

# **Testing Guide For UniAud 32**

**Version 1.2**

**October 3, 2022**

# Table of Contents

<b><u>BACKGROUND</u></b> .....	<b>3</b>
<u>INSTALLATION OF TEST VERSIONS</u> .....	3
<u>TESTING TOOLS</u> .....	4
<u>PMUnimix</u> .....	4
<u>Unimix.exe</u> .....	5
<u>PCI.exe</u> .....	6
<u>Testlog.EXE</u> .....	7
<b><u>WORKAROUNDS</u></b> .....	<b>8</b>
<u>LOW SOUND VOLUME</u> .....	8
<u>COLD BOOT AS OPPOSED TO WARM BOOT</u> .....	8
<u>AUTO-MUTE</u> .....	9
<b><u>FILING DEVELOPER REPORTS</u></b> .....	<b>10</b>
<b><u>ACKNOWLEDGEMENTS</u></b> .....	<b>10</b>
<b><u>GUIDE VERSIONS</u></b> .....	<b>10</b>

## **Testing Guide - UniAud32 Driver**

**Version 1.2 – October 3, 2022**

**Compiled by Mark Szkolnicki (Mark@paladinenvironmental.com)**

### **BACKGROUND**

The UniAud driver is a project designed to create one driver which is compatible with all sound chipsets and cards, without requiring a specific driver for each item of equipment installed in the computer system. It is currently based on development of a similar driver within the Linux community.

The following Wiki has been written to provide guidance as to tools and procedures available to both determine and test functionality of the driver, to allow developers to enhance the existing driver for use in the on-going development of the ArcaOS Operating System and provide functionality to previous existing OS/2 and eComStation (eCS) systems.

Current development of advanced versions of UniAud32 for our systems has been undertaken by Paul Smedley, with assistance from Steve Levine.

### **INSTALLATION OF TEST VERSIONS**

To beta test development versions of UniAud32, an existing installation of the driver, which is included with ArcaOS, must be installed (refer to the ArcaOS installation process for information). For other existing legacy OS/2 and eCS systems refer to installation procedures for the respective system.

The driver is normally installed in the directory X:\MMOS2, where X is the installation drive for the operating system. Lines related to UniAud are also added to the CONFIG.SYS file on the installation drive as follows:

```
DEVICE=X:\MMOS2\UNIAUD32.SYS  
DEVICE=X:\MMOS2\UNIAUD16.SYS /C
```

Beta Test versions of the UniAud32 driver can be installed in the following manner:

- 1 Before installing the development UniAud32 driver, be sure to backup the existing UniAud32.sys driver, either by copying it to another location or renaming it, so it can be restored, as required
- 2 Unzip the appropriate UniAud32 package provided by the developer to a directory
- 3 Usually two folders with drivers are created called RELEASE and DEBUG
- 4 Open the folder containing the driver you wish to install, and copy the development version of UniAud32 (UniAud32.sys, UniAud.sym, and Uniaud32.map) to X:\MMOS2
- 5 Reboot the system

**NOTE:** UniAud16.sys **does not** need to be removed from X:\MMOS2 or commented out in CONFIG.SYS, in order to test the system. Doing so may cause problems to occur within your ArcaOS, eCS or OS/2 system

The following switch may be added to UniAud32 in CONFIG.SYS, as required, if more than one chipset / sound card is present and active in the system (information of this nature may be obtained from PMUnimix - see below):

DEVICE=D:\MMOS2\UNIAUD32.SYS /a:x

where x is the appropriate number (1,2,3 etc.) of the sound card / chipset you wish to use (only one sound card / chipset may be enabled at a time – if you only have one sound card chipset this switch is not necessary – if you are testing the sound card chipset marked “0” this switch is also not necessary).

## **TESTING TOOLS**

The following 4 software tools have been found useful to undertake testing of this driver:

### **PMUnimix**

**Most Current Version** - 0.8.2

**Available at:** Hobbes File Repository

(<https://hobbes.nmsu.edu/?dir=/pub/os2> as PMUnimix\_0-82.zip)

PMUnimix is a visual front end program which provides volume information settings for Unimix.exe, which is located in X:\MMOS2. These settings are generated by the sound card(s) / chipsets being used and passed to Unimix.exe. These software settings are used by UniAud32 to adjust sound volume within the operating system.

PMUnimix can both enable and adjust the various sound controls related to Unimix. Various sound volume sections may be present in this display, based on the information obtained from the sound card or chipset. Various sections may be enabled by clicking on them, and sliders can be used to adjust the volumes setting as necessary.

## **Installation**

After unzipping the PMUnimix archive, the file(s) contained in it may be installed to any directory. (NOTE: if more than one ArcaOS / eCS / OS/2 partition is present on your system, it is recommended to place a copy of PMUnimix in a directory on each separate installation partition if you plan to do testing on one partition only).

If you want, you may create a program object for PMUnimix on the desktop or in some other location.

## Use

When PMUnimix is run for the first time it will create two files in the directory it is installed in:

PMUnimix.ini  
PMUniMixDebug.txt

These files are generated based on the state of the system at the time the program is run. The first contains the sound volume settings present the first time it is run, and subsequently after volume setting are adjusted, and will retain the settings you have adjusted for subsequent sessions.

PMUniMixDebug.txt contains information on the adapters recognized by Unimix.exe within your system. If a switch (/a:x) is not present in the UniAud32 line of CONFIG.SYS, PMUniMixDebug.txt will report all sound cards / sound chipsets present in the system, but PMUnimix will only address the sound card / chipset designated as "card number 0"

## Testing

If UniAud32 is operating correctly with the sound chipset or card, the sliders on the visual display should adjust the volumes generated by the sound equipment being used. If not, including this information in any report may assist the developer in the debugging process. Also, if PMUniMix is not reporting all the normal volume sections present within the visual display, this may also assist the developer.

## Unimix.exe

**Most Current Version** - included with ArcaOS / OS2 / eCS, based on the version installed  
**Available at:** present in the X:\MMOS2 directory on the installation partition

In addition to the volume information which PMUnimix shows (see above), Unimix can also generate the following information, separately:

Command line mixer for UNIAUD32. Version 1.1

Usage:

-dev<num>	- card to work (default=0)
-id<num>	- use control num (for list or set value). Begins from 1
-cnt<num>	- set value for count number <num> in control. Begins from 0
-val<num>	- set value for control
-card	- show card info
-pcms	- show PCM instances info
-list	- full (id,name,bounds,value) control(s) list

-names	- just control(s) name(s) and ID
-get	- get value switch. uses with -id and/or -cnt
-powerget	- get power state
-powerset<num>	- set power state
-save<file>	- save dump of all values of all controls to file
-load<file>	- load dump from file to all values of all controls

All arguments are processed in order. -dev, -id, and -cnt simply set a value that is retained until changed. All other arguments perform a command.

Example: unimix -id5 -cnt0 -val15 -cnt1 -val16 -list  
sets id5 cnt0 to 15, then sets id5 cnt1 to 16, then lists id5

## Use

Unimix commands above can be used from a command prompt under ArcaOS. As detailed information presented on the command line may scroll off the screen before reading, the switch | More may be added to the command prompt to slow the display of information, or the key Pause / Break on your keyboard may be used.

## PCI.exe

**Most Current Version** - v1.0.4 vkA included with ArcaOS

**Available at:** present in the X:\sys\install\DETECT directory in the installation partition (also in Hobbes File Repository (<https://hobbes.nmsu.edu/?dir=/pub/os2>))

PCI is a program written by Craig Hart which can provide detailed operating information related to all components installed in the computer system, including sound card / chipset components. This may be useful to developers needing additional information about the system, card or chipset in question.

## Use

Using a Command Prompt, change to the directory above (NOTE: the directory may also be included in the PATH statement in CONFIG.SYS for ease of use).

Typing PCI at the command prompt will generate a report of all components within the system.

Specific reports may also be generated using the following options:

Usage: PCI [-Option ..]    [] indicates optional parameter

-H : Use direct hardware access (instead of the BIOS) to retrieve PCI Info  
May be required for accurate reporting on Intel 430FX chipset+Award BIOS

- D : Do a hex-dump of each device's entire configuration space
- S : Create a brief, summary report only; only devices and IRQs listed
- T : Disable the test ROM IRQ Routing Table function
- P : Enable display of PCI slot routing data
- I : Installer mode: produce raw data dump (for use with auto-setup programs)
- R : Draw a Tree of Busses, Devices and Device Functions
- ? : Displays this help screen!
- N : Do not write to registers (PM readout, ExpRom size)
- 5 : keep textmode
- 7 : Display tree using ASCII chars
- M : Output HTML

PCI Supports generating reports to a file or printer using MS-DOS pipes; i.e.

PCI -D > REPORT.TXT (Save report to file), PCI > LPT1: (Print report)

## Testing

If UniAud32 is not operating correctly, including specific information generated by PCI in any report may assist the developer in the debugging process. PCI can also be used for information purposes by the tester, to expand on the limited information generated by the PMUnimix debug report.

## Testlog.EXE

**Most Current Version** - v3.5.0 (v3.4.6 is included with the latest version of ArcaOS)

**Available at:** V3.4.6 is present in the X:\sys\install\DETECT directory in the installation partition (the latest version is available at <https://88watts.net/dlfile.php?F=Testlog>)

Written by David Azarewicz, Testlog creates a log file that you can submit with your test results when reporting a problem with a driver. This program collects all the data about your system and the driver under test, and puts it into a single log file with appropriate dividers between the sections in the file.

## Use

Opening a command prompt, typing testlog uniaud will generate a report providing system information related specifically to the UniAud driver, for use by a developer. Prior to generating the report, you will be asked to provide a summary description of the problem encountered which will be included in the log file generated. Once this is done the report will be generated and placed in the directory X:\var\temp of the operating system partition, where X: is your operating system partition. The log file may be forwarded as an email attachment or posted at the request of the developer

## **WORKAROUNDS**

During testing, a number of workarounds have been found to problems which may be encountered – prior to filing a developers report, you may wish to try these workarounds, to see if they solve a problem you may have encountered.

### **LOW SOUND VOLUME**

Lack of sound may be a result of sound being extremely quiet but still present within MMOS2, instead of the driver not working. If increasing sound volume through the Volume object within the Multimedia folder (if present) does not work, a manual workaround suggested by a previous tester may product sound, or not.

To test:

- 1 Go to the Operating System partition on your system and find the MMOS2 directory
- 2 Find the file MMPM2.INI (this is a plain text file, as opposed to MMPM.INI which requires an INI editor and does not contain the information to be edited)
- 3 Right click on this file and at the Menu click on OPEN AS – either Text View or Text Editor
- 4 Find and change the following entries:

Change `mastervolume=77` to `mastervolume=100`

Change `PARMSTRING=BALANCE=50,VOL=80,INPUT=LINE,GAIN=70,OUTPUT=SPEAKER`

To `PARMSTRING=BALANCE=50,VOL=100,INPUT=LINE,GAIN=70,OUTPUT=SPEAKER`

### **COLD BOOT AS OPPOSED TO WARM BOOT**

Lack of sound may also occur with certain chipsets and UniAud32, if the operating system partition is not started from a cold reboot (ie. The system using ArcaOS is totally shut and powered down, and then rebooted by powering up)

If sound is present and then this occurs, in the case of a Warm Boot of the Operating System partition (ie. Rebooting the same OS partition without shutting and powering down the system totally), it most likely will be restored by simply shutting down the system totally, then powering the system up again.

In the case of a warm reboot from another operating system partition on your system, to the partition you are testing (eg. from a Windows, Linus, or an additional ArcaOS partition), sound may be present after the Warm Reboot. If it is not, shut down the system totally and do a Cold Reboot – most likely sound will be restored.



## **AUTO-MUTE**

Certain sound chipsets have been found during testing, under UniAud32, to be functioning perfectly, based on data shown graphically using PMUniAud, or in other ways, but produce no sound, audibly.

A workaround to this may be to disable a setting called “Auto-Mute” which has been found to be enabled as default in the drivers from which our UniAud drivers are derived.

To do this, try the following:

- 1 Open a Command Prompt window, in the operating system partition containing the driver.
- 2 In the window type “Unimix –names” (without the parentheses)
- 3 If the window scrolls too fast, use the “pause break” key on your keyboard or |more at the end of the prompt, to pause the initial output, then press the Enter key to view the subsequent output
- 4 Find the id number for your sound chipset or adapter which refers to “Auto-Mute” (Note: this may not be present based on the card / chipset used – if so, this workaround may not work)
- 5 Note the ID number (referred to as “x” below – enter this value where indicated)
- 6 From the command prompt type “unimix –idx -val0=0”
- 7 A message should appear indicating success.
- 8 Try a multimedia program or perform an action which generates a system sounds to see if sound has been enabled

If you have more than one sound card or chipset present in your system, you may need to the switch statement –val0 to –val1 or –val2 instead, based on whether it is sound adapter 0, 1, 2, etc., to do this You can refer to this value in the file PMUniMixDebug.txt, in the directory for PMUnimix, if you have this testing tool installed.

Upon reboot, the effect of this workaround may disappear, and need to be re-enabled

You can have what you enter be entered automatically every reboot by performing the following:

- 1 After performing the actions listed above and confirming sound is present, open and type into the command prompt the following - unimix –save**somefilename** (choose a name for this file)
- 2 A file with the name you chose will be created in the root directory of the Operating System partition you are using
- 3 If you already have the file STARTUP.CMD in your root directory, you do not need to create it again – if you do not have one yet, the easiest way to create it is to go to the Templates folder created by ArcaOS /eCS/OS2 originally, and drag a copy of the template file Command Line.CMD to the desktop
- 4 With the new file open, or by opening the exiting STARTUP.CMD type the following into the file

`unimix -loadsomefilename` (where somefilename is the name of the file you created in 1) above)

If you want to have the command window to close automatically after executing the command file also add “exit” (without parentheses) just below the `-load` line (if it is already present in `STARTUP.CMD`, place the new line above the exit statement).

- 5 Save a new file as `STARTUP.CMD` or save the existing `STARTUP.CMD`.
- 6 If you created the new file on the desktop, move it to the root directory of the Operating System partition you are using.
- 7 Reboot the system – if everything is entered correctly the Auto-Mute should be disabled automatically at boot up

## **FILING DEVELOPER REPORTS**

The surest way to get reports of problems to the developer is to open a ticket at <https://trac.netlabs.org/uniaud>.

As required or as requested, include any report files generated, from the test programs mentioned previously, as attachments

## **ACKNOWLEDGEMENTS**

**Paul Smedley** – Thanks to Paul for taking on the job of producing these advanced versions of UniAud32 for this operating system, trying to keep in line with the latest Linux versions

**Arca Noae** – for continuing development of this operating system

And a special thanks to all members of the community who have undertaken Beta Testing of the UniAud drivers and have provided suggestions to improve the testing experience

## **GUIDE VERSIONS**

1 – First release of this Testing Guide

1.1 – Some slight modifications and corrections in the text. Added reference to Testlog 3.5.0 with a link location, as well as reference to Developer Reports being filed at <https://trac.netlabs.org/uniaud>.

1.2 – Added comments related to not deleting or altering UniAud16. Added a Workarounds section, with tips to enhance the testing experience. Added a Table of Contents at front of the guide