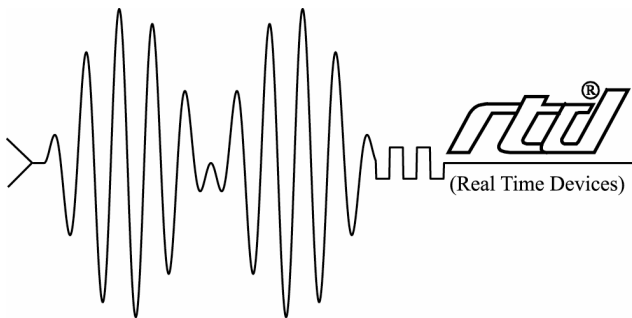


APPLICATION NOTE

Using Video4Linux2 SAA7134 Driver with VFG73xx



RTD Embedded Technologies, Inc.
"Accessing the Analog World"®

SWM-640020017
rev A

APPLICATION NOTE

Using Video4Linux2 SAA7134 Driver with VFG73xx



RTD Embedded Technologies, Inc.

103 Innovation Boulevard
State College, PA 16803-0906

Phone: +1-814-234-8087

FAX: +1-814-234-5218

E-mail

sales@rtd.com

techsupport@rtd.com

web site

<http://www.rtd.com>

Revision History

01/13/2006	Revision A issued
------------	-------------------

Published by:

RTD Embedded Technologies, Inc.
103 Innovation Boulevard
State College, PA 16803-0906

Copyright (C) RTD Embedded Technologies, Inc. All Rights Reserved.
Printed in U.S.A.

The RTD logo and utilityModule are registered trademarks of RTD Embedded Technologies. Linux is a registered trademark of Linus Torvalds. All other trademarks appearing in this document are the property of their respective owners.

Introduction

To use an RTD VFG7330 or VFG7350 frame grabber utilityModule™ under Linux, you need a device driver. The kernel already provides a Video4Linux2 device driver for boards using the Philips SAA7130, SAA7133, SAA7134, and SAA7135 chips. Support for RTD VFG73xx devices has been added to this driver.

This document provides a general overview of setting up and utilizing the Video4Linux2 SAA7134 driver. Fedora Core 4 running a 2.6.15 kernel built from kernel.org source was used for the creation of this application note.

Procedure for 2.4 Kernels

The Video4Linux2 project maintainers may or may not be submitting software updates for inclusion in the official 2.4 kernel source tree. Therefore, the 2.6 kernel should be used.

Procedure for 2.6 Kernels

Preparing the SAA7134 driver for use on a 2.6 kernel requires that you perform the following steps: 1) create a video-aware kernel, and 2) load the necessary kernel modules.

Creating a Video-Aware Kernel

Beginning with the 2.6.15 release, the official kernel source tree SAA7134 driver supports RTD VFG73xx devices. To obtain kernel source code, download it from the Linux Kernel Archives at <http://www.kernel.org>. Once you have the kernel source, you can configure and build the kernel.

I2C Support

VFG73xx devices require I2C support from the kernel. To enable kernel I2C functionality, make the following kernel configuration changes:

- * Under the **Device Drivers / I2C support** configuration submenu, **I2C support** should be compiled as a module.
- * Under the **Device Drivers / I2C support / I2C support / Miscellaneous I2C Chip support** submenu, **Philips PCF8574 and PCF8574A** should be compiled as a module if you want to use digital I/O.

SAA7134 Support

Since RTD VFG73xx devices use Philips SAA713x chips, they need the kernel SAA7134 device driver. To build the SAA7134 driver, make the following kernel configuration changes:

- * Under the **Device Drivers / Multimedia devices** configuration submenu, **Video For Linux** should be compiled as a module.
- * Under the **Device Drivers / Multimedia devices / Video For Linux / Video For Linux** configuration submenu, **Philips SAA7134 support** should be compiled as a module.

SAA7134 Sound Support

This section applies only if you want to use VFG73xx audio.

You need to be aware of some audio issues before proceeding. As of version 2.6.15, the kernel provides SAA7134 ALSA (Advanced Linux Sound Architecture) support and deprecates the OSS (Open Source Sound) driver. Because SAA7134 OSS support seems more mature than ALSA support, you are strongly encouraged to use OSS even though it is deprecated. Consequently, the remainder of this document assumes you will be using OSS.

To turn on the necessary sound features, make the following kernel configuration changes:

- * Under the **Device Drivers / Sound** configuration submenu, **Sound card support** should be compiled as a module.
- * Under the **Device Drivers / Sound / Open Sound System** configuration submenu, **Open Sound System (DEPRECATED)** should be compiled as a module.
- * Under the **Device Drivers / Sound / Open Sound System / Open Sound System (DEPRECATED)** configuration submenu, **OSS sound modules** should be compiled as a module.
- * Under the **Device Drivers / Multimedia devices / Video For Linux / Video For Linux / Philips SAA7134 support** configuration submenu, **Philips SAA7134 DMA audio support (OSS, DEPRECATED)** should be compiled as a module.

Once the kernel builds, install it and install the loadable modules. Boot the new kernel to continue.

Loading the Kernel Modules

Now that the proper kernel has been created, the appropriate kernel modules must be loaded into the executing kernel. To use a VFG73xx board, the following modules should be installed in the order given (see the paragraphs below for additional information):

- * videodev
- * ir-common
- * i2c-core
- * ir-kbd-i2c
- * v4l1-compat
- * v4l2-common
- * video-buf
- * saa7134
- * saa6752hs
- * soundcore
- * saa7134-oss
- * saa7134-empress
- * pcf8574

Some of the kernel modules above may automatically load when your system boots. This behavior likely depends upon the Linux distribution used. Modules required may change as kernel releases change.

Note that the saa7134 driver module automatically loads the saa6752hs driver if it detects a VFG73xx device which supports MPEG compression. For the saa7134 driver module, if audio functionality is desired then the `oss=1` module parameter must be given when the module is loaded.

Load the soundcore and saa7134-oss driver modules only if you wish to use VFG73xx audio.

The saa6752hs and saa7134-empres driver modules apply only to VFG73xx devices which support MPEG compression. Furthermore, they are required only if MPEG compression is desired.

If the saa6752hs driver is to correctly access the SAA6752 chips, it requires special module parameters when loaded. Please see the “Using the SAA6752 Driver” section below for more information.

The pcf8574 driver is required only if you want to use digital I/O. If the pcf8574 driver is to correctly access the PCF8574 chips, it needs special module parameters when loaded. Please see the “Using the PCF8574 Driver” section below for more information.

For those modules that require special parameters when loaded, it can grow cumbersome typing in the parameters each time the modules are loaded. To save time and to ensure that the arguments are present each time the modules are loaded, place the parameters in the `/etc/modprobe.conf` file.

At this point, you should be able to invoke `xawtv` (a Linux TV application) and watch uncompressed video coming from a VFG73xx card. If you encounter trouble getting the application to start successfully, you may need to invoke it with the `-nogl` command line option to disable OpenGL support.

MPlayer (a Linux movie player application) can be used to view the compressed video transport stream.

Video4Linux2 Device Files

The saa7134, saa7134-empres, and saa7134-oss driver modules create device file entries in `/dev` for each interface they find. These device files provide access to uncompressed video, compressed data, and sound resources respectively.

As an example, consider a single VFG7350 in a system. Assuming that the saa7134-empres and saa7134-oss drivers are loaded, the following device files are created:

File	Interface	Purpose
<code>/dev/dsp</code>	first	Uncompressed digital audio
<code>/dev/dsp1</code>	second	Uncompressed digital audio
<code>/dev/mixer</code>	first	Mixer control
<code>/dev/mixer1</code>	second	Mixer control
<code>/dev/vbi0</code>	first	Raw VBI samples
<code>/dev/vbi1</code>	second	Raw VBI samples
<code>/dev/video0</code>	first	Uncompressed video
<code>/dev/video1</code>	second	Uncompressed video
<code>/dev/video2</code>	first	Compressed audio and video
<code>/dev/video3</code>	second	Compressed audio and video

Using the PCF8574 Driver

The Linux kernel provides a driver for the Philips PCF8574 Digital I/O Expander chip. To make use of the VFG73xx digital I/O capabilities, the PCF8574 driver must be loaded. This driver looks at a limited set of I2C bus addresses for the presence of a PCF8574 chip because it's not possible to ask the chip for identification information. Unfortunately, the VFG73xx PCF8574 I2C address of 0x40 is not included in the set of addresses the driver looks at. Therefore, you must tell the driver which I2C addresses to assume are PCF8574 chips and which I2C addresses to ignore (due to other I2C devices on the VFG73xx and elsewhere in the system).

To instruct the driver to use a particular I2C address, specify the force parameter when loading the driver. The force parameter consists of a series of I2C adapter/address pairs and takes the form "force=<adapter>,<address>".

To instruct the driver to ignore a particular I2C address, specify the ignore parameter when loading the driver. The ignore parameter consists of a series of I2C adapter/address pairs and takes the form "ignore=<adapter>,<address>".

An I2C adapter essentially is an I2C bus number. An I2C address is a hexadecimal number indicating where on the bus the device sits; note that this number is actually shifted right one bit from the true address, therefore if a device sits at I2C address 0x40 you would specify 0x20. The adapter depends upon other I2C devices in the system, so you must know what other I2C devices exist in the system to correctly form the force and ignore parameters.

Consider the following command example to load the driver and set the probe/ignore addresses:

```
modprobe pcf8574 force=0x1,0x20,0x2,0x20 ignore=0x0,0x20,0x1,0x21,0x2,0x21
```

Let's analyze this command a bit more closely. The "force=0x1,0x20,0x2,0x20" part indicates there are two chips which should be assumed to be at address 0x40 on the second and third I2C busses. The "ignore=0x0,0x20,0x1,0x21,0x2,0x21" part indicates there are three non-PCF8574 chips (one at address 0x40 on the first I2C bus and two at address 0x42 on the second and third I2C busses) which should be skipped.

For an example of how to use VFG73xx digital I/O, please see the Software Product SWP-700010065 "Linux Software (VFG73xx)" available from the RTD web site.

Using the SAA6752 Driver

The Linux kernel provides a driver for the Philips SAA6752 MPEG Encoder chip. To make use of the VFG73xx MPEG compression capabilities, the SAA6752 driver must be loaded. This driver looks at a limited set of I2C bus addresses for the presence of a SAA6752 chip because it's not possible to ask the chip for identification information. Unfortunately, the VFG73xx SAA6752 I2C address of 0x42 is not included in the set of addresses the driver looks at. Therefore, you must tell the driver which I2C addresses to assume are SAA6752 chips and which I2C addresses to ignore (due to other I2C devices on the VFG73xx and elsewhere in the system).

To instruct the driver to use a particular I2C address, specify the force parameter when loading the driver. The force parameter consists of a series of I2C adapter/address pairs and takes the form "force=<adapter>,<address>".

To instruct the driver to ignore a particular I2C address, specify the ignore parameter when loading the driver. The ignore parameter consists of a series of I2C adapter/address pairs and takes the form “ignore=<adapter>,<address>”.

An I2C adapter essentially is an I2C bus number. An I2C address is a hexadecimal number indicating where on the bus the device sits; note that this number is actually shifted right one bit from the true address, therefore if a device sits at I2C address 0x42 you would specify 0x21. The adapter depends upon other I2C devices in the system, so you must know what other I2C devices exist in the system to correctly form the force and ignore parameters.

Consider the following command example to load the driver and set the probe/ignore addresses:

```
modprobe saa6752hs force=0x1,0x21,0x2,0x21 ignore=0x0,0x20,0x1,0x20,0x2,0x20
```

Let’s analyze this command a bit more closely. The “force=0x1,0x21,0x2,0x21” part indicates there are two chips which should be assumed to be at address 0x42 on I2C busses two and three. The “ignore=0x0,0x20,0x1,0x20,0x2,0x20” part indicates there are three non-SAA6752 chips (one at address 0x40 on the first I2C bus and two at address 0x40 on the second and third I2C busses) which should be skipped.

Audio/Video Inputs

A VFG73xx board has two stereo audio inputs and six video inputs for each of its SAA7133 video decoder chips. The audio inputs are 1) external audio input 1 and 2) external audio input 2. The video inputs are 1) composite 0, 2) composite 1, 3) composite 2, 4) composite 3, 5) s-video 0, and 6) s-video 1.

The driver assigns an audio input to each video input. Associations between video inputs and audio inputs can be found in the following table:

Video Input	Audio Input
composite 0	external audio input 1
composite 1	external audio input 2
composite 2	external audio input 1
composite 3	external audio input 2
s-video 0	external audio input 1
s-video 1	external audio input 2

Selecting a particular video input also selects the associated audio input.

Audio Signal Routing

Audio signals travel two paths through a single SAA7133 chip on the VFG73xx board. The first path is an analog audio pass-through where no processing is performed on the audio signal. The second path goes through the SAA7133 on-chip DSP where processing may be performed on the audio signal.

When audio signal is passed through, it makes its way through the board as follows:

- * audio input connector
- * enters SAA7133 chip
- * analog input crossbar
- * analog output crossbar
- * exits SAA7133 chip
- * audio output connector

Signal taking this path remains analog and cannot be captured via DMA.

When audio signal is not passed through, it makes its way through the board as follows:

- * audio input connector
- * enters SAA7133 chip
- * analog input crossbar
- * baseband A/D converter
- * digital input crossbar
- * DSP
- * digital output crossbar
- * stereo D/A converter
- * analog output crossbar
- * exits SAA7133 chip
- * audio output connector

Signal taking this path is converted from analog to digital then back to analog and may be captured via DMA.

The analog input crossbar controls which of the two external audio inputs provide the audio signal for the SAA7133 chip. To change the signal allowed through this crossbar, you need to change the audio input source or the recording source.

The digital input crossbar controls the digital matrix and signal source for each of the four DSP paths (MAIN, Dolby ProLogic, AUX1, and AUX2). The AUX1, AUX2, and Dolby ProLogic paths are not used. The MAIN path digital matrix is always set to AB stereo with automatrix off and the signal source depends upon the recording source.

The digital output crossbar controls the source of the DMA, D/A converter, and I2S channels. The MAIN left path always feeds DMA channel 1, the D/A converter left channel, and the I2S 1 & 2 left channels. The MAIN right path always feeds DMA channel 2, the D/A converter right channel, and the I2S 1 & 2 right channels. DMA channels 3, 4, 5, and 6 are not used.

The analog output crossbar controls which internal audio signal is allowed out of the SAA7133 chip. To change the signal passed off chip, you need to change the audio input source or change the recording source.

Audio Capture

Audio data can be captured via DMA from each of the two external inputs. To prepare the board for capture, you need to set the recording source. The SAA7134 driver supports three recording sources: Video, Line1, and Line2.

The Video recording source is associated with the TV tuner. Since the VFG73xx does not have a tuner, selecting this source results in passing no signal through and feeding no signal to the DMA.

The Line1 recording source is associated with external audio input 1. Selection of this source results in passing external audio input 1 through and also feeding it to the DMA.

The Line2 recording source is associated with external audio input 2. Selection of this source results in passing external audio input 2 through and also feeding it to the DMA.

When changing the recording source, the analog input, digital input, and analog output crossbars are updated. If source Video is used, the digital input crossbar directs the SIF demodulator/decoder output to the DSP MAIN path. If source Line1 or Line2 is used, the digital input crossbar pipes the baseband A/D converter output to the DSP MAIN path.

Known Limitations

PCF8574 I/O Expander Chip Driver

The standard Linux PCF8574 driver controls the I/O expander chip. This driver provides access to the chip via attribute files within the sysfs file system. The sysfs file system imposes limitations on reading attribute files. One such limitation is that actual device data can be read exactly once each time the attribute file is opened. Subsequent reads without moving the file offset return an end-of-file condition and reading data after moving the file offset returns stale data. Because of these shortcomings, you must open the PCF8574 "read" attribute file each time you wish to read it.

The 2.4 kernel does not provide a PCF8574 driver.

Even though the Philips PCF8574 chip provides an interrupt capability, the chip's interrupt pin is not connected on VFG73xx boards and the Linux driver does not support interrupts.

SAA7134 Driver Suitability

The Video4Linux2 SAA7134 driver has been minimally modified to support VFG73xx boards. If the SAA7134 driver does not work the way you want it to, you are free to adapt it in whatever way benefits you most. RTD Embedded Technologies neither developed nor maintains the Video4Linux2 project.

Video4Linux2 API Usage

No example programs exist to demonstrate the Video4Linux2 API (Application Programming Interface). Many Open Source projects use the API and illustrate its video programming concepts.

Video4Linux2 ioctl() Request Codes

The Video4Linux2 SAA7134 driver does not fully support all audio-related ioctl() request codes. Specifically:

- * VIDIOC_ENUMAUDIO and VIDIOC_ENUMAUDOUT are not implemented.
- * VIDIOC_G_AUDIO always sets the v4l2_audio structure name[] field to the string "audio".
- * VIDIOC_S_AUDIO succeeds but does nothing. To change the audio input, you must do so via the OSS sound mixer interface.

MPEG Stream Format

The Video4Linux2 SAA7134 driver supports only the MPEG transport stream. It does not support elementary audio, elementary video, packetized elementary, program, pack, or elementary audio/video streams.

Performance

VFG73xx device and Video4Linux2 SAA7134 driver performance depend upon many factors such as PCI bus traffic, IDE bus traffic (when saving data to disk), behavior of other kernel drivers, system activity, other devices in the system, and interaction between compressed and uncompressed data.

Technical Support

RTD does not offer technical assistance for 1) any kernel version lacking native VFG73xx device support, 2) OSS drivers, 3) video4linux CVS code, 4) ALSA drivers, and 5) configuring/building the kernel.

Conclusion

Although the Video4Linux2 code can be downloaded from the LinuxTV.org video4linux CVS (Concurrent Versions System) repository, working with it may be frustrating. For 2.4 kernels, there is no guarantee that the code will compile. For 2.6 kernels, there is no guarantee that the source code will compile on kernel versions older than 2.6.15. The CVS code is developed using a small number of kernel releases, so even if you are using a version newer than 2.6.15 the code may not compile because the kernel source is different enough from the kernel source used to develop Video4Linux2. Because the repository contains development code, the code may not compile no matter which kernel version you are using.

For more information, please consult the following resources:

- * ALSA <http://www.alsa-project.org>
- * Linux kernel documentation distributed with each kernel release and <http://www.tldp.org>
- * SAA713x chip Philips Semiconductor document [SAA7133; SAA7135 User Manual](#)
- * Video4Linux2 <http://www.linuxtv.org> and the video4linux mailing list
- * Video4Linux2 API Video4Linux2 source code API and <http://www.linuxtv.org>
- * modprobe modprobe(8) and modprobe.conf(5) man pages.
- * mplayer mplayer(1) man page and <http://www.mplayerhq.hu>
- * sysfs file system file system documentation distributed with each kernel release
- * xawtv xawtv(1) man page and <http://linux.bytesex.org/xawtv>

Getting Technical Support

If you require additional support with this product, or any other products from RTD Embedded Technologies, contact us using the information below:

RTD Embedded Technologies, Inc.
103 Innovation Boulevard
State College, PA 16803 USA

Telephone: (814) 234-8087
Fax: (814) 234-5218
Sales Information and Quotes: sales@rtd.com
Technical Assistance: techsupport@rtd.com
Web Site: <http://www.rtd.com>