Set-Top Box Configuration Guide

Amino x4x, x5x and x0xx series set-top boxes

July 2014

Document no. AM-000500-TC v119

Copyright

Set-top box Configuration Guide

July 2014

version 119

At the time of issue this guide applies to the set-top boxes (STBs) and software versions listed below:

Platform	Software version	Notes
x4x series STBs		
A129 A140 H140 A540PVR M540	3.3.1	Software version 3.3.1 is an upgrade that runs on all set-top boxes listed under x4x series STBs
x5x series STBs		
A150	4.0.0	
x0xx series STBs		
L1050	3.3.5	

This guide describes components that undergo continual development. The information in this guide is subject to change without notice at any time.

There may be visual deviations between graphics in the guide and the released software.

Comments about Amino documentation are welcome. Please submit feedback to docs@aminocom.com.

For further information about Amino or Amino products, please e-mail info@aminocom.com.

This guide is to be used in conjunction with the Amino set-top boxes (STBs) and software versions listed above. It does not refer to older STBs, unless required for clarification of a particular instruction.

Amino currently supports the Opera browser version 11 (for x4x series STBs) and version 12 (for x5x series STBs from v4.0.0 and x0xx series STBs from v3.4.0).

Operating System

The operating system required is a 32 bit x86 (ia32) Debian Squeeze, with bash as the default shell and Linux kernel 2.6.

It is possible to get the tools working on 64 bit Linux operating systems as long as the 32 bit compatibility libraries are installed (ia32-libs), however this is not a configuration supported by Amino.

Likewise, it is possible to use variations of x86 Linux other than Debian and to use a virtual machine such as VMWare or VirtualBox. Again, these configurations are not officially supported by Amino. Distributions other

than Debian may have different pre-requisites: they may require a certain cluster package or the addition of optional packages. Amino is currently unable to advise further on this subject.

This is **not** the embedded operating system in the STB.

Contents

Introduction

About this document

Amino set-top box (STB) management offers various means of editing/reading the configuration and controlling STB operation (for example, in order to initiate a software upgrade or reboot). This guide describes the configurable settings and options for Amino STBs. It assumes you have an STB and access to a suitable software release. It also assumes a basic understanding of the technologies involved, which include Linux command line operation.

For information on installing and upgrading Amino software, see the *Amino Software Installation and Upgrade Guide*.

Many of the control and customisation options described in this guide can also be implemented using JavaScript Media Access Control Extensions (JMACX) or C API calls using the AmiNET SDK. For more information about JMACX refer to the Amino *JMACX API specification* (AM-000502-TC) and the Amino *SDK User Guide* (AM-001394-TC).

Document conventions

Formatting	Usage
< ...>	Indicates a value that you need to replace with a system specific value (except where used in HTML or XML examples, where it is used in tags, as normal).
[...]	Indicates optional parameters - for example in commands or functions.
... | ...	Indicates choices – for example where an input can take one of a number of values.
`code font`	Indicates input and output values (for example, at a command line), as well as function, configuration, parameter and file names.
bold text	Used for emphasis and to indicate specific key presses. For example: Press the **Esc** key.
grey text	Commands or settings which are not in general use (for example, configuration settings that are reserved for Amino internal use).
blue text	Cross-reference (this is a “clickable” hyperlink if you are viewing the document electronically).

Document structure

Chapter	Outline
Introduction	General introduction.
Chapter 1, ”Configuration techniques“	This chapter introduces the options available for configuring STBs, and details how to configure certain configuration areas.
Chapter 2, ”STB configuration files“	This chapter lists some of the main STB configuration settings.
Chapter 3, ”Using STBremoteconf“	This chapter introduces **`STBremoteconf`** and explains how to use it to configure and control an STB.
Chapter 4, ”Using libconfig“	This chapter introduces **`libconfig`** and explains how to construct commands to view and/or edit STB configuration.
Chapter 5, ”Customising a software upgrade image“	This chapter describes how to customise the software upgrade before it is signed and loaded to the STB.
Chapter 6, ”STB configuration pages for Opera 11 onwards“	This chapter introduces the Management and Preferences pages for Opera 11 builds and describes how to use them to configure and control an STB.
Appendices	
Appendix A, ”NOR Flash settings“	Provides configuration examples and other supporting information for NOR Flash.
Appendix B, ”Settings file contents“	Provides configuration examples and other supporting information for Settings file instructions.
Appendix C, ”DHCPC file contents“	Provides configuration examples and other supporting information for **`dhcpc`** instructions.
Appendix D, ”Using the configuration techniques“	Gives methods of using the configuration techniques.
Appendix E, ”Media source URLs and HLS“	Provides information on constructing media stream URLs, plus information about HTTP Live Streaming (HLS).
Appendix F, ”Recovery from invalid video mode“	Describes a method of recovering from setting an invalid video mode.
Appendix G, ”Video output formats“	Lists the video output formats for the STBs.
Appendix H, ”Pairing an IR remote control with an STB“	Describes the method of pairing an IR remote control with an STB.
Appendix I, ”STBremoteconf stats command“	Describes the descriptors returned by the **`STBRemoteconf stats`** command.
Appendix J, ”Frequently Asked Questions (FAQs)“	

Document history

Issue	Date issued	Changes
119	July 2014	Changes and updates for x4x series STB releases 3.2.1 to 3.3.1. Information relating to Amino x5x release 4.0.0 and x0xx release 3.3.5 added. References to Amino x3x STBs now removed.
118	May 2013	Changes and updates for x4x series STB releases 2.9.0.
117	April 2013	Changes and updates for Ax4x series STB release 2.7.1, 2.7.2, 2.8.1 and 2.8.2
116	October 2012	Changes and updates for Ax4x series STB release 2.6.2. Changes and updates for Ax3x series STB release 0.18.10. Chapter on using configuration pages for Opera 9 now removed (classed as legacy information).
115	February 2012	Corrections: stbremoteconf SETCONFIG and GETCONFIG syntax corrected. NOR FLASH typo corrected to NORFLASH in code examples syntax.
114	December 2011	Changes and updates for Ax4x series STB releases 2.2.0 - 2.2.6, 2.3.0 - 2.3.5 and 2.4.2 - 2.4.3. Changes and updates for Ax3x series STB releases 0.18.3 - 0.18.6.
113	January 2011	Added RTSP server source address filtering information (settings file)
112	December 2010	Changes and updates for version 0.17.5 and 0.17.6 (for Ax3x series) and version 2.0.0 and 2.1.0 (for Ax4x series) releases.
111	November 2010	Changes and updates for version 0.17.3 (for Ax3x series) and version 1.0.0 (for Ax4x series) releases. Not issued - incorporated into issue 112.
110	August 2010	Changes and updates for version 0.17.2 release.
109	April 2010	New settings options added.
108	March 2010	General updates and corrections.
107	November 2009	New appendix listing **DHCPC** options added. **Netconf** appendix updated. Galio support information removed. Please see earlier version of this guide for information relating to the Galio browser.
106	November 2009	Chapter 7 added describing the new management and preference pages Configuration page settings are now included in Chapters 6 and 7. Appendix A, B and C have been updated. Guide updated for Opera 9 builds. Appendix G and H are both new. Support information for the AmiNET 120 and 124 removed. Please see earlier version of this guide for information relating to those platforms.

Issue	Date issued	Changes
105	June 2009	Removed confidentiality requirement. Password protection warnings added.
104	June 2009	Appendix G updated. Minor corrections.
103	May 2009	General Updates. Board Revision Numbers updated. **`STBremoteconf REFORMAT`** added.
102	January 2009	File Access Policy information added. **`OUTRES`** info updated. Summary tables updated. Scripts to control **`STBRemoteconf`** added. **`SAVE`** command info corrected.

Audience

This guide is intended for computer-literate people, who have a working knowledge of computing and networking principles. It is expected that readers are familiar with the principles of IPTV and have already performed a basic installation of an Amino STB.

Chapter 1—Configuration techniques

In this Chapter:

This chapter provides an outline of the configuration techniques for Amino STBs. More detailed descriptions will be provided in later chapters.

Not all settings described in this document are available to all STBs. An STB may or may not use a particular configuration option depending on the software build used to create the software image.

1.1 Configuration techniques

Amino STB management offers various means of editing/reading configuration and controlling basic STB operation (for example, to initiate a software upgrade or reboot). The techniques that make this possible are described below:

- **STBremoteconf** – A remote configuration tool that enables you to send configuration and control commands across the network to one or more STBs.
- **Configuration pages** – The local configuration pages can be accessed using an Amino Infra-red (IR) keyboard and television display to change configuration settings and carry out simple commands such as rebooting and initiating software upgrades. The term “Configuration pages” encompasses both the Management pages (for administrators) and the Preferences pages (for users). Some software releases with certain middlewares have their own configuration pages.
- **Imagecomponents** – The imagecomponents are the set of files that a software upgrade image is built from. These include configuration files and graphics files (for example, those used in the display to show that the STB is loading new software). In Table 1.2 below, changing `imagecomponents` means editing the contents of the `imagecomponents` directory before creating and signing a new upgrade image. If new files are added they should be added to the `flashcontents` file.
- **libconfig commands (via Telnet/SSH)** – If an STB has either the Telnet or SSH component installed this can be used to log in to it remotely and carry out configuration using libconfig commands.

These components are usually not installed for security reasons, so you will need to add them to the software image installed on the STB if you want to use them.

- **JMACX (JavaScript) –** The JavaScript Media Access Control Extensions (JMACX) API offers a full set of functions for controlling a range of STB operational areas via JavaScript embedded in web pages. See the Amino *JMACX API specification* (AM-000502-TC) for more information on using the JavaScript extensions.

Table 1.1 summarises which configuration files are used with which techniques.

Configuration files	Technique				
	STBRemoteconfig	libconfig commands	JMACX API	Configuration pages	Image components
SETTINGS	Y	Y	Y	Y	Y
HOSTNAME			Y	Y	Y
DHCP (see note below)					

Table 1.1:

Configuration files	Technique				
	STBRemoteconfig	libconfig commands	JMACX API	Configuration pages	Image components
TRUSTED DOMAINS			Y	Y	Y
CHANNELS			Y	Y	Y
FUNCTION KEYS			Y		Y
BROWSER CONFIGURATION			Y	Y	Y
PASSWORD					Y
NOR Flash	Y	Y	Y	Y	Y

Table 1.1:

The use of DHCP is as an ON/OFF configuration: configuration values supplied by the DHCP server are read only.

The configuration techniques described in the tables below are each suited to particular circumstances and contexts. For example, the Configuration pages offer local configuration, while `STBremoteconf` enables remote configuration. Editing image component files before the software is upgraded enables the same configuration to be set on multiple STBs, whereas changes via Telnet can only apply to a single STB.

Context	STBremoteconf	libconfig commands	JMACX API	Configuration pages	Image components
Before software is loaded					Y
After software is loaded	Y	Y	Y	Y	
Remote	Y	Y			Y
Local			Y	Y	
Single STB	Y	Y	Y	Y	Y[a]
Large deployment	Y[b]		Y		Y
Test settings	Y	Y		Y	

Table 1.2:

a. Not generally recommended, unless performing operations that cannot be done in another way, for example, adding a client program.
b. Not recommended – settings are only received if the STB is powered.

In Table 1.3 below:

- *Reflash* refers to deleting all the contents of the NAND flash and completely replacing the main software image. All configuration options, except those stored in NOR Flash, will be replaced with the configuration contained in the new software image.
- *Upgrade* refers to replacing just the main software image. In this case, some of the configuration files, for example the user settings, can be preserved. This can be controlled when the software image is built.

Operation	**STBremoteconf**	**libconfig commands**	**JMACX API**	**Configuration pages**	**Image components**
read configuration	Y	Y	Y	Y	Y
write configuration	Y	Y	Y	Y	Y
reboot	Y		Y	Y	
upgrade software	Y		Y	Y	
reflash	Y		Y	Y	

Table 1.3:

Chapter 2—STB configuration files

In this Chapter:

This chapter describes the configurable settings and options for Amino STBs, including the settings available from the configuration pages.Configuration files

In all the following instructions, **<release_name>** indicates the full name of the software release image directory that contains your Amino software release**.**For example, **Ax4x-3.2.1-Opera11** installs in **3.2.1-Ax4x-opera11**, hence **<release_name>** is **3.2.1-Ax4x-opera11**.

Configuration settings that persist over STB reboots and power-cycles are generally stored in configuration files that can be read or edited via the provided APIs (SDK or JMACX), administration tools or in some cases by directly editing the files.The exception is the NOR Flash, which can only be configured via the administration tools.

2.1 The settings file

The **settings** file contains most STB configuration items. The file provides various default settings for the STB software. Some of these can then be over-ridden by user settings held in the **usersettings** file. Configuration held in the **usersettings** file takes precedence over the **settings** file contents, but not all items in the **settings** file can be over-ridden. See details on **usersettings** and which configuration items it supports in the usersettings section of this document.

The upgrade system can be configured to specify whether, on an upgrade, the settings file on the STB should be retained or replaced by one held in the new software upgrade image. See the Amino *Installation and Upgrade Guide* (*AM-000501-TC*) for information on how to control this behaviour.

2.1.1 How to view or change settings

An initial settings file is supplied as part of a software release, in the following location:

<release_name>/upgradeimage/imagecomponents/

You can edit the **settings** file directly or replace it before creating a software upgrade image. Alternatively, you can use the various administration tools available to change settings once software is loaded to the STB. Management pages, **STBremoteconf** and **libconfig** (for example, via Telnet/SSH) can all change values in this file.

Once loaded on the STB, the settings file is stored in the following location:

`/mnt/nv/`

The settings used depends on the software and hardware build.

2.1.2 Example settings file

```
DOLBY_RF=""
REMOTECONF=""
IGMP_END_TIMEOUT="30"
IGMP_START_TIMEOUT="30"
RTSP_SCALE="6"
BROWSER_ENABLECOOKIES="Y"
BROWSER_HOMEPAGE="about:/start.htm"
BROWSER_HELPPAGE=""
HTTP_PROXY_IGNORE=""
```

```
HTTPS_PROXY=""
HTTP_PROXY=""
USE_PROXY="Y"
TOOLBAR_STATE="1"
#######DISPLAY_MODE="letterbox"
DISPLAY_MODE="panscan"
RTSP_END_TIMEOUT="6"
RTSP_START_TIMEOUT="4"
RTSP_SERVER="ncube"
DELAY_FACTOR="4"
REPEAT_RATE="8"
MAXIMUM_VOLUME="100"
MINIMUM_VOLUME="0"
DEFAULT_VOLUME="100"
FULLSCREEN="N"
```

2.1.3 Settings file contents

See Appendix B, ”Settings file contents“.

2.2 NOR Flash configuration

The NOR Flash contains settings required by the ROM. In some cases, this can include static network settings.

2.2.1 How to view or change settings

NOR Flash settings are not edited via a separate configuration file, instead, they can be accessed via the administration tools.

If you want to define values before loading the software image to the STB, you can do so by adding **`libconfig`** calls to the **`upgrade.sh`** file. This is an upgrade script that the STB executes when the new software image is loaded. If this file does not already exist, create it in the following location:

```
<release_name>/upgradeimage/imagecomponents/
```

The file will then be included when you create a software upgrade image. The NOR Flash settings example below gives an example of how to add **`libconfig`** calls to the upgrade script. See Chapter 4, ”Using libconfig“ for general information on using **`libconfig`**.

Alternatively, you can use the administration tools available to change settings once software is loaded to the STB.

The settings used depend on the software and hardware build.

For more information on which settings are used by Opera, see Appendix A, ”NOR Flash settings“.

2.2.2 Example script to set NOR Flash settings

NOR Flash settings cannot be edited via a configuration file, instead they need to be changed directly. The following example shows how to add **`libconfig`** calls that change NOR Flash settings to the upgrade script (**`upgrade.sh`**). This is included in the software upgrade image, and executed when the software is loaded on the STB.

The settings in this example define static multicast values:

```
#!/bin/sh
libconfig-set NORFLASH.MULTICAST_BOOTSTRAP_GROUP 239.255.1.1
libconfig-set NORFLASH.MULTICAST_BOOTSTRAP_PORT 11111
libconfig-set NORFLASH.MULTICAST_UPGRADE_GROUP 239.255.1.2
libconfig-set NORFLASH.MULTICAST_UPGRADE_PORT 11111
```

2.2.3 NOR Flash - list of settings

For further information see Appendix A, ”NOR Flash settings“.

2.3 Channel changer application

If you need a basic channel changer application some example HTML code is available for download from the Amino Support Site at Online Support > Downloads > JMACX and Application Development > channel_changer.tgz

The easiest way to use this is to install it on a webserver editing the **`init`** function (below) to include your stream addresses:

```
function init()
                {
                /* Setup channel list */
                add(0, "igmp://239.1.1.1:11111", "description1");
                add(1, "igmp://239.1.1.2:11111", "description2");
                add(2, "igmp://239.1.1.3:11111", "description3");
                add(3, "igmp://239.1.1.4:11111", "description4");
                add(4, "igmp://239.1.1.5:11111", "description5");
               addChannelList(0,5); // the 5 should be replaced with the number
of channels you are adding above
                //addChannelList();
                /* Select initial stream */
                start(0);
                }
```

Set the homepage on your box(es) to point to this page when they boot up.

2.4 Trusted domains file

The **`trsdmns.txt`** file is not normally supplied as part of a software release, instead, you will need to create this manually, as shown in 2.4.1. The **`trsdmns.txt`** file defines trusted domains and specifies whether JMACX and Macrovision are enabled for each. The **`trsdmns.txt`** file can be edited manually on the STB using the VI text editor.

The trusted domains file is only used with the Opera browser.

2.4.1 How to view or change settings

For further details, see

Chapter 6, ”STB configuration pages for Opera 11 onwards“

If you want to define trusted domains before loading the software image to the STB, you can do so by creating the file in the following location:

```
<release_name>/upgradeimage/imagecomponents/
```

The file will then be included when you create a software upgrade image.

Alternatively, you can use the various administration tools available to change settings once software is loaded to the STB.

Once created or loaded on the STB, the **trsdmns.txt** file is stored in the following location:

```
/mnt/nv/
```

2.4.2 Example trsdmns.txt file

```
3 macrovision.aminocom.com
2 *.bbc.co.uk
2 *.aminocom.com
0 *.
```

2.4.3 Trusted domains file contents

The file contains a list of trusted domains, each one on a separate line, with a code that indicates what is enabled. The format for each line is as follows:

```
<code> <url>
```

where:

<code>	Code that indicates what is enabled for the specified domain. Values can be as follows: **0** – neither Macrovision nor JMACX enabled **1** – Macrovision enabled, JMACX disabled (unlikely to need setting) **2** – JMACX enabled, Macrovision disabled **3** – Both JMACX and Macrovision enabled
<url>	The URL for the trusted domain. This can use asterisks as wildcards. For example: `3 *.aminocom.com`

Important: The rules specified in the **trsdmns.txt** file are applied in the order they are stated, so when adding domains ensure that the rules are applied as you intend.

For example, in the following file, **3** specifies that Macrovision and JMACX are enabled for the **macrovision.aminocom.com** domain, and **2** specifies that only JMACX is enabled for any other **aminocom.com** domain:

```
3 macrovision.aminocom.com
2 *.aminocom.com
```

In the following alternative example - with the same settings re-ordered - only JMACX is enabled for any **aminocom.com** domain, and so Macrovision is not enabled for **macrovision.aminocom.com**, even though **3** would normally enable Macrovision - since Macrovision for this domain has already been ruled out in the previous line:

```
2 *.aminocom.com
3 macrovision.aminocom.com
```

2.5 Usersettings file

The USERSETTINGS file contains settings that the STB user can control (through the Preferences pages).

These override the values specified for the same settings in the **SETTINGS** file.

2.5.1 How to view or change settings

The **usersettings** file is not normally supplied as part of a software release, instead, it is created when the user first configures preferences via the Preferences pages.

If you want to define local user settings before loading the software image to the STB, you can do so by creating the file in the following location:

```
<release_name>/upgradeimage/imagecomponents/
```

The file will then be included when you create a software upgrade image.

Alternatively, you can use the various administration tools available to change settings once software is loaded to the STB. Preferences pages and libconfig (via Telnet/SSH) can be used to change values in this file.

Values set via the **usersettings** file override the value for the same configuration setting elsewhere, for example, in the **settings** file.

Once created or loaded on the STB, the **usersettings** file is stored in the following location:

```
/mnt/nv/
```

The settings used depend on the software build.

For more information about which settings are used by Opera 11, see Chapter 6, ”STB configuration pages for Opera 11 onwards“

2.5.2 Example usersettings file

```
DOLBY_RF=""
DISPLAY_MODE="panscan"
SUBTITLES_SECOND_LANG="ita/it"
SUBTITLES_PREF_LANG="eng/en"
SECONDARY_LANG="fra/fre/fr"
PREFERRED_LANG="eng/en"
```

2.5.3 Usersettings file contents

The **USERSETTINGS** file can contain any of the settings normally specified in the **SETTINGS** file. Software builds for some browsers write certain settings to the **usersettings** file.

2.6 Hostname file

The **HOSTNAME** file contains just a single value, specifying the name of the STB.

2.6.1 How to view or change settings

The **HOSTNAME** file is not normally supplied as part of a software release. It is created when you specify a hostname via the Management pages.

If you want to define a hostname before loading the software image to the STB (not recommended), you can do so by creating the file in the following location:

```
<release_name>/upgradeimage/imagecomponents/
```

The file will then be included when you create a software upgrade image. Alternatively, you can use the various administration tools available to change settings once software is loaded to the STB. Management pages and **libconfig** (via Telnet/SSH) can be used to change values in this file.

Once created or loaded on the STB, the **HOSTNAME** file is stored in the following location:

```
/etc
```

2.6.2 Example Hostname file

The **HOSTNAME** file contains just a single line, such as the following:

```
AMINET
```

2.6.3 Hostname file contents

The following table describes the settings available in the **HOSTNAME** file.

Default values listed here are used if no value is supplied.

The example shows how to enter the setting if you are editing the **hostname** file directly (note that the values are all enclosed in quotation marks). See the appropriate documentation for information on how to enter values using other means (for example, JMACX or **libconfig** via Telnet).

Setting	Default value	Description
HOSTNAME		The hostname of the STB. **Default:** **“”** **Example:** **HOSTNAME=”AMINET”**

2.7 Function keys file

The **fkeys.conf** file specifies how the STB responds to remote control keys. If you want to change the configuration, you are not recommended to edit this file directly. Instead, it is recommended that a JavaScript key handler is used to determine the functionality of the remote control keys. Although the JMACX **SetKeyFunction** and **SaveKeyFunction** commands can be used (see How to view or change settings below), these two commands are now deprecated.

fkeys.conf was deprecated in software version 2.5.7, but has been reinstated from version 2.6.1. If you are using version 2.5.7 you should use the standard JavaScript key handling mechanisms to provide equivalent functionality in your portals. Alternatively, the Opera user.js file can be used to provide system-wide key handling.

2.7.1 How to view or change settings

An initial **fkeys.conf** file is supplied as part of a software release, in the following location:

<release_name>/upgradeimage/imagecomponents/

Use the JMACX functions to set the key functions:

ASTB.SetKeyFunction
ASTB.SaveKeyFunction

See the Amino *JMACX API specification* (AM-000502-TC) for more information, and for a list of the functionality that can be mapped to keys.

2.7.2 Example fkeys.conf file

The following section shows a **partial** example of a **fkeys.conf** file, containing entries for IR keyboard and the Amino IR remote control.

<...> shows where several lines have been removed to reduce the size of this example. For a complete list of all the remote control key codes, see the Amino *JMACX API specification* (AM-000502-TC).

```
# First, the keyboard mappings
0x40000072 1      # Browser Home
0x40000007 17     # Help
0xC0000001 1      # Browser Home
0x40000066 3      # Browser Back
0x40000067 4      # Browser Forwards
0x40000068 6      # Browser Reload
0x40000069 5      # Browser Stop
0x4000006A 2      # Browser Goto
<...>

# Now the amino remote mappings
0x40000003 0      # Last Ch
0x4000012C 18     # Channel Up= Browser Ch Up
0x4000012E 19     # Channel Down= Browser Ch Down
0x4000012F 41     # Volume up
0x40000130 42     # Volume down
0x40000131 40     # Mute
0x40000132 43     # Power
0x40000144 16     # Menu= Browser Preferences
<...>
```

2.8 Browser configuration files

Software releases will normally include the configuration file for the browser. For Opera it is **opera.ini**. In some builds, the STB reads some of its settings from this file.Refer to the documentation for your browser for details of the browser configuration file settings.

2.8.1 How to view or change settings for Opera

The browser configuration file for Opera is the **opera.ini** file. This is located in the **opera_home.tgz** in **<release_name>/upgradeimage/imagecomponents/**.

Once created or loaded on the STB, the browser configuration file is stored in **/mnt/nv**.

Also see Appendix B, "Settings file contents" for browser configuration settings that the configuration pages read from and write to. For a full explanation of the Opera Settings File see http://www.opera.com/support/usingopera/operaini/

2.8.2 Using the Ekioh browser with Beenius middleware

The following parameters are mandatory for running the Ekioh browser. If omitted, performance may be degraded or SSL connectivity affected (At present, SSL is not used).

STB vendors should include these parameters in the **ekioh.cfg** file, included in the STB software image:

```
browser.image.accelerate.all:true
browser.image.rendering.quality:medium
font.accelerate:true
font.cache.capacity:1MB
graphics.filters.accelerate:true
svg.image.accelerate.png:true
svg.image.accelerate.jpg:true
svg.image.accelerate.bmp:true
svg.image.load.timeout:0
svg.jsheap.full.warning:1MB
svg.jsheap.size:16MB
ssl.capath:/<path_to_permanent_storage>/ekioh/ssl
ssl.cale:/<path_to_permanent_storage>/ekioh/ssl/<ca_pem_file>
ssl.certs.js.install:true
ssl.validate.hostname:false
ssl.verify:false
svg.database.path:/<path_to_permanent_storage>/ekioh
svg.webstorage.database:/<path_to_permanent_storage>/ekioh/ekioh.db
```

For first time users who are not familiar with modifying software images, there is a publicly available pre-signed Ekioh image with these changes made here:

http://stbsupport.aminocom.com/upgrade/3.2.1.beenius.mcfs

2.9 The no_analog file

This feature applies to hospitality STBs only. If you add the **no_analog** file to your image components it will disable video from the analogue output of the STB after upgrading. Disabling the analogue output can be a requirement for some content providers who require that content can only be played via HDMI.

If you add a **no_analog** file to your image and upgrade, you can only remove the no analogue feature by removing the file from your image, re-signing the image and upgrading again without the file present.

2.10 HDCP policy file

For x4x series STBs, the default state of HDCP authentication is off, whereas for x5x and x0xx series STBs the default state is on. The HDCP policy file allows you to override the default state of HDCP authentication. The setting used within the file indicates that HDCP is either always on or always off (but HDCP may be dynamically enabled or disabled by the middleware using an SDK).

Create a text file named **`hdcp_policy`** then add it to **`imagecomponents`** and **`flashcontents`** as read only.

This file should contain:

"**`0`**" to disable HDCP

"**`1`**" to enable HDCP

"**`2`**" to control HDCP via the media control API

2.11 Dynamic DHCP settings file

The dynamic network and multicast upgrade settings retrieved from the DHCP server are stored in a file called **`dhcpcd-eth0.info`**. These values are only set if the STB uses dynamic network settings, otherwise, the static values are defined in the NOR Flash. The DHCPC values are shown in Appendix C, "DHCPC file contents" and are read-only.

2.11.1 How to view settings

The DHCPC values are stored in the following location on the STB:

`/var/dhcpc/dhcpcd-eth0.info`

The values can be accessed via **`libconfig`** commands (using Telnet/SSH) or JMACX calls.

2.11.2 DHCPC - list of settings

The full list of DHCPC settings available depends on the DHCP server installed and the Amino vendor options configured, that is, it depends on what values the DHCP server sends to the STB. The following table describes the main settings that are generally available.

All values are read-only.

Setting	Description
`BROWSER_HOMEPAGE`	The address of the page set as the homepage.
`DI`	The deployment index.
`DIMIN`	The minimum deployment index
`DNS`	The IP address of the DNS that the STB should be using.
`DOMAIN`	The network domain.
`GATEWAY`	The gateway for the STB.
`IPADDR`	The IP address of the STB.
`LEASE`	The DHCP lease time.

Setting	Description
MULTICAST_FS_ADDR	The multicast file system IP address.
MULTICAST_FS_PORT	The multicast file system port.
NETMASK	The netmask for the STB.
REBIND	The DHCP rebind time.
RENEWAL	The DHCP renewal time.

2.12 File access policy

The file access policy is a security feature used to control which files the user can access through the web browser. The default policies shown in Section 2.12.1 are hard coded in the system. For additional policies a file is created called **policy.def** in which you add any other local files that you might want to access – if they are not in the 'default' list (such as the Management pages).

All access policies are loaded from the file **/mnt/nv/policy.def** to avoid conditional compilation.

This approach means that the only changes to default policy are needed in the policy file and in some cases the policy file will not be needed at all.

2.12.1 The policy file

Each line of this file is treated as separate policy **<ACCESS_POLICY>:<ACCESS_URL>**, unless it begins with **#** in which case it will be treated as a comment.

ACCESS_POLICY has to be one of the following:

- **REJECT**: Access to URL should be rejected
- **ACCEPT**: Access to URL should be accepted
- **PASSWORD**: Access to URL should be accepted only if a valid password was entered
- **FIRST_PAGE_ACCEPT**: Access to URL should be accepted only if it is first page accessed by a browser
- **FIRST_PAGE_PASSWORD**: Access to URL should be accepted only if it is first page accessed by a browser **and** a valid password was entered
- **IGNORE**: Access to URL will be granted, but policy will not invalidate password-protection on entering new URL.

All policies are checked in the same order in which they are located in the policy file, the first **ACCESS_POLICY** from matching policy is returned.

If all policies are checked and no match was found, then the **REJECT** is returned.
A policy is matching if its **ACCESS_URL** is the same as the URL that is being checked, unless **ACCESS_URL** ends with ***** in which case all URLs that begin with **ACCESS_URL** (without *****) are also found as matched.

2.12.1.1 Default policies

```
#Anything not in this file will be automatically rejected
ACCEPT:file:///etc/htdocs/preference.html
ACCEPT:file:///etc/htdocs/keyboard.html
ACCEPT:file:///etc/htdocs/subtitles.html
ACCEPT:file:///etc/htdocs/video-pref.html
```

```
ACCEPT:file:///etc/htdocs/audio-pref.html
ACCEPT:file:///etc/htdocs/pvr.html
ACCEPT:file:///opt/fresco/htdocs/dvbt*
ACCEPT:file:///opt/fresco/htdocs/ozone*
#
#Password protect rest of the docs
PASSWORD:file:///etc/htdocs/*
```

2.12.1.2 Example policy file

```
ACCEPT:file:///mnt/nv/startup.html
ACCEPT:file:///mnt/nv/nonetaccess.html
```

2.13 Password

Software releases store the passwords in encrypted form in /mnt/nv/passwd and /mnt/nv/shadow. During STB boot these are copied to /etc/passwd and /etc/shadow.

Warning:
Amino strongly recommends that you change the default passwords when deploying our STBs.
See the Amino *Installation and Upgrade Guide* for information on changing these default passwords.

Chapter 3—Using STBremoteconf

In this Chapter:

This chapter describes how to set up and use the **STBremoteconf** Linux client to control and manage Amino STBs. It assumes a basic multicast upgrade system is installed and that the user is familiar with simple Linux command line functionality. For information on how to set up and manage the multicast upgrade system, see the Amino *Install and Upgrade Guide*.

3.1 STBremoteconf

STBremoteconf provides a mechanism for remotely configuring and controlling a local network of Amino STBs. It can be run on the command line by sending commands individually to one or more STBs, or commands can be collected into a script to save time. Commands are signed when they are created, so that the STB can check that the command has been sent by an authorised source.

Example functions include rebooting, modifying output modes or updating software, as well as viewing information about the STB.

Information in this chapter relates to **STBremoteconf** version 1.29 which is the latest version at time of writing. Earlier versions of **STBremoteconf** may not contain all the features described in this chapter.

3.1.1 How it works

STBremoteconf consists of a client application on a local PC and a server application **STBrcd** installed on the STB. **STBremoteconf** is operated by entering commands at the command line or by collecting commands into a script.

A normal operational sequence is as follows:

1. **Enter command:** The user enters a command at the command line on the PC. This includes specifying whether the command is to be sent to a single STB, sent to a multicast group, or saved for later use.
2. **Sign command:** The user is prompted to enter the passphrase for the configuration key, and then **STBremoteconf** signs the command. It also adds an expiry time after which the command will no longer be valid. By default this is 30 seconds after the time at which the command was signed.
3. **Send command:** If the command is to be sent to a single STB or to a multicast group, it is sent immediately. If the command is saved to be sent later, a confirmation confirms that it has been saved, and a separate **STBremoteconf** command allows saved commands to be sent.
4. **Verify command:** When the STB receives a command, the STB also uses its stored public configuration key to check that the command has been authorised, and rejects commands that are not authorised. It then checks the expiry time for the command against its internal clock. If the current time is later than the expiry time, it rejects it. This ensures that even if signed commands are used in a capture and replay attack, replaying them later will have no effect on the STBs. For this reason, the STB and the PC on which the **STBremoteconf** client is installed must use correct date and time settings.
5. **Execute command:** The STB executes the command using **STBrc**, and generates the appropriate response.
6. **Command response:** A response containing the information requested or confirmation that the command was executed (or an error report) is returned to the sending PC for display.

3.2 Installation

STBremoteconf is normally installed as part of an Amino multicast system. If an Amino multicast system is not installed, follow the multicast system installation instructions in the Amino *Installation and Upgrade Guide.*

3.2.1 Before you start

The instructions in this section assume that **STBremoteconf** and other components are to be installed as part of a multicast system. You will need the following:

- You will need to know the passphrase for the configuration key that you will be using to sign commands. For the Amino engineering key this is **stbrckey**. For further information on the use of passkeys, see the Amino *Installation and Upgrade Guide*.
- To send commands via **STBremoteconf**, you will need to know the IP address of individual STBs, or the IP address for a multicast group. STBs must be powered and connected to the network.
- For some of the commands (for example, multicasting commands), the DHCP server and multicast server must be running, and appropriate upgrade images must be available.
- You will need to log in as **root** or have **sudo** rights to send commands via **STBremoteconf**.

3.2.2 Installing STBremoteconf

You will need the following:

1. Operating system.

   STBremoteconf is supplied for use on Linux. The recommended platform is Debian Squeeze available from http://www.debian.org/releases/. The instructions here assume the use of this platform.

2. Installation files

   You will need the following files to install and use **STBremoteconf**:

   - **STBremoteconf**
   - **imgcfg**
   - **commands**

   These files are normally supplied in the **utils/** subdirectory of a full software release, and are installed as part of the multicast system set-up. If this has already been done, then there is nothing else to do to complete the installation.

3. In order to sign commands that you send with **STBremoteconf**, you will also need:

   - A private configuration key file – if you are using the Amino engineering keys, this is **STBrc-KEY.private** (this must match the public key file on the STB)
   - Perl modules

     The **STBremoteconf** client has dependencies on various external Perl modules. If the required modules are not installed on your system, you will see errors similar to the following when you try to use **STBremoteconf** for the first time:

     ```
     Can't locate IO/Socket/Multicast.pm in @INC (@INC contains:
                  /usr/lib/perl5/5.8.1/i386-linux-thread-multi...
     ```

   If this happens, you can download and install the required files.

3.2.3 Installing prerequisite Perl modules

The **STBremoteconf** client has dependencies on various external Perl modules. You can either install these before you start installing **STBremoteconf**, or try using **STBremoteconf** and then install the Perl modules that are required only if errors are reported.

3.2.3.1 To install the Perl modules required by STBremoteconf

The following procedure assumes that you have the Perl **CPAN** module installed and have internet connectivity. You must be logged in as root or have **sudo** rights.

1. Enter the following command:

   sudo perl -MCPAN -e shell
2. If the **cpan>** prompt is not displayed, then you are asked if you want to configure automatically. Press **Enter** or type **yes**.
3. If you are prompted to install any modules, press **Enter** or type **yes**.
4. Enter the following command:

   install Bundle::CPAN

   If you are prompted to install dependencies, enter **yes**.
5. Enter the following command:

   install Date::Manip

   If you are prompted to install dependencies, enter **yes**.
6. Enter the following command:

   install MIME::Base64

   If you are prompted to install dependencies, enter **yes**.
7. Enter the following command:

   install File::Copy

   If you are prompted to install dependencies, enter **yes**.
8. Enter the following command:

   install IO::Socket::Multicast

   If you are prompted to install dependencies, enter **yes**.
9. Enter the following command:

   quit

You now have all of the required Perl modules installed and your STBremoteconf client should now work.

If you have a recent Debian-based platform installed you **may** be able to replace the above stages with the following:

Enter the following command:

apt-get install libdate-manip-perl libmime-base64-perl libio-socket-multicast-perl

3.2.4 Upgrading STBremoteconf

STBremoteconf client can be installed or upgraded separately from upgrades to STB software, but if the client and STB applications are at different versions, only the functionality of the older version will be available. For

example, if the new STBremoteconf client application includes commands that are not yet implemented in the STB software, then the commands will not be available until the STB software is also upgraded.

In all the following instructions, **<release_name>** indicates the full name of the software release image directory that contains your Amino software release. For example, **Ax4x-3.2.1-Opera11** installs in **3.2.1-Ax4x-opera11**, hence **<release_name>** is **3.2.1-Ax4x-opera11**.

1. Log in as root, and copy the **imgcfg** to a directory in your path (the other Amino binaries, including **mcastbootd** must also be in this location).

   For example:

   ```
   cp <release_name>/utils/imgcfg /usr/local/bin
   ```

2. Copy the **STBremoteconf** and **commands** files to the location you want to run them from. This can be in the system path or in another location, but the two files must be stored in the same directory as each other.

   For example, to copy the files to a directory in the path:

   ```
   cp <release_name>/utils/STBremoteconf /usr/local/bin
   cp <release_name>/utils/commands /usr/local/bin
   ```

3. Copy the key file to an accessible location. Instructions in this guide assume it is in the following subdirectory of an Amino release: **/utils/keys/amino/**.

The upgrade is now complete.

3.2.5 Removing STBremoteconf

In order to remove STBremoteconf you will need to delete the following files:

- **STBremoteconf**
- **imgcfg**
- **commands**

These files are normally supplied in the **utils** subdirectory of a full software release.

3.3 Sending commands with STBremoteconf

The following steps outline an example command line interaction with **STBremoteconf**.

3.3.1 To send a command at the command line

The following steps assume you have **not** copied the **STBremoteconf** files to a directory in the system path (for example, **/usr/local/bin**). If you **have** copied the files, you do not need to navigate to the directory that the **STBremoteconf** file is stored in, and you do not need to enter **./** at the start of each command.

1. Set the **STBKEY** environment variable by entering a command in the following format:

   ```
   export STBKEY=<release_name>/utils/keys/amino/STBrc-KEY.private
   ```

   In order to set **STBKEY** as a permanent environment variable, add the above **export** command to your **.bashrc** file.

2. Navigate to the directory that the **STBremoteconf** file is located in.

   For example:

```
cd <release_name>/utils
```

3. Enter a command in the following format:

```
./STBremoteconf [<optional_arguments>] <host>
                                <command_option [parameter]>
```

See Example commands for examples of this syntax in use.

4. You are prompted to enter the pass phrase for the configuration key. For the Amino engineering configuration key this is **stbrckey**.

The passphrase can be set as the environment variable **STBPASS**, in the same way as for the **STBKEY** variable

If the command syntax and IP address(es) entered are valid, the client command is sent. For unicast operations (that is, commands sent to a single STB) a confirmation message is displayed which includes a status code returned by the STB. For multicast operations, only some commands will receive a confirmation message, for example GETVERSION.

If you have your own keys then you will have your own, different, passphrase.

You can now enter further commands as required.

3.3.1.1 Example commands

For example, enter the following commands to find out the software version on an A540 STB:

Unicast

```
./STBremoteconf -p 540 10.172.247.235 GETVERSION
```

Multicast

```
./STBremoteconf -p 540 225.10.10.10 GETVERSION
```

To find out the software version on an M540 STB:

Multicast

```
./STBremoteconf -p M540 225.10.10.10 GETVERSION
```

3.4 Using scripts to control STBremoteconf

STBremoteconf typically reads the location of the private key file from an environment variable, and the passphrase associated with it from input from the user when a message is sent. In some cases, such as where some degree of autonomy may be required, it is desirable not to be prompted for a passphrase. To do so, the value of the passphrase required can be stored in the **STBPASS** variable. For example, assuming that the Amino engineering keys are being used, the commands required to set this up would be:

```
export STBKEY=/usr/local/amino/keys/amino/STBrc-KEY.private
export STBPASS=stbrckey
```

These environment settings would only remain valid in the current shell session. For complete autonomy, these export statements could be added to the user's **.bashrc** file, or other shell settings file, depending on the Linux distribution being used.

If the two export variables were set, no request for a passphrase would be made by **STBremoteconf** when sending a message. Anyone using this feature should be aware of the potential security implications of storing the passphrase for the private key in this unprotected form.

3.5 STBremoteconf parameter syntax

STBremoteconf is operated by entering commands with the following syntax:

```
./STBremoteconf [-p <product_list>] [-e <expiry>] [-P <port>] <host> <com-
mand_options [<parameters>...]>
```

Argument	Usage
-p <product_list>	A comma-separated list of product codes that you want the command to apply to. For example: -p 140,140H (cannot be used in conjunction with -m or -s)
-e <expiry>	Time at which the message will expire. If you do not include this argument, it defaults to 30 seconds from the current time. The time specified should be acceptable to the **ParseDate** Perl command. For example: "**today**" "**1st Thursday in June 2008**" "**05/10/07**" "**12:30 Dec 12th 1880**" "**8:00pm December tenth**"
-P <port>	Port to send the command on. For unicast, the default port is **54321**. For multicast, the default port is **22222**.
<host>	The **host** argument that specifies where the command will be sent or enables you to save the command for use later. See below for more information.
-m	optional mac address in format 00:02:02:xx:xx:xx (cannot be used in conjunction with -s or -p). A maximum of five STBs can be contacted using MAC addresses.
<command_options [<parameters>...]>	The command that will be executed (see <command_options> argument)
-t	optional TTL value for multicast (defaults to 10).
-s	optional serial number (cannot be used in conjunction with -m or -p).
-u	optional UDP response timeout (number 1-999) (defaults to 10).

Alternatively, you can enter the command **./STBremoteconf** without any arguments to display help at the command line.

If you have copied the **STBremoteconf** to a directory in the system path (for example, **/usr/bin**), you can enter the commands without the **./** at the start.

3.5.1 <host> argument

The **<host>** argument must be included in all commands. It can be set with one of the following values, depending on where you want the command to be sent:

<multicast_group>

To send the command to all STBs listening on a particular multicast group, enter the IP address of the group (for example, as defined in the configuration of the STB or the DHCP server configuration). If no group has been configured on the STB, the default address **225.10.10.10** is used.

Use the multicast option with care to ensure that you are sending the command to the multicast group you have specified.

<unicast_IPaddress>

To send the command to a single STB listening on the network, enter the IP address for that box. Alternatively, you can enter a valid name (which has a DNS entry). You should send commands to a single box in this way if you want to configure settings for a single box, or if the command returns information such as status or version.

If the IP address is invalid, then the following error is returned:

Couldn't convert to internet address: Resource temporarily unavailable.

If the IP address is valid, but the STB does not respond to the command (for example, the STB is not connected or the address has not been allocated by the DHCP server), then the following error is returned:

Couldn't connect: Connection refused

3.5.2 Save

To sign a command and save it for multicasting or unicasting later, replace **<host>** with **SAVE**. This is useful as it allows you to collect multiple signed commands together to be transmitted altogether later, or to give a signed command to a third party that it is not appropriate to give keys to.

STBremoteconf saves the signed command to a **SAVEFILE.msg** file in the current directory, and you can send it later using the **SIGNEDCMD** command option.

Although the **SAVE** command can be used to create a batch of multicast or unicast commands, the expiry time from when the command is signed still applies (default is 30 seconds). This means that each command must be multicast/unicast before its expiry time is reached. The **-e** parameter can be used to specify a longer expiry time.

For example:

```
STBremoteconf -e "December 16th 2009" SAVE REBOOT
```

3.5.3 <command_options> argument

The **<command_options>** argument must be included in all commands, and specifies the operation you want to execute. Some options also require additional parameters. See Overview of command options for a list of available options.

3.5.4 Overview of command options

You can add one of the following to the **`STBremoteconf`** command. Alternatively, enter the command without adding any options to view **`STBremoteconf`** version information and a full list of available commands at the command line (see the full command list for details of when they should be used). You should only use these if you are fully aware of the implications for your STB

Not all commands will give a response.

Command	Usage
CEC_TVON	Switches on the TV using CEC*
CEC_TVOFF	Switches off the TV using CEC*
CEC_CMD	Sends a CEC command*
	*See the Amino J*MACX API specification* (AM-000502-TC) for more information about CEC commands.
CHANGEPAGE **`<url>`**	Displays the page specified on the STB browser.
DUMPCONFIG **`<location to store output file> <optional prefix>`**	Requests a dump of the libconfig settings.
EAS **`<message>`**	Sends an EAS (Emergency Alert System) message.
ETHERNET **`<value>`**	Sets the Ethernet speed.
GETCONFIG see SETCONFIG	Allows a key defined by libconfig (see Using libconfig to get and set values) to be obtained.
GETDI	Returns the deployment index.
GETSTAT	Returns STB statistics (not currently supported on x0xx series STBs).
GETVERSION	Returns the current software version on the STB.
HWINFO	Returns information about the STB hardware.
LOGREAD	Returns last few entries from syslog.
OUTFMT **`<format>`**	Sets the output format of the STB.
OUTRES	Sets the output resolution. (HD-capable boxes only)
PING **`<remote_host> <count> <backoff_secs>`**	Instructs the STB to send ICMP echo (ping) requests to a remote host.
PROC **`<proc_entry>`**	Sends specified **`proc`** entry to the STB.
PS	Runs the **`ps`** process on the STB.
REBOOT	Reboots the STB.
REFLASH	Wipes the NAND flash on the STB, then downloads new software.
REFORMAT (Minerva and Evo only)	Sets a flag on the STB to reformat the hard drive after a reboot. The STB will reboot to perform this command.

Command	Usage
RFMODE `<option>`	Enables/disables or resets the RF modulator (if present) on the STB.
SETCONFIG	Allows setting of a key defined by libconfig (except MAC address, Serial ID or Lock value.)
SIGNEDCMD	Sends a pre-prepared signed command out.
SNAPSHOT	Runs a script to obtain debugging information from the STB.
STATS	Returns status information about the STB.
SYSLOG `<new_host>`	Redirects the STB's `syslog` to the specified new host.
TVSYSTEM `<option>`	Sets the TV standard to be used.
UPGRADE `http://<host>/mc2.mcfs`	Instructs the STB to upgrade to software available at the host address specified.
UPGRADEMCAST `<IPaddress> <port>`	Instructs the STB to upgrade to software available at multicast address specified.
UPGRADEMCAST_VER `<IPaddress> <port> <software_version_string>`	Instructs STBs running different software versions to upgrade to the specified version.
VOD `<option>`	Executes Video-on-Demand operations (not currently supported on x0xx series STBs).

3.6 List of commands

The following is a list of the STBremoteconf commands supported at present.

CEC_TVON

Switches on the TV using CEC source and destination codes.

Usage

```
./STBremoteconf <host> CEC_TVON [source] [destination]
```

Example

```
./STBremoteconf 10.172.227.145 CEC_TVON
```

CEC_TVOFF

Switches off the TV using CEC source and destination codes.

Usage

```
./STBremoteconf <host> CEC_TVOFF [source] [destination]
```

Example

```
./STBremoteconf 10.172.227.145 CEC_TVOFF
```

CEC_CMD

Sends a CEC command.

Usage

```
./STBremoteconf <host> CEC_CMD <cmd byte string> [source] [destination]
```

Examples

To mute the TV

```
./STBremoteconf 10.172.227.145 CEC_CMD 43
```

To bring the TV from standby (an alternative to CEC_TVON)

```
./STBremoteconf 10.172.227.145 CEC_CMD 04
```

To send the TV to standby (an alternative to CEC_TVOFF)

```
./STBremoteconf 10.172.227.145 CEC_CMD 36
```

CHANGEPAGE

Instructs the web browser to go to a specific URL. Whatever the user is viewing at the time will be interrupted.

Usage

```
./STBremoteconf <host> CHANGEPAGE <url>
```

This does not currently work for igmp urls when using Ekioh 3.4.0.

Example

```
./STBremoteconf 225.10.10.10 CHANGEPAGE "http://www.aminocom.com"
```

The CHANGEPAGE command was previously unable to use fragment identifiers (#) in URLs. The fragment identifier # has now been replaced by a space character.

For example, instead of sending:

"**#|changepage|http://www.jsperf.com/faq#autorun|#1380642103#**"

the command sent is:

" **|changepage|http://www.jsperf.com/faq autorun| 1380642103** "

DUMPCONFIG

Requests a dump of the libconfig settings.

Usage

```
./STBremoteconf <host> DUMPCONFIG <location to store output file> <optional
prefix>
```

Example

```
./STBremoteconf 225.10.10.10 DUMPCONFIG /tmp/tempfile
```

EAS

Sends an EAS (Emergency Alert System) message. It is the responsibility of the middleware running on the STB to act on this message and perform any required operations.

Usage:

```
./STBremoteconf <host> EAS <message>
```

Example:

```
./STBremoteconf 225.10.10.10 EAS "<XML><Message>
                                  This is a message</Message><XML>"
```

To receive the message on the STB you need to use the AminoGeneric class. The service number for the **STBRemoteconf** EAS service is 13.

Usage example

```
// The event handler itself
function eventhandler(xml_event_string)
{
alert ("Message sent to eventhandler - " +
xml_event_string) ;
}
// Register the event handler for unique service identifier ID of 13 (EAS)
AminoGeneric.onEvent13="eventhandler";
```

For further information see Amino *Technical Note 024 - Using AminoGeneric*.

Certain characters cannot be used in the EAS message:

A message containing # or | will fail to get through.

A message containing ' will fail to get through and will change the AminoGeneric.onEvent13 registration to *undefined* which causes all following EAS messages to fail to get through.

ETHERNET

Restricted: Not for use with OTP STBs.

Sets the Ethernet speed to one of the following values:

```
AUTO
10HD
10FD
100HD
100FD
```

10BaseT Ethernet is not supported by Ax4x series STBs.

Usage

```
/STBremoteconf <host> ETHERNET <value>
```

Example

```
/STBremoteconf 225.10.10.10 ETHERNET 100FD
```

GETCONFIG see SETCONFIG

GETDI

Returns the deployment index. This command is only useful via unicast.

Usage

```
./STBremoteconf <host> GETDI
```

Example

```
./STBremoteconf 10.172.227.145 GETDI
```

GETSTAT

Returns the following statistics:

maximum, minimum and current temperature, system uptime and HDD self-monitoring (SMART)

Usage (Non x4x STBs)

```
./STBremoteconf <host> GETSTAT System.Uptime
./STBremoteconf <host> GETSTAT System.TotalUptime
./STBremoteconf <host> GETSTAT Fan.OffDuration
./STBremoteconf <host> GETSTAT Fan.SlowDuration
./STBremoteconf <host> GETSTAT Fan.OnDuration
```

```
./STBremoteconf <host> GETSTAT Fan.TotalOffDuration
./STBremoteconf <host> GETSTAT Fan.TotalSlowDuration
./STBremoteconf <host> GETSTAT Fan.TotalOnDuration
```

Usage (x4x STBs only)

```
./STBremoteconf <host> GETSTAT Temp.Current
./STBremoteconf <host> GETSTAT Temp.Max
./STBremoteconf <host> GETSTAT Temp.Min
./STBremoteconf <host> GETSTAT System.Uptime
```

Usage (all HDD STBs)

```
./STBremoteconf <host> GETSTAT HardDrive0.SMART
```

Example

```
./STBremoteconf 10.172.227.145 GETSTAT Temp.Current
```

GETVERSION

Returns the software version of a box. This command is only available via unicast.

Usage

```
./STBremoteconf <host> GETVERSION
```

Example

```
./STBremoteconf 10.172.227.145 GETVERSION
```

Example response

```
Got back: 2.7.1-opera11
```

HWINFO

Returns the product, board revision and ROM type of the STBs. For unicast only.

Usage

```
./STBremoteconf <host> HWINFO
```

Example

```
./STBremoteconf 10.172.227.145 HWINFO
```

Example response

```
aminet130 (revision 2) ROM type: 0xc222-SS-S4
```

LOGREAD

Returns the last few entries from syslog. This command is only useful via unicast.

Usage

```
./STBremoteconf <host> LOGREAD
```

Example

```
./STBremoteconf 10.172.227.145 LOGREAD
```

OUTFMT

Sets the output format.

Restricted: Not for use with OTP STBs.

Usage

```
./STBremoteconf <host> OUTFMT <new output format>
```

Example

```
./STBremoteconf 225.10.10.10 OUTFMT CVBS-LC
```

The options are:

Value	Meaning
"CVBS-RGBOFF"	Enable composite output without RGB (where available)
"CVBS-RGBON"	Enable composite output with RGB (where available)
"LC-RGBOFF"	Enable S-video output without RGB (where available)
"LC-RGBON"	Enable S-video output with RGB (where available)
"CVBS-LC"	Enable composite with S-Video (where available)
"CVBS-LBR"	Enable composite and component (YPbPr) (where available)
"LBR-CVBS-LC"	Enable composite, S-video and component (YPbPr) (where available)
"RGB-CVBS-LC"	Enable RGB with composite with S-video outputs (where available)

OUTRES

Sets the output resolution. This command is only supported by HD-capable STBs (for example, A130, A530).

The correct frequencies for these settings are detected using **TVSYSTEM**. This is done to try and limit accidental misconfiguration.

The output resolution will be one of the following:

```
SD
480p
576p
720p
1080i
```

Usage

```
./STBremoteconf <host> OUTRES <new output resolution>
```

Example

```
./STBremoteconf 225.10.10.10 OUTRES 720p
```

PING

Instructs the STB to send a number of ICMP echo (ping) requests to a remote host. In a network with multiple STBs, this command can be multicast and then the network can be monitored for replies, to determine which STBs received the message.

Usage

```
./STBremoteconf <host> PING <remote_host> <count> <backoff_secs>
```

where

<count> Specifies the number of echo requests to send.

<backoff_secs> Specifies how long to wait before sending the first request (in seconds). If you specify a **backoff_secs** parameter of zero, the STB sends the ICMP echo requests as soon as it receives the message. If **backoff_secs** is greater than zero, the STB waits a random time interval between zero and **backoff_secs** seconds before sending the first ICMP echo, followed by one second between each request.

Example

```
./STBremoteconf 225.10.10.10 PING 139.255.1.1 10 0
```

PROC

Queries the STB for the value of the specified **proc** entry, for example, to query memory usage.

The Linux command **cat /proc/<proc_entry>** will give the same information as using **PROC <proc_entry>.** This command is only useful via unicast.

Usage

```
./STBremoteconf <host> PROC <proc_entry>
```

Example

```
./STBremoteconf 10.172.227.145 PROC meminfo
```

PS

Performs a linux **ps** on the STB. This command is only useful via unicast.

ps displays information about a selection of the active processes.

Usage

```
./STBremoteconf <host> PS [option]
```

The only option presently available is the **-w** switch, which increases the number of characters displayed on each line of output from the **ps** command from **79** to **255**.

Example

```
./STBremoteconf 10.172.227.145 PS -w
```

REBOOT

Triggers a reboot of the STB, which is most useful when restarting the device after making configuration changes in unicast mode. The command returns a status code indicating whether the command was accepted. Not permitted in multicast mode.

The STB automatically reboots after some commands, such as a **MCAST_UPGRADE**.

Usage

```
./STBremoteconf <host> REBOOT
```

Example

```
./STBremoteconf 10.172.227.145 REBOOT
```

REFLASH

Requests a reflash of the STB. Reflashing refers to deleting all the contents of the NAND flash and replacing the main software image (**mcs.mfs**). The update is done by

- formatting the flash memory and reloading it from the appropriate server for Ax3x series STBs.
- using the Recovery Image (RI) stored in the NAND flash for Ax4x series STBs.

Before using this command, ensure that valid software exists, and that no attempt is made to interact with the device (such as disconnecting mains power) during the update process. This command wipes any persistent setting stored in the NAND flash (**/mnt/nv**).

If an upgrade is required rather than a reflash then see the UPGRADE or UPGRADEMCAST command. For further information on the differences between upgrading and reflashing the STB see the Amino *Install and Upgrade Guide.*

Usage

```
./STBremoteconf <host> REFLASH
```

Example

```
./STBremoteconf 225.10.10.10 REFLASH
```

REFORMAT

The REFORMAT command is for Minerva and Evo clients only. It sets a flag on the STB to reformat the hard drive after a reboot.

STBs that do not have a hard drive will not be affected.

The STB will reboot to perform this command.

The command returns a status code indicating whether the command was accepted.

Warning:
All the content of the hard drive will be erased if you issue this command

This command is not permitted in multicast.

Usage

```
./STBremoteconf <host> REFORMAT
```

Example

```
./STBremoteconf 10.172.247.235 REFORMAT
```

RFMODE

Enables or disables the RF modulator (if present) and/or changes the current RF channel.

The following options are available:

Option	Description
<channel> ENABLE	Sets the RF channel to **<channel>** and enables the RF modulator.
<channel> DISABLE	Sets the RF channel to **<channel>** and disables the RF modulator.

Option	Description
<channel> PRESERVE	Sets the RF channel to **<channel>** and does not change the current setting of the RF modulator.
<channel>	Sets the RF channel to **<channel>** and does not change the current setting of the RF modulator.
ENABLE	Enables the RF modulator for the current RF channel.
DISABLE	Disables the RF modulator for the current RF channel

Usage

```
./STBremoteconf <host> RFMODE <option>
```

Example

This sets the RF channel to **38** and enables the RF modulator.

```
./STBremoteconf 225.10.10.10 RFMODE 38 ENABLE
```

SAVEFILE

Restricted: Not for use with OTP STBs.

Uploads a local file to the STB file store. This is particularly useful when setting a unique configuration to a device by replacing its **config.txt** file. Specifying the local file name followed by the full path to store the file on the STB.

The local file name is validated to ensure that it exists. The remote name is not validated. If the remote name is specified without directory information, the file is stored in the root folder. You can specify the complete path for the remote file name, but only existing folder names can be used (it is not possible to create sub-folders within the STB storage with this command).

Usage

```
./STBremoteconf <host> SAVEFILE <local_filepath> <remote_filepath>
```

Example

```
./STBremoteconf 225.10.10.10 SAVEFILE config.txt abcde.txt
```

This stores the file **config.txt** as **abcde.txt** in the root folder of the STB.

SETCONFIG, GETCONFIG

Sets or gets a key defined by libconfig

You cannot use SETCONFIG to change the MAC address, Serial ID or Lock value.

Usage

```
./STBremoteconf <host> SETCONFIG <config option> <value>
./STBremoteconf <host> GETCONFIG <config option>
```

Unicast example

```
./STBremoteconf 10.172.247.78 GETCONFIG NORFLASH.OUTPUT_RESOLUTION
Got back: HD720P50
```

Multicast example

```
./STBremoteconf 225.10.10.10 SETCONFIG NORFLASH.OUTPUT_RESOLUTION HD720P50
Command Set $Revision: 1.25 $
Multicasting...
This command will affect every box listening on this network
Are you sure? [Y/N]: y
Sending following command: #|setconfig|NORFLASH.OUTPUT_RESOLUTION|#1329299052#
copying from '/tmp/STBremoteconf.24757/TMPFILE' to '/tmp/STBremoteconf.24757/
TMPFILE.signed'
Signed image written to '/tmp/STBremoteconf.24757/TMPFILE.signed'
Awaiting response(s) on port 22223 (timeout = 10)
Response from STB: "MAC:00:02:02:2c:f5:46 SERIAL:GB2510D0000168 REPLY:OK"
Response from STB: "MAC:00:02:02:2d:db:3e SERIAL:GF2710D0005377 REPLY:OK"
```

Multicast example using a command option

In this multicast example the command argument **-p <product_list>** is used to get the output resolution from a group of A140 STBs:

```
./STBremoteconf -p A140 225.10.10.10 GETCONFIG NORFLASH.OUTPUT_RESOLUTION
Amino STBremoteconf $Revision: 1.29 $:
Command Set $Revision: 1.25 $
Multicasting...
This command will affect every box listening on this network
Are you sure? [Y/N]: y
Sending following command:
#|getconfig|NORFLASH.OUTPUT_RESOLUTION|#1327318039#|A140|#
copying from '/tmp/STBremoteconf.31324/TMPFILE' to '/tmp/STBremoteconf.31324/
TMPFILE.signed'
Signed image written to '/tmp/STBremoteconf.31324/TMPFILE.signed'
Awaiting response(s) on port 22223 (timeout = 10)
Response from STB: "MAC:00:02:02:2c:f5:46 SERIAL:GB2510D0000168 REPLY:HDNONE"
Response from STB: "MAC:00:02:02:2d:db:3e SERIAL:GF2710D0005377 REPLY:HDAUTO"
Response from STB: "MAC:00:02:02:37:07:d8 SERIAL:GB1011D0035194 REPLY:HDNONE"
Response from STB: "MAC:00:02:02:37:6e:dd SERIAL:GB1611D0058168 RE-
PLY:HD720P50"
Response from STB: "MAC:00:02:02:2c:f4:d6 SERIAL:GB2510D0000056 REPLY:HDAUTO"
```

```
Response from STB: "MAC:00:02:02:2c:f5:49 SERIAL:GB2510D0000171 REPLY:HDAUTO"
Response from STB: "MAC:00:02:02:2c:f4:eb SERIAL:GB2510D0000077 RE-
PLY:HD720P59"
```

SIGNEDCMD

Sends a pre-prepared signed command out. This requires a file generated by STBremoteconf using the SAVE command and allows the user to send out a command without needing a password or a private key, so it can be useful to allow untrusted parties to submit commands.

Usage

```
./STBremoteconf <host> SIGNEDCMD <file>
```

Example

```
./STBremoteconf 10.172.227.145 SIGNEDCMD SAVEFILE.msg
```

SNAPSHOT

Runs a script to obtain debugging information from the STB.

Usage

```
./STBremoteconf <host> SNAPSHOT <path/filename>
```

Example

```
./STBremoteconf 10.172.227.145 SNAPSHOT <path/tmp>
```

STATS

Returns a set of status values for the STB. This command is only useful via unicast.The STBremoteconf stats command returns information which can be accessed by looking at specific directories on the STB. The MAC address is returned, and also selected information from **`/proc/avcore, /proc/meminfo`** and from running an **`ifconfig`** command on the STB.

Further information about the **`stats`** command and a description of the returned parameters is available in Appendix I, ”STBremoteconf stats command“ .

Usage

```
./STBremoteconf <host> STATS
```

Example

```
./STBremoteconf 10.172.227.145 STATS
```

SYSLOG

Redirects the syslog of the STB to the new host you specify. Syslog handles logging of various event and debug messages on the STB. You can redirect the output to an alternative host, such as a local PC.

Depending on your system configuration, you may want to redirect syslogs individually.

You will need to set up your server to accept remote syslog messages. For further information on Syslog, see http://www.syslog.org/wiki/Main/SyslogHome.

Usage

```
./STBremoteconf <host> SYSLOG <new_host>
```

Example

```
./STBremoteconf 10.172.227.145 SYSLOG 234.56.78.0
```

TVSYSTEM

Changes the TV standard to one of a predefined set of values.

The following options are available:

Value	Example region
PAL-I	UK
PAL-M	Brazil
PAL-N	Argentina
PAL-G	Europe
PAL-B	Australia, Ghana
PAL-H	Belgium
NTSC-J	Japan
NTSC-M	USA

A complete list of the Worldwide TV Standards can be found in the Amino *TV Standards Guide*.

Usage

```
./STBremoteconf <host> TVSYSTEM <option>
```

Example

```
./STBremoteconf 225.10.10.10 TVSYSTEM PAL-I
```

UPGRADE

The UPGRADE command is used with Ax4x STBs, and instructs an STB to upgrade to the software offered at a particular address.

Usage

Unicast upgrade

```
./STBremoteconf <host> UPGRADE http://<location>/mc2.mcfs
```

Multicast upgrade

```
./STBremoteconf <host> UPGRADE igmp:// <Multicast Upgrade address>: <Upgrade
port number>
```

Examples

Unicast upgrade

```
./STBremoteconf 10.172.227.145 UPGRADE http://10.172.227.145/mc2.mcfs
```

Multicast upgrade

```
./STBremoteconf 10.172.227.145 UPGRADE igmp://239.255.230.100: 12345
```

UPGRADEMCAST

The UPGRADEMCAST command is used with x3x and legacy STBs and instructs one or more STBs to upgrade to the software offered on a particular multicast group. This command is only valid for software versions supporting multicast upgrade (it is not valid for TFTP booting software versions). A software version must be multicasting on the specified group when the command is sent. Once the upgrade is complete, the STB will automatically reboot into the new version, you can use the **GETVERSION** command to check that the new software version is in use.

This command retains various persistent settings files held in the flash, such as **settings**. For more information on the settings files Appendix B, ”Settings file contents“.

Usage

Upgrading one STB:

```
./STBremoteconf <host> UPGRADEMCAST <IPaddress> <port>
```

Upgrading multiple STBs:

```
./STBremoteconf -P <port> <Multicast Address of STBs Listen> UPGRADEMCAST
<Multicast Upgrade address> <Upgrade port number>
```

Examples

Upgrading one STB:

```
./STBremoteconf 10.172.227.145 UPGRADEMCAST 239.255.1.1 12345
```

Upgrading multiple STBs:

```
./STBremoteconf –P 1111 239.39.39.1 UPGRADEMCAST 239.240.1.1 11111
```

UPGRADEMCAST_VER

Instructs STBs to upgrade to a specified software version. It is only valid for software versions supporting multicast upgrade (it is not valid for TFTP booting software versions). The command takes three parameters: a numeric multicast address, a port number and the full software version string. A software version must be

multicasting on this group when the command is sent. The STB compares its current software version string with the upgrade version string specified, and only upgrades if the strings are different. Once the upgrade is complete, the STB will automatically reboot into the new version.

Usage

```
./STBremoteconf <host> UPGRADEMCAST_VER <IPaddress>
                                          <port> <version_string>
```

Example

```
./STBremoteconf 225.10.10.10
       UPGRADEMCAST_VER 255.50.50.51 11111 2.7.1-opera11
```

VOD

Allows remote video on demand operations to be executed. For example, a **PAUSE** operation may be required before you can send another command.

The following video on demand operations are supported:

PLAY
PAUSE
STOP
FF
RW

VOD PLAY should only be used to resume from a **PAUSE** state. To initiate a new video on demand session remotely, use the **CHANGEPAGE** command to point to the required video on demand resource.

Usage

```
./STBremoteconf <host> VOD <option>
```

Example

```
./STBremoteconf 225.10.10.10 VOD PAUSE
```

3.7 STBremoteconf key changing

It is possible to generate your own STBremoteconf key on Ax4x STBs from software version 2.7.1. Having your own key will therefore allow you to change the STBrc key for units in the field without involving Amino.

Before STBrc key changing:

for sending STBremoteconf commands you will have:

`STBrc-KEY.public`:
the public Amino engineering key, installed on the STB at manufacture

and

`STBrc-KEY.private`:
the private Amino engineering key, stored on the management hosts, used for signing STBremoteconf commands.

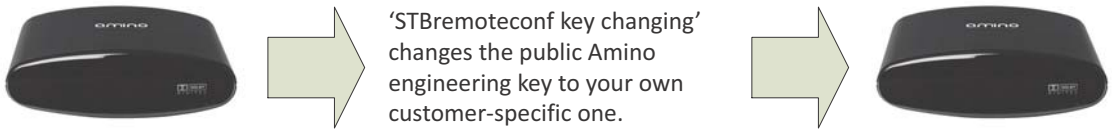
After STBrc key changing:

for sending STBremoteconf commands you will have:

`STBrc-KEY.public`:
which is now changed to your own specific key

and

`STBrc-KEY.private`:
which is now changed to your own specific key



'STBremoteconf key changing' changes the public Amino engineering key to your own customer-specific one.

3.7.1 Requirements

- A Linux Debian system with configured Amino utilities (**`imgcfg, rsakey`**)
- Amino keys on your system (**`KEY.private`**, which is delivered with each software build).

3.7.2 Preparing a new STBrc-KEY

1. Generate a new keyfile and passphrase using the RSA key generating utility **`rsakey`**:
   `$ rsakey -g -f STBrc-KEY`.
2. Enter your own passphrase when prompted (twice). The result will be 2 files:

   `STBrc-KEY.public`

   `STBrc-KEY.private`

   Sign the new public key with your Customer key (this example uses the generic Amino customer key):
   `$ imgcfg sign -f STBrc-KEY.public -o STBrc-KEY.public.signed -k /usr/local/keys/amino/KEY.private -w "markskey"`
3. Copy the newly generated **`STBrc-KEY.public.signed`** into your software image (in this example **`Ax4x-2.7.1-opera11`** is used):
   `$ cp STBrc-KEY.public.signed Ax4x-2.7.1-opera11/upgradeimage/imagecomponents/bin`
4. Copy the **`STBrc-KEY.private`** into **`/usr/local/keys/MYKEYS/`** on your PC.
5. Run the signupgrade image as normal and copy the result to the usual place.
6. Flash the STB. View the serial debug log - you will see the following message confirming your key is being installed:

Chapter 4—Using libconfig

In this Chapter:

The libconfig configuration tool enables you to interact with the STB configuration. In most cases, you can retrieve or set any of the available configuration settings.

4.1 Before you start

The instructions in this guide assume that the STBs you want to communicate with are powered and connected to the network. You will need the following:

- To send commands via **libconfig** you will need to know the IP address of individual STBs that you want to communicate with.
- To send commands via **libconfig** you will need a command line directly onto the box. This can either be via a **telnet/ssh** session or using a debug cable.
- You will need to know the password for the **root** user on the STB. This is **root2root** if you have not changed the default.

  If you want to follow the instructions here on using Telnet to send **libconfig** commands you will need to have software with Telnet enabled. See Enabling remote log-in tools for more information on including Telnet. Both Telnet and SSH are normally disabled for roll-out, so you will need to enable it by adding it to the software image on the STB.

4.2 libconfig configuration areas

libconfig can be used to access various areas of configuration, using a unique prefix for each area. In most cases, **libconfig** can read or write settings, but there are a few exceptions for which **libconfig** access is read-only.

4.2.1 NOR Flash configuration

The NOR Flash contains settings required by the ROM. In some cases, this can include static network settings.

4.2.1.1 libconfig prefix

To create the key for use in libconfig commands, prefix the setting name with **NORFLASH**. For example, for the **DI** setting, the libconfig key is **NORFLASH.DI**.

4.2.2 Settings configuration

The **settings** file contains various STB settings, such as languages, volume ranges and keyboard configuration.

4.2.2.1 libconfig prefix

To create the key for use in **libconfig** commands, prefix the setting name with **SETTINGS**. For example, to access the **RTSPSERVER** setting, the libconfig key is **SETTINGS.RTSPSERVER**.

For example:

```
libconfig-set SETTINGS.RTSPSERVER bitband
```

4.2.3 Hostname configuration

The **hostname** file contains just a single setting, that specifies the name of the STB.

4.2.3.1 libconfig prefix

To create the key for use in libconfig commands, prefix the setting name with **HOSTNAME**. For the **HOSTNAME** setting, the libconfig key is **HOSTNAME.HOSTNAME**.

4.2.4 Usersettings configuration

The **usersettings** file contains settings that the STB user controls (for example, through the Preferences pages). These override the values specified for the same settings in the **settings** file. The values persist when a new software image is loaded to the STB, but not if the box is reflashed.

4.2.4.1 libconfig prefix

To create the key for use in libconfig commands, prefix the setting name with **USERSETTINGS**. For example, for the **PREFERRED_LANG** setting, the libconfig key is **USERSETTINGS.PREFERRED_LANG**.

4.2.5 DHCPC configuration

The dynamic network and multicast upgrade settings retrieved from the DHCP server are stored in a file called **dhcpcd-eth0.info**. These values are only set if the STB uses dynamic network and multicast upgrade settings, otherwise, the static values are defined in the NOR Flash. The DHCPC values are read-only.

4.2.5.1 libconfig prefix

To create the key for use in libconfig commands, prefix the setting name with **DHCPC**. For example, for the **DNS** setting, the libconfig key is **DHCPC.DNS**.

4.3 Using libconfig to get and set values

The following section describes how to set new configuration values. We provide a command line utility **libconfig-set** to set configuration values. **libconfig-get** can be used to query the values.

4.3.1 libconfig-set command syntax

The libconfig commands for setting and querying STB configuration have similar syntax:

```
libconfig-set <config_key> "value"
```

or

```
libconfig-get <config_key>
```

Argument	Usage
`<config_key>`	Key formed from a prefix that relates to the configuration area and the name of the setting. For example, for the **`DEFAULT_VOLUME`** setting in the **`settings`** file, the key is **`SETTINGS.DEFAULT_VOLUME`**.
`<value>`	Specifies the new value for the configuration setting. Only required for the **`libconfig-set`** command. **Note**: If this value contains any spaces, it must be enclosed in quotation marks (for example: **`"file name.txt"`**), otherwise, the quotation marks are not required.

For example:

```
libconfig-set SETTINGS.DEFAULT_VOLUME "80"
```

or

```
libconfig-get SETTINGS.DEFAULT_VOLUME
```

4.3.2 To set a value via Telnet

The following steps outline how to send **`libconfig`** commands via Telnet.

This is usually disabled for security reasons.

1. From a console window, Telnet to the STB.

   For example, if your STB has an IP address 123.4.5.67, enter the following command:

   ```
   telnet 123.4.5.67
   ```

2. You are prompted to enter log-in details. The user name to log in as is **`root`**, the password is **`root2root`** (unless you have changed the password for the root user).

3. Enter **`libconfig`** commands, as required.

   For example:

   ```
   libconfig-get SETTINGS.DELAY_FACTOR
   ```

   This command returns the **`DELAY_FACTOR`** setting from the **`SETTINGS`** file on the STB, and displays it in the console, if you request a setting for which no value is set, then no return value is displayed.

   For example:

   ```
   libconfig-set SETTINGS.DELAY_FACTOR 5
   ```

```
[root@AMINET]# libconfig-set SETTINGS.DELAY_FACTOR 5
Setting 'SETTINGS.DELAY_FACTOR'='5'
SETTINGS.DELAY_FACTOR currently '(null)'
[root@AMINET]# libconfig-get SETTINGS.DELAY_FACTOR
[root@AMINET]#
```

The new value (5) is confirmed in the console, followed by a note of the value you replaced:

```
Setting 'SETTINGS.DELAY_FACTOR'='5'
SETTINGS.DELAY_FACTOR currently '(null)'
```

4. When you have finished, exit Telnet by typing the following command:

   ```
   quit
   ```

4.3.3 To set a value using commands in a script

This is particularly useful if you need to use set values in multiple STBs – for example, to set static network settings.

For an example of this, see Creating an upgrade script on page 65.

4.3.4 To set a value using the JMACX API

The **`ASTB.SetConfig/ASTB.GetConfig`** functions in the JMACX API call **`libconfig`** functions (see the Amino *JMACX API specification* AM-000502-TC for more information).

The STB management pages also use the **`ASTB.SETConfig/GetConfig`** function to modify settings.

Commands take the form:

```
ASTB.SetConfig (<password> , <config_item> , <value>)
```

Example use

```
ASTB.SetConfig (<password>), "SETTINGS.DELAY_FACTOR", "5")
ASTB.GetConfig ("SETTINGS.DELAY_FACTOR")
```

Parameters	Allowed values	Description
`password`		The password for the **`mngwrite`** user, as required for changing settings on the Management pages of the STB.
`config_item`		String containing the configuration item to set.
`value`		String containing the value to set the specified configuration item. Suitable values depend on the specified **`config_item`**.

Chapter 5—Customising a software upgrade image

In this Chapter:

You can customise software before it is loaded onto an STB by configuring the image components used to create the software upgrade image. The sections in this chapter describe how to configure the image components and provide additional information on some of the important components that you may need to use.

5.1 Configuring image components

Configuring image components enables you to customise the files used to create a software upgrade image. When the **signupgradeimage** script creates a new image, it includes the files that are listed in the **flashcontents** file and stored in the **imagecomponents/** subdirectory of the software release. You can edit these files or add new ones in order to customise the software upgrade image to your requirements.

For example, you can:

- edit configuration files before they are made available to STBs.
- add files needed to enable Telnet, SSH or Dropbear.
- add/edit an **upgrade.sh** script file that the STB will execute when it installs the new software upgrade image.
- replace graphics (displayed during the boot stages) with customised versions. For example, the **splash.gif** graphic.

All files in the **imagecomponents/** directory must be listed in the **flashcontents** file, otherwise, the **signupgradeimage** script will fail.

5.1.1 To configure image components

The general procedure for editing image components is as follows:

In all the following instructions, **<release_name>** indicates the full name of the software release image directory that contains your Amino software release**.** For example, **Ax4x-3.2.1-Opera11** installs in **3.2.1-Ax4x-opera11**, hence **<release_name>** is **3.2.1-Ax4x-opera11**.

1. Open the file that you want to edit. The image component files are in the following location:

   <release_name>/upgradeimage/imagecomponents

   Alternatively, you can add a new file to this location, such as a **chnls.txt** file you have copied from a pre-configured STB or an **upgrade.sh** file that contains **libconfig** commands to set NOR Flash values.

2. Edit the file, and save your changes.
3. Open the **flashcontents** file, and ensure that the file you have added/edited is listed there. This file is in the following location:

   <release_name>/upgradeimage

   The **flashcontents** file lists the files that will be included in a software upgrade image, and defines permissions for the files.

4. Use the **signupgradeimage** script to create the software image, as usual.

The new software upgrade image includes your new or edited files, and you can use it to upgrade your STBs.

5.2 Editing the flashcontents file

You are only recommended to edit the **flashcontents** file if you are adding new files to the image components used to create a software image, and the files are not currently listed in it. The **flashcontents** file is supplied in the following location in a software release:

```
<release_name>/upgradeimage
```

5.2.1 flashcontents file

The **flashcontents** file lists the files that can be included in a software upgrade image, and defines the permissions associated with these files when the software image is installed on the STB.

The file contains a list of file names and the permission associated with them, in the following format:

```
<permission> <file name>
```

For example:

```
R AMINET.img
```

5.2.1.1 Permissions

The permissions are as follows:

Permission	
R	File cannot be altered by the STB (for example the kernel module, application). If files with **R** permission are altered or deleted, the STB detects an error and reboots.
W	File can be altered by the STB (for example a configuration file). If these files are deleted from the STB, it will not cause an error.
E	File is read-only and can be deleted after it is executed without affecting STB operation (e.g. **upgrade.sh** script).

- Executable image component files must be listed as either **R** or **E**, otherwise the **signupgradeimage** script will fail to create the software upgrade image.
- All files in the **imagecomponents/** directory must be listed in the **flashcontents** file, otherwise, the **signupgradeimage** script will fail. If a file is listed in the **flashcontents** file but is not present in the **imagecomponents/** directory, it will not affect the **signupgradeimage** script.

5.2.2 File naming

Files names must not be longer than 12 characters. For further information on the **flashcontents** file, see the Amino *Installation and Upgrade Guide*.

5.3 Creating an upgrade script

An upgrade script **upgrade.sh** contains commands that the STB executes when a new software upgrade image is installed. The script is created and installed on the STB as part of a software image, and is only executed once, then deleted.

5.3.1 To use an upgrade script

The **upgrade.sh** script is not normally supplied as part of a software release, but you can create your own and add the commands you want the STB to execute. For example you can add **libconfig** commands to set static multicast values. Add the file to the image components used to create a software upgrade image and ensure that the script is listed in the **flashcontents** file, with its permission set to **E**.

The script is executed automatically when this software image is installed on the STB and is then deleted.

5.4 Enabling remote log-in tools

Telnet or SSH(Dropbear) can be included in the software installed on an STB (the tool available depends on the platform), in order to enable remote log-in – for example, to access configuration via **libconfig** commands.

Amino advise that you us SSH(dropbear) where possible as some content providers insist that STBs do not have Telnet support for security reasons. If you want to use Telnet please contact Amino Customer Support for assistance.

5.4.1 Remote log-in tool availability

On the Ax4x platform, the remote log-in tools (Telnet/SSH(Dropbear)) are included as part of a standard software build or software image but are disabled by default for security. For example to enable startup of the Telnet daemon at bootup, include a script in **imagecomponents**, for example, **rc.ca_app** or **rc.dmn_app**.

For example:

```
#!/bin/sh
#
# modules start / stop.

. /etc/rc.config

case "$1" in
        start)
        echo "Loading telnetd"
        telnetd &
        ;;
        stop)
        echo "Killing telnetd"
        kill -9 `pidof telnetd`
        ;;
        reload|restart)
        $0 stop && $0 start
        ;;
        *)
        echo "Usage: $0 {start|stop|reload|restart}"
        exit 1
esac
```

For information on how to install Dropbear onto the STB, see *Amino Technical Note 015 Running the Dropbear SSH Daemon on an Amino STB.*

5.5 Customising graphics

You can replace the graphic displayed during STB booting. For TFTP booting STBs, it is also possible to replace the graphic shown during software upgrades.

5.5.1 Graphics used by the STB

The following default graphics are supplied with a software release:

5.5.1.1 splash.gif

The default graphic shown on screen when the STB is booting is as shown:

5.5.1.2 loading.gif

This is the graphic shown on screen when the STB is upgrading. The **Upgrading, do not unplug!** screen is shown in multicast and TFTP upgrades, once the STB has downloaded a software upgrade image from the server.

5.5.2 Replacing the default graphics with your own

You can replace the default graphics with your own by replacing the GIF files in the image components used to build a new software upgrade image, then upgrading the software on an STB to use this new software image. When using your own graphics, please note the following points:

- Each graphic must be a GIF with the same name as the file you are replacing.
- Animated GIFs are not supported (using an animated GIF will prevent the STB from booting)
- GIFs must be less than 640 x 400 pixels. If the GIF is larger than the screen dimensions, it will not be displayed.
- Transparent GIFs are not supported, but if you set the transparent colour to `0`, the background colour will be undefined.

- GIFs with profile data are not supported. Profiles can be stripped using tools such as Graphics Magick (http://www.graphicsmagick.org/) and the command **`gm convert +profile "*" in.gif out.gif`**

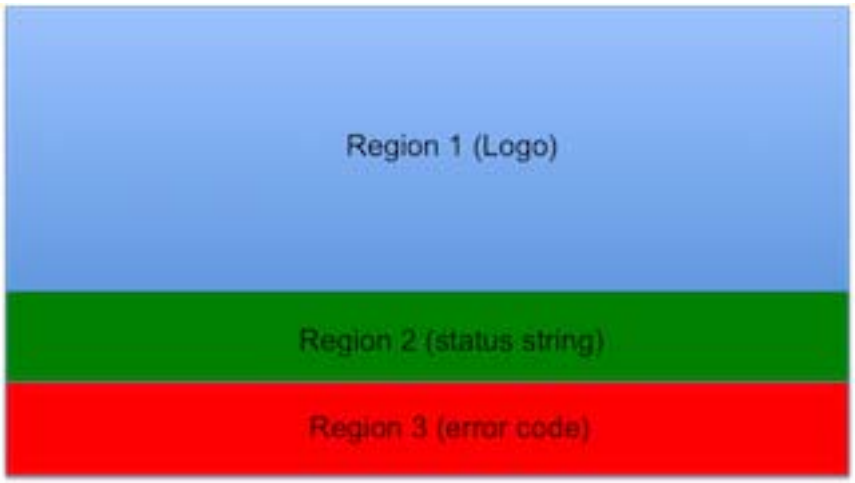
The graphic is always centred on the screen when it is displayed.

5.6 Customising upgrade splashscreens

The upgrade splashscreens can be configured to display your corporate logo or choice of graphic, and the status of the upgrade can be displayed in your corporate font and re-phrased if required. Furthermore the error codes that are represented by the flashing LED can be displayed numerically on the screen.

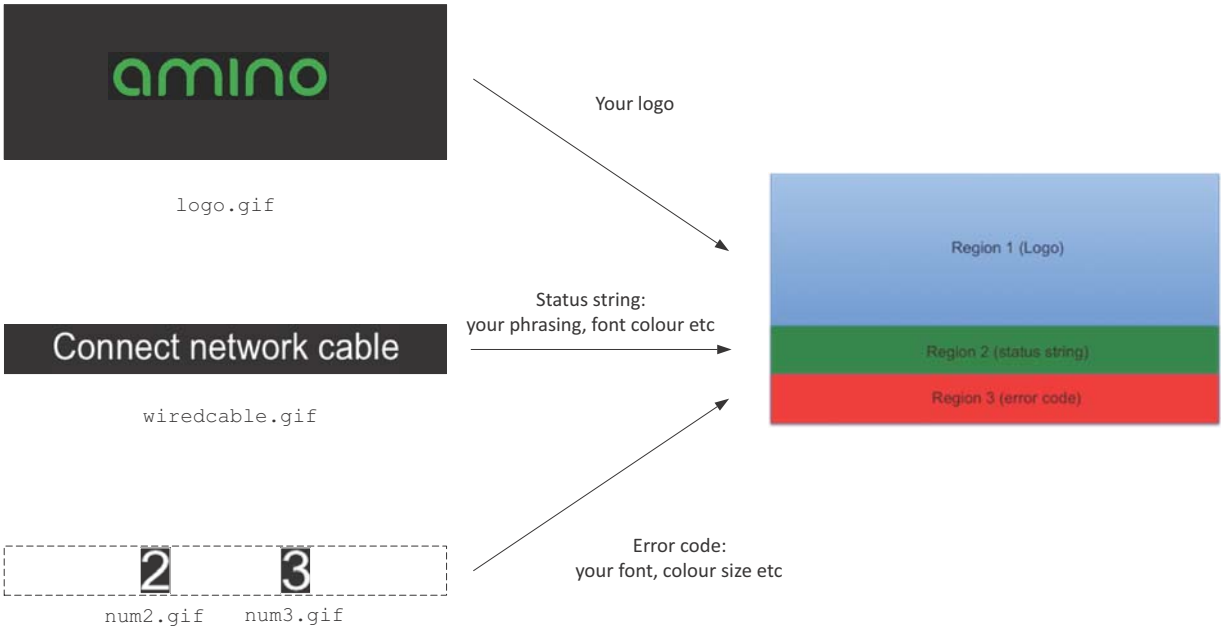
Configuration of the splashscreen uses three "regions" on the screen. Region 1 is the top 2/3 and is used for displaying the logo. The remaining 1/3 is split into two regions where region 2 displays the stage of the upgrade (for example ‘fetching data’ or ‘Installing. Do not unplug’) and region 3 is used to display the two digits of the error code.

Customising the upgrade display consists of simply replacing the graphics used for the display with your own graphics to create a display in your preferred format, font, colour etc.



The supplied graphics are located in **`upgradeimage\imagecomponents\upgrade_branding.tgz`**. Replace the graphics in the .tgz file with your own. When you use your own graphics, you **must** keep the same filenames as those given in the release because the upgrade script will look for those filenames. For example, your logo must be called `logo.gif`.

The following diagram shows how you can assemble your customised upgrade display by substituting the graphics provided in the release image with your own:


amino
logo.gif
Your logo
Region 1 (Logo)
Connect network cable
wiredcable.gif
Status string:
your phrasing, font colour etc
Region 2 (status string)
Region 3 (error code)
2
3
num2.gif
num3.gif
Error code:
your font, colour size etc

Chapter 6—STB configuration pages for Opera 11 onwards

In this Chapter:

This chapter describes how to use the Amino STB configuration pages that provide customisation options for both administrators and end-users.

The screenshots for the Preferences and Management pages shown in this chapter are examples from an H140 STB, with a standard Amino/Opera homepage. Your installation may not show exactly the same screenshots if you are using a different model of STB.

6.1 About the STB configuration pages

STBs with browsers include HTML configuration pages that enable local access and editing of configuration areas. The configuration pages consist of a user area (Preferences pages) and an administrator area (Management pages).

After booting, the STB will present the home page.

Preferences pages main menu (H140)

Management pages main menu (H140)

6.2 Getting started

The instructions in this section assume that the STB you want to communicate with is powered and connected to the network.

To use the Preferences pages, you will need the following:

- An IR remote control or Amino IR keyboard

To use the Management pages, you will need the following:

- An Amino IR keyboard (optionally also an IR remote control)

- You will need to know the password for the **mngread** and **mngwrite** user names on the STB. These are **leaves** and **snake** respectively if you have not changed the defaults.

Warning:
Amino strongly recommends that you change the default passwords when deploying our STBs.
See the Amino *Installation and Upgrade Guide* for information on changing these default passwords.

6.2.1 Using a keyboard or remote control with the configuration pages

The table below shows which remote control buttons and keyboard keys enable important functionality.

Functionality	Remote control	Keyboard
Display Preferences page main menu.	**MENU** button	**Alt**+**P** key combination or **Settings** button.
Display Management pages main menu.	-	**Alt**+**M** key combination, then password.
Move around the items on the page.	Arrow buttons	Arrow keys
Display sub menu.	**OK** button or right arrow button	**Enter** key or right arrow key
Select item.	**OK** button	**Enter** key
Delete text you have entered.	**P<P** button	**Back Space** key.
Exit configuration pages.	**MENU** button or left arrow button	Left arrow key

In general, Management pages require the use of a keyboard, whereas Preferences pages are designed for use with a remote control.

6.3 Preferences pages

The Preferences pages enable users to change basic STB configuration settings such as video and subtitle set-up. The pages are accessed locally via an IR remote control or keyboard and are designed to be used by end-users.

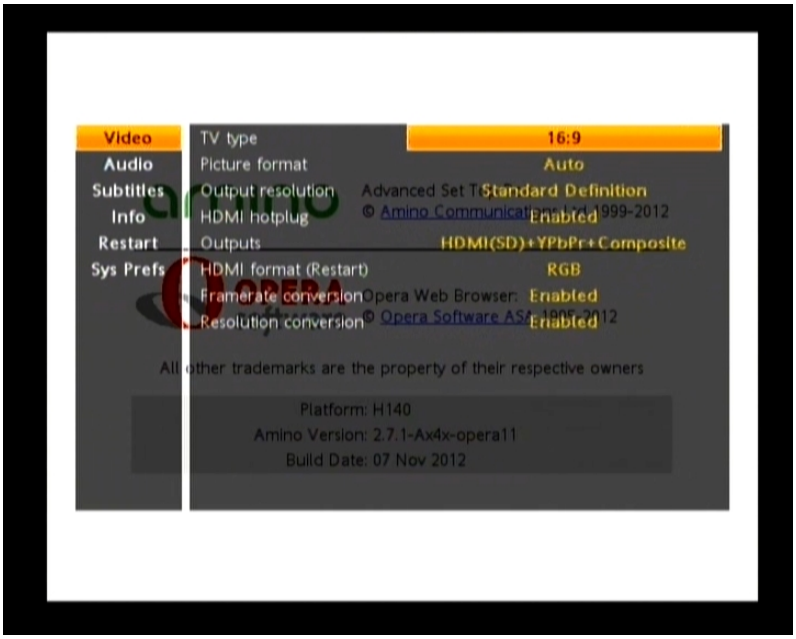
On hospitality boxes (e.g. 140H), the Preferences pages are not available to end users - they can only be accessed from the Management pages.

6.3.1 Changing the Preferences pages settings

To change configuration settings on the Preferences pages using a remote control:

1. Access the pages by pressing the **MENU** button on the remote control.

2. Use the arrow buttons to navigate to the setting you want to change.
3. Press **OK** or the right arrow button to display the secondary menu on the right hand side of the page.



4. Use the arrow buttons to navigate to the setting you want to change.
5. Press **OK** - the setting will display arrows if there is a choice of values. Navigate through these values using the left or right arrow buttons. When you have found the setting you want press **OK** again.
6. Change any other settings you require, then exit the Preferences pages.

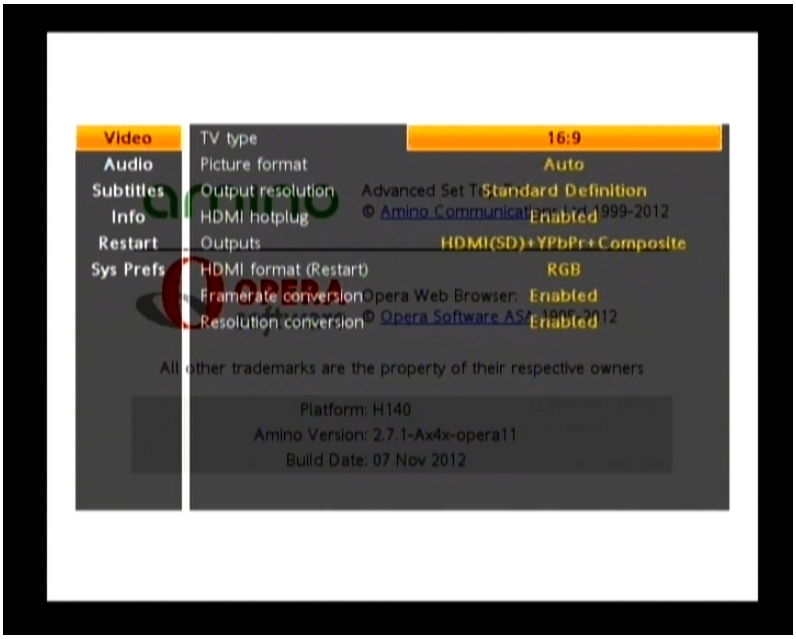
6.3.2 Using the Preferences pages

The Preferences pages are divided into several areas. When you first access the pages, the main menu is displayed. This menu includes the following items:

Page	Function:
Video	View and edit video display and output configuration.
Audio	View and edit audio language settings.
Subtitles	Enable or disable subtitles and select primary and secondary subtitle language.
RF (not available on all models)	Configure the RF output settings.
Info	View the STB serial number, MAC address and software version numbers. There are no user-configurable settings on this page.
Restart	Restart the STB.
Sys Prefs (H140)	Move to the Management pages

The sections that follow detail the main settings available. Each section relates to a page accessible from the main menu.

6.3.3 Video



TV type Sets how video is formatted for the TV. See also **DISPLAY_MODE**.

The following options are available:

4:3 and 16:9

Picture format

Sets how video is formatted for the TV. See also **DISPLAY_MODE** and Amino *Technical Note 018 - Aspect Ratio Handling*.

The options available change depending on the **TV type** setting.

With TV type set to 16:9	
Value	**Description**
Stretch (4:3 to 16:9)	Ignores the aspect ratio of the video, and assume that it is the same as the ratio of the output. So on 16:9 display, a 4:3 picture will appear horizontally distorted. On a 4:3 display, a 16:9 picture will appear vertically distorted.
Auto	For televisions which have automatic ratio switching. In this mode, the television switches between 4:3 and 16:9 depending on the video content, and full content is displayed for both.
Zoom (4:3 to 16:9)	This mode zooms 4:3 video to fill a 16:9 screen, cropping the top and bottom of the picture.

Output Resolution

Selects the output resolution, which will be one of the following options for a PAL system and will also depend on whether **HDMI hotplug** is enabled:

SD

1080p50

1080i50

720p50

576p50 (or 480p depending on whether the value of **TVSYSTEM** is PAL or NTSC).

The box must be re-booted in order for the changes to take effect. See also OUTPUT_RESOLUTION.

HDMI hotplug

Automatically switches the output between SD and HD depending on whether the HDMI cable is plugged in or not. When HDMI hotplug is enabled, the "Output Resolution" option (above) refers to the "Preferred" resolution when the HDMI cable is connected.

Outputs

Sets the output format of the STB. See also **OUTFMT.**

Ax4x builds do not require a restart.

The following table shows which video output formats are available for each type of STB:

Ax4x series STBs	
STB type	**Output format choices offered**
A129	COMPOSITE COMPOSITE + RGB COMPOSITE + S-VIDEO COMPOSITE + YPbPr
A140, H140, A540PVR	COMPOSITE S-VIDEO COMPOSITE + RGB COMPOSITE + S-VIDEO COMPOSITE + YPbPr RF
M540	COMPOSITE COMPOSITE + RGB COMPOSITE + S-VIDEO

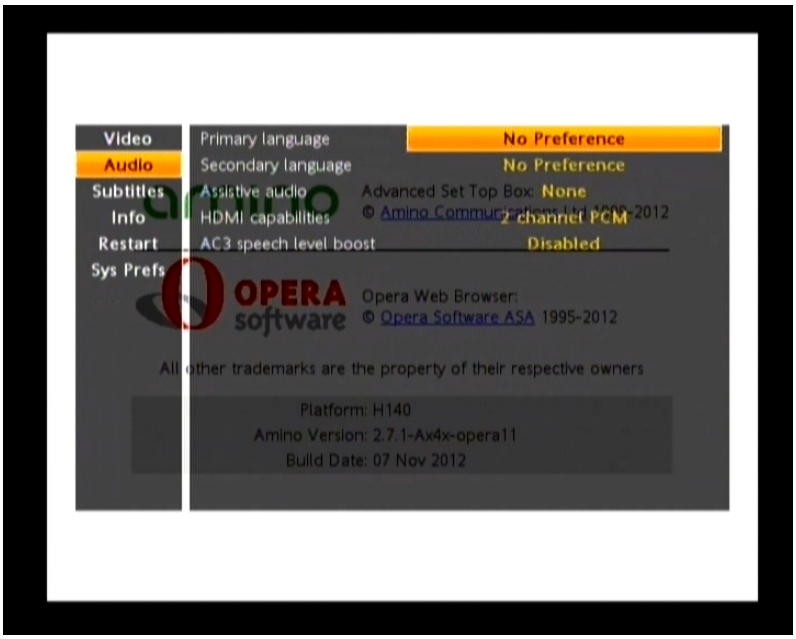
HDMI format (Restart) Sets the video format from the HDMI output: RGB, YUV444 or YUV422. See also **HDMI_VIDEO_FORMAT**.

Framerate conversion When framerate conversion is enabled, then native framerate passthrough is disabled, that is the STB will **not** automatically change its output framerate to match that of the incoming (native) video stream being played: instead the framerate will be converted.

For example, if the box is set to PAL (50Hz) and an NTSC clip is played (59Hz), the STB will attempt to play the clip at PAL (50Hz). See also **NATIVE_FRAMERATE_PASSTHRU_ENABLE**

Resolution conversion When resolution conversion is enabled, then native passthrough is disabled, that is the STB will **not** change its output resolution (HD) to SD if the incoming (native) video stream is in SD. See also **NATIVE_PASSTHRU_ENABLE.**

6.3.4 Audio



Primary language — The primary language for the audio stream. See also **PREFERRED_LANG**.

Secondary language — The secondary language for the audio stream. See also **SECONDARY_LANG**.

Assistive audio — Visually impaired: selects the audio stream labelled visually impaired (the assistance is usually a voice describing the programme scene and sound effects, supplementary to the programme dialogue).

Hearing impaired: selects the audio stream labelled hearing impaired.

None: disables the assistive audio.

HDMI capabilities — The HDMI options are: 2 channel PCM, 5.1 Dolby Digital & 7.1 Dolby Digital Plus. This should be set to match the decoding capabilities of the HDMI device that the STB is connected to. See also **HDMI_AUDIO_FORMAT**.

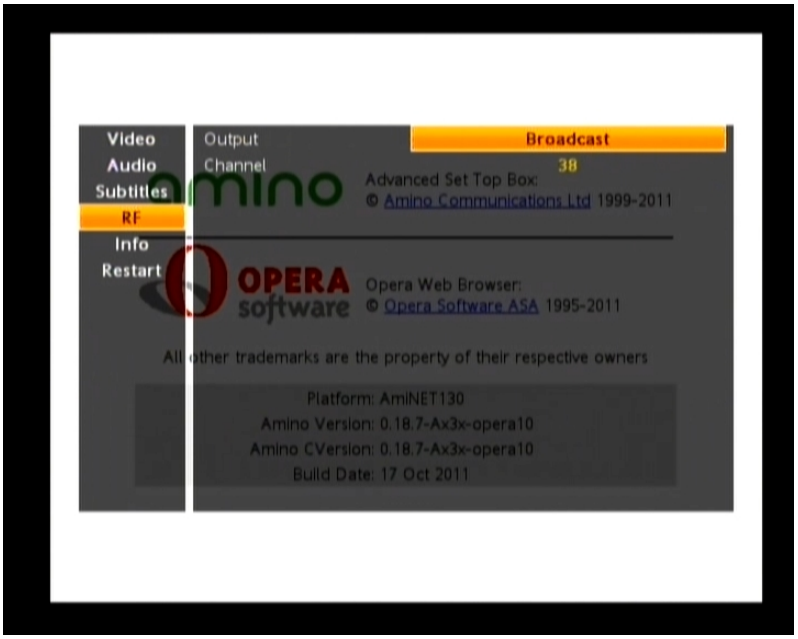
6.3.5 Subtitles

Subtitles Enables or disables subtitles.

Primary language Sets the primary language for the default subtitle display. See also **SUBTITLES_PREF_LANG.**

Secondary language Sets the secondary language for the default subtitle display. If the STB can find no subtitle information in the Primary language it uses the information in the Secondary language, See also **SUBTITLES_SECOND_LANG.**

6.3.6 RF (not available on all models)



Output Enables or disables the RF output (for STBs with RF output) and selects the type of frequency. See also **RFMOD_ENABLE** and **RFMOD_FREQTABLE**.

Value	Description
Disabled	Enables or disables the RF output
Broadcast	Configures the RF modulator to use broadcast frequencies
Cable	Configures the RF modulator to use cable frequencies

Channel Sets the RF channel for the output.See also **RFMOD_CHAN**.

6.3.7 Info

The Info page gives you the following information about the STB:

- Serial number
- MAC address
- Software version
- CR version
- OEM version
- PBL version
- RI version

6.3.8 Restart

The Restart page allows you to restart the STB.

6.3.9 Sys Prefs (H140)

The Sys Prefs page allows you to return to the Management pages.

6.4 Management pages

The Management pages enable administrators to change STB configuration areas such as networking, channel list and browser set-up, as well as initiate basic operations such as rebooting and updating software. In addition, read-only access is available to areas such as version and STB identity settings. The pages are password-protected and accessed locally via an Amino IR keyboard (limited functionality is also supported via a remote control).

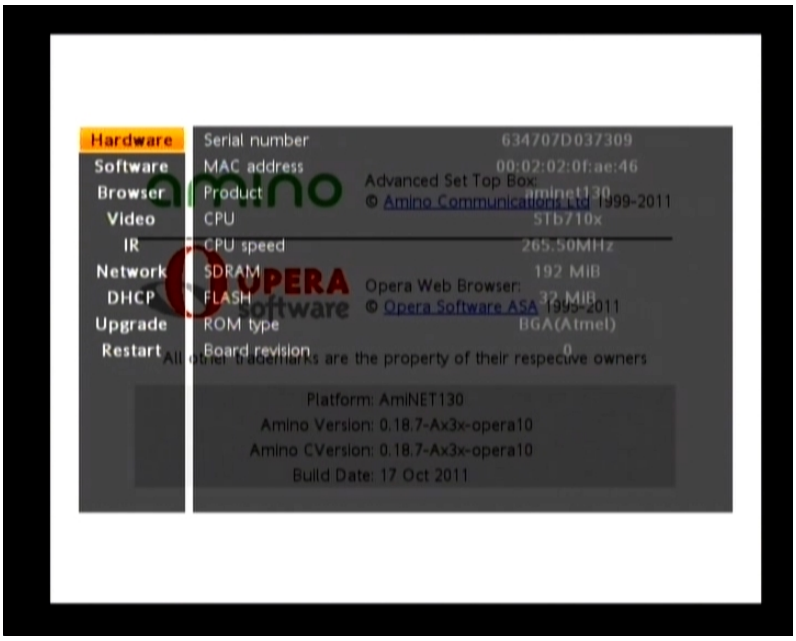
Use **Alt-M** on the keyboard to access the Management pages. You will have to enter the management password, see Getting started (the password is **leaves** if you have not changed the default). Management pages allow access to advanced configuration functions and are designed for use by administrators rather than end-users.

6.4.1 Using the Management pages

The Management pages are divided into several sections. When you first access the pages, the main menu is displayed. This menu includes the following section headings:

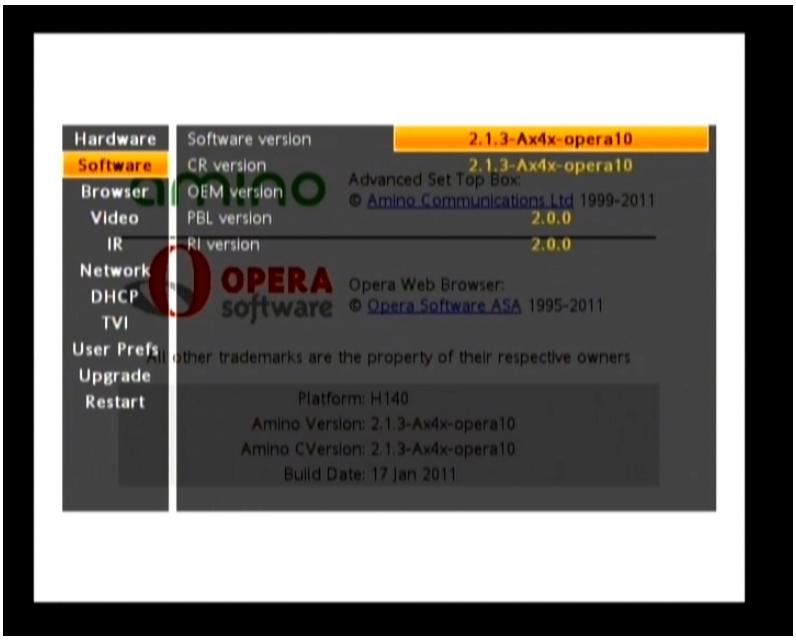
Page	Function
Hardware	View the STB hardware details. There are no configurable settings on this page.
Software	View the STB software details. There are no configurable settings on this page.
Browser	View and edit the channel URLs and graphics support information.
Video	View and edit video display and timeout settings.
IR	View and edit the IR device settings.
Network	View and edit static address information.
DHCP	View and edit the DHCP server information.
Upgrade	Upgrade the STB
Restart	Reboot the STB.
TVI (H140)	Initiate a software upgrade (multicast) or reflash.
User Prefs	Enter the Preferences pages (H140 only)

6.4.2 Hardware



Serial number	The STB serial number
MAC address	The STB MAC address
Product	The STB type.
CPU	The processor type.
CPU speed	The processor speed.
SDRAM	The size of the SDRAM.
FLASH	The size of the Flash.
ROM type	The type of ROM and manufacturer
Board revision	The board revision number.

6.4.3 Software



H140 software page: note TVI and User Prefs options

Software version The version string for the software build.

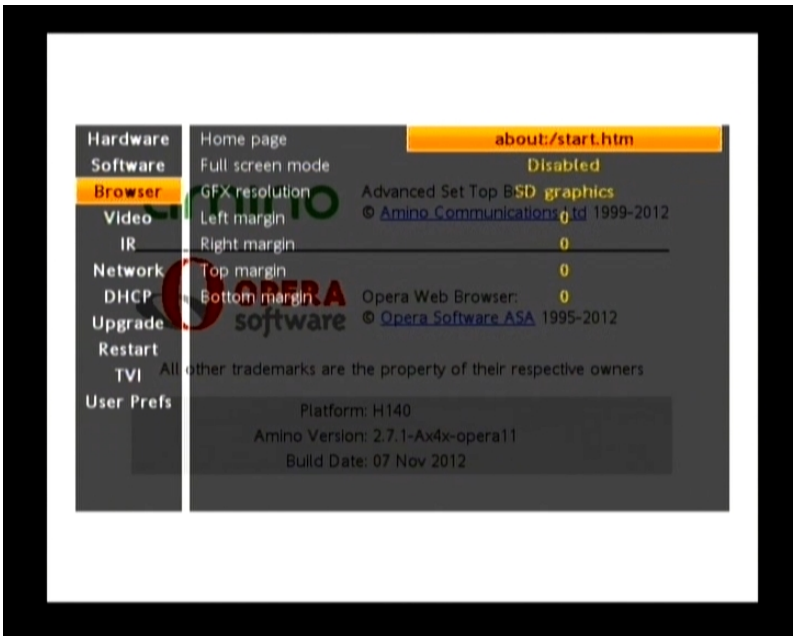
CR version The Custom Resource (CR) version for the software build.

OEM version The OEM version as defined by the system integrator in **`/mnt/nv/oemversion.`**

PBL version The Primary Boot Loader version.

RI version The Recovery Image version.

6.4.4 Browser



Home page — Sets the home page. This is the URL of the page the browser goes to when the STB has booted successfully and the Home key is pressed. The URL entered must be in the format http://<address>. See also **BROWSER_HOMEPAGE**. (The STB can go directly to this page without pressing a key if you set the appropriate permissions in the File access policy.)

Full screen mode — Enables or disables full screen mode. See also **FULLSCREEN**.

GFX resolution — Sets the graphics resolution. See also **GFX_RESOLUTION**. (When this setting is set to SD Graphics, the value of **TVSYSTEM** determines the final screen resolution.)

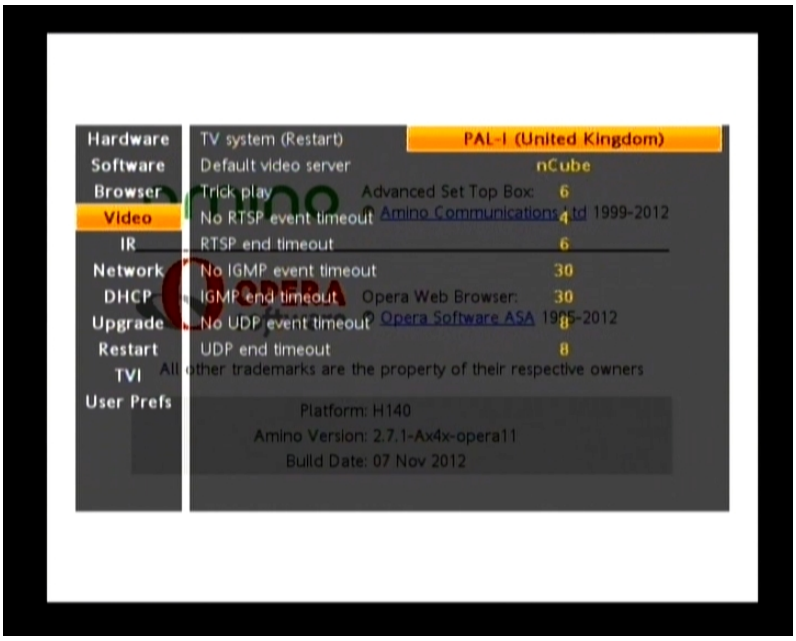
Left Margin — Sets the left browser margin, as a percentage of the graphics plane size (that is, in the range 0 - 100). Opera also allow margins to be specified in pixels. See also **BROWSER_MARGIN_LEFT**.

Right Margin — Sets the right browser margin, as a percentage of the graphics plane size (that is, in the range 0 - 100). Opera also allow margins to be specified in pixels. See also **BROWSER_MARGIN_RIGHT**.

Top Margin — Sets the top browser margin, as a percentage of the graphics plane size (that is, in the range 0 - 100). Opera also allow margins to be specified in pixels. **BROWSER_MARGIN_TOP**.

Bottom Margin — Sets the bottom browser margin, as a percentage of the graphics plane size (that is, in the range 0 - 100). Opera also allow margins to be specified in pixels. See also **BROWSER_MARGIN_BOTTOM**.

6.4.5 Video



TV system Sets the TV system

Default Video Server Sets the default video server type. It is now only necessary to do this for EONA - for other servers the STB will detect the type from the RTSP stream. The available options are:

nCube	Oracle	Sapphire	SeaChange
Tandberg Openstream	Telefonica	Anevia	BitBand
Concurrent	EONA	InfoValue	Kasenna MediaBase

See also **RTSP_SERVER**.

Trick play Sets the default fast forward and rewind **multiplier** (0-6), for example a trick play value of 2 means the fast forward and rewind speeds will be multiplied by twice normal speed. A value of 0 will pause the stream. See also **RTSP_SCALE**.

No RTSP event timeout Sets the number of seconds to wait for video data after sending an RTSP play command, before a no-video event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **RTSP_START_TIMEOUT**.

RTSP end timeout Sets the number of seconds to wait after no more RTSP data is seen, before an end-of-media event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **RTSP_END_TIMEOUT**.

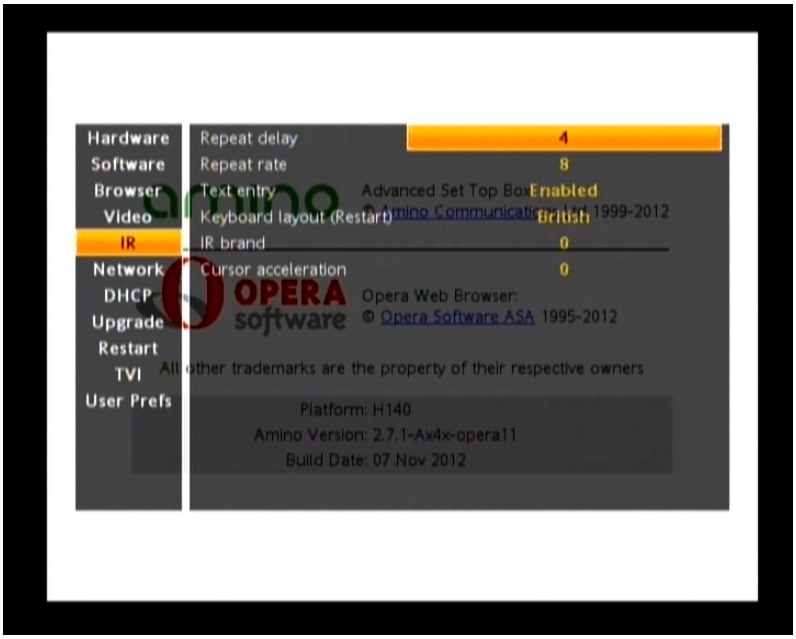
No IGMP event timeout Sets the number of seconds to wait for video data after sending an IGMP join command, before a no-video event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **IGMP_START_TIMEOUT**.

IGMP end timeout Sets the number of seconds to wait after no more video data is seen, before an end-of-media event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **IGMP_END_TIMEOUT**.

No UDP event timeout Sets the number of seconds to wait for video data after trying to connect to a UDP media stream, before a no-video event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **UDP_START_TIMEOUT**.

UDP end timeout Sets the number of seconds to wait when no more UDP data is seen, before an end-of-media event is raised. This must be in the range 0 - 30, where 0 means that there will never be a timeout raised. See also **UDP_END_TIMEOUT**.

6.4.6 IR



Repeat delay Sets the length of time that a user has to depress a key on an IR input device (for example the keyboard) to make it repeat. See also **DELAY_FACTOR.**

Value	Description
1	16cs delay rate.
2	32cs delay rate.
3	64cs delay rate.
4	96cs delay rate.

Repeat rate Sets the key repeat rate in centiseconds, to control the number of characters per second that a user can enter. Set an even-number value in the range 2 - 30 (rate in centi-seconds), or set 0 for no repeat. See also **REPEAT_RATE.**

Text entry Enables or disables remote text entry via the remote control. If text entry is enabled, the user enters text by holding down the appropriate number key. See also **TEXT_ENTRY.**

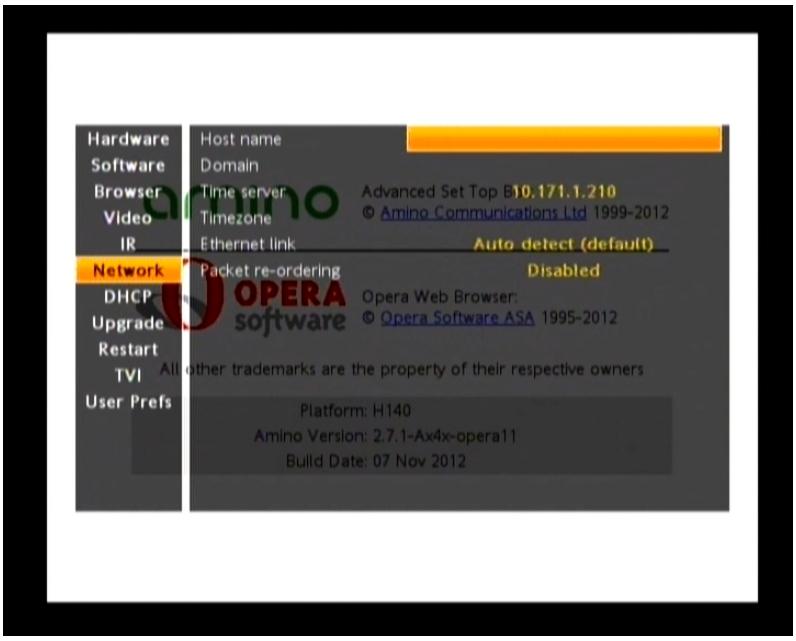
> NOTE The Repeat rate must be set to 0 (zero) and remote text entry must be enabled for remote text entry to work.

Keyboard layout Sets the keyboard layout.

IR brand Sets the brand code used to pair the STB with a particular Amino remote control. The value must be in the range 000 - 015 (or 100 –115 for the A130), and must match the code on the remote control that you want to use to with the STB. See also **IR_BRAND**.

Cursor acceleration Sets the speed of the cursor. See also CURSOR_ACCELERATION

6.4.7 Network



Hostname Sets the hostname for the STB.

Domain Sets the domain for the STB. This is only needed if the STB does not retrieve its network settings from a DHCP server. See also **DOMAIN**.

Time server The IP address of the NTP client (either static or from the DHCP server). See **TIME_SERVER**.

Timezone The time zone in which the STB is operating. See **TIME_ZONE**.

Also see http://www.twinsun.com/tz/tz-link.htm or http://en.wikipedia.org/wiki/List_of_tz_zones_by_name

Ethernet Link Sets the default speed for the main Ethernet port to one of the following values:

Auto detect (default)

10Mbit/s Half Duplex

10Mbit/s Full Duplex

100Mbit/s Half Duplex

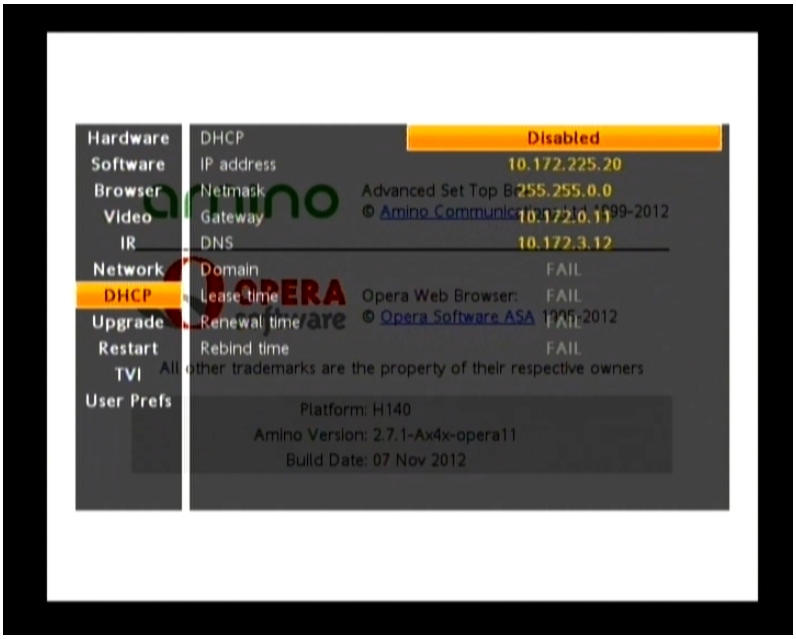
100Mbit/s Full Duplex

10BaseT Ethernet is not supported by Ax4x series STBs.

See also **ETHERNET**.

Packet re-ordering Enables or disables packet ordering. If packet ordering is enabled, the STB buffers out-of-order packets. Amino does **not** recommend that you use packet ordering as it can have an adverse effect on system performance.

6.4.8 DHCP



DHCP Sets whether the STB will be getting its network settings dynamically from the DHCP server (DHCP enabled).

If the STB will **not** be getting its network settings dynamically from the DHCP server (that is, you have selected **Disabled**), then you will need to ensure that the appropriate static settings are specified on this page (for example, IP address and netmask). See also **DHCP**.

IP Address Sets the static IP address for the STB. This is only needed if the STB does not retrieve its network settings from a DHCP server. See also **IPADDR**.

Netmask Sets the netmask. This is only needed if the set-top box does not retrieve its network settings from a DHCP server. See also **NETMASK**.

Gateway Sets the IP address for the gateway for the set-top box. This is only needed if the set-top box does not retrieve its network settings from a DHCP server. See also **GATEWAY**.

DNS Sets the IP addresses for the domain name servers for the STB. This is only needed if the set-top box does not retrieve its network settings from a DHCP server. See also **DNS**.

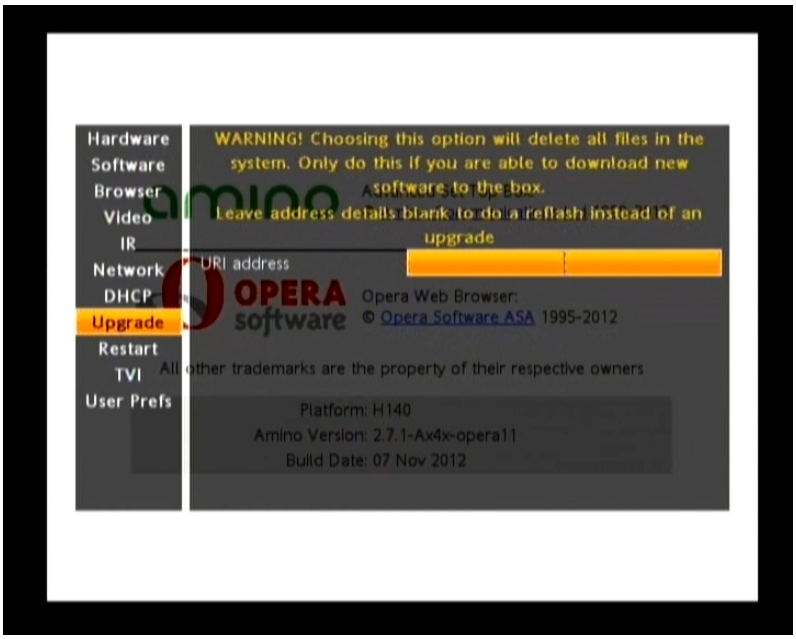
Domain Sets the domain for the STB. This is only needed if the set-top box does not retrieve its network settings from a DHCP server. See also **DOMAIN**.

Lease time The DHCP lease time as returned by the DHCP server. See also **LEASETIME**.

Renewal time The DHCP renewal time as returned by the DHCP server. See also **RENEWALTIME**.

Rebind time The DHCP rebind time as returned by the DHCP server. See also **REBINDTIME**.

6.4.9 Upgrade



Ax4x upgrade screen

The Upgrade page allows you to upgrade the software via the Management pages. However, before you initiate an upgrade, you must ensure that new software is available. This means that you will need an HTTP server containing the upgrade image for a unicast upgrade, and a multicast server containing the upgrade image for a multicast upgrade.

For information about setting up an HTTP server or multicast server, refer to the Amino *Software Installation and Upgrade Guide.*

When the STB does a **software upgrade**, it only replaces the main software image and some user configuration can be preserved. If there is a problem during the upgrade, the new software can revert to the old software.

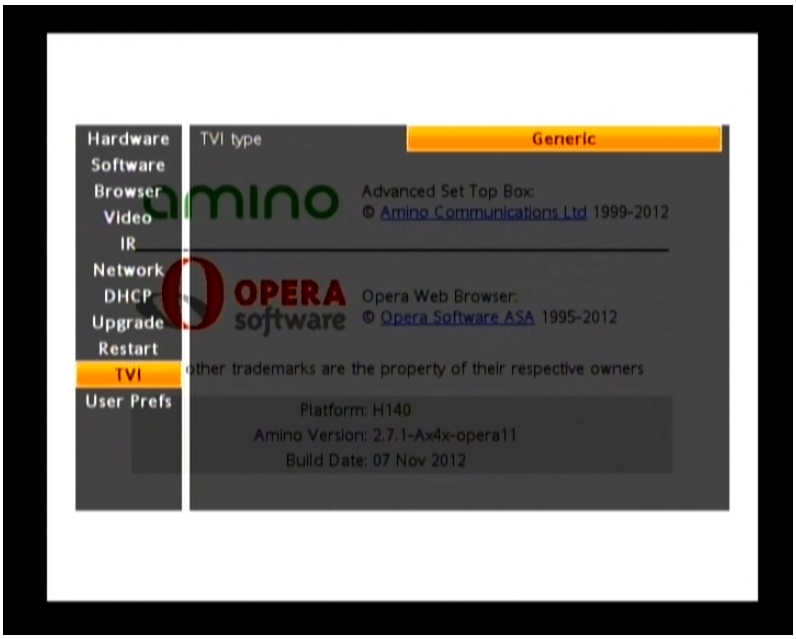
When the STB does a **reflash**, it wipes all its current software - except configuration options stored in the NOR Flash - before installing new software, so it may not be able to revert to the old software if there is a problem.

For Ax4x upgrade	For Ax4x reflash
URI address - enter the URI for the upgrade image (either IGMP for multicast or HTTP for unicast).	URI address - leave blank

6.4.10 Restart

The restart page allows you to restart the STB.

6.4.11 TVI (H140)



The H140 has an additional TVI page that allows you to configure the TVI interface. For more information about TVI refer to the Amino *TV Connectivity Guide.*

Generic The generic TVI setting - the Amino generic protocol will be used.

Samsung The Samsung TVI setting - the Samsung Serial Protocol for TVI operation will be used.

6.4.12 User Prefs

The User Prefs page allows you to move into the Preferences pages directly from the Management pages. If you have this page available, then there will be a Sys Prefs page in the Preferences pages that allows you to return directly to the Management pages.

6.5 Configuration page settings for Opera 11

The main settings needed to configure the STB can be accessed via the configuration pages, as detailed in the previous sections of this chapter. These settings are read from the configuration file or NOR Flash depending on the software build. The tables in 6.5.1 Preferences pages and 6.5.2 Management pages show the areas to which the configuration pages read and write settings for Opera 11 browsers. The format for the table is as shown:

Preferences or Management page setting	Settings file <setting> or NOR Flash <setting>

You may need this information if you want to try out settings via the configuration pages and then copy the new configuration to the image components used to create a new software image.

6.5.1 Preferences pages

Area	Settings file setting or NOR Flash setting
IR Control pages	
Repeat delay	Settings **DELAY_FACTOR**
Repeat rate	Settings **REPEAT_RATE**
Text entry	Settings **TEXT_ENTRY**
IR brand code	Settings **IR_BRAND**
A/V setup pages	
Minimum Volume	Settings **MINIMUM_VOLUME**
Maximum Volume	Settings **MAXIMUM_VOLUME**
TV type	Settings **DISPLAY_MODE**
Display mode	Settings **DISPLAY_MODE**
Output resolution	NOR Flash OUTPUT_RESOLUTION
Video output	NOR Flash **OUTFMT**
RF enable	NOR Flash **RFMOD_ENABLE**
Frequency table	NOR Flash **RFMOD_FREQTABLE**
RF Channel	NOR Flash **RFMOD_CHAN**
TV System	NOR Flash **TVSYSTEM**
HDMI audio mode	NOR Flash **HDMI_AUDIO_FORMAT**
Native Resolution Passthrough	NOR Flash **NATIVE_PASSTHRU_ENABLE**
Native Framerate Passthrough	NOR Flash **NATIVE_FRAMERATE_ PASSTHRU_ENABLE**
Languages setup pages	
Primary language	Settings **PREFERRED_LANG**

Area	Settings file setting or NOR Flash setting
Secondary language	Settings **SECONDARY_LANG**
Subtitle enable CC enable	NOR Flash **SUBTITLES**
Primary subtitle language	Settings **SUBTITLES_PREF_LANG** (or **usersettings** file if the file exists)
Secondary subtitle language	Settings **SUBTITLES_SECOND_LANG** (or **usersettings** file if the file exists)

6.5.2 Management pages

Area	Hard coded location, Settings file setting or NOR Flash setting
STB Information pages	
Product	hard coded
CPU	hard coded
CPU speed	hard coded
SDRAM	hard coded
FLASH	hard coded
ROM type	hard coded
Board Revision	hard coded
Software version	**/etc/version**
CR version	**/mnt/nv/cversion**
OEM version	**/mnt/nv/oemversion**
MAC Address	NOR Flash **MAC_ADDRESS**
Serial Number	NOR Flash **SERIAL_ID**
Browser setup pages	
Home page	Settings **BROWSER_HOMEPAGE**
Full screen mode	Settings **FULLSCREEN**
GFX resolution	Settings **GFX_RESOLUTION**
Left Margin	Settings **BROWSER_MARGIN_LEFT**
Right Margin	Settings **BROWSER_MARGIN_RIGHT**

Area	Hard coded location, Settings file setting or NOR Flash setting
Top Margin	Settings **BROWSER_MARGIN_TOP**
Bottom Margin	Settings **BROWSER_MARGIN_BOTTOM**
Video setup pages	
Default Video Server	Settings **RTSP_SERVER**
Trick play	Settings **RTSP_SCALE**
RTSP start timeout	Settings **RTSP_START_TIMEOUT**
RTSP end timeout	Settings **RTSP_END_TIMEOUT**
IGMP start timeout	Settings **IGMP_START_TIMEOUT**
IGMP end timeout	Settings **IGMP_START_TIMEOUT**
UDP start timeout	Settings **UDP_START_TIMEOUT**
UDP end timeout	Settings **UDP_END_TIMEOUT**
Network setup pages	
DHCP	NOR Flash **DHCP**
IP Address	NOR Flash **IPADDR**
Netmask	NOR Flash **NETMASK**
Gateway	NOR Flash **GATEWAY**
DNS	NOR Flash **DNS**
Hostname	hostname **HOSTNAME**
Domain	NOR Flash **DOMAIN**
Time Server	NOR Flash **TIME_SERVER**
NTP Client timeout	Settings **NTPCLIENT_TIMEOUT**

Area	Hard coded location, Settings file setting or NOR Flash setting
Time zone	NOR Flash **TIME_ZONE**
Ethernet Link	NOR Flash **ETHERNET**
IP Address	NOR Flash **IPADDR**
Netmask	NOR Flash **NETMASK**
Gateway	NOR Flash **GATEWAY**
DNS	NOR Flash **DNS**
Lease time	DHCPC **LEASETIME**
Renewal time	DHCPC **RENEWALTIME**
Rebind time	DHCPC **REBINDTIME**

Appendix A—NOR Flash settings

In this Chapter:

A.1 NOR Flash settings

The following table describes the settings available in the NOR Flash. For convenience, these are listed in alphabetical order. Note that whether these values are used or not depends on the software and hardware build.

Default values listed here are used if no value is supplied. In reality the actual default is build-specific, so may not be as specified.

The examples show how to enter the setting if you are editing the settings file directly. See the appropriate documentation for information on how to enter values using other means, for example, JMACX or `libconfig`.

Setting	Values	Description

`BOARDREV`	16-bit integer	Specifies the board type (relates to the region that the STB is designed to work in and indicates board features).
		For further information on Board revision numbers see Appendix 5, ”Board revision numbers“
		This setting is read-only.
		Default:
		None.
		Example:
		`libconfig-get NORFLASH.BOARDREV`
		`> 16`

`CUSTOMER_DATA`	< 12 characters.	A string containing no more than 12 characters, this can be anything the customer wants, its use is not prescribed. **Default:** **Example:** `libconfig-set NORFLASH.CUSTOMER_DATA ”PANIC BUTTON”`
`DHCP_USER_CLASS`	< 12 characters.	A string containing no more than 12 characters, this can be anything the customer wants, its use is not prescribed. **Default:** **Example:** `libconfig-set NORFLASH.DHCP_USER_CLASS “CLASS_1”`

Setting	Values	Description
DHCP	**Y** = Always use DHCP **N** = Do not use DHCP	Configures whether network configuration is carried out by contacting a DHCP server. Note that if **DHCP** is set to **N**, the multicast upgrade values and other settings are read from the settings listed in this table (for example, the **MULTICAST_<name>** settings). **Default:** **Example:** `libconfig-set NORFLASH.DHCP "Y"`
DHCP_ AUTHENTICATION	**None** **Tele2** **Samsung**	None - no DHCP authentication Tele2 - DHCP authentication using the Tele2 vendor string. Samsung _ DHCP authentication using the Samsung vendor string. **Default:** **None** **Example:** `libconfig-set NORFLASH.DHCP_ AUTHENTICATION "Tele2"`
DHCPTIMEOUT	Time in seconds passed as a string	Sets how long (in seconds) the DHCP timeout value should be. **Default:** **Example:** `libconfig-set NORFLASH.DHCPTIMEOUT "3600"`

DI		This contains a string representation of an integer that is increased on a per build basis to ensure that during a multicast upgrade an older build is not put onto the STB.
		Read-only.
		Default:
		0
		Example:
		`libconfig-get NORFLASH.DI`
		`> 2`

Setting	Values	Description
DNS	Valid IP address.	Specifies the IP address of the DNS server that the STB should be using. This is the value used if **DHCP** is set to **N** (or if the STB cannot retrieve an IP address from the DHCP server). **Default:** **Example:** `libconfig-set NORFLASH.DNS "123.4.5.67"`
ETHERNET	**AUTO** = auto-detect. **10HD** = 10Mbit/s half duplex. **10FD** = 10Mbit/s full duplex. **100HD** = 100Mbit/s half duplex. **100FD** = 100Mbit/s full duplex.	Sets the default speed for the main Ethernet port. **Default:** **AUTO** **Example:** `libconfig-set NORFLASH.ETHERNET "10HD"` **Note**: 10BaseT Ethernet is not supported by Ax4x series STBs.
GATEWAY	Valid IP address.	Specifies the gateway for the STB. This is the value used if **DHCP** is set to **N** (or if the STB cannot retrieve an IP address from the DHCP server). **Default:** **Example:** `libconfig-set NORFLASH.GATEWAY "123.4.5.67"`
IGMP_MAX_VER	**2** or **3**	Specifies the maximum version of the IGMP protocol to use. **Default:** **2** **Example:** `libconfig-set NORFLASH.IGMP_MAX_VER "3"`
IPADDR	Valid IP address.	Configures the static IP address of the STB. This is the value used if **DHCP** is set to **N**. **Default:** **Example:** `libconfig-set NORFLASH.IPADDR "123.45.6.7"`

Setting	Values	Description
LOCK	0 = The NOR Flash is not locked. 1 = The NOR Flash is locked.	Specifies the lock state of the NORFlash. If this is set to 1, values such as the MAC address of the STB, the serial number of the STB and some CA settings that identify an individual STB cannot be changed. Read-only. **Default:** 0 **Example:** **libconfig-get NORFLASH.LOCK** **>1**
MAC_ADDRESS		On Ax4x STBs, the **MAC ADDRESS** is read-only. For Ax3x STBs, the **MAC ADDRESS** sets the unique MAC address of the STB, using six 8-bit values separated by colons, e.g. **aa:bb:cc:dd:ee:ff.** If the MAC address is changed, then the STB needs to be rebooted for the new value to take effect. Amino recommends that you do not change this configuration. **Default:** **""** **Example:** **libconfig-get NORFLASH.MAC_ADDRESS** **>00:02:02:ff:fc:64**
NETMASK	Valid IP address.	Specifies the static netmask for the STB. This is the value used if **DHCP** is set to N (or if the STB cannot retrieve an IP address from the DHCP server). **Default:** **Example:** **libconfig-set NORFLASH.NETMASK "255.255.0.0"**

Setting	Values	Description

OUTPUT_RESOLUTION	**HDNONE** = Output resolution is defined by **TVSYSTEM**.
	HD480P59 = 480p@59.94Hz.
	HD576P50 = 576p@50Hz
	HD720P50 = 720p@50Hz
	HD720P59 = 720p@59.94Hz
	HD1080I50 = 1080i@50Hz
	HD1080I59 = 1080i@59.94Hz
	HD1080P50 = 1080p@50Hz
	HD1080P59 = 1080p@59.94Hz
	HDAUTO = for x4x series STBs this allows switching between SD and HD depending on whether an HDMI cable is connected or not.
	HDAUTO = for x5x and x0xx series STBs this reads the EDID of the TV (via HDMI) and configures the STB to match the native resolution of the TV.
		Used to set the output video resolution.
		HDNONE means that the HD-capable outputs are set to the **TVSYSTEM** defined mode.
		Any analogue SD output is always defined by **TVSYSTEM**, regardless of the value of **OUTPUT_RESOLUTION**.
		Also see Primary and Secondary video outputs for further information.
		Default:
		`720p`
		Example:
		`libconfig-set NORFLASH.OUTPUT_ RESOLUTION "HD576P50"`
		NOTE If the EDID data provides a resolution that is not supported by the TV, then the closest supported resolution will be used instead.

Setting	Values	Description
OUTFMT	**CVBS-RGBOFF** Composite **CVBS-RGBON** Composite + RGB **LC-RGBOFF** S-video **LC-RGBON** S-video + RGB **CVBS-LC** Composite + S-video **CVBS-LBR** Composite + Component **LBR-CVBS-LC** Component + Composite + S-video **RGB-CVBS-LC** Composite + S-video + RGB	Defines the output format of any standard definition output from the STB The output formats available depend on the STB and video cable(s) used. **Default:** **Example:** `libconfig-set NORFLASH.OUTFMT "CVBS-RGBOFF"` See Appendix G, "Video output formats" and Overview of command options for further details.

PPP_PASSWORD	A maximum of 32 characters.	Configures the password to use in PPP builds.
		PPP builds only.
		Default:
		`""`
		Example:
		`libconfig-set NORFLASH.PPP_PASSWORD "MyPassword"`

PPP_USERNAME	A maximum of 64 characters.	Configures the user name to use in PPP builds.
		PPP builds only.
		Default:
		`""`
		Example:
		`libconfig-set NORFLASH.PPP_USERNAME "MyUserName"`

RFMOD_CHAN	Integer less than **128**.	Sets the RF channel for boards that have an RF modulator.
		Default:
		`""`
		Example:
		`libconfig-set NORFLASH.RFMOD_CHAN "3"`

Setting	Values	Description
RFMOD_ENABLE	**Y** = Enable the RF tuner. **N** = Disable the RF tuner.	Enables or disables the RF modulator for boards that have one. **Default:** **Example:** `libconfig-set NORFLASH.RFMOD_ENABLE "Y"`
RFMOD_FREQTABLE	**BROADCAST** **CABLE**	Configures whether the RF modulator uses cable or broadcast frequencies. **Default:** **Example:** `libconfig-set NORFLASH.RFMOD_ FREQTABLE "CABLE"`
SERIAL_ID		The **SERIAL_ID** is read-only on Ax4x STBs. For Ax3x STBs, **SERIAL_ID** allows configuration of the unique serial ID of the STB. Amino recommends that you do not change this configuration. **Default:** The serial number of the box is the usual value. **Example:** `libconfig-get NORFLASH.SERIAL_ID >C01008F000276`
SOFTWARE_URI		The URI for an image when recovering the STB if there is no DHCP response. **Default:** **Example:** `libconfig-set NORFLASH.SOFTWARE_URI <http_URL>`

NOTE

The **NORFLASH.SOFTWARE_URI** setting is overidden by the **DHCP** setting where shown in this table.

DHCP	SOFTWARE URI	Result
Y	Y	DHCP used
Y	N	DHCP used
N	Y	SOFTWARE URI used
N	N	Further DHCP requests attempted

Setting	Values	Description
SUBTITLES	**Y** = Display subtitles/closed captions **N** = Do not display subtitles/closed captions	Enable or disable the subtitle/closed caption display when playing video that contains subtitles/closed captions. **Default:** **N** **Example:** **libconfig-set NORFLASH.SUBTITLES "Y"**
TIME_SERVER	Valid IPv4 address.	Specifies the location of the time server for the STB to use. This is the value used if **DHCP** is set to **N** (or if the STB cannot retrieve an IP address from the DHCP server). The time server address must be in dot-quad format (234.56.78.9). **Default:** **Example:** **libconfig-set NORFLASH.TIME_SERVER "234.56.78.9"**
TIME_ZONE (deprecated - you can set the Amino vendor option 16 "AMINO.timezone" to a timezone string in the DHCP server. See the Amino *Software Installation and Upgrade Guide* for information about DHCP options.		A string in the format sss+n or sss-n. This implementation inverts the sign, so for a time zone of GMT plus 1 hour the string would be GMT-1. The time zone can also be set to a value from the IANA Time Zone Database. See http://en.wikipedia.org/wiki/List_of_tz_database_time_zones **Default:** **Example:** **libconfig-set NORFLASH.TIME_ZONE "GMT+1"** Time zone is GMT minus one hour (that is, if GMT=12:00, then TZ = 11:00)
TVSYSTEM	**PAL-B** = Australia, Ghana. **PAL-G** = Europe. **PAL-H** = Belgium. **PAL-I** = UK. **PAL-M** = Brazil. **PAL-N** = South America. **PAL-CN** = Argentina. **NTSC-J** = Japan. **NTSC-M** = US.	Sets the TV system that the STB will operate in. **Note:** After the value has been altered the STB will need to be rebooted for the change to take effect. **Default:** **Example:** **libconfig-set NORFLASH.TVSYSTEM "PAL-B"**

Setting	Values	Description
WIRELESS_REGION	The Wifi region identification string.	Specifies the WiFi region as listed in *Amino Technical Note 052 WiFi Support.* The Wifi region identification strings for particular geographical regions are listed in Technical Note 052. **Default:** **Example:** `libconfig-set NORFLASH.WIRELESS_REGION "GR-A09"`
WIRELESS_SSID		Specifies the SSID of a configured wireless access point (WAP). **Default:** **Example:** `libconfig-set NORFLASH.WIRELESS_SSID "AMINO_AP_TEST"`
WIRELESS_ PASSPHRASE		A passphrase that allows access to a WAP. **Default:** **Example:** `libconfig-set NORFLASH.WIRELESS_ PASSPHRASE "My_passphrase"`
WIRELESS_SECURITY _MODE	OPEN WEP WPA-PSK-TKIP WPA-PSK-AES WPA2-PSK-TKIP WPA2-PSK-AES WPA-ENT-TKIP WPA-ENT-AES WPA2-ENT-TKIP WPA2-ENT-AES	Sets the security mode for a configured WAP. **Default:** **Example:** `libconfig-set NORFLASH.WIRELESS_ SECURITY_MODE "WPA-PSK-TKIP"`

A.2 Primary and Secondary video outputs

x4x STBs have a primary and a secondary video output. The primary video output can render both the video and GFX planes, whilst the secondary video output only renders the video plane.

On x0xx STBs, the GFX plane is present on both outputs.

A.2.1 OUTPUT_RESOLUTION

NORFLASH.OUTPUT_RESOLUTION controls the primary output. If this is set to an HD resolution (720p, 1080i and so on) then the primary interface will feed the HDMI and/or HD-YUV outputs only. Only if it is set to **HDNONE**, or left undefined will the primary interface feed the HDMI, CVBS, S-video, YUV/RGB outputs at SD resolution.

The secondary interface is used if the primary interface is feeding HD outputs. In this case, CVBS and S-video will be fed from the secondary interface.

In summary:I

Output Resolution setting	**Primary Interface feeds:**	**Primary Interface Resolution**	**Secondary Interface (SD only)**
HD (720p, 1080i and so on)	HDMI HD-YUV	HD	CVBS S-video
HDNONE	HDMI CVBS S-video YUV/RGB	SD	-

Appendix B—Settings file contents

In this Chapter:

The following table describes the settings available in the **settings** file. Settings are described in alphabetical order. Note that whether these values are used or not depends on the software and hardware build.

Default values listed here are used if no value is supplied, but again default values depend on the software and hardware build so may not be exactly as listed here.

The examples show how to enter the setting if you are using a command line interface.

For information on how to enter values using other methods, see

Chapter 6, ”STB configuration pages for Opera 11 onwards“

Appendix D, ”Using the configuration techniques“

B.1 Available settings

Setting	Allowed values	Description
ALT_TIME_SERVERS	A string that holds a comma separated list of FQDNs (ntp.domain.com) or dot quad formatted addresses (up to a total of four). The list can contain a mix of FQDNs and dot quad addresses.	In deployments where a DHCP server is not present, this setting defines alternative NTP servers if the one stored in **TIME_SERVER** does not respond. Up to 4 server addresses can be added for static configuration. The IP addresses can be IPv6 addresses on systems that support IPv6. **Default:** **Example:** `libconfig-set SETTINGS.ALT_TIME_SERVERS “10.172.0.3,,,”`
ANY_KEY_STANDBY_RECOVER	**Y** = Any key can be used to bring the STB out of standby. **N** = Only the power button can be used to bring the STB out of standby.	Sets which remote control keys can bring the STB out of standby. Use either **Y**, **N** or **NULL**. If the value has not been set, use the default value, **N**. **Default:** **N** **Example:** `libconfig-set SETTINGS.ANY_KEY_STANDBY_RECOVER “Y”`
AUDIO_DESCRIPTION_STREAM_TYPES		A comma separated string of values representing the audio descriptor PID of the allowed streams. **Default:** **Example:** `libconfig-set SETTINGS.AUDIO_DESCRIPTION_STREAM_TYPES “0x0C0”`

Setting	Allowed values	Description
AVSYNC_WAIT	**Y** = Enable wait. **N** = Disable wait.	Configures whether the channel changes wait for AV sync before displaying. The value can be overridden by settings in the URL. If the value has not been set, use the default value. **Default:** **N** **Example:** **libconfig-set SETTINGS.AVSYNC_ WAIT "Y"**
BROWSER_ALLOC_EXTRA		Controls the amount of extra allocation space (in MiB) on top of the cache to use in a browser. **Default:** **18** **Example:** **libconfig-set SETTINGS. BROWSER_ALLOC_EXTRA "10"**
BROWSER_CACHE_DOCUMENT		Controls the amount of RAM (in MiB) to be allocated for document caching in a browser. **Default:** **4** **Example:** **libconfig-set SETTINGS. BROWSER_CACHE_DOCUMENT "2"**
BROWSER_CACHE_IMAGES		Controls the amount of RAM (in MiB) to be allocated for image caching in a browser. **Default:** **20** **Example:** **libconfig-set SETTINGS.BROWSER_ CACHE_IMAGES "10"**
BROWSER_CACHE_RAM		Controls the amount of RAM (in MiB) to be allocated for RAM caching in a browser. **Default:** **6** **Example:** **libconfig-set SETTINGS.BROWSER_ CACHE_RAM "1"**

Setting	Allowed values	Description
BROWSER_ENABLECOOKIES	**Y** = Enable cookies **N** = Disable cookies.	Enables or disables cookies. This setting cannot be accessed via **libconfig**. See also **BROWSER_PRESERVECOOKIES**. **Default:** **Y** **Example:** **libconfig-set SETTINGS.BROWSER_ENABLECOOKIES "N"**
BROWSER_HEAP_EXTRA		Controls the amount of extra heap space (in MiB) on top of the allocation to use in a browser. **Default:** **10** **Example:** **libconfig-set SETTINGS.BROWSER_HEAP_EXTRA "8"**
BROWSER_HELPPAGE	URL	Sets the help page. This is the URL for the page that the browser goes to when the **Help** key is pressed. **Default:** **Example:** **libconfig-set SETTINGS.BROWSER_HELPPAGE "http://www.aminocom.com/help"**
BROWSER_HOMEPAGE	URL	Sets the home page. This is the URL for the page that the browser goes to when the **Home** key is pressed and when the STB has booted successfully. **Default:** **about:/start.htm** **Example:** **libconfig-set SETTINGS.BROWSER_HOMEPAGE "http://www.aminocom.com"**

Setting	Allowed values	Description
BROWSER_MARGIN_LEFT **BROWSER_MARGIN_RIGHT** **BROWSER_MARGIN_TOP** **BROWSER_MARGIN_BOTTOM**	**0 - 100**	Sets the browser margins as a percentage (**%**) of the graphics plane size. Must be defined as percentage and will not work without being defined. Valid values are 0 to 100**%**. Setting these values to 100% will set the margins to cover the entire graphics plane. **Default:** **Example:** **libconfig-set SETTINGS.BROWSER_ MARGIN_LEFT "20"**
BROWSER_PRESERVECOOKIES	**Y** = preserve cookies over a reboot **N** = do not preserve cookies over a reboot **NULL** = disabled	Allows the browser to run the **rc.cookies** script so that cookies are persistent over a reboot of the STB. **Default:** **NULL** **Example:** **libconfig-set SETTINGS.BROWSER_ PRESERVECOOKIES "Y"**
BROWSER_UNLOAD_VIDEO	**Y** = Always stop video. **N** = Never stop video.	Sets the default for whether video is stopped when the page changes. The setting can be overwritten on a per page basis by the HTML page. This setting cannot be accessed via **libconfig**. **Default:** **Y** **Example:** **libconfig-set SETTINGS.BROWSER_ UNLOAD_VIDEO "N"**
CAPTIONING_GFX_ RESOLUTION	**SD** = standard definition (this depends on the **TVSYSTEM** setting). **HD720** equivalent to 1280 x 720 resolution. **HD1080** equivalent to 1920 x 1080 resolution.	Allows the desired captioning or subtitle GFX resolution to be configured independently from the main GFX resolution. **Default:** **SD** **Example:** **libconfig-set SETTINGS.CAPTIONING_ GFX_RESOLUTION "HD720"**

Setting	Allowed values	Description
CAPTIONING_WINDOW_LEVEL	**top** **middle** **bottom**	When initialising the window for subtitles or closed captions the window position can be specified in the z-plane. **Default:** **bottom** **Example:** **libconfig-set SETTINGS.CAPTIONING_WINDOW_LEVEL "top"**
CC_DIGITAL_SERVICE	**0** = Disable 708 digital service. **1** = Default value (same as not set) value. **2** to **63** = Range of valid custom service numbers.	Sets whether EIA-608B captions are used in preference to EIA-708 captions when both are present in a stream. This setting avoids disruption to a service if stream configuration issues result in invalid EIA-708 closed caption data (that is, the 708 packets are available but do not contain usable captions). **Default:** **1** **Example:** **libconfig-set SETTINGS.CC_DIGITAL_SERVICE "0"**
CC_FORCE_SCROLL_ON_CR	**Y** = Enable CC scrolling when <CR> detected **N** = Disable CC scrolling when <CR> detected	Controls whether CC text lines will be scrolled when detecting <CR> characters in the data. **Default:** **N** **Example:** **libconfig-set SETTINGS.CC_FORCE_SCROLL_ON_CR "Y"**
CC_PREFERENCE	**ASTC** = The closed captions will use ASTC encapsulation. **SCTE** = The closed captions will use SCTE encapsulation.	Sets which closed captions encapsulation should be used. If left blank, closed captions will use the ASTC encapsulation **Default**: **ASTC** **Example**: **libconfig-set SETTINGS.CC_ PREFERENCE "ASTC"**

Setting	Allowed values	Description
CURSOR_ACCELERATION	**-5** to **+5**	Increases the sensitivity of the mouse cursor (the pointer speed). Making a change to this setting will require the STB to be rebooted before the change takes effect. This setting is used with Opera 11 only and has no effect with STBs using Opera 12 (x5x and x0xx series). **Default:** **0** **Example:** **libconfig-set SETTINGS.CURSOR_ACCELERATION "2"**
DEFAULT_VOLUME	**0** to **100**	Sets the default volume that the STB uses when it boots. This must be between **MINIMUM_VOLUME** and **MAXIMUM_VOLUME**, that is, 0 – 100. You are recommended not to change the value from the default. **Default:** **100** **Example:** **libconfig-set SETTINGS.DEFAULT_VOLUME "80"**
DELAY_FACTOR	**1** = 16cs delay. **2** = 32cs delay. **3** = 64cs delay. **4** = 96cs delay.	Sets the length of time that a user has to depress a key on an IR input device (for example, the keyboard) to make it repeat. **Default:** **2** **Example:** **libconfig-set SETTINGS.DELAY_FACTOR "1"**
DISABLE_CAPTIONING_IN_PIG	**Y** - disable captioning in PIG **N** - enable captioning in PIG **SCALE** - enable scaling of captions	Disables or enables captioning in PIG. Selecting **SCALE** will automatically scale the captions to the size of the PIG. Default: **SCALE** **Example:** **libconfig-set SETTINGS.DISABLE_CAPTIONING_IN_PIG "Y"**

Setting	Allowed values	Description
DISPLAY_MODE		Configures how the video is formatted for the TV.
	panscan 4:3 TV; 16:9 content displayed in pan and scan presentation mode.	Select **panscan** for televisions which **do not** have automatic aspect ratio switching, and where you want the video picture to fill the full 4:3 screen. In this mode, 4:3 content fits the screen correctly, and any 16:9 video content is cropped on the left and right sides.
	letterbox 4:3 TV; 16:9 content displayed in letterbox presentation mode.	Select **letterbox** for televisions which **do not** have automatic aspect ratio switching, and where you want to display the full 16:9 content. In this mode, 4:3 content fits the screen correctly, and any 16:9 content is displayed in full, with black bars above and below it ('letterbox' style).
	widescreen 16:9 TV; aspect ratio of content signalled on SD outputs, 4:3 content displayed in pillarbox presentation mode on HD outputs.	Select **widescreen** for televisions which have automatic ratio switching. In this mode, the television switches between 4:3 and 16:9 depending on the video content, and full content is displayed for both.
	ignore 16:9 TV; all content stretched to fill the screen.	Select **ignore** to ignore to ignore the aspect ratio of the video and stretch all video to the equivalent of 16:9 ratio. A 4:3 ratio picture will appear distorted.
	widezoom 16:9 TV; 4:3 content displayed in zoom presentation mode.	Select **widezoom** to zoom in on 4:3 content so that it fills the width of a 16:9 screen. The top and bottom of the picture will be lost. This option is only available in software releases from 2.6.2 onwards. For more information on aspect ratios refer to Amino *Technical Note 018 Aspect Ratio Handling* and the Amino *TV Standards Guide*. **Default:** **Example:** `libconfig-set SETTINGS.DISPLAY_MODE "widescreen"`

Setting	Allowed values	Description
DOLBY_RF	**0** = Line mode. **1** = RF mode.	Sets the Dolby dynamic range control. Line mode is light compressions and RF is heavy compression. **Default:** **0** **Example:** **libconfig-set SETTINGS.DOLBY_RF "1"**
DO_PMT_CHECK	**Y** = check program numbers in PMT and PAT are the same. **N** or no value = do not perform additional check (keep legacy behaviour).	Enables additional stream checking to ensure the PMT program_number is the same as the PAT program_number. This allows non-compliant customer streams to work. **Default:** **N** **Example:** **libconfig-set SETTINGS.DO_PMT_CHECK "Y"**
DSCP_GENERIC	0 to 63	The DSCP value to use for general outgoing IP traffic. **Default:** **Example:** **libconfig-set SETTINGS.DSCP_GENERIC "26"**
DSCP_MIDDLEWARE	0 to 63	The DSCP value to use for outgoing traffic to the specific middleware addresses specified by SETTINGS.DSCP_MWAR_ADDRS **Default:** **Example:** **libconfig-set SETTINGS.DSCP_MIDDLEWARE "26"**
DSCP_MULTICAST	0 to 63	The DSCP value to use for multicast IP traffic. **Default:** **Example:** **libconfig-set SETTINGS.DSCP_MULTICAST "26"**

Setting	Allowed values	Description
DSCP_MWARE_ADDRS	A string containing a comma separated list of IP address[/mask] where the **SETTINGS.DSCP_MIDDLEWARE** value should be applied	The domains where the DSCP_MIDDLEWARE value should be applied. **Default:** **Example:** `libconfig_set LIBCONFIG_DSCP_MWARE_ADDRS, "10.172.243.0/24,192.168.0.2,192.168.1.5"`
DSCP_VOD	0 to 63	The DSCP value to use for RTSP related outgoing IP traffic. **Default:** **Example:** `libconfig-set SETTINGS.DSCP_VOD "26"`
Dual network interfaces		
ENABLED_NETWORK_INTERFACES	For details of the input values and their definitions, refer to *Amino Technical Note 064: Dual network interfaces.*	Configures multiple concurrent network interfaces.
DNS_INTERFACES		Specifies the network interfaces used for DNS support.
IGMP_ROUTING_INTERFACES		Specifies the network interfaces used for IGMP routing support.
DEFAULT_ROUTE_INTERFACES		Specifies the network interfaces used for default IP routing support.
NTP_INTERFACES		Specifies the network interfaces used for NTP support.
ENABLE_EXTERNAL_RECEIVER	1 enable 0 disable	Enables or disables the external infra-red (IR) receiver. This command controls pin 1 on the DE9 TVI connector. **Default:** 0 **Example:** `libconfig-set SETTINGS.ENABLE_EXTERNAL_RECEIVER "1"`

Setting	Allowed values	Description
ENABLE_SUBS_OR_CC	**NONE** = if TVSYSTEM is NTSC then softCC will run, dvbttx will not. If TVSYSTEM is PAL then dvbttx will run, softCC will not. **SOFTCC** = only run softCC regardless of TVSYSTEM setting. **DVBTTX** = only run dvbttx regardless of TVSYSTEM setting. **BOTH** = run both softCC and dvbttx regardless of TVSYSTEM setting.	Sets whether closed captions or subtitles are set, depending on the value selected and the television system type. **Default:** **NONE** **Example:** **libconfig-set SETTINGS.ENABLE SUBS_OR_CC "BOTH"**
	 If you want to display SCTE-27 subtitles, you must have **ENABLE_SUBS_OR_CC** set to **BOTH**. Also .srt and .sub subtitle support on NTSC-M systems is not enabled by default. To enable it you must have **ENABLE_SUBS_OR_CC to BOTH.**	
ENABLE_TRUSTED_DOMAINS	**Y** = Any page that uses JMACX and / or Macrovision calls must be checked against the list of known trusted hosts. **N** = Execute JMACX and/or Macrovision calls to any web page.	Enable or disable the use of trusted domains. If the use of trusted domains is enabled, pages that use JavaScript or Macrovision are checked against the list of trusted hosts. If trusted domains are disabled, then JavaScript and Macrovision calls may be executed from any web page. **Default:** **N** **Example:** **libconfig-set SETTINGS.**ENABLE_ TRUSTED_DOMAINS "Y"
EXTRA_STREAM_BUFFERING	0 to disable or a value in Mibs	Used to specify an amount of additional buffering allocated for use with stream transfers when HTTP OTT streaming. The buffer size is specified in MiB. **Default:** 0 **Example:** **libconfig-SETTINGS.EXTRA_STREAM_ BUFFERING "10"**

Setting	Allowed values	Description
FULLSCREEN **Note:** Amino recommends setting this to 'Y' and controlling the size of the browser window by using the **BROWSER_MARGIN** settings.	**Y** = Full screen. **N** = Not full screen.	Enables or disables full screen mode. In full screen mode, the browser window is set to 100% of the video plane. In non full screen mode, the browser window is set to 90% (safe area), so that the entire window will be visible on all televisions. **Default:** **N** **Example:** **libconfig-set SETTINGS.FULLSCREEN "Y"**
GFX_MEM_SIZE Not used in v3.4.0 (Ax0xx). Used in v4.0.0 (Ax5x)		Specifies the amount of memory in MB that should be used for GFX processing. This setting will depend on the resolution selected by **GFX_RESOLUTION**, and the **LEGACY_SUBS** setting. The **GFX_MEM_SIZE** limit is 320MB. **Default:** **24MB (SD)** **48MB (HD)** **Example:** **libconfig-set SETTINGS.GFX_MEM_SIZE "36"**

Setting	Allowed values	Description
GFX_RESOLUTION	**HD720** = equivalent to 1280x720. HD1080 = equivalent to 1920 x 1080 resolution. **NULL** = standard definition	Defines the resolution at which the GFX plane **in the browser** is rendered: x pixels wide by y pixels high. Any value other than those shown will be treated as standard definition. Note on x4x STBs, even when set to SD resolution, if an HD output is being used no graphics will be seen on the SD outputs. This is because the secondary interface is feeding the SD outputs, and this interface does not render the GFX plane. On x0xx STBs, the GFX plane is present on both outputs. The range of supported values will be platform-dependent and thus may be affected by the current **NORFLASH.**OUTPUT_RESOLUTION setting. Note that at higher resolutions, you may find some performance reduction. **Default:** **NULL** **Example:** **libconfig-set SETTINGS.GFX_ RESOLUTION "HD1080"**
	 The actual SD resolution will be either 720 x480 or 720 x576 based on the value of **NORFLASH.TVSYSTEM**. It is good practice to set **GFX_RESOLUTION** to **"NULL"** to achieve a standard definition graphics plane.	
GFX_ROTATION Opera 11 onwards	0 90 180 270	Rotates the graphics plane anticlockwise through 90°, 180° or 270° degrees to display portrait orientation. Teletext and subtitles are rotated with the graphics, and scale to the PIG window. Notethat the video plane is **not** rotated when rotating the graphics plane. Making a change to this setting will require the STB to be rebooted before the change takes effect. **Default:** 0 **Example:** libconfig-set SETTINGS.GFX_ROTATION "90"

Setting	Allowed values	Description
HDMI_AUDIO_FORMAT	**0** = Auto **1** = PCM **2** = Dolby Digital **3** = Dolby Digital Plus For x0xx series STBs the STB now checks the EDID of the HDMI-attached device. If the attached device does not support the option selected by the end-user, then the **minimum** supported option will be selected (for example, if the user selects Dolby Digital but the TV does not support it, then the STB will output PCM).	The tag to use when reading or writing the HDMI audio bitstream format option to the settings file. **Default:** **0** **Example:** **libconfig-set SETTINGS.HDMI_AUDIO_FORMAT "1"**
HDMI_FALLBACK_MODE	**DVI** - consider the monitor as a DVI device when reading the EDID fails **HDMI** - consider the monitor as an HDMI device when reading the EDID fails.	Allows users to override the type of sink device (TV) that the STB is connected to. **Default:** **DVI** **Example:** **libconfig-set SETTINGS.HDMI_FALLBACK_MODE "HDMI"**
HDMI_VIDEO_FORMAT	**RGB** **YUV422** **YUV444**	Allows a choice of video format from the HDMI output. This setting determines the colour range encoding, for example changing from RGB to YUV can produce a deeper black. **Default:** **RGB** **Example:** **libconfig-set SETTINGS.HDMI_VIDEO_FORMAT "RGB"**

Setting	Allowed values	Description
HLS_INITIAL_BITRATE_PREFERENCE	**First** - select the first variant in the playlist. **Lowest** - select the lowest bitrate variant in the playlist. **Highest** - select the highest bitrate variant in the playlist. **Previous** - select the bitrate that was used for the previous playback session. **<integer>** - use <integer> as an absolute bitrate value.	Indicates the preferred HLS bitrate variant to be selected from an HLS playlist. **Default:** **Previous** **Example:** **libconfig-set SETTINGS.HLS_INITIAL_BITRATE_PREFERENCE**
HLS_STARTUP_MODE	**BestQuality** - the STB will try to play the highest quality content. If a higher bitrate can be selected during startup, already fetched lower quality data will be discarded and the higher quality data fetched instead. This can result in longer startup times but better quality video being displayed. **Fastest** - disables the best quality startup in preference to displaying something quickly. Better quality will be introduced as the content plays.	Sets the preferred HLS start up mode. **Default:** **Example:** **libconfig-set SETTINGS.HLS_STARTUP_MODE**

Setting	Allowed values	Description
HOSPITALITY	**Y** = the user preferences menu is not available for all users (that is, the standard hospitality configuration is retained). **N** = the user preferences menu is available for all users	For hospitality STBs, the user preferences are not accessible to normal users, the user preferences menu is only accessible through the management page menu. This is the standard hospitality configuration. Setting the value to '**N**' makes the user preferences menu available for all users. Also, setting this to '**Y**' on a non-hospitality STB can make it operate in hospitality mode See User Prefs. **Default:** **Y** **Example:** **libconfig-set SETTINGS.HOSPITALITY "Y"**
HTTP_PROXY		Sets the proxy server for HTTP transactions. Enter the IP address and port in the format **host:port**, for example: **123.4.5.67:1234**, or the address can be given in the form **hostname.domain:1234** **Default:** **Example:** **libconfig-set SETTINGS.HTTP_PROXY "123.4.5.67:1234"**
HTTP_PROXY_IGNORE	A comma separated list of domains.	Configures a flag to ignore the current HTTP proxy setting. **Default:** **Example:** **libconfig-set SETTINGS.HTTP_PROXY_IGNORE "www.google.co.uk,www.bbc.co.uk"**
HTTPS_PROXY		Configure the proxy server for HTTPS transactions. Enter the IP address and port in the format **host:port**, for example: **123.4.5.67:1234** **Default:** **Example:** **libconfig-set SETTINGS.HTTPS_PROXY "123.4.5.67:1234"**

Setting	Allowed values	Description
IGMP_END_TIMEOUT	0 - 30	Sets the number of seconds to wait after no more video data is seen, before an end-of-media event is raised. If the wait for more data extends past the timeout, the playback is stopped by the STB. Set to 0 for no timeout. **Default:** 30 **Example:** libconfig-set SETTINGS.IGMP_END_TIMEOUT "15"
IGMP_START_TIMEOUT	0 - 30	Sets the number of seconds (0-30) to wait for video data after sending an IGMP join command, before a no-video event is raised. If no video is seen before the time-out, the playback is stopped by the STB. Set to 0 for no timeout. **Default:** 30 **Example:** libconfig-set SETTINGS.IGMP_START_TIMEOUT "15"
INITIAL_LED	ON = LED is on when the STB is active. OFF = LED is off when the STB is active.	Sets the initial state of the main/IR LED, when the STB is active (that is, powered and not in standby). For a description of the LED configuration for different STBs see LED Configuration operation by STB product. **Default:** **Example:** libconfig-set SETTINGS.INITIAL_LED "ON"
IR_BRAND	0 - 15	Defines the prefix expected within the IR signal sent by a remote control in order to pair the remote with a particular STB. This can be useful in lab/demo environments when multiple STBs may receive the signal from a remote control. See Appendix H, "Pairing an IR remote control with an STB" for details of how to pair a remote control with an STB. **Default:** **Example:** libconfig-set SETTINGS.IR_BRAND "10"

Setting	Allowed values	Description
KEYMAP	**us** = United States **uk** = British **el** = Greek **ge** = German **ru** = Russian **sl** = Slovenian **uk-lrk** = British - New Amino keyboard **ge-lrk** = German - New Amino keyboard **fr-lrk** = French - New Amino keyboard	Sets the layout for the keyboard. Language specified in ISO-639 notation. **Default:** **Example:** **libconfig-set SETTINGS.KEYMAP "us"**
LEGACY_SUBS	**Y** = Use old behaviour. **N** = Use new behaviour.	New behaviour is defined as having subtitles displayed on a second framebuffer on top of the browser graphics. This means that the STB can display subtitles and browser graphics at the same time. Old behaviour is defined as displaying either browser graphics or subtitles. **Default:** **N** **Example:** **libconfig-set SETTINGS.LEGACY_ SUBS "Y"**
LOW_DISK_SPACE_LEVEL	**0** = OFF, otherwise an integer value of the free space available in kB.	Monitors free disk space and sends an event to the mpeg control client when the free disk space drops below the level specified. **Default:** **OFF** **Example:** **libconfig-set SETTINGS.LOW_DISK_ SPACE_LEVEL "8000000"**
MAXIMUM_VOLUME	**0 - 100**	Sets the maximum volume. **Default:** **100** **Example:** **libconfig-set SETTINGS.MAXIMUM_ VOLUME "100"**

Setting	Allowed values	Description
MINIMUM_VOLUME	**0 - 100**	Sets the minimum volume. **Default:** **0** **Example:** **libconfig-set SETTINGS.MINIMUM_ VOLUME "20"**
Multiple Interface settings see Dual network interfaces		
NATIVE_FRAMERATE_ PASSTHRU_ENABLE (HD products only)	**Y** = Enable native framerate passthrough. **N** = Disable native framerate passthrough.	When this is set the framerate of the video output will be set to match the framerate of the video stream being played, temporarily overriding the user preference setting if this is different. **Default:** **Example:** **libconfig-set SETTINGS.NATIVE_ FRAMERATE_PASSTHRU_ENABLE "N"**
NATIVE_PASSTHRU_ENABLE (HD products only)	**Y** = Enable native passthrough. **N** = Disable native passthrough.	Used to enable temporary switching of output resolution to SD when playing SD video, when the output resolution is set to HD. **Default:** **Example:** **libconfig-set SETTINGS.NATIVE_ PASSTHRU_ENABLE "N"**

Setting	Allowed values	Description
NETWORK_DEVICE_POLICY	**Middleware** - the network manager will not perform automatic network switching. Middleware has full control of which interface is used, and can switch at any time using netman_set_ active_device(). **ConnectionLost** - the network manager will perform automatic switching if the current device is no longer available and another device is ready. Middleware can still switch to another device which is ready using netman_set_ active_device(). **WiredAvailable** (Default) - the network manager will always switch to the wired interface if it is available. If the STB is currently connected via wireless and the wired interface is plugged in, it will switch to use it. Middleware should not use netman_set_ active_device().	Used to control how netman will switch active devices between Wi-Fi and Ethernet. By default, netman will always switch to a wired connection when it is available. **Default:** **WiredAvailable** **Example:** **libconfig-set SETTINGS.NETWORK_DEVICE_POLICY "Middleware"**

Setting	Allowed values	Description
NTPCLIENT_TIMEOUT		Sets the NTP client timeout in seconds. **Default:** **Example:** `libconfig-set SETTINGS.NTPCLIENT_TIMEOUT "10"`
PLT_SPOOLTIME (Valid for PVR capable platforms only)	An unsigned integer in range from **10** to **1440.**	The maximum length of the PLT buffer in minutes. **Default:** **60** **Example:** `libconfig-set SETTINGS.PLT_SPOOLTIME "60"`
PLT_START_DELAY (Valid for PVR capable platforms only)	An unsigned integer in range from **0** to **60.**	Used to set the PLT start delay (in seconds). Sets the number of seconds after stream begin before the PLT buffer is started. **Note:** If the value of PLT_START_DELAY is **0** (zero) or not set, PLT buffering is disabled. **Default:** **Not set (PLT disabled by default)** **Example:** `libconfig-set SETTINGS.PLT_START_DELAY "5"`
PLT_START_OF_BUFFER_ACTION	**PLAY** **PAUSE**	Controls the BOS behaviour for the PLT buffer **Default:** **Example:** `libconfig-set SETTINGS.PLT_START_OF_BUFFER_ACTION "Play"`
POD_SENSITIVITY	**0 - 10**	The remote sends a signal every 10 centiseconds, this setting configures how many signals are required before the mouse moves. Hence a setting of **2** will indicate that two signals need to be received before the mouse moves. **Default:** **Example:** `libconfig-set SETTINGS.POD_SENSITIVITY "5"`

Setting	Allowed values	Description
PREFERRED_CAPTIONING_SYSTEM	**NONE** = default – select **SOFTCC** = select SoftCC output by default if both systems are running. **DVBTTX** = select Dvbttx output by default if both systems are running	Sets a preferred captioning system (either SoftCC or Dvbttx) if both systems are available. **Default:** NONE **Example:** `libconfig-set SETTINGS.PREFERRED_CAPTIONING_SYSTEM "SOFTCC"`
PREF_HD_RESOLUTION	**576P** – PAL **480P** – NTSC **720P** **1080I** **1080P**	Used to store the preferred output resolution to switch to when automatic output resolution is enabled. (when the **NORFLASH**. **OUTPUT_RESOLUTION** setting is unset or is set to **HDAUTO**) **Default:** **720P** **Example:** `libconfig-set SETTINGS.PREF_HD_RESOLUTION "576P"`
PREFERRED_LANG	ISO 639-2 language code	Sets the preferred audio language from a video stream. Set to **""** for automatic language selection. **Default:** **None** **Example:** `libconfig-set SETTINGS.PREFERRED_LANG "fr"`
PVR_SPOOLTIME (PVR capable platforms only)	An unsigned integer in range from **60** to **1440.**	The maximum length of the PVR buffer in minutes. The maximum length of recording allowed. **Default:** **1440 if not set** **Example:** `libconfig-set SETTINGS.PVR_SPOOLTIME "180"`

Setting	Allowed values	Description
RCUCURSOR_CONTROLS_MOUSE	**Y** = remote control arrow keys moves the mouse. **N** = remote control arrow keys acts as cursor keys	Allows the cursor keys to either be mouse keys or keyboard arrow keys when the mouse pointer is active. **Default:** **Example:** `libconfig-set SETTINGS.RCUCURSOR_CONTROLS_MOUSE`
REMOTE_LAYOUT	**Default** **Hospitality** **Myrio** **Think** **TotalVision** **VillaFontaine** **Guide** **GuestTek** **GETC** **Sejinalt** **Myrio2think** **RAW**	Sets the default IR remote control layout. Different remote layouts specify how certain keys on the remote control are interpreted. Customer-specific layout values also exist. See Remote layout options for further information. See Volume control for details of controlling the volume using the IR remote. **Default:** **NULL** **Example:** `libconfig-set SETTINGS.REMOTE_LAYOUT "Hospitality"`
REMOTECONF	IP address	Sets the multicast IP address that the STB listens on for **STBremoteconf** commands. **Default:** **225.10.10.10** **Example:** `libconfig-set SETTINGS.REMOTECONF "234.56.78.9"`
REMOTECONFPORT	0 - 65535	Sets the multicast port that the STB listens on for **STBremoteconf** commands. (See **STBRCPORT** for information on how to set the unicast port). **Default:** **22222** **Example:** `libconfig-set SETTINGS.REMOTECONFPORT "12345"`

Setting	Allowed values	Description

REMOTECONF_URI		Sets the URI that the STB listens on for source-specific multicast (SSM) commands via **STBremoteconf**. This is similar to SOFTWARE_URI, but the SSM include/exclude options are added after the multicast address and port. Up to eight 'include' and eight 'exclude' addresses can be added as a comma separated list. If **REMOTECONF_URI** is not set the existing **REMOTECONF** and **REMOTECONFPORT** will be read and non-SSM behaviour will be used.
		Note: only the first option that appears in the string is supported, so if both the 'include' and 'exclude' options are present, one will be ignored
		Default:
		Example with 'include' options
		`libconfig-set SETTINGS.REMOTECONF_ "igmp://239.255.250.1:11111?incl= 10.172.2.1,10.172.2.2"`
		Example with 'exclude' options
		`libconfig-set SETTINGS.REMOTECONF_ "igmp://239.255.250.1:11111?excl= 10.172.2.1,10.172.2.2"`

REPEAT_RATE	**0** = Do not repeat any keys **2** - **30** repeat rate (in cs)	Sets the key repeat rate in centiseconds, to control the number of characters per second that a user can enter via an IR input device (for example a keyboard). **"0"** indicates no key repeat. **Default:** **8** **Example:** `libconfig-set SETTINGS.REPEAT_ RATE "0"`
RFBOOST_DOLBY	**Y** = Enable Dolby audio level boosting over RF. **N** = Disable Dolby audio level boosting over RF.	Sets whether audio level boosting over RF for Dolby audio type is enabled or disabled. **Default:** **N** **Example:** `libconfig-set SETTINGS.RFBOOST_ DOLBY "Y"`

Setting	Allowed values	Description
RFBOOST_MPEG	**Y** = Enable mpeg audio level boosting over RF. **N** = Disable mpeg audio level boosting over RF.	Sets whether audio level boosting over RF for mpeg audio type is enabled or disabled. **Default:** **N** **Example:** **libconfig-set SETTINGS.RFBOOST_ MPEG "Y"**
RTP_AUTO_SKIP_ENABLE	**Y** = Enable **rtpskip**. **N** = Disable **rtpskip**.	Sets whether **rtpskip** is automatically enabled or disabled for a stream with RTP encapsulation. **Default:** **N** **Example:** **libconfig-set SETTINGS. RTP_AUTO_SKIP_ENABLE "Y"**
RTCP_RETRANSMISSION_ ENABLE	**Y** = Enable RTCP retransmission **N** = Disable RTCP retransmission	Sets the global default behaviour for whether RTCP retransmission is used or not. **Default:** **N** **Example:** **libconfig-set SETTINGS.RTCP__ RETRANSMISSION_ENABLE "Y"** The default behaviour can be overridden on a per-stream basis by adding **"rtcpretrans=yes"** or **"rtcpretrans=no"** to the stream URI.
RTSP_BASE_PORT	**1 - 65535**	Sets the initial RTSP UDP video port number the STB is to use. By default, the STB starts at 11111 and increases in increments of 10. **Default:** **554** **Example:** **libconfig-set SETTINGS.RTSP_ BASE_PORT "1234"**
RTSP_END_TIMEOUT	**0 - 30**	Sets the number of seconds (0-30) to wait after no more RTSP data is seen, before an end-of-media event is raised. If the wait for more data extends past the timeout, the playback is stopped by the STB. Set to **0** for no timeout. **Default:** **6** **Example:** **libconfig-set SETTINGS.RTSP_END_ TIMEOUT "15"**

Setting	Allowed values	Description
RTSP_SCALE	**0 - 6**	The default fast forward and rewind multiplier. A scale of **2** will mean that you move forward through the stream twice as fast. RTSP speed gives the whole data stream at that speed. A 4Mbit/s stream at a scale of **2** gives the whole stream at 8Mbit/s. A value of **0** will pause the stream. **Default:** **6** **Example:** **libconfig-set SETTINGS.RTSP_ SCALE "2"**
RTSP_SERVER	**ncube** **mediabase** **oracle** **concurrent** **bitband** **infovalue** **seachange** **eona** **smartvision**	Sets the default video server type. **Default:** **ncube** **Example:** **libconfig-set SETTINGS.RTSP_ SERVER "mediabase"**
	 It is now only necessary to set the default video server type for **eona** - for other servers the STB will detect the type from the RTSP stream.	
RTSP_SIMPLE_TRANSPORT	**Y**= Offer **RAW/ RAW/UDP** transport only and omit destination details. **N**= Offer all supported transports	This gives the option of forcing the use of transport RAW/RAW/UDP;unicast and prevents adding destination details to the SETIUP request. **Default** **N** **Example:** **libconfig-set SETTINGS.RTSP_ SIMPLE_TRANSPORT "Y"**
RTSP_START_TIMEOUT	**0 - 30**	Sets the number of seconds (0-30) to wait for video data after sending an RTSP play command, before a no-video event is raised. If no video is seen before the time-out, the playback is stopped by the STB. Set to **0** for no timeout. **Default**: **4** **Example**: **libconfig-set SETTINGS.RTSP_START_ TIMEOUT "15"**

Setting	Allowed values	Description
SECONDARY_LANG	ISO 639-2 language code	The secondary audio language from a video stream selection. Set **“”** for automatic language selection. **Default** **Example:** **libconfig-set SETTINGS.SECONDARY_ LANG “fr”**
SELECT_RTSP_STREAM_BY_ PORT_ONLY	**Y**= selects stream by port only **N**= selects stream by IP address	If VOD servers send video data with a source IP address different to that from their control IP address (that is, the IP address you use to control it over RTSP), and they do not tell the STB this in their response to the SETUP command, then video will not play. Setting this command turns off the IP address check. Reboot the STB for it to take effect (the setting will be lost if you upgrade the STB again.) To have this command in the upgrade image, edit the **settings** file by adding the following line: **SELECT_RTSP_STREAM_BY_PORT_ONLY="Y"** Create the mc2 upgrade image and upgrade the STB with that image. **Default** **Y** (on Minerva builds only) **Example:** **libconfig-set SETTINGS.SELECT_ RTSP_ STREAM_BY_PORT_ONLY “Y”**
SKIP_DHCP_LEASE_CHECK	**Y**= skip the DHCP lease check and continue to boot. **N** = remain at runlevel 3 until DHCP lease check is completed.	During the boot sequence, the STB will stay at runlevel 3 until it gets a DHCP lease (or reaches the end of a 60s timeout). From version 2.3.2, this setting allows the STB to bypass the check for a DHCP lease and continue to boot as in previous releases. **Default** **N** (on Minerva builds only) **Example:** **libconfig-set SETTINGS.SKIP_DHCP_ LEASE_CHECK “Y”**

Setting	Allowed values	Description
STANDBY_LED	**ON** = LED is on when the STB is in standby. **OFF** = LED is off when the STB is in standby.	Sets the state of the standby LED when the STB is in standby. For a description of the LED configuration for different STBs see LED Configuration operation by STB product. **Default:** **Example:** `libconfig-set SETTINGS.STANDBY_LED "OFF"`

STBRCPORT	0 - 65535	Sets the unicast port that the **STBremoteconf** daemon listens on for **STBremoteconf** commands. (See **REMOTECONFPORT** for information on how to set the multicast port).
		Default:
		`22222`
		Example:
		`libconfig-set SETTINGS.STBRCPORT "12345"`

STC_OFFSET
(HD products only)
	-10000 to -40000	The System Time Clock (STC) offset is the value in 90KHz ticks by which the STC is offset from the Programme Clock Reference (PCR). The offset determines the size of the data buffer which needs to be set so as to avoid data underrun or overrun as the result of network jitter. The STC offset has a consequential effect on the channel change time; the shorter the offset the shorter the channel change time. However this must be balanced against the possibility of data underrun as the result of a smaller buffer.
		The current STC offset value is **-10000** and this is now configurable. The channel change time can be reduced by approximately 100ms for every STC offset decrease of **10000**, however **STC_OFFSET** must be in the range **-10000** to **-40000** inclusive..
		Default:
		`-10000`
		Example:
		`libconfig-set SETTINGS.STC_OFFSET "-10000"`

Setting	Allowed values	Description
STEREOSCOPIC_ PASSTHROUGH	**on** = 3D streams will be output as 3D even if the TV does not support it. **off** = the TV will not be switched to 3D and 3D streams will be scaled down to 2D. **auto** = the TV will be switched to 3D if it is 3D-capable and a 3D stream is being played.	Sets the behaviour when 3D streams are present. Note that 3D is only output when a single full-screen video window is visible. If the video is scaled, or if more than one window is visible, output is forced to 2D. **Default:** **Example:** `libconfig-set SETTINGS. STEREOSCOPIC_PASSTHROUGH "on"`
STREAM_RATE_LIMIT	**0** or not set = no limit	Limits the maximum data fetch rate for HTTP OTT streams. The limit size is specified as a percentage of the playback rate, where 100 is normal playback rate, 150 would be 50% faster and so on. **Note:** the actual maximum data fetch rate achieved may be lower than expected due to network limitations. This setting does not affect HLS. **Default:** **Example:** `libconfig-set SETTINGS.STREAM_ RATE_LIMIT "150"`
SUBTITLES_OPTION	**Y** = Turn subtitles on. **1** = Turn subtitles on. **N** = Turn subtitles off. **0** = Turn subtitles off.	Sets the mode of the subtitles. **Default:** **N** **Example:** `libconfig-set SETTINGS.SUBTITLES_ OPTION "N"`

Setting	Allowed values	Description
SUBTITLES_PREF_LANG	ISO 639-2 language code.	Sets the primary language for default subtitle display. Set **""** for automatic language selection. **Default:** **Example:** **libconfig-set SETTINGS.SUBTITLES_PREF_LANG "fr"**
	 If both dvb subtitles and teletext subtitles are present in the stream **and** they have the same language descriptor **and** this matches the primary language (or secondary language if the primary language does not match) **then** dvb subtitles will always be selected. If neither primary nor secondary language match then the lowest-numbered subtitle PID will be selected. There is no mechanism to select teletext subtitles over dvb subtitles as teletext subtitles is considered to be a more limited legacy technology and therefore inferior to dvb subtitles.	
SUBTITLES_SECOND_LANG	ISO 639-2 language code.	Sets the secondary language for default subtitle display. Set **""** for automatic language selection. **Default:** **Example:** **libconfig-set SETTINGS.SUBTITLES_SECONDARY_LANG "fr"**
SYSLOG_BUFFER_SIZE	**4** **16** **64**	Sets the size of the syslog/logread buffer in kilobytes. **Default:** **Example:** **libconfig-set SETTINGS.SYSLOG_BUFFER_SIZE "16"**
SYSLOG_KERNEL_REDIRECT	**Y** = show kernel debug **N** = do not show kernel debug	Allows syslog to show kernel debug. **Default:** **Example:** **libconfig-set SETTINGS.SYSLOG_KERNEL_REDIRECT "Y"**

Setting	Allowed values	Description
SYSLOG_REMOTE_ADDR		The IP address of the network syslog server. An address must be specified before remote logging can be enabled. After changing this settings, the **rc.syslogd** script must be executed with the **restart** command for the changes to take effect. **Default:** **Example:** **libconfig-set SETTINGS.SYSLOG_ REMOTE_ADDR "123.45.67.89"**
SYSLOG_REMOTE_LOG_ ENABLED	**Y** = enable. **N** = disable.	Controls whether the remote logging is enabled (Y) or not (N). After changing this settings, the **rc.syslogd** script must be executed with the **restart** command for the changes to take effect. **Default:** **N** **Example:** **libconfig-set SETTINGS.SYSLOG_ REMOTE_LOG_ENABLED "Y"**
SYSLOG_REMOTE_LOG_TIME	**UNTIL:XXX** - where **XXX** is a number of seconds (since epoch - 00:00 01/ 01/1970). This allows an absolute time to be specified. **FOR:XXX** - where **XXX** is a number of seconds.	Controls if and when remote logging is automatically disabled. If nothing is specified, remote logging will be enabled until it is manually disabled. The disable time can be specified in two formats: **UNTIL:XXX** **FOR:XX** where **XXX** is a number of seconds. This allows a relative time from remote logging being enabled to be specified. If the STB reboots after remote logging is started with a 300 second delay, the logging time will start at 300 seconds when the STB boots again. After changing this settings, the **rc.syslogd** script must be executed with the **restart** command for the changes to take effect. **Default:** **Example:** **libconfig-set SETTINGS.SYSLOG_ REMOTE_LOG_TIME "FOR:300"**

Setting	Allowed values	Description
SYSLOG_REMOTE_PORT		The port number of the network syslog server. If not specified, the default 514 port number will be used. After changing this settings, the **rc.syslogd** script must be executed with the **restart** command for the changes to take effect. **Default:** **514** **Example:** **libconfig-set SETTINGS.SYSLOG_ REMOTE_PORT "100"**
SYSTEM		Allows information about the STB and software version to be read. **Default:** **Example:** **libconfig-get SYSTEM "STB_MODEL"**
	SOFTWARE_VERSION = the version of the AmiNET software **CUSTOM_VERSION** = the version of any custom resource applied **OEM_VERSION** = any customer applied version information **STB_MODEL** = the STB model (for example, A540) **STB_FAMILY** = the STB family (for example, Ax4x) **SOFTWARE_BUILD_TIME** = the date and time that the AmiNET software was built **OS_VERSION** = the version of the Operating System **CPU_MODEL** = the type of CPU used on the STB **SOC_VERSION** = the version of the SoC on the STB **SOC_SERIAL_ID** = the serial number of the SoC on the STB **TOTAL_MEMORY** = the total physical DRAM available on the STB **INTERNAL_HDD_PRESENT** = returns whether a HDD is present or not (**Y** or **N**)	
TELETEXT_FULLSCREEN	**Y** = enable full screen operation. **N** = disable full screen operation.	Enables or disables teletext full screen operation. (Opera subtitle builds only.) **Default:** **Y** **Example:** **libconfig-set SETTINGS.TELETEXT_ FULLSCREEN "N"**

Setting	Allowed values	Description
TEXT_ENTRY	1 = Enable text entry. 0 = Disable text entry.	Enables or disables text entry via the remote control. If text entry is enabled, the user enters text by holding down the appropriate number key. **Default:** **Example:** `libconfig-set SETTINGS.TEXT_ENTRY "1"`

TOOLBAR_STATE	1 = Toolbar is visible.
	0 = Toolbar is not visible.
		Sets the default for whether the browser's toolbar is visible or not when the STB boots. This can be overridden by settings for individual pages.
		Default:
		`1`
		Example:
		`libconfig-set SETTINGS.TOOLBAR_STATE "1"`

TVI_TYPE	0 = Philips 1 = Zenith 2 = LG 3 = Mate (Sony/ Panasonic) 4 = TVLink 5 = Generic 6 = Samsung	Configures which TV type the TVI interface needs to use on the programmable interface controller (PIC). **Note**: this setting is used on Hospitality platforms only. **Default:** **Example:** `libconfig-set SETTINGS.TVI_TYPE "0"`
UDP_END_TIMEOUT	0 – 30	Sets the number of seconds to wait when no more UDP data is seen, before an end-of-media event is raised. If the wait for more data extends past the timeout, the playback is stopped by the STB. Set to 0 for no timeout. **Default:** **Example:** `libconfig-set SETTINGS.UDP_END_TIMEOUT "15"`

Setting	Allowed values	Description
UDP_START_TIMEOUT	**0 – 30**	Sets the number of seconds (0-30) to wait for video data after trying to connect to a UDP media stream, before a no-video event is raised. If no video is seen before the time-out, the playback is stopped by the STB. Set to **0** for no timeout. **Default:** **Example:** **libconfig-set SETTINGS.UDP_START_TIMEOUT "15"**
USB_SPEED	**""** = Use autodetection. **lowspeed** = 1.5 Mbits/s (USB 1). **fullspeed** = 12 Mbits/s (USB 1). **highspeed** = 480 Mbits/s (USB 2 only).	Configures whether the USB interface should auto-negotiate a speed with a client (default) or whether it should be limited to a particular speed. For the Tira dongle, this must be set to **lowspeed**. If you are setting this via **libconfig**, you will need to reboot the STB in order for the setting to take effect. **Default:** **""** **Example:** **libconfig-set SETTINGS.USB_SPEED "lowspeed"**
USE_PROXY	**Y** = Enable HTTP proxy. **N** = Disable HTTP proxy.	Enables or disables the use of HTTP, HTTPs and FTP proxy servers. **Default:** **N** **Example:** **libconfig-set SETTINGS.USE_PROXY "Y"**
USER_AGENT_STRING	A string of up to 60 characters in length.	Appends a 60 character ASCII string to the user agent string. This feature can be cleared by setting an empty string. Note: Making this change will require the STB to be rebooted for the change to take effect. **Default:** **Example:** **libconfig-set SETTINGS.USER_AGENT_STRING "Aminotech"**

Setting	Allowed values	Description
VIDEO_ERROR_MODE	**None** = Lowest error recovery mode. This will result in macroblocks on error. **Partial** = Partial error recovery mode. **High** = High error recovery mode. **Full** = Highest error recovery mode. No macroblocking is seen.	Used to set the video error recovery mode used by the video decoder. **Default:** **Partial** **Example:** **libconfig-set SETTINGS.VIDEO_ ERROR_MODE**
VQE_DISABLE	**Y** = Disable VQE **N** = Enable VQE	Disables VQE **Default:** **N** **Example:** **libconfig-set SETTINGS.VQE DISABLE "Y"**
WATCHDOG_HEARTBEAT x0xx series only		Used to set the heartbeat interval in seconds for the hardware watchdog. The heartbeat interval cannot be more than half of the **WATCHDOG_TIMEOUT** duration (for example, if you set a timeout of 30 s and a heartbeat of 40 s, the heartbeat will be set to 15 s. **Default:** **4** **Example:** **libconfig-set SETTINGS.WATCHDOG_ HEARTBEAT "10"**
WATCHDOG_TIMEOUT x0xx series only	**0 - 65535**	Used to set the timeout in seconds for the hardware watchdog. If the timeout is non-zero, the **WATCHDOG_HEARTBEAT** must also be non-zero. **Default:** **30** **Example:** **libconfig-set SETTINGS.WATCHDOG_ TIMEOUT "60"**

B.2 Remote layout options

These are the options for **`libconfig-set SETTINGS.REMOTE_LAYOUT`**.

Layout	Description
`Default`	The default layout.
`Hospitality`	As the default layout but with the following changes: • CH_UP/CH_DOWN and number buttons become TVPOWER. • TVPOWER/STB become HOME. • VID_PLAY becomes VID_PLAYPAUSE.
`Think`	Default setting for all Amino Minerva builds.
`Totalvision`	As default with the following changes: • CH_UP/CH_DOWN become TVPOWER • TVPOWER/STB become HOME • RED becomes VID_REVERSE • GREEN becomes VID_PLAYPAUSE • YELLOW becomes VID_STOP • BLUE becomes VID_FORWARD
`Villafontaine`	As the default layout with the following changes: • CH_UP/CH_DOWN become TVPOWER • TVPOWER/STB become HOME
`Myrio`	To operate the STB using a Myrio remote control.
`Guide`	As the default layout but maps the -/-- to be an EPG key
`GuestTek`	Affects Sejin keyboard remote only in the following ways: • Right mouse button will send a CIR_BTN_RIGHT_MOUSE key press regardless of the mouse being hidden or not. • Bookmarks becomes Vol+. • Help becomes Vol-. • Toolbar becomes CH+. • Ins becomes CH-. • Del becomes power.
`CETC`	Default setting for all CETC builds.
`Sejinalt`	Alternative Sejin remotes provided by Koolconnect.
`Myrio2think`	Used for Myrio UEI Sejin remotes to map keys to Minerva think codes.
`RAW`	Sends unmapped key codes from the RCU, allowing custom RCUs to be configured as required. When using the **`RAW`** setting the keycodes will not match those of the Willow remote.

B.3 LED Configuration operation by STB product

The **SETTINGS.INITIAL_LED** value only has an effect when the STB is active. It has no effect on the LEDs when the STB is in standby.

The **SETTINGS.STANDBY_LED** value only has an effect when the STB is in standby. It has no effect on the LEDs when the STB is in active.

To change the default state for all boxes, modify the file in your **imagecomponents.settings** to include these 2 lines:

SETTINGS.INITIAL_LED ="**ON**" or "**OFF**" as required

SETTINGS.STANDBY_LED ="**ON**" or "**OFF**" as required

B.3.1 STB in ACTIVE mode

STB	SETTINGS.INITIAL_LED value		
	unset or ""	"on"	"off"
Single LED STBs: A129, A140, H140, A150	ON	ON	OFF
A540PVR	Main LED OFF Standby LED GREEN	Main LED ON Standby LED GREEN	Main LED OFF Standby LED GREEN

B.3.2 STB in STANDBY mode

STB	SETTINGS.STANDBY_LED value		
	unset or ""	"on"	"off"
Single LED STBs: A129, A140, H140, A150	OFF	ON	OFF
A540PVR	Main LED OFF Standby LED RED	Main LED OFF Standby LED RED	Main LED OFF Standby LED RED

B.4 Volume control

In the default condition, the Amino remote will send out TV commands for the volume control +/- functions.

The **AudioControl.SetVolume()** / **GetVolume()** JMACX calls can then be used to implement a volume control (see the Amino *JavaScript Media Access Control Extensions API specification* for further information).

Action codes 41 (up) and 42 (down) in **`fkeys.conf`** can also be used (see Function keys file), but this would not allow for any visual feedback by returning the currently set volume.

Remote	Default behaviour	How to enable volume control
Grey remote 	When a TV brand code **is** configured the remote will emit the appropriate TV codes regardless of the mode of the remote. When a TV brand code **is not** configured the remote will not emit any code when the volume +/- buttons are pressed, regardless of the mode of the remote.	Enter a TV brand code of **2 3 1** as follows: • Press **TV** and **OK** together until the TV button lights up • Enter **2 3 1** and the TV light will go out It is also necessary to ensure that the **`REMOTE_LAYOUT`** setting in the settings file is set to **`AMINOTV`**. This remote will only control volume on the A110(H) product range and not the newer products.
AVC remote 	When a TV brand code **is** configured the remote will emit the appropriate TV codes regardless of the mode of the remote. When a TV brand code **is not** configured the remote will not emit any code when the volume +/- buttons are pressed, regardless of the mode of the remote.	Press and hold the **STB** button until it flashes. If **9 9 3 Volume Up** is now pressed, the remote will emit codes for the STB in all modes. If **9 9 3 Volume Down** is now pressed, the remote will emit codes for the STB in the current mode (for example, TV if the TV mode was selected prior to entering programming mode). If **9 9 3 Mode** is entered, this will cause the remote to emit volume codes for the selected device in the currently selected mode.
Willow remote 	When a TV brand code **is** configured the remote will emit the appropriate TV codes. When a TV brand code **is not** configured the remote will emit the STB volume codes (not the TV volume codes) when the volume +/- buttons are pressed.	Configure the remote so that no TV brand code is available. An existing brand code can be deleted as follows: • Press and hold the **1** and **6** buttons simultaneously for approximately 3 seconds until the TV standby button is lit. • Enter the key sequence **9 9 6**.

Appendix C—DHCPC file contents

In this Chapter:

The following table describes the settings available in the **dhcpd** file. Note that whether these values are available depends on the DHCP server configuration. For convenience, the values are listed in alphabetical order.

C.1 DHCPC file settings

The examples show how to obtain the value provided by the DHCP server. Defaults are configured using other settings. Any values received via DHCP will be used in preference to these default settings. See the appropriate documentation for information on how to set defaults using other methods

In each case **null** is returned if:

- The DHCP server failed to reply
- DHCP is not enabled
- The relevant setting is not configured in the DHCP server

Setting	Description

BROWSER_HOMEPAGE	Access to the browser homepage returned by the DHCP server
	The browser homepage returned by the DHCP server which overrides both the default page and the one set by **LIBCONFIG_HOMEPAGE**. Value is read only and only available if DHCP is enabled.
	Example:
	`libconfig-get DHCPC.BROWSER_ HOMEPAGE`
	`> aminocom.com`

DNS	The DNS server returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-set DHCPC.DNS`
	`>192.168.0.0`

DOMAIN	The domain returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.DOMAIN`
	`> aminocom.com`

GATEWAY	The gateway returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.GATEWAY`
	`>`

IPADDR	The IP address returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.IPADDR`
	`> 192.168.0.0`

Setting	Description

LEASETIME	The DHCP lease time returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.LEASETIME`
	`> 600`

NETMASK	The netmask returned by the DHCP server if DHCP is enabled. This is read only
	Example:
	`libconfig-get DHCPC.NETMASK`
	`> 255.255.0.0`

REBINDTIME	The DHCP rebind time returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.REBINDTIME`
	`> 2268000`

REMOTECONF_URI	The URI returned by the DHCP server that STBremoteconfd will listen on for incoming commands. This overrides the **SETTINGS.REMOTECONF_URI** setting. **Example:** `libconfig-get DHCP_REMOTECONF_URI <igmp URL>` See REMOTECONF_URI for examples of source specific multicast URIs.

RENEWALTIME	The DHCP renewal time returned by the DHCP server if DHCP is enabled. This is read only.
	Example:
	`libconfig-get DHCPC.RENEWALTIME`
	`> 1296000`

Appendix D—Using the configuration techniques

In this Chapter:

The following appendix gives examples of how to use some of the configuration techniques.

D.1 Configuring image components

Configuring image components enables you to customise the files used to create a software upgrade image. When the **signupgradeimage** script creates a new image, it includes the files that are listed in the **flashcontents** file and stored in the **imagecomponents** subdirectory of the software release. You can edit these files or add new ones in order to customise the software upgrade image to your requirements.

For example, you can:

- edit configuration files before they are made available to STBs.
- add files needed to enable Telnet, or SSH(Dropbear).
- add/edit an **upgrade.sh** script file that the STB will execute when it installs the new software upgrade image.
- replace graphics (displayed during the bootstrapping and upgrading stages) with customised versions – for example, the **splash.gif** graphic.

All files in the **imagecomponents** directory must be listed in the **flashcontents** file, otherwise, the **signupgradeimage** script will fail.

D.1.1 How to configure image components

In all the following instructions, **<release_name>** indicates the full name of the software release image directory that contains your Amino software release. For example, **Ax4x-3.2.1-Opera11** installs in **3.2.1-Ax4x-opera11**, hence **<release_name>** is **3.2.1-Ax4x-opera11**.

The general procedure for editing image components is as follows:

1. Open the file that you want to edit. The image component files are in the following location:

   <release_name>/upgradeimage/imagecomponents/

   Alternatively, you can add a new file to this location - such as a **chnls.txt** file you have copied from a pre-configured STB or an **upgrade.sh** file that contains **libconfig** commands to set NOR Flash values.

2. Edit the file, and save your changes.

3. Open the **flashcontents** file, and ensure that the file you have added/edited is listed there. This file is in the following location:

   <release_name>/upgradeimage/flashcontents

   The **flashcontents** file lists the files that will be included in a software upgrade image, and defines permissions for the file.

   Example flashcontents file

   This shows a small part of a typical **flashcontents** file.

```
# R == file must exist
#      must be read-only
#      must match the checksum in listfile.sig
# W == file may exist
#      must not be executable
```

```
#      we don't care about the checksum
# E == file may exist
#      if it does exist:
#      it must be read-only
#     must match the checksum in listfile.sig

R AMINET.img
R libm223.so
R libc223.so
R xfresco
R xfresco.amem
R mkfs.ext2
R fsck.ext2
R tune2fs
R mkfs.xfs
R font_opt.bin
R texttvd
R ttsub
R bbsdk.cfg
R irb_keys.txt
W noformat
W !poweron
W cookies.txt
W history.txt
```

4. Use the **signupgradeimage** script to create the software image, as usual.

The new software upgrade image includes your new or edited files, and you can use it to upgrade your STBs.

D.1.1.1 Using Telnet or SSH to send libconfig commands

See Chapter 4, "Using libconfig" for details of how to access STB configuration via Telnet or SSH. These remote log in tools are not normally included in software builds, but you can enable the appropriate tool by adding the file to the image components used to create the software image loaded onto the STB.

D.2 Configuring a large deployment

It is possible to configure software before deploying it to a large number of STBs. The following steps describe recommendations on how this can be done for both the initial STB configuration and for future upgrade images.

D.2.1 Prerequisites

The configuration procedure outlined here assumes you already have an STB with a valid software image installed. This must include one of the remote log-in tools (Telnet/SSH/Dropbear).

1. Configure the software on the box (for example, via the Management pages), and test the configuration to ensure the STB now operates as intended.
2. When you are happy with the configuration, use the remote log-in tool to connect to the STB.
3. Copy the contents of the files that have been modified.

On the STB use:

```
mkdir /nfs
mount -o nolock <ip addr of host>:/path_to_nfs_export/ /nfs cd /nfs cp /mnt/nv/settings /nfs/
```

The configuration files are in **/mnt/nv**, and you will need to copy them to the **<release_name>/upgradeimage/imagecomponents** directory on your PC. If you are not sure which files have been modified, then copy all the contents of the directory. Normally only the **settings** file will have been modified.

4. If you have added any new files that need to be included in the upgrade image, add them to the **flashcontents** file in the **<release_name>/upgradeimage/** directory.
5. Create a new signed upgrade image (and add it to the directory specified in the multicast server's configuration file). Ensure that the new image has an incremented deployment index, if you are using deployment indexes to control software versions installed on the STB (this is **not** recommended for a trial system).
6. Upgrade the software on a single STB with this new upgrade image (for example, by using **STBremoteconf**).
7. Check that the new software loads and works as expected. If changes need to be made, repeat steps 1. to 6. until the configuration is correct.
8. Instruct all STBs on your network to upgrade to this new image (for example, use the deployment index mechanism to force an automatic upgrade).

Appendix E—Media source URLs and HLS

In this Chapter:

Streamed media will always have a valid, publicly accessible URL. These URLs, with optional attributes, are used to join media sessions. They can be used the Amino API functions and in the selected browser's URL bar.

A media server is a device that stores and shares media. It may be a simple attached storage, such as an external disk drive, or a commercial web server that hosts media for a large web site. It may also be used to run special applications that allow users to access the media from a remote location via the internet. The only requirements for a media server is a method of storing media and a network connection with enough speed to allow access to that media.

If Macrovision is enabled then it is not possible to play a media URL from a toolbar as the Macrovision mode byte can only be set from inside an HTML page. However, JMACX calls could be used to play the content. (This also applies to **`tshttp`** URLs.)

E.1 Media source types

AmiNET products support various streaming protocols, including Real Time Streaming Protocol (RTSP), Internet Group Management Protocol (IGMP) and User Datagram Protocol (UDP). In some cases, the format of the URL for a media stream depends on the media server, but the following sections give general guidelines on URLs. The URL types supported are as follows:

- **`rtsp://`** - Real Time Streaming Protocol, supporting servers from Kasenna, nCube and others.
- `igmp://` - Multicast UDP streaming with IGMP group membership
- **`http://`** - HTTP streaming is supported when using the full BitBand client.
- **`tshttp://`** - A transport stream downloaded over an HTTP connection. (Can only be used with JMACX calls.)
- **`udp://`** - UDP streaming of video
- **`tsfile://`** - Local file playback

The AmiNET products support the video streaming protocols RTSP, IGMP and UDP, as defined below:

RTP is **not** supported. Ax4x STBs will automatically detect RTP headers in UDP streams and strip them out. See **`rtpskip`** under Media source attributes and **`RTP_AUTO_SKIP_ENABLE`**.

E.1.1 RTSP

URLs for an RTSP stream can have the following format:

`rtsp://server.domain.com:554/videos/movie.ts`

`rtsp` indicates that this is an RTSP URL.

`server.domain.com` is the hostname of the video-on-demand RTSP server.

`554` is the port that the server listens on for connections.

The other information included is a path to the media source and additional attributes.

```
rtsp://server.domain.com:554/videos/movie2.ts;audiopid=4195;
        pcrpid=4149;videopid=4149;servertype=eona;eomfreeze=yes
        ;unloadvideo=no
```

or

```
rtsp://server.domain.com//server.domain.com/assetname;servertype=eona
```

Some of these attributes are described in Media source attributes.

E.1.1.1 Playing an RTSP stream, with playback a number of seconds into the video

This is done by adding the **offset=nnn** parameter to the RTSP request URL, where nnn is the number of seconds into the stream you wish to start. This works on most compliant RTSP servers.

For example, to start 5 mins into the video:

rtsp://<server>:<port>/<path>/<asset>;offset=300

E.1.1.2 Using Pause-Live-TV (PLT) with a VoD server

From version 3.2.1 it is possible to Pause-Live-TV on Mediabase VoD servers. Timeshifting the media source is defined by the **assettype=nPLT** and **offset_clock=yyyymmddThhmmss.00Z.**

The full URL in this instance would be:

rtsp://10.172.4.26
BBC1;ServiceType=TSTV;assettype=nPLT;offset_clock=yyyymmddThhmmss.00Z

E.1.2 IGMP

URLs for an IGMP (UDP multicast) stream can be as follows:

igmp://239.255.250.2:11111

igmp indicates that this is an IGMP stream.

239.255.250.2 is the IP address associated with the stream.

11111 is the port number for the multicast stream.

In a stream that contains multiple programs, the URL can also include additional attributes if required (see Media source attributes), for example:

igmp://239.255.250.2:11111;audiopid=17;pcrpid=16;videopid=17

E.1.2.1 Playing a multicast video stream

Assuming your stream is being sent to a valid multicast address and nothing in your network infrastructure stops multicast traffic, then the URL to get the STB to play will be:

igmp://:port

For example

igmp://239.192.128.1:11111

E.1.3 HTTP

URLs for an HTTP stream can be as follows:

http://exampledomain.com/

http indicates that this is an http URL.

exampledomain.com is the domain associated with the stream .

E.1.4 Streaming MP3s

From version 3.2.1, mp3 URL types are no longer supported. IGMP can still be used for multicast MP3 streams by using an IGMP URL with the 'type' attribute:

```
igmp://<address>:<port>;type=mp3
```

E.1.5 UDP

URLs for a unicast UDP stream can be as follows:

```
udp://192.168.0.1:11111
```

udp indicates that this is a UDP stream.

192.168.0.1 is the unicast destination address associated with the stream (that is, the IP address of the STB).

11111 is the port number for the unicast stream.

The URL can also include additional attributes if required (see Media source attributes), for example:

```
udp://192.168.0.1:11111;audiopid=20
```

You should also ensure that the setting **SELECT_RTSP_STREAM_BY_PORT_ONLY** is set to **Y**.

UDP streams will be setup on the server to be sent to a particular port on the IP address of the STB. The required parameters in the URL are the source IP address (to correctly title traffic from the source only) and the destination port the content is addressed to.

E.1.5.1 Playing a unicast video stream

Make sure your streamer is sending data to the IP address of the STB, then use the following URL:

```
udp://<STB IP address>:<STB port>
```

For example, you have an STB at 192.168.1.103 and a PC running VLC at 192.168.1.4

The PC would stream to: 192.168.1.103 port 1234

The URL to play the stream would be:

```
udp://192.168.1.103:1234
```

E.2 Media source attributes

Attributes can be included in media source URLs to provide additional settings relating to the stream. In the case of audio, video and PCR PIDs, the values are normally autodetected. If you manually specify any PIDs, then this auto-detection system is disabled.

audiopid — Sets the audio PID value. Audio PIDs are normally auto-selected based on the audio language preferences. By setting a manual PID the default auto PID detection mechanism is disabled.

audiotype — The audiotype attribute sets the audio codec to use for the media. Possible types are:

```
aac-latm
aac-adts
ac3
mpeg112
```

Any codec not matching the above will default to **mpeg1l2**. This attribute is used when setting an audio PID manually using **audiopid**.

clock_abs an absolute time used for timeshift TV to specify where in the stream to play from, in RTSP format 20130522T113430Z.

EOMfreeze The **EOMFreeze** setting — which can be turned on by using the '**EOMfreeze=yes**' media source attribute or by calling **AVMedia.EOMFreeze(1)** — determines what happens to the video picture when you call **AVMedia.Play()** to start a new stream when video is already playing.

Normally the video picture will be cleared immediately and remain black until the new stream begins, but if **EOMFreeze** is set the final picture of the old video stream will be left frozen on the screen instead. (Note that if **AVMedia.Kill()** is used to stop a stream it overrides **EOMfreeze** so the picture is always cleared, regardless of the **EOMFreeze** setting.)

monomix Selects which audio channels are output according to the following values:

stereo both left and right channels (default).
left left channel only.
right right channel only.
monomix mix left and right channels together.

offset Number of seconds to offset into the stream when starting an RTSP asset.

password Password to log onto servers that require a password before they stream video.

pcr Sets the PCR PID value.

pcrpid Sets the PCR PID value. PCR PIDs are normally auto-detected. By manually setting a PCR PID the auto detection mechanism is disabled.

pmtpid Selects the program map table (PMT) which in turn identifies the component streams such as audio and video required to recreate a program.

progid Selects the program to play in a multiple program transport stream (MPTS), for example `igmp://239.255.250.2:11111;Progid=14`

rtpskip For IGMP, RTSP and UDP stream types, RTP headers can be skipped by adding the attribute **rtpskip=yes** to the URL.

rtspping The **rtspping** attribute sets the RTSP **ping** timeout value. The default value used is 30 seconds if no attribute is set. The RTSP **ping** is used as a 'heartbeat' by the client so that the server will continue to serve video.

The STB checks that data is being received from an RTSP server every second and records the last time that video data was received.

If the time since the last video data received reaches the **rtspping** threshold then a ping is sent to the server.

serverdata It is possible to define additional attributes specific to certain video server types using the **serverdata** attribute.

An example is for the Seachange server where supported serverdata extensions include "SeaChange-Version", "SeaChange-MayNotify" and "SeaChange-Server-Data" to set a device ID. This only applies to PPC products.

servertype Specifies the type of server to which the request is being made.

speed	For PVR builds that can be used to specify what speed the content should be played at. The argument needs to be of type `float`, for example, `1.0`.
src	Specifies the URL of a valid media source. For example, `src=igmp://239.255.250.1:11111`
subpid	Selects which subtitle language to use, if present, in the media source.
txtpid	Selects which teletext language to use, if present, in the media source.
type	This is used to define other media sources instead of video. Currently only **`MP3`** is defined which allows for the incoming media to be decoded as MP3 instead of video.
username	Username to log onto servers that require a username before they stream video.
videopid	Sets the video PID value, for example `igmp://239.255.250.2:11111;videopid-17`**.** Video PIDs are normally auto-detected. By setting a manual PID the default auto PID detection mechanism is disabled. Note that on IPTV single program transport streams are supported with one video PID.
videotype	Sets the video codec. Options include: **`h264`** **`mp42`** (MPEG4pt2/MPEG4-ASP/H.263) **`mpeg2`** Any other setting defaults to "**`mpeg2`**". This attribute is used when setting a vdeo PID manually using **`videopid`**.

E.3 HTTP Live Streaming (HLS)

E.3.1 Overview of HLS

External information about HLS can be found by following the link below:

https://developer.apple.com/resources/http-streaming/

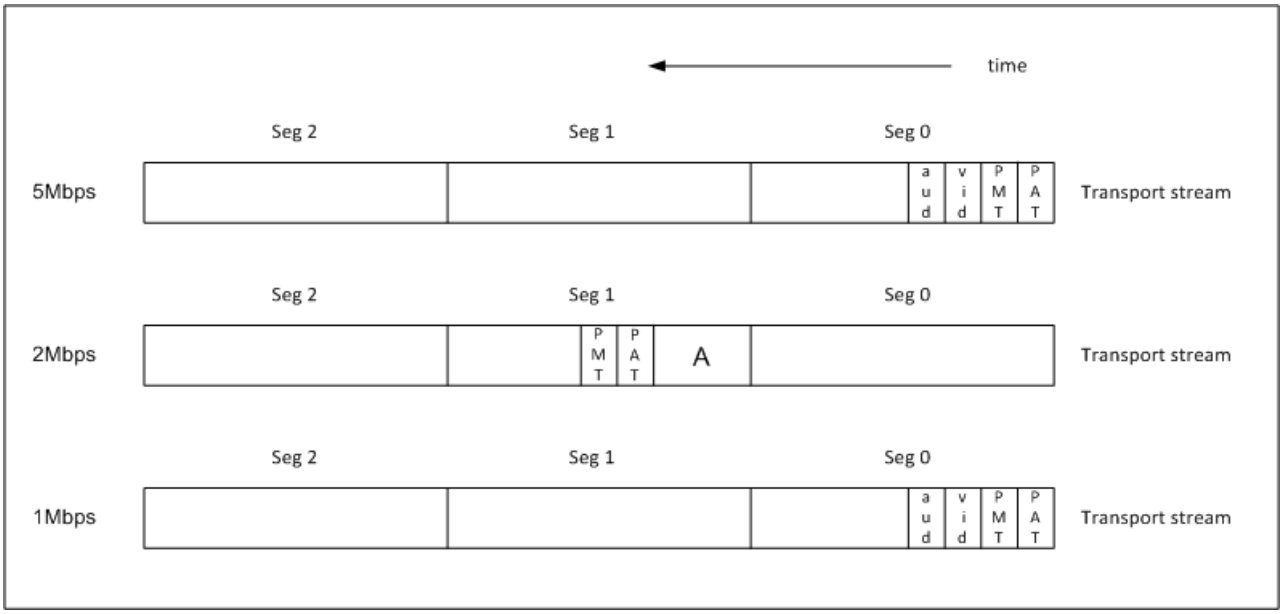
HLS allows client devices to fetch a *variant* or *non-variant* playlist (an index file) of transport stream segments to play in order. It uses H.264 compression, AAC audio, mpeg audio etc. The fetched stream contains playback indexing files as well as the transport stream.

The HLS server provides at least one non-variant playlist file. The non-variant playlist file is a list of URLs for the transport stream segments.

The stream consists of segments of a specific time (the HLS specification defines the 'target' duration of the segments) and theoretically the STB could cope with unlimited size segments because the file is streamed rather than downloaded.

The variant playlist can, for example, consist of several streams at different bit rates (5, 2 and 1 Mbps) as shown below. The individual streams are known as non-variant streams.

NOTE The maximum bitrate set for an HLS stream is lowered internally to allow for network overheads.



E.3.1.1 Switching between non-variant streams

Say for example that the current playback is segment 0 of the 5Mbps stream. If, during playback, the network slows for some reason, then the decoder will need to get the next segment of the HLS stream from a lower bitrate non variant stream, that is, segment 1 from the 2Mbps stream. To enable seamless switching without data loss, the timestamps across the streams must be identical.

E.3.1.2 Clean start

For a clean start to playback, the PAT and PMT are needed and should be at the front of the segment as shown in the 5Mbps stream. If they are positioned in a different part of the segment as shown in the 2Mbps stream, then on switching to this stream the decoder needs to find the PAT and PMT for correct playback. Any data before the PAT and PMT - shown as time A - is likely to be lost (at least the first 770k).

E.3.1.3 Stream change

On stream change, the STB notices that the PMT has changed, and playback will not be smooth until the STB finds the required PIDs for the new stream segment, after which it can continue playing the stream at the lower bitrate.

E.3.1.4 Playback

From the selection of streams offered in the variant playlist, convention is to start with the first stream (the stream the operator thinks is the best default bitrate will be placed first).

The maximum bitrate to be used can be specified in the playback URL at present.

The HLS start up mode defaults to 'best quality' and checks the available bandwidth for best quality - it will switch to another non variant stream if a higher quality (that is, greater bandwidth) stream is available, or if the requested one is not available. Any lower quality data already fetched is discarded.

The lowest bitrate stream may be audio only. Audio only streams are currently only supported by Minerva.

E.3.2 HLS playback using JMACX functions

To be able to play HLS content, the HLS mpegcontrol plugin must be present in the build (hls.ppg).

HLS streams are treated in the same way as any other stream type except that you add **`servertype=hls`** to the URL, for example:

`http://exampledomain.com/exampleplaylist.m3u8;servertype=hls`

Currently HLS Video-on-Demand and HLS live streams are supported, as are HLS variant playlists (a playlist of a stream encoded at different bitrates) and non-variant playlists (the URLs of the streams to be played). The HLS plugin initially downloads the last bitrate variant used, but if a faster bit rate is detected the HLS will move to it when it starts the next segment.

The following examples show the JMACX functions that can be used with HLS:

1. Playback of a *variant* or *non-variant* playlist using AVMedia.Play:

   AVMedia.Play("src=http://devimages.apple.com/iphone/samples/bipbop/bipbopall.m3u8;servertype=hls")

2. Set position in seconds using AVMedia.SetPos( )

Set position will have a resolution of the file segment size for the HLS stream being played. For example if file segments are 10 seconds long, the set position will be the beginning of the 10-second segment that contains your chosen value.

For example, if you set AVMedia.SetPos(127), the actual position set will be AVMedia.SetPos(120), that is, the start of the 120s to 130s segment.

3. Get position using AVMedia.GetPos( )

   Returns value in seconds, for example:

   var pos = AVMedia.GetPos( )

4. Get duration using AVMedia.GetDuration( )

   Returns duration in seconds, for example:

   var duration = AVMedia.GetDuration( )

5. Toggle between pause and play using AVMedia.Pause( ) and AVMedia.Continue( )

   AVMedia.Pause( )

   AVMedia.Continue( )

E.3.3 HLS Debug

For debug purposes:

- the client can automatically choose a variant playlist .
- you can manually fetch the variant playlist then get the URLs of the non-variant playlists and tell the HLS client to just play that non-variant playlist URL you want to debug.
- you can point the STB at the MPEG TS segment URLs, in turn, using tshttp.
- you can fetch the segment files yourself on a PC, concatenate them together and use vlc/FFplay on the PC, or tsfile on the STB.

Appendix F—Recovery from invalid video mode

In this Chapter:

This appendix gives information about invalid video modes and how to recover from them.

F.1 Invalid video modes

There are a number of ways in which an STB can be configured with the result that the user sees no output on their TV. The more common are:

- Using the composite, S-video, SCART or RF output when the box is configured for HD. In this case the output will only show a black screen until video starts playing. Downscaled video will be shown on these outputs but the middleware graphics will not.
- Using HDMI and having the box configured for a mode the TV does not support (some TVs do not support 480i or 576i over HDMI, for example).
- Using an S-video, Component or RF cable but having the box configured with the corresponding output turned off.
- Connecting to an NTSC TV when the box is configured for PAL, and to a lesser extent vice-versa (many PAL TVs will show something usable when receiving an NTSC signal even if there is no colour, but most NTSC TVs will not display a PAL signal at all).

F.2 Invalid video mode recovery

A simple method of video mode recovery is as follows:

- Powercycle the box, and while it is booting hold down the OK button.
- Initially the LED will not flash as the remote has not yet been read.
- The LED will start flashing as soon as the STB starts to read the remote. The STB will recognise the signal from the remote and will set the output to HDAUTO on boot up (if an HDMI is connected then the output will be set to HD and not SD).
- The box will be reconfigured and rebooted.
- The LED will stop flashing as the box reboots (Release OK).
- The STB should now be reconfigured and displaying video again.

The LED transitions can be used to describe when to hold the OK button and when to let it go. It transitions from lit to flashing to lit. The Restore button (OK in this case) can be changed if needed.

F.2.1 How it works

There is a script on **`/mnt/nv`** called **`rc.restore`**. This can be replaced or changed by customers if needed. This script is run just before the browser/middleware. It checks if a predefined key has been held down for more than two seconds while booting. If so, it will reset the resolution to **`HDAUTO`** as default.

However the resolution can still be changed by changing the rc script on flash. The setting **`NORFLASH.OUTPUT_RESOLUTION`** can be:

- **`HDNONE`** if no HDMI cable is connected.
- **`HD480P59`** if HDMI is connected and **`NORFLASH.TVSYSTEM`** is **`NTSC-M`**, **`NTSC-J`** or **`PAL-M`**.
- **`HD576P50`** if HDMI is connected and **`NORFLASH.TVSYSTEM`** is anything else.

All boxes always produce a composite output and all cable types have a composite signal available, so you will be able to get a picture by using this, though it may mean switching connections to the TV.

Appendix G—Video output formats

In this Chapter:

The following table shows the output format possibilities for the STBs listed, depending on the cables connected. The output format is normally set as part of the configuration (for example, via the Management pages or `libconfig`), but in some cases it is selected automatically.

The name of the setting or command used to change the output format depends on which configuration tool you are using.

When the HDMI cable is connected, the HD format is selected automatically.

In the following tables Composite is the same as CVBS and Component is the same as YPbPr.

G.1 A129, A140, H140, A540PVR, A150

Part no.	Cables	Output format configuration options
502-418	10 pin Mini DIN to SCART	Composite Composite, RGB Composite, S-video Composite, Component
502-594	10 pin Mini DIN to S-video and Composite 1RCA	Composite, S-video
502-419	10 pin Mini DIN to Component video and RGB 6 x RCA	Component, RGB
502-523	10 pin Mini DIN to Composite 3 x RCA	Composite
510-885	HDMI	HDMI (not A129)

Appendix H—Pairing an IR remote control with an STB

In this Chapter:

You can program an Amino IR remote control so that it can only control an STB that is configured to respond to it.

The Amino IR remote controls prefix a number to each STB command. This number is set to `000` by default, but you can program it to be another number in the range `001` to `015`.

All Amino STBs respond to commands prefixed with `0` but you can configure them to respond to commands prefixed with another number as well.

For example, a remote control can be set to prefix commands with `3`, and an STB can be configured to respond to commands prefixed with `3`.

- The remote control can only control STBs that expect `3` as a command prefix.
- The STB will respond to commands prefixed with `3` and also continues to respond to commands prefixed with `0` (so that all STBs respond to the default Amino remote control).

The following steps use the Configuration pages to set the IR brand code. This can also be set with the other configuration tools. See Chapter 6, "STB configuration pages for Opera 11 onwards" for details.

H.1 On the grey Amino IR remote control

1. Hold down the **STB** and **OK** buttons on the remote control until the STB button lights up permanently.
2. Enter the number that you want to set as the prefix code for this remote control. This must be a three-digit number in the range `000 - 015`.

   For example:

   `111`

   The remote control will now prefix all STB commands with this code.

> NOTE To revert to the default code (that is, to unpair the remote control and STB), repeat these steps and set the code to `000`.

H.2 On the AVC 'Boat' IR remote control

1. Hold down the **STB** button until the LED flashes twice
2. Enter `9 8 2`. The STB light should flash four times. If it flashes twice, repeat this part of the procedure. The `9 8 2` sequence unlocks/locks remote configuration programming.
   - 4 flashes = unlocked
   - 2 flashes = locked
3. Hold down the **STB** key until the LED flashes twice and continue holding it until it flashes twice again.
4. Enter the number that you want to set as the prefix code for this remote control. This must be a three-digit number in the range `000 - 015`.

H.2.1 Pairing the STB to the remote control

1. From the Amino IR keyboard, press Alt-M and then enter the Management password ('`leaves`' is the default), to access the Management pages.

Warning:
Amino strongly recommends that you change the default passwords when deploying our STBs.
See the Amino *Installation and Upgrade Guide* for information on changing these default passwords.

2. Open the IR configuration page.

NOTE For navigation on this page, it is easier to use the arrow keys on the Amino IR keyboard.

3. Enter the two-digit code that you programmed the remote control with, this is the **last two** digits of the 3-digit code. (The first digit indicates the frequency used).
4. This can also be set with `libconfig-set SETTINGS.IR_BRAND` "`11`".
5. Save the changes (the default password is `snake`).

   The remote control and the STB are now paired.

H.3 On the SRC ‘Willow’ IR remote control

1. Press **2** and **7** simultaneously for 3 seconds. During the initial 3 seconds, the LEDs will remain off. After 3 seconds the STB LED turns on.
2. Press the **GUIDE** key. The remote will perform a confirmation blink (200ms on/off repeated twice) and the STB LED will turn on.
3. Enter a 4 digit code, from the Pairing Code list below.
4. Only digit keys are allowed. When a digit is pressed, the STB LED will turn off for the duration of the key-press. Any other key press will result in an invalid sequence.
5. If the entered digits are valid, the remote control will issue a confirmation blink, and then return to normal Use mode.
6. If the entered digits are invalid, the remote control will issue an error blink (50ms on/off repeated 5 times) then return to normal Use mode.

The remote can be paired to individual STBs by programming with one of 16 Pairing Codes. The default pairing code will be 0100.

Pairing Code	Address Field
0100	0x00
0101	0x01
0102	0x02
0103	0x03
0104	0x04
0105	0x05
0106	0x06
0107	0x07
0108	0x08
0109	0x09
0110	0x0A
0111	0x0B
0112	0x0C
0113	0x0D
0114	0x0E
0115	0x0F

Appendix I—STBremoteconf stats command

In this Chapter:

This section describes the descriptors returned by the **`STBremoteconf stats`** command. The command returns a set of status values for the STB. This command is only useful via unicast.

In future versions the parameters that are currently included in the stats report may be changed, or further parameters may be added, depending on customer feedback.

I.1 Usage

```
./STBremoteconf <host> STATS
```

I.2 Example

```
./STBremoteconf 10.172.227.145 STATS
```

I.3 Terms explained

The STBremoteconf stats command returns information which can be accessed by looking at specific directories on the STB. The MAC address is returned, and also selected information from **`/proc/avcore, /proc/meminfo`** and from running an **`ifconfig`** command on the STB.

I.4 MAC address

The **`stats`** command returns the MAC address of the STB:

Term	Example value	Description
`MACADDRESS`	00:02:02:03:BD:97	The MAC address of the STB.

I.5 AVCore

The operating system collects and presents a range of useful statistical information whilst the STB is running. These are all presented in the pseudo-filesystem **`/proc/avcore/`** where they can be inspected. All counts and timestamps are unsigned 32-bit integer values unless otherwise indicated.

Timestamps are in 'jiffies', a Linux time unit (see the appropriate Linux documentation for a precise definition).

Many of the statistics are presented in a standard (three value) form which is the time of the most recent occurrence (in jiffies), the number of occurrences that have occurred during the playing of the current asset, and the total number of occurrences that occurred prior to the playing of the current asset.

Term	Example value	Description
The video decoder will determine whether to decode a picture for display, or to skip a picture if synchronisation requires it.		
`last_skip_timestamp`	`0`	The last video skip timestamp.

skips_in_this_movie	0	The number of skips in the movie that is currently playing .
total_skips	0	The total skips since last boot.
Sequence errors are reported by the video decoder if error recovery is required to handle an unsupported frame rate or when the specified dimensions for a sequence do not match the expected MPEG profile.		
last_serror_timestamp	0	The last sequence error timestamp.
serrors_in_this_movie	0	The sequence errors in this movie.
total_serrors	0	The total sequence errors since last boot.
TIME returns timing information about the current asset being played. Timestamps are in 'jiffies', a Linux time unit (see the appropriate Linux documentation for a precise definition).		
timestamp_of_movie_start	0	The timestamp of the start of the current movie.
current_time	76319	The current timestamp.
Audio PTS		
audio first pts	0	The timestamp of the first audio buffer overflow.
audio last pts (repeated below as **last_overflow_timestamp**)	0	The timestamp of the last audio buffer overflow.
AUDBUF returns information about how well audio buffering has been performing. The audio buffer statistics returned are specified as follows.		
last_overflow_timestamp	0	The timestamp of the last audio buffer overflow.
overflows_in_this_movie	0	The number of overflows for the current asset.
total_overflows	0	The total number of overflows since last boot.
last_underflow_timestamp	0	The timestamp of the last underflow.
underflows_in_this_movie	0	The number of underflows for the current asset.

total_underflows	0	The total number of underflows since last boot.
fullness	0	The fullness of the audio buffer.
The video buffer statistics returned are specified as follows.		
last_overflow_timestamp	0	The timestamp of the last video buffer overflow.
overflows_in_this_movie	0	The number of overflows for the current movie.
total_overflows	0	The total number of overflows since last boot.
last_underflow_timestamp	0	The timestamp of the last underflow.
underflows_in_this_movie	0	The number of underflows for the current movie.
total_underflows	0	The total number of underflows since last boot.
fullness	0	The fullness of the video buffer.
Video PTS		
topbit, lower 32bits	0:0	Retrieves the timestamp of the last video pts in a 33-bit presentation

I.6 Netstat

Displays generic net statistics of the host you are currently connected to:

Interface	Received bytes	packets	errs	drop	fifo	frame	compressed	multicast	Transmit bytes	packets	errs	drop	fifo	colls	carrier	compressed
lo	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0
eth0	438308	3116	0	0	0	0	0	2925	0	135	0	0	0		0	0

Where:

`Interface` is the type of interface used in which lo is the loopback or local interface, and eth0 is the external Ethernet port.

`Received bytes` is the total number of bytes received per second.

`packets` is the totaL number of packets received per second.

`errs` is the total number of errors occurring during received packets per second.

`drop` is the total number of packets that were dropped per second.

`fifo` is the total number of overrun errors in the received queue.

`frame` is the number of framing errors transmitted or received.

`compressed` is the total number of compressed packets received per second.

`multicast` is the total number of multicast packets received per second.

`Transmit bytes` is the total number of transmitted bytes.

`packets` is the number of packets transmitted per second.

`errs` is the total number of errors occurring during received packets per second.

`drop` is the number of packets that were dropped per second.

`fifo` is the total number of overrun errors in the transmit queue.

`colls` is the number of collisions that were detected per second.

`carrier` is the number of carrier errors that happened on transmitted packets per second.

`compressed` is the number of compressed packets transmitted per second.

I.7 Meminfo

Much of the information here is used by the **`free`**, **`top`** and **`ps`** Linux commands. The output of the **`free`** command is similar in appearance to the contents and structure of **`/proc/meminfo`**, although looking directly at **`/proc/meminfo`** gives more details:

	Description
`MemTotal`	Total usable RAM, in kilobytes (that is, physical RAM minus a few reserved bits and the kernel binary code).
`MemFree`	Is the sum of LowFree+HighFree (overall stat).

	Description
MemShared	The amount of shared memory.
Buffers	The amount of memory used by system buffers.
Cached	Memory in the page cache (disk cache) minus SwapCache
SwapCached	Memory that once was swapped out, is swapped back in but still also is in the swapfile (if memory is needed it does not need to be swapped out AGAIN because it is already in the swapfile. This saves I/O.
Active	Memory that has been used more recently and usually not reclaimed unless absolutely necessary.
Inactive	Memory which has been less recently used and is more eligible to be reclaimed for other purposes.
HighTotal	The total amount of memory in the high region. Highmem is all memory above (approx) 860MB of physical RAM. Kernel uses indirect tricks to access the high memory region. Data cache can go in this memory region.
HighFree	The amount of free memory of the high memory region.
LowTotal	The total amount of non-high mem memory. This is the memory the kernel can address directly. All kernel data structures need to go into low memory.
LowFree	The amount of free memory of the low memory region.
SwapTotal	Total amount of physical swap memory.
SwapFree	Total amount of swap memory free.
Dirty	The total amount of memory, in kilobytes, waiting to be written back to the disk.
Writeback	The total amount of memory, in kilobytes, actively being written back to the disk.
Mapped	The total amount of memory, in kilobytes, which have been used to map devices, files, or libraries using the mmap command.
Slab	The total amount of memory, in kilobytes, used by the kernel to cache data structures for its own use.
CommitLimit	Total amount of memory currently available to be allocated on the system.
Committed_AS	The amount of memory presently allocated on the system.
PageTables	The total amount of memory, in kilobytes, dedicated to the lowest page table level.
VmallocTotal	The total amount of memory, in kilobytes, of total allocated virtual address space.
VmallocUsed	The total amount of memory, in kilobytes, of used virtual address space.
VmallocChunk	The largest contiguous block of memory, in kilobytes, of available virtual address space.

I.8 Example of a stats output

```
    Got back: MACADDRESS: 00:02:02:03:bd:97
     (last_skip_timestamp skips_in_this_movie total_skips)
     0 0 0
     (last_serror_timestamp serrors_in_this_movie total_serrors)
     0 0 0
     (timestamp_of_movie_start current_time)
     0 76319
     0 0
     (last_overflow_timestamp overflows_in_this_movie total_overflows last_underflow_timestamp under-
flows_in_this_movie total_underflows fullness)
     0
     0 0 0
     0 0 0
     0
     (last_overflow_timestamp overflows_in_this_movie total_overflows last_underflow_timestamp under-
flows_in_this_movie total_underflows fullness)
     0 0 0
     0 0 0
     0
     (top_bit:other_32)
     0:0
     Inter-|   Receive                                                |  Transmit
      face |bytes    packets errs drop fifo frame compressed multicast|bytes    packets errs drop fifo
colls carrier compressed
    lo:       0       0    0    0    0     0          0         0        0       0      0    0    0    0     0       0
    eth0:  438308    3116    0    0    0     0          0      2925        0     135    0    0    0
0       0          0

             total:    used:    free:  shared: buffers:  cached:
     Mem:  31207424 18276352 12931072        0   176128 11268096
     Swap: 27258880        0 27258880
     MemTotal:        30476 kB
     MemFree:         12628 kB
     MemShared:           0 kB
     Buffers:           172 kB
     Cached:          11004 kB
     SwapCached:          0 kB
     Active:           2012 kB
     Inactive:        12956 kB
     HighTotal:           0 kB
     HighFree:            0 kB
     LowTotal:        30476 kB
     LowFree:         12628 kB
     SwapTotal:       26620 kB
     SwapFree:        26620 kB
```

Appendix J—Frequently Asked Questions (FAQs)

In this Chapter:

The following sections contain the answers to FAQs raised in support tickets. The FAQs are listed under various topic headings, and may appear under more than one if the FAQ crosses two topics. Note that these FAQs only relate to topics that are covered in this STB Configuration Guide: for other FAQs and the Amino Support Knowledgebase, please visit the Amino website (www.aminocom.com) and click on **Customer Support.**

J.1 Leds

Q. On the A140 STB, is it possible to switch off the LED when the STB is in the Off state, and turn it on to indicate it is in the ON state?

To change the default state for all boxes, modify the file in your **`imagecomponents.settings`** to include these 2 lines:

SETTINGS.**`INITIAL_LED "ON"`** or **`"OFF"`** as required

SETTINGS.**`STANDBY_LED "ON"`** or **`"OFF"`** as required

You can also change individual boxes to other settings by using tools such as STBremoteconf.

Refer to LED Configuration operation by STB product to see the effect of the **`INITIAL_LED`** and **`STANDBY_LED`** settings on LED operation.

J.2 STBremoteconf

Q. Where do I find the tools for STBremoteconf?

They are located in the **utils** directory of the software release. Follow the instructions in this guide to install and run the utility. See Using STBremoteconf.

Q. I am looking for information about how to monitor event logs from the Amino 140, by using Telnet etc.

You can use our STBremoteconf utility and its protocol. There are several documents on the support site explaining the benefits and limitations. See Using STBremoteconf.

There is also a lot of information outputted by the serial debug port and a lot of telemetry is available from the "/proc" filesystem under **`/proc/avcore`**. There are several documents on the support site describing each entry. See the Amino website (www.aminocom.com) and click on **Customer Support**.

J.3 Recovery image

Q. Is it possible to set a software URL for the recovery image in the STBimage?

It can be stored in NOR Flash with the parameter **`NORFLASH.SOFTWARE_URI`**, see SOFTWARE_URI.

J.4 IR and remotes

Q. I cannot get the IR passthrough to work.

There is a configuration setting required to enable the pass-through. From debug, enter the following:

```
libconfig-set SETTINGS.ENABLE_EXTERNAL_RECEIVER 1
```

You will need to restart the STB for this to take effect. See **`ENABLE_EXTERNAL_ RECEIVER`**.

Q. Can the 'willow' remote be programmed with macros?

No, the ‘willow’ remote does not support macros.

J.5 Miscellaneous

Q. Do Amino STBs support IGMPv3?

The Ax4x STBs can support either IGMPv2 or IGMPv3 (limited).

ALL v3 devices will drop back to v2 if they see a v2 message (the entire network will fall back to the lowest common level) and we have seen compatability issues when sending v3 messages on a v2 network.

By default the STBs are configured to use v2 to get over the issues seen above. You can override the default and raise the version to v3 using the command

```
libconfig-set IGMP_MAX_VER 3
```

Q. How do you set the current date and time in the system?

The time can be set in several ways depending on your requirements:

By default the STB will get the time information from the DHCP server it connects to. You can point the STB to a time server using the STB management pages. See Network on page 88.

If you have telnet or console access to the STB you can set it manually at the command line (not normally required). See **TIME_SERVER**.

Q. Is there a way to modify the images that appear when upgrading?

You can replace these images with your own but you must follow certain guidelines. See Replacing the default graphics with your own on page 67.

Q. The STB now shows a graphic while the kernel is loading. Is this graphic configurable?

No, this graphic is the bootloader graphic and cannot be configured because it is in a proprietary format. However, we can create a bootloader graphic based on a graphic that you provide, but it must meet the following conditions:

- a gif at the lowest resolution that will provide an acceptable graphic
- a minimum amount of colours
- no animations
- a black background rather than transparency.

When you design or provide the graphic, note that it will be displayed centred against a black background.

Q. Can I rotate the video plane as well as the graphics plane?

No, only the graphics plane can be rotated.

Q. How do I enable or disable cookies?

Cookies can be enabled or disabled by using the **BROWSER_ENABLECOOKIES** setting.

Q. How do I enable subtitles?

Subtitles or closed captions can be enabled by using the **ENABLE_SUBS_OR_CC** setting. Alternatively, if you are using the ‘Willow’ remote control unit, you can go into the management pages and enable Subs/CC by pressing the menu key, clicking on "Subtitles/CC" and then clicking enable.

Q. How do I configure the STB for 1080p and what is meant by the STB "can de-interlace content for progressive display"?

The output resolution can be configured using the OUTPUT_RESOLUTION setting. While the STB does not support a direct 1080p input (because the video hardware does not have the bandwidth for that much data), the interlaced 1080i video can be output in 1080p format. So a TV which supports 1080p will have a valid video signal - the difference is that it will have less video information contained in it than a native 1080p decoded signal.

1080i (interlaced) content means that the video content is sent with the odd and even video lines interlaced to build up one complete frame of video, so depending on the frame rate in use and the content, the video quality will be acceptable. In most normal situations it is unlikely you will notice any difference, but with high motion content you may see artifacts/distortion in the video if the frames contain drastically different content for much of the time.

Q. My IP camera outputs a stream as RTSP - will the A140 support this?

The A140 will support RTSP via UDP (but not RTP).

J.6 Using VLC

(http://www.videolan.org/vlc/)

Q. I have errors or gliches on my stream when played through the STB, but they play without problems when using VLC .

Video Lan Client (VLC) is a popular free player/streamer that can be used to give a quick confirmation that a stream is available and what the content is (provided it is not encrypted). It will show teletext and subtitles and can be used to quickly stream out a file to confirm it is the correct video content etc.

However, because VLC uses the resources of the computer which it is running on, it is not a good comparison against an STB and you may think that because VLC plays the video then the STB should also play it. This is **not** the case and VLC can often buffer and play non-conforming video that an STB cannot.

It is simple to use and free, so it makes a useful addition to the test tools, but it should be used for quick checks rather than proper diagnosis.

J.7 Hard Disk Drives and recording

Q. I have a hard disk drive which I want to connect to my STB via the USB port to use for recording. Can I just plug it in?

You will need to do some configuration before connecting a hard disk drive to an STB. The hard disk drive must be validated using our validation tools and entered into the `PVR.map` file (formerly the `PVR.usermap` file) to be recognised and formatted by our STB. For more information about using a HDD for recording, see the Amino *JMACX API specification* AM-000502-TC and the *USB Hard Disk Performance Test Specification and Procedure*, document number AM-003378-TE.

Q. How many hours of video can I record on the PVR with a particular video bitrate?

It is difficult to state a specified time because it depends on parameters such as compression, bitrates, network overheads etc. for which each ecosystem is different. It could be 20 hours to 65+ hours. MPEG-4 (h.264) SD could perhaps be 80 hours depending on how low the bitrate is.

As a *rough* guide for A540 recordings with a 250GB HDD:

8 Megabits (Mb) = 1 MegaByte (MB)

1 GigaByte (GB) = 1024 MB

230 GB = 235,520 MB

So if an HD stream is playing @ 8 Mbps then 1 MB of space is written per second onto the HDD which is equivalent to 60 MB per minute.

235 GB/60 = 3,925.3 minutes.

3,925.3 minutes/60 = 65.4 hours.

So the amount of video in hours that can be recorded is approximately 65 hours of HD or 130 hours of SD (SD streams are up to 4 Mbps, so for this example the HD hours have been doubled).

J.8 Debug

Q. Is there a way to obtain the debug logs from the STB without using the debug cable, for when the STB is in a remote location?

There are three options for remote log retrieval:

1. use the LOGREAD option with STBremoteconf. This will retrieve the circular log buffer.
2. ssh in, and run the command logread. this will retrieve same as 1)
3. set up remote syslog, so the STB outputs its syslog over the network to a syslog daemon on another server.

Index

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W