

README.md

build passing

## PUAE LIBRETRO

Based on PUAE 2.6.1, git commit 0186c1b16f7181ffa02d73e6920d3180ce457c46

Credits to:

- Mustafa 'GnoStiC' TUFAN
- Richard Drummond, <http://www.rcdrummond.net/uae/>

This port was based at start on PS3 version E-UAE 0.8.29-WIP4 release 8 (so also credits to Ole.)

E-UAE is based on the work of dozens of contributors including Bernd Schmidt (the original author and maintainer of UAE), Bernie Meyer (the author of the x86 JIT compiler), Toni Wilen (the current maintainer of WinUAE), and many more.

## Default controls

RetroPad button	Action
D-Pad	Joystick
Left Analog	Mouse
Right Analog	Mouse
B	Fire button 1 / Red
A	Fire button 2 / Blue
X	Space
L2	Left mouse button
R2	Right mouse button
Select	Toggle virtual keyboard

Keyboard key	Action
F12	Toggle statusbar
RControl	Switch between joystick/mouse

## Virtual keyboard controls

Button	Action
B / Enter	Keypress
A	Toggle transparency
Y	Toggle CapsLock
X	Toggle position
Start	Press Return

Long press for sticky keys. Stickyng the third key will replace the second.

## ↪ Joyport control

Some games use mouse instead of joystick. D-Pad can be switched between joystick and mouse control in several ways:

- Use the core option: Quick Menu -> Options -> RetroPad Joystick/Mouse
- Bring up the virtual keyboard with `Select` button, and press the key labeled `J/M`
- Press the default keyboard shortcut `Right Control`
- Assign `Switch Joystick/Mouse` to any RetroPad button under `Quick Menu -> Options`

## ↪ Configuration

To generate the temporary uae configuration file the core will use the core options configured in RetroArch.

The following model presets are provided:

Short	Long	Chipset	Memory
A500OG	Amiga 500	OCS	0.5MB Chip RAM
A500	Amiga 500	OCS	0.5MB Chip RAM + 0.5MB Slow RAM
A500+	Amiga 500+	ECS	1MB Chip RAM
A600	Amiga 600	ECS	2MB Chip RAM + 8MB Fast RAM
A1200OG	Amiga 1200	AGA	2MB Chip RAM
A1200	Amiga 1200	AGA	2MB Chip RAM + 8MB Fast RAM
A4030	Amiga 4000/030	AGA	2MB Chip RAM + 8MB Fast RAM
A4040	Amiga 4000/040	AGA	2MB Chip RAM + 8MB Fast RAM
CDTV	Amiga CDTV	ECS	1MB Chip RAM
CD32	Amiga CD32	AGA	2MB Chip RAM

Short	Long	Chipset	Memory
CD32FR	Amiga CD32	AGA	2MB Chip RAM + 8MB Fast RAM

The configuration file is generated at launch and at core restart.

## 🔗 Kickstart ROMs

The following Kickstart ROMs are required in RetroArch `system` directory:

*It is critical to use ROMs with the correct MD5, otherwise the core might not start!*

*The core has a built-in AROS fallback Kickstart, which is used when the real Kickstart is not found. It can be compatible enough for some A500 games.*

Amiga Forever BIOS files must be renamed accordingly.

System	Version	Filename	Amiga Forever	Size	
A500	KS v1.2 rev 33.180 !	<b>kick33180.A500</b>	amiga-os- 120.rom	262 144	85ad74194e87c0f
A500	KS v1.3 rev 34.005	<b>kick34005.A500</b>	amiga-os- 130.rom	262 144	82a21c1890cae84
A500+	KS v2.04 rev 37.175	<b>kick37175.A500</b>	amiga-os- 204.rom	524 288	dc10d7bdd1b6f4
A600	KS v3.1 rev 40.063	<b>kick40063.A600</b>	amiga-os- 310- a600.rom	524 288	e40a5dfb3d017b
A1200	KS v3.1 rev 40.068	<b>kick40068.A1200</b>	amiga-os- 310- a1200.rom	524 288	646773759326fb
A4000	KS v3.1 rev 40.068	<b>kick40068.A4000</b>	amiga-os- 310- a4000.rom	524 288	9bdedde6a4f335
CDTV	CDTV extended ROM 1.00	<b>kick34005.CDTV</b>	amiga-os- 130-cdtv- ext.rom	262 144	89da1838a24460

v1.00

Amiga

(!) Kickstart v1.2 only needed for WHDLoad Arcadia games.

For CD32 you need either separate ROMs (Kickstart + extended ROM) or the combined ROM:

System	Version	Filename	Amiga Forever	Size	
CD32	KS + extended v3.1 rev 40.060	<b>kick40060.CD32</b>		1 048 576	f2f241bf09416
		OR			
CD32	KS v3.1 rev 40.060	<b>kick40060.CD32</b>	amiga- os-310- cd32.rom	524 288	5f8924d013dc
CD32	Extended ROM rev 40.060	<b>kick40060.CD32.ext</b>	amiga- os-310- cd32- ext.rom	524 288	bb72565701b

## Resolution and rendering

These parameters control the output resolution of the core (default bolded):

Name	Values
Video Standard	<b>PAL 50Hz</b> , NTSC 60Hz
Video Resolution	<b>Automatic</b> , Low, High, Super-High
Video Line Mode	<b>Automatic</b> , Single Line, Double Line
Aspect Ratio	<b>Automatic</b> , PAL, NTSC

With these settings all the standard resolutions are available:

PAL 50Hz Resolution	Description
360x288	Lores
720x288	Hires Single Line
720x576	Hires Double Line
1440x288	SuperHires Single Line

PAL 50Hz Resolution	Description
1440x576	SuperHires Double Line
NTSC 60Hz Resolution	Description
360x240	Lores
720x240	Hires Single Line
720x480	Hires Double Line
1440x240	SuperHires Single Line
1440x480	SuperHires Double Line

When using low resolution mode, rendering will be halved horizontally and forced into "**Single Line**" mode. Scaling shaders looks great, but high resolution games and Workbench are badly rendered.

When using high resolution "**Double Line**" mode, rendering will be doubled vertically. It is compatible with high resolution games and Workbench, but scaling shaders will look ugly. "**Double Line**" shows interlaced fields separately (weave) and is suited for deinterlacing shaders.

When using high resolution "**Single Line**" mode, rendering is presented as is. It delivers the best of both worlds, and looks great with high resolution games, Workbench and shaders. "**Single Line**" combines interlaced fields into one field (bob), which will make high resolution images blocky and jittery.

- Automatic "Resolution" defaults to "**Hires**" and selects "**SuperHires**" when needed (practically only in Workbench and Super Skidmarks)
- Automatic "Line Mode" defaults to "**Single Line**" and selects "**Double Line**" on interlaced screens

## 🔗 Model overriding

You can force a specific model if a game needs one (AGA games for instance) either by the "Model" core option or by file path tags.

The "Model" core option at "**Automatic**" will default to A500 when booting floppy disks, A600 when booting hard drives, and CD32 when booting CD images.

The whole path (filename and directory) will be searched for the following tags if the model is "**Automatic**":

Floppy	HD/LHA	CD	String	Result
--------	--------	----	--------	--------

Floppy	HD/LHA	CD	String	Result
x	x		(A500OG), (512K)	Amiga 500 (0.5MB Chip RAM)
x	x		(A500), OCS	Amiga 500 (0.5MB Chip RAM + 0.5MB Slow RAM)
x	x		(A500+), (A500PLUS)	Amiga 500+ (1MB Chip RAM)
x	x		(A600), ECS	Amiga 600 (2MB Chip RAM + 8MB Fast RAM)
x	x		(A1200OG), (A1200NF)	Amiga 1200 (2MB Chip RAM)
x	x		(A1200), AGA, CD32, AmigaCD	Amiga 1200 (2MB Chip RAM + 8MB Fast RAM)
x	x		(A4030), (030)	Amiga 4000/030 (2MB Chip RAM + 8MB Fast RAM)
x	x		(A4040), (040)	Amiga 4000/040 (2MB Chip RAM + 8MB Fast RAM)
		x	CDTV	Amiga CDTV (1MB Chip RAM)
		x	(CD32), (CD32NF)	Amiga CD32 (2MB Chip RAM)
		x	(CD32FR), FastRAM	Amiga CD32 (2MB Chip RAM + 8MB Fast RAM)
x	x	x	NTSC, (USA)	NTSC 60Hz
x	x	x	PAL, (Europe) !	PAL 50Hz
x			(MD) !!	Insert each disk in different drives
x	x	x	(CE)	Force CPU Cycle-exact

- (!) Additional tags: (Denmark), (Finland), (France), (Germany), (Italy), (Spain), (Sweden)
- (!! ) Maximum 4 disks

Example: When launching "Alien Breed 2 AGA.hdf" or "AGA/Alien Breed 2.hdf" the model will be Amiga 1200.

Note: **CD32** and **AmigaCD** are a bit misleading, since they have nothing to do with actual CDs. They are for automatically selecting the appropriate model with certain WHDLoad slaves and AmigaCD-to-HDF conversions.

## 🔗 Disk images, WHDLoad and M3U support

Supported formats are:

- **ADF, ADZ, IPF, DMS, FDI** for floppy disk images
- **ISO, CUE, CCD, NRG, MDS** for compact disc images
- **HDF, HDZ, LHA** for hard drive images
- **M3U** for multiple image playlist
- **ZIP** for various content (FD, HD, CD, WHDLoad)

When launching these files the core will generate a temporary configuration file in RetroArch `saves` directory and use it to start the emulation.

## 🔗 Floppy drive sounds

The core has embedded internal floppy drive samples. External sound samples have to be copied from [https://github.com/libretro/libretro-uae/tree/master/sources/uae\\_data](https://github.com/libretro/libretro-uae/tree/master/sources/uae_data) into a directory named `uae_data` or `uae` in RetroArch `system` directory.

## 🔗 IPF support

Most full-price commercial Amiga games had some form of custom disk format and/or copy protection. For this reason, most commercial Amiga games cannot be stored in ADF files unaltered, but there is an alternative called Interchangeable Preservation Format (IPF) which was specifically designed for this purpose.

IPF support is done through CAPSIMG library. To enable it you have to put the dynamic library called `capsimg.dll` (Windows) or `capsimg.so` (Linux, macOS) in RetroArch `system` or `executable` directory.

Compatible CAPSIMG libraries for Windows, macOS and Linux can be found at <http://www.softpres.org/download> and <https://fs-uae.net/download#plugins>

Compatible CAPSIMG libraries for Android can be found at <https://github.com/rsn8887/capsimg/releases/latest>

Please be aware that there are 32-bits and 64-bits versions of the library. Choose the one corresponding to your RetroArch executable.

## ZIP support

ZIPs are extracted to a temporary directory in `saves`, bypassing the default frontend extraction. The temporary directory is emptied but not removed on exit. ZIP is not repacked, which means saves and highscores are lost.

This allows:

- Automatic M3U playlist generation of all files
- The use of zipped images in M3Us
- Hard drive images will be treated one by one and only the first file found is selected for launch
- If no disk/drive images are found, the ZIP will be treated as a directory

## M3U support

When you have a multi disk game, you can use a M3U playlist file to be able to change disks via RetroArch Disc Control interface.

A M3U file is a simple text file with one disk per line (Wikipedia).

Example:

```
Simpsons, The - Bart vs. The Space Mutants.m3u
```

```
Simpsons, The - Bart vs. The Space Mutants_Disk1.adf  
Simpsons, The - Bart vs. The Space Mutants_Disk2.adf
```

Path can be absolute or relative to the location of the M3U file.

When the game asks for it, you can change the current disk in the RetroArch "Disc Control" menu:

- Eject the current disk with "Eject Disc"
- Select the right disk index with "Current Disc Index"
- Insert the new disk with "Insert Disc"

By default, RetroArch will display the filename (without extension) of each M3U entry when selecting a disk via the `Current Disc Index` drop-down menu. Custom display labels may be set for each disk using the syntax: `DISK_FILE|DISK_LABEL`. For example, the following M3U file:

```
Valhalla & the Fortress of Eve.m3u
```

```
Valhalla & the Fortress of Eve_Disk1.adf|Game Disk  
Valhalla & the Fortress of Eve_Disk2.adf|Data Disk
```



```
Valhalla & the Fortress of Eve_Disk3.adf|Level 1 Disk
Valhalla & the Fortress of Eve_Disk4.adf|Level 2 Disk
Valhalla & the Fortress of Eve_Disk5.adf|Level 3 Disk
Valhalla & the Fortress of Eve_Disk6.adf|Level 4 Disk
```

...will be shown in RetroArch's disk selection menu as:

```
1: Game Disk
2: Data Disk
3: Level 1 Disk
4: Level 2 Disk
5: Level 3 Disk
6: Level 4 Disk
```

If `DISK_LABEL` is intentionally left blank (i.e. `DISK_FILE|` ) then only the disk index will be displayed.

For games that require a dedicated save disk, one may be generated automatically by entering the following line in an M3U file: `#SAVEDISK:VolumeName` . `VolumeName` is optional and may be omitted. For example, this will create a blank, unlabelled disk image at disk index 5:

```
Secret of Monkey Island.m3u
```

```
Secret of Monkey Island_Disk 1.adf
Secret of Monkey Island_Disk 2.adf
Secret of Monkey Island_Disk 3.adf
Secret of Monkey Island_Disk 4.adf
#SAVEDISK:
```

Some games require save disks to have a specific label - for example, `It Came from the Desert` will only save to a disk named `DSAVE` :

```
It Came from the Desert.m3u
```

```
It Came from the Desert_Disk 1.adf
It Came from the Desert_Disk 2.adf
It Came from the Desert_Disk 3.adf
#SAVEDISK:DSAVE
```

Although one save disk is normally sufficient, an arbitrary number of `#SAVEDISK:VolumeName` lines may be included. Save disks are located in the frontend's save directory, with the following name: `[M3U_FILE_NAME].save[DISK_INDEX].adf` .

Save disks generated by the `#SAVEDISK:` keyword are automatically assigned the label: `Save Disk [SAVE_DISK_INDEX]` .

For games that support multiple disk drives, append "**(MD)**" as in "MultiDrive" to the M3U filename to insert each disk in different drives. Only possible with maximum 4 disks!

## 🔗 Extra features

- `#SAVEDISK:<label>`
  - Create a save disk in `saves`
- `<disk>.adf|<label>`
  - Set a friendly name (shown in "Disc Control")
- `<disks>.zip#<disk>.adf`
  - Specify a disk inside a ZIP with multiple disks (not needed with single file ZIPs)

M3U playlist supports floppy disks and compact discs.

## 🔗 WHDLoad

Pre-installed WHDLoad LHA archives can be launched directly without any kind of manual preparing and downloading.

- WHDLoad helper files (Directory or HDF) will be generated to `saves`, `WHDLoad.prefs` will be generated to `system`
- `WHDLoad.prefs` & `WHDLoad.key` will be copied from `system` to the helper image
- Kickstarts will be copied automatically to the helper image
- To update WHDLoad: simply delete the directory or the HDF

## 🔗 Overrides at startup

- **(Red)** Hold fire button for launch selector
  - For alternate `.info` launching
- **(Red+Blue)** Hold fire + 2nd fire for `ReadMe` + `MkCustom`
  - For creating default `CUSTOM` parameters

## 🔗 WHDLoad Splash Screen core option overrides

- **(Blue)** Hold 2nd fire for WHDLoad Config
  - Waits for user input if the slave supports splash screen configurations
- **(LMB)** Hold left mouse button for WHDLoad Splash
  - Briefly shows the splash screen while preloading (default WHDLoad behavior)
- **(RMB)** Hold right mouse button for WHDLoad Config+Splash
  - Always waits for user input at the splash screen

## 🔗 New WHDLoad.hdf features for old users

## 🔗 Major changes

- Slave no longer needs to be renamed to `game.slave`. The first one is selected.
- Kickstarts will be copied automatically from `system` if they do not exist in the helper image.
- `WHDLoad.key` will be copied from `system` if it does not exist in the helper image.
- `WHDLoad.prefs` will be copied from `system` on every run.
- Supports a file named `custom` in the root of the `game.hdf` for passing specific WHDLoad parameters when the slave does not support the config screen or when it should be the default, for example `Custom1=1`. It always overrides `WHDLoad.prefs`.
  - ~~The easiest way to create `custom` is to quit WHDLoad (default Numpad\*), type `echo custom1=1 >custom`, press enter and reboot the Amiga.~~
  - Script called `MkCustom` for simplest `custom` file handling. Launches after quitting WHDLoad.
  - `MkCustom` will create a slave-based `custom_$SLAVE` in `WHD Saves:`. Essential with readonly images.
- Supports a file named `load` in the root of the `game.hdf` which overrides the whole launch command, aimed at non-WHDLoad installs.
- If `.slave` is not in the root of the HDF, it will also be searched under the first found directory.
- Saves will be redirected to a separate `WHD Saves.hdf`. Repo provides an empty 4MiB HDF.

## 🔗 Minor changes

- Both HDF-files ( `WHDLoad.hdf` & `WHD Saves.hdf` ) can be located in either `RA system` OR `saves`.
- "WHDLoad Support" core option does not need to be disabled when launching a non-WHDLoad HDF which has `S/startup-sequence`.
- `NTSC` parameter can be used with WHDLoad.
- Included `ClickNot` for suppressing drive clicking when drive sound emulation is enabled.
- Included `MEmacs` for file editing ( `custom` & `load` ).
- Updated WHDLoad to the latest one (18.5 2019-03-09).
- New WHDLoad defaults:
  - `ButtonWait` (Waits for a button press in certain slaves when loading is so fast that you can't enjoy a picture or a tune).
  - `ReadDelay=0` & `WriteDelay=50` (These speed up OS switching on loadings and savings. Saves tend to corrupt HDFs with `WriteDelay` below 50).

## 🔗 Latest changes

- Support for Retroplay LHA installs.

- Support for Arcadia installs (requires KS 1.2, `kick33180.A500` will be copied automatically).
- Fixed `xpkmaster.library`.
- Show `ReadMe` before running `MkCustom` after quitting.
- Changed HDF filesystem from OFS to FFS. Prevents image corruption on unclean shutdowns.
- Default launch method changed to `WBRUN`.
- Better support for games with multiple icons.
- Hold down fire button at boot for `.info` selector. Selector will be launched always when there is no exact match for `.slave`.
- `WHDLoad.hdf` + `WHDSaves.hdf` + `WHDLoad.prefs` baked in the core. No more separate downloads!
- `WHDLoad:` and `WHDSaves:` volumes defaulted to directory mode.

## 🔗 Create a HDF image for a game

### 🔗 Not recommended or required anymore, because:

- WHDLoad specific LHA archives will work directly as a read-only hard drive image
- Directories will work as a regular hard drive image
- ZIP files will be extracted to a temporary directory and can contain any type of supported content

### 🔗 But if you must:

To do this you can use ADF Opus or amitools.

Example, to create a HDF file from a zipped WHDLoad game:

- Extract files from the ZIP to a directory
- Pack the directory in a HDF file with:
  - ADFOpus: see Allan Lindqvist's tutorial
  - amitools: `xdftool -f <NAME_OF_HDF> pack <GAME_DIRECTORY> size=<SIZE_OF_HDF>`

Note the size of the HDF specified by `SIZE_OF_HDF` must be greater than size of the directory to store the additional filesystem informations (f.ex a 1.25 ratio).

## 🔗 Using configuration files

You can pass an `.uae` configuration file and the core will load the settings and start emulation.

Look at the temporary configuration file `puae_libretro.uae` in RetroArch saves as a starting point for your own configuration files.

If the file `puae_libretro_global.uae` exists in RetroArch saves it will be appended to the temporary configuration file.

***Note that the use of configuration files is no longer encouraged or necessary. The core has been modified to always use the core options as a base, so that all custom configurations will be appended to the created configuration, effectively overriding the core options. The problem with this is that changing any core option while the core is running will reset all duplicate configurations. Therefore only add configurations which will require a restart or do not exist in the core options, if you must use a custom uae. If there is an option missing that is a must have, please make an issue about it.***

Example 1: You want to mount four non-RDB HDF files. You have one bootable 1000 MB file called `System.hdf` created with `surfaces=1`, and three non-bootable 2000 MB files called `WHDGamesA.hdf`, `WHDGamesB.hdf`, `WHDGamesC.hdf` created with `surfaces=2`. Your HDF files are located in the folder with absolute path `/emuroms/amiga/hdf/`. For that scenario, you should create a `.uae` text file with the following content:

```
hardfile=read-write,32,1,2,512,/emuroms/amiga/hdf/System.hdf
hardfile=read-write,32,2,2,512,/emuroms/amiga/hdf/WHDGamesA.hdf
hardfile=read-write,32,2,2,512,/emuroms/amiga/hdf/WHDGamesB.hdf
hardfile=read-write,32,2,2,512,/emuroms/amiga/hdf/WHDGamesC.hdf
```

Example 2: You want to mount a directory full of extracted data as a hard drive:

```
filesystem2=rw,DH0:data:/emuroms/amiga/,0
```

Windows tip:

- If paths are enclosed with quotes, Windows needs double backslashes: `filesystem2=rw,DH0:data:"c:\\emuroms\\amiga",0`.

Linux tip:

- Leave the ending slash to the path to make sure UAE sees it as a directory.

If you are using RDB HDF files, please use `0,0,0,0` instead of geometry numbers like `32,1,2,512`. The geometry will then be read from the file. This only works for RDB HDF files.

## Latest features

- Full framerate double line interlace backport (sonninnos)

- Core-based ZIP extraction with automatic M3U generation (sonninnos)
- CDTV support (sonninnos)
- CD turbo speed backport (sonninnos)
- Touch control VKBD (sonninnos)
- File-based WHDLoad.hdf (sonninnos)
- Memory-based savestates (jdgleaver)
- Analog Joystick (sonninnos)
- SuperHires (sonninnos)
- Retroplay WHDLoad LHA support (sonninnos)
- CD32 support (sonninnos)
- WHDLoad.prefs override core option (sonninnos)
- Internal+external floppy drive sounds (sonninnos)
- Single line high resolution mode (sonninnos)
- Savestate support (sonninnos)
- Parallel port four player joystick adapter is emulated with working two, three and four-player controls (sonninnos and rsn8887)
- CD32 controller support (sonninnos)
- Multiple mice support using raw input driver on Windows (sonninnos)
- Autozoom and autocentering options for large game display without borders or distortion (sonninnos)
- Mappable hotkeys to bring up statusbar and on-screen keyboard (sonninnos)
- Analog stick mouse control (sonninnos and rsn8887)
- On screen keyboard with easy joystick/dpad control (sonninnos and rsn8887)
- Implement new core options with some small print explanations (sonninnos)
- Implement hotkeys for status bar and virtual keyboard that can be remapped instead of using hardcoded buttons (sonninnos)
- Allow starting the emulated Amiga without content (sonninnos)
- Add many core options, such as a cycle exact option (sonninnos and rsn8887)
- Allow to load any single, bootable .hdf hard drive file directly via load content (rsn8887)
- Allow disabling the use of system/WHDLoad.hdf (rsn8887)
- Fix quickmenu->disk control options for swapping disks on the fly (rsn8887)
- Change option defaults to more sensible values (sonninnos)
- Add Switch version of the core (rsn8887)
- ...