

Application Developer Kit

Documentation

for HiCO.SH7780

Copyright



© Copyright 2009 emtrion GmbH

All rights reserved. Without written permission this documentation may neither be photocopied nor stored on electronic media. The information contained in this documentation is subject to change without prior notification. We do not assume any liability for erroneous information or its consequences. The trademarks of other companies that are used identify the products of these companies exclusively. Microsoft, Windows, Windows95, Windows98, Windows NT, Windows 2000, Windows XP, Windows CE and MS-DOS are registered trademarks of the Microsoft Corporation.

| Revision No. | Changes | Date |
|--------------|--|---------------|
| 1 | First edition | 25.08.2008/Rr |
| 2 | Display settings, HiCOCAN-104 support, minor changes | 27.03.2009/Rr |
| 3 | Minor corrections | 18.08.2009/Rr |

This document is published by:

Emtrion GmbH

Greschbachstr. 12

D-76229 Karlsruhe

Tel: +49 (721) 62725-0

Fax: +49 (721) 62725-19

E-mail: mail@emtrion.de

Internet: <http://www.emtrion.de>

March 2009-001

Index

| | | |
|----------|--|-----------|
| 1 | Introduction | 5 |
| 2 | BSP contents | 6 |
| 2.1 | Applications – End User | 6 |
| 2.2 | Applications and Services Development | 6 |
| 2.3 | Communication Services and Networking | 7 |
| 2.4 | Core OS Services | 7 |
| 2.5 | Device Management | 8 |
| 2.6 | File Systems and Data Store | 8 |
| 2.7 | Fonts | 8 |
| 2.8 | Graphics and Multimedia Technologies | 8 |
| 2.9 | International | 9 |
| 2.10 | Internet Client Services | 9 |
| 2.11 | Security | 9 |
| 2.12 | Shell and User Interface | 9 |
| 2.13 | Device Driver | 10 |
| 2.13.1 | Display settings | 10 |
| 2.13.2 | Special hints for using HiCOCAN-104 | 12 |
| 2.13.3 | Jumpers of HiCOCAN-104 | 15 |
| 2.13.4 | Board Number, X4 | 15 |
| 3 | Application development | 16 |
| 3.1 | VS2005 for Application Development | 16 |
| 3.1.1 | Creating a new Managed Project | 16 |
| 3.1.2 | Deploying to the HiCO.SH7780 | 17 |
| 4 | Starting an Application at system start | 20 |
| 4.1 | Autostart with launchXX and dependXX | 20 |
| 4.2 | Emtrion's Autostart Mechanism | 21 |
| 5 | Preparing the board for operation | 22 |
| 5.1 | Serial Ports | 23 |
| 6 | Data Exchange between Target and Workstation | 25 |
| 6.1 | Preparing the Desktop Computer | 25 |
| 6.2 | Preparing the Windows CE Device | 25 |
| 6.3 | USB ActiveSync | 25 |
| 6.4 | Remarks on Using ActiveSync | 26 |
| 7 | Persistent Registry | 30 |
| 7.1 | Deleting the Persistent Registry | 31 |
| 7.1.1 | Deleting the Registry by means of the Bootloader | 31 |
| 7.1.2 | Deleting the Registry by means of an Application under Windows CE | 31 |
| 8 | Replacing the Starter Kit Kernel | 32 |
| 8.1 | The Bootloader Menu | 33 |
| 8.2 | Bootloader Menu Items | 34 |
| 8.2.1 | Execute stored Image | 34 |
| 8.2.2 | Download Image via serial port and store in Flash | 34 |
| 8.2.3 | Download Image via serial port, store in SDRAM and execute | 34 |
| 8.2.4 | Download Image via LAN9218 Ethernet controller and store in Flash | 34 |
| 8.2.5 | Download Image via LAN9218 Ethernet controller, store in SDRAM and execute | 35 |
| 8.2.6 | Extended functionality | 35 |
| 8.2.7 | Download image over standard TFTP connection | 35 |
| 8.2.8 | Self test | 35 |
| 8.2.9 | Set real time clock | 35 |

| | | |
|-----------|---|-----------|
| 8.2.10 | Upload stored data | 36 |
| 8.2.11 | Delete the persistent registry of Windows CE | 36 |
| 8.2.12 | Download contents for flash sectors (not OS) over serial port | 36 |
| 8.2.13 | Download contents for flash sectors (not OS) over LAN9218 Ethernet contr. | 36 |
| 8.2.14 | Return to main menu | 36 |
| 8.3 | Supported File Formats | 37 |
| 9 | Emtrion Tools for Windows CE | 38 |
| 9.1 | Getting the Version of the Operating System Kernel | 38 |
| 9.2 | Setipadr | 38 |
| 10 | Licence Agreements | 39 |

1 Introduction

This application package addresses all software engineers who don't want to care about the details of the Windows embedded CE 6.0 R2 kernel and who directly want to start writing applications. That's what emtrion GmbH bundled packages with all the files which are necessary for writing applications. Mainly two kernels and their SDKs with a typical set of components are available. They base on rich experiences with different customer specific kernels and are therefore suitable for the most applications. One kernel is a GUI based kernel and the second one is a so called headless kernel. The details of the kernel are depicted in latter chapters.

Anyway, if these kernels don't fit to the applications needs there are two possibilities to fix this problem. The first one is that the customer purchases a developer kit and does all the modifications of the BSP by means of the MS Platform Builder by himself or he asks emtrion GmbH to do this work for him.

The following chapters of this documentation give a brief overview about all what is good to know when starting writing applications for emtrion's HiCO.SH7780 single board computer. Parts of the following chapters are excerpts of the documents which are part of the developer kit packages.

Important:

The kernels coming with this package are not licensed and have a limited live time. You can use them for demonstration, tests and application development.

If you need a kernel for your series production it is mandatory to activate the kernel by emtrion GmbH and to have a license from Microsoft for each CPU on which this kernel runs. Please contact emtrion GmbH about the details of licensing and on how to get such licenses from Microsoft.

2 BSP contents

This chapter describes briefly the contents of the two online available kernels. These kernels are some sort of typical kernels which should be suitable for most of the applications. If modifications are necessary, you may consider purchasing a developer kit from emtrion or just asking emtrion to design an individual kernel for your needs.

2.1 Applications – End User

| Kernel | Headless | Starterkit |
|--------------------------------|----------|------------|
| Active Sync | X | X |
| File Sync | X | X |
| CAB File Installer/Uninstaller | - | X |

2.2 Applications and Services Development

| Kernel | Headless | Starterkit |
|---|----------|------------|
| .NET Compact Framework 2.0 | | |
| .NET Compact Framework 2.0 | - | - |
| .NET Compact Framework 2.0 – headless | - | - |
| OS Dependencies for .NET CF 2.0 | - | - |
| OS Dependencies for .NET CF 2.0 - headless | - | - |
| .NET Compact Framework 3.5 | | |
| .NET Compact Framework 3.5 | - | X |
| .NET Compact Framework 3.5 – headless | X | - |
| OS Dependencies for .NET CF 3.5 | - | X |
| OS Dependencies for .NET CF 3.5 – headless | X | - |
| Active Template Library (ATL) | - | X |
| C Libraries and Runtimes | | |
| C++ Runtime Support for Exception Handling and Runtime Type Information | X | X |
| Full C Runtime | X | X |
| Standard I/O (STDIO) | X | X |
| Standard I/O ASCII (STDIOA) | X | X |
| Standard String Functions – ASCII | X | X |
| COM and DCOM | | |
| DCOM, incl. COM Storage | | X |
| Minimal COM (No OLE Support) | X | |
| String Safe Utility Functions | X | X |
| XML | | |
| XML Core Services and Document Object Model | X | X |
| XML Query Languages (XQL) | X | X |

2.3 Communication Services and Networking

| Kernel | Headless | Starterkit |
|--|----------|------------|
| Networking General | | |
| NDIS User-mode I/O Protocol Driver | X | X |
| Network Driver Architecture (NDIS) | X | X |
| Network Utilities (Ipconfig, ping, route, netstat) | X | X |
| TCP/IP | X | X |
| IP Helper API | X | X |
| TCl/Ipv6 Support | X | X |
| Windows Networking API/Redirector (SMB/CIFS) | X | X |
| Winsock Support | X | X |
| Networking – Local Area Network (LAN) | | |
| Wired Local Area Network (802.3, 802.5) | X | X |
| Networking – Wide Area Network (WAN) | | |
| Dial Up Networking (RAS/PPP) | X | X |
| AutoDial | X | X |
| Standard Modem Support for Dial Up Networking | X | X |
| Telephony API (TAPI 2.0) | X | X |
| Unimodem Support | X | X |
| Servers | | |
| Core Server Support | X | X |
| File Server (SMB/CIFS) | X | X |
| File Server Customizable Web UI | X | X |
| FTP Server | X | X |
| RAS Server/PPTP Server (Incoming) | X | X |
| SNTP Client with DST | X | X |
| Telnet Server | X | X |
| Web Server (HTTPD) minimum for file server | X | X |

2.4 Core OS Services

| Kernel | Headless | Starterkit |
|--|----------|------------|
| Toolhelp API | X | X |
| Device Manager | X | X |
| Display Support | X | X |
| Serial Port Support | X | X |
| UI Proxy for Kernel-Mode Drivers | X | X |
| Windows Embedded CE Driver Development Kit Support Library | X | X |
| Kernel Funtionality | | |
| Fiber API | - | X |
| Format Message API | X | X |
| Format Message API – System Error Messages | X | X |
| Memory Mapped Files | X | X |
| Message Queue – Point-to-Point | X | X |
| Target Control Support (Shell.exe) | X | X |
| Notification | | |
| Non UI based Notification | X | - |
| UI based Notification | - | X |
| Power Management | | |
| Power Management (Full) | | X |
| Power Management (Minimal) | X | - |
| USB Host Support | | |

| | | |
|---|---|---|
| USB Function Driver | X | X |
| USB Host Support | X | X |
| USB Human Input Device (HID) Class Driver | X | X |
| USB HID Keyboard and Mouse | - | X |
| USB HID Keyboard only | - | X |
| USB HID Mouse only | - | X |
| USB Printer Class Driver (PCL-3) | X | X |
| USB Storage Class Driver | X | X |

2.5 Device Management

| | | |
|---|----------|------------|
| Kernel | Headless | Starterkit |
| Simple Network Management Protocol (SNMP) | X | X |
| Device Management Client | - | X |

2.6 File Systems and Data Store

| | | |
|--|----------|------------|
| Kernel | Headless | Starterkit |
| Compression | X | X |
| Database Support | | |
| CEDB Database Engine | X | X |
| File and Database Replication, bit-based | X | X |
| File Cache Manager | - | - |
| File System – Internal | | |
| RAM and ROM File System | X | X |
| ROM-only File System | - | - |
| Registry Storage | | |
| Hive-based Registry | - | - |
| RAM-based Registry | X | X |
| Storage Manager | | |
| FAT File System | X | X |
| Partition Driver | X | X |
| Release Directory File System | X | X |
| Storage Manager Control Applet | - | X |
| System Password | X | X |

2.7 Fonts

| | | |
|---------------------------|----------|------------|
| Kernel | Headless | Starterkit |
| Arial (Subset 1_30) | - | X |
| Courier New (Subset 1_30) | - | X |
| Symbol | - | X |
| Tahoma (Subset 1_30) | - | X |
| Tahoma Bold | - | X |
| Wingding | - | X |

2.8 Graphics and Multimedia Technologies

| | | |
|---------------------------|----------|------------|
| Kernel | Headless | Starterkit |
| Audio | | |
| Audio Compression Manager | X | X |
| Waveform Audio | X | X |

| Graphics | | |
|---|---|---|
| DirectDraw | - | X |
| Gradient Fill Support | X | X |
| Still Image Codec Support (Encode and Decode) | - | X |
| Still Image Encoders and Decoders | - | X |
| BMP Decoder | - | X |
| GIF Decoder | - | X |
| JPG Decoder | - | X |
| PNG Decoder | - | X |

2.9 International

| Kernel | Headless | Starterkit |
|---|----------|------------|
| Input Method Manager (IMM) | - | X |
| Local Services | | |
| English (US) National Language Support only | - | - |
| National Language Support (NLS) | X | X |

2.10 Internet Client Services

| Kernel | Headless | Starterkit |
|---|----------|------------|
| Internet Explorer 6.0 for Windows Embedded CE – Standard Components | - | X |
| Internet Explorer 6.0 Sample Browser | - | X |
| Internet Explorer 6.0 for Windows Embedded CE Components | - | X |
| Internet Explorer Browser Control Host | - | X |
| Internet Explorer HTML/DHTML API | - | X |
| Internet Explorer Multiple-Language Base API | - | X |
| URL Moniker Services | - | X |
| Windows Internet Services | - | X |
| Internet Options Control Panel | - | X |
| Jscript 5.6 | - | X |
| VBScript 5.6 | - | X |

2.11 Security

| Kernel | Headless | Starterkit |
|---|----------|------------|
| Authentication Services | X | X |
| NTLM | X | X |
| Schannel (SSL/TLS) | X | X |
| Credential Manager | X | X |
| Cryptography Services (CryptoAPI 1.0) with High Encryption Provider | X | X |
| Certificates (CryptoAPI 2.0) | X | X |

2.12 Shell and User Interface

| Kernel | Headless | Starterkit |
|--------------------------------------|----------|------------|
| Graphics, Windowing and Events | | |
| Minimal GDI Configuration | X | X |
| Minimal GWES Configuration | X | X |
| Minimal Input Configuration | X | X |
| Minimal Window Manager Configuration | X | X |
| Command Shell | | |

| | | |
|-----------------------------------|---|---|
| Aygshell API Set | X | X |
| Command Processor | X | X |
| Console Window | X | X |
| Graphical Shell | | |
| Standard Shell | - | X |
| Common Controls | | |
| Common Control | X | X |
| Common Dialog Support | X | X |
| Control Panel Applets | X | X |
| Mouse | - | X |
| Network User Interface | X | X |
| Software-based Input Panel (SIP) | - | X |
| SIP for Small Screens | - | X |
| Software-based Input Panel Driver | - | X |
| Touch Screen (Stylus) | - | X |

2.13 Device Driver

| Kernel | Headless | Starterkit |
|--|-------------------|-------------------|
| Audio Driver for TLV320AIC23 ¹⁾ | X ¹⁾ | X ¹⁾ |
| NULL Display Driver (Stub) | X | - |
| Display Driver (SM502) ²⁾ | - | X ²⁾ |
| TFT640x480 | - | X ²⁾ |
| Native keyboard support | - | X |
| CAN driver (ML9620) | X | X |
| LAN 9218 | X | X |
| Asynctac NDIS Driver | X | X |
| PC Card driver (onchip) ¹⁾ | X ¹⁾ | X ¹⁾ |
| Serial interfaces (onchip + SM502) | X | X |
| I2C (SM502) | X | X |
| Storage Devices | | |
| Flash File System | X | X |
| RAM Disk ¹⁾ | X ¹⁾ | X ¹⁾ |
| MMC/SDC ¹⁾ | X ¹⁾ | X ¹⁾ |
| Compact Flash ¹⁾ | X ¹⁾ | X ¹⁾ |
| Touch Driver (TSC2000) | - | X |
| USB Function with ISP1181B | X | X |
| USB Function Client Serial | X | X |
| USB Host (SM502) | X | X |
| PC104-Support | X ¹⁾ | X ¹⁾ |
| HiCOCAN-104-x | X ¹⁾³⁾ | X ¹⁾³⁾ |

1) Only available via HiCO.Nect

2) Contact emtrion for displays and adaptors

3) Also see special registry settings

2.13.1 Display settings

There are two ways to configure the display. The first is by the registry and the second by the settings of the dip switches SW1.

In the listing below is shown how the registry has to be configured if a special display setting should be used within WindowsCE. The display driver reads the parameters for the panel and the screen from the registry and checks the internal database if a valid entry is available. But note, this is a feature for supporting customer

specific displays which have been implemented by emtrion for a customer. If you need a special display support, please contact emtrion.

```

;-----
; Settings for 640x480 at 16bpp
;-----
[HKEY_LOCAL_MACHINE\Drivers\Display\SMIVGX]
  "Bpp"=dword:10
  "CxPanel"=dword:280
  "CyPanel"=dword:1E0
  "CxScreen"=dword:280
  "CyScreen"=dword:1E0
  "RefreshRate"=dword:3C

[HKEY_LOCAL_MACHINE\Drivers\Display\SMIVGX\MONITOR0]
  "Bpp"=dword:10
  "CxPanel"=dword:280
  "CyPanel"=dword:1E0
  "CxScreen"=dword:280
  "CyScreen"=dword:1E0
  "RefreshRate"=dword:3C

```

With the following setting, which is also the default setting of this kernel, the display resolution is controlled by the dip switches.

```

;-----
; Settings for the display
;-----
[HKEY_LOCAL_MACHINE\Drivers\Display\SMIVGX]
  "CxPanel"=dword:0
  "CyPanel"=dword:0
  "CxScreen"=dword:0
  "CyScreen"=dword:0

[HKEY_LOCAL_MACHINE\Drivers\Display\SMIVGX\MONITOR0]
  "CxPanel"=dword:0
  "CyPanel"=dword:0
  "CxScreen"=dword:0
  "CyScreen"=dword:0

```

The first 3 columns show the settings of the dip switch SW1 and the last column shows if this dip switch setting is also supported by the bootloader.

| 3 | 2 | 1 | | Bootloader |
|-----|-----|-----|----------------------------|------------|
| off | off | off | 320x640, 16bpp, 60Hz TFT | yes (8bpp) |
| off | off | on | 640x480, 16bpp, 60Hz TFT | yes (8bpp) |
| off | on | off | 800x600, 16bpp, 60Hz TFT | yes (8bpp) |
| off | on | on | 800x480, 16bpp, 60Hz TFT | no |
| on | off | off | 1024x768, 16bpp, 60Hz TFT | yes (8bpp) |
| on | off | on | | |
| on | on | off | 1280x1024, 16bpp, 60Hz TFT | no |
| on | on | on | | |

By the way, the panel parameters describe the dimensions of the display and the screen parameters describe the size of the frame used by the windows manager. Usually they have the same values but it is possible to use different values.

If the values of the screen parameters are smaller than the values of the panel, the display controller is configured in a way that the screen is zoomed to the panel size. On the other way round the panel shows only a part of the screen. If the mouse cursor touches the borders of the panel, the panel is moved so that now the missing parts of the screen are shown on the panel.

2.13.2 Special hints for using HiCOCAN-104

This feature is available with kernel revision 6 and higher. The board HiCOCAN-104 from emtrion has a PC104 connector. This board can be used together with the board HiCO.nect-openbase which has a PC104 connector as well. The WindowsCE kernel supports the PC104 interface and HiCOCAN-104 as a member of the PC104 bus.

It is possible to use the Onboard CAN controller ML9620 and HiCOCAN-104 simultaneously since the drivers have the same api. But some restrictions have to be obeyed. In addition some special registry settings and jumper settings on HiCOCAN-104 are necessary to distinguish the different CAN nodes.

The first table describes a system where the ML9620 will be used in combination with HiCOCAN-104. Only three additional HiCOCAN-104 boards are supported now.

| Node number | Board number | description | Internal handle |
|-------------|--------------|--|-----------------|
| 0 | 0 | ML9620 | HCN1: |
| 1 | | unused | - |
| 2 | 1 | HiCOCAN-104, first channel, set node jumper to 1 | HCN2: |
| 3 | 1 | HiCOCAN-104, sec. channel | HCN2: |
| 4 | 2 | HiCOCAN-104, first channel, set node jumper to 2 | HCN3: |
| 5 | 2 | HiCOCAN-104, sec. channel | HCN3: |
| 6 | 3 | HiCOCAN-104, first channel, set node jumper to 3 | HCN4: |
| 7 | 3 | HiCOCAN-104, sec. channel | HCN4: |

This table describes also the default setting of the registry.

The listing shows the corresponding settings of the registry. It is very important to check very carefully the jumper settings of the HiCOCAN-104 boards. Please jumper very carefully the settings for the memory base addresses, the ISA bus interrupts and the board numbers.

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\ML9620_0]
  "Prefix"="HCN"
  "Dll"="ml9620.dll"
  "Order"=dword:0
  "Index"=dword:1           ; driver index 1 (HCN1:)
  "SysIntr"=dword:18
  "Priority256"=dword:63    ; = Priority level 100
  "MemBase"=dword:A2800000
  "MemLen"=dword:1000
  "BaudTable"=hex:\
    0a,00,0c,01,31,\
    14,00,0c,01,18,\
    32,00,0c,01,09,\
    64,00,0c,01,04,\
    7d,00,0c,01,03,\
    fa,00,0c,01,01,\
    f4,01,0c,01,00,\
    20,03,06,01,00,\
    e8,03,04,01,00,\
    00,00,0c,01,18

[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HiCOCANPC104_0]
  "Prefix"="HCN"
  "Dll"="HiCOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:2           ; board number 1
                               ; driver index 2 (HCN2:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:28       ; IRQ5 at the ISA bus
  "Priority256"=dword:6b
  "MemBase"=dword:D8000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels

[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HiCOCANPC104_1]
  "Prefix"="HCN"
  "Dll"="HiCOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:3           ; board number 2
                               ; driver index 3 (HCN3:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:23       ; IRQ9 at the ISA bus
  "Priority256"=dword:6c
  "MemBase"=dword:D9000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels

[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HiCOCANPC104_2]
  "Prefix"="HCN"
  "Dll"="HiCOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:4           ; board number 3
                               ; driver index 4 (HCN4:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:25       ; IRQ11 at the ISA bus
  "Priority256"=dword:6d
  "MemBase"=dword:DA000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels
```

The second table describes a system where the ML9620 is NOT used in combination with HiCOCAN-104. In this case up to four HiCOCAN-104 boards are supported.

| Node number | Board number | description | Internal handle |
|-------------|--------------|--|-----------------|
| 0 | 0 | HiCOCAN-104, first channel, set node jumper to 0 | HCN1: |
| 1 | 0 | HiCOCAN-104, sec. channel | HCN1: |
| 2 | 1 | HiCOCAN-104, first channel, set node jumper to 1 | HCN2: |
| 3 | 1 | HiCOCAN-104, sec. channel | HCN2: |

| | | | |
|---|---|--|-------|
| 4 | 2 | HiCOCAN-104, first channel, set node jumper to 2 | HCN3: |
| 5 | 2 | HiCOCAN-104, sec. channel | HCN3: |
| 6 | 3 | HiCOCAN-104, first channel, set node jumper to 3 | HCN4: |
| 7 | 3 | HiCOCAN-104, sec. channel | HCN4: |

The listing shows the corresponding settings of the registry. It is very important to check very carefully the jumper settings of the HiCOCAN-104 boards. Please jumper very carefully the settings for the memory base addresses, the ISA bus interrupts and the board numbers.

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HICOCANPC104_0]
  "Prefix"="HCN"
  "Dll"="HICOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:2          ; board number 1
                          ; driver index 2 (HCN2:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:28      ; IRQ5 at the ISA bus
  "Priority256"=dword:6b
  "MemBase"=dword:D8000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels

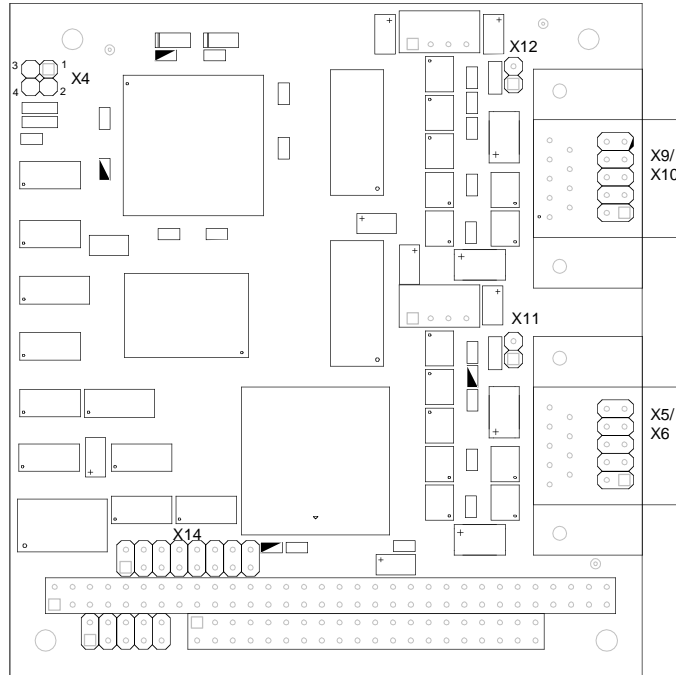
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HICOCANPC104_1]
  "Prefix"="HCN"
  "Dll"="HICOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:3          ; board number 2
                          ; driver index 3 (HCN3:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:23      ; IRQ9 at the ISA bus
  "Priority256"=dword:6c
  "MemBase"=dword:D9000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels

[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HICOCANPC104_2]
  "Prefix"="HCN"
  "Dll"="HICOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:4          ; board number 3
                          ; driver index 4 (HCN4:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:25      ; IRQ11 at the ISA bus
  "Priority256"=dword:6d
  "MemBase"=dword:DA000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels

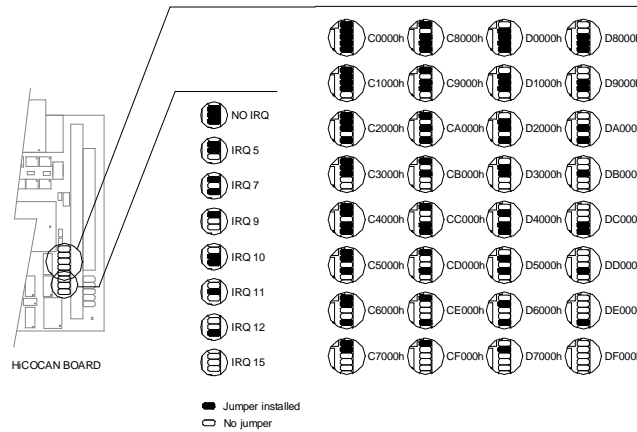
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\HICOCANPC104_3]
  "Prefix"="HCN"
  "Dll"="HICOCANPC104.dll"
  "Order"=dword:0
  "Index"=dword:1          ; board number 0
                          ; driver index 1 (HCN1:)
  "InterfaceType"=dword:1 ; ISA bus
  "SysIntr"=dword:2A      ; IRQ7 at the ISA bus
  "Priority256"=dword:6e
  "MemBase"=dword:DB000
  "MemLen"=dword:1000
  "ChanAnz"=dword:2       ; number of channels
```

2.13.3 Jumpers of HiCOCAN-104

Find below the position of the jumpers on the board.



The following figure shows the jumpers X14 settings of the resources (addresses, interrupts) on the board.



2.13.4 Board Number, X4

| Jumper | open | closed | Jumper | open | closed | Board No. |
|--------|------|--------|--------|------|--------|-----------|
| 1-3 | x | | 2-4 | x | | 0 |
| 1-3 | | x | 2-4 | x | | 1 |
| 1-3 | X | | 2-4 | | x | 2 |
| 1-3 | | x | 2-4 | | x | 3 |

3 Application development

Application development is possible with VS2005. Developing with Embedded Visual C++ is not supported with Windows Embedded CE 6.0 kernel. If there is a need for this, please don't hesitate to contact emtrion GmbH.

VS2005 supports unmanaged and managed code and programming languages like C++, C#, Visual Basic and more are available. Since applications have to be written for embedded devices it is necessary to install an appropriate SDK before. The communication interface for debugging can be Ethernet as well as USB ActiveSync. Sole Ethernet communication without a little help from USB ActiveSync is possible, but needs some special tools instead. For a more detailed description see the next chapters. We recommend a combination of Ethernet and USBF and all the kernels are supporting this.

An evaluation version of VS2005 is available via

<http://www.microsoft.com/windows/embedded/eval/trial.msp>

3.1 VS2005 for Application Development

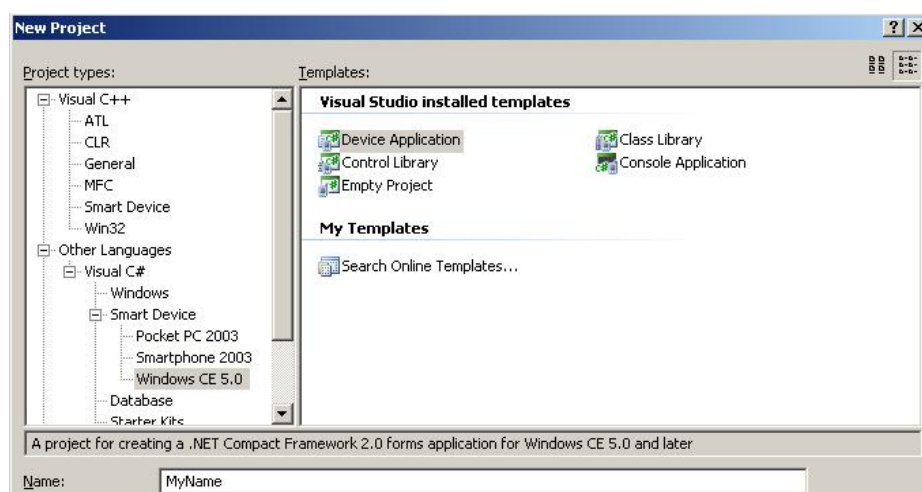
When you using VS2005 for application development based on a specific kernel some hints shall be considered.

- Since Win32-API functions may be omitted in the creation of operating system kernels under Windows® Embedded CE, there is no SDK (Software Development Kit) which suits all Windows® CE operating system kernels. Actually, each operating system kernel has an SDK of its own which must be installed in addition to the development environment.
- Please consult the online help of VS2005 (search for "Requirements") to obtain information on whether a specific function is available in an SDK. If the name of the OS kernel (e.g. HiCOSH7780SKit_CE600 for developer kit kernel) is listed, the function will be supported by this OS kernel.
- The list of parameters of the Win32-API functions is identical with that of the corresponding API functions of Windows® 9x/NT/XP. With some functions certain parameters must be set to 0. For more information, please refer to the online help.

3.1.1 Creating a new Managed Project

The first step is starting a new instance of VS2005.

1. Select **File/NewProject ...** from the Visual Studio menu.
2. In the **NewProject** window select **Visual C#/SmartDevice/WindowsCE5.0**
3. Select the **Device Application** template
4. Name your project **MyName** and click **ok**.



Now you can start with your application development.

3.1.2 Deploying to the HiCO.SH7780

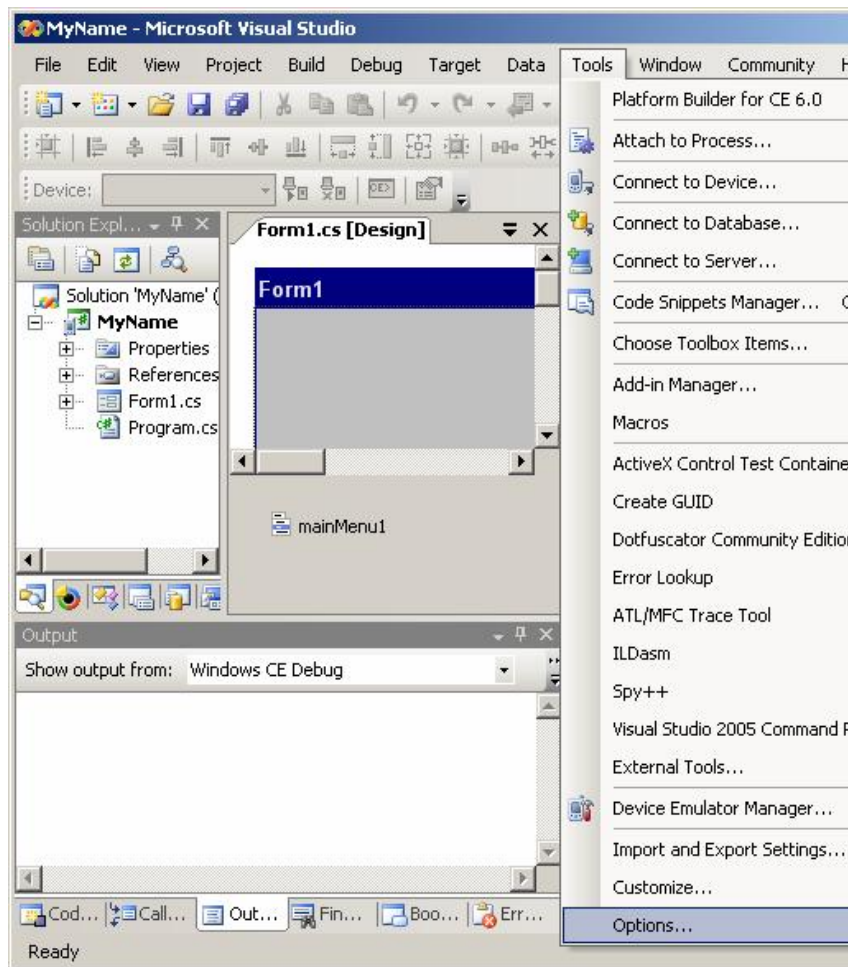
The connection between VS2005 and the target is made by network. For this reason the IP address of the HiCO.SH7780 is necessary. To make the IP address known to VS2005 there are two options.

The first option is to ask manually by means of the function „ipconfig“ on the device. For that “ipconfig” must be executed from the command shell.

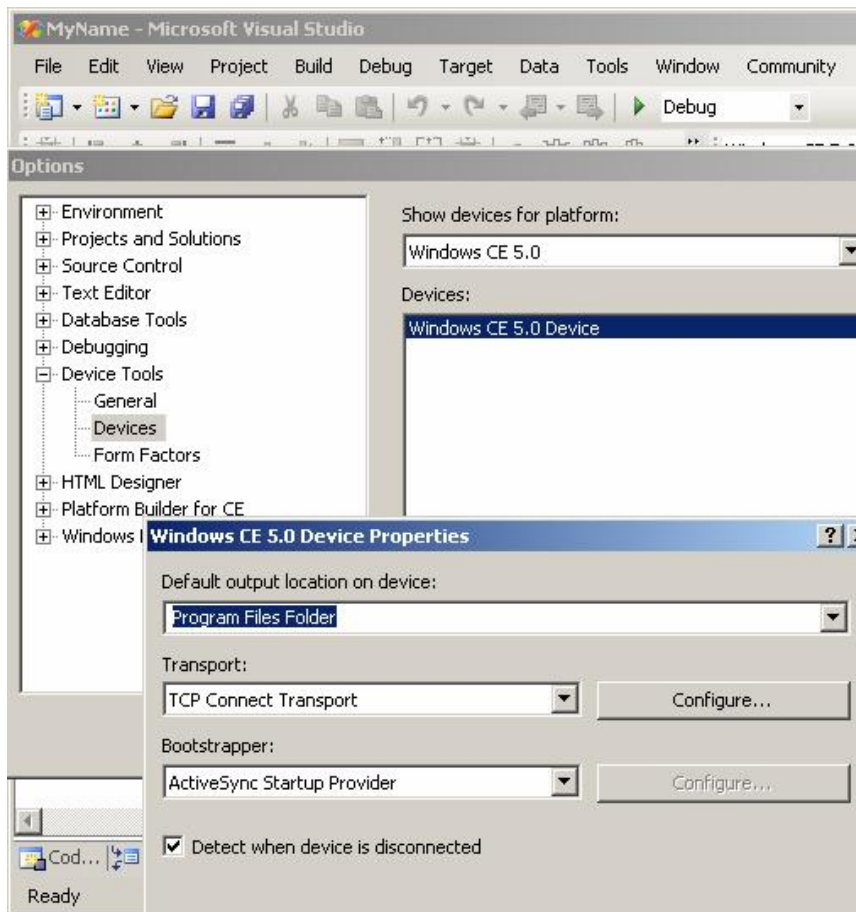
The second is receiving the IP address automatically by an existing USB ActiveSync connection.

After getting this address, the next step is preparing the managed application development environment for deployment.

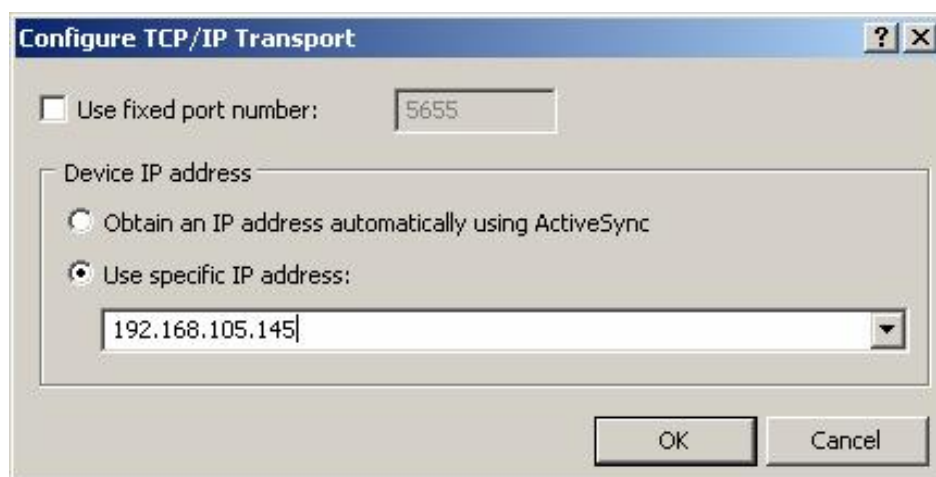
1. In VS2005 select **Tools/Options** from the menu.



2. In the options window, expand the **Device Tools** node and select **Devices**.
3. In the **Show device for platform:** drop down box select **WindowsCE5.0 Device**.
4. Click on **WindowsCE5.0 Device** and select **Properties**.



5. Click the **Configure** button beside the **Transport** drop down box. We are going to configure the TCP Connect Transport.



6. In the case when an USB ActiveSync connection exists between the workstation and the device, the option "Obtain an IP address ..." is to select. In the other case the option "Use specific IP address" is to select and the IP address you got on the device by ipconfig has to be inserted.
7. Click **OK** through all of the dialogs.

Preparing the target HiCO.SH7780:

1. At the command shell, type **ConmanClient2**.
2. Then, type **cmaccept**. You have 3 minutes to establish a connection with your managed application.

Deploying the managed application:

1. Select **Debug/Start Debugging** from VS2005.
2. Select **WindowsCE5.0 Device** from the list of devices in the **Deploy MyName** box and click **Deploy**. VS2005 will deploy several cab files to the device in addition to your application. Your application is executed on the target device.

4 Starting an Application at system start

There are two options to start an application at start up. Using only one option is allowed at any time. In any case, the execution of the explorer is controlled by emtrion's autostart mechanism described in the section below. This means no launch entry may be existed in the registry for the explorer.

To use the autostart functionality the registry must be modified. For it you can use the remote Registry Editor of VS2005. Making the changes persistent the emtrion tool WriteReg.exe is available.

Additional items have to be considered when you want to realise starting your application at system start.

- Folder structures and shortcuts get lost at power off or reset when it have been created at runtime.
- The same as above happens with files that have been copied to the object store of the kernel.
- Changes to the registry also getting lost when it have not been made persistent.
- Due to supporting autostart of applications from a storage device, the kernel is configured with search paths of any supported storage device.

Dependent on the supported storage devices of the Developer Kit the storage folders are specified as following.

- ...\\SD Card for storage device microSD or SD Card
- ...\\MultiMediaCard for storage device MMC
- ...\\Hard Disk for storage device USB stick
- ...\\FlashDisk for storage device flash
- ...\\NAND Flash for storage device nand

4.1 Autostart with launchXX and dependXX

This section describes how you can modify the kernel in order to start your application at system start by the two registry entries launchXX and dependXX.

Using this option, be sure the registry key "**CustomerApp**" in [Emtrion's Autostart Mechanism](#) is empty. In the case of using the explorer by the kernel, the registry key "**FailureOptions**" has to be set to 00000020h for execution.

The application and its own DLLs that shall be started have to be stored in one of the supported storage devices.

- Here are the settings to do on the registry key [HKEY_LOCAL_MACHINE\\init]:
 - Entry 1:
 - Name: LaunchXX, where XX stands for a two-digit number. This number must be between 51 and 99 with the operating system kernel belonging to the starter kit kernel.
 - Type: REG_SZ
 - Value: Name of the file that shall be executed
 - Entry 2:
 - Name: DependXX, where XX must be the **same** two-digit number as specified with entry 1.
 - Type: REG_BINARY
 - Value: Hexadecimal digits that specify the dependency on a previously started program.
- Please also note section "Persistent Registry".

Note

Entry 2 specifies the dependency of the application. An application stored on a mass storage cannot be started until the corresponding entry has been made in the folder structure. It may happen in this context that the registry entries in the [HKEY_LOCAL_MACHINE\\init] key will be processed faster than the mass storages are entered in the folder structure. This must be taken into account in the start sequence.

4.2 Emtrion's Autostart Mechanism

Another possibility to start your application at system start is using the autostart mechanism from emtrion. Therefore an autostart program was written and included into the kernel.

The registry of the Developer Kit kernel includes a launch link to that autostart program and is starting it at system start. Additional, a shortcut of the autostart program is located in the "StartUp" folder of the explorer, so that the start up mechanism of the explorer is also useable. Now, the autostart program itself looks in a specific registry key whether or not what application shall be started. The registry key includes several entries.

But this is not the only task of the autostart program. The autostart program is also controlling the starting of the explorer.

The autostart program looks in **[HKEY_LOCAL_MACHINE\Software\emtrion\AutoStart]** for the entry "**CustomerApp**". The entry is of type REG_SZ and is provided for the filename and if necessary the path of the application that shall be started. When the entry is empty no further action will be done. In the case of a valid entry the autostart program tries to start the application. When a path is not stated, the system searches in the folder "Windows" and in the folders of the supported storage devices of the Developer Kit, mentioned in the introduction of the chapter.

The entry "**AppParameter**" can be found in the same registry key. The entry is also a type of REG_SZ and is provided for the command line parameters of the application.

Optionally, the same registry key may also contain an entry called "**WaitCycles**". This entry, a type of REG_DWORD, lets you extend the time that the autostart program is waiting for the directories to be linked, e.g. a FlashDisk. If the entry is not found, a default value of 100 is taken.

Optionally, the same registry key may also contain an entry called "**FailureOptions**". This entry is also a type of REG_DWORD and is bit-coded. Please set only these bits that are really necessary. The meaning of the individual bits is shown in the table below.

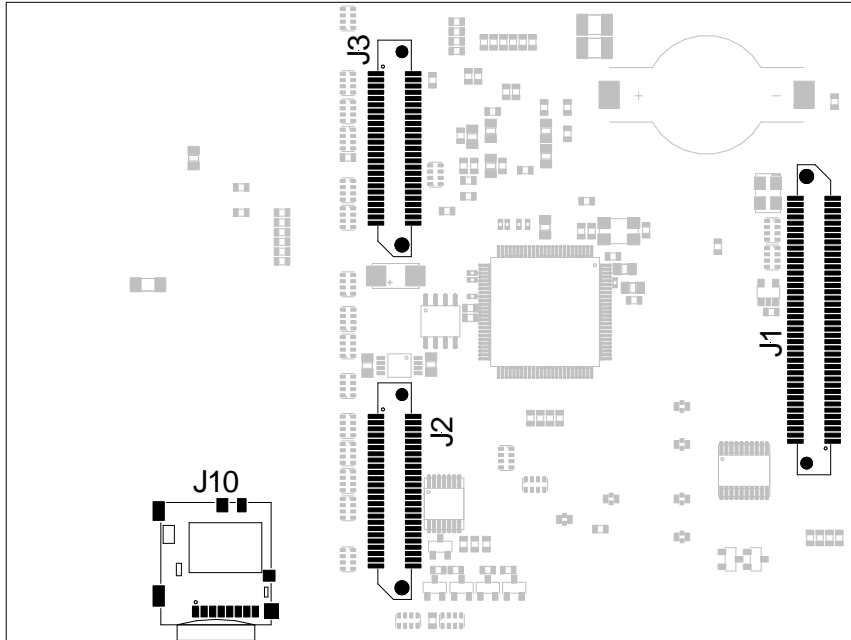
| Bit No. | description |
|---------|---|
| 0 | Setting this bit, a message is displaying in a message box, when the custom application cannot be started. |
| 1 | With this bit you can specify the message showing in the message box. '0': message → "Cannot start the application which was specified to launch at startup." is showing. '1': → more detailed information is showing, like the name and search path of the application. |
| 2 | Setting this bit, the autostart program doesn't exit and is showing continuously the message when the custom application cannot be started. |
| 3 | Setting this bit, the explorer is started when the custom application cannot be started. |
| 4 | Specifies the start up process for the custom application 0: → the explorer is not started and the custom application is started at launch time of the autostart program. 1: → the explorer is started and the custom application is started by the startup mechanism of the explorer |
| 5 | Setting this bit, the explorer will be started in any case. |
| 6 | Setting this bit when the kernel is a headless one. Due to no graphic exists, the console is started instead. |
| 7..31 | Reserved for future extensions |

A further entry is "**Timeout**". On that you can specify the time between two consecutive WaitCycles. The type of Timeout is REG_DWORD.

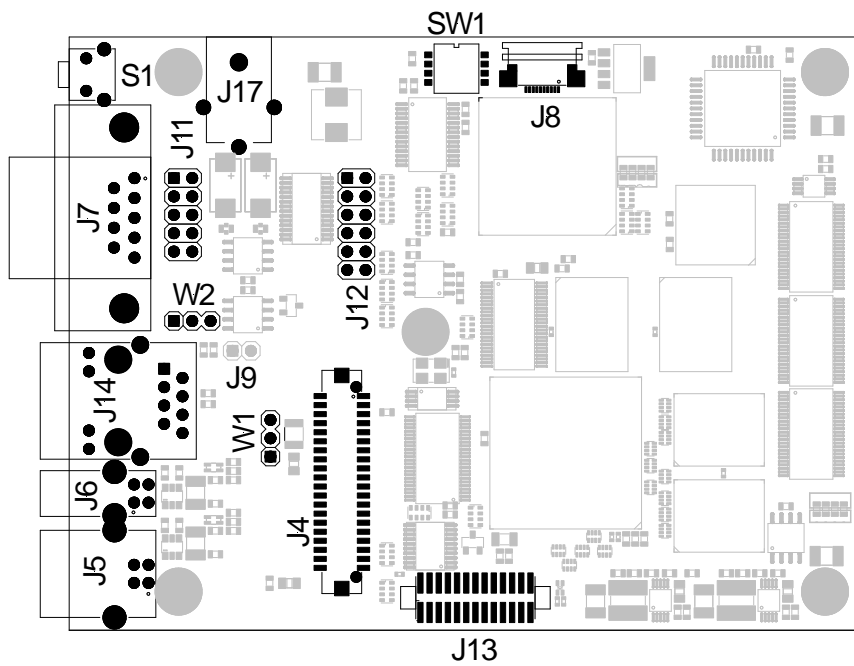
5 Preparing the board for operation

The following description gives a brief overview on what has to be how connected. For more details please have a look at the hardware manual of this board.

Bottom view of the HiCO.SH7780



Top view of the HiCO.SH7780



1. Connect one side of the power cable to the backlight inverter and the other side to the J9 connector on the HiCO.SH7780 board. The red wire has to be connected to pin 1 that is closed to the digit 9 in the figure above.
2. The TFT display has to be connected to J4.
3. The cable of the touch is combined with the cable of the TFT.
4. Connect the USB hub to J6, and then a USB keyboard and or a USB mouse to the hub.
5. Connect the Null Modem cable to the port COM1 on the HiCO.SH7780 board (connector J7).
6. Connect the HiCO.SH7780 to your network with the Ethernet cable. On the starter kit kernel a DHCP client is running.
7. Connect the power supply to the HiCO.SH7780 (J17).
8. Connect the RS232-Adapters to J12. COM2 is the line where pin 1 is located. Pin 1 is marked as square. The red wire of the adapter for COM2 has to be connected to pin 1. The adapter for COM3 must be connected reverse.

5.1 Serial Ports

Three serial ports are available on HiCO.SH7780. Two of the serial ports are based on the UART0 and UART1 of the SM502 and the other one is based on the SCIF0 of the SH7780.

In combination with the HiCO.OpenBase a further serial port is supported. The serial port is based on the ST16C650. But the corresponding driver is deactivated by default.

This driver may only be activated when a corresponding device is present. To enable the driver some of its registry entries have to be modified. The entries are listed below.

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\COM16x50_2]
```

```
"Dll"="" ;"Dll"="16Cx50.Dll"
"IsrDll"="" ;"IsrDll"="isr16Cx50.dll"
"IsrHandler"="" ;"IsrHandler"="ISRHandler"
```

The entries shown after the semicolon enables the driver for loading at system start.

Here is one possibility to modify the registry entries.

1. One is using the remote tool "registry editor"

The empty string of each key name above has to be replaced to its corresponding value that is shown after the semicolon. Following this step the changes have to be made persistent as with the tool `writereg.exe`. When you restart HiCO.SH7780 the driver will be loaded.

The relation of the names to the serial ports from the hardware and software point of view and some more supported features are summarised in the table below.

| | UART0 | UART1 | SCIF0 | ST16C650 |
|----------------------------|--------------------|--------------------------------------|--------------------------------------|----------------|
| Hardware | UART-A | UART-B | UART-C | UART-D |
| Software | COM1 | COM2 | COM3 | COM4 |
| Signals | RS232 | LVTTTL* | LVTTTL* | RS232 |
| Connector on HiCO.SH7780 | D-Sub connector J7 | pin header J12, strip includes pin 1 | pin header J12, strip includes pin 2 | - |
| Connector on HiCO.OpenBase | D-Sub(bottom) J11 | pin header J20, strip includes pin 1 | Pin header J20, strip includes pin 2 | D-Sub(top) J11 |
| HW-Handshake RTS/CTS | yes | yes | yes | yes |
| Hardware Flow | | | | |

| | | | | |
|--------------------|--------|--------|--------|--------|
| Control RTS/CTS | yes | yes | no | yes |
| HW-FiFo | | | | |
| - Receive | 64Byte | 64Byte | 64Byte | 32Byte |
| - Transmit | 64Byte | 64Byte | 64Byte | 32Byte |
| Baudrates | | | | |
| - 1200 | yes | yes | yes | yes |
| - 2400 | yes | yes | yes | yes |
| - 4800 | yes | yes | yes | yes |
| - 9600 | yes | yes | yes | yes |
| - 14400 | yes | yes | yes | yes |
| - 19200 | yes | yes | yes | yes |
| - 38400 | yes | yes | yes | yes |
| - 57600 | yes | yes | yes | yes |
| - 115200 | yes | yes | yes | yes |
| - > 115200 | yes** | yes** | no | yes** |
| | | | | |

(*) Suitable RS232-adapters are delivered by the Starter Kit too

(**) There are higher baudrates possible. While the input clock of some devices differ each other, higher baudrates are strongly customized. When you require higher baudrates please contact Emtrion.

All the serial ports can be accessed by the WIN32-API of WINCE. But with the debug version of the starter kit kernel **COM1** is reserved for debug purposes.

6 Data Exchange between Target and Workstation

The starter kit kernel allows the exchange of data between HiCO.SH7780 and your workstation. For this, Microsoft's ActiveSync program is needed. The latest version is available on the internet at:

<http://search.microsoft.com/search/results.aspx?st=b&qu=ActiveSync&view=en-us>

If you have not yet worked with Microsoft's ActiveSync, we recommend to you a look at section [#Remarks on Using ActiveSync](#).

6.1 Preparing the Desktop Computer

If Microsoft ActiveSync has not yet been installed on your desktop computer, it must be prepared as follows:

Windows 2000:

1. From the Control panel choose "Modem" and install the communications cable between two computers.
2. Start **ActiveSync_45_eng_setup.exe** or **ActiveSync_45_ger_setup.exe**. Both have been placed in the subdirectory **ActiveSync** of the Starter Kit CD.

With all other Windows versions:

1. Start **ActiveSync_45_eng_setup.exe** or **ActiveSync_45_ger_setup.exe** in the **ActiveSync** directory of the starter kit CD.

6.2 Preparing the Windows CE Device

USB function is the default setting for communication with ActiveSync. This communication is sufficient for file transfer and for application debugging.

When you do application development by VS2005 communication via Ethernet is also possible. In this case USB ActiveSync can be used to get the ip-address of the device.

6.3 USB ActiveSync

- Using a USB cable to connect the device to the workstation.
- When not yet powered on, turn on the power of HiCO.SH7780. The Windows® CE OS kernel which is stored in the flash will be executed. The running kernel will initiate an ActiveSync connection at the desktop computer. When HiCO.SH7780 is unknown as USB-Device at the workstation ActiveSync will ask for the corresponding USB-function-driver which has to be installed. The driver is located in the subdirectory **USBFDriver** of the installation.
- When ActiveSync does not start automatically on the workstation then you must perform this manually. Please note, USB has to be selected in the connectivity settings of ActiveSync

HiCO.SH7780 establishes a connection with the desktop computer and possibly prompts you to login and enter your password. Enter the same name and password as you are using for your desktop computer.

Note:

When the connection is **not** automatically established between the HiCO.SH7780 and desktop, you should start the "repllog" program on the HiCO.SH7780 platform manually. To do this, select "Run" from the "Start" menu, then enter **repllog** and click the **OK** button.


The **New Partnership** dialog appears on the desktop computer.

- Select the options as described in section [#Remarks on Using ActiveSync](#).
- After establishing a connection, click **Explore** to open a window. Via this window, you may then exchange files with the HiCO.SH7780 target platform using drag and drop. For this, drag the files onto the desired directory on the HiCO.SH7780.

Note

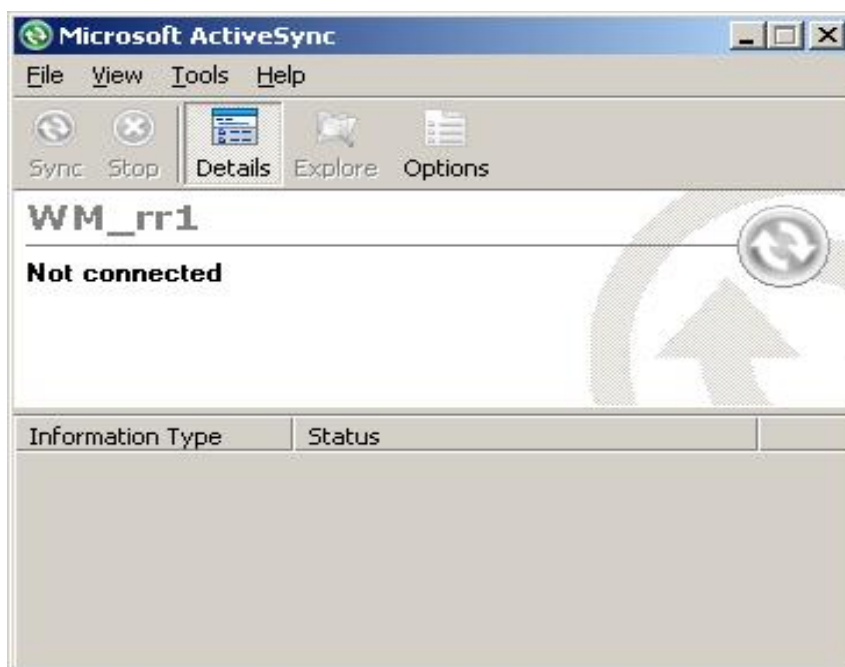
The folders of the HiCO.SH7780 kernel and their contents are created when Windows® CE starts up. If you copy any files to these folders, they are not available any more when Windows® CE is restarted again. This does not apply to folders and files which are directly located in mass storages like the “..\FlashDisk” or the “..\Storage” Card.

These folders are available only if the corresponding devices are detected in the system during the startup phase. A special case are also all subfolders of the "Network" folder. These folders represent drives when a connection via network was established.

To terminate the connection click the  symbol on the Windows® CE's task bar. The connection can be detached in the dialog that appears. Another easy way to stop ActiveSync is just to unplug the USB cable.

6.4 Remarks on Using ActiveSync

After installing ActiveSync, there is a "Microsoft ActiveSync" link in the "Programs" group of the Windows "Start" menu. When pointing to this link, the following window appears:



Please verify that the USB connection is enabled (**Connection Settings** menu item of the **File** menu). If a connection has been established for the first time after power on, the following dialog box displays:



The window shows that no partnership has been established between the Windows® CE computer and the desktop PC.

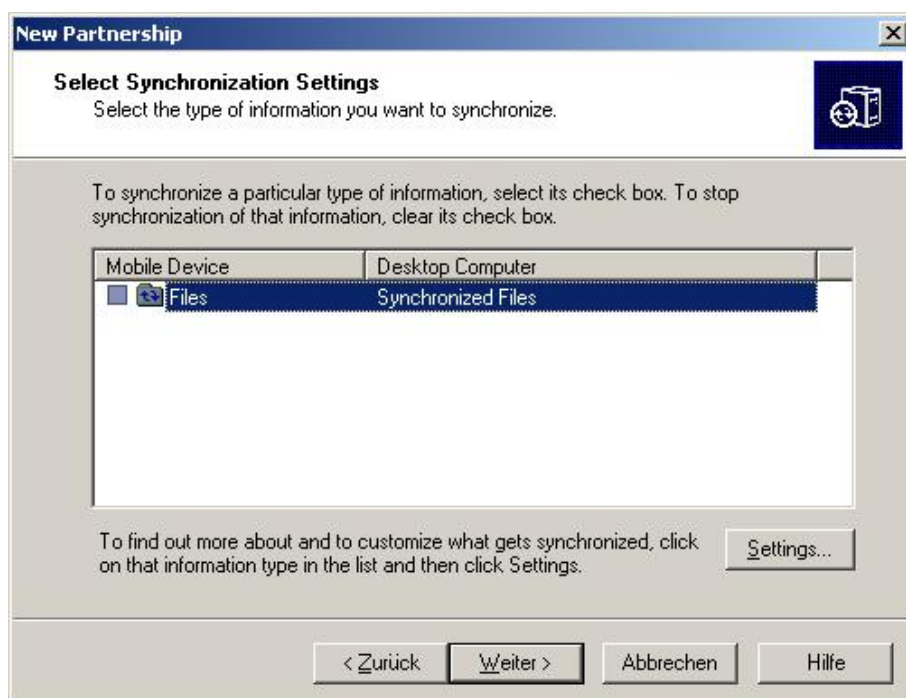
Hint

If a partnership has already been created with a device with persistent registry, the dialogs displayed here do not show up.

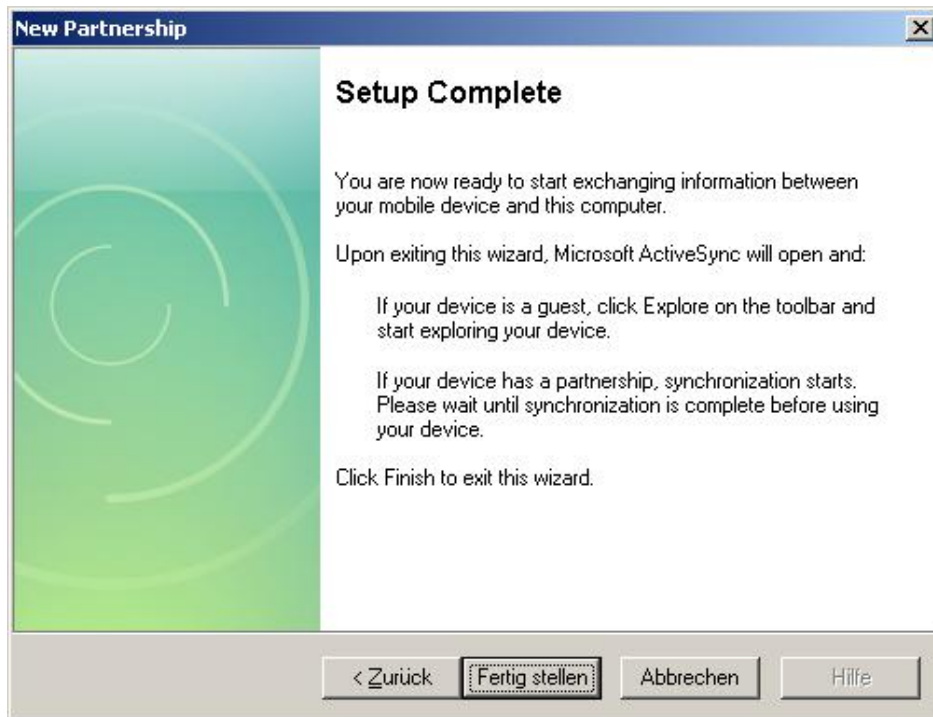
Any of the offered options will be using the USB interface with following property.

"**No**" means that the Windows® CE computer is connected with the desktop computer as a guest only. The connection as a guest is sufficient if you wish to exchange data or to debug an application via an USB connection. "**Yes**" will establish a partnership, which is necessary for data synchronisation.

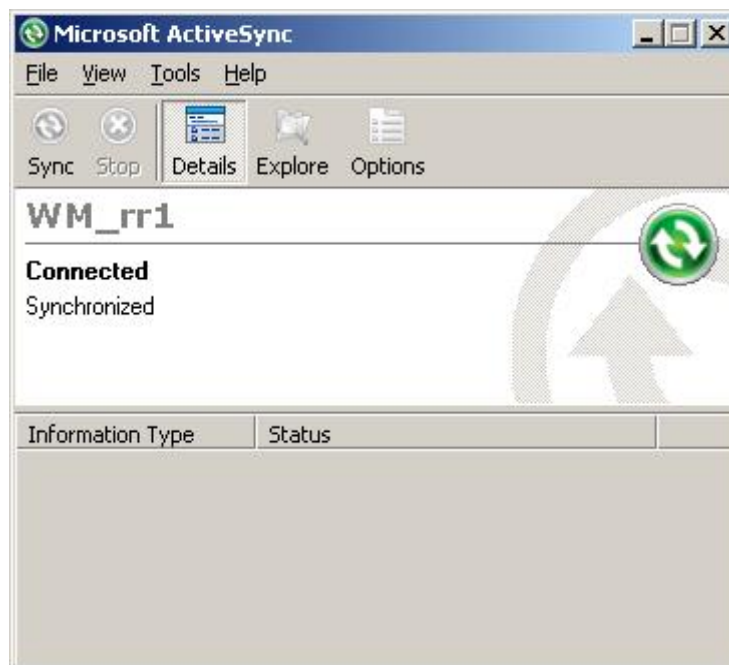
The following dialogs appear.



Accept the default setting or activate "Files" by clicking on the small box left of "Files" and click "Next".



After that, a dialog box appears telling you that the partnership has been successfully established. After clicking the **Finish** button, the ActiveSync program will open on your desktop:



The connection has been established.

Data transfer

Clicking the "Explore" button will bring up a window where you can search the Windows® CE computer.

The data exchange between the Windows® CE computer and the desktop computer can be initiated by dragging a file from the window to another window that does not belong to ActiveSync, or vice versa.

7 Persistent Registry

By default, Microsoft's Windows® Embedded CE does not support a persistent registry. That is, all modifications made to the registry while Windows® CE is running will not be kept when Windows® CE is restarted.

To permanently keep these modifications, two functions are available which have to be implemented by the hardware manufacturer. These functions are already implemented in the HiCO.SH7780. They will be called by the operating system, in order to write the registry onto the Flash in the last 512 kBytes.

Note

Systems with a non-persistent registry have the advantage that a system cannot be damaged due to possibly wrong registry settings made while the system is running.

In order to have the registry write to the flash disk, the "RegFlushKey" function must be called by the application. However, this function should not be called after each modification to avoid performance bottlenecks. Call this function after having made several changes to the registry.

The starter kit's operating system kernels include a tool called WriteReg.exe. This tool calls the RegFlushKey function to make the desired changes. For example, it can be used for system modifications that are to be kept permanently.

"writereg -default" sets the registry to its default values, this means that all changes are lost and the settings of the default registry are used.

A programming sample:

```
HKEY  hRegKey;
DWORD  retWert;

/* open the registry key */
retWert = RegOpenKeyEx(HKEY_LOCAL_MACHINE, _T("Software"), 0, 0, &hRegKey);

if (retWert != ERROR_SUCCESS)
{
    /* Error handling */
}

/*
:
:   Here new entries are made, entries changed or deleted.
:
*/

/* Save registry to Flash */
retWert = RegFlushKey ( hRegKey );
/* Hint: Here a valid handle for the registry key will need to
   be specified (see the online help).
   The complete registry will always be saved to Flash. */
if (retWert != ERROR_SUCCESS)
{
    /* Error handling */
}

/* Close registry key */
retWert = RegCloseKey( hRegKey );
if (retWert != ERROR_SUCCESS)
{
    /* Error handling */
}
```

7.1 Deleting the Persistent Registry

The persistent registry can be deleted in either of the following ways:

- 1.) Using the bootloader
- 2.) Via an application under Windows CE

7.1.1 Deleting the Registry by means of the Bootloader

The persistent registry can be deleted via menu item 4 of the second bootloader menu. For more detailed information, please refer to the user manual of the bootloader.

7.1.2 Deleting the Registry by means of an Application under Windows CE

A Windows CE application can be enabled to use the persistent registry by calling the Win32-API function `KernelloControl`. For more informations about the function `KernelloControl` please have a look at the online help of the platformbuilder.

For this purpose, emtrion has created the following device-specific I/O control code:

```
#define IOCTL_HAL_SETREGISTRYTODEFAULT CTL_CODE(FILE_DEVICE_HAL, 0x810,
METHOD_BUFFERED, FILE_ANY_ACCESS )
```

In this code, `CTL_CODE` is a macro, `FILE_DEVICE_HAL`, `METHOD_BUFFERED` and `FILE_ANY_ACCESS` are specified as `#define` statements. Both the macro and the definitions were defined by Microsoft in the `winiocctl.h` header file. This file is usually linked to applications with the `#include <windows.h>` statement.

The `KernelloControl` is described in the online help of the Platform Builder. At the end of the chapter a short extract of this Win32-API-function is printed.

IOCTL_HAL_SETREGISTRYTODEFAULT

This control code will delete the currently stored persistent registry. As a result, the default registry will be used when the system is started again. However, a reset will **not** be issued automatically. Such a reset can be performed by using the `IOCTL_HAL_REBOOT` control code.

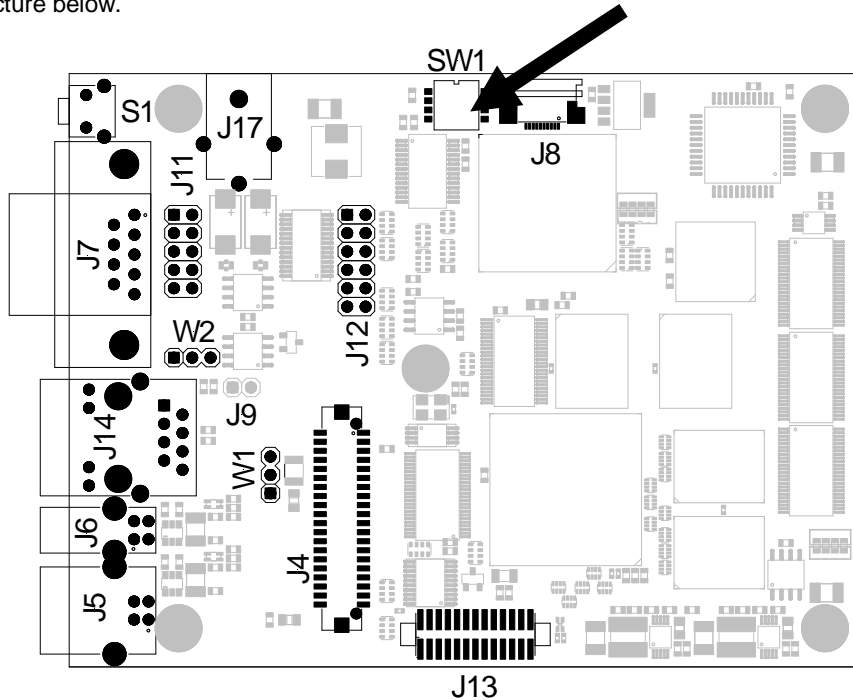
`IOCTL_HAL_SETREGISTRYTODEFAULT` expects the following parameters of the `KernelloControl` function:

| | |
|--|---|
| <code>lpInBuf</code> , <code>nInBufSize</code> , <code>lpOutBuf</code> , <code>nOutBufSize</code> | These parameters will not be analyzed |
| <code>lpBytesReturned</code> | Pointer to a <code>DWORD</code> variable containing the number of bytes returned (here 0) |

The control code deletes the flash sectors (where the registry is saved) by means of the erase function supplied by the bootloader.

8 Replacing the Starter Kit Kernel

For updating a kernel or even the bootloader itself the operation mode of the bootloader has to be changed so that the bootloader menu shown on COM1. For this purpose the switch 4 of SW1 has to be set to ON which is shown in the picture below.



As communication partner for the download, the "KernelDownload" tool from emtrion can be used. The Tool is described in a separate document. The document file is "**KernelDnld_v....pdf**".

Important note

Whenever the operating system kernel is exchanged, the persistent registry should also be deleted by means of the bootloader. If the persistent registry is not deleted, the operating system may not start up correctly, or it may not start at all

8.1 The Bootloader Menu

If the jumper is set to the bootloader position, the following menu is output by the bootloader via COM3. The following parameters are used: 115,200 Baud, 8 data bits, no parity, 1 stop bit, hardware handshake. As opposite device any terminal program may be used, e.g. Windows' supplied Hyperterminal.

```
emtrion GmbH
Bootloader for HiCO.SH7780
Version $ProjectRevision: V1.6 $ from $Date: 2008/05/27 10:04:49Z $
Copyright (c) 2000 – 2008
All rights reserved

Bootloader menu

1.) Execute stored Image
2.) Download Image via serial port and store in Flash
3.) Download Image via serial port, store in SDRAM and execute
4.) Download Image via LAN9218 Ethernet controller and store in Flash
5.) Download Image via LAN9218 Ethernet controller, store in SDRAM and
execute
6.) Extended functionality
7.) Download image over standard TFTP connection

>
```

The individual menu items' functions are detailed in the following section.

When the item "6.) Extended functionality" is chosen, the following (second) menu is output to the serial port:

```
emtrion GmbH
Bootloader for HiCO.SH7780
Version $ProjectRevision: V1.6 $ from $Date: 2008/05/27 10:04:49Z $
Copyright (c) 2000 – 2008
All rights reserved

Bootloader menu

1.) Self test
2.) Set real time clock
3.) Upload stored data
4.) Delete the persistent registry of Windows CE
5.) Download contents for flash sectors (not OS) over serial port
6.) Download contents for flash sectors (not OS) over LAN9218 Ethernet contr.
7.) Reset this device
8.) Show PCI configuration
e.) Return to main menu

>
```

The individual menu items' functions are detailed in the following section.

8.2 Bootloader Menu Items

8.2.1 Execute stored Image

The menu item „Execute stored Image“ executes the operating system (OS) software which is currently stored in the Flash. To do this, the bootloader checks if the OS must be copied to the SDRAM (if it is located for execution from Flash). If this is so, it copies the OS to the SDRAM. Then the bootloader launches the OS by jumping to the starting address of the OS.

8.2.2 Download Image via serial port and store in Flash

The menu item "Download Image via serial port and store in Flash" loads a new image file using the serial port via which the boot menu is output. The new image will be stored in the flash. As transfer protocol the Xmodem protocol with a packet size of 128 bytes and the CRC16 checksum are used. Alternatively, the Xmodem protocol with a packet size of 1024 bytes may be used. The selection is made in the transmitting terminal program.

Example

Starting the serial data transfer on the desktop computer via the Hyperterminal:

- Selection of the menu item with key 2
The bootloader detects the selection and starts the transfer by transmitting a "C". This also appears in Hyperterminal's output window.
- Selection of the menu item "Send File..." in the "Transfer" menu.
- Selection of the file to be transferred and of the 'Xmodem' (or 1K Xmodem) protocol in the dialog and click on "Send".

The transfer will be executed.

8.2.3 Download Image via serial port, store in SDRAM and execute

The menu item "Download Image via serial port, store in SDRAM and execute" loads a new image file via the serial port, via which the boot menu is output. The new image will be stored in the SDRAM. The image file to be loaded must be located such that the addresses specified are in the same address region as the SDRAM. As transfer protocol the Xmodem protocol with a packet size of 128 bytes and the CRC16 checksum are used. Alternatively, the Xmodem protocol with a packet size of 1024 bytes may be used. The selection is made in the transmitting terminal program.

Example

Starting the serial data transfer on the desktop computer by means of the Hyperterminal:

- Selection of the menu item with key 3
The bootloader detects the selection and starts the transfer by transmitting a "C". This also appears in Hyperterminal's output window.
- Selection of the menu item "Send File..." in the "Transfer" menu.
- Selection of the file to be transferred and of the 'Xmodem' (or 1K Xmodem) protocol in the dialog and click on "Send".

The transfer will be executed.

8.2.4 Download Image via LAN9218 Ethernet controller and store in Flash

The menu item "Download Image via Ethernet and store in Flash" loads a new image file via the 100 MBit on-board ethernet controller and saves it in the Flash. The 100 MBit ethernet controller uses the connector on the HiCO.SH7780 board

The required IP address is obtained from a DHCP server in the network. The IP address may also be entered manually as soon as the following message displays in the terminal:

Wait for DHCP or enter new IP address:

The image file to be loaded must have one of the data formats supported (see section [#Supported File Formats](#)).

Note

Both start- and end address of the image's memory region area will also be stored in the flash.

8.2.5 Download Image via LAN9218 Ethernet controller, store in SDRAM and execute

The menu item "Download Windows CE Image via Ethernet, store in SDRAM and execute" loads an image file via the on-board 100 MBit Ethernet controller and stores it in the RAM. The image file to be loaded must be located such that the addresses specified are in the same address region as the SDRAM. After the load process is terminated, the jumps to the yielded start address). The 100 MBit Ethernet controller uses the connector on the HiCO.SH7780 board.

The required IP address is obtained from a DHCP server in the network. The IP address may also be entered manually as soon as the following message displays in the terminal:

```
Wait for DHCP or enter new IP address:
```

The image file to be loaded must have one of the data formats supported (see section [#Supported File Formats](#)).

8.2.6 Extended functionality

The menu item „Extended functionality“ switches to the second menu, which allows additional operations.

8.2.7 Download image over standard TFTP connection

This menu item is similar to the menu item 5. The main difference is that a standard TFTP protocol can be used or transferring an image to the flash. On the host site please call

```
tftp -i <ip-adresse> put nk.bin nk.bin
```

to initiate such an transfer.

8.2.8 Self test

The "Self test" menu item starts a self test of the board. It first detects the size of memory available on the board. Also, the MAC address and both current date and time of the real-time clock are listed. After that, the SDRAM that is not used by the bootloader will be tested. Finally, an internal and external loopback test of the ethernet controller is run. If the optional debug board is found, the self test displays the MAC address of the debug ethernet controller and also tests the debug ethernet controller with an internal and external loopback test. Before the external loop test for the optional debug ethernet controller is executed, you are asked whether the external loopback connector is attached. If you enter No, this test will be skipped.

8.2.9 Set real time clock

This menu item shows both current date and time. If you send the character s via the serial port to the bootloader, you can enter a new data and time.

8.2.10 Upload stored data

This menu item makes it possible to upload some memory regions.

8.2.11 Delete the persistent registry of Windows CE

This menu item clears the last 256 kByte of the on-board flash. Windows CE operating system kernels with a persistent registry, which are created by emtrion, store the persistent registry in this area. If this region is deleted, the OS kernel will use the default registry of the corresponding OS kernel the next time the system is started.

Important note

Use this menu item only if you are absolutely sure that the OS kernel provides a registry that is located in this region. If there are other data in that region (e.g. parts of the OS kernel in use), they will be deleted. As a result, the OS might not run any more and needs to be loaded to the on-board flash again.

8.2.12 Download contents for flash sectors (not OS) over serial port

This menu item is available since bootloader revision 1.9. With this menu item you can change the contents of any sector in the flash except the bootloader sector and the first sector used for the OS. The new data is downloaded over the serial port, via which the boot menu is output. The new data can be provided in a file with one format described in described in the section [#Supported File Formats](#).

Important notes:

- **As the storage location are extracted from the address information in the downloaded files. Please note that there is NO validation of the addresses.**
 - **If the starting address is in the middle of a flash sector the complete flash sector will be erased. Because no data will be saved before, you can loss data in this case.**
-

8.2.13 Download contents for flash sectors (not OS) over LAN9218 Ethernet contr.

This menu item is available since bootloader revision 1.9. With this menu item you can change the contents of any sector in the flash except the bootloader sector and the first sector used for the OS. The new data is downloaded via the 100 MBit on-board Ethernet controller (LAN9218 ethernet controller). The new data can be provided in a file with one format described in the section [#Supported File Formats](#).

The required IP address is obtained from a DHCP server in the network. The IP address may also be entered manually as soon as the following message displays in the terminal:

```
Wait for DHCP or enter new IP address:
```

Important notes:

- **As the storage location are extracted from the address information in the downloaded files. Please note that there is NO validation of the addresses.**
 - **If the starting address is in the middle of a flash sector the complete flash sector will be erased. Because no data will be saved before, you can loss data in this case.**
-

8.2.14 Return to main menu

This menu item causes that the bootloader changes back to the main menu.

8.3 Supported File Formats

The bootloader supports the following file formats:

1. Microsoft Windows CE Binary Image Data Format
2. Motorola 32-Bit Binary Data Format (S3-Records)

The bootloader recognizes the file format by the first data packet.

If the first data packet starts with "B000FFx0A", a type 1 file will be recognized; if it begins with an "S0" string, it is a type 2 file.

Important note

The bootloader detects by the first address entry it finds whether the file contents are located in the Flash or the SDRAM. If the first address entry is in the flash, the bootloader assumes that the file contents were generated for execution in the flash. In this case, all data that are to be stored in the RAM, according to the address input, will be discarded. If data are discarded, a warning will be output in the terminal.

Hint on files in the Motorola format:

Files in the Motorola 32-Bit Binary Data Format may contain the following record types in addition to the S3-Records:

- An S0-Record at the beginning of the file.
- S5-Records; this record type will be ignored.
- An S7-Record that specifies the start address of the OS at the end of file.

9 Emtrion Tools for Windows CE

In order to do various settings or get information, emtrion provides a variety of tools running on the Windows CE computer. Some of the tools can be found in

Start Menu -> Programs -> emtrion Tools.

Writing the Persistent Registry

In order to enable the user to selectively write to the registry, emtrion has implemented a tool called WriteReg in the operating system kernel. When started it writes the persistent registry by calling the Win32-API function RegFlushKey().

When it is called with the additional parameter “-default” the actual persistent registry is deleted and after restarting the OS the default registry is active.

Note

User inputs will be of no significance while the persistent registry is being written.

9.1 Getting the Version of the Operating System Kernel

The RevisionInfo tool implemented in the operating system kernel helps the user determine the currently running operating system kernel.

The following information is displayed in a dialog box:

- Name of the operating system kernel, e.g. HiCOSH7780SKit_CE600
- Version number of the operating system kernel, e.g. V6
- Release date of the operating system kernel, e.g. 2009-09-03

9.2 Setipadr

This tool is not displayed within the folder structure stated above. The tool setipadr is a console application and is equipped with some command line options. You can get current information about the network adapter or you can set some properties like the IP-address or the subnetmask.

Calling the tool without any option all the available network adapters will be listed.

| Option | Value | Meaning |
|--------|--|--|
| | | |
| -a | name of the network adapter | displays the current network settings of the specified adapter |
| -d | specify 0 or 1 | disables or enables the DHCP for the specified adapter in -a |
| -h | No value is required | available parameter are listed |
| -i | specify the IP-address in dot notation | sets the IP-address for the specified adapter in -a |
| -n | specify the subnetmask in dot notation | sets the network mask for the specified adapter in -a |

Example:

Disable the DHCP and set the IP-address and network mask for the network adapter TEST
 setipadr -a TEST -d 0 -i 192.168.110.98 -n 255.255.255.0

10 Licence Agreements

SOFTWARE LICENSE TERMS

WINDOWS EMBEDDED CE 6.0 CUSTOMIZED OEM SOFTWARE DEVELOPMENT KIT for

Windows Embedded CE 6.0 Starterkit for HiCO.SH7780, 02.00.0000

These license terms are an agreement between you and [emtrion GmbH] "Company". Please read them. They apply to the software named above which includes software licensed by Microsoft Corporation or their affiliates ("Microsoft") to Company. The software also includes documentation and any media on which you received it.

The terms also apply to any

- updates,
- supplements,
- Internet-based services, and
- support services

for this software, unless other terms accompany those items. If so, those terms apply.

By using the software, you accept these terms. If you do not accept them, do not use the software. Instead, contact Company [www.emtrion.com] to determine its return policy. If you comply with these license terms, you have the rights below.

- 1. INSTALLATION AND USE RIGHTS.** You may install and use one copy of the software on each computer on your premises to design, develop, test and demonstrate your programs running validly licensed copies of Microsoft Visual Studio 2005, Team Suite, Visual Studio 2005 Team Edition for Software Architects, Visual Studio 2005 Team Edition for Software Developers, Visual Studio 2005 Team Edition for Software Testers, Visual Studio 2005 Professional Edition, Visual Studio 2005 Tools for the Microsoft Office System or Visual Studio 2005 Standard Edition (including evaluation versions).
- 2. SPEECH RECOGNITION.** If the software includes speech recognition component(s), you understand that speech recognition is an inherently statistical process and that recognition errors are inherent in the process. Neither Company nor Microsoft or their suppliers shall be liable for any damages arising out of errors in the speech recognition process.
- 3. SCOPE OF LICENSE.** The software is licensed, not sold. This agreement only gives you some rights to use the software. Company and Microsoft reserve all other rights. Unless applicable law gives you more rights despite this limitation, you may use the software only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the software that only allow you to use it only in certain ways. For more information, see the software documentation. Except and only to the extent permitted by applicable law despite these limitations, you may not:
 - work around any technical limitations in the software;
 - reverse engineer, decompile or disassemble the software;
 - make more copies of the software than specified in this agreement;
 - publish the software for others to copy;
 - rent, lease or lend the software;
 - transfer the software or this agreement to any third party; or
 - use the software for commercial software hosting services.
- 4. BACK UP MEDIA.** You may make one backup copy of the software. You may use it only to reinstall the software.
- 5. TRADEMARKS.** This agreement does not grant you any rights in connection with any trademarks or service marks of Company, Microsoft, or their suppliers.

6. EXPORT RESTRICTIONS. The software is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the software. These laws include restrictions on destinations, end users and end use. For additional information, see www.microsoft.com/exporting.

7. PRODUCT SUPPORT. Contact Company for support options. Refer to the support number provided with the device. Microsoft does not provide support services for components provided to you in this software.

8. HIGH RISK ACTIVITIES. The software is not fault-tolerant and is not designed, manufactured or intended for any use requiring fail-safe performance in which the failure of the software could lead to death, serious personal injury or severe physical and environmental damage ("High Risk Activities"), such as the operation of aircraft or nuclear facilities. You agree not to use, or license the use of, the software in connection with any High Risk Activities. Company customized this software and is responsible for how it operates.

9. ENTIRE AGREEMENT. This agreement, and the terms for supplements, updates, Internet-based services and support services that you use, are the entire agreement for the software and support services.

10. APPLICABLE LAW.

a. United States. If you acquired the software in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.

b. Outside the United States. If you acquired the software in any other country, the laws of that country apply.

11. LEGAL EFFECT. This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the software. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.

12. NO WARRANTIES FOR THE SOFTWARE. The software is provided "as is". You bear all risks of using it. Microsoft gives no express warranties, guarantees or conditions. When allowed by your local laws, Company and Microsoft exclude implied warranties of merchantability, fitness for a particular purpose and non-infringement. If your local laws give you any implied warranties, guarantees or conditions, despite this exclusion, your remedies are limited to Company and its suppliers using reasonable efforts to repair or replace the software to the extent permitted by your local laws. Any warranties you receive regarding the software do not originate from, and are not binding on, Microsoft or their affiliates.

13. LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. You can recover from Microsoft and their suppliers only direct damages up to U.S. \$5.00. Except for any refund Company may provide, you cannot recover any other damages, including consequential, lost profits, special, indirect or incidental damages.

It also applies even if

- anything related to the software, services, content (including code) on third party internet sites, or third party programs, and
- claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent permitted by applicable law.
- Company or Microsoft knew or should have known about the possibility of the damages.

The above limitation may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

Please note: As this software is distributed in Quebec, Canada, some of the clauses in this agreement are provided below in French.

Remarque : Ce logiciel étant distribué au Québec, Canada, certaines des clauses dans ce contrat sont fournies ci-dessous en français.

EXONÉRATION DE GARANTIE. Le logiciel visé par une licence est offert « tel quel ». Toute utilisation de ce logiciel est à votre seule risque et péril. L'entreprise ou Microsoft n'accorde aucune autre garantie expresse. Vous pouvez bénéficier de droits additionnels en vertu du droit local sur la protection des consommateurs, que ce contrat ne peut modifier. La ou elles sont permises par le droit local, les garanties implicites de qualité marchande, d'adéquation à un usage particulier et d'absence de contrefaçon sont exclues.

LIMITATION DES DOMMAGES-INTÉRÊTS ET EXCLUSION DE RESPONSABILITÉ POUR LES DOMMAGES. Vous pouvez obtenir de Microsoft et de ses fournisseurs une indemnisation en cas de dommages directs uniquement à hauteur de 5,00 \$ US. Cette indemnisation ne couvre pas les réductions qu'une entreprise vous ait données. Vous ne pouvez prétendre à aucune indemnisation pour les autres dommages, y compris les dommages spéciaux, indirects ou accessoires et pertes de bénéfices.

Cette limitation s'applique aussi à :

- tout ce qui est relié au logiciel, aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers ; et
- les réclamations au titre de violation de contrat ou de garantie, ou au titre de responsabilité stricte, de négligence ou d'une autre faute dans la limite autorisée par la loi en vigueur.
- Ce que l'entreprise ou Microsoft savaient ou auraient du savoir sur les dommages-intérêts.

Elle s'applique également, même si Microsoft connaissait ou devrait connaître l'éventualité d'un tel dommage. Si votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit, il se peut que la limitation ou l'exclusion ci-dessus ne s'appliquera pas à votre égard.

EFFET JURIDIQUE. Le présent contrat décrit certains droits juridiques. Vous pourriez avoir d'autres droits prévus par les lois de votre pays. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre pays si celles-ci ne le permettent pas.

15. **ADDITIONAL TERMS** (if applicable).

End User License Agreement

Emtrion GmbH Greschbachstrasse 12, 76229 Karlsruhe, Germany (hereinafter referred to as "**Emtrion**") provides to businesspersons which are "Unternehmer" in the sense of Sect. 14 BGB German Civil Code (hereinafter: "the **Customer**") certain Software to be run as embedded software on hardware boards (hereinafter: "the Software") subject to Emtrion's Terms and Conditions:

I. Definitions

1. "**Software**" shall mean the operating software to be embedded into the Hardware, including any documentation hereto.
2. "**Confidential Information**" shall mean any information, know-how and data owned or controlled by Emtrion or by Third Parties license Rights.

II. Limited Use Rights

With respect to Software, Customer is granted a non-exclusive, non-transferable, non-sub licensable, worldwide, license to use or have used the Software for validating or testing. Against an extra fee and an advanced EULA Customer can use Software for projects.

Customer shall not - and shall prevent others from doing so - copy, translate, modify, create derivative works, disassemble, reverse engineer, decompile or otherwise use the Software, except as specifically authorized hereunder in this item, or except provided otherwise by compulsory law. Whenever compulsory law permits the above actions, Recipient shall in any case

- (i) inform Emtrion of these actions in advance, and
- (ii) request support of Emtrion, which Emtrion may offer against reasonable consideration.

All rights granted to Software hereunder shall only apply to the code format to be provided as agreed upon, which is, if not explicitly agreed upon otherwise, solely the binary code to the Software.

Customer is aware that Software may consist of or contains third party software, distributed by Emtrion in altered or unaltered form. The use of such Software may be further restricted by

additional terms and conditions (hereinafter also “**Third Parties License Rights**”), with priority to the terms set out herein.

III. **ADDITIONAL RESTRICTION ON USE**

THE OBJECT CODE, SOURCE CODE AND THIRD PARTY MATERIALS ARE NOT FAULT TOLERANT AND ARE NOT DESIGNED, MANUFACTURED, OR INTENDED FOR USE OR RESALE AS ONLINE CONTROL EQUIPMENT IN HAZARDOUS ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS IN THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH THE FAILURE OF THE OBJECT CODE, SOURCE CODE AND THIRD PARTY MATERIALS COULD LEAD DIRECTLY TO DEATH, PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE.

IV. **Confidentiality Restrictions**

1. **Restrictions.** Without limiting and in addition to the use restrictions under Sect. II and Sect III above, all Confidential Information delivered pursuant to this agreement shall
 - (a) be used by Customer for the Purpose and subject to Section II above only; and
 - (b) be treated by Customer with the same degree of care to avoid unauthorized disclosure to any third party as with respect to Customer's own confidential information of like importance but with no less than the degree of care described in this agreement; and
 - (c) not be distributed, disclosed, or disseminated in any way or form by Customer to anyone except its own employees, who have a reasonable need to know such Confidential Information with respect to the Purpose and who are bound to confidentiality by written agreements not less stringent than under the obligations of this agreement; and
 - (d) be stored in secure areas only that are protected against unauthorized access and are regularly monitored to assure their sufficient security,
 - (e) be treated by Customer in compliance with any additional security guidelines the parties may agree.
2. **Exceptions.** The obligations as per Sect. IV. 1 above shall not apply to any information which the Customer can prove,
 - (a) is at the time of disclosure already in the public domain or becomes available to the public through no breach by the Customer of this agreement, except that Confidential Information shall not be deemed to be in the public domain merely because any part of the Confidential Information is embodied in general disclosures or because individual features, components or combinations thereof are now or become known to the public;
 - (b) is received by the Customer from a third party free to lawfully disclose such information to Customer;
 - (c) was in the Customer's lawful possession prior to receipt from the Discloser as evidenced by written documentation;
 - (d) is independently developed by the Customer without the benefit of any of the Confidential Information as evidenced by written documentation;
 - (e) is approved for release by written agreement of Emtrion;
 - (f) is required to be disclosed to comply with legal mandatory regulations, a judicial or official order or decree, provided that written advance notice of such judicial action was timely given to Emtrion.

V. **Prohibition on Development of Competing Products**

Customer shall at no time during the term of this agreement and hereafter for a term of five (5) years use Confidential Information for the purpose of developing or distributing products which are competing with Emtrion's Hardware as a stand alone product.

VI **WARRANTIES.**

Emtrion makes no warranties, either express or implied, with respect to the object code or source code or with respect to Software or other products or intellectual property obtained from Third Parties. Emtrion expressly disclaims any such warranties, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.