HENGSTLER

Thermal Printer Family
eXtendo

Linux Driver
S684018-R3-V1.04e-BETA-2018AUG14

Installation Guidelines
Operation Manual
Driver Synopsis
Programming Manual
Printer Tools

E

HENGSTLER GmbH

Uhlandstr. 49

78554 Aldingen / Germany

Tel. +49 (0) 7424-89 0

Fax +49 (0) 7424-89 500

eMail: info@hengstler.com

www.hengstler.com

Safety Precautions

- Please read and understand these specifications thoroughly before using the printer driver in combinations with your application. Please keep the specifications carefully in a place where they may be easily consulted when the printer driver is used.

- Please do not modify this printer driver as this may cause unpredictable behavior.

- The printer driver is not intended to be installed in computer systems or in combination with devices, such as those used in life-support medical equipment, undersea relays, and aerospace applications or for nuclear power control, in which extremely high reliability is required. If you are considering such applications, please consult our customer service department.

- There is a general possibility of component failure. Every effort has been made to improve product quality but such failures cannot be completely excluded. Please assume that such failure may occur before using the printer driver.

- Take care that the contents of this document matches the version of your printer driver.

We would urge that these specifications should be thoroughly understood to assure that the printer driver is used safely in your company or associated organization. Please indicate or describe in your products and in the user manuals those items, which are related to the prevention or avoidance of danger and draw these to the attention of the eventual client (the user).

Any approval or order based upon this manual will not be accepted by HENGSTLER. Any approval must be made upon samples of the product.

Contents

Document History

Revision	Mod. Nr.	Status	Description
1	4 290908 TKA	Closed	Initial version: created for eXtendo CUPS Linux driver R2-V1.01-2008SEP30.
2	4 150709 LT1	Closed	Version: created for eXtendo CUPS Linux driver R2-V1.01-FINAL 2009JUL15.
3	4 201211 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.02a-BETA 2011DEC20.
4	4 070515 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.02d-BETA 2015MAY07.
5	4 130217 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.03a-BETA 2017FEB13.
6	4 230318 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.04a-BETA 2018MAR23.
7	4 310718 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.04d-BETA 2018JUL31.
7	4 140818 LT1	Closed	Version: created for eXtendo CUPS Linux driver R3-V1.04e-BETA 2018AUG14.

Comments

Revision	Comments
-	-

Driver History

Version	Date	Major Modification
R1-V1.00	2008MAR14	1st release of eXtendo CUPS driver for Linux
R2-V1.01b	2008SEP30	2st release of eXtendo CUPS driver for Linux
R2-V1.01	2009JUL15	3rd release of eXtendo CUPS driver for Linux
R3-V1.02a	2011DEC20	4th release of eXtendo CUPS driver for Linux
R3-V1.02d	2015MAY07	6th release of eXtendo CUPS driver for Linux
R3-V1.03a	2017FEB13	7th release of eXtendo CUPS driver for Linux
R3-V1.04a	2018MAR23	8th release of eXtendo CUPS driver for Linux
R3-V1.04d	2018JUL31	9th release of eXtendo CUPS driver for Linux
R3-V1.04e	2018AUG14	10th release of eXtendo CUPS driver for Linux

Applicable Documentation

<u>Following documentation is applicable to this document:</u>

[1]   Extendo Emulation Command Set Reference Manual
      Part-Nr: Hengstler D684112

[2]   Extendo Firmware Release Note
      Part-Nr: Hengstler D684117

[3]   Extendo Linux Driver Release Note
      Part-Nr: Hengstler D684126

Current Driver

The current eXtendo CUPS printer driver with part number S684018 has version "**R3-V1.04e-BETA**" and contains various sub-components with own version history.

Abbreviations

Abbreviation	Description
<cd>	Absolute path to the installation CD–ROM (e.g. "D:")
<root>	Sub-directory on <cd> that contains eXtendo driver package.
SDK	*Software Development Kit*, a programming package that enables a programmer to develop applications for a specific platform. Typically an SDK includes one or more APIs, programming tools, and documentation.
API	*Application Program Interface*, a set of routines, protocols, and tools for building software applications. A good API makes it easier to develop a program by providing all the building blocks. A programmer puts the blocks together.
Shell	Terminal windows in Linux (konsole)
CUPS	**C**ommon **U**nix **P**rinting **S**ystem

1   Introduction

This reference manual describes the eXtendo CUPS printer driver for Linux which has Hengstler part number S684018. The eXtendo printer driver is suitable for usage with Hengstler thermal printers: X56(203 dpi) and X80(203 dpi).

This manual is divided into the following sections:

- Installation Guidelines

- Operation Manual

- Driver Synopsis

- Programming Manual

- Printer Tools

2 Installation Guidelines

2.1 Introduction

This chapter describes the basic installation and un-installation of the CUPS printer driver within the Linux operating system. This CUPS printer driver can be used with X56 and/or X80 printers using the USB printer port.

Following subjects are described throughout this chapter:

- The CUPS printer driver distribution

- Installing the printer driver

- The printer driver files

- Un-installing the printer driver

- Re-installing the printer driver

Usage of this driver under operations other than Linux is prohibited.

2.2 The CUPS printer driver distribution

The eXtendo printer driver package contains out of several sub-directories containing driver files and tool. In the adjacent text, the root directory of this driver package is called **"<root>".**

Directory **"<root>"** contains following files:

- **.\ S-684-018-Extendo-LinuxDriver-R3-V1_04e-BETA-2018AUG14.zip**
  This is the compressed representation of the eXtendo printer driver (ZIP)

- **.\README.TXT**
  Information about eXtendo Linux driver package

- **.\install_x86**
  Instalation script under Linux **x86** systems

- **.\install_x64**
  Instalation script under Linux **x64** systems

- **.\install_arm**
  Instalation script under Linux **arm** systems

- **.\uninstall_x86**
  Un-Installation script under Linux **x86** systems

- **.\uninstall_x64**
  Un-Installation script under Linux **x64** systems

- **.\uninstall_arm**
  Un-Installation script under Linux **arm** systems

\

Directory "**<root >**" contains following sub-directories:

- **.\Driver\x86   or   .\Driver\arm   or   .\Driver\x64**

  This directory contains all printer driver files for installation under Linux **x86**, **x64** and **arm** :

.\hengstlerx56.ppd	= Information file used for installation with eXtendo X56 printers (PPD)
.\hengstlerx80.ppd	= Information file used for installation with eXtendo X80 printers (PPD)
.\hengstlerx56.ppd.gz	= Information file used for installation with eXtendo X56 printers (PPD) in gzip format
.\hengstlerx80.ppd.gz	= Information file used for installation with eXtendo X80 printers (PPD) in gzip format
.\rastertoextendo	= Raster file for eXtendo printers (Driver)

- **.\Api\x86   or   .\Api\arm   or   .\Api\x64**

.\Exo_Api.h	= Include file specifying the eXtendo Application Programming Interface
.\libExoApi.so.1.0	= Application Programming Interface/API (Linux library for **x86/x64/arm**)
.\libExoTb.so.1.0	= ToolBox library (Linux library for **x86/x64/arm**)

- **.\Documents**

  This directory contains all relevant documents of eXtendo driver :
  .\S-684-018-Extendo-LinuxDriver-ReleaseNote.pdf
  = Release note for the eXtendo printer driver (PDF)
  .\D-684-140-Extendo-LinuxDriver-Manual.pdf
  = Manual for the eXtendo printer driver (PDF)

- **.\Tools\x86   or   .\Tools\arm   or   .\Tools\x64**

  This directory contains various eXtendo driver related tools:

.\Exo_InquiryTool	= eXtendo inquiry tool for different printer packaged
.\Exo_Demo_FormFeed	= eXtendo demo application for form feed
.\Exo_Demo_InquiryStatus	= eXtendo demo application for inquiring printer status package
.\Exo_Demo_PageLength	= eXtendo demo application for settings page length
.\libExoApi.so.1.0	= Application Programming Interface/API (Linux library for **x86/x64/arm**)
.\libExoTb.so.1.0	= ToolBox library (Linux library for **x86/x64/arm**)
.\README.txt	= Instruction to demo applications

- **.\Sources**

.\Exo_Api.h	= Include file specifying the eXtendo Application Programming Interface
.\Exo_Demo_FormFeed.c	**=** Source code of demo application for setting form feed
.\Exo_Demo_InquiryStatus.c	**=** Source code of demo application for inquiring printer status
.\Exo_Demo_PageLength.c	**=** Source code of demo application for page length settings
.\README.txt	**=** Instruction for demo application on how to compile sources and how to run executables

2.3   Installing the printer driver

Installation of the CUPS printer driver for a Hengstler eXtendo printer can be done as follows:

- Installation of CUPS

- Installation of printer using the Printers dialog

- Installation of printer by shell

- Installation of printer using **localhost:631** in browser

Note that, you need to have at least CUPS v1.2 installed in your Linux distribution.

Note that the screen shots below were made during an installation using the Linux Debian Platform and Gnome environment. They may appear differently in other Linux Platforms.

2.3.1   Installing CUPS

Download CUPS package from www.cups.org, file "cups-1.2.12–" <version> "-sources.tar.gz" (e.g. "cups-1.2.12–sources. tar.gz").

- Compile and install CUPS with command …

  ➢ `rpmbuild –ta cups-<version>-sources.tar.gz`

- … or unzip with command …

  ➢ `tar xvf cups-<version>-sources.tar.gz`

- … and install with commands

  ➢ `./configure`

  ➢ `make`

  ➢ `make install`

Note that the GCC compiler must be installed for compiling and installing CUPS.

2.3.2 Printer installation using the Printers dialog

- Manual installation of the printer driver under Linux is done as standard dialog "Printers" installation in following steps: Selecting the "Printers" dialog

- Adding a new printer to "Printer" dialog

- Going through the "Add"

2.3.2.1 Selecting the "Printers" dialog

The "Printers" dialog is started through the menu by selecting ...

[Desktop] ·· [Preferences] ·· [Printers]

... as is shown in Figure 1 below.

Figure 1: Selecting the "Printers" dialog

Within the „Printer" dialog, installed printers are listed as shown in Figure 2 below.
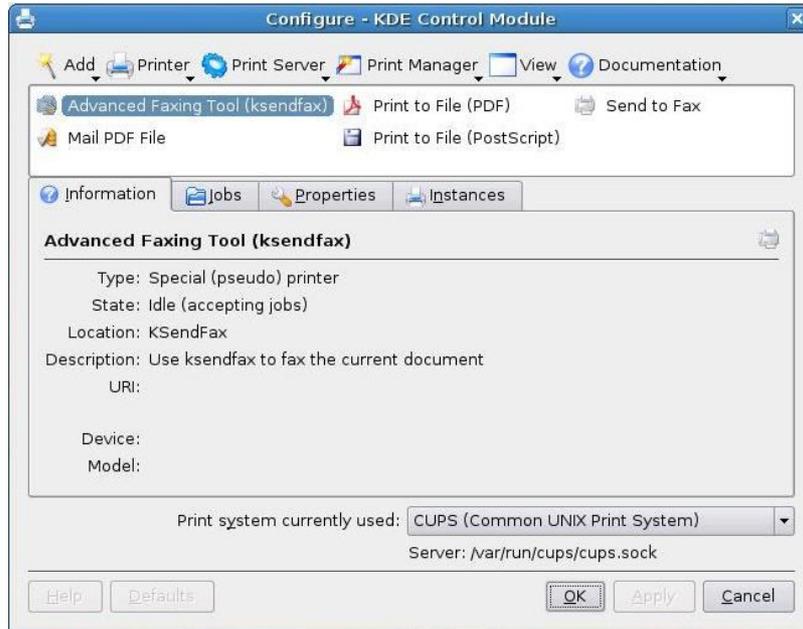
Figure 2: List of installed printers

Note that you may install only one printer of model „ Hengstler X56/X80" (e.g. USB001, COM1, etc.), whereas all of them must use the same eXtendo driver version.

2.3.2.2 Adding a new printer to "Printer" dialog

The eXtendo printer driver is installed with use of the „Add". The wizard is started from the „Printers" dialog, simply by selecting the „Add printer" item in „Add" as shown in Figure 3 below.
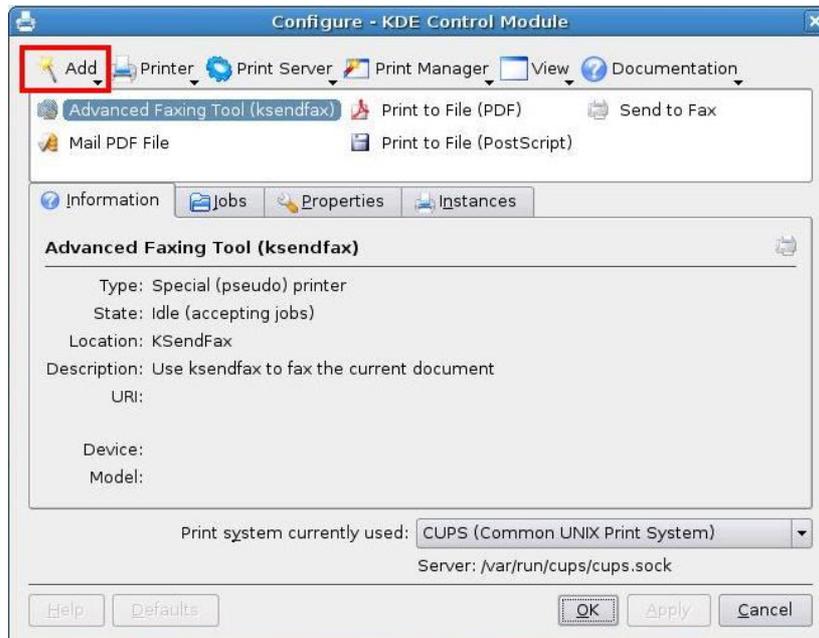
Figure 3: Adding a printer from the „Printers" dialog

The „Add" guides you through this installation.

2.3.2.3 Going through the "Add"

This chapter guides you through the „Add ".

2.3.2.3.1 The „Welcome...." page

The first page of the „Add" (see Figure 4) displays information on the wizard and plug and play printers. This information on „plug and play ...." can be ignored when installing the eXtendo printer driver.
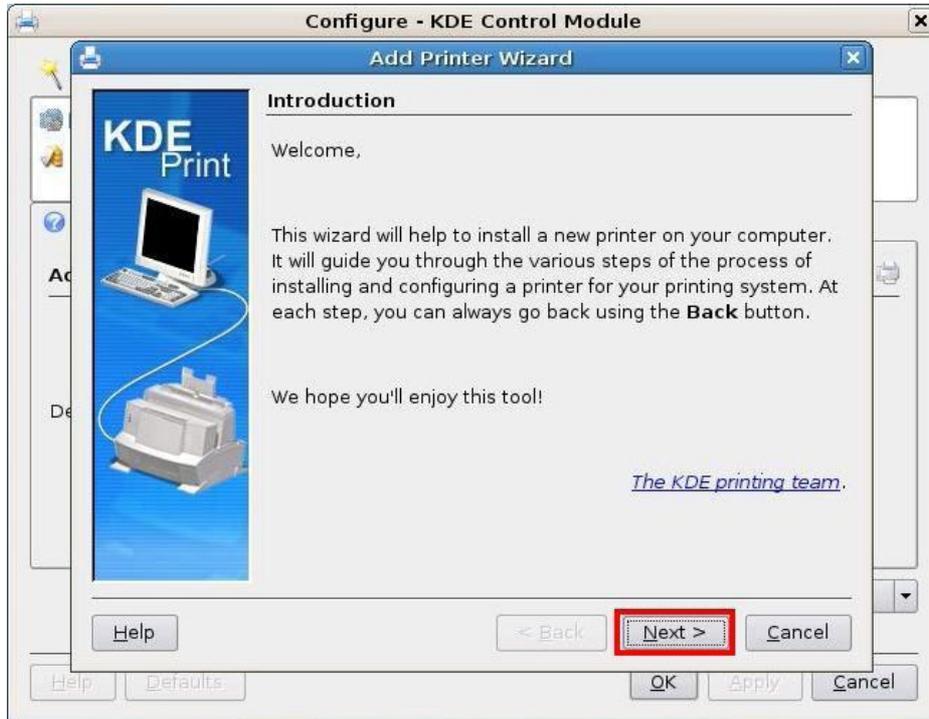
Figure 4: The „Welcome" page of the „Add"

2.3.2.3.2 The „Backend Selection" page

The 2nd page of the „Add" is used to select either a local or a network printer. The eXtendo thermal printer can only be attached as local printer. See Figure 5 for the appropriate settings.

Figure 5: The „Backend Selection" page of the „Add"

2.3.2.3.3 The „Local Port Selection" page

The 3rd page of the „Add" is used to attach a printer. Since the eXtendo thermal printers only support the USB or RS232 connection, you need to select an USB or SERIAL printer. Figure 6 shows the "Local Port Selection" page.
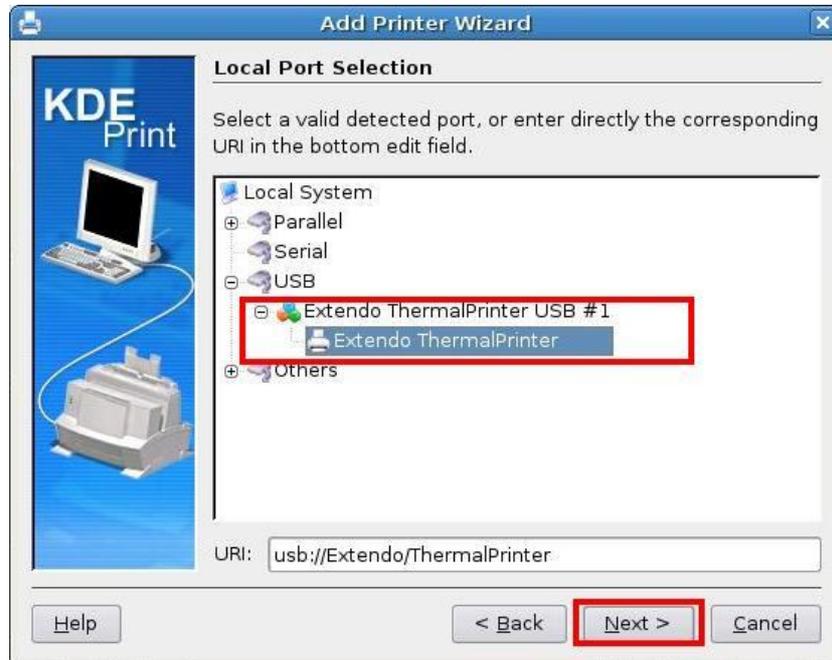
Figure 6: The „Local Port Selection" page of the „Add"

2.3.2.3.4 The „Printer Model Selection" page

The 4th page of the „Add" is used to select the printer model. The printer model is selected by ...

- … specifying the path to the PPD file **„hengstlerx56.ppd"** or **„hengstlerx80.ppd"**

- … selecting **„HENGSTLER X56"** or **„HENGSTLER X80"** from the list of available printers.

Figure 7 shows all steps required for selecting the appropriate printer model.
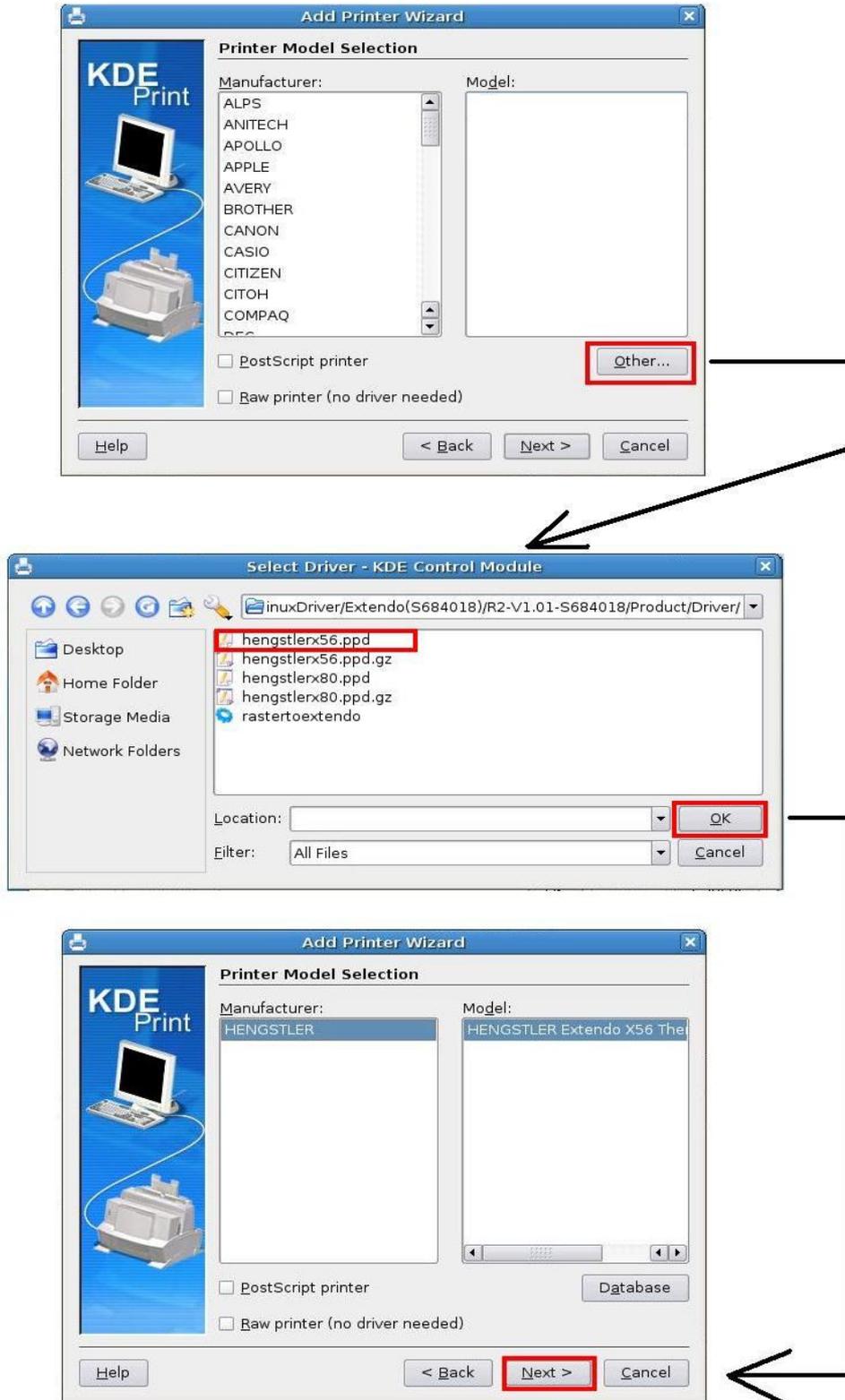
Figure 7: The „Printer Model Selection" page of the „Add"

2.3.2.3.5 The „Printer Test " page

The 5th page of the „Add" allows you to print "Test" page.
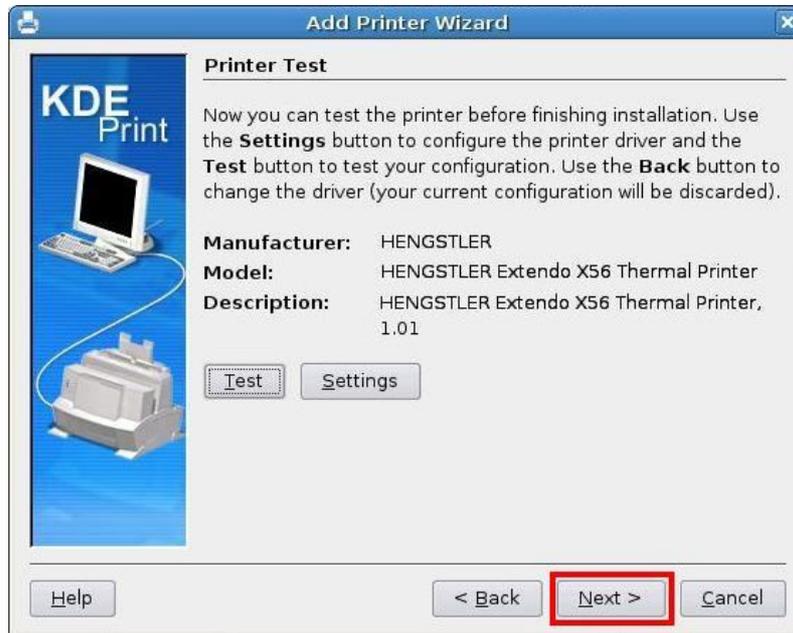
Figure 8: The „Printer Test" page of the „Add"

2.3.2.3.6 The „Banner Selection" page

The 6th page is used to set up extra banners at Starting and Ending printing.

Figure 9: The „Banner Selection" page of the „Add"

2.3.2.3.7 The „Printer Quota Settings" page

The 7th page is used to set up Quota Settings and Page Limits.
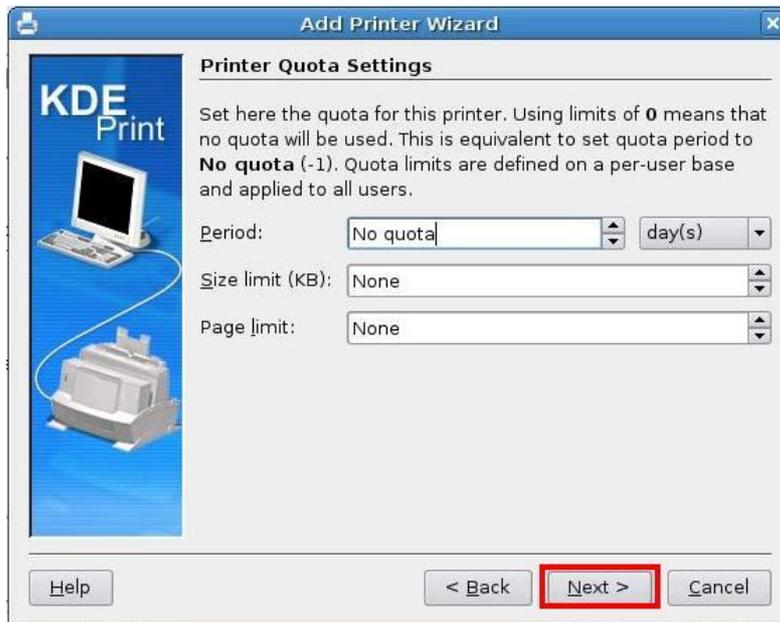
Figure 10: The „Printer Quota Settings" page of the „Add"

2.3.2.3.8 The "User Access Settings"

The 8th page is used to set up User Access to this printer.
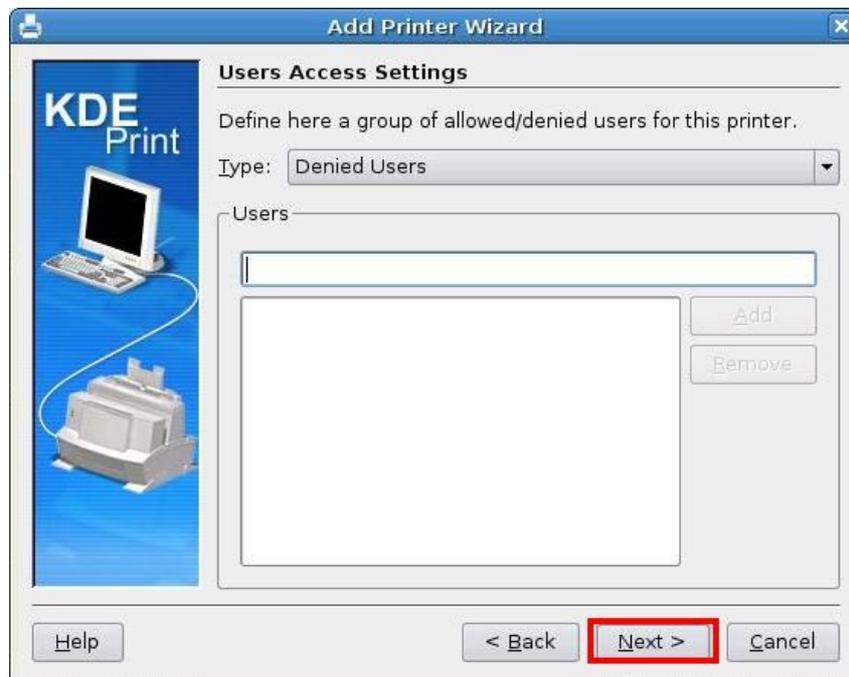
Figure 11: The „User Access Settings" page of the „Add"

2.3.2.3.9 The "General Information" Page

The 9th page is used to set Printer Name and Location of the printer.
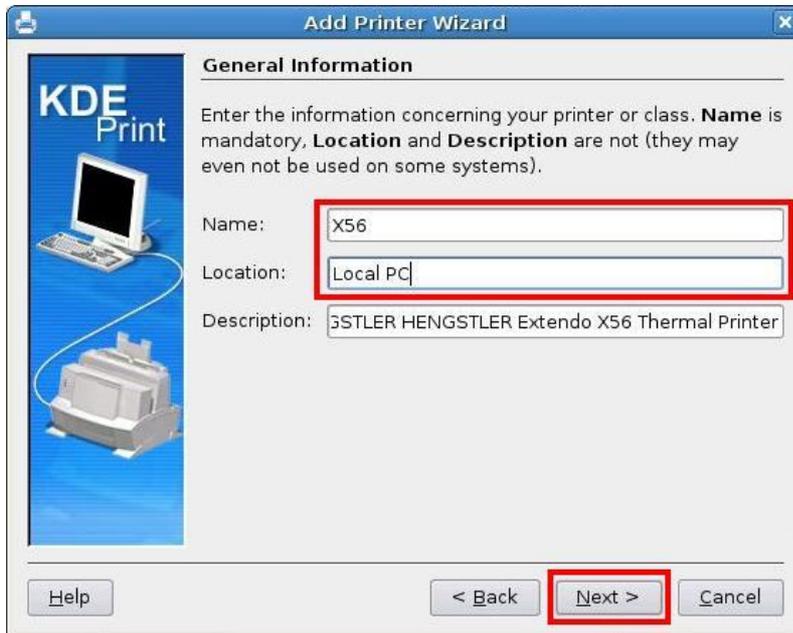
Figure 12: The „General information" page of the „Add"

2.3.2.3.10 The "Confirmation"

The 10th page is used to confirm all settings.
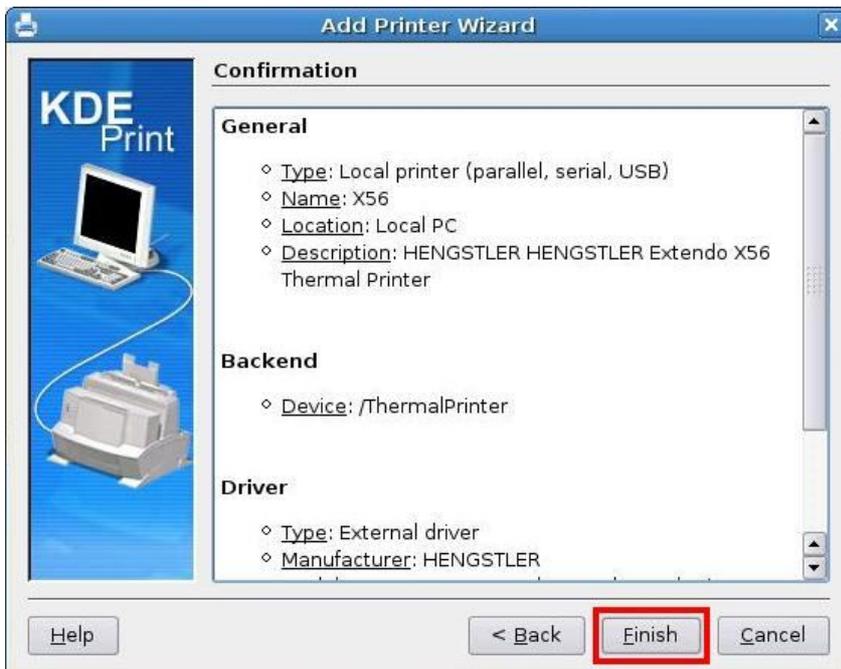
Figure 13: The „General information" page of the „Add"

2.3.2.3.11 The "Printers" dialog

The final "Printers" dialog is used to see if the printer was correctly installed to the system.
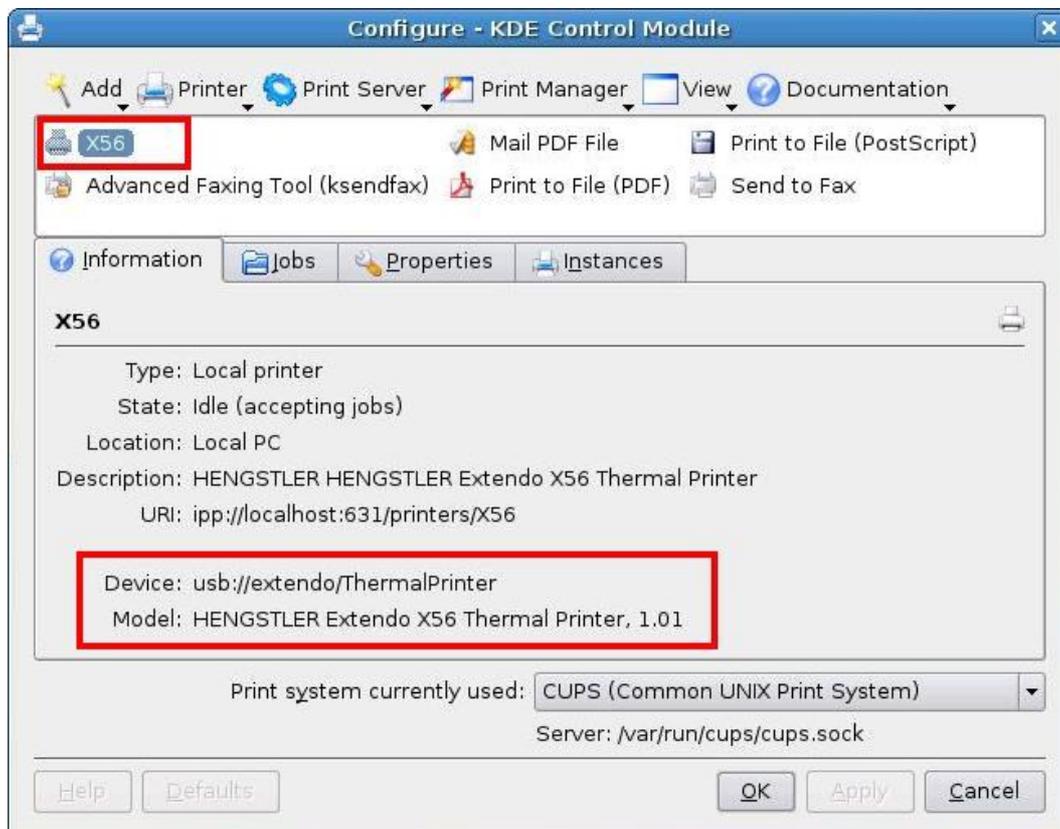
Figure 14: The „Printers" dialog

2.3.3 Printer installation using shell terminal

Sometimes it is required that the installation of the printer driver becomes integrated into a customer specific installation utility. For those cases Linux permits you to install a printer from the shell whereas this command might be easily to integrate into the customer specific installation utility.

The commands for Hengstler printer installation under Linux is:

```
root:/# lpadmin –p PrinterName –E –v Device-Uri –m ppd-filename ENTER
```

Sample printer installation commands:

- Installation of the Hengstler X56 printer driver on USB1:

```
root:/# lpadmin –p HengstlerX56 –E –v usb://dev/usb/lp0 –m hengstlerx56.ppd
ENTER
```

Detailed information on the printer user interface can be retrieved by shell command:

```
root:/# info lpadmin
```

2.3.4 Printer installation using browser localhost:631

For installation of printer you can also you browser with address localhost:631 where you can manually add printer to the system. This address is reachable from all Linux platforms.
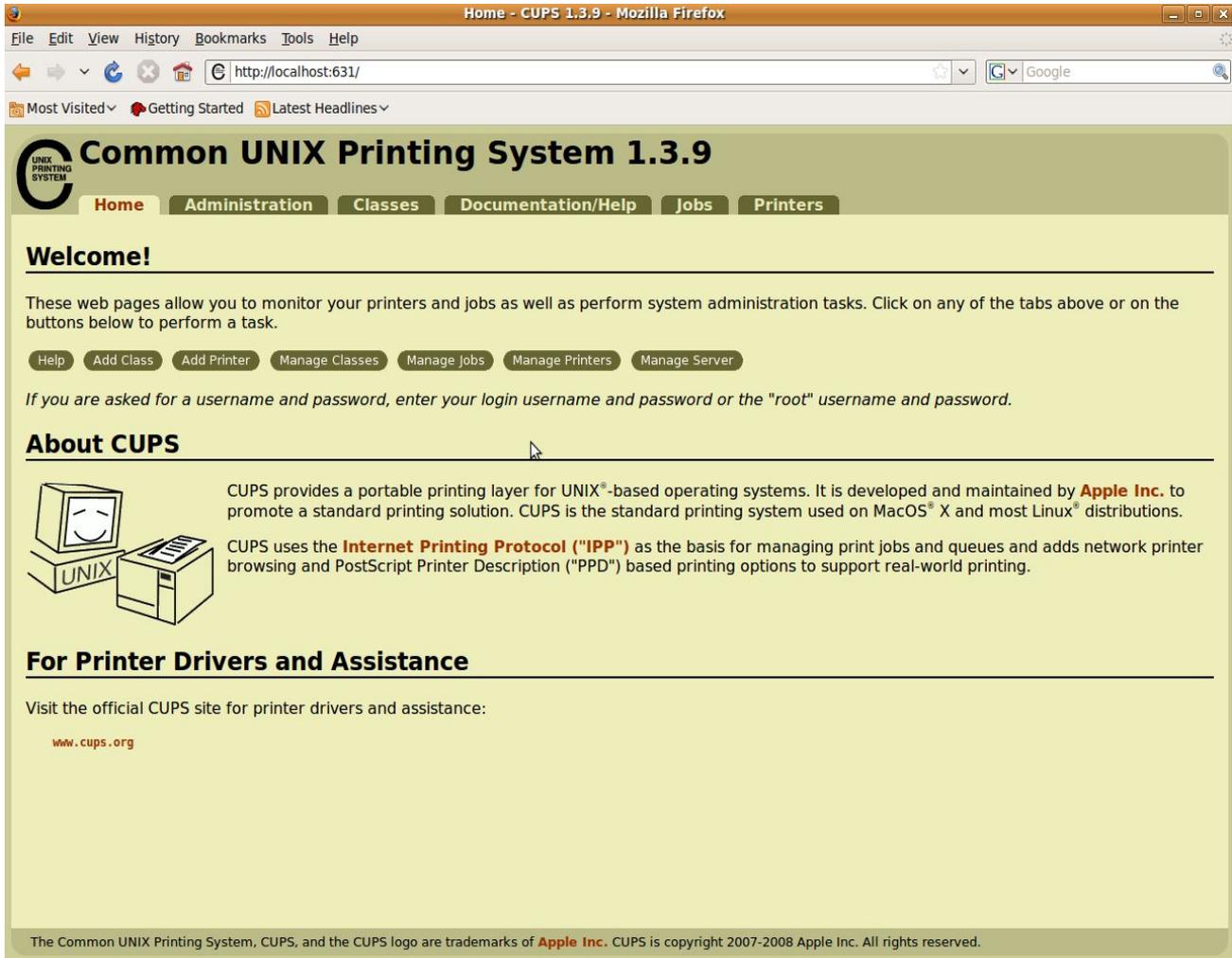
Figure 15: localhost:631 address

2.3.5 Printer driver files

After adding printer to the system, it is necessary to copy also printer driver files to specific destinations.

In **<root>** directory of the driver package run script:

```
root:/# ./install_x86
```

- for **x86** system types

```
root:/# ./install_x64
```

- for **x64** system types

```
root:/# ./install_arm
```

- for **arm** system types

This script will copy all files to destination directories. The output of the shell window can looks like this:

```
Hengstler
Copy CUPS FILTER to /usr/lib/cups/filter
Copy libExoTb.so.1.0 library to '/usr/lib'
Copy libExoApi.so.1.0 library to '/usr/lib'
 * Stopping Common Unix Printing System: cupsd        [ OK ]
 * Starting Common Unix Printing System: cupsd        [ OK ]

Install Complete
Now you can add a printer queue using the Cups tool, go
to address http://localhost:631 with a simpe browser and
set up the printer
```

Figure 16: Shell window after copying driver files

You can find information on copied files and their destination directories in chapter 2.3.6.

2.3.6 Printer driver files overview

After installation of the printer driver, various files are copied into specific directories as follows:

- **/usr/lib/cups/filter**
  .\rastertoextendo   = Raster printer driver for eXtendo printers

- **/usr/lib**
  .\libExoApi   = Application Programming Interface/API (Linux library)
  .\libExoTb   = ToolBox functions (Linux library)

2.4 Un-installing the printer driver

Un-installation of the printer driver must be done with the Un-Installation script.

2.4.1 The eXtendo un-installation script

The eXtendo printer driver un-installation must be done with the eXtendo Un-installation script "uninstall". This script is located in the driver package in directory "**<root>**".

The "uninstall" script will uninstall (remove) all installed eXtendo printer driver files from the system together with API library.

```
root:/# ./uninstall_x86
```

- for **x86** system types

```
root:/# ./uninstall_x64
```

- for **x64** system types

```
root:/# ./uninstall_arm
```

- for **arm** system types

Removal of all printer driver files is required to avoid a eXtendo driver file version mismatch after installing a different eXtendo driver version.

Note that deleting a eXtendo printer from the "Printers" dialog is also necessary. You can also do it in shell command:

```
root:/# lpadmin -x PrinterName
```

The eXtendo un-installation script is show in Figure 17.


```
Hengstler
Deleting CUPS FILTER from /usr/lib/cups/filter
Deleting libExoApi.so.1.0 library from '/usr/lib'
Deleting libExoTb.so.1.0 library from '/usr/lib'
 * Stopping Common Unix Printing System: cupsd        [ OK ]
 * Starting Common Unix Printing System: cupsd        [ OK ]

Un-Install Complete
```

Figure 17: The eXtendo Un-installer script

2.5   Re-installing the printer driver

For manual re-installation of the printer driver following steps must be done:

- Delete printer using **[Desktop] ·· [Preferences] ·· [Printers]**
  In SUSE 10.1 KDE **[Start] ·· [Utilities] ·· [Printing Manger] ·· [Remove]**

- Un-install the driver by opening a terminal in the driver package and by entering "./uninstall_x86" or "./uninstall_arm" based on system type. Reboot the computer afterwards. (see chapter 2.4)

- Add the printer using **[Desktop] ·· [Preferences] ·· [Printers]**
  In SUSE 10.1 **[Start] ·· [Utilities] ·· [Printing Manger] ·· [Add]**
  Note that the USB printer must be connected before adding the printer. (see chapter 2.3)

- Install the driver by opening a terminal in the driver package and by entering "./install_x86" or "./install_arm" based on system type. Reboot the computer afterwards.

3   Operation Manual

After successful installation, the printer driver can be referenced to by applications that have a printing function (such as kWord, Firefox …) using the GDI.

3.1   Configuring the printer driver

The printer driver is configured from the printer configure dialog. Among other, this dialog allows you to

- Specify the print density

- Specify the print speed

- Specify if trailing blank dot lines must be removed from the printed page

- Specify cutter control at end of print

- Specify if a form feed is required at end of print

- Specify a limited form length in dot lines

- Specify which paper path is to be selected

3.1.1   Selecting the "Printer Properties" dialog

The „Configure" dialog is started from the „Printers" dialog as shown in Figure 18,
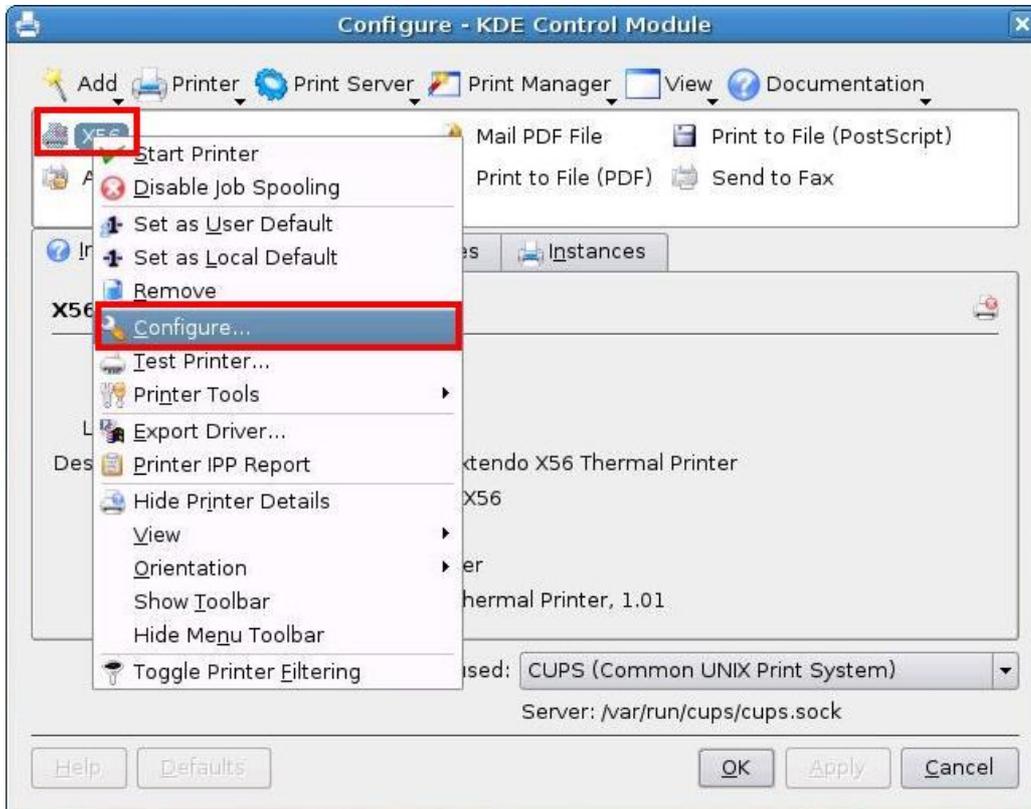

Figure 18: Starting the „Printer Properties" dialog for the printer driver

3.1.2   Overview of attributes in the "Configure" dialog

Within the "Configure" dialog, following attributes are available:

| Attributes	Function
Driver Information	Displays Information on the eXtendo driver.
Paper Control	Paper settings related to the printout: remove trailing blanks flag, reduced form length, initiate form feed flag and cutter control setting for end of page.
Printer Settings	Printer related settings: print density and print speed.
Others	Print resolution settings.
General	Media size (paper size) settings.
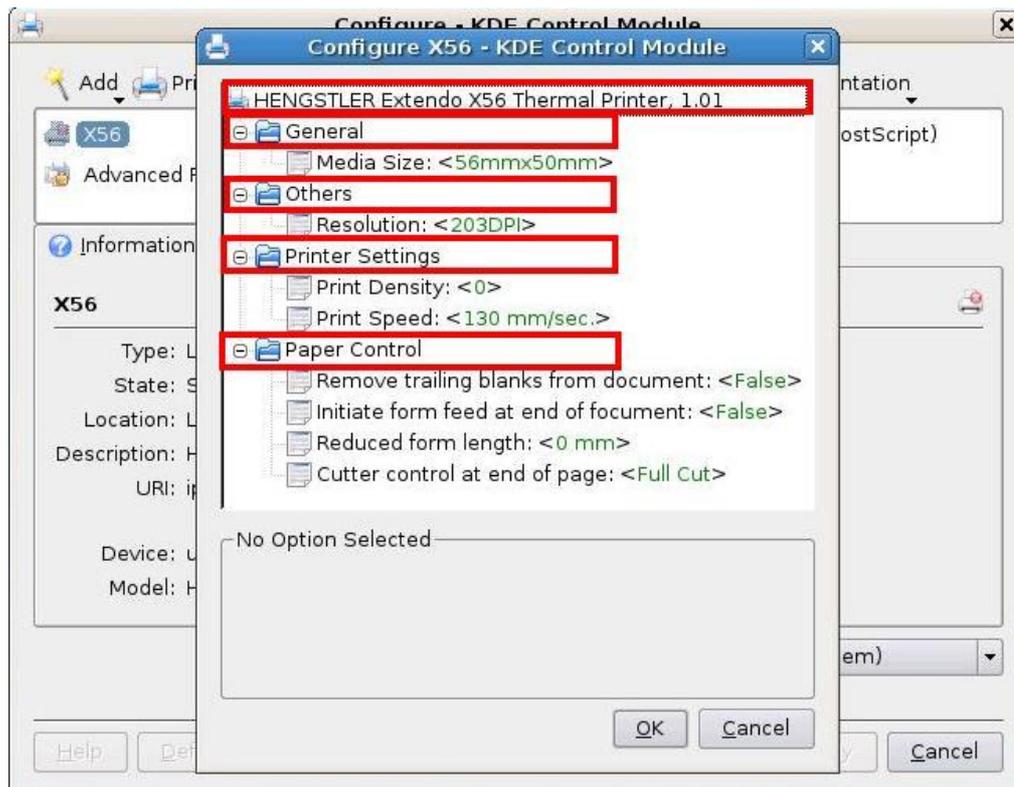
Figure 19: Selecting Attributes in the „Configure" dialog

3.1.3 The "Driver Information" attribute

The „Driver Information"attribute displays information on the printer driver as shown in Figure 20. The most important information is the version/release (here R2-V**1.01**) of the printer driver.
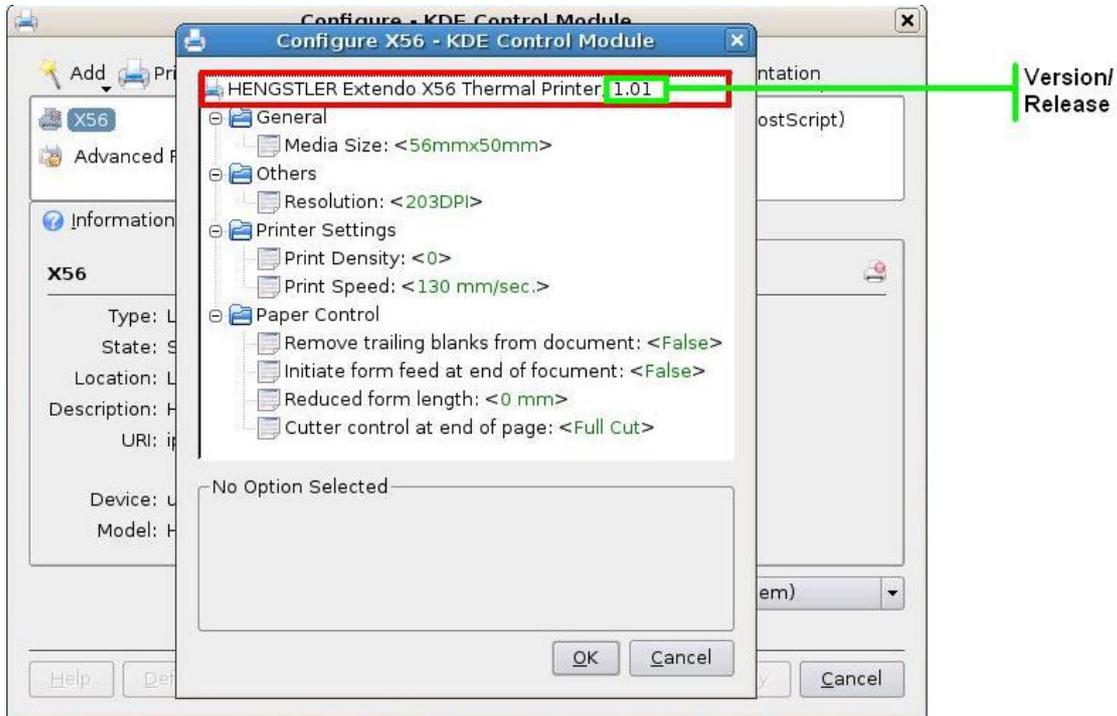
Figure 20: The „Driver Information" attribute in the „Configure" dialog

3.1.4 The "Paper Control" attributes

The „Paper Control" attributes are used to specify:

- Remove trailing blanks

- Reduced form length

- Form feed at end of document,

- Cutter control at end of page

- Path selection at start of page
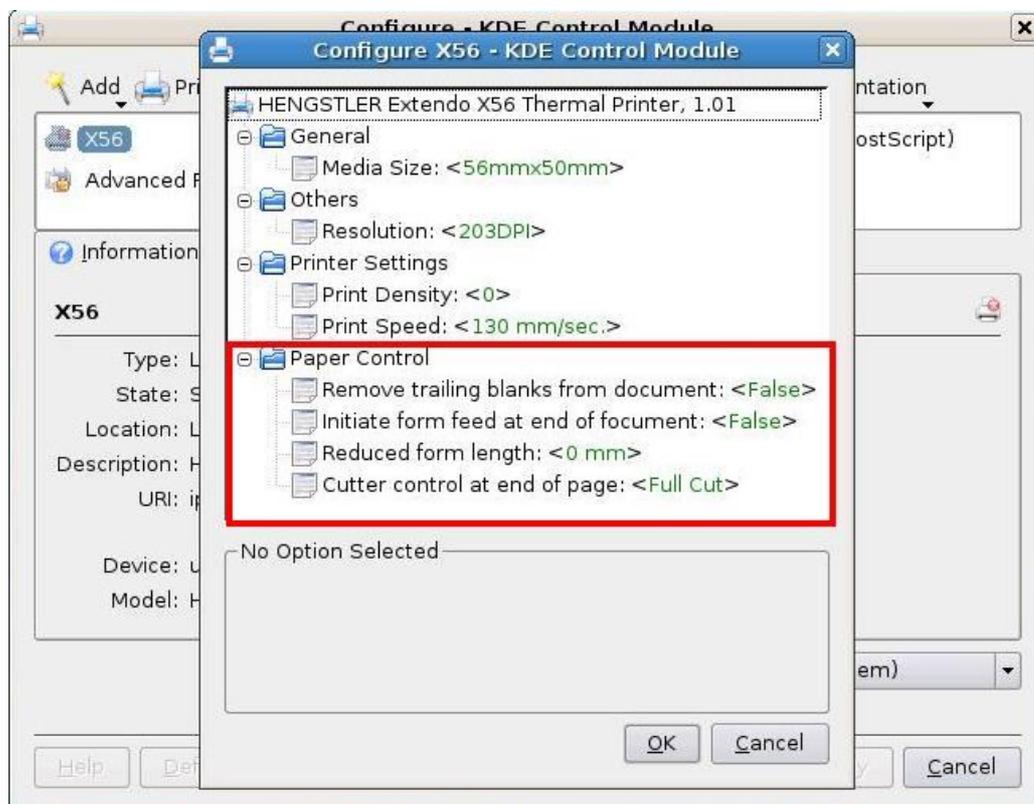
Figure 21: The „Paper Control" attributes in the „Configure" dialog

The "reduced form length" is specified in mm.

The "remove trailing blanks" option is helpful when the length of the printout needs to be variable (not fixed). If enabled, then all trailing blank dot lines of GDI printouts will not be printed.

A variable size printout is done in following steps:

In your GDI application, select one of the eXtendo specific paper sizes and large height ("3000mm" is the largest height currently available in the driver)

- Clear the complete page (e.g. fill the page with white color)

- Setup the top part of the page with data (e.g. page contents) that is to be printed

- Print the whole page as portrait, whereas only the top part (e.g. page contents) will be printed
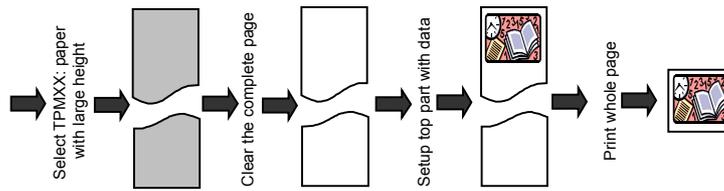

Figure 22: Printing with variable paper size

The maximum document length option allows limitation of the page length. The part of the page that exceeds the specified document length is not sent to the printer, when this option is enabled. Note that the initial and final paper feed specification do not affect the page length.

The "reduced form length" option is helpful when the length of the selected form needs to be reduced. This might be the case when paper with black marks is used, whereas the selected form exceeds the distance between two adjacent black marks.

Following keys in the [PaperControl/Paper Control] section of the hengstlerx56/ hengstlerx80.ppd files relate to this attribute:

Key	Description	Units	Domain	Comment
FormLength	Reduced form length	mm	[0..20]	Printer domain might differ
CutterControl	Cutter control at end of page	N/A	[0,1,2,3]	0 = NO CUT, 1 = FULL CUT, 2 = PARTIAL CUT,

3.1.5 The "Printer Settings" attributes

The „Printer Settings" attributes are used to:

- Specify the print density of the thermal printer

- Specify the print speed of the thermal printer

The print density can be specified in the range from –100 to +100 percent, whereas 0% is default. It specifies the energization level for the thermal line, which might need to be corrected depending on the type of thermal paper used. The print density of the printer is set in advance to each page printout.

The print speed can be specified in the range from 30 mm/sec to 250 mm/sec, whereas 130 mm/sec is default. It specifies the maximum speed which the printer may use for printing the document. The print speed of the printer is set in advance to each page printout.
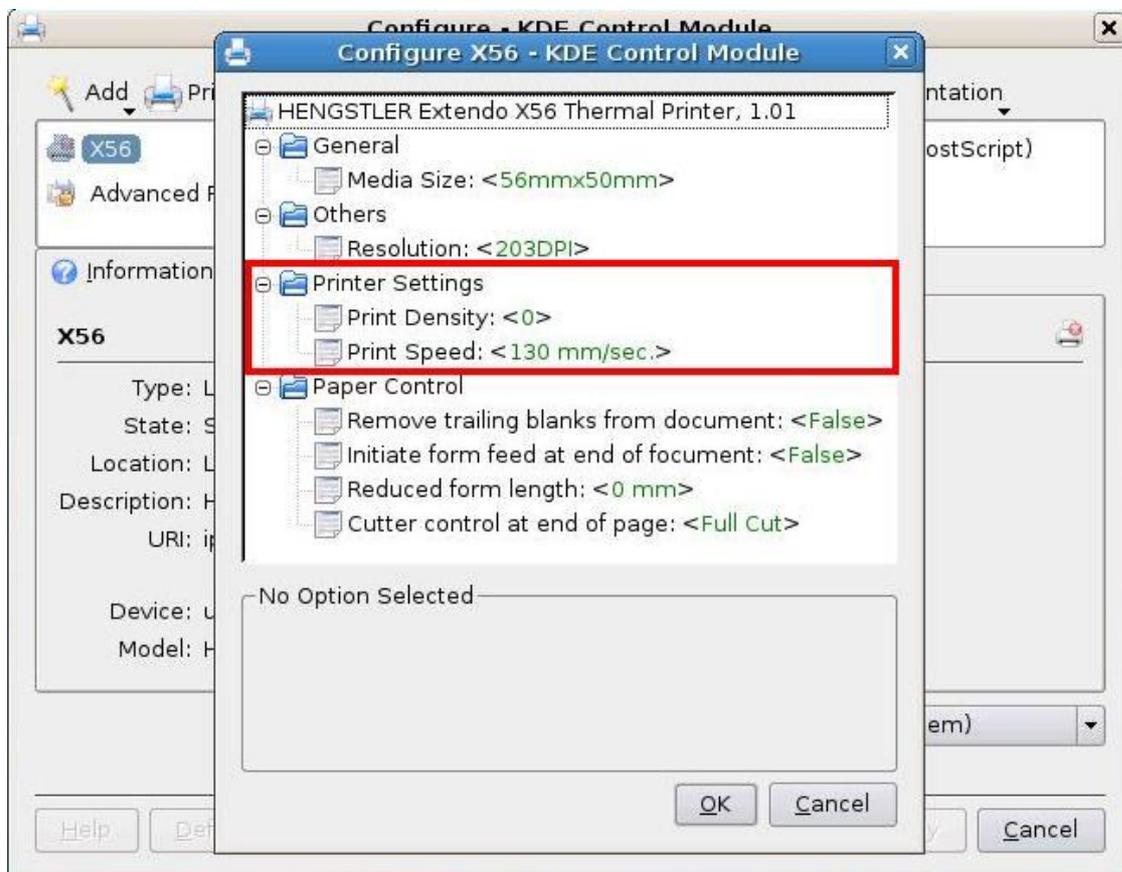

Figure 23: The „Printer Settings" attributes in the „Configure" dialog

Following keys in the [PrinterSettings/Printer Settings] section of the hengstlerx56/ hengstlerx80.ppd files relate to this attribute:

Key	Description	Units	Domain	Comment
PrintDensity	Energation level for the thermal line	Percent	[-100..+100, Disabled]	0=-0%, 15=+15%
PrintSpeed	Maximum speed for printed	Mm/sec	[30..250, Disabled]	130=130 mm/sec

3.1.6 The "General" attribute

The „General" attribute is used to select media (paper) size for printing. The media size can be specified in the range from 50mm to 3000mm long paper plus specific paper sizes (A2, A3, A4…).
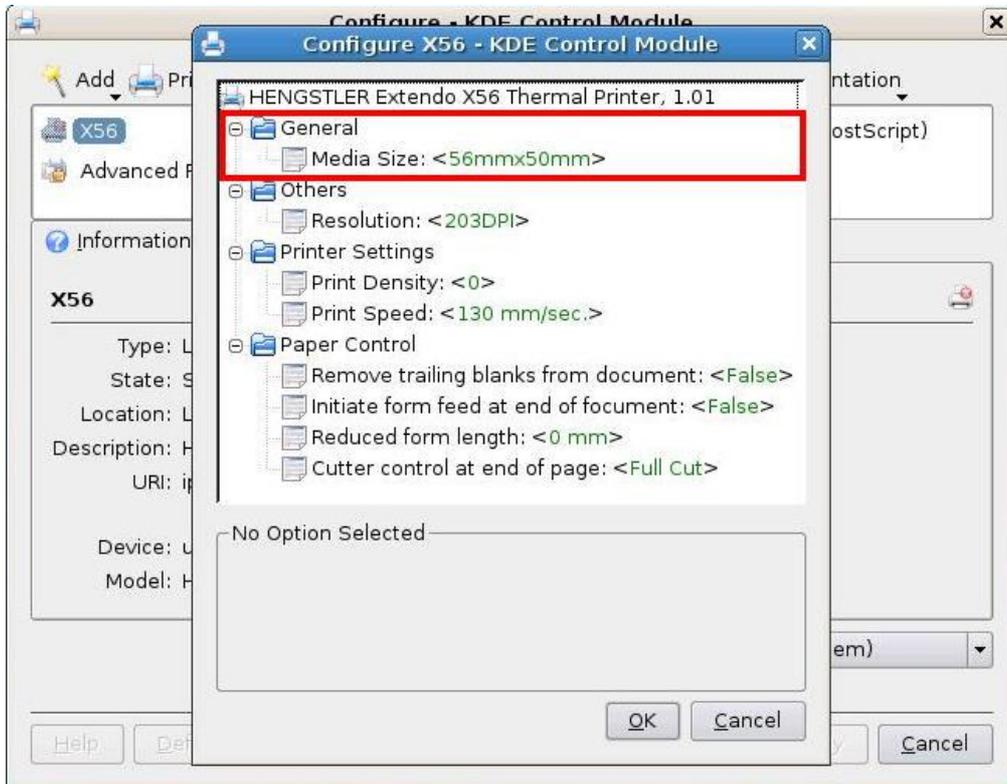

Figure 24: The „General" attribute in the „Configure" dialog

Following keys in the [PageRegion/Media Size] section of the hengstlerx56/ hengstlerx80.ppd files relate to this attribute:

Key	Description	Units	Domain	Comment
PageSize	Maximum page size settings	Mm	[50..3000, A3, A4, A5, B4, B5, LETTER, LEGAL]	50=50mm 1000=1000mm

3.1.7 The "Others" attribute

The „Others" attribute is used to select resolution for printing. Default resolution for X56 and X80 is 203DPI,
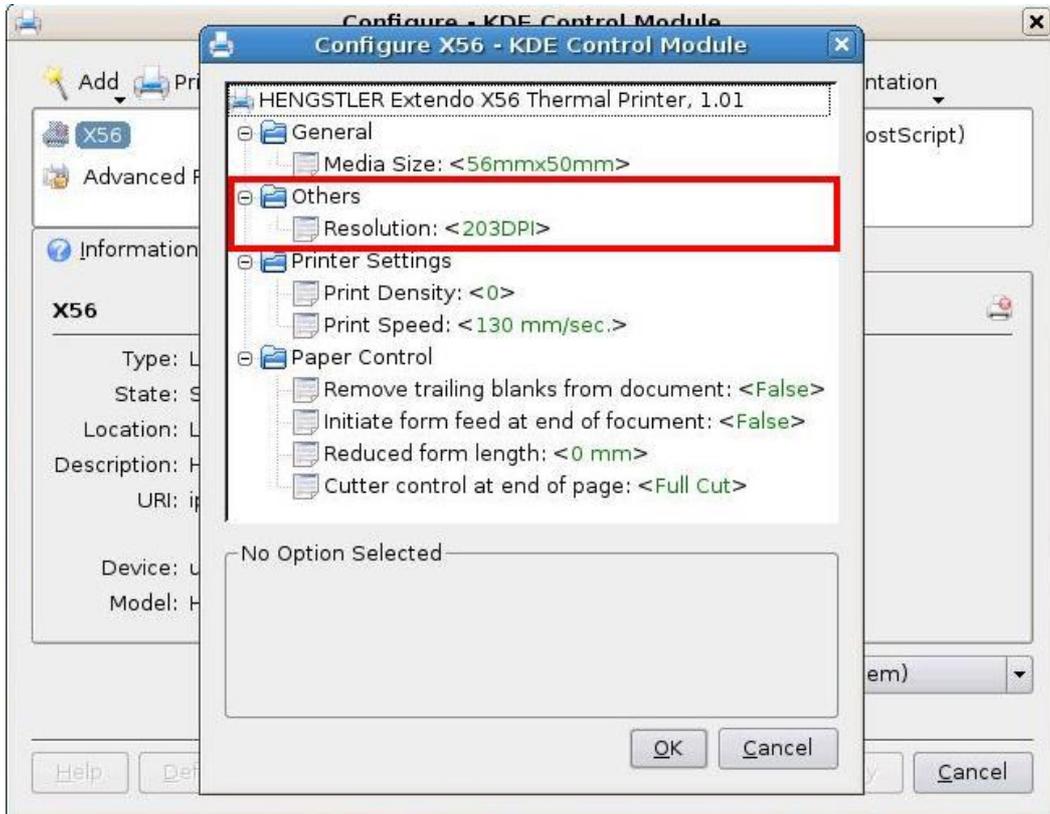

Figure 25: The „Others" attributes in the „Configure" dialog

3.2   Using the printer driver

The printer (CUPS driver) can be used by standard applications (e.g. kWord) through the Graphical Device Interface (GDI), or, by special applications (e.g. Receipt-Systems) through both the GDI and the Software Development Kit (SDK=>API).

3.2.1   Using the printer driver in common applications

The printer can be used in common applications that use GDI for printing (such as kWord), however with following restrictions:

• The selected paper size must match one of the eXtendo predefined paper sizes (e.g. "X56 56: A4").

• If the selected paper size differs from an eXtendo pre-defined paper size, then the selected paper will be used.

Note that you can setup the default paper size in the "Configure" dialog (see the "General" attribute in the "Printers" dialog).

3.2.2   Using the printer driver in special applications

The printer driver can be used in special applications that use the SDK or SDK + GDI interface of the printer driver for printing.

The SDK offers an API (Application Programming Interface) that supports:

• Direct printer access (open, close, read, write, reset)

• … and more …

4 Driver Synopsis

This chapter describes the conception of the eXtendo CUPS printer driver components and environment.

4.1   The eXtendo CUPS printer driver

The eXtendo CUPS printer driver is designed for usage with Hengstler X56 and X80 printers whereas it offers an interface to the application that has printing, status inquiry and printer control capabilities.
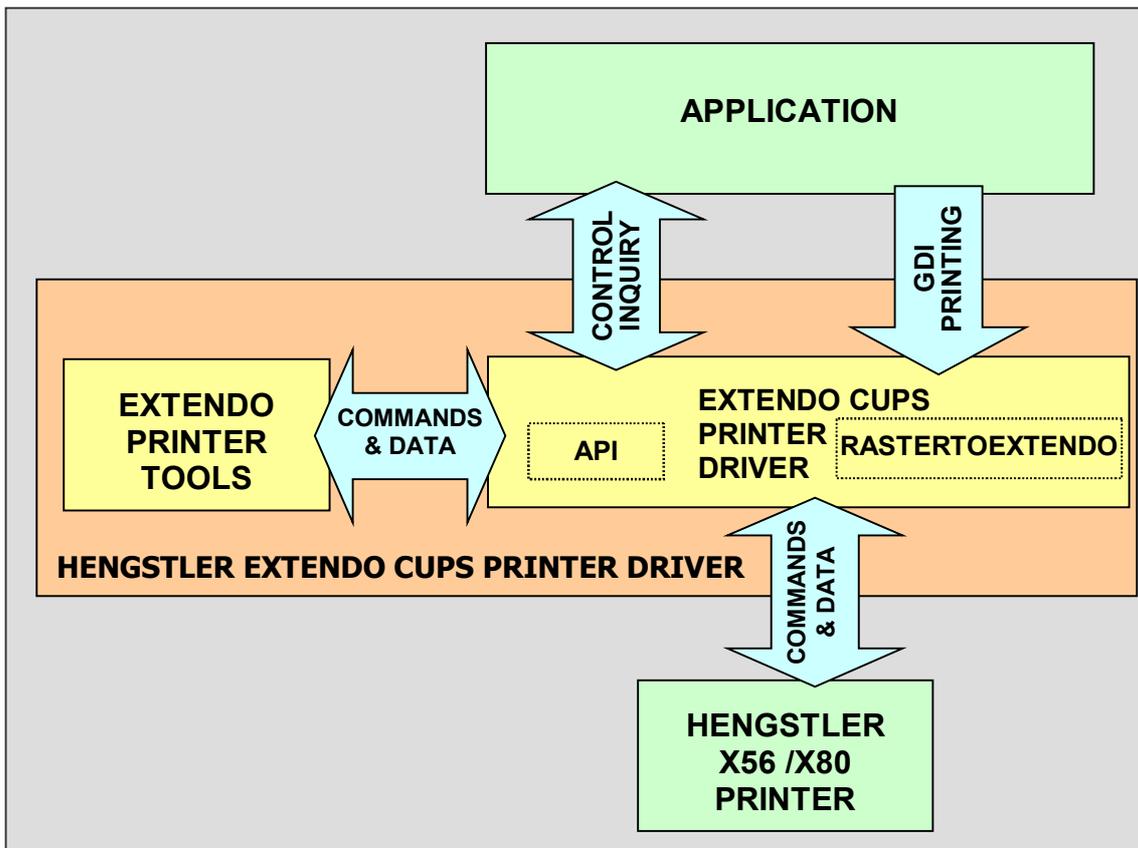

Figure 26: The eXtendo CUPS printer driver environment

The application may use one or more of following eXtendo interfaces:

- GDI printing
  This graphical device interface allows printing from standard applications, whereas eXtendo driver renders the printout that is send to the printer.

- Application Programming Interface (API)
  This interface allows the application to open/close the connection to the printer and inquire information or control paper movement (e.g. cut, etc.).

Figure 26 shows the eXtendo CUPS printer driver environment.

5   Programming Manual

This chapter describes the application programming interfaces (API) that is part of the printer driver package.

5.1   Integrating the eXtendo CUPS printer driver in an application

The eXtendo offers two different interfaces to the application:

- **GDI**
  The graphical device interface is a Linux standard for representing graphical objects and transmitting them to output devices, such as monitors and printers. An application printout through GDI will result in one or more pages that are passed to the eXtendo raster, whereas this component converts it to the data, witch are send to the printer.

- **API**
  An application can use the application programming interface to directly interact with the printer by means of open(), close(), read(), write(), etc.

  From CUPS v1.4, CUPS are not using for communication "usblp" library anymore. New library "libusb" is used. Therefore new API in latest eXtendo Linux driver R3-V1.02a-BETA is extended to this communication also, so now it supports both ways "usblp" and "libusb".

  Differences for opening printer connection for each library type:

  usblp – to open connection to printer you need to type **"/dev/usb/lpX"** as printer port(X- means port number 0-99)

  > Example: "/dev/usb/lp0"

  libusb – to open connection to printer you need to type **"/dev/bus/usb/00X/00Y"** as printer port(X-means USB bus-number and Y-means device)

  > Example: "/dev/bus/usb/005/002"

  To check what is bus and device number for the printer connected to your system, type in shell:

  ```
  >> lsusb
  ```

  In output search for row where **"1bf1:0001"** is. "1bf1" is VID number and "0001" is PID number for eXtendo printer.

  After each printer disconnect or reset this BUS number are changed. To ensure that your printer is always mapped to same path you need to crate symlinks. More information about how to create / write udev rules study Linux manual.

  ```
  Example:

      /dev/bus/usb/00X/00Y is always mapped to > /dev/usb/ExtendoX56
  ```

5.2   The eXtendo application programming interface (API)

5.2.1   Installation eXtendo API into Linux system

Library libExoApi.so.1.1 is automatically installed to system with driver installation.

-To check if library was installed in your system, type in shell:

```
>>    ldconfig –p
```

-It will list all installed libraries in your system, where you should find also libExoApi.so.1.1 together with symbolic links.

If library was not installed into your system you can do it manually in shell terminal:

-Go to <root>Api folder and copy library to /usr/lib, then create symbolic links

```
>>    cp libExoApi.so.1.1 /usr/lib

>>    ln –sf /usr/lib/libExoApi.so.1.1 /usr/lib/libExoApi.so

>>    ln –sf /usr/lib/libExoApi.so.1.1 /usr/lib/libExoApi.so.1

>>    ldconfig
```

Same steps are applicable for manual installing libExoTb.so.1.0 library.

5.2.2   Integrating eXtendo API in an application

The Extendo API (application programming interface) is represented by file **libExoApi.so.1.0**, a dynamic library (.so) that exports all functions of the Extendo API. File **libExoApi.so.1.0** is located in /usr/lib/.

Three additional files allow integration of the libExoApi.so.1.0 in your application:

- **Exo_Api.h**
  This is the header file of the Extendo API which describes the API through macro definitions, type definitions, function prototypes, etc. This header file must be included in your application C-source files that call Extendo API functions. Note that this file is used during compile time to check if your function calls are syntactically correct.

This file can be found on the distribution CD in directory "<root>\Api".

Note that libExoApi.so.1.0 is a vital component of your final application environment and that its availability is a must. File Exo_Api.h, however, is only used during the application development phase and is not needed for the final application environment.

If you want to use the library in your application, you have to include header file in your application sources and in compilation you have to add library (-lExoApi) as parameter.

Note that, you need add time library (–lrt) as parameter in compilation in some Linux distribution.

5.2.3 The interface definition file Exo_Api.h

The eXtendo API is defined in header file Exo_Api.h which is divided into following parts:

- INCLUDES   -> list of include files

- MACROS   -> macro definitions

- TYPES   -> type definitions

- FUNCTIONS   -> function prototypes

- VARIABLES   -> exported variables

5.2.3.1 Includes Part (API-INCLUDES)

The "INCLUDES" part in file Exo_Api.h includes all header files required for successful compilation of this header file. Currently no "INCLUDES" are available.

5.2.3.2 Macros Part (API-MACROS)

The "MACROS" part in file Exo_Api.h specifies following macros:

- Printer Types

- Error Code

- Printer Path

5.2.3.2.1 Error Code

Following macros specify error codes that are used for informing the caller about errors in printer access functions.

Macro	Description
D_EXO_API_OPEN_FAILED	This error code informs that the opening printer communication failed
D_EXO_API_INVALID_DEVICE	This error code informs that the device is not correct (USB or SERIAL)
D_EXO_API_DETACH_FAILED	This error code informs that the device cannot be detached (from R3-V1.02a).
D_EXO_API_CLAIM_FAILED	This error code informs that the device cannot be claimed (from R3-V1.02a).
D_EXO_API_GET_ACTIVE_CONFIG_FAILED	This error code informs that the device can not set active config (from R3-V1.04a).
D_EXO_API_NO_INTERFACE	This error code informs that the device has no interface (from R3-V1.04a).
D_EXO_API_ENDPOINT_COUNT_FAILURE	This error code informs that the device has no endpoints (from R3-V1.04a).

5.2.3.3 Types Part (API-TYPES)

The "TYPES" part in file Exo_Api.h specifies following basic data type definitions. Currently no "TYPES" are available.

5.2.3.4 Functions Part (API-FUNCTIONS)

The "FUNCTIONS" part in file Exo_Api.h specifies following function types:

- Printer Access Functions

- Library Functions

5.2.3.4.1 Printer Access Functions

The following functions are available for accessing the printer:

Function	Description	
	Return Value	**Parameter**
exo_api_printer_open	Function exo_api_printer_open (...) opens the connection to printer.	
	intptr_t　　　　printer_handler	char　　　　　　* port
exo_api_printer_close	Function exo_api_printer_close (...) closes the printer connection	
	VOID	intptr_t　　　　printer_handler
exo_api_printer_read	Function exo_api_printer_read (...) reads data sent by printer until all data has been received or until the timeout runs out	
	int　　　　nr_of_read_bytes	intptr_t　　　　printer_handler
		unsigned char　　* read_data
		int　　　　size_of_data
		unsigned long　　timeout_ms
exo_api_printer_clear	Function exo_api_printer_clear (...) clears all buffered data that has been or has not yet been sent by printer.	
	int　　　　　result	intptr_t　　　　printer_handler
exo_api_printer_write	Function exo_api_printer_write (...) sends data to the printer until all data has been sent or until the timeout runs out	
	int　　　nr_of_write_data	intptr_t　　　　printer_handler
		unsigned char　　* write_data
		int　　　　size_of_data
		unsigned long　　timeout_ms
exo_api_printer_set_serial	Function exo_api_printer_set_serial (...) sets up the serial printer port	
	int　　　　fResult	intptr_t　　　　printer_handler
		unsigned long　　baud_rate
		unsigned char　　data_bits
		unsigned char　　parity
		unsigned char　　stop_bits

5.2.3.4.2 Library Functions

The following functions are available for accessing the API version:

Function	Description	
	Return Value	**Parameter**
exo_api_get_library_info	Function exo_api_get_library_info (...) Retrieves info about the library	
	char * library_info	VOID

5.2.3.5 Variables Part (API-VARIABLES)

There are no variables exported by the eXtendo API.

5.2.4   eXtendo API Coding Examples

This chapter contains following eXtendo API coding examples:

- Printer Communication Coding Example

5.2.4.1 Printer Communication Coding Example

Document printing through the eXtendo API is done to initiate print outs on a specific printer type and port.

Following C-Programming example is an extract from a demo application and shows the procedure for document printout through the eXtendo API:

```
void InquirePrinterStatusThroughExo_Api
   (char *PrinterType, int PrinterPort)
{
   intptr_t iPrinter    = 0; // Handler to printer
   int iNrReadData = 0;  // Number of read bytes from printer

   //-- Open connection to printer
   iPrinter = exo_api_printer_open (PrinterPort);

   //-- Check if printer was opened
   if (   (iPrinter != D_EXO_API_OPEN_FAILED)
      || (iPrinter != D_EXO_API_INVALID_DEVICE)
      || (iPrinter != D_EXO_API_DETACH_FAILED)
      || (iPrinter != D_EXO_API_CLAIM_FAILED)
      || (iPrinter != D_EXO_API_GET_ACTIVE_CONFIG_FAILED)
      || (iPrinter != D_EXO_API_NO_INTERFACE)
      || (iPrinter != D_EXO_API_ENDPOINT_COUNT_FAILURE)
      ) {

      BYTE aubEcPrinterDataRequest[] = {0x1d,0x61,0x00, 0x01}; // GS a+[n]+[m],m=0x00,n=0x01 =>
                                       // emulation command for requesting
                                       // printer status

   //-- Clear receive buffer
      exo_api_printer_clear (iPrinter);
```

Call exo_api_printer_open() to open the connection with the printer

Declaration of the emulation command "GS 'a'+[n]+[m]" with [n]==[00] and [m]==[01] will inquire the printer status.

Call exo_api_printer_clear () to clear receive buffer.

```
    //-- Request printer  status
    if (    exo_api_printer_write
                (iPrinter, aubEcPrinterDataRequest, sizeof(aubEcPrinterDataRequest), 1000)
        == sizeof(aubEcPrinterStatusCmd)
      ) {



    BYTE ubEcPrinterDataResponse[255];



    //-- Receive printer status packet header
    iNrReadData = exo_api_printer_read
                (iPrinter, ubEcPrinterDataResponse, sizeof(ubEcPrinterDataResponse), 1000);

    //-- Check if response data is a printer status
    if (        iNrReadData > 4)
        && (ubEcPrinterDataResponse[0] == 0x1B)
        && (ubEcPrinterDataResponse[1] == 0xFF)
        && (ubEcPrinterDataResponse[2] == 0x02) // Printer status packet
      ) {

        //-- Process printer status
        ....
    }
    }
  }

  //-- Close connection with printer
  Exo_Api_PrinterClose(iPrinter);
}
```

> Call exo_api_printer_write () to send the emulation command to the printer.

> Declaration of the 255 bytes packet will be used to identify the printer status packet.

> Call exo_api_printer_read () to read the packet which precedes the printer status

> Call exo_api_printer_close () to close connection with the printer

6 Printer Tools

6.1 Introduction

This chapter describes the eXtendo printer tools, which are distributed as part of the printer driver package and which are available in directory "<root>\Tools".

Currently following printer tools are available:

- The Exo Print Terminal

- The Exo Api Test

6.2 The Exo Print Terminal

The Exo Print Terminal tool is used for immediate communication with the printer using emulation commands. The tool is started by writing in shell window in directory <root>/Tools/x86 or <root>/Tools/arm based on system type:

```
>>    ./Exo_PrintTerminal
```
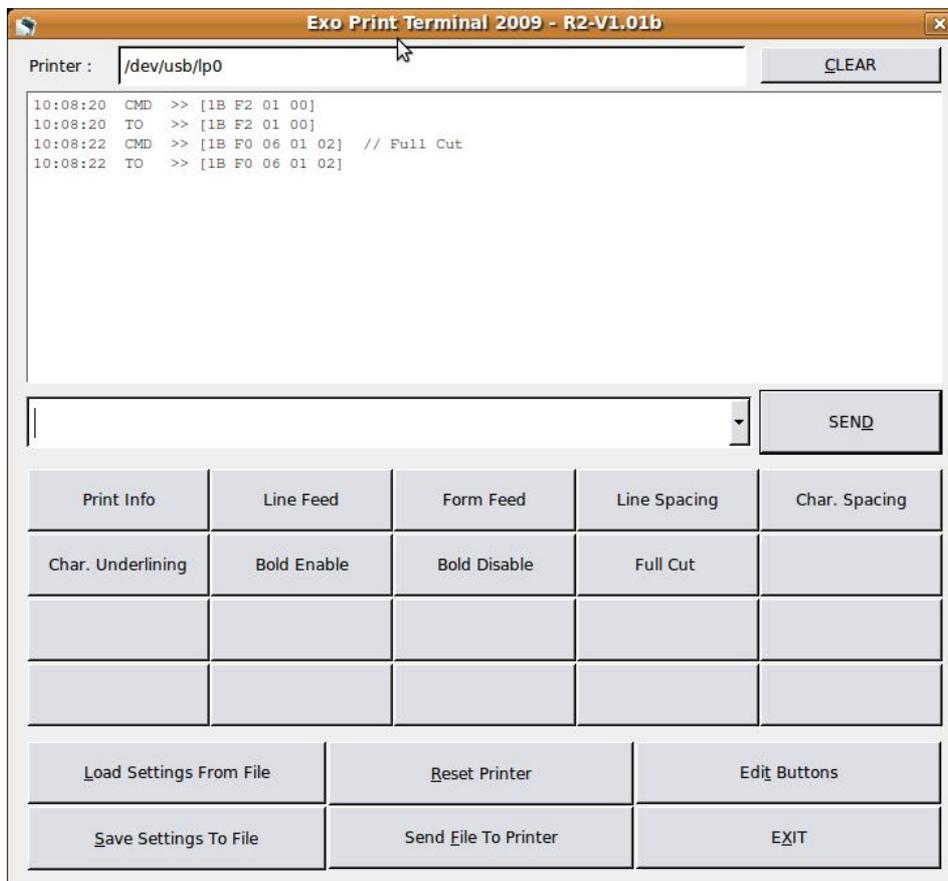

Figure 27: The Exo Print Terminal-main dialog
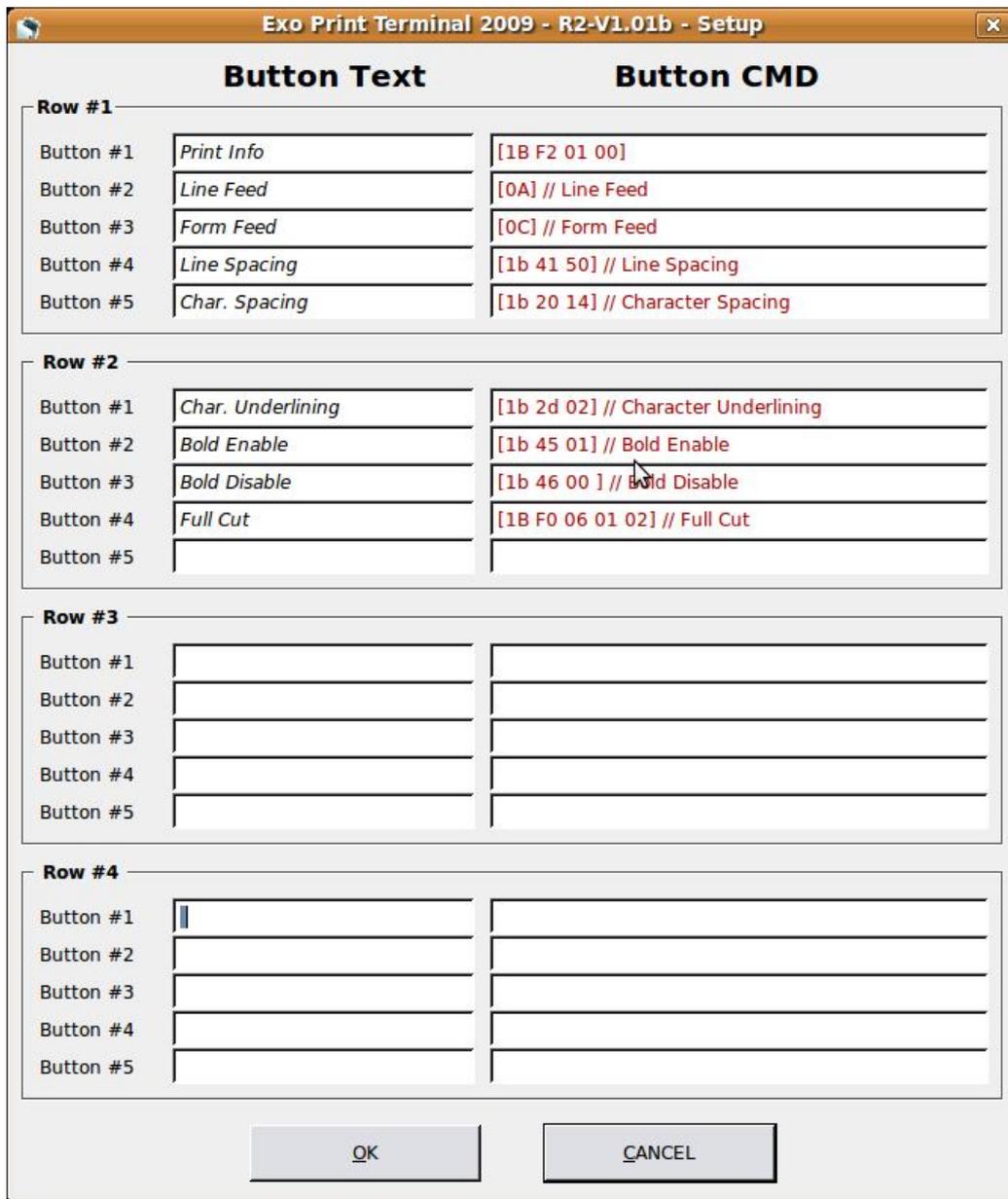
Figure 28: The Exo Print Terminal-setup dialog

The Exo Print Terminal tool allows simultaneous input in TEXT and HEX mode whereas the mode is selected by special character combinations:

- **\X switches to hex mode.**
  All hex digit pairs entered afterwards represent HEX values that are converted to bytes when the data is transmitted to the printer.
  Sample: »\X0a0A0a« are three linefeeds whereas 'LF' 'LF' 'LF' is send to the printer

- **[ ] start and end of hex sequence.**
  All hex digit pairs between the brackets represent HEX values that are converted to bytes when the data is transmitted to the printer.
  Sample: »**[0a 0A 0a]**« are three linefeeds whereas 'LF' 'LF' 'LF' is send to the printer

HENGSTLER

- **\A switches to text mode**
  All data entered afterwards represents ASCII text that is transmitted to the printer without conversion:
  Sample: »\ATEXT« are 4 characters whereas 'T' 'E' 'X' 'T' is send to the printer

- **"   start and end of text sequence.**
  All characters between quotation marks (e.g. ") are regarded as text characters
  Example: **"[0a]"**   the four characters '[' '0' 'a' ']' are sent to the printer

Other special character combinations are:

- \n   a line feed

- \l   a form feed

- \\   a single backslash

- \"   a single quotation mark

- //   beginning of a comment

Note that, this tool is not available from release R3-V1.02d-BETA – 2015MAY07 or higher.

6.3   The Exo Api Test

The Exo Api Test tool is used to test functionality of Exo API and communication with printer. The tool is started by writing in shell window in directory <root>/Tools/x86 or <root>/Tools/x64 or <root>/Tools/arm based on system type:

```
>>   ./Exo_Api_Test "/path/to/printer"

Example:

>>   ./Exo_Api_Test "/dev/bus/usb/002/011"
```

Source code of the Api Test Tool:

```c
#include <stdio.h>
#include "../../Libraries/Exo_Api.h"
void main (int argc, char *argv[])
   {
    int            iPrinter       =0;
    int            iWritenData    =0;
    unsigned char  cmd_FormFeed[] = {0x0C};                    // Form feed
    unsigned char  cmd_Cut[]      = {0x1B,0xF0,0x06,0x01,0x02};  // Cut cmd_Cut
    char printer_name[32];

    if (argc < 2){
      printf("\nERROR: No printer path available , example usage : Exo_Api_Test
\"/dev/bus/usb/001/011\" \n\n");
      goto FAILURE;
    }
    //-- Copy printer name
    memcpy(printer_name,argv[1], strlen(argv[1]) );
    printer_name[strlen(argv[1])] = '\0';

    //-- Open printer connection
    iPrinter = exo_api_printer_open(&printer_name[0]);

    //-- Check if printer was opened
    if (  iPrinter <0 )
       ) goto FAILURE;

    //-- Clear pending data
    exo_api_printer_clear (iPrinter);

    //-- Write cmd to printer
    if ((iWritenData = exo_api_printer_write(iPrinter, cmd_FormFeed, 1, 5000)) != 1)
       goto FAILURE;

    //-- Write cmd to printer
    if ((iWritenData = exo_api_printer_write(iPrinter, cmd_Cut, sizeof(cmd_Cut), 5000)) !=
sizeof(cmd_Cut))
       goto FAILURE;
FAILURE :
    //-- Close printer
    exo_api_printer_close(iPrinter);
    return;
   }
```

Note that, this tool is not available from release R3-V1.04a-BETA or higher.

6.4　The Exo Inquiry Tool

The Exo Inquiry Tool is used to inquire on printer status. This tool is a console application.

The tool is started by typing in a shell window in directory <root>/Tools/x86 or <root>/Tools/x64 or <root>/Tools/arm based on system type:

```
>>    ./Exo_InquiryTool [-s] [-h] [-v] Printer Code
```

Example:

```
>>    ./Exo_InquiryTool -v -s "/dev/usb/lp0" [3F]

      or

>>    ./Exo_InquiryTool -v -s "/dev/bus/usb/002/003" [3F]
```

XPM Inquiry Tool parameters:

[-s]	=	enables silent mode
[-h]	=	enables help display
[-v]	=	enables verbose output
Printer	=	name of the printer port to which the printer is connected starting with "\\"
Code	=	a 2 hex digit inquiry code enclosed in square brackets [00]

The hex digit inquiry code is a 1 unsigned char bitmask that corresponds in emulation command \"GS A+[n]. The individual bits in the inquiry code generate the following inquiries:

[01] = printer status

[02] = general information regarding the printer (part number, firmware, etc.)

[04] = information regarding the printer sensors,

[08] = information regarding the flash memory (contents, etc.)\n");

[10] = status information regarding the presenter unit\n");

[20] = life data\n");

The inquired data is output as hex data string.

6.5   The Demo Applications

The Demo Application are used to demonstrate functionality of eXtendo API. The tools are started by typing in a shell window in directory <root>/Tools/x86 or <root>/Tools/x64 or <root>/Tools/arm based on system type:

```
>>      ./Exo_Demo_PageLength "Printer path" PageLength

>>      ./Exo_Demo_FormFeed "Printer path"

>>      ./Exo_Demo_InquiryStatus "Printer path"
```

Example:

```
>>      ./Exo_Demo_PageLength "/dev/bus/usb/003/025" 1255

>>      ./Exo_Demo_FormFeed "/dev/bus/usb/003/025"

>>      ./Exo_Demo_InquiryStatus "/dev/bus/usb/003/025"
```

All these demo applications in driver package are available also with sources and description how to compile it and evaluate on each environment. Feel free to use and modify them.

LIST OF FIGURES