

Pfaadt Software

Ingenieurbüro Hartmut & Klaus Pfaadt GbRmbH

Papiergaessle 7d
D-79183 Waldkirch
Tel. +49 7681 / 493 997 2
Fax +49 7681 / 493 997 8

info@pfaadtsoft.de
www.pfaadtsoft.de

Embedded Software, Windows CE.NET,
BIOS Adaptations, Driver Development,
Realtime Programming, Applications

User Documentation for the Bootloader Toolkit for WindowsCE

Copyright © 2003 Pfaadt Software

Author: Klaus Pfaadt
Date: 11/21/2003
Revision: 1.3

Introduction

Since version 4.1 Microsoft offers a boot loader that allows you to load and start a WindowsCE image from a disk. This is an alternative to the older DOS utility LOADCEPC, that requires a DOS license. The source codes for this loader and the accompanying SYS-command can be found in the directory %_WINCEROOT%\PUBLIC\COMMON\CSP\I486\BIOSLOADER.

However the use of this loader suffers from certain implementation problems that makes handling difficult or sometimes impossible. In addition the loader is kept quite simple and often users wish to have a graphical logo displayed while WindowsCE gets loaded.

With our Bootloader SDK these problems have been fixed and you will get some new and hopefully interesting features.

Installation on the target system

First you must SYS your card with e.g. "CESYS C:". Next, modify the parameters in BOOT.INI according to your needs and copy it onto the disk. Finally, copy BOOT.PCX and the WindowsCE image on your disk.

Restrictions of the Demo Version

The Demo Version is fully functional. However you can not replace BOOT.PCX by your own graphic and starting the loader without the graphic is not possible. As soon as the WindowsCE image has been loaded, the loader waits for any user key press before WindowsCE gets started.

Components delivered in the Full Version

The full version doesn't have the restrictions of the demo version. You will receive the boot loader in full source code and you can compile it in your Platform Builder environment. This allows you to make your own additions to the boot loader. The utilities CESYS.COM and PCXINFO.EXE will still be provided as binaries. In addition, you will get a description what needs to be changed in the WindowsCE kernel to use the registry override feature.

Features and Fixes

The SYS Command

The SYS command of WindowsCE uses BIOS INT13 calls to read from or write to the boot device. This means that you can only write to fixed disks that get managed by the BIOS or a BIOS extension. You can not write to any drive that is installed by a DOS device driver. For example you can not SYS an ATA card that is inserted into your PCMCIA slot. Since most embedded designs are using CompactFlash, software installation typically takes place in the PCMCIA slot of a notebook.

We are using a special SYS-command which is based on FreeDOS that allows you to write on any DOS compatible disk device.

The Boot Block

The boot block is written by the SYS command. With the existing boot block provided in WindowsCE, all blocks of the boot loader must reside in continuous blocks. This means that you must format your drive before you can SYS it. The boot block also expects a fixed size of your boot loader code.

The boot block of our SYS command is more flexible. It tries to detect the boot loader in the root directory and loads it, independent of its size or whether the blocks are continuous.

Graphical Logo

After SYSing the drive you can put a PCX file boot.pcx onto your disk. The boot loader will automatically display the picture with the appropriate resolution. The PCX graphic may have the formats 640x480x8, 800x600x8, 1024x768x8 or 1280x1024x8. We have chosen the PCX format instead of the common BMP format because the graphic files are relatively small. The VGA BIOS of your video controller must support the VESA/VBE standard in order to display the pictures properly.

Progress Bar

After the logo has been displayed, the boot loader may display a progress bar that visualizes the loading process. The progress bar can be easily configured to have a specific color and coordinates. We provide a DOS based tool that make this configuration easy to do.

Multiple-Region Images

A WindowsCE image can be split into several smaller images. This allows you, for example, to move your WindowsCE kernel and your applications into different files. To update your applications, you would only need to send the application image to your customer. This loader contains support for these kind of images. You can specify the CHAIN.LST file in order to load an image from multiple files.

Registry Override

A lot of drivers and system components of WindowsCE are using registry keys for system configuration. If one of these registry keys will be changed by accident, your WindowsCE system may not come up. This boot loader gives you the possibility to disable the persistent registry on power up by pressing the LEFT SHIFT and CTRL key simultaneously. In such a case, the ROM defaults will be used instead and the persistent (or hive-based) registry will be reset to the ROM defaults. To visualize this special mode, the boot loader will print the message "Registry Override Active!" on a specific position.

VESA 2.0 based Flat Driver Support

WindowsCE offers a generic video driver (ddi_flat.dll) that runs with all VESA/VBE 2.0 compatible VGA cards. Typically all PCI VGA cards support this standard. This driver requires that the chosen video mode has already been set before WindowsCE gets started and that some video mode informations have been provided through the BOOTARG structure passed to WindowsCE.

Our boot loader allows you to specify a video mode in BOOT.INI which will be set before WindowsCE gets started. You can use the PCXINFO utility on your target system to find out more about the supported VESA modes of your display adapter.

Configuration Files

The old LOADCEPC offered several command line parameters that could be used to pass some informations to WindowsCE. This allows you to build a generic WindowsCE image whose behaviour can be modified by parameters. The original WindowsCE boot loader does not provide these possibilities. It uses fixed parameters that can not be changed after compilation.

Our boot loader tries to load and analyze the file BOOT.INI. If this file is not present, then default values are used. The parser allows you to use comments ("#") and to specify values either in decimal or hexadecimal (prefix "0x").

Alternate Boot Configuration

By pressing RIGHT SHIFT and CTRL simultaneously immediately at the start of the boot loader, an alternate configuration file ALTBOOT.INI can be loaded. This allows you to load a different image for in-the-field services. This may either be a different WindowsCE image for recovery that allows to replace or repair the standard NK.BIN or the ethernet boot loader EBOOT.BIN.

BOOT.INI switches

Whenever the boot loader gets started, it will look on the boot device for the configuration file BOOT.INI. BOOT.INI contains several environment switches, that will be passed over to WindowsCE. BOOT.INI will not be installed automatically with CESYS.COM. You should copy it seperately to the disk.

IMAGE=<filename>

The filename of your WindowsCE image. This is typically either NK.BIN or EBOOT.BIN. (default: NK.BIN)

If the filename is "CHAIN.LST", the bootloader will load a multi-region image.

PBAR_ENABLE

Enables the progress bar. (default: disabled)

PBAR_X=<x-position>

The column of the upper left corner of the progress bar. (default: 0)

PBAR_Y=<y-position>

The row of the upper left corner of the progress bar. (default: 0)

PBAR_COLOR=<color-index>

The color index of the progress bar. (default: 0)

OVERRIDE_ENABLE

Enables the Registry Override option. (default: disabled)

OVERRIDE_X=<x-position>

The column of the upper left corner of the registry override message. (default: 0)

OVERRIDE_Y=<y-position>

The row of the upper left corner of the registry override message. (default: 0)

OVERRIDE_COLOR=<color-index>

The color index of the "Registry Override Active!" message. (default: 0)

COM_PORT=<port number>

Select a COM port for serial debugging. A 0 means, that serial debugging is disabled and the port is available for WindowsCE applications. (default: 0)

COM_BAUD=<baud rate>

Select a baud rate for serial debugging. Possible values are:
9600/19200/38400/57600/115200

ETH_ADAPTER=<adapter type>

Select the ethernet adapter type used for kernel debugging. Possible values are:

0=SMC9000/1=NE2000

If this key is not specified, support for kernel debugging is disabled.

(default: disabled)

ETH_IO=<address>

Define the I/O address used by the ethernet card. (default: 0)

ETH_IRQ=<interrupt>

Define the interrupt used by the ethernet card. (default: 0)

ETH_IP=<ip address>

Define the ip address used for the ethernet card, e.g. 192.168.0.13. (default: 0.0.0.0)

VIDEO_MODE=<mode>

Define the video mode that will be used by WindowsCE. The loader will initialize this mode before WindowsCE gets started. Any values below 0x100 will be considered as standard video modes while modes above 0x100 are considered as VESA video modes. If this key is not specified, there will be no video initialization. Use the PCXINFO tool to find out more about supported VESA modes.

(default: 0 "disabled")

VERBOSE

Displays various debug informations on COM1 (38400,n,8,1). (default: disabled)

#

This starts a comment until the end of line.

PCXINFO

PCXINFO.EXE helps you to configure BOOT.INI correctly. The syntax of this utility is as follows:

PCXINFO <filename>

This utility must be started in a DOS or Windows98 environment. Windows2000 does not support VESA modes which is required to run this utility.

After PCXINFO has been started, the following menu will show up:

```
PCXINFO VERSION 1.0  
Copyright (c) 2003 Pfaadt Software
```

```
PCX File:    BOOT.PCX  
Resolution: 800x600x8  
Video Mode: 0103
```

```
Progress Bar: HIDDEN  
Position: (0/0)   Color: 000
```

```
Registry Override Message: HIDDEN  
Position: (0/0)   Color: 000
```

```
F1:    Show Picture and determine Progress Bar position  
F2:    Edit Progress Bar Color  
F3:    Edit Color of Registry Override Message  
F4:    Display VESA Mode Informations  
ESC:   Exit Program
```

The items "Position" and "Color" in the sections "Progress Bar" and "Registry Override Message" will be updated to the values determined in the Picture Viewer (F1) and Color Table Viewer (F2 and F3). These values can be used later in BOOT.INI to fill out the fields "PBAR_X", "PBAR_Y" and "PBAR_COLOR" as well as "OVERRIDE_X", "OVERRIDE_Y" and "OVERRIDE_COLOR".

After pressing the F1 key, the Picture Editor comes up:

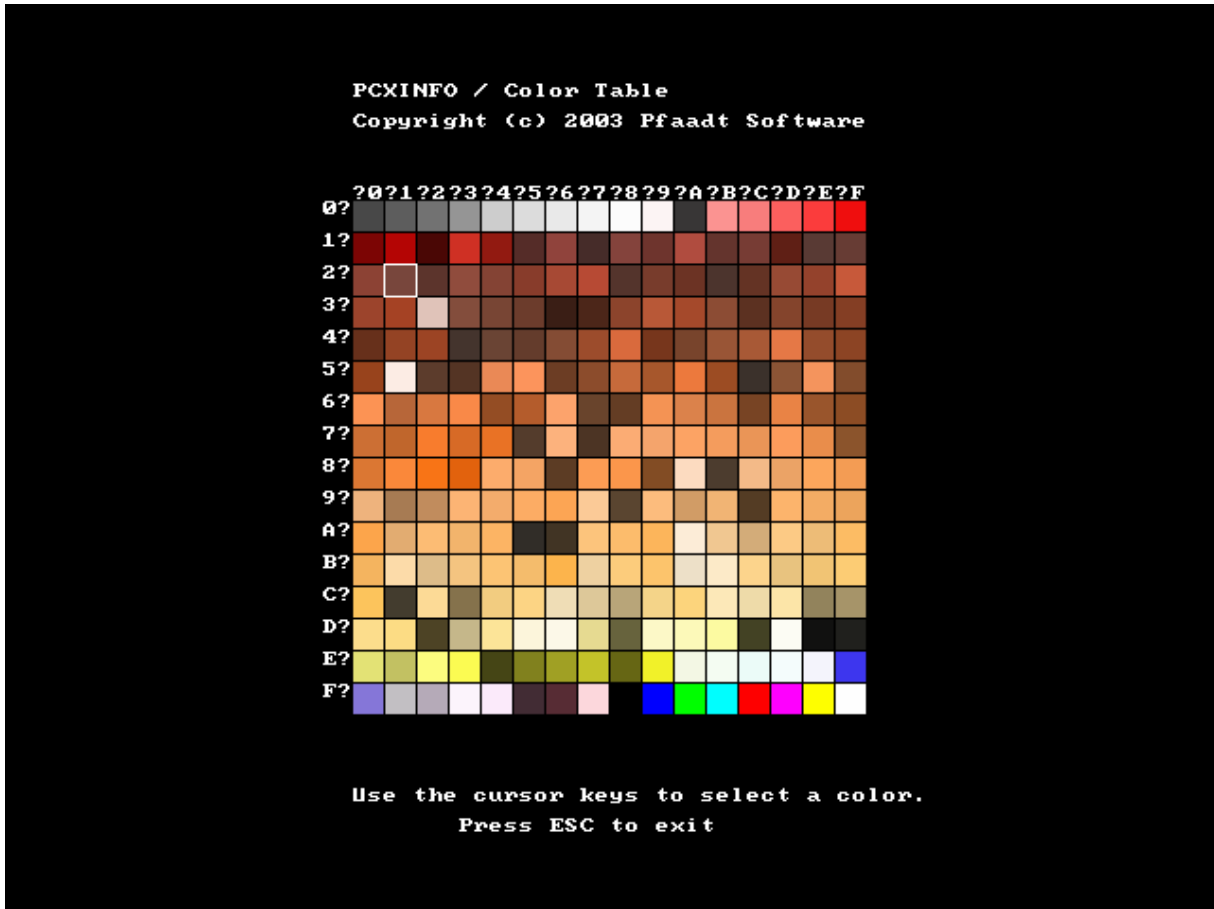


You can use the cursor keys to move the cross to the position, where you want your progress bar to be displayed. You can move the cross faster by pressing the SHIFT key simultaneously.

When you have reached your position, press F1 and the progress bar will be displayed in the selected color. Press F2 to draw the "Registry Override Active!" message.

You can leave this editor by pressing ESC or RETURN.

By pressing F2 or F3, you get into the Color Editor where you select the color of the progress bar or “Registry Override Active!” message:



The graphics displays the PCX color table in rows and columns. You can receive the hexadecimal value of the color index by adding the row and column values. The actual color is highlighted by a white frame. You can move the frame by using the cursor keys.

After you have found your color, you can leave this editor by pressing ESC or RETURN.

When you are pressing the F4 key, PCXINFO dumps all available VESA video modes. You can use these informations to fill out the VIDEO_MODE option appropriately.

Vesa BIOS Version: 3.0

```
101: 640x 480x 8 (0/0/0) #1 D4000000 Supported
110: 640x 480x16 (5/5/5) #1 D4000000
111: 640x 480x16 (5/6/5) #1 D4000000 Supported
112: 640x 480x32 (8/8/8) #1 D4000000 Supported
100: 640x 400x 8 (0/0/0) #1 D4000000 Supported
102: 800x 600x 4 (0/0/0) #4 00000000
103: 800x 600x 8 (0/0/0) #1 D4000000 Supported
113: 800x 600x16 (5/5/5) #1 D4000000
114: 800x 600x16 (5/6/5) #1 D4000000 Supported
115: 800x 600x32 (8/8/8) #1 D4000000 Supported
105: 1024x 768x 8 (0/0/0) #1 D4000000 Supported
116: 1024x 768x16 (5/5/5) #1 D4000000
117: 1024x 768x16 (5/6/5) #1 D4000000 Supported
118: 1024x 768x32 (8/8/8) #1 D4000000 Supported
107: 1280x1024x 8 (0/0/0) #1 D4000000 Supported
119: 1280x1024x16 (5/5/5) #1 D4000000
11A: 1280x1024x16 (5/6/5) #1 D4000000 Supported
11B: 1280x1024x32 (8/8/8) #1 D4000000 Supported
10A: 132x 43x 4 (0/0/0) #4 00000000
```

Press any key to continue.

This option makes sense, if you run PCXINFO in your target system. The boot loader requires at least VESA BIOS Version 2.0. Please verify that this is supported on your target.

For each VESA mode available, PCXINFO dumps the following fields:

- Mode
- Resolution (X-Res x YRes x BitsPerPixel)
- Color-Resolution R/G/B
- Number of Planes
- Base Address of Frame Buffer
- Status whether this mode is supported by DDI_FLAT.DLL

The DDI_FLAT driver only supports 16 bit modes with a color resolution of 5/6/5. A linear frame buffer must be available and the number of planes must be 1.