sig

COLLABORATORS			
	TITLE : sig		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		August 23, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

sig

1.1 sig.guide

This document describes SIRDS_GEN V3.7, a shareware SIRDS-Maker for
the Amiga. Many options, random dots as well as patterns are supported.
You can choose between a picture and a mathematical function as source.

1.2 sig.guide/Installation

Installation
****

To install, you just have to copy (drag) the file to the appropiate directory. If you want to have it localized, you have to copy the appropiate catalog too. This is easily done by clicking on the install_catalog icon.

1.3 sig.guide/Registration

Registration
****

Starting with V3.1, SIRDS_GEN is now SHAREWARE. There is only a small cripple, and this is, that the function-parser won't calculate the following functions:

sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, exp, log, log10, sqrt.

And without registration you wont be able to use pattern mode 4.

Take a look at the picture pic2.sirds.gif. This one is calculated with the formula "z=sin(x)+cos(y)" and pattern mode 4.

The registration fee is US$10, or if you live in germany, you can send DM15 to me. My address is:

Michael Mutschler
Somborer Weg 11
71067 Sindelfingen
Germany

Only US$ or DM are accepted. All I need is your full address with your name, Street, City and country (just like mine above.)

You will be shipped a disk with your personal keyfile, and the newest version of the SIRDS_GEN, with some patterns (and pics, if i get some via ftp...)

For all germans: Man kann mir auch die 15DM auch überweisen. Meine Konto-Nr ist:

Kreissparkasse Böblingen
BLZ: 603 501 130
Konto-Nr: 3684791

Und nicht vergessen, Name, Strasse & Ort mit draufschreiben. Das Land ist in diesem Falle überflüssig :-))

1.4 sig.guide/Distribution

Distribution
****

SIRDS_GEN V3.7 is Shareware. See
Registration
, for more info.

Good picture (preferrable the source picture, for generating others...), are always welcome via ftp to ftp.rus.uni-stuttgart.de in the directory pub/systems/amiga/incoming.

You may copy the program as you like, as long as no money is taken for it, except a small fee for copying which should be < $5. Inclusion in PD-collections, such as the Fish-Disk, or Aminet is allowed, as long as the following files stay together:

cave.pic
cave.pic.sirds
cave.pic.sis
pic2.sirds.iff
SIRDS_GEN
SIRDS_GEN.info
SIRDS_GEN000
SIRDS_GEN000.info
SIRDS_GEN.guide
SIRDS_GEN.guide.info
testpattern.iff
testpattern2.iff

Any Picture generated with SIRDS_GEN may not be used in any commercial manner without registration.

The newest Versions will be available
- via anonymous FTP: all aminet sites in the directory gfx/3d. Take a look at ftp.rus.uni-stuttgart.de in pub/systems/amiga/gfx/sirds I am collecting some pics there too, so send them!!!

- Mailbox: The Abyss: +49-711-617291 & +49-711-6159399. Type u1;16 at the main prompt to get in the right subboard. The program is free download there.

1.5 sig.guide/Introduction

Introduction
****

SIRDS means Single Image Random Dot Stereogram

The trick of SIRDS is, that you have to know how to view them. When just looking at them, like you lok at normal picture, then you will just see some random placed pixels, which seem to not make any sense at all.

So how do you look at them? There are basically two ways of viewing SIRDS: Wide-Eye view (WE) and Cross-Eye view (CE). WE ist the easier way, though I have a friend, which can only see them with CE.

WE: You have to try to look behind the picture, actually the same distance, as you are away from it. To help doing this, you can put a glass over it, and look at your own mirrored face, an the try to get the SIRDS sharp. Then you should see the picture with a real 3D effect. For better help the "Eyes" in the picture can be used: When you got it, you see 3 of them. And the middle one must be sharp. The sharp middle one, is on the farest plane.

CE: Instead of looking behind, you must cross your eyes in front of the picture. A pencil is useful to hold between the eyes and the picture, concentrate on the pencil, and make the SIRDS sharp.

The difference of viewing CE and WE is, that CE swaps the depth of the picture: the farest plane ist the nearest, and vice versa.

Another useful help is the flimmering. Just calculate some pics (~10: save one; recalc; save next ...) and put them together as an animation (e.g. DPaint can do this). This way you can’t concentrate on a particular point in the picture, and you only see something, if you have focused on the right depth. (2 frames is not enough, so flimmer is discarded now; and making an anim is so easy...) And remeber: This works only with SIRDS and not with the pattern mode!

How does it work? When looking normally, you look with both eyes on ONE point. When looking on SIRDS, you have to look on TWO points. Each eye is looking at a different point. This way, the brain thinks it is one point with a virtual depth. Now, you can vary the depth with inserting/leaving out pixels. Inserting means the point more far away.

This is not limited to graphics. You can make them out of plain ASCII too, but they don’t look that good.

Example_SIRTS

1.6 sig.guide/Example_SIRTS

```
Example_SIRTS
=============

   Here is how to make SIRTS (Single Image Random Text Stereogram)

     #include <stdlib.h>
     #include <stdio.h>

     main()
     {
     char m[100], s[80];
     int j,i,e;

     srand(time(0));
     for(e=0; e<6; printf("X%13s","",),e++);
     for(puts(""),scanf("%d\n",&j);gets(m),j>=0;puts(s),j--)
     for(e=s[79]=i=0;i<79;s[i++]=(e||i<14)?'!'+rand()%92:s[i-14])
     for(e=0;m[i-14]=='#'&&i<79&&i>13;e=1,s[i++]=s[i-13]);
     }

   compile the program and start it with "a.out <sirt.inp"

   for an input (e.g. sird.inp) you can use the following:

     17
     ------------------------------------------------------------------
     ------------------------------------------------------------------
     ------------------------------------------------------------------
     -----####-------####----####-------------################--------
     -----####-------####----####-----------####################------
     -----####-------####------------------####--------------####-----
     -----####-------####------------------####---###----###---####----
     -----####-------####----####----------####---###----###---####----
     -----###############----####----------####----------------####----
     -----###############----####----------####----------------####----
     -----####-------####----####----------####--####----####--####----
     -----####-------####----####----------####---##########---####----
     -----####-------####----####----------####-------------####-----
     -----####-------####----####-----------####################------
     -----####-------####----####-------------################--------
     ------------------------------------------------------------------
     ------------------------------------------------------------------
     ------------------------------------------------------------------

   The first line ist the number of lines that follow. a "#" means a
plane above the other.  An example output can be:

     X             X             X             X             X             X
```

```
    %Dc>qx[B]|+"i%Dc>qx[B]|+"i%Dc>qx[B]|+"i%Dc>qx[B]|+"i%Dc>qx[B]|+"i%Dc>qx[B]|+" ↩
        i
    xcnM@?N%8/j%DxcnM@?N%8/j%DxcnM@?N%8/j%DxcnM@?N%8/j%DxcnM@?N%8/j%DxcnM@?N%8/j% ↩
        D
    srE@K^M|CB1LKsrE@K^M|CB1LKsrE@K^M|CB1LKsrE@K^M|CB1LKsrE@K^M|CB1LKsrE@K^M| ↩
        CB1LK
    B)PCj=$/J5*3BB)PCj=/J5*53BB)PC=/J50*53B)PCo=/J50*53B)PCo/J50*53B)PCo/J50J*53B ↩
        )
    )(S6E@k.AtCfQ)(S6E@.AtCPfQ)(S6@.At_CPf)(S6n@.At_CPf)(Sn@.At_CPf)(Sn@.At_CP-f) ↩
        (
    P;:107Ne,C^5*P;:107e,C^’5*P;:17e,CN^’5*P;:17e,CN^’5*P:17e],CN^’5*P:17e]CN^’t5 ↩
        *
    [:-<7N=t+:m‘c[:-<7Nt+:m.‘c[:-<Nt+:ym.‘c[:-<Nt+:ym.‘c:-<N0t+ym.[‘c:<N0Jt+m.[‘) ↩
        c
    x[r9p+>%,w6y‘x[r9p+%,w6oy‘x[r9+%,wj6oyx[r9Y+%,wj6oyxr9Y+0%,j6o[yxrY+0v%,6o[ ↩
        y9x
    l/FY‘;^mD[J1ll/FY‘;mD[J1ll/FY‘;mD[7J1l/FY‘r;mD[7J1l/Y‘r;]mD[7J1l/Y‘r;]mD7J1lT ↩
        /
    /6!p/rqpoVEHw/6!p/rpoVEHw/6!p/rpoV2EHw6!p/MrpoV2EHw6p/MrtpoV2EHw6p/ ↩
        Mrtpo2EHwS6
    ZE@sr5DK.ed[*ZE@sr5K.edI[*ZE@s5K.ehdI[ZE@s;5K.ehdI[Z@s;5rKehdIE[Z@;5rKPedIE[ ↩
        CZ
    ri0/F5xoZ=h7zri0/F5oZ=hU7zri0/5oZ=‘hU7ri0/O5oZ=‘hU7r0/O5&oZ‘hU7r0/O5& ↩
        aoZhU7rd0
    W6]t/65|3J-87W6]t/6|3J-r87W6]t6|3Jq-r8W6]tD6|3Jq-r8W6tD6|[3Jq-r8W6tD6|[Jq- ↩
        rf8W
    y|K^%L_NEL/v5y|K^%LNEL/|v5y|K^LNEL3/|vy|K^FLNEL3/|vy|KFLNEL3/|vy|KFLNEL3/|yvy ↩
        |
    4[V9T’zaPC"9X4[V9T’aPC"s9X4[V9’aPCF"s94[V9Q’aPCF"s94[V9QaPCF"s94[V9QaPCFh"s94 ↩
        [
    #RA(gBIxKr54O#RA(gBIxKr54O#RA(gBIxKr54O#RA(gBIxKr54O#RA(gBIxKr54O#RA( ↩
        gBIxKr54O
    j-hwRAH+&=DCNj-hwRAH+&=DCNj-hwRAH+&=DCNj-hwRAH+&=DCNj-hwRAH+&=DCNj-hwRAH+&= ↩
        DCN
    =D’zm@kv9HWf-=D’zm@kv9HWf-=D’zm@kv9HWf-=D’zm@kv9HWf-=D’zm@kv9HWf-=D’zm@kv9HWf ↩
        -
```

can you see it?

1.7 sig.guide/Features

```
Features
********

   SIRDS_GEN V3.7 takes a picture or a formula as input. When using a
picture, the value of the pixels determine the depth-position of the
SIRDS there. e.g. if you have a 32-color picture, then you have a
maximum of 32 layers in the picture, where the background is the lowest
area, and the pixels with number 31 will be the highest one. But if you
prefer using a grayscale, then you can sort the colors first, to let
the darkest color describe the lowest area, and the lightest one the
highest.

   The second input form is a mathematical function.
```

The output is either a SIRDS or a SIS, when you supply a pattern.

Here is a further list of options:
* function plotting, and viewing as SIRDS

* free choice of screen-mode

* scaling of the picture

* should run on Gfx-cards too (not much tested, but Picasso II is working)

* automatic correction of the eyewidth to the displaymode

* uses datatypes for reading the picture

* 32-bit color-funktions are used.

* uses a symmetric algorithm

* generation of SIS possible

* flimmering

* various Settings possible

* 4 different pattern-modes

1.8 sig.guide/Requirements

Requirements
****

SIRDS_GEN V3.7 requires Kickstart 2.0+ & Workbench V2.0+, though some function work only with Kickstart 3.0+.

An accelerator with FPU is nice, but not recommended (use the 68000er version if you haven't got one).

Here is a list of functions disabled when operating under Kickstart 2.0:

* If you only have Workbench 2.0 you can't change the screenmode in the Prefs-window, and no localization is possible. Only possible with Workbench V2.1+

* The picture to be converted can't be viewed before

* The preview-Window of the Prefs-window is disabled

* only loading of IFF-ILBM pictures possible

1.9 sig.guide/Usage

```
                Usage
*****

   Choose the right version: If you own a computer with at least a
68020 AND a 68881 then you can use the normal version. Otherwise you
have to use the 68000-version.

   After starting you are prompted with the
                Preferences window
                 to do
the various settings there. Either enter a formula, or (after
finishing) you are prompted for a picture to load. Now the picture will
be loaded into a screen (the PIC-Screen). Another Screen (the
SIRDS-Screen) will be opened, and the SIRDS will be calculated.

   Due to the fact, that the same loop, for both screens (PIC & SIRDS)
is used you have the same menus, and keyboard functions. So saving for
example effects the current active window.
```

1.10 sig.guide/Keys-Menus

```
                Keys/Menus
**********



                 PIC-SIRDS-Screen Keys
                     Keys available on the screens

                 PIC-SIRDS-Screen Menu
                     Menu avaliable on the screens

                 Preferences-Keys
                          Keys available while in prefs-window

                 Preferences-Menu
                          Menu available while in prefs-window
```

1.11 sig.guide/PIC-SIRDS-Screen Keys

```
                PIC/SIRDS-Screen Keys
====================

l
     Loading of a new picture
```

```
s
     saving of the current screen as ILBM

g
     saving of the current screen as GIF

q
     exiting the program

ESC
     exiting the program

r
     Perform a new calculation

t
     switch to the other screen

p
     brings up the
                Preferences window
                . All functions there correspond
     to the ToolTypes.
```

1.12 sig.guide/PIC-SIRDS-Screen Menu

```
                PIC/SIRDS-Screen Menu
=====================

Load Pic
     Loading of a new picture

Save Pic
    ILBM
          saving of the current screen as ILBM

    GIF
          saving of the current screen as GIF

Quit
     exiting the program

ReCalc
     Perform a new calculation

Switch Screen
     switch to the other screen

Preferences
     Brings up the
                Preferences window
                . All funtions there correspond to
     the ToolTypes.
```

1.13 sig.guide/Preferences-Menu

```
Preferences-Menu
================

Project
    Use
          same as clicking on the USE button: Accept the changes

    Open
          You are asked for a config-file to load. The loaded config
          will be displayed in the Prefs-Window

    Save
          Saves the config to ENVARC:SIRDS_GEN.prefs

    Save As
          Saves the config to a specified file

    Cancel
          Leave the Prefs-Window.

Edit
    Default
          Gets the default values

    Last saved
          reads the config from ENVARC:SIRDS_GEN.prefs

    Restore
          reads the config from ENV:SIRDS_GEN.prefs
```

1.14 sig.guide/Preferences-Keys

```
Preferences-Keys
================

   The Keys usable in the Prefs-Window are all the underscored ones,
plus a few more:

q
     CANCEL

ESC
     CANCEL

h
     hidden
```

```
e
     Auto Eye-Width

E
     activate the Eye-Width Gadget when possible

u
     USE

U
     Save

RET
     USE

s
     Get Source Screenmode

d
     Get Destination Screenmode

b
     beep

a
     Auto Source Screenmode

y
     cycle EyePos

c
     switch Colors

g
     switch camg-mask

i
     switch Invers

f
     switch function-mode

w
     activates the width-gadget

p
     switch pattern-mode
```

1.15 sig.guide/Preferences window

```
                Preferences window
******************
```

```
   All the settings here reflect the
               Options
                via Toolstypes or CLI. See
the descriptions there for their meaning.

   A few things about the Prefs-Window. When klicking on the gadgets
right next to the Screen-mode text-Gadgets, you get a screen-mode
requester.  The Gadget on the left of the Source-Screen-Mode is for the
(not) visibility of the Picture-Screen.

   When clicking on the gadget labeled "Preview" next to the
Pattern-dimension area, a window will open, and you get to see the
pattern. The viewing is done (how could it be else?) via datatype. This
way, it can be (and is) done asynchronously. So if you load e.g. a GIF
or even a JPEG, it can take a while before it is visible. You can do
everything else what you want.

   The preview window is an AppWindow. This means (if you have WBPREFS
set) you can move your patterns on it, and they will be used, as if you
clicked on the load pattern button, right of the display of the
pattern-file-name.

   The save-Gadget saves the current confirguration to
ENV:SIRDS_GEN.prefs & ENVARC:SIRDS_GEN.prefs. You can edit the options
there if you like; they're saved as ASCII. If used is clicked, the
options are saved only to ENV:SIRDS_GEN.prefs.
```

1.16 sig.guide/Options

```
               Options
*******

   Here are the Options for configuring the Program.
   - you can use them as ToolTypes (e.g. HIDDEN)

   - if you want to have an option disabled, add NO in front of it. (e.g
     NOHIDDEN)

   - use them as CLI-Argument (e.g. SIRDS_GEN SIRDS_SCREEN="PAL:HighRes
     Interlace" EYEPOS BOTTOM NOBEEP)

   - When starting from CLI, you can specify an options-file to load.
     It is specified with a  @ followed by a filename. Note: The  @ has
     to be the first symbol in the commandline. Everything which
     follows will be parsed too.  (example: SIRDS_GEN
     @ENV:SIRDS_GEN.prefs FILE cave.pic INVERSE) will load the prefs
     twice).  If the file contains spaces, you can surround the file
     with quotes: SIRDS_GEN @"t:SIG file"

   - or click on the corresponding gadget in the Prefs window

   Options marked with (Startup only) are only setable on startup (as
CLI-option or Tooltype). If you want to set them for default, start the
```

program with the appropiate options, enter the prefs-window and click
on save, or you can edit the prefs-file, and add the option there.

```
[NO]BEEP
        Display a beep after drawing

[NO]CAMG_MASK
   Save IFF with modified CAMG_MASK

[NO]COLORS
      Sort colors of loaded pic before drawing

DARKNESS
        Set the percentage of black pixels for SIRDS

EYEPOS
          define where the eyes should be placed

EYEWIDTH
        define the width of the eyes

FILE
            specify the picture to be loaded as source (Startup  ↩
               only)

FUNCTION
        specify the function to be drawn (Startup only)

[NO]HIDDEN
      use HIDDEN mode

[NO]INVERSE
     create inverse-SIRDS

MINX-MAXX
       define the x-dimension of the function

MINY-MAXY
       define the y-dimension of the function

MINZ-MAXZ
       define the z-dimension of the function

MIX
             mix function & picture

PATTERN
         choose the pattern for SIS

PAT_MODE
        choose the pattern mode

[NO]PICOSAVE
    save IFFs on Piccolo (Startup only)
```

```
PIC_DEPTH
       set the virtual depth of the SIRDS

[NO]PREFS_FIRST
 show the preferences before drawing (Startup only)

RASTER
          set the raster for a function

SAVEFILE
        set the file to be saved (Startup only)

SAVEGIFFILE
     set the GIF-file to be saved (Startup only)

SEED
            set the initial seed for SIRDS (Startup only)

[NO]SHOW_SRC
    view the source picture

SIRDS_SCREEN
    set the display-mode for the SIRDS

SIRDS_HEIGHT
    set the height for the SIRDS

SIRDS_WIDTH
     set the width for the SIRDS

SRC_SCREEN
      set the mode for the source-picture

[NO]WBPREFS
     show Preferences on the workbench (Startup only)
```

1.17 sig.guide/FUNCTION

```
FUNCTION
========

   Starting with version 2.7, you are able to plot 3-dimensional
function with a SIRDS-algorithm. Really great if you can't think of
what a function will look like. The way you see it, is straight from
top down to the function. You can set all ranges of the function as you
desire.

The complete EBNF-syntax of the function-plotter is:
     func := 'z' '=' expr.
     expr := CmpOp ('<'|'>'|'<='|'>='|'='|'<>') CmpOp.
     CmpOp := term {('+'|'-') term}.
     term := factor {('*'|'/'|'%'|'div'|'mod') factor}.
     factor := value {('^'|'**') value}.
```

```
     value := ['+'|'-'] number | 'x' | 'y' | '(' expr ')'.
     value := ('abs'|'asin'|'acos'|'atan'|'cos'|'cosh'|'exp'|'log') '(' expr ')'.
     value := ('log10'|'sin'|'sinh'|'sqrt'|'tan'|'tanh') '(' expr ')'.
     value := 'if' '(' expr ',' expr ',' expr ')'.
     value := ('rad'|'radius') '(' expr ',' expr ')'.
     value := 'dist' '(' expr ',' expr ',' expr ',' expr ')'.
```

The function must contain a variable "z" at the beginning followed by a "=". The rest must be a valid function, else an error will occure.

The function parser understands the standard amount of functions:

```
        "*"         "abs"          "cosh"
        "/"         "acos"         "sinh"
        "+"         "asin"         "tanh"
        "-"         "atan"         "exp"
        "^"         "cos"          "log"
        "**"        "sin"          "log10"
        "()"        "tan"          "sqrt"
```

non standard:

```
     "div"    -  div-operator           5.7 div 0.5 (-> 11)
     "mod"    -  modulo                 5.7 mod 0.5 (-> 0.2)
     "%"      -  same as modulo
     "if"     -  if clause (see below)
     "<", ">", "=", "<=", ">=", "<>" - boolean operators (see below)
     "radius" - radius(x,y) = dist(x, y, 0, 0)
     "rad"    - same as radius
     "dist"   - distance of 2 points. syntax: dist(expr, expr, expr, expr)
                                      dist(2,4,6,7) -> (2,4) to (6,7) -> 5
```

Numbers can be written as you like. e.g the following will be accepted:

```
     1.2e-3
     .67
     -23.6
```

Another feature of the parser is, that a minus in front of a term, will be treated, as if there stands -1*term. e.g if you want to enter a term like z=-1*sin(x) you could just enter z=-sin(x). Therefore this construction is valid too: z=2-x which would result in z=2-(-1*x)

Of course * and / have a higher priority than + and -. And ^ or ** have a higher priority than * and /. So there is no need to use braces all the time, like 2+3*x.

There doesn't exist any limit for the amount of braces. The only limitation is the length of 256 bytes for the whole function, which should be enough.

The boolean functions return a value of (1.0) for true and (0.0) for false. e.g. z=(x<0)*x would result in:

```
     x<0  : z=x
     x>=0 : z=0
```

don't forget the braces; boolean expressions have the lowest priority. e.g z=x<0*x would be the same as z=x<(0*x) which is z=x<0

```
   The if-clause syntax is: if (expr, true-expr, false-expr) The
expression is tested, against 0.0. If it's not 0.0 then the expr is
true, and the true expression is calculated, otherwise the
false-expression will be used.  It is useful, to use the boolean
expression for the first expression.

now a few examples:
     z=if(x>0, 1, -1)
    This would result in 1 if x>0, and -1 when x<=0.

lets simulate the signum function:
     x>0: z=1
     x=0: z=0
     x<0: z=-1
    just do something like z=if(x > 0, 1, if (x=0, 0, -1)) but you can
do it faster if you type z=(x>0)-(x<0)

A great thing to do with the dist()-function is the drawing of
interfering sine-curves. If you draw one starting from (0/0), then you
would do z=cos(radius(x,y)) which will result in a circular sine-curve,
starting at the origin, just if you throw a stone in the lake, and
watch the waves.  You can set any middle point with the
dist()-function. i.e.  z=cos(dist(x,y,0,3)) would start in (0/3).
To overlap these two, just add them together. Multiplication is funny
too.

   As I often use the rad()-function, there exists a shortcut r for
rad(x,y). You can simply type z=sin(r) to get a circular sine-wave.

here is an example of three points:
     z = cos(radius(x,y)) * cos(dist(x,y,0,5)) * cos(dist(x,y,5,5))
     range: xmin = -11, xmax = 13
            ymin =  -8, ymax = 10
            zmin =  -1, zmax = 1

There exists a default function, which is z=-0.3*(x*x+y*y)+2
```

1.18 sig.guide/MINX-MAXX

```
MINX/MAXX
=========

   MINX and MAXX define the x-range of the function to be plotted.
Default is from -6 to 6.  !
```

1.19 sig.guide/MINY-MAXY

```
MINY/MAXY
=========
```

MINY and MAXY define the y-range of the function to be plotted.
Default is from -6 to 6. !

1.20 sig.guide/MINZ-MAXZ

MINZ/MAXZ
=========

MINZ and MAXZ define the z-range of the function to be plotted.
Default is from -2 to 2. !

1.21 sig.guide/DARKNESS

DARKNESS
========

Set the percentage of dark pixels, when drawing a SIRDS. 0 means all
white 100 means all dark. Note: when using 50, the program is slightly
faster. ! DEFAULT: 50

1.22 sig.guide/SEED

SEED
====

Set the initial seed for a SIRDS. If you pass 0, then the timer will be
used for the seed -> every time another SIRDS.

DEFAULT: 0

1.23 sig.guide/SAVEFILE

SAVEFILE
========

   When using this option, you have to pass a filename, which the SIRDS
will be saved to. You can only save IFF-files this way. The picture is
saved immediately after drawing, and the program then terminates.
Useful for making a bunch of pictures, e.g. for an animation.

DEFAULT: <none>

1.24 sig.guide/SRC_SCREEN

```
SRC_SCREEN
==========

   Screenmode for the Pic-Screen. If no valid Screenmode is found,
BestModeID() is used for getting the right mode.

DEFAULT: PAL:LowRes
```

1.25 sig.guide/SIRDS_SCREEN

```
SIRDS_SCREEN
============

   Screenmode for the SIRDS-Screen.

DEFAULT: NTSC:HighRes Interlace
```

1.26 sig.guide/SIRDS_WIDTH

```
SIRDS_WIDTH
===========

   Width of the SIRDS-Screen. If zero, the STANDARD Overscan width of
the screenmode will be used. Try bigger value than StdOscan. The
Autoscrolling looks really nice.

DEFAULT: 0
```

1.27 sig.guide/SIRDS_HEIGHT

```
SIRDS_HEIGHT
============

   Height of the SIRDS-Screen. If zero, the STANDARD Overscan height of
the screenmode will be used.

DEFAULT: 0
```

1.28 sig.guide/EYEPOS

```
EYEPOS
======

   Position of the Eyes:
TOP
     At the Top (default)

MID
     in the Mid of the Screen (if you like it...)

BOTTOM
     at the bottom

NONE
     No Eyes (for those you dont like it at all)

DEFAULT: none
```

1.29 sig.guide/EYEWIDTH

```
EYEWIDTH
========

   The space between the eyes. If you specify "0", the space will be
adjusted to the screenmode: EYEWIDTH = OSCAN_STANDARD / 10.  Actually
EYEWIDTH is the number of pixels per inch. You can use this option if
you want to calculate a SIRDS for another Media, e.g. for printing.

DEFAULT: 0
```

1.30 sig.guide/[NO]HIDDEN

```
HIDDEN
======

   If set, an algorithm for removing hidden layers is used.

DEFAULT: OFF
```

1.31 sig.guide/PIC_DEPTH

```
PIC_DEPTH
=========

   The virtual depth of the SIRDS. It is calculation is the following:
visible_depth = 20 / PIC_DEPTH * max_visible_depth. Due to this formula
```

PIC_DEPTH has to be >=20.

DEFAULT: 55

1.32 sig.guide/[NO]BEEP

BEEP
====

If TRUE, a DisplayBeep(0) is generated after each calculation, to indicate a picture is finished. Some people find this nerving, right Jens?

DEFAULT: TRUE

1.33 sig.guide/[NO]CAMG_MASK

CAMG_MASK
=========

When saving as ILBM, some (in fact one) want to mask the screenmode in the CAMG-chunk to apply a default-monitor. If this flag is true, the screen-mode will be masked with INTERLACE | HIRES_KEY.

DEFAULT: FALSE

1.34 sig.guide/[NO]INVERSE

INVERSE
=======

If set, the vitual depth of the SIRDS will be reversed: The highest area will be the lowest, and vice versa. Useful for peole who cross their view before the picture for viewing SIRDS.

DEFAULT: FALSE

1.35 sig.guide/[NO]COLORS

COLORS
======

If set, the colors are sorted. So the highest color will be the

front-most position in the SIRDS. The colors are sorted in the
following way: r-Val + g-Val + b-Val, and the sorted.

DEFAULT: FALSE

1.36 sig.guide/FILE

FILE
====

   Here you can specify a file for loading. If none specified, you will
be asked for one.

DEFAULT: <none>

1.37 sig.guide/PATTERN

PATTERN
=======

   requires a File, which will be used as pattern for SIS. The loading
is done via datatypes, so you can use any format you like. The SIS-mode
is automatically activated, when this options is specified.

DEFAULT: <none>

1.38 sig.guide/PAT_MODE

PAT_MODE
========

   4 different Pattern-modes are possible:
  1. The Pattern will be displayed normally on the left, and adjusted
     to the right

  2. The Pattern will be displayed normally in the mid, and adjusted to
     both sides

  3. The Pattern will be displayed normally on the right, and adjusted
     to the left

  4. The Pattern is scaled to the farest point on each line, and
     centered. This is the slowest mode, but the best one. (only
     available with keyfile)

   If you pass a wrong value, a SIRDS will be generated.

DEFAULT: 2

1.39 sig.guide/[NO]SHOW_SRC

SHOW_SRC
========

If OFF, no screen for the source-picture will be opened. Just to save a little Chip-Mem for bigger SIRDSs.

DEFAULT: ON

1.40 sig.guide/[NO]WBPREFS

WBPREFS
=======

If set, the Prefs-Window will open on the default PubScreen.. Otherwise it will open on the current screen.

DEFAULT: TRUE

1.41 sig.guide/[NO]PREFS_FIRST

PREFS_FIRST
===========

When set to on, The Prefs-Window will show before the SIRDS will be drawn. This way, you can select a new screenmode first, if you like.

DEFAULT: ON

1.42 sig.guide/RASTER

RASTER
======

This option works in conjunktion with the function-plotter. You can speed up the drawing, if you increase the raster. Setting RASTER to 1 will plot every pixel, and is really slow. A RASTER of 2 will draw a square of 4 Pixels which have the same value, and so on. Useful, when you want to see what a function will look like. Try using a high value, such as 50. Looks nice too.

DEFAULT: 3

1.43 sig.guide/SAVEGIFFILE

```
                SAVEGIFFILE
===========
```

CLI-option only. You have to specify a file, which the GIF-file will be saved to. The difference to SAVEFILE is:
a. The file beeing saved is a GIF-file.

b. The SIRDS is not shown!
You have to specify
SIRDS_WIDTH
&
SIRDS_HEIGHT
, but they can be any value. So you can generate huge pictures with this option, without having to worry about memory. The only limitations are 256 colors, and a width < 65536, but that's the limitation of the GIF-format. After drawing, the program will end.

1.44 sig.guide/[NO]PICOSAVE

```
PICOSAVE
========
```

Due to the fact that saving iff-pictures didn't work on a Piccolo-Gfx-card, this option activates a small workaround: The Bitmap of the picture is copied to a temporary Bitmap first, which is saved then. I advise you to not set it if not necessary, due to extra memory usage when saving.

DEFAULT: OFF

1.45 sig.guide/MIX

```
MIX
===
```

This Options enables the mix of a picture with a formula. To get things right, you need to know, that all heights are scaled to an area between 0 and 1. 0 is the back plane and 1 is the front-most plane.

There are 4 different ways to mix:

```
ADD
     This calculates the arithmetic mid of the function an the formula.
     i.e.  (pic_height + fml_height) / 2

MUL
     Multiplication of picture and formula. Useful if you want to cut
     out something of a formula. Just take a 2-color picture.

MAX
     takes the highest area's of picture or formula. You can have a
     Text fly above a sine wave for example.

MIN
     takes the minimum of picture and formula. Just for completeness.

none
     well, this is the default and doesn't mix the two.

DEFAULT: none
```

1.46 sig.guide/Address

```
Address
*******

   Bugs/Suggestions/registration to the following address:

     Michael Mutschler
     Somborer Weg 11
     71067 Sindelfingen
     Germany

     EMAIL:
     Internet: amiga2@info2.rus.uni-stuttgart.de
     UUCP: micha@agnus.tynet.sub.org
     FIDO: 2:246/1115.1
```

1.47 sig.guide/Thanks

```
Thanks
******

   Greetings:
   * Markus Wolf for the nice Test-Picture(s).

   * Hans-Jörg Malthaner for the GIF-Save routine

   * Christophe Wegmuller for the french catalog
```

"The Graphics Interchange Format(c) is the Copyright property of
CompuServe Incorporated. GIF(sm) is a Service Mark property of
CompuServe Incorporated."

1.48 sig.guide/History

History
****

Version 3.1

Version 3.2

Version 3.3

Version 3.4

Version 3.5

Version 3.6

Version 3.7

Version 3.8

1.49 sig.guide/Version 3.1

Version 3.1
===========

changes for Version 3.1:

* Bug Fix: when in the Prefs-Win the Source screen-mode was
disabled, and you disabled the function, the program would crash.

* Internal changes to the parser.

* The Parser now translates ^2 ^3 ^4 to multiplications. ^2 is now
twice as fast.

1.50 sig.guide/Version 3.2

Version 3.2
===========

```
   changes for Version 3.2:

   * complete rewrite of the options. Now using ReadArgs() instead of
     ArgArrayInit(). See
                Options
                .

   * Now using ReadPixelArray8() for GIF-saving -> twice as fast as b4.

   * When the pattern can't be loaded, a SIRDS will be drawn.

   * Sometimes, when you clicked the pattern-filerequest Gadget, the
     screen wasn't refreshed. Fixed.

   * Localization. If you want to make a catalog in a language not
     currently supported, please read the readme in the catalogs-dir.

   * Due to localization, I found some strings, which were still
     german. oops.

   * When using
                [NO]PREFS_FIRST
                , and the prefs-window is cancelled, the
     program will quit now.
```

1.51 sig.guide/Version 3.3

```
                Version 3.3
===========

   changes for Version 3.3:

   * Localized the string "Lines to do" in the Requester when saving a
     gif

   * added new
                PAT_MODE
                 4 for registered users. Without Keyfile you
     wont be able to use it.

   * Enforcer hit removed, when saving the config

   * the dimension of the function will be saved now too

   * serious bug during startup fixed.

   * added cli-option
                SAVEGIFFILE
                .

   * changed DST_WIDTH, DST_HEIGHT, DST_SCREEN to SIRDS_*. But the old
     options will work for a while too... :-)
```

```
   * PREFS_FIRST is now true for default

   * added another picture in the archive

   * Now loading locale.library with V38 instead of V39.

   * The error-messages relating the 68020er version of the lack of a
     020 & 881 now localized.
```

1.52 sig.guide/Version 3.4

```
                Version 3.4
===========

   changes for Version 3.4:

   * removed the flimmering. Didn't look that good, and wasted 1.5kb
     code.  See
                Introduction
                , for a workaround.

   * added "if" function to the function-parser

   * when the source screen wasn't open, the main loop could guru.
     fixed.

   * f & p - key now active again in prefs window.

   * "radius" & "dist" function in parser.

   * Prefs-Win now has the RMBTRAP flag set.
```

1.53 sig.guide/Version 3.5

```
                Version 3.5
===========

   changes for Version 3.5:

   * removed history of pre 3.1

   * a SIRDS_WIDTH of 640 is now guru-free

   * New option:
                [NO]PICOSAVE
                   * french catalog included
```

1.54 sig.guide/Version 3.6

```
                Version 3.6
===========

   changes for Version 3.6:

   * added
                MIX
                -mode

   * fixed pattern mode 1

   * New documentation

   * Made the Preview-Window an AppWindow. You can move patterns
     directly onto it

   * Now the KEYPATH variable is used for searching for the Keyfile. If
     it is not present, S: will be used. Just like MUI.
```

1.55 sig.guide/Version 3.7

```
Version 3.7
===========

   V3.6 was only for beta-testing; same as V3.7.

   changes for Version 3.7:

   * added "r" to function-parser (shortcut for rad(x,y))

   * when file loading fails, the program will not quit anymore.
     instead it continues, as if the requester has been cancelled

   * the getpattern filerequester now doesn't automatically disables the
     pattern-mode, when cancelled.

   * added 2.0-flag CLI-only option, which forces use of 2.0 function on
     newer OS-versions. This means, no datatypes are used!

   * now the darkest color is used for the BARDETAILPEN and the
     brightest for BARBLOCKPEN. This way you can always see the menu.
     This works on Kick3.0 only.

   * complete rewrite of the internal bitmaps: no more datatypes'
     bitmaps, but my own.  This should solve the big trouble.

   * due to internal bitmaps, you can see the loaded picture under
     Kick2.0 too
```

1.56 sig.guide/Version 3.8

```
                Version 3.8
===========

   changes for Version 3.7:

   * fixed wrong saving of mix-mode ADD

   * darkness is now saved correctly too

   * enhanced the commmand-line parser to accept an options-file. (see

                Options
                )

   * added menu to the Prefs-Window. (see
                Preferences-Menu
                )
```