

**PicCon**

**COLLABORATORS**

	<i>TITLE :</i> PicCon		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 5, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>PicCon</b>	<b>1</b>
1.1	PicCon Help . . . . .	1
1.2	PicCon/Introduction . . . . .	1
1.3	PicCon/Requirements . . . . .	2
1.4	PicCon/Installation . . . . .	3
1.5	PicCon/FilesInDistribution . . . . .	3
1.6	PicCon/Usage . . . . .	3
1.7	PicCon/Disclaimer and Author Info . . . . .	4
1.8	PicCon/How To Register . . . . .	5
1.9	PicCon/Project-menu . . . . .	5
1.10	PicCon/Restore palette . . . . .	6
1.11	PicCon/Load prefs . . . . .	6
1.12	PicCon/Save prefs . . . . .	6
1.13	PicCon/Open picture . . . . .	7
1.14	PicCon/Save image . . . . .	7
1.15	PicCon/Save palette . . . . .	7
1.16	PicCon/Save data as... . . . .	7
1.17	PicCon/Change screenmode . . . . .	8
1.18	PicCon/About . . . . .	9
1.19	PicCon/Quit . . . . .	9
1.20	PicCon/Edit . . . . .	9
1.21	PicCon/Box frame . . . . .	9
1.22	PicCon/Set boxsize . . . . .	10
1.23	PicCon/Cut frame . . . . .	10
1.24	PicCon/Set frame . . . . .	10
1.25	PicCon/Flip . . . . .	11
1.26	PicCon/Autocrop . . . . .	11
1.27	PicCon/Palettesave . . . . .	11
1.28	PicCon/4bits . . . . .	11
1.29	PicCon/8bits . . . . .	12

---

---

1.30 PicCon/32bits . . . . .	12
1.31 PicCon/LoadRGB32 . . . . .	12
1.32 PicCon/LoadRGB4 . . . . .	12
1.33 PicCon/Copperlist . . . . .	13
1.34 PicCon/Imageformat . . . . .	13
1.35 PicCon/WBIcon . . . . .	14
1.36 PicCon/Iconformat . . . . .	14
1.37 PicCon/WBDISK . . . . .	14
1.38 PicCon/WBDRAWER . . . . .	15
1.39 PicCon/WBTOOL . . . . .	15
1.40 PicCon/WBPROJECT . . . . .	15
1.41 PicCon/WBGARBAGE . . . . .	15
1.42 PicCon/WBDEVICE . . . . .	15
1.43 PicCon/WBKICK . . . . .	15
1.44 PicCon/WBAPPICON . . . . .	15
1.45 PicCon/Selectframe . . . . .	16
1.46 PicCon/Bitplanes . . . . .	16
1.47 PicCon/BlitterMask . . . . .	16
1.48 PicCon/Maskformat . . . . .	17
1.49 PicCon/Bitplanemask . . . . .	17
1.50 PicCon/Interleaved masks . . . . .	18
1.51 PicCon/Interleaved . . . . .	18
1.52 PicCon/Chunky . . . . .	18
1.53 PicCon/Sprite . . . . .	19
1.54 PicCon/Spriteformat . . . . .	19
1.55 PicCon/Width 16 . . . . .	19
1.56 PicCon/Width 32 . . . . .	20
1.57 PicCon/Width 64 . . . . .	20
1.58 PicCon/Attached . . . . .	20
1.59 PicCon/CTRLdata . . . . .	20
1.60 PicCon/History . . . . .	21
1.61 PicCon/Thanks to... . . . .	22

---

# Chapter 1

## PicCon

### 1.1 PicCon Help

Introduction  
Requirements  
Installation  
Files in Distribution  
Usage  
Disclaimer and Author Info  
How To Register  
Thanks to...  
History

### 1.2 PicCon/Introduction

PicCon

Copyright © November 27, 1993

All Rights Reserved

Written by

Morten Eriksen

=====  
PicCon is short for Picture Converter. This is a utility made for programmers, which will convert any pictureformat you've got support for in your datatypes library to an appropriate image format. This is an essential stage mainly in the development of games, but is also useful in development of other software (like demos, applications, etc.). Not only whole pictures can be converted, but also parts of

---

pictures can be cut out to be saved as e.g. sprites or small bitplanes.

Features include:

- uses `datatypes.library` to read pictures (can load and process e.g. JPEG's and GIF's if you've got the datatypes).
- compatible with OCS, ECS and AGA-chipsets. Device-independent code should make it compatible with future chipsets as well.
- display independent code and use of Commodore's standard screenmode-requester secures that PicCon will run in any displaymode on a capable monitor.
- correct handling of all screenmodes, including extra halfbrite (EHB), Hold-And-Modify (both HAM6 and HAM8) and Dual Playfields (DPF).
- saving in numerous image formats, like ordinary bitplanes, chunkymode, sprites (even AGA-specific spriteformats), workbench icons ++
- saving of palettes in numerous formats, both ECS-compatible (4-bits-per-gun) and AGA-compatible (8-bits-per-gun), OCS/ECS copperlists, AGA-copperlists, and more.
- manipulation of cutted frame (autocropping and flipping).
- choose to save any data as binary, assemblysource or C source for including directly into your own sourcecode.
- automatic bytealigning and blanking of trailing bits (no need to keep your images on 8-pixel bounds).

NOTE: PicCon is shareware software. If you find it useful, register to get the full version. Please refer to the `How to Register` section of this documentation.

The difference between the demoverison and the registered version is that the registered version also handles AGA graphics (more than 5 (or 6 on EHB or HAM) bitplanes, sprites wider than 16 pixels and AGA palettes and copperlists). The full version can also save workbench icons bigger than 64x64 pixels in size.

Please notice that you are not entitled to use the demoverison on a regular basis without registration. The demoverison is only for evaluation of the program.

Name and a personal code is incorporated in the registered version of PicCon.

### 1.3 PicCon/Requirements

---

## REQUIREMENTS

=====

- a) Any Amiga with Kickstart 3.0 (or greater).
- b) Datatypes for picture formats installed on your systemdisk(s).

Tested on:

- \* A500, A1200, A4000
- \* 68000, 68EC020, 68EC030, 68040
- \* kickstarts 39 and 40 (in ROM and ReKicked)
- \* 0.5 - 2.0MB CHIPRAM, 0 - 6MB FASTRAM
- \* all screenmodes on all monitortypes distributed with OS3.0 (except the A2024 monitortype, which I haven't been able to test)

- also fully tested with enforcer and mungwall. As far as I can tell, PicCon is completely free from hits.

If you find any bugs under your configuration, please contact me (address in the

Disclaimer and Author Info  
section).

## 1.4 PicCon/Installation

## Installation

=====

Make a new drawer, then copy all files in this archive to the new drawer. "That's it, and that's that."

## 1.5 PicCon/FilesInDistribution

## FILES IN DISTRIBUTION

=====

PicCon	- main program
PicCon.info	- workbench icon for program
PicCon.guide	- AmigaGuide Help/Documentation for PicCon
PicCon.guide.info	- workbench icon for docsfile

## 1.6 PicCon/Usage

---

## Usage

=====

PicCon can either be started from CLI by typing it's name in a shell/cli-window, or by doubleclicking it's icon.

On startup you'll be presented with information on the version of the program and the license (or information on how to register if you are using the demoverision).

After clicking "Ok" you'll have the program waiting for input through the menubar. Click below for full explanation on each of the menuchoices:

Project

Edit

Palettesave

Imageformat

## 1.7 PicCon/Disclaimer and Author Info

### PicCon Information

=====

PicCon Copyright © 1993 Morten Eriksen, All Rights Reserved.

The registered version is copyrighted, and is a personal license only. The demoverision is freely distributable as long as all of its files are included in their original form without additions, deletions, or modifications of any kind, and only a nominal fee can be charged for its distribution. This software is provided "AS IS" without warranty of any kind, either expressed or implied. By using PicCon, you agree to accept the entire risk as to the quality and performance of the program.

Please send your comments, wishes, and bug reports for PicCon to:

Morten Eriksen  
Jonsborgv. 18  
7563 Malvik  
NORWAY

Or by email: [mortene@idt.unit.no](mailto:mortene@idt.unit.no)

Remember to state the version number you are using, in all correspondence.



## 1.8 PicCon/How To Register

How To Register  
=====

To register, send me 2 errorfree double-density disks and USD \$15 (or GBP £10, FFR 75, DM 25, NKR 100, SKR 100, DKR 100, or an equal amount in your own currency). Cash only, please. This will earn you the latest full version plus one free upgrade of your own choice.

Softwarehouses and other companies can contact me to get a licensed in-house version.

If you've got the possibility of receiving the package through email, send only money (no disks), and let me know how to contact you.

Notice that the demoverision is not freeware. If you use it regularly, you should register, even if you're not going to use any AGA specific saveformats.

Mail to:

Morten Eriksen  
Jonsborgv. 18  
7563 Malvik  
NORWAY

Email: [mortene@idt.unit.no](mailto:mortene@idt.unit.no)

Contact me also For questions, ideas for future releases, bug-reports, wanted new formats (no matter how weird! Anyone want to have Sega Megadrive/Genesis sprites? C64 sprites? Sure! Just inform me on the format specifications/limits), or anything else.

Please support Amiga shareware, don't spread the registered version, nor use a pirated version. Increased support means better and more software for the Amiga range of computers.

## 1.9 PicCon/Project-menu

Menuchoices on the "Project"-menu:

Open picture  
Save image  
Save palette  
Save data as  
Change screenmode

---

Restore palette  
Load prefs  
Save prefs  
About  
Quit

## 1.10 PicCon/Restore palette

If you load a picture with a palette that causes PicCon's menus  $\leftrightarrow$  and requesters to be unreadable, use this option to restore the 4 first colors of the palette to the ones specified in Workbench preferences. As it's hard to 'find' this option in the menu with such palettes, you'd better remember the hotkey: RAmiga+'r' .

This function works as a switch, hit RAmiga+'r' (or select it in the menu) to get back the original palette again.

If you choose

Save palette  
while you have the workbench  
palette active, PicCon will still save the original palette.

## 1.11 PicCon/Load prefs

Load and set the preferences in the S:PicCon.prefs file. This is  $\leftrightarrow$  also automatically done upon starting PicCon. See Save prefs too.

## 1.12 PicCon/Save prefs

Save settings to S:PicCon.prefs. This file will be automatically  $\leftrightarrow$  loaded upon startup, but can also be done manually by choosing Load prefs from the menu.

Settings included in the preferences file are: savetype (binary/assemblysource/C source), image format, palettesave format, last used directory and filenames on loading/saving, plus some internal data.

This option is handy if you grow tired if reselecting the same directory/

---

filename/menuoptions each time you load PicCon.

### 1.13 PicCon/Open picture

Choose the picturefile to load. PicCon will load a picture in any picture-format which you have support for in your datatypes library. The IFF-ILBM datatype specification is delivered as standard with the distribution of WorkBench3.0, but there's many more in the Public Domain (as of now (November '93) I know that these datatypes exists, either as PD or as Shareware; JPEG, GIF, PCX and BMP).

After the file has been loaded and depacked (this might take some time, especially with JPEG-pictures), you'll be presented with a requester displaying some information on the picture. Click "Ok", and the picture will now be displayed on your screen. From here on you can start cutting out and saving pieces of the picture in many different image formats, save this picture's palette, etc.

### 1.14 PicCon/Save image

Save the picture information contained in the frame in your chosen  $\leftrightarrow$  image format. If no frame is cut out, this option will save the whole picture in your specified format. See also:

```
Imageformat
.
```

### 1.15 PicCon/Save palette

Save this picture's palette in your specified format. See also:

```
Palettesave
.
```

### 1.16 PicCon/Save data as...

Choose if the saved data should be output as RAW binary, assemblysource or C source.

Data output as assembly sourcecode will look like this:

```
SECTION mydata,DATA
```

---

```

dc.b  $45,$fe,$12,$56,$12,$56,$12,$56,$45,$fe,$12,$45,$fe,$12,$33
dc.b  $14,$2e,$32,$52,$1f,$fe,$ee,$16,$23,$23,$23,$4f,$ff,$d2,$cd
.
.
.
dc.b  $a2,$ba,$11,$01

```

C sourcecode will be saved in this format:

```

UBYTE myData[] =
{
    0x45,0xfe,0x12,0x56,0x12,0x56,0x12,0x56,0x45,0xfe,0x12,0x45,0xfe,0x12,
    0x33,0x14,0x2e,0x32,0x52,0x1f,0xfe,0xee,0x16,0x23,0x23,0x23,0x4f,0xff,
    .
    .
    .
    0xa2,0xba,0x11,0x01
};

```

These two examples show how a sourcecode will be saved in byte-format, but PicCon recognizes if word- or longwordformat would be more appropriate. E.g.: 16 pixels wide sprites will be saved with two words on each line, while 64 pixels wide sprites will be saved with 4 longwords on each line.

NOTE: Often you want the saved data to reside in CHIP RAM. To obtain this, you'll have to specify for your compiler to place it in a chipdata-section. In assembly you'll do this by appending either ',CHIP' or '\_C' to the 'DATA' keyword, e.g.:

```
SECTION mydata,DATA_C
```

All assemblycompilers may not be completely compatibel with this use of the CHIPMEM option, check your manual.

In C, you'll insert the keyword 'chip' between the 'UBYTE' keyword and the name of your datasection (e.g. myData):

```

UBYTE chip myData[] =
{
    .
    .
    .
};

```

## 1.17 PicCon/Change screenmode

Sometimes when you load a picture it will be loaded in a screenmode which is not very convenient for cutting out pieces of the picture (pixelperfect framing in SuperHires is not very easy...). Then use this menuchoice to change the screenmode. All screenmodes contained in your Monitor-drawer

(in DEVS:) will be listed out for you to choose from.

## 1.18 PicCon/About

Displays information about the program, the picture and the cutted frame.

## 1.19 PicCon/Quit

Exits program.

## 1.20 PicCon/Edit

Menuchoices on the "Edit"-menu:

- Cut frame
- Box frame
- Set frame
- Set boxsize
- Autocrop
- Flip X
- Flip Y

## 1.21 PicCon/Box frame

This function let you cut out your frame with a fixed sized box. ←

Use

- Set boxsize  
to set the dimensions of the box.

This is handy if you want to cut several pieces with the same size out of your picture. See also

- Cut frame
- and
- Set frame

.

---

## 1.22 PicCon/Set boxsize

Brings up a window where you can specify the boxdimensions of the selectbox you use in

Box frame  
function.

## 1.23 PicCon/Cut frame

After selection of this menuitem, your mousepointer will be followed

by a cross. Upon pressing the left mousebutton, you now have the possibility of selecting an appropriate piece of your picture for saving in RAW format. Just keep the left mousebutton pressed while you drag the mousepointer so that you boxes in the piece you want to save. See also

Set frame  
and  
Box frame  
.

After letting go of the mousepointer, you can select the

About  
menuitem in the  
Project  
menu to check that the box

you cut out is of the wanted size.

If your picture is 'open' (clearly seperated items on the screen) you can use the

Autocrop  
option after selection to avoid

having to do pixelperfect boxing of your frame.

If you misplace the selection, you might want to use the

Set frame  
option

to correct the coordinates.

## 1.24 PicCon/Set frame

Instead of selecting a frame by dragging out the framebox, you can use

this function to manually input the coordinates of the corners of the framebox. See also

Cut frame  
.

This function is also handy if you miss slightly when you use the mouse to select the framebox; just pop up the coordinates inputbox and modify the coordinates.

---

## 1.25 PicCon/Flip

Will flip your selected frame in the chosen direction (if no frame is cutted, the whole picture will be flipped).

This option will be useful if you don't want to have flip-routines in your code, but rather include the flipped frames as binary data.

## 1.26 PicCon/Autocrop

After framing a piece of the picture, you might want to use this option to make the framing 'perfect' (no extra space included). This is to avoid the need to do pixelperfect framing.

## 1.27 PicCon/Palettesave

The choices on this menu is for selecting the way the picturepalette will be saved when choosing Save palette in the Project menu.

Menuchoices on the "Palettesave"-menu:

- 4 bits
- 8 bits
- 32 bits
- LoadRGB4
- LoadRGB32
- Copperlist

## 1.28 PicCon/4bits

---

Will save out your palette in the standard old chipset and enhanced chipset formats (4-bits-per-gun, 1 word per color).

E.g.:

White is \$0fff, red is \$0f00 and purple is \$0f0f.

```

    ^blue component
    ^green component
    ^red component
    ^unused

```

## 1.29 PicCon/8bits

Will save out your palette in the new AGA chipset format (8-bits-per-gun, 2 words (1 longword) per color).

E.g.:

White is \$00ffffff, red is \$00ff0000 and purple is \$00ff00ff.

```

    ^^blue component
    ^^green component
    ^^red component
    ^^unused

```

## 1.30 PicCon/32bits

Will save out your palette in the internal systemformat (which will probably be used in the future AAA chipset (32-bits-per-gun, 3 longwords per color)).

E.g.:

White is \$ffffffffffffffffffffffff and purple is \$ffffffff00000000ffffffff.

```

    ^^^^^^^^blue component
    ^^^^^^^^green component
    ^^^^^^^^red component

```

## 1.31 PicCon/LoadRGB32

Will save the palette as a table ready to be passed to the graphics.library function "LoadRGB32".

Look up the autodocs specification of "LoadRGB32" if this is unclear.

## 1.32 PicCon/LoadRGB4



Will save the palette as a table ready to be passed to the graphics.library function "LoadRGB4".

Look up the autodocs specification of "LoadRGB4" if this is unclear.

### 1.33 PicCon/Copperlist

Will save out your palette as a list of commands that the Amiga graphics-coprocessor (a.k.a. COPPER) understands. Insert the palette coppercommands into your own custom copperlist to make use of the paletteentries.

PicCon knows the format of both OCS/ECS and AGA chipsets, and will save in the appropriate format according to the number of bits-per-gun specified:

4 bits-per-gun: OCS/ECS copperlist.

8 bits-per-gun: AGA copperlist.

32 bits-per-gun: this is an unknown future format that is not in use today (1993), which is why you will not be able to save 32-bit palettes as copperlists.

LoadRGB4: internal format for use in graphics.library function "LoadRGB4", no copperversion available.

LoadRGB32: internal format for use in graphics.library function "LoadRGB32", no copperversion available.

### 1.34 PicCon/Imageformat

The choices on this menu is for selecting the way the boxed frame will be saved when choosing

Save image  
in the

Project  
menu.

Menuchoices on the "Imageformat"-menu:

Bitplanes

Interleaved

Chunky

Blittermask

Sprite

---

WBIcon  
Spriteformat  
Iconformat  
Maskformat

### 1.35 PicCon/WBIcon

This option sets the image format of the savefile to standard ↔  
Amiga  
workbenchicon. The format and type of the icon is set with the options  
in the

Iconformat  
submenus.

When you specify the filename of the workbench icon, don't include the  
.info extension, as this will be done automatically.

This option will not work in conjunction with the Save as.. asmsource or  
C source, as there's no point in saving an icon as sourcecode (or is  
it ?).

### 1.36 PicCon/Iconformat

Specify your wanted iconformat here. Available options is:

WBDISK  
WBDRAWER  
WBTOOL  
WBPROJECT  
WBGARBAGE  
WBDEVICE  
WBKICK  
WBAPPICON  
Selectframe

### 1.37 PicCon/WBDISK

---

Same type as e.g. the RAM Disk icon.

### 1.38 PicCon/WBDRAWER

Same type as any drawer in your workbench.

### 1.39 PicCon/WBTOOL

Executable files (and scriptfiles).

### 1.40 PicCon/WBPROJECT

Files that need a program to "treat them", e.g. docfiles.

### 1.41 PicCon/WBGARBAGE

Same type as the standard Amiga garbage can.

### 1.42 PicCon/WBDEVICE

I don't know what kind of files that use this icontype, as I don't think I have ever seen a "device-icon". I can't save icons of this format either, as they have no default in the 3.0 workbench (well, not in my workbench, that is). Maybe this is a future format?

### 1.43 PicCon/WBKICK

These icons can be saved under my configuration, but I still don't know what they are good for, as I have never seen any files with a WBKICK icon.

### 1.44 PicCon/WBAPPICON

Same story here as with  
WBDEVICE  
, it don't  
seem to be possible for me to obtain specifications on this  
iconformat, as the defaulticon can't be loaded with the icon.library  
function "GetDefDiskObject".

---



object(s). Just set up the blitter (or use the CPU) to perform this logic function on all common background- and objectplanes:

bckgnd	mask	result
0	0	0
0	1	0
1	0	1
1	1	0

This is referred to as the cookie-cut method in the Amiga Hardware Reference Manual. See also  
Maskformat  
.

## 1.48 PicCon/Maskformat

The choices on this menuitem is for selecting the way the masks will be saved when choosing

Save image  
in the

Project  
menu.

Menuchoices on the "Maskformat"-submenu:

Bitplanemask

Interleaved masks  
See also  
Blittermask  
.

## 1.49 PicCon/Bitplanemask

If this option is set, the selected frame will be saved as a blittermask in this format:

```

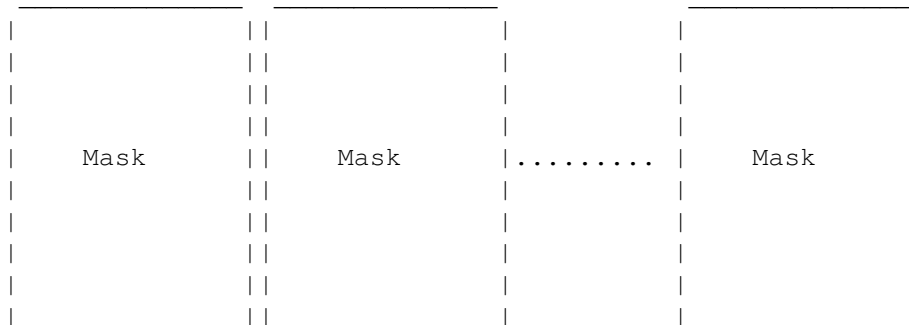
|-----|
|       |
|       |
|       |
|  Mask  |
|       |
|       |
|       |
|-----|

```

...where the bits in "Mask" is made up by OR'ing together the individual bitplanes in the frame.

## 1.50 PicCon/Interleaved masks

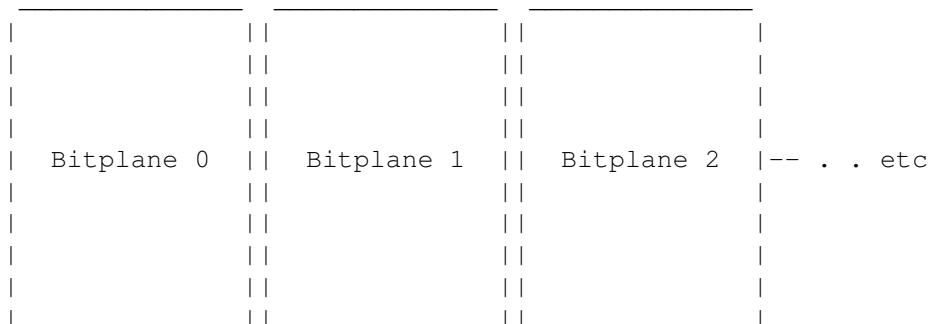
If this option is set, the selected frame will be saved as a blittermask in this format:



The mask will be saved as many times as you've got bitplanes. This format makes it possible to mask out unwanted bits in all bitplanes with only one blitteroperation (if you've made your screen interleaved).

## 1.51 PicCon/Interleaved

Set RAW save to interleaved format (useful for blittergraphics):



## 1.52 PicCon/Chunky

Chunkymode is an often used IBM-PC format, but will also be used in the Amigascene with the introduction of the Akikochip in the CD32-console.

In chunkymode, 1 byte is saved for each pixel in the picture.

E.g.:

If you've got 5 bitplanes, with bits set in planes 0 and 3, and unset in planes 1, 2 and 4, the pixel will have this value:

% 0 0 0 0 1 0 0 1 = \$09 = dec 9

bit# 7 6 5 4 3 2 1 0 (bits in unused bitplanes will be unset)

### 1.53 PicCon/Sprite

Will set RAW saveformat to Amiga sprites. Set the wanted type of ←  
 sprites  
 in the  
 Spriteformat  
 submenus.

### 1.54 PicCon/Spriteformat

The choices on this menuitem is for selecting the way the sprites  
 will be saved when choosing  
 Save image  
 in the  
 Project  
 menu.

Menuchoices on the "Spriteformat"-submenu:

Width 16

Width 32

Width 64

Attached

CTRLdata

### 1.55 PicCon/Width 16

Will set wanted sprite savewidth to 16 pixels each sprite (height unlimited). Sprites under OCS and ECS chipsets might only be 16 pixels wide.

---

### 1.56 PicCon/Width 32

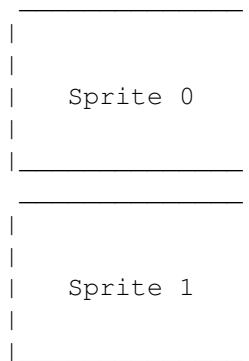
Will set wanted sprite savewidth to 32 pixels each sprite (height unlimited). 32 pixels wide sprites may only be displayed under the AGA chipset.

### 1.57 PicCon/Width 64

Will set wanted sprite savewidth to 64 pixels each sprite (height unlimited). 64 pixels wide sprites may only be displayed under the AGA chipset.

### 1.58 PicCon/Attached

For displaying sprites with more than 3 colors, you'll have to make them attached. This means that you must overlap two sprites to get 15 colors to choose from. The sprites will be saved like this:



Look up your copy of the Amiga Hardware Reference Manual.

### 1.59 PicCon/CTRLdata

Setting this option will make the program append some blank bytes in front of and behind the sprite(s) when saved. The bytes in front of the sprite is used to set position, attachmentbit etc. The bytes behind the actual spritedata is used for spritetermination.

The number of bytes appended to the actual spritedata will depend upon your selected spritewidth. If you're using 16 pixels wide sprites, 2 words will be appended in front of the sprite(s) and 2 words behind. For 32 pixels wide sprites, you'll get 2 longwords extra in front and behind, and for 64 pixels wide sprites it'll be 2 double longwords.

Look up your copy of the Amiga Hardware Reference Manual.

---



## 1.60 PicCon/History

History:

v1.00:

Initial release.

v1.01:

- bitmaps < 320x200 is now expanded when loaded
- added "Cancel" button on the loaded-requester
- some minor bugs removed
- many small quirks in the userinterface fixed
- palette can now be saved out as copperlist if preferred (support for both OCS/ECS and AGA copperlists)
- added options to save data as assembly- or C-source
- added loading and saving of settings (preferences)

v1.02:

- fixed a bug that caused crash on Amigas with MMUs
- fixed a bug that caused 5 enforcerhits (4 on startup, 1 on exit)

v1.03:

- fixed a bug in the loading and saving of boolean preferences
- "planelines" renamed to "interleaved"
- included option to output palette as a LoadRGB32 array (thanks to Vídar Petursson for this suggestion)
- included option to save as blittermask (thanks to Børge Nøst for this suggestion)
- "Load prefs" now works the way it was intended to
- made more functional demoverision

v1.04:

- included option to manually input framecoordinates
- improved some details in the filehandling and the userinterface
- improved sourcesave (autodetection of byte-, word- or longword format) (e.g.: 16 pixels wide sprites will now be saved in word-format with 2 words on each line, copperlists will be saved in wordformat, 8-bits palettes will be saved in longword format etc)
- new type of blittermask included: interleaved, multiple bitplanes (thanks Børge)
- added panic-choice "Restore palette" (for pictures with 'tight' palettes)
- included option to output palette as a LoadRGB4 array

v1.05:

- added option to save frame(s) as workbench icon

v1.06:

- improved code on saveoperations (5x faster on bitplanes, interleaved, wbicons and blittermasks, 10x faster planar to chunky conversion)
  - removed major bug in menuhandling (setting several options by clicking left mousebutton while holding down right button now works)
  - removed a "bug" in the demoverision which made it impossible to save any palettes at all ("Save as 4-bits" and "Save as LoadRGB4")
-

should work in the demoverison, only AGA-specific modes are left out)  
- removed some minor bugs

## 1.61 PicCon/Thanks to...

Thanks to...

- \* my brother Mads (for doing excellent graphics)
- \* my girlfriend Guri (for being so patient about my hobby (well, most of the time, that is..))
- \* Vídar Petursson (for betatesting)
- \* Stian (for nice coding (you still owe me a six-pack, mate ;-))
- \* Mika Saastamoinen for being my first registered user :)

... and to everybody else who has helped me with bugreports and clever suggestions for enhancements.

---