

**YACDP**

**COLLABORATORS**

	<i>TITLE :</i> YACDP		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 1, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>YACDP</b>	<b>1</b>
1.1	YACDP.guide . . . . .	1
1.2	copyright note . . . . .	2
1.3	disclaimer . . . . .	2
1.4	system requirements . . . . .	3
1.5	features . . . . .	3
1.6	getting started . . . . .	3
1.7	gui . . . . .	6
1.8	menu . . . . .	7
1.9	icon . . . . .	9
1.10	text . . . . .	9
1.11	buttons . . . . .	10
1.12	all the rest . . . . .	11
1.13	keyboard shortcuts . . . . .	12
1.14	save audio as iff-cdr . . . . .	13
1.15	arexx port . . . . .	16
1.16	compatibility . . . . .	20
1.17	toshiba xm-3301bme . . . . .	21
1.18	toshiba xm-3401ta . . . . .	22
1.19	toshiba xm-4101ta . . . . .	22
1.20	apple cd300e plus . . . . .	22
1.21	philips cm215 . . . . .	22
1.22	apple cd300 . . . . .	22
1.23	pioneer drm-604x . . . . .	23
1.24	texel dm-xx24 . . . . .	23
1.25	apple powercd . . . . .	23
1.26	hitachi cdr3750 . . . . .	23
1.27	sanyo crd-400i . . . . .	23
1.28	nec cd-rom's . . . . .	24
1.29	to do . . . . .	24

---

---

1.30	how to support yacdp . . . . .	24
1.31	thanks . . . . .	25
1.32	history . . . . .	26
1.33	songfiles and the hashfile . . . . .	28
1.34	author . . . . .	29

---

# Chapter 1

## YACDP

### 1.1 YACDP.guide

Yet Another CD Player 1.2  
© 1994 by Frank Würkner  
All Rights Reserved

#### COPYRIGHT NOTE

Support Shareware!

#### DISCLAIMER

Who pays if all blows up

#### SYSTEM REQUIREMENTS

What do you need to run the programm

#### FEATURES

What features does YACDP have

#### GETTING STARTED

The description of the tooltypes

#### GUI

What are those buttons, sliders, etc. good for

#### SAVE AUDIO AS IFF/CDR

How to save songs of the CD as IFF or CDR

#### AREXX PORT

Controlling YACDP with AREXX

#### COMPATIBILITY

What CDROM's have been tested

#### TO DO

What is planned in the future

#### HOW TO SUPPORT YACDP

How else can YOU help

---

THANKS

Who has helped

HISTORY

What changed since the last version

## 1.2 copyright note

COPYRIGHT NOTE

-----  
The program YACDP is copyrighted © 1994 by Frank Würkner.  
All rights reserved.

YACDP is SHAREWARE. This program may be freely distributed as long as  
- the executable and documentation remain unchanged and are included  
in the distribution

- no other charge is made than to cover time and copying costs

WHY SHAREWARE

-----  
Well, what started as fun turned out to be work (e.g. the hours I spent  
adjusting YACDP to run with various controllers and CD-ROM's). However  
this version is FULLY functional, because I don't like those nasty  
requesters, keyfiles, etc. I hope that the program itself convinces you  
to pay the Shareware fee.

HOW TO REGISTER

-----  
Just print out the supplied 'Registration form' and fill it out. Put it  
along with the fee of 20DM or US\$15 (only CASH or EUROCHEQUE please)  
into an envelope. By sending the fee to the  
author  
you support the  
further developement. You will recieve the next update by mail. If you  
are interested in updates by mail beyond the first you must include 5DM  
or US\$5 for each further update. But I recommend to get further updates  
from the Aminet, Aminet CD, FreshFish(TM) CD or one of your local BBSes.  
If you have EMail I can notify you of new versions or send you updates  
lhaed and uuencoded.

## 1.3 disclaimer

DISCLAIMER

-----  
No warranties of any kind are made as to the functionality of this  
program. You are using it ENTIRE at your own risk.

---

## 1.4 system requirements

### SYSTEM REQUIREMENTS

-----

- o OS2.0 (OS2.1 for using localization) or higher
- o reqtools.library © 1991-1994 by Nico François (thanks Nico!)
- o a SCSI bus adapter capable of sending SCSI direct commands
- o a SCSI-2 CD-ROM which can read CD-DA ("Red Book") encoded discs (i.e. you should see a headphone jack on the front panel)
- o optionally your CD-ROM should be capable of sending the audio as 16-bit digital data over the SCSI bus

## 1.5 features

### FEATURES

-----

- o GUI that let's you access all functions of a normal CD-Player
- o is a Commodity
- o powerfull AREXX port
- o localized for 2.1 and up (currently German)
- o can display songnames and CD title, of course you once have to enter the names (about 150 CD texts are already included)
- o doesn't wastes memory
- o SCSI errors are displayed as text (95 known)
- o realtime stereo playback of audio on your Amiga, even on 68000 machines (achieved by highly optimised assembler routines)
- o ability to save (parts of) songs as stereo IFF (8SVX) or RAW (CDR)

For the last two items the CD-ROM must be able to send audio over the SCSI bus (see

SYSTEM REQUIREMENTS  
)

## 1.6 getting started

---

## GETTING STARTED

-----

If you often use audio CD's you might like to put the program into the WBStartup drawer (don't forget to change the 'SONGDIR' and 'AREXXDIR'). But any other place is fine too. Aside TOOLPRI all other tooltypes can also be used when starting the program from CLI, e.g. 'YACDP CX\_POPUP=NO UNIT=6'. The default values are shown in brackets. Following tooltypes are known:

## TOOLPRI [0]

The taskpriority of YACDP (must be a number from -128 to 127). If you often want to use the 'via SCSI' mode, I recommend a taskpriority of 1

## CX\_PRIORITY [0]

The priority for the commodity (must be a number from -128 to 127)

## CX\_POPUP [YES]

YES = show GUI when started  
NO = don't show GUI

## CX\_POPKEY [CTRL SHIFT ALT C]

Hotkey to bring up the GUI

## DEVICE [AUTOMATIC]

Your SCSI device name such as scsi.device (caution: the devicename is casesensitive). The default is to search a suitable device (compares with a builtin list), but if your device is not in the list this fails.

## BOARD [0]

Controller board number (0 to 7), normally 0

## UNIT [AUTOMATIC]

SCSI target ID (0 to 7). The default is to search for the first CD-ROM.

## LUN [0]

SCSI logical unit (0 to 7), normally 0

## VIASCSIMODE [AUTOMATIC]

The command to send digital audio data over the SCSI bus is a nonstandard command and therefore differs from drive to drive. The default is to guess the right command for your drive. Valid values are:

- 0 = CD-ROM doesn't support to send digital audio data over the SCSI bus.  
So hide the menuentries and buttons that can't be used
- 1 = Command used by Sony drives
- 2 = Command used by Toshiba drives, i.e. READ(10) plus setting the right block format
- 3 = Simple read command, i.e. READ(10)

## SONGDIR [PROGDIR:Songs/]

Valid directory where to store the songfiles

---



AREXXDIR [PROGDIR:AREXX/]

Valid directory where to search for the AREXX menu definition ('YACDP.menu', see also MENU) and for the corresponding AREXX macros

EDITOR [C:Ed %s]

Your favorite texteditor which will be used if you edit the songnames. If your editor doesn't detach, you may want to use 'run C:Ed %s' so YACDP is not blocked. The songfile will be reread by YACDP every time you change the songfile (file notification).

IFFDIR [RAM:]

Default directory where the IFF files are saved

WINDOWX [0]

Windows x position

WINDOWY [11]

Windows y position

ZIPWINDOWX [0]

x position of the zipped window

ZIPWINDOWY [11]

y position of the zipped window

VIASCSI [NO]

Content of 'via SCSI' at startup:  
YES = checked  
NO = not checked

AUDIOCHANNELS [---]

If the 'via SCSI' mode is used the playback is done by the Amiga which has four audio channels (two for the left and two for the right). The default is to allocate any left-right pair. Using more than one channel for each side is not supported and also wouldn't make any sense. Normally there should be no reason to change the default, but you can. The four chars mean:

```

                XXXX
Amiga's left  _____|||_____ Amiga's left
channel 0      ||          channel 1
                /  \
Amiga's right<      >Amiga's right
channel 0          channel 1

```

Where X may be: - = don't use

L = use for left audio data

R = use for right audio data

For example to play the right channel on the left and vice versa you would write 'AUDIOCHANNELS=RL--'.

AMIGAVOL [64]

Start volume for Amiga (0 to 64)

CDROMVOL [KEEP]

The default is to use the current volume of the CD-ROM (the CDROMBALANCE is also kept). Start volume for the CD-ROM (0 to 255)

AMIGABALANCE [0]

The balance for Amiga (-100 to 100). Zero means equal loudness left and right. Values smaller than zero will turn down the right channel. Values greater than zero will turn down the left channel

CDROMBALANCE [0]

The balance for the CD-ROM (-100 to 100)

RETRYs [10]

Some CD-ROM's report errors when starting play operations or reading audio data over the SCSI bus. The operation can only be executed after some retrys. This specifies the maximum of retrys for these operations

CLOSEWINQUITS [NO]

This tooltype describes the behaviour when you push the close gadget of the window. The default is to hide the window. Set this to 'YES' if the program should quit if the window is closed.

NOREQ []

Don't show requesters for nonfatal scsi errors. Set this if you get requesters on every startup

AUTOSTART []

Start playback whenever a new CD is inserted

AUTOEJECT []

Only works in conjunction with 'AUTOSTART'. The CD is ejected when the end of the CD is reached. This is avoided if you press the stop button.

ALTERNATETIME []

Show the time rather in '00:00:00' format than in '0h 0m 0s' format

CHIPMEM | FASTMEM | 24BITDMAMEM [any]

If any of the keywords is given the buffer for audio playback will be either chipmemory, fastmemory or 24Bit-DMA-memory

TESTCDROM []

Write information about your computer and CD-ROM into the specified file and exit. Please include this output when sending bug reports.

## 1.7 gui

--- GUI

---

MENU  
What you get by pushing the right mousebutton

ICON  
What does the icon in the upper left mean

TEXT  
What do these six textlines tell me

BUTTONS  
What do those nice buttons do

ALL THE REST  
How about the sliders, checkboxes, ...

KEYBOARD (SHORTCUTS)  
Control by keyboard

## 1.8 menu

----- MENU

'About Program'  
Display a requester about the program (what surprise).

'About CD-ROM'  
Display a requester about the CD-ROM including vendor, the productname and the revision.

'About CD'  
Display a requester with the songnames (if present) and the playing time of each song.

'Edit Songs'  
Start the texteditor so you can enter the songnames. Replace the lines in '<>' by the real names. If you are intrested in how the filename is given read  
Songfiles and the hashfile  
.

'Update'  
You may need this if you change something in the corresponding songfile of the CD and file notification isn't supported or the data displayed about the CD isn't correct.

'Hide'  
Hide the GUI.

'Quit'  
Quit the program.

'SCSI'

---

All menuitems change the value of the corresponding tooltype.

'Filter'

The condition of the lowpass filter (defaults to off).

'Release Device'

If you first play back audio data 'via SCSI', some additional buffers and soundchannels of the sound device are allocated. Use this if you want the program to give them back.

'Save Audio'

Show a window where you can enter the parameters which part of the CD you want to save. (see

```
SAVE AUDIO AS IFF/CDR
)
```

'Trace'

If checked, show every AREXX command send to YACDP in the status/message line.

'Flush waitqueue'

YACDP supplies an AREXX command 'WAIT' (see

```
AREXX PORT
)
```

to wait for

special states. When testing AREXX script you can use this to abort all waiting scripts.

Menuitems beyond this can be configured in the file 'AREXX/YACDP.menu'. The textfile contains an even number of lines. Two following lines describe one menuitem. The first line is the name of the menuitem (if the line only contains a minus it is treated as seperatorline). The second is the command that is executed when you choose the menuitem. You can also start other programms here.

Here is a description of the default macros:

'Test CD'

Play the first 10 seconds of every song on the CD.

'Upside down'

Play the CD from the last song to the first song.

'Fade out'

Continual reduce the volume to zero

'Fade in'

Continual increase the volume to the maximum

'Show current position'

Show the actual positon. The first number at 'Abs. block' is the blocknumber used in 'Save Audio', where the number in brackets is the real blocknumber.

'Dump vars'

Show all AREXX variables currently valid.

---

## 1.9 icon

ICON

----

- a SCSI cable with a flashing question-mark  
the program couldn't connect to your CD-ROM. You should have seen a requester with more detailed information. Play around with the parameters in the menu 'SCSI'
- a CD with a flashing question-mark  
the program found a CD-ROM, but you didn't insert a CD yet
- a CD with 0's and 1's on it  
you inserted a CD, but it's a data CD
- a CD with notes on it  
you inserted an audio CD

## 1.10 text

TEXT

----

- track line  
The current track, you can also say song.
  - index line  
The current index. Very few CD's do have index (some classic CD's do have them).
  - time line  
The first time shows you the amount of time already played from the total CD. The second shows you the amount of time still remaining.
  - song line  
The first time shows you the amount of time already played from this track. The second shows you the amount of time still remaining on this track. If you only see one time, the playback is in the pause area between two tracks.
  - songname line  
This line shows you the name of the current song. If you can't see it you perhaps should  
    Edit Songs  
    .
  - status/message line
-

This line either shows you a message, that was created e.g. by

AREXX

. Or it will display the status of the playback if you are in

Continue mode

The line is empty.

Shuffle mode

The blue numbers indicate the tracks that were excluded or already played. The white number shows the track that is currently playing. And the black numbers stand for the tracks that will be played.

Program mode

The blue numbers indicate the tracks that were already played. The white number shows the track that is currently playing. And the black numbers stand for the tracks that will be played.

## 1.11 buttons

BUTTONS

For most functions you will obviously have to put an audio CD into your CD-ROM or the gadgets will stay ghosted.

the '1-39' buttons and the '+' button

If your CD has more than 39 tracks you can use the '+' button on lower right of the window, which will bring up a requester where you can enter your desired track. Clicking on one of them will start the audio replay at this track (song), if you are in 'Continue'-mode. If you are in 'Shuffle'-mode this track is excluded from the playlist. If it was already excluded it is included again. If you are in 'Program'-mode the track is added to playlist. When holding down SHIFT while pressing the button the name of the song will be showed in the message line

.

'>'

If no playback is in progress it will be started.  
If the playback is paused it will start again.  
If the CD is currently playing, it will restart the playback at the begin of the current track.

'||'

(Un)pause playback

' , '

stop playback

`|<`

Start playback of previous track, not possible for 1st track :-).  
If in 'Program'-mode the previous entry of the playlist is played.  
This button has no effect in 'Shuffle'-mode.

`>|`

Start playback of next track, not possible if you are already at  
the last track. If in 'Program'-mode the next entry of the  
playlist is played. In 'Shuffle'-mode the next accidental track is  
played.

`<<`

Skip n seconds back

`>>`

Skip n seconds forward. Where n depends on the qualifier that is  
pressed with the button:

qualifier		n
SHIFT		10 secs
none		20 secs
ALT		40 secs
CTRL		60 secs

`««`

Skip to previous index

`»»`

Skip to next index ('««' and '»»' only work if not in 'via  
SCSI' mode)

`^`

Eject the CD

## 1.12 all the rest

ALL THE REST

-----

the play mode chooser

You can choose either

'Continue'

The tracks are played one after each other.

'Shuffle'

The tracks are played accidental. You can exclude tracks that  
should not be played.

'Program'

You first have to enter the list which tracks should be  
played. Then the tracks are played as you entered them.

'Repeat all' checkbox

If checked the playback is restarted when the end is reached.

**'Repeat track' checkbox**

If checked the current track is played forever.

**'via SCSI' checkbox**

If checked the audio data will be send over the SCSI bus, converted and played by the audio device. This will result in a heavy CPU load (at least for 68000er's), so you might not be able to 'multitask' much or the playback is not continous anymore. If not checked the playback is done by the CD-ROM. Only the GUI is updated every second.

**'Stereo' checkbox**

This is only available if the 'via SCSI' checkbox is checked, if the playback is done by the CD-ROM it is always stereo. So if checked the playback is stereo otherwise it is mono that means left and right channel play the same. This can be usefull because some CD's have nasty stereo effects.

**'Vol' slider**

The slider let's you control the volume for the CD-ROM and the Amiga depending on the state of the 'via SCSI' checkbox. Moving the slider to the top makes the sound louder.

**'Bal' slider**

The slider let's you control the balance for the CD-ROM and the Amiga depending on the state of the 'via SCSI' checkbox. If the slider is in the centre the loudness left and right is the same. Moving the slider to the buttom turns down the left channel. Moving it up turns down the right channel.

## 1.13 keyboard shortcuts

### KEYBOARD SHORTCUTS

**'0'..'9'**

Start playback at entered track. If the track consists of two digits the second digit must be entered within a second. If you enter more than two digits the last is shifted out, e.g. if you type 5 - 1 - 6 the track 16 will be played.

**'G'**

see

'> ' button  
'P'

see

'||' button  
'S' or 'RETURN'

see

' ' button  
'Cursor Down'

see



```

        `|<' button
        `Cursor Up'
see
        `>|' button
        `Cursor Left'
see
        `<<' button
        (also same qualifier)

`Cursor Right'
see
        `>>' button
        (also same qualifier)

`<'
see
        `««' button
        `>'
see
        `»»' button
        `^'
see
        `^ ' button
        `+'
Increase volume

`-'
Decrease volume

`HELP'
see
        `About CD' menuitem
        `SPACE'
Restore statusline (if a message is shown instead)

`BACKSPACE'
Has no effect in `Continue'-mode. In `Shuffle'-mode all songs are
included into the playlist. In `Program'-mode the last track in
the playlist is deleted (if not currently played).

```

## 1.14 save audio as iff-cdr

SAVE AUDIO AS IFF/CDR

-----

COPYRIGHT

-----

Though you own the CD you do not own the rights for the CD. Especially the CDR mode enables you to copy CD's without loss of data. Have a look at the `Copystatus'-line to see if the author of the CD allows digital copying. YOU are responsible that no existing rights are violated.

BACKGROUND

-----  
 CD-DA discs have their audio data stored in blocks of 2352 bytes plus additional information each containing 588 signed 16-bit numbers for the left and the right channel. The samplingrate of an audio CD is 44100, which means 44100 signed 16-bit numbers for each channel reconstruct the original waveform for one second. So 75 (=44100/588) blocks on a CD-DA discs make one second. For example 1 minute in CD quality has a total size of 75\*2352\*60 = 10584000 (about 10MB). The Amiga soundchip (Paula) can play back signed 8-bit numbers with a maximum sampling rate of 28867 (more for ECS Amigas).

#### CONVERTING AUDIO

-----  
 So the audio data send by the CD-ROM has to be reduced, but how? The program let's you controll the conversion by the

##### o Samplingrate

```
44100 (*) ==> 1/1 original samplingrate
22050      ==> 1/2      "      "
14700      ==> 1/3      "      "
11025      ==> 1/4      "      "
8820       ==> 1/5      "      "
```

...

Reducing the samplingrate beyond 11025 will result in very pure quality.

(\*) most players will refuse this rate (read 'BACKGROUND' for why)

##### o Channels

###### 'Stereo (CDR)'

Do absolute no conversion (the file created is simply a dump of the data present on the CD, i.e. the file is not in IFF format). This format can be used to convert to other formats using sox (PL11, Amiga port by David Champion, aminet/mus/edit/AmiSOX33.lha). For example you can create a 16Bit Windows .wav file by simply executing 'sox test.cdr test.wav'. The format of the .cdr file is (beware the Intel byte order!):

Byte | Sample | Meaning

Byte	Sample	Meaning
1	1	Least significant byte (lowbyte) left channel
2	1	Most significant byte (highbyte) left channel
3	1	Least significant byte (lowbyte) right channel
4	1	Most significant byte (highbyte) right channel
5	2	Least significant byte (lowbyte) left channel
...	...	...

###### 'Stereo (IFF)'

Both channels are written seperately.

Note:

Some players don't understand stereo IFF's!

Recommended players are:

- o DSound 1.50 © 1991-1994 by Dave Schreiber  
(aminet/mus/play/DSound150.lha)

- o DeliTracker 2.07 © 1991-1994 by Peter Kunath and Frank Riffel (aminet/mus/play/DeliTracker\_207.lha)

`Mono (IFF)'

The data for the left and the right channel is combined, i.e. the size is only  $\frac{1}{2}$  of the stereo IFF

`Left only (IFF)'

only the audio data for the left channel is taken into account

`Right only (IFF)'

only the audio data for the left channel is taken into account

o Scale samples

Sometimes the audio data will sound rather low, so you can scale them by any amount, especially by a power of 2.

Try 2, 4, 8, 16.

Example:

Let's say you have choosen a samplingrate of 14700 and `Left & Right' and scale the samples by 2, how would this input be reduced:

L	R	L	R	L	R
-9785,	456,	-23764,	135,	-32768,	-343

o First all samples will be summed up: -66069

o The divisor is  $(256 \times 2 \times 3) / 2 = 768$

$\begin{array}{cccc} \wedge & \wedge & \wedge & \wedge \\ 16\text{-bit to 8-bit} & | & | & \text{scale samples with 2} \\ \text{left \& right channel} & | & & \\ & & & 14700, \text{ i.e. } 1/3 \end{array}$

o The final output would be  $-66069 / 768 = -86$

Note:

If you had choosen to scale with 4 the result would have been -172, which doesn't fit in 8-bit and therefore would have been reduced to -128 (As well as numbers greater than 127 will be reduced to 127).

If this case happens too often your audio data will sound pretty ugly.

o The scrollers on the left side represent the start track the relative time in this track and finally the absolute blocks (remember 1 second makes 75 blocks?). The scrollers on the right side represent the same for the end position.

o buttons

`PLAY START'

plays 10 seconds from your current start position

`PLAY END'

plays 10 seconds before your current end position

`PLAY ALL'

plays all of your selected audio data

`STOP'

stop any audio playback

`OK'

to proceed and a filerequester let's you select the filename

'CANCEL'

brings you back to the main window.

If all goes fine you should see a little window showing the success of the conversion. Press ESC or CTRL-C in the window if you want to abort the conversion.

## 1.15 arexx port

### AREXX PORT

The program sets up an AREXX port named 'YACDP'. Some commands need to have arguments that are parsed via templates in the same way as CLI commands. If the command is not known or the arguments are invalid (that means a mandatory argument is missing, a numeric value is incorrect, ...) the command will return with error 10. If a numeric value is out of the expected range it will be set to the next valid value. This is not considered to be an error. For example you send the command 'PLAYTRACK 20', but the CD only has 15 tracks the 15th track will actually be started. In case of failure (RC>5) the AREXX variable ERR will contain a description of what went wrong.

RC	Description
0	no error
5	requester aborted
10	any non-SCSI and nonfatal error, e.g. the command is not known
15	SCSI error
20	fatal error that leads to the termination of the main program, e.g. an out of memory error

Following commands are known:

- o PLAYTRACK TRACK/N,INDEX/N [1,1]  
Play whole CD starting at specified track and index
- o PLAYBLOCK BLOCK/N [first valid block of CD]  
Play whole CD starting at specified block. Do not assume the CD to start at block 0. Use the 'GETVAR TRACKSTART NUM=n' command to find out about where to start.
- o PLAY  
Same as if you push the '>' button
  - o PAUSE
- Same as if you push the '||' button
  - o STOP
- Same as if you push the '\' button
  - o LASTSONG
- Same as if you push the '|<' button

- o NEXTSONG  
Same as if you push the  
`>|` button
  - o BACKWARD BLOCKS/N [20\*75=1500]  
Skip blocks backward (75=one second), i.e. same as the  
`<<` button
    - o FORWARD BLOCKS/N [20\*75=1500]  
Skip blocks forward (75=one second), i.e. same as the  
`>>` button
    - o LASTINDEX  
Same as if you push the  
`««` button
    - o NEXTINDEX  
Same as if you push the  
`»»` button
    - o EJECT  
Same as if you push the  
`^` button
- o SAVEAUDIO FILENAME/A, STARTBLOCK/N/A, ENDBLOCK/N/A, SCALE/N, COMP ←  
/N,  
TYPE/N, NOWIN/S [ , , , 1, 2, 2, ]  
This command correspondes with the menuitem (  
Save Audio  
) . Just  
supply the filename, the startblock and the endblock. Optionally  
you can scale the samples, set the samplingrate (see below) and  
the fileformat (see below). If you set 'NOWIN' the the  
progress-window will not be opened.

COMP	samplingrate	TYPE	channels and fileformat
1	44100	1	Stereo (CDR)
2	22050	2	Stereo (IFF)
3	14700	3	Mono (IFF)
4	11025	4	Left only (IFF)
5	8820	5	Right only (IFF)
6	7350		
7	6300		
8	5512		
9	4900		
10	4410		

- o SHOW  
Show GUI
- o HIDE  
Hide GUI
- o QUIT  
Quit YACDP
- o REQUESTER VARNAME/A, TITLE/A, TEXT/A, BUTTONS/A, DEFAULT/N [ , , , , 0]  
Display a requester with the title and text you specify. The  
buttons are separated with the char `|`. The buttons are numbered  
from the left to the righth starting with 1, but the rightmost  
button is numbered with 0. The default button is bold. The  
resulting number is stored in VARNAME.

- o GETSTRING VARNAME/A,TITLE/A,DEFAULT [ , ,"" ]  
Display a string requester with the title and the default text you specify. The resulting strings is stored in VARNAME. If the User aborts the requester RC will be set to 5.
- o GETNUMBER VARNAME/A,TITLE/A,DEFAULT/N,MIN/N/K,MAX/N/K,DONTSHOW/S [ , ,0,-2147483648,2147483647, ]  
Display a number requester with the title and the default number you specify. You can restrict the number with MIN and MAX. Set DONTSHOW if the default number shall not be displayed. The resulting number is stored in VARNAME. If the User aborts the requester RC will be set to 5.
- o PUTMSG MSG/A,SECS/N [ ,10]  
The message MSG will be displayed SECS seconds in the status/message line.
- o TRON  
Show every AREXX command send to YACDP in the status/message line.
- o TROFF  
Don't show AREXX commands in the status/message line.
- o LOCKVARS  
When reading more variables that are not independent from each other, you should always lock the variables before this. Otherwise the variables will perhaps be inconsistent. For example you read RELBLOCK and TRACK without locking. It could happen that RELBLOCK has a very high amount, and between the next call of 'GETVAR' the track changes and you get the false TRACK.  
Don't lock the variables longer than absolutly necessary. Make sure that no error can happen between 'LOCKVARS' and 'UNLOCKVARS'.
- o UNLOCKVARS  
The counterpart to 'LOCKVARS'. Every 'LOCKVARS' call must have a corresponding 'UNLOCKVARS' call.
- o GETVAR VAR/A,VARNAME,NUM/N/K,MODE/N/K [-,same name as VAR,1,0]  
This command let's you get some of YACDP internal variables. NUM is only used for some variables. Following types of variables exist:

```

string      : any sequence of characters
number      : a number from -2147483648 to 2147483647
boolean     : either 1 (representing true) or 0 (representing false)
blocknumber : the absolut block on CD, if MODE is
              0 : just the number, e.g. "31363"
              1 : the position in hour, minute, second, e.g. " 6m 58s"
              2 : as 1 but in alternate style, e.g. "00:06:58"
address     : the memory location

```

varname	type	description
DEVICE	(1)   string	the SCSI devicename used
BOARD	(1)   number	controller board number (0 to 7)
UNIT	(1)   number	SCSI target ID (0 to 7)

LUN	(1)	number	SCSI logical unit (0 to 7)
CDROMVOL	(1,4)	number	volume for CD-ROM (0 to 255)
CDROMBAL	(1,4)	number	balance for CD-ROM (-100 to 100)
AMIVOL	(1)	number	volume for Amiga (0 to 65)
AMIBAL	(1)	number	balance for Amiga (-100 to 100)
PLAYMODE	(1)	number	0 = Continue   1 = Shuffle   2 = Program
REPEATALL	(1)	boolean	state of "Repeat all" checkbox
REPEATTRACK	(1)	boolean	state of "Repeat track" checkbox
VIASCSI	(1)	boolean	state of "via SCSI" checkbox
STEREO	(1)	boolean	state of "Stereo" checkbox
LOWPASSFILTER	(1)	boolean	1 if lowpassfilter is turned on
ALTERNATETIME	(1)	boolean	same as tooltype ALTERNATETIME
NOREQ	(1)	boolean	same as tooltype NOREQ
RETRYS	(1)	number	same as tooltype RETRYS
SHUFFLELIST	(1,4)	string	(only available if PLAYMODE=1)   a list of the songs not yet played, the   current is marked with a '*' e.g.   2,5,*8,12 (means 2,5 or 12 will be played   after the current i.e. 8 is ready)
PROGRAMLIST	(1,4)	string	(only available if PLAYMODE=2)   a list of the songs, the current is marked   with a '*' e.g. 2,5,*8,12 (means 2 and 5   where already played, 8 is currently playing   and 12 will be played after this)
STATUS		number	0 = No valid CD-ROM found   1 = No CD inserted   2 = Data CD inserted   3 = playback is stopped   4 = playback is paused   5 = playback in progress
GUIOPEN		boolean	1 if GUI is presently shown
VIASCSIMODE		number	same as tooltype VIASCSIMODE
LANGUAGE		string	preferred language from the user   (for localized AREXX scripts)
WINDOWADR	(2)	address	the address of YACDP's window
VENDORID	(3)	string	8 chars identifying the vendor
VENDOR	(3)	string	the real name of the vendor if known,   otherwise the VENDORID
PRODUCT	(3)	string	16 chars identifying the CD-ROM
REVISION	(3)	string	revision number of the CD-ROM
SCSIVER	(3)	number	SCSI version
MAXBUSWIDE	(3)	number	Max. bus wide (8, 16 or 32)
MAXTRACK	(4)	number	number of tracks
CDLEN	(4)	blocknumber	number of blocks on CD
CDNAME	(4)	string	Name of the CD
BY	(4)	string	by whom ???
TRACKSTART	(4)	blocknumber	starting block of track NUM
TRACKLEN	(4)	blocknumber	length in blocks of track NUM
SONGNAME	(4)	string	song name of track NUM
TRACK	(4)	number	current track
INDEX	(4)	number	current index
BLOCK	(4)	blocknumber	current block
RELBLOCK	(4)	blocknumber	current block (relativ to start of current   track)

- (1) these variables can be changed with PUTVAR
- (2) only available if GUIOPEN is 1 \
- (3) only available if STATUS>1 > otherwise RC is set to 10
- (4) only available if STATUS>2 /

- o PUTVAR VAR/A,VALUE/A [-,-]

This command let's you set some of YACDP internal variables. See above.

- o WAIT BLOCK/N/A,FLAGS/N [-,0]

The command waits till the currently played block is greater than BLOCK or any of the conditions defined in FLAGS (see below) appear. You can check ERR to see which flags actually made WAIT return.

n	2^n	condition
0	1	rather wait till the played block is less than BLOCK
1	2	user started playback at an other position
2	4	window is open (GUIOPEN=1)
3	8	window is closed (GUIOPEN=0)
4	16	valid CD-ROM is found (STATUS>0)
5	32	no valid CD-ROM is found (STATUS=0)
6	64	a CD is inserted (STATUS>1)
7	128	no CD is inserted (STATUS<=1)
8	256	the CD is an audio CD (STATUS>2)
9	512	no audio CD (STATUS<=2)
10	1024	"via SCSI"-checkbox is checked (VIASCSI=1)
11	2048	"via SCSI"-checkbox is not checked (VIASCSI=0)
12	4096	CD-ROM is playing (STATUS=5)
13	8192	playback is paused (STATUS=4)
14	16384	playback is stopped (STATUS=3)

For example:

- o wait for the window to be opened: 'WAIT 0 5'. Actually you are waiting for the window to be opened or the current block to be less than 0 (which is impossible!)
- o wait till the playback reaches block 100000 or the CD is removed or or playback is paused or playback is stopped you would send the command 'WAIT 100000 24704' (because 128+8192+16384=24704). This might return ERR 8192, so you know that the user has paused the playback.
- o wait for a CD to be inserted or an audio CD: 'WAIT 0 321' (64+256+1=321). Nothing very special. But ERR might be 320, that means both conditions are true (e.g. the user inserted an audio CD).

## 1.16 compatibility

COMPATIBILITY

-----



Following drives are known to work with YACDP:

	'via SCSI'		set
	works		volume
	-----	+	-----
Apple CD300	yes		yes
Apple CD300e plus	yes		no
Apple PowerCD	no		yes
Hitachi CDR3750	no		no
Nec CD-ROM's	no		no
Philips CM215	no		yes
Pioneer DRM-604x	no		yes
Sanyo CRD-400I	no		no
Texel DM-XX24	no		yes
Toshiba XM-3301BME	yes		yes
Toshiba XM-3401TA	yes		yes
Toshiba XM-4101TA	yes		yes

## 1.17 toshiba xm-3301bme

Vendor : International Business Machines [IBM]  
 Product : CDRM00101 !D  
 via SCSI : yes  
 Problems : none known

## 1.18 toshiba xm-3401ta

Vendor : Toshiba Japan [TOSHIBA]  
Product : CD-ROM XM-3401TA  
via SCSI : yes  
Problems : none known

## 1.19 toshiba xm-4101ta

Vendor : Toshiba Japan [TOSHIBA]  
Product : CD-ROM XM-4101TA  
via SCSI : yes  
Problems : none known

## 1.20 apple cd300e plus

Vendor : Matsushita [MATSHITA]  
Product : CD-ROM CR-8004  
via SCSI : yes  
Problems : Setting the volume is currently not supported, this may be fixed in future versions. Setting the NOREQ tooltype is strongly recommended to avoid the requesters at startup.

## 1.21 philips cm215

Vendor : Laser Magnetic Storage International Company [LMS]  
Product : CM215  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : none known

## 1.22 apple cd300

Vendor : Sony Corporation Japan [SONY]  
Product : CD-ROM CDU-8003A or CD-ROM CDU-561  
via SCSI : yes  
Problems : none known

---

### 1.23 pioneer drm-604x

Vendor : Pioneer Electronic Corp. [PIONEER]  
Product : CD-ROM DRM-604X  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : none known

### 1.24 texel dm-xx24

Vendor : Texel [TEXEL]  
Product : CD-ROM DM-XX24 K  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : none known

### 1.25 apple powercd

Vendor : Philips [PHILIPS]  
Product : CDF080  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : none known

### 1.26 hitachi cdr3750

Vendor : Unisys [UNISYS]  
Product : CD-ROM  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : Setting the volume is currently not supported, this may be fixed in future versions. Setting the NOREQ tooltype is strongly recommended to avoid the requesters at startup.

### 1.27 sanyo crd-400i

Vendor : SANYO Electric Co., Ltd. [SANYO]  
Product : CRD-400I  
via SCSI : no (set tooltype VIASCSIMODE=0)  
Problems : Setting the volume is currently not supported, this may be fixed in future versions. Setting the NOREQ tooltype is strongly recommended to avoid the requesters at startup.

---

## 1.28 nec cd-rom's

Vendor : NEC [NEC]  
 Product : CD-ROM DRIVE:211 or CD-ROM DRIVE:400 or CD-ROM DRIVE:500 or  
 CD-ROM DRIVE:841  
 via SCSI : no (set tooltype VIASCSIMODE=0)  
 Problems : Setting the volume is currently not supported, this may be  
 fixed in future versions. Setting the NOREQ tooltype is  
 strongly recommended to avoid the requesters at startup.  
 Positioning within songs is not supported by the drive  
 (at least not by the the standard command 'PLAY AUDIO(12)').

## 1.29 to do

TO DO  
 -----

- o A-B range play, cuing
- o Ability to save CD specific settings such as volume, favourite tracks, 'via SCSI', ...
- o add an option to replace buttons by listview
- o find optimum 'Scale Sample' value
- o jog-shuttle-wheel
- o fontsensitive GUI

## 1.30 how to support yacdp

HOW TO SUPPORT YACDP  
 -----

Even if you have (not yet?) registered you can support YACDP ...

- o It would be nice if you could write me about your success/problems with the program, especially if your CD-ROM isn't yet in the

COMPATIBILITY

list or if you have problems not mentioned there.

Please include information about your Amiga model and the output that you can create by setting the tooltype 'TESTCDROM=RAM:CDROM.log'.

- o If you know your CD-ROM to be capable of sending audio data over the SCSI bus, but YACDP doesn't yet support your CD-ROM you might ask the vendor about how to access audio data and send the information to me. I will try to support your CD-ROM at the next

release.

- o If you entered the songnames of some CD's you should send the files to me (best would be compressed with lha and uuencoded to my email address) so I can include them with the next release.
- o If you have written an AREXX macro that you think is usefull to other users don't bother sending it to me so I can include it.
- o Tell me what features you would like to see in YACDP, aside those already present in the  
    TO DO  
    list.

## 1.31 thanks

THANKS

-----

Many thanks to following persons which helped to improve YACDP

Alessandro Agostinelli  
Anders Melchiorson  
Andreas M. Kirchwitz  
Bruno Vella  
Carsten Meyer  
David Ekholm  
David Kay  
Derrick Puckett  
Dieter Kaye  
Dietmar Krause  
Eric Bachfischer  
Eric J. M. Weber  
Florian Wolff  
Georg Bertram  
George Bernard  
Hartmut Jäger  
Hubertus Schneider  
James Cooper  
James M. Kerr  
Jim Gonzalez  
Karlheinz Neumann  
Lance Kibblewhite  
Mario Kemper  
Mark Peters  
Marquart Franz  
Matthias Scheler  
Mattias Johansson  
Michael Wirth  
Niels Van Willigenburg  
Oliver Wallusch  
Patrizio Rusconi  
Paul Moore  
Peter Steinseifer

---

Rainer Hess  
Robin Vermaat  
Stefan Kaintoch  
Stephan Fuhrmann  
Sven Thoennissen  
Thomas Freudling  
Urs Utzinger-Brunner  
Willi Raeckers

Thanks must also go to

Nico François for ReqTools 2.2  
Jochen Wiedmann for FlexCat 1.3  
Reinhard Spisser and Sebastiano Vigna for makeinfo 1.49

## 1.32 history

### HISTORY

1.0

first release

1.1

- o removed a bug that caused some CD-ROM's to report "INVALID FIELD IN CDB" when starting audio playback
  - o 'via SCSI' now works for Toshiba drives (of course only those who support to send audio as digital data over the SCSI bus)
  - o most buttons now have keyboard shortcuts
  - o volume now adjustable
  - o new tooltype AMIGAVOL and CDROMVOL to set the start volume
  - o IFF format now more correct
  - o removed SCSI error text that didn't belong to CD-ROM's
  - o updated vendor identification list
  - o device, board, unit, LUN now changeable within the program
  - o and therefore invalid start parameter don't quit the program anymore
  - o tooltype EDITOR now also accepts %S for editors that expect AmigaDOS wildcards
  - o new tooltype VIASCSI to allow "via SCSI" to be checked at startup
  - o new tooltype AUDIOCHANNELS to specify which of the four audio channels are used
-

- o new tooltype AUTOSTART to start playback whenever a CD is inserted
- o new tooltype AUTOEJECT to eject CD at end of playback
- o new tooltype ALTERNATETIME (time in 00:00:00 format)
- o new tooltype NOVIASCSI to hide menu and button that can only be used by CD-ROM's which are able to send audio data over the SCSI bus
- o windows now open activated
- o Now included the command TestCDROM to simplify (bug) reports

## 1.2

- o YACDP is now (uncrippled!) Shareware
  - o 'via SCSI' playback is now stereo
  - o saveing stereo IFF's is now also possible
  - o added AREXX-port and menu for AREXX macros
  - o added some things well known from CD players (shuffle/program mode, repeat track, repeat all)
  - o added balance slider
  - o volume slider now has his maximum at top
  - o localized for 2.1 and up (currently German)
  - o new look menus and standard wait pointer for 3.0 and up
  - o beautified buttons
  - o documentation now in AmigaGuide®
  - o better support for NEC CD-ROM's
  - o when using 'via SCSI' the scsi\_Length was set too large fixed (memory was trashed with some scsi devices)
  - o fixed problem with some older versions of gvpscsi.device that returned IOERR\_BADLENGTH when using odd buffer length
  - o didn't support multi-menu-select fixed
  - o the AUTOSTART feature didn't work in conjunction with CX\_POPUP=NO
  - o window looked ugly for titlefonts with a height greater 8
  - o songnaming has changed (old still accepted) and the songfiles moved into one big hashfile, see
-

- Songfiles and the hashfile
- o changes in the songfile are now displayed immediate ( ← file notification)
  - o the programm TestCDROM is now integrated into YACDP (tooltype TESTCDROM)
  - o the tooltype NOVIASCSI has vanished, use VIASCSIMODE=0 instead
  - o some new tooltypes and new keyboard shortcuts (check them!)

### 1.33 songfiles and the hashfile

Songfiles and the hashfile  
-----

A unique name for each CD  
-----

The program generates a unique number for each CD, which is a kind of key to find the songfile again when the CD is reinsterted. The number is created from the number of songs on the CD and the length of the first four songs. This may look like this: '15000500000329000429000550'. Pre 1.2 versions used either the Universal-Product-Code (bar code) of the CD or an identifier built with the number of tracks, the total playing time and the playing time of the first track. This was unsatisfying for a number of reasons:

- o some CD-ROM's have problems to read the Universal-Product-Code
- o reading the Universal-Product-Code stops playback in progress and needs far more time than reading the table of content of the CD (which is buffered in the CD-ROM)
- o few CD's have a bar code
- o CD's that are sold as a pack have the same bar code
- o the identifier was too short to guarantee that it is unique for all CD's
- o the identifier created contained the char '|' which is used for pattern maching in AmigaDOS and therefore confused editors and made it difficult to process these files

YACDP automatically renames old style filenames (the old filename is set as filenote). If you already have send me songfiles I would ask you to resend them as soon as all files are renamed (please preserve the filenote). I'm sorry for the inconvenience.

The hashfile  
-----

The hashfile was introduced to save both space and time (ever listed a directory with some 100 files ???). I recommend not to include your own

---



files into the hashfile until you have MANY own songfiles. Have a look at the 'Songs/Index' file if you want to know what's inside the hashfile.

#### Manipulating the hashfile

-----

The program 'MakeHash' included in the 'c/' drawer is used for manipulating the hashfile and has following template:  
FROMDIR, NUM/N/K, EX=EXTRACT/K, V=VERBOSE/S

#### FROMDIR

The directory where you keep your songfiles, otherwise the current directory.

#### NUM/N/K

The length of hashtable. Should be at least twice as big than the number of files you want to store.

#### EX=EXTRACT/K

Rather extract songfiles from the hashfile. Extract only those songfiles written from the author matching the pattern ('EXTRACT=#?' for all).

#### V=VERBOSE/S

Show (much!) additional information.

If you want to update your hashfile, use the script 'c/Update'. The program takes a look at the filenote to find out about who entered the songfile. So before updating you may want execute 'filenote #[0-9] "written by <your name>"'.

## 1.34 author

Frank Würkner  
Georgenschwaigstr. 22  
D-80807 München  
Germany

or

E-Mail: wuerkner@informatik.tu-muenchen.de