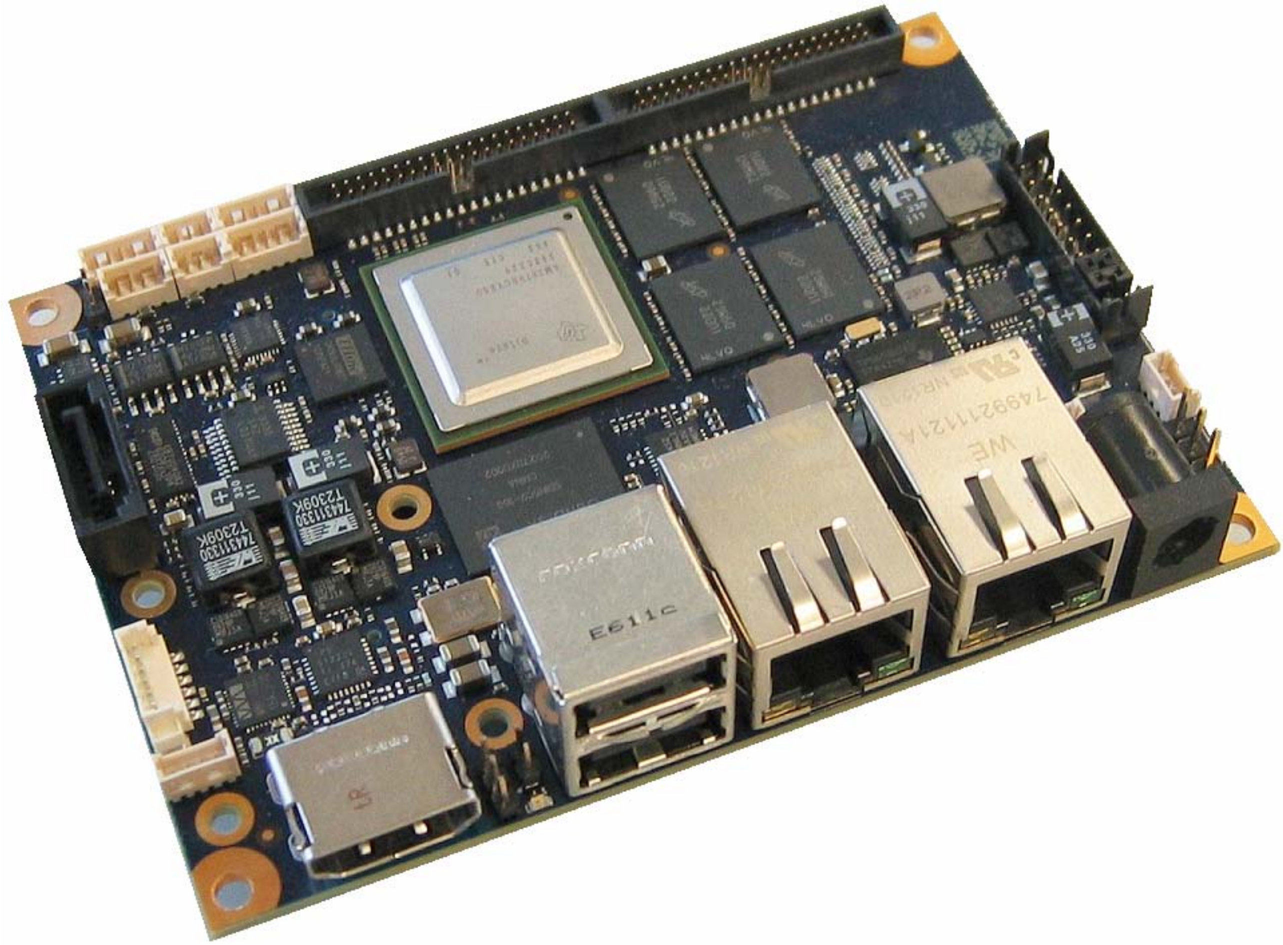
» Kontron Software Guide «



KTAM3874/pITX

KTD-S0057-I

» Table of Contents «

1 User Information

1.1 About This Document

This document provides information about products from KONTRON Technology A/S and/or its subsidiaries. No warranty of suitability, purpose or fitness is implied. While every attempt has been made to ensure that the information in this document is accurate the information contained within is supplied "as-is" - no liability is taken for any inaccuracies. Manual is subject to change without prior notice.

KONTRON assumes no responsibility for the circuits, descriptions and tables indicated as far as patents or other rights of third parties are concerned.

1.2 Copyright Notice

1.3 Trademarks

Brand and product names are trademarks or registered trademarks of their respective owners.

1.4 Standards

KONTRON Technology A/S is certified to ISO 9000 standards.

1.5 Warranty

This product is warranted against defects in material and workmanship for the warranty period from the date of shipment. During the warranty period KONTRON Technology A/S will at its discretion decide to repair or replace defective products.

Within the warranty period the repair of products is free of charge as long as warranty conditions are observed.

The warranty does not apply to defects resulting from improper or inadequate maintenance or handling by the buyer, unauthorized modification or misuse, operation outside of the product's environmental specifications or improper installation or maintenance.

KONTRON Technology A/S will not be responsible for any defects or damages to third party products that are caused by a faulty KONTRON Technology A/S product.

1.6 Life Support Policy

KONTRON Technology's products are not for use as critical components in life support devices or systems without express written approval of the general manager of KONTRON Technology A/S.

As used herein:

Life support devices or systems are devices or systems which

a) are intended for surgical implant into body or

b) support or sustain life and whose failure to perform, when properly used in accordance with instructions for use provided in the labelling, can be reasonably expected to result in significant injury to the user.

A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

1.7 Technical Support

Please consult our web site at **http://www.kontron.com/support** for the latest product documentation, utilities, drivers and support contacts or use the special e-mail address **sbc-support@kontron.com** for a technical problem. In any case you can always contact your board supplier for technical support.

Before contacting support please be prepared to provide as much information as possible:

Board identification:

- ❐ Type
- ❐ Part number (find PN on label)
- ❐ Serial number (find SN on label)

System environment:

- ❐ O/S type and version
- ❐ Driver origin and version
- ❐ Attached hardware (drives, USB devices, LCD panels ...)

2 U-Boot Setup

The sense of a special Setup part is to avoid expendable changes in the proper operating systems. The Setup entries are valid for all supported operating systems (e.g. Linux®, Android™ and Windows® Embedded Compact). For example if you switch from Linux® to Android™ or vice versa in the ideal case no changes will be necessary. The Setup data records are stored in a non-volatile memory (EEPROM) and not in an erasable script. A possible password input protects from unauthorized access.
The KONTRON Setup provides two configurable main menus:

- **display**
- **devices**

The display menu allows to define one or more boot displays, resolution and mapping of a LCD panel and backlight parameters. The devices part involves some hardware device settings, e.g. audio controller enabled or disabled.

ATTENTION

Only the original KONTRON BSPs guarantee the realization of the U-Boot Setup features.

2.1 Setup Command

For a help screen type the command without an argument

setup

Now you can see all the supported sub-commands (menus). Normally the following entries are displayed

❒	**display**	display settings
❒	**devices**	onboard device configuration
❒	**password**	password input
❒	**defaults**	reset all settings to the default values
❒	**summary**	show all actual settings
❒	**save**	save all settings to EEPROM
❒	**showdid**	show the DisplayID™ settings

Syntax example: **setup display** <Enter>

2.2 Setup Usage

After the execution of a sub-command (e.g. setup devices) the selection of a submenu only requires a numeric value. Thereafter the real settings are visible. Now you can choose between an alphanumeric (default) or a numeric input. The alphanumeric presentation illustrates intuitive the right choice. If you favor the numeric input delete all chars with the <Backspace> key, type the number and then press the <Enter> key.
After the completion of all changes it is reasonable to control the settings with the summary command.
Syntax: **setup summary** <Enter>

2.3 Display Menu

This menu part includes several display settings:

- **First boot display**
- **Second boot display**
- **Boot display mirror mode**
- **LCD panel resolution**
- **LCD panel mapping**
- **Backlight output control**
- **Backlight enable level**
- **Backlight brightness**

2.3.1 Boot Display

The TI® AM3874 implies a graphics subsystem with two independent HD display controllers. Without a restriction (except duplicate usage but 'none' on both settings is possible) each controller interface can be configured as:

- **none** switch off the display controller
- **lcd**
- **hdmi**

Examples:

First boot display	hdmi
Second boot display	none
or	
First boot display	none
Second boot display	lcd
or	
First boot display	hdmi
Second boot display	lcd
or	
First boot display	lcd
Second boot display	hdmi

2.3.2 Mirror Mode

This feature is only meaningful if both boot displays are active. **In enabled state always the first boot display determines the real resolution**. In this case it would be advisable to use the LCD panel (setting 'lcd') as first boot display. If mirror mode is disabled the second boot display remains black but it can be used with custom programs.

2.3.3 LCD Panel Resolution

You have the choice to select a panel resolution with a fixed timing or a special setting 'auto' for a free definable timing based on the VESA® DisplayID™ specification. For further details about DisplayID™ see the chapter 'VESA® DisplayID™'. The KTAM3874/pITX supports following resolutions:

❐ **auto**	free timing based on VESA® DisplayID™
❐ **vga**	fixed timing 640x480 pixel, 18 bit color depth, single channel
❐ **wvga**	fixed timing 800x480 pixel, 18 bit color depth, single channel
❐ **svga**	fixed timing 800x600 pixel, 18 bit color depth, single channel
❐ **xga**	fixed timing 1024x768 pixel, 24 bit color depth, single channel
❐ **sxga**	fixed timing 1280x1024 pixel, 24 bit color depth, dual channel
❐ **uxga**	fixed timing 1600x1200 pixel, 24 bit color depth, dual channel
❐ **fullhd**	fixed timing 1920x1080 pixel, 24 bit color depth, dual channel

2.3.4 LCD Panel Mapping

There are two interface modes existing at 24-bit color depth: **FPDI** (Flat Panel Display Interface) or **LDI** (LVDS Display Interface). For 18-bit color depth this setting has no validity. More information can be found in chapter ' VESA® DisplayID™/24 Bit Color Mapping Tips'.

- ❐ **fpdi**
- ❐ **ldi**

2.3.5 Backlight Output Control

Almost all LCD panels support either analog or PWM backlight brightness, therefore two options are available.

❐ **analog**	
❐ **pwm**	frequency approx. 600 Hz

2.3.6 Backlight Output Level

Some backlight inverters need a low level for the enable signal, other inverters a high level. Use this submenu to configure the right enable output level.

❐ **low**	voltage = 0V
❐ **high**	voltage = +3.3V

2.3.7 Backlight Brightness

This submenu allows the definition of the analog backlight brightness (voltage range: 0V to +5V). The input format is represented by a decimal number with maximal three digits.

Examples:

Brightness: 0	minimal value = 0V
or	
Brightness: 255	maximal value = +5V

2.4 Devices Menu

This menu part defines several hardware device settings:

- mPCIe port settings
- Audio settings
- LAN ports settings
- SATA settings
- CAN port 0 settings
- CAN port 1 settings
- UART port 5 settings
- First GPIO port settings
- Second GPIO port settings

2.4.1 mPCI Express® Port

The selection is limited to the enable respectively disable feature.

- disabled
- enabled

2.4.2 Audio Device

The selection is limited to the enable respectively disable feature.
Note: This setting only concerns the audio codec and not the HDMI® audio part.

- disabled
- enabled

2.4.3 LAN Devices

The selection is limited to the enable respectively disable feature.
Note: This setting affects both LAN controllers.

- disabled
- enabled

2.4.4 S-ATA® Device

The selection is limited to the enable respectively disable feature.

- disabled
- enabled

2.4.5 CAN Port 0

The selection is limited to the enable respectively disable feature.

- ❐ **disabled**
- ❐ **enabled**

2.4.6 CAN Port 1

The selection is limited to the enable respectively disable feature.

- ❐ **disabled**
- ❐ **enabled**

2.4.7 UART Port 5

You can choose between three modes:

- ❐ **rs232**
- ❐ **rs422**
- ❐ **rs485**

2.4.8 First GPIO Port

All interface signals can be defined as GPIOs (General Purpose Input Output) or some special signals have another function (Timer 4 to 7). For a detailed overview about these signals see the KTAM3874/pITX User's Guide chapter 'Digital I/O Interface'. The signals are named GPIO21 to 24.

Note: the operating systems do not support these special functions because the possible applications can be too different.

The main menu shows following options:

- ❐ **Multiplexed GPIO/TIMER4+6 part**
- ❐ **Multiplexed GPIO/TIMER5+7 part**

The submenu allows the selection between:

- ❐ **gpio**
- ❐ **timer**

2.4.9 Second GPIO Port

All interface signals can be defined as GPIOs (General Purpose Input Output) or some special signals have another function (UART2, UART4 or SPI3). For a detailed overview about these signals see the KTAM3874/pITX User's Guide chapter 'Digital I/O Interface'. The signals are named GPIO12 to 18 resp. GPIO4 to 7.

The main menu shows following options:

- ❐ **Multiplexed GPIO/UART2 part**
- ❐ **Multiplexed GPIO/UART4 part**
- ❐ **Multiplexed GPIO/SPI3 part**

The submenu for UART2 and UART4 allows the selection between:

- **gpio**
- **uart**

and the submenu for SPI3:

- **gpio**
- **spi**

2.5 Password Command

If you want to control the access to the Setup settings it is possible to use a password protection. Maximal eight alphanumeric chars, numbers or special characters are admissible. You can delete an old password respectively cancel the password protection with the input of an empty string.

Syntax: **setup password** <Enter>

Example:

```
New password: ********          e.g.  12%&fgWQ
Verify password: ********         the same input
```

2.6 Defaults Command

In some cases it can be useful to reset quickly the Setup settings. For an example there is a problem with driving of a single display - preferably a LCD panel - and the connection of a HDMI® monitor is possible.

Syntax: **setup defaults** <Enter>

```
First boot display              hdmi
Second boot display             none
Boot display mirror mode        enabled
LCD panel resolution            auto
LCD panel mapping               fpdi
Backlight output control        pwm
Backlight enable level          high
Backlight brightness            128
mPCI Express interface          enabled
Audio interface                 enabled
LAN interfaces                  enabled
SATA interface                  enabled
CAN interface 0                 enabled
CAN interface 1                 enabled
UART interface 5                rs232
GPIO/TIMER4+6 interface         gpio
GPIO/TIMER5+7 interface         gpio
GPIO/UART2 interface            gpio
GPIO/UART4 interface            gpio
GPIO/SPI3 interface             gpio
```

2.7 Save Command

This is one of the most important sub-commands. Without this calling all Setup changes are lost after power off. The save instruction writes the temporary Setup settings into the non-volatile memory device (EEPROM).

Syntax: **setup save** <Enter>

2.8 Summary Command

This Setup command gives a quick overview about all actual settings.

Syntax: **setup summary** <Enter>

Example:

```
DISPLAY PART:
     First boot display              : hdmi
     Second boot display             : lcd
     Boot display mirror mode        : enabled
     LCD panel resolution            : sxga
     LCD panel mapping               : ldi
     Backlight output control        : pwm
     Backlight enable level          : high
     Backlight brightness            : 255
DEVICES PART:
     mPCI Express interface          : enabled
     Audio interface                 : disabled
     LAN interfaces                  : enabled
     SATA interface                  : disabled
     CAN interface 0                 : enabled
     CAN interface 1                 : enabled
     UART interface 5                : rs485
     GPIO/TIMER4+6 interface         : gpio
     GPIO/TIMER5+7 interface         : gpio
     GPIO/UART2 interface            : uart
     GPIO/UART4 interface            : gpio
     GPIO/SPI3 interface             : gpio
```

2.9 ShowDID Command

The showdid command provides the necessary information for checking the DisplayID™ values. Some additional messages are possible, for example:

```
No DisplayID record present !
ATTENTION: DisplayID record is not activated !
```

Syntax: **setup showdid** <Enter>

Example:

```
DisplayID data:
----------------------------
    Type I Timing Data Block
      Pixel clock                : 83500000 Hz
      Horizontal active          : 1280 clocks
      Horizontal sync start      : 1430 clocks
      Horizontal sync end        : 1530 clocks
      Horizontal total           : 1680 clocks
      Horizontal sync pol.       : positive
      Vertical active            : 800 lines
      Vertical sync start        : 815 lines
      Vertical sync end          : 816 lines
      Vertical total             : 831 lines
      Vertical sync pol.         : positive
    Display Device Data Block
      Color depth                : 24 bit
    Display Interface Data Block
      Number of channels         : 1
      Color mapping              : FPDI
```

3 U-Boot Environment

You can use the U-Boot environment to configure the behavior of U-Boot as you like. Default settings:

```
autoload=yes
baudrate=115200
bootcmd=run sdboot;run mmcboot;run spiboot;run nfsboot
bootdelay=3
bootfile=uImage
bootmmc=setenv root_bootargs root=/dev/mmcblk${device}p${rootpart} rw rootwait;setenv interface mmc;
                                                  mmc rescan ${device} && run fsdetect
cpuspeed=cpu_maxfreq=800
cputype=AM3874
debug_bootargs=earlyprink
default_bootargs=console=ttyO0,115200n8 vram=50M notifyk.vpssm3_sva=0xBF900000 ti814xfb.vram=0:16M,1:16M
                                                  noinitrd --no-log
doboot=run setmemarg;run loadscript;run setbootarg;run loadkernel;
ethaddr=xx:xx:xx:xx:xx:xx
eth1addr=xx:xx:xx:xx:xx:xx
ethact=cpsw
extra_bootargs=bootdisp1=hdmi bootdisp2=none mirrordisp=1 pcie=1 audio=1 lan=1 sata=1 can0=1 can1=1
                                                  uart5_mode=rs232 tm46_enable=0 tm57_enable=0
                                                  uart2_enable=0 uart4_enable=0 spi3_enable=0
fsdetect=setenv fsloader ext2load ${interface} ${device};fatinfo ${interface} ${device} && setenv fsloader fatload ${interface}
                                                  ${device};run doboot
lcdmode=0,0,0,0,0,0,0
loadaddr=0x80009000
loadkernel=${fsloader} ${loadaddr} ${bootfile} && bootm ${loadaddr}
loadscript=${fsloader} ${scriptaddr} ${scriptfile} && source ${scriptaddr}
mmcboot=setenv device 1; setenv rootpart 1; run bootmmc
memsize=2048
nfsboot=setenv autoload no; dhcp;setenv root_bootargs root=/dev/nfs ${nfsroot} ip=${ipaddr}:${serverip}:${gatewayip}:
                                                  ${netmask}:${hostname}:eth0:off;setenv fsloader tftp;run doboot;
nfsroot=nfsroot=192.168.1.1:/nfsroot
```

```
ramdisk_file=ramdisk.gz
rootpart=2
scriptaddr=0x80900000
scriptfile=boot.scr
sdboot=setenv  device 0; setenv rootpart 2; run bootmmc
setbootarg=setenv lcd_bootargs lcdmode=${lcdmode} videomode=${videomode};setenv bootargs ${default_bootargs}
                                        ${lowmem} ${highmem} ${debug_bootargs} ${root_bootargs}
                                        ${lcd_bootargs} ${extra_bootargs} ${cpuspeed}
                                        eth0=${ethaddr} eth1=${eth1addr}
setmemarg=setenv lowmem mem=417M;test ${cputype} = DM8148 && setenv lowmem mem=364M;setenv highmem
                                        mem=320M@0x9FC00000;test ${memsize} = 2048 &&
                                        setenv highmem ${highmem} mem=1023M@0xC0000000;
                                        setenv highmem ${highmem} vmalloc=512M
spiboot=setenv root_bootargs root=/dev/mmcblk0p${rootpart} rw rootwait;run setbootarg;sf probe 0 && sf read ${loadaddr}
                                        0xE2000 0x31E000 && bootm ${loadaddr}
verify=yes
videomode=25180000,640,656,752,800,480,490,492,525,0,1,1
```

ATTENTION

Do not change the arguments in red color because U-Boot generates these parameters itself.

The U-Boot command 'printenv' lists all variables, 'setenv' modifies the values and 'saveenv' stores the new environment.

4 VESA® DisplayID™

4.1 LCD/LVDS Technology Overview

4.1.1 Detailed Timing Descriptor

The input fields Pixel Clock, Horizontal Active, Horizontal Blank, Horizontal Sync Offset, Horizontal Sync Width, Vertical Active, Vertical Blank, Vertical Sync Offset and Vertical Sync Width must be filled in with the correct values according to the panel's data sheet. In many cases the value for Horizontal/Vertical Blank cannot be read directly from the data sheet. Instead terms such as Display Period (active pixels/lines) or Horizontal/Vertical Total appear.
In this case the following calculation can be made:

⇒ **Blank Value = Total Value – Active Value.**

Sometimes the datasheet does not specify Sync Offset and/or Sync Width. In this case the permissible values can only be determined through testing. However the rule is:

⇒ **The sum of Sync Offset and Sync Width must not exceed the value for Horizontal/Vertical Blank.**

Also datasheets are often different for displays with double pixel clock. If Pixel Clock and Horizontal Values seem to be halved this must be corrected for input:

⇒ **The values must always be entered as though it were a panel with single pixel clock.**

Example 1:

PRIMEVIEW PM070WL4 (single pixel clock)
Data sheet specifications:

Clock Frequency [typ.]	32 MHz	
HSync Period [typ.]	1056 Clocks	(equivalent to Horizontal Total)
HSync Display Period [typ.]	800 Clocks	(equivalent to Horizontal Active)
HSync Pulse Width [typ.]	128 Clocks	
HSync Front Porch [typ.]	42 Clocks	
HSync Back Porch [typ.]	86 Clocks	
VSync Period [typ.]	525 Lines	(equivalent to Vertical Total)
VSync Display Period	480 Lines	(equivalent to Vertical Active)
VSync Pulse Width [typ.]	2 Lines	
VSync Front Porch [typ.]	10 Lines	
VSync Back Porch [typ.]	33 Lines	

Result:

Pixel Clock	32	
Horizontal Active	800	
Horizontal Blank	256	((128 + 42 + 86) → H. Pulse Width + H. Front Porch + H. Back Porch)
Horizontal Sync Offset	42	(H. Front Porch)
Horizontal Sync Width	128	(H. Pulse Width)
Vertical Active	480	
Vertical Blank	45	((2 + 10 + 33) → V. Pulse Width + V. Front Porch + V. Back Porch)
Vertical Sync Offset	10	(V. Front Porch)
Vertical Sync Width	3	(V. Pulse Width)

Example 2:

SHARP LQ190E1LW01 (double pixel clock)
Data sheet specifications (no definition of Sync Offset and Sync Width):

Clock Frequency [typ.]	54 MHz	
Horizontal Period (1) [typ.]	844 Clocks	(equivalent to Horizontal Total)
Horizontal Display Period	640 Clocks	(equivalent to Horizontal Active)
Vertical Period [typ.]	1066 Lines	(equivalent to Vertical Total)
Vertical Display Period	1024 Lines	(equivalent to Vertical Active)

Result:

Pixel Clock	108	(2 x 54 MHz)
Horizontal Active	1280	(2 x 640 Clocks)
Horizontal Blank	408	((844 – 640) x 2 Clocks)
Horizontal Sync Offset	45	(normally approx. 10 – 15 % of Horizontal Blank)
Horizontal Sync Width	140	(normally approx. 30 – 70 % of Horizontal Blank)
Vertical Active	1024	
Vertical Blank	42	(1066 – 1024 Lines)
Vertical Sync Offset	1	(normally approx. 1 – 3 Lines)
Vertical Sync Width	3	(normally approx. 1 – 15 Lines)

Example 3:

LG-PHILIPS LM170E01-TLA1 (double pixel clock)
Data sheet specifications:

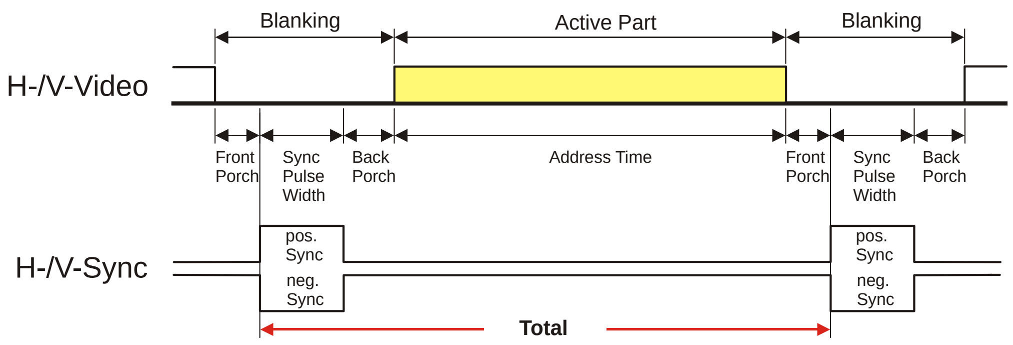
Clock Frequency [typ.]	54 MHz
Hsync Period [typ.]	844 Clocks
Horiz. Valid [typ.]	640 Clocks
Horiz. Back Porch [typ.]	124 Clocks
Horiz. Front Porch [typ.]	24 Clocks
Vsync Period [typ.]	1066 Lines
Vert. Valid [typ.]	1024 Lines
Vert. Back Porch [typ.]	38 Lines
Vert. Front Porch [typ.]	1 Line

Result:

Pixel Clock	108	(2 x 54 MHz)
Horizontal Active	1280	(2 x 640 Clocks → Horizontal Addr. Time)
Horizontal Blank	408	((844 – 640) x 2 Clocks)
Horizontal Sync Offset	48	(2 x 24 Clocks → Horizontal Front Porch)
Horizontal Sync Width	112	(((408/2 – 124 – 24) x 2) → H. Blank – H. Back Porch – H. Front Porch)
Vertical Active	1024	(Vertical Addr. Time)
Vertical Blank	42	(1066 – 1024 Lines)
Vertical Sync Offset	1	(Vertical Front Porch)
Vertical Sync Width	3	(Vertical Blank – Vertical Back Porch – Vertical Front Porch)

The following picture shows the typical video timing.

Timing Parameter Definitions



4.1.2 24 Bit Color Mapping Tips

The double pixel clock or 24-bit color depth can generally be taken from the datasheet. There are two interface modes existing at 24-bit color depth: **FPDI** (Flat Panel Display Interface) or **LDI** (LVDS Display Interface). Some panels use the line SELL LVDS (SELect Lvds data order). The LVDS data assignment in the datasheet can give you an indication by the last channel (e.g. RX3/TX3 – SELL LVDS = low) wether it is a LDI panel (contains the lowest bits).

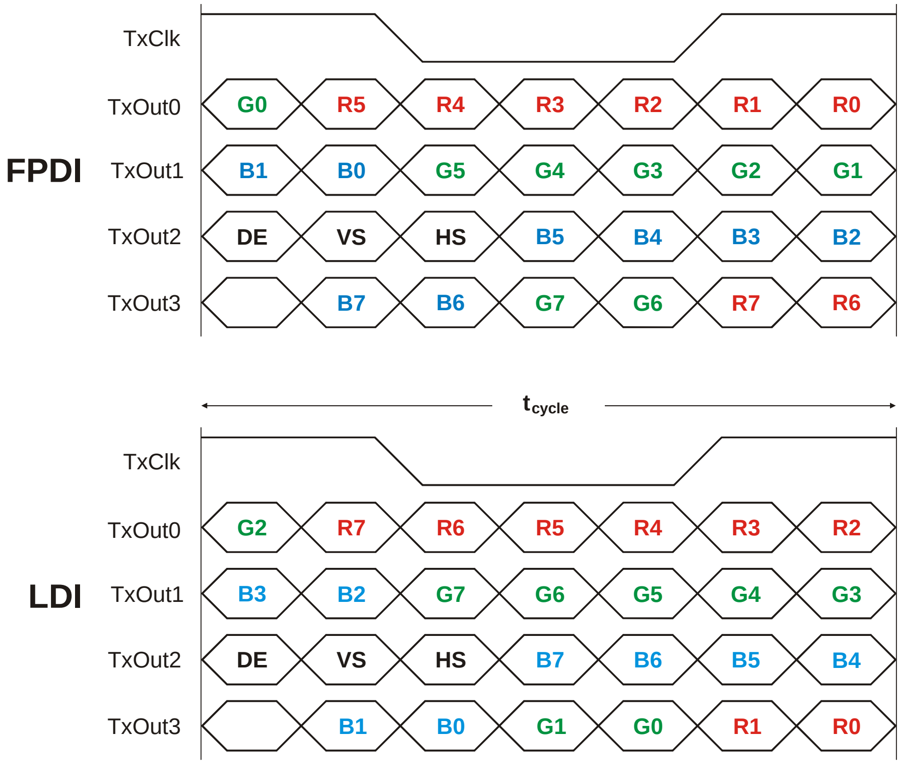
Example:

FPDI data assignment (LVDS channel 3 even or odd):

Tx/Rx27	Red 6 (e.g. even: RE6 or ER6)
Tx/Rx5	Red 7
Tx/Rx10	Green 6 (e.g. even: GE6 or EG6)
Tx/Rx11	Green 7
Tx/Rx16	Blue 6 (e.g. even: BE6 or EB6)
Tx/Rx17	Blue 7
Tx/Rx23	not used

LDI data assignment (LVDS channel 3 even or odd):

Tx/Rx27	Red 0 (e.g. even: RE0 or ER0)
Tx/Rx5	Red 1
Tx/Rx10	Green 0 (e.g. even: GE0 or EG0)
Tx/Rx11	Green 1
Tx/Rx16	Blue 0 (e.g. even: BE0 or EB0)
Tx/Rx17	Blue 1
Tx/Rx23	not used


FPDI
TxClk
TxOut0 G0 R5 R4 R3 R2 R1 R0
TxOut1 B1 B0 G5 G4 G3 G2 G1
TxOut2 DE VS HS B5 B4 B3 B2
TxOut3 B7 B6 G7 G6 R7 R6
t_{cycle}
LDI
TxClk
TxOut0 G2 R7 R6 R5 R4 R3 R2
TxOut1 B3 B2 G7 G6 G5 G4 G3
TxOut2 DE VS HS B7 B6 B5 B4
TxOut3 B1 B0 G1 G0 R1 R0

4.2 EDID 1.3 Specification (VESA®)

The EDID (Extended Display Identification Data) record has a fixed structure. The first 8 bytes contain the distinctive identification 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00. The end of the record is marked by the checksum (1 byte). The result of the addition of all bytes including the checksum has to be zero.
For a comprehensive support of the majority of available panels you don't need all fields of the EDID record. The **Detailed Timing Descriptor** (18 bytes) is the most important field. No 24bit panels (FPDI/LDI) are supported though. This means EDID should only be used for 18bit panels.
For further information please consult the official EDID specification from the VESA® comitee which has to be payed.

4.3 DisplayID™ Specification (VESA®)

Intended as a replacement for all previous EDID versions DisplayID™ contains many new features. It's a structure with several well defined elements (tags). Not every element that is listed in the specification has to be part of the resulting data set (basic section).
KONTRON has decided to use this selection of tags (mandatory presence).

Tag	Description
0x00	Product Identification Data Block (Vendor ID, Product Code, Manufacturing Date ...)
0x03	Type I Detailed Timing Data Block (Pixel Clock, Horizontal/Vertical Data ...)
0x0C	Display Device Data Block (Device Technology, Operating Mode, Color Depth ...)
0x0D	Interface Power Sequencing Data Block (Power On/Off Timing)
0x0F	Display Interface Data Block (Interface Type, Interface Attribute ...)

4.3.1 DisplayID™ Parameter Summary

Only a part of the parameters used in the DisplayID™ Windows® tool are interpreted by a specific board. The following table shows a summary of the used parameters (valid for KTAM3874/pITX).

Group	Parameter	Comment
Type I Timing	**Pixel Clock**	
Type I Timing	**Horizontal Active**	
Type I Timing	**Horizontal Blank**	
Type I Timing	**Horizontal Sync Offset**	Front porch
Type I Timing	**Horizontal Sync Width**	
Type I Timing	**Vertical Active**	
Type I Timing	**Vertical Blank**	
Type I Timing	**Vertical Sync Offset**	Front porch
Type I Timing	**Vertical Sync Width**	
Display Interface 1	**Bits per Pixel**	Color depth (18 or 24bit)
Display Interface 1	**Pixel per Clock**	Single or dual channel
Display Interface 1	**24 Bit Color Mapping**	FPDI or LDI

4.3.2 DisplayID™ Restrictions

Depending on the graphic controller not all features can be used. The following table shows the most important restrictions.

Restrictions for KTAM3874/pITX
Variable power sequencing not supported

4.3.3 LCD Panel Selection

The choice of a LCD display is basically defined by two parameters.

Parameter	Value
Pixel per Clock (Channels)	**1 or 2**
Pixel Clock Range	**165 MHz**

Currently this leads to a maximum resolution of

1920 x 1080 Pixel

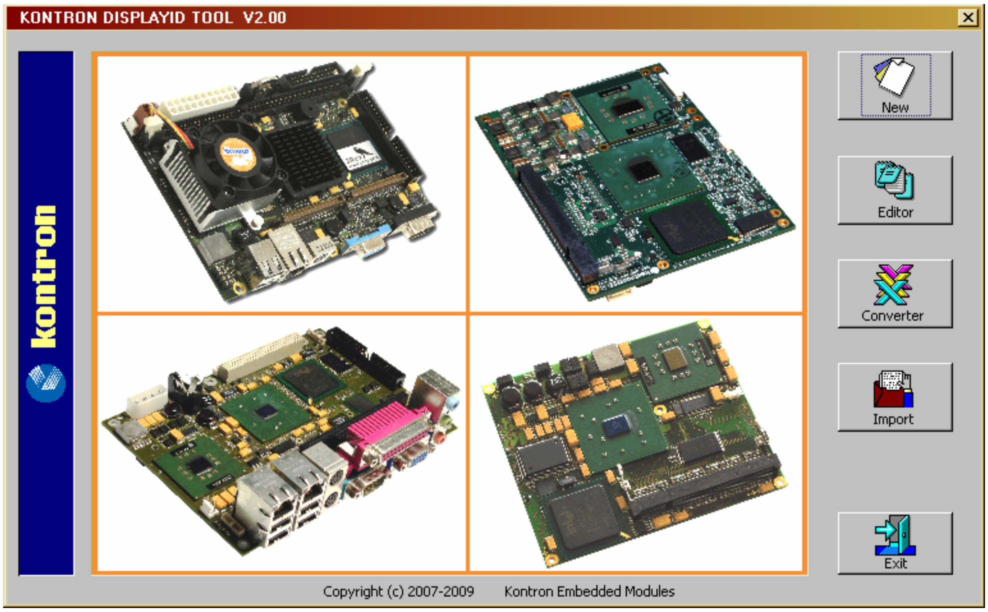
With TIs® graphic driver it is not guaranteed that every resolution can be achieved. KONTRON does not guarantee the correct function of the board for untypical resolution. In principal the use of DisplayID™ allows realizing every special display resolution. For this a valid DisplayID™ dataset must be written to the onboard EEPROM. Additionally the U-Boot Setup entry

setup display → submenu [3] → Set LCD panel resolution

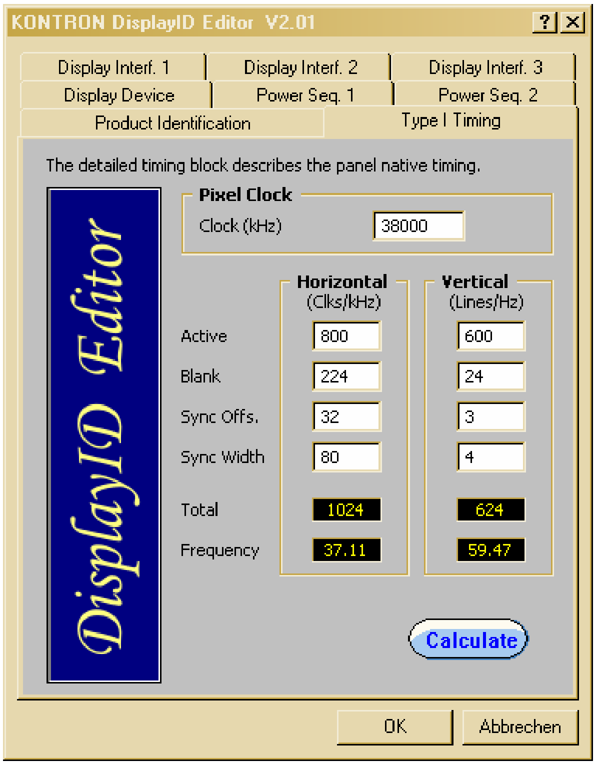
must be set to **auto**.

4.3.4 DisplayID^{TM} Windows$^{®}$ Tool

The DisplayID^{TM} parameter can be modified with the DisplayID^{TM} Windows$^{®}$ tool.


KONTRON DISPLAYID TOOL V2.00
kontron
New
Editor
Converter
Import
Exit
Copyright (c) 2007-2009 Kontron Embedded Modules

For an example the following picture shows the input fields for the **Detailed Timing** parameters.



The detailed timing block describes the panel native timing.

Pixel Clock

Clock (kHz) 38000

	Horizontal (Clks/kHz)	Vertical (Lines/Hz)
Active	800	600
Blank	224	24
Sync Offs.	32	3
Sync Width	80	4
Total	1024	624
Frequency	37.11	59.47

For more information see the documentation of the DisplayID^{TM} tool (software can be downloaded from kontron.com).

The DisplayID™ Editor saves the parameters in a intermediate file format. The file extension is 'KDD' (Kontron DisplayID™ Data). This file format cannot be used to program the onboard EEPROM.
For transferring this file format into the binary file format for the EEPROM apply the Converter.

4.3.5 Building DisplayID™ File

❶ Start the Windows® tool **DisplayID.exe**.
❷ Use the **Editor** if you want to modify an existing DisplayID™ file or select **New** to create a complete new record.
❸ Change respectively enter new parameters.
❹ Save the parameters in a file with the extension 'KDD'.
❺ Open the saved 'KDD'-file using the **Converter**.
❻ Save the binary file with the extension 'KDB' (Kontron DisplayID™ Binary).
❼ Program the onboard EEPROM using the board specific update tool.

4.3.6 Erasing DisplayID™ Record

Create a dummy file with a size of 128 bytes filled with the value 0xFFh and program this file using the U-Boot update tool. This treatment deletes a valid DisplayID™ record.

4.3.7 U-Boot EEPROM Update Tool

The update tool is a new component of the U-Boot bootloader. You need two commands to program a DisplayID™ file into the EEPROM:

❒ ext2load, fatload, loadb or loady
❒ writedid

File Operation

The following example gives an overview:
The storage medium is a microSD™ card formatted with a Linux® partition and the DisplayID™ file **wvga.kdb** is located in the root directory. The file size of **wvga.kdb** amounts 81 bytes.
For loading the file into memory type the following standard U-Boot command lines

```
mmc rescan 0
ext2load mmc 0 81000000 wvga.kdb
```

The memory address (0x81000000) is free selectable. With the md command you can control the result

```
md.b 81000000 80
```

Now you can load the memory content into the EEPROM. Type the new KONTRON U-Boot command

```
writedid 81000000 51
```

The 'count' respectively the size argument is a very important parameter. Do not use another value as the file size of your DisplayID™ file.

Serial Download

The following example demonstrates a serial download via the ymodem protocol:
The KTAM3874/pITX board is connected to a desktop computer with a suitable terminal program (e.g. Hyper-Terminal or TeraTerm). The file size of **wvga.kdb** amounts 81 bytes.
For downloading the file into memory type the following standard U-Boot command line

```
loady
```

Now U-Boot waits for reply. User input to the desktop terminal program starts the download session. After the download of **wvga.kdb** ends you can control the result with the md command

```
md.b  80009000  80
```

Now you can load the memory content into the EEPROM. Type the new KONTRON U-Boot command

```
writedid  80009000  51
```

The 'count' respectively the size argument is a very important parameter. Do not use another value as the file size of your DisplayID^{TM} file.

Note: The memory parameter of md.b and writedid depends on the environment entry loadaddr because loady uses this definition as base address, for example if loadaddr equals 0x81000000 the new KONTRON U-Boot command changes to

```
writedid  81000000  51
```

5 U-Boot Bootloader

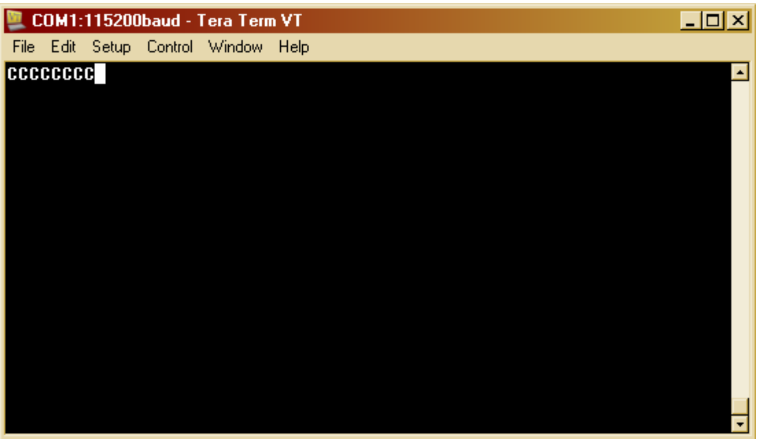
TI® requires a two-part U-Boot version. The ROM code loads the first part (named U-Boot-MIN) into the on-chip prozessor RAM and starts automatically this part. After setting up the clocks and initializing all necessary peripherals U-Boot-MIN loads the final version. The KTAM3874/pITX board offers four different load mechanisms: **SPI^{TM}** (default, see chapter 'SPI^{TM} Bootloader'), MMC, Ethernet and UART.
For development purposes the UART and Ethernet methods promise some advantages.

5.1 UART Bootloader

The first serial port (**UART0**) on connector J1304 provides the serial console interface. For communication a standard terminal program on a desktop computer is necessary (e.g. TeraTerm). The next line shows the default serial console settings:

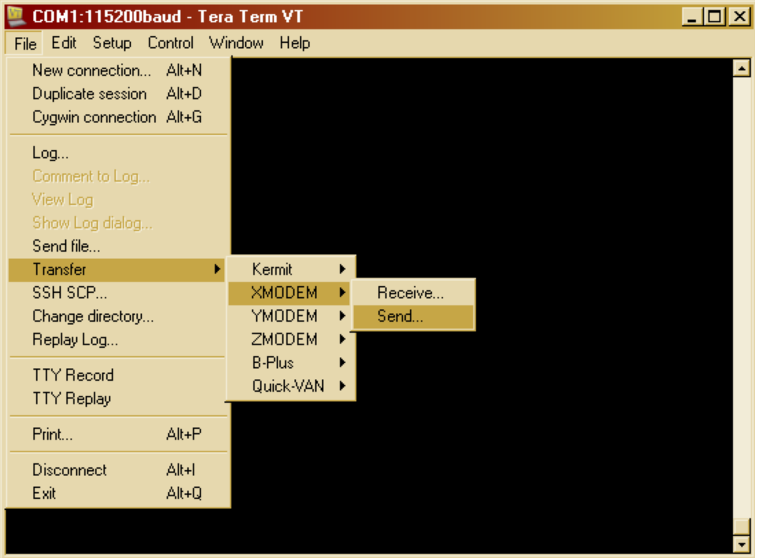
- 115 kBaud / 8 data bits / 1 stop bit / no parity

The UART method strictly needs a jumper on pin header J1101 (see User's Guide chapter 'Serial Console'). After power on the following screen appears:


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

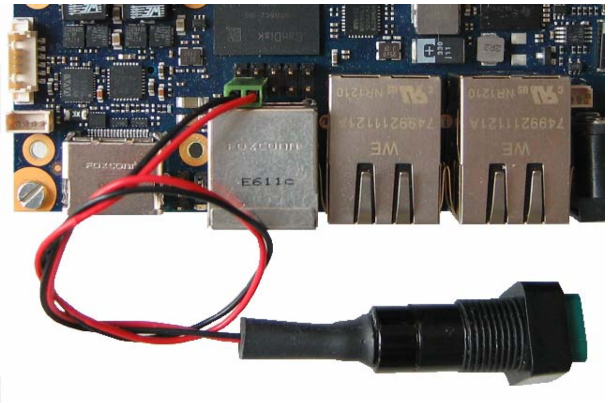
```
CCCCCCCC
```

Now you have to select the XMODEM transfer mode.

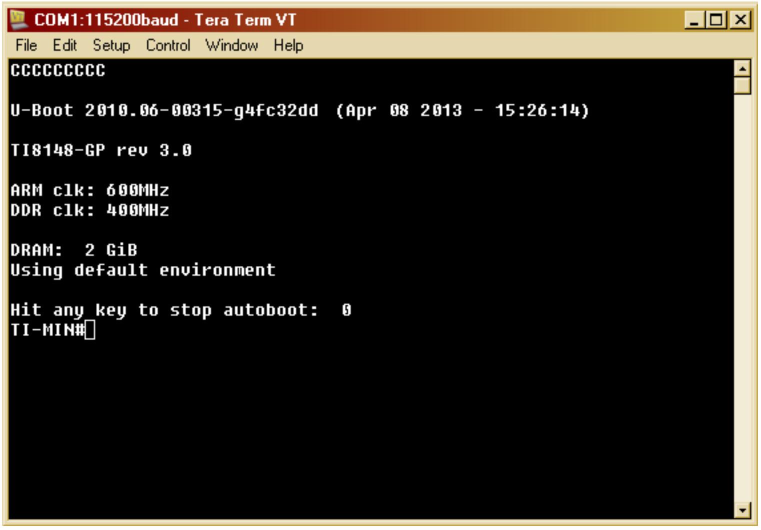


Note: *Do not start the transmission before all eight C's are visible - otherwise the download can be confused.*

Thereafter select the U-Boot-MIN file (normally named 'u-boot.min.uart') and immediately press the Reset button to start the download process.

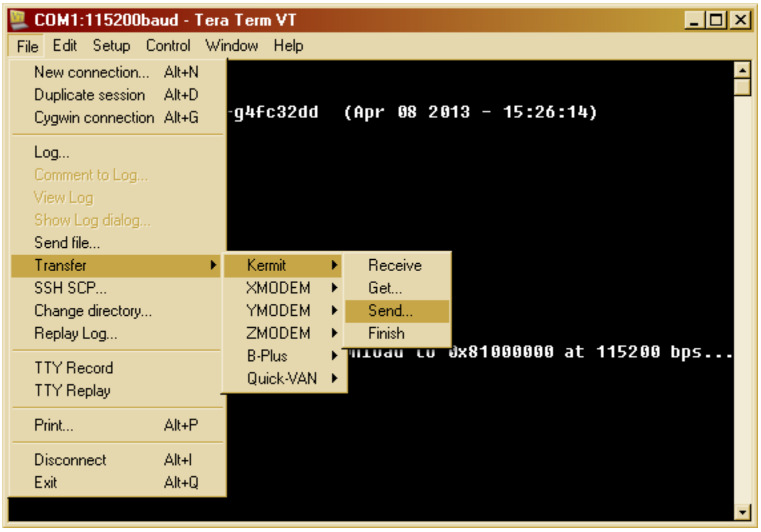


The terminal program shows the following result:


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

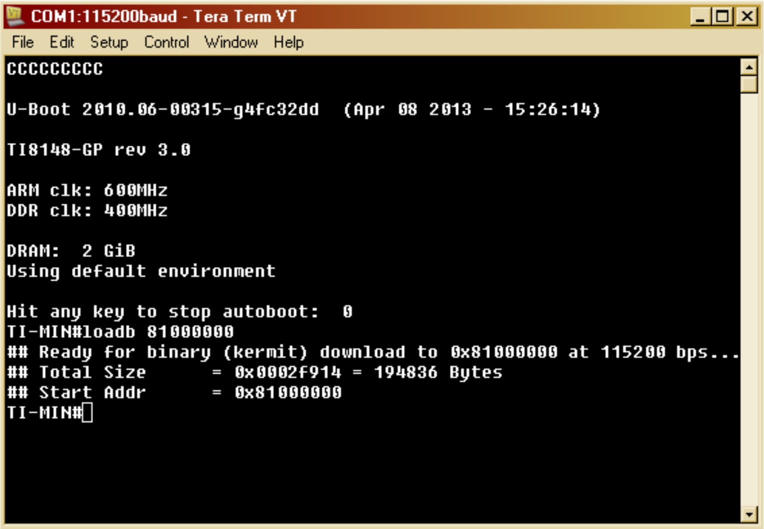
```
CCCCCCCCC

U-Boot 2010.06-00315-g4fc32dd (Apr 08 2013 - 15:26:14)

TI8148-GP rev 3.0

ARM clk: 600MHz
DDR clk: 400MHz

DRAM:  2 GiB
Using default environment

Hit any key to stop autoboot:  0
TI-MIN#
```

Type 'loadb 81000000' at the 'TI-MIN#' prompt and press 'Enter'. Then choose the Kermit transfer mode.



```
-g4fc32dd  (Apr 08 2013 - 15:26:14)

load to 0x81000000 at 115200 bps...
```

The next step is to load the final U-Boot version (normally named 'u-boot.bin'). This time the transfer starts automatically. The result looks as follows:


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

```
CCCCCCCCC

U-Boot 2010.06-00315-g4fc32dd  (Apr 08 2013 - 15:26:14)

TI8148-GP rev 3.0

ARM clk: 600MHz
DDR clk: 400MHz

DRAM:  2 GiB
Using default environment

Hit any key to stop autoboot:  0
TI-MIN#loadb 81000000
## Ready for binary (kermit) download to 0x81000000 at 115200 bps...
## Total Size      = 0x0002f914 = 194836 Bytes
## Start Addr      = 0x81000000
TI-MIN#
```

Now you can start the final U-Boot version with 'go 81000000'.

5.2 Ethernet Bootloader

For download over Ethernet you need a TFTP (Trivial File Transfer Protocol) server with DHCP ability.
Note: Use only the first Ethernet port (J1600).
Some preference changes are necessary:

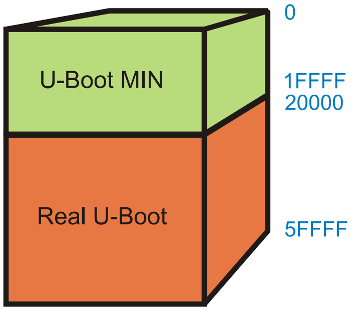
- This method strictly needs a jumper on pin header J1101 (see User's Guide chapter 'Serial Console').
- Rename the U-Boot-MIN file named 'u-boot.min.eth' in 'MLO'.
- Copy 'MLO' and u-boot.bin' into the same directory.

The ROM code tries first to load the U-Boot-MIN part over the serial interface but this approach fails. Afterwards the ROM code etablish the Ethernet connection and loads MLO and subsequently the real U-Boot version. As a result of the serial interface timeout the network connection is rather slowly.

5.3 SPI™ Bootloader

This chapter describes the programming of the SPI™ flash with the serial interface. Alternative you can load both U-Boot versions for example from the microSD™ card.
Assumption: the default boot order is active (SPI™ flash first boot device).

The picture shows the fixed address ranges within the SPI™ flash.

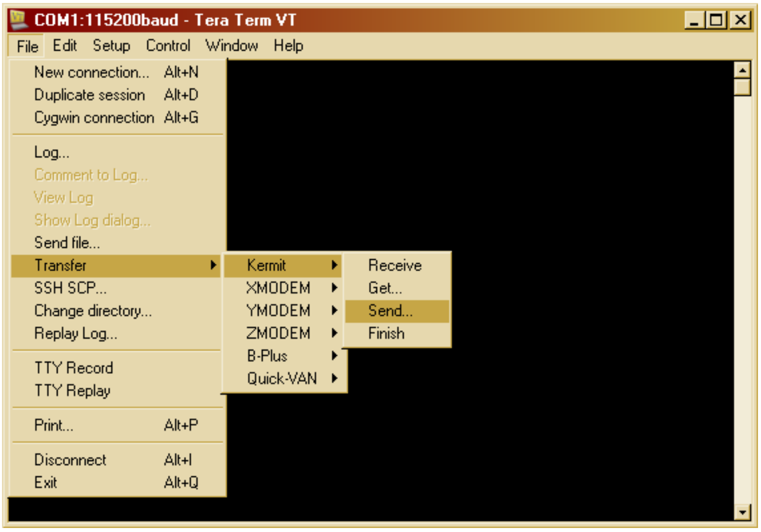


After power on U-Boot starts from SPI™. Hit any key to stop the autoboot process. As a first step you should erase the bootloader part of the SPI™ flash with the following commands:

```
sf probe 0:0
sf erase 0 60000
```

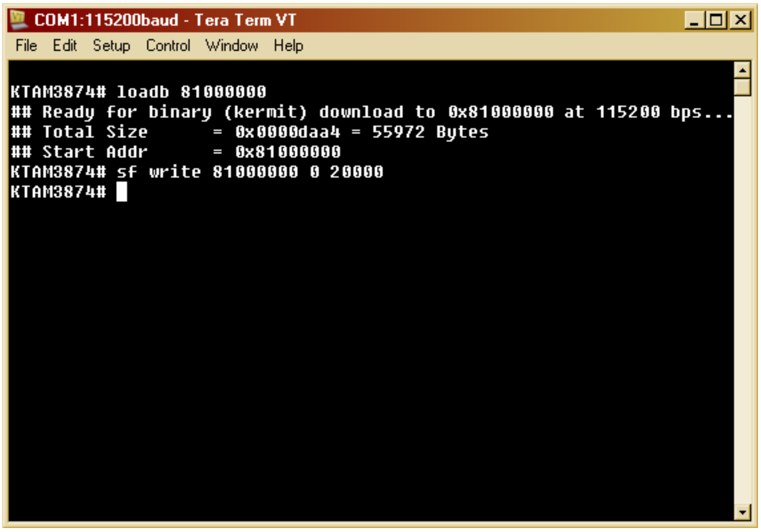
Thereafter type 'loadb 81000000' at the 'KTAM3874#' prompt and press 'Enter'. Then choose the Kermit transfer mode. Before the download you can clear the memory with

```
mw.b 81000000 ff 20000
```



Load the U-Boot-MIN file (normally named 'u-boot.min.spi') and then type

```
sf write 81000000 0 20000
```


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

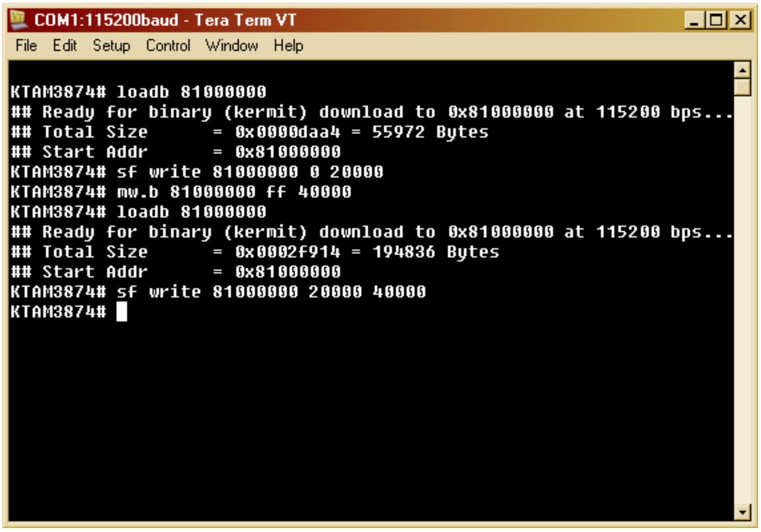
```
KTAM3874# loadb 81000000
## Ready for binary (kermit) download to 0x81000000 at 115200 bps...
## Total Size      = 0x0000daa4 = 55972 Bytes
## Start Addr      = 0x81000000
KTAM3874# sf write 81000000 0 20000
KTAM3874#
```

Now the first update part is finished. For the second part type again 'loadb 81000000' at the 'KTAM3874#' prompt and press 'Enter'. Then choose the Kermit transfer mode. Before the download you can clear the memory with

```
mw.b 81000000 ff 40000
```

Load the real U-Boot file (normally named 'u-boot.bin') and then input with another offset and size

```
sf write 81000000 20000 40000
```


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

```
KTAM3874# loadb 81000000
## Ready for binary (kermit) download to 0x81000000 at 115200 bps...
## Total Size      = 0x0000daa4 = 55972 Bytes
## Start Addr      = 0x81000000
KTAM3874# sf write 81000000 0 20000
KTAM3874# mw.b 81000000 ff 40000
KTAM3874# loadb 81000000
## Ready for binary (kermit) download to 0x81000000 at 115200 bps...
## Total Size      = 0x0002f914 = 194836 Bytes
## Start Addr      = 0x81000000
KTAM3874# sf write 81000000 20000 40000
KTAM3874#
```

After power off and power on U-Boot-MIN loads automatically the second part. No additional steps are necessary.

5.4 MMC Bootloader

For booting from the microSD™ card you have to change some preferences:

- ❐ Replug the boot order jumper to MMC ➔ SPI ➔ ... (see User's Guide chapter 'Boot Order').
- ❐ Rename the U-Boot-MIN file named 'u-boot.min.sd' in 'MLO'.
- ❐ Format the microSD™ card with FAT16 or FAT32 (Linux® partitions are not accepted). For previous Linux® formatted cards it would be advisable to use for example the 'HP® USB Disk Storage Format Tool' otherwise the detection can fail.

  Note: You need an active FAT partition.
- ❐ Copy 'MLO' and u-boot.bin' to the microSD™ card.

After power on U-Boot-MIN (MLO) loads automatically the second part.

6 Debian™ Linux® BSP

6.1 User Login Arguments

For both, serial remote system or terminal root privileges with 'sudo', the arguments are the same (necessary input in green color):

ktam3874 login: **ktam3874**
Password: **ktam3874**

6.2 Video Decoding

The KTAM3874/pITX is not qualified for video decoding, the major focus lies on industrial applications. If you still want to apply a video player use MPlayer ('sudo apt-get install mplayer'). Other deployable programs (e.g. Totem Movie Player or SMPlayer, VLC does not work with the default settings) show strong dropouts.
For measuring the graphics performance use the glmark2-es2 benchmark.

6.3 Audio Support

If you have trouble with audio input/output please check the 'alsamixer' settings.
Examples with the command line tool 'amixer':

- ❐ First select a card with amixer -c KTAM3874 (WM8903 codec) or use amixer -c HDMI (HDMI® monitor). Card names are readable in the directory /proc/asound/
- ❐ For an overview type amixer contents
- ❐ To change a value type e.g. amixer cset numid=27 on

You can find some audio files in the directory /usr/share/sounds/. Play these files with the 'aplay' tool, for example aplay login.wav.

6.4 KEAPI Interface

The menu item 'Applications ➔ System Tools ➔ KEAPI' offers the practical KEAPI GUI tool. A password is needed (default: **Kontron**) - this information can also be detected in the 'Kontron EAPI for Linux' documentation but rather hidden.
The KEAPI GUI includes four important components: SerialBus, GPIO, Perf(ormance) and Misc(ellaneous). Three I2C™ busses are selectable in the SerialBus tab:

- ❐ #0 for onboard devices. Normally the access to these devices should be avoided - further information on request.
- ❐ #1 for the HDMI® DDC EEPROM. The graphic driver blocks the access.
- ❐ #2 provides free access to the I2C™ bus at J2105 (pins 3 and 4).

For GPIO pin assignment see the KTAM3874/pITX User's Guide (chapter 'Digital I/O Interface'). The Perf tab only allows the switch between 600 and 800 MHz (all available frequencies are listed in the file '/sys/devices/system/cpu/cpu0/cpufreq/scaling_available_frequencies'). Finally the Misc tab includes the Watchdog and System State control (only RESTART possible - TURN OFF and HIBERNATE without function).

6.5 CAN Bus Utilities (SocketCAN)

The directory /usr/bin contains all necessary CAN bus utilities you may need. An overview:

Utility	Short Description
canbusload	statistic tool
can-calc-bit-timing	CAN baudrate calculation
candump	shows the received message from the CAN bus, e.g. candump can0
canfdtest	full-duplex test program (slave and host part)
cangen	CAN frames generator for testing purpose
cangw	CAN-to-CAN gateway to route (and modify) messages between multiple buses
canlogserver	capture data as ASCII logfile format
canplayer	send CAN frames from a file to a CAN interface
cansend	send CAN-frames via CAN_RAW sockets, e.g. cansend can0 500#1E.10.10 (500 = CAN ID)
cansniffer	capture and analyze CAN traffic

Additional information can be retrieved in the directory /proc/net/can. If you want to overwrite the default baudrate of 500 kbps change the value in the file /etc/default/can0 resp. can1 or use the following commands (example, requires root rights):

```
sudo ifconfig can0 down
sudo ip link set can0 up type can bitrate 125000
```

After boot up it is not necessary to enter the command ifconfig can0/can1 up because this is already done.
More information about SocketCAN can be found on following webpage:

http://www.mjmwired.net/kernel/Documentation/networking/can.txt

6.6 S-ATA® Interface

The S-ATA® port supports two interfaces: either the standard L-type connector J903 or the mSATA® interface on the mini PCI Express® connector J900. After plug in a mSATA® card the hardware switches automatically from the connector J903 to J900 and disables the mini PCI Express® lanes.

6.7 Suspend Mode

Some of TI's® graphics drivers do not support the suspend/resume feature. Due to this limitation the board cannot be put into suspend mode if graphics output is enabled and active.
To work around this limitation you must stop the X session and unload the graphics driver modules before entering the suspend mode. When resuming load the modules first and start the X session afterwards.

6.8 Screensaver

The X screensaver overlays the Gnome™ screensaver. In order to solve this issue type xset s off in terminal mode and press Enter.

6.9 Sysfs Support

Alternative to the KEAPI interface you can use Sysfs to manipulate the GPIOs or the backlight. The example reads the content and direction of GPIO0 (GP0[10]) and sets the direction to 'output' and the content to 'high'.

```
                                                            Comment:
root@ubuntu:#  echo  10  >  /sys/class/gpio/export          enable GP0[10] access
root@ubuntu:#  cat  /sys/class/gpio/gpio10/value            read content
1                                                           value = high
root@ubuntu:#  cat  /sys/class/gpio/gpio10/direction        read direction
in                                                          direction = input
root@ubuntu:#  echo  "out"  >  /sys/class/gpio/gpio10/direction   set direction to output
root@ubuntu:#  echo  1  >  /sys/class/gpio/gpio10/value     set content to high
root@ubuntu:#  echo  10  >  /sys/class/gpio/unexport        disable GP0[10] access
```

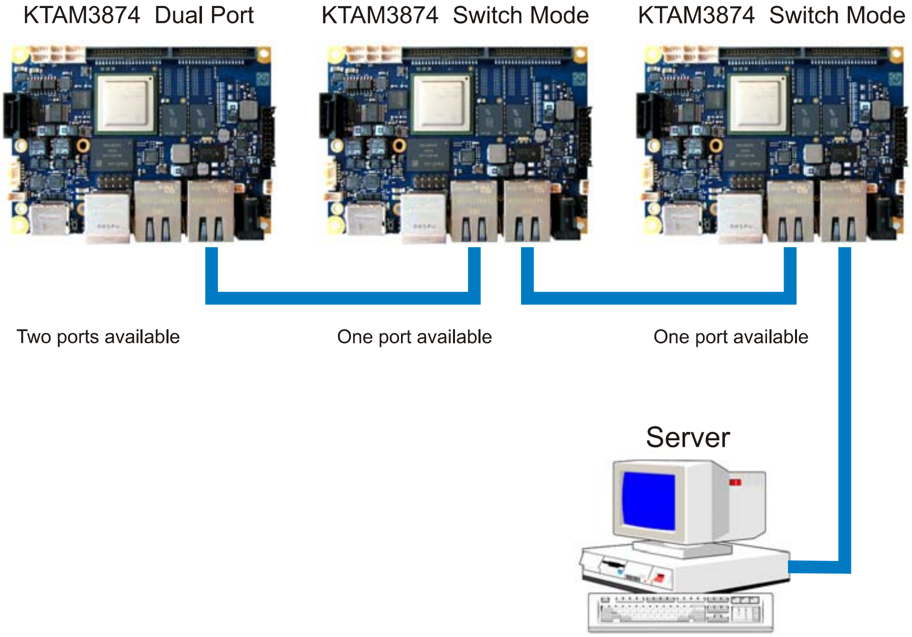
7 Ethernet Switch

TI®'s AM3874 SoC knows two different Ethernet operating modes: the normal dual port and the optional three port switch setting. The switch contains one port with internal connection and two ports which are brought out externally on connector J1600 respectively J1601. You can choose between both modes with an additional U-Boot environment variable (default: dual port mode if the variable is absent).

dual_emac=0 ☚ Three port switch mode
dual_emac=1 ☚ Dual port mode

The environment argument default_bootargs represents a good place for attachment with the U-Boot command editenv. Do not forget to save this setting using the command saveenv.

The following picture shows an example with three boards which have at least one free port. There is no difference between the connectors J1600 and J1601.



For detection of the actual mode use the command ifconfig within Linux®. As a result for switch mode you see only eth0 and for dual port mode eth0/eth1.

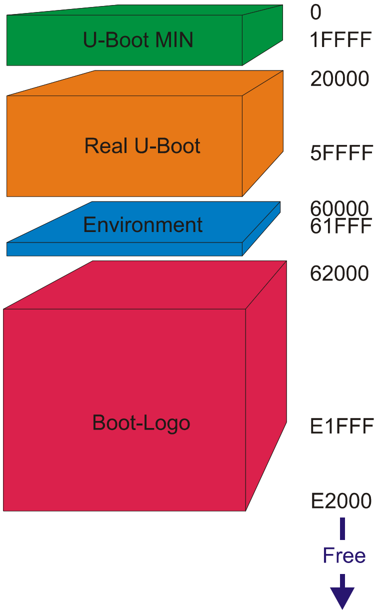
There are no restrictions regarding the connection of other computer boards for example an x86 board with the Windows® operating system.

8 Kernel Boot from SPI™ Flash

In some cases it is necessary to program the Linux® kernel into the SPI™ flash. U-Boot cannot boot the kernel from the S-ATA® or USB interface (for S-ATA® example see next chapter). The most obvious option insists in the use of the U-Boot flash command 'sf'.

```
# sf probe 0:0
    4096 KiB SST25VF032B at 0:0 is now current device
# mw.b 81000000 ff 320000
# loadb 81000000
    Ready for binary (kermit) download to 0x81000000 at 115200 bps...
    Total Size      = ... = ... Bytes
    Start Addr      = 0x81000000
# sf write 81000000 e2000 31e000
```

The following picture shows the SPI™ flash structure:



Note: The SPI™ flash boot requires an additional modification in the U-Boot environment.

```
loadaddr = 0x81000000
```

Another option represents the programming of the SPI™ flash within the Linux® operating system (e.g. boot from microSD™ card). First check the partition layout with

```
# cat /proc/partitions
  major minor  #blocks   name
   31     0     128       mtdblock0
   31     1     256       mtdblock1
   31     2     8         mtdblock2
   31     3     512       mtdblock3
   31     4     3192      mtdblock4
   179    0     7761920   mmcblk0
   179    1     7757824   mmcblk0p1
          .........
```

Thereafter type the following line

```
# sudo dd if=/boot/uImage of=/dev/mtdblock4
  4626+1 records in
  4626+1 records out
  … bytes (… MB) copied, … s, 46.0 kB/s
```

9 Root File System Boot from S-ATA® Drive

The S-ATA® part (mSATA® or standard S-ATA® connector) offers the fastest board interface. This is the only way of achieving transfer rates about 150 MB/s, normally only with SSD drives. The kernel does not automatically mount the drive therefore you have to modify the configuration file etc/fstab (per default this file is empty). The example uses a freely chosen drive identifier (sda1) and target directory.

```
# UNCONFIGURED FSTAB FOR BASE SYSTEM
/dev/sda1  /media/sata  ext4  defaults  0  1
```

After update you should create the new directory /media/sata. But this method has a serious disadvantage: if you remove the drive the kernel stops with an error message.
Furthermore the U-Boot environment needs an additional modification:

```
spiboot =setenv root_bootargs root=/dev/sda1 rw rootwait;run setbootarg;sf probe 0 && sf read ${loadaddr}
                                                                          0xE2000 0x31E000 && bootm ${loadaddr}
```

10 Linux® Programming Examples (Debian™)

10.1 TI® AM3874 Register Access

With the simple known program devmem2 you can read and write all registers of TI®'s AM3874 SoC. The following sourcecode for devmem2 contains some differences to the original program (compiled on Ubuntu™ distribution).

```
/*
* devmem2.c: Simple program to read/write from/to any location in memory.
* Copyright (C) 2000, Jan-Derk Bakker (J.D.Bakker@its.tudelft.nl)

* This software has been developed for the LART computing board
* (http://www.lart.tudelft.nl/). The development has been sponsored by
* the Mobile MultiMedia Communications (http://www.mmc.tudelft.nl/)
* and Ubiquitous Communications (http://www.ubicom.tudelft.nl/)
* projects.

* The author can be reached at:
* Jan-Derk Bakker
* Information and Communication Theory Group
* Faculty of Information Technology and Systems
* Delft University of Technology
* P.O. Box 5031
* 2600 GA Delft
* The Netherlands

* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License, or
* (at your option) any later version.

* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.

* You should have received a copy of the GNU General Public License
* along with this program; if not, write to the Free Software
* Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <fcntl.h>
#include <sys/mman.h>

#define FATAL do { fprintf(stderr, "Error at line %d, file %s (%d) [%s]\n", \
                        __LINE__, __FILE__, errno, strerror(errno)); exit(1); } while(0)

#define MAP_SIZE 4096UL
#define MAP_MASK (MAP_SIZE - 1)
```

```
int main (int argc, char **argv)
{
    int fd;
    void *map_base, *virt_addr;
    unsigned long read_result, writeval;
    off_t target;
    int access_type = 'd';

    if (argc < 2)
    {
      fprintf (stderr,  "\nUsage:\t%s {address} [type [data]]\n"
                        "\taddress : memory address to act upon\n"
                        "\ttype : access operation type : [b]yte, [w]ord, [d]word\n"
                        "\tdata : data to be written\n\n", argv[0]);
      exit (1);
    }

    target = strtoul (argv[1], NULL, 16);
    if (argc > 2)
      access_type = tolower (argv[2][0]);

    if ((fd = open ("/dev/mem", O_RDWR | O_SYNC)) == -1) FATAL;
    map_base = mmap (0, MAP_SIZE, PROT_READ | PROT_WRITE, MAP_SHARED, fd, target & ~MAP_MASK);
    if (map_base == (void *) -1) FATAL;
    virt_addr = map_base + (target & MAP_MASK);

    switch (access_type)
    {
      case 'b': read_result = *((unsigned char *) virt_addr); break;
      case 'w': read_result = *((unsigned short *) virt_addr); break;
      case 'd': read_result = *((unsigned long *) virt_addr); break;
      default : fprintf (stderr, "Illegal data type '%c'.\n", access_type); exit (2);
    }
    printf ("Value at address 0x%lX: 0x%lX\n", target, read_result);
    fflush (stdout);

    if (argc > 3)
    {
      writeval = strtoul (argv[3], NULL, 16);
      switch (access_type)
      {
        case 'b': *((unsigned char *) virt_addr) = writeval;
                read_result = *((unsigned char *) virt_addr);
                break;
        case 'w': *((unsigned short *) virt_addr) = writeval;
                read_result = *((unsigned short *) virt_addr);
                break;
        case 'd': *((unsigned long *) virt_addr) = writeval;
                read_result = *((unsigned long *) virt_addr);
                break;
      }
      printf ("Written: 0x%lX - Readback: 0x%lX\n", writeval, read_result);
      fflush (stdout);
    }
    if (munmap (map_base, MAP_SIZE) == -1) FATAL;
    close (fd);
    return 0;
}
```

The following lines demonstrate the access to an AM3874 register (all values are hexadecimal):

Read the GPIO port 0 debounce register	`./devmem2  48032150`
Enable debouncing of GP0[10] (write access)	`./devmem2  48032150  d  400`

ATTENTION

You need root rights for use of 'devmem2'.

10.1.1 Register Access Restrictions

The functionality of the GPIO connectors deserves special interest. You can use the respective device driver interface or for more flexibility direct register access (e.g. with 'devmem2'). The table shows two cases where the register access fails.

Function	Short Desciption	Memory Address	Direct Access
I2C2	I^2C^{TM} - 3rd controller	0x4819C000	**YES**
SPI2	SPI^{TM} - 3rd controller	0x481A2000	**NO**
SPI3	SPI^{TM} - 4th controller	0x481A4000	**NO**
UART2	UART - 3rd controller	0x48024000	**YES**
UART4	UART - 5th controller	0x481A8000	**YES**
TIM4	Timer 4	0x48044000	**YES**
TIM5	Timer 5	0x48046000	**YES**
TIM6	Timer 6	0x48048000	**YES**
TIM7	Timer 7	0x4804A000	**YES**

The problem with the SPI^{TM} interface resides in the fact that outside of the device driver the clock domain is disabled. You can solve this issue with the following register programming (needed root rights):

Enable clock domain in CM_ALWON_SPI_CLKCTRL	`./devmem2  48181590  d  30002`

Now all SPI^{TM} registers are fully accessible.

10.2 SPI™ Interface Examples

10.2.1 SPI™ Loopback Test Utility

A simple way to check the SPI™ interface consists in shorting of the data lines (MISO, MOSI). After that you can use the following program with root rights (compiled on Ubuntu™ distribution):

```
/*
* SPI testing utility (using spidev driver)

* Copyright (c) 2007  MontaVista Software, Inc.
* Copyright (c) 2007  Anton Vorontsov

* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License.
*/

#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
#include <getopt.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/spi/spidev.h>

#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))

static const char *device = "/dev/spidev3.0";          /* Connector J2105 pin 5 to 8 */
/******  Do not forget to enable SPI3 (= spidev4.0) in U-Boot Setup  ******/
/*static const char *device = "/dev/spidev4.0";*/      /* Connector J2105 pin 23 to 26 */
static uint8_t mode   = 0;
static uint8_t bits   = 8;
static uint32_t speed = 1000000;
static uint16_t delay;

static void pabort (const char *s)
{
    perror (s);
    abort ();
}


static void transfer (int fd)
{
    int ret;
    uint8_t tx[] = {
      0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
      0x40, 0x00, 0x00, 0x00, 0x00, 0x95,
      0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
      0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
      0xFF, 0xFF, 0xFF, 0xFF, 0xDE, 0xAD,
      0xBE, 0xEF, 0xBA, 0xAD, 0xF0, 0x0D };
    uint8_t rx[ARRAY_SIZE(tx)] = {0, };
    struct spi_ioc_transfer tr = {
      .tx_buf = (unsigned long) tx,
      .rx_buf = (unsigned long) rx,
      .len = ARRAY_SIZE(tx),
      .delay_usecs = delay,
      .speed_hz = 0,
      .bits_per_word = 0 };

    ret = ioctl (fd, SPI_IOC_MESSAGE (1), &tr);
    if (ret == 1)
      pabort ("Cannot send SPI message");

    for (ret = 0; ret < ARRAY_SIZE(tx); ret++)
    {
      if (! (ret % 6))
        puts ("");
      printf ("%.2X ", rx[ret]);
    }
    puts ("");
}


void print_usage (const char *prog)
{
    printf ("Usage: %s [-DsbdHOC]\n", prog);
    puts ("  -D --device   device to use (default /dev/spidev1.1)\n"
                        "  -s --speed      max speed (Hz)\n"
                        "  -d --delay      delay (usec)\n"
                        "  -b --bpw        bits per word \n"
                        "  -H --cpha       clock phase\n"
                        "  -O --cpol       clock polarity\n"
                        "  -C --cs-high    chip select active high\n");
    exit (1);
}


void parse_opts (int argc, char *argv[])
{
    while (1)
    {
      static const struct option lopts[] = {
        { "device",    1, 0, 'D' },
        { "speed",     1, 0, 's' },
        { "delay",     1, 0, 'd' },
        { "bpw",       1, 0, 'b' },
        { "cpha",      0, 0, 'H' },
        { "cpol",      0, 0, 'O' },
        { "cs-high",   0, 0, 'C' },
        { NULL,        0, 0, 0 }};
      int c;

      c = getopt_long (argc, argv, "D:s:d:b:HOC", lopts, NULL);
      if (c == -1) break;
```

```
        switch (c)
        {
           case 'D':  device = optarg; break;
           case 's':   speed = atoi (optarg); break;
           case 'd':   delay = atoi (optarg); break;
           case 'b':   bits = atoi (optarg); break;
           case 'H':   mode |= SPI_CPHA; break;
           case 'O':   mode |= SPI_CPOL; break;
           case 'C':   mode |= SPI_CS_HIGH; break;
           default:    print_usage (argv[0]); break;
        }
     }
}


int main (int argc, char *argv[])
{
     int ret = 0;
     int fd;

     parse_opts (argc, argv);

     fd = open (device, O_RDWR);
     if (fd < 0)
        pabort ("Cannot open device");

     /* SPI mode */
     ret = ioctl (fd, SPI_IOC_WR_MODE, &mode);
     if (ret == -1)
        pabort ("Cannot set SPI mode");

     ret = ioctl (fd, SPI_IOC_RD_MODE, &mode);
     if (ret == -1)
        pabort ("Cannot get SPI mode");

     /* Bits per word */
     ret = ioctl (fd, SPI_IOC_WR_BITS_PER_WORD, &bits);
     if (ret == -1)
        pabort ("Cannot set bits per word");

     ret = ioctl (fd, SPI_IOC_RD_BITS_PER_WORD, &bits);
     if (ret == -1)
        pabort ("Cannot get bits per word");

     /* Max speed in Hz */
     ret = ioctl (fd, SPI_IOC_WR_MAX_SPEED_HZ, &speed);
     if (ret == -1)
        pabort ("Cannot set max speed in Hz");

     ret = ioctl (fd, SPI_IOC_RD_MAX_SPEED_HZ, &speed);
     if (ret == -1)
        pabort ("Cannot get max speed in Hz");

     printf ("SPI mode: %d\n", mode);
     printf ("Bits per word: %d\n", bits);
     printf ("Max speed: %d Hz (%d kHz)\n", speed, speed/1000);
```

```
    transfer (fd);

    close (fd);
    return ret;
}
```

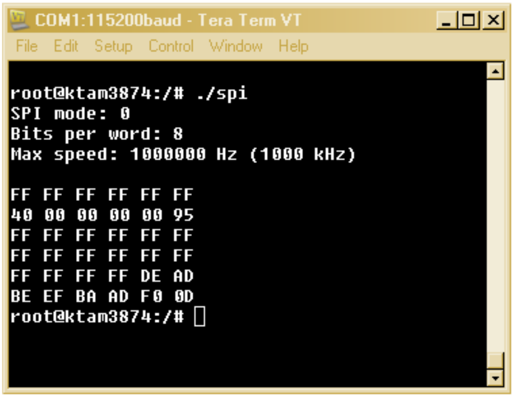
You should consider following parameter limits:

Parameter	MIN	MAX	Units
Databits (Bits per Word)	4	32	Bits
Transmission Speed	2000	48000000	Hz
Transmission Mode	Only full duplex possible (driver restriction)		
Bit Transfer Mode	Only MSB first mode possible		

Note: The test array of the original sourcecode contains 38 bytes but this number can cause a crash if you use 32 data bits (number should be 36 bytes).
Further information is available on https://www.kernel.org/doc/Documentation/spi/spi-summary.
Examples:

Using default values	./spi
Set speed to 10 MHz	./spi -s 10000000
Set mode to 3	./spi -H -O

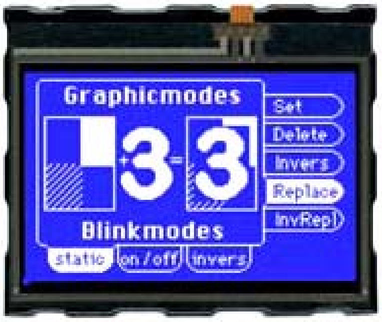
As result you should see this screen output:


COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

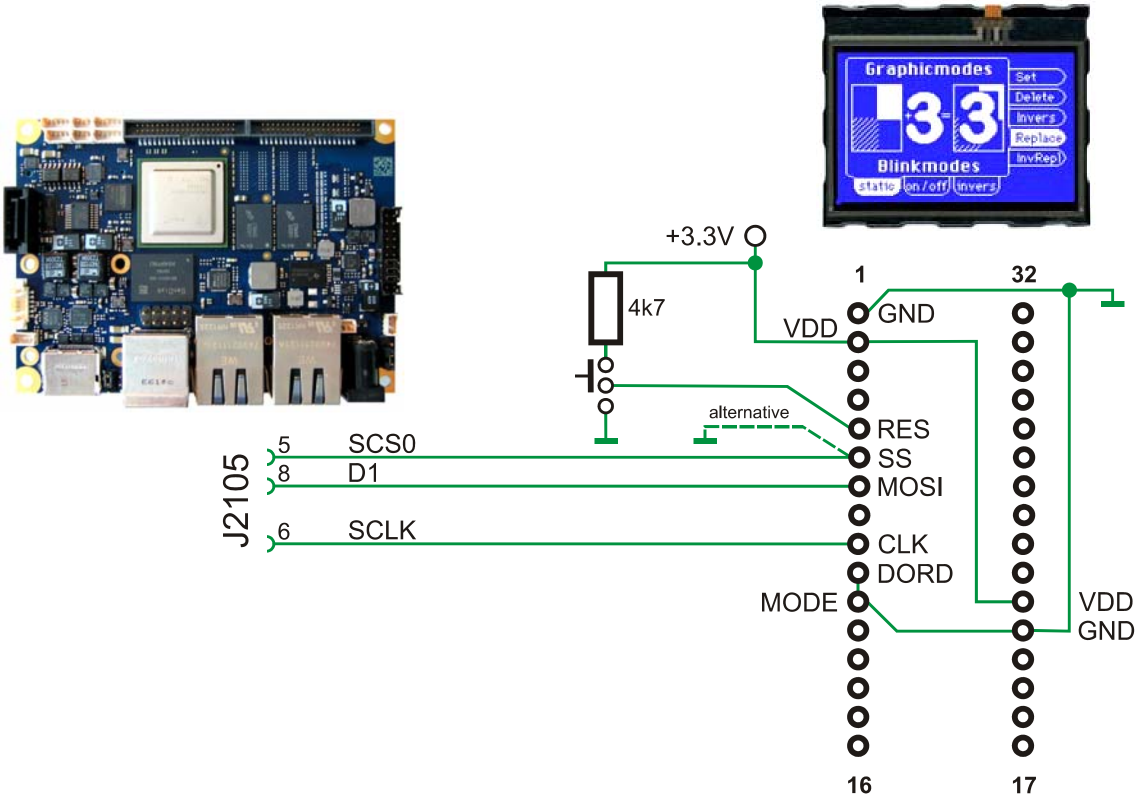
```
root@ktam3874:/# ./spi
SPI mode: 0
Bits per word: 8
Max speed: 1000000 Hz (1000 kHz)

FF FF FF FF FF FF
40 00 00 00 00 95
FF FF FF FF FF FF
FF FF FF FF FF FF
FF FF FF FF DE AD
BE EF BA AD F0 0D
root@ktam3874:/#
```

10.2.2 SPI^{TM} Display Interface

The SPI^{TM} interface is well suited to connect uncomplicated intelligent small graphic modules for example **Electronic Assembly eDIP128-6.**



The schematic below shows the connection to I/O connector J2105 pin 5 to 8 (SPI2). Please notice that this draft only serves for demonstration purposes but not for real applications.



In general the following pinout applies for all SPI^{TM} interfaces:

D0 ☛ **MISO** (Master In Slave Out = Receive)

D1 ☛ **MOSI** (Master Out Slave In = Transmit)

Note: Please perform a RESET before you execute the test program.

Now the corresponding test program (needs root rights):

```
/* SPI test program for
 *    Electronic Assembly graphic module eDIP128-6

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
#include <getopt.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/spi/spidev.h>

#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))
#define SPI_OPEN_ERROR            0x10
#define SPI_MODE_ERROR            0x11
#define SPI_BITS_ERROR            0x12
#define SPI_SPEED_ERROR           0x13

static const char *device = "/dev/spidev3.0";
/******  Do not forget to enable SPI3 (= spidev4.0) in U-Boot Setup  ******/
/*static const char *device = "/dev/spidev4.0";*/
static uint8_t mode   = 3;
static uint8_t bits     = 8;
static uint32_t speed = 100000;
static uint16_t delay  = 10;
static int fd;

static int transmit_spi_test_string (void)
{
    uint8_t tx[] = {
        0x11,                                   /* DC1 */
        0x05,                                   /* Data len */
        0x48, 0x65, 0x6C, 0x6C, 0x6F,           /* Data = 'Hello' */
        0x0A                                    /* Checksum */
        };
    struct spi_ioc_transfer xfer = {
        .tx_buf = (unsigned long) tx,
        .rx_buf = (unsigned long) NULL,
        .len = ARRAY_SIZE(tx),
        .delay_usecs = delay,
        .speed_hz = 0,
        .bits_per_word = 0 };

    if (ioctl (fd, SPI_IOC_MESSAGE (1), &xfer) == 1)
      return -1;
    return 0;
}
```

```
int init_spi_interface (void)
{
    fd = open (device, O_RDWR);
    if (fd < 0)
      return SPI_OPEN_ERROR;

    if (ioctl (fd, SPI_IOC_WR_MODE, &mode) == -1)
      return SPI_MODE_ERROR;

    if (ioctl (fd, SPI_IOC_WR_BITS_PER_WORD, &bits) == -1)
      return SPI_BITS_ERROR;

    if (ioctl (fd, SPI_IOC_WR_MAX_SPEED_HZ, &speed) == -1)
      return SPI_SPEED_ERROR;

    return 0;
}


int main (void)
{
    int ret;

    printf ("\nSPI Test for eDIP128-6\n");


    if ((ret = init_spi_interface ()))
    {
      close (fd);
      printf ("SPI init error: 0x%02X\n\n", ret);
      return -1;
    }

    if (transmit_spi_test_string ())
    {
      close (fd);
      printf ("SPI transmission fails!\n\n");
      return -1;
    }

    close (fd);
    printf ("SPI transmission successfully finished.\n\n");
    return 0;
}
```

10.3 I^{2}C^{TM} Interface Examples

10.3.1 I^{2}C^{TM} Detection Utility

The following sourcecode represents an optimized lean version of the standard I^{2}C^{TM} detection program without unnecessary overhead (compiled on Ubuntu^{TM} distribution, needs root rights).

```
/*
i2cdetect.c - a user-space program to scan for I2C devices
Copyright (C) 1999-2004  Frodo Looijaard <frodol@dds.nl>, and
                         Mark D. Studebaker <mdsxyz123@yahoo.com>
Copyright (C) 2004-2012  Jean Delvare <khali@linux-fr.org>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
MA 02110-1301 USA.
*/

#include <errno.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <linux/types.h>

#define MODE_AUTO                              0
#define MODE_QUICK                             1
#define MODE_READ                              2
#define MODE_FUNC                              3

#define I2C_FUNC_I2C                           0x00000001
#define I2C_FUNC_10BIT_ADDR                    0x00000002
#define I2C_FUNC_PROTOCOL_MANGLING             0x00000004
#define I2C_FUNC_SMBUS_PEC                     0x00000008
#define I2C_FUNC_SMBUS_BLOCK_PROC_CALL         0x00008000
#define I2C_FUNC_SMBUS_QUICK                   0x00010000
#define I2C_FUNC_SMBUS_READ_BYTE               0x00020000
#define I2C_FUNC_SMBUS_WRITE_BYTE              0x00040000
#define I2C_FUNC_SMBUS_READ_BYTE_DATA          0x00080000
#define I2C_FUNC_SMBUS_WRITE_BYTE_DATA         0x00100000
#define I2C_FUNC_SMBUS_READ_WORD_DATA          0x00200000
#define I2C_FUNC_SMBUS_WRITE_WORD_DATA         0x00400000
#define I2C_FUNC_SMBUS_PROC_CALL               0x00800000
```

```
#define I2C_FUNC_SMBUS_READ_BLOCK_DATA          0x01000000
#define I2C_FUNC_SMBUS_WRITE_BLOCK_DATA         0x02000000
#define I2C_FUNC_SMBUS_READ_I2C_BLOCK           0x04000000
#define I2C_FUNC_SMBUS_WRITE_I2C_BLOCK          0x08000000

#define I2C_SMBUS_READ                          1
#define I2C_SMBUS_BYTE                          1
#define I2C_SLAVE                               0x0703
#define I2C_FUNCS                               0x0705
#define I2C_SMBUS                               0x0720
#define I2C_SMBUS_BLOCK_MAX                     32

struct func
{
    long value;
    const char* name;
};

union i2c_smbus_data
{
    __u8 byte;
    __u16 word;
    __u8 block[I2C_SMBUS_BLOCK_MAX + 2];
};

struct i2c_smbus_ioctl_data
{
    char read_write;
    __u8 command;
    int size;
    union i2c_smbus_data *data;
};

static void help (void)
{
    fprintf (stderr, "Usage: i2cdetect [I2CBUS] [FIRST LAST]\n"
                    "           i2cdetect -F [I2CBUS]\n"
                    "  I2CBUS is an integer or an I2C bus name\n"
                    "  If provided, FIRST and LAST limit the probing range.\n");
}


static inline __s32 i2c_smbus_access (int file, char read_write, __u8 command,
                                      int size, union i2c_smbus_data *data)
{
    struct i2c_smbus_ioctl_data args;

    args.read_write = read_write;
    args.command = command;
    args.size = size;
    args.data = data;
    return ioctl (file, I2C_SMBUS, &args);
}
```

```
static inline __s32 i2c_smbus_read_byte (int file)
{
    union i2c_smbus_data data;

    if (i2c_smbus_access (file, I2C_SMBUS_READ, 0, I2C_SMBUS_BYTE, &data))
      return -1;
    else
      return 0x0FF & data.byte;
}


static int scan_i2c_bus (int file, int mode, int first, int last)
{
    int i, j;

    printf ("     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f\n");

    for (i = 0; i < 128; i += 16)
    {
      printf ("%02x: ", i);
      for (j = 0; j < 16; j++)
      {
        fflush (stdout);

        /* Skip unwanted addresses */
        if (i+j < first || i+j > last)
        {
          printf ("   "); continue;
        }

        /* Set slave address */
        if (ioctl (file, I2C_SLAVE, i+j) < 0)
        {
          if (errno == EBUSY)
          {
            printf ("UU "); continue;
          }
          else
          {
            /* ERROR: Could not set address */
            printf ("** "); continue;
          }
        }

        /* Probe this address */
        /* This is known to lock SMBus on various write-only chips (mainly clock chips) */
        if (i2c_smbus_read_byte (file) < 0)
          printf ("-- ");
        else
          printf ("%02x ", i+j);
      }
      printf ("\n");
    }
    return 0;
}
```

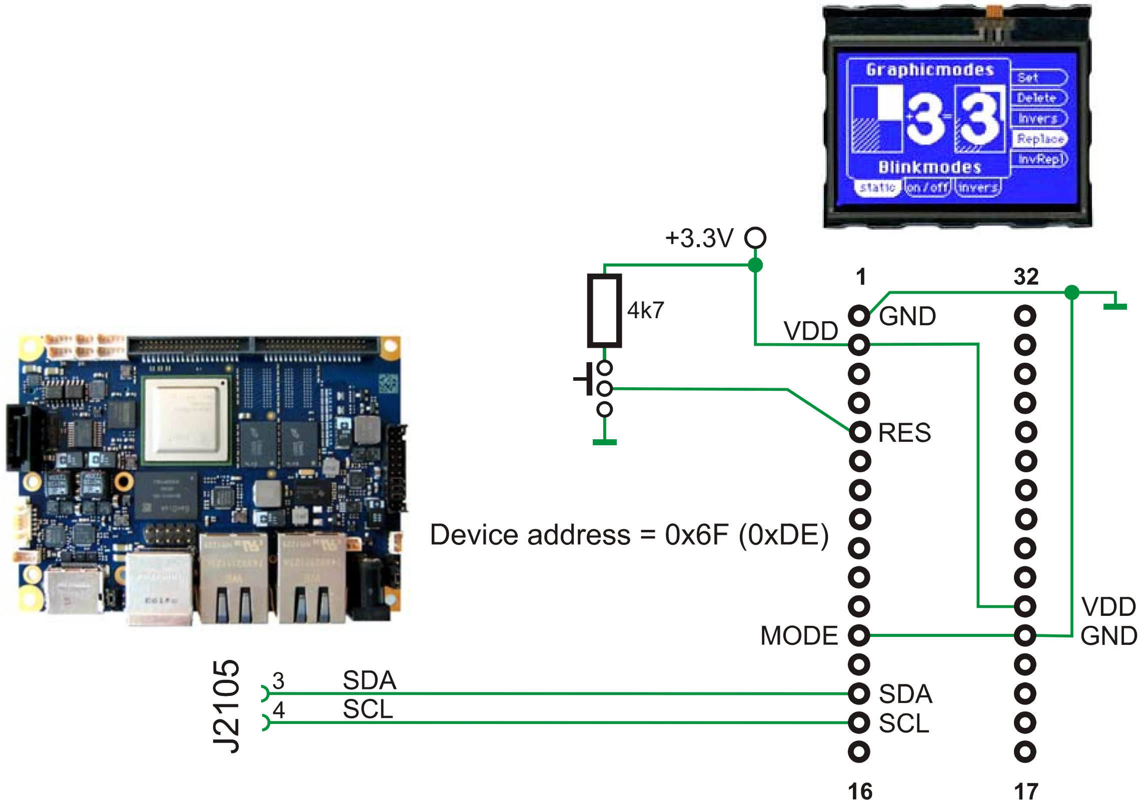
```
static const struct func all_func[] = {
     { .value = I2C_FUNC_I2C,
       .name = "I2C" },
     { .value = I2C_FUNC_SMBUS_QUICK,
       .name = "SMBus Quick Command" },
     { .value = I2C_FUNC_SMBUS_WRITE_BYTE,
       .name = "SMBus Send Byte" },
     { .value = I2C_FUNC_SMBUS_READ_BYTE,
       .name = "SMBus Receive Byte" },
     { .value = I2C_FUNC_SMBUS_WRITE_BYTE_DATA,
       .name = "SMBus Write Byte" },
     { .value = I2C_FUNC_SMBUS_READ_BYTE_DATA,
       .name = "SMBus Read Byte" },
     { .value = I2C_FUNC_SMBUS_WRITE_WORD_DATA,
       .name = "SMBus Write Word" },
     { .value = I2C_FUNC_SMBUS_READ_WORD_DATA,
       .name = "SMBus Read Word" },
     { .value = I2C_FUNC_SMBUS_PROC_CALL,
       .name = "SMBus Process Call" },
     { .value = I2C_FUNC_SMBUS_WRITE_BLOCK_DATA,
       .name = "SMBus Block Write" },
     { .value = I2C_FUNC_SMBUS_READ_BLOCK_DATA,
       .name = "SMBus Block Read" },
     { .value = I2C_FUNC_SMBUS_BLOCK_PROC_CALL,
       .name = "SMBus Block Process Call" },
     { .value = I2C_FUNC_SMBUS_PEC,
       .name = "SMBus PEC" },
     { .value = I2C_FUNC_SMBUS_WRITE_I2C_BLOCK,
       .name = "I2C Block Write" },
     { .value = I2C_FUNC_SMBUS_READ_I2C_BLOCK,
       .name = "I2C Block Read" },
     { .value = 0, .name = "" }
};


static void print_functionality (unsigned long funcs)
{
     int i;

     for (i = 0; all_func[i].value; i++)
     {
        printf ("%-32s %s\n", all_func[i].name, (funcs & all_func[i].value) ? "yes" : "no");
     }
}
```

```
int open_i2c_dev (const int i2cbus, char *filename, const int quiet)
{
    int file;

    sprintf (filename, "/dev/i2c-%d", i2cbus);
    file = open (filename, O_RDWR);

    if (file < 0 && !quiet)
    {
      if (errno == ENOENT)
        fprintf (stderr, "Error: Could not open file `/dev/i2c-%d': %s\n", i2cbus, strerror (ENOENT));
      else
      {
        fprintf (stderr, "Error: Could not open file `%s': %s\n", filename, strerror (errno));
        if (errno == EACCES)
          fprintf (stderr, "Run as root?\n");
      }
    }
    return file;
}


int main (int argc, char *argv[])
{
    int i2cbus = 3;                 /* Default I2C bus on J2105 */
    int file, res;
    char filename[20];
    unsigned long funcs;
    int mode = MODE_READ;           /* Quick mode not supported */
    int first = 0x03, last = 0x77;
    int flags = 0;

    /* Handle (optional) flags first */
    while (1+flags < argc && argv[1+flags][0] == '-')
    {
      switch (argv[1+flags][1])
      {
        case 'F': mode = MODE_FUNC; break;
        default : fprintf (stderr, "Error: Unsupported option \"%s\"!\n", argv[1+flags]);
                  help ();
                  return -1;
      }
      flags++;
    }

    if (argc >= flags + 2)
    {
      i2cbus = atoi (argv[flags+1]);
      if ((i2cbus < 1) || (i2cbus > 4))
      {
        fprintf (stderr, "Error: No valid number for i2c-bus!\n");
        return -1;
      }
    }
    /* Read address range if present */
    if (argc == flags + 4 && mode != MODE_FUNC)
    {
      int tmp;

      tmp = (int) strtoul (argv[flags+2], NULL, 16);
      if ((tmp < first) || (tmp > last))
      {
        fprintf (stderr, "Error: FIRST argument out of range (0x%02x-0x%02x)!\n", first, last);
        return -1;
      }
      first = tmp;

      tmp = (int) strtoul (argv[flags+3], NULL, 16);
      if ((tmp < first) || (tmp > last))
      {
        fprintf (stderr, "Error: FIRST argument out of range (0x%02x-0x%02x)!\n", first, last);
        return -1;
      }
      last = tmp;
    }

    file = open_i2c_dev (i2cbus, filename, 0);
    if (file < 0)
      return -1;

    /* Special case, we only list the implemented functionalities */
    if (mode == MODE_FUNC)
    {
      if (ioctl (file, I2C_FUNCS, &funcs) < 0)
      {
        fprintf (stderr, "Error: Could not get the adapter functionality matrix: %s\n", strerror (errno));
        close (file);
        return -1;
      }
      close (file);
      printf ("Functionalities implemented by %s:\n", filename);
      print_functionality (funcs);
      return 0;
    }

    res = scan_i2c_bus (file, mode, first, last);

    close (file);
    exit (res?1:0);
}
```

Further information about the I^{2}C^{TM} bus is available on https://www.kernel.org/doc/Documentation/i2c/dev-interface.

10.3.2 I^{2}C™ Display Interface

The graphic module from Electronic Assembly can also be driven via the I^{2}C™ bus. The schematic is a bit different. Please notice that this draft only serves for demonstration purposes but not for real applications.



Note: Please perform a RESET before you execute the test program.
The sourcecode is based on the SPI™ example (needs root rights):

```
/* I2C test program for
 *    Electronic Assembly graphic module eDIP128-6

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <errno.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <linux/types.h>
```

```
#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))
#define I2C_SMBUS_WRITE          0
#define I2C_SMBUS_BYTE           1
#define I2C_SLAVE                0x0703
#define I2C_SMBUS                0x0720
#define I2C_SMBUS_BLOCK_MAX      32
#define EDIP128_ADDR             0x6F

static const char *device = "/dev/i2c-3";
static int fd;

union i2c_smbus_data
{
    __u8 byte;
    __u16 word;
    __u8 block[I2C_SMBUS_BLOCK_MAX + 2];
};

struct i2c_smbus_ioctl_data
{
    char read_write;
    __u8 command;
    int size;
    union i2c_smbus_data *data;
};

static inline __s32 i2c_smbus_access (int file, char read_write, __u8 command,
                                     int size, union i2c_smbus_data *data)
{
    struct i2c_smbus_ioctl_data args;

    args.read_write = read_write;
    args.command = command;
    args.size = size;
    args.data = data;
    return ioctl (file, I2C_SMBUS, &args);
}


static inline __s32 i2c_smbus_write_byte (int file, __u8 value)
{
    return i2c_smbus_access (file, I2C_SMBUS_WRITE, value, I2C_SMBUS_BYTE, NULL);
}


static int transmit_i2c_test_string (void)
{
    int i;
    __u8 tx[] = {
            0x11,                               /* DC1 */
            0x05,                               /* Data len */
            0x48, 0x65, 0x6C, 0x6C, 0x6F,       /* Data = 'Hello' */
            0x0A                                /* Checksum */
            };
```

```
     for (i = 0; i < ARRAY_SIZE(tx); i++)
     {
        if (i2c_smbus_write_byte (fd, tx[i]))
           return -1;
     }
     return 0;
}


int init_i2c_interface (int dev_addr)
{
     fd = open (device, O_RDWR);
     if (fd < 0)
        return -1;

     /* Set slave address */
     return ioctl (fd, I2C_SLAVE, dev_addr);
}


int main (void)
{
     int ret;

     printf ("\nI2C Test for eDIP128-6\n");

     if (init_i2c_interface (EDIP128_ADDR))
     {
        close (fd);
        printf ("I2C init error!\n\n");
        return -1;
     }

     if (transmit_i2c_test_string ())
     {
        close (fd);
        printf ("I2C transmission fails!\n\n");
        return -1;
     }

     close (fd);
     printf ("I2C transmission successfully finished.\n\n");
     return 0;
}
```

10.4 UART Interface Examples

10.4.1 UART Loopback Test Utility

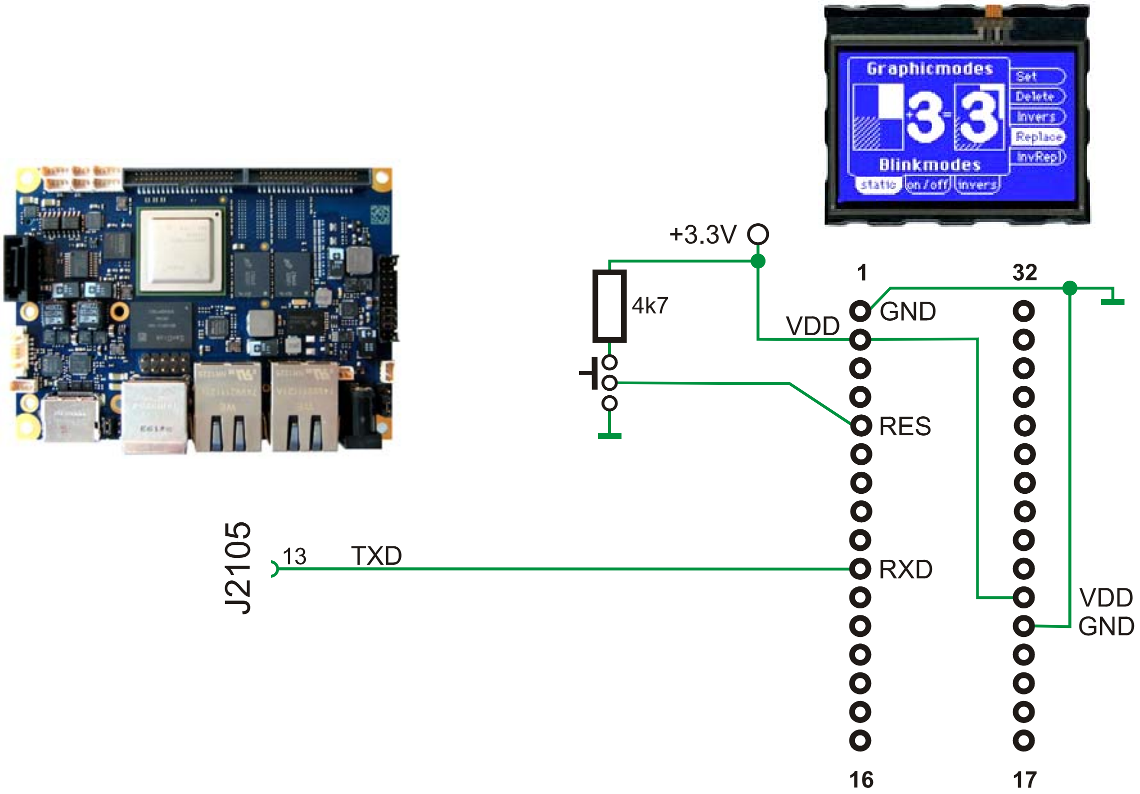
A simple way to check the UART interface consists in shorting of the data lines (TXD, RXD). After that you can use the following program with root rights (compiled on Ubuntu™ distribution):

```
/* UART loopback test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License. */

#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <fcntl.h>
#include <errno.h>
#include <sys/types.h>

#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))

/******  Do not forget to enable UART2 and/or UART4 in U-Boot Setup  ******/
static const char *device = "/dev/ttyO4";                /* Connector J2105 pin 11 to 14 */
/*static const char *device = "/dev/ttyO2";*/            /* Connector J2105 pin 29 to 32 */
static int fd;


int init_uart_interface (int baudrate)
{
   struct termios new_values;

   fd = open (device, O_RDWR);
   if (fd < 0 )
     return -1;

   if (tcflush (fd, TCIOFLUSH))
     return -1;

   switch (baudrate)
   {
     case 300          : baudrate = B300; break;
     case 600          : baudrate = B600; break;
     case 1200         : baudrate = B1200; break;
     case 2400         : baudrate = B2400; break;
     case 4800         : baudrate = B4800; break;
     case 9600         : baudrate = B9600; break;
     case 19200        : baudrate = B19200; break;
     case 38400        : baudrate = B38400; break;
     case 57600        : baudrate = B57600; break;
     case 115200       : baudrate = B115200; break;
     default           : return -1;
   }
```

```
    memset (&new_values, 0, sizeof (new_values));
    new_values.c_cflag = CS8 | CLOCAL | CREAD;
    new_values.c_iflag = IGNBRK;
    new_values.c_cc[VMIN] = 1;

    if (cfsetispeed (&new_values, baudrate))
      return -1;
    if (cfsetospeed (&new_values, baudrate))
      return -1;
    return tcsetattr (fd, TCSANOW, &new_values);
}


int uart_loopback (void)
{
    char out_str[] = "the quick brown fox jumps over the lazy dog";
    char in_str [64] = "\0";

    if (write (fd, out_str, sizeof (out_str)) !=  sizeof (out_str))
      return -1;

    if (tcdrain (fd))
      return -1;

    read (fd, in_str, sizeof (out_str));

    return memcmp (out_str, in_str, sizeof (out_str));
}


int main (void)
{
    int baud[] = { 9600, 38400, 115200 };
    int i;

    printf ("\nUART loopback test program\n");

    for (i = 0; i < ARRAY_SIZE(baud); i++)
    {
      if (init_uart_interface (baud[i]))
      {
        close (fd);
        printf ("UART init error!\n\n");
        return -1;
      }

      if (uart_loopback ())
      {
        close (fd);
        printf ("UART loopback test fails with baudrate %d!\n\n", baud[i]);
        return -1;
      }

      close (fd);
    }
    printf ("UART loopback test successfully finished.\n\n");
    return 0;
}
```

10.4.2 UART Display Interface

The last possible interface of the graphic module from Electronic Assembly represents the standard UART interface with 115.2 kBaud, 8 data bits, 1 stop bit and no parity. Please notice that this draft only serves for demonstration purposes but not for real applications.



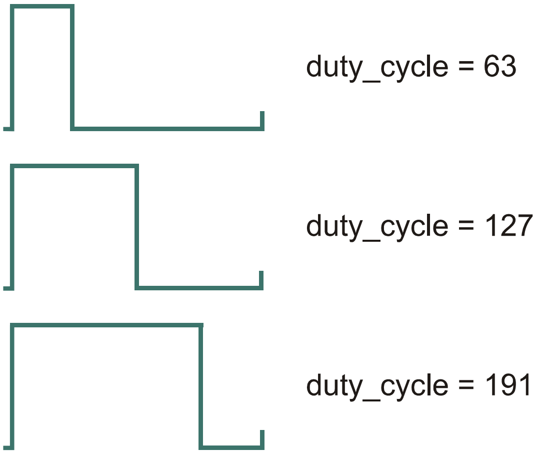
Note: Please perform a RESET before you execute the test program.
Below you will find the already known varied sourcecode (needs root rights).

```
/* UART test program for
 *    Electronic Assembly graphic module eDIP128-6
 *
 * Copyright (c) 2013  Kontron Technology A/S
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <errno.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <termios.h>
#include <linux/types.h>
```

```
#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))

/****** Do not forget to enable UART2 and/or UART4 in U-Boot Setup ******/
static const char *device = "/dev/ttyO4";               /* Connector J2105 pin 11 to 14 */
/*static const char *device = "/dev/ttyO2";*/           /* Connector J2105 pin 29 to 32 */
static int fd;

static int transmit_uart_test_string (void)
{
    __u8 tx[] = {
            0x11,                                       /* DC1 */
            0x05,                                       /* Data len */
            0x48, 0x65, 0x6C, 0x6C, 0x6F,               /* Data = 'Hello' */
            0x0A                                        /* Checksum */
            };

    if (write (fd, tx, ARRAY_SIZE(tx)) != ARRAY_SIZE(tx))
      return -1;
    return 0;
}


int init_uart_interface (void)
{
    struct termios new_values;

    fd = open (device, O_RDWR);
    if (fd < 0)
      return -1;

    memset (&new_values, 0, sizeof (new_values));
    new_values.c_cflag = CS8 | CLOCAL;

    if (cfsetospeed (&new_values, B115200))
      return -1;

    return tcsetattr (fd, TCSANOW, &new_values);
}


int main (void)
{
    printf ("\nUART Test for eDIP128-6\n");

    if (init_uart_interface ())
    {
      close (fd);
      printf ("UART init error!\n\n");
      return -1;
    }

    if (transmit_uart_test_string ())
    {
      close (fd);
      printf ("UART transmission fails!\n\n");
      return -1;
    }
    close (fd);
    printf ("UART transmission successfully finished.\n\n");
    return 0;
}
```

10.5 Timer Example

The timer module offers a variety of possibilities, for example realization of PWM outputs. The demonstration program only shows an easy (simple) implementation to start and stop the PWM interface and cannot be called twice without a reboot process. The frequency range is well suited for motor control applications (5 to 30 kHz).
The picture demonstrates the use of several different duty cycle values.



There is a problem with the maximum value of 255. The equation for val in routine start_pwm gives the right result of 0xFFFFFFFF but the PWM output level approaches zero. You can avoid this issue with following additional lines

```
val = (FREQ_VAL * duty_cycle) / DUTY_MAX;
if (duty_cycle == 255)
  write_timer_reg (TMAR, 0xFFFFFFFE);
else
  write_timer_reg (TMAR, RELOAD_VAL + val);
```

or alternative with

```
val = ((FREQ_VAL * duty_cycle) / DUTY_MAX) - 1;
write_timer_reg (TMAR, RELOAD_VAL + val);
```

but one fact remains: both limits (0 and 255) are generating short spikes. For a zero value it is possible to stop the timer. The program execution needs root rights (compiled on Ubuntu$^{\text{TM}}$ distribution).

```
/* Timer test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */
```

```
#include <errno.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <sys/mman.h>

/******  Do not forget to enable TIMER4 to TIMER7 in U-Boot Setup  ******/
#define TIMER4_BASE_ADDR        0x48044000
#define TIMER5_BASE_ADDR        0x48046000
#define TIMER6_BASE_ADDR        0x48048000
#define TIMER7_BASE_ADDR        0x4804A000
#define TIMER_BASE              TIMER4_BASE_ADDR

/* Register definitions */
#define TIOCP_CFG               0x10
#define TCLR                    0x38
#define TCRR                    0x3C
#define TLDR                    0x40
#define TTGR                    0x44
#define TMAR                    0x4C
#define TSICR                   0x54

/* Register bits */
#define TIOCP_SOFTRES           0x01
#define TIOCP_SMARTIDLE         0x08
#define TCLR_ST                 0x01
#define TCLR_AR                 0x02
#define TCLR_CE                 0x40
#define TCLR_SCPWM              0x80
#define TCLR_TRGOVFLMAT         0x800
#define TCLR_TRG                0xC00
#define TCLR_PT                 0x1000

#define TCLR_GPOCFG             0x4000
#define TSICR_SFT               0x02
#define TSICR_POSTED            0x04

#define RESET_TIMEOUT           100000
#define STOP_TIMEOUT            2000000

/* Frequency related definitions */
#define FREQ_VAL                1024          /* VAL = 128 approx. 155 kHz - VAL = 1024 approx. 20 kHz */
#define RELOAD_VAL              0xFFFFFFFF - FREQ_VAL
#define DUTY_MAX                255

#define MAP_SIZE                4096UL
#define MAP_MASK                (MAP_SIZE - 1)

static int fd;
static void *virt_addr;
static void *map_base;
```

```
int open_reg_range (void)
{
    fd = open ("/dev/mem", O_RDWR | O_SYNC);
    if (fd < 0)
      return -1;

    map_base = mmap (0, MAP_SIZE, PROT_READ | PROT_WRITE, MAP_SHARED, fd,
                                                                    TIMER_BASE & ~MAP_MASK);
    if (map_base == (void *) -1)
      return -1;

    virt_addr = map_base + (TIMER_BASE & MAP_MASK);
    return 0;
}


void close_reg_range (void)
{
    munmap (map_base, MAP_SIZE);
    close (fd);
}


inline unsigned long read_timer_reg (unsigned long reg)
{
    return *((unsigned long *) (virt_addr + reg));
}


inline void write_timer_reg (unsigned long reg, unsigned long val)
{
    *((unsigned long *) (virt_addr + reg)) = val;
}


int timer_reset (void)
{
    int i;

    /* Reset the timer */
    write_timer_reg (TSICR, TSICR_SFT | TSICR_POSTED);
    for (i = 0; i < RESET_TIMEOUT; i++)
      if (! (read_timer_reg (TIOCP_CFG) & TIOCP_SOFTRES)) break;
    if (i >= RESET_TIMEOUT)
      return -1;

    /* Set to smart-idle mode */
    write_timer_reg (TIOCP_CFG, read_timer_reg (TIOCP_CFG) | 0x200 | TIOCP_SMARTIDLE);

    /* Match hardware reset default of posted mode */
    write_timer_reg (TSICR, TSICR_POSTED);
    /* Set PWM mode */
    write_timer_reg (TCLR, (read_timer_reg (TCLR) &
                                    ~(TCLR_GPOCFG | TCLR_PT | TCLR_TRG | TCLR_SCPWM)) |
                                    (TCLR_PT | TCLR_TRGOVFLMAT));

    return 0;
}


int start_pwm (unsigned long duty_cycle)
{
    unsigned long val;

    /* Check the duty cycle value */
    if ((duty_cycle < 0) || (duty_cycle > DUTY_MAX))
      return -1;

    /* Set the reload value */
    write_timer_reg (TCLR, read_timer_reg (TCLR) | TCLR_AR);
    write_timer_reg (TLDR, RELOAD_VAL);
    write_timer_reg (TTGR, 0);

    /* Set the match value */
    write_timer_reg (TCLR, read_timer_reg (TCLR) | TCLR_CE);
    val = (FREQ_VAL * duty_cycle) / DUTY_MAX;
    write_timer_reg (TMAR, RELOAD_VAL + val);

    /* Write the counter value */
    write_timer_reg (TCRR, -1);

    /* Start the timer */
    if (! (read_timer_reg (TCLR) & TCLR_ST))
      write_timer_reg (TCLR, read_timer_reg (TCLR) | TCLR_ST);

    return 0;
}


int stop_pwm (void)
{
    int i;

    if (read_timer_reg (TCLR) & TCLR_ST)
    {
      /* Stop the timer */
      write_timer_reg (TCLR, read_timer_reg (TCLR) & ~TCLR_ST);

      /* Check if timer off */
      for (i = 0; i < STOP_TIMEOUT; i++)
        if (! (read_timer_reg (TCLR) & TCLR_ST)) break;
      if (i >= STOP_TIMEOUT)
      return -1;
    }

    return 0;
}
```

```
int main (void)
{
    printf ("\nTIMER test program\n");

    if (open_reg_range ())
    {
      printf ("Cannot open register range !\n\n");
      return -1;
    }

    if (timer_reset ())
    {
      close_reg_range ();
      printf ("TIMER init error !\n\n");
      return -1;
    }

    /* Use only values from 0 to 255 to define the duty cycle, 127 generates a duty cycle of 50% */
    if (start_pwm (127))
    {
      close_reg_range ();
      printf ("Wrong duty cycle value !\n\n");
      return -1;
    }

    close_reg_range ();
    printf ("\n");
    return 0;
}
```

10.6 GPIO Example

An easy way to design an application with GPIO access consists in use of the sysfs interface. The directory /sys/class/gpio contains the following entries:

export		
gpiochip0	TI® AM3874 SoC	32 GPIOs
gpiochip32	TI® AM3874 SoC	32 GPIOs
gpiochip64	TI® AM3874 SoC	32 GPIOs
gpiochip96	TI® AM3874 SoC	32 GPIOs
gpiochip128	PMU	9 GPIOs
unexport		

ATTENTION

Do not use other GPIO pins as are described in the User's Guide chapter 'Digital I/O Interface'.

The kernel occupies some GPIO pins but not all are declared (for example GP1[0] and GP1[1] comprise the first CAN bus interface). The command line below shows incomplete information (debugfs is already mounted):

```
# cat /sys/kernel/debug/gpio
GPIOs 0-31, gpio:

GPIOs 32-63, gpio:
 gpio-38 (mmc_cd    ) in lo irq-198 edge-both

GPIOs 64-95, gpio:

GPIOs 96-127, gpio:
 gpio-111 (PWR_RS485   ) out hi
 gpio-112 (EN_RS485    ) out lo
 gpio-113 (HALF_DUPLEX   ) out lo

GPIOs 128-136, i2c/1-002d, tps65911, can sleep:
 gpio-129 (tps65910:led1   ) out lo
 gpio-131 (tps65910:led2   ) out lo
```

Further information about GPIO programming is available on https://www.kernel.org/doc/Documentation/gpio.txt

The following table gives an overview about the numeration of GPIO pins.

I/O Pin	TI® Label	Number (decimal)	Connector
GPIO0	GP0[10]	10	J2105
GPIO1	GP0[11]	11	J2105
GPIO2	GP0[12]	12	J2105
GPIO3	GP0[13]	13	J2105
GPIO4	GP0[14]	14	J2105
GPIO5	GP0[15]	15	J2105
GPIO6	GP0[16]	16	J2105
GPIO7	GP0[17]	17	J2105
GPIO8	GP0[18]	18	J2105
GPIO9	GP0[19]	19	J2105
GPIO10	GP0[20]	20	J2105
GPIO11	GP0[21]	21	J2105
GPIO12	GP0[22]	22	J2105
GPIO13	GP0[23]	23	J2105
GPIO14	GP0[24]	24	J2105
GPIO15	GP0[25]	25	J2105
GPIO16	GP0[26]	26	J2105
GPIO17	GP0[27]	27	J2105
GPIO18	GP0[28]	28	J2105
GPIO19	GP2[02]	66	J2105
GPIO20	GP2[16]	80	J2105
GPIO21	GP2[17]	81	J2105
GPIO22	GP2[18]	82	J2105
GPIO23	GP2[19]	83	J2105
GPIO24	GP2[20]	84	J2105
GPIO25	GP1[09]	41	J2105
GPIO26	GP1[11]	43	J2104
GPIO27	GP1[12]	44	J2104
GPIO28	GP1[27]	59	J2104
GPIO29	GP1[28]	60	J2104
GPIO30	GP1[29]	61	J2104
GPIO31	GP1[30]	62	J2104
GPIO32	GP2[00]	64	J2104
GPIO33	GP2[02]	66	J2104
GPIO34	GP2[07]	71	J2104
GPIO35	GP2[08]	72	J2104
GPIO36	GP2[09]	73	J2104
GPIO37	GP2[10]	74	J2104
GPIO38	GP2[11]	75	J2104
GPIO39	GP2[12]	76	J2104

The program below generates a square wave (ratio of 1:1) as an output demonstration and subsequently it reads the voltage level from the same pin as an input example (compiled on Ubuntu^{TM} distribution, needs root rights).

```
/* GPIO test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <errno.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>

#define MAX_GPIO                    136
#define GPIO_IN                     0
#define GPIO_OUT                    1
#define DIR_OUT                     "out"
#define DIR_IN                      "in"
#define GP2_17                      81                  /* J2105 Pin 38 */

#define MAX_CNT                     1000
#define SLP_TIME                    100

static const char *dev_export       = "/sys/class/gpio/export";
static const char *dev_unexport     = "/sys/class/gpio/unexport";
static const char *dev_dir          = "/sys/class/gpio/gpio%d/direction";
static const char *dev_val          = "/sys/class/gpio/gpio%d/value";
static int fd [MAX_GPIO];


int gpio_export (int gpio_pin)
{
    int fd_tmp;
    char str[8];

    fd_tmp = open (dev_export, O_WRONLY);
    if (fd_tmp < 0)
      return -1;

    sprintf (str, "%d", gpio_pin);
    if (write (fd_tmp, str, strlen (str)) != strlen (str))
    {
      close (fd_tmp);
      return -1;
    }

    close (fd_tmp);
    return 0;
}
```

```
int gpio_unexport (int gpio_pin)
{
    int fd_tmp;
    char str[8];

    fd_tmp = open (dev_unexport, O_WRONLY);
    if (fd_tmp < 0)
      return -1;

    sprintf (str, "%d", gpio_pin);
    if (write (fd_tmp, str, strlen (str)) != strlen (str))
    {
      close (fd_tmp);
      return -1;
    }

    close (fd_tmp);
    return 0;
}


int gpio_set_direction (int gpio_pin, int out_in)
{
    int fd_tmp;
    char str[128];

    sprintf (str, dev_dir, gpio_pin);
    fd_tmp = open (str, O_WRONLY);
    if (fd_tmp < 0)
      return -1;

    if (out_in)
    {
      if (write (fd_tmp, DIR_OUT, sizeof(DIR_OUT)-1) != sizeof(DIR_OUT)-1)
      {
        close (fd_tmp);
        return -1;
      }
    }
    else
    {
      if (write (fd_tmp, DIR_IN, sizeof(DIR_IN)-1) != sizeof(DIR_IN)-1)
      {
        close (fd_tmp);
        return -1;
      }
    }

    close (fd_tmp);
    return 0;
}
```

```
int gpio_open_write (int gpio_pin)
{
    char str[128];

    sprintf (str, dev_val, gpio_pin);
    fd [gpio_pin] = open (str, O_WRONLY);
    if (fd [gpio_pin] < 0)
      return -1;

    return 0;
}


int gpio_write (int gpio_pin, int value)
{
    char str[8];

    if ((value < 0) || (value > 1))
      return -1;
    sprintf (str, "%d", value);

    if (write (fd [gpio_pin], str, strlen (str)) != strlen (str))
      return -1;

    return lseek (fd [gpio_pin], 0, SEEK_SET);
}


void gpio_close_write (int gpio_pin)
{
    close (fd [gpio_pin]);
}


int gpio_open_read (int gpio_pin)
{
    char str[128];

    sprintf (str, dev_val, gpio_pin);
    fd [gpio_pin] = open (str, O_RDONLY);
    if (fd [gpio_pin] < 0)
      return -1;

    return 0;
}


int gpio_read (int gpio_pin, int *value)
{
    char str[8];

    if (read (fd [gpio_pin], str, sizeof (str)) < 0)
      return -1;

    str [1] = '\0';
    *value = atoi (str);

    return lseek (fd [gpio_pin], 0, SEEK_SET);
}
```

```
void gpio_close_read (int gpio_pin)
{
    close (fd [gpio_pin]);
}


int main (void)
{
    int i, value;

    printf ("\nGPIO test program\n");

    if ((! gpio_export (GP2_17)) && (! gpio_set_direction (GP2_17, GPIO_OUT)))
    {
      if (! gpio_open_write (GP2_17))
      {
        for (i = 0; i < MAX_CNT; i++)
        {
          gpio_write (GP2_17, 1);
          usleep (SLP_TIME);
          gpio_write (GP2_17, 0);
          usleep (SLP_TIME);
        }
        gpio_close_write (GP2_17);
      }
      else
      {
        printf ("\nCannot open GPIO !\n");
        gpio_unexport (GP2_17);
        return -1;
      }
      gpio_unexport (GP2_17);
    }
    else
    {
      printf ("\nGPIO init error !\n");
      gpio_unexport (GP2_17);
      return -1;
    }

    printf ("GPIO output part successfully finished. Please press Enter.");
    getchar ();

    if ((! gpio_export (GP2_17)) && (! gpio_set_direction (GP2_17, GPIO_IN)))
    {
      if (! gpio_open_read (GP2_17))
      {
        if (! gpio_read (GP2_17, &value))
          printf ("Input value on GP2[17] is %d\n", value);

        gpio_close_read (GP2_17);
      }
```

```
         else
         {
            printf ("\nCannot open GPIO !\n");
            gpio_unexport (GP2_17);
            return -1;
         }
         gpio_unexport (GP2_17);
      }
      else
      {
         printf ("\nGPIO init error !\n");
         gpio_unexport (GP2_17);
         return -1;
      }

      printf ("\n");
      return 0;
}
```

You can add interrupt support with a simple extension (only an extract):

```
#include <poll.h>

#define IRQ_FALL                 0
#define IRQ_RISE                 1
#define EDGE_RISE                "rising"
#define EDGE_FALL                "falling"
#define POLL_TIMEOUT             20000

static const char *dev_edge      = "/sys/class/gpio/gpio%d/edge";

int gpio_set_edge (int gpio_pin, int rise_fall)
{
      int fd_tmp;
      char str[128];

      sprintf (str, dev_edge, gpio_pin);
      fd_tmp = open (str, O_WRONLY);
      if (fd_tmp < 0)
         return -1;

      if (rise_fall)
      {
         if (write (fd_tmp, EDGE_RISE, sizeof(EDGE_RISE)-1) != sizeof(EDGE_RISE)-1)
         {
            close (fd_tmp);
            return -1;
         }
      }
```

```
   else
   {
      if (write (fd_tmp, EDGE_FALL, sizeof(EDGE_FALL)-1) != sizeof(EDGE_FALL)-1)
      {
         close (fd_tmp);
         return -1;
      }
   }

   close (fd_tmp);
   return 0;
}


int main (void)
{
   int i, ret, value;
   struct pollfd poll_fd;

   printf ("\nGPIO IRQ test program\n");

   if ((! gpio_export (GP2_17)) && (! gpio_set_direction (GP2_17, GPIO_IN)))
   {
      if ((! gpio_set_edge (GP2_17, IRQ_RISE)) && (! gpio_open_read (GP2_17)))
      {
         poll_fd.fd      = fd [GP2_17];
         poll_fd.events  = POLLPRI;
         poll_fd.revents = 0;

         printf ("Waiting for interrupt ...\n");

         /* Read before Poll is necessary */
         gpio_read (GP2_17, &value);
         ret = poll (&poll_fd, 1, POLL_TIMEOUT);

         switch (ret)
         {
            case -1: printf ("Interrupt routine fails !\n"); break;
            case  0: printf ("Timeout has occured !\n"); break;
            case  1: if (poll_fd.revents & POLLPRI)
                        printf ("IRQ successfully detected.\n");
                     else
                        printf ("Unexpected problem !\n");
                     break;
            default : printf ("Unexpected problem !\n"); break;
         }

         gpio_close_read (GP2_17);
      }
      else
      {
         printf ("\nCannot set edge or open GPIO !\n");
         gpio_unexport (GP2_17);
         return -1;
      }
      gpio_unexport (GP2_17);
   }
```

```
    else
    {
      printf ("\nGPIO init error !\n");
      gpio_unexport (GP2_17);
      return -1;
    }

    printf ("\n");
    return 0;
}
```

10.7 Watchdog Example

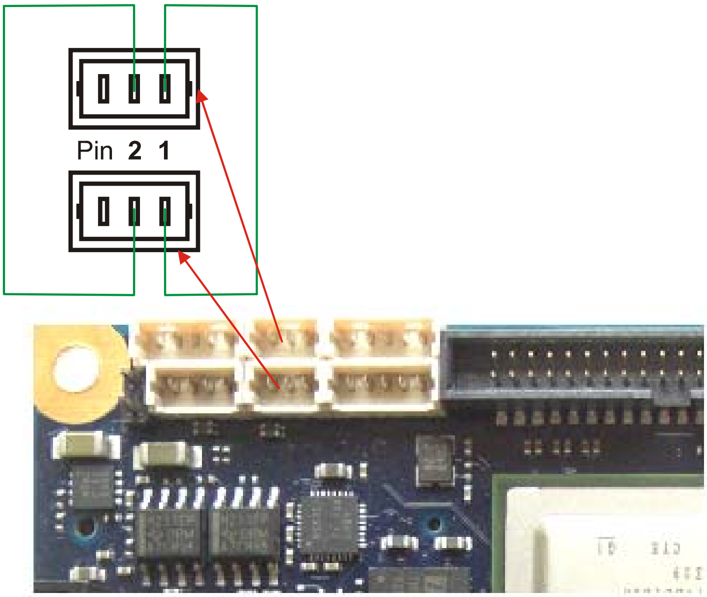
The watchdog option represents a nice feature for industrial applications. You can also download a watchdog daemon with 'sudo apt-get install watchdog' for additional features.
Further information about watchdog programming is available on https://www.kernel.org/doc/Documentation/watchdog/watchdog-api.txt. The demonstration program needs root rights (compiled on Ubuntu™ distribution).

```
/* Watchdog test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <stdio.h>
#include <fcntl.h>
#include <errno.h>
#include <asm/types.h>
#include <linux/watchdog.h>

#define WDT_TIMEOUT          20                   /* Seconds */
#define WDT_TRIG_TIME        3
#define WDT_DELAY            10                   /* Seconds */
#define WDT_MARKUP           ((WDT_TIMEOUT - WDT_DELAY) + 10)

static const char *device = "/dev/watchdog";

int main (void)
{
    int fd, i;
    /*int bootstat;*/
    int wdt_time = WDT_TIMEOUT;

    fd = open (device, O_WRONLY);
    if (fd < 0)
    {
      printf ("\nWatchdog init error !\n\n");
      return -1;
    }

    /* Check if last boot is caused by watchdog - NOT WORKING PROPERLY */
    /*if (ioctl (fd, WDIOC_GETBOOTSTATUS, &bootstat))
      printf ("\nCannot read watchdog status !\n");
    else
      printf ("\nLast boot is caused by : %s\n", (bootstat != 0) ? "Watchdog" : "Power-On-Reset"); */

    if (ioctl (fd, WDIOC_SETTIMEOUT, &wdt_time))
      printf ("\nSetting of watchdog timeout fails !\n");
    else
      printf ("\nNew timeout value is : %d seconds\n", wdt_time);
    for (i = 0; i < WDT_TRIG_TIME; i++)
    {
      if (write (fd, "\0", 1) != 1)                    /* Trigger watchdog */
        break;
      sleep (WDT_DELAY);
    }

    /* Should be reboot after 40 seconds from program start */
    if (i >= WDT_TRIG_TIME)
      sleep (wdt_time + WDT_MARKUP);                   /* Cause a timeout */
    else
      printf ("Trigger process fails !\n\n");

    /* Should never be reached */
    write (fd, "V", 1);                                /* Magic stop */
    close (fd);                                        /* Deactivates watchdog */
    return 0;
}
```

10.8 CAN Bus Examples

10.8.1 CAN Loopback Test Utility

The following picture shows a simple wiring plan for the loopback test. With very short cables you need no termination resistors.



It is possible that the program does not work correctly after start-up. In this case you should execute the commands below (naturally you can use another baudrate than 125 kBaud):

```
sudo ifconfig can0 down
sudo ifconfig can1 down
sudo ip link set can0 up type can bitrate 125000
sudo ip link set can1 up type can bitrate 125000
```

Further details about CAN bus and socket programming is available on https://www.kernel.org/doc/Documentation/networking/can.txt and also on http://www.can-cia.org/fileadmin/cia/files/icc/13/hartkopp.pdf. As an exception this program can run with user and root rights (compiled on Ubuntu^{TM} distribution).

```
/* CAN bus test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <stdio.h>
#include <fcntl.h>
#include <errno.h>
#include <string.h>
```

```
#include <net/if.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <sys/ioctl.h>
#include <linux/can.h>
#include <linux/can/raw.h>

#define CAN0_INDEX          0
#define CAN1_INDEX          1
#define MAX_CAN_DEV         1
#define MAX_CAN_LEN         8
#define MAX_CAN_SSF         0x7FFUL          /* CAN 2.0A */
#define DEF_ERR_MASK        0x1FFUL
#define WAIT_TIME           10000            /* 10 ms */
#define MAX_NAME_LEN        8

static int skt;
static const char *dev_name = "can%d";

int init_can_bus (canid_t id_min, canid_t id_max)
{
    can_err_mask_t err_mask = DEF_ERR_MASK;
    struct sockaddr_can addr;
    struct can_filter rfilter;

    if ((skt = socket (PF_CAN, SOCK_RAW, CAN_RAW)) < 0)
      return -1;

    if (setsockopt (skt, SOL_CAN_RAW, CAN_RAW_ERR_FILTER, &err_mask, sizeof (err_mask)) < 0)
      return -1;
    if (id_min && id_max)
    {
      /* Define an identifier filter */
      if ((id_min > id_max) || (id_min > MAX_CAN_SSF) || (id_max > MAX_CAN_SSF))
        return -1;

      rfilter.can_id   = id_min;
      rfilter.can_mask = (~ (id_max - id_min)) & MAX_CAN_SSF;
      if (setsockopt (skt, SOL_CAN_RAW, CAN_RAW_FILTER, &rfilter, sizeof (rfilter)) < 0)
        return -1;
    }

    addr.can_family  = AF_CAN;
    addr.can_ifindex = 0;                    /* Bind all controllers */
    return bind (skt, (struct sockaddr *) &addr, sizeof (addr));
}


int close_can_bus (void)
{
    close (skt);
}
```

```
int disable_id_filter (void)
{
    return setsockopt (skt, SOL_CAN_RAW, CAN_RAW_FILTER, NULL, 0);
}


int send_can_frame (int device, canid_t can_id, unsigned char data[], int data_len)
{
    int i, nbytes;
    struct ifreq ifr;
    struct sockaddr_can addr;
    struct can_frame frame;
    char str[MAX_NAME_LEN];

    if ((device < 0) || (device > MAX_CAN_DEV) ||
            (data_len < 0) || (data_len > MAX_CAN_LEN) ||
            (can_id < 0) || (can_id > MAX_CAN_SSF))
      return -1;

    sprintf (str, dev_name, device);
    memset (&ifr, 0, sizeof (struct ifreq));
    strcpy (ifr.ifr_name, str);
    if (ioctl (skt, SIOCGIFINDEX, &ifr) < 0)
      return -1;

    frame.can_id  = can_id;
    frame.can_dlc = (unsigned char) data_len;
    for (i = 0; i < data_len; i++)
      frame.data[i] = data[i];

    addr.can_family  = AF_CAN;
    addr.can_ifindex = ifr.ifr_ifindex;
    if ((nbytes = sendto (skt, &frame, sizeof (struct can_frame), 0, (struct sockaddr *) &addr, sizeof (addr))) < 0)
      return -1;

    if (nbytes != sizeof (frame))
      return -1;

    return 0;
}


int recv_can_frame (char name[], unsigned char data[], int *data_len)
{
    int i;
    struct ifreq ifr;
    struct sockaddr_can addr;
    struct can_frame frame;
    socklen_t skt_len = sizeof (addr);

    if (recvfrom (skt, &frame, sizeof (struct can_frame), MSG_WAITALL, (struct sockaddr *) &addr, &skt_len) < 0)
      return -1;
    for (i = 0; i < frame.can_dlc; i++)
      data[i] = frame.data[i];
    *data_len = frame.can_dlc;

    /* Search for name */
    memset (&ifr, 0, sizeof (struct ifreq));
    ifr.ifr_ifindex = addr.can_ifindex;
    if (ioctl (skt, SIOCGIFNAME, &ifr) < 0)
      return -1;

    strncpy (name, ifr.ifr_name, MAX_NAME_LEN);
    name[MAX_NAME_LEN-1] = '\0';
    for (i = 0; i < strlen (name); i++)
      name[i] = toupper (name[i]);

    return 0;
}


int main (void)
{
    int data_len;
    char name [MAX_NAME_LEN];
    unsigned char data [MAX_CAN_LEN];

    printf ("\nCAN bus test program\n");

    /* For example only the identifier range from 0x200 to 0x2FF is valid.
       If there are no restrictions regarding identifiers use init_can_bus (0, 0) */
    if (init_can_bus (0x200UL, 0x2FFUL))
    {
      close_can_bus ();
      printf ("Cannot initialize the CAN interface !\n\n");
      return -1;
    }

    memset (&data, 0, sizeof (data));
    data[0]  = 0x11;
    data[1]  = 0x22;
    data[2]  = 0x33;
    data[3]  = 0x44;
    data[4]  = 0x55;
    data[5]  = 0x66;
    data[6]  = 0x77;
    data[7]  = 0x88;
    data_len = 8;

    if (send_can_frame (CAN0_INDEX, 0x201UL, data, data_len))
    {
      close_can_bus ();
      printf ("Cannot transmit CAN bus data !\n\n");
      return -1;
    }

    printf ("\nSome data are successfully transmitted from: CAN0\n");
    usleep (WAIT_TIME);
    memset (&data, 0, sizeof (data));
    if (recv_can_frame (name, data, &data_len))
    {
      close_can_bus ();
      printf ("CAN bus receive function fails !\n\n");
      return -1;
    }
    else
    {
      printf ("Some data are successfully received from: %s\n", name);
      printf ("Data:  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X\n\n",
                                   data[0], data[1], data[2], data[3], data[4], data[5], data[6], data[7]);
    }

    close_can_bus ();
    return 0;
}
```

If problems occur then you can try the following commands:

❶ Reset network interfaces with

```
# sudo service networking restart
```

❷ Get more information with

```
# sudo ifconfig can0
can0        Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
            UP RUNNING NOARP  MTU:16  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:10
            RX bytes:0 (0.0 B)  TX bytes:32 (32.0 B)
            Interrupt:52
```

```
# sudo ifconfig can1
can1        Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
            UP RUNNING NOARP  MTU:16  Metric:1
            RX packets:4 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:10
            RX bytes:32 (32.0 B)  TX bytes:0 (0.0 B)
            Interrupt:55
```

❸ or with (example with 'can0' for general interface information)

```
# ip -details link show can0
2: can0:    <NOARP,UP,LOWER_UP,ECHO> mtu 16 qdisc pfifo_fast state UNKNOWN mode DEFAULT ...
            link/can
            can <TRIPLE-SAMPLING> state ERROR-ACTIVE (berr-counter tx 0 rx 0) restart-ms 0
            bitrate 125000 sample-point 0.875
            tq 500 prop-seg 6 phase-seg1 7 phase-seg2 2 sjw 1
            d_can: tseg1 1..16 tseg2 1..8 sjw 1..4 brp 1..1024 brp-inc 1
            clock 20000000
```

❹ or with (transmit part shortened, 'lo' implies 'Local Loopback')

```
# cat /proc/net/dev
Inter-|   Receive                                                            |Transmit
 face |   bytes   packets  errs   drop   fifo   frame   compr. multicast |bytes  packets ..
   lo:    ???     ???      0      0      0      0       0      0          ???    ???    ....
 can0:    0       0        0      0      0      0       0      0          32     4      ....
 can1:    32      4        0      0      0      0       0      0          0      0      ....
 eth0:    0       0        0      0      0      0       0      0          0      0      ....
 eth1:    0       0        0      0      0      0       0      0          0      0      ....
```

The socket interface offers an REC(EI)V(E)_OWN_MSGS option which does not work properly. The call of the following routine has no effect.

```
int enable_own_msgs (void)
{
    int own_msgs = 1;

    return setsockopt (skt, SOL_CAN_RAW, CAN_RAW_RECV_OWN_MSGS, &own_msgs, sizeof (own_msgs));
}
```

After entering some commands this option is still available.

```
sudo ifconfig can0 down
sudo ip link set can0 type can loopback on
sudo ip link set can0 up type can bitrate 125000
```

Now you can expand/change the previous example (extract):

```
#define DELAY_TIME          100              /* 100 us */
#define MAX_LOOPS           10000            /* Overall 1 second */

int detect_can_frame (void)
{
    struct sockaddr_can addr;
    struct can_frame frame;
    socklen_t skt_len = sizeof (addr);

    if (recvfrom (skt, &frame, sizeof (struct can_frame), MSG_DONTWAIT | MSG_PEEK,
                                                 (struct sockaddr *) &addr, &skt_len) < 0)

      return -1;

    if (frame.can_dlc > 0)
      return 0;

    return -1;
}


int main (void)
{
    int i, data_len;
    char name [MAX_NAME_LEN];
    unsigned char data [MAX_CAN_LEN];
    printf ("\nCAN bus test program\n");

    /* For example only the identifier range from 0x200 to 0x2FF is valid.
       If there are no restrictions regarding identifiers use init_can_bus (0, 0) */
    if (init_can_bus (0x200UL, 0x2FFUL))
    {
      close_can_bus ();
      printf ("Cannot initialize the CAN interface !\n\n");
      return -1;
    }

    memset (&data, 0, sizeof (data));
    data[0]  = 0x11;
    data[1]  = 0x22;
    data[2]  = 0x33;
    data[3]  = 0x44;
    data[4]  = 0x55;
    data[5]  = 0x66;
    data[6]  = 0x77;
    data[7]  = 0x88;
    data_len = 8;

    if (send_can_frame (CAN0_INDEX, 0x201UL, data, data_len))
    {
      close_can_bus ();
      printf ("Cannot transmit CAN bus data !\n\n");
      return -1;
    }

    printf ("\nSome data are successfully transmitted from: CAN0\n\n");

    for (i = 0; i < MAX_LOOPS; i++)
    {
      usleep (DELAY_TIME);
      if (! detect_can_frame ())
      {
        memset (&data, 0, sizeof (data));
        if (recv_can_frame (name, data, &data_len))
        {
          close_can_bus ();
          printf ("CAN bus receive function fails !\n\n");
          return -1;
        }
        else
        {
          printf ("Some data are successfully received from: %s\n", name);
          printf ("Data:  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X  0x%02X\n\n",
                                  data[0], data[1], data[2], data[3], data[4], data[5], data[6], data[7]);
        }
      }
    }

    close_can_bus ();
    return 0;
}
```

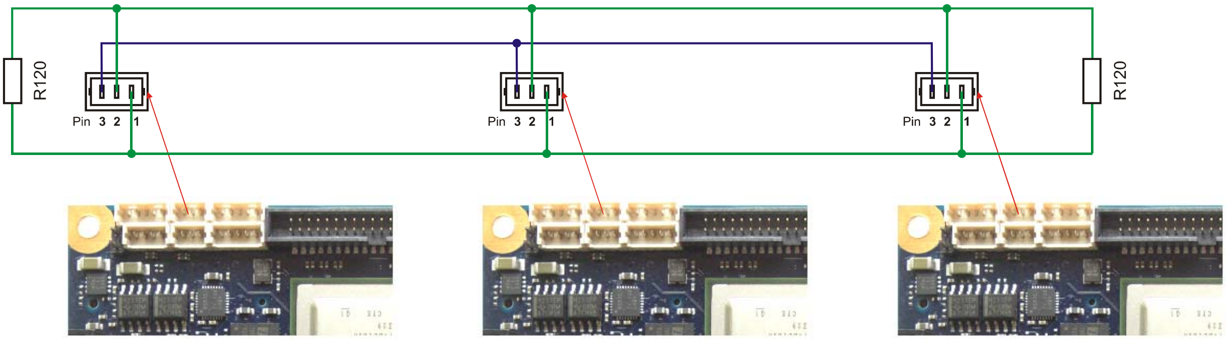
10.8.2 Single CAN Controller Usage

You can replace the 'sendto' and 'recvfrom' calls in single CAN controller applications with the standard 'write' and 'read' commands. For example (sourcecode simplified):

```
/********************** * INIT ROUTINE ***** ***************/

int init_can_bus (canid_t id_min, canid_t id_max)
{
    struct sockaddr_can addr;
    struct ifreq ifr;

    if ((skt = socket (PF_CAN, SOCK_RAW, CAN_RAW)) < 0)
      return -1;

    memset (&ifr, 0, sizeof (struct ifreq));
    strcpy (ifr.ifr_name, "can0");
    if (ioctl (skt, SIOCGIFINDEX, &ifr) < 0)
      return -1;

    addr.can_family  = AF_CAN;
    addr.can_ifindex = ifr.ifr_ifindex;                          /* Bind only CAN0 controller */
    return bind (skt, (struct sockaddr *) &addr, sizeof (addr));
}

/*********************** TRANSMIT ROUTINE ********************/

int send_can_frame (canid_t can_id, unsigned char data[], int data_len)
{
    int i, nbytes;
    struct can_frame frame;

    frame.can_id  = can_id;
    frame.can_dlc = (unsigned char) data_len;
    for (i = 0; i < data_len; i++)
      frame.data[i] = data[i];

    if (write (skt, &frame, sizeof (struct can_frame)) < 0)
      return -1;

    return 0;
}

/*********************** RECEIVE ROUTINE ********************/

int recv_can_frame (unsigned char data[], int *data_len)
{
    int i;
    struct can_frame frame;

    if (read (skt, &frame, sizeof (struct can_frame)) < 0)
      return -1;
    for (i = 0; i < frame.can_dlc; i++)
      data[i] = frame.data[i];
    *data_len = frame.can_dlc;

    return 0;
}
```

10.8.3 CAN Wiring

For longer distances it is recommended to terminate the CAN bus at both ends by terminating resistors. Another important measure: all GND contacts (pin 3) should be tied together.



10.9 Backlight Example

An easy way to modify the backlight brightness value consists in use of the sysfs interface. The directory /sys/class/backlight/ktam3874-lcd/ contains the following entries (please note - this directory appears only if an LCD panel is enabled):

actual_brightness	Actual brightness value (read only)
brightness	Changeable brightness value
max_brightness	Maximal brightness value (read only)
bl_power	Changeable power state (on/off)

The program below gives an overview regarding the possible setting options.

```
/* Backlight test program

 * Copyright (c) 2013  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 */

#include <stdio.h>
#include <fcntl.h>
#include <errno.h>
#include <string.h>

#define MAX_BKL_VAL         255
#define MAX_BKL_POW         1               /* 0 = enable, 1 = disable */
#define WRITE_DELAY         20
#define POWER_DELAY         100

static const char *dev_pow = "/sys/class/backlight/ktam3874-lcd/bl_power";
static const char *dev_val = "/sys/class/backlight/ktam3874-lcd/brightness";
static const char *dev_act = "/sys/class/backlight/ktam3874-lcd/actual_brightness";


int write_bkl_val (int val)
{
    int fd;
    char str[8];

    if ((val < 0) || (val > MAX_BKL_VAL))
      return -1;

    fd = open (dev_val, O_WRONLY);
    if (fd < 0)
      return -1;

    sprintf (str, "%d", val);
    if (write (fd, str, strlen (str)) != strlen (str))
    {
      close (fd);
      return -1;
    }

    close (fd);
    return 0;
}
```

```
int read_bkl_val (int *val)
{
    int fd;
    char str[8];

    fd = open (dev_act, O_RDONLY);
    if (fd < 0)
      return -1;

    if (read (fd, str, sizeof (str)) < 0)
    {
      close (fd);
      return -1;
    }

    str [3] = '\0';                                   /* Maximal value is 255 */
    *val = atoi (str);
    close (fd);
    return 0;
}


int write_bkl_power (int pow_state)
{
    int fd;
    char str[8];

    if ((pow_state < 0) || (pow_state > MAX_BKL_POW))
      return -1;

    fd = open (dev_pow, O_WRONLY);
    if (fd < 0)
      return -1;

    sprintf (str, "%d", pow_state);
    if (write (fd, str, strlen (str)) != strlen (str))
    {
      close (fd);
      return -1;
    }

    close (fd);
    return 0;
}


int main (void)
{
    int value;

    printf ("\nBacklight test program\n");

    if (! read_bkl_val (&value))
      printf ("Actual backlight value: %d\n", value);
    else
      printf ("Backlight read function fails !\n");
    value = 96;
    if (! write_bkl_val (value))
      printf ("New backlight value   : %d\n", value);
    else
      printf ("Backlight write function fails !\n");

    usleep (WRITE_DELAY);

    if (! read_bkl_val (&value))
      printf ("Check backlight value : %d\n", value);
    else
    printf ("Backlight read function fails !\n");

    if (! write_bkl_power (1))
      printf ("-- Disable backlight --\n");
    else
      printf ("Backlight power function fails !\n");

    usleep (POWER_DELAY);

    if (! read_bkl_val (&value))
      printf ("Check backlight value : %d\n", value);
    else
      printf ("Backlight read function fails !\n");

    if (! write_bkl_power (0))
      printf ("-- Enable backlight --\n");
    else
      printf ("Backlight power function fails !\n");

    usleep (POWER_DELAY);

    if (! read_bkl_val (&value))
      printf ("Check backlight value : %d\n\n", value);
    else
      printf ("Backlight read function fails !\n\n");

    return 0;
}
```

10.10 RS485 Interface Example

The application programming of the RS485 interface can be a little tricky (up to Kernel version 1.03). Normally the RTS status line should switch between transmit and receive mode but after initialization of the serial interface the status line does not work correctly. If you use the devmem2 utility directly after the boot process with

sudo ./devmem2 481AA010 d 2 resp. **sudo ./devmem2 481AA010 d 0**

you can toggle the RTS status line without a problem but the same approach fails after initialization by an application program (auto-RTS and auto-CTS not enabled, clearing the complete EFR = Enhanced Feature Register also has no effect). If you want to control the RTS pin by calling ioctl() with TIOCMGET/TIOCMSET parameters the result belies over the true state. The only way out of this situation is to redefine the RTS pin as a GPIO pin. The program below uses the sysfs interface to modify the pin multiplexing and initialize the GPIO part.

```
/* RS485 test program
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License. */

#include <termios.h>
#include <stdio.h>
#include <string.h>
#include <fcntl.h>

#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))
#define FALSE                    0
#define TRUE                     1
#define DEF_BAUDRATE             115200
#define PIN138_GPIO              "E0080"
#define PIN138_RTS               "E0020"
#define GP2_3                    67
#define DIR_OUT                  "out"
#define MIN_RCV_CHARS            8

static const char *device        = "/dev/ttyO5";
static const char *pin_138       = "/sys/kernel/debug/omap_mux/vin0_hsync0";
static const char *dev_export    = "/sys/class/gpio/export";
static const char *dev_unexport  = "/sys/class/gpio/unexport";
static const char *dev_dir       = "/sys/class/gpio/gpio%d/direction";
static const char *dev_val       = "/sys/class/gpio/gpio%d/value";
static int fd;


int update_device (const char *dev, char *str, int seek_ctrl)
{
    int fd_tmp;

    fd_tmp = open (dev, O_WRONLY);
    if (fd_tmp < 0)
       return -1;
    if (write (fd_tmp, str, strlen (str)) != strlen (str))
    {
      close (fd_tmp);
      return -1;
    }

    if (seek_ctrl)
      lseek (fd_tmp, 0, SEEK_SET);

    close (fd_tmp);
    return 0;
}


int rts_as_gpio_open (void)
{
    char dev [64];
    char str [16];

    strcpy (str, PIN138_GPIO);
    if (update_device (pin_138, str, FALSE))
      return -1;

    sprintf (str, "%d", GP2_3);
    if (update_device (dev_export, str, FALSE))
      return -1;

    sprintf (dev, dev_dir, GP2_3);
    strcpy (str, DIR_OUT);
    if (update_device (dev, str, FALSE))
      return -1;

    return 0;
}


int rts_as_gpio_close (void)
{
    char str [16];

    sprintf (str, "%d", GP2_3);
    if (update_device (dev_unexport, str, FALSE))
      return -1;

    strcpy (str, PIN138_RTS);
    if (update_device (pin_138, str, FALSE))
      return -1;

    return 0;
}
```

```
int rts_transmit_control (int tx_enable)
{
    char dev [64];
    char str [16];

    sprintf (dev, dev_val, GP2_3);
    sprintf (str, "%d", tx_enable);
    if (update_device (dev, str, TRUE))
      return -1;

    return 0;
}


int init_uart_interface (int baudrate)
{
    struct termios new_values;

    fd = open (device, O_RDWR | O_NOCTTY);
    if (fd < 0 )
      return -1;

    if (tcflush (fd, TCIOFLUSH))
      return -1;

    switch (baudrate)
    {
      case 300      : baudrate = B300; break;
      case 600      : baudrate = B600; break;
      case 1200     : baudrate = B1200; break;
      case 2400     : baudrate = B2400; break;
      case 4800     : baudrate = B4800; break;
      case 9600     : baudrate = B9600; break;
      case 19200    : baudrate = B19200; break;
      case 38400    : baudrate = B38400; break;
      case 57600    : baudrate = B57600; break;
      case 115200   : baudrate = B115200; break;
      default       : return -1;
    }
    memset (&new_values, 0, sizeof (new_values));
    new_values.c_cflag = CS8 | CLOCAL | CREAD;
    new_values.c_iflag = IGNBRK | ICRNL;
    new_values.c_lflag = ICANON;
    new_values.c_cc[VMIN] = 1;

    if (cfsetispeed (&new_values, baudrate))
      return -1;
    if (cfsetospeed (&new_values, baudrate))
      return -1;

    return tcsetattr (fd, TCSANOW, &new_values);
}
```

```
int rs485_outgoing_msg (void)
{
    char out_str[] = "the quick brown fox jumps over the lazy dog";

    if (rts_transmit_control (TRUE))
        return -1;

    if (write (fd, out_str, sizeof (out_str)) != sizeof (out_str))
        return -1;

    if (tcdrain (fd))
        return -1;

    if (rts_transmit_control (FALSE))
        return -1;

    return 0;
}


int rs485_incoming_msg (void)
{
    int i;
    unsigned char data [MIN_RCV_CHARS];

    if (tcflush (fd, TCIFLUSH))
        return -1;

    memset (&data, 0, sizeof (data));
    if (read (fd, data, MIN_RCV_CHARS) < 0)
        return -1;

    printf ("Received data = ");
    for (i = 0; i < MIN_RCV_CHARS; i++)
        printf ("0x%02X ", data [i]);
    printf ("\n");

    return 0;
}


int main (void)
{
    printf ("\nRS485 test program\n");

    if (init_uart_interface (DEF_BAUDRATE))
    {
        close (fd);
        printf ("UART init error!\n\n");
        return -1;
    }

    if (rts_as_gpio_open () || rts_transmit_control (FALSE))
    {
        rts_as_gpio_close (); close (fd);
        printf ("RTS control error!\n\n");
        return -1;
    }
    if (rs485_outgoing_msg ())
    {
      rts_as_gpio_close (); close (fd);
      printf ("Error during transmit operation!\n\n");
      return -1;
    }

    if (rs485_incoming_msg ())
    {
      rts_as_gpio_close (); close (fd);
      printf ("Error during receive operation!\n\n");
      return -1;
    }

    rts_as_gpio_close ();
    close (fd);

    printf ("RS485 test successfully finished.\n\n");
    return 0;
}
```

10.11 Sound Programming with ALSA

The following examples assume that the compilation runs on an x86 Desktop PC using a cross compiler. One possible Makefile might look like this (Ubuntu^{TM} environment):

```
CC  :=  arm-linux-gnueabihf-gcc  -march=armv7
LDFLAGS  :=  -l asound  -l c  -l m

all:  pcm

pcm.o:  pcm.c

pcm:  pcm.o
    $(CC)  -o  $@  pcm.o  $(LDFLAGS)

clean:
    rm  pcm  pcm.o
```

You can see the special shared library libasound.so (libc and libm are standard libraries and referenced by libasound) which is typically not available. The library package also contains some necessary Include-files. One possible way to install the binary library consists in the download of libasound2-dev on the target hardware with

```
sudo  apt-get  install  libasound2-dev
```

and copy it from the Archive-directory to a storage medium

```
sudo  cp  /var/cache/apt/archives/libasound2-dev_..._armhf.deb  <storage medium>
```

On the x86 Desktop PC extract the files to a temporary directory with

```
sudo  dpkg  -x  libasound2-dev_..._armhf.deb
```

Now you can find out the library search paths by adding the -v option into the Makefile:

```
LDFLAGS :=  -l asound  -l c  -l m  -v
```

and then copy libasound.so into a search path.

The easiest way to install the Include-files is to install libasound2-dev also on the x86 Desktop PC (no differences between x86 and ARM® Include-files, likely the compiler uses the x86 search paths). The cross compiler ignores the x86 library file.
There are some interesting sourcecode examples on the webpage

http://www.alsa-project.org/alsa-doc/alsa-lib/examples.html

The PCM demo provides a comprehensive playback program (sinus frequency range 50 - 22000 Hz).

```
/**************************************************************
*  This small demo sends a simple sinusoidal wave to your audio output

*  This program is free software; you can redistribute it and/or modify
*  it under the terms of the GNU General Public License as published by
*  the Free Software Foundation; either version 2 of the License, or
*  (at your option) any later version.
*  This program is distributed in the hope that it will be useful,
*  but WITHOUT ANY WARRANTY; without even the implied warranty of
*  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
*  GNU General Public License for more details.
*  You should have received a copy of the GNU General Public License
*  along with this program; if not, write to the Free Software
*  Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307 USA
**************************************************************/
```

```
#include <stdio.h>
#include <getopt.h>
#include <math.h>
#include <alsa/asoundlib.h>

static char *device = "plughw:0,0";                       /* playback device */
static snd_pcm_format_t format = SND_PCM_FORMAT_S16;      /* sample format */
static unsigned int rate = 44100;                         /* stream rate */
static unsigned int channels = 2;                         /* count of channels */
static unsigned int buffer_time = 500000;                 /* ring buffer length in us */
static unsigned int period_time = 100000;                 /* period time in us */
static double freq = 440;                                 /* sinusoidal wave frequency in Hz */
static int verbose = 0;                                   /* verbose flag */
static int resample = 1;                                  /* enable alsa-lib resampling */
static int period_event = 0;                              /* produce poll event after each period */
static snd_pcm_sframes_t buffer_size;
static snd_pcm_sframes_t period_size;
static snd_output_t *output = NULL;

struct async_private_data
{
    signed short *samples;
    snd_pcm_channel_area_t *areas;
    double phase;
};

struct transfer_method
{
    const char *name;
    snd_pcm_access_t access;
    int (*transfer_loop)(snd_pcm_t *handle, signed short *samples, snd_pcm_channel_area_t *areas);
};


static void generate_sine (const snd_pcm_channel_area_t *areas,  snd_pcm_uframes_t offset,
                                                          int count, double *_phase)
{
    static double max_phase = 2. * M_PI;
    double phase = *_phase;
    double step = max_phase * freq / (double) rate;
    unsigned char *samples[channels];
    int steps[channels];
    unsigned int chn;
    int format_bits = snd_pcm_format_width (format);
    unsigned int maxval = (1 << (format_bits - 1)) - 1;
    int bps = format_bits / 8;                                  /* bytes per sample */
    int phys_bps = snd_pcm_format_physical_width (format) / 8;
    int big_endian = snd_pcm_format_big_endian (format) == 1;
    int to_unsigned = snd_pcm_format_unsigned (format) == 1;
    int is_float = (format == SND_PCM_FORMAT_FLOAT_LE || format == SND_PCM_FORMAT_FLOAT_BE);
```

```
/* verify and prepare the contents of areas */
for (chn = 0; chn < channels; chn++)
{
  if ((areas[chn].first % 8) != 0)
  {
    printf ("areas[%i].first == %i, aborting...\n", chn, areas[chn].first);
    exit (EXIT_FAILURE);
  }
  samples[chn] = (((unsigned char *) areas[chn].addr) + (areas[chn].first / 8));
  if ((areas[chn].step % 16) != 0)
  {
    printf ("areas[%i].step == %i, aborting...\n", chn, areas[chn].step);
    exit (EXIT_FAILURE);
  }
  steps[chn] = areas[chn].step / 8;
  samples[chn] += offset * steps[chn];
}

/* fill the channel areas */
while (count-- > 0)
{
  union {
    float f;
    int i;
  } fval;
  int res, i;

  if (is_float)
  {
    fval.f = sin (phase) * maxval;
    res = fval.i;
  }
  else
    res = sin (phase) * maxval;

  if (to_unsigned)
    res ^= 1U << (format_bits - 1);

  for (chn = 0; chn < channels; chn++)
  {
    /* generate data in native endian format */
    if (big_endian)
    {
      for (i = 0; i < bps; i++)
        *(samples[chn] + phys_bps - 1 - i) = (res >> i * 8) & 0xff;
    }
    else
    {
      for (i = 0; i < bps; i++)
        *(samples[chn] + i) = (res >>  i * 8) & 0xff;
    }
    samples[chn] += steps[chn];
  }
```

```
        phase += step;
        if (phase >= max_phase)
            phase -= max_phase;
    }
    *_phase = phase;
}


static int set_hwparams (snd_pcm_t *handle, snd_pcm_hw_params_t *params, snd_pcm_access_t access)
{
    unsigned int rrate;
    snd_pcm_uframes_t size;
    int err, dir;

    /* choose all parameters */
    err = snd_pcm_hw_params_any (handle, params);
    if (err < 0)
    {
        printf ("Broken configuration for playback: no configurations available: %s\n", snd_strerror (err));
        return err;
    }

    /* set hardware resampling */
    err = snd_pcm_hw_params_set_rate_resample (handle, params, resample);
    if (err < 0)
    {
        printf ("Resampling setup failed for playback: %s\n", snd_strerror (err));
        return err;
    }

    /* set the interleaved read/write format */
    err = snd_pcm_hw_params_set_access (handle, params, access);
    if (err < 0)
    {
        printf ("Access type not available for playback: %s\n", snd_strerror (err));
        return err;
    }

    /* set the sample format */
    err = snd_pcm_hw_params_set_format (handle, params, format);
    if (err < 0)
    {
        printf ("Sample format not available for playback: %s\n", snd_strerror (err));
        return err;
    }

    /* set the count of channels */
    err = snd_pcm_hw_params_set_channels (handle, params, channels);
    if (err < 0)
    {
        printf ("Channels count (%i) not available for playbacks: %s\n", channels, snd_strerror (err));
        return err;
    }
    /* set the stream rate */
    rrate = rate;
    err = snd_pcm_hw_params_set_rate_near (handle, params, &rrate, 0);
    if (err < 0)
    {
      printf ("Rate %iHz not available for playback: %s\n", rate, snd_strerror (err));
      return err;
    }

    if (rrate != rate)
    {
      printf ("Rate doesn't match (requested %iHz, get %iHz)\n", rate, err);
      return -EINVAL;
    }

    /* set the buffer time */
    err = snd_pcm_hw_params_set_buffer_time_near (handle, params, &buffer_time, &dir);
    if (err < 0)
    {
      printf ("Unable to set buffer time %i for playback: %s\n", buffer_time, snd_strerror (err));
      return err;
    }

    err = snd_pcm_hw_params_get_buffer_size (params, &size);
    if (err < 0)
    {
      printf ("Unable to get buffer size for playback: %s\n", snd_strerror (err));
      return err;
    }
    buffer_size = size;

    /* set the period time */
    err = snd_pcm_hw_params_set_period_time_near (handle, params, &period_time, &dir);
    if (err < 0)
    {
      printf ("Unable to set period time %i for playback: %s\n", period_time, snd_strerror (err));
      return err;
    }

    err = snd_pcm_hw_params_get_period_size (params, &size, &dir);
    if (err < 0)
    {
      printf ("Unable to get period size for playback: %s\n", snd_strerror (err));
      return err;
    }
    period_size = size;

    /* write the parameters to device */
    err = snd_pcm_hw_params (handle, params);
    if (err < 0)
    {
      printf ("Unable to set hw params for playback: %s\n", snd_strerror (err));
      return err;
    }
    return 0;
}
```

```
static int set_swparams (snd_pcm_t *handle, snd_pcm_sw_params_t *swparams)
{
    int err;

    /* get the current swparams */
    err = snd_pcm_sw_params_current (handle, swparams);
    if (err < 0)
    {
      printf ("Unable to determine current swparams for playback: %s\n", snd_strerror (err));
      return err;
    }

    /* start the transfer when the buffer is almost full: (buffer_size / avail_min) * avail_min */
    err = snd_pcm_sw_params_set_start_threshold (handle, swparams, (buffer_size / period_size) * period_size);
    if (err < 0)
    {
      printf ("Unable to set start threshold mode for playback: %s\n", snd_strerror (err));
      return err;
    }

    /* allow the transfer when at least period_size samples can be processed */
    /* or disable this mechanism when period event is enabled (aka interrupt like style processing) */
    err = snd_pcm_sw_params_set_avail_min (handle, swparams, period_event ? buffer_size : period_size);
    if (err < 0)
    {
      printf ("Unable to set avail min for playback: %s\n", snd_strerror (err));
      return err;
    }

    /* enable period events when requested */
    if (period_event)
    {
      err = snd_pcm_sw_params_set_period_event (handle, swparams, 1);
      if (err < 0)
      {
        printf ("Unable to set period event: %s\n", snd_strerror (err));
        return err;
      }
    }

    /* write the parameters to the playback device */
    err = snd_pcm_sw_params (handle, swparams);
    if (err < 0)
    {
      printf ("Unable to set sw params for playback: %s\n", snd_strerror (err));
      return err;
    }

    return 0;
}
```

```
/* Underrun and suspend recovery */
static int xrun_recovery (snd_pcm_t *handle, int err)
{
    if (verbose)
      printf("Stream recovery\n");

    if (err == -EPIPE)                                        /* under-run */
    {
      err = snd_pcm_prepare (handle);
      if (err < 0)
        printf ("Can't recovery from underrun, prepare failed: %s\n", snd_strerror (err));
      return 0;
    }
    else if (err == -ESTRPIPE)
    {
      while ((err = snd_pcm_resume (handle)) == -EAGAIN)
      sleep (1);                                              /* wait until the suspend flag is released */
      if (err < 0)
      {
        err = snd_pcm_prepare (handle);
        if (err < 0)
          printf ("Can't recovery from suspend, prepare failed: %s\n", snd_strerror (err));
      }
      return 0;
    }

    return err;
}


/* Transfer method - write only */
static int write_loop (snd_pcm_t *handle, signed short *samples, snd_pcm_channel_area_t *areas)
{
    double phase = 0;
    signed short *ptr;
    int err, cptr;

    while (1)
    {
      generate_sine (areas, 0, period_size, &phase);
      ptr = samples;
      cptr = period_size;
      while (cptr > 0)
      {
        err = snd_pcm_writei (handle, ptr, cptr);
        if (err == -EAGAIN)
          continue;
        if (err < 0)
        {
          if (xrun_recovery (handle, err) < 0)
          {
            printf ("Write error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
          }
```

```
                break;                                      /* skip one period */
            }
            ptr += err * channels;
            cptr -= err;
        }
    }
}


/* Transfer method - write and wait for room in buffer using poll */
static int wait_for_poll (snd_pcm_t *handle, struct pollfd *ufds, unsigned int count)
{
    unsigned short revents;

    while (1)
    {
        poll (ufds, count, -1);
        snd_pcm_poll_descriptors_revents (handle, ufds, count, &revents);
        if (revents & POLLERR)
            return -EIO;
        if (revents & POLLOUT)
            return 0;
    }
}


static int write_and_poll_loop (snd_pcm_t *handle, signed short *samples, snd_pcm_channel_area_t *areas)
{
    struct pollfd *ufds;
    double phase = 0;
    signed short *ptr;
    int err, count, cptr, init;

    count = snd_pcm_poll_descriptors_count (handle);
    if (count <= 0)
    {
        printf ("Invalid poll descriptors count\n");
        return count;
    }

    ufds = malloc (sizeof (struct pollfd) * count);
    if (ufds == NULL)
    {
        printf ("Not enough memory\n");
        return -ENOMEM;
    }

    if ((err = snd_pcm_poll_descriptors (handle, ufds, count)) < 0)
    {
        printf ("Unable to obtain poll descriptors for playback: %s\n", snd_strerror (err));
        return err;
    }
```

```
init = 1;
while (1)
{
   if (! init)
   {
      err = wait_for_poll (handle, ufds, count);
      if (err < 0)
      {
         if (snd_pcm_state (handle) == SND_PCM_STATE_XRUN ||
                                     snd_pcm_state (handle) == SND_PCM_STATE_SUSPENDED)
         {
            err = snd_pcm_state (handle) == SND_PCM_STATE_XRUN ? -EPIPE : -ESTRPIPE;
            if (xrun_recovery (handle, err) < 0)
            {
                printf ("Write error: %s\n", snd_strerror (err));
                exit (EXIT_FAILURE);
            }
            init = 1;
         }
         else
         {
            printf ("Wait for poll failed\n");
            return err;
         }
      }
   }

   generate_sine (areas, 0, period_size, &phase);
   ptr = samples;
   cptr = period_size;
   while (cptr > 0)
   {
      err = snd_pcm_writei (handle, ptr, cptr);
      if (err < 0)
      {
         if (xrun_recovery (handle, err) < 0)
         {
            printf ("Write error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
         }

         init = 1;
         break;                                                /* skip one period */
      }

      if (snd_pcm_state (handle) == SND_PCM_STATE_RUNNING)
         init = 0;
      ptr += err * channels;
      cptr -= err;
      if (cptr == 0)
         break;
```

```
            /* it is possible that the initial buffer cannot store all data from the last period - so wait awhile */
            err = wait_for_poll (handle, ufds, count);
            if (err < 0)
            {
              if (snd_pcm_state (handle) == SND_PCM_STATE_XRUN ||
                                                snd_pcm_state (handle) == SND_PCM_STATE_SUSPENDED)
              {
                err = snd_pcm_state (handle) == SND_PCM_STATE_XRUN ? -EPIPE : -ESTRPIPE;
                if (xrun_recovery (handle, err) < 0)
                {
                  printf ("Write error: %s\n", snd_strerror (err));
                  exit (EXIT_FAILURE);
                }
                init = 1;
              }
              else
              {
                printf ("Wait for poll failed\n");
                return err;
              }
            }
          }
        }
}

/* Transfer method - asynchronous notification */
static void async_callback (snd_async_handler_t *ahandler)
{
    snd_pcm_t *handle = snd_async_handler_get_pcm (ahandler);
    struct async_private_data *data = snd_async_handler_get_callback_private (ahandler);
    signed short *samples = data->samples;
    snd_pcm_channel_area_t *areas = data->areas;
    snd_pcm_sframes_t avail;
    int err;

    avail = snd_pcm_avail_update (handle);
    while (avail >= period_size)
    {
      generate_sine (areas, 0, period_size, &data->phase);
      err = snd_pcm_writei (handle, samples, period_size);
      if (err < 0)
      {
        printf ("Write error: %s\n", snd_strerror (err));
        exit (EXIT_FAILURE);
      }
      if (err != period_size)
      {
        printf ("Write error: written %i expected %li\n", err, period_size);
        exit (EXIT_FAILURE);
      }
      avail = snd_pcm_avail_update (handle);
    }
}
```

```
static int async_loop (snd_pcm_t *handle, signed short *samples, snd_pcm_channel_area_t *areas)
{
    struct async_private_data data;
    snd_async_handler_t *ahandler;
    int err, count;

    data.samples = samples;
    data.areas = areas;
    data.phase = 0;
    err = snd_async_add_pcm_handler (&ahandler, handle, async_callback, &data);
    if (err < 0)
    {
      printf ("Unable to register async handler\n");
      exit (EXIT_FAILURE);
    }
    for (count = 0; count < 2; count++)
    {
      generate_sine (areas, 0, period_size, &data.phase);
      err = snd_pcm_writei (handle, samples, period_size);
      if (err < 0)
      {
        printf ("Initial write error: %s\n", snd_strerror (err));
        exit (EXIT_FAILURE);
      }
      if (err != period_size)
      {
        printf ("Initial write error: written %i expected %li\n", err, period_size);
        exit (EXIT_FAILURE);
      }
    }
    if (snd_pcm_state (handle) == SND_PCM_STATE_PREPARED)
    {
      err = snd_pcm_start (handle);
      if (err < 0)
      {
        printf ("Start error: %s\n", snd_strerror (err));
        exit (EXIT_FAILURE);
      }
    }
    /* because all other work is done in the signal handler suspend the process */
    while (1)
    {
      sleep (1);
    }
}
```

```
/* Transfer method - asynchronous notification + direct write */
static void async_direct_callback (snd_async_handler_t *ahandler)
{
    snd_pcm_t *handle = snd_async_handler_get_pcm (ahandler);
    struct async_private_data *data = snd_async_handler_get_callback_private (ahandler);
    const snd_pcm_channel_area_t *my_areas;
    snd_pcm_uframes_t offset, frames, size;
    snd_pcm_sframes_t avail, commitres;
    snd_pcm_state_t state;
    int first = 0, err;

    while (1)
    {
      state = snd_pcm_state (handle);
      if (state == SND_PCM_STATE_XRUN)
      {
        err = xrun_recovery (handle, -EPIPE);
        if (err < 0)
        {
          printf ("XRUN recovery failed: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
      }
      else if (state == SND_PCM_STATE_SUSPENDED)
      {
        err = xrun_recovery (handle, -ESTRPIPE);
        if (err < 0)
        {
          printf ("SUSPEND recovery failed: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
      }

      avail = snd_pcm_avail_update (handle);
      if (avail < 0)
      {
        err = xrun_recovery (handle, avail);
        if (err < 0)
        {
          printf ("avail update failed: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
        continue;
      }

      if (avail < period_size)
      {
        if (first)
        {
```

```
        first = 0;
        err = snd_pcm_start (handle);
        if (err < 0)
        {
          printf ("Start error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
      }
      else
        break;
      continue;
    }

    size = period_size;
    while (size > 0)
    {
      frames = size;
      err = snd_pcm_mmap_begin (handle, &my_areas, &offset, &frames);
      if (err < 0)
      {
        if ((err = xrun_recovery (handle, err)) < 0)
        {
          printf ("MMAP begin avail error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
      }

      generate_sine (my_areas, offset, frames, &data->phase);
      commitres = snd_pcm_mmap_commit (handle, offset, frames);
      if (commitres < 0 || (snd_pcm_uframes_t) commitres != frames)
      {
        if ((err = xrun_recovery (handle, commitres >= 0 ? -EPIPE : commitres)) < 0)
        {
          printf ("MMAP commit error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
      }
      size -= frames;
    }
  }
}


static int async_direct_loop (snd_pcm_t *handle, signed short *samples ATTRIBUTE_UNUSED,
                                        snd_pcm_channel_area_t *areas ATTRIBUTE_UNUSED)
{
  struct async_private_data data;
  snd_async_handler_t *ahandler;
  const snd_pcm_channel_area_t *my_areas;
  snd_pcm_uframes_t offset, frames, size;
  snd_pcm_sframes_t commitres;
  int err, count;
```

```
    data.samples = NULL;                    /* we do not require the global sample area for direct write */
    data.areas = NULL;                      /* we do not require the global areas for direct write */
    data.phase = 0;
    err = snd_async_add_pcm_handler (&ahandler, handle, async_direct_callback, &data);
    if (err < 0)
    {
      printf ("Unable to register async handler\n");
      exit (EXIT_FAILURE);
    }

    for (count = 0; count < 2; count++)
    {
      size = period_size;
      while (size > 0)
      {
        frames = size;
        err = snd_pcm_mmap_begin (handle, &my_areas, &offset, &frames);
        if (err < 0)
        {
          if ((err = xrun_recovery (handle, err)) < 0)
          {
            printf ("MMAP begin avail error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
          }
        }

        generate_sine (my_areas, offset, frames, &data.phase);
        commitres = snd_pcm_mmap_commit (handle, offset, frames);
        if (commitres < 0 || (snd_pcm_uframes_t) commitres != frames)
        {
          if ((err = xrun_recovery (handle, commitres >= 0 ? -EPIPE : commitres)) < 0)
          {
            printf ("MMAP commit error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
          }
        }
        size -= frames;
      }
    }

    err = snd_pcm_start (handle);
    if (err < 0)
    {
      printf ("Start error: %s\n", snd_strerror (err));
      exit (EXIT_FAILURE);
    }

    /* because all other work is done in the signal handler, suspend the process */
    while (1)
    {
      sleep(1);
    }
}
```

```
/* Transfer method - direct write only */
static int direct_loop (snd_pcm_t *handle, signed short *samples ATTRIBUTE_UNUSED,
                                    snd_pcm_channel_area_t *areas ATTRIBUTE_UNUSED)
{
    double phase = 0;
    const snd_pcm_channel_area_t *my_areas;
    snd_pcm_uframes_t offset, frames, size;
    snd_pcm_sframes_t avail, commitres;
    snd_pcm_state_t state;
    int err, first = 1;

    while (1)
    {
      state = snd_pcm_state (handle);
      if (state == SND_PCM_STATE_XRUN)
      {
        err = xrun_recovery (handle, -EPIPE);
        if (err < 0)
        {
          printf ("XRUN recovery failed: %s\n", snd_strerror (err));
          return err;
        }
        first = 1;
      }
      else if (state == SND_PCM_STATE_SUSPENDED)
      {
        err = xrun_recovery (handle, -ESTRPIPE);
        if (err < 0)
        {
          printf ("SUSPEND recovery failed: %s\n", snd_strerror (err));
          return err;
        }
      }

      avail = snd_pcm_avail_update (handle);
      if (avail < 0)
      {
        err = xrun_recovery (handle, avail);
        if (err < 0)
        {
          printf ("avail update failed: %s\n", snd_strerror (err));
          return err;
        }
        first = 1;
        continue;
      }

      if (avail < period_size)
      {
        if (first)
        {
          first = 0;
          err = snd_pcm_start (handle);
        if (err < 0)
        {
          printf ("Start error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
      }
      else
      {
        err = snd_pcm_wait (handle, -1);
        if (err < 0)
        {
          if ((err = xrun_recovery (handle, err)) < 0)
          {
            printf ("snd_pcm_wait error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
          }
          first = 1;
        }
      }
      continue;
    }

    size = period_size;
    while (size > 0)
    {
      frames = size;
      err = snd_pcm_mmap_begin (handle, &my_areas, &offset, &frames);
      if (err < 0)
      {
        if ((err = xrun_recovery (handle, err)) < 0)
        {
          printf ("MMAP begin avail error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
      }

      generate_sine (my_areas, offset, frames, &phase);
      commitres = snd_pcm_mmap_commit (handle, offset, frames);
      if (commitres < 0 || (snd_pcm_uframes_t) commitres != frames)
      {
        if ((err = xrun_recovery (handle, commitres >= 0 ? -EPIPE : commitres)) < 0)
        {
          printf ("MMAP commit error: %s\n", snd_strerror (err));
          exit (EXIT_FAILURE);
        }
        first = 1;
      }
      size -= frames;
    }
  }
}
```

```
/* Transfer method - direct write only using mmap_write functions */
static int direct_write_loop (snd_pcm_t *handle, signed short *samples, snd_pcm_channel_area_t *areas)
{
    double phase = 0;
    signed short *ptr;
    int err, cptr;

    while (1)
    {
      generate_sine (areas, 0, period_size, &phase);
      ptr = samples;
      cptr = period_size;
      while (cptr > 0)
      {
        err = snd_pcm_mmap_writei (handle, ptr, cptr);
        if (err == -EAGAIN)
          continue;
        if (err < 0)
        {
          if (xrun_recovery (handle, err) < 0)
          {
            printf ("Write error: %s\n", snd_strerror (err));
            exit (EXIT_FAILURE);
          }
          break;                                         /* skip one period */
        }
        ptr += err * channels;
        cptr -= err;
      }
    }
}



static struct transfer_method transfer_methods[] =
{
    { "write", SND_PCM_ACCESS_RW_INTERLEAVED, write_loop },
    { "write_and_poll", SND_PCM_ACCESS_RW_INTERLEAVED, write_and_poll_loop },
    { "async", SND_PCM_ACCESS_RW_INTERLEAVED, async_loop },
    { "async_direct", SND_PCM_ACCESS_MMAP_INTERLEAVED, async_direct_loop },
    { "direct_interleaved", SND_PCM_ACCESS_MMAP_INTERLEAVED, direct_loop },
    { "direct_noninterleaved", SND_PCM_ACCESS_MMAP_NONINTERLEAVED, direct_loop },
    { "direct_write", SND_PCM_ACCESS_MMAP_INTERLEAVED, direct_write_loop },
    { NULL, SND_PCM_ACCESS_RW_INTERLEAVED, NULL }
};


static void help (void)
{
    int k;
    printf ("Usage: pcm [OPTION]... [FILE]...\n"
      "-h,--help      help\n"
      "-D,--device    playback device\n"
      "-r,--rate      stream rate in Hz\n"
      "-c,--channels  count of channels in stream\n"
      "-f,--frequency sine wave frequency in Hz\n"
      "-b,--buffer    ring buffer size in us\n"
      "-p,--period    period size in us\n"
      "-m,--method    transfer method\n"
      "-o,--format    sample format\n"
      "-v,--verbose   show the PCM setup parameters\n"
      "-n,--noresample  do not resample\n"
      "-e,--pevent    enable poll event after each period\n\n");

    printf ("Recognized sample formats are:");
    for (k = 0; k < SND_PCM_FORMAT_LAST; ++k)
    {
      const char *s = snd_pcm_format_name (k);
      if (s)
        printf(" %s", s);
    }

    printf ("\n");
    printf ("Recognized transfer methods are:");
    for (k = 0; transfer_methods[k].name; k++)
      printf (" %s", transfer_methods[k].name);
    printf ("\n");
}


int main (int argc, char *argv[])
{
    struct option long_option[] = {
      {"help", 0, NULL, 'h'},
      {"device", 1, NULL, 'D'},
      {"rate", 1, NULL, 'r'},
      {"channels", 1, NULL, 'c'},
      {"frequency", 1, NULL, 'f'},
      {"buffer", 1, NULL, 'b'},
      {"period", 1, NULL, 'p'},
      {"method", 1, NULL, 'm'},
      {"format", 1, NULL, 'o'},
      {"verbose", 1, NULL, 'v'},
      {"noresample", 1, NULL, 'n'},
      {"pevent", 1, NULL, 'e'},
      {NULL, 0, NULL, 0}  };
    snd_pcm_t *handle;
    int err, morehelp;
    snd_pcm_hw_params_t *hwparams;
    snd_pcm_sw_params_t *swparams;
    int method = 0;
    signed short *samples;
    unsigned int chn;
    snd_pcm_channel_area_t *areas;
```

```
snd_pcm_hw_params_alloca (&hwparams);
snd_pcm_sw_params_alloca (&swparams);
morehelp = 0;

while (1)
{
  int c;

  if ((c = getopt_long (argc, argv, "hD:r:c:f:b:p:m:o:vne", long_option, NULL)) < 0)
    break;
  switch (c)
  {
    case 'h':
        morehelp++;
        break;
    case 'D':
        device = strdup (optarg);
        break;
    case 'r':
        rate = atoi (optarg);
        rate = rate < 4000 ? 4000 : rate;
        rate = rate > 196000 ? 196000 : rate;
        break;
    case 'c':
        channels = atoi (optarg);
        channels = channels < 1 ? 1 : channels;
        channels = channels > 2 ? 2 : channels;
        break;
    case 'f':
        freq = atoi (optarg);
        freq = freq < 50 ? 50 : freq;
        freq = freq > 22000 ? 22000 : freq;
        break;
    case 'b':
        buffer_time = atoi (optarg);
        buffer_time = buffer_time < 1000 ? 1000 : buffer_time;
        buffer_time = buffer_time > 1000000 ? 1000000 : buffer_time;
        break;
    case 'p':
        period_time = atoi (optarg);
        period_time = period_time < 1000 ? 1000 : period_time;
        period_time = period_time > 1000000 ? 1000000 : period_time;
        break;
    case 'm':
        for (method = 0; transfer_methods[method].name; method++)
          if (! strcasecmp(transfer_methods[method].name, optarg))
            break;
        if (transfer_methods[method].name == NULL)
          method = 0;
        break;
    case 'o':
        for (format = 0; format < SND_PCM_FORMAT_LAST; format++)
        {
          const char *format_name = snd_pcm_format_name (format);
          if (format_name)
            if (! strcasecmp (format_name, optarg))
              break;
        }
        if (format == SND_PCM_FORMAT_LAST)
          format = SND_PCM_FORMAT_S16;
        if (! snd_pcm_format_linear (format) && ! (format == SND_PCM_FORMAT_FLOAT_LE ||
                                                   format == SND_PCM_FORMAT_FLOAT_BE))
        {
          printf ("Invalid (non-linear/float) format %s\n", optarg);
          return 1;
        }
        break;
    case 'v':
        verbose = 1;
        break;
    case 'n':
        resample = 0;
        break;
    case 'e':
        period_event = 1;
        break;
  }
}

if (morehelp)
{
  help ();
  return 0;
}

err = snd_output_stdio_attach (&output, stdout, 0);
if (err < 0)
{
  printf ("Output failed: %s\n", snd_strerror (err));
  return 0;
}

printf ("Playback device is %s\n", device);
printf ("Stream parameters are %iHz, %s, %i channels\n", rate, snd_pcm_format_name (format), channels);
printf ("Sine wave rate is %.4fHz\n", freq);
printf ("Using transfer method: %s\n", transfer_methods[method].name);

if ((err = snd_pcm_open (&handle, device, SND_PCM_STREAM_PLAYBACK, 0)) < 0)
{
  printf ("Playback open error: %s\n", snd_strerror (err));
  return 0;
}
```

```
    if ((err = set_hwparams (handle, hwparams, transfer_methods[method].access)) < 0)
    {
      printf ("Setting of hwparams failed: %s\n", snd_strerror (err));
      exit (EXIT_FAILURE);
    }
    if ((err = set_swparams (handle, swparams)) < 0)
    {
      printf ("Setting of swparams failed: %s\n", snd_strerror (err));
      exit (EXIT_FAILURE);
    }
    if (verbose > 0)
      snd_pcm_dump (handle, output);
    samples = malloc ((period_size * channels * snd_pcm_format_physical_width (format)) / 8);
    if (samples == NULL)
    {
      printf ("Not enough memory\n");
      exit (EXIT_FAILURE);
    }
    areas = calloc (channels, sizeof (snd_pcm_channel_area_t));
    if (areas == NULL)
    {
      printf ("No enough memory\n");
      exit (EXIT_FAILURE);
    }
    for (chn = 0; chn < channels; chn++)
    {
      areas[chn].addr = samples;
      areas[chn].first = chn * snd_pcm_format_physical_width (format);
      areas[chn].step = channels * snd_pcm_format_physical_width (format);
    }
    err = transfer_methods[method].transfer_loop (handle, samples, areas);
    if (err < 0)
      printf ("Transfer failed: %s\n", snd_strerror (err));
    free (areas);
    free (samples);
    snd_pcm_close (handle);
    return 0;
}
```

The most important PCM ALSA interfaces use the identifiers plughw and hw. For the plughw identifier you need no information about the sound hardware - data will be automatically converted. In contrast you have to check whether the hw interface supports your hardware configuration. The two statements after the colon represent the card and the device number which can be reliably determined by

```
aplay  -l
```

The following sourcecode introduces a simple program for recording (capturing) and stores the data in a WAV-file.

```
/****************************************************************
 * Capture demo program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
****************************************************************/

#include <stdio.h>
#include <stdint.h>
#include <alsa/asoundlib.h>

#define FILENAME        "test.wav"
#define FRAMES          0x8000
#define MONO            1
#define STEREO          2
#define BUFSIZE         FRAMES * MONO
#define FMT_PCM         1
#define RATE_44kHz      44100
#define RATE_48kHz      48000
#define SAMPLEBITS      16

static char *device = "plughw:0,0";
static short buf[BUFSIZE];

typedef struct {
    char chunkId[4];
    uint32_t chunkSize;
    char format[4];
    char subChunkId[4];
    uint32_t subChunkSize;
    uint16_t audioFormat;
    uint16_t numChannels;
    uint32_t sampleRate;
    uint32_t byteRate;
    uint16_t blockAlign;
    uint16_t bitsPerSample;
    char subChunkId2[4];
    uint32_t subChunkSize2;
} wav_header;
```

```
static void set_wav_header (wav_header *w)
{
    strcpy (w->chunkId, "RIFF");
    w->chunkSize = BUFSIZE + sizeof (wav_header) - 8;
    strcpy (w->format, "WAVE");
    strcpy (w->subChunkId, "fmt ");
    w->subChunkSize = 16;                              /* Length of sub-header - 8 */
    w->audioFormat = FMT_PCM;                          /* PCM, uncompressed */
    w->numChannels = MONO;
    w->sampleRate = RATE_44kHz;
    w->byteRate = RATE_44kHz * (SAMPLEBITS / 8) * w->numChannels;       /* only valid for 8, 16 and 24 bits */
    w->blockAlign = (SAMPLEBITS / 8) * w->numChannels;                  /* only valid for 8, 16 and 24 bits */
    w->bitsPerSample = SAMPLEBITS;
    strcpy (w->subChunkId2, "data");
    w->subChunkSize2 = BUFSIZE;
}


static int write_wav_file (wav_header w)
{
    FILE *wavfile;

    wavfile = fopen (FILENAME, "w");
    if (! wavfile)
      return EXIT_FAILURE;

    if (fwrite (&w, sizeof (wav_header), 1, wavfile) != 1)
      return EXIT_FAILURE;

    if (fwrite (&buf, sizeof (short), FRAMES, wavfile) != FRAMES)
      return EXIT_FAILURE;

    fclose(wavfile);
    return EXIT_SUCCESS;
}


int main (void)
{
    int err;
    unsigned int rrate;
    wav_header wavhdr;
    snd_pcm_t *capture_handle;
    snd_pcm_hw_params_t *hw_params;

    if ((err = snd_pcm_open (&capture_handle, device, SND_PCM_STREAM_CAPTURE, 0)) < 0)
    {
      fprintf (stderr, "Cannot open audio device %s (%s)\n", device, snd_strerror (err));
      return EXIT_FAILURE;
    }

    if ((err = snd_pcm_hw_params_malloc (&hw_params)) < 0)
    {
      fprintf (stderr, "Cannot allocate hardware parameter structure (%s)\n", snd_strerror (err));
      return EXIT_FAILURE;
    }
```

```
if ((err = snd_pcm_hw_params_any (capture_handle, hw_params)) < 0)
{
  fprintf (stderr, "Cannot initialize hardware parameter structure (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

if ((err = snd_pcm_hw_params_set_access (capture_handle, hw_params,
                                                   SND_PCM_ACCESS_RW_INTERLEAVED)) < 0)
{
  fprintf (stderr, "Cannot set access type (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

if ((err = snd_pcm_hw_params_set_format (capture_handle, hw_params, SND_PCM_FORMAT_S16_LE)) < 0)
{
  fprintf (stderr, "Cannot set sample format (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

rrate = RATE_44kHz;
if ((err = snd_pcm_hw_params_set_rate_near (capture_handle, hw_params, &rrate, 0)) < 0)
{
  fprintf (stderr, "Cannot set sample rate (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

if ((err = snd_pcm_hw_params_set_channels (capture_handle, hw_params, MONO)) < 0)
{
  fprintf (stderr, "Cannot set channel count (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

if ((err = snd_pcm_hw_params (capture_handle, hw_params)) < 0)
{
  fprintf (stderr, "Cannot set parameters (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

snd_pcm_hw_params_free (hw_params);

if ((err = snd_pcm_prepare (capture_handle)) < 0)
{
  fprintf (stderr, "Cannot prepare audio interface for use (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

if ((err = snd_pcm_readi (capture_handle, buf, BUFSIZE)) != BUFSIZE)
{
  fprintf (stderr, "read from audio interface failed (%s)\n", snd_strerror (err));
  return EXIT_FAILURE;
}

snd_pcm_close (capture_handle);
```

```
    set_wav_header (&wavhdr);
    if (write_wav_file (wavhdr))
    {
      fprintf (stderr, "Cannot write wav file\n");
      return EXIT_FAILURE;
    }

    return EXIT_SUCCESS;
}
```

<u>Note:</u> The audio driver does not support the full duplex mode.
If you complement the previous example with few lines of code for full duplex mode (red color) the program generates an error message:

```
int main (void)
{
    int err;
    unsigned int rrate;
    wav_header wavhdr;
    snd_pcm_t *capture_handle;
    snd_pcm_t *playback_handle;
    snd_pcm_hw_params_t *hw_params;

    if ((err = snd_pcm_open (&playback_handle, device, SND_PCM_STREAM_PLAYBACK, 0)) < 0)
    {
      fprintf (stderr, "Cannot open audio device %s (%s)\n", device, snd_strerror (err));
      return EXIT_FAILURE;
    }

    if ((err = snd_pcm_open (&capture_handle, device, SND_PCM_STREAM_CAPTURE, 0)) < 0)
    {
      fprintf (stderr, "Cannot open audio device %s (%s)\n", device, snd_strerror (err));
      return EXIT_FAILURE;
    }

    if ((err = snd_pcm_hw_params_malloc (&hw_params)) < 0)
    {
      fprintf (stderr, "Cannot allocate hardware parameter structure (%s)\n", snd_strerror (err));
      return EXIT_FAILURE;
    }
                    .

                    .
```

This is the reason why you cannot use the Latency example from http://www.alsa-project.org/alsa-doc/alsa-lib/examples.html.

The sourcecode below shows an example for mixer programming. However you cannot use the well-known identifiers for the card name - only default and hw:0 (without device number) are possible.

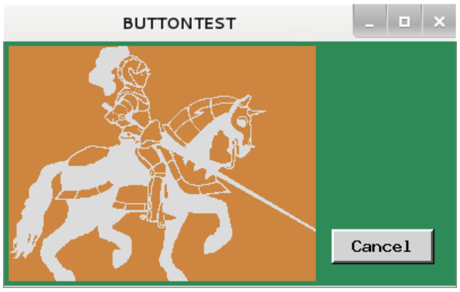
```
/**************************************************************
 * Mixer demo program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
***************************************************************/


#include <stdio.h>
#include <alsa/asoundlib.h>

static const char *card = "default";                    /* do not use plughw:0,0 or hw:0,0 */
static const char *playb_name = "Line Out";

typedef struct {
    snd_mixer_t *handle;
    snd_mixer_elem_t *elem;
} mixer_dev;

typedef struct {
    long left;
    long right;
} playb_vol;


static int init_mixer (mixer_dev *dev)
{
    const char *elem_name;
    snd_mixer_elem_t *elem;

    if (snd_mixer_attach (dev->handle, card) < 0)
    {
      fprintf (stderr, "Cannot attach card device\n");
      return EXIT_FAILURE;
    }

    if (snd_mixer_selem_register (dev->handle, NULL, NULL) < 0)
    {
      fprintf (stderr, "Cannot register simple element class\n");
      return EXIT_FAILURE;
    }

    if (snd_mixer_load (dev->handle) < 0)
    {
      fprintf (stderr, "Cannot open an empty mixer\n");
      return EXIT_FAILURE;
    }

    elem = snd_mixer_first_elem (dev->handle);
    while (elem)
    {
      elem_name = snd_mixer_selem_get_name (elem);
      if (! strcasecmp (elem_name, playb_name))
      {
        dev->elem = elem;
        return EXIT_SUCCESS;
      }
      elem = snd_mixer_elem_next (elem);
    }

    fprintf (stderr, "Cannot find mixer element\n");
    return EXIT_FAILURE;
}


int get_playb_vol (mixer_dev *dev, playb_vol *vol)
{
    long val;

    if (snd_mixer_selem_get_playback_dB (dev->elem, SND_MIXER_SCHN_FRONT_LEFT, &val) < 0)
    {
      fprintf (stderr, "Cannot get playback volume\n");
      return EXIT_FAILURE;
    }
    vol->left = val / 100L;

    if (snd_mixer_selem_get_playback_dB (dev->elem, SND_MIXER_SCHN_FRONT_RIGHT, &val) < 0)
    {
      fprintf (stderr, "Cannot get playback volume\n");
      return EXIT_FAILURE;
    }
    vol->right = val / 100L;

    return EXIT_SUCCESS;
}


int set_playb_vol (mixer_dev *dev, playb_vol *vol)
{
    if (snd_mixer_selem_set_playback_dB (dev->elem, SND_MIXER_SCHN_FRONT_LEFT, vol->left * 100L, 0) < 0)
    {
      fprintf (stderr, "Cannot set playback volume\n");
      return EXIT_FAILURE;
    }

    if (snd_mixer_selem_set_playback_dB (dev->elem, SND_MIXER_SCHN_FRONT_RIGHT,
                                                                  vol->right * 100L, 0) < 0)

    {
      fprintf (stderr, "Cannot set playback volume\n");
      return EXIT_FAILURE;
    }

    return EXIT_SUCCESS;
}
```

```
int main (void)
{
    mixer_dev mixdev;
    playb_vol pbvol;
    snd_mixer_t *mixer_handle;

    if (snd_mixer_open (&mixer_handle, 0) < 0)
    {
      fprintf (stderr, "Cannot open mixer device\n");
      return EXIT_FAILURE;
    }
    mixdev.handle = mixer_handle;

    if (init_mixer (&mixdev))
      return EXIT_FAILURE;

    if (get_playb_vol (&mixdev, &pbvol))
      return EXIT_FAILURE;

    printf ("Volumen: Left = %ld dB   Right = %ld dB\n", pbvol.left, pbvol.right);

    pbvol.left -= 3;
    pbvol.right -= 7;

    if (set_playb_vol (&mixdev, &pbvol))
      return EXIT_FAILURE;

    if (get_playb_vol (&mixdev, &pbvol))
      return EXIT_FAILURE;

    printf ("Volumen: Left = %ld dB   Right = %ld dB\n", pbvol.left, pbvol.right);

    snd_mixer_close (mixer_handle);
    return EXIT_SUCCESS;
}
```

10.12 Graphic Programming

10.12.1 X11^{TM}

X11^{TM} realizes a client/server architecture and supports graphical screen output as well as keyboard and mouse input. The following example only gives a small overview about the X11^{TM} features. First you can see two buttons and after click on the centered button the program shows the screen below.



But there exist two restrictions: program execution via the serial remote interface is not possible and the program is only checked with an HDMI$^{®}$ monitor. For cross compilation the addition of the X11^{TM} lib within the Makefile may be necessary, e.g. LDFLAGS := -l X11. If so copy the library from the target hardware to a search path and if required create a new link or rename the file to libX11.so (see previous chapter).

```
/*****************************************************************
 * X11 demonstration program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
*****************************************************************/

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include "picture.xbm"

#define TRUE                1

#define LEFT                200
#define TOP                 200
#define WIDTH               400
#define HEIGHT              216
#define BORDER              4
#define BACKGND_COL         "seagreen"
#define HEADER              "BUTTONTEST"

#define BTN_WIDTH           90
#define BTN_HEIGHT          30
#define BTN_LINECOL         "gray32"
#define BTN_GRAYCOL         "lightgray"
#define BTN_BLUECOL         "cornflowerblue"
#define BTN_TEXTCOL         "navy"
```

```
#define BTN_CANCEL          1
#define BTN_FONT1           "9x15bold"
#define BTN_TEXT1           "Cancel"

#define BTN_PICTURE         2
#define BTN_FONT2           "lucidasans-bold-12"
#define BTN_TEXT2           "Picture"

#define PICT_X              4
#define PICT_BG             "peru"
#define PICT_FG             "gainsboro"

typedef struct {
    Display *disp;
    Window win;
    GC gc;
    int screen;
    Colormap cmap;          /* only for freeing */
    XFontStruct *font1;     /* only for freeing */
    XFontStruct *font2;     /* only for freeing */
    Pixmap pict;            /* only for freeing */
} XSystem;

typedef struct {
    XRectangle rc;
    unsigned long backcol;
    unsigned long linecol1;
    unsigned long linecol2;
    unsigned long textcol;
    XFontStruct *font;
    char text[16];
} XButton;

static XRectangle btn1 = {
    .x      = WIDTH - BTN_WIDTH - 20,
    .y      = HEIGHT - BTN_HEIGHT - 20,
    .width  = BTN_WIDTH,
    .height = BTN_HEIGHT
};

static XRectangle btn2 = {
    .x      = WIDTH / 2 - BTN_WIDTH / 2,
    .y      = HEIGHT / 2 - BTN_HEIGHT,
    .width  = BTN_WIDTH,
    .height = BTN_HEIGHT * 2
};


static void draw_button (XSystem sys, XButton btn)
{
    int tmp, xp, yp;
    XCharStruct tsize;

    XSetForeground (sys.disp, sys.gc, btn.backcol);
    XFillRectangle (sys.disp, sys.win, sys.gc, btn.rc.x+1, btn.rc.y+1, btn.rc.width-2, btn.rc.height-2);
    XSetForeground (sys.disp, sys.gc, BlackPixel (sys.disp, sys.screen));
    XDrawRectangle (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y, btn.rc.width, btn.rc.height);
    XSetForeground (sys.disp, sys.gc, btn.linecol1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y, btn.rc.x, btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y, btn.rc.x + btn.rc.width-1, btn.rc.y);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+1, btn.rc.y, btn.rc.x+1, btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y+1, btn.rc.x + btn.rc.width-1, btn.rc.y+1);

    XSetForeground (sys.disp, sys.gc, btn.linecol2);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+1, btn.rc.y + btn.rc.height-1, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x + btn.rc.width-1, btn.rc.y+1, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+2, btn.rc.y + btn.rc.height-2, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-2);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x + btn.rc.width-2, btn.rc.y+2, btn.rc.x + btn.rc.width-2,
                                                                        btn.rc.y + btn.rc.height-1);

    if (strlen (btn.text))
    {
      XSetFont (sys.disp, sys.gc, btn.font->fid);
      XSetForeground (sys.disp, sys.gc, btn.textcol);
      XSetBackground (sys.disp, sys.gc, btn.backcol);
      XTextExtents (btn.font, btn.text, strlen (btn.text), &tmp, &tmp, &tmp, &tsize);
      xp = (btn.rc.width / 2) - (tsize.width / 2) + btn.rc.x;
      yp = (btn.rc.height / 2) + (tsize.ascent) - ((tsize.ascent + tsize.descent) / 2) + btn.rc.y;
      XDrawImageString (sys.disp, sys.win, sys.gc, xp, yp, btn.text, strlen (btn.text));
    }
}


/* Not the optimal solution ! */
static void reverse_button (XSystem sys, XButton btn)
{
    XSetForeground (sys.disp, sys.gc, BlackPixel (sys.disp, sys.screen));
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y, btn.rc.x, btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y, btn.rc.x + btn.rc.width-1, btn.rc.y);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+1, btn.rc.y, btn.rc.x+1, btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x, btn.rc.y+1, btn.rc.x + btn.rc.width-1, btn.rc.y+1);

    XSetForeground (sys.disp, sys.gc, WhitePixel (sys.disp, sys.screen));
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+1, btn.rc.y + btn.rc.height-1, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x + btn.rc.width-1, btn.rc.y+1, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-1);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x+2, btn.rc.y + btn.rc.height-2, btn.rc.x + btn.rc.width-1,
                                                                        btn.rc.y + btn.rc.height-2);
    XDrawLine (sys.disp, sys.win, sys.gc, btn.rc.x + btn.rc.width-2, btn.rc.y+2, btn.rc.x + btn.rc.width-2,
                                                                        btn.rc.y + btn.rc.height-1);
}
```

```
static int check_button (XSystem sys, XButton btn, int x, int y, int suppr)
{
    if (x >= btn.rc.x && x <= btn.rc.x + btn.rc.width && y >= btn.rc.y && y <= btn.rc.y + btn.rc.height)
    {
      if (! suppr)
        reverse_button (sys, btn);
      return EXIT_SUCCESS;
    }
    return EXIT_FAILURE;
}


static void free_resources (XSystem sys)
{
    if (sys.pict)
      XFreePixmap (sys.disp, sys.pict);
    if (sys.font1 != (XFontStruct *) NULL)
      XFreeFont (sys.disp, sys.font1);
    if (sys.font2 != (XFontStruct *) NULL)
      XFreeFont (sys.disp, sys.font2);
    if (sys.gc)
      XFreeGC (sys.disp, sys.gc);
    if (sys.win)
      XDestroyWindow (sys.disp, sys.win);
    if (sys.cmap)
      XFreeColormap (sys.disp, sys.cmap);
    XCloseDisplay (sys.disp);
}


int main (void)
{
    Display *disp;
    XVisualInfo vinfo;
    Visual *visual;
    int depth, screen,
        quit = 0,
        curbtn = 0,
        suppress = 0;
    Colormap cmap;
    XSetWindowAttributes wa;
    Window win;
    GC gc;
    XColor col, backgnd;
    XColor backcol, linecol;
    XFontStruct *font1, *font2;
    XCharStruct tsize;
    XSizeHints sh;
    XTextProperty name;
    Atom closeatom;
```

```
XEvent e;
Pixmap pict;
XSystem xsys;
XButton xbtn;
char *txt = HEADER;

disp = XOpenDisplay (NULL);
if (disp == (Display *) NULL)
{
  fprintf (stderr, "Cannot connect the X server\n");
  return EXIT_FAILURE;
}

memset ((void *) &xsys, 0, sizeof (xsys));
memset ((void *) &xbtn, 0, sizeof (xbtn));
xsys.disp = disp;

if (!XMatchVisualInfo (disp, 0, 32, TrueColor, &vinfo))
  if (!XMatchVisualInfo (disp, 0, 24, TrueColor, &vinfo))
    if (!XMatchVisualInfo (disp, 0, 16, TrueColor, &vinfo))
    {
      fprintf (stderr, "Cannot get TrueColor Visual\n");
      free_resources (xsys);
      return EXIT_FAILURE;
    }

visual = vinfo.visual;
depth = vinfo.depth;
screen = vinfo.screen;
cmap = XCreateColormap (disp, RootWindow (disp, screen), visual, AllocNone);

wa.background_pixel = WhitePixel (disp, screen);
wa.border_pixel = BlackPixel (disp, screen);
wa.colormap = cmap;
win = XCreateWindow (disp, RootWindow (disp, screen), LEFT, TOP, WIDTH, HEIGHT, BORDER, depth,
                              InputOutput, visual, CWBorderPixel | CWBackPixel | CWColormap, &wa);

sh.min_width       = WIDTH;
sh.min_height      = HEIGHT;
sh.flags           = PMinSize;
XSetWMNormalHints (disp, win, &sh);
XStringListToTextProperty (&txt, 1, &name);
XSetWMName (disp, win, &name);

xsys.win           = win;
xsys.screen        = screen;
xsys.cmap          = cmap;

gc = XCreateGC (disp, win, 0, NULL);
if (! gc)
{
  fprintf (stderr, "Cannot create graphics context\n");
  free_resources (xsys);
  return EXIT_FAILURE;
}
```

```
xsys.gc = gc;
XSelectInput (disp, win, ButtonPressMask | ButtonReleaseMask | ExposureMask | StructureNotifyMask);

closeatom = XInternAtom (disp, "WM_DELETE_WINDOW", TRUE);
if (closeatom)
  XSetWMProtocols (disp, win, (Atom *) & closeatom, 1);

font1 = XLoadQueryFont (disp, BTN_FONT1);
if (font1 == (XFontStruct *) NULL)
{
  fprintf (stderr, "Cannot load text font\n");
  free_resources (xsys);
  return EXIT_FAILURE;
}
xsys.font1 = font1;

font2 = XLoadQueryFont (disp, BTN_FONT2);
if (font2 == (XFontStruct *) NULL)
{
  fprintf (stderr, "Cannot load text font\n");
  free_resources (xsys);
  return EXIT_FAILURE;
}
xsys.font2 = font2;

XMapWindow (disp, win);

while (! quit)
{
  XNextEvent (disp, &e);
  if (e.type == ClientMessage && (Atom) (e.xclient.data.l[0]) == closeatom)
    quit = TRUE;

  else if (e.type == ButtonPress)
  {
    xbtn.rc = btn1;
    if (! check_button (xsys, xbtn, e.xbutton.x, e.xbutton.y, 0))
      curbtn = BTN_CANCEL;

    xbtn.rc = btn2;
    if (! check_button (xsys, xbtn, e.xbutton.x, e.xbutton.y, suppress))
      curbtn = BTN_PICTURE;
  }

  else if (e.type == ButtonRelease)
  {
    if (curbtn)
    {
      switch (curbtn)
      {
        case BTN_CANCEL:
                 XAllocNamedColor (disp, cmap, BTN_LINECOL, &linecol, &linecol);
                 XAllocNamedColor (disp, cmap, BTN_GRAYCOL, &backcol, &backcol);
                 xbtn.rc        = btn1;
```

```
                    xbtn.backcol   = backcol.pixel;
                    xbtn.linecol1  = WhitePixel (disp, screen);
                    xbtn.linecol2  = linecol.pixel;
                    xbtn.textcol   = BlackPixel (disp, screen);
                    xbtn.font      = font1;
                    strcpy (xbtn.text, BTN_TEXT1);
                    draw_button (xsys, xbtn);
                    quit = TRUE;
                    break;

              case BTN_PICTURE:
                    XAllocNamedColor (disp, cmap, PICT_FG, &col, &col);
                    XSetForeground (disp, gc, col.pixel);
                    XAllocNamedColor (disp, cmap, PICT_BG, &col, &col);
                    XSetBackground (disp, gc, col.pixel);
                    pict = XCreateBitmapFromData (disp, win, xbm_bits, xbm_width, xbm_height);
                    XCopyPlane (disp, pict, win, gc, 0, 0, xbm_width, xbm_height, PICT_X,
                                                                       (HEIGHT - xbm_height) / 2, 1);
                    xsys.pict = pict;
                    suppress = TRUE;
                    break;
            }
            curbtn = 0;
          }
        }

        else if (e.type == ConfigureNotify)
        {
          if (e.xconfigure.width != WIDTH || e.xconfigure.height != HEIGHT)
            ; /* Do something */
        }

        else if (e.type == Expose)
        {
          if (e.xexpose.count == 0)
          {
            XClearWindow (disp, win);
            XAllocNamedColor (disp, cmap, BACKGND_COL, &backgnd, &backgnd);
            XSetForeground (disp, gc, backgnd.pixel);
            XFillRectangle (disp, win, gc, 0, 0, WIDTH, HEIGHT);

            XAllocNamedColor (disp, cmap, BTN_LINECOL, &linecol, &linecol);

            XAllocNamedColor (disp, cmap, BTN_GRAYCOL, &backcol, &backcol);
            xbtn.rc         = btn1;
            xbtn.backcol    = backcol.pixel;
            xbtn.linecol1   = WhitePixel (disp, screen);
            xbtn.linecol2   = linecol.pixel;
            xbtn.textcol    = BlackPixel (disp, screen);
            xbtn.font       = font1;
            strcpy (xbtn.text, BTN_TEXT1);
            draw_button (xsys, xbtn);
```

```
                XAllocNamedColor (disp, cmap, BTN_BLUECOL, &backcol, &backcol);
                XAllocNamedColor (disp, cmap, BTN_TEXTCOL, &col, &col);
                xbtn.rc             = btn2;
                xbtn.backcol        = backcol.pixel;
                xbtn.linecol1       = WhitePixel (disp, screen);
                xbtn.linecol2       = linecol.pixel;
                xbtn.textcol        = col.pixel;
                xbtn.font           = font2;
                strcpy (xbtn.text, BTN_TEXT2);
                draw_button (xsys, xbtn);
            }
        }
    }

    free_resources (xsys);
    return EXIT_SUCCESS;
}
```

The previous example requires an additional file named 'picture.xbm'.

```
#define xbm_width 272
#define xbm_height 208
static const unsigned char xbm_bits[] = {
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0xFE,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xFF,0xFF,0x03,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0xFF,0x03,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0xFC,0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0xFE,0xFF,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0xFE,0xFF,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0xFF,
  0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF,0x07,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF,0x7F,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF,0xA3,0x03,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x80,0xFF,0xFF,0xFF,0xA0,0x06,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,
0xFF,0xFF,0x3F,0xF0,0x09,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0xFF,
0x3F,0x1D,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0xFF,0xCF,0x03,
0x70,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0xFF,0x87,0x01,0xD0,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xE0,0xFF,0xFF,0xC7,0x00,0x10,0x01,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0xFF,0xFF,0x43,0x00,0x18,0x03,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,
0xFF,0xFF,0x63,0x00,0x08,0x02,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0xFF,
0x23,0x00,0x0C,0x06,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x3F,0x1F,0x1C,
0x04,0x06,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0x1F,0x11,0x3C,0x84,0x1F,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xFC,0xFF,0x0F,0x1F,0x64,0xE2,0x1F,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xFE,0xFF,0x07,0x11,0x44,0xF2,0x0C,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x07,0x11,0xC4,0x3E,0x04,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x0F,
0x11,0x88,0x07,0x3C,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x3F,0x13,0x18,
0x60,0x18,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x3F,0x1A,0x10,0x00,0x08,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x80,0xFF,0xFF,0x3F,0x3A,0x20,0x00,0x0C,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0xFF,0xFF,0x3F,0x3E,0x20,0x00,0x04,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,
0xFF,0x1F,0x68,0x40,0x00,0x06,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x0F,
0xF8,0xC0,0x00,0x02,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x1F,0xE0,0x83,
0x00,0x02,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0x3F,0xE0,0x1F,0x01,0x02,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xFC,0xFF,0x7F,0xC0,0x3F,0x03,0x03,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0xFF,0x7F,0x40,0x78,0x06,0x01,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xFF,0x7F,0xC0,0xC0,0xBF,0x01,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0x7F,
0xC0,0x03,0xFE,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x7F,0xE0,0x0F,
0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x7F,0x30,0x00,0x41,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF0,0x3F,0x18,0x00,0x68,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xF0,0x3F,0x8C,0xD7,0xC0,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xE0,0x1F,0x76,0xC0,0x83,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0x07,
0x8F,0x3F,0x9E,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0x03,0xC5,0xE0,
0x70,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x18,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x73,0x80,0xC1,0x03,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x0F,0x00,0x7C,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x39,0x00,0x07,0x07,0x00,0x00,
0x00,0x00,0x00,0x00,0xE0,0x03,0xF0,0x00,0x4F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x80,0x3D,0x00,0x0E,0x04,0x00,0x00,0x00,0x00,
0x00,0xE0,0x1F,0x00,0x00,0xC3,0x67,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x80,0xFC,0x00,0x3A,0x0C,0x00,0x00,0x00,0x00,0x00,0x3E,
0x08,0x00,0x00,0xFC,0x23,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,
0x9E,0x03,0x24,0x0C,0x00,0x00,0x00,0x00,0x80,0x03,0x0C,0x00,
0x00,0xFE,0x21,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x0F,0x06,
0xC4,0x18,0x00,0x00,0x00,0x00,0x60,0x00,0x0C,0x00,0x00,0x3F,
0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x70,0xC0,0x07,0x08,0x84,0x1F,
0x00,0x00,0x00,0x00,0x18,0x00,0x0C,0xF8,0xFF,0x1F,0x18,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF8,0xC0,0x0F,0x08,0x04,0x1E,0x00,0x00,
0x00,0x00,0x06,0x00,0x8C,0x07,0x80,0x2F,0x08,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFC,0xC1,0x30,0x10,0x04,0x10,0x00,0x00,0x00,0x00,
0x03,0x00,0x7C,0x00,0xC0,0x5F,0x0C,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xF8,0x67,0x40,0x20,0x06,0x10,0x00,0x00,0x00,0x80,0x00,0x80,
0x0F,0x00,0xE0,0xB7,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x2F,
0x80,0x20,0x06,0x10,0x00,0x00,0x00,0xF0,0x00,0xF8,0x0C,0x00,
0xE0,0xE3,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0x3F,0x00,0x20,
0x06,0x18,0x00,0x00,0x00,0xD8,0x00,0x0E,0x08,0x00,0xF0,0x03,
0x30,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x1F,0x00,0x61,0x07,0x0E,
0x00,0x00,0x00,0xC6,0x00,0x03,0x08,0x00,0xF0,0x01,0x78,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x0F,0x00,0xF0,0xFF,0x0B,0x00,0x00,
0x00,0xC1,0xC0,0x01,0x08,0x00,0xF8,0x00,0x88,0xFF,0x0F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x13,0x00,0xFA,0x03,0x18,0x00,0x00,0x80,0x80,
0xE0,0x1F,0x18,0x00,0xF8,0x00,0x08,0xE0,0x03,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0x51,0x00,0xBE,0x00,0x10,0x00,0x00,0x40,0x80,0xF0,0x0F,
0x18,0x80,0x7F,0x00,0x08,0x3C,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,
0x3D,0x3F,0x00,0x10,0x00,0x00,0x20,0x80,0x00,0x07,0x10,0xE0,
0x3F,0x00,0x98,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x60,0x00,0xC6,0x3F,
0x00,0x10,0x00,0x00,0x10,0x80,0xC1,0x01,0x20,0xF8,0x3F,0x00,
0xF0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x20,0x00,0x8A,0x1F,0x00,0x10,
0x00,0x00,0x08,0x00,0x71,0x00,0x20,0xFE,0x3F,0xF0,0x63,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x20,0x00,0xB2,0x1F,0x00,0x30,0x00,0x00,
0x04,0x00,0x1D,0x00,0xE0,0xFF,0x1F,0x08,0x44,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xC0,0x10,0xC4,0x3F,0x00,0x20,0x00,0x00,0x02,0x00,
0x07,0x00,0xC0,0xFF,0x1F,0x04,0x44,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0x11,0x3E,0xFF,0x00,0x20,0x0F,0x00,0x01,0x80,0x03,0x00,
0xE0,0xFF,0x1F,0xFC,0x49,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x93,
0x01,0xFE,0x01,0xF0,0x11,0x80,0x00,0x60,0x06,0x00,0xF0,0xFF,
0x1F,0xFC,0x6B,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x00,0xF8,
0x03,0xF0,0x20,0xC0,0x00,0x18,0x04,0x00,0xF8,0xFF,0x1F,0xFC,
0x2B,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0x01,0xF0,0x07,0x30,
0x20,0x60,0x00,0x0C,0x08,0x00,0xFE,0xFF,0x1F,0xFC,0x27,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x80,0x03,0xFC,0x0F,0x18,0x20,0x3F,
0x00,0x06,0x00,0x00,0xFF,0xFF,0x3F,0xF8,0x63,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x87,0x1F,0x1C,0x0C,0x20,0x75,0x00,0x01,
0x10,0x80,0xFF,0xFF,0x3F,0xF0,0x43,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFC,0x00,0x78,0x06,0x20,0x41,0x80,0x00,0x20,0xC0,
0xFF,0xFF,0x7F,0xC0,0x41,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x78,0x00,0xF0,0x03,0x60,0xC2,0x40,0x00,0x40,0xE0,0xFF,0xFF,
0xFF,0x00,0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x20,0x00,
0xC4,0x01,0x40,0x82,0x21,0x00,0x80,0xF0,0xFF,0xFF,0xFF,0x01,
0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x00,0x88,0x00,
0x40,0x04,0x11,0x00,0x00,0xF9,0xFF,0xF7,0xFF,0x03,0x40,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x10,0x00,0xF0,0x00,0x40,0x0C,
0x1A,0x00,0x00,0xFE,0xFF,0xC3,0xFF,0x0F,0x40,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF0,0x0F,0x40,0x00,0xC0,0x0F,0x0C,0x00,
0x00,0xFE,0xFF,0xC1,0xFF,0x3F,0x40,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xF8,0x39,0x60,0x00,0xC0,0x1F,0x08,0x00,0x00,0xFF,
0xFF,0xC1,0xFF,0x7F,0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xFC,0xF1,0x23,0x00,0x80,0x3F,0x18,0x00,0x80,0xFF,0xFF,0x81,
0xFF,0xFF,0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFB,
0x3F,0x00,0x80,0x7F,0x30,0x00,0xC0,0xFF,0x3F,0x00,0xFF,0xFF,
0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE6,0x9F,0x1F,0x00,
0x00,0xFF,0x60,0x00,0xE0,0xFF,0x3F,0x00,0xFC,0xFF,0x40,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x0E,0x00,0x1F,0x00,0x00,0xFF,
0xC1,0x00,0xF0,0xFF,0x1F,0x00,0xF8,0xFF,0x60,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xFC,0x7F,0x1F,0x00,0x80,0xFF,0x83,0x01,
0xF8,0xFF,0x1F,0x00,0xF8,0xFC,0x60,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFC,0x7F,0x1F,0xFE,0xC1,0xFF,0x07,0x02,0xFC,0xFF,
0x1F,0x00,0xF8,0x79,0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xF8,0xFF,0x11,0x83,0xFF,0xFF,0x0F,0x04,0xFE,0xFF,0x0F,0x00,
0x38,0x7B,0x70,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,
0x9C,0x01,0xC0,0xFF,0x3F,0x18,0xFF,0xFF,0x0F,0x00,0x38,0x7B,
0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0xFF,0x00,
0xFC,0xFF,0x7F,0xF0,0xFF,0xFF,0x0F,0x00,0x38,0x7B,0x50,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x80,0xFD,0xFF,0x1F,0x80,0xFF,0xFF,
0xFF,0xE0,0xFF,0xFF,0x0F,0x00,0xF8,0x7D,0xAE,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0x3F,0x00,0xE0,0xFF,0xFF,0x1F,0xE0,0x87,0xFF,0xFF,0xF3,
0xFF,0xFF,0x0F,0x00,0x78,0xF3,0xFF,0x03,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xE0,
0x3F,0xFE,0xFF,0xFF,0x3F,0x98,0x07,0xFF,0xFF,0xEF,0xFF,0xFF,
0x0F,0x00,0xF8,0xF1,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x18,0x00,0xF0,0xFF,
0xFF,0xFF,0xED,0x0F,0x07,0xFC,0xFF,0xFF,0xFF,0xFF,0x07,0x00,
0x7C,0xF3,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x0E,0x00,0x00,0xFE,0xFF,0xFF,
0x04,0x00,0x3E,0xF0,0xFF,0xFF,0xFF,0xFF,0x03,0x00,0x3E,0xF3,
0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x80,0x03,0x00,0x00,0xFE,0xFF,0xFF,0x04,0x00,
0x7C,0xC0,0xFF,0xFF,0xFF,0xFF,0x03,0x00,0x3E,0xF3,0x3F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0x00,0x00,0x00,0xFE,0xFF,0xFF,0x04,0x00,0x78,0xC0,
0xFE,0xFF,0xFF,0xFF,0x03,0x00,0x3C,0xF3,0x1F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x38,
0x00,0x00,0x00,0xFF,0xFF,0xFF,0x0C,0x00,0xF8,0xC0,0xF3,0xFF,
0xFF,0xFF,0x03,0x00,0x38,0xF3,0x0F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0C,0x00,0x00,
0x00,0xFF,0xFF,0xFF,0x08,0x00,0xF0,0x80,0xE1,0xFF,0xFF,0xFF,
0x03,0x00,0x30,0xF3,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x80,0xFF,
0xFF,0xFF,0x19,0x00,0xF0,0x01,0x87,0xFF,0xFF,0xFF,0x03,0x00,
0x20,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x80,0x01,0x00,0x00,0x80,0xFF,0xFF,0xFF,
0x33,0x00,0xE0,0x01,0x03,0xFE,0xFF,0xFF,0x03,0x00,0x20,0xFD,
0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xC0,0x00,0x00,0x00,0xF0,0xFF,0xFF,0xFF,0xE3,0x00,
0xE0,0x03,0x02,0xF8,0xFF,0xFF,0x03,0x00,0x20,0x71,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x60,0x00,0x00,0x00,0xF8,0xFF,0xFF,0xFF,0xC7,0x01,0xC0,0x03,
0x06,0xFC,0xFF,0xFF,0x03,0x00,0x20,0x01,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x00,
0x00,0x00,0xFE,0xFF,0xFF,0xFF,0x87,0x03,0xC0,0x07,0x05,0xFC,
0xFF,0xFF,0x03,0x00,0x20,0x01,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x3C,0x00,0x00,0x80,
0xFF,0xFF,0xFF,0xFF,0x0F,0x06,0xC0,0xC7,0x0F,0x78,0xFE,0xFF,
0x1F,0x00,0xA0,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xF0,0x1F,0x00,0x00,0xE0,0xFF,0xFF,
0xFF,0xFF,0x1F,0x0C,0xC0,0x6F,0x0C,0x38,0xFC,0xFF,0xFF,0x00,
0xB0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xFE,0x1F,0x00,0x00,0xF0,0xFF,0xFF,0xFF,0xFF,
0x3F,0x18,0x70,0x3E,0x08,0x38,0xF0,0xFF,0xFF,0x07,0x9E,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xE0,0xFF,0x1F,0x00,0x80,0xFF,0xFF,0xFF,0xFF,0xFF,0x7F,0x30,
0x1C,0x1C,0x18,0x10,0xE0,0xFF,0xFF,0xFF,0x9F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,
0x1F,0x00,0xE0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x60,0x06,0x18,
0x10,0x10,0xE0,0xFF,0xFF,0xFF,0x9F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xC1,0x03,0x38,0x30,0x00,
0xF0,0xFF,0xFF,0xFF,0xDF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xC3,0x01,0x38,0x20,0x00,0xF0,0xFF,
0xFF,0x8F,0x69,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xC0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xE7,0x00,0x78,0x60,0x00,0xF0,0xE3,0xFF,0x07,
0x3E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xBF,0x01,0x78,0x40,0x00,0xF0,0x80,0xFF,0x03,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x1F,
0x01,0x78,0x40,0x00,0x10,0x80,0xFF,0x01,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0xBF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x5F,0x02,0xF8,
0xC0,0x00,0x00,0x80,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x9F,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x5F,0xFE,0xFF,0x80,0x00,
0x00,0x80,0xFF,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0xC7,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0x5F,0x02,0xF8,0x01,0x01,0x00,0xC0,
0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xF8,0xFF,0xC3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0x3F,0x03,0xF8,0x01,0x01,0x00,0xC0,0xCF,0xFF,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xF8,0xFF,0xC1,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0x7F,0xFC,0x03,0x03,0x00,0xC0,0x83,0xFE,0x03,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,
0xFF,0xC0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xC1,0xFF,0x03,0x07,0x00,0xC0,0x81,0xF8,0x07,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0x7F,0xC0,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x3F,0x81,0xFF,
0xC7,0x05,0x00,0xC0,0x80,0xE0,0x1F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x7F,0xC0,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x3F,0x01,0xFF,0x6F,0x0C,
0x00,0x00,0x80,0x81,0x7F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xFE,0x3F,0xC0,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFE,0xFF,0xFF,0xFF,0x1F,0x01,0xFF,0x3F,0x18,0x00,0x00,
0x80,0x03,0xFE,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xFF,0x3F,0xC0,0xFF,0xFF,0xFF,0xFF,0x7F,0xFC,
0xFF,0xFF,0xFF,0x9F,0x00,0xFF,0x1F,0x70,0x00,0x00,0xE0,0x07,
0xFC,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xC0,0xFF,0x3F,0xC0,0xFF,0xFF,0xFF,0xFF,0x3F,0xF8,0xFF,0xFF,
0xFF,0x8F,0x80,0xFF,0x3F,0xC0,0x00,0x00,0x3E,0x1C,0xF0,0x0F,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,
0x3F,0x80,0xFF,0xFF,0xFF,0xFF,0x0F,0xE0,0xFF,0xFF,0xFF,0x8F,
0x80,0xF7,0x3F,0x00,0x1F,0xF8,0x03,0x30,0xC0,0x1F,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x3F,0x80,
0xFF,0xFF,0xFF,0xFF,0x00,0xF0,0xFF,0x03,0x80,0xCF,0x80,0xF3,
0x7F,0x00,0xF0,0x0F,0x00,0x60,0x00,0x7F,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0x3F,0x80,0xFF,0xFF,
0xFF,0x7F,0x00,0xF8,0x1F,0x00,0x00,0x48,0x80,0xF3,0x7F,0x00,
0x00,0x01,0x00,0xC0,0x00,0xFE,0x01,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF8,0xFF,0x3F,0x80,0xFF,0xFF,0xFF,0x47,
0x00,0xFC,0x03,0x00,0x00,0x48,0x80,0xF3,0xFF,0x00,0x00,0x01,
0x00,0x80,0x01,0xF8,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFC,0xFF,0x7F,0x80,0xFF,0xFF,0xFF,0x40,0x00,0xFE,
0x03,0x00,0x00,0x48,0x80,0xF3,0xFF,0x01,0x00,0x01,0x00,0x00,
0x03,0xE0,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xFC,0xFF,0x7F,0x80,0xFF,0xFF,0x00,0x80,0x80,0xFF,0x01,0x00,
0x00,0x48,0x80,0xF3,0xFF,0x03,0x00,0x01,0x00,0x00,0x06,0x80,
0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,
0x7F,0xC0,0x01,0x20,0x00,0x80,0xE0,0xFF,0x01,0x00,0x00,0x48,
0xC0,0xF3,0xFF,0x07,0x00,0x02,0x00,0xC0,0x0F,0x00,0xFE,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,0x7F,0x60,
0x00,0x30,0x00,0x00,0xF9,0xFF,0x01,0x00,0x00,0x48,0xC0,0xE3,
0xFF,0x0F,0x00,0x02,0xC0,0xFF,0x1F,0x00,0xFC,0x01,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,0x7F,0x70,0x00,0x10,
0x00,0x00,0xFF,0xFF,0x00,0x00,0x00,0x68,0xC0,0xE3,0xFF,0x7F,
0x00,0x86,0xFF,0x03,0x00,0x00,0xF0,0x07,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xFC,0xFF,0x7F,0x30,0x00,0x10,0x00,0xC0,
0xFF,0xFF,0x00,0x00,0x00,0x28,0xC0,0xE3,0xFF,0xFF,0xFF,0xFF,
0x03,0x00,0x00,0x00,0xC0,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFC,0xFF,0x7F,0x18,0x00,0x18,0x00,0xFC,0xFF,0xFF,
0x00,0x00,0x00,0x28,0xC0,0xE3,0xFF,0xFF,0xFF,0xFF,0x00,0x00,
0x00,0x00,0x00,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xFC,0xFF,0x7F,0x0C,0x00,0x0C,0x00,0xFF,0xFF,0x7F,0x00,0x00,
0x00,0x38,0xC0,0xE3,0xFF,0xFF,0xFF,0xFF,0x07,0x00,0x00,0x00,
0x00,0xFE,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,
0x7F,0x0E,0x00,0x84,0xFF,0xFF,0xFF,0x7F,0x00,0x00,0x00,0x38,
0xC0,0xF7,0xFF,0xFF,0xFF,0xFF,0x7F,0x00,0x00,0x00,0x00,0xF8,
0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0x3F,0xFE,
0xFF,0xFF,0xFF,0xFF,0xFF,0x7F,0x00,0x00,0x00,0x38,0xC0,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0x01,0x00,0x00,0x00,0xE0,0x07,0x00,
0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x3F,0x00,0xF8,0xFF,
0xFF,0xFF,0xFF,0x7F,0x00,0x00,0x00,0x38,0xC0,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0x1F,0x00,0x00,0x00,0x80,0x1F,0x00,0x00,0x00,
0x00,0x00,0x00,0x80,0xFF,0xFF,0x3F,0x00,0xF8,0xFF,0xFF,0xFF,
0xFF,0xFF,0x03,0x00,0x00,0x38,0xC0,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0x3F,0x00,0x00,0x00,0x00,0x7E,0x00,0x00,0x00,0x00,0x00,
0x00,0xC0,0xFF,0xFF,0x1F,0x00,0xF8,0xFF,0xFF,0xFF,0xFF,0x1F,
0x7C,0x00,0x00,0x68,0xC0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x00,0x00,0x00,0x00,0xFC,0x01,0x00,0x00,0x00,0x00,0x00,0xC0,
0xFF,0xFF,0x1F,0x00,0xF0,0xFF,0xFF,0xFF,0xFF,0x1F,0x80,0xFF,
0xFF,0x6F,0x40,0xFE,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x01,0x00,
0x00,0x00,0xF0,0x03,0x00,0x00,0x00,0x00,0x00,0xC0,0xFF,0xFF,
0x0F,0x00,0xF0,0xFF,0xFF,0xFF,0xFF,0x0F,0x80,0x00,0x00,0x59,
0x60,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x01,0x00,0x00,0x00,
0xC0,0x0F,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x0F,0x00,
0xE0,0xFF,0xFF,0xFF,0xFF,0x0F,0x80,0x00,0x00,0x59,0x20,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x01,0x00,0x00,0x00,0x00,0x3F,
0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x07,0x00,0xE0,0xFF,
0xFF,0xFF,0xFF,0x0F,0xC0,0x00,0x00,0x59,0x20,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0x7E,0x00,0x00,
0x00,0x00,0x00,0xC0,0xFF,0xFF,0x07,0x00,0xE0,0xFF,0xFF,0xFF,
0xFF,0x7F,0xC0,0x00,0x00,0x79,0x20,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0xF8,0x01,0x00,0x00,0x00,
0x00,0xE0,0xFF,0xFF,0x03,0x00,0xE0,0xFF,0xFF,0xFF,0xFF,0xFF,
0x7F,0x00,0x00,0x39,0x30,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x03,0x00,0x00,0x00,0x00,0xE0,0x07,0x00,0x00,0x00,0x00,0xF0,
0xFF,0xFF,0x03,0x00,0xE0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0x2F,0x10,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x03,0x00,
0x00,0x00,0x00,0x80,0x0F,0x00,0x00,0x00,0x00,0xF0,0xFF,0xFF,
0x00,0x00,0xE0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x6F,
0x10,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x03,0x00,0x00,0x00,
0x00,0x00,0x3F,0x00,0x00,0x00,0x00,0xF8,0xFF,0xFF,0x00,0x00,
0xF0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x07,0x00,0x48,0x10,0xFC,
0xFF,0x7F,0xF8,0xFF,0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,
0xFC,0x00,0x00,0x00,0x00,0xF8,0xFF,0xFF,0x00,0x00,0xF0,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x00,0x4C,0x18,0xF0,0xFF,0x7F,
0x00,0x00,0x00,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0xF0,0x01,
0x00,0x00,0x00,0xF0,0xFF,0x7F,0x00,0x00,0xF0,0xFF,0xFF,0xFF,
0xFF,0xFF,0x7F,0x00,0xE0,0x45,0x18,0x82,0xFF,0x7F,0x00,0x00,
0x00,0xFE,0x07,0x00,0x00,0x00,0x00,0x00,0xC0,0x07,0x00,0x00,
0x00,0xF0,0xFF,0x7F,0x00,0x00,0xF0,0xFF,0xFF,0xFF,0xFF,0xFF,
0x3F,0x00,0xF0,0x47,0x18,0x02,0xFC,0x3F,0x00,0x00,0x00,0xFE,
0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x1F,0x00,0x00,0x00,0xF0,
0xFF,0x7F,0x00,0x00,0xF0,0xFF,0xFF,0xFF,0xFF,0xFF,0x1F,0x00,
0xB0,0x67,0x18,0x02,0xF0,0x3F,0x00,0x00,0x00,0xFC,0x07,0x00,
0x00,0x00,0x00,0x00,0x00,0x3E,0x00,0x00,0x00,0xF0,0xFF,0x7F,
0x00,0x00,0xF8,0xFF,0xFF,0xFF,0xFF,0xFF,0x0F,0x00,0xE0,0x25,
0x18,0x02,0xF0,0x1F,0x00,0x00,0x00,0xFC,0x0F,0x00,0x00,0x00,
0x00,0x00,0x00,0xF8,0x00,0x00,0x00,0xF8,0xFF,0x7F,0x00,0x00,
0xFC,0xFF,0xFF,0xFF,0xFF,0xFF,0x07,0x00,0xC0,0x2C,0x08,0x02,
0xFE,0x1F,0x00,0x00,0x00,0xF8,0x0F,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0x03,0x00,0x00,0xFC,0xFF,0x7F,0x00,0x00,0xFC,0xFF,
0xFF,0xFF,0xFF,0xFF,0x03,0x00,0x00,0x7C,0x1F,0xFC,0xFF,0x0F,
0x00,0x00,0x00,0xF8,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x80,
0x07,0x00,0x00,0xFE,0xFF,0x7F,0x00,0x00,0xFC,0xFF,0xFF,0xFF,
0xFF,0xFF,0x01,0x00,0x00,0xE0,0x9F,0xFC,0xFF,0x0F,0x00,0x00,
0x00,0xF8,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x1F,0x00,
0x00,0xFF,0xFF,0x7F,0x00,0x00,0xFE,0xFF,0xFF,0xFF,0xFF,0x7F,
0x00,0x00,0x00,0x00,0x50,0xFC,0xFF,0x07,0x00,0x00,0x00,0xF8,
0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7C,0x00,0x80,0xFF,
0xFF,0x7F,0x00,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0x3F,0x00,0x00,
0x00,0x00,0x50,0xFD,0xFF,0x03,0x00,0x00,0x00,0xF8,0x1F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x01,0x80,0xFF,0xFF,0x7F,
0x00,0x80,0xFF,0xFF,0xFF,0xFF,0xFF,0x0F,0x00,0x00,0x00,0x00,
0xD0,0xFF,0xFF,0x03,0x00,0x00,0x00,0xF8,0x1F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xC0,0x03,0x80,0xFF,0xFF,0x7F,0x00,0xC0,
0xFF,0xFF,0xFF,0xBF,0xFF,0x0F,0x00,0x00,0x00,0x00,0x30,0xFA,
0xFF,0x01,0x00,0x00,0x00,0xF8,0x1F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x80,0x0F,0x80,0xFF,0xFF,0x3F,0x00,0xE0,0xFF,0xFF,
0xFF,0x9F,0xFF,0x0F,0x00,0x00,0x00,0x00,0xE0,0xFB,0xFF,0x01,
0x00,0x00,0x00,0xFC,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x3E,0x80,0xFF,0xFF,0x3F,0x00,0xF0,0xFF,0xFF,0xFF,0x8F,
0xFF,0x0F,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0x00,0x00,0x00,
0x00,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x78,
0x80,0xFF,0xFF,0x3F,0x00,0xFC,0xFF,0xFF,0xFF,0x87,0xFF,0x3F,
0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0x00,0x00,0x00,0x00,0xFF,
0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0x80,0xFF,
0xFF,0x3F,0x00,0xFE,0xFF,0xFF,0xFF,0x83,0xFF,0xFF,0x00,0x00,
0x00,0x00,0x00,0xFC,0xFF,0x00,0x00,0x00,0x00,0xFF,0x3F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x80,0xFF,0xFF,0x3F,
0x00,0xFF,0xFF,0xFF,0xFF,0x00,0xFF,0xFF,0x07,0x00,0x00,0x00,
0x00,0xFC,0x7F,0x00,0x00,0x00,0x80,0xFF,0x3F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0xFF,0x1F,0x80,0xFF,
0xFF,0xFF,0x7F,0x00,0xFF,0xFF,0x1F,0x00,0x00,0x00,0x00,0xFC,
0x7F,0x00,0x00,0x00,0xC0,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x80,0xFF,0xFB,0x1F,0xE0,0xFF,0xFF,0xFF,
0x07,0x00,0xFF,0xFF,0x7F,0x00,0x00,0x00,0x00,0xFC,0x3F,0x00,
0x00,0x00,0xC0,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x80,0xFF,0xF3,0x0E,0xF0,0xFF,0xFF,0xFF,0x03,0x00,
0xFF,0xFF,0xFF,0x01,0x00,0x00,0x00,0xFC,0x3F,0x00,0x00,0x00,
0xFC,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0xFF,0x7B,0x0E,0xF8,0xFF,0xFF,0x3F,0x00,0x00,0xFF,0xFF,
0xFF,0x07,0x00,0x00,0x00,0xFC,0x3F,0x00,0x00,0x00,0xFE,0xFF,
0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xFF,
0x7B,0x06,0xFC,0xFF,0xFF,0x1F,0x00,0x00,0xFE,0xFF,0xFF,0x0F,
0x00,0x00,0x00,0xFC,0x1F,0x00,0x00,0x00,0xFF,0xFF,0x1F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0x39,0x83,
0xFF,0xFF,0xFF,0x07,0x00,0x00,0xF0,0xFF,0xFF,0x0F,0x00,0x00,
0x00,0xFC,0x1F,0x00,0x00,0x80,0xFF,0xFF,0x1F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0xB9,0x83,0xFF,0xFF,
0x7F,0x00,0x00,0x00,0x80,0xFF,0xFF,0x1F,0x00,0x00,0x00,0xFC,
0x0F,0x00,0x00,0x80,0xFF,0xFF,0x0F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF0,0xFF,0xBD,0xC1,0xFF,0xFF,0x3F,0x00,
0x00,0x00,0x00,0xFF,0xFF,0x1F,0x00,0x00,0x00,0xFE,0x07,0x00,
0x00,0xC0,0xFF,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xF0,0xFF,0x9D,0xC0,0xFF,0xFF,0x1F,0x00,0x00,0x00,
0x00,0xF0,0xFF,0x1F,0x00,0x00,0x00,0xFE,0x07,0x00,0x00,0xC0,
0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x70,0xFF,0xDF,0xC0,0xFF,0xFF,0x0F,0x00,0x00,0x00,0x00,0xC0,
0xFF,0x1F,0x00,0x00,0x00,0xFE,0x07,0x00,0x00,0xE0,0xFF,0xFF,
0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x90,0xFF,
0xDF,0xC0,0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0x80,0xFF,0x1F,
0x00,0x00,0x00,0xFE,0x03,0x00,0x00,0xE0,0xFF,0xFF,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x90,0xFF,0xFF,0xC0,
0xFF,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0xFF,0x1F,0x00,0x00,
0x00,0xFE,0x03,0x00,0x00,0xE0,0xFF,0x3F,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x90,0xFF,0x7F,0xE0,0xFF,0xFF,
0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x1F,0x00,0x00,0x00,0xFE,
0x03,0x00,0x00,0xE0,0xFF,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xD0,0xFF,0x3F,0xE0,0xFF,0x7F,0x00,0x00,
0x00,0x00,0x00,0x00,0xFC,0x1F,0x00,0x00,0x00,0xFE,0x03,0x00,
0x00,0xE0,0xFF,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xE0,0xFF,0x3F,0xE0,0xFF,0x3F,0x00,0x00,0x00,0x00,
0x00,0x00,0xF8,0x1F,0x00,0x00,0x00,0xFF,0x01,0x00,0x00,0xE0,
0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xE0,0xFF,0x1F,0xF0,0xFF,0x1F,0x00,0x00,0x00,0x00,0x00,0xE0,
0xFF,0x1F,0x00,0x00,0x00,0xFF,0x01,0x00,0x00,0xE0,0xFF,0x01,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x78,0xFF,
0x1F,0xF0,0xFF,0x0F,0x00,0x00,0x00,0x00,0x00,0xE0,0xFF,0x1F,
0x00,0x00,0x80,0xFF,0x01,0x00,0x00,0xE0,0xFF,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xBE,0xCF,0x0F,0xF0,
0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x1F,0x00,0x00,
0x80,0xFF,0x01,0x00,0x00,0xE0,0x3F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xE7,0x03,0xF0,0xFF,0x03,
0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x1F,0x00,0x00,0x80,0xFF,
0x00,0x00,0x00,0xE0,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x80,0xE3,0x01,0xF0,0xFF,0x03,0x00,0x00,
0x00,0x00,0x00,0xF0,0xFF,0x0F,0x00,0x00,0xC0,0xFF,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xC0,0xF3,0x00,0xF0,0xFF,0x01,0x00,0x00,0x00,0x00,
0x00,0xF0,0xFF,0x0F,0x00,0x00,0xC0,0xFF,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xC0,0xC1,0x00,0xE0,0xFF,0x01,0x00,0x00,0x00,0x00,0x00,0xF0,
0xFF,0x07,0x00,0x00,0xC0,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x40,
0x00,0xE0,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x07,
0x00,0x00,0xE0,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,
0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x03,0x00,0x00,
0xE0,0xFF,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xFF,0x0F,
0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x01,0x00,0x00,0xE0,0xFF,
0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xFF,0x0F,0x00,0x00,
0x00,0x00,0x00,0xF0,0xFF,0x01,0x00,0x00,0xE0,0xFF,0x01,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x80,0xFF,0x1F,0x00,0x00,0x00,0x00,
0x00,0xF0,0xFF,0x00,0x00,0x00,0xC0,0xFF,0x01,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0xF0,
0x3F,0x00,0x00,0x00,0xC0,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFE,0x7F,0x00,0x00,0x00,0x00,0x00,0xF0,0x1F,0x00,
0x00,0x00,0xC0,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0xFC,0x7F,0x00,0x00,0x00,0x00,0x00,0xE0,0x07,0x00,0x00,0x00,
0xC0,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,
0x00,0x00,0x00,0x00,0x00,0xE0,0x03,0x00,0x00,0x00,0x00,0xFF,
0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0xFF,0x03,0x00,
0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0xFE,0x1F,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x07,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0x7F,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xF0,0xFF,0x7F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xFC,0xFF,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xE0,0xFF,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xF8,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0xFF,0xFF,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xF0,0xFF,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,
0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0x03,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x0F,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0x03,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x0F,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0x07,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0xF0,0xFF,0x1F,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xFE,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0xE0,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xFE,0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xE0,0xFF,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,
0xFF,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,
0xFF,0x7F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00};
```

A detailed overview regarding the supported fonts can be created with xlsfonts and a single font can be displayed with xfd -fn <fontname>. Another important point relates to the supported colors. A good choice to have a look on this part consists in the call of

http://www.graphviz.org/doc/info/colors.html

You can use these colors with the very simple instruction XAllocNamedColor. The following program implies twenty randomly chosen colors which can be replaced by other colors. Remarkably two colors are not supported: Crimson and Indigo.

```
/*****************************************************************
 * X11 color demonstration program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
*****************************************************************/

#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define TRUE                    1
#define LEFT                    200
#define TOP                     200
#define WIDTH                   600
#define HEIGHT                  600
#define BORDER                  4
#define WAIT                    3
#define HEADER                  "COLORTEST"
#define FONT                    "9x15"

#define RC_SIZE                 280
#define RC_X                    10
#define RC_Y                    10
#define RC_SPC                  20

struct cols {
  int val;
  const char* name;
};

static const struct cols colors[] = {
    {.val  = 1,
    .name = "aliceblue" },
    {.val  = 2,
    .name = "bisque" },
    {.val  = 3,
    .name = "coral" },
    {.val  = 4,
    .name = "cyan3" },
    {.val  = 5,
    .name = "crimson" },
    {.val  = 6,
    .name = "firebrick" },
    {.val  = 7,
    .name = "gold" },
    {.val  = 8,
    .name = "gray22" },
    {.val  = 9,
    .name = "greenyellow" },
    {.val  = 10,
    .name = "indigo" },
    {.val  = 11,
    .name = "mediumpurple" },
    {.val  = 12,
    .name = "orange" },
    {.val  = 13,
    .name = "orchid" },
    {.val  = 14,
    .name = "royalblue" },
    {.val  = 15,
    .name = "sandybrown" },
    {.val  = 16,
    .name = "seagreen" },
    {.val  = 17,
    .name = "steelblue" },
    {.val  = 18,
    .name = "tomato" },
    {.val  = 19,
    .name = "violet" },
    {.val  = 20,
    .name = "wheat" },
    {.val  = 0, .name = "" }
};


int main (void)
{
    Display *disp;
    XVisualInfo vinfo;
    Visual *visual;
    int depth, screen, index,
        xp, tmp, quit = 0;
    Colormap cmap;
    XSetWindowAttributes wa;
    Window win;
    GC gc;
    XFontStruct *font;
    XCharStruct tsize;
    XColor col;
    XSizeHints sh;
    XTextProperty name;
    XEvent e;
    Atom close;
    char *txt = HEADER;

    disp = XOpenDisplay (NULL);
    if (disp == (Display *) NULL)
    {
      fprintf (stderr, "Cannot connect the X server\n");
      return EXIT_FAILURE;
    }
```

```
if (! XMatchVisualInfo (disp, 0, 32, TrueColor, &vinfo))
   if (! XMatchVisualInfo (disp, 0, 24, TrueColor, &vinfo))
      if (! XMatchVisualInfo (disp, 0, 16, TrueColor, &vinfo))
      {
         fprintf (stderr, "Cannot get TrueColor Visual\n");
         XCloseDisplay (disp);
         return EXIT_FAILURE;
      }

visual = vinfo.visual;
depth = vinfo.depth;
screen = vinfo.screen;
cmap = XCreateColormap (disp, RootWindow (disp, screen), visual, AllocNone);

wa.background_pixel = WhitePixel (disp, screen);
wa.border_pixel = BlackPixel (disp, screen);
wa.colormap = cmap;
win = XCreateWindow (disp, RootWindow (disp, screen), LEFT, TOP, WIDTH, HEIGHT, BORDER, depth,
                                             InputOutput, visual, CWBorderPixel | CWBackPixel | CWColormap, &wa);

sh.min_width = WIDTH;
sh.min_height = HEIGHT;
sh.flags = PMinSize;
XSetWMNormalHints (disp, win, &sh);
XStringListToTextProperty (&txt, 1, &name);
XSetWMName (disp, win, &name);

gc = XCreateGC (disp, win, 0, NULL);
if (! gc)
{
   fprintf (stderr, "Cannot create graphics context\n");
   XDestroyWindow (disp, screen);
   XFreeColormap (disp, cmap);
   XCloseDisplay (disp);
   return EXIT_FAILURE;
}

font = XLoadQueryFont (disp, FONT);
if (font == (XFontStruct *) NULL)
{
   fprintf (stderr, "Cannot load text font\n");
   XFreeGC (disp, gc);
   XDestroyWindow (disp, screen);
   XFreeColormap (disp, cmap);
   XCloseDisplay (disp);
   return EXIT_FAILURE;
}
XSetFont (disp, gc, font->fid);
XSetBackground (disp, gc, WhitePixel (disp, screen));

XSelectInput (disp, win, ButtonPressMask);
close = XInternAtom (disp, "WM_DELETE_WINDOW", TRUE);
if (close)
   XSetWMProtocols (disp, win, (Atom *) &close, 1);
```

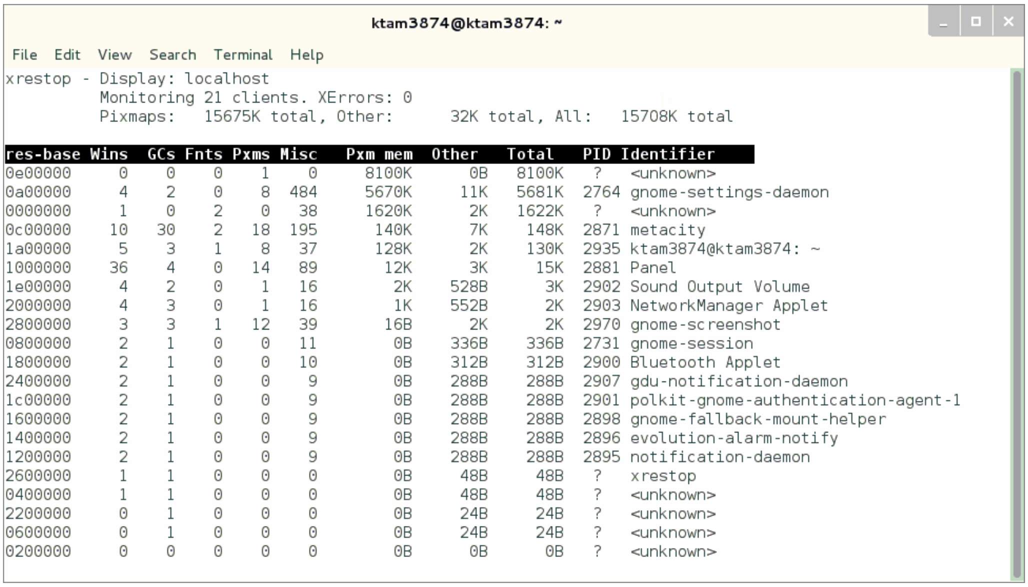
```
XMapWindow (disp, win);
index = 0;

while (! quit)
{
   if (XPending (disp))
   {
      XNextEvent (disp, &e);
      if ((e.type == ClientMessage && (Atom) (e.xclient.data.l[0]) == close) || (e.type == ButtonPress))
         quit = TRUE;
   }
   else
   {
      XClearWindow (disp, win);

      XAllocNamedColor (disp, cmap, colors[index].name, &col, &col);
      XSetForeground (disp, gc, col.pixel);
      XFillRectangle (disp, win, gc, RC_X, RC_Y, RC_SIZE, RC_SIZE);
      XSetForeground (disp, gc, BlackPixel (disp, screen));
      XTextExtents (font, colors[index].name, strlen (colors[index].name), &tmp, &tmp, &tmp, &tsize);
      xp = (RC_SIZE / 2) - (tsize.width / 2) + RC_X;
      XDrawImageString (disp, win, gc, xp, RC_Y+RC_SPC, colors[index].name, strlen (colors[index].name));
      index++;

      XAllocNamedColor (disp, cmap, colors[index].name, &col, &col);
      XSetForeground (disp, gc, col.pixel);
      XFillRectangle (disp, win, gc, RC_X+RC_SIZE+RC_SPC, RC_Y, RC_SIZE, RC_SIZE);
      XSetForeground (disp, gc, BlackPixel (disp, screen));
      XTextExtents (font, colors[index].name, strlen (colors[index].name), &tmp, &tmp, &tmp, &tsize);
      xp = (RC_SIZE / 2) - (tsize.width / 2) + RC_X+RC_SIZE+RC_SPC;
      XDrawImageString (disp, win, gc, xp, RC_Y+RC_SPC, colors[index].name, strlen (colors[index].name));
      index++;

      XAllocNamedColor (disp, cmap, colors[index].name, &col, &col);
      XSetForeground (disp, gc, col.pixel);
      XFillRectangle (disp, win, gc, RC_X, RC_Y+RC_SIZE+RC_SPC, RC_SIZE, RC_SIZE);
      XSetForeground (disp, gc, BlackPixel (disp, screen));
      XTextExtents (font, colors[index].name, strlen (colors[index].name), &tmp, &tmp, &tmp, &tsize);
      xp = (RC_SIZE / 2) - (tsize.width / 2) + RC_X;
      XDrawImageString (disp, win, gc, xp, RC_Y+(2*RC_SPC)+RC_SIZE, colors[index].name, strlen
                                                                            (colors[index].name));
      index++;

      XAllocNamedColor (disp, cmap, colors[index].name, &col, &col);
      XSetForeground (disp, gc, col.pixel);
      XFillRectangle (disp, win, gc, RC_X+RC_SIZE+RC_SPC, RC_Y+RC_SIZE+RC_SPC, RC_SIZE, RC_SIZE);
      XSetForeground (disp, gc, BlackPixel (disp, screen));
      XTextExtents (font, colors[index].name, strlen (colors[index].name), &tmp, &tmp, &tmp, &tsize);
      xp = (RC_SIZE / 2) - (tsize.width / 2) + RC_X+RC_SIZE+RC_SPC;
      XDrawImageString (disp, win, gc, xp, RC_Y+(2*RC_SPC)+RC_SIZE, colors[index].name, strlen
                                                                            (colors[index].name));
      index++;

      XFlush (disp);
      sleep (WAIT);
```

```
            if (! colors[index].val)
                index = 0;
        }
    }

    XFreeFont (disp, font);
    XFreeGC (disp, gc);
    XDestroyWindow (disp, win);
    XFreeColormap (disp, cmap);
    XCloseDisplay (disp);
    return EXIT_SUCCESS;
}
```

The X Server™ dislikes resources which have not been freed after program termination. One possibility to examine this fact is the use of xrestop. Download the tool with

```
sudo apt-get install xrestop
```

The picture below shows the appearance:



```
xrestop - Display: localhost
          Monitoring 21 clients. XErrors: 0
          Pixmaps:   15675K total, Other:      32K total, All:   15708K total

res-base Wins  GCs Fnts Pxms Misc   Pxm mem  Other   Total   PID Identifier
0e00000     0    0    0    1    0     8100K     0B   8100K   ?   <unknown>
0a00000     4    2    0    8  484     5670K    11K   5681K  2764 gnome-settings-daemon
0000000     1    0    2    0   38     1620K     2K   1622K   ?   <unknown>
0c00000    10   30    2   18  195      140K     7K    148K  2871 metacity
1a00000     5    3    1    8   37      128K     2K    130K  2935 ktam3874@ktam3874: ~
1000000    36    4    0   14   89       12K     3K     15K  2881 Panel
1e00000     4    2    0    1   16        2K   528B      3K  2902 Sound Output Volume
2000000     4    3    0    1   16        1K   552B      2K  2903 NetworkManager Applet
2800000     3    3    1   12   39       16B     2K      2K  2970 gnome-screenshot
0800000     2    1    0    0   11        0B   336B    336B  2731 gnome-session
1800000     2    1    0    0   10        0B   312B    312B  2900 Bluetooth Applet
2400000     2    1    0    0    9        0B   288B    288B  2907 gdu-notification-daemon
1c00000     2    1    0    0    9        0B   288B    288B  2901 polkit-gnome-authentication-agent-1
1600000     2    1    0    0    9        0B   288B    288B  2898 gnome-fallback-mount-helper
1400000     2    1    0    0    9        0B   288B    288B  2896 evolution-alarm-notify
1200000     2    1    0    0    9        0B   288B    288B  2895 notification-daemon
2600000     1    1    0    0    0        0B    48B     48B   ?   xrestop
0400000     1    1    0    0    0        0B    48B     48B   ?   <unknown>
2200000     0    1    0    0    0        0B    24B     24B   ?   <unknown>
0600000     0    1    0    0    0        0B    24B     24B   ?   <unknown>
0200000     0    0    0    0    0        0B     0B      0B   ?   <unknown>
```

10.12.2 Cairo

Cairo designates a vector-based 2D graphics library and supports output to many different backends - also named surfaces -, for example: X11^{TM}, PDF, PostScript® and SVG files. On an x86 Desktop PC using a cross compiler (Ubuntu^{TM} environment) you need some additional libraries (application dependent):

cairo
expat
fontconfig
freetype
pixman-1
png12
pthread
X11
Xrender
m (libm)
z (libz)

All dynamic libraries are available within the standard Debian^{TM} image (most of them in the directory /usr/lib/arm-linux-gnueabihf). The best way for development consists in the use of these libraries which should be copied from the target hardware to the cross compiler environment.
Another possible option represents the download of libcairo2-dev (includes all necessary libraries, but only as static version) on the target hardware with

```
sudo apt-get install libcairo2-dev
```

but this approach is not really recommended. For further steps see the chapter Sound Programming with ALSA. Maybe a link creation or renaming of library files could be required. Please note that the development libraries are not needed for the execution of Cairo applications on the target hardware because the standard Debian^{TM} image contains all necessary files.

> **ATTENTION**
>
> If you download the 'libcairo2-dev' environment on the target hardware without special precautions the image will likely be corrupted. The integrated files 'libcairo.so.2' and 'libcairo-gobject.so.2' are special versions and should not be overwritten.

Without special precautions the following download message appears (extract):

```
The following extra packages will be installed:
   binutils gcc gcc-4.6 libc-dev-bin libc6-dev libcairo-gobject2
   libcairo-script-interpreter2 libcairo2 libelf1 libexpat1-dev
   libfontconfig1-dev libfreetype6-dev libglib2.0-bin libglib2.0-dev libice-dev
   libpcre3-dev libpcrecpp0 libpixman-1-dev libpng12-dev libpthread-stubs0
   libpthread-stubs0-dev libsm-dev libx11-dev libx11-doc libxau-dev
   libxcb-render0-dev libxcb-shm0-dev libxcb1-dev libxdmcp-dev libxrender-dev
   linux-libc-dev manpages-dev x11proto-core-dev x11proto-input-dev
   x11proto-kb-dev x11proto-render-dev xorg-sgml-doctools xtrans-dev zlib1g-dev
Suggested packages:
   binutils-doc gcc-multilib make autoconf automake1.9 libtool flex bison gdb
   gcc-doc libmudflap0-4.6-dev gcc-4.6-doc gcc-4.6-locales libgcc1-dbg
   libgomp1-dbg libquadmath-dbg libmudflap0-dbg binutils-gold glibc-doc
   libcairo2-doc libglib2.0-doc libice-doc libsm-doc libxcb-doc
```

```
The following NEW packages will be installed:
    binutils gcc gcc-4.6 libc-dev-bin libc6-dev libcairo-script-interpreter2
    libcairo2-dev libelf1 libexpat1-dev libfontconfig1-dev libfreetype6-dev
    libglib2.0-bin libglib2.0-dev libice-dev libpcre3-dev libpcrecpp0
    libpixman-1-dev libpng12-dev libpthread-stubs0 libpthread-stubs0-dev
    libsm-dev libx11-dev libx11-doc libxau-dev libxcb-render0-dev
    libxcb-shm0-dev libxcb1-dev libxdmcp-dev libxrender-dev linux-libc-dev
    manpages-dev x11proto-core-dev x11proto-input-dev x11proto-kb-dev
    x11proto-render-dev xorg-sgml-doctools xtrans-dev zlib1g-dev
The following packages will be upgraded:
    libcairo-gobject2 libcairo2
```

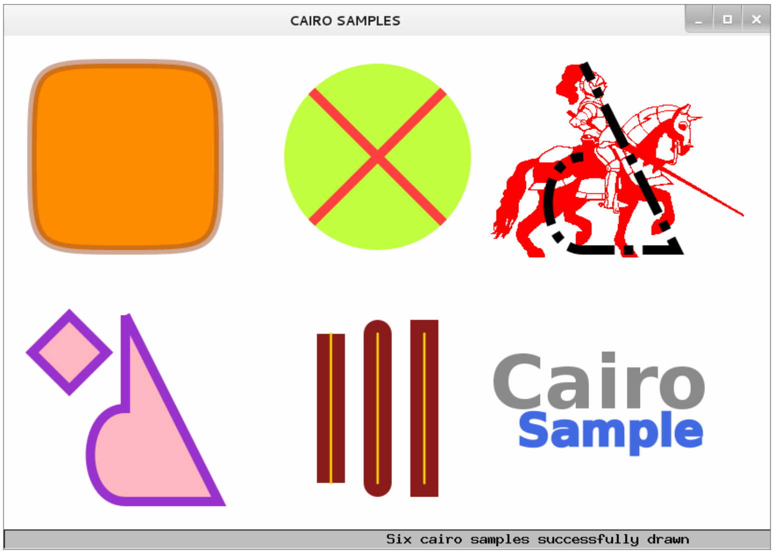
One possible Makefile might look like this:

```
CC := arm-linux-gnueabihf-gcc -march=armv7
LDFLAGS := -l cairo -l png12 -l pixman-1 -l fontconfig -l freetype -l expat -l pthread -l X11 -l Xrender -l z -l m

all: cairo

cairo.o: cairo.c

cairo: cairo.o
    $(CC) -o $@ cairo.o $(LDFLAGS)

clean:
    rm cairo cairo.o
```

There are some sourcecode examples on the webpage

http://cairographics.org/examples/

An interesting fact of Cairo represents the ability to combine X11^{TM} and Cairo graphic output:



Six cairo samples successfully drawn

The following demonstration program realizes six independent Cairo samples (the top right sample combines X11^{TM} and Cairo) and a simple X11^{TM} status bar:

```
/*********************************************************************
 * Cairo demonstration program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
 *********************************************************************/
```

```
#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include <cairo/cairo.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "picture.xbm"

#define TRUE                    1

#define LEFT                    200
#define TOP                     200
#define WIDTH                   820
#define HEIGHT                  550
#define BORDER                  4
#define HEADER                  "CAIRO SAMPLES"

#define OBJ_WIDTH               200.0
#define OBJ_HEIGHT              200.0
#define ALPHA                   0.5
#define LINE_WIDTH              10.0
#define FONT_SIZE               80.0
#define OBJ1_X                  30.0
#define OBJ1_Y                  30.0
#define OBJ2_X                  300.0
#define OBJ2_Y                  30.0
#define OBJ3_X                  520.0
#define OBJ3_Y                  30.0
#define OBJ4_X                  30.0
#define OBJ4_Y                  300.0
#define OBJ5_X                  300.0
#define OBJ5_Y                  300.0
#define OBJ6_X                  500.0
#define OBJ6_Y                  300.0

#define OBJ1_COL1               "darkorange"
#define OBJ1_COL2               "sienna"
#define OBJ2_COL1               "olivedrab1"
#define OBJ2_COL2               "brown1"
#define OBJ3_COL                "black"
#define OBJ4_COL1               "lightpink"
#define OBJ4_COL2               "darkorchid"
#define OBJ5_COL1               "firebrick4"
#define OBJ5_COL2               "gold"
#define OBJ6_COL1               "gray54"
#define OBJ6_COL2               "royalblue"
#define OBJ6_FONT               "Sans"

#define PICT_BG                 "white"
#define PICT_FG                 "red"
#define STAT_BG                 "gray"
#define STAT_HEIGHT             20
#define STAT_BORDER             5
#define STAT_FONT               "9x15bold"
```

```
#define STAT_TXT                "Six cairo samples successfully drawn"

#define RGB_R(c)                ((double)((c>>16)&255)/255.0)
#define RGB_G(c)                ((double)((c>>8)&255)/255.0)
#define RGB_B(c)                ((double)(c&255)/255.0)
#define C(c)                    (c.pixel)

#define HALF(x)                 (x/2.0)
#define QUART(x)                (x/4.0)
#define FIFTH(x)                (x/5.0)
#define TENTH(x)                (x/10.0)

typedef struct {
    Display *disp;
    Window win;
    Colormap cmap;
    Visual *visual;
    int screen;
    GC gc;
    XFontStruct *font;
    Pixmap pict;
    Atom close;
} XSystem;

static void draw_cairo_object1 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x0 = 0.0, y0 = 0.0,
            width  = OBJ_WIDTH,
            height = OBJ_HEIGHT,
            radius = HALF(OBJ_WIDTH),
            x1 = x0 + width,
            y1 = y0 + height;

    cairo_save (cr);

    cairo_move_to  (cr, x0, y0 + radius);
    cairo_curve_to (cr, x0 , y0, x0 , y0, x0 + radius, y0);
    cairo_line_to (cr, x1 - radius, y0);
    cairo_curve_to (cr, x1, y0, x1, y0, x1, y0 + radius);
    cairo_line_to (cr, x1 , y1 - radius);
    cairo_curve_to (cr, x1, y1, x1, y1, x1 - radius, y1);
    cairo_line_to (cr, x0 + radius, y1);
    cairo_curve_to (cr, x0, y1, x0, y1, x0, y1- radius);
    cairo_close_path (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ1_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_fill_preserve (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ1_COL2, &col, &col);
    cairo_set_source_rgba (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)), ALPHA);
    cairo_set_line_width (cr, LINE_WIDTH);
    cairo_stroke (cr);

    cairo_restore (cr);
}
```

```
static void draw_cairo_object2 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x = 0.0, y = 0.0,
           width  = OBJ_WIDTH,
           height = OBJ_HEIGHT,
           radius = HALF(OBJ_WIDTH);

    cairo_save (cr);

    cairo_arc (cr, x + radius, y + radius, radius, 0.0, 2.0 * M_PI);
    cairo_clip (cr);

    cairo_new_path (cr);                        /* current path is not consumed by cairo_clip */
    XAllocNamedColor (sys.disp, sys.cmap, OBJ2_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_rectangle (cr, x, y, width, height);
    cairo_fill (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ2_COL2, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_move_to (cr, x, y);
    cairo_line_to (cr, x + width, y + height);
    cairo_move_to (cr, x + width, y);
    cairo_line_to (cr, x, y + height);
    cairo_set_line_width (cr, LINE_WIDTH);
    cairo_stroke (cr);

    cairo_restore (cr);
}


static void draw_cairo_object3 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x = 0.0, y = 0.0,
           width  = OBJ_WIDTH,
           height = OBJ_HEIGHT,
           dashes[] = { 50.0, 10.0, 10.0, 10.0 },
           offset = -dashes[0];
    int ndash = sizeof (dashes) / sizeof (dashes[0]);

    cairo_save (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ3_COL, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_set_dash (cr, dashes, ndash, offset);
    cairo_set_line_width (cr, LINE_WIDTH);
    cairo_move_to (cr, x + HALF(width), y);
    cairo_line_to (cr, x + width, y + height);
    cairo_rel_line_to (cr, -HALF(width), 0.0);
    cairo_curve_to (cr, x + QUART(width), y + height, x + QUART(width), y + HALF(height),
                                                       x + HALF(width), y + HALF(height));
    cairo_stroke (cr);

    cairo_restore (cr);
}
```

```
static void draw_cairo_object4 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x = 0.0, y = 0.0,
            width  = OBJ_WIDTH,
            height = OBJ_HEIGHT;

    cairo_save (cr);

    cairo_move_to (cr, x + HALF(width), y);
    cairo_line_to (cr, x + width, y + height);
    cairo_rel_line_to (cr, -HALF(width), 0.0);
    cairo_curve_to (cr, x + QUART(width), y + height, x + QUART(width), y + HALF(height),
                                                              x + HALF(width), y + HALF(height));
    cairo_rel_line_to (cr, 0.0, -HALF(height));

    cairo_move_to (cr, x + FIFTH(width), y);
    cairo_rel_line_to (cr, FIFTH(width), FIFTH(height));
    cairo_rel_line_to (cr, -FIFTH(width), FIFTH(height));
    cairo_close_path (cr);

    cairo_set_line_width (cr, LINE_WIDTH);
    XAllocNamedColor (sys.disp, sys.cmap, OBJ4_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_fill_preserve (cr);
    XAllocNamedColor (sys.disp, sys.cmap, OBJ4_COL2, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_stroke (cr);

    cairo_restore (cr);
}


static void draw_cairo_object5 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x = 0.0, y = 0.0,
            width  = OBJ_WIDTH,
            height = OBJ_HEIGHT,
            cap = TENTH(height);

    cairo_save (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ5_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_set_line_width (cr, LINE_WIDTH * 3.0);
    cairo_set_line_cap  (cr, CAIRO_LINE_CAP_BUTT);
    cairo_move_to (cr, x + QUART(width), y + cap);
    cairo_line_to (cr, x + QUART(width), y + height - cap);
    cairo_stroke (cr);
    cairo_set_line_cap  (cr, CAIRO_LINE_CAP_ROUND);
    cairo_move_to (cr, x + HALF(width), y + cap);
    cairo_line_to (cr, x + HALF(width), y + height - cap);
    cairo_stroke (cr);
    cairo_set_line_cap  (cr, CAIRO_LINE_CAP_SQUARE);
    cairo_move_to (cr, x + QUART(width) * 3.0, y + cap);
    cairo_line_to (cr, x + QUART(width) * 3.0, y + height - cap);
    cairo_stroke (cr);

    /* draw helping lines */
    cairo_set_line_width (cr, QUART(LINE_WIDTH));
    cairo_move_to (cr, x + QUART(width), y + cap);
    cairo_line_to (cr, x + QUART(width), y + height - cap);
    cairo_move_to (cr, x + HALF(width), y + cap);
    cairo_line_to (cr, x + HALF(width), y + height - cap);
    cairo_move_to (cr, x + QUART(width) * 3.0, y + cap);
    cairo_line_to (cr, x + QUART(width) * 3.0, y + height - cap);
    XAllocNamedColor (sys.disp, sys.cmap, OBJ5_COL2, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_stroke (cr);

    cairo_restore (cr);
}


static void draw_cairo_object6 (XSystem sys, cairo_t *cr)
{
    XColor col;
    double x = 0.0, y = 0.0,
            width  = OBJ_WIDTH,
            height = OBJ_HEIGHT;

    cairo_save (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ6_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (col.pixel), RGB_G (col.pixel), RGB_B (col.pixel));
    cairo_select_font_face (cr, OBJ6_FONT, CAIRO_FONT_SLANT_NORMAL, CAIRO_FONT_WEIGHT_BOLD);
    cairo_set_font_size (cr, FONT_SIZE);
    cairo_move_to (cr, x + TENTH(width), y + HALF(height));
    cairo_show_text (cr, "Cairo");

    XAllocNamedColor (sys.disp, sys.cmap, OBJ6_COL2, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (col.pixel), RGB_G (col.pixel), RGB_B (col.pixel));
    cairo_set_font_size (cr, TENTH(FONT_SIZE) * 6.0);
    cairo_move_to (cr, x + QUART(width), y + TENTH(height) * 7.0);
    cairo_text_path (cr, "Sample");
    cairo_fill_preserve (cr);
    cairo_set_line_width (cr, QUART(LINE_WIDTH));
    cairo_stroke (cr);

    cairo_restore (cr);
}
```

```
static void draw_x11_object (XSystem *sys)
{
    XColor col;
    Pixmap pict;

    XAllocNamedColor (sys->disp, sys->cmap, PICT_FG, &col, &col);
    XSetForeground (sys->disp, sys->gc, col.pixel);
    XAllocNamedColor (sys->disp, sys->cmap, PICT_BG, &col, &col);
    XSetBackground (sys->disp, sys->gc, col.pixel);
    pict = XCreateBitmapFromData (sys->disp, sys->win, xbm_bits, xbm_width, xbm_height);
    XCopyPlane (sys->disp, pict, sys->win, sys->gc, 0, 0, xbm_width, xbm_height, OBJ3_X, OBJ3_Y, 1);
    sys->pict = pict;
}


static void draw_x11_status_bar (XSystem sys)
{
    XColor col;
    XWindowAttributes attr;
    char *txt = STAT_TXT;

    XGetWindowAttributes (sys.disp, sys.win, &attr);
    XAllocNamedColor (sys.disp, sys.cmap, STAT_BG, &col, &col);
    XSetForeground (sys.disp, sys.gc, col.pixel);
    XFillRectangle (sys.disp, sys.win, sys.gc, 0, attr.height - STAT_HEIGHT, attr.width, attr.height);
    XSetForeground (sys.disp, sys.gc, BlackPixel (sys.disp, sys.screen));
    XDrawRectangle (sys.disp, sys.win, sys.gc, 0, attr.height - STAT_HEIGHT, attr.width, attr.height);
    XSetFont (sys.disp, sys.gc, sys.font->fid);
    XSetForeground (sys.disp, sys.gc, BlackPixel (sys.disp, sys.screen));
    XSetBackground (sys.disp, sys.gc, col.pixel);
    XDrawImageString (sys.disp, sys.win, sys.gc, HALF(attr.width), attr.height - STAT_BORDER, txt, strlen (txt));
}


static int create_X11_window (XSystem *sys)
{
    Display *disp;
    XVisualInfo vinfo;
    Visual *visual;
    int depth, screen;
    Colormap cmap;
    XSetWindowAttributes wa;
    Window win;
    GC gc;
    XFontStruct *font;
    XSizeHints sh;
    XTextProperty name;
    Atom close;
    char *txt = HEADER;
```

```
disp = XOpenDisplay (NULL);
if (disp == (Display *) NULL)
{
   fprintf (stderr, "Cannot connect the X server\n");
   return EXIT_FAILURE;
}

if (!XMatchVisualInfo (disp, 0, 32, TrueColor, &vinfo))
   if (!XMatchVisualInfo (disp, 0, 24, TrueColor, &vinfo))
      if (!XMatchVisualInfo (disp, 0, 16, TrueColor, &vinfo))
      {
         fprintf (stderr, "Cannot get TrueColor Visual\n");
         XCloseDisplay (disp);
         return EXIT_FAILURE;
      }

visual = vinfo.visual;
depth = vinfo.depth;
screen = vinfo.screen;
cmap = XCreateColormap (disp, RootWindow (disp, screen), visual, AllocNone);

wa.background_pixel = WhitePixel (disp, screen);
wa.border_pixel = BlackPixel (disp, screen);
wa.colormap = cmap;
win = XCreateWindow (disp, RootWindow (disp, screen), LEFT, TOP, WIDTH, HEIGHT, BORDER, depth,
                                    InputOutput, visual, CWBorderPixel | CWBackPixel | CWColormap, &wa);

gc = XCreateGC (disp, win, 0, NULL);
if (! gc)
{
   fprintf (stderr, "Cannot create graphics context\n");
   XDestroyWindow (disp, win);
   XFreeColormap (disp, cmap);
   XCloseDisplay (disp);
   return EXIT_FAILURE;
}

font = XLoadQueryFont (disp, STAT_FONT);

sh.min_width = WIDTH;
sh.min_height = HEIGHT;
sh.flags = PMinSize;
XSetWMNormalHints (disp, win, &sh);
XStringListToTextProperty (&txt, 1, &name);
XSetWMName (disp, win, &name);

XSelectInput (disp, win, ButtonPressMask | ExposureMask | StructureNotifyMask);
close = XInternAtom (disp, "WM_DELETE_WINDOW", TRUE);
if (close)
   XSetWMProtocols (disp, win, (Atom *) &close, 1);

XMapWindow (disp, win);
```

```
    sys->disp       = disp;
    sys->win        = win;
    sys->cmap       = cmap;
    sys->visual     = visual;
    sys->screen     = screen;
    sys->gc         = gc;
    sys->font       = font;
    sys->close      = close;
    return EXIT_SUCCESS;
}


static void destroy_X11_window (XSystem sys)
{
    if (sys.pict)
      XFreePixmap (sys.disp, sys.pict);
    if (sys.font != (XFontStruct *) NULL)
      XFreeFont (sys.disp, sys.font);
    if (sys.gc)
      XFreeGC (sys.disp, sys.gc);
    if (sys.win)
      XDestroyWindow (sys.disp, sys.win);
    if (sys.cmap)
      XFreeColormap (sys.disp, sys.cmap);
    XCloseDisplay (sys.disp);
}


int main (void)
{
    int quit = 0;
    XSystem xsys;
    XEvent e;
    cairo_t *cr;
    cairo_surface_t *surface;

    memset ((void *) &xsys, 0, sizeof (xsys));
    if (create_X11_window (&xsys))
      return EXIT_FAILURE;

    surface = (cairo_surface_t *) cairo_xlib_surface_create (xsys.disp, xsys.win, xsys.visual, WIDTH, HEIGHT);
    if (cairo_surface_status (surface) != CAIRO_STATUS_SUCCESS)
    {
      fprintf (stderr, "Cannot create cairo surface\n");
      destroy_X11_window (xsys);
      return EXIT_FAILURE;
    }
    cr = cairo_create (surface);
    if (cairo_status (cr) != CAIRO_STATUS_SUCCESS)
    {
      fprintf (stderr, "Cannot create cairo context\n");
      cairo_surface_destroy (surface);
      destroy_X11_window (xsys);
      return EXIT_FAILURE;
    }

    while (! quit)
    {
      XNextEvent (xsys.disp, &e);
      if ((e.type == ClientMessage && (Atom) (e.xclient.data.l[0]) == xsys.close) || (e.type == ButtonPress))
        quit = TRUE;
      else if (e.type == ConfigureNotify)
      {
        if (e.xconfigure.width != WIDTH || e.xconfigure.height != HEIGHT)
          cairo_xlib_surface_set_size (surface, e.xconfigure.width, e.xconfigure.height);
      }
      else if (e.type == Expose)
      {
        draw_x11_object (&xsys);
        cairo_surface_set_device_offset (surface, OBJ1_X, OBJ1_Y);
        draw_cairo_object1 (xsys, cr);
        cairo_surface_set_device_offset (surface, OBJ2_X, OBJ2_Y);
        draw_cairo_object2 (xsys, cr);
        cairo_surface_set_device_offset (surface, OBJ3_X, OBJ3_Y);
        draw_cairo_object3 (xsys, cr);
        cairo_surface_set_device_offset (surface, OBJ4_X, OBJ4_Y);
        draw_cairo_object4 (xsys, cr);
        cairo_surface_set_device_offset (surface, OBJ5_X, OBJ5_Y);
        draw_cairo_object5 (xsys, cr);
        cairo_surface_set_device_offset (surface, OBJ6_X, OBJ6_Y);
        draw_cairo_object6 (xsys, cr);
        draw_x11_status_bar (xsys);
      }
    }

    cairo_destroy (cr);
    cairo_surface_destroy (surface);
    destroy_X11_window (xsys);
    return EXIT_SUCCESS;
}
```

When using static libraries there is one limitation: the gradient feature does not work.

The time has come to explain the PDF, PostScript$^{®}$ and SVG-file surface programming. It would seem that the Include-file selection will have a special significance. The following table gives an overview:

Surface	Only use the Include-file
X11^{TM}	cairo/cairo.h
PDF	cairo/cairo-pdf.h
PostScript$^{®}$	cairo/cairo-ps.h
SVG-file	cairo/cairo-svg.h

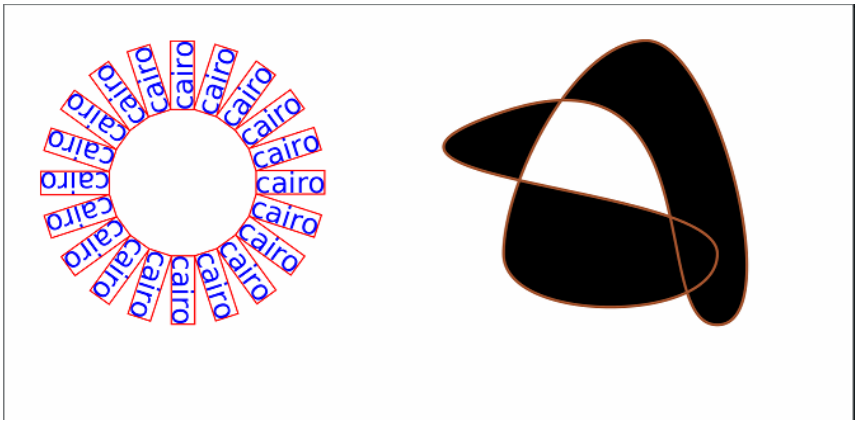
For example if you want to use the PostScript$^{®}$ surface and you only integrate the cairo.h file the application generates an empty PS-file (however, the header part is present). The demonstration program also involves the X Server^{TM} for color definition but without displaying a window on the monitor. Another major difference affects the surface size. The X11^{TM} environment needs the use of pixel, but PDF, PostScript$^{®}$ and SVG 1/72 inch (72 DPI). Within the Cairo sourcecode package you can find the following extract:

```
static const cairo_page_standard_media_t _cairo_page_standard_media[] =
{
    { "A0",             2384, 3371 },
    { "A1",             1685, 2384 },
    { "A2",             1190, 1684 },
    { "A3",             842, 1190 },
    { "A4",             595, 842 },
    { "A5",             420, 595 },
    { "B4",             729, 1032 },
    { "B5",             516, 729 },
    { "Letter",         612, 792 },
    { "Tabloid",        792, 1224 },
    { "Ledger",         1224, 792 },
    { "Legal",          612, 1008 },
    { "Statement",      396, 612 },
    { "Executive",      540, 720 },
    { "Folio",          612, 936 },
    { "Quarto",         610, 780 },
    { "10x14",          720, 1008 },
};
```

Calculation example for A4 format (1 inch = 25,4 mm):

```
                (595 / 72) * 25.4 = 209.903 [mm]
                (842 / 72) * 25.4 = 297,04 [mm]
```

Following sourcecode generates a PDF file (A4 format) with two pictures:



```
/*****************************************************************
 * Cairo PDF demonstration program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
*****************************************************************/

#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include <cairo/cairo-pdf.h>
/* #include <cairo/cairo-ps.h>
#include <cairo/cairo-svg.h> */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>

#define WIDTH                         595
#define HEIGHT                        842

#define OBJ1_COL1                     "black"
#define OBJ1_COL2                     "sienna"
#define OBJ2_COL1                     "red"
#define OBJ2_COL2                     "blue"

#define OBJ2_XOFFS                    300
#define OBJ2_YOFFS                    0
#define NUM_TEXT                      20
#define TEXT_WIDTH                    250.0
#define TEXT_HEIGHT                   250.0
#define TEXT_SIZE                     20.0

#define RGB_R(c) ((double)((c>>16)&255)/255.0)
#define RGB_G(c) ((double)((c>>8)&255)/255.0)
#define RGB_B(c) ((double)(c&255)/255.0)
#define C(c) (c.pixel)

typedef struct {
    Display *disp;
    Colormap cmap;
} XSystem;


static void draw_quadrant (XSystem sys, cairo_t *cr, const char *text, const cairo_text_extents_t *extents,
                              const cairo_matrix_t *transform, int x_off, int y_off)
{
    XColor col;
    int i;

    for (i = 0; i < NUM_TEXT / 4; i++)
    {
      cairo_save (cr);

      cairo_rotate (cr, 2.0 * M_PI * (double) i / (double) NUM_TEXT);
      cairo_transform (cr, transform);
      cairo_set_line_width (cr, 1.0);
      cairo_rectangle (cr, (double) x_off - 0.5, (double) y_off - 0.5, extents->width + 1.0, extents->height + 1.0);
```

```
        XAllocNamedColor (sys.disp, sys.cmap, OBJ2_COL1, &col, &col);
        cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
        cairo_stroke (cr);

        cairo_move_to (cr, x_off - extents->x_bearing, y_off - extents->y_bearing);
        XAllocNamedColor (sys.disp, sys.cmap, OBJ2_COL2, &col, &col);
        cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
        cairo_show_text (cr, text);

        cairo_restore (cr);
    }
}


static void draw_cairo_object1 (XSystem sys, cairo_t *cr)
{
    cairo_text_extents_t extents;
    cairo_font_options_t *font_options;
    const char text[] = "cairo";
    int x_off, y_off;
    cairo_matrix_t m;

    cairo_save (cr);

    cairo_select_font_face (cr, "Sans", CAIRO_FONT_SLANT_NORMAL, CAIRO_FONT_WEIGHT_NORMAL);
    cairo_set_font_size (cr, TEXT_SIZE);
    font_options = cairo_font_options_create ();
    cairo_get_font_options (cr, font_options);
    cairo_font_options_set_hint_metrics (font_options, CAIRO_HINT_METRICS_OFF);
    cairo_set_font_options (cr, font_options);
    cairo_font_options_destroy (font_options);

    cairo_translate (cr, TEXT_WIDTH / 2.0, TEXT_HEIGHT / 2.0);
    cairo_text_extents (cr, text, &extents);
    x_off = floor (0.5 + (extents.height + 1.0) / (2.0 * tan (M_PI / (double) NUM_TEXT)));
    y_off = -floor (0.5 + extents.height / 2.0);

    cairo_save (cr);
    cairo_matrix_init_identity (&m);
    draw_quadrant (sys, cr, text, &extents, &m, x_off, y_off);
    cairo_matrix_init (&m, 0.0, 1.0, -1.0, 0.0, 0.0, 0.0);
    draw_quadrant (sys, cr, text, &extents, &m, x_off, y_off);
    cairo_restore (cr);

    cairo_save (cr);
    cairo_scale (cr, -1.0, -1.0);
    cairo_matrix_init_identity (&m);
    draw_quadrant (sys, cr, text, &extents, &m, x_off, y_off);
    cairo_matrix_init (&m, 0.0, 1.0, -1.0, 0.0, 0.0, 0.0);
    draw_quadrant (sys, cr, text, &extents, &m, x_off, y_off);
    cairo_restore (cr);

    cairo_restore (cr);
}
```

```
static void draw_cairo_object2 (XSystem sys, cairo_t *cr)
{
    XColor col;

    cairo_save (cr);

    cairo_translate (cr, 0.0, -25.0);
    cairo_move_to  (cr, 50.0, 200.0);
    cairo_curve_to (cr, 50.0, 150.0, 100.0, 50.0, 150.0, 50.0);
    cairo_curve_to (cr, 200.0, 50.0, 250.0, 250.0, 200.0, 250.0);
    cairo_curve_to (cr, 150.0, 250.0, 200.0, 50.0, 50.0, 100.0);
    cairo_curve_to (cr, -100.0, 150.0, 200.0, 150.0, 200.0, 200.0);
    cairo_curve_to (cr, 200.0, 250.0, 50.0, 250.0, 50.0, 200.0);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ1_COL1, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_fill_preserve (cr);

    XAllocNamedColor (sys.disp, sys.cmap, OBJ1_COL2, &col, &col);
    cairo_set_source_rgb (cr, RGB_R (C(col)), RGB_G (C(col)), RGB_B (C(col)));
    cairo_stroke (cr);

    cairo_restore (cr);
}


static int create_X11_window (XSystem *sys)
{
    Display *disp;
    XVisualInfo vinfo;
    Visual *visual;
    int screen;
    Colormap cmap;

    disp = XOpenDisplay (NULL);
    if (disp == (Display *) NULL)
    {
      fprintf (stderr, "Cannot connect the X server\n");
      return EXIT_FAILURE;
    }

    if (!XMatchVisualInfo (disp, 0, 32, TrueColor, &vinfo))
      if (!XMatchVisualInfo (disp, 0, 24, TrueColor, &vinfo))
        if (!XMatchVisualInfo (disp, 0, 16, TrueColor, &vinfo))
        {
          fprintf (stderr, "Cannot get TrueColor Visual\n");
          XCloseDisplay (disp);
          return EXIT_FAILURE;
        }

    visual = vinfo.visual;
    screen = vinfo.screen;
    cmap = XCreateColormap (disp, RootWindow (disp, screen), visual, AllocNone);

    sys->disp = disp;
    sys->cmap = cmap;
    return EXIT_SUCCESS;
}
```

```
static void destroy_X11_window (XSystem sys)
{
    if (sys.cmap)
      XFreeColormap (sys.disp, sys.cmap);
    XCloseDisplay (sys.disp);
}


int main (void)
{
    XSystem xsys;
    cairo_t *cr;
    cairo_surface_t *surface;

    memset ((void *) &xsys, 0, sizeof (xsys));
    if (create_X11_window (&xsys))
      return EXIT_FAILURE;

    surface = (cairo_surface_t *) cairo_pdf_surface_create ("test.pdf", WIDTH, HEIGHT);
    /* surface = (cairo_surface_t *) cairo_ps_surface_create ("test.ps", WIDTH, HEIGHT);
    surface = (cairo_surface_t *) cairo_svg_surface_create ("test.svg", WIDTH, HEIGHT); */
    if (cairo_surface_status (surface) != CAIRO_STATUS_SUCCESS)
    {
      fprintf (stderr, "Cannot create cairo surface\n");
      destroy_X11_window (xsys);
      return EXIT_FAILURE;
    }

    cr = cairo_create (surface);
    if (cairo_status (cr) != CAIRO_STATUS_SUCCESS)
    {
      fprintf (stderr, "Cannot create cairo context\n");
      cairo_surface_destroy (surface);
      destroy_X11_window (xsys);
      return EXIT_FAILURE;
    }

    draw_cairo_object1 (xsys, cr);
    cairo_surface_set_device_offset (surface, OBJ2_XOFFS, OBJ2_YOFFS);
    draw_cairo_object2 (xsys, cr);

    cairo_show_page (cr);          /* Only necessary for PDF and PS */

    cairo_destroy (cr);
    cairo_surface_destroy (surface);
    destroy_X11_window (xsys);
    return EXIT_SUCCESS;
}
```

10.12.3 GTK+ 3.x

GTK+ represents a multi-platform Graphical User Interface Toolkit with a wide range of widgets and controls. On an x86 Desktop PC using a cross compiler (Ubuntu™ environment) you need some additional libraries (application dependent):

gtk-3
gdk-3
gobject-2.0
glib-2.0
pango-1.0
pangocairo-1.0
gdk_pixbuf-2.0
cairo
png12
expat
selinux
pcre
gio-2.0
z (libz)

All dynamic libraries are available within the standard Debian™ image (most of them in the directories /usr/lib/arm-linux-gnueabihf and /lib/arm-linux-gnueabihf). The best way for development consists in the use of these libraries which should be copied from the target hardware to the cross compiler environment. Maybe you also need some additional Include-files like glibconfig.h or the directory gdk-pixbuf. Normally all x86 Include-files are compatible with the ARM® files and could be copied to the ARM® environment.
Another possible option represents the download of libgtk-3-dev (includes all necessary libraries and Include-files) on the target hardware with

```
sudo apt-get install libgtk-3-dev
```

but this approach is not really recommended. For further steps see the chapter Sound Programming with ALSA. Maybe a link creation or renaming of library files could be required. Please note that the development libraries are not needed for the execution of GTK+ applications on the target hardware because the standard Debian™ image contains all necessary files.

ATTENTION

If you download the 'libgtk-3-dev' environment on the target hardware without special precautions the image will likely be corrupted. The integrated files 'libcairo.so.2' and 'libcairo-gobject.so.2' are special versions and should not be overwritten.

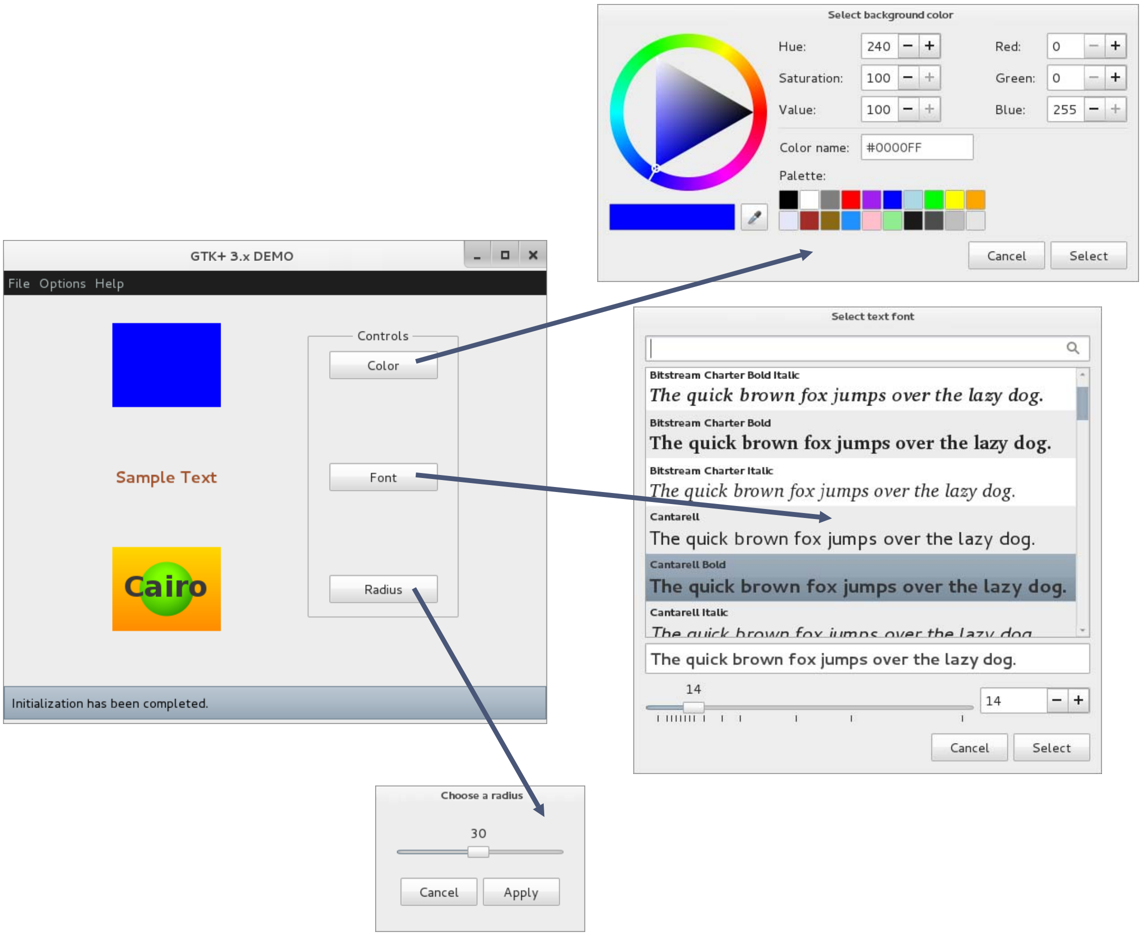
Without special precautions the following download message appears (extract):

```
The following extra packages will be installed:
  autopoint binutils build-essential debhelper dpkg-dev fakeroot g++ g++-4.6
  gcc gcc-4.6 gettext gettext-base git git-man html2text intltool-debian
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl
  libasprintf0c2 libatk1.0-dev libc-dev-bin libc6-dev libcairo-gobject2
  libcairo-script-interpreter2 libcairo2 libcairo2-dev libdpkg-perl libelf1
  liberror-perl libexpat1-dev libfile-fcntllock-perl libfontconfig1-dev
  libfreetype6-dev libgdk-pixbuf2.0-dev libgettextpo0 libglib2.0-bin
  libglib2.0-dev libice-dev libmail-sendmail-perl libpango1.0-dev libpcre3-dev
  libpcrecpp0 libpixman-1-dev libpng12-dev libpthread-stubs0
  libpthread-stubs0-dev libsm-dev libstdc++6-4.6-dev libsys-hostname-long-perl
  libx11-dev libx11-doc libxau-dev libxcb-render0-dev libxcb-shm0-dev
  libxcb1-dev libxcomposite-dev libxcursor-dev libxdamage-dev libxdmcp-dev
  libxext-dev libxfixes-dev libxft-dev libxi-dev libxinerama-dev libxrandr-dev
  libxrender-dev linux-libc-dev make manpages-dev patch po-debconf rsync
  x11proto-composite-dev x11proto-core-dev x11proto-damage-dev
  x11proto-fixes-dev x11proto-input-dev x11proto-kb-dev x11proto-randr-dev
  x11proto-render-dev x11proto-xext-dev x11proto-xinerama-dev
  xorg-sgml-doctools xtrans-dev zlib1g-dev
Suggested packages:
  binutils-doc dh-make debian-keyring gcc-4.6-doc libstdc++6-4.6-dbg
  gcc-multilib autoconf automake1.9 libtool flex bison gdb gcc-doc
  libmudflap0-4.6-dev gcc-4.6-locales libgcc1-dbg libgomp1-dbg libquadmath-dbg
  libmudflap0-dbg binutils-gold gettext-doc git-daemon-run git-daemon-sysvinit
  git-doc git-el git-arch git-cvs git-svn git-email git-gui gitk gitweb
  glibc-doc libcairo2-doc libglib2.0-doc libgtk-3-doc libice-doc
  libpango1.0-doc imagemagick libsm-doc libstdc++6-4.6-doc libxcb-doc
  libxext-doc make-doc ed diffutils-doc libmail-box-perl
The following NEW packages will be installed:
  autopoint binutils build-essential debhelper dpkg-dev fakeroot g++ g++-4.6
  gcc gcc-4.6 gettext gettext-base git git-man html2text intltool-debian
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl
  libasprintf0c2 libatk1.0-dev libc-dev-bin libc6-dev
  libcairo-script-interpreter2 libcairo2-dev libdpkg-perl libelf1
  liberror-perl libexpat1-dev libfile-fcntllock-perl libfontconfig1-dev
  libfreetype6-dev libgdk-pixbuf2.0-dev libgettextpo0 libglib2.0-bin
  libglib2.0-dev libgtk-3-dev libice-dev libmail-sendmail-perl libpango1.0-dev
  libpcre3-dev libpcrecpp0 libpixman-1-dev libpng12-dev libpthread-stubs0
  libpthread-stubs0-dev libsm-dev libstdc++6-4.6-dev libsys-hostname-long-perl
  libx11-dev libx11-doc libxau-dev libxcb-render0-dev libxcb-shm0-dev
  libxcb1-dev libxcomposite-dev libxcursor-dev libxdamage-dev libxdmcp-dev
  libxext-dev libxfixes-dev libxft-dev libxi-dev libxinerama-dev libxrandr-dev
  libxrender-dev linux-libc-dev make manpages-dev patch po-debconf rsync
  x11proto-composite-dev x11proto-core-dev x11proto-damage-dev
  x11proto-fixes-dev x11proto-input-dev x11proto-kb-dev x11proto-randr-dev
  x11proto-render-dev x11proto-xext-dev x11proto-xinerama-dev
  xorg-sgml-doctools xtrans-dev zlib1g-dev
The following packages will be upgraded:
  libcairo-gobject2 libcairo2
```

One possible Makefile might look like this:

```
INC    := -I/usr/include/gtk-3.0 \
          -I/usr/include/glib-2.0 \
          -I/usr/include/pango-1.0 \
          -I/usr/include/cairo \
          -I/usr/include/atk-1.0
CC     := arm-linux-gnueabihf-gcc -march=armv7 $(INC)
LDFLAGS := -l gtk-3 -l gdk-3 -l gobject-2.0 -l glib-2.0 -l pango-1.0 -l pangocairo-1.0 -l gdk_pixbuf-2.0 -l cairo \
              -l png12 -l expat -l selinux -l pcre -l gio-2.0 -l z

all: gtk

gtk.o: gtk.c

gtk: gtk.o
    $(CC) -o $@ gtk.o $(LDFLAGS)

clean:
    rm gtk gtk.o
```

The following demonstration program realizes two standard dialogs (color and font), one user-defined dialog and a Cairo object with color gradients. The sourcecode is fully compatible to GTK+ 3.x but does not take some updates from version 3.2 respectively 3.4 into account. In the past GTK has too often gone through substantial changes.



Select background color

Hue: 240

Saturation: 100

Value: 100

Red: 0

Green: 0

Blue: 255

Color name: #0000FF

Select text font

Bitstream Charter Bold Italic
The quick brown fox jumps over the lazy dog.

Bitstream Charter Bold
The quick brown fox jumps over the lazy dog.

Bitstream Charter Italic
The quick brown fox jumps over the lazy dog.

Cantarell
The quick brown fox jumps over the lazy dog.

Cantarell Bold
The quick brown fox jumps over the lazy dog.

Cantarell Italic
The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog.

14

Choose a radius

30

Initialization has been completed.

```
/*****************************************************************
* GTK+ 3.x demonstration program
* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License.
*
* PLEASE NOTE: THE SAMPLE DOES NOT SUPPORT ALL NEW FEATURES
*                     FROM VERSION 3.2 RESP. 3.4
*****************************************************************/

#include <gtk/gtk.h>

static GtkWidget * create_color_area (GtkWidget *, gboolean);
static GtkWidget * create_font_area (GtkWidget *, gboolean);
static GtkWidget * create_radius_area (GtkWidget *, gboolean);
```

```
#define ID_COLOR              1
#define ID_FONT               2
#define ID_RADIUS             3
#define INITCOL               {0, 0, 0, 0xFFFF}      /* BLUE */
#define RADIUS_BORDER         20
#define RADIUS_SPACE          15
#define INITRADIUS            QUART(width)
#define ZERO                  0.0
#define ONE                   1.0
#define FRAME_SCALE           0.6
#define CAIRO_FONT            "Sans"
#define FONT_SIZE             32.0
#define INFO_PADDING          3
#define CENTER                0.5
#define FRAME_BOTTOM          12
#define FRAME_LEFT            59
#define FRAME_BORDER          34
#define MENU_PADDING          0
#define MAIN_BORDER           0

#define SAMPLE_DIMENS         600,500
#define TABLE_COLOR           1,2,1,4
#define TABLE_FONT            1,2,6,7
#define TABLE_RADIUS          1,2,9,12
#define BUTTON_COLOR          {3,2,1,1}
#define BUTTON_FONT           {3,6,1,1}
#define BUTTON_RADIUS         {3,10,1,1}
#define BUTTON_FRAME          {2,0,3,13}

#define RECT_COL1             "gold"
#define RECT_COL2             "darkorange"
#define SPHERE_COL1           "chartreuse"
#define SPHERE_COL2           "darkgreen"
#define CAIRO_TEXTCOL         "gray22"
#define TEXT_COLOR            "sienna"
#define TEXT_FONT             "Cantarell Bold 14"

#define HALF(x)               (x/2.0)
#define QUART(x)              (x/4.0)
#define TENTH(x)              (x/10.0)
#define FOURTEENTH(x)         (x*4.0/10.0)
#define COL(x)                ((double)x/65535.0)    /* Do not use BLACK with this statement */

#define MAIN_TITLE            "GTK+ 3.x DEMO"
#define COLOR_TITLE           "Select background color"
#define FONT_TITLE            "Select text font"
#define RADIUS_TITLE          "Choose a radius"
#define ABOUT_TITLE           "GTK+ 3.x Example"
#define CANCEL_TXT            "Cancel"
#define APPLY_TXT             "Apply"
#define ABOUT_VERSION         "Version 1.0"
#define ABOUT_COPYRGT         "(c) Kontron"
#define ABOUT_COMMENT         "GTK+ example for GTK version %d.%d.%d"
#define ABOUT_WEBSITE         "http://www.kontron.com"
#define ABOUT_ICONDIR         "usr/share/pixmaps/gnome-tigert.png"
```

```
#define CAIRO_TXT              "Cairo"
#define MSG_INIT               "Initialization has been completed."
#define MSG_COLOR              "Color button has been pressed."
#define MSG_FONT               "Font button has been pressed."
#define MSG_RADIUS             "Radius button has been pressed."
#define FRAME_TITLE            " Options "
#define MENU_FILE              "File"
#define MENU_OPTIONS           "Options"
#define MENU_HELP              "Help"
#define MENU_QUIT              "Quit"
#define MENU_COLOR             "Color"
#define MENU_FONT              "Font"
#define MENU_RADIUS            "Radius"
#define MENU_ABOUT             "About"
#define SAMPLE_TXT             "Sample Text"
#define BUTTON_COLTXT          "Color"
#define BUTTON_FONTTXT         "Font"
#define BUTTON_RADTXT          "Radius"

#define GET_COLOR_WIDGET       create_color_area ((GtkWidget *) NULL, FALSE)
#define GET_FONT_WIDGET        create_font_area ((GtkWidget *) NULL, FALSE)
#define GET_RADIUS_WIDGET      create_radius_area ((GtkWidget *) NULL, FALSE)


static void color_changed (GtkWidget *widget, GtkColorSelection *colorsel)
{
    GdkColor colortmp;

    gtk_color_selection_get_current_color (colorsel, &colortmp);
    gtk_widget_modify_bg (GET_COLOR_WIDGET, GTK_STATE_NORMAL, &colortmp);
}


static void color_dialog (void)
{
    GtkWidget *colorwgt;
    GtkColorSelection *colorsel;
    GtkColorSelectionDialog *colorseldlg;
    static GdkColor color = INITCOL;

    colorwgt = gtk_color_selection_dialog_new (COLOR_TITLE);
    colorseldlg = GTK_COLOR_SELECTION_DIALOG (colorwgt);
    colorsel = GTK_COLOR_SELECTION (gtk_color_selection_dialog_get_color_selection (colorseldlg));
    gtk_color_selection_set_previous_color (colorsel, &color);
    gtk_color_selection_set_current_color (colorsel, &color);
    gtk_color_selection_set_has_palette (colorsel, TRUE);

    g_signal_connect (G_OBJECT (colorsel), "color_changed", G_CALLBACK (color_changed), (gpointer) colorsel);
    if (gtk_dialog_run (GTK_DIALOG (colorwgt)) == GTK_RESPONSE_OK)
      gtk_color_selection_get_current_color (colorsel, &color);
    else
      gtk_widget_modify_bg (GET_COLOR_WIDGET, GTK_STATE_NORMAL, &color);
    gtk_widget_destroy (colorwgt);
}
```

```
static void font_dialog (void)
{
    GtkWidget *fontwgt;
    PangoContext *context;
    PangoFontDescription *fontdesc;
    gchar *tmp;
    static gchar font [64] = "\0";

    fontwgt = gtk_font_chooser_dialog_new (FONT_TITLE, (GtkWindow *) NULL);
    if (! g_ascii_isalpha (font[0]))
    {
      context = gtk_widget_create_pango_context (GET_FONT_WIDGET);
      fontdesc = pango_context_get_font_description (context);
      tmp = (gchar *) pango_font_description_to_string (fontdesc);
      g_stpcpy (font, tmp);
      g_free (tmp);
      pango_font_description_free (fontdesc);
    }
    gtk_font_chooser_set_font (GTK_FONT_CHOOSER (fontwgt), font);

    if (gtk_dialog_run (GTK_DIALOG (fontwgt)) == GTK_RESPONSE_OK)
    {
      tmp = gtk_font_chooser_get_font (GTK_FONT_CHOOSER (fontwgt));
      g_stpcpy (font, tmp);
      fontdesc = pango_font_description_from_string (tmp);
      gtk_widget_override_font (GET_FONT_WIDGET, fontdesc);
      pango_font_description_free (fontdesc);
      g_free (tmp);
    }
    gtk_widget_destroy (fontwgt);
}


static gint radius_dialog_value (gboolean update, gint val)
{
    static gint radius_value = 0;

    if (update)
      radius_value = val;
    return radius_value;
}


static void radius_dialog (void)
{
    GtkWidget *dialog,
              *content,
              *scale;
    gdouble width;
    gint value;
    dialog = gtk_dialog_new_with_buttons (RADIUS_TITLE, (GtkWindow *) NULL, 0,
                        CANCEL_TXT, GTK_RESPONSE_CANCEL, APPLY_TXT, GTK_RESPONSE_APPLY, NULL);
    content = gtk_dialog_get_content_area (GTK_DIALOG (dialog));

    gtk_container_set_border_width (GTK_CONTAINER (dialog), RADIUS_BORDER);
    gtk_box_set_spacing (GTK_BOX (content), RADIUS_SPACE);

    width = (gdouble) gtk_widget_get_allocated_width (GET_RADIUS_WIDGET);
    scale = gtk_scale_new_with_range (GTK_ORIENTATION_HORIZONTAL, ONE, HALF(width), TENTH(width));
    gtk_container_add (GTK_CONTAINER (content), scale);

    value = radius_dialog_value (FALSE, 0);
    gtk_range_set_value (GTK_RANGE (scale), (gdouble) value);
    gtk_widget_show_all (dialog);

    if (gtk_dialog_run (GTK_DIALOG (dialog)) == GTK_RESPONSE_APPLY)
    {
      value = (gint) gtk_range_get_value (GTK_RANGE (scale));
      radius_dialog_value (TRUE, value);
      gdk_window_invalidate_rect (gtk_widget_get_window (GET_RADIUS_WIDGET),
                                                  (GdkRectangle *) NULL, TRUE);
    }
    gtk_widget_destroy (dialog);
}


void about_dialog (GtkWidget *widget, gpointer data)
{
    GdkPixbuf *pixbuf;
    GtkWidget *aboutwgt;
    gchar version [64];

    aboutwgt = gtk_about_dialog_new ();
    gtk_about_dialog_set_program_name (GTK_ABOUT_DIALOG (aboutwgt), ABOUT_TITLE);
    gtk_about_dialog_set_version (GTK_ABOUT_DIALOG (aboutwgt), ABOUT_VERSION);
    gtk_about_dialog_set_copyright (GTK_ABOUT_DIALOG (aboutwgt), ABOUT_COPYRGT);
    g_sprintf (version, ABOUT_COMMENT, gtk_major_version, gtk_minor_version, gtk_micro_version);
    gtk_about_dialog_set_comments (GTK_ABOUT_DIALOG (aboutwgt), version);
    gtk_about_dialog_set_website (GTK_ABOUT_DIALOG (aboutwgt), ABOUT_WEBSITE);
    pixbuf = gdk_pixbuf_new_from_file (ABOUT_ICONDIR, (GError **) NULL);
    if (pixbuf != (GdkPixbuf *) NULL)
    {
      gtk_about_dialog_set_logo (GTK_ABOUT_DIALOG (aboutwgt), pixbuf);
      g_object_unref (pixbuf), pixbuf = (GdkPixbuf *) NULL;
    }
    gtk_dialog_run (GTK_DIALOG (aboutwgt));
    gtk_widget_destroy (aboutwgt);
}
```

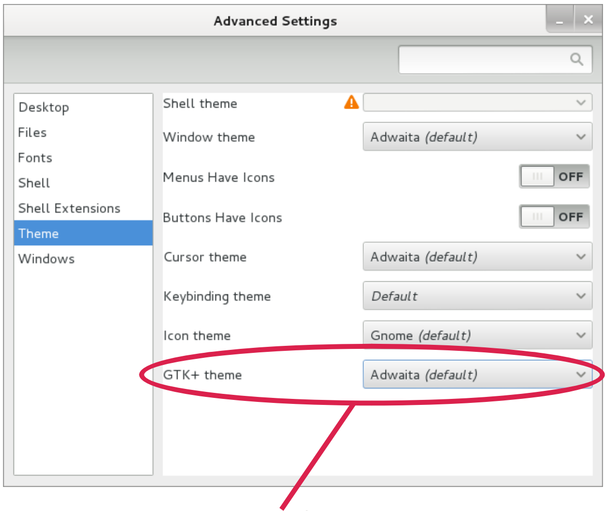
```
static gboolean cairo_draw_event (GtkWidget *parent, cairo_t *cr, gpointer data)
{
    GdkColor col;
    double width = (double) gtk_widget_get_allocated_width (parent),
            height,
            radius,
            radius0 = TENTH(width),
            radius1 = HALF(width);
    cairo_pattern_t *pat;

    height = (double) gtk_widget_get_allocated_height (parent);
    if (! (gint) (radius = (double) radius_dialog_value (FALSE, 0)))
      radius = (double) radius_dialog_value (TRUE, INITRADIUS);

    pat = cairo_pattern_create_linear (ZERO, ZERO, ZERO, height);
    gdk_color_parse (RECT_COL2, &col);
    cairo_pattern_add_color_stop_rgba (pat, ONE, COL(col.red), COL(col.green), COL(col.blue), ONE);
    gdk_color_parse (RECT_COL1, &col);
    cairo_pattern_add_color_stop_rgba (pat, ZERO, COL(col.red), COL(col.green), COL(col.blue), ONE);
    cairo_rectangle (cr, ZERO, ZERO, width, height);
    cairo_set_source (cr, pat);
    cairo_fill (cr);
    cairo_pattern_destroy (pat);

    pat = cairo_pattern_create_radial (HALF(width), FOURTEENTH(height), radius0, FOURTEENTH(width),
                                        FOURTEENTH(height), radius1);
    gdk_color_parse (SPHERE_COL1, &col);
    cairo_pattern_add_color_stop_rgba (pat, ZERO, COL(col.red), COL(col.green), COL(col.blue), ONE);
    gdk_color_parse (SPHERE_COL2, &col);
    cairo_pattern_add_color_stop_rgba (pat, ONE, COL(col.red), COL(col.green), COL(col.blue), ONE);
    cairo_set_source (cr, pat);
    cairo_arc (cr, HALF(width), HALF(height), radius, 0, 2 * G_PI);
    cairo_fill (cr);

    gdk_color_parse (CAIRO_TEXTCOL, &col);
    cairo_set_source_rgb (cr, COL(col.red), COL(col.green), COL(col.blue));
    cairo_select_font_face (cr, CAIRO_FONT, CAIRO_FONT_SLANT_NORMAL, CAIRO_FONT_WEIGHT_BOLD);
    cairo_set_font_size (cr, FONT_SIZE);
    cairo_move_to (cr, TENTH(width), HALF(height) + QUART(FONT_SIZE));
    cairo_show_text (cr, CAIRO_TXT);
    cairo_stroke (cr);

    cairo_pattern_destroy (pat);
    return FALSE;
}


static GtkWidget * create_environment (GtkWidget *win)
{
    GtkWidget *vb;

    vb = gtk_box_new (GTK_ORIENTATION_VERTICAL, 0);
    gtk_container_add (GTK_CONTAINER (win), vb);
    return vb;
}
```

```
static GtkWidget * create_infobar (GtkWidget *parent)
{
    GtkWidget *infobar;

    infobar = gtk_info_bar_new ();
    gtk_box_pack_end (GTK_BOX (parent), infobar, FALSE, FALSE, INFO_PADDING);
    gtk_info_bar_set_message_type (GTK_INFO_BAR (infobar), GTK_MESSAGE_INFO);
    return infobar;
}


static GtkWidget * create_infolabel (GtkWidget *parent)
{
    GtkWidget *msglabel,
                *msgarea;

    msglabel = gtk_label_new ((gchar *) NULL);
    msgarea = gtk_info_bar_get_content_area (GTK_INFO_BAR (parent));
    gtk_container_add (GTK_CONTAINER (msgarea), msglabel);
    return msglabel;
}


static void set_infobar_msg (GtkWidget *msglabel, gchar *msg)
{
    gtk_label_set_text (GTK_LABEL (msglabel), msg);
    gtk_widget_show_now (msglabel);
}


static void color_callback (GtkWidget *widget, gpointer data)
{
    if (data != (gpointer) NULL)
      set_infobar_msg ((GtkWidget *) data, MSG_COLOR);
    color_dialog ();
}


static void font_callback (GtkWidget *widget, gpointer data)
{
    if (data != (gpointer) NULL)
      set_infobar_msg ((GtkWidget *) data, MSG_FONT);
    font_dialog ();
}


static void radius_callback (GtkWidget *widget, gpointer data)
{
    if (data != (gpointer) NULL)
      set_infobar_msg ((GtkWidget *) data, MSG_RADIUS);
    radius_dialog ();
}
```

```
static void create_button (GtkWidget *parent, GtkWidget *msglabel, GdkRectangle rc, gint id, gchar *msg)
{
    GtkWidget *button;

    button = gtk_button_new_with_label (msg);
    gtk_table_attach (GTK_TABLE (parent), button, rc.x, rc.x + rc.width, rc.y, rc.y + rc.height,
                                                                GTK_FILL, GTK_FILL, 0, 0);
    switch (id)
    {
      case ID_COLOR:
           g_signal_connect (G_OBJECT (button), "clicked", G_CALLBACK (color_callback), (gpointer) msglabel);
           break;
      case ID_FONT:
           g_signal_connect (G_OBJECT (button), "clicked", G_CALLBACK (font_callback), (gpointer) msglabel);
           break;
      case ID_RADIUS:
           g_signal_connect (G_OBJECT (button), "clicked", G_CALLBACK (radius_callback), (gpointer) msglabel);
           break;
    }
}


static void create_button_frame (GtkWidget *parent, GdkRectangle rc)
{
    GtkWidget *frame,
              *align;

    frame = gtk_frame_new (FRAME_TITLE);
    gtk_frame_set_shadow_type (GTK_FRAME (frame), GTK_SHADOW_ETCHED_IN);
    gtk_frame_set_label_align (GTK_FRAME (frame), CENTER, CENTER);

    align = gtk_alignment_new (ZERO, ZERO, FRAME_SCALE, ONE);
    gtk_alignment_set_padding (GTK_ALIGNMENT (align), 0, FRAME_BOTTOM, FRAME_LEFT, 0);
    gtk_container_set_border_width (GTK_CONTAINER (frame), FRAME_BORDER);

    gtk_table_attach (GTK_TABLE (parent), align, rc.x, rc.x + rc.width, rc.y, rc.y + rc.height,
                                                                GTK_FILL, GTK_FILL, 0, 0);
    gtk_container_add (GTK_CONTAINER (align), frame);
}


static void create_menubar (GtkWidget *parent)
{
    GtkWidget *menubar,
              *filemenu, *optsmenu, *helpmenu,
              *file, *opts, *help,
              *quit, *color, *font, *radius, *about;

    menubar = gtk_menu_bar_new ();
    filemenu = gtk_menu_new ();
    optsmenu = gtk_menu_new ();
    helpmenu = gtk_menu_new ();
    file = gtk_menu_item_new_with_label (MENU_FILE);
    opts = gtk_menu_item_new_with_label (MENU_OPTIONS);
    help = gtk_menu_item_new_with_label (MENU_HELP);
    quit = gtk_menu_item_new_with_label (MENU_QUIT);
    color = gtk_menu_item_new_with_label (MENU_COLOR);
    font  = gtk_menu_item_new_with_label (MENU_FONT);
    radius = gtk_menu_item_new_with_label (MENU_RADIUS);
    about = gtk_menu_item_new_with_label (MENU_ABOUT);

    gtk_menu_item_set_submenu (GTK_MENU_ITEM (file), filemenu);
    gtk_menu_shell_append (GTK_MENU_SHELL (filemenu), quit);
    gtk_menu_shell_append (GTK_MENU_SHELL (menubar), file);

    gtk_menu_item_set_submenu (GTK_MENU_ITEM (opts), optsmenu);
    gtk_menu_shell_append (GTK_MENU_SHELL (optsmenu), color);
    gtk_menu_shell_append (GTK_MENU_SHELL (optsmenu), font);
    gtk_menu_shell_append (GTK_MENU_SHELL (optsmenu), radius);
    gtk_menu_shell_append (GTK_MENU_SHELL (menubar), opts);

    gtk_menu_item_set_submenu (GTK_MENU_ITEM (help), helpmenu);
    gtk_menu_shell_append (GTK_MENU_SHELL (helpmenu), about);
    gtk_menu_shell_append (GTK_MENU_SHELL (menubar), help);
    gtk_box_pack_start (GTK_BOX (parent), menubar, FALSE, FALSE, MENU_PADDING);

    g_signal_connect (G_OBJECT (quit), "activate", G_CALLBACK (gtk_main_quit), (gpointer) NULL);
    g_signal_connect (G_OBJECT (color), "activate", G_CALLBACK (color_callback), (gpointer) NULL);
    g_signal_connect (G_OBJECT (font), "activate", G_CALLBACK (font_callback), (gpointer) NULL);
    g_signal_connect (G_OBJECT (radius), "activate", G_CALLBACK (radius_callback), (gpointer) NULL);
    g_signal_connect (G_OBJECT (about), "activate", G_CALLBACK (about_dialog), (gpointer) NULL);
}


static GtkWidget * create_color_area (GtkWidget *parent, gboolean flag)
{
    GdkColor colorinit = INITCOL;
    static GtkWidget *colarea;

    if (flag)
    {
      colarea = gtk_drawing_area_new ();
      gtk_table_attach (GTK_TABLE (parent), colarea, TABLE_COLOR, GTK_FILL, GTK_FILL, 0, 0);
      gtk_widget_modify_bg (colarea, GTK_STATE_NORMAL, &colorinit);
      return (GtkWidget *) NULL;
    }
    else
      return colarea;
}
```

```
static GtkWidget * create_font_area (GtkWidget *parent, gboolean flag)
{
    GdkColor colorfont;
    PangoFontDescription *fontdesc;
    static GtkWidget *fontarea;

    if (flag)
    {
      fontarea = gtk_label_new (SAMPLE_TXT);
      gdk_color_parse (TEXT_COLOR, &colorfont);
      gtk_widget_modify_fg (fontarea, GTK_STATE_NORMAL, &colorfont);
      fontdesc = pango_font_description_from_string (TEXT_FONT);
      gtk_widget_override_font (fontarea, fontdesc);
      pango_font_description_free (fontdesc);
      gtk_table_attach (GTK_TABLE (parent), fontarea, TABLE_FONT, GTK_FILL, GTK_FILL, 0, 0);
      return (GtkWidget *) NULL;
    }
    else
      return fontarea;
}


static GtkWidget * create_radius_area (GtkWidget *parent, gboolean flag)
{
    static GtkWidget *cairoarea;

    if (flag)
    {
      cairoarea = gtk_drawing_area_new ();
      gtk_table_attach (GTK_TABLE (parent), cairoarea, TABLE_RADIUS, GTK_FILL, GTK_FILL, 0, 0);
      g_signal_connect (G_OBJECT (cairoarea), "draw", G_CALLBACK (cairo_draw_event), (gpointer) NULL);
      return (GtkWidget *) NULL;
    }
    else
      return cairoarea;
}


gint main (gint argc, gchar *argv[])
{
    GtkWidget *window,
              *vbox,
              *info,
              *msg,
              *table;
    GdkRectangle rc_color  = BUTTON_COLOR,
                 rc_font   = BUTTON_FONT,
                 rc_radius = BUTTON_RADIUS,
                 rc_frame  = BUTTON_FRAME;
    gtk_init (&argc, &argv);

    window = gtk_window_new (GTK_WINDOW_TOPLEVEL);
    gtk_window_set_default_size (GTK_WINDOW (window), SAMPLE_DIMENS);
    gtk_window_set_title (GTK_WINDOW (window), MAIN_TITLE);
    gtk_container_set_border_width (GTK_CONTAINER (window), MAIN_BORDER);
    g_signal_connect (G_OBJECT (window), "destroy", G_CALLBACK (gtk_main_quit), (gpointer) NULL);

    vbox = create_environment (window);
    create_menubar (vbox);
    info = create_infobar (vbox);
    msg = create_infolabel (info);
    set_infobar_msg (msg, MSG_INIT);

    table = gtk_table_new (5, 5, TRUE);
    gtk_container_add (GTK_CONTAINER (vbox), table);

    create_button (table, msg, rc_color, ID_COLOR, BUTTON_COLTXT);
    create_button (table, msg, rc_font, ID_FONT, BUTTON_FONTTXT);
    create_button (table, msg, rc_radius, ID_RADIUS, BUTTON_RADTXT);
    create_button_frame (table, rc_frame);

    create_color_area (table, TRUE);
    create_font_area (table, TRUE);
    create_radius_area (table, TRUE);

    gtk_widget_show_all (window);
    gtk_main ();
    return 0;
}
```

Some unpleasant surprises are waiting behind the pretty facade, for example the disappearing of the frame-widget with the standard theme Adwaita. Have a look on the dialog 'Applications → System Tools → Preferences → Advanced Settings'.



Shell theme
Window theme: Adwaita *(default)*
Menus Have Icons: OFF
Buttons Have Icons: OFF
Cursor theme: Adwaita *(default)*
Keybinding theme: *Default*
Icon theme: Gnome *(default)*
GTK+ theme: Adwaita *(default)*

Change only this entry

The existing themes do not support the frame-widget, therefore it is necessary to download some additional themes. A first option would be:

```
sudo apt-get install clearlooks-phenix-theme
```

This theme provides only one additional GTK3 entry named Clearlooks-Phenix and is largely compatible with 'Adwaita' but you can see the frame-widget in user-/root-mode. Another option insists in the use of:

```
sudo apt-get install gtk2-engines-murrine
```

This theme contains three additional GTK3 entries: Albatross (see the demonstration program figure), Bluebird and Greybird. If you select the 'Albatross' entry the frame-widget appears in root-mode but disappears in user-mode (depending on the use of gtk_table_attach or gtk_table_attach_defaults).
Other problems might not be discovered yet.

10.12.4 Qt^{TM} 4.8

Qt^{TM} is a cross-platform environment and UI framework with native C++ libraries, declarative UI language and tools in order to build complex applications for desktop, embedded and mobile computers. On an x86 Desktop PC using a cross compiler (Ubuntu^{TM} environment) you need some additional libraries (application dependent):

QtCore
QtGui
QtDBus
glib-2.0
png12
expat
pcre
uuid
z (libz)

Some Qt^{TM} dynamic libraries are available within the standard Debian^{TM} image (have a look at the directories /usr/lib/arm-linux-gnueabihf and /lib/arm-linux-gnueabihf), especially these five libraries: QtCore.so.4.8.2, QtGui.so.4.8.2, QtDBus.so.4.8.2, QtCLucene.so.4.8.2 and QtXml.so.4.8.2. The best way for development consists in the use of these libraries which should be copied from the target hardware to the cross compiler environment. Maybe a link creation or renaming of library files could be required. You also need some additional Include-files which typically are not available. For smaller applications the download of libqt4-dev on the x86 Desktop PC offers a sufficient way with

```
sudo  apt-get  install  libqt4-dev
```

Now you have access to the new Include parent directory /usr/include/qt4 and also to the important new base directory /usr/share/qt4. The following examples will not need the Qt^{TM} Creator or qmake, therefore a small manual adjustment is necessary. After the download of libqt4-dev the development environment uses x86 settings. By switching two definitions in the file

/usr/include/qt4/QtCore/qconfig.h

Qt^{TM} compiles for ARM environment and creates an ARM executable (changes in orange color):

```
/* Everything */

/* Qt Edition */
#ifndef QT_EDITION
# define QT_EDITION QT_EDITION_OPENSOURCE
#endif

/* Machine byte-order */
#define Q_BIG_ENDIAN 4321
#define Q_LITTLE_ENDIAN 1234
/*#define QT_BUILD_KEY "i386 linux g++-4 full-config"*/
/*#define QT_BUILD_KEY_COMPAT "i686 Linux g++-4 full-config"*/
#define QT_BUILD_KEY "arm linux g++-4 full-config"

#ifdef QT_BOOTSTRAPPED
#define Q_BYTE_ORDER Q_LITTLE_ENDIAN
#else
#define Q_BYTE_ORDER Q_LITTLE_ENDIAN
#endif
```

```
/* Machine Architecture */
#ifndef QT_BOOTSTRAPPED
/*# define QT_ARCH_I386*/
# define QT_ARCH_ARM
#else
/*# define QT_ARCH_I386*/
# define QT_ARCH_ARM
#endif
/* Compile time features */
#define QT_LARGEFILE_SUPPORT 64
#define QT_POINTER_SIZE 4
```

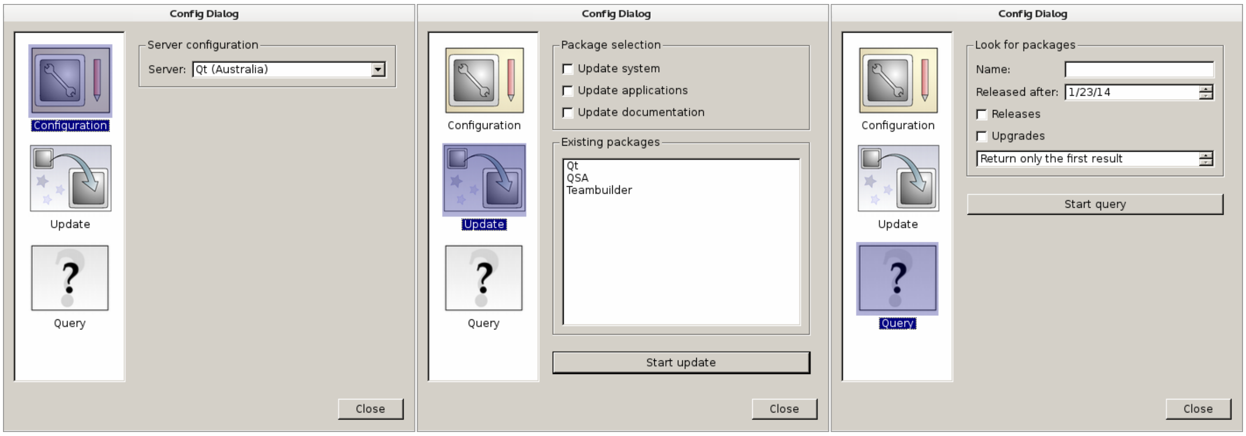
The Qt^{TM} project provides a set of examples on the following webpage

http://qt-project.org/doc/qt-4.8/all-examples.html

The first example uses the sourcecode from 'Dialog Examples/Config Dialog' and needs the creation of a subdirectory named 'images' for the PNG-files. One possible Makefile might look like this:

```
DEFINES  = -DQT_WEBKIT -DQT_NO_DEBUG -DQT_GUI_LIB -DQT_CORE_LIB -DQT_SHARED

CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDBus -l glib-2.0 -l png12 -l expat -l pcre -l uuid -l z

all: qt

qrc_configdialog.cpp: configdialog.qrc images/config.png images/update.png images/query.png
    /usr/bin/rcc -name configdialog configdialog.qrc -o qrc_configdialog.cpp

moc_configdialog.cpp: configdialog.h
    /usr/bin/moc-qt4 $(DEFINES) -I/usr/include/qt4 configdialog.h -o moc_configdialog.cpp

main.o: main.cpp
configdialog.o: configdialog.cpp
pages.o: pages.cpp

qt: main.o configdialog.o pages.o qrc_configdialog.o moc_configdialog.o
    $(CXX) -o $@ main.o configdialog.o pages.o qrc_configdialog.o moc_configdialog.o $(LDFLAGS)

clean:
    rm qt main.o configdialog.o pages.o qrc_configdialog.o moc_configdialog.o
```

The program produces the following screen output.



Config Dialog

Server configuration

Server: Qt (Australia)

Config Dialog

Package selection

□ Update system

□ Update applications

□ Update documentation

Existing packages

Qt
QSA
Teambuilder

Config Dialog

Look for packages

Name:

Released after: 1/23/14

□ Releases

□ Upgrades

Return only the first result

Some minor deviations from the original sourcecode are marked with orange color.

File configdialog.qrc:

```
<!DOCTYPE RCC><RCC version="1.0">
<qresource>
    <file>images/config.png</file>
    <file>images/query.png</file>
    <file>images/update.png</file>
</qresource>
</RCC>
```

File main.cpp:

```
/****************************************************************************
 ** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
 ** Contact: http://www.qt-project.org/legal
 ** This file is part of the examples of the Qt Toolkit.
 **
 ** $QT_BEGIN_LICENSE:BSD$
 ** You may use this file under the terms of the BSD license as follows:
 ** "Redistribution and use in source and binary forms, with or without
 ** modification, are permitted provided that the following conditions are
 ** met:
 **   * Redistributions of source code must retain the above copyright
 **     notice, this list of conditions and the following disclaimer.
 **   * Redistributions in binary form must reproduce the above copyright
 **     notice, this list of conditions and the following disclaimer in
 **     the documentation and/or other materials provided with the distribution.
 **   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
 **     of its contributors may be used to endorse or promote products derived
 **     from this software without specific prior written permission.
 **
 ** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 ** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 ** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 ** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 ** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 ** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 ** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 ** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 ** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 ** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 ** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
 ** $QT_END_LICENSE$
****************************************************************************/

#include <QtGui/QApplication>
#include "configdialog.h"

int main (int argc, char *argv[])
{
    Q_INIT_RESOURCE (configdialog);

    QApplication app (argc, argv);
    ConfigDialog dialog;
    return dialog.exec();
}
```

File configdialog.h (header with Qt^{TM} license see 'main.cpp'):

```
#ifndef CONFIGDIALOG_H
#define CONFIGDIALOG_H

#include <QtGui/QDialog>

class QListWidget;
class QListWidgetItem;
class QStackedWidget;

class ConfigDialog : public QDialog
{
    Q_OBJECT

public:
    ConfigDialog ();

public slots:
    void changePage (QListWidgetItem *current, QListWidgetItem *previous);

private:
    void createIcons ();

    QListWidget *contentsWidget;
    QStackedWidget *pagesWidget;
};

#endif
```

File configdialog.cpp (header with Qt^{TM} license see 'main.cpp'):

```
#include <QtGui/QtGui>
#include "configdialog.h"
#include "pages.h"

ConfigDialog::ConfigDialog ()
{
    contentsWidget = new QListWidget;
    contentsWidget->setViewMode (QListView::IconMode);
    contentsWidget->setIconSize (QSize(96, 84));
    contentsWidget->setMovement (QListView::Static);
    contentsWidget->setMaximumWidth (128);
    contentsWidget->setSpacing (12);

    pagesWidget = new QStackedWidget;
    pagesWidget->addWidget (new ConfigurationPage);
    pagesWidget->addWidget (new UpdatePage);
    pagesWidget->addWidget (new QueryPage);

    QPushButton *closeButton = new QPushButton (tr ("Close"));

    createIcons ();
    contentsWidget->setCurrentRow (0);

    connect (closeButton, SIGNAL (clicked ()), this, SLOT (close ()));
    QHBoxLayout *horizontalLayout = new QHBoxLayout;
    horizontalLayout->addWidget (contentsWidget);
    horizontalLayout->addWidget (pagesWidget, 1);

    QHBoxLayout *buttonsLayout = new QHBoxLayout;
    buttonsLayout->addStretch (1);
    buttonsLayout->addWidget (closeButton);

    QVBoxLayout *mainLayout = new QVBoxLayout;
    mainLayout->addLayout (horizontalLayout);
    mainLayout->addStretch (1);
    mainLayout->addSpacing (12);
    mainLayout->addLayout (buttonsLayout);
    setLayout (mainLayout);

    setWindowTitle (tr ("Config Dialog"));
}


void ConfigDialog::createIcons ()
{
    QListWidgetItem *configButton = new QListWidgetItem (contentsWidget);
    configButton->setIcon (QIcon (":/images/config.png"));
    configButton->setText (tr ("Configuration"));
    configButton->setTextAlignment (Qt::AlignHCenter);
    configButton->setFlags (Qt::ItemIsSelectable | Qt::ItemIsEnabled);

    QListWidgetItem *updateButton = new QListWidgetItem (contentsWidget);
    updateButton->setIcon (QIcon (":/images/update.png"));
    updateButton->setText (tr ("Update"));
    updateButton->setTextAlignment (Qt::AlignHCenter);
    updateButton->setFlags (Qt::ItemIsSelectable | Qt::ItemIsEnabled);

    QListWidgetItem *queryButton = new QListWidgetItem (contentsWidget);
    queryButton->setIcon (QIcon (":/images/query.png"));
    queryButton->setText (tr ("Query"));
    queryButton->setTextAlignment (Qt::AlignHCenter);
    queryButton->setFlags (Qt::ItemIsSelectable | Qt::ItemIsEnabled);

    connect (contentsWidget, SIGNAL (currentItemChanged (QListWidgetItem*, QListWidgetItem*)),
                     this, SLOT (changePage (QListWidgetItem*, QListWidgetItem*)));
}


void ConfigDialog::changePage (QListWidgetItem *current, QListWidgetItem *previous)
{
    if (!current)
      current = previous;

    pagesWidget->setCurrentIndex (contentsWidget->row (current));
}
```

File pages.h (header with Qt™ license see 'main.cpp'):

```
#ifndef PAGES_H
#define PAGES_H

#include <QtGui/QWidget>

class ConfigurationPage : public QWidget
{
public:
    ConfigurationPage (QWidget *parent = 0);
};

class QueryPage : public QWidget
{
public:
    QueryPage (QWidget *parent = 0);
};

class UpdatePage : public QWidget
{
public:
    UpdatePage (QWidget *parent = 0);
};

#endif
```

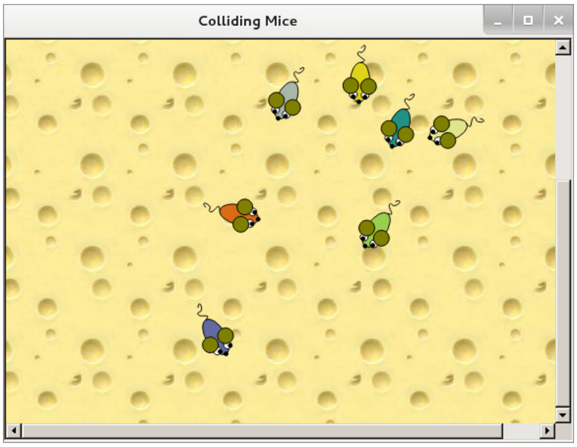
File pages.cpp (header with Qt™ license see 'main.cpp'):

```
#include <QtGui/QtGui>
#include "pages.h"

ConfigurationPage::ConfigurationPage (QWidget *parent)
        : QWidget (parent)
{
    QGroupBox *configGroup = new QGroupBox (tr ("Server configuration"));

    QLabel *serverLabel = new QLabel (tr ("Server:"));
    QComboBox *serverCombo = new QComboBox;
    serverCombo->addItem (tr ("Qt (Australia)"));
    serverCombo->addItem (tr ("Qt (Germany)"));
    serverCombo->addItem (tr ("Qt (Norway)"));
    serverCombo->addItem (tr ("Qt (People's Republic of China)"));
    serverCombo->addItem (tr ("Qt (USA)"));

    QHBoxLayout *serverLayout = new QHBoxLayout;
    serverLayout->addWidget (serverLabel);
    serverLayout->addWidget (serverCombo);

    QVBoxLayout *configLayout = new QVBoxLayout;
    configLayout->addLayout (serverLayout);
    configGroup->setLayout (configLayout);
    QVBoxLayout *mainLayout = new QVBoxLayout;
    mainLayout->addWidget (configGroup);
    mainLayout->addStretch (1);
    setLayout (mainLayout);
}


UpdatePage::UpdatePage (QWidget *parent)
        : QWidget (parent)
{
    QGroupBox *updateGroup = new QGroupBox (tr ("Package selection"));
    QCheckBox *systemCheckBox = new QCheckBox (tr ("Update system"));
    QCheckBox *appsCheckBox = new QCheckBox (tr ("Update applications"));
    QCheckBox *docsCheckBox = new QCheckBox (tr ("Update documentation"));

    QGroupBox *packageGroup = new QGroupBox (tr ("Existing packages"));

    QListWidget *packageList = new QListWidget;
    QListWidgetItem *qtItem = new QListWidgetItem (packageList);
    qtItem->setText (tr ("Qt"));
    QListWidgetItem *qsaItem = new QListWidgetItem (packageList);
    qsaItem->setText (tr ("QSA"));
    QListWidgetItem *teamBuilderItem = new QListWidgetItem (packageList);
    teamBuilderItem->setText (tr ("Teambuilder"));

    QPushButton *startUpdateButton = new QPushButton (tr ("Start update"));

    QVBoxLayout *updateLayout = new QVBoxLayout;
    updateLayout->addWidget (systemCheckBox);
    updateLayout->addWidget (appsCheckBox);
    updateLayout->addWidget (docsCheckBox);
    updateGroup->setLayout (updateLayout);

    QVBoxLayout *packageLayout = new QVBoxLayout;
    packageLayout->addWidget (packageList);
    packageGroup->setLayout (packageLayout);

    QVBoxLayout *mainLayout = new QVBoxLayout;
    mainLayout->addWidget (updateGroup);
    mainLayout->addWidget (packageGroup);
    mainLayout->addSpacing (12);
    mainLayout->addWidget (startUpdateButton);
    mainLayout->addStretch (1);
    setLayout (mainLayout);
}


QueryPage::QueryPage (QWidget *parent)
        : QWidget (parent)
{
    QGroupBox *packagesGroup = new QGroupBox (tr ("Look for packages"));

    QLabel *nameLabel = new QLabel (tr ("Name:"));
    QLineEdit *nameEdit = new QLineEdit;

    QLabel *dateLabel = new QLabel (tr ("Released after:"));
    QDateTimeEdit *dateEdit = new QDateTimeEdit (QDate::currentDate ());
    QCheckBox *releasesCheckBox = new QCheckBox (tr ("Releases"));
    QCheckBox *upgradesCheckBox = new QCheckBox (tr ("Upgrades"));
    QSpinBox *hitsSpinBox = new QSpinBox;
    hitsSpinBox->setPrefix (tr ("Return up to "));
    hitsSpinBox->setSuffix (tr (" results"));
    hitsSpinBox->setSpecialValueText (tr ("Return only the first result"));
    hitsSpinBox->setMinimum (1);
    hitsSpinBox->setMaximum (100);
    hitsSpinBox->setSingleStep (10);

    QPushButton *startQueryButton = new QPushButton (tr ("Start query"));
    QGridLayout *packagesLayout = new QGridLayout;
    packagesLayout->addWidget (nameLabel, 0, 0);
    packagesLayout->addWidget (nameEdit, 0, 1);
    packagesLayout->addWidget (dateLabel, 1, 0);
    packagesLayout->addWidget (dateEdit, 1, 1);
    packagesLayout->addWidget (releasesCheckBox, 2, 0);
    packagesLayout->addWidget (upgradesCheckBox, 3, 0);
    packagesLayout->addWidget (hitsSpinBox, 4, 0, 1, 2);
    packagesGroup->setLayout (packagesLayout);
    QVBoxLayout *mainLayout = new QVBoxLayout;
    mainLayout->addWidget (packagesGroup);
    mainLayout->addSpacing (12);
    mainLayout->addWidget (startQueryButton);
    mainLayout->addStretch (1);
    setLayout (mainLayout);
}
```

The second example uses the sourcecode from 'Graphics View Examples/Colliding Mice' and needs the creation of a subdirectory named 'images' for the JPG-file. Remark: the JPG-file does not exist - you can download it with the 'qt4-demos' package. One possible Makefile might look like this:

```
CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDBus -l glib-2.0 -l png12 -l expat -l pcre -l uuid -l z
all: qt
qrc_mice.cpp: mice.qrc images/cheese.jpg
    /usr/bin/rcc -name mice mice.qrc -o qrc_mice.cpp
main.o: main.cpp
mouse.o: mouse.cpp
qt: main.o mouse.o qrc_mice.o
    $(CXX) -o $@ main.o mouse.o qrc_mice.o $(LDFLAGS)
clean:
    rm qt main.o mouse.o qrc_mice.o
```

and the associated screen output:



File mice.qrc:

```
<RCC>
<qresource prefix="/" >
    <file>images/cheese.jpg</file>
</qresource>
</RCC>
```

File main.cpp:

```
/****************************************************************************
 ** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
 ** Contact: http://www.qt-project.org/legal
 ** This file is part of the examples of the Qt Toolkit.
 **
 ** $QT_BEGIN_LICENSE:BSD$
 ** You may use this file under the terms of the BSD license as follows:
 ** "Redistribution and use in source and binary forms, with or without
 ** modification, are permitted provided that the following conditions are
 ** met:
 **   * Redistributions of source code must retain the above copyright
 **     notice, this list of conditions and the following disclaimer.
 **   * Redistributions in binary form must reproduce the above copyright
 **     notice, this list of conditions and the following disclaimer in
 **     the documentation and/or other materials provided with the distribution.
 **   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
 **     of its contributors may be used to endorse or promote products derived
 **     from this software without specific prior written permission.
 **
 ** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 ** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 ** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 ** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 ** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 ** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 ** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 ** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 ** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 ** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 ** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
 ** $QT_END_LICENSE$
****************************************************************************/
```

```
#include <QtGui/QtGui>
#include <math.h>
#include "mouse.h"

static const int MouseCount = 7;

int main (int argc, char **argv)
{
    QApplication app (argc, argv);
    qsrand (QTime (0,0,0).secsTo (QTime::currentTime ()));

    QGraphicsScene scene;
    scene.setSceneRect (-300, -300, 600, 600);
    scene.setItemIndexMethod (QGraphicsScene::NoIndex);

    for (int i = 0; i < MouseCount; ++i)
    {
      Mouse *mouse = new Mouse;
      mouse->setPos (::sin ((i * 6.28) / MouseCount) * 200,
                     ::cos ((i * 6.28) / MouseCount) * 200);
      scene.addItem (mouse);
    }

    QGraphicsView view (&scene);
    view.setRenderHint (QPainter::Antialiasing);
    view.setBackgroundBrush (QPixmap (":/images/cheese.jpg"));
    view.setCacheMode (QGraphicsView::CacheBackground);
    view.setViewportUpdateMode (QGraphicsView::BoundingRectViewportUpdate);
    view.setDragMode (QGraphicsView::ScrollHandDrag);
    view.setWindowTitle (QT_TRANSLATE_NOOP (QGraphicsView, "Colliding Mice"));
#if defined (Q_WS_S60) || defined (Q_WS_MAEMO_5) || defined (Q_WS_SIMULATOR)
    view.showMaximized ();
#else
    view.resize (400, 300);
    view.show ();
#endif

    QTimer timer;
    QObject::connect (&timer, SIGNAL (timeout ()), &scene, SLOT (advance ()));
    timer.start (1000 / 33);

    return app.exec ();
}
```

File mouse.h (header with Qt^{TM} license see 'main.cpp'):

```
#ifndef MOUSE_H
#define MOUSE_H

#include <QtGui/QGraphicsItem>

class Mouse : public QGraphicsItem
{
public:
    Mouse ();
    QRectF boundingRect () const;
    QPainterPath shape () const;
    void paint (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget);

protected:
    void advance (int step);

private:
    qreal angle;
    qreal speed;
    qreal mouseEyeDirection;
    QColor color;
};

#endif
```

File mouse.cpp (header with Qt^{TM} license see 'main.cpp'):

```
#include <QtGui/QGraphicsScene>
#include <QtGui/QPainter>
#include <QtGui/QStyleOption>
#include <math.h>
#include "mouse.h"

static const double Pi = 3.14159265358979323846264338327950288419717;
static double TwoPi = 2.0 * Pi;

static qreal normalizeAngle (qreal angle)
{
    while (angle < 0)
      angle += TwoPi;
    while (angle > TwoPi)
      angle -= TwoPi;
    return angle;
}


Mouse::Mouse ()
      : angle (0), speed (0), mouseEyeDirection (0), color (qrand () % 256, qrand () % 256, qrand () % 256)
{
    setRotation (qrand () % (360 * 16));
}


QRectF Mouse::boundingRect () const
{
    qreal adjust = 0.5;
    return QRectF (-18 - adjust, -22 - adjust, 36 + adjust, 60 + adjust);
}
```

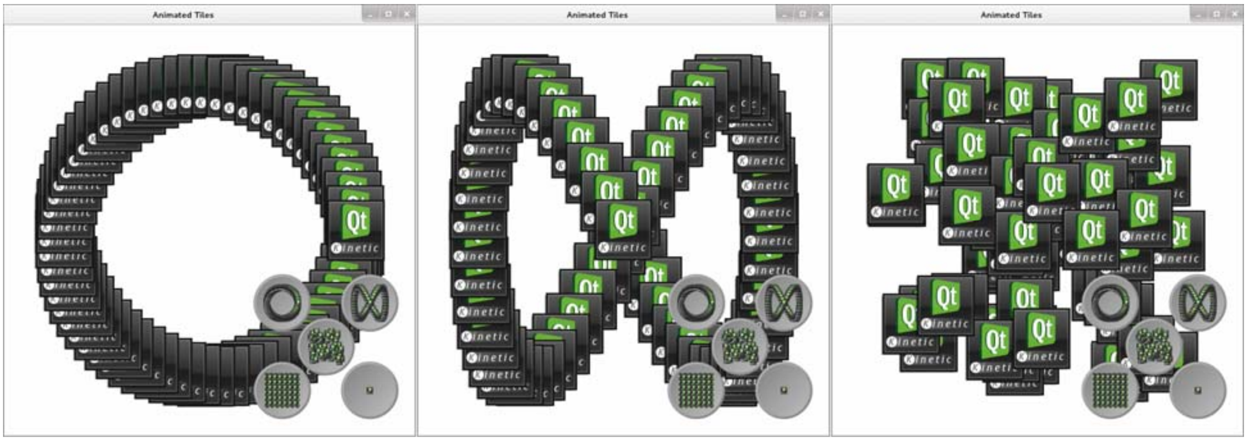
```
QPainterPath Mouse::shape () const
{
    QPainterPath path;
    path.addRect (-10, -20, 20, 40);
    return path;
}


void Mouse::paint (QPainter *painter, const QStyleOptionGraphicsItem *, QWidget *)
{
    // Body
    painter->setBrush (color);
    painter->drawEllipse (-10, -20, 20, 40);

    // Eyes
    painter->setBrush (Qt::white);
    painter->drawEllipse (-10, -17, 8, 8);
    painter->drawEllipse (2, -17, 8, 8);

    // Nose
    painter->setBrush (Qt::black);
    painter->drawEllipse (QRectF (-2, -22, 4, 4));

    // Pupils
    painter->drawEllipse (QRectF (-8.0 + mouseEyeDirection, -17, 4, 4));
    painter->drawEllipse (QRectF (4.0 + mouseEyeDirection, -17, 4, 4));

    // Ears
    painter->setBrush (scene ()->collidingItems (this).isEmpty () ? Qt::darkYellow : Qt::red);
    painter->drawEllipse (-17, -12, 16, 16);
    painter->drawEllipse (1, -12, 16, 16);

    // Tail
    QPainterPath path (QPointF (0, 20));
    path.cubicTo (-5, 22, -5, 22, 0, 25);
    path.cubicTo (5, 27, 5, 32, 0, 30);
    path.cubicTo (-5, 32, -5, 42, 0, 35);
    painter->setBrush (Qt::NoBrush);
    painter->drawPath (path);
}


void Mouse::advance (int step)
{
    if (! step)
      return;

    // Don't move too far away
    QLineF lineToCenter (QPointF (0, 0), mapFromScene (0, 0));
    if (lineToCenter.length () > 150)
    {
      qreal angleToCenter = ::acos (lineToCenter.dx () / lineToCenter.length ());
      if (lineToCenter.dy () < 0)
        angleToCenter = TwoPi - angleToCenter;
      angleToCenter = normalizeAngle ((Pi - angleToCenter) + Pi / 2);
    if (angleToCenter < Pi && angleToCenter > Pi / 4)
    {
        // Rotate left
        angle += (angle < -Pi / 2) ? 0.25 : -0.25;
    }
    else if (angleToCenter >= Pi && angleToCenter < (Pi + Pi / 2 + Pi / 4))
    {
        // Rotate right
        angle += (angle < Pi / 2) ? 0.25 : -0.25;
    }
}
else if (::sin (angle) < 0)
{
    angle += 0.25;
}
else if (::sin(angle) > 0)
{
    angle -= 0.25;
}

// Try not to crash with any other mice
QList<QGraphicsItem *> dangerMice = scene ()->items (QPolygonF ()
                                                   << mapToScene (0, 0)
                                                   << mapToScene (-30, -50)
                                                   << mapToScene (30, -50));
foreach (QGraphicsItem *item, dangerMice)
{
    if (item == this)
        continue;

    QLineF lineToMouse (QPointF (0, 0), mapFromItem (item, 0, 0));
    qreal angleToMouse = ::acos (lineToMouse.dx () / lineToMouse.length ());
    if (lineToMouse.dy () < 0)
        angleToMouse = TwoPi - angleToMouse;
    angleToMouse = normalizeAngle ((Pi - angleToMouse) + Pi / 2);

    if (angleToMouse >= 0 && angleToMouse < Pi / 2)
    {
        // Rotate right
        angle += 0.5;
    }
    else if (angleToMouse <= TwoPi && angleToMouse > (TwoPi - Pi / 2))
    {
        // Rotate left
        angle -= 0.5;
    }
}
```

```
    // Add some random movement
    if (dangerMice.size () > 1 && (qrand () % 10) == 0)
    {
      if (qrand () % 1)
        angle += (qrand () % 100) / 500.0;
      else
        angle -= (qrand () % 100) / 500.0;
    }

    speed += (-50 + qrand () % 100) / 100.0;

    qreal dx = ::sin (angle) * 10;
    mouseEyeDirection = (qAbs (dx / 5) < 1) ? 0 : dx / 5;

    setRotation (rotation () + dx);
    setPos (mapToParent (0, -(3 + sin (speed) * 3)));
}
```

The third example refers to the sourcecode from 'Animation Framework Examples/Animated Tiles' and needs also a subdirectory named 'images' for the PNG-files. Remark: the background JPG-file is absent but you can download it with the 'qt4-demos' package - however you can do without it.
This example has to be changed on a large scale because the output from the MOC (Meta-Object Compiler) is required. Create a new 'main.h' file, copy all class definitions into this file and delete the definitions in 'main.cpp'.
One possible Makefile might look like this:

```
DEFINES  = -DQT_WEBKIT -DQT_NO_DEBUG -DQT_GUI_LIB -DQT_CORE_LIB -DQT_SHARED

CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDBus -l glib-2.0 -l png12 -l expat -l pcre -l uuid -l z

all: qt

qrc_animatedtiles.cpp: animatedtiles.qrc images/centered.png images/ellipse.png \
                       images/figure8.png images/kinetic.png images/random.png images/tile.png
    /usr/bin/rcc -name animatedtiles animatedtiles.qrc -o qrc_animatedtiles.cpp

moc_main.cpp: main.h
    /usr/bin/moc-qt4 $(DEFINES) -I/usr/include/qt4 main.h -o moc_main.cpp

main.o: main.cpp moc_main.cpp

qt: main.o qrc_animatedtiles.o
    $(CXX) -o $@ main.o qrc_animatedtiles.o $(LDFLAGS)

clean:
    rm qt main.o qrc_animatedtiles.o
```

Three out of five views look as follows:



File animatedtiles.qrc:

```
<!DOCTYPE RCC><RCC version="1.0">
<qresource>
    <file>images/centered.png</file>
    <file>images/ellipse.png</file>
    <file>images/figure8.png</file>
    <file>images/kinetic.png</file>
    <file>images/random.png</file>
    <file>images/tile.png</file>
</qresource>
</RCC>
```

File main.cpp:

```
/****************************************************************************
 ** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
 ** Contact: http://www.qt-project.org/legal
 ** This file is part of the examples of the Qt Toolkit.
 **
 ** $QT_BEGIN_LICENSE:BSD$
 ** You may use this file under the terms of the BSD license as follows:
 ** "Redistribution and use in source and binary forms, with or without
 ** modification, are permitted provided that the following conditions are
 ** met:
 **   * Redistributions of source code must retain the above copyright
 **     notice, this list of conditions and the following disclaimer.
 **   * Redistributions in binary form must reproduce the above copyright
 **     notice, this list of conditions and the following disclaimer in
 **     the documentation and/or other materials provided with the distribution.
 **   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
 **     of its contributors may be used to endorse or promote products derived
 **     from this software without specific prior written permission.
 **
 ** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 ** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 ** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 ** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 ** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 ** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 ** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 ** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 ** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 ** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
 ** $QT_END_LICENSE$
****************************************************************************/

#include <QtGui/QtGui>
#include "moc_main.cpp"

int main (int argc, char **argv)
{
    Q_INIT_RESOURCE (animatedtiles);

    QApplication app (argc, argv);

    QPixmap kineticPix (":/images/kinetic.png");
    /* QPixmap bgPix (":/images/Time-For-Lunch-2.jpg"); */

    QGraphicsScene scene (-350, -350, 700, 700);

    QList<Pixmap *> items;
    for (int i = 0; i < 64; ++i)
    {
      Pixmap *item = new Pixmap (kineticPix);
      item->setOffset (-kineticPix.width () / 2, -kineticPix.height () / 2);
      item->setZValue (i);
      items << item;
      scene.addItem (item);
    }

    // Buttons
    QGraphicsItem *buttonParent = new QGraphicsRectItem;
    Button *ellipseButton = new Button (QPixmap (":/images/ellipse.png"), buttonParent);
    Button *figure8Button = new Button (QPixmap (":/images/figure8.png"), buttonParent);
    Button *randomButton = new Button (QPixmap (":/images/random.png"), buttonParent);
    Button *tiledButton = new Button (QPixmap (":/images/tile.png"), buttonParent);
    Button *centeredButton = new Button (QPixmap (":/images/centered.png"), buttonParent);

    ellipseButton->setPos (-100, -100);
    figure8Button->setPos (100, -100);
    randomButton->setPos (0, 0);
    tiledButton->setPos (-100, 100);
    centeredButton->setPos (100, 100);

    scene.addItem (buttonParent);
    buttonParent->scale (0.75, 0.75);
    buttonParent->setPos (200, 200);
    buttonParent->setZValue (65);

    // States
    QState *rootState = new QState;
    QState *ellipseState = new QState (rootState);
    QState *figure8State = new QState (rootState);
    QState *randomState = new QState (rootState);
    QState *tiledState = new QState (rootState);
    QState *centeredState = new QState (rootState);
    // Values
    for (int i = 0; i < items.count (); ++i)
    {
      Pixmap *item = items.at (i);
      // Ellipse
      ellipseState->assignProperty (item, "pos", QPointF (cos ((i / 63.0) * 6.28) * 250,
                                                          sin ((i / 63.0) * 6.28) * 250));

      // Figure 8
      figure8State->assignProperty (item, "pos", QPointF (sin ((i / 63.0) * 6.28) * 250,
                                                          sin (((i * 2) / 63.0) * 6.28) * 250));

      // Random
      randomState->assignProperty (item, "pos", QPointF (-250 + qrand () % 500,
                                                          -250 + qrand () % 500));

      // Tiled
      tiledState->assignProperty (item, "pos", QPointF (((i % 8) - 4) * kineticPix.width () + kineticPix.width () / 2,
                                               ((i / 8) - 4) * kineticPix.height () + kineticPix.height () / 2));

      // Centered
      centeredState->assignProperty (item, "pos", QPointF ());
    }

    // Ui
    View *view = new View (&scene);
    view->setWindowTitle (QT_TRANSLATE_NOOP (QGraphicsView, "Animated Tiles"));
    view->setViewportUpdateMode (QGraphicsView::BoundingRectViewportUpdate);
    /* view->setBackgroundBrush (bgPix); */
    view->setCacheMode (QGraphicsView::CacheBackground);
    view->setRenderHints (QPainter::Antialiasing | QPainter::SmoothPixmapTransform);
#ifdef Q_OS_SYMBIAN
    view->showMaximized ();
#else
    view->show();
#endif

    QStateMachine states;
    states.addState (rootState);
    states.setInitialState (rootState);
    rootState->setInitialState (centeredState);

    QParallelAnimationGroup *group = new QParallelAnimationGroup;
    for (int i = 0; i < items.count (); ++i)
    {
      QPropertyAnimation *anim = new QPropertyAnimation (items[i], "pos");
      anim->setDuration (750 + i * 25);
      anim->setEasingCurve (QEasingCurve::InOutBack);
      group->addAnimation (anim);
    }
    QAbstractTransition *trans = rootState->addTransition (ellipseButton, SIGNAL (pressed ()), ellipseState);
    trans->addAnimation (group);

    trans = rootState->addTransition (figure8Button, SIGNAL (pressed ()), figure8State);
    trans->addAnimation (group);
    trans = rootState->addTransition (randomButton, SIGNAL (pressed ()), randomState);
    trans->addAnimation (group);

    trans = rootState->addTransition (tiledButton, SIGNAL (pressed ()), tiledState);
    trans->addAnimation (group);

    trans = rootState->addTransition (centeredButton, SIGNAL (pressed ()), centeredState);
    trans->addAnimation (group);

    QTimer timer;
    timer.start (125);
    timer.setSingleShot (true);
    trans = rootState->addTransition (&timer, SIGNAL (timeout ()), ellipseState);
    trans->addAnimation (group);

    states.start ();

#ifdef QT_KEYPAD_NAVIGATION
    QApplication::setNavigationMode (Qt::NavigationModeCursorAuto);
#endif
    return app.exec ();
}
```

New file main.h (header with Qt™ license see 'main.cpp'):

```
#include <QtGui/QtGui>
/* #include <QtCore/qstate.h> */

class Pixmap : public QObject, public QGraphicsPixmapItem
{
    Q_OBJECT
    Q_PROPERTY (QPointF pos READ pos WRITE setPos)
public:
    Pixmap (const QPixmap &pix)
           : QObject (), QGraphicsPixmapItem (pix)
    {
      setCacheMode (DeviceCoordinateCache);
    }
};


class Button : public QGraphicsWidget
{
    Q_OBJECT
public:
    Button (const QPixmap &pixmap, QGraphicsItem *parent = 0)
           : QGraphicsWidget (parent), _pix (pixmap)
    {
      setAcceptHoverEvents (true);
      setCacheMode (DeviceCoordinateCache);
    }

    QRectF boundingRect () const
    {
      return QRectF (-65, -65, 130, 130);
    }
    QPainterPath shape () const
    {
      QPainterPath path;
      path.addEllipse (boundingRect ());
      return path;
    }

    void paint (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *)
    {
      bool down = option->state & QStyle::State_Sunken;
      QRectF r = boundingRect ();
      QLinearGradient grad (r.topLeft (), r.bottomRight ());
      grad.setColorAt (down ? 1 : 0, option->state & QStyle::State_MouseOver ? Qt::white : Qt::lightGray);
      grad.setColorAt (down ? 0 : 1, Qt::darkGray);
      painter->setPen (Qt::darkGray);
      painter->setBrush (grad);
      painter->drawEllipse (r);
      QLinearGradient grad2 (r.topLeft (), r.bottomRight ());
      grad.setColorAt (down ? 1 : 0, Qt::darkGray);
      grad.setColorAt (down ? 0 : 1, Qt::lightGray);
      painter->setPen (Qt::NoPen);
      painter->setBrush (grad);
      if (down)
        painter->translate (2, 2);
      painter->drawEllipse (r.adjusted (5, 5, -5, -5));
      painter->drawPixmap (-_pix.width () / 2, -_pix.height () / 2, _pix);
    }

signals:
    void pressed ();

protected:
    void mousePressEvent (QGraphicsSceneMouseEvent *)
    {
      emit pressed ();
      update ();
    }

    void mouseReleaseEvent (QGraphicsSceneMouseEvent *)
    {
      update ();
    }

private:
    QPixmap _pix;
};
```

```
class View : public QGraphicsView
{
public:
    View (QGraphicsScene *scene) : QGraphicsView (scene) { }

protected:
    void resizeEvent (QResizeEvent *event)
    {
        QGraphicsView::resizeEvent (event);
        fitInView (sceneRect (), Qt::KeepAspectRatio);
    }
};
```

The fourth example handles the OpenGL® interface from 'OpenGL Examples/Grabber'. However, this requires additional steps. The standard Debian™ image contains the OpenGL® library named 'libGL.so.1' but you need also the Qt OpenGL® library named 'libQtOpenGL.so.4.8.2'. In order to do that download the Qt OpenGL® environment on the target hardware with

```
sudo apt-get install libqt4-opengl-dev
```

The following download message appears (extract):

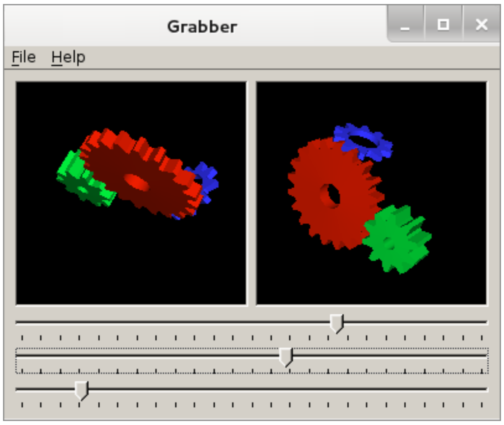
```
The following extra packages will be installed:
  libgl1-mesa-dev libglu1-mesa-dev libmysqlclient18 libpthread-stubs0
  libpthread-stubs0-dev libqt4-declarative libqt4-designer libqt4-dev
  libqt4-dev-bin libqt4-help libqt4-network libqt4-opengl libqt4-qt3support
  libqt4-script libqt4-scripttools libqt4-sql libqt4-sql-mysql libqt4-svg
  libqt4-test libqt4-xmlpatterns libqtwebkit-dev libqtwebkit4 libx11-dev
  libx11-doc libxau-dev libxcb1-dev libxdmcp-dev libxext-dev mesa-common-dev
  mysql-common qt4-linguist-tools qt4-qmake x11proto-core-dev
  x11proto-input-dev x11proto-kb-dev x11proto-xext-dev xorg-sgml-doctools
  xtrans-dev
Suggested packages:
  libqt4-declarative-folderlistmodel libqt4-declarative-gestures
  libqt4-declarative-particles libqt4-declarative-shaders qt4-qmlviewer
  firebird-dev libmysqlclient-dev libpq-dev libsqlite0-dev libsqlite3-dev
  qt4-dev-tools qt4-doc unixodbc-dev libxcb-doc libxext-doc
The following NEW packages will be installed:
  libgl1-mesa-dev libglu1-mesa-dev libmysqlclient18 libpthread-stubs0
  libpthread-stubs0-dev libqt4-declarative libqt4-designer libqt4-dev
  libqt4-dev-bin libqt4-help libqt4-network libqt4-opengl libqt4-opengl-dev
  libqt4-qt3support libqt4-script libqt4-scripttools libqt4-sql
  libqt4-sql-mysql libqt4-svg libqt4-test libqt4-xmlpatterns libqtwebkit-dev
  libqtwebkit4 libx11-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
  libxext-dev mesa-common-dev mysql-common qt4-linguist-tools qt4-qmake
  x11proto-core-dev x11proto-input-dev x11proto-kb-dev x11proto-xext-dev
  xorg-sgml-doctools xtrans-dev
```

Now you can find some additional Qt™ libraries in the directory /usr/lib/arm-linux-gnueabihf (extract): libQtDeclarative.so.4.8.2, libQtNetwork.so.4.8.2, libQtOpenGL.so.4.8.2, libQtScript.so.4.8.2, libQtScriptTools .so.4.8.2, libQtSql.so.4.8.2, libQtSvg.so.4.8.2 and libQtWebKit.so.4.9.0.
The best way for development consists in the use of these libraries which should be copied from the target hardware to the cross compiler environment. Maybe a link creation or renaming of library files could be required.

One possible Makefile might look like this:

```
DEFINES  = -DQT_WEBKIT -DQT_NO_DEBUG -DQT_GUI_LIB -DQT_CORE_LIB -DQT_SHARED

CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDBus -l QtOpenGL -l GL -l glib-2.0 -l png12 -l expat -l pcre -l uuid -l z

all: qt

moc_glwidget.cpp: glwidget.h
	/usr/bin/moc-qt4 $(DEFINES) -I/usr/include/qt4 glwidget.h -o moc_glwidget.cpp
moc_mainwindow.cpp: mainwindow.h
	/usr/bin/moc-qt4 $(DEFINES) -I/usr/include/qt4 mainwindow.h -o moc_mainwindow.cpp

glwidget.o: glwidget.cpp moc_glwidget.cpp
main.o: main.cpp moc_mainwindow.cpp
mainwindow.o: mainwindow.cpp

qt: main.o glwidget.o mainwindow.o
	$(CXX) -o $@ main.o glwidget.o mainwindow.o $(LDFLAGS)

clean:
	rm qt main.o glwidget.o mainwindow.o
```

As a result you can see this picture:



File main.cpp:

```
/****************************************************************************
 ** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
 ** Contact: http://www.qt-project.org/legal
 ** This file is part of the examples of the Qt Toolkit.
 **
 ** $QT_BEGIN_LICENSE:BSD$
 ** You may use this file under the terms of the BSD license as follows:
 ** "Redistribution and use in source and binary forms, with or without
 ** modification, are permitted provided that the following conditions are
 ** met:
 **   * Redistributions of source code must retain the above copyright
 **     notice, this list of conditions and the following disclaimer.
 **   * Redistributions in binary form must reproduce the above copyright
 **     notice, this list of conditions and the following disclaimer in
 **     the documentation and/or other materials provided with the distribution.
 **   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
 **     of its contributors may be used to endorse or promote products derived
 **     from this software without specific prior written permission.
 ** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 ** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 ** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 ** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 ** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 ** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 ** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 ** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 ** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
 ** $QT_END_LICENSE$
**************************************************************************/

#include <QtGui/QApplication>
#include "mainwindow.h"
#include "moc_mainwindow.cpp"

int main (int argc, char *argv[])
{
    QApplication app (argc, argv);
    MainWindow mainWin;
    mainWin.show ();
    return app.exec ();
}
```

File glwidget.h (header with Qt^{TM} license see 'main.cpp'):

```
#ifndef GLWIDGET_H
#define GLWIDGET_H

#include <QtOpenGL/QGLWidget>

class GLWidget : public QGLWidget
{
    Q_OBJECT

public:
    GLWidget (QWidget *parent = 0);
    ~GLWidget ();

    int xRotation () const { return xRot; }
    int yRotation () const { return yRot; }
    int zRotation () const { return zRot; }

public slots:
    void setXRotation (int angle);
    void setYRotation (int angle);
    void setZRotation (int angle);

signals:
    void xRotationChanged (int angle);
    void yRotationChanged (int angle);
    void zRotationChanged (int angle);

protected:
    void initializeGL ();
    void paintGL ();
    void resizeGL (int width, int height);
    void mousePressEvent (QMouseEvent *event);
    void mouseMoveEvent (QMouseEvent *event);

private slots:
    void advanceGears ();

private:
    GLuint makeGear (const GLfloat *reflectance, GLdouble innerRadius, GLdouble outerRadius, GLdouble thickness,
                                                            GLdouble toothSize, GLint toothCount);
    void drawGear (GLuint gear, GLdouble dx, GLdouble dy, GLdouble dz, GLdouble angle);
    void normalizeAngle (int *angle);

    GLuint gear1;
    GLuint gear2;
    GLuint gear3;
    int xRot;
    int yRot;
    int zRot;
    int gear1Rot;
    QPoint lastPos;
};

#endif
```

File glwidget.cpp (header with Qt™ license see 'main.cpp'):

```
#include <QtGui/QtGui>
#include <QtOpenGL/QtOpenGL>
#include <math.h>
#include "glwidget.h"
#include "moc_glwidget.cpp"

GLWidget::GLWidget (QWidget *parent)
        : QGLWidget (parent)
{
    gear1 = 0;
    gear2 = 0;
    gear3 = 0;
    xRot = 0;
    yRot = 0;
    zRot = 0;
    gear1Rot = 0;

    QTimer *timer = new QTimer (this);
    connect (timer, SIGNAL (timeout ()), this, SLOT (advanceGears ()));
    timer->start (20);
}
```

```
GLWidget::~GLWidget ()
{
    makeCurrent ();
    glDeleteLists (gear1, 1);
    glDeleteLists (gear2, 1);
    glDeleteLists (gear3, 1);
}


void GLWidget::setXRotation (int angle)
{
    normalizeAngle (&angle);
    if (angle != xRot)
    {
      xRot = angle;
      emit xRotationChanged (angle);
      updateGL ();
    }
}


void GLWidget::setYRotation (int angle)
{
    normalizeAngle (&angle);
    if (angle != yRot)
    {
      yRot = angle;
      emit yRotationChanged (angle);
      updateGL ();
    }
}


void GLWidget::setZRotation (int angle)
{
    normalizeAngle (&angle);
    if (angle != zRot)
    {
      zRot = angle;
      emit zRotationChanged (angle);
      updateGL ();
    }
}


void GLWidget::initializeGL ()
{
    static const GLfloat lightPos[4] = { 5.0f, 5.0f, 10.0f, 1.0f };
    static const GLfloat reflectance1[4] = { 0.8f, 0.1f, 0.0f, 1.0f };
    static const GLfloat reflectance2[4] = { 0.0f, 0.8f, 0.2f, 1.0f };
    static const GLfloat reflectance3[4] = { 0.2f, 0.2f, 1.0f, 1.0f };
    glLightfv (GL_LIGHT0, GL_POSITION, lightPos);
    glEnable (GL_LIGHTING);
    glEnable (GL_LIGHT0);
    glEnable (GL_DEPTH_TEST);

    gear1 = makeGear (reflectance1, 1.0, 4.0, 1.0, 0.7, 20);
    gear2 = makeGear (reflectance2, 0.5, 2.0, 2.0, 0.7, 10);
    gear3 = makeGear (reflectance3, 1.3, 2.0, 0.5, 0.7, 10);

    glEnable (GL_NORMALIZE);
    glClearColor (0.0f, 0.0f, 0.0f, 1.0f);
}


void GLWidget::paintGL ()
{
    glClear (GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glPushMatrix ();
    glRotated (xRot / 16.0, 1.0, 0.0, 0.0);
    glRotated (yRot / 16.0, 0.0, 1.0, 0.0);
    glRotated (zRot / 16.0, 0.0, 0.0, 1.0);

    drawGear (gear1, -3.0, -2.0, 0.0, gear1Rot / 16.0);
    drawGear (gear2, +3.1, -2.0, 0.0, -2.0 * (gear1Rot / 16.0) - 9.0);

    glRotated (+90.0, 1.0, 0.0, 0.0);
    drawGear (gear3, -3.1, -1.8, -2.2, +2.0 * (gear1Rot / 16.0) - 2.0);

    glPopMatrix ();
}


void GLWidget::resizeGL (int width, int height)
{
    int side = qMin (width, height);
    glViewport ((width - side) / 2, (height - side) / 2, side, side);

    glMatrixMode (GL_PROJECTION);
    glLoadIdentity ();
    glFrustum (-1.0, +1.0, -1.0, 1.0, 5.0, 60.0);
    glMatrixMode (GL_MODELVIEW);
    glLoadIdentity ();
    glTranslated (0.0, 0.0, -40.0);
}


void GLWidget::mousePressEvent (QMouseEvent *event)
{
    lastPos = event->pos ();
}
```

```
void GLWidget::mouseMoveEvent (QMouseEvent *event)
{
    int dx = event->x () - lastPos.x ();
    int dy = event->y () - lastPos.y ();

    if (event->buttons () & Qt::LeftButton)
    {
      setXRotation (xRot + 8 * dy);
      setYRotation (yRot + 8 * dx);
    }
    else if (event->buttons () & Qt::RightButton)
    {
      setXRotation (xRot + 8 * dy);
      setZRotation (zRot + 8 * dx);
    }
    lastPos = event->pos ();
}


void GLWidget::advanceGears ()
{
    gear1Rot += 2 * 16;
    updateGL ();
}


GLuint GLWidget::makeGear (const GLfloat *reflectance, GLdouble innerRadius,
              GLdouble outerRadius, GLdouble thickness, GLdouble toothSize, GLint toothCount)
{
    const double Pi = 3.14159265358979323846;

    GLuint list = glGenLists (1);
    glNewList (list, GL_COMPILE);
    glMaterialfv (GL_FRONT, GL_AMBIENT_AND_DIFFUSE, reflectance);

    GLdouble r0 = innerRadius;
    GLdouble r1 = outerRadius - toothSize / 2.0;
    GLdouble r2 = outerRadius + toothSize / 2.0;
    GLdouble delta = (2.0 * Pi / toothCount) / 4.0;
    GLdouble z = thickness / 2.0;
    int i, j;

    glShadeModel (GL_FLAT);

    for (i = 0; i < 2; ++i)
    {
      GLdouble sign = (i == 0) ? +1.0 : -1.0;

      glNormal3d (0.0, 0.0, sign);

      glBegin (GL_QUAD_STRIP);
```

```
  for (j = 0; j <= toothCount; ++j)
  {
    GLdouble angle = 2.0 * Pi * j / toothCount;
    glVertex3d (r0 * cos (angle), r0 * sin (angle), sign * z);
    glVertex3d (r1 * cos (angle), r1 * sin (angle), sign * z);
    glVertex3d (r0 * cos (angle), r0 * sin (angle), sign * z);
    glVertex3d (r1 * cos (angle + 3 * delta), r1 * sin (angle + 3 * delta), sign * z);
  }
  glEnd ();

  glBegin (GL_QUADS);
  for (j = 0; j < toothCount; ++j)
  {
    GLdouble angle = 2.0 * Pi * j / toothCount;
    glVertex3d (r1 * cos (angle), r1 * sin (angle), sign * z);
    glVertex3d (r2 * cos (angle + delta), r2 * sin (angle + delta), sign * z);
    glVertex3d (r2 * cos (angle + 2 * delta), r2 * sin (angle + 2 * delta), sign * z);
    glVertex3d (r1 * cos (angle + 3 * delta), r1 * sin (angle + 3 * delta), sign * z);
  }
  glEnd ();
}

glBegin (GL_QUAD_STRIP);
for (i = 0; i < toothCount; ++i)
{
  for (j = 0; j < 2; ++j)
  {
    GLdouble angle = 2.0 * Pi * (i + (j / 2.0)) / toothCount;
    GLdouble s1 = r1;
    GLdouble s2 = r2;
    if (j == 1)
      qSwap (s1, s2);

    glNormal3d (cos (angle), sin (angle), 0.0);
    glVertex3d (s1 * cos (angle), s1 * sin (angle), +z);
    glVertex3d (s1 * cos (angle), s1 * sin (angle), -z);

    glNormal3d (s2 * sin (angle + delta) - s1 * sin (angle), s1 * cos (angle) - s2 * cos (angle + delta), 0.0);
    glVertex3d (s2 * cos (angle + delta), s2 * sin (angle + delta), +z);
    glVertex3d (s2 * cos (angle + delta), s2 * sin (angle + delta), -z);
  }
}
glVertex3d (r1, 0.0, +z);
glVertex3d (r1, 0.0, -z);
glEnd ();

glShadeModel (GL_SMOOTH);
glBegin (GL_QUAD_STRIP);
for (i = 0; i <= toothCount; ++i)
{
  GLdouble angle = i * 2.0 * Pi / toothCount;
  glNormal3d (-cos (angle), -sin (angle), 0.0);
  glVertex3d (r0 * cos (angle), r0 * sin (angle), +z);
  glVertex3d (r0 * cos (angle), r0 * sin (angle), -z);
}
```

```
    glEnd ();
    glEndList ();
    return list;
}


void GLWidget::drawGear (GLuint gear, GLdouble dx, GLdouble dy, GLdouble dz, GLdouble angle)
{
    glPushMatrix ();
    glTranslated (dx, dy, dz);
    glRotated (angle, 0.0, 0.0, 1.0);
    glCallList (gear);
    glPopMatrix ();
}


void GLWidget::normalizeAngle (int *angle)
{
    while (*angle < 0)
      *angle += 360 * 16;
    while (*angle > 360 * 16)
      *angle -= 360 * 16;
}
```

File mainwindow.h (header with Qt^{TM} license see 'main.cpp'):

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QtGui/QMainWindow>

class QAction;
class QLabel;
class QMenu;
class QScrollArea;
class QSlider;
class GLWidget;

class MainWindow : public QMainWindow
{
    Q_OBJECT
public:
    MainWindow ();
private slots:
    void renderIntoPixmap ();
    void grabFrameBuffer ();
    void clearPixmap ();
    void about ();
private:
    void createActions ();
    void createMenus ();
    QSlider *createSlider (const char *changedSignal, const char *setterSlot);
    void setPixmap (const QPixmap &pixmap);
    QSize getSize ();

    QWidget *centralWidget;
    QScrollArea *glWidgetArea;
    QScrollArea *pixmapLabelArea;
    GLWidget *glWidget;
    QLabel *pixmapLabel;
    QSlider *xSlider;
    QSlider *ySlider;
    QSlider *zSlider;

    QMenu *fileMenu;
    QMenu *helpMenu;
    QAction *grabFrameBufferAct;
    QAction *renderIntoPixmapAct;
    QAction *clearPixmapAct;
    QAction *exitAct;
    QAction *aboutAct;
    QAction *aboutQtAct;
};

#endif
```

File mainwindow.cpp (header with Qt™ license see 'main.cpp'):

```
#include <QtGui/QtGui>
#include <QtOpenGL/QtOpenGL>
#include "glwidget.h"
#include "mainwindow.h"

MainWindow::MainWindow ()
{
    centralWidget = new QWidget;
    setCentralWidget (centralWidget);

    glWidget = new GLWidget;
    pixmapLabel = new QLabel;

    glWidgetArea = new QScrollArea;
    glWidgetArea->setWidget (glWidget);
    glWidgetArea->setWidgetResizable (true);
    glWidgetArea->setHorizontalScrollBarPolicy (Qt::ScrollBarAlwaysOff);
    glWidgetArea->setVerticalScrollBarPolicy (Qt::ScrollBarAlwaysOff);
    glWidgetArea->setSizePolicy (QSizePolicy::Ignored, QSizePolicy::Ignored);
    glWidgetArea->setMinimumSize (50, 50);

    pixmapLabelArea = new QScrollArea;
    pixmapLabelArea->setWidget (pixmapLabel);
    pixmapLabelArea->setSizePolicy (QSizePolicy::Ignored, QSizePolicy::Ignored);
    pixmapLabelArea->setMinimumSize (50, 50);
    xSlider = createSlider (SIGNAL (xRotationChanged (int)), SLOT (setXRotation (int)));
    ySlider = createSlider (SIGNAL (yRotationChanged (int)), SLOT (setYRotation (int)));
    zSlider = createSlider (SIGNAL (zRotationChanged (int)), SLOT (setZRotation (int)));

    createActions ();
    createMenus ();

    QGridLayout *centralLayout = new QGridLayout;
    centralLayout->addWidget (glWidgetArea, 0, 0);
    centralLayout->addWidget (pixmapLabelArea, 0, 1);
    centralLayout->addWidget (xSlider, 1, 0, 1, 2);
    centralLayout->addWidget (ySlider, 2, 0, 1, 2);
    centralLayout->addWidget (zSlider, 3, 0, 1, 2);
    centralWidget->setLayout (centralLayout);

    xSlider->setValue (15 * 16);
    ySlider->setValue (345 * 16);
    zSlider->setValue (0 * 16);

    setWindowTitle (tr ("Grabber"));
    resize (400, 300);
}


void MainWindow::renderIntoPixmap ()
{
    QSize size = getSize ();
    if (size.isValid ())
    {
      QPixmap pixmap = glWidget->renderPixmap (size.width (), size.height ());
      setPixmap (pixmap);
    }
}


void MainWindow::grabFrameBuffer ()
{
    QImage image = glWidget->grabFrameBuffer ();
    setPixmap (QPixmap::fromImage (image));
}


void MainWindow::clearPixmap ()
{
    setPixmap (QPixmap ());
}


void MainWindow::about ()
{
    QMessageBox::about (this, tr ("About Grabber"),
      tr ("The <b>Grabber</b> example demonstrates two approaches for rendering OpenGL into a Qt pixmap."));
}
```

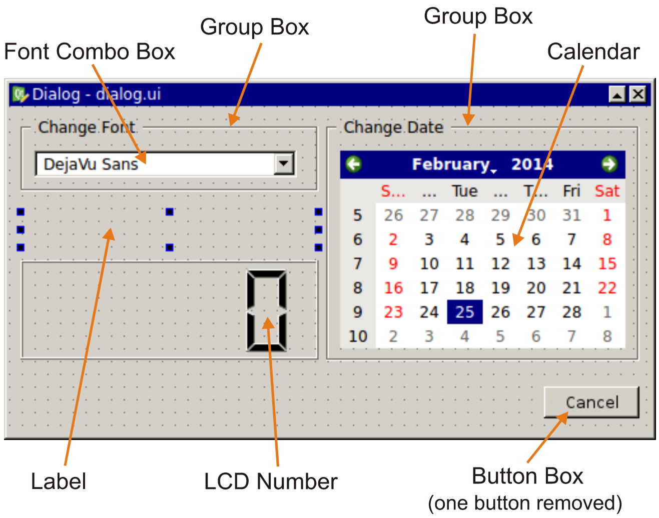
```
void MainWindow::createActions ()
{
    renderIntoPixmapAct = new QAction (tr ("&Render into Pixmap..."), this);
    renderIntoPixmapAct->setShortcut (tr ("Ctrl+R"));
    connect (renderIntoPixmapAct, SIGNAL (triggered ()), this, SLOT (renderIntoPixmap ()));

    grabFrameBufferAct = new QAction (tr ("&Grab Frame Buffer"), this);
    grabFrameBufferAct->setShortcut (tr ("Ctrl+G"));
    connect (grabFrameBufferAct, SIGNAL (triggered ()), this, SLOT (grabFrameBuffer ()));

    clearPixmapAct = new QAction (tr ("&Clear Pixmap"), this);
    clearPixmapAct->setShortcut (tr ("Ctrl+L"));
    connect (clearPixmapAct, SIGNAL (triggered ()), this, SLOT (clearPixmap ()));

    exitAct = new QAction (tr ("E&xit"), this);
    exitAct->setShortcuts (QKeySequence::Quit);
    connect (exitAct, SIGNAL (triggered ()), this, SLOT (close ()));

    aboutAct = new QAction (tr ("&About"), this);
    connect (aboutAct, SIGNAL (triggered ()), this, SLOT (about ()));

    aboutQtAct = new QAction (tr ("About &Qt"), this);
    connect (aboutQtAct, SIGNAL (triggered ()), qApp, SLOT (aboutQt ()));
}


void MainWindow::createMenus ()
{
    fileMenu = menuBar ()->addMenu (tr ("&File"));
    fileMenu->addAction (renderIntoPixmapAct);
    fileMenu->addAction (grabFrameBufferAct);
    fileMenu->addAction (clearPixmapAct);
    fileMenu->addSeparator ();
    fileMenu->addAction (exitAct);

    helpMenu = menuBar ()->addMenu (tr ("&Help"));
    helpMenu->addAction (aboutAct);
    helpMenu->addAction (aboutQtAct);
}


QSlider *MainWindow::createSlider (const char *changedSignal, const char *setterSlot)
{
    QSlider *slider = new QSlider (Qt::Horizontal);
    slider->setRange (0, 360 * 16);
    slider->setSingleStep (16);
    slider->setPageStep (15 * 16);
    slider->setTickInterval (15 * 16);
    slider->setTickPosition (QSlider::TicksRight);
    connect (slider, SIGNAL (valueChanged (int)), glWidget, setterSlot);
    connect (glWidget, changedSignal, slider, SLOT (setValue (int)));
    return slider;
}
```

```
void MainWindow::setPixmap (const QPixmap &pixmap)
{
    pixmapLabel->setPixmap (pixmap);
    QSize size = pixmap.size ();
    if (size - QSize (1, 0) == pixmapLabelArea->maximumViewportSize ())
      size -= QSize (1, 0);
    pixmapLabel->resize (size);
}


QSize MainWindow::getSize ()
{
    bool ok;
    QString text = QInputDialog::getText (this, tr ("Grabber"), tr ("Enter pixmap size:"),
        QLineEdit::Normal, tr ("%1 x %2").arg (glWidget->width ()).arg (glWidget->height ()), &ok);
    if (! ok)
      return QSize ();

    QRegExp regExp (tr ("([0-9]+) *x *([0-9]+)"));
    if (regExp.exactMatch (text))
    {
      int width = regExp.cap (1).toInt ();
      int height = regExp.cap (2).toInt ();
      if (width > 0 && width < 2048 && height > 0 && height < 2048)
        return QSize (width, height);
    }

    return glWidget->size ();
}
```

Finally, for a change, the Guide now presents a self-created Qt™ sample program which shows a calendar with date display in plain text and a clock widget with hour/minute display. The graphic surface has been made with the Qt Designer. If on the Desktop PC the Designer is not already installed use the command line

```
sudo apt-get install qt4-designer
```

and then call designer. The first variant applies the Dialog with Buttons Bottom template. The figure shows further details (without assignment of a layout).



For integration you can use the User Interface Compiler (UIC) in the Makefile or produce an Include-file with Form ➔ Show Code ➔ Icon Save. One possible Makefile might look like this (lines in orange are optional):

```
CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l glib-2.0 -l png12 -l expat -l pcre -l uuid -l z

all: qt

moc_main.cpp: main.h
    /usr/bin/moc-qt4 -I/usr/include/qt4 main.h -o moc_main.cpp

ui_dialog.h: dialog.ui
    /usr/bin/uic-qt4 dialog.ui -o ui_dialog.h

main.o: main.cpp moc_main.cpp ui_dialog.h

qt: main.o
    $(CXX) -o $@ main.o $(LDFLAGS)

clean:
    rm qt main.o
```

The associated UI Include-file (ui_dialog.h) has the following content:

```
/********************************************************************************
** Form generated from reading UI file 'dialog.ui'
**
** Created: .......
**      by: Qt User Interface Compiler version 4.8.1
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_DIALOG_H
#define UI_DIALOG_H

#include <QtCore/QVariant>
#include <QtGui/QAction>
#include <QtGui/QApplication>
#include <QtGui/QButtonGroup>
#include <QtGui/QCalendarWidget>
#include <QtGui/QDialog>
#include <QtGui/QDialogButtonBox>
#include <QtGui/QFontComboBox>
#include <QtGui/QFormLayout>
#include <QtGui/QGroupBox>
#include <QtGui/QHeaderView>
#include <QtGui/QLCDNumber>
#include <QtGui/QLabel>

QT_BEGIN_NAMESPACE
```

```
class Ui_Dialog
{
public:
    QGroupBox *groupBox_2;
    QFormLayout *formLayout;
    QFontComboBox *fontComboBox;
    QLCDNumber *lcdNumber;
    QGroupBox *groupBox;
    QCalendarWidget *calendarWidget;
    QDialogButtonBox *buttonBox;
    QLabel *label;

    void setupUi(QDialog *Dialog)
    {
        if (Dialog->objectName().isEmpty())
            Dialog->setObjectName(QString::fromUtf8("Dialog"));
        Dialog->resize(504, 258);
        groupBox_2 = new QGroupBox(Dialog);
        groupBox_2->setObjectName(QString::fromUtf8("groupBox_2"));
        groupBox_2->setGeometry(QRect(9, 9, 234, 58));
        formLayout = new QFormLayout(groupBox_2);
        formLayout->setObjectName(QString::fromUtf8("formLayout"));
        fontComboBox = new QFontComboBox(groupBox_2);
        fontComboBox->setObjectName(QString::fromUtf8("fontComboBox"));

        formLayout->setWidget(0, QFormLayout::LabelRole, fontComboBox);

        lcdNumber = new QLCDNumber(Dialog);
        lcdNumber->setObjectName(QString::fromUtf8("lcdNumber"));
        lcdNumber->setGeometry(QRect(9, 121, 234, 78));
        groupBox = new QGroupBox(Dialog);
        groupBox->setObjectName(QString::fromUtf8("groupBox"));
        groupBox->setGeometry(QRect(249, 9, 246, 191));
        calendarWidget = new QCalendarWidget(groupBox);
        calendarWidget->setObjectName(QString::fromUtf8("calendarWidget"));
        calendarWidget->setGeometry(QRect(11, 26, 224, 155));
        buttonBox = new QDialogButtonBox(Dialog);
        buttonBox->setObjectName(QString::fromUtf8("buttonBox"));
        buttonBox->setGeometry(QRect(336, 219, 160, 27));
        buttonBox->setOrientation(Qt::Horizontal);
        buttonBox->setStandardButtons(QDialogButtonBox::Cancel);
        label = new QLabel(Dialog);
        label->setObjectName(QString::fromUtf8("label"));
        label->setGeometry(QRect(9, 83, 234, 28));
        QFont font;
        font.setPointSize(14);
        label->setFont(font);
        label->setAlignment(Qt::AlignJustify|Qt::AlignVCenter);

        retranslateUi(Dialog);
        QObject::connect(buttonBox, SIGNAL(accepted()), Dialog, SLOT(accept()));
        QObject::connect(buttonBox, SIGNAL(rejected()), Dialog, SLOT(reject()));

        QMetaObject::connectSlotsByName(Dialog);
    } // setupUi
    void retranslateUi(QDialog *Dialog)
    {
      Dialog->setWindowTitle(QApplication::translate("Dialog", "Dialog", 0, QApplication::UnicodeUTF8));
      groupBox_2->setTitle(QApplication::translate("Dialog", " Change Font ", 0, QApplication::UnicodeUTF8));
      groupBox->setTitle(QApplication::translate("Dialog", " Change Date ", 0, QApplication::UnicodeUTF8));
      label->setText(QString());
    } // retranslateUi

};

namespace Ui {
    class Dialog: public Ui_Dialog {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_DIALOG_H
```

The lines in green color contain some key terms for the main program but the main Include-file (main.h) is listed first.

```
#ifndef MAIN_H
#define MAIN_H

class Dialog : public QDialog
{
    Q_OBJECT

public:
    explicit Dialog (QWidget *parent = 0);
    ~Dialog ();

private slots:
    void showTime ();
    void drawDate ();

private:
    static const int TIMEOUT = 200;
    static const int FONTSIZE = 14;
    Ui::Dialog *ui;
};

#endif
```

and now follows the real main program (main.cpp):

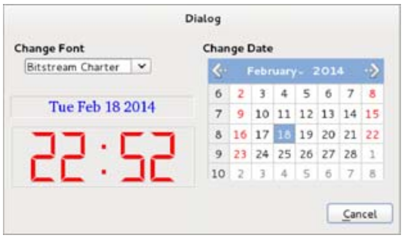
```
/***************************************************************
 * Qt demonstration program

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License.
****************************************************************/
```

```
#include <QtGui/QtGui>
#include "ui_dialog.h"
#include "moc_main.cpp"

Dialog::Dialog (QWidget *parent) : QDialog (parent), ui (new Ui::Dialog)
{
    ui->setupUi (this);

    // only an example for subsequent modifications of calendar features
    ui->calendarWidget->setMinimumDate (QDate (2014, 1, 1));
    ui->calendarWidget->setMaximumDate (QDate (2020, 1, 1));
    ui->calendarWidget->setSelectedDate (QDate (2014, 2, 28));
    ui->calendarWidget->setGridVisible (true);
    ui->calendarWidget->setHorizontalHeaderFormat (QCalendarWidget::SingleLetterDayNames);

    // change color and other properties of the label widget (date display)
    QPalette p_lab = ui->label->palette ();
    p_lab.setColor (QPalette::WindowText, Qt::blue);
    ui->label->setAutoFillBackground (true);
    ui->label->setPalette (p_lab);
    ui->label->setFrameStyle (QFrame::Panel | QFrame::Sunken);
    ui->label->setAlignment (Qt::AlignCenter);

    // update the label widget and link Font Combo Box and Calendar to the drawDate routine
    QObject::connect (ui->fontComboBox, SIGNAL (currentFontChanged (const QFont &)), this,
                                                                    SLOT (drawDate ()));
    QDate date = ui->calendarWidget->selectedDate ();
    ui->label->setText (date.toString ());
    QObject::connect (ui->calendarWidget, SIGNAL (selectionChanged ()), this, SLOT (drawDate ()));

    // change the color of LCD Number widget
    QPalette p_lcd = ui->lcdNumber->palette ();
    p_lcd.setColor (QPalette::WindowText, Qt::red);
    ui->lcdNumber->setPalette (p_lcd);

    // link a timer to the showTime routine and start the timer
    QTimer *timer = new QTimer (this);
    QObject::connect (timer, SIGNAL (timeout ()), this, SLOT (showTime ()));
    timer->start (TIMEOUT);
}


Dialog::~Dialog ()
{
    delete ui;
}


void Dialog::drawDate ()
{
    QFont font = ui->fontComboBox->currentFont ();
    font.setPointSize (FONTSIZE);
    ui->label->setFont (font);
    QDate date = ui->calendarWidget->selectedDate ();
    ui->label->setText (date.toString ());
}
```

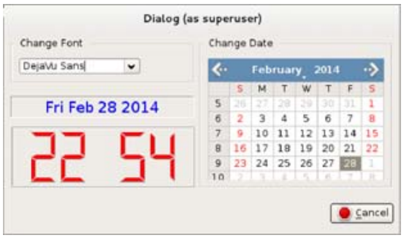
```
void Dialog::showTime ()
{
    QTime time = QTime::currentTime ();
    QString text = time.toString ("hh:mm");
    // realizes blinking ':' for seconds ticking
    if (! (time.second () % 2))
      text[2] = ' ';
    ui->lcdNumber->display (text);
}


int main (int argc, char *argv[])
{
    QApplication app (argc, argv);
    Dialog *dialog = new Dialog;
    dialog->show ();
    return app.exec ();
}
```

If you run the program with user or root rights you will experience a surprise - the appearance differs strongly between the modes (without QtConfig and with the standard theme Adwaita).

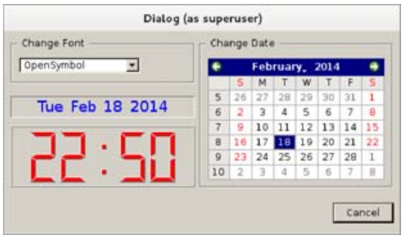
user



root (su)



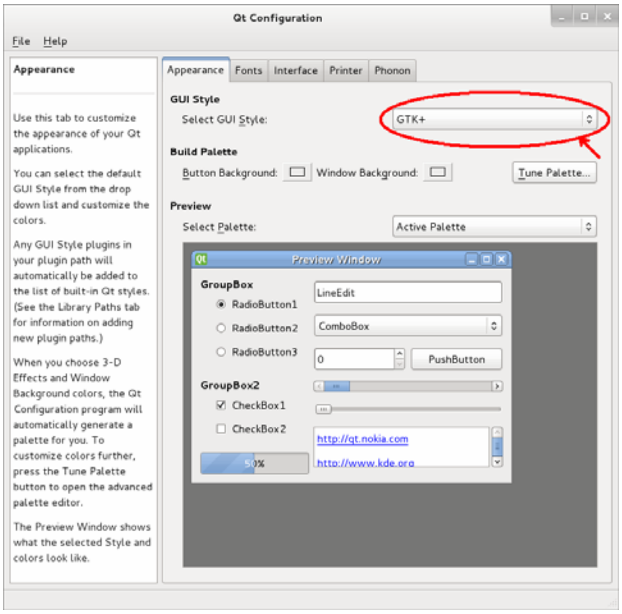
root (sudo -i)



The use of QtConfig can minimize the differences. Load this tool <u>on the target hardware</u> with

```
sudo apt-get install qt4-qtconfig
```

After call of qtconfig you see for example the following screen:



Appearance

Use this tab to customize the appearance of your Qt applications.

You can select the default GUI Style from the drop down list and customize the colors.

Any GUI Style plugins in your plugin path will automatically be added to the list of built-in Qt styles. (See the Library Paths tab for information on adding new plugin paths.)

When you choose 3-D Effects and Window Background colors, the Qt Configuration program will automatically generate a palette for you. To customize colors further, press the Tune Palette button to open the advanced palette editor.

The Preview Window shows what the selected Style and colors look like.

GUI Style

Select GUI Style: GTK+

Build Palette

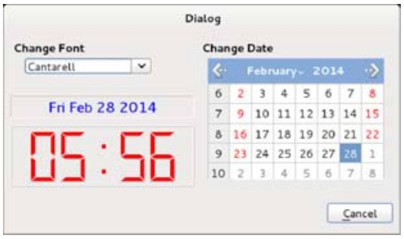
Button Background: □ Window Background: □ Tune Palette...

Preview

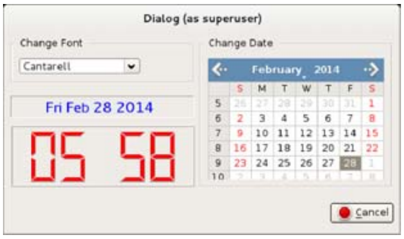
Select Palette: Active Palette

The figure already implies the absence of the Groupbox border if the GTK+ style is active. With this style the sample program looks like this (not much better):

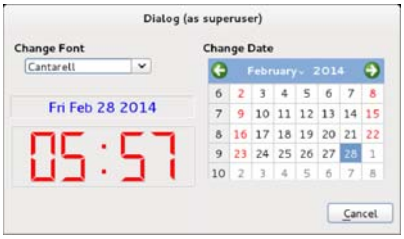
user



root (su)



root (sudo -i)



The web search refers to QtCurve as a way out of this issue but the download on the target hardware with

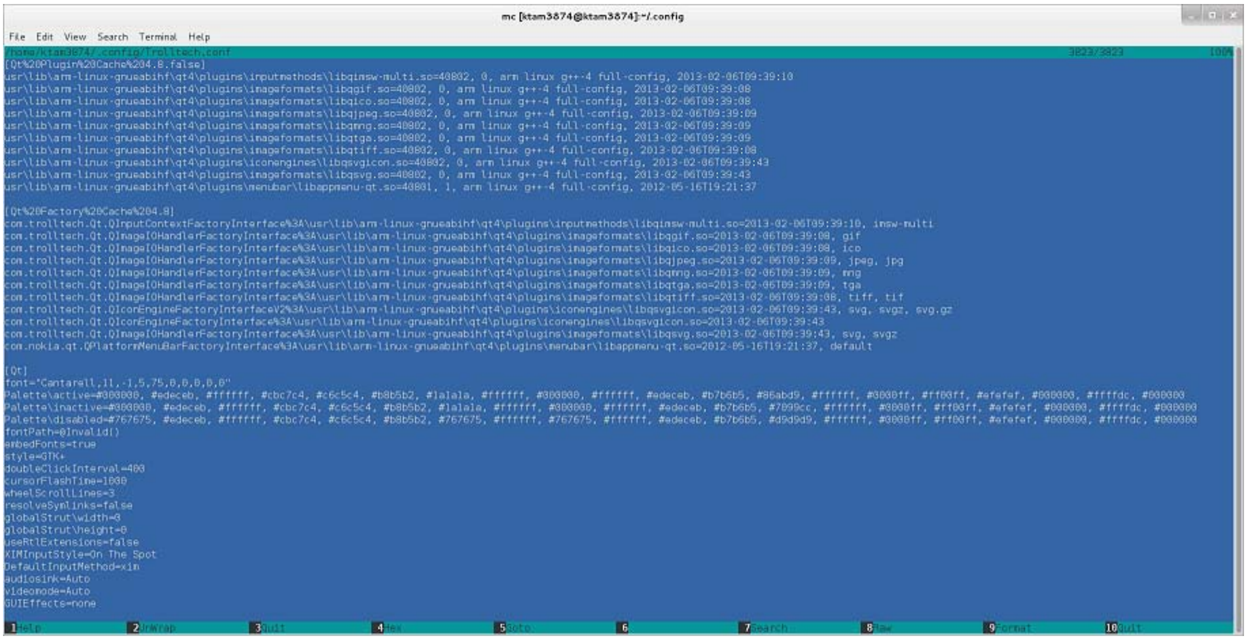
```
sudo apt-get install qtcurve
```

has no effect. If you analyze the download message (extract):

```
The following NEW packages will be installed:
  docbook-xml docbook-xsl fonts-freefont-ttf gtk2-engines-qtcurve kate-data
  katepart kde-runtime kde-runtime-data kde-style-qtcurve kdelibs-bin
  kdelibs5-data kdelibs5-plugins kdoctools kwin-style-qtcurve liba52-0.7.4
  libattica0 libavformat53 libcddb2 libclucene0ldbl libdirac-decoder0
  libdlrestrictions1 libdvbpsi7 libebml3 libfam0 libgif4 libilmbase6 libiodbc2
                                  .
                                  .
                                  .
  phonon-backend-vlc plasma-scriptengine-javascript qtcurve qtcurve-i18n
  sgml-data shared-desktop-ontologies soprano-daemon ttf-dejavu
  ttf-dejavu-extra virtuoso-minimal virtuoso-opensource-6.1-bin
  virtuoso-opensource-6.1-common vlc vlc-data vlc-nox vlc-plugin-notify
  vlc-plugin-pulse xml-core
```

you only see the gtk2-engine and KDE^{TM} plugins - the gtk3-engine is absent.

Another method to control the appearance may consist in manually modifying the file ~/.config/Trolltech.conf


mc [ktam3874@ktam3874]:~/.config
File Edit View Search Terminal Help
1Help 2UnWrap 3Quit 4Hex 5Goto 6 7Search 8Raw 9Format 10Quit

```
/home/ktam3874/.config/Trolltech.conf                                    3823/3823   100%
[Qt%20Plugin%20Cache%204.8.false]
usr\lib\arm-linux-gnueabihf\qt4\plugins\inputmethods\libqimsw-multi.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:10
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqgif.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:08
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqico.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:08
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqjpeg.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:09
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqmng.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:09
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqtga.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:09
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqtiff.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:08
usr\lib\arm-linux-gnueabihf\qt4\plugins\iconengines\libqsvgicon.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:43
usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqsvg.so=40802, 0, arm linux g++-4 full-config, 2013-02-06T09:39:43
usr\lib\arm-linux-gnueabihf\qt4\plugins\menubar\libappmenu-qt.so=40801, 1, arm linux g++-4 full-config, 2012-05-16T19:21:37

[Qt%20Factory%20Cache%204.8]
com.trolltech.Qt.QInputContextFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\inputmethods\libqimsw-multi.so=2013-02-06T09:39:10, imsw-multi
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqgif.so=2013-02-06T09:39:08, gif
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqico.so=2013-02-06T09:39:08, ico
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqjpeg.so=2013-02-06T09:39:09, jpeg, jpg
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqmng.so=2013-02-06T09:39:09, mng
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqtga.so=2013-02-06T09:39:09, tga
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqtiff.so=2013-02-06T09:39:08, tiff, tif
com.trolltech.Qt.QIconEngineFactoryInterfaceV2%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\iconengines\libqsvgicon.so=2013-02-06T09:39:43, svg, svgz, svg.gz
com.trolltech.Qt.QIconEngineFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\iconengines\libqsvgicon.so=2013-02-06T09:39:43
com.trolltech.Qt.QImageIOHandlerFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\imageformats\libqsvg.so=2013-02-06T09:39:43, svg, svgz
com.nokia.qt.QPlatformMenuBarFactoryInterface%3A\usr\lib\arm-linux-gnueabihf\qt4\plugins\menubar\libappmenu-qt.so=2012-05-16T19:21:37, default

[Qt]
font="Cantarell,11,-1,5,75,0,0,0,0,0"
Palette\active=#000000, #edeceb, #ffffff, #cbc7c4, #c6c5c4, #b8b5b2, #1a1a1a, #ffffff, #000000, #ffffff, #edeceb, #b7b6b5, #86abd9, #ffffff, #0000ff, #ff00ff, #efefef, #000000, #ffffdc, #000000
Palette\inactive=#000000, #edeceb, #ffffff, #cbc7c4, #c6c5c4, #b8b5b2, #1a1a1a, #ffffff, #000000, #ffffff, #edeceb, #b7b6b5, #7099cc, #ffffff, #0000ff, #ff00ff, #efefef, #000000, #ffffdc, #000000
Palette\disabled=#767675, #edeceb, #ffffff, #cbc7c4, #c6c5c4, #b8b5b2, #767675, #ffffff, #767675, #ffffff, #edeceb, #b7b6b5, #d9d9d9, #ffffff, #0000ff, #ff00ff, #efefef, #000000, #ffffdc, #000000
fontPath=@Invalid()
embedFonts=true
style=GTK+
doubleClickInterval=400
cursorFlashTime=1000
wheelScrollLines=3
resolveSymlinks=false
globalStrut\width=0
globalStrut\height=0
useRtlExtensions=false
XIMInputStyle=On The Spot
DefaultInputMethod=xim
audiosink=Auto
videomode=Auto
GUIEffects=none
```

The second variant applies the Main Window template. The optional lines in the Makefile change according to kind of form:

```
ui_mainwindow.h: mainwindow.ui
    /usr/bin/uic-qt4 mainwindow.ui -o ui_mainwindow.h

main.o: main.cpp moc_main.cpp ui_mainwindow.h
```

and the associated UI Include-file (ui_mainwindow.h) as follows:

```
/********************************************************************************
** Form generated from reading UI file 'mainwindow.ui'
**
** Created: .......
**      by: Qt User Interface Compiler version 4.8.1
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_MAINWINDOW_H
#define UI_MAINWINDOW_H

#include <QtCore/QVariant>
#include <QtGui/QAction>
#include <QtGui/QApplication>
#include <QtGui/QButtonGroup>
#include <QtGui/QCalendarWidget>
#include <QtGui/QFontComboBox>
#include <QtGui/QFormLayout>
#include <QtGui/QGroupBox>
#include <QtGui/QHeaderView>
#include <QtGui/QLCDNumber>
#include <QtGui/QLabel>
#include <QtGui/QMainWindow>
#include <QtGui/QPushButton>
#include <QtGui/QWidget>

QT_BEGIN_NAMESPACE

class Ui_MainWindow
{
public:
    QWidget *centralwidget;
    QGroupBox *groupBox_2;
    QFormLayout *formLayout;
    QFontComboBox *fontComboBox;
    QLCDNumber *lcdNumber;
    QGroupBox *groupBox;
    QCalendarWidget *calendarWidget;
    QLabel *label;
    QPushButton *pushButton;
    void setupUi(QMainWindow *MainWindow)
    {
      if (MainWindow->objectName().isEmpty())
        MainWindow->setObjectName(QString::fromUtf8("MainWindow"));
      MainWindow->resize(504, 258);
      centralwidget = new QWidget(MainWindow);
      centralwidget->setObjectName(QString::fromUtf8("centralwidget"));
      groupBox_2 = new QGroupBox(centralwidget);
      groupBox_2->setObjectName(QString::fromUtf8("groupBox_2"));
      groupBox_2->setGeometry(QRect(9, 9, 234, 58));
      formLayout = new QFormLayout(groupBox_2);
      formLayout->setObjectName(QString::fromUtf8("formLayout"));
      fontComboBox = new QFontComboBox(groupBox_2);
      fontComboBox->setObjectName(QString::fromUtf8("fontComboBox"));

      formLayout->setWidget(0, QFormLayout::LabelRole, fontComboBox);

      lcdNumber = new QLCDNumber(centralwidget);
      lcdNumber->setObjectName(QString::fromUtf8("lcdNumber"));
      lcdNumber->setGeometry(QRect(9, 121, 234, 78));
      groupBox = new QGroupBox(centralwidget);
      groupBox->setObjectName(QString::fromUtf8("groupBox"));
      groupBox->setGeometry(QRect(249, 9, 246, 191));
      calendarWidget = new QCalendarWidget(groupBox);
      calendarWidget->setObjectName(QString::fromUtf8("calendarWidget"));
      calendarWidget->setGeometry(QRect(11, 26, 224, 155));
      label = new QLabel(centralwidget);
      label->setObjectName(QString::fromUtf8("label"));
      label->setGeometry(QRect(9, 83, 234, 28));
      QFont font;
      font.setPointSize(14);
      label->setFont(font);
      label->setAlignment(Qt::AlignJustify|Qt::AlignVCenter);
      pushButton = new QPushButton(centralwidget);
      pushButton->setObjectName(QString::fromUtf8("pushButton"));
      pushButton->setGeometry(QRect(410, 221, 80, 25));
      MainWindow->setCentralWidget(centralwidget);

      retranslateUi(MainWindow);
      QObject::connect(pushButton, SIGNAL(clicked()), MainWindow, SLOT(close()));

      QMetaObject::connectSlotsByName(MainWindow);
    } // setupUi

    void retranslateUi(QMainWindow *MainWindow)
    {
      MainWindow->setWindowTitle(QApplication::translate("MainWindow", "MainWindow", 0,
                                                          QApplication::UnicodeUTF8));
      groupBox_2->setTitle(QApplication::translate("MainWindow", " Change Font ", 0,
                                                          QApplication::UnicodeUTF8));
      groupBox->setTitle(QApplication::translate("MainWindow", " Change Date ", 0, QApplication::UnicodeUTF8));
      label->setText(QString());
      pushButton->setText(QApplication::translate("MainWindow", "Cancel", 0, QApplication::UnicodeUTF8));
    } // retranslateUi
};
```

```
namespace Ui {
    class MainWindow: public Ui_MainWindow {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_MAINWINDOW_H
```

and now the main Include-file (main.h):

```
#ifndef MAIN_H
#define MAIN_H

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow (QWidget *parent = 0);
    ~MainWindow ();

private slots:
    void showTime ();
    void drawDate ();

private:
    static const int TIMEOUT = 200;
    static const int FONTSIZE = 14;
    Ui::MainWindow *ui;
};

#endif
```

and finally the changes (orange color) in main.cpp:

```
#include <QtGui/QtGui>
#include "ui_mainwindow.h"
#include "moc_main.cpp"

MainWindow::MainWindow (QWidget *parent) : QMainWindow (parent), ui (new Ui::MainWindow)

MainWindow::~MainWindow ()

void MainWindow::drawDate ()

void MainWindow::showTime ()

int main (int argc, char *argv[])
{
    QApplication app (argc, argv);
    MainWindow *mw = new MainWindow;
    mw->show ();
    return app.exec ();
}
```

10.12.5 Qt^{TM} Quick 1.x / QML

Qt^{TM} Quick 1.x (QUICK = Qt User Interface Creation Kit) is based on Qt^{TM} 4.x and uses the QPainter/ QGraphics-View API. For the newer Qt^{TM} Quick 2.x (based on SceneGraph, an OpenGL® ES 2.x abstraction layer) you need Qt^{TM} 5.x which libraries are not included in the Debian^{TM} Wheezy image. However, JsCore (JavaScript^{TM} engine of Webkit^{TM}) serves the platform for the QML (Qt Meta Language or Qt Modeling Language) part of Qt^{TM} Quick 1.x. QML is also a scripting language which does not require compilation and can instantly be run on many devices. Qt^{TM} Quick uses the Qt^{TM} Declarative module resp. library.
On an x86 Desktop PC using a cross compiler (Ubuntu^{TM} environment) you need some additional libraries (application dependent):

```
QtCore
QtGui
QtDeclarative
QtScript
QtSql
QtNetwork
QtXmlPatterns
glib-2.0
png12
expat
pcre
uuid
z (libz)
```

Some Qt^{TM} dynamic libraries are available within the standard Debian^{TM} image (have a look at the directories /usr/lib/arm-linux-gnueabihf respectively /lib/arm-linux-gnueabihf), especially these five libraries: QtCore.so.4.8.2, QtGui.so.4.8.2, QtDBus.so.4.8.2, QtCLucene.so.4.8.2 and QtXml.so.4.8.2. The best way for development consists in the use of these libraries which should be copied from the target hardware to the cross compiler environment. For further information see the previous chapter.
You need five additional libraries: QtDeclarative.so.4.8.2, QtScript.so.4.8.2, QtSql.so.4.8.2, QtNetwork.so.4.8.2 and QtXmlPatterns.so.4.8.2. One way to download these files on the target hardware insists in the use of

```
sudo apt-get install libqt4-dev
```

or with less effort

```
sudo apt-get install libqt4-opengl-dev
```

and thereafter these five files should also be copied to the cross compiler environment.

You can find a detailed list of all available Debian^{TM} Wheezy packages and their associated content on

https://packages.debian.org/wheezy/allpackages

Qt^{TM} Quick requires two additional special files: qmlapplicationviewer.h and qmlapplicationviewer.cpp. Normally Qt^{TM} Creator generates these files, but at this point we will publish the content to keep it simple.

Note: The content is only valid for Qt^{TM} 4.x and not for Qt^{TM} 5.x. Furthermore the cpp-file does not contain any debug or version reference.

File qmlapplicationviewer.h:

```
/****************************************************************************
**
** Copyright (C) 2012 Nokia Corporation and/or its subsidiary(-ies).
** All rights reserved.
** Contact: Nokia Corporation (qt-info@nokia.com)
**
** This file is part of the QtCore module of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:BSD$
** You may use this file under the terms of the BSD license as follows:
**
** "Redistribution and use in source and binary forms, with or without
** modification, are permitted provided that the following conditions are
** met:
**   * Redistributions of source code must retain the above copyright
**     notice, this list of conditions and the following disclaimer.
**   * Redistributions in binary form must reproduce the above copyright
**     notice, this list of conditions and the following disclaimer in
**     the documentation and/or other materials provided with the
**     distribution.
**   * Neither the name of Nokia Corporation and its Subsidiary(-ies) nor
**     the names of its contributors may be used to endorse or promote
**     products derived from this software without specific prior written
**     permission.
**
** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
** $QT_END_LICENSE$
**
****************************************************************************/

#ifndef QMLAPPLICATIONVIEWER_H
#define QMLAPPLICATIONVIEWER_H

#include <QtDeclarative/QDeclarativeView>

class QmlApplicationViewer : public QDeclarativeView
{
    Q_OBJECT
public:
    enum ScreenOrientation
    {
      ScreenOrientationLockPortrait,
      ScreenOrientationLockLandscape,
      ScreenOrientationAuto
    };

    explicit QmlApplicationViewer (QWidget *parent = 0);
    virtual ~QmlApplicationViewer ();

    static QmlApplicationViewer *create ();

    void setMainQmlFile (const QString &file);
    void addImportPath (const QString &path);

    // Note that this will only have an effect on Symbian and Fremantle.
    void setOrientation (ScreenOrientation orientation);

    void showExpanded ();

private:
    explicit QmlApplicationViewer (QDeclarativeView *view, QWidget *parent);
    class QmlApplicationViewerPrivate *d;
};

QApplication *createApplication (int &argc, char **argv);

#endif // QMLAPPLICATIONVIEWER_H
```

File qmlapplicationviewer.cpp:

```
/* This file was generated by the Qt Quick Application wizard of Qt Creator.
   QmlApplicationViewer is a convenience class containing mobile device specific
   code such as screen orientation handling. Also QML paths and debugging are
   handled here.
   It is recommended not to modify this file, since newer versions of Qt Creator
   may offer an updated version of it. */

#include <QtCore/QDir>
#include <QtCore/QFileInfo>
#include <QtGui/QApplication>
#include <QtDeclarative/QDeclarativeComponent>
#include <QtDeclarative/QDeclarativeEngine>
#include <QtDeclarative/QDeclarativeContext>

#include "qmlapplicationviewer.h"


class QmlApplicationViewerPrivate
{
    QString mainQmlFile;
    friend class QmlApplicationViewer;
    static QString adjustPath (const QString &path);
};
```

```
QString QmlApplicationViewerPrivate::adjustPath (const QString &path)
{
    QString pathInInstallDir = QString::fromLatin1 ("%1/../%2").arg (QCoreApplication::applicationDirPath (), path);
    if (QFileInfo (pathInInstallDir).exists ())
      return pathInInstallDir;
    pathInInstallDir = QString::fromLatin1 ("%1/%2").arg (QCoreApplication::applicationDirPath (), path);
    if (QFileInfo (pathInInstallDir).exists ())
      return pathInInstallDir;
    return path;
}


QmlApplicationViewer::QmlApplicationViewer (QWidget *parent)
     : QDeclarativeView (parent)
     , d (new QmlApplicationViewerPrivate ())
{
    connect (engine (), SIGNAL (quit ()), SLOT (close ()));
    setResizeMode (QDeclarativeView::SizeRootObjectToView);
}


QmlApplicationViewer::~QmlApplicationViewer ()
{
    delete d;
}


QmlApplicationViewer *QmlApplicationViewer::create ()
{
    return new QmlApplicationViewer ();
}


void QmlApplicationViewer::setMainQmlFile (const QString &file)
{
    d->mainQmlFile = QmlApplicationViewerPrivate::adjustPath (file);
    setSource (QUrl::fromLocalFile (d->mainQmlFile));
}


void QmlApplicationViewer::addImportPath (const QString &path)
{
    engine ()->addImportPath (QmlApplicationViewerPrivate::adjustPath (path));
}


void QmlApplicationViewer::setOrientation (ScreenOrientation orientation)
{
    Qt::WidgetAttribute attribute;
    switch (orientation)
    {
      case ScreenOrientationLockPortrait:
            attribute = Qt::WA_LockPortraitOrientation;
            break;
      case ScreenOrientationLockLandscape:
            attribute = Qt::WA_LockLandscapeOrientation;
            break;
      default:
      case ScreenOrientationAuto:
            attribute = Qt::WA_AutoOrientation;
            break;
    };
    setAttribute (attribute, true);
}


void QmlApplicationViewer::showExpanded ()
{
    show ();
}


QApplication *createApplication (int &argc, char **argv)
{
    return new QApplication (argc, argv);
}
```

The Qt^{TM} Quick project provides a set of examples on the following webpage

http://qt-project.org/doc/qt-4.8/qdeclarativeexamples.html

The first example uses the sourcecode from 'Web Browser'. This project requires the WebKit plugin, an open source web browser rendering engine. You can find a more detailed description on

http://qt-project.org/doc/qt-4.8/qtwebkit.html#details

Plugins for Qt^{TM} are normally available in the following directory (target device)

/usr/lib/arm-linux-gnueabihf/qt4/imports

If this directory imports or the subdirectory QtWebKit do not exist download the plugin with

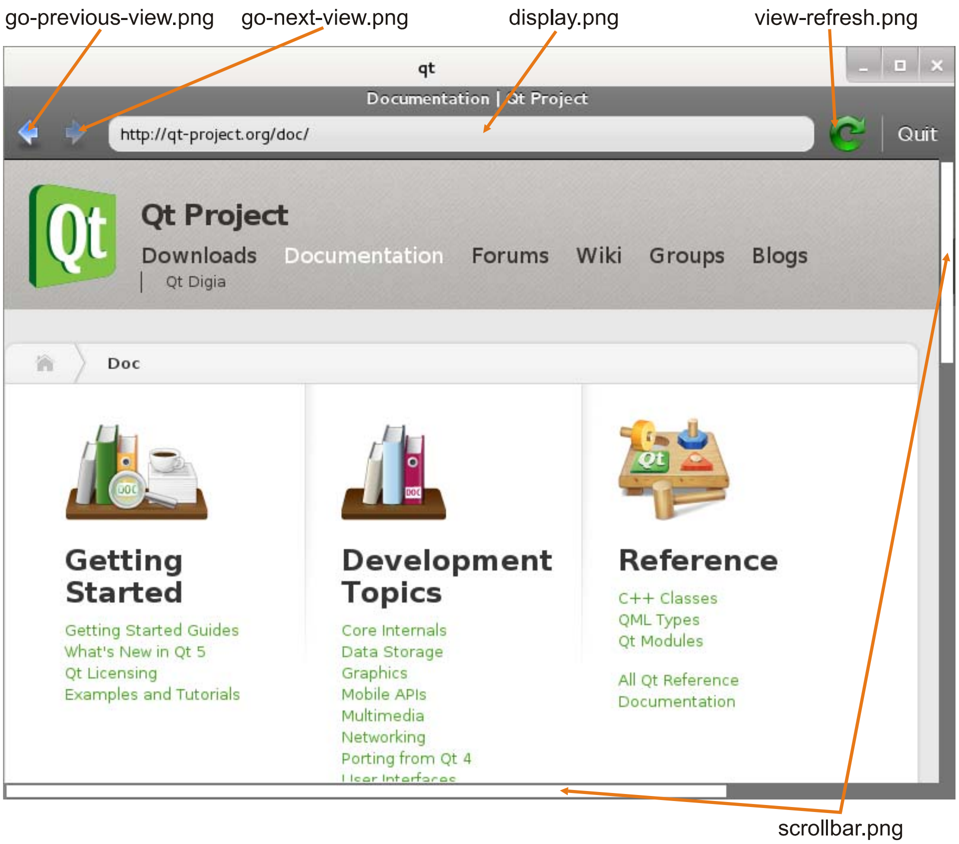
```
sudo apt-get install libqtwebkit-qmlwebkitplugin
```

Now let us discuss the main project. You should download the sourcecode from the 'Web Browser' page (the file webbrowser.svg is not really needed). Then create a new directory on the target hardware, for example '/home/ktam3874/qt/webbrowser/qml', and copy all qml-files into this directory. However, there is a problem with the png-files (integrated in the qml-files) - the qt4-demos package does not contain these files but this issue can be solved by a web search with little effort. Necessary files:

display.png, edit-delete.png, go-jump-locationbar.png, go-next-view.png
go-previous-view.png, scrollbar.png, titlebar-bg.png and view-refresh.png

Thereafter create a new subdirectory '/home/ktam3874/qt/webbrowser/qml/pics' and copy the png-files into this directory.

The following picture gives an impression of the program.



One possible Makefile might look like this:

```
CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDeclarative -l QtScript -l QtSql -l QtNetwork -l QtXmlPatterns -l glib-2.0 \
           -l png12 -l expat -l pcre -l uuid -l z

all: qt

moc_qmlapplicationviewer.cpp: qmlapplicationviewer.h
    /usr/bin/moc-qt4 -I/usr/include/qt4 qmlapplicationviewer.h -o moc_qmlapplicationviewer.cpp

qmlapplicationviewer.o: qmlapplicationviewer.cpp
main.o: main.cpp moc_qmlapplicationviewer.cpp qmlapplicationviewer.cpp

qt: main.o qmlapplicationviewer.o
    $(CXX) -o $@ main.o qmlapplicationviewer.o $(LDFLAGS)

clean:
    rm qt main.o qmlapplicationviewer.o
```

Some minor deviations from the original sourcecode are marked with orange color.

File main.cpp:

```
/****************************************************************************
** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
** Contact: http://www.qt-project.org/legal
** This file is part of the examples of the Qt Toolkit.
**
```

```
** $QT_BEGIN_LICENSE:BSD$
** You may use this file under the terms of the BSD license as follows:
** "Redistribution and use in source and binary forms, with or without
** modification, are permitted provided that the following conditions are
** met:
**   * Redistributions of source code must retain the above copyright
**     notice, this list of conditions and the following disclaimer.
**   * Redistributions in binary form must reproduce the above copyright
**     notice, this list of conditions and the following disclaimer in
**     the documentation and/or other materials provided with the
**     distribution.
**   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
**     of its contributors may be used to endorse or promote products derived
**     from this software without specific prior written permission.
**
** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
** $QT_END_LICENSE$
****************************************************************************/

#include <QtGui/QApplication>
#include "qmlapplicationviewer.h"
#include "moc_qmlapplicationviewer.cpp"

Q_DECL_EXPORT int main (int argc, char *argv [])
{
    QApplication app (argc, argv);

    QmlApplicationViewer viewer;
    viewer.setOrientation (QmlApplicationViewer::ScreenOrientationAuto);
    viewer.setMainQmlFile (QLatin1String ("qml/webbrowser.qml"));
    viewer.showExpanded ();

    return app.exec ();
}
```

File webbrowser.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0
import QtWebKit 1.0

/* directory for the other qml-files - not required if all qml-files are available in the same directory ! */
/* import "content" */
```

```
Rectangle
{
    id: webBrowser
    property string urlString : "http://qt-project.org/doc/"
    width: 800
    height: 600
    color: "#343434"

    FlickableWebView
    {
        id: webView
        url: webBrowser.urlString
        onProgressChanged: header.urlChanged = false
        anchors
        {
            top: headerSpace.bottom
            left: parent.left
            right: parent.right
            bottom: parent.bottom
        }
    }

    Item
    {
        id: headerSpace
        width: parent.width
        height: 62
    }

    Header
    {
        id: header
        editUrl: webBrowser.urlString
        width: headerSpace.width
        height: headerSpace.height
    }

    ScrollBar
    {
        scrollArea: webView
        width: 14
        anchors
        {
            right: parent.right
            top: header.bottom
            bottom: parent.bottom
        }
    }

    ScrollBar
    {
        scrollArea: webView
        height: 14
        orientation: Qt.Horizontal
        anchors
        {
            right: parent.right
            rightMargin: 14
            left: parent.left
            bottom: parent.bottom
        }
    }
}
```

File Button.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    property alias image: icon.source
    property variant action
    signal clicked
    width: 40
    height: parent.height

    Image
    {
        id: icon
        anchors.centerIn: parent
        opacity: if (action != undefined) { action.enabled ? 1.0 : 0.4 } else 1
    }

    MouseArea
    {
        anchors
        {
            fill: parent
            topMargin: -10
            bottomMargin: -10
        }

        onClicked:
        {
            if (action != undefined)
                action.trigger ()
            parent.clicked ()
        }
    }
}
```

File FlickableWebView.qml (header with Qt™ license see 'main.cpp'):

```
import QtQuick 1.0
import QtWebKit 1.0

Flickable
{
    property alias title: webView.title
    property alias icon: webView.icon
    property alias progress: webView.progress
    property alias url: webView.url
    property alias back: webView.back
    property alias stop: webView.stop
    property alias reload: webView.reload
    property alias forward: webView.forward

    id: flickable
    width: parent.width
    contentWidth: Math.max (parent.width, webView.width)
    contentHeight: Math.max (parent.height, webView.height)
    anchors.top: headerSpace.bottom
    anchors.bottom: parent.top
    anchors.left: parent.left
    anchors.right: parent.right
    pressDelay: 200

    onWidthChanged:
    {
      // Expand (but not above 1:1) if otherwise would be smaller that available width.
      if (width > webView.width*webView.contentsScale && webView.contentsScale < 1.0)
        webView.contentsScale = width / webView.width * webView.contentsScale
    }

    WebView
    {
      id: webView
      transformOrigin: Item.TopLeft

      function fixUrl (url)
      {
        if (url == "")
          return url
        if (url[0] == "/")
          return "file://" + url
        if (url.indexOf (":") < 0)
        {
          if (url.indexOf (".") < 0 || url.indexOf (" ") >= 0)
          {
            // Fall back to a search engine; hard-code Wikipedia
            return "http://en.wikipedia.org/w/index.php?search=" + url
          }
          else
          {
            return "http://" + url
          }
```

```
  }
  return url
}
url: fixUrl (webBrowser.urlString)
smooth: false     // We don't want smooth scaling, since we only scale during (fast) transitions
focus: true
onAlert: console.log (message)
function doZoom (zoom, centerX, centerY)
{
  if (centerX)
  {
    var sc = zoom*contentsScale
    scaleAnim.to = sc
    flickVX.from = flickable.contentX
    flickVX.to = Math.max (0, Math.min (centerX - flickable.width / 2, webView.width * sc - flickable.width))
    finalX.value = flickVX.to
    flickVY.from = flickable.contentY
    flickVY.to = Math.max (0, Math.min (centerY - flickable.height / 2, webView.height * sc - flickable.height))
    finalY.value = flickVY.to
    quickZoom.start ()
  }
}
Keys.onLeftPressed: webView.contentsScale -= 0.1
Keys.onRightPressed: webView.contentsScale += 0.1
preferredWidth: flickable.width
preferredHeight: flickable.height
contentsScale: 1
onContentsSizeChanged:
{
  // zoom out
  contentsScale = Math.min (1, flickable.width / contentsSize.width)
}
onUrlChanged:
{
  // got to topleft
  flickable.contentX = 0
  flickable.contentY = 0
  if (url != null)
  {
    header.editUrl = url.toString ()
  }
}
onDoubleClick:
{
  if (! heuristicZoom (clickX, clickY, 2.5))
  {
    var zf = flickable.width / contentsSize.width
    if (zf >= contentsScale)
      zf = 2.0 * contentsScale     // zoom in (else zooming out)
    doZoom (zf, clickX * zf, clickY * zf)
  }
}

SequentialAnimation
{
  id: quickZoom
  PropertyAction
  {
    target: webView
    property: "renderingEnabled"
    value: false
  }
  ParallelAnimation
  {

    NumberAnimation
    {
      id: scaleAnim
      target: webView
      property: "contentsScale"
      // the to property is set before calling
      easing.type: Easing.Linear
      duration: 200
    }

    NumberAnimation
    {
      id: flickVX
      target: flickable
      property: "contentX"
      easing.type: Easing.Linear
      duration: 200
      from: 0     // set before calling
      to: 0     // set before calling
    }

    NumberAnimation
    {
      id: flickVY
      target: flickable
      property: "contentY"
      easing.type: Easing.Linear
      duration: 200
      from: 0     // set before calling
      to: 0     // set before calling
    }
  }

  // Have to set the contentXY, since the above 2 size changes may have started a correction if
  // contentsScale < 1.0.
```

```
          PropertyAction
          {
            id: finalX
            target: flickable
            property: "contentX"
            value: 0      // set before calling
          }
          PropertyAction
          {
            id: finalY
            target: flickable
            property: "contentY"
            value: 0      // set before calling
          }
          PropertyAction
          {
            target: webView
            property: "renderingEnabled"
            value: true
          }
        }
        onZoomTo: doZoom (zoom, centerX, centerY)
      }
}
```

File Header.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Image
{
    id: header
    property alias editUrl: urlInput.url
    property bool urlChanged: false
    source: "pics/titlebar-bg.png"
    fillMode: Image.TileHorizontally

    x: webView.contentX < 0 ? -webView.contentX : webView.contentX > webView.contentWidth - webView.width
                                          ? -webView.contentX + webView.contentWidth - webView.width : 0
    y:
    {
      if (webView.progress < 1.0)
        return 0
      else
      {
        webView.contentY < 0 ? -webView.contentY : webView.contentY > height ? -height : -webView.contentY
      }
    }

    Column
    {
      width: parent.width
```

```
Item
{
  width: parent.width
  height: 20
  Text
  {
    anchors.centerIn: parent
    text: webView.title
    font.pixelSize: 14
    font.bold: true
    color: "white"
    styleColor: "black"
    style: Text.Sunken
  }
}

Item
{
  width: parent.width
  height: 40

  Button
  {
    id: backButton
    action: webView.back
    image: "pics/go-previous-view.png"
    anchors
    {
      left: parent.left
      bottom: parent.bottom
    }
  }

  Button
  {
    id: nextButton
    anchors.left: backButton.right
    action: webView.forward
    image: "pics/go-next-view.png"
  }

  UrlInput
  {
    id: urlInput
    anchors
    {
      left: nextButton.right
      right: reloadButton.left
    }
    image: "pics/display.png"
    onUrlEntered:
    {
      webBrowser.urlString = url
      webBrowser.focus = true
      header.urlChanged = false
    }
    onUrlChanged: header.urlChanged = true
  }
  Button
  {
    id: reloadButton
    anchors
    {
      right: quitButton.left
      rightMargin: 10
    }
    action: webView.reload
    image: "pics/view-refresh.png"
    visible: webView.progress == 1.0 && !header.urlChanged
  }
  Text
  {
    id: quitButton
    color: "white"
    style: Text.Sunken
    anchors.right: parent.right
    anchors.top: parent.top
    anchors.bottom: parent.bottom
    verticalAlignment: Text.AlignVCenter
    horizontalAlignment: Text.AlignHCenter
    font.pixelSize: 18
    width: 60
    text: "Quit"
    MouseArea
    {
      anchors.fill: parent
      onClicked: Qt.quit ()
    }
    Rectangle
    {
      width: 1
      y: 5
      height: parent.height - 10
      anchors.right: parent.left
      color: "darkgray"
    }
  }
  Button
  {
    id: stopButton
    anchors
    {
      right: quitButton.left
      rightMargin: 10
    }
```

```
                action: webView.stop
                image: "pics/edit-delete.png"
                visible: webView.progress < 1.0 && !header.urlChanged
            }
            Button
            {
                id: goButton
                anchors
                {
                    right: parent.right
                    rightMargin: 4
                }
                onClicked:
                {
                    webBrowser.urlString = urlInput.url
                    webBrowser.focus = true
                    header.urlChanged = false
                }
                image: "pics/go-jump-locationbar.png"
                visible: header.urlChanged
            }
        }
    }
}
```

File ScrollBar.qml (header with Qt™ license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: container
    property variant scrollArea
    property variant orientation: Qt.Vertical
    opacity: 0
    function position ()
    {
        var ny = 0
        if (container.orientation == Qt.Vertical)
            ny = scrollArea.visibleArea.yPosition * container.height
        else
            ny = scrollArea.visibleArea.xPosition * container.width
        if (ny > 2)
            return ny
        else
            return 2
    }
```

```
function size ()
{
  var nh, ny

  if (container.orientation == Qt.Vertical)
    nh = scrollArea.visibleArea.heightRatio * container.height
  else
    nh = scrollArea.visibleArea.widthRatio * container.width
  if (container.orientation == Qt.Vertical)
    ny = scrollArea.visibleArea.yPosition * container.height
  else
    ny = scrollArea.visibleArea.xPosition * container.width

  if (ny > 3)
  {
    var t

    if (container.orientation == Qt.Vertical)
      t = Math.ceil (container.height - 3 - ny)
    else
      t = Math.ceil (container.width - 3 - ny)
    if (nh > t)
      return t
    else
      return nh
  }
  else
    return nh + ny
}

Rectangle
{
  anchors.fill: parent
  color: "Black"
  opacity: 0.5
}

BorderImage
{
  source: "pics/scrollbar.png"
  border
  {
    left: 1
    right: 1
    top: 1
    bottom: 1
  }
  x: container.orientation == Qt.Vertical ? 2 : position ()
  width: container.orientation == Qt.Vertical ? container.width - 4 : size ()
  y: container.orientation == Qt.Vertical ? position () : 2
  height: container.orientation == Qt.Vertical ? size () : container.height - 4
}
```

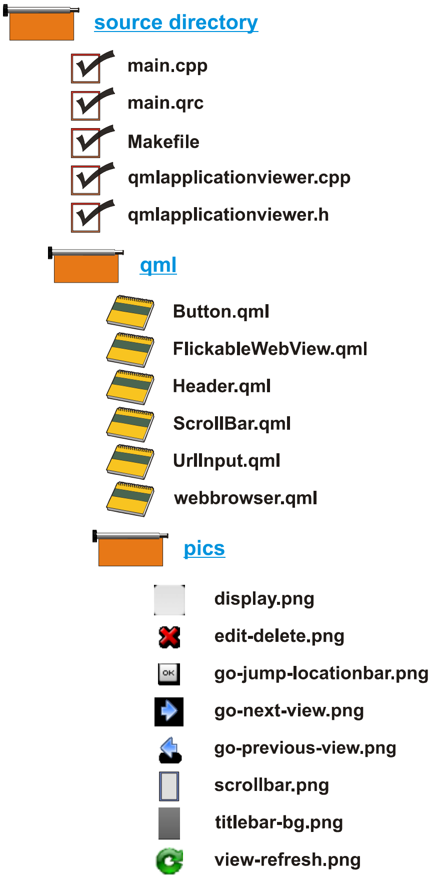
```
    states: State
    {
      name: "visible"
      when: container.orientation == Qt.Vertical ? scrollArea.movingVertically : scrollArea.movingHorizontally
      PropertyChanges
      {
        target: container
        opacity: 1.0
      }
    }
    transitions: Transition
    {
      from: "visible"
      to: ""
      NumberAnimation
      {
        properties: "opacity"
        duration: 600
      }
    }
}
```

File UrlInput.qml (header with Qt™ license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: container
    property alias image: bg.source
    property alias url: urlText.text

    signal urlEntered (string url)
    signal urlChanged

    width: parent.height
    height: parent.height

    BorderImage
    {
      id: bg
      rotation: 180
      x: 8
      width: parent.width - 16
      height: 30
      anchors.verticalCenter: parent.verticalCenter
      border
      {
        left: 10
        top: 10
        right: 10
        bottom: 10
      }
    }
```

```
        Rectangle
        {
          anchors.bottom: bg.bottom
          x: 18
          height: 4
          color: "#63b1ed"
          width: (bg.width - 20) * webView.progress
          opacity: webView.progress == 1.0 ? 0.0 : 1.0
        }

        TextInput
        {
          id: urlText
          horizontalAlignment: TextEdit.AlignLeft
          font.pixelSize: 14
          onTextChanged: container.urlChanged ()

          Keys.onEscapePressed:
          {
            urlText.text = webView.url
            webView.focus = true
          }

          Keys.onEnterPressed:
          {
            container.urlEntered (urlText.text)
            webView.focus = true
          }

          Keys.onReturnPressed:
          {
            container.urlEntered (urlText.text)
            webView.focus = true
          }

          anchors
          {
            left: parent.left
            right: parent.right
            leftMargin: 18
            rightMargin: 18
            verticalCenter: parent.verticalCenter
          }
        }
    }
```

For debugging purposes this approach maintains good results but the real application should have binary format. To this end create a new subdirectory 'qml' and into this subdirectory another subdirectory named 'pics' on the x86 Desktop PC with the cross compiler. The picture illustrates the directory structure:



How the integration of QML-files could be carried out? To realize that you have to create a new resource file like this

File main.qrc:

```
<!DOCTYPE RCC><RCC version="1.0">
<qresource>
<file>qml/webbrowser.qml</file>
<file>qml/Button.qml</file>
<file>qml/FlickableWebView.qml</file>
<file>qml/Header.qml</file>
<file>qml/ScrollBar.qml</file>
<file>qml/UrlInput.qml</file>
<file>qml/pics/display.png</file>
<file>qml/pics/go-next-view.png</file>
<file>qml/pics/go-previous-view.png</file>
<file>qml/pics/scrollbar.png</file>
<file>qml/pics/view-refresh.png</file>
<file>qml/pics/edit-delete.png</file>
<file>qml/pics/go-jump-locationbar.png</file>
<file>qml/pics/titlebar-bg.png</file>
</qresource>
</RCC>
```

and change the content of the Makefile.

```
CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDeclarative -l QtScript -l QtSql -l QtNetwork -l QtXmlPatterns -l glib-2.0 \
              -l png12 -l expat -l pcre -l uuid -l z

all: qt

qrc_main.cpp: main.qrc
    /usr/bin/rcc -name main main.qrc -o qrc_main.cpp

moc_qmlapplicationviewer.cpp: qmlapplicationviewer.h
    /usr/bin/moc-qt4 -I/usr/include/qt4 qmlapplicationviewer.h -o moc_qmlapplicationviewer.cpp

qmlapplicationviewer.o: qmlapplicationviewer.cpp
main.o: main.cpp moc_qmlapplicationviewer.cpp qmlapplicationviewer.cpp

qt: main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o
    $(CXX) -o $@ main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o $(LDFLAGS)

clean:
    rm qt main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o
```

You also need some minor modifications (orange color) in main.cpp:

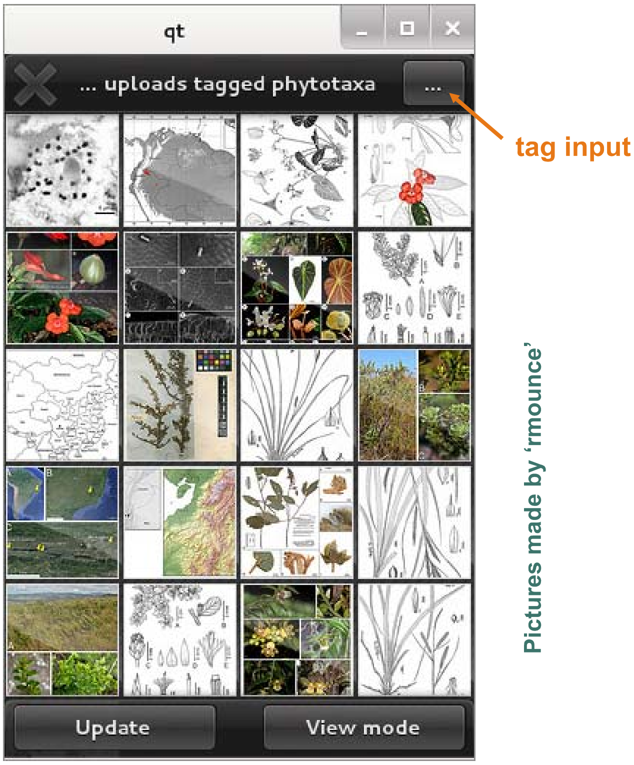
```
/****************************************************************************
 ** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
 ** Contact: http://www.qt-project.org/legal
** This file is part of the examples of the Qt Toolkit.
 **
 ** $QT_BEGIN_LICENSE:BSD$
 ** You may use this file under the terms of the BSD license as follows:
** "Redistribution and use in source and binary forms, with or without
 ** modification, are permitted provided that the following conditions are
 ** met:
 **   * Redistributions of source code must retain the above copyright
 **     notice, this list of conditions and the following disclaimer.
 **   * Redistributions in binary form must reproduce the above copyright
 **     notice, this list of conditions and the following disclaimer in
 **     the documentation and/or other materials provided with the
 **     distribution.
 **   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
 **     of its contributors may be used to endorse or promote products derived
 **     from this software without specific prior written permission.
 **
** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 ** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 ** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 ** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 ** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 ** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 ** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 ** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 ** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 ** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 ** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
** $QT_END_LICENSE$
****************************************************************************/
```

```
#include <QtGui/QApplication>
#include "qmlapplicationviewer.h"
/* #include "moc_qmlapplicationviewer.cpp" */      /* not necessary, see the Makefile modifications */

Q_DECL_EXPORT int main (int argc, char *argv [])
{
    QApplication app (argc, argv);

    QmlApplicationViewer viewer;
    viewer.setOrientation (QmlApplicationViewer::ScreenOrientationAuto);
    viewer.setSource (QUrl ("qrc:///qml/webbrowser.qml"));
    viewer.showExpanded ();

    return app.exec ();
}
```

The second example realizes a Flickr™ browser with sourcecode of 'Flickr Mobile'. Flickr™ represents a web-based photo management and sharing service by using web feeds as data format. A web feed implies typically HTML™ for content or links to webpages. This example is based on RSS 2 web feed format. Many browsers can support RSS 1.x/2.x and the Atom 1.x format.

However, there is one big disadvantage: Flickr™ only provides the latest 20 items for a single query. An anonymous data request cannot receive more items.

The picture below shows the initial screen:



Please note: the tag 'phytotaxa' is licensed under 'Creative Commons Attribution License' and can be published by naming the author. What means the term 'tag'? You can equip your photos and videos with a tag (optional), which is like a keyword or category label.

The next two pictures demonstrate subfunctions:

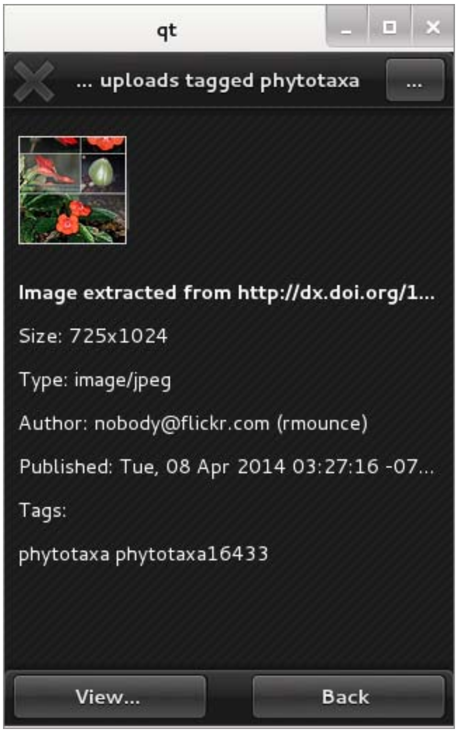


Image extracted from http://dx.doi.org/1...

Size: 725x1024

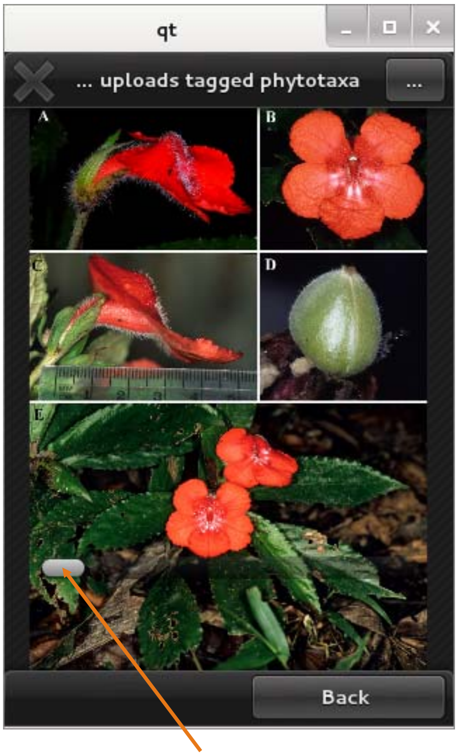
Type: image/jpeg

Author: nobody@flickr.com (rmounce)

Published: Tue, 08 Apr 2014 03:27:16 -07...

Tags:

phytotaxa phytotaxa16433



zoom slider

One possible Makefile might look like this:

```
CXX   := arm-linux-gnueabihf-g++ -march=armv7 -I/usr/include/qt4
LDFLAGS := -l QtCore -l QtGui -l QtDeclarative -l QtScript -l QtSql -l QtNetwork -l QtXmlPatterns -l glib-2.0 \
             -l png12 -l expat -l pcre -l uuid -l z

all: qt

qrc_main.cpp: main.qrc
    /usr/bin/rcc -name main main.qrc -o qrc_main.cpp

moc_qmlapplicationviewer.cpp: qmlapplicationviewer.h
    /usr/bin/moc-qt4 -I/usr/include/qt4 qmlapplicationviewer.h -o moc_qmlapplicationviewer.cpp

qmlapplicationviewer.o: qmlapplicationviewer.cpp
main.o: main.cpp moc_qmlapplicationviewer.cpp qmlapplicationviewer.cpp

qt: main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o
    $(CXX) -o $@ main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o $(LDFLAGS)

clean:
    rm qt main.o qmlapplicationviewer.o qrc_main.o moc_qmlapplicationviewer.o
```

Some minor deviations from the original sourcecode are marked with orange color.

File main.cpp:

```
/****************************************************************************
** Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
** Contact: http://www.qt-project.org/legal
** This file is part of the examples of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:BSD$
** You may use this file under the terms of the BSD license as follows:
** "Redistribution and use in source and binary forms, with or without
** modification, are permitted provided that the following conditions are
** met:
**   * Redistributions of source code must retain the above copyright
**     notice, this list of conditions and the following disclaimer.
**   * Redistributions in binary form must reproduce the above copyright
**     notice, this list of conditions and the following disclaimer in
**     the documentation and/or other materials provided with the
**     distribution.
**   * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
**     of its contributors may be used to endorse or promote products derived
**     from this software without specific prior written permission.
**
** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
** "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
** LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
** A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
** OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
** SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
** LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
** DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
** THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
** (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
** OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
** $QT_END_LICENSE$
****************************************************************************/

#include <QtGui/QApplication>
#include "qmlapplicationviewer.h"

Q_DECL_EXPORT int main (int argc, char *argv [])
{
    QApplication app (argc, argv);

    QmlApplicationViewer viewer;
    viewer.setOrientation (QmlApplicationViewer::ScreenOrientationLockLandscape);
    viewer.setSource (QUrl ("qrc:///qml/flickr.qml"));
    viewer.showExpanded ();

    return app.exec ();
}
```

In contrast to the example which uses the subdirectories 'common' and 'mobile' all qml-files are located in the same subdirectory 'qml'. This approach involves some changes in the qml-files (orange color).

File main.qrc:

```
<!DOCTYPE RCC><RCC version="1.0">
<qresource>
<file>qml/flickr.qml</file>
<file>qml/Button.qml</file>
<file>qml/GridDelegate.qml</file>
<file>qml/ImageDetails.qml</file>
<file>qml/ListDelegate.qml</file>
<file>qml/Progress.qml</file>
<file>qml/RssModel.qml</file>
<file>qml/ScrollBar.qml</file>
<file>qml/Slider.qml</file>
<file>qml/TitleBar.qml</file>
<file>qml/ToolBar.qml</file>
<file>qml/images/gloss.png</file>
<file>qml/images/lineedit.png</file>
<file>qml/images/quit.png</file>
<file>qml/images/stripes.png</file>
<file>qml/images/titlebar.png</file>
<file>qml/images/toolbutton.png</file>
<file>qml/images/lineedit.sci</file>
<file>qml/images/titlebar.sci</file>
<file>qml/images/toolbutton.sci</file>
</qresource>
</RCC>
```

File flickr.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

// Both lines are not necessary
// import "common" as Common
// import "mobile" as Mobile

Item
{
    id: screen
    width: 320
    height: 480
    property bool inListView : false

    Rectangle
    {
      id: background
      anchors.fill: parent
      color: "#343434"
```

```
Image
{
  source: "images/stripes.png"
  fillMode: Image.Tile
  anchors.fill: parent
  opacity: 0.3
}
RssModel
{
  id: rssModel
}
Item
{
  id: views
  width: parent.width
  anchors.top: titleBar.bottom
  anchors.bottom: toolBar.top
  GridView
  {
    id: photoGridView
    model: rssModel
    delegate: GridDelegate {}
    cacheBuffer: 1000
    cellWidth: (parent.width - 2) / 4
    cellHeight: cellWidth
    width: parent.width
    height: parent.height
  }
  ListView
  {
    id: photoListView
    model: rssModel
    delegate: ListDelegate {}
    width: parent.width
    height: parent.height
    x: -(parent.width * 1.5)
    cacheBuffer: 100
  }
  states: State
  {
    name: "ListView"
    when: screen.inListView == true
    PropertyChanges
    {
      target: photoListView
      x: 0
    }
```

```
      PropertyChanges
      {
         target: photoGridView
         x: -(parent.width * 1.5)
      }
   }
   transitions: Transition
   {
      NumberAnimation
      {
         properties: "x"
         duration: 500
         easing.type: Easing.InOutQuad
      }
   }
   ImageDetails
   {
      id: imageDetails
      width: parent.width
      anchors.left: views.right
      height: parent.height
   }
   Item
   {
      id: foreground
      anchors.fill: parent
   }
}
TitleBar
{
   id: titleBar
   width: parent.width
   height: 40
   opacity: 0.9
}
ToolBar
{
   id: toolBar
   height: 40
   anchors.bottom: parent.bottom
   width: parent.width
   opacity: 0.9
   button1Label: "Update"
   button2Label: "View mode"
   onButton1Clicked: rssModel.reload ()
   onButton2Clicked:
      if (screen.inListView == true)
         screen.inListView = false
      else
         screen.inListView = true
}
```

```
Connections
{
    target: imageDetails
    onClosed:
    {
        if (background.state == "DetailedView")
        {
            background.state = ''
            imageDetails.photoUrl = ""
        }
    }
}

states: State
{
    name: "DetailedView"

    PropertyChanges
    {
        target: views
        x: -parent.width
    }

    PropertyChanges
    {
        target: toolBar
        button1Label: "View..."
    }

    PropertyChanges
    {
        target: toolBar
        onButton1Clicked:
            if (imageDetails.state=='')
                imageDetails.state='Back'
            else
                imageDetails.state=''
    }

    PropertyChanges
    {
        target: toolBar
        button2Label: "Back"
    }

    PropertyChanges
    {
        target: toolBar
        onButton2Clicked: imageDetails.closed ()
    }
}
```

```
      transitions: Transition
      {
        NumberAnimation
        {
          properties: "x"
          duration: 500
          easing.type: Easing.InOutQuad
        }
      }
    }
}
```

File Button.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: container
    signal clicked
    property string text

    BorderImage
    {
      id: buttonImage
      source: "images/toolbutton.sci"
      width: container.width
      height: container.height
    }

    BorderImage
    {
      id: pressed
      opacity: 0
      source: "images/toolbutton.sci"
      width: container.width
      height: container.height
    }

    MouseArea
    {
      id: mouseRegion
      anchors.fill: buttonImage
      onClicked:
      {
        container.clicked ()
      }
    }

    Text
    {
      color: "white"
      anchors.centerIn: buttonImage
      font.bold: true
      font.pixelSize: 15
```

```
        text: container.text
        style: Text.Raised
        styleColor: "black"
    }
    states: [
        State
        {
            name: "Pressed"
            when: mouseRegion.pressed == true
            PropertyChanges
            {
                target: pressed
                opacity: 1
            }
        }
    ]
}
```

File GridDelegate.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: wrapper
    width: GridView.view.cellWidth
    height: GridView.view.cellHeight
    function photoClicked ()
    {
        imageDetails.photoTitle = title
        imageDetails.photoTags = tags
        imageDetails.photoWidth = photoWidth
        imageDetails.photoHeight = photoHeight
        imageDetails.photoType = photoType
        imageDetails.photoAuthor = photoAuthor
        imageDetails.photoDate = photoDate
        imageDetails.photoUrl = url
        imageDetails.rating = 0
        scaleMe.state = "Details"
    }
    Item
    {
        anchors.centerIn: parent
        scale: 0.0
        Behavior on scale
        {
            NumberAnimation
            {
                easing.type: Easing.InOutQuad
            }
        }
        id: scaleMe
```

```
Item
{
  width: 77
  height: 77
  anchors.centerIn: parent
  Rectangle
  {
    id: whiteRect
    width: 77
    height: 77
    color: "#dddddd"
    smooth: true
    Image
    {
      id: thumb
      source: imagePath
      x: 1
      y: 1
      smooth: true
    }

    Image
    {
      source: "images/gloss.png"
    }
  }
}

Connections
{
  target: toolBar
  onButton2Clicked:
    if (scaleMe.state == 'Details' ) scaleMe.state = 'Show'
}

states: [
  State
  {
    name: "Show"
    when: thumb.status == Image.Ready
    PropertyChanges
    {
      target: scaleMe
      scale: 1
    }
  },

  State
  {
    name: "Details"
    PropertyChanges
    {
      target: scaleMe
      scale: 1
    }
```

```
      ParentChange
      {
        target: whiteRect
        x: 10
        y: 20
        parent: imageDetails.frontContainer
      }

      PropertyChanges
      {
        target: background
        state: "DetailedView"
      }
    }
  ]

  transitions: [
    Transition
    {
      from: "Show"
      to: "Details"
      ParentAnimation
      {
        via: foreground
        NumberAnimation
        {
          properties: "x,y"
          duration: 500
          easing.type: Easing.InOutQuad
        }
      }
    },

    Transition
    {
      from: "Details"
      to: "Show"
      ParentAnimation
      {
        via: foreground
        NumberAnimation
        {
          properties: "x,y"
          duration: 500
          easing.type: Easing.InOutQuad
        }
      }
    }
  ]
}
```

```
    MouseArea
    {
      anchors.fill: wrapper
      onClicked: photoClicked ()
    }
}
```

File ImageDetails.qml (header with Qt™ license see 'main.cpp'):

```
import QtQuick 1.0

// This line is not necessary
// import "../common" as Common

Flipable
{
    id: container
    property alias frontContainer: containerFront
    property string photoTitle: ""
    property string photoTags: ""
    property int photoWidth
    property int photoHeight
    property string photoType
    property string photoAuthor
    property string photoDate
    property string photoUrl
    property int rating: 2
    property variant prevScale: 1.0

    signal closed

    transform: Rotation
    {
      id: itemRotation
      origin.x: container.width / 2
      axis.y: 1
      axis.z: 0
    }

    front: Item
    {
      id: containerFront
      anchors.fill: container

      Rectangle
      {
        anchors.fill: parent
        color: "black"
        opacity: 0.4
      }
```

```
Column
{
  spacing: 10
  anchors
  {
    left: parent.left
    leftMargin: 10
    right: parent.right
    rightMargin: 10
    top: parent.top
    topMargin: 120
  }
  Text
  {
    font.bold: true
    color: "white"
    elide: Text.ElideRight
    text: container.photoTitle
    width: parent.width
  }
  Text
  {
    color: "white"
    elide: Text.ElideRight
    text: "Size: " + container.photoWidth + 'x' + container.photoHeight
    width: parent.width
  }
  Text
  {
    color: "white"
    elide: Text.ElideRight
    text: "Type: " + container.photoType
    width: parent.width
  }
  Text
  {
    color: "white"
    elide: Text.ElideRight
    text: "Author: " + container.photoAuthor
    width: parent.width
  }
  Text
  {
    color: "white"
    elide: Text.ElideRight
    text: "Published: " + container.photoDate
    width: parent.width
  }
```

```
            Text
            {
               color: "white"
               elide: Text.ElideRight
               text: container.photoTags == "" ? "" : "Tags: "
               width: parent.width
            }

            Text
            {
               color: "white"
               elide: Text.ElideRight
               text: container.photoTags
               width: parent.width
            }
         }
      }

      back: Item
      {
         anchors.fill: container
         Rectangle
         {
            anchors.fill: parent
            color: "black"
            opacity: 0.4
         }

         Progress
         {
            anchors.centerIn: parent
            width: 200
            height: 22
            progress: bigImage.progress
            visible: bigImage.status != Image.Ready
         }

         Flickable
         {
            id: flickable
            anchors.fill: parent
            clip: true
            contentWidth: imageContainer.width
            contentHeight: imageContainer.height

            function updateMinimumScale ()
            {
               if (bigImage.status == Image.Ready && bigImage.width != 0)
               {
                  slider.minimum = Math.min (flickable.width / bigImage.width, flickable.height / bigImage.height)
                  if (bigImage.width * slider.value > flickable.width)
                  {
                     var xoff = (flickable.width / 2 + flickable.contentX) * slider.value / prevScale
                     flickable.contentX = xoff - flickable.width / 2
                  }
                  if (bigImage.height * slider.value > flickable.height)
                  {
                     var yoff = (flickable.height / 2 + flickable.contentY) * slider.value / prevScale
                     flickable.contentY = yoff - flickable.height / 2
                  }
                  prevScale = slider.value
               }
            }

            onWidthChanged: updateMinimumScale ()
            onHeightChanged: updateMinimumScale ()

            Item
            {
               id: imageContainer
               width: Math.max (bigImage.width * bigImage.scale, flickable.width)
               height: Math.max (bigImage.height * bigImage.scale, flickable.height)

               Image
               {
                  id: bigImage
                  source: container.photoUrl
                  scale: slider.value
                  anchors.centerIn: parent
                  smooth: !flickable.movingVertically
                  onStatusChanged:
                  {
                     // Default scale shows the entire image.
                     if (bigImage.status == Image.Ready && bigImage.width != 0)
                     {
                        slider.minimum = Math.min (flickable.width / bigImage.width, flickable.height / bigImage.height)
                        prevScale = Math.min (slider.minimum, 1)
                        slider.value = prevScale
                     }
                  }
               }
            }
         }

         Text
         {
            text: "Image Unavailable"
            visible: bigImage.status == Image.Error
            anchors.centerIn: parent
            color: "white"
            font.bold: true
         }

         Slider
         {
            id: slider
            visible:
            {
               bigImage.status == Image.Ready && maximum > minimum
            }
            anchors
            {
               bottom: parent.bottom
               bottomMargin: 65
               left: parent.left
               leftMargin: 25
               right: parent.right
               rightMargin: 25
            }

            onValueChanged:
            {
               if (bigImage.width * value > flickable.width)
               {
                  var xoff = (flickable.width / 2 + flickable.contentX) * value / prevScale
                  flickable.contentX = xoff - flickable.width / 2
               }

               if (bigImage.height * value > flickable.height)
               {
                  var yoff = (flickable.height / 2 + flickable.contentY) * value / prevScale
                  flickable.contentY = yoff - flickable.height / 2
               }
               prevScale = value
            }
         }
}

states: State
{
   name: "Back"

   PropertyChanges
   {
      target: itemRotation
      angle: 180
   }

   PropertyChanges
   {
      target: toolBar
      button2Visible: false
   }

   PropertyChanges
   {
      target: toolBar
      button1Label: "Back"
   }
}
```

```
    transitions: Transition
    {
      SequentialAnimation
      {
        PropertyAction
        {
          target: bigImage
          property: "smooth"
          value: false
        }
        NumberAnimation
        {
          easing.type: Easing.InOutQuad
          properties: "angle"
          duration: 500
        }
        PropertyAction
        {
          target: bigImage
          property: "smooth"
          value: !flickable.movingVertically
        }
      }
    }
}
```

File ListDelegate.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Component
{
    Item
    {
      id: wrapper
      width: wrapper.ListView.view.width
      height: 86
      Item
      {
        id: moveMe
        Rectangle
        {
          color: "black"
          opacity: index % 2 ? 0.2 : 0.4
          height: 84
          width: wrapper.width
          y: 1
        }
```

```
Rectangle
{
  x: 6
  y: 4
  width: 77
  height: 77
  color: "white"
  smooth: true

  Image
  {
    source: imagePath
    x: 1
    y: 1
  }

  Image
  {
    source: "images/gloss.png"
  }
}

Column
{
  x: 92
  width: wrapper.ListView.view.width - 95
  y: 15
  spacing: 2

  Text
  {
    text: title
    color: "white"
    width: parent.width
    font.pixelSize: 14
    font.bold: true
    elide: Text.ElideRight
    style: Text.Raised
    styleColor: "black"
  }

  Text
  {
    text: photoAuthor
    width: parent.width
    font.pixelSize: 14
    elide: Text.ElideLeft
    color: "#cccccc"
    style: Text.Raised
    styleColor: "black"
  }
```

```
                Text
                {
                  text: photoDate
                  width: parent.width
                  font.pixelSize: 14
                  elide: Text.ElideRight
                  color: "#cccccc"
                  style: Text.Raised
                  styleColor: "black"
                }
              }
            }
          }
}
```

File Progress.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    property variant progress: 0

    Rectangle
    {
      anchors.fill: parent
      smooth: true
      border.color: "white"
      border.width: 0
      radius: height / 2 - 2
      gradient: Gradient
      {
        GradientStop
        {
          position: 0
          color: "#66343434"
        }
        GradientStop
        {
          position: 1.0
          color: "#66000000"
        }
      }
    }

    Rectangle
    {
      y: 2
      height: parent.height - 4
      x: 2
      width: Math.max (parent.width * progress - 4, 0)
      opacity: width < 1 ? 0 : 1
      smooth: true
```

```
        gradient: Gradient
        {
            GradientStop
            {
                position: 0
                color: "lightsteelblue"
            }
            GradientStop
            {
                position: 1.0
                color: "steelblue"
            }
        }
        radius: height / 2 - 2
    }

    Text
    {
        text: Math.round (progress * 100) + "%"
        anchors.horizontalCenter: parent.horizontalCenter
        anchors.verticalCenter: parent.verticalCenter
        color: "white"
        font.bold: true
        font.pixelSize: 15
    }
}
```

File RssModel.qml (header with Qt$^{\text{TM}}$ license see 'main.cpp'):

```
import QtQuick 1.0

XmlListModel
{
    property string tags : ""

    function commasep (x)
    {
        return x.replace (' ',',')
    }

    source: "http://api.flickr.com/services/feeds/photos_public.gne?"+(tags ? "tags="+commasep(tags)+"&" :
                                                                                              "")+"format=rss2"
    query: "/rss/channel/item"
    namespaceDeclarations: "declare namespace media=\"http://search.yahoo.com/mrss/\";"

    XmlRole { name: "title"; query: "title/string ()" }
    XmlRole { name: "imagePath"; query: "media:thumbnail/@url/string()" }
    XmlRole { name: "url"; query: "media:content/@url/string()" }
    XmlRole { name: "description"; query: "description/string()" }
    XmlRole { name: "tags"; query: "media:category/string()" }
    XmlRole { name: "photoWidth"; query: "media:content/@width/string()" }
    XmlRole { name: "photoHeight"; query: "media:content/@height/string()" }
    XmlRole { name: "photoType"; query: "media:content/@type/string()" }
    XmlRole { name: "photoAuthor"; query: "author/string()" }
    XmlRole { name: "photoDate"; query: "pubDate/string()" }
}
```

File ScrollBar.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: container
    property variant flickableArea

    Rectangle
    {
      radius: 5
      color: "black"
      opacity: 0.3
      border.color: "white"
      border.width: 2
      x: 0
      y: flickableArea.visibleArea.yPosition * container.height
      width: parent.width
      height: flickableArea.visibleArea.heightRatio * container.height
    }

    states: [
      State
      {
        name: "show"
        when: flickableArea.movingVertically
        PropertyChanges
        {
          target: container
          opacity: 1
        }
      }
    ]

    transitions: [
      Transition
      {
        from: "*"
        to: "*"
        NumberAnimation
        {
          target: container
          properties: "opacity"
          duration: 400
        }
      }
    ]
}
```

File Slider.qml (header with Qt™ license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: slider
    width: 400
    height: 16

    // value is read / write.
    property real value: 1
    onValueChanged: updatePos ()
    property real maximum: 1
    property real minimum: 1
    property int xMax: width - handle.width - 4
    onXMaxChanged: updatePos ()
    onMinimumChanged: updatePos ()

    function updatePos ()
    {
      if (maximum > minimum)
      {
        var pos = 2 + (value - minimum) * slider.xMax / (maximum - minimum)
        pos = Math.min (pos, width - handle.width - 2)
        pos = Math.max (pos, 2)
        handle.x = pos
      }
      else
      {
        handle.x = 2
      }
    }

    Rectangle
    {
      anchors.fill: parent
      border.color: "white"
      border.width: 0
      radius: 8
      gradient: Gradient
      {
        GradientStop
        {
          position: 0.0
          color: "#66343434"
        }
        GradientStop
        {
          position: 1.0
          color: "#66000000"
        }
      }
    }
    Rectangle
    {
      id: handle
      smooth: true
      y: 2
      width: 30
      height: slider.height - 4
      radius: 6
      gradient: Gradient
      {
        GradientStop
        {
          position: 0.0
          color: "lightgray"
        }
        GradientStop
        {
          position: 1.0
          color: "gray"
        }
      }

      MouseArea
      {
        id: mouse
        anchors.fill: parent
        drag.target: parent
        drag.axis: Drag.XAxis
        drag.minimumX: 2
        drag.maximumX: slider.xMax + 2
        onPositionChanged:
        {
          value = (maximum - minimum) * (handle.x - 2) / slider.xMax + minimum
        }
      }
    }
}
```

File TitleBar.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
    id: titleBar
    property string untaggedString: "Uploads from everyone"
    property string taggedString: "Recent uploads tagged "
```

```
BorderImage
{
  source: "images/titlebar.sci"
  width: parent.width
  height: parent.height + 14
  y: -7
}
Item
{
  id: container
  width: (parent.width * 2) - 55
  height: parent.height
  function accept ()
  {
    imageDetails.closed ()
    titleBar.state = ""
    background.state = ""
    rssModel.tags = editor.text
  }
  Image
  {
    id: quitButton
    anchors.left: parent.left
    anchors.verticalCenter: parent.verticalCenter
    source: "images/quit.png"
    MouseArea
    {
      anchors.fill: parent
      onClicked: Qt.quit ()
    }
  }
  Text
  {
    id: categoryText
    anchors
    {
      left: quitButton.right
      right: tagButton.left
      leftMargin: 10
      rightMargin: 10
      verticalCenter: parent.verticalCenter
    }
    elide: Text.ElideLeft
    text: (rssModel.tags == "" ? untaggedString : taggedString + rssModel.tags)
    font.bold: true
    font.pixelSize: 15
    color: "White"
    style: Text.Raised
    styleColor: "Black"
  }
```

```
Button
{
  id: tagButton
  x: titleBar.width - 50
  width: 45
  height: 32
  text: "..."
  onClicked:
  if (titleBar.state == "Tags")
    container.accept ()
  else
    titleBar.state = "Tags"
  anchors.verticalCenter: parent.verticalCenter
}

Item
{
  id: lineEdit
  y: 4
  height: parent.height - 9
  anchors
  {
    left: tagButton.right
    leftMargin: 5
    right: parent.right
    rightMargin: 5
  }

  BorderImage
  {
    source: "images/lineedit.sci"
    anchors.fill: parent
  }

  TextInput
  {
    id: editor
    anchors
    {
      left: parent.left
      right: parent.right
      leftMargin: 10
      rightMargin: 10
      verticalCenter: parent.verticalCenter
    }
    cursorVisible: true
    font.bold: true
    color: "#151515"
    selectionColor: "Green"
  }

  Keys.forwardTo: [(returnKey), (editor)]
```

```
            Item
            {
               id: returnKey
               Keys.onReturnPressed: container.accept ()
               Keys.onEnterPressed: container.accept ()
               Keys.onEscapePressed: titleBar.state = ""
            }
         }
      }

      states: State
      {
         name: "Tags"

         PropertyChanges
         {
            target: container
            x: -tagButton.x + 5
         }

         PropertyChanges
         {
            target: tagButton
            text: "OK"
         }

         PropertyChanges
         {
            target: editor
            focus: true
         }
      }

      transitions: Transition
      {
         NumberAnimation
         {
            properties: "x"
            easing.type: Easing.InOutQuad
         }
      }
}
```

File ToolBar.qml (header with Qt^{TM} license see 'main.cpp'):

```
import QtQuick 1.0

Item
{
      id: toolbar
      property alias button1Label: button1.text
      property alias button2Label: button2.text
      property alias button2Visible: button2.visible

      signal button1Clicked
      signal button2Clicked
```

```
    BorderImage
    {
      source: "images/titlebar.sci"
      width: parent.width
      height: parent.height + 14
      y: -7
    }

    Row
    {
      anchors.right: parent.right
      anchors.rightMargin: 5
      y: 3
      height: 32
      spacing: 30

      Button
      {
        id: button1
        width: 140
        height: 32
        onClicked: toolbar.button1Clicked ()
      }

      Button
      {
        id: button2
        width: 140
        height: 32
        onClicked: toolbar.button2Clicked ()
      }
    }
}
```

10.13 Interprocess Communication (Root/User)

Many examples in this guide require root access. If you want to retain the user permissions the easiest way is based on InterProcess Communication (IPC) between a root (server) and a user (client) program. For background operation the server part can be realized as a Daemon. A number of possible communication types exist: shared memory, mapped memory, pipes, FIFOs (named pipes) and sockets. The following demonstration programs only use FIFOs. Any process has the ability to open or close the FIFO; the processes on either end of the pipe need not be related to each other. A FIFO can possess multiple readers or multiple writers.
As a first step the sample programs run as normal applications in a terminal window, one as root (server) and the other one as user (client).

One possible Makefile (server) might look like this:

```
CC := arm-linux-gnueabihf-gcc -march=armv7

all: server

server.o: server.c

server: server.o
	$(CC) -o $@ server.o

clean:
	rm server server.o
```

One possible Makefile (client) might look like this:

```
CC := arm-linux-gnueabihf-gcc -march=armv7

all: client

client.o: client.c

client: client.o
	$(CC) -o $@ client.o

clean:
	rm client client.o
```

and now the Include-file 'fifo.h':

```
/* Include file for server & client application

 * Copyright (c) 2014  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License. */

#ifndef FIFO_H
#define FIFO_H

#define RD_SIZE                 32
#define WR_SIZE                 32

#define POLL_TIME               5                  /* milliseconds */
```

```
#define CMD_SIZE                4
#define CMD_WDIO                "wdio"
#define CMD_GPIO                "gpio"
#define CMD_EXIT                "exit"

#define ACK_SIZE                3
#define ACK_STR                 "ok\n"
#define ACK_APP                 "    :"
#define NAK_SIZE                4
#define NAK_STR                 "nok\n"
#define DATA_STR                "ok - input value = %d\n"
#define CMD_STR                 " cmd>"
#define TIMEOUT_STR             "    :timeout !\n"

#define WD_ON                   "on"
#define WD_OFF                  "off"
#define WD_TRIG                 "trig"

#define IO_OPEN                 "open"
#define IO_CLOSE                "close"
#define IO_DATA                 "data"
#define IO_IN                   "in"
#define IO_OUT                  "out"
#define IO_LOW                  "0"
#define IO_HIGH                 "1"
#define IO_DIR_IN               0
#define IO_DIR_OUT              1

#define ERROR do { fprintf (stderr, "Error: %s\n", strerror (errno)); exit (EXIT_FAILURE); } while(0)

static const char *root_rd          = "/tmp/root_rd";
static const char *root_wr          = "/tmp/root_wr";

static const char *wdt              = "/dev/watchdog";
static const char *io_exp           = "/sys/class/gpio/export";
static const char *io_unexp         = "/sys/class/gpio/unexport";
static const char *io_dir           = "/sys/class/gpio/gpio%d/direction";
static const char *io_val           = "/sys/class/gpio/gpio%d/value";

static const int io_num [] = {
        10, 11, 12, 13, 14, 15, 16, 17,
        18, 19, 20, 21, 22, 23, 24, 25,
        26, 27, 28, 66, 80, 81, 82, 83,
        84, 41, 43, 44, 59, 60, 61, 62,
        64, 66, 71, 72, 73, 74, 75, 76 };

#endif
```

following the sourcecode of the server application 'server.c':

```
/* Server application

 * Copyright (c) 2014  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License. */
```

```
#include <fcntl.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <linux/limits.h>
#include <linux/watchdog.h>
#include "fifo.h"

int parse_wdio (char cmd_line[])
{
    static int fd_wd = 0;
    const char delim[] = " ,;:-";
    char *param1, *param2;

    cmd_line [strlen (cmd_line) - 1] = '\0';                /* remove CR */
    strsep (&cmd_line, delim);                              /* suppress the command */
    param1 = strsep (&cmd_line, delim);
    param2 = strsep (&cmd_line, delim);

    if (param1 == (char *) NULL)
      return -1;

    if (!strcasecmp (param1, WD_ON) && param2 != (char *) NULL && !fd_wd)
    {
      int time;

      time = atoi (param2);
      if ((time < 1) || (time > 600))
        return -1;
      if ((fd_wd = open (wdt, O_WRONLY)) < 0)
        return -1;
      if (ioctl (fd_wd, WDIOC_SETTIMEOUT, &time))
      {
        close (fd_wd);
        fd_wd = 0;
        return -1;
      }
      return 0;
    }

    if (!strcasecmp (param1, WD_OFF) && fd_wd)
    {
      if (write (fd_wd, "V", 1) != 1)
      {
        close (fd_wd);                                      /* matter of opinion */
        fd_wd = 0;                                          /* the same */
        return -1;
      }
      close (fd_wd);
      fd_wd = 0;
      return 0;
    }
    if (!strcasecmp (param1, WD_TRIG) && fd_wd)
    {
      if (write (fd_wd, "\0", 1) != 1)
        return -1;                                   /* fd_wd is not closed to give a second chance */
      return 0;
    }

    return -1;
}


int parse_gpio (char cmd_line[], int *val)
{
    static int fd_io = 0;
    static int gp_num = 0;
    static int gp_dir = -1;
    const char delim[] = " ,;:-";
    char *param1, *param2, *param3;

    cmd_line [strlen (cmd_line) - 1] = '\0';         /* remove CR */
    strsep (&cmd_line, delim);                       /* suppress the command */
    param1 = strsep (&cmd_line, delim);
    param2 = strsep (&cmd_line, delim);
    param3 = strsep (&cmd_line, delim);

    *val = -1;
    if ((param1 == (char *) NULL) || (param2 == (char *) NULL))
      return -1;

    if (!strcasecmp (param2, IO_OPEN) && param3 != (char *) NULL && !fd_io)
    {
      int fd_tmp, gpio;
      char str [64];

      gpio = atoi (param1);
      if ((gpio < 0) || (gpio > 39))
        return -1;
      if (strcasecmp (param3, IO_IN) && strcasecmp (param3, IO_OUT))
        return -1;

      if ((fd_tmp = open (io_exp, O_WRONLY)) < 0)
        return -1;
      sprintf (str, "%d", io_num[gpio]);
      if (write (fd_tmp, str, strlen (str)) != strlen (str))
      {
        close (fd_tmp);
        return -1;
      }
      close (fd_tmp);

      sprintf (str, io_dir, io_num[gpio]);
      if ((fd_tmp = open (str, O_WRONLY)) < 0)
        return -1;
      if (write (fd_tmp, param3, strlen (param3)) != strlen (param3))
      {
        close (fd_tmp);
        return -1;
      }
      close (fd_tmp);

      sprintf (str, io_val, io_num[gpio]);
      if (!strcasecmp (param3, IO_IN))
      {
        if ((fd_io = open (str, O_RDONLY)) < 0)
          return -1;
        gp_dir = IO_DIR_IN;
      }
      else
      {
        if ((fd_io = open (str, O_WRONLY)) < 0)
          return -1;
        gp_dir = IO_DIR_OUT;
      }
      gp_num = gpio;
      return 0;
    }

    if (!strcasecmp (param2, IO_CLOSE) && fd_io)
    {
      int fd_tmp, gpio;
      char str [8];

      gpio = atoi (param1);
      if (gpio != gp_num)
        return -1;

      close (fd_io);
      fd_io = 0;

      if ((fd_tmp = open (io_unexp, O_WRONLY)) < 0)
        return -1;
      sprintf (str, "%d", io_num[gpio]);
      if (write (fd_tmp, str, strlen (str)) != strlen (str))
      {
        close (fd_tmp);
        return -1;
      }
      close (fd_tmp);
      return 0;
    }

    if (!strcasecmp (param2, IO_DATA) && fd_io)
    {
      int gpio;
      char str [8];

      gpio = atoi (param1);
      if (gpio != gp_num)
        return -1;
```

```
        if (!gp_dir)
        {
            if (read (fd_io, str, sizeof (str)) < 0)
                return -1;
            str [1] = '\0';
            *val = atoi (str);
            return lseek (fd_io, 0, SEEK_SET);
        }
        else
        {
            if (param3 != (char *) NULL)
            {
                if (strcasecmp (param3, IO_LOW) && strcasecmp (param3, IO_HIGH))
                    return -1;
            }
            if (write (fd_io, param3, strlen (param3)) != strlen (param3))
                return -1;
            return lseek (fd_io, 0, SEEK_SET);
        }
    }

    return -1;
}


int main (void)
{
    int fifo_rd, fifo_wr, value;
    mode_t oldmask = umask (0);
    static char buff_rd [RD_SIZE];
    static char buff_wr [WR_SIZE];

    if (mkfifo (root_rd, S_IWUSR | S_IRUSR | S_IWGRP | S_IRGRP | S_IWOTH | S_IROTH) < 0)
    {
        umask (oldmask);
        ERROR;
    }

    if (mkfifo (root_wr, S_IWUSR | S_IRUSR | S_IWGRP | S_IRGRP | S_IWOTH | S_IROTH) < 0)
    {
        unlink (root_rd);
        umask (oldmask);
        ERROR;
    }

    if ((fifo_rd = open (root_rd, O_RDONLY)) < 0 || (fifo_wr = open (root_wr, O_WRONLY)) < 0)
    {
        unlink (root_rd);
        unlink (root_wr);
        umask (oldmask);
        ERROR;
    }
```

```
   while (read (fifo_rd, buff_rd, RD_SIZE) > 0)
   {
      if (!strncasecmp (buff_rd, CMD_WDIO, CMD_SIZE))
      {
         if (!parse_wdio (buff_rd))
            write (fifo_wr, ACK_STR, ACK_SIZE);
         else
            write (fifo_wr, NAK_STR, NAK_SIZE);
      }

      if (!strncasecmp (buff_rd, CMD_GPIO, CMD_SIZE))
      {
         if (!parse_gpio (buff_rd, &value))
         {
            if (value != -1)
            {
               sprintf (buff_wr, DATA_STR, value);
               write (fifo_wr, buff_wr, strlen (buff_wr));
            }
            else
               write (fifo_wr, ACK_STR, ACK_SIZE);
         }
         else
            write (fifo_wr, NAK_STR, NAK_SIZE);
      }

      if (!strncasecmp (buff_rd, CMD_EXIT, CMD_SIZE))
         break;

      memset (buff_rd, 0, RD_SIZE);
   }

   close (fifo_rd);
   unlink (root_rd);
   close (fifo_wr);
   unlink (root_wr);
   umask (oldmask);
   return 0;
}
```

and finally the sourcecode of the client application 'client.c':

```
/* Client application

 * Copyright (c) 2014  Kontron Technology A/S

 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License. */

#include <fcntl.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <poll.h>
```

```
#include <linux/limits.h>
#include "fifo.h"

int main (void)
{
    int fifo_rd, fifo_wr, n;
    struct pollfd pfd;
    static char buff_rd [RD_SIZE];
    static char buff_wr [WR_SIZE];

    if ((fifo_rd = open (root_wr, O_RDONLY | O_NDELAY)) < 0 || (fifo_wr = open (root_rd, O_WRONLY)) < 0)
      ERROR;

    while (1)
    {
      write (fileno (stdout), CMD_STR, strlen (CMD_STR));
      memset (buff_wr, 0, WR_SIZE);
      memset (buff_rd, 0, RD_SIZE);
      n = read (fileno (stdin), buff_wr, WR_SIZE);
      write (fifo_wr, buff_wr, n);

      if (! strncasecmp (buff_wr, CMD_EXIT, CMD_SIZE))
        break;

      pfd.fd = fifo_rd;
      pfd.events = POLLIN;
      if (poll (&pfd, 1, POLL_TIME) > 0)                /* wait 5 milliseconds */
      {
        write (fileno (stderr), ACK_APP, strlen (ACK_APP));
        while ((n = read (fifo_rd, buff_rd, RD_SIZE)) > 0)
          write (fileno (stderr), buff_rd, n);
      }
      else
        write (fileno (stderr), TIMEOUT_STR, strlen (TIMEOUT_STR));
    }

    close (fifo_rd);
    close (fifo_wr);
    return 0;
}
```

First open a terminal window, type su (password root) or sudo -i (password ktam3874), change to your target directory with the server application and start the server program with './server'. Thereafter open a second new terminal window, go to your target directory with the client application and run this program with './client' (not as root).

Both samples are based on practical experience and aimed at simple and quick implementation of inter-process communication, therefore only watchdog and GPIO functionality are implemented.

Command table:

wdio on 30	Activate the watchdog with timeout of 30 seconds
wdio off	Disable the watchdog
wdio trig	Trigger the watchdog

gpio 11 open in	Enable (export) GPIO11 as input
gpio 11 open out	Enable (export) GPIO11 as output
gpio 11 close	Disable (unexport) GPIO11
gpio 11 data	Read GPIO11 data line (should be opened as input)
gpio 11 data 0	Set GPIO11 data line to low level (should be opened as output)
gpio 11 data 1	Set GPIO11 data line to high level (should be opened as output)
exit	Terminate client and server program

Remark: the GPIO assignment (0 to 39) uses the table on page 61.

If you perform a more detailled analysis you can notice that the programs apply two separate FIFOs, one for sending and one for receiving, and not a bidirectional FIFO. For speed optimization it is possible to reduce the poll time (client application) and/or use binary data instead of ASCII chars.

To run the server application as a normal program does not offer the best solution. The following source-code realizes a Daemon for background operation. It is a good habit to rename the server program to 'serverd'. The listing only shows the main routine and two additional subroutines (extensions in green color):

```
#include <errno.h>
#include <signal.h>
#include <linux/limits.h>


mode_t get_mask (int cmd_set)
{
    static mode_t oldmask;

    if (cmd_set)
      oldmask = umask (0);
    return oldmask;
}


void signal_handler (int sig)
{
    switch (sig)
    {
      case SIGHUP: break;            /* ignore this signal or do something */

      case SIGTERM:
          unlink (root_rd);
          unlink (root_wr);
          umask (get_mask (0));
          exit (EXIT_SUCCESS);
    }
}
```

```
int main (void)
{
    int fifo_rd, fifo_wr, value;
    pid_t pid;
    static char buff_rd [RD_SIZE];
    static char buff_wr [WR_SIZE];

    /* Fork off the parent process */
    if ((pid = fork ()) < 0)
      ERROR;

    /* If we got a good PID then we can exit the parent process */
    if (pid > 0)
      exit (EXIT_SUCCESS);

    /* process becomes session leader, group leader and has no controlling shell */
    if (setsid () < 0)
      ERROR;

    signal (SIGHUP, signal_handler);
    signal (SIGTERM, signal_handler);
    get_mask (1);


    if (mkfifo (root_rd, S_IWUSR | S_IRUSR | S_IWGRP | S_IRGRP | S_IWOTH | S_IROTH) < 0)
    {
      umask (get_mask (0));
      ERROR;
    }

    if (mkfifo (root_wr, S_IWUSR | S_IRUSR | S_IWGRP | S_IRGRP | S_IWOTH | S_IROTH) < 0)
    {
      unlink (root_rd);
      umask (get_mask (0));
      ERROR;
    }

    oldmask = umask (0);

    if ((fifo_rd = open (root_rd, O_RDONLY)) < 0 || (fifo_wr = open (root_wr, O_WRONLY)) < 0)
    {
      unlink (root_rd);
      unlink (root_wr);
      umask (get_mask (0));
      ERROR;
    }

    while (read (fifo_rd, buff_rd, RD_SIZE) > 0)
    {
      if (!strncasecmp (buff_rd, CMD_WDIO, CMD_SIZE))
      {
        if (!parse_wdio (buff_rd))
          write (fifo_wr, ACK_STR, ACK_SIZE);
        else
          write (fifo_wr, NAK_STR, NAK_SIZE);
      }
```

```
        if (!strncasecmp (buff_rd, CMD_GPIO, CMD_SIZE))
        {
          if (!parse_gpio (buff_rd, &value))
          {
            if (value != -1)
            {
              sprintf (buff_wr, DATA_STR, value);
              write (fifo_wr, buff_wr, strlen (buff_wr));
            }
            else
              write (fifo_wr, ACK_STR, ACK_SIZE);
          }
          else
            write (fifo_wr, NAK_STR, NAK_SIZE);
        }

        if (!strncasecmp (buff_rd, CMD_EXIT, CMD_SIZE))
          break;

        memset (buff_rd, 0, RD_SIZE);
      }

      close (fifo_rd);
      unlink (root_rd);
      close (fifo_wr);
      unlink (root_wr);
      umask (get_mask (0));
      return 0;
}
```

Remark: it will probably be necessary to change the file permissions of the daemon program and the init script below with chmod.

Some subtleties, for example the closing or redirection (/dev/null) of the standard I/O devices stdin, stdout and stderr, are not really important and a brief explanation of pidfile usage follows later.

A good idea for the target path of the daemon would be the directory /usr/sbin/. Open a terminal window and type

```
sudo start-stop-daemon --start --exec /usr/sbin/serverd
```

and then close the terminal window (ignore the 'process kill warning').

Before you close the window you can check the activity of the daemon with

```
pgrep -lf serverd
```

For terminating the daemon open again a terminal window and input the line

```
sudo start-stop-daemon --stop --exec /usr/sbin/serverd
```

or as an equivalent solution use the 'exit' command from the client program (see page 266/267 command table).

What do you have to do in order to integrate the daemon into the boot process? The first step consists in the preparation of an init script named 'serv', for example:

```
#! /bin/sh

### BEGIN INIT INFO
# Provides:          serverd
# Required-Start:    $remote_fs
# Required-Stop:     $remote_fs
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: Provide routines for root access
# Description:       Provide routines for root access
### END INIT INFO

DAEMON=/usr/sbin/serverd

case "$1" in
   start)
     start-stop-daemon --start --quiet --exec $DAEMON
     ;;
   stop)
     start-stop-daemon --stop --quiet --exec $DAEMON
     ;;
   restart|force-reload)
     $0 stop
     $0 start
     ;;
   status)
     ;;
   *)
     echo "Usage: $N {start|stop|restart|force-reload|status}" >&2
     exit 1
     ;;
esac

exit 0
```

and then copy this script into the directory /etc/init.d. You can start or stop the script manually with

sudo ./serv start respectively **sudo ./serv stop**

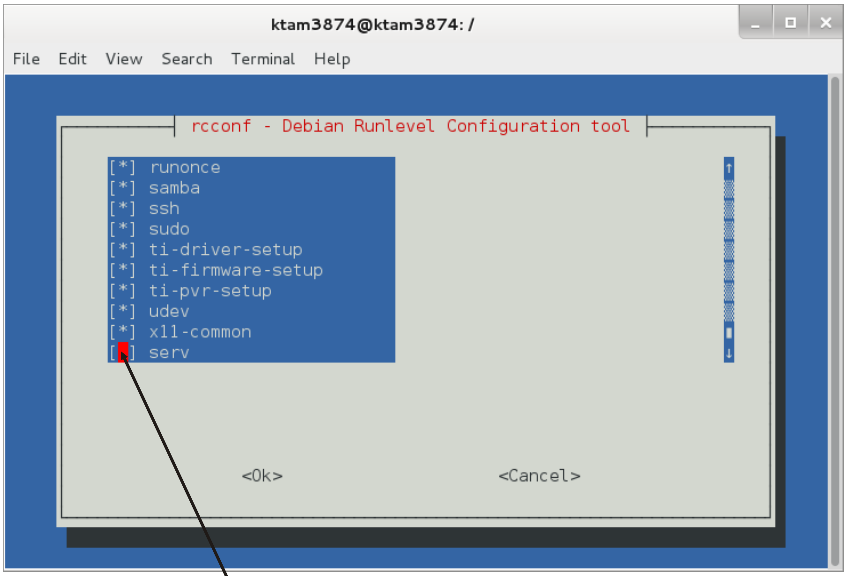
The use of the rcconf tool represents a very simple way to integrate the daemon into the boot process. If you need more setting options the program update-rc.d is more suitable. By adopting this approach only the client application has to be started manually. The rcconf tool can be obtained by

sudo apt-get install rcconf

After the input of

```
sudo rcconf
```

you see the following screen



```
rcconf - Debian Runlevel Configuration tool

[*] runonce
[*] samba
[*] ssh
[*] sudo
[*] ti-driver-setup
[*] ti-firmware-setup
[*] ti-pvr-setup
[*] udev
[*] x11-common
[ ] serv
```

press Space and then Enter

The next item concerns the pidfile topic. The presence of a locked pidfile prevents two instances of a daemon with the same name from running at the same time. The pidfile contains a single line of text, being the numeric process ID of the process that currently holds the lock. The following suggestion for the server application:

```
static int pidfile;

int create_pidfile (void)
{
    char str [32];

    if ((pidfile = open ("/var/run/serverd.pid", O_RDWR | O_CREAT)) < 0)
       return -1;

    /* Try to lock the file and test the presence of an existing pidfile */
    if (lockf (pidfile, F_TLOCK, 0) < 0)
    {
       close (pidfile);
       return -1;
    }

    sprintf (str, "%d\n", getpid ());
    if (write (pidfile, str, strlen (str)) < 0)
    {
       close (pidfile);
       return -1;
    }

    return 0;
}
```

```
int main (void)
{
             .
             .
     if (create_pidfile ())
        ERROR;
             .
```

There is evidence that the create_pidfile() call should be made before you fork off the parent process. Naturally you have to close the file descriptor 'pidfile' at the end of the program and in the signal handler (SIGTERM). If you use the 'exit' command you should also delete the physical pidfile in the server application otherwise you can add an 'rm' command in the stop-path of the init script.

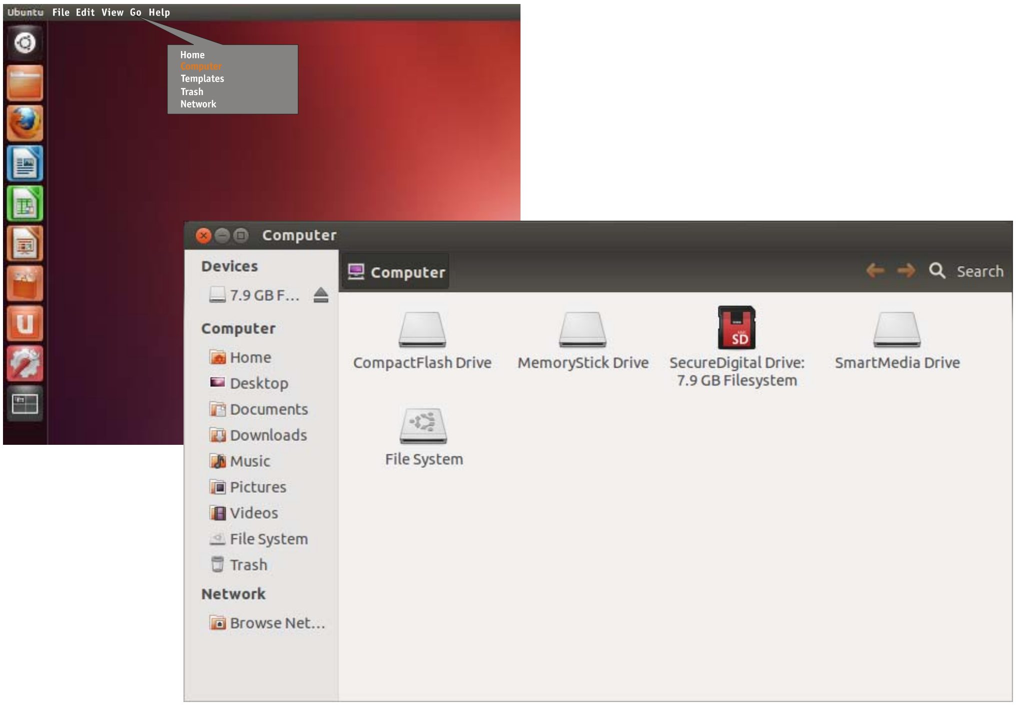
Appendix A: Linux® on microSD™ Card

A.1 Ubuntu™ 12.04 LTS

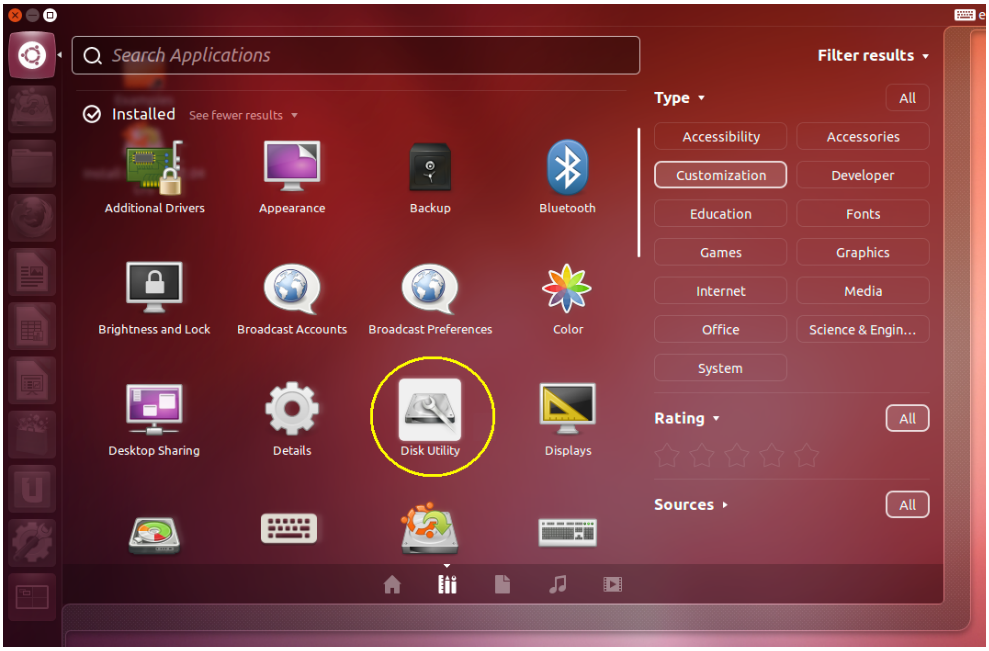
The description below is based on:

Computer: x86 Desktop PC
Storage medium: Live-CD
microSD™ card carrier: USB Card Reader

As a first step you should check the presence of the microSD™ card.



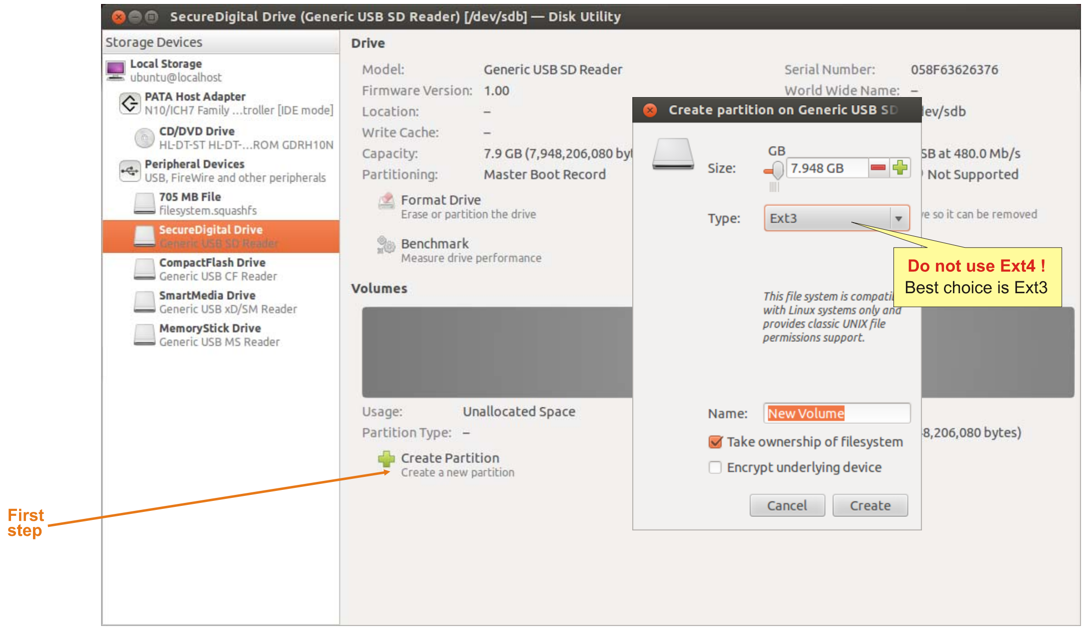
Thereafter you need the Disk Utility for partitioning and formatting.



The easiest way to prepare the microSD^{TM} card is to use one partition. The user space really needs an Ext-format; therefore only Ext2 to Ext4 are possible.

ATTENTION

Do not use the Ext4 format because U-Boot cannot boot the Kernel from an Ext4 partition.


SecureDigital Drive (Generic USB SD Reader) [/dev/sdb] — Disk Utility

Storage Devices: Local Storage (ubuntu@localhost); PATA Host Adapter (N10/ICH7 Family ...troller [IDE mode]); CD/DVD Drive (HL-DT-ST HL-DT-...ROM GDRH10N); Peripheral Devices (USB, FireWire and other peripherals); 705 MB File (filesystem.squashfs); SecureDigital Drive (Generic USB SD Reader); CompactFlash Drive (Generic USB CF Reader); SmartMedia Drive (Generic USB xD/SM Reader); MemoryStick Drive (Generic USB MS Reader)

Drive

Model: Generic USB SD Reader
Firmware Version: 1.00
Location: –
Write Cache: –
Capacity: 7.9 GB (7,948,206,080 byt
Partitioning: Master Boot Record
Serial Number: 058F63626376
World Wide Name: –

Volumes

Usage: Unallocated Space
Partition Type: –

Create partition on Generic USB SD

Size: 7.948 GB
Type: Ext3

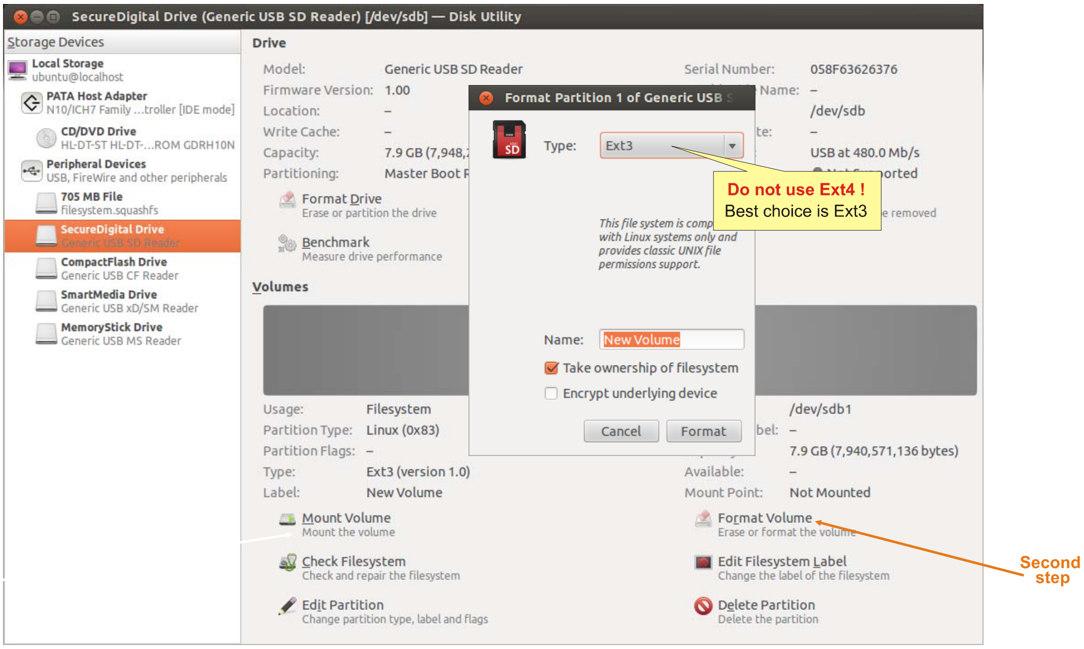
This file system is compati... with Linux systems only and provides classic UNIX file permissions support.

Name: New Volume
☑ Take ownership of filesystem
☐ Encrypt underlying device

Do not use Ext4 !
Best choice is Ext3

First step

After partitioning the microSD^{TM} card requires formatting.


SecureDigital Drive (Generic USB SD Reader) [/dev/sdb] — Disk Utility

Drive

Model: Generic USB SD Reader
Firmware Version: 1.00
Location: –
Write Cache: –
Capacity: 7.9 GB (7,948,2
Partitioning: Master Boot R
Serial Number: 058F63626376

Format Partition 1 of Generic USB S

Type: Ext3

This file system is comp... with Linux systems only and provides classic UNIX file permissions support.

Name: New Volume
☑ Take ownership of filesystem
☐ Encrypt underlying device

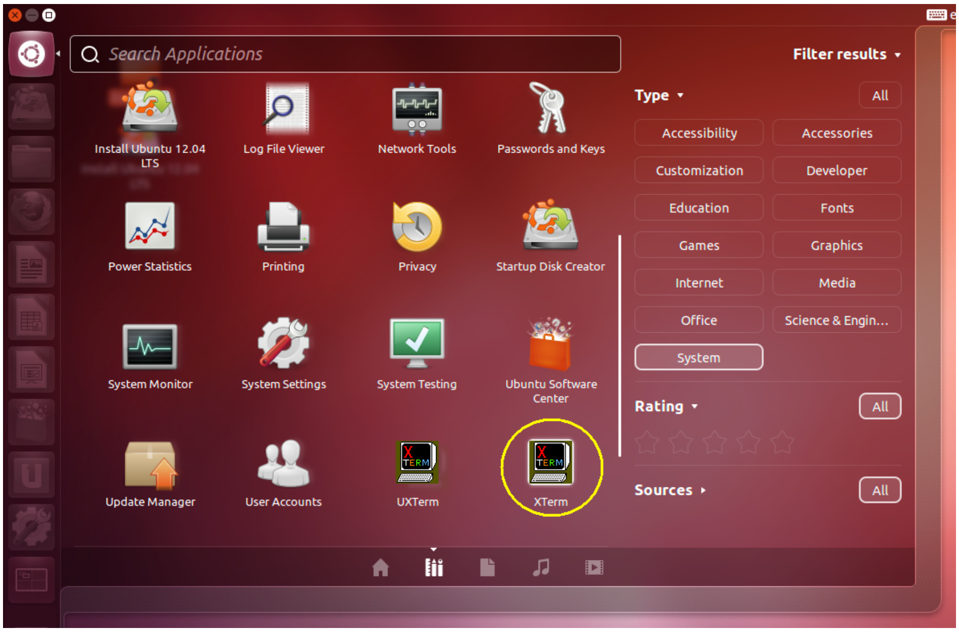
Do not use Ext4 !
Best choice is Ext3

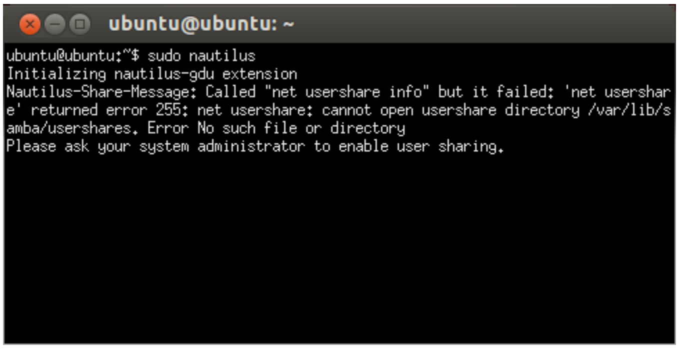
Volumes

Usage: Filesystem
Partition Type: Linux (0x83)
Partition Flags: –
Type: Ext3 (version 1.0)
Label: New Volume
/dev/sdb1
7.9 GB (7,940,571,136 bytes)
Available: –
Mount Point: Not Mounted

Second step

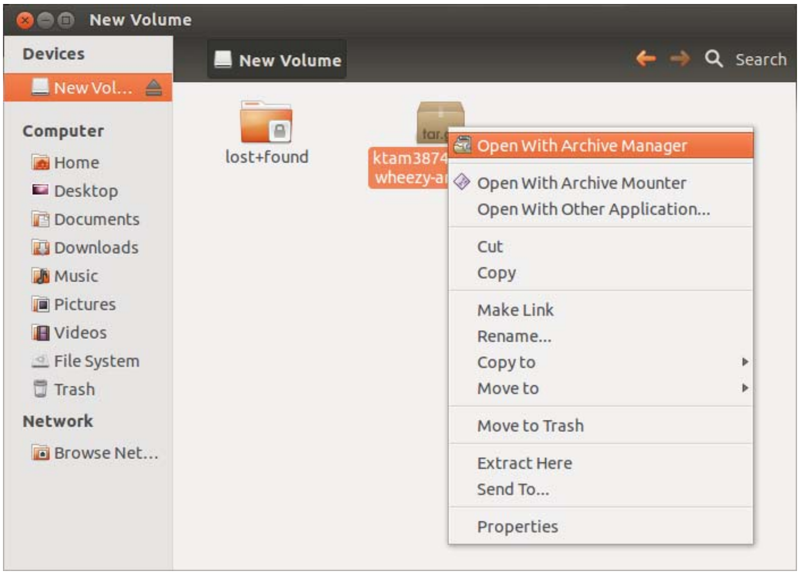
The next step involves the copying of the tarball ktam3874-gnome-wheezy-armhf.tgz (Debian™ version) from a USB key (for example formatted with FAT32) to the microSD™ card with 'Go/Computer' (see first step). The standard file manager allows the extraction of the tgz-file but you need root rights. The only way to do this leads trough the terminal program Terminal or XTerm (ignore the error messages).

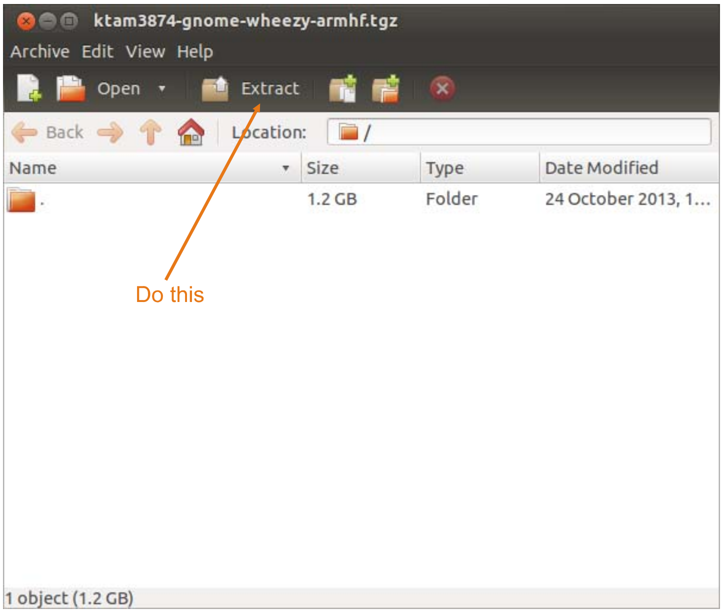




```
ubuntu@ubuntu:~$ sudo nautilus
Initializing nautilus-gdu extension
Nautilus-Share-Message: Called "net usershare info" but it failed: 'net usershar
e' returned error 255: net usershare: cannot open usershare directory /var/lib/s
amba/usershares. Error No such file or directory
Please ask your system administrator to enable user sharing.
```

The next pictures illustrate the approach.



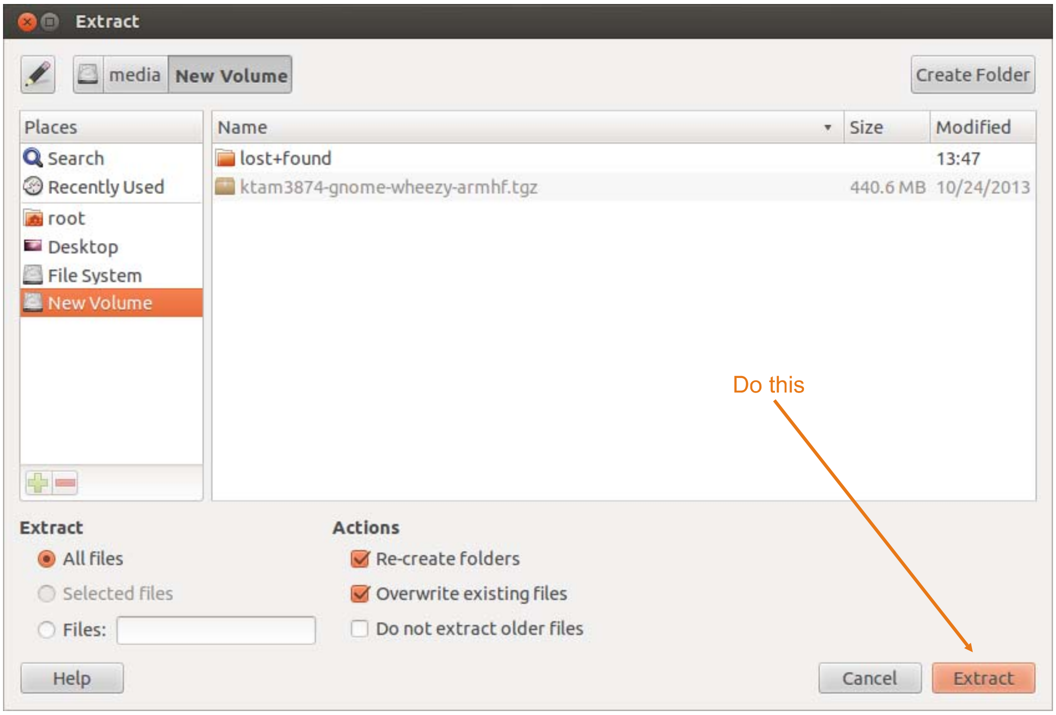


Location: /

Name	Size	Type	Date Modified
.	1.2 GB	Folder	24 October 2013, 1…

1 object (1.2 GB)

Do this



Name	Size	Modified
lost+found		13:47
ktam3874-gnome-wheezy-armhf.tgz	440.6 MB	10/24/2013

Extract

- ● All files
- ○ Selected files
- ○ Files:

Actions

- ☑ Re-create folders
- ☑ Overwrite existing files
- □ Do not extract older files

Do this

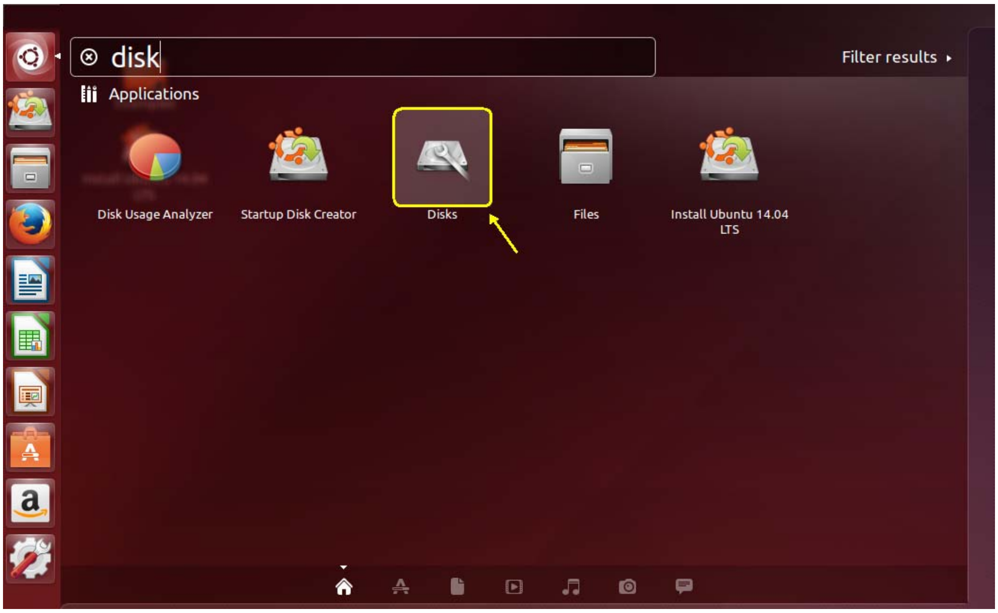
Thereafter the microSD^{TM} card is ready for use.

A.2 Ubuntu^{TM} 14.04 LTS

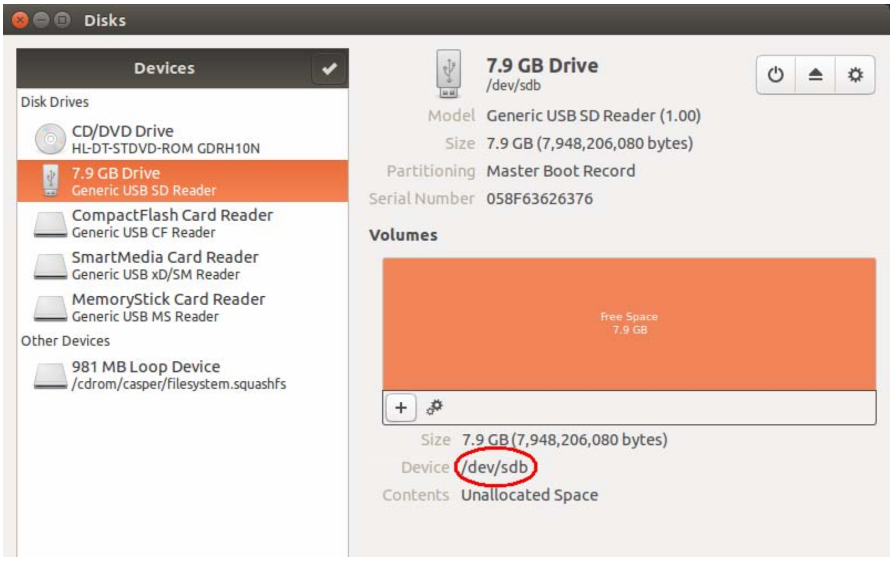
The description below is based on:

Computer: x86 Desktop PC
Storage medium: Live-CD
microSD^{TM} card carrier: USB Card Reader

As a first step you need the Disk Utility to obtain some information (type 'disk').



It is appropriate to delete the existing partition(s) and create a new single partition respectively a new formatting. The red bordered field below designates the drive identification (for example: /dev/sdb).

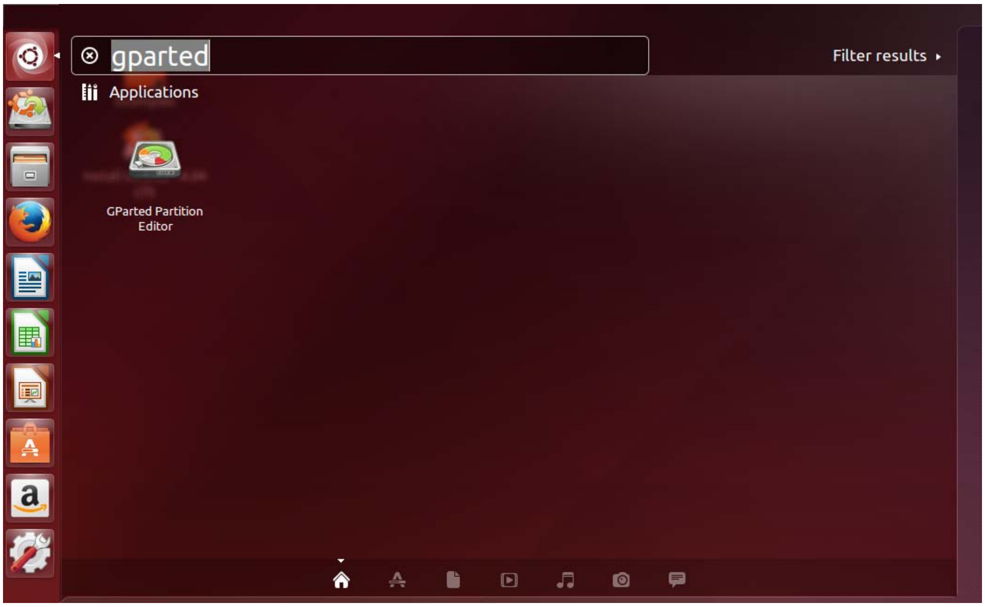


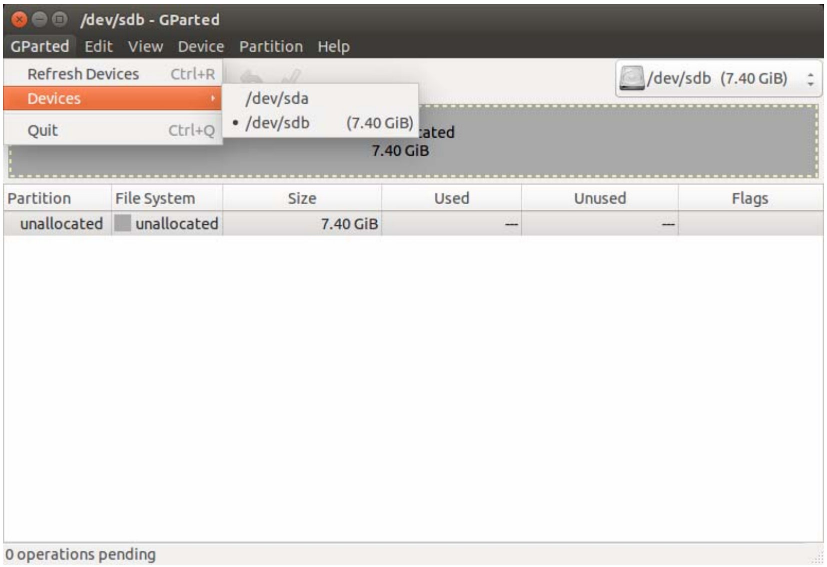
7.9 GB Drive
/dev/sdb

Model Generic USB SD Reader (1.00)
Size 7.9 GB (7,948,206,080 bytes)
Partitioning Master Boot Record
Serial Number 058F63626376

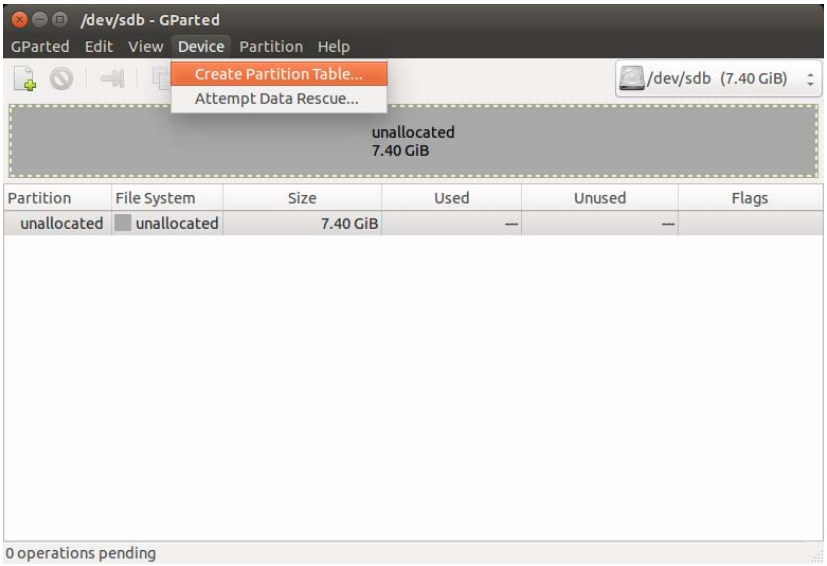
Size 7.9 GB (7,948,206,080 bytes)
Device /dev/sdb
Contents Unallocated Space

As a next step use the Partition Editor (type 'gparted'). Most of the following pictures are self-explanatory.

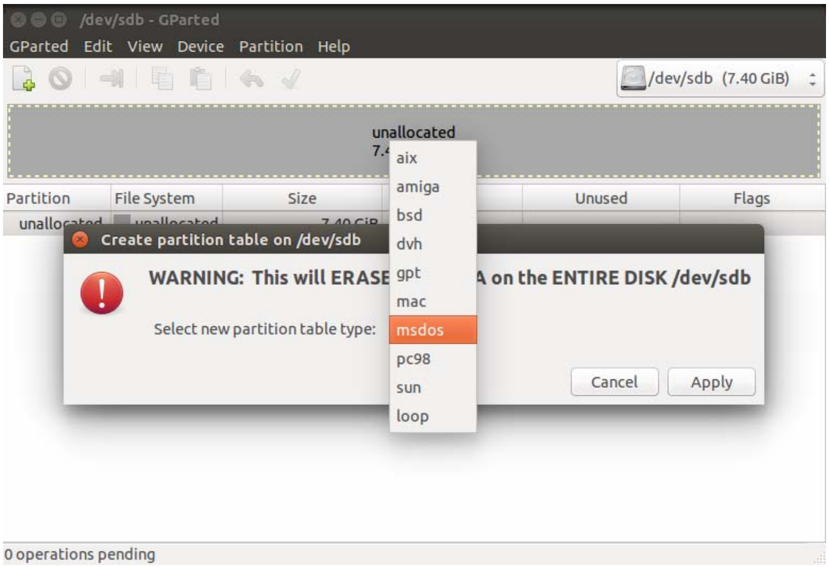




Partition	File System	Size	Used	Unused	Flags
unallocated	unallocated	7.40 GiB	---	---	



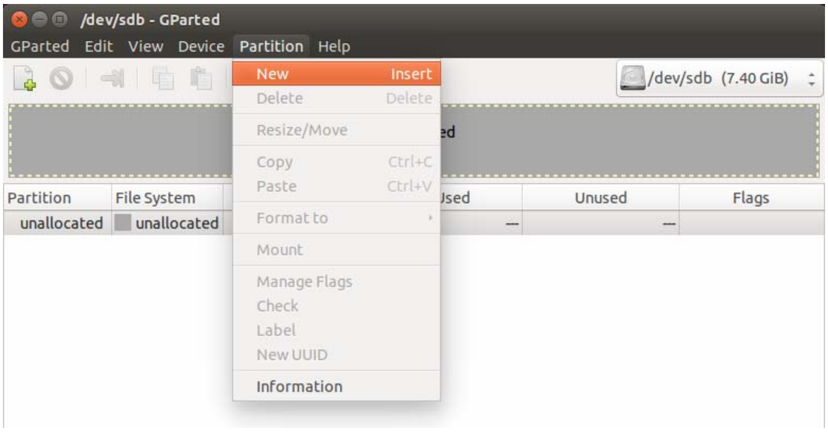
Partition	File System	Size	Used	Unused	Flags
unallocated	unallocated	7.40 GiB	---	---	

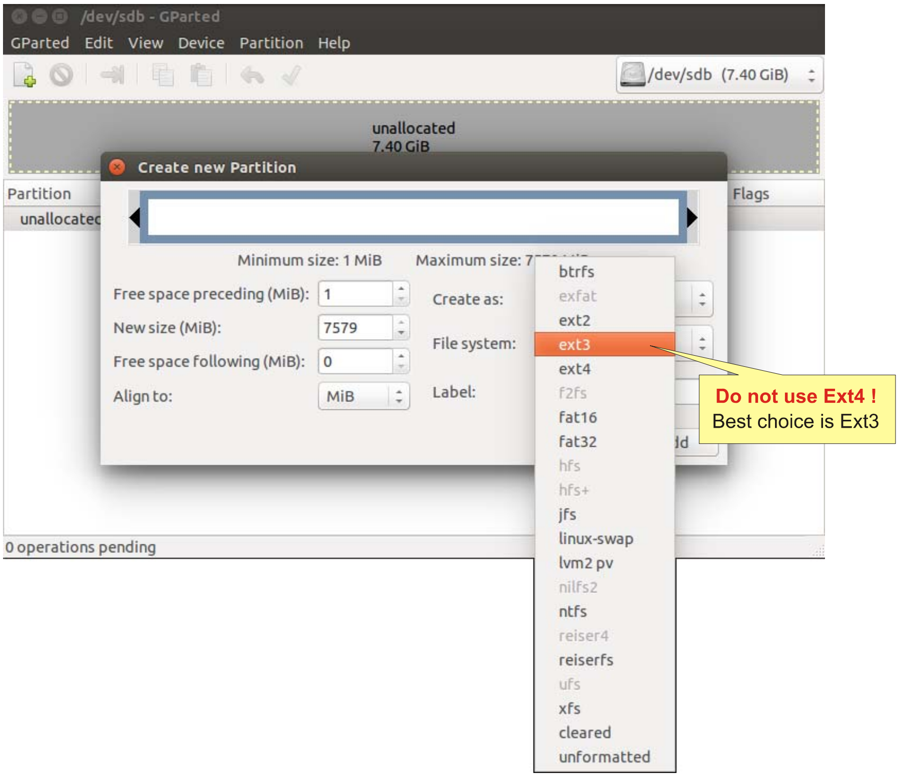


Create partition table on /dev/sdb

WARNING: This will ERASE [...] A on the ENTIRE DISK /dev/sdb

Select new partition table type: msdos





Create new Partition

Minimum size: 1 MiB Maximum size: 7[illegible] MiB

Free space preceding (MiB): 1

New size (MiB): 7579

Free space following (MiB): 0

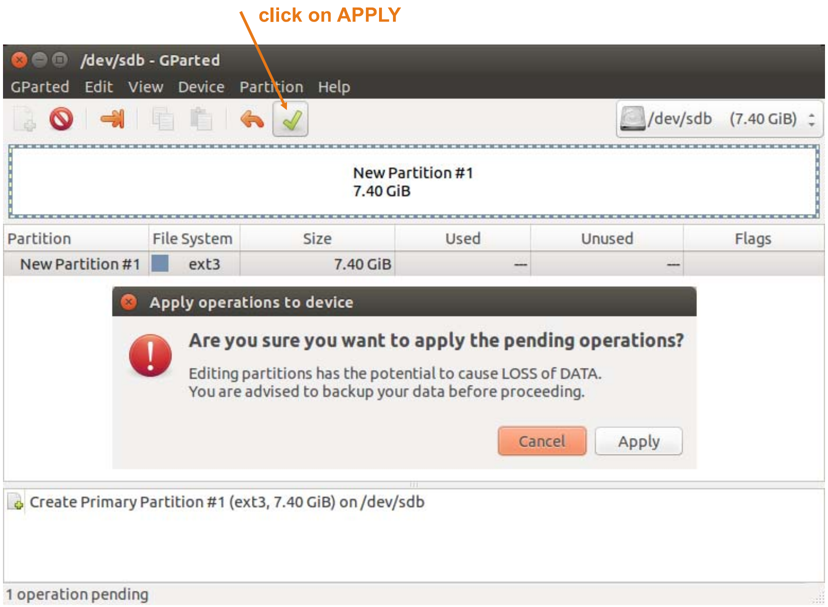
Align to: MiB

Create as:

File system: ext3

Label:

Do not use Ext4 !
Best choice is Ext3



click on APPLY

Partition	File System	Size	Used	Unused	Flags
New Partition #1	▨ ext3	7.40 GiB	---	---	

Apply operations to device

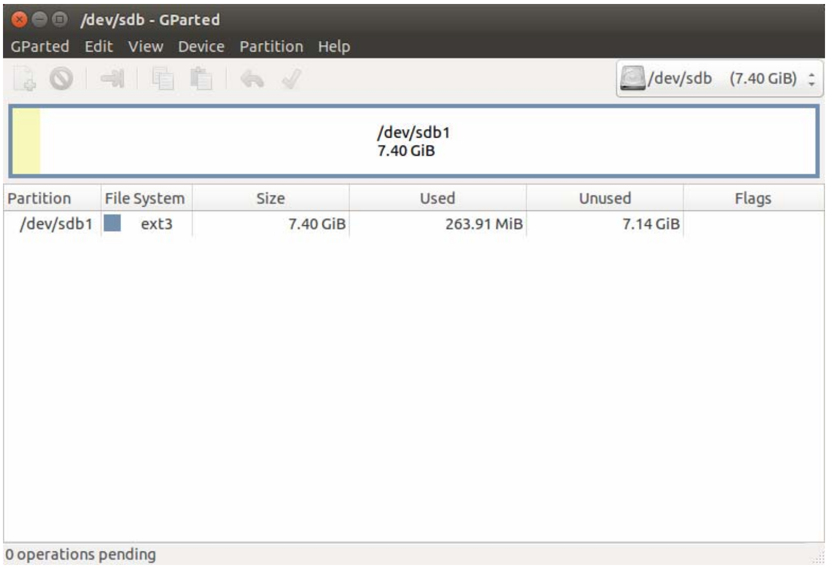
Are you sure you want to apply the pending operations?

Editing partitions has the potential to cause LOSS of DATA.
You are advised to backup your data before proceeding.

Create Primary Partition #1 (ext3, 7.40 GiB) on /dev/sdb

1 operation pending

and finally you should see

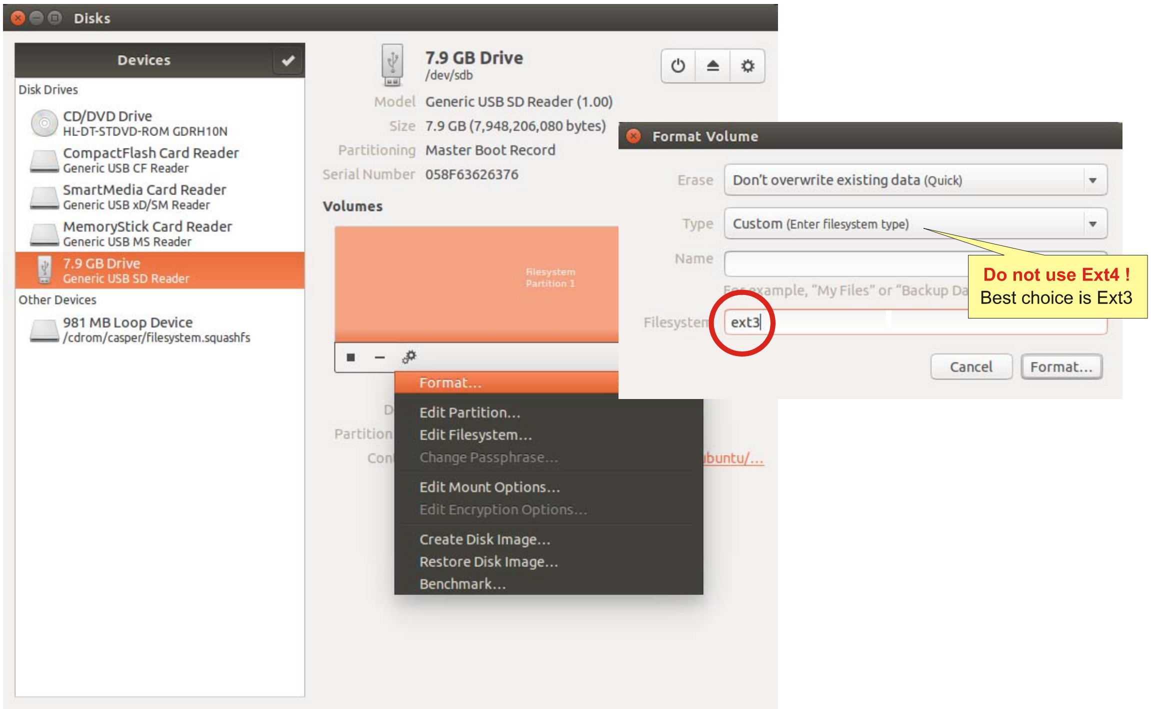


Partition	File System	Size	Used	Unused	Flags
/dev/sdb1	▨ ext3	7.40 GiB	263.91 MiB	7.14 GiB	

0 operations pending

Therefore, the microSD^{TM} partitioning process is completed. The next step will show the formatting of the microSD^{TM} card - but Ubuntu^{TM} 14.04 no longer supports 'ext3' as a standard setting. You have to select the 'custom' entry and input 'ext3' as Filesystem.

The next step requires the known Disk Utility.



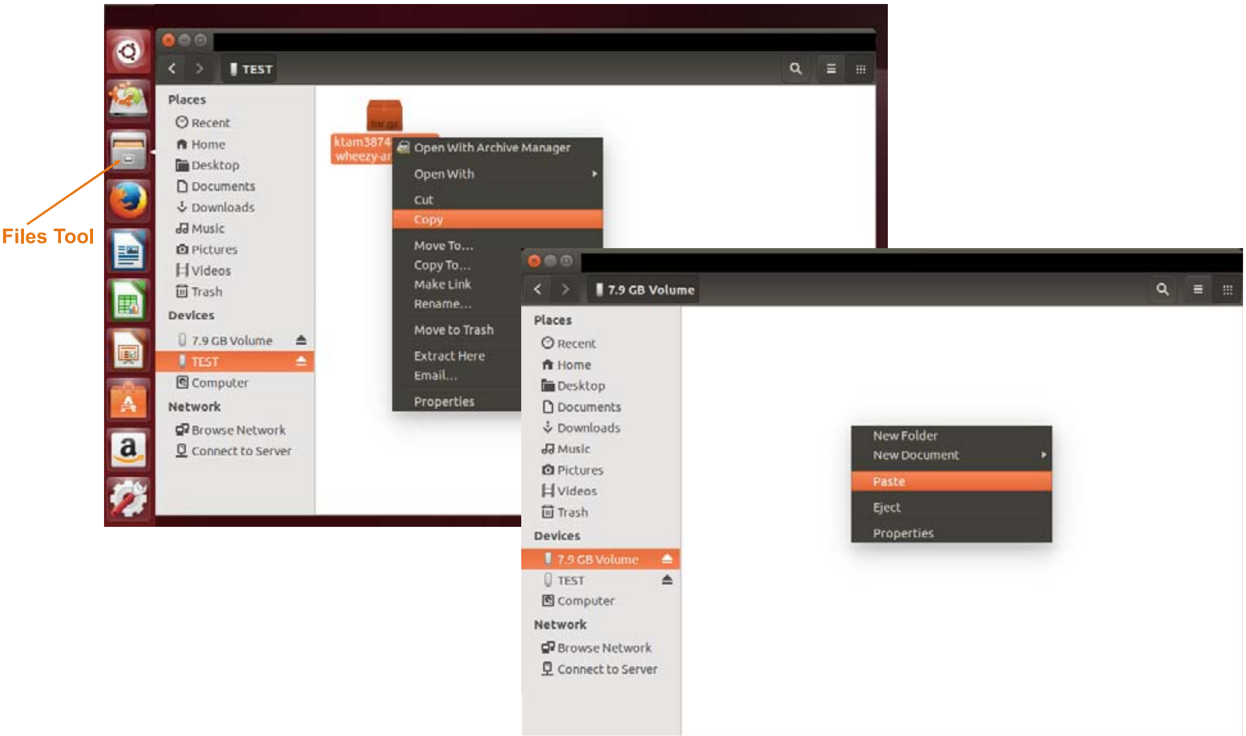
7.9 GB Drive
/dev/sdb
Model Generic USB SD Reader (1.00)
Size 7.9 GB (7,948,206,080 bytes)
Partitioning Master Boot Record
Serial Number 058F63626376

Format Volume

Erase Don't overwrite existing data (Quick)
Type Custom (Enter filesystem type)
Name
For example, "My Files" or "Backup Da
Filesystem ext3

Do not use Ext4 !
Best choice is Ext3

Thereafter copy the tarball ktam3874-gnome-wheezy-armhf.tgz (Debian^{TM} version) from a USB key (for example formatted with FAT32) to the microSD^{TM} card with the Files tool.

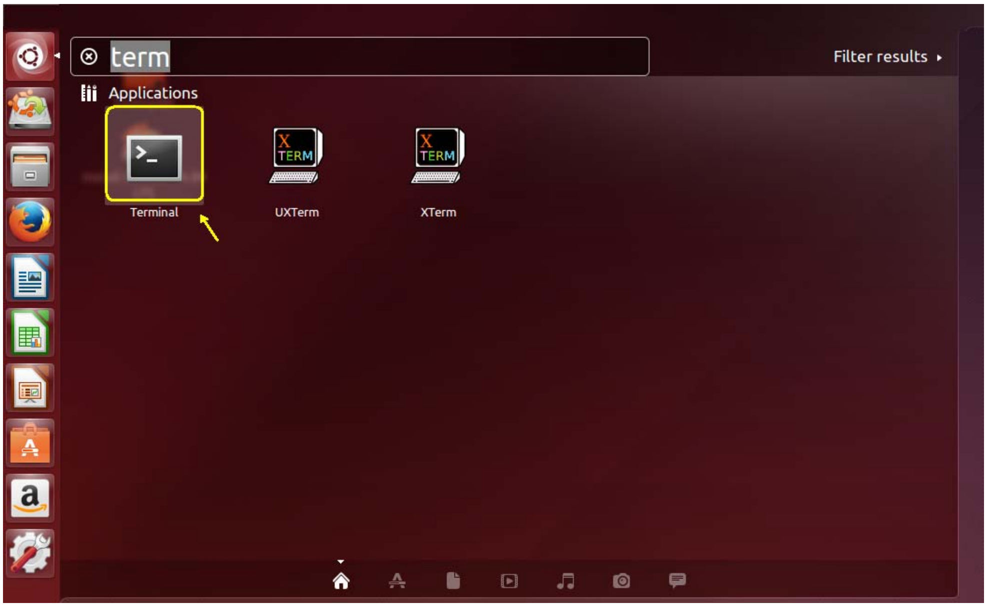


ATTENTION

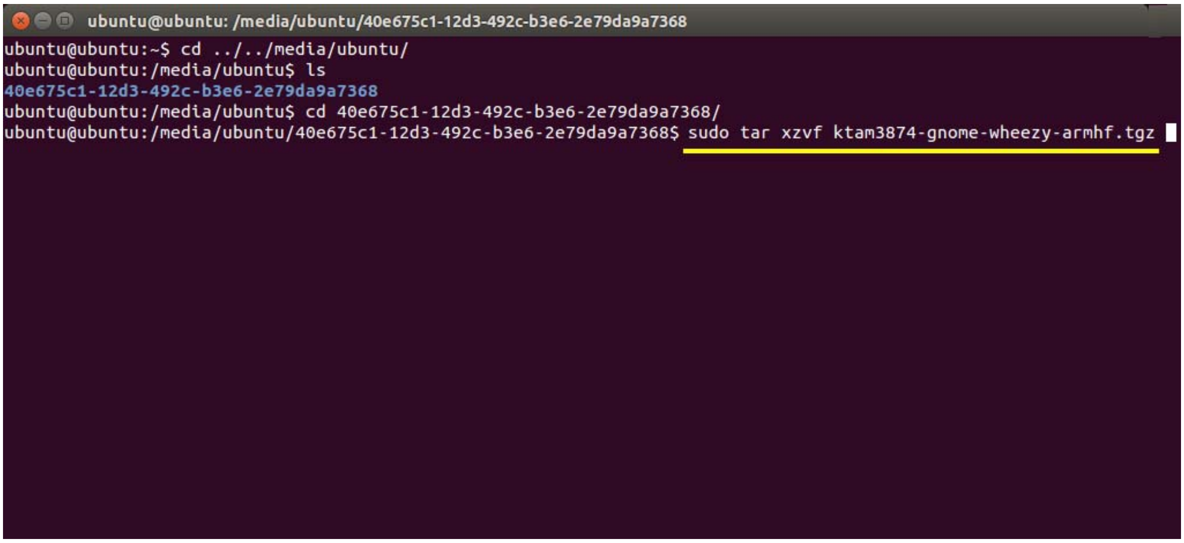
In the first release of Ubuntu^{TM} 14.04 the Archive Manager ignores the permissions, for example with 'sudo file-roller'. You have the same effect if you use the 'tar' command without 'sudo'.

DO NOT USE THE ARCHIVE MANAGER!

An alternative would be the console command 'tar' within the Terminal application.



The figure shows one way to extract the tarball (do not forget the 'sudo' statement).

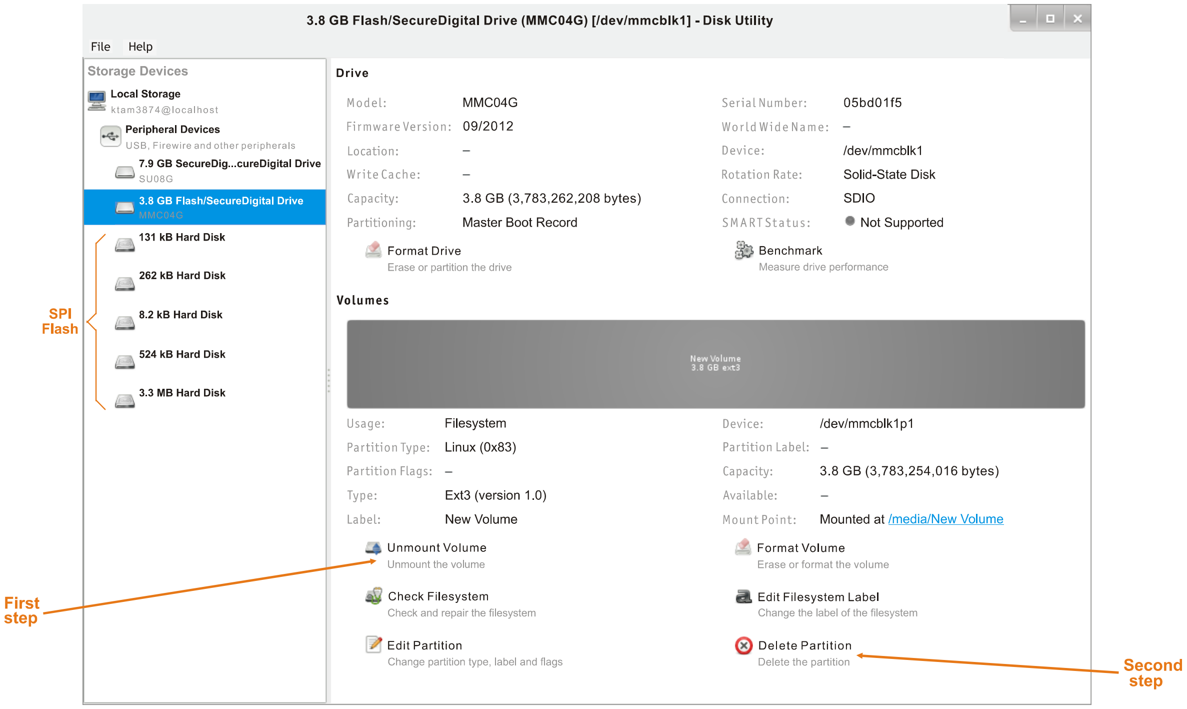

ubuntu@ubuntu: /media/ubuntu/40e675c1-12d3-492c-b3e6-2e79da9a7368

```
ubuntu@ubuntu:~$ cd ../../media/ubuntu/
ubuntu@ubuntu:/media/ubuntu$ ls
40e675c1-12d3-492c-b3e6-2e79da9a7368
ubuntu@ubuntu:/media/ubuntu$ cd 40e675c1-12d3-492c-b3e6-2e79da9a7368/
ubuntu@ubuntu:/media/ubuntu/40e675c1-12d3-492c-b3e6-2e79da9a7368$ sudo tar xzvf ktam3874-gnome-wheezy-armhf.tgz
```

Now the microSD^{TM} card is ready for use.

Appendix B: User Space on Onboard Flash

Requirement: an executable Linux® environment on a microSD™ card.
After booting Linux® you might call the disk utility from the Gnome™ shell (Applications/Accessories/Disk Utility) and select the 3.8 GB Flash/SecureDigital Drive. If the drive is already formatted the tool looks like this:



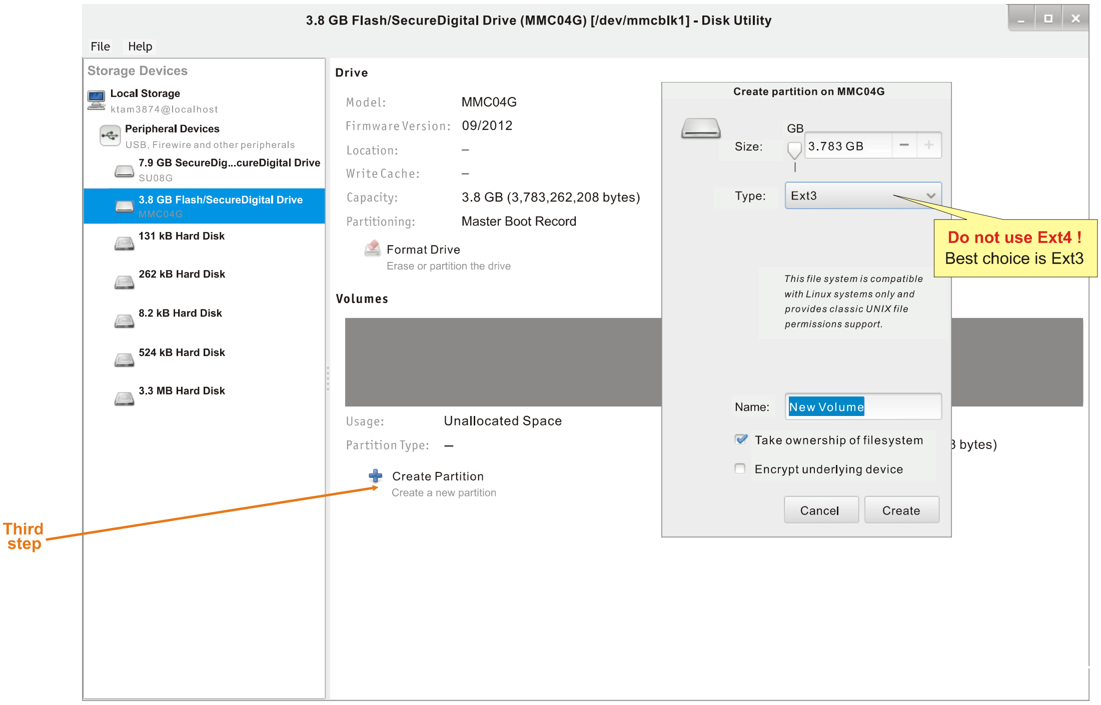
3.8 GB Flash/SecureDigital Drive (MMC04G) [/dev/mmcblk1] - Disk Utility

Drive

Model:	MMC04G	Serial Number:	05bd01f5
Firmware Version:	09/2012	World Wide Name:	–
Location:	–	Device:	/dev/mmcblk1
Write Cache:	–	Rotation Rate:	Solid-State Disk
Capacity:	3.8 GB (3,783,262,208 bytes)	Connection:	SDIO
Partitioning:	Master Boot Record	SMART Status:	● Not Supported

Usage:	Filesystem	Device:	/dev/mmcblk1p1
Partition Type:	Linux (0x83)	Partition Label:	–
Partition Flags:	–	Capacity:	3.8 GB (3,783,254,016 bytes)
Type:	Ext3 (version 1.0)	Available:	–
Label:	New Volume	Mount Point:	Mounted at /media/New Volume

otherwise the tool shows the following view:



3.8 GB Flash/SecureDigital Drive (MMC04G) [/dev/mmcblk1] - Disk Utility

Drive

Model:	MMC04G
Firmware Version:	09/2012
Location:	–
Write Cache:	–
Capacity:	3.8 GB (3,783,262,208 bytes)
Partitioning:	Master Boot Record

Usage: Unallocated Space
Partition Type: –

Create partition on MMC04G

Size: 3.783 GB (GB)
Type: Ext3

This file system is compatible with Linux systems only and provides classic UNIX file permissions support.

Name: New Volume

☑ Take ownership of filesystem
□ Encrypt underlying device

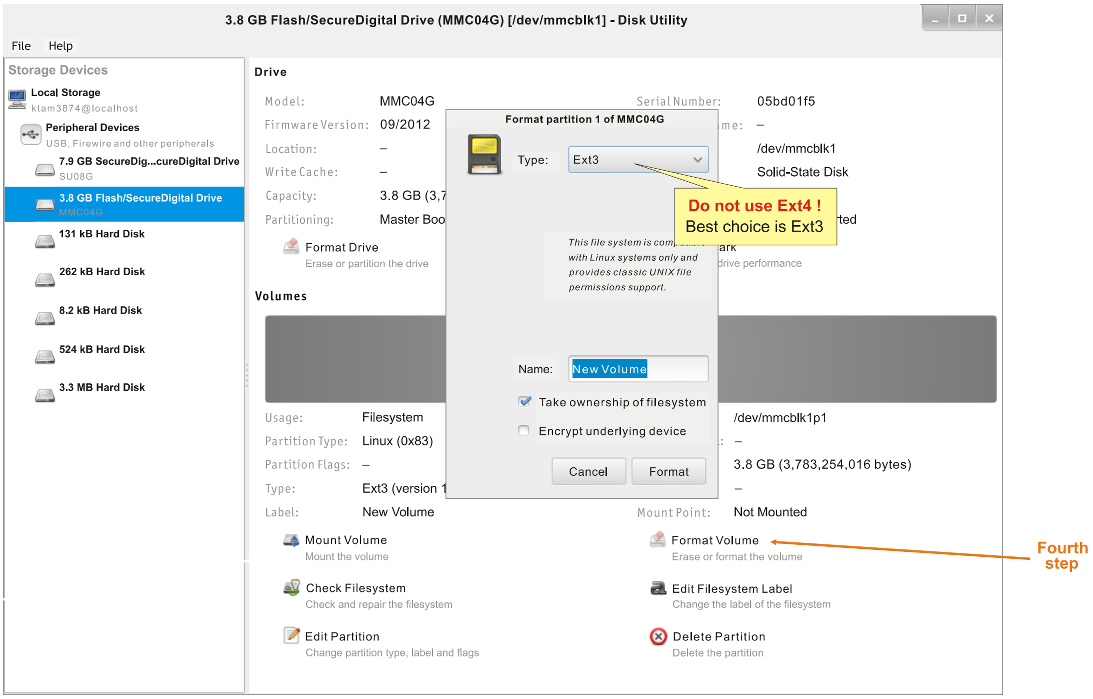
Do not use Ext4 !
Best choice is Ext3

The easiest way to prepare the onboard flash is to use one partition. The user space really needs an Ext-format; therefore only Ext2 to Ext4 are possible.

ATTENTION

Do not use the Ext4 format because U-Boot cannot boot the Kernel from an Ext4 partition.

After partitioning the onboard flash requires formatting.


3.8 GB Flash/SecureDigital Drive (MMC04G) [/dev/mmcblk1] - Disk Utility

Model: MMC04G
Serial Number: 05bd01f5
Firmware Version: 09/2012
/dev/mmcblk1
Location: –
Solid-State Disk
Write Cache: –
Capacity: 3.8 GB (3,7
Partitioning: Master Boo

Format partition 1 of MMC04G

Type: Ext3

Do not use Ext4 !
Best choice is Ext3

This file system is com... with Linux systems only and provides classic UNIX file permissions support.

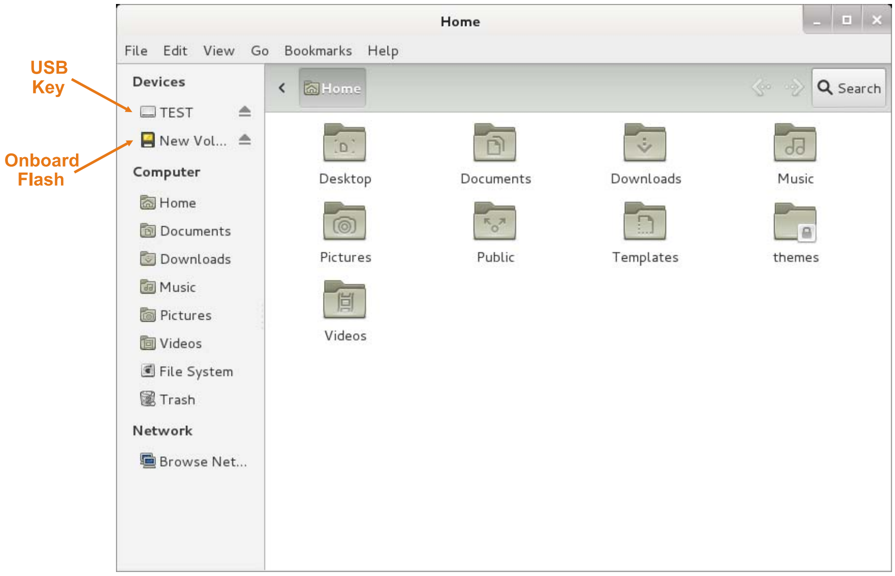
Name: New Volume

☑ Take ownership of filesystem

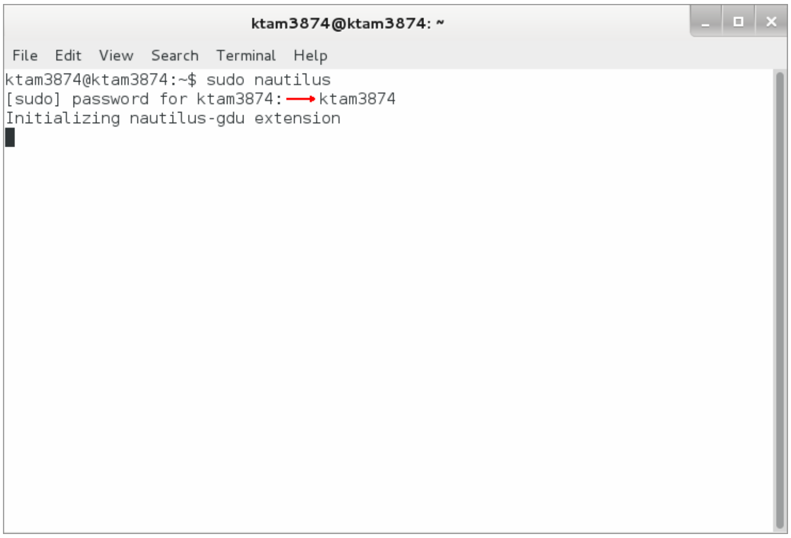
☐ Encrypt underlying device

Usage: Filesystem
/dev/mmcblk1p1
Partition Type: Linux (0x83)
Partition Flags: –
3.8 GB (3,783,254,016 bytes)
Type: Ext3 (version 1
Label: New Volume
Mount Point: Not Mounted

Subsequently copy the tarball ktam3874-gnome-wheezy-armhf.tgz (Debian^{TM} version) from the Desktop PC to a USB key (for example formatted with FAT32) or to the present microSD^{TM} card. On the target platform copy this file from the storage medium to the onboard flash, e.g. with the shell tool ' Applications/Accessories/Files'.

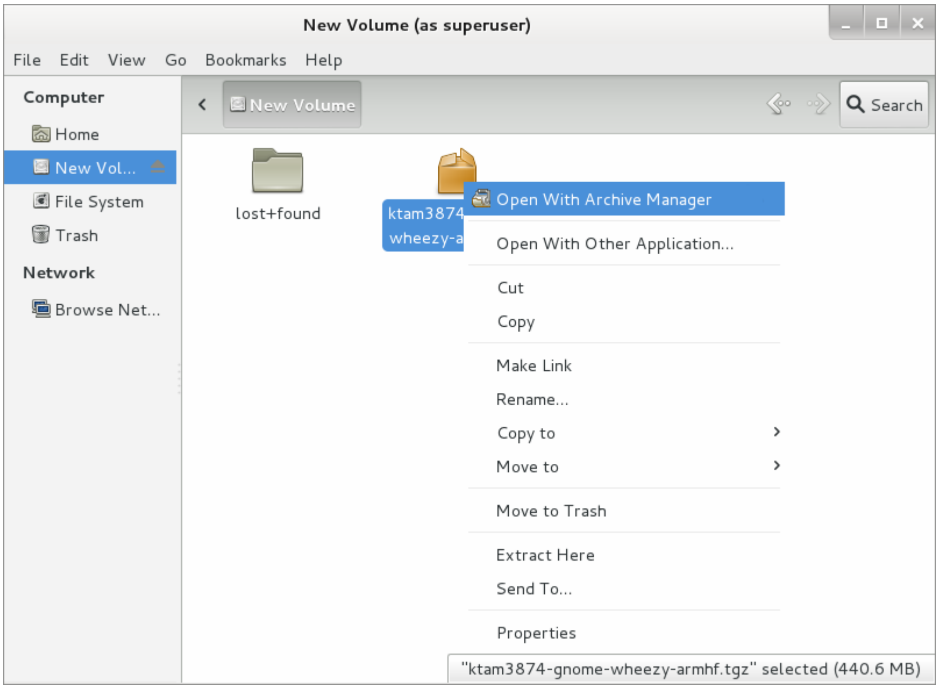


The same tool allows the extraction of the tgz-file but first you should check the current date - if the system date is older than the file date the extraction tool generates error messages. As a second condition you need root rights. The only way to do this leads trough the terminal program.

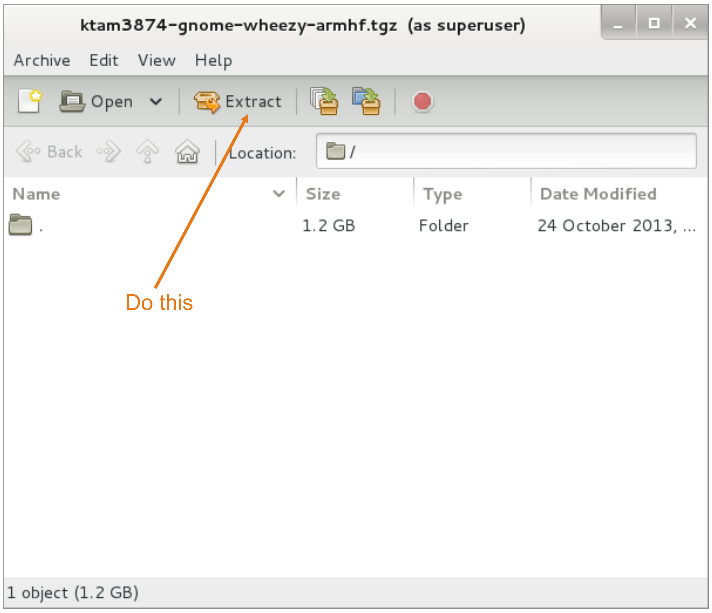


```
ktam3874@ktam3874:~$ sudo nautilus
[sudo] password for ktam3874: → ktam3874
Initializing nautilus-gdu extension
```

The next pictures illustrate the approach.

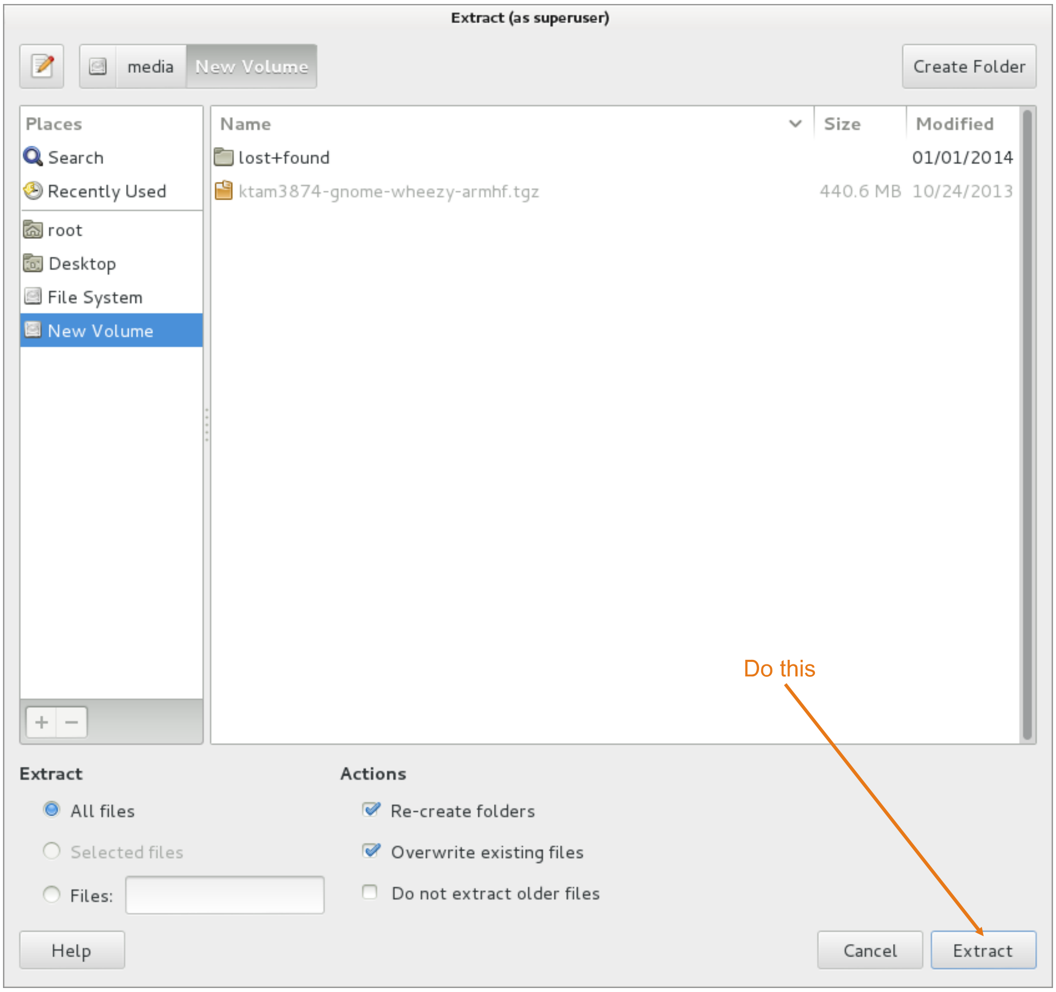


"ktam3874-gnome-wheezy-armhf.tgz" selected (440.6 MB)



Name	Size	Type	Date Modified
.	1.2 GB	Folder	24 October 2013, ...

Do this



Name	Size	Modified
lost+found		01/01/2014
ktam3874-gnome-wheezy-armhf.tgz	440.6 MB	10/24/2013

Extract

- ● All files
- ○ Selected files
- ○ Files:

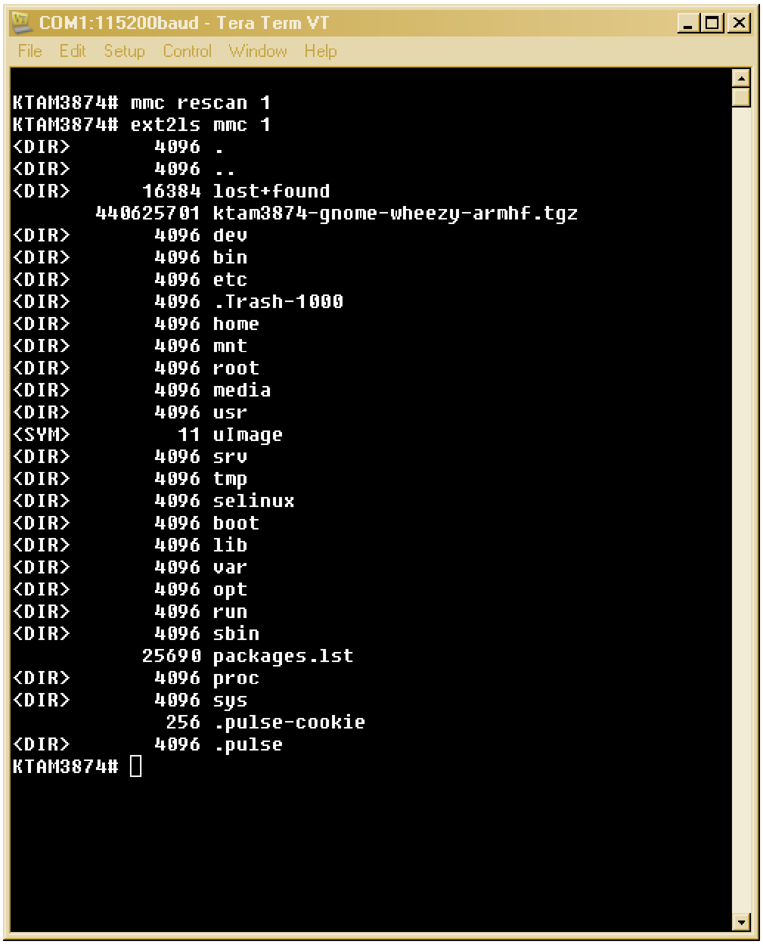
Actions

- ☑ Re-create folders
- ☑ Overwrite existing files
- □ Do not extract older files

Do this

Thereafter shutdown the target board and remove the microSD™ card. The next boot process runs Linux® from the onboard flash.

If you have trouble booting the Kernel please check the onboard flash partition in U-Boot with the following commands:



```
KTAM3874# mmc rescan 1
KTAM3874# ext2ls mmc 1
<DIR>        4096 .
<DIR>        4096 ..
<DIR>       16384 lost+found
       440625701 ktam3874-gnome-wheezy-armhf.tgz
<DIR>        4096 dev
<DIR>        4096 bin
<DIR>        4096 etc
<DIR>        4096 .Trash-1000
<DIR>        4096 home
<DIR>        4096 mnt
<DIR>        4096 root
<DIR>        4096 media
<DIR>        4096 usr
<SYM>          11 uImage
<DIR>        4096 srv
<DIR>        4096 tmp
<DIR>        4096 selinux
<DIR>        4096 boot
<DIR>        4096 lib
<DIR>        4096 var
<DIR>        4096 opt
<DIR>        4096 run
<DIR>        4096 sbin
           25690 packages.lst
<DIR>        4096 proc
<DIR>        4096 sys
             256 .pulse-cookie
<DIR>        4096 .pulse
KTAM3874# 
```

Appendix C: Reference Documents

KONTRON Technology A/S cannot guarantee the availability of internet addresses.

Document	Internet Address
General overview about relevant documents	http://processors.wiki.ti.com/index.php/Category:PSP
TI81XX PSP User Guide	http://processors.wiki.ti.com/index.php/TI81XX_PSP_User_Guide
TI® AM3874 Datasheet and Reference Manual	http://www.ti.com/product/am3874
TI® E2E™ Community	http://e2e.ti.com/support/dsp/davinci_digital_media_processors/f/716.aspx
Linux® EZ Software Development Kit	http://www.ti.com/tool/linuxezsdk-sitara
High Definition Multimedia Interface (HDMI®)	http://www.hdmi.org/manufacturer/specification.aspx
Open LVDS Display Interface Standard Spec.	http://www.national.com/analog/displays/open_ldi
IEEE 802.3® Specification (Ethernet)	http://standards.ieee.org/getieee802
Universal Serial Bus Specification (USB)	http://www.usb.org/developers/docs
SD™ Specification (SD Card)	http://www.sdcard.org/developers/tech/sdio/sdio_spec
High Speed Serialized AT Attachment (S-ATA®)	http://www.sata-io.org/developers
PCI Express® Base Spezification (PCI Express®)	http://www.pcisig.com/specifications
CAN Bus Specification (CAN)	http://www.semiconductors.bosch.de/media/pdf/canliteratur/can2spec.pdf
CAN Bus Background Information (CAN)	http://www.canbus.us

Appendix D: Document Revision History

Revision	Date	Author	Changes
S0057-I	06/05/14	M. Hüttmann	Added 'Qt™ Quick 1.x / QML' chapter, Interprocess Communication chapter and Appendix Ubuntu 14.04 LTS
S0057-H	03/13/14	M. Hüttmann	Added 'RS485 Interface Example' chapter, new Appendix A and B and an additional program in 'Qt™ 4.8' chapter
S0057-G	01/30/14	M. Hüttmann	Added 'Qt™ 4.8' chapter
S0057-F	01/14/14	M. Hüttmann	Added GTK3 example in chapter 'GTK 3.x'
S0057-E	12/19/13	M. Hüttmann	Added 'Graphic Programming (X11™, Cairo, GTK 3.x)' chapter
S0057-D	11/18/13	M. Hüttmann	Added 'Sound Programming with ALSA' chapter
S0057-C	10/31/13	M. Hüttmann	Added 'Kernel Boot from SPI™ Flash' and 'Root File System Boot from S-ATA® Drive' chapter
S0057-B	09/17/13	M. Hüttmann	Added 'Ethernet switch' and 'Linux® Programming Examples' chapter
S0057-A	07/09/13	M. Hüttmann	Added 'ShowDID' subchapter, 'U-Boot Environment' and 'Debian™ Linux® BSP' chapter
S0057-0	05/08/13	M. Hüttmann	Created preliminary manual

Corporate Offices

Europe, Middle East & Africa

Oskar-von-Miller-Str. 1
85386 Eching/Munich
Germany
Tel.: +49 (0)8165/ 77 777
Fax: +49 (0)8165/ 77 219
info@kontron.com

North America

14118 Stowe Drive
Poway, CA 92064-7147
USA
Tel.: +1 888 294 4558
Fax: +1 858 677 0898
info@us.kontron.com

Asia Pacific

17 Building,Block #1,ABP
188 Southern West 4th Ring Road
Beijing 100070, P.R.China
Tel.: + 86 10 63751188
Fax: + 86 10 83682438
info@kontron.cn

Texas Instruments