FastLife Version 2.7 documentation

COLLABORATORS

	TITLE : FastLife Version 2.7 documentation		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		September 19, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

FastLife Version 2.7 documentation

1.1 FastLife Version 2.7 documentation

```
            FastLife Program Documentation
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

  Version 2.7 16-Jan-1994
```

FastLife is an Amiga implementation of Conway's Game of Life.

FastLife may be freely distributed.

Amiga Software Release 2.04+ and LIBS:reqtools.library 38+ are required
to run FastLife.

```
              Quick Start

              Credits

              Features and Changes

              The Game of Life

              About FastLife

              Starting FastLife

              Using FastLife

              Menus

              Notes

              File Format

              Interesting Patterns

              If You Have Trouble

              FastLife and Pre-2.04 AmigaDOS
```

```
                    Tomas Rokicki's Life Program

                    Author
```

1.2 Quick Start

NOTE - Kickstart 2.04+ and LIBS:reqtools.library 38.961+ are required to run FastLife.

```
1. Double click on the FastLife Icon
2. Select "Cancel" in the screen mode requester
3. Select the "Control-Run" menu item

     OR

1. Double-click on the FastLife Icon
2. Select "Cancel" in the screen mode requester
3. Select "Project-Open" menu item
4. Select "acorn.life" and press <return>
5. Click in the center of the screen
6. Select the "Control-Run" menu item
```

1.3 Credits

```
o  Intuition interface and integration by Ron Charlton
o  Blitter life routines by Tomas Rokicki
o  Requesters use ReqTools by Nico Francois
```

1.4 Features and Changes

```
              Features

              Changes from FastLife 2.2
```

1.5 Features

```
o  Full Intuition interface
o  Any screen width and height (if you have enough memory)
o  Screen mode and screen size requester
o  Random Field Generation
o  Torus mode (screen edge wraps around)
o  Speed control
o  Run for a specified number of generations
o  Run to a specified generation
o  Runs from Workbench and CLI under Kickstart 2.04+
o  150+ Life patterns in text file format
o  14 new "SpaceFiller" patterns
```

1.6   Changes from FastLife 2.2

FastLife uses new, blazingly fast Life routines written by Tomas
Rokicki. FastLife can now update a 320 by 200 screen at up to 170
generations per second on a 25 MHz Amiga 3000, depending on what pattern
is displayed. Your mileage may vary.

You can get Tomas' Life program, version 6.1, on Fred Fish disk 837.
FastLife no longer has a CPU mode and BLITTER mode; it always uses
Tomas' routines.

FastLife will generate a random field on request.

FastLife has a torus option that connects the top edge of the life
universe to the bottom edge and the left edge to the right edge.

FastLife has AmigaGuide format documentation.

Miscelleanous corrections (that I hope no one has found a need for) have
been made.

1.7   The Game of Life

In 1970 John Horton Conway, a mathematician at the University of
Cambridge, invented the game of Life. It was introduced by Martin
Gardner in his October 1970 Scientific American column. Conway's goal
was to create a board "game" with simple rules that would exhibit
complex behavior. The game is played on an infinite checkerboard. Each
cell on the checkerboard, is either alive or dead (ON or OFF). Time
moves in steps; each step marks the life of one generation.

Each cell's condition (ON or OFF) in the next generation is based on the
life in it and its eight neighboring cells. A cell is represented by
the asterisk (*) below; its eight neighbors are numbered –

    123
    4*5
    678

Three rules decide a cell's state in the next generation:

    1. If there are 2 ON neighbors the cell stays ON or OFF
    2. If there are 3 ON neighbors the cell is ON
    3. 0, 1, 4, 5, 6, 7 or 8 ON neighbors the cell is OFF

The analogy to real life is this: too many neighbors and the cell dies
from overcrowding, too few neighbors and the cell dies from loneliness.
Three cells results in trisexual mating.

The ON/OFF state is determined for all cells on the board before
changing to the next generation.

FastLife uses one pixel on the screen to represent one cell on the
board. The screen dimensions determine how many cells are available in

FastLife, so the board is not infinite, as defined by Conway.

An excellent reference for learning more about the history and
philosophical implications of Life is

       "The Recursive Universe" by William Poundstone,
       Contemporary Books, Chicago, 1985

The book consists of alternating chapters about Conway's game of Life
and modern physics, and what the game of Life can reveal about the
beginning of biological life and its replication.

1.8   About FastLife

                   I developed FastLife because I wanted to combine the speed of ↩
                      Tomas
Rokicki's Life routines with an Intuition interface and access to many
of the classic Life patterns.

The Life patterns supplied with FastLife are stored in a simple text
file (see
                   File Format
                   below). Although FastLife does not allow
on-screen editing of Life patterns, you may use your favorite text
editor to create text file patterns to load into FastLife.

Conway defined an infinite board for Life; FastLife limits itself to the
size of a screen on your Amiga. Patterns that spread to the edge of the
screen will be "eaten" away there, unless you have the
                   Torus
                   option
turned on, so you should be sure that you select a screen size that will
contain the entire pattern you want to view. This corruption of the
Life pattern at the edge of the screen is not detected by FastLife; you
must observe for yourself when this happens.

The corruption occurs at the edge of the screen, which may not be the
same as edge of what is visible to you, depending on whether you choose
a screen size larger than the default for the screen mode.

1.9   Starting FastLife

              You may start FastLife from either the Workbench or CLI (Shell).

          WORKBENCH

          SHELL or CLI

1.10 WORKBENCH

You may start FastLife from Workbench by double-clicking its icon.
Several ToolTypes may be used. To set a ToolType click once on
FastLife's icon and then press Right-Amiga-I. Click on New or an
existing ToolType to edit. Enter for example,

```
ALIVECHAR=X
```

Finally, select Save. FastLife will then honor the ToolType. The
supplied icon has parentheses around most of the ToolTypes. Remove the
parentheses to use that ToolType.

The ToolTypes available are:

```
ALIVECHAR=x

FILEPAT=#?name.???
```

You must use uppercase for the text to the left of the equal sign for
FastLife to recognize the ToolType.

Do not change the WINDOW= ToolType. It is used to display error messages
under unusual circumstances.

The character that follows ALIVECHAR= indicates what character in
Picture files represents an ON cell. All other characters in the pattern
will indicate OFF cells.

The string that follows FILEPAT= determines what file names appear in
the Open file requester. Amiga wildcards are supported. "?" (no quotes)
in the string matches any one character. "#?" (no quotes) matches any
number of characters. Upper/lowercase do not matter. Some examples:

```
  #?.life     any name ending in ".life"
  #?.???      any name that has a three-character extension
  #?breeder#? any name that contains "breeder" anywhere in it
```

1.11 SHELL or CLI

You may start FastLife from a Shell or CLI by typing

```
    1> FastLife          or

    1> run FastLife
```

FastLife's template is ALIVECHAR/K,FILEPAT/K.   Typing

```
    1> FastLife ?
```

will display the template and wait for you to enter arguments.

To specify the Picture file character for ON pixels as the uppercase
letter 'O' use

```
    1> FastLife ALIVECHAR O
```

Finally,

```
    1> FastLife FILEPAT #?.lif
```

will set the file requester pattern to "#?.lif" initially. You can
change the settings for CALCMODE and FILEPAT after FastLife has started.

NOTE – The Life files supplied with FastLife all end in ".life".
"#?.life" is FastLife's default pattern.

1.12   Using FastLife

                    Effects of Screen Mode & Size

                    Choosing a Screen Mode & Size

                    Run Life Run! See Life Run!

                    A Bug by Any Other Name

1.13   Effects of Screen Mode & Size

                    WARNING – The Amiga is capable of producing screen modes that will ←↩
                         cause
physical damage to certain types of monitors. See your Amiga
documentation for further information. Do not select screen modes that
your monitor does not support.

You may choose a screen width, height, and one of several screen modes
for the FastLife screen. The screen mode determines the maximum number
of pixels visible vertically and horizontally. Width and height
determine the actual size of the screen, part of which may not be
visible.

For instance, if you select a screen mode of NTSC:Lores, width of
 600 and height of 512, then a life pattern up to 600 by 512 will fit on
the screen. But you will only be able to see a 320 by 200 portion of the
screen at once. Pushing the mouse pointer against the edge of the screen
will cause the screen to scroll.

Another way of looking at this: The screen mode determines how big the
pixels are, width and height determines how many there are.

You may select any screen width or height greater than 32 and smaller
than about 16384. Screen sizes smaller than 320 by 200 may prevent
displaying requesters and menus. You may want to remember the menu
keyboard shortcuts, especially the ScreenSize shortcut, if you choose a
small screen size.

```
Be sure to read
                A Bug by Any Other Name
                below to learn how FastLife
updates your view of the Life universe.
```

1.14 Choosing a Screen Mode & Size

When you start Fastlife it will open a screen mode requester. Select a
screen mode, height and width. You may click on a mode to select it, or
double click to select and close the requester. NOTE - You must press
either <Return> or <Tab> after changing the width or height. You may
click a checkmark to select a default. Click "Ok" to create a screen as
you've selected, or click "Cancel" to quit FastLife. You may use the
right Amiga key with an underlined letter in the requester to select a
button.

FastLife will exit if memory is insufficient for the selected screen
size. If you can't get the screen size you want, try closing other
screens, windows and projects/tools to free up memory. You may even want
to reboot your Amiga in case memory has been fragmented by other
programs.

1.15 Run Life Run! See Life Run!

```
                After opening its screen, FastLife will display its name. You may  ←
                     select
```
the menu item Control-Run and see FastLife run for over 3100 generations
before the pattern becomes stable, or you may immediately select menu
item Project-New or Project-Open to clear the screen (and open a file).

You can use the Left-Amiga-N and Left-Amiga-M key combinations to expose
the FastLife screen or hide it behind the Workbench screen. You can
also click in the small FastLife window on the Workbench to bring
FastLife's screen to the front or use the Project-
```
                Screen To Back
                menu
```
item.

1.16 A Bug by Any Other Name

A bug by any other name would smell... so I call it a feature.

FastLife does its Life calculations in a buffer, then transfers the
result to your screen. The transfer takes a significant amount of time.
When you have a screen that is near the default for a particular screen
mode (e.g., 320x200 for LORES) FastLife updates the entire screen every
generation. When you select a screen size that is very much larger than
the default then FastLife gets lazy and only updates the visible part of

the screen. This helps FastLife run as fast as it can. A side effect
of this might be considered a bug: FastLife only updates the "standard"
part of the screen and not the overscan portion. So, in some cases you
will see a small band at the right edge and the bottom edge of your view
that is not being updated. Note that it is only your view that is not
being updated entirely; FastLife always updates its entire buffer.

If you use one of the number keys '2' through '9' to slow FastLife, then
it decides you are not in a hurry and it will update the entire screen
every generation. If the static band at the edge of your view is a
bother, press the '2' key. FastLife will then update the whole screen,
and you lose a little speed.

1.17 Menus

                   FastLife has the following menus:

            Project

            New

            Open...

            Open Again

            About

            Screen To Back

            Generation

            Show
                    <G>

            Zero
                    <Z>

            Run For...
              <F>

            Stop At...
              <H>

            Screen Size
                 <E>

            Priority

            Lower

            Normal

            Higher

```
                    Quit
                                <Q>

                    Control

                    Once
                                </>

                    Speed

                    Run
                                  <R>
      -----

                    Torus
                                <T>

                    Random

                    Random Field
                    <P>
          ------------

                    Random Seed

                    Set Seed...

                    Set Count...
                    <C>
```

The hierarchy follows:

```
                    Project

                    Control
```

1.18 Project

```
                    New

                    Open...

                    Open Again

                    About

                    Screen To Back

                    Generation

                    Screen Size

                    Priority
```

```
            Quit
```

1.19 New

New clears the screen. Use it when you want to start with an empty
screen.

1.20 Open...

```
            Open allows you to load a new Life pattern from disk.  The  ←
                supplied Life
```
file names end with ".life" (no quotes).

A ReqTools requester will display files that match the displayed
pattern. Upper and lower case do not matter. Click on a name or enter
one. Double click on a file name or select "Ok" to open it. The "get"
button will re-read the directory. NOTE – Earlier versions of FastLife
added ".life" to a file name if it was not supplied. FastLife no longer
does.

The mouse pointer will change to a picture of a glider to indicate
FastLife is waiting for a click. Click on the screen to indicate the
position of the top-left corner of the pattern. FastLife can read four
file types: Absolute, Relative, Picture, and RLE (denoted in the files
with "#A", "#R", "#P" or "#E"). See
```
                File Format
                below.  You can use
```
the right mouse button to cancel picking a screen location.

FastLife will indicate if part of the Life pattern falls off the
screen's edge.

NOTE – The Life files are separated into directories by screen size.
Within the 320 by 200 directory the files are in sub-directories solely
so floppy disk users won't have to wait for the file requester to load
one hundred and fifty filenames when opening a file.

1.21 Open Again

Open Again opens the previous file again. You must again click to show
where to load the pattern on the screen. You may want to select "New"
before using "Open Again". The file name is not lost when "New" is
selected.

1.22 About

About shows the current version number and date, along with credits to
the authors.

1.23 Screen To Back

Screen To Back sends the FastLife screen behind other screens.

1.24 Generation

            Show

            Zero

            Run For...

            Stop At...

1.25 Show

Show shows the current generation number. The first generation is
numbered zero.

1.26 Zero

Zero sets the generation number to zero.

1.27 Run For...

Run For opens a requester. You may select a number of generations to
run. For example, if the current generation is 112, and you select 100,
FastLife will run to generation 212 and stop. You may stop FastLife
before the "Run For" limit is reached.

1.28 Stop At...

Stop At opens a requester. You may select the generation number to run
to. You may stop FastLife before the "Stop At" limit is reached.

1.29 Screen Size

Screen Size opens a screen mode requester. NOTE – The current Life
pattern on the screen will be lost when this menu item is selected. See

Choosing a Screen Mode & Size

.

1.30 Priority

Set FastLife's task priority.

Caveat: Task priorities on the Amiga range from -127 to +127. It might
appear that increasing or decreasing Fastlife's priority by only one
would have little effect. This is not true since the priority number
only establishes an ORDER. The difference between the priorities
matters not. If you select a Higher priority for FastLife it can
prevent any program at a lower priority (such as zero) from running at
all. Since most programs run at priority zero, FastLife with Higher
Priority (1) can stall them completely. Conversely, if FastLife's
priority is Lower (-1) other programs can stop FastLife completely. A
complication to all this is the interaction of programs when they are
waiting. A program with higher priority that is doing nothing can allow
lower priority programs to run (as when a word processor is waiting for
you to type something). Polite programs yield to lower priority programs
as much as possible. If FastLife is running at its fastest (i.e.,
number key '1' selected) it will never yield to lower priority programs,
thus stalling them completely. If you select a number key '2' through
'9' FastLife will be waiting part of the time and will yield to other
programs. If you select a Lower priority for FastLife and a higher
priority program does no waiting (as when you "TYPE" a file in a CLI or
Shell) FastLife will not respond to your commands. The cure for this is
to cause the other program to wait (For instance, stop the mandelbrot
generator, or send Break to the CLI or Shell, etc.).

Lower

Normal

Higher

1.31 Lower

Lower sets Fastlife's task priority to -1. AmigaOS will then allow most
programs to run before FastLife since most programs run at priority
zero.

1.32   Normal

Normal sets FastLife's priority to zero. Most programs run at priority
zero.

1.33   Higher

Higher sets FastLife's priority to one. Most programs must wait on
FastLife when its priority is one since programs typically run at
priority zero.

1.34   Quit

Quit closes FastLife. The <Esc> key will also close FastLife.

1.35   Control

Once

Speed

Run

Torus

Random

1.36   Once

Once calculates one generation. You may want to use the Right-Amiga-/
key combination to step FastLife one generation at a time.

1.37   Speed

Speed will display a brief description of how to slow FastLife.   ←
Slowing
FastLife will allow you to see what is happening in more detail and will
cause FastLife to update the entire screen.

Pressing one of the number keys '1' thru '9' will select fastest through
slowest speed, respectively. '1' introduces no delay. Each higher
numbered key slows FastLife more. '9' introduces about ten seconds of
delay between each generation. Also see the caveat under
Priority
.

1.38   Run

Run is a toggle menu item. It causes the screen to update continuously
or to stop. It will be checkmarked when FastLife is running.

1.39   Torus

Torus is a toggle menu item: Each time you select it you change the
state of the torus option, either on or off. The screen edges wrap
around when the torus option is on: the left edge of the screen touches
the right edge, and the bottom edge of the screen touches the top.

Torus will be checkmarked when the torus option is on.

1.40   Random

                Sometimes you may want FastLife to create a random field and see ↩
                   what
develops.
                Random
                has several sub-items that allow you to control how
random pixels are set. You may control how many pixels are set. You
may select a seed for the random number generator that picks which
pixels to set, or you may ask the computer to select a seed based on the
system time.

                Random Field

                Random Seed

                Set Seed...

                Set Count...

1.41   Random Field

                Random Field sets pixels randomly on the screen.

                Set Count...
                lets you determine how many pixels will be set.

You may ask FastLife to select a
                Random Seed
                   .

You may select a seed with
                Set Seed...

.

Regardless of how you select the seed it will be used once. If you wish
to regenerate a particular field you must use "Set Seed" again. You
cannot (at least you probably cannot) recreate a particular field that
you generated based on a "Random Seed".

1.42 Random Seed

                  Random Seed lets the computer select a seed randomly. The seed ←
                  will be
used when setting a
                  Random Field
                  . The computer will select one of 2.8
x 10**14 seeds randomly based on the system time. You only get one use
of a seed, so you cannot reproduce a field that you generated with
"Random Seed".

Also see
                  Set Seed...
                  .

1.43 Set Seed...

                  Set Seed... lets you select one of 65535 seeds to be used when ←
                  creating
a
                  Random Field
                  . See also
                  Random Seed
                  .

1.44 Set Count...

                  Set Count lets you determine how many pixels will be set when ←
                  creating a

                  Random Field
                  . The default is five thousand pixels. You can use
"Random Field" several times in a row to set more pixels.

Because the pixels are selected randomly, fewer than you specify may be
turned on since a pixel may be selected multiple times but can be turned
on only once.

1.45   Notes

FastLife is VERY dependent on having FAST RAM available. It will run
properly without FAST RAM, but, as you might guess, it will be much
slower. A lifeform that can be updated at 82 generations per second with
FAST RAM will slow to 34 generations per second without it. If you have
FAST RAM and FastLife seems too slow, try running "avail flush" from the
CLI. This is necessary because the Amiga will keep libraries in FAST
RAM, even when no one has them open and provide CHIP RAM to requests
until it runs low on CHIP RAM.

Selecting "Control-Once" or "Control-Run" will appear to do nothing if
there is no Life pattern on the screen. Open a pattern file first.
Also note that selecting speed key <9> may make it appear FastLife is
stalled when it is not. Press the <1> key to run as fast as possible.

You may make multiple menu selections at one time. For example: Press
and hold the right mouse button, point at "Project-New" and click the
left mouse button; point at "Project-Open" and click the left mouse
button. Release the right mouse button. This will clear the screen and
bring up "Open" in one menu action. Multiple menu selections are
explained in your Amiga software manual.

Often FastLife can calculate generations faster than the display rate of
thirty (NTSC) or twenty-five (PAL) frames per second. You may see cases
where gliders or other life forms appear to move without being updated
properly. Don't worry, FastLife is calculating each generation
correctly. Try pressing a number key, <2> through <9>, to slow FastLife
and allow you to see all of the action.

FastLife uses a buffer and a screen to do its work. The buffer will be
in FAST RAM if it is available; the screen memory must be CHIP RAM. The
buffer and the screen each use approximately the same amount of memory:
(ScreenHeight * ScreenWidth / 8) bytes.

1.46   File Format

                    FastLife's patterns are stored in files with names that end in ". ↩
                        life".
Many of the patterns were obtained with xlife version 2.0 for the
X-Windows System, and retain a subset of the formats used there.

You can make your own Life patterns using your favorite text editor.
Follow the format described below and save the file as plain text.
Multitasking on the Amiga will allow you to switch back and forth
between FastLife and your editor to speed your work.

Lines in the files that begin with "#C" are comment lines.

                        Picture File Format

                        Absolute File Format

```
            Relative File Format

            Run-Length Encoded File Format
```

1.47 Picture File Format

The Picture file format has a "#P" in a line before the pattern itself.
Each line that follows represents a line of cells across the screen,
with ALIVECHAR being an ON cell and any other character being an OFF
cell. ALIVECHAR defaults to '*'.

An example of a Picture file is

```
    #P
    **.
    .**
    .*.
```

1.48 Absolute File Format

The Absolute file format consists of pairs of absolute X-Y addresses of
the cells to turn ON, one pair per line. The file may or may not have
"#A" before the pattern.

1.49 Relative File Format

The Relative file must contain a line with "#R" before the pattern. Each
following line has two numbers interpreted as an X-Y address relative to
the position on the screen where you clicked. The upper left of the
image will normally be placed where you click the mouse on the screen.
Two integers may follow the "#R" to offset the image.

1.50 Run-Length Encoded File Format

The Run-Length Encoded (RLE) file format has a line beginning with "#E"
before the pattern. It may also have a line of the form

```
    x = <width>, y = <height>
```

to indicate the size of the life form. The pattern then follows. Here is
an example pattern:

```
    10ob5o3b5o$10bobo12b20
    3b4o!
```

'o' indicates an On cell. 'b' indicates a Blank cell. '$' indicates the
end of the current line on the screen. A number before any of these

means to repeat it that number of times. '!' marks the end of the
pattern. Whitespace in the pattern is ignored.

1.51 Interesting Patterns

Some patterns I find particularly interesting are:

```
Alu.life
BreederRLE.life (Use a SuperHiRes, 1280x350 screen)
bunnies.life
f2.life
glider1_2.life
glider2_2.life
gliderglidergun.life
oscillators.life
p24.life
p26.life
p55.life
rabbit.life
randomgun.life
round.life
switchengine.life
```

1.52 If You Have Trouble

Be sure you have a pattern on the screen before you RUN or it will
appear that nothing is happening.

Be sure the file you are trying to open has #P, #R, #A or #E before the
pattern, otherwise nothing will be displayed.

Press the <1> key to make FastLife run as fast as possible, or press the
<2> key to avoid the static bottom and right edge. Note that dragging
the screen down with the left-amiga key and LMB will confuse FastLife
and the screen may not be updated correctly. Either don't drag the
screen down, or if you do, press the <2> key so that the entire screen
is updated.

Make sure you have FAST RAM available. FastLife runs much slower when it
cannot get FAST RAM.

Close other screens, windows, projects and tools (or reboot) to free
CHIP and FAST memory if you can't get the screen size you want.

Reboot to remove memory fragmentation.

1.53 FastLife and Pre-2.04 AmigaDOS

If you have an earlier release of AmigaDOS you can run an earlier
version (1.1) of FastLife found on Fred Fish's Library disk number 608.
It should run with AmigaDOS 1.2, 1.3 and 2.03 and will use the ARP
library if it is available.

1.54   Tomas Rokicki's Life Program

Tomas Rokicki's Life program, version 6.1, can be found on Fred Fish's
Library disk number 837. Tomas' 6.1 Life routines are responsible for
FastLife's speed.

1.55   Author

```
Ron Charlton
9002 Balcor Circle
Knoxville, TN 37923 USA

(615)694-0800 evenings Eastern Time

Internet:  charlton@cs.utk.edu
           charltr@euclid.eng.ornl.gov
           charltoncr@ornl.gov
```

Please send any comments, bug reports, suggestions, etc., to the author.

FastLife was developed with SAS/C version 6.50 and has been tested with
Amiga Software Release 2.1. It should work with release 2.04 and later.