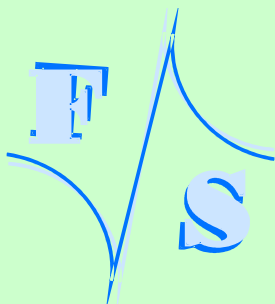


PicoCOM First Steps

Documentation

Version 2.4
(2012-04-26)

PicoCOM
Windows CE



About This Document

This documentation is a step by step introduction to how to use PicoCOM2, PicoCOM3, PicoCOM4 and PicoCOM5 with a PicoCOM2-Startinterface. The connectors referenced to in this documentation are on the PicoCOM2-Startintface. The documentation does NOT explain how to use a PicoCOM without a PicoCOM2-Startintf.

The PicoCOM2-Startinterface was designed primarily for PicoCOM2, but as all later modules within the PicoCOM family are pin compatible, the PicoCOM2-Startinterface is also used for all later modules (PicoCOM3, PicoCOM4, ...). The differences of each PicoCOM module are pointed out accordingly in this documentation.

The following abbreviations of terms are used in this documentation:

PC2 - PicoCOM2

PC3 - PicoCOM3

PC4 - PicoCOM4

PC5 - PicoCOM5

The latest version of this document can be found at:

<http://www.picocom.de>

Additional support information and software examples can be found in our discussion forum at:

<http://forum.fs-net.de>

© 2012

F&S Elektronik Systeme GmbH
Untere Waldplätze 23
D-70569 Stuttgart
Germany



History

Date	V	Platform	A,M,R	Chapter	Description	Au
2012-01-16	2.0	*	A,M	*	Merged to latest DIN A4 format template. Common documentation for all PicoCOMs.	MK
2012-01-18	2.1	PC3	M	*	Starterkit includes ET07008DH6 instead of ET057090DH.	MK
2012-01-20	2.1	*	*	*	Overall review and corrections	JG
2012-02-02	2.2	PC4	A	*	Display ET035080 added	KU
2012-03-30	2.3	*	A	2.1.1	Configuration not regarding ET035080.	MK
2012-04-26	2.4	PC2	A,M	2.1	Display backlight connecting	MK

V Version

A,M,R Added, Modified, Removed

Au Author

© 2012

F&S Elektronik Systeme GmbH
Untere Waldplätze 23
D-70569 Stuttgart
Germany



Table of Contents

1	Getting latest Software / Firmware	1
2	Connecting basic peripheral devices	2
2.1.1	Connecting the Display.....	3
3	Power-on PicoCOM	7
4	Bootup sequence	10
4.1	Nboot.....	10
4.2	Eboot.....	10
5	Firmware Update	11
5.1	The NetDCU-USBLoader utility.....	11
5.2	Updating – WindowsCE kernel image.....	12
5.2.1	Download via USB:.....	12
5.2.2	Download via Ethernet.....	13
5.2.3	Partitioning the Flash storage.....	13
5.3	Updating – EBOOT.....	15
5.3.1	Updating Eboot over the serial debug line.....	15
5.4	Updating – NBoot.....	15
6	Configuration	16
6.1	Using NDCUCFG utility	16
6.2	Network interface configuration.....	17
6.2.1	Network – General Facts.....	17
6.2.2	Network – Network address.....	18
6.2.3	Network – Set up with NDCUCFG.....	18
6.2.4	Network – Saving the parameters to registry.....	19
6.3	Touch panel calibration.....	20
6.3.1	Remote Tools (VS2005 / 2008).....	21
7	Software Development	22
7.1	Visual Studio – managed Code.....	23



7.2	Visual Studio – native Code.....	24
7.3	Visual Studio – debugging a device application.....	25
8	Support	26
9	Appendix	27
	Listings.....	27
	List of Figures.....	27
	List of Tables.....	27
	Important Notice.....	28



1 Getting latest Software / Firmware

All development programs and the latest firmware components mentioned in this document are available in the password protected download area of PicoCOM. We release and provide updated firmware versions periodically.

<http://download.fs-net.de>

The PicoCOM download area is password protected. You will find the required credentials on a small accompanying ticket that is included in your starterkit package.



2 Connecting basic peripheral devices

Next figure shows the PicoCOM2-Startintf with the position and description of connectors.

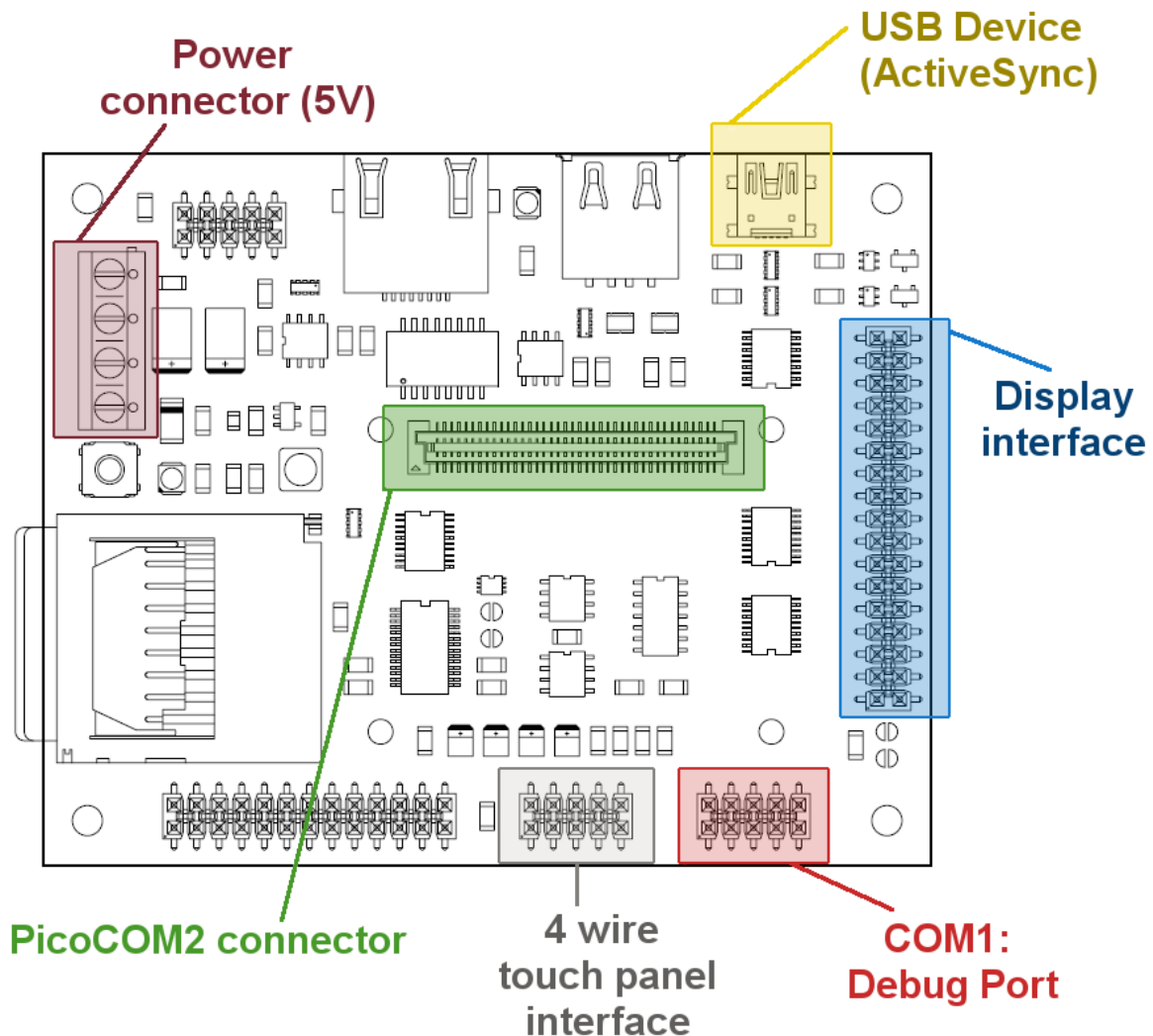


Figure 1: PicoCOM2-Startintf without PicoCOM.

All required cables and adapters are come with the Starterkit package. When connecting these cables **please take account of the pin1 marker on the cable and the connector.**

For power supply only one single 5V source is required. The input current VCFL is not required by the supplied Display (Hitachi TX09), as the contrast voltage is used for background power supply.

More information about the PicoCOM carrier board can be found in the PicoCOM2-Startintf documentation on our website.

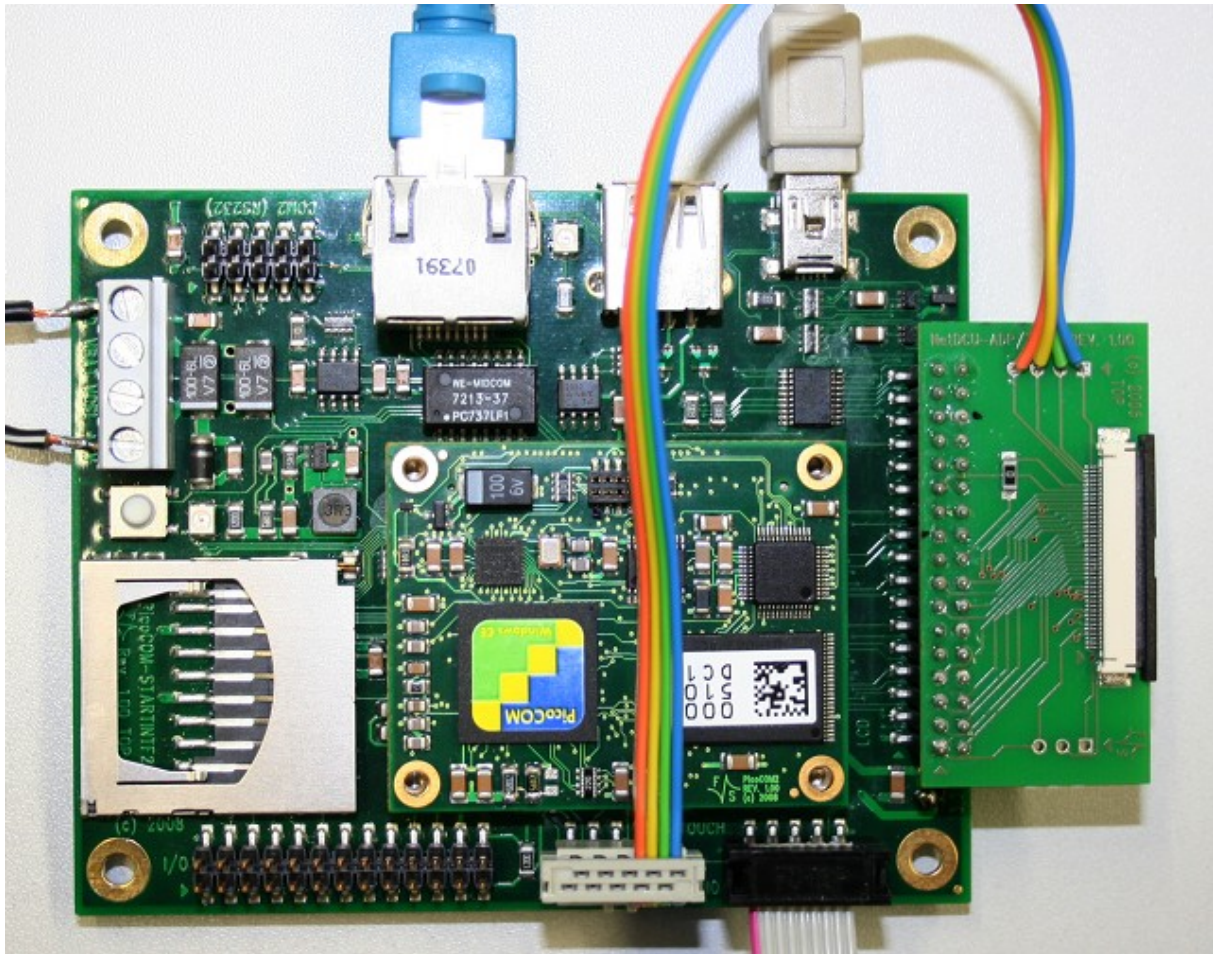


Figure 2: PicoCOM2-Startintf basic buildup

2.1.1 Connecting the Display

The display attached to the starterkit depends on the PicoCOM starterkit you have acquired. Table 1 shows a list of these displays.

PicoCOM Module	Display	Backlight powersupply
PicoCOM2	Hitachi TX-09	VEEK (3,3V)
PicoCOM3/ 5	EDT-ET070080DH6	VCFL (3,3V)
PicoCOM4	Hitachi TX-09 or EDT ET035080	VEEK (3,3V)

Table 1: List of Starterkit displays.

Connecting basic peripheral devices

To connect the Hitachi TX09 / EDT ET035080 / ET070080 display an adapter board is required (see figure 2). The display itself will be connected via a ribbon cable. The blue side of this cable must always remain on the upper side of the connector. On the adapter board the black pusher must be used to fix the cable (see figure 3).

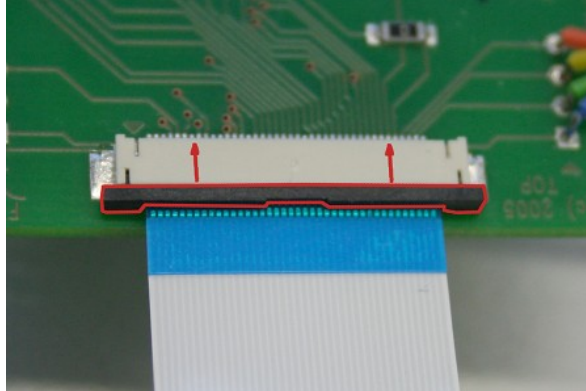


Figure 3: Ribbon cable on PicoCOM display adapter.

On the display itself the small white plastic bar must be pushed down to fix the cable (see figure 4,5 and 6).

Note:

The connectors for ribbon cable are very fragile. Please avoid handling by force.

Hitachi TX09

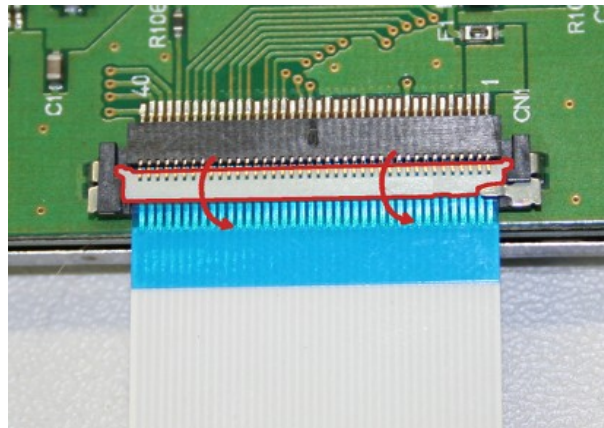


Figure 4: Ribbon cable Hitachi TX09

EDT ET035080

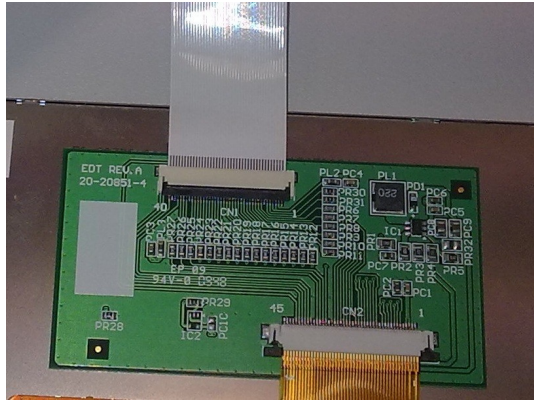


Figure 5: Ribbon cable on ET035080 display.

Note:

By default the PicoCOM4 is configured for a *Hitachi TX9* display. To reconfigure the display interface for *ETD ET035080* you need to run the following `nducfg` commands:

```
display contrast set 0xFF
display mode set 0
reg save
```

EDT ET70080

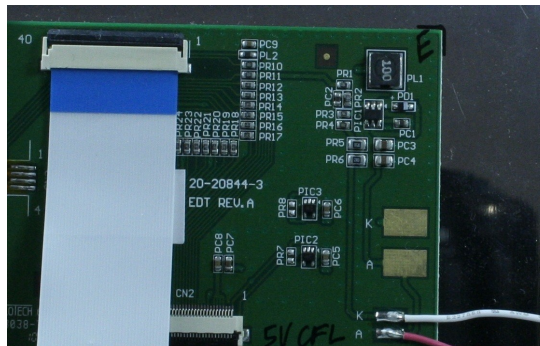


Figure 6: Ribbon cable on EDT ET70080 display

Note:

As the power-consumption of the smaller screens is very low, VEEK voltage can be used directly to regulate backlight. When using **EDT-ET070080DH6** the maximum current avail-

Connecting basic peripheral devices

able on VEEK isn't sufficient, therefore you need to connect 3,3V@1,5A to the VCFL pin of the carrier board.

Note:

More informations about the adapter boards you find in the documentation:

NetDCU-ADP/TX9 (Hitachi TX09) or

NetDCU-ADP/EDT1 (EDT ET035080/ ET070080)

3 Power-on PicoCOM

Before powering on PicoCOM, you should establish a serial connection between PicoCOM and your development PC. Please use the cables shipped with PicoCOM-Starterkit to connect your PC with the Debug-Port of PicoCOM (COM1). On PC side, you should have installed DCUTerm as terminal program, which is available in the password protected download area of PicoCOM.

Follow the steps below, to make a connection:

- Install DCUTerm on your PC
- Configure DCUTerm as shown in the following figure (the COM-Port may be different on your computer):

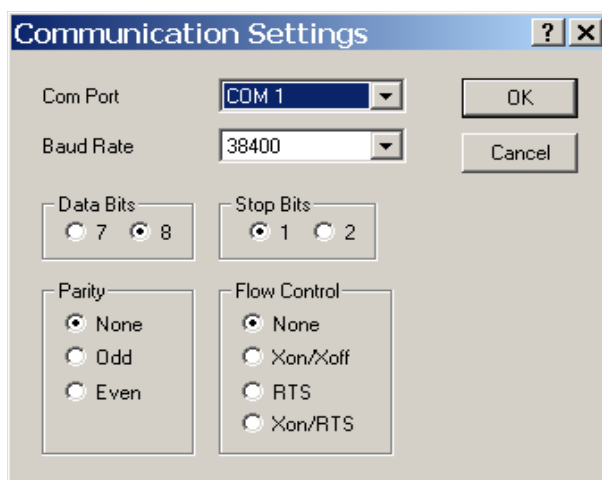


Figure 7: DCUTerm configuration: CommPort -> Settings

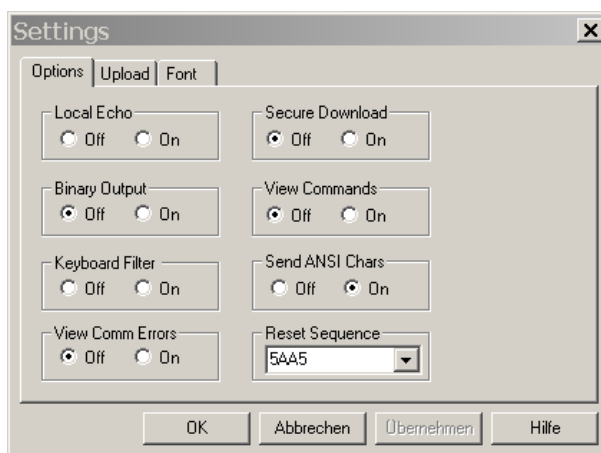


Figure 8: DCUTerm configuration: View -> Options

Power-on PicoCOM

- Press the connect button in DCUTerm

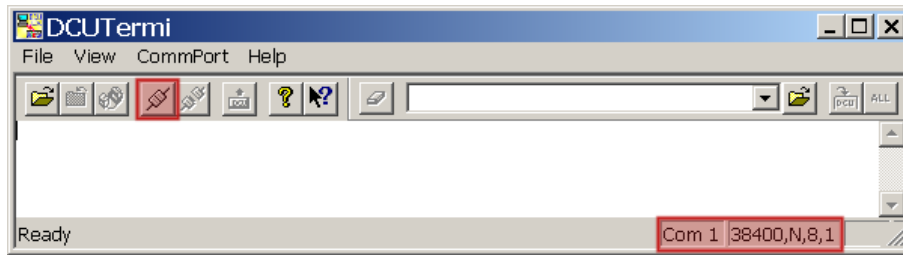


Figure 9: Opening serial connection in DCUTerm.

- Use the serial cable shipped with PicoCOM starterkit to make connection between COM1 and your PC.
- Apply the power connector to PicoCOM2-Startintf's power supply. At this moment you only need to connect ground and +5V (2A).

After these steps LED1 should be on and your serial output in the DCUTerm window should be similar to following:

```
Microsoft Windows CE Ethernet Bootloader Common Library Version 1.1
Built Dec 12 2008 16:31:35
Microsoft Windows CE Bootloader for PICOCOM2 Built Dec 15 2008
Portions copyright (c) 2007 F&S Elektronik Systeme GmbH
Boot Loader, Version 1.0
NBoot, Version C215

32 MB Ram
INFO:OALLogSetZones: dpCurSettings.ulZoneMask: 0xb
Board Revision: 2.0
System ready!
Preparing for download...
Press >S< to step into monitor...

Watchdog disabled

PICOCOM2 - WindowsCE Bootloader

:>
```

Listing 1: PicoCOM first boot-up, entering Eboot menu prompt.

Now you have entered PicoCOM bootloader prompt (:>). Next step will be downloading a Windows CE kernel image which will be described in chapter 5.

If you don't see a text similar to the above one:

- Check serial connection
- Check power supply
- Check settings of DCUTerm

Note:

You may enter the bootloader every time by pressing **<SHIFT> + 's'** (capital s) while powering on the PicoCOM. See chapter 4 for more details.



4 Bootup sequence

The start up process of PicoCOM is divided into three steps:

- **Nboot** (pcboot)
 - Responsible for low level initialization tasks
 - Loads the WindowsCE bootloader
- **Eboot** (WindowsCE bootloader)
 - Loads the WindowsCE kernel image
- **WindowsCE kernel image**
 - WindowsCE operating system and all drivers for PicoCOM
 - Offers you the possibility to develop and debug custom applications

Both bootloaders (Nboot and Eboot) feature a small configuration menu, that is accessible via serial debug port (COM1).

To open one of these boot menus, the following characters must be entered while booting the PicoCOM.

- Nboot: 's'
- Eboot: <SHIFT>+'S' (capital 's')

4.1 Nboot

The command menu of Nboot bootloader should only be entered in emergencies. For that reason it does not offer a huge amount of commands and is not be described in more detail in this documentation.

4.2 Eboot

By contrast, Eboot bootloader supports a wide range of commands that are used, for example, to setup the partitioning of flash memory, erase the Registry, enable or disable the debug output, etc. . The command '?' gives you a full list of all available commands including a short description. Some fundamental commands, which are required to update firmware components, are described in the following chapter.

Note:

Operating with the Nboot and Eboot menu gives you the possibility to erase the complete flash. Please always keep in mind that all data stored in flash and even the Eboot itself could get lost.

5 Firmware Update

All three firmware components of the PicoCOM, described in the chapter before, can be updated separately. The following chapter describes these operations in more detail.

5.1 The NetDCU-USBLoader utility

The preferred method to update PicoCOM is using the NetDCU-USBLoader which offers the possibility to download bootloaders and the WindowsCE kernel to PicoCOM via USB. The NetDCU-USBLoader is available in the PicoCOM download area on our website.

First you need to install NetDCU-USBLoader on your development PC. Please follow the instructions of the installation process step by step. If you are asked to install the unsigned driver, please acknowledge with 'Yes' (compare figure 10). If you disagree, the program may not work properly afterwards.

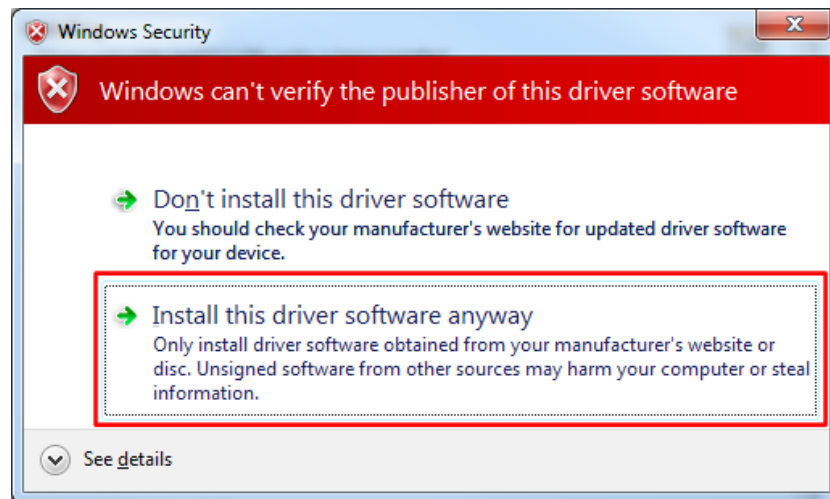


Figure 10: Acknowledge unsigned driver.

If there are problems during installation please refer to the installation documentation , shipped with NetDCU-USBLoader package.

The procedure of downloading a new bootloader or WindowsCE kernel with this utility is described in chapters 5.2 to 5.4.

5.2 Updating – WindowsCE kernel image

You can update the Windows CE kernel via Ethernet or by using the NetDCU-USBLoader utility via USB which is the preferred method. Therefore you must enter the WindowsCE bootloader (Eboot) first by pressing **<SHIFT> + 's'** while powering on the PicoCOM.

5.2.1 Download via USB:

To download the WindowsCE Kernel via USB, start the NetDCU-USBLoader utility on your desktop PC and use the command `DU` to enter the USB download mode (1). As soon as the connection is established, the button in the top right corner of NetDCU-USBLoader turns from red to green (2). Select the respective `<NK-kernel-image>.bin` file (3) and click the 'Start' button (4). You should now see progress similar to figure 11.

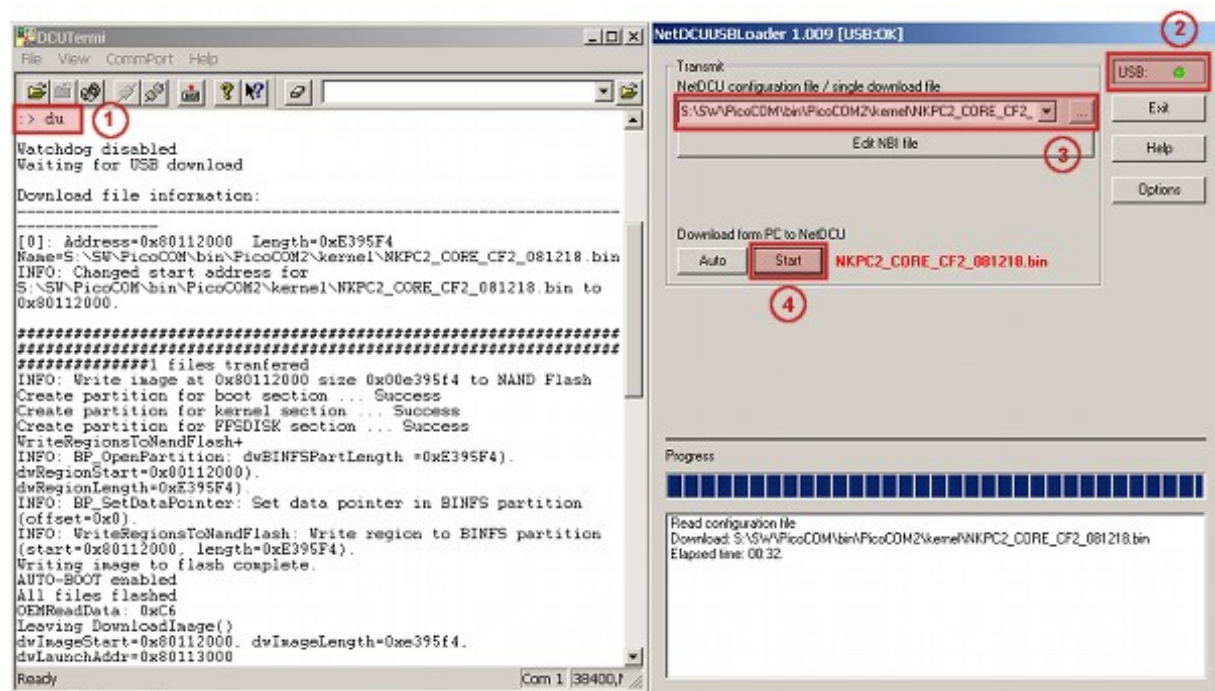


Figure 11: Using the NetDCU-USBLoader utility.

After the download operation has finished and the kernel has been started, the Windows desktop should appear on the connected display.

On serial debug line, the NDCUCFG utility will be activated by default. Eboot menu is no longer prompted on start up. Please refer to chapter 6.1 to get more details about the NDCUCFG utility.

Note:

The WindowsCE kernel can be downloaded to RAM or Flash. This can be configured with Eboot commands:

- MR - Store kernel in RAM memory
- MF - Store kernel on Flash disk

If the WindowsCE kernel does not start automatically after reboot, you have to execute the following Eboot command:

- L1 - Launch previously stored kernel after boot

5.2.2 Download via Ethernet

Implemented on: PC2, PC3, PC5

In addition to using NetDCU-USBLoader to update WindowsCE kernel image via USB, the Microsoft utility called Eshell can be used to download a kernel image via Ethernet. For this purpose, enter DE command within Eboot and start the Eshell utility on your development machine. As soon as PicoCOM sends a network broadcast ('Sent BOOTME to 255.255.255.255') you should be able to select the device in the 'Select Device' dialog of Eshell. Then choose 'Select image' from the 'File'-Menu of Eshell and select the respective <NK-kernel-image>.bin file. Press 'open' – this starts the download to PicoCOM.

5.2.3 Partitioning the Flash storage

Depending on the Windows CE kernel image you are going to run on PicoCOM, the default size of kernel partition might not be sufficient for your needs and it might be required to re-partition the flash storage.

There are up to 3 partitions possible on PicoCOM:

- **OS-Image (BINFS):** The WindowsCE kernel is stored in this partition.
- **FFSDISK:** This partitions can be used to store user data and applications. It is available under '\FFSDISK' on a running WindowsCE system.
- **Extended Partition:** The extended partition must be administrated in WindowsCE. There are no partitions available by default, so you have to create them using the Storage-Manager. The size of this partition might be 0 in most cases.

Partitioning the flash memory must be performed within the Eboot menu. With the command ? you get a list of all available Eboot commands. Partitioning is setup with the command P.

```
:> P
-----PARTITION CONFIGURATION-----
Current settings:
```



Firmware Update

```
Total      : 33 MB
OS-Image: 15 MB
FFSDISK   : 18 MB, Part type: FAT
SECOND    : 0 MB, Part type: EXTENDED
Enter maximal size for OS-Image [15]:
```

Listing 2: Flash partitioning (part1).

At this point you may resize the partitions for BINFS, FFSDISK and the EXTENDED partition. Follow the instructions printed in the terminal program. After confirming the updated partition settings, you should see an output similar to the one printed below.

```
WindowsCE image and all data in FFSDISK will be erased.
Continue ? (Y/n) Y
FMD: Can't erase block 0x0
FMD: Can't erase block 0x1
.
.
FMD: Can't erase block 0x11
done
Create partition for boot section .....
Success
Create partition for kernel section ...
Success
Create partition for FFSDISK section ...
Success
```

Listing 3: Flash partitioning (part2).

You can ignore the FMD messages. Some blocks are reserved for storing the bootloader and you should not worry about these messages.

Please wait for Eboot prompt (:>) to return before continuing.

5.3 Updating – EBOOT

Updating Eboot is done similar to updating WindowsCE Kernel with the NetDCU-USBLoader utility. Enter the menu of the currently installed Eboot version by pressing **<SHIFT>+s'** while powering on the PicoCOM. To download the new Eboot version (eboot_<version>.nb0) press **DU** and start the NetDCU-USBLoader utility on your development PC. In NetDCU-USBLoader select the respective <eboot>.nb0 file and click the **'Start'** button.

5.3.1 Updating Eboot over the serial debug line

You can also download the Eboot bootloader via the serial debug port. Therefore enter the Nboot menu (**'s'**) and press **d** to start the serial download. In the **'File'**-Menu of the DCUTerm utility choose **'Transmit binary file'**, select the respective <eboot>.nb0 file and press **'open'**. After the download has finished use **f** command to save the eboot image on flash.

Note:

Serial downloads are very prone to errors. For this reason, please try avoiding this method. To make this update variant more robust, a checksum is calculated and displayed after downloading has finished. You can verify this checksum with a small utility (bootloader-checksum) that can be downloaded from the PicoCOM download area.

5.4 Updating – NBoot

Updating the Nboot bootloader can also be accomplished with NetDCU-USBLoader. The process is similar to updating a Windows CE kernel or an Eboot image. After starting USB transfermode in Eboot menu (**DU**), you simply select the Nboot image for installing in NetDCU-USBLoader and press **'Start'**.



6 Configuration

The configuration of a PicoCOM device is provided by different means. Though the most powerful and acceptable way is running the NDCUCFG software utility. In fact, this is a standard command prompt program, allowing you to adjust a variety of system parameters.

Most of changes done to a PicoCOM device through NDCUCFG utility, are stored in persistent system registry, and take effect after next reboot of the PicoCOM device.

According to the device's software architecture, this utility gets automatically started on the COM1. As well, the utility can be remotely executed over a Telnet connection, once you have got network access to the PicoCOM device.

All in all, software components and the core of the operating system running on PicoCOM, offer you an easy and effective way to make necessary settings.

Chapter 6.2 contains a description of network adjustments, to make it possible to connect and transfer files via Ethernet. It also demonstrates quite well, how to use the NDCUCFG utility.

6.1 Using NDCUCFG utility

You can either enable debug messages or the NDCUCFG utility on COM1. The NDCUCFG utility will automatically run on COM1. To select the usage of this serial port, you must enter the Eboot menu. With the command `o` you can enable or disable the serial debug output during booting. If you choose `y`, the NDCUCFG utility gets started on COM1.

After booting the PicoCOM module, wait until following prompt message appears:

```
NetDCU Config Utility Ready
Version: 030
Type help for commands

!>
```

Listing 4: NDCUCFG command prompt.

When this command prompt appears in the terminal program, you are ready to pass commands to the NDCUCFG utility. Otherwise something went wrong and the kernel image could not be booted correctly. Please check various parameters described in chapter 1.2.

If NDCUCFG is running successfully on the serial debug port, you can start passing commands to this utility. It is recommended to issue `help` as first command. This is the final part of what you see when issuing it:

```
!>help
... list of messages starts earlier ...
backlight off
start <file name>
quit
```

```
help
help <command>
!>
```

Listing 5: NDCUCFG help command.

You definitely know how to use such trivial (but important!) commands as `help` and `quit`. For all other commands, you can use the hint given you in the last string of the output above. i.e., if you do not know how to issue the command `backlight`, then you type the following command and then press *Enter*:

```
!>help backlight
```

Two possible ways of executing this command are shown you in response. If you're still interested in what the command `backlight off` does, just type the following and finish with *Enter*:

```
!>help backlight off
```

and you get a satisfying answer to you help-request. To save the changes, execute the command:

```
!>reg save
```

You have to reboot the device to make the changes effective. Upper examples demonstrate how the NDCUCFG utility functions in general. Now, let us set up the Network.

6.2 Network interface configuration

PicoCOM implements a powerful and stable Ethernet interface, which allows customers to create a variety of modern hardware Internet applications, highly required by the modern market of data processing and transporting appliances.

The Ethernet interface implemented in PicoCOM meets 802.3 10BaseT specifications by IEEE, and provides safe data transfer on speeds up to 100 Mbit/sec.

6.2.1 Network – General Facts

To be integrated into an IP-network, in order to get directly referred by other network devices, every PicoCOM device must obtain its own IP-address, unique within the entire network segment. Such an address, along with other necessary parameters, must generally be confirmed by a network administrator.

Get a preferred IP-address from a range of currently available IP-addresses (for example 192.168.5.5), and mark this address as one currently being assigned to PicoCOM. Ask your network administrator if you don't know how to obtain unused IP-addresses, or see "Network – Network address".

The hardware layer which communicates between network devices, assumes every device to have one more address. This other kind of address is a so-called MAC-address, or 'Ether-



Configuration

net address', or 'physical address'. It is formed of a six-byte sequence, and, in accordance to corresponding IEEE's regulations, is unique for every network device across the World.

6.2.2 Network – Network address

Every IP-Address can be split into the network address and station address. It's not part of this documentation to describe all details but we want to explain how you can obtain your network address from your PC.

Open command window and type:

```
C:> ipconfig
```

then press *Enter*. The output you get should be relative to the following:

Windows IP configuration:

Ethernet Adapter RTL81391:

IP address: 192.168.5.131

Subnet mask: 255.255.255.0

Standard Gateway:

With this information you can calculate your network address. Interpret the values as hexadecimal values and do a logical AND of IP address and subnet mask. The result is the network address.

192	.	168	.	5	.	131
255	.	255	.	255	.	0
192	.	168	.	5	.	0

So, for our example the network address is 192.168.5.0 and the station address within this network is 192.168.5.131. Only stations that are in the same network can communicate with each other.

6.2.3 Network – Set up with NDCUCFG

Almost all device settings can be configured by registry. Therefore you use the reg commands of the NDCUCFG utility as described below:

```
!>help reg
reg open
reg open <key>
reg opencu <key>
reg enum key <#>
reg enum key *
reg enum value <#>
reg enum value *
```

```

reg set value <name> dword <value>
reg set value <name> string <value>
reg set value <name> multi <value1>;<value2> ;<valueN>
reg set value <name> hex <value>,<value>,<value>
reg create key <name>
reg del value <name>
reg del key <name>
reg save
reg erase
!>

```

Listing 6: NDCUCFG reg commands.

The Network parameters for PicoCOM can be found under:

```
[HKLM\Comm\ETHNETA1\Parms\TcpIp]
```

So execute the command:

```
!>reg open \Comm\ETHNETA1\Parms\TcpIp
OK
```

to access the network parameters. The output ok tells you that NDCUCFG could successfully open the path.

I.e. you can change the value IpAddress with the command:

```
!>reg set val IpAddress string "10.0.0.111"
OK
!>reg enum
OK -> reg enum key \
OK -> reg enum value \
00 "IpAddress"=string:10.0.0.111 \
01 "EnableDHCP"=dword:0 \
02 "UseZeroBroadcast"=dword:0 \
03 "DefaultGateway"=string:192.168.0.1 \
04 "Subnetmask"=string:255.0.0.0 \
05 "DNS"=string:0.0.0.0 \
06 "WINS"=string:0.0.0.0 \
OK
```

Listing 7: Setting Ethernet IP-address.

6.2.4 Network – Saving the parameters to registry

After changes as for type of network were correctly done, and the special checkings following it, have approved this fact, it's suitable time to save those changes from RAM memory to physical media, so they take an effect after the next reboot of your PicoCOM device.

Enter the following:

```
!>reg save
```

Listing 8: Saving the registry permanently.



Configuration

and press *Enter*. The procedure of physical saving takes about two seconds – do nothing during this period! If you can see the 'OK' message again, it means that all the changes provided to the PicoCOM system during the current session of working with NDCUCFG utility, are stored in persistent registry.

6.3 Touch panel calibration

The TX09 display, available with the starterkit, is equipped with a 4 wire touch panel. At delivery status this touch panel should already be calibrated satisfyingly. But when using another display, or if the touch panel is not working properly, it might be required to calibrate the touch panel driver again. This is performed by using NDCUCFG.

The desired command is “touch calibrate”. After executing this command, the touch panel calibrating routine should be displayed on the LCD screen.

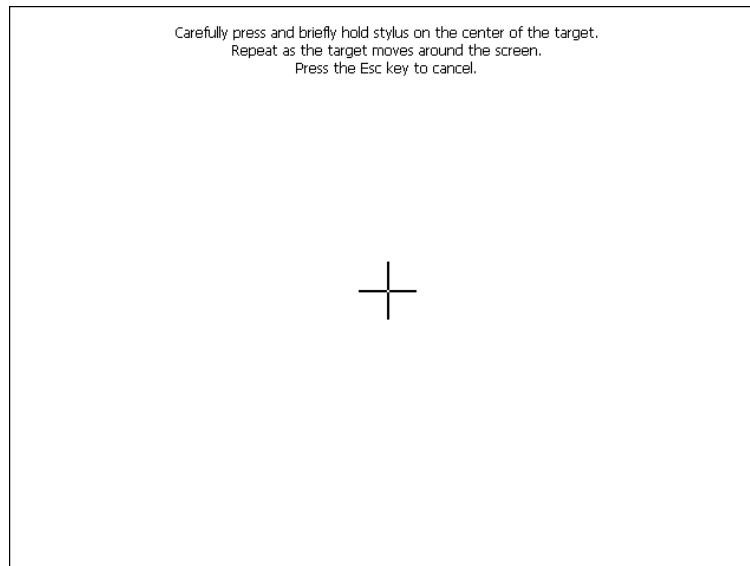


Figure 12: Initial screen of the touch panel calibration program.

After holding the stylus on the cross in the center of the screen, the cross moves around to each corner where it must be 'touched' again. When the calibration is finished, the cross disappears and you are requested to press the Enter key. If there is no keyboard available, you can simply hold the stylus anywhere on the screen to close the calibration program. The regular Windows desktop should appear again and the calibration is finished.

As the calibration settings are stored in registry, the registry must be saved to accept the new settings permanently (“reg save” in NDCUCFG). Please note that calibration settings will be reset every time the registry is reset.

6.3.1 Remote Tools (VS2005 / 2008)

Microsoft Visual Studio 2005 / 2008 is shipped with a couple of useful Remote Tools.

- Remote File Viewer – File Explorer
- Remote Heap Walker – lists Heap per Process
- Remote Process Viewer – Task Manager
- Remote Spy - displays messages received by windows associated with applications running on a target device
- Remote Zoom In - On a development workstation, Remote Zoom-in displays a screen image from a target device



7 Software Development

For software development you have to use Visual Studio 2005 or later. You can download the kernel-image from the website www.picocom.de. It already includes Microsoft Compact Framework 3.5. This enables the developer to write managed code in C# or VB.NET. It is also possible to develop applications in native code (C++), by using Win32 API or MFC. To use the native code, you need to install the PicoCOM SDK, which you also can download from the above mentioned website.

To connect Visual Studio to PicoCOM for software development, you can use an USB device connection or an Ethernet connection.

A detailed description of how to connect via Ethernet is given by “Connect VS 2005 to Net-DCU using LAN” which is analog to PicoCOM2. Download it from <http://download.picocom.de> .

The best way (because easy to handle) is to connect via USB, using Microsoft ActiveSync. For this, install the latest version of ActiveSync on your PC (download ActiveSync from <http://www.microsoft.com>) and connect the PicoCOM and the PC by using the USB device connector shipped with the PicoCOM2-Startint. The connection is established automatically.

7.1 Visual Studio – managed Code

The application programmer can develop the application in C# or VB.NET, using Compact Framework 3.5, which is part of the WindowsCE kernel for PicoCOM.

The next figure shows the project settings when using managed code.

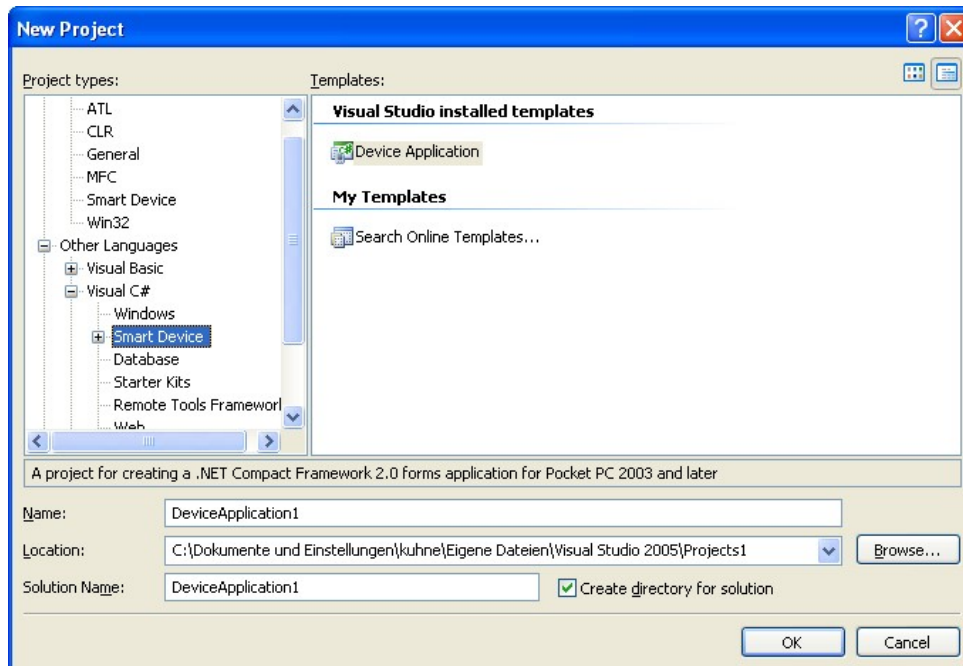


Figure 13: Creating new managed Smart Device Project.

7.2 Visual Studio – native Code

The application programmer can develop the application in C++, using the PicoCOM SDK, which can be downloaded from www.picocom.de/download.

The next figure shows the project settings when using native code.

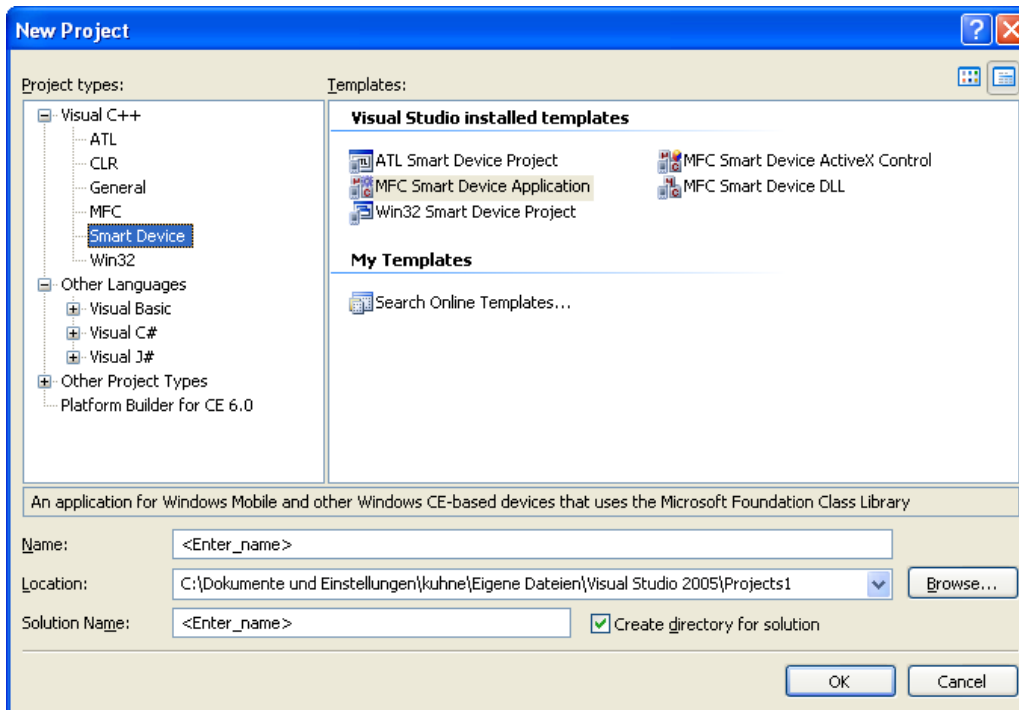


Figure 14: Creating new unmanaged Smart Device Project.

7.3 Visual Studio – debugging a device application

The application developer can debug an application via Ethernet or via USB – which is the recommended way. To debug via USB, establish a connection between your development PC and PicoCOM with ActiveSync. As soon as the connection is set up, you can start debugging the application with breakpoints etc., as you know from applications for desktop PCs.

Note:

In case you start your application in Visual Studio with 'Start debugging' you may get memory problems on your PicoCOM, please disable deploying the latest version of Compact Framework. Therefore select menu Project- Properties- Devices and deselect:

Deploy the latest version of the .NET Compact Framework (including Service Packs)

Support

8 Support

For questions regarding PicoCOM, application development or other support requests, we offer free forum support at <http://forum.fs-net.de> . The forum is sponsored by F&S' hard- and software developers which assist you promptly.

It also features a lot of examples and additional information about PicoCOM- and other product-families.

9 Appendix

Listings

Listing 1: PicoCOM first boot-up, entering Eboot menu prompt.....	8
Listing 2: Flash partitioning (part1).....	14
Listing 3: Flash partitioning (part2).....	14
Listing 4: NDCUCFG command prompt.....	16
Listing 5: NDCUCFG help command.....	17
Listing 6: NDCUCFG reg commands.....	19
Listing 7: Setting Ethernet IP-address.....	19
Listing 8: Saving the registry permanently.....	19

List of Figures

Figure 1: PicoCOM2-Startintf without PicoCOM.....	2
Figure 2: PicoCOM2-Startintf basic buildup.....	3
Figure 3: Ribbon cable on PicoCOM display adapter.....	4
Figure 4: Ribbon cable Hitachi TX09.....	4
Figure 5: Ribbon cable on ET035080 display.....	5
Figure 6: Ribbon cable on EDT ET70080 display.....	5
Figure 7: DCUTerm configuration: CommPort -> Settings.....	7
Figure 8: DCUTerm configuration: View → Options.....	7
Figure 9: Opening serial connection in DCUTerm.....	8
Figure 10: Acknowledge unsigned driver.....	11
Figure 11: Using the NetDCU-USBLoader utility.....	12
Figure 12: Initial screen of the touch panel calibration program.....	20
Figure 13: Creating new managed Smart Device Project.....	23
Figure 14: Creating new unmanaged Smart Device Project.....	24

List of Tables



Table 1: List of Starterkit displays.....3

Important Notice

The information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication. F&S Elektronik Systeme assumes no responsibility, however, for possible errors or omissions, or for any consequences resulting from the use of the information contained in this documentation.

F&S Elektronik Systeme reserves the right to make changes in its products or product specifications or product documentation with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

F&S Elektronik Systeme makes no warranty or guarantee regarding the suitability of its products for any particular purpose, nor does F&S Elektronik Systeme assume any liability arising out of the documentation or use of any product and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

Products are not designed, intended, or authorised for use as components in systems intended for applications intended to support or sustain life, or for any other application in which the failure of the product from F&S Elektronik Systeme could create a situation where personal injury or death may occur. Should the Buyer purchase or use a F&S Elektronik Systeme product for any such unintended or unauthorised application, the Buyer shall indemnify and hold F&S Elektronik Systeme and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorised use, even if such claim alleges that F&S Elektronik Systeme was negligent regarding the design or manufacture of said product.