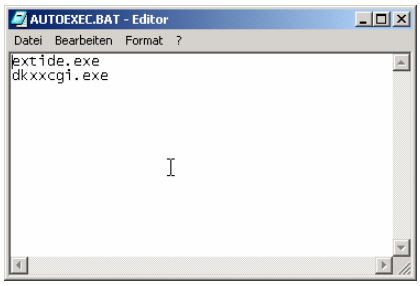
2.2.7 Step 6 – Configuring the CF Card

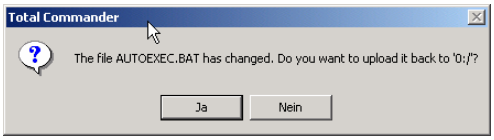
In order to put a CompactFlash card in DK50 into operation, you need the CompactFlash driver "EXTIDE.EXE" from the IDE Driver package on drive A: of the IPC@CHIP. First disconnect the DK50 from the power supply. Insert the CompactFlash card into the DK50. Reconnect the DK50 to the power supply.

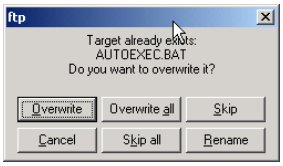
Copy the "EXTIDE.EXE" driver onto the drive of the IPC@CHIP using TotalCommander.



Now edit the Autoexec.bat file on the IPC@CHIP by selecting the file and clicking the "F4 Edit" button. Add the entry “exitde.exe” in the first line of the Autoexec.bat file. Save the file and close the editor. In the dialog prompt that follows, you must then confirm that you wish to upload the modified Autoexec.bat file onto the IPC@CHIP. You then have to confirm that you wish to overwrite the existing file.







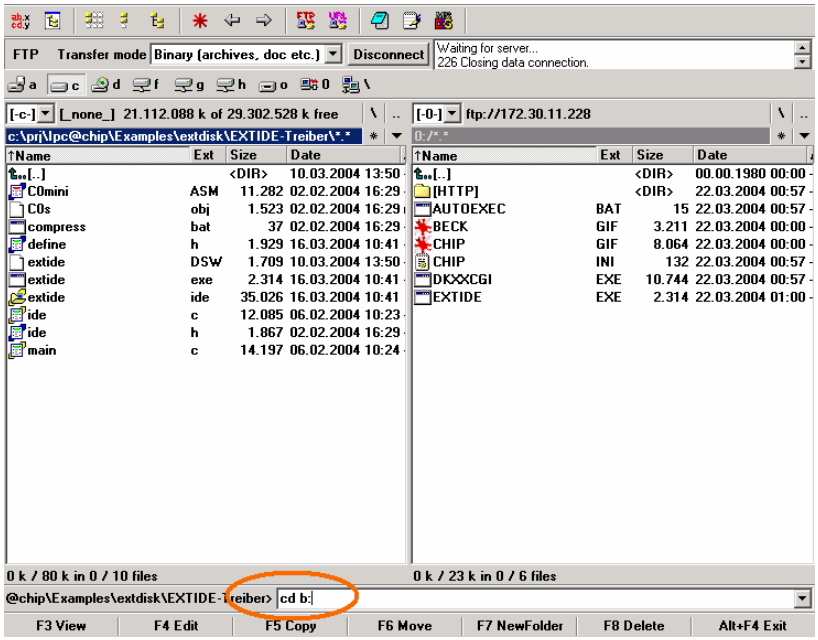
Close the FTP connection once more by clicking the "Disconnect" button and restart the DK50 via the Reset button.

The following message should then appear in the HyperTerminal program during the startup. This confirms that the CompactFlash has been installed correctly as drive B:.

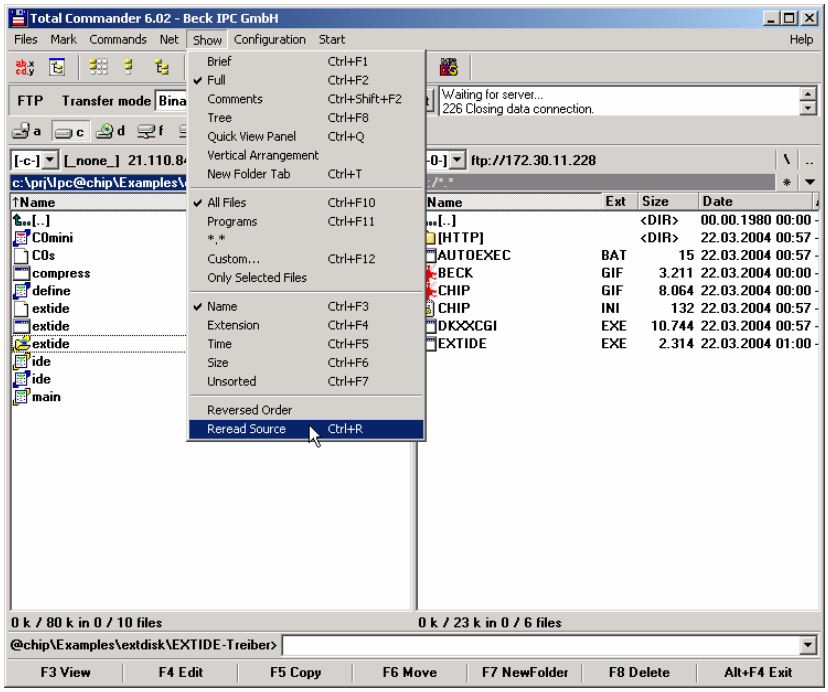
```
External IDE disk driver V1.14 Interface Version (0x0001)
Mounting partition: 1
Drive B: installed
```

© 2005 Beck IPC GmbH

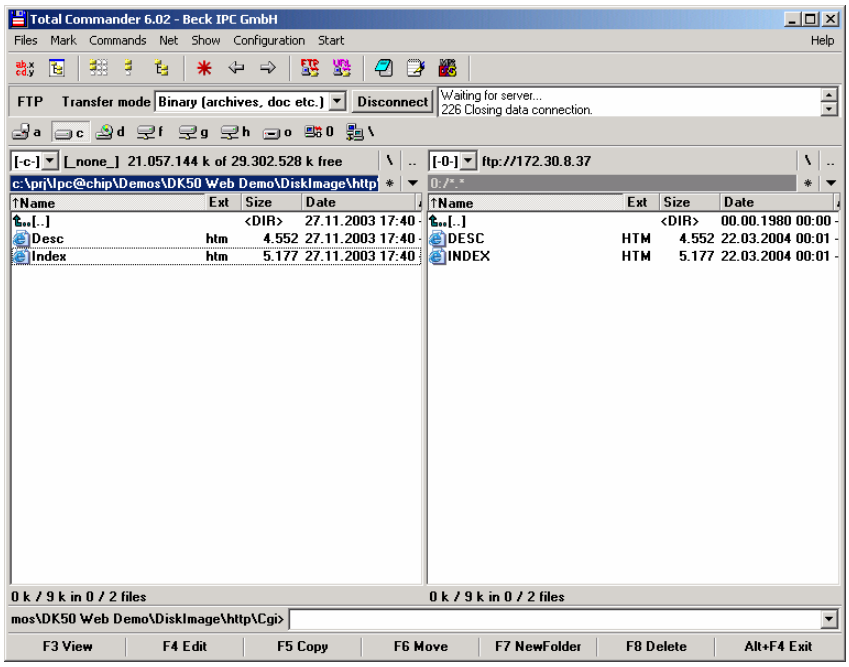
TotalCommander must then be used as the FTP client in order to transfer the files to the CompactFlash card. After the FTP connection has been established, you must switch the current drive on the IPC@CHIP from A: to B:. To do this enter the command "cd b:" in the command line shown below and confirm with <ENTER>.



The view must then be refreshed by choosing the “Reread Source” command in the “Show" menu.



The drive content of the CompactFlash card will then be displayed in the right-hand pane and you can also copy files to it.



Remember the following:
Files must be in 8.3 format.

You can enter the commands "B: <ENTER>" and "DIR <ENTER>" in the HyperTerminal program in order to view the content of the Compact Flash card:

```
A:\>b:

B:\>dir

DESC      HTM      4552  03-22-04   00:01
INDEX     HTM      5177  03-22-04   00:01

2 files use 9729 bytes
63922176 bytes free
```

© 2005 Beck IPC GmbH

2.3 DK50 Development Board Start-up (SC11)

The DK51 Development Kit can also be used in conjunction with an SC11. The SC11 is the same as the SC13 apart from the fact that it does not have an Ethernet network interface. It is therefore commissioned differently to the SC13 and is described here separately.

2.3.1 What do we need to get started?

- DK50 Development Board
- Embedded Web Controller SC11 (on the DK50)
- Power supply unit PSE10
- Null-modem cable ZK14

- A PC running Windows 95/98/NT/2000/XP
- Serial port on your PC
- Windows HyperTerminal program
- Optional: Compact Flash Card, if you want to configure the CF Interface (not needed for start-up)

- Zip-files from the Beck Getting Started web site (http://www.beck-ipc.com/ipc/gettingstarted):
  - FTP client TotalCommander as shareware
  - CHIPTOOL Software
  - Application Visualisation Demo for DK50/DK51
  - IDE Driver package (Containing EXTIDE.EXE)

The Getting Started web site contains all the tools and programs required to put the Development Kit into operation. The FTP client is a shareware program. For more details please read the online documentation of this tool. You can use any FTP program which provides file transfer support.

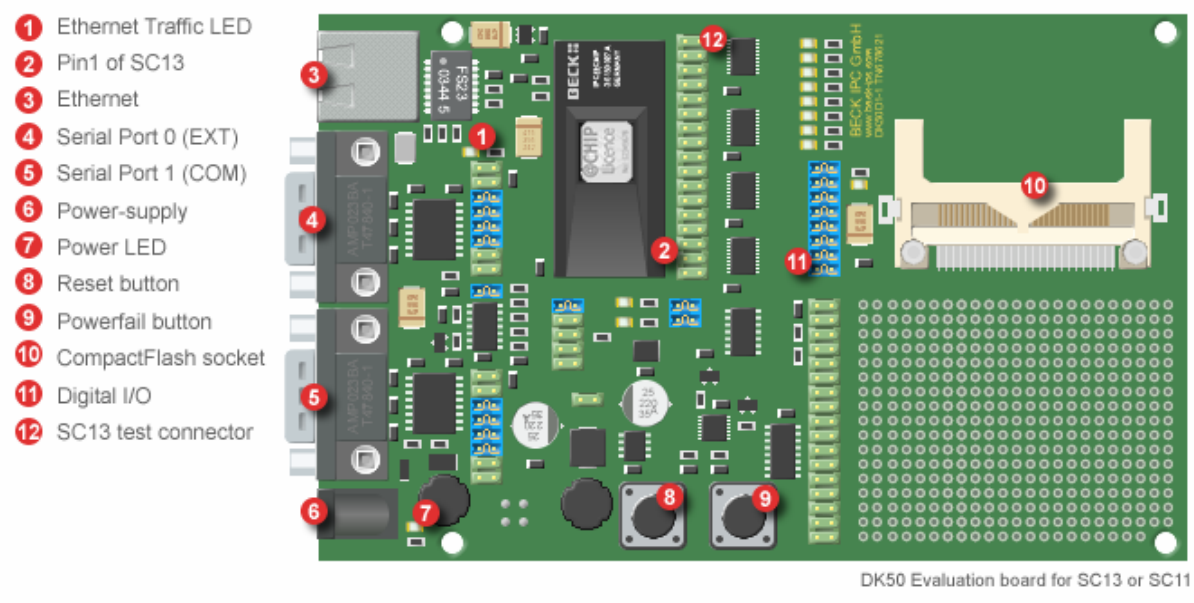


Be aware, that the Embedded Controller SC13 and DK50 Development Board are Electrostatic Sensitive Devices. Observe general precautions for handling.

© 2005 Beck IPC GmbH

2.3.2 Step 1 – Switching on the DK50 Development Board

The DK50 Development Board is an evaluation platform for the Embedded Controller SC13 and SC11. Among other functions, it provides two serial ports, an Ethernet interface, eight digital I/O, a 'traffic' LED, CompactFlash Socket and a switching DC/DC converter.



DK50 Evaluation board for SC13 or SC11

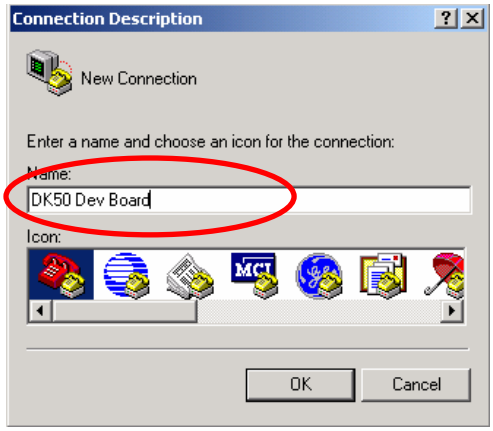
The AC power supply (PSE10) is delivered with country specific AC adapters. Please choose the right adapter for your country. Then connect the AC power supply to your main supply and to the DK50 Development Board. The Power LED (7) will light up and Traffic LED will light up for less than half a second.

© 2005 Beck IPC GmbH

2.3.3 Step 2 – Setting up the serial communication

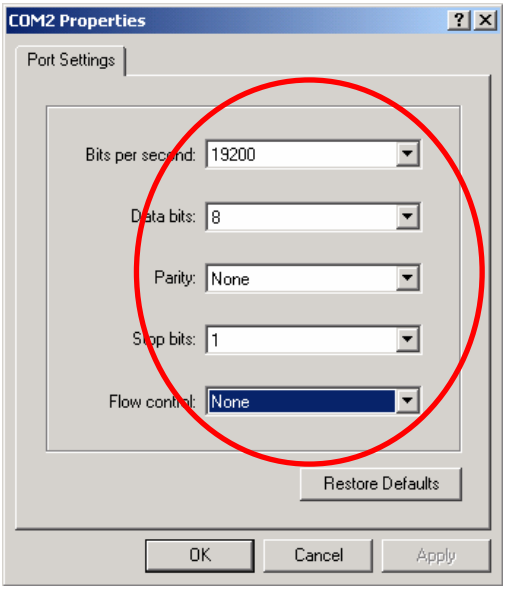
Please connect the null modem cable to your PC's COM port and to the COM port of the DK50 (5).

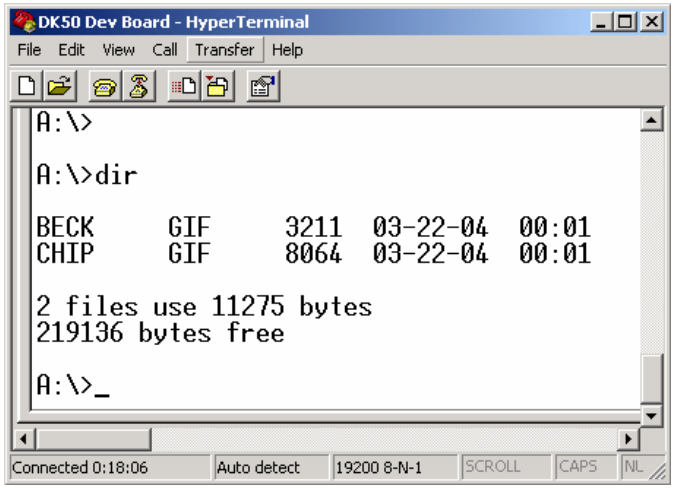
Next, you need the Windows HyperTerminal program. If it's not part of your current Windows installation, you should install the application from your Windows CD.
Start the HyperTerminal program. Enter a name for the new connection (e.g. DK50 Dev Board). Choose the connection via the COM port of your PC.





Set the communication parameters to 19200 baud, 8 data bits, no parity, one stop bit, no handshake. The HyperTerminal program now connects the PC with the DK50. Press Enter when the terminal window is opened to see the 'A:>' prompt for the drive A: on the Embedded Controller. Enter the command 'dir' and you will see the root directory of the Embedded Controller.





The Embedded Controller is factory shipped with the two graphics files "BECK.GIF" and "CHIP.GIF" on the drive.

© 2005 Beck IPC GmbH

2.3.4 Step 3 – Setting up the PPP communication

The following chapter describes how to set up communication via PPP using Windows 2000[8]. The SC11 does not have an Ethernet network interface. However, TCP/IP can be implemented with PPP, so that it is still possible to use the features of the IPC@CHIP such as FTP and Telnet.
In the following description, it is assumed that the SC11 functions as a PPP server and is connected to your PC via the null-modem cable, i.e. your PC must establish a connection to the passive SC11.
The PPP server of the IPC@CHIP must be configured in the CHIP.INI configuration file so that it can be used. You can create this file on your PC and then transfer it to the SC11.
Open any text editor (e.g. Notepad) and enter the following text[9].

```
[PPPSERVER]
ENABLE=1
COMPORT=EXT
ADDRESS=192.168.200.1
REMOTEADDRESS=192.168.200.2
NETMASK=255.255.255.0
IDLETIME=0
FLOWCTRL=2
MODEM=1
AUTH=2
BAUD=115200
USER1=ppps
PASSWORD1=ppps

CONNECTMSG0=CLIENT
CONNECTANSWER0=CLIENTSERVER
CONNECTMSG1=NULL
CONNECTANSWER1=NULL
CONNECTTIMEOUT0=0
CMDMODE=NULL

[SERIAL]
EXT_RECVQUEUE=4096
EXT_SENDQUEUE=4096
```

Only the most important entries are explained as the exact meaning of each entry in the configuration file does not have to be understood for commissioning:

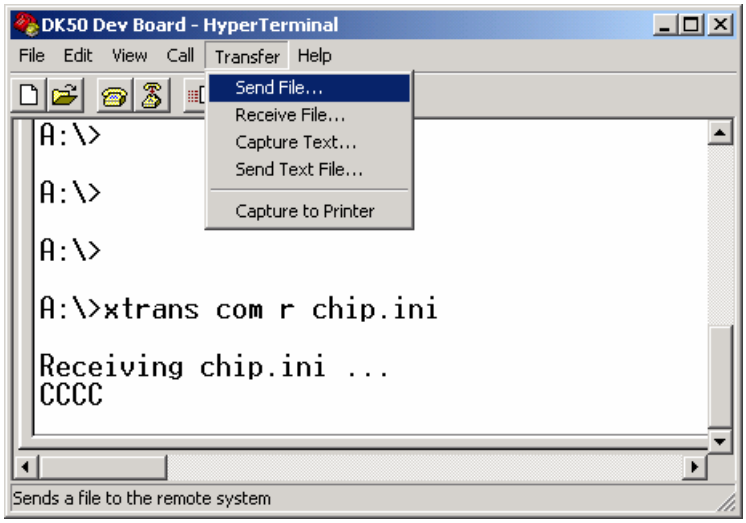
- ENABLE=1 activates the PPP server.
- COMPORT=EXT specifies that the EXT interface of the IPC@CHIP is to be used for the connection.
- ADDRESS=192.168.200.1 defines the IP address of the IPC@CHIP for the PPP connection. This IP address can be used to access the IPC@CHIP after the configuration is completed. Please note it down.
- REMOTEADDRESS=192.168.200.1 defines the address that is to be assigned to your PC for the PPP connection. If you have installed several components on your PC that use TCP/IP, you should ensure that the IP addresses defined here are not in the same subnet as those of other components.
- USER1=ppps and PASSWORD1=ppps define the access data with which the client (your PC) must later log on to the server. You can also define other user name password combinations. (e.g. USER2=ppps2 and PASSWORD2=ppps2)

[8] The procedure described is slightly different for other Windows versions.

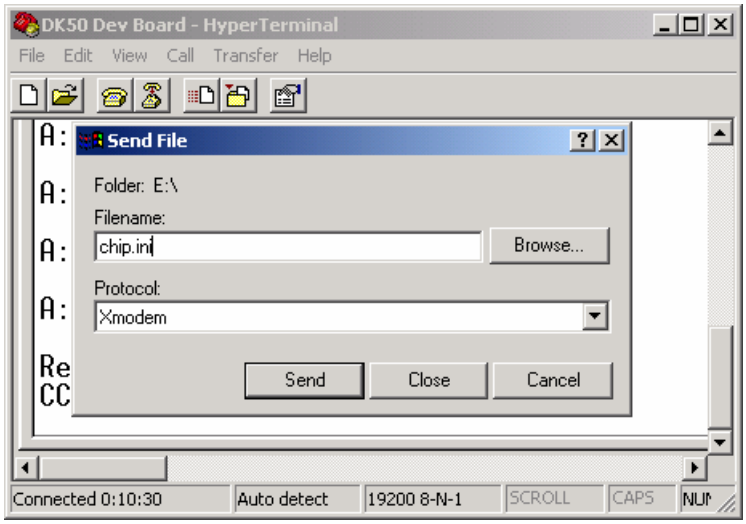
[9] When using other Windows versions, a slightly different form of the chip.ini configuration file shown here must be used. You can download configuration files for different versions from the Beck Getting Started web site (http://www.beck-ipc.com/ipc/gettingstarted).

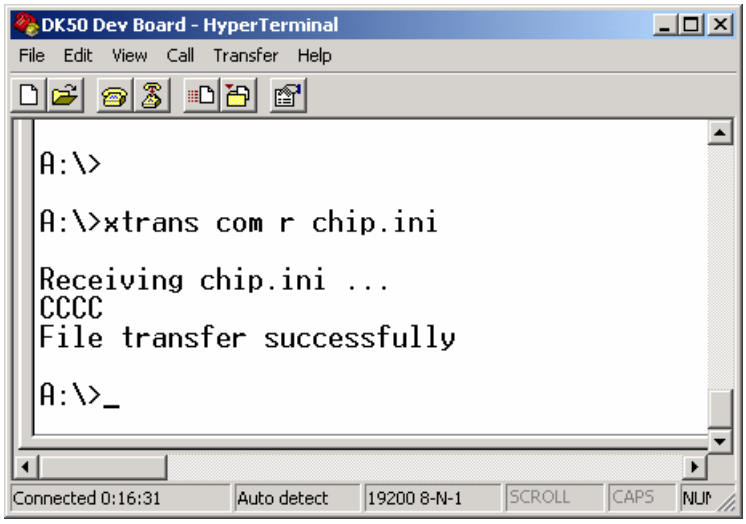
© 2005 Beck IPC GmbH

Save the file on your PC. The serial connection established in the previous section is used for transferring the file. In the HyperTerminal window, enter the following command:
`xtrans com r CHIP.INI`
It initiates the data transfer using the Xmodem protocol. Choose Transfer -> Send File... in the main menu.



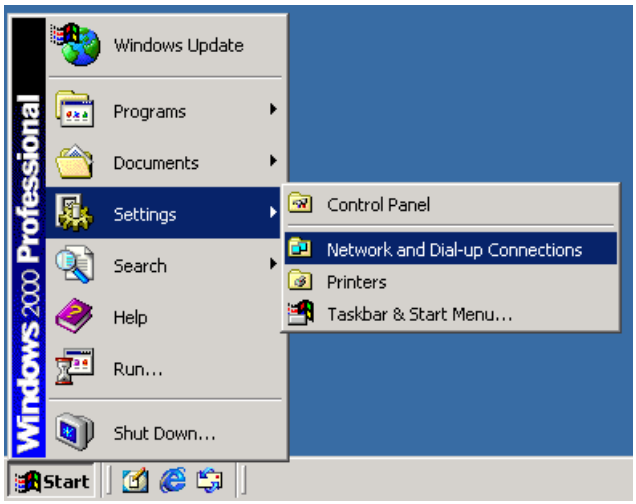
In the dialog that appears, enter under Filename the path of the newly created file and choose Xmodem under Protocol. Start the transfer by clicking the Send button. (Attention: after the xtrans command is entered, you only have a minute to start the transfer! Otherwise you will have to reenter the xtrans command.)



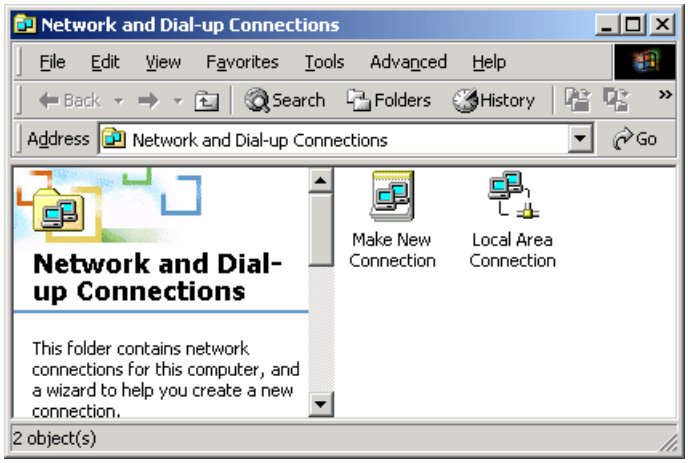


If you reboot the SC11, for example by entering the “reboot” command, the `PPP server` line will appear under the boot messages. This indicates that the PPP server is now active and is waiting for incoming connection requests. Now close the HyperTerminal program so that the serial interface used is available.

To establish a connection to the PPP server of the IPC@CHIP, you still have to set up a connection on your PC. To do this choose Settings -> Network and Dial-up Connections in the Start menu.



This will open the Network and Dial-up Connections window. Click the Make New Connection icon.

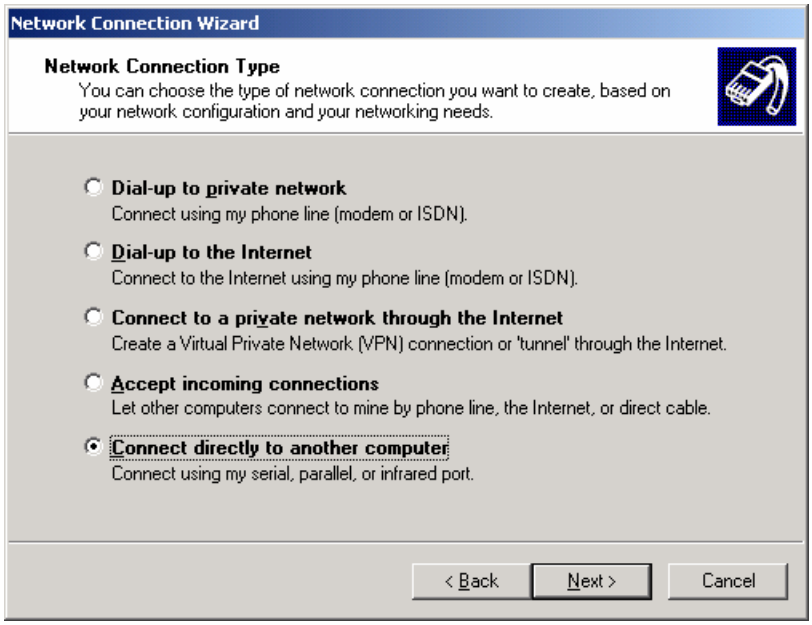


© 2005 Beck IPC GmbH

This will start the Network Connection Wizard which will help you to create the new connection. Click Next in the Welcome dialog of the wizard.



In the Network Connection Type dialog select the Connect directly to another computer option. Click Next to continue.

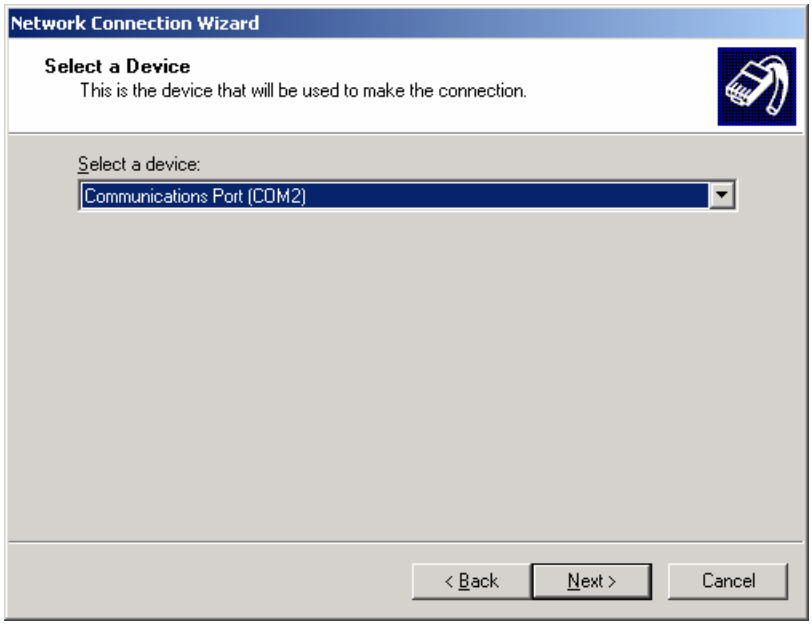


© 2005 Beck IPC GmbH

The Guest option must be selected in the Host or Guest dialog as your PC is to function as a client within the communication connection with the IPC@CHIP via PPP. Click Next to continue.

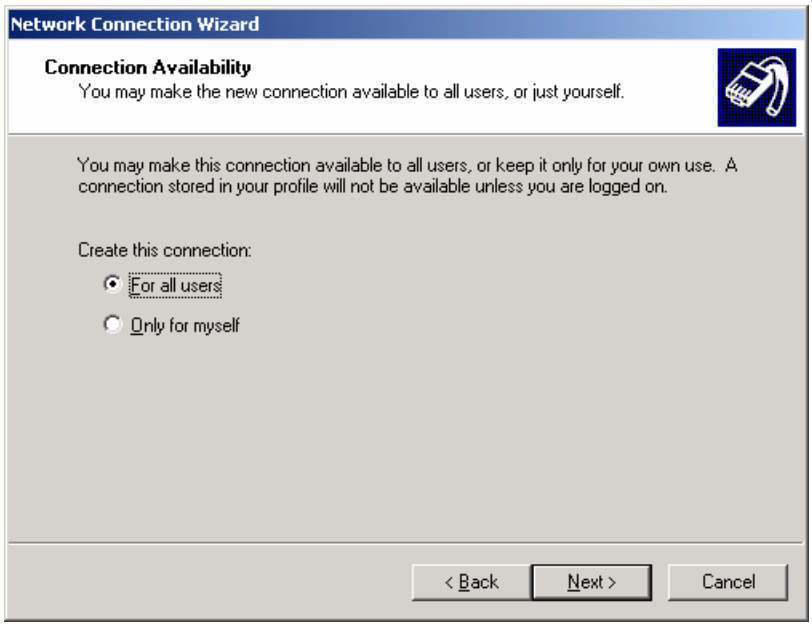


In the Select a Device dialog you can select the connection by means of which the IPC@CHIP is connected with your PC. Select here the serial interface to which the null-model cable is connected.



© 2005 Beck IPC GmbH

In the subsequent Connection Availability dialog, you can define whether the connection is only available to you or all users of the computer. Select the required the option and click Next to continue.

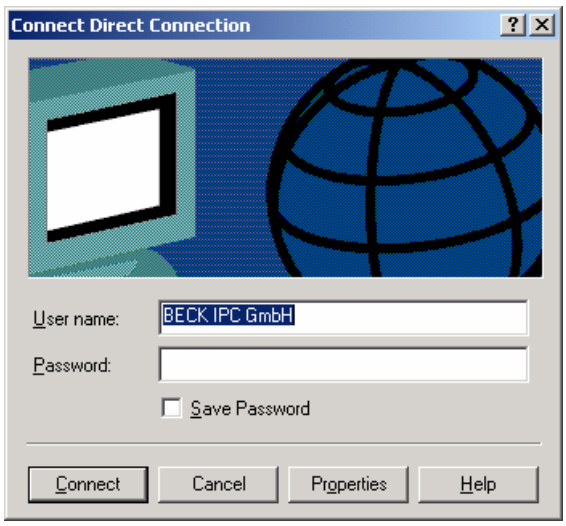


In the last dialog of the wizard you can give the connection a name. Click the Finish button to close the Network Connection Wizard.

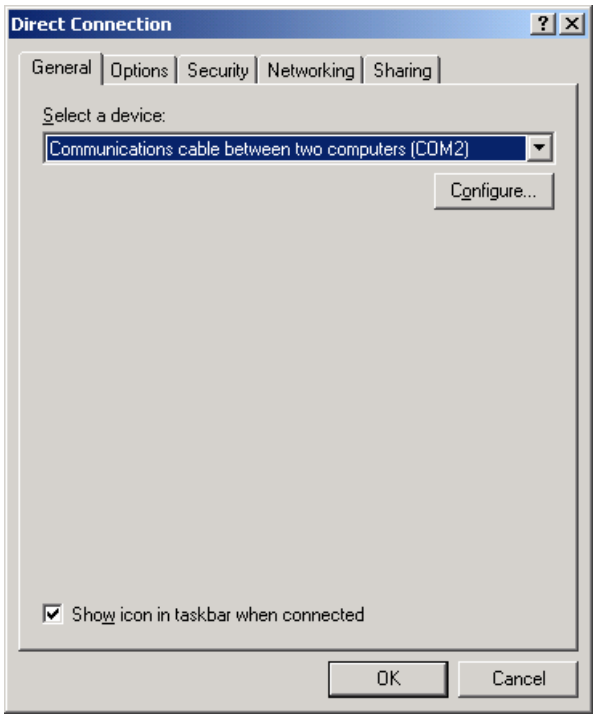


© 2005 Beck IPC GmbH

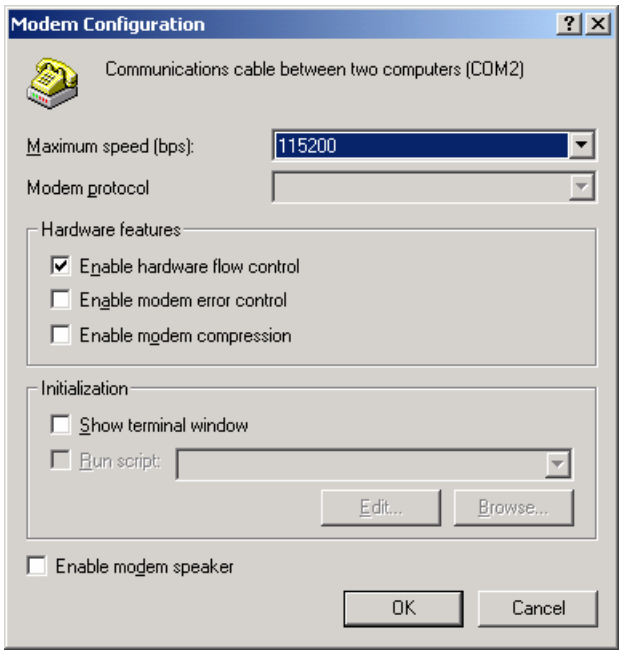
The Connect... dialog will now appear. Before, however, a connection can be established, it is still necessary to modify some properties that the Network Connection Wizard has not queried.
To do this, click Properties.



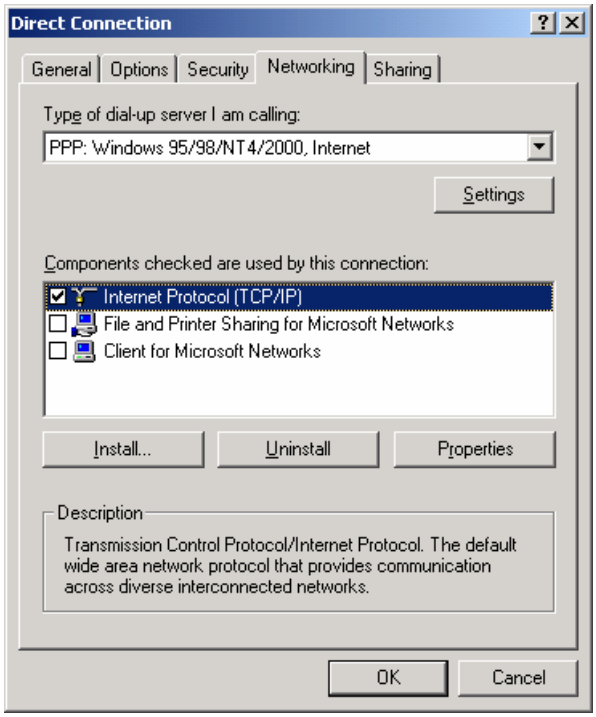
The maximum transmission speed of the serial interface first has to be increased. Click the Configure... button to open the Modem Configuration dialog.



At Maximum speed (bps) set the value 115200 and click OK in order to return to the previous dialog.

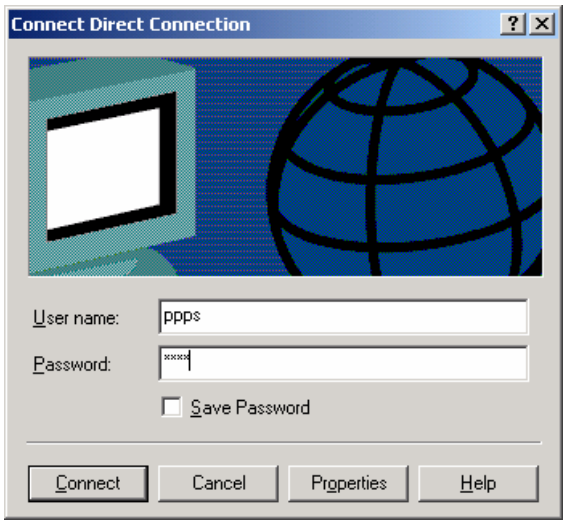


Then select the Networking tab. Deactivate here all components apart from Internet Protocol (TCP/IP). Click OK to close the dialog.



© 2005 Beck IPC GmbH

The Connect... dialog will now be reopened. A connection to the PPP server of the IPC@CHIP can now be established as all connection properties and the interface used are correctly configured.
First connect the serial interface of your PC to the EXT interface of the IPC@CHIP via the null-modem cable.
Now enter the user name password combination that you have defined in the CHIP.INI file (“ppps” and “ppps”). Click Connect to establish the connection.



After a short time a message will appear to indicate that connection has been established successfully.



The IPC@CHIP can now be accessed via the IP address configured above. All the TCP/IP based services of the IPC@CHIP can now be used via the PPP connection in exactly the same way as if an Ethernet connection was available.

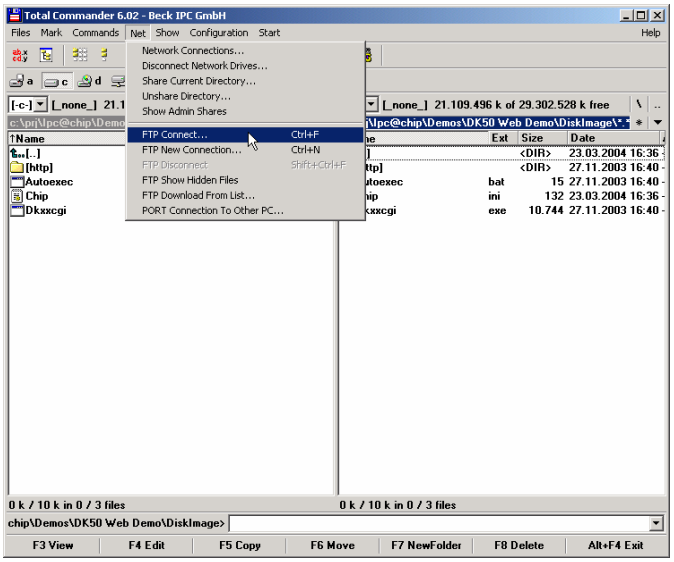
© 2005 Beck IPC GmbH

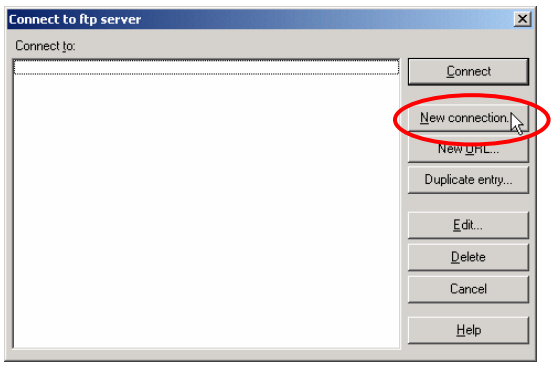
2.3.5 Step 4 – Downloading the DK50 Online demo program

The following describes how to download files to the IPC@CHIP and put the DK50 online demo into operation on your DK50.

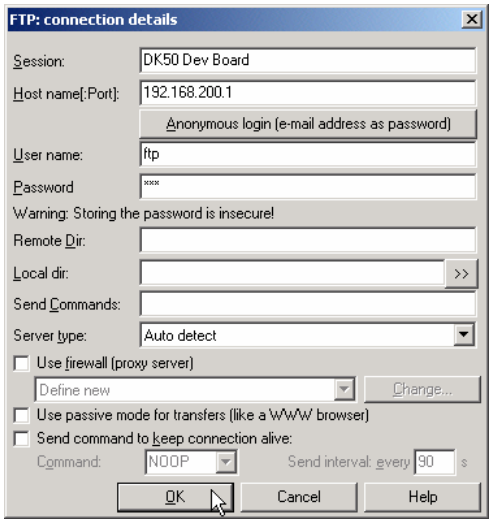
The RTOS of the IPC@CHIP features an FTP server for transferring files. FTP stands for File Transfer Protocol and has become a standard for transferring files over TCP/IP networks. All you need to transfer a file from or to the IPC@CHIP is an FTP client. Your Windows system only provides an FTP client with a command line interface. Clearly, a graphical user interface, such as with the WS-FTP or TotalCommander, is much more user-friendly.
The following explains the data transfer procedure using the TotalCommander shareware program.

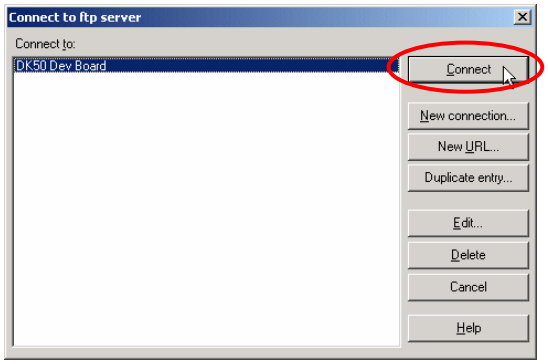
Install and start TotalCommander. In the left-hand pane select the path where the DK50 online demo files are located. Switch to the right-hand pane and choose FTP Connect from the Net menu. Click “New connection” in the dialog that opens.





Under "Session" enter a name for the connection, such as “DK50 Dev Board”. The IP address of the IPC@CHIP that you should have noted down before must be specified in the “Host name” field. To connect to the FTP server, you need also a user name and a password. For the IPC@CHIP you use ‘ftp’ as user name and also ‘ftp’ as password. These are default settings that can be changed at a later time.
Click OK to confirm and close the dialog. In the previous dialog select the new connection and click “Connect".





© 2005 Beck IPC GmbH

You should now be connected to the FTP server of the IPC@CHIP and the drive content of the IPC@CHIP should now be visible in the right-hand pane of TotalCommander. Note that the FTP connection will timeout if you don't issue any actions for some time. You will then have to reconnect. The procedure now depends on whether you have configured the IP address of the IPC@CHIP by hand or with DHCP.

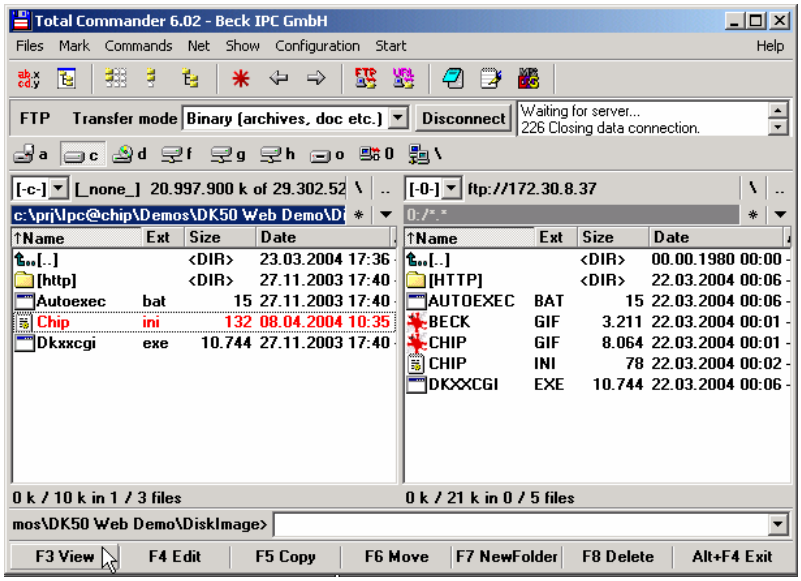
2.3.5.1 IP address was configured using DHCP

Select all files and subfolders in the path of the DK50 Online Demo. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.

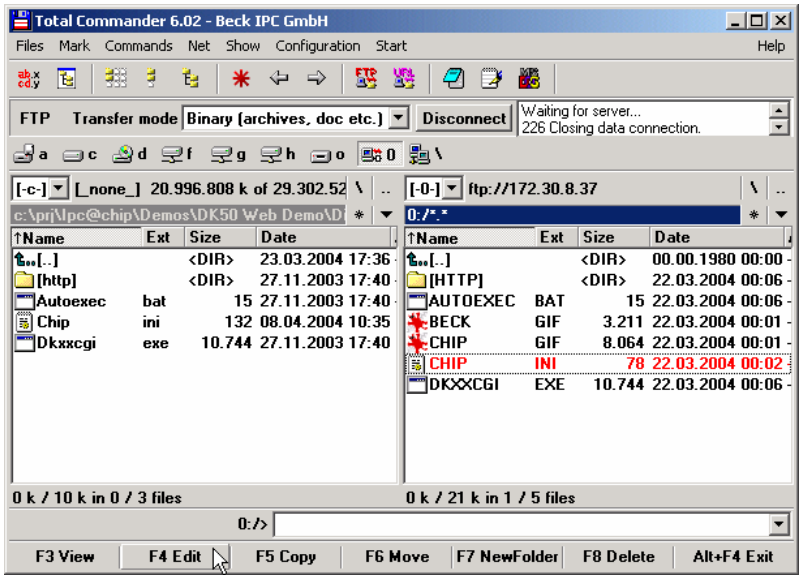
The DK50 must be restarted in order to activate the demo. To do this, press the Reset button.

2.3.5.2 IP address was configured by hand

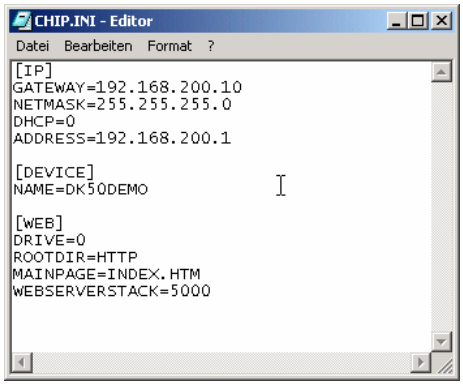
Select all files and subfolders in the path of the DK50 Online Demo apart from the CHIP.INI file. This will ensure that you do not overwrite the IP configuration you have entered beforehand. Click the "F5 Copy" button to copy the files to drive A: of the IPC@CHIP.
Now select the CHIP.INI file on the PC and click the "F3 View" button to view the content of the file. Select all the text apart from the section [IP] and copy it to the Clipboard with CTRL+C. Then close the window.



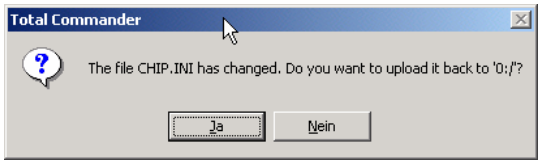
Now choose the CHIP.INI file on the IPC@CHIP and click “F4 Edit” to edit the file.



Paste the content of the Clipboard into the file after the IP section with CTRL+V.



Save the file and close the editor. In the dialog prompt that follows, you must then confirm that you wish to upload the modified CHIP.INI file onto the IPC@CHIP. You then have to confirm that you wish to overwrite the existing file.





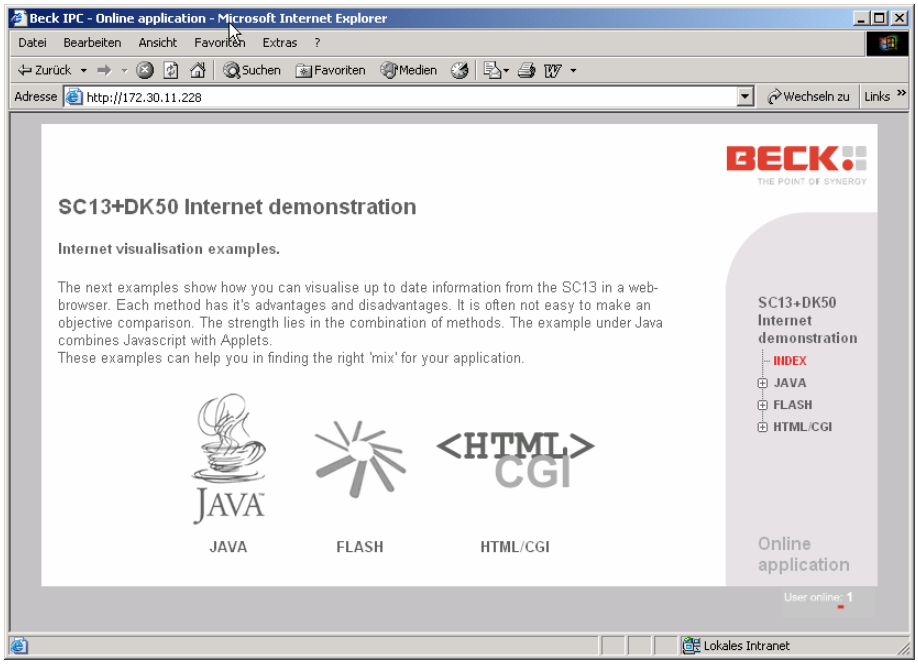
You can close the FTP connection by clicking the “Disconnect” button once all files have been transferred.

The DK50 must be restarted in order to activate the demo. To do this, press the Reset button.

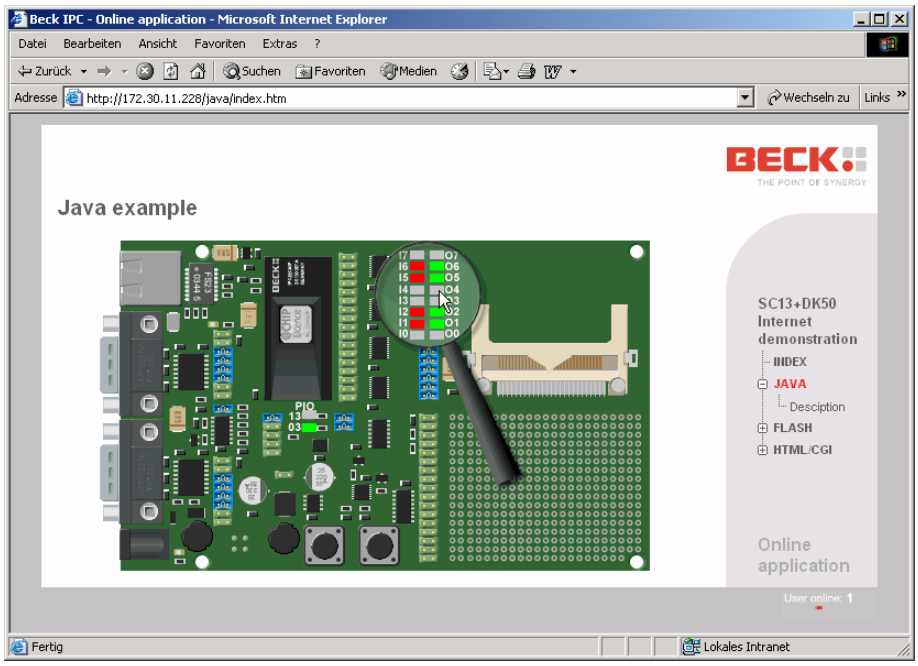
© 2005 Beck IPC GmbH

2.3.6 Step 5 – Operating the Demo program

Start your Internet browser and enter the IP address of your IPC@CHIP (the one that you have noted down) in the address bar. The start page of the DK50 Online Demo will appear. Please choose the Java example from the start page. The security settings of your browser must allow the use of Java applets.



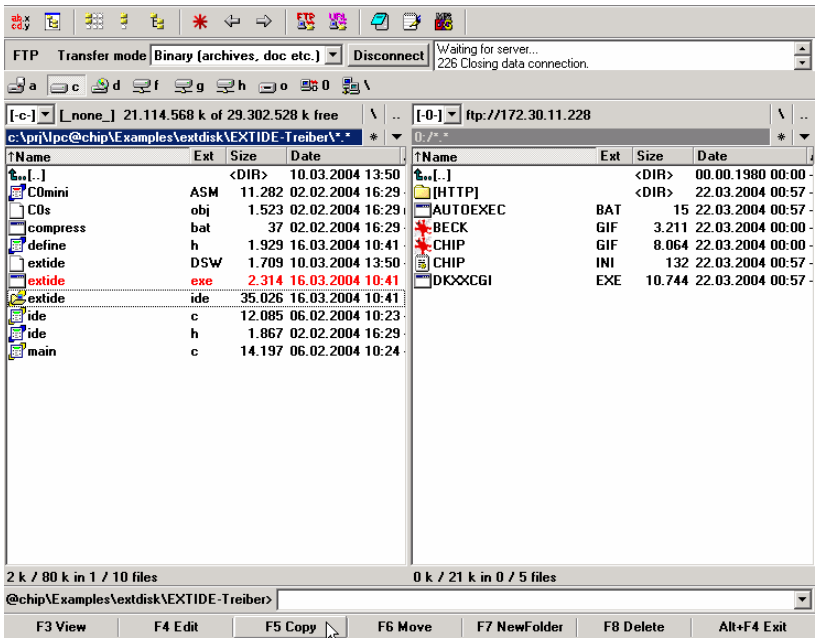
A Java applet is then loaded that visually represents the digital I/Os of the DK50. Click the fields marked O7 to O0 in order to switch the outputs of the DK50 on and off. The LEDs on PIOs 13 and 3 can also be switched on and off in this way.



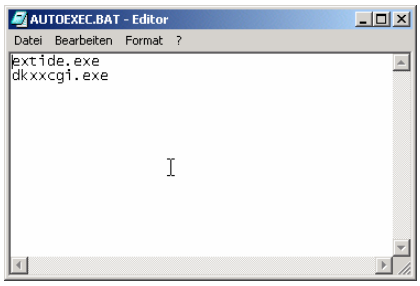
2.3.7 Step 6 – Configuring the CF Card

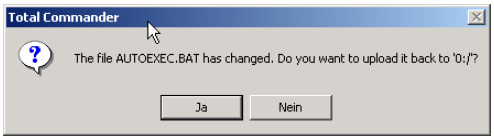
In order to put a CompactFlash card in DK50 into operation, you need the CompactFlash driver "EXTIDE.EXE" from the IDE Driver package on drive A: of the IPC@CHIP. First disconnect the DK50 from the power supply. Insert the CompactFlash card into the DK50. Reconnect the DK50 to the power supply.

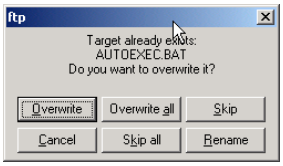
Copy the "EXTIDE.EXE" driver onto the drive of the IPC@CHIP using TotalCommander.



Now edit the Autoexec.bat file on the IPC@CHIP by selecting the file and clicking the "F4 Edit" button. Add the entry “exitde.exe” in the first line of the Autoexec.bat file. Save the file and close the editor. In the dialog prompt that follows, you must then confirm that you wish to upload the modified Autoexec.bat file onto the IPC@CHIP. You then have to confirm that you wish to overwrite the existing file.







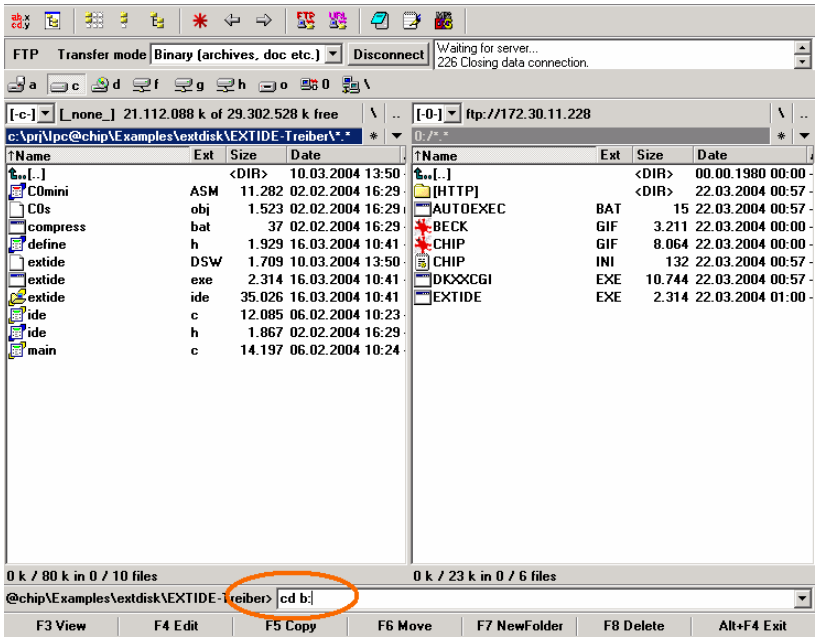
Close the FTP connection once more by clicking the "Disconnect" button and restart the DK50 via the Reset button.

The following message should then appear in the HyperTerminal program during the startup. This confirms that the CompactFlash has been installed correctly as drive B:.

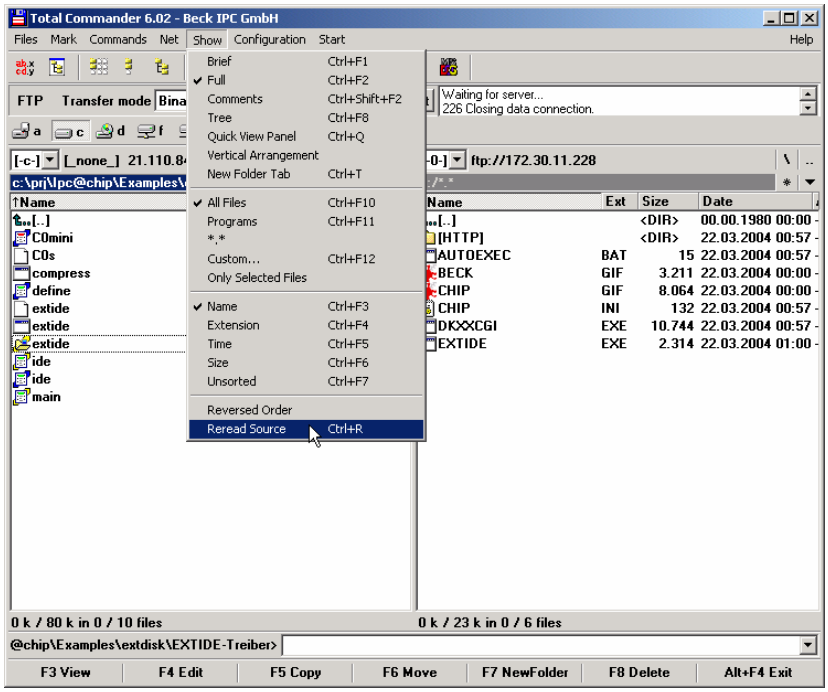
```
External IDE disk driver V1.14 Interface Version (0x0001)
Mounting partition: 1
Drive B: installed
```

© 2005 Beck IPC GmbH

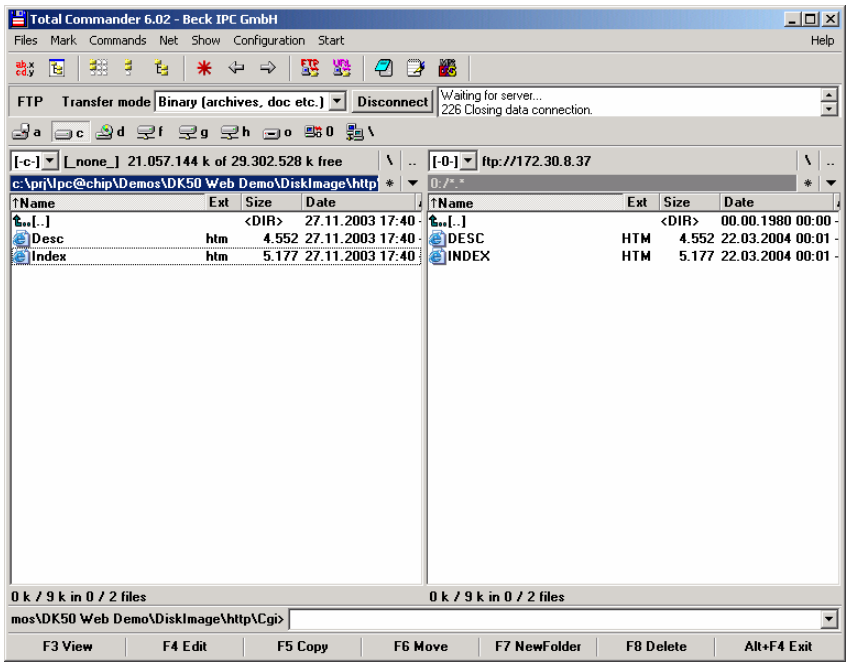
TotalCommander must then be used as the FTP client in order to transfer the files to the CompactFlash card. After the FTP connection has been established, you must switch the current drive on the IPC@CHIP from A: to B:. To do this enter the command "cd b:" in the command line shown below and confirm with <ENTER>.



The view must then be refreshed by choosing the “Reread Source” command in the “Show" menu.



The drive content of the CompactFlash card will then be displayed in the right-hand pane and you can also copy files to it.



Remember the following:
Files must be in 8.3 format.

You can enter the commands "B: <ENTER>" and "DIR <ENTER>" in the HyperTerminal program in order to view the content of the Compact Flash card:

```
A:\>b:

B:\>dir

DESC      HTM      4552  03-22-04   00:01
INDEX     HTM      5177  03-22-04   00:01

2 files use 9729 bytes
63922176 bytes free
```

© 2005 Beck IPC GmbH

3 Information For Advanced Users

3.1 Manual IP Configuration via the Terminal

To change the IP configuration of the Ethernet device interface, we can use the commands ‘ip’, ‘netmask’, ‘gateway’ and ‘dhcp’, each followed by a blank and the new entry. For example:

```
A:\>
A:\>ip 192.168.200.1
A:\>netmask 255.255.255.0
A:\>dhcp 0
A:\>ipeth
A:\>
```

In this example, we defined an IP address, a network mask and the DHCP was disabled. Please note that the gateway is not needed at this stage. When an application or service communicates to another network device that can only be reached via a specific ‘router’, you need to enter the address of the router as the ‘gateway’.

The IP configuration information is now stored in the file ‘CHIP.INI' on the flash disk. Now either cycle the power on the DK40/DK50 or use the ‘ipeth’ command to apply the new IP configuration.

© 2005 Beck IPC GmbH

3.2 IPC@CHIP @CHIP-RTOS

3.2.1 Updates

New @CHIP-RTOS versions for SC12 and SC13 are available for download via the Internet at the download area of http://www.beck-ipc.com/ipc . The downloaded ZIP file contains complete @CHIP-RTOS API[10] documentation in HTML format. Six different variant configurations of the @CHIP-RTOS are included to tailor the @CHIP-RTOS to your needs.

Important:
Use an SC12 @CHIP-RTOS version exclusively on an IPC@CHIP SC12.
Use an SC13 @CHIP-RTOS version exclusively on an IPC@CHIP SC13.
Don't mix up the targets!

The table below shows the different RTOS variants and their supported features.

	Tiny	Small	Medium	Medium PPP	Large	Large PPP
RTOS-Kernel	x	x	x	x	x	x
Serial port fossil driver	x	x	x	x	x	x
RTOS filesystem	x	x	x	x	x	x
External Disk Drive			x	x	x	x
XMODEM filetransfer	x	x	x	x	x	x
TCP/IP Ethernet driver		x	x	x	x	x
TCP/IP PPP client/server				x		x
Ethernet packet interface	x	x	x	x	x	x
TCP/IP protocol stack		x	x	x	x	x
I2C Interface	x	x	x	x	x	x
Software SPI	x	x	x	x	x	x
Hardware API	x	x	x	x	x	x
CFG server		x	x	x	x	x
Web Server					x	x
FTP Server			x	x	x	x
Telnet Server			x	x	x	x

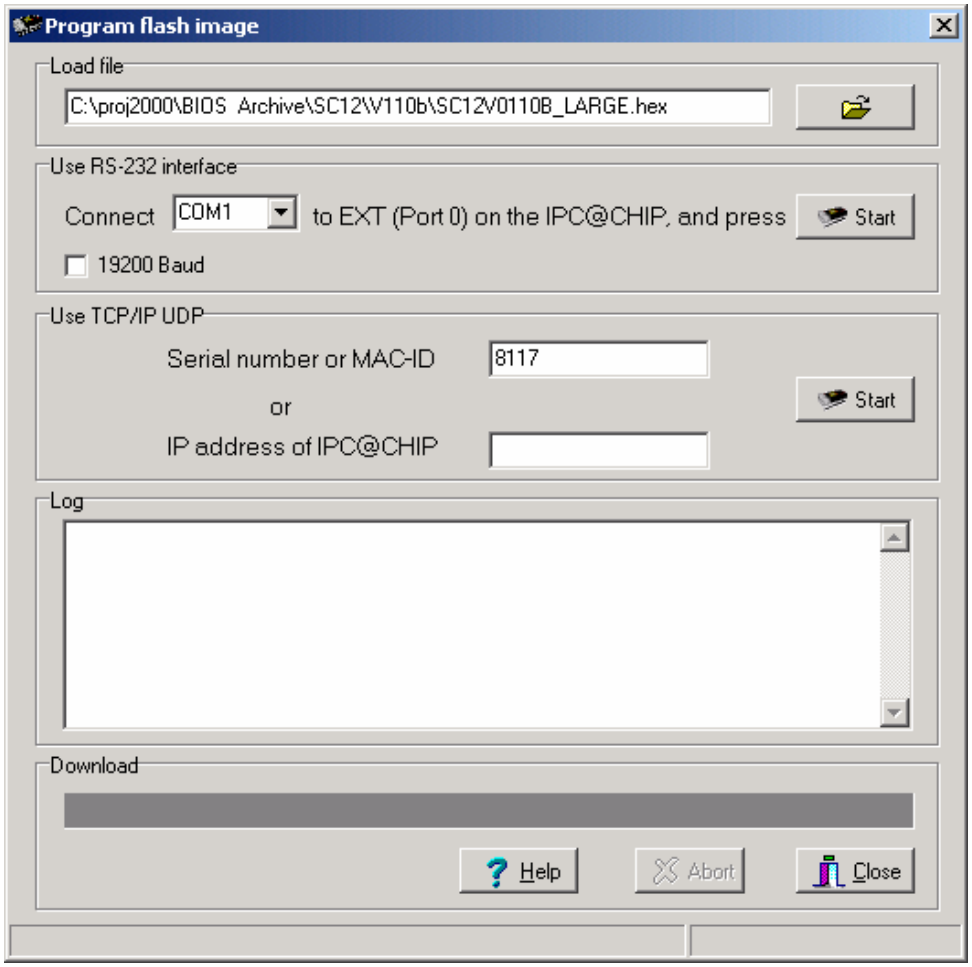
Please note:
Additional features of the @CHIP-RTOS (TFTP server, SNMP MIB support, Webfile upload,...) are not a part of the six official @CHIP-RTOS variants, but customers can ask for a special @CHIP-RTOS variant which includes the required features. Each @CHIP-RTOS variant with TCP/IP protocol stack includes a DHCP client.

[10] Application Programmer's Interface

© 2005 Beck IPC GmbH

3.2.2 Downloading RTOS updates

New @CHIP-RTOS versions can be easily downloaded to the IPC@CHIP using the CHIPTOOL. Select the @CHIP-RTOS | Program Flash menu option to get the following dialog. Here select the Intel-Hex image file that you want to load. The 'fileopen' button on the right of the Load file field allows you to browse for the desired Intel-Hex image file containing the new @CHIP-RTOS version.



You have two options for transferring a new @CHIP-RTOS into the IPC@CHIP. The transfer can be done either via the IPC@CHIP's EXT RS-232 interface (Port 0) or via the TCP/IP network path. The RS-232 interface relies only upon a valid Bootstrap loader being present in the IPC@CHIP which should always be the case. So when all else fails, use the EXT RS-232 port (Port 0) on the IPC@CHIP for @CHIP-RTOS download.

The RS-232 communication for downloading is handled by either an existing @CHIP-RTOS or the IPC@CHIP's Bootstrap loader. If for some reason your existing @CHIP-RTOS is not valid[11], you will need to cycle the power on the IPC@CHIP so that the CHIPTOOL can get the attention of the resident Bootstrap loader.

[11] The existing @CHIP-RTOS could be invalid due to, for example, an IPC@CHIP power loss during the @CHIP-RTOS Flash programming cycle.

© 2005 Beck IPC GmbH

For transfer via the TCP/IP network, the IPC@CHIP can be identified by either the serial number or IP address.[12]

Press the respective start button for the type of transfer you have selected to use. After the hex file transfer has been completed, the Flash programming cycle will require about 15 seconds during which 'P' characters can be seen coming out to the terminal. The terminal will indicate "Done" when the programming cycle has completed, as shown below. This will be followed by an IPC@CHIP reboot.

```
A:\>

Programming flash
PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
Done
```

At the CHIPTOOL display you may see some error messages if the CHIPTOOL is not able to re-establish communication following the @CHIP-RTOS download. This can often be the case because the flash disk needs to be reformatted due to a change in size of the @CHIP-RTOS and the consequent loss of set-up information (e.g. IP configuration). You will then need to repeat the IP configuration steps.

3.2.3 CHIP.INI configuration file

The @CHIP-RTOS holds its non-volatile configuration information in a CHIP.INI file found in the root directory of the A: Flash disk drive. This ASCII text file can be adjusted either directly with a text editor or with particular commands from the terminal. The figure below shows a few example entries that might be found in a CHIP.INI file.

```
[IP]
ADDRESS=192.168.200.1
NETMASK=255.255.255.0
GATEWAY=192.168.200.10
DHCP=0

[STDIO]
STDOUT=COM TELNET
FOCUSKEY=6

[SERIAL]
COM_BAUD=9600
```

Refer to the HTML @CHIP-RTOS API Documentation for a complete specification of the entries possible in this configuration file.

[12] Since *SC12@CHIP-RTOS version 1.10Beta* / *SC13@CHIP-RTOS version 0.91Beta*, it is also possible to enter the MAC-address of the internal Ethernet controller, instead of the serial number.

© 2005 Beck IPC GmbH

3.2.4 The command shell

The @CHIP-RTOS has a command shell task which accepts commands from a standard I/O device. This standard I/O device can be either or both of the RS-232 ports (COM and EXT) and/or the Telnet port. The selection is controllable from the CHIP.INI file[13]. The standard I/O can be configured to be shared between application programs and the @CHIP-RTOS command shell. A hot key (by default Ctrl+F) will switch the standard I/O focus between application and @CHIP-RTOS command shell.

For Telnet you use 'tel' as user name and also 'tel' as password. These are default settings that can be changed at a later time.

For a list of the commands available you can type in 'help' at the command line.

```
A:\>help
DIR          DEL          TYPE         COPY         REN
MD           CD           RD           FORMAT       CON
IW           OW           IB           OB           PCS
ALE          ADR          PIO          IP           NETMASK
GATEWAY      DHCP         WAIT         FTP          BATCHMODE
MEMOPT       XTRANS       PING         IPCFG        IPETH
TCPIPMEM     REBOOT       VER          MEM          HELP
ERRORS       TASKS        UTASKS       WEBSTAT      CGISTAT
CLOSETELNET
A:\>
```

Refer to the @CHIP-RTOS API Documentation for a detailed description of these commands.

3.2.5 The PPP Server/Client

PPP (the Point to Point Protocol) is a mechanism for creating and running TCP/IP over a serial link - be that a direct serial connection (using a null-modem cable), or a link made using an analog modem.

Other computers can dial into the IPC@CHIP PPP server and communicate via the TCP/IP link using FTP, Telnet, Web, etc. in the same manner as with an Ethernet TCP/IP link. One major difference between a PPP and an Ethernet connection is of course the speed. A standard Ethernet connection operates at 10 Mbs maximum theoretical throughput, whereas an analog modem operates at speeds up to 56 kbps.

PPP is strictly a peer to peer protocol; there is no difference between the machine that dials in and the machine that is dialled into. However, it is still useful to think in terms of servers and clients. When you dial into a site to establish a PPP connection, you are considered the client. The machine to which you connect (e.g. the IPC@CHIP) is considered the server. The @CHIP-RTOS operating system provides TCP/IP API functions for either side of this client/server relationship.

PPP at the IPC@CHIP includes the subprotocols LCP and NCP (IPCP). Supported authentication protocols are PAP and CHAP.

[13] For example, the figure shows the stdio configured for both the RS-232 COM (Port 1) *and* Telnet ports.

© 2005 Beck IPC GmbH

3.2.6 The HTTP Web Server

The @CHIP-RTOS includes a web server. This means that you can store so called ‘html’ pages on the device and retrieve (show) them using a web browser such as Netscape or the Microsoft Internet Explorer. These pages can include images, Java applets, sounds, etc. The web server includes a CGI interface. This allows you to define functions within your program that are called by the web server if certain page names are requested by the browser. Your program code then produces the contents of the page that is presented to the browser.

One CGI page is activated if you enter the IP address of the IPC@CHIP device, without specifying any page name. Assuming your IPC@CHIP has the IP address 192.168.200.2, this would be the address http://192.168.200.2 .

Remember:
As long as the IPC@CHIP is connected to your network, you only need to enter the IP address in your Browser and you are connected to the WEB@CHIP.
To find what IP address your IPC@CHIP is using, take a look at the main network scan window of the CHIPTOOL application or enter the ‘ipcfg’ command in the Terminal window.

On the page shown, you will find a second link to ‘Current chip configuration’. This link will produce a page showing you the current IP configuration. Possibly not very useful, but it shows that with CGI you can produce html pages with a changing content.

You can produce your own html pages and store them on the IPC@CHIP. In this case bear in mind the following:

- Use ‘short’ file names in the 8.3 format. (extension htm instead of html)
- Only a limited number of files can be stored on the root directory (32 with version 1.01). Use a subdirectory if you need more files.

Remember:
After creating your own Web pages and downloading them into the @CHIP you need to enter the IPC address followed by a "slash" and the name of the first page you want to open.

© 2005 Beck IPC GmbH

3.3 Writing Execution Programs for the IPC@CHIP

3.3.1 Introduction

The IPC@CHIP Embedded Web Controller can execute multiple DOS programs concurrently. These 16 bit DOS programs can be created with a variety of development tools. However, the instruction opcodes in these programs must be confined to the 80186 instruction set.

For C/C++ programming, Beck offers the Borland 5.02 development environment.

Important note for installation of Borland 5.02 on your Windows PC:
Installation of Borland 5.02 has some trouble with long Windows directory and filenames. As installation directory you must enter a directory name with 8 characters or less, e.g. C:\BC5.

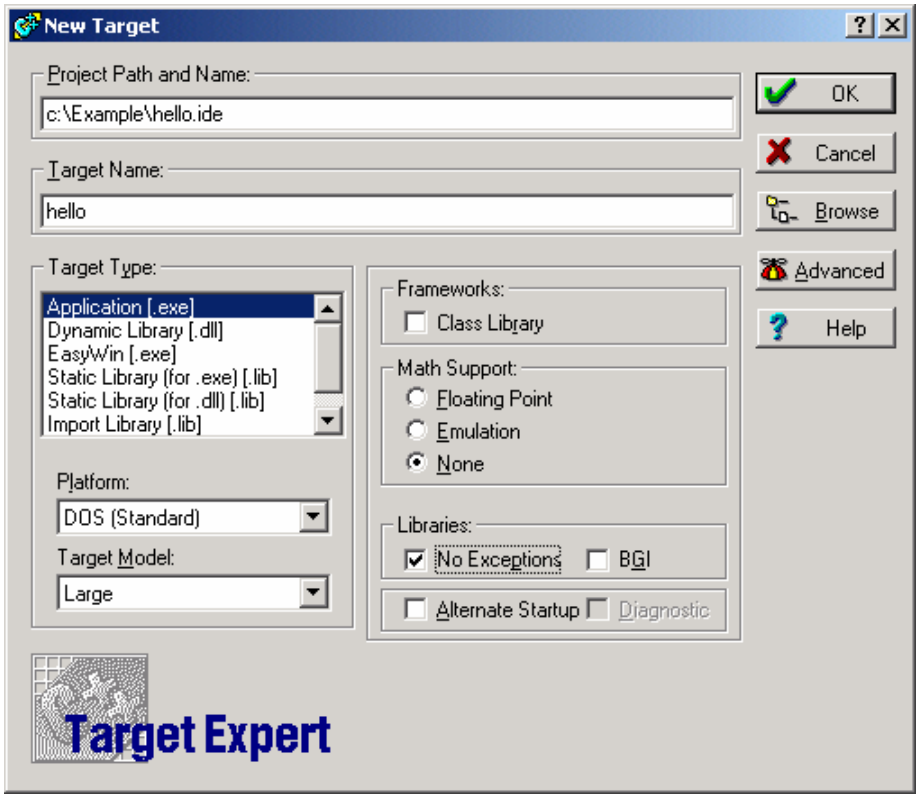
3.3.2 IPC@CHIP API documentation

HTML help files are available which fully document the IPC@CHIP API offered by Real Time Operating System @CHIP-RTOS.

3.3.3 IPC@CHIP example programs

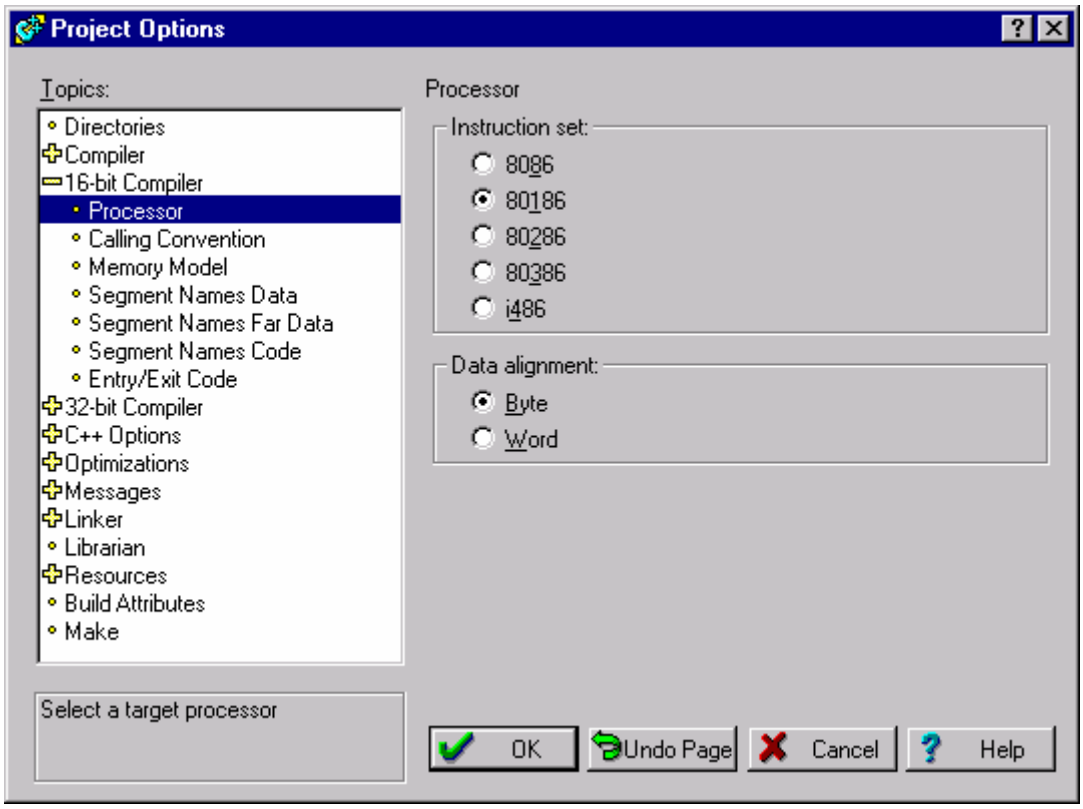
Many example DOS programs for the IPC@CHIP SC12/SC13 are available with C-source code from the download area at http://www.beck-ipc.com.

Here is an example of a simple “Hello World” program written for the IPC@CHIP using Borland 5.02. We start from the Borland’s “File | New | Project” menu selection which provides the following dialog interface.



It is important here to select the “DOS (Standard)” for the Platform. Select any Target Memory Model that is appropriate for your project. For inexperienced application programmers, we recommend the use of Model Large. Enable the “No Exceptions” check box; this will reduce the size of the resulting executable by at least 26 **KByte.**

The next important step is to select the processor type. This selection is made from the Options | Project menu option's dialog below.



If you forget to make this processor selection it is likely that your program will fail due to an invalid opcode exception!

Next we will write a short program shown below. This program includes examples of using the @CHIP-RTOS API functions, reached through @CHIP-RTOS interrupts. Refer to the @CHIP-RTOS API documentation for more details on these interfaces[14].

[14] Dozens of examples for programming with the @CHIP RTOS API interfaces are available for download via the Internet.

© 2005 Beck IPC GmbH

```
#include <dos.h>
#include <stdio.h>

#define RTOS_INTERRUPT     (0xAD)
#define @CHIP-RTOS_INTERRUPT_A0  (0xA0)
#define @CHIP-RTOS_INTERRUPT_10  (0x10)
#define SHELL_FOCUS        (1)
#define APPLICATION_FOCUS  (2)

static void Set_Stdio_Focus (int to_whom)
{
    _AL = to_whom ;
    _AH = 0x11 ;      // Service index
    asm { int @CHIP-RTOS_INTERRUPT_A0 }
}

static unsigned int Key_Pressed (void)
{
    _AH = 0x01 ;     // Service index
    asm { int @CHIP-RTOS_INTERRUPT_10 }
    return _AX;   //Register AX contains the result
}

static void Sleep (int how_long)
{
    _BX = how_long ;  // [ms]
    _AH = 0x0 ;       // Service index
    asm { int RTOS_INTERRUPT }
}

int main (void)
{
    Set_Stdio_Focus (APPLICATION_FOCUS) ;
    printf ("Hello World\n") ;
    do
    {
        Sleep (100) ;
    } while (!Key_Pressed()) ;
    printf ("Good Bye\n") ;
    Set_Stdio_Focus (SHELL_FOCUS) ;
    return 0 ;
}
```

The program first transfers the stdio focus to the application so that the printf output can be seen on the terminal. Use of the RTOS Sleep is a good practice for DOS programs which dwell for any significant amount of time, so that lower priority system tasks such as the Web server or FTP server can still operate[15]. This hello program spins in the loop until a key on the console is touched and then it exits with the “Good Bye” message.

```
A:\>hello
Hello World
Good Bye
```

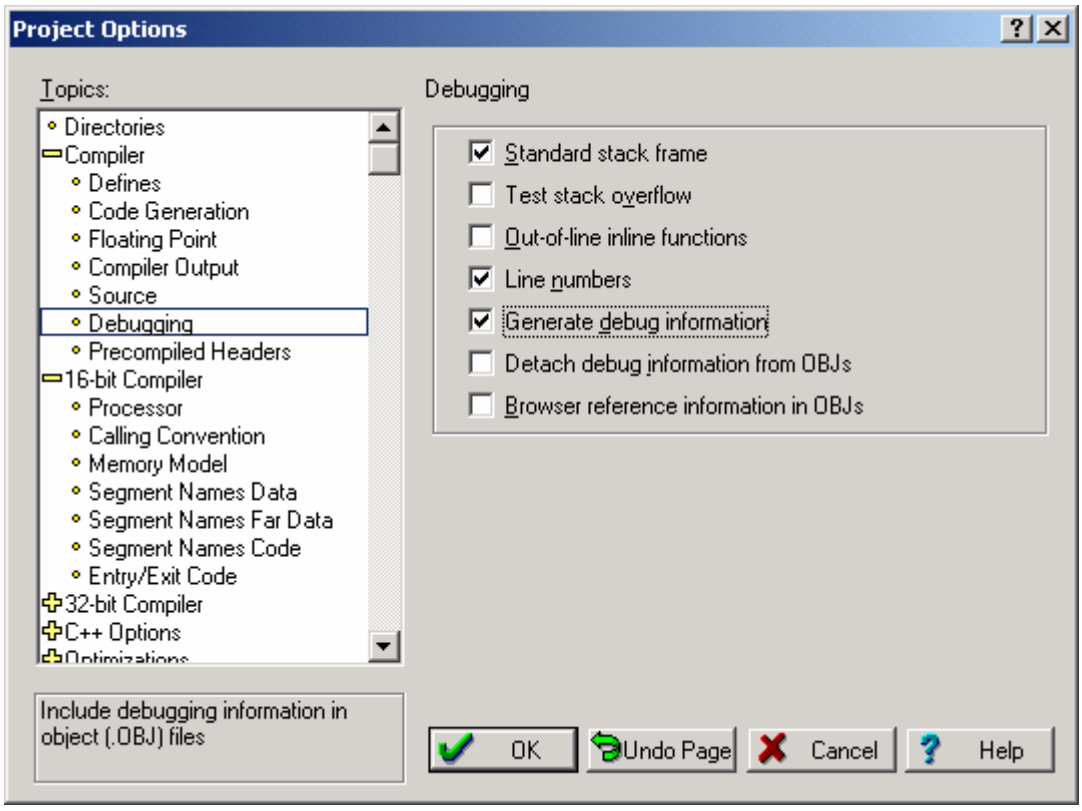
[15] This also allows multiple DOS programs to execute concurrently with a co-operative multitasking.

© 2005 Beck IPC GmbH

3.3.4 Program debugging

Debug@Chip is a Windows based source level debugger made for the IPC@CHIP. It operates with a PROBE.EXE program executing on the IPC@CHIP embedded side, which monitors and controls the operation of target programs. This probe communicates with Debug@Chip on the Window's PC over either an RS-232 or TCP/IP port, allowing breakpoint control and real-time monitoring of memory locations in the target system.

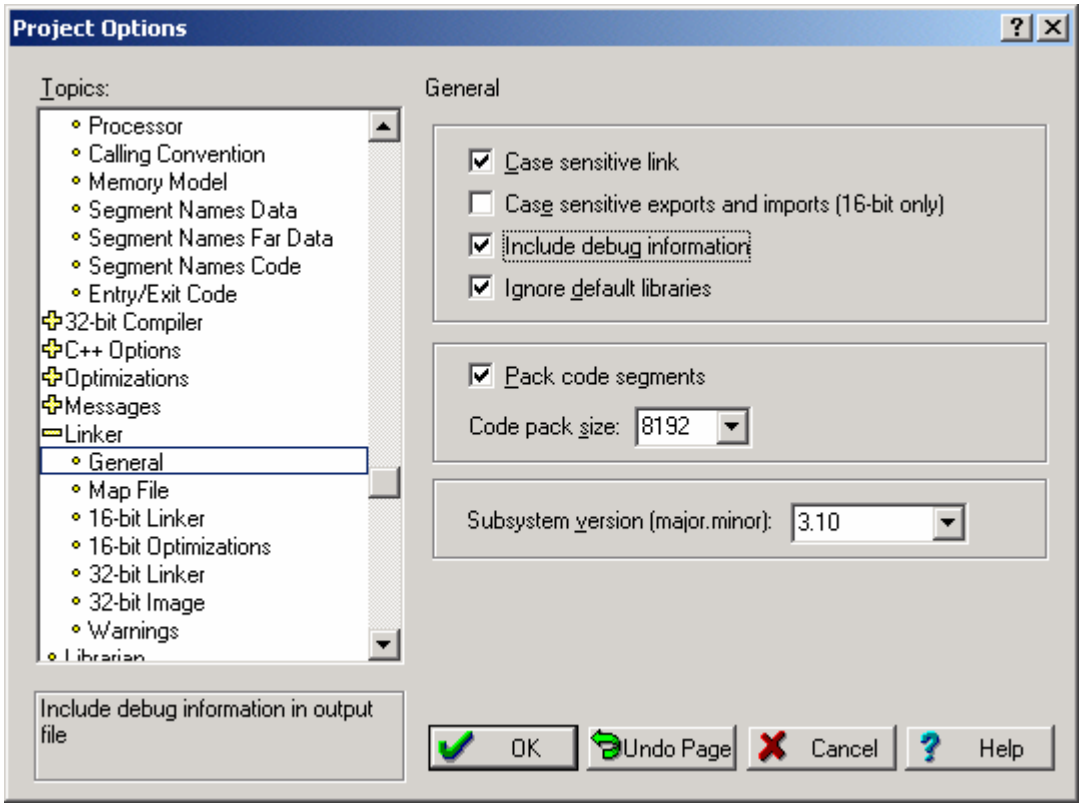
Debug@Chip requires Borland debug information in the executable files. Debug information is enabled from the Borland Options | Project dialog shown below. Ensure here that the "Line Numbers" and "Generate debug information" check boxes are selected as shown.



This selection instructs the compiler to place the debug information into the project's intermediate OBJ files.

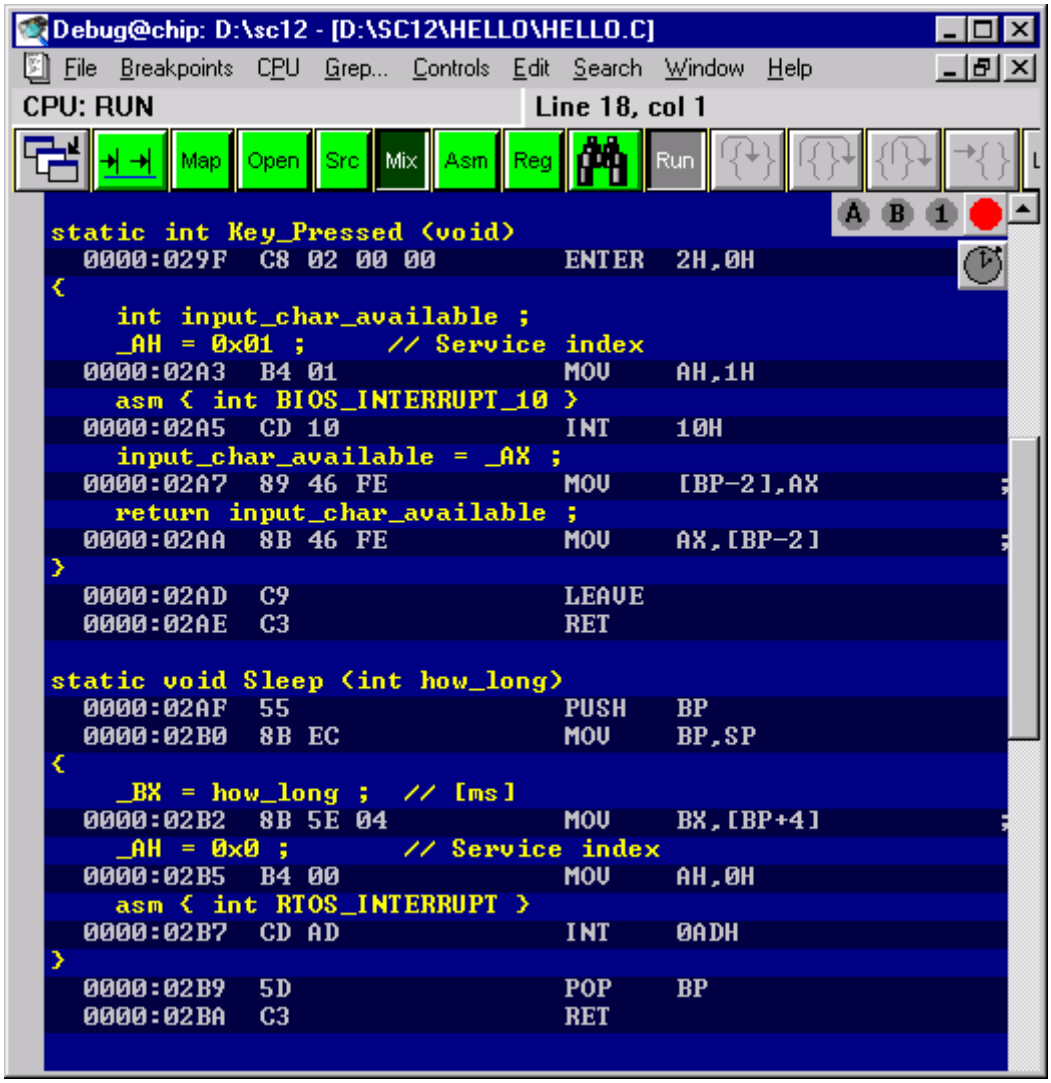
The next step we need then is to instruct the linker to pass this information along further to the resulting executable. This is done by checking the Linker option "Include debug information" shown in the following dialog.

© 2005 Beck IPC GmbH



Now with this debug information, the Debug@Chip tool is aware of the names and locations of your program's variables. Breakpoints can be set at source level. A further benefit is that you can see between the source lines to inspect the code produced by the compiler. For example, the API interface used in hello.c shown above was a bit daring. How can we be sure that the compiler is not using the registers that we are setting directly? (In general, there is no guarantee that the compiler will not immediately write over our register settings with a subsequent instruction.) Through the Debug@Chip's Mix window this code inspection is straightforward.

© 2005 Beck IPC GmbH



For example, in the Sleep procedure we can see that register BX is safely set with the 'how_long' value read from off the input parameter stack, with no subsequent writes to BX by the compiler prior to the interrupt API call.

An alternative to inspecting code is to use the compiler's int86 or int86x procedures. While these procedures are certainly safe, they are costly in terms of execution time. For real-time programs you may want to do some code inspecting.

For more detailed information, see the documentation of DEBUG@CHIP. You will find this documentation after installation at the subdirectory ‘Doc’ of the Debugger’s installation path.

3.3.5 Stripping debug information from executables prior to download

The Borland debug info can require a lot of space and serves no purpose in the target system. It is enough to have a copy of the executable with debug info on the Windows PC where the Debug@Chip is located. The 'tdstrip' program provided by Borland will remove the debug information from an executable file. This operation is recommended prior to downloading programs to the target IPC@CHIP to conserve Flash disk space.

© 2005 Beck IPC GmbH

4 Appendix

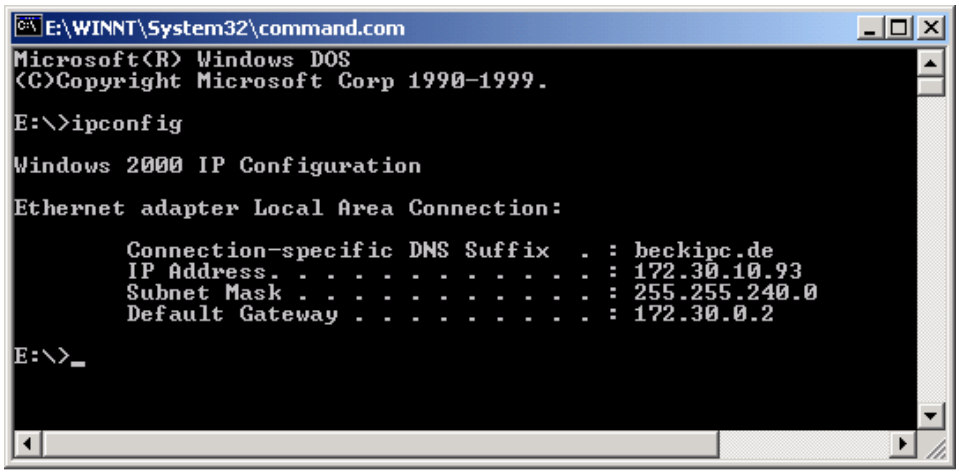
4.1 Basics of IP networking

On an IP (Internet Protocol) network we use IP addresses to define the source and the destination of an information packet. An intelligent 'trick' of the IP address is that it not only identifies the device, it also tells us where to find the device in a large network. This is best explained by an example:
Suppose we have the address 192.168.200.1 and the network mask 255.255.255.0. Let us look at this data in a 'binary' format:

Address	192.168.200.1	1100 0000. 1010 1000. 1100 1000. 0000 0001
Network mask	255.255.255.0	1111 1111. 1111 1111. 1111 1111. 0000 0000
Logic 'AND'	192.168.200.0	1100 0000. 1010 1000. 1100 1000. 0000 0000

The first 24 bits of the network mask are '1'. This means that the first 24 bits of the address identify the network. The last 8 bits (those that are '0' in the network mask) identify the device within this network. This would mean that in the given example, that we can give the devices in our network the addresses 192.168.200.0 through 192.168.200.255, thus allowing for 256 devices. This is almost correct, but not 100%. The lowest (here 192.168.200.0) and the highest address (here 192.168.200.255) are not to be used thus limiting us to 254 devices on this network. This we call a 'Class C' network. The highest address (here 192.168.200.255) has a special meaning: this is the address used for a broadcast to all devices on this section of the network. Any message sent to this address is received by all devices.

Any address that is not within the range of the current network is passed to the gateway address (the router). A gateway is a physical machine, hardwired into two or more physical networks. A gateway knows how to transfer an IP packet from one network to another.
You can find the network configuration of the network that you are connected to by entering the command 'ipconfig' in a Windows command shell.



On a network, no two devices are allowed to have the same address. For this reason, only one person can tell you what IP address you are allowed to use: your network supervisor or administrator. Tell him where you want to connect your IPC@CHIP and ask for a valid IP address, network mask and optionally the default gateway.

This is a short introduction and if you are looking for more detailed information about TCP/IP, we suggest the following books:

1. "Internetworking with TCP/IP, Volume 1" by Douglas Comer
2. "TCP/IP Illustrated, Volume 1" by W. Richard Stevens

© 2005 Beck IPC GmbH

4.2 Where to find other information on the IPC@CHIP

Regularly check the contents of our Internet pages at http://www.beck-ipc.com
Here you will find application interfaces, tools, examples, new @CHIP-RTOS versions, etc, etc.

Remember that the IPC@CHIP is an evolving product. It makes sense to visit the IPC@CHIP home page often.

4.3 Where to get support

There are two possible ways to get support:

1. To get support please use our newsgroup on the Internet:
You can visit the IPC@CHIP forum at http://www.beck-ipc.com/ipc/support/forum. Here you will find a lot of answers to customer problems and you can also add your own ideas and questions.
Please use email as the main means of communication. This not only reduces the time you spend on the phone, it also allows us to easily manage the questions and identify problem areas. Important or frequent questions and the corresponding answers are becoming part of our 'Frequently asked question list' that is published on the Internet.

2. To get support via email:
Please send your emails with your request to mailto:support@beck-ipc.com

You will find also a lot of answers in our Frequently Asked Questions documentation at http://www.beck-ipc.com/ipc/support.

4.4 Useful web links

Here is a list of some useful Internet links:

- http://hp.vector.co.jp/authors/VA002416/teraterm.html
  Home page for Tera Term Pro. A fine freeware terminal program for Windows 95/98/NT/2000/XP. (Note: This Internet address is case-sensitive.)
- http://www.beck-ipc.com/ipc. Our start page for all IPC@CHIP information.
- http://www.ghisler.com: Here you can find a fine file manager program named TotalCommander, including the FTP client functionality for Win 95/98/NT/2000/XP

4.5 Order numbers

Any part of the DK41 and DK51 Development Kits can be ordered separately. Please visit our Online Shop. The order numbers are:

20041011	DK41 Starter kit
20041010	DK40 Evaluation Module
192681	SM14 programming cable for DK40
20040112	SC12 Embedded Web Controller
190555	PSE4 power supply for DK40
538823	DK51 Development Kit
539361	DK50 Development Board
195837	ZK14 programming cable for DK51
536079	SC13 Embedded Web Controller
538934	PSE10 power supply for DK50
20040110	Borland C++ 5.02 SDE

End of this document

© 2005 Beck IPC GmbH