

SPRITESLIB

Conversion program

COLLABORATORS

	<i>TITLE :</i> SPRITESLIB		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Conversion program	October 9, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	SPRITESLIB	1
1.1	Overview of SPRITESLIB	1
1.2	SPRITESLIB	1
1.3	SPRITESLIB	2
1.4	SPRITESLIB	2
1.5	SPRITESLIB	2
1.6	SPRITESLIB	3
1.7	SPRITESLIB	3
1.8	SPRITESLIB	3
1.9	SPRITESLIB	4
1.10	SPRITESLIB	4
1.11	SPRITESLIB	4

Chapter 1

SPRITESLIB

1.1 Overview of SPRITESLIB

Overview

An Acid Software Library

Converted to AmigaGuide by

Red When Excited Ltd

Used with the permission of Acid Software

Edited, fixed and cleaned by Toby Zuijdveld 27/02/1999.
mailto:hotcakes@abacus.net.au

1.2 SPRITESLIB

Statement: GetaSprite

Modes :

Syntax : GetaSprite Sprite#,Shape#

To be able to display a sprite, you must first create a sprite object. This will contain the image information for the sprite. GetaSprite will transfer the graphic data contained in a shape object into a sprite object. This allows you to perform any of the Blitz 2 shape manipulation commands (eg Scale or Rotate) on a shape before creating a sprite from the shape.

Once GetaSprite has been executed, you may not require the shape object anymore. In this case, it is best to free up the shape object (using Free Shape) to conserve as much valuable chip memory as possible.

1.3 SPRITESLIB

Statement: ShowSprite

Modes :

Syntax : ShowSprite Sprite#,X,Y,Sprite Channel

ShowSprite is the command used to actually display a sprite through a sprite channel. X and Y specify the position the sprite is to be displayed at. These parameters are ALWAYS given in lo-resolution pixels. Sprite Channel is a value 0 through 7 which decides which sprite channel the sprite should be display through.

1.4 SPRITESLIB

Statement: InFront

Modes :

Syntax : InFront Sprite Channel

A feature of sprites is that they may be displayed either 'in front of' or 'behind' the bitmap graphics they are appearing in. The InFront command allows you to determine which sprites appear in front of bitmaps, and which sprites appear behind.

Sprite Channel must be an even number in the range 0 through 8. After executing an InFront command, sprites displayed through sprite channels greater than or equal to Sprite Channel will appear BEHIND any bitmap graphics. Sprites displayed through channels less than Sprite Channel will appear IN FRONT OF any bitmap graphics.

For example, after executing an InFront 4, any sprites displayed through sprite channels 4,5,6 or 7 will appear behind any bitmap graphics, while any sprites displayed through sprite channels 0,1,2 or 3 will appear in front of any bitmap graphics.

InFront should only be used in non-dualplayfield slices. For dualplayfield slices, use InFrontF and InFrontB.

1.5 SPRITESLIB

Statement: InFrontF

Modes :

Syntax : InFrontF Sprite Channel

InFrontF is used on dualplayfield slices to determine sprite/playfield priority with respect to the foreground playfield. Using combinations of InFrontF and InFrontB (used for the background playfield), it is possible to display sprites at up to 3 different depths - some in front

of both playfields, some between the playfields, and some behind both playfields.

Please refer to `InFront` for more information on the `Sprite Channel` parameter.

1.6 SPRITESLIB

Statement: `InFrontB`

Modes :

Syntax : `InFrontB Sprite Channel`

`InFrontB` is used on `dualplayfield` slices to determine `sprite/playfield` priority with respect to the background playfield. Using combinations of `InFrontB` and `InFrontF` (used for the foreground playfield), it is possible to display sprites at up to 3 different depths - some in front of both playfields, some between the playfields, and some behind both playfields.

Please refer to `InFront` for more information on the `Sprite Channel` parameter.

1.7 SPRITESLIB

Statement: `SaveSprites`

Modes :

Syntax : `SaveSprites Sprite#,Sprite#,Filename$`

`SaveSprites` allows you to create a file containing a range of `sprite` objects. This file may be later loaded using the `LoadSprites` command.

The range of `sprites` to be saved is specified by `Sprite#,Sprite#`, where the first `Sprite#` refers to the lowest `sprite` to be saved and the second `Sprite#` the highest.

1.8 SPRITESLIB

Statement: `LoadSprites`

Modes :

Syntax : `LoadSprites Sprite#[,Sprite#],Filename$`

`LoadSprites` lets you load a 'range' of `sprites` from disk into a series of `sprite` objects. The file specified by `Filename$` should have been

created using the SaveSprites command.

The first Sprite# parameter specifies the number of the first sprite object to be loaded. Further sprites will be loaded into increasingly higher sprite objects.

If a second Sprite# parameter is supplied, then only sprites up to and including the second Sprite# value will be loaded. If there are not enough sprites in the file to fill this range, any excess sprites will remain untouched.

1.9 SPRITESLIB

Statement: SpriteMode

Modes :

Syntax : SpriteMode 0=16 1=32 2=64

For use with the capabilities of the new Display library SpriteMode is used to define the width of sprites to be used in the program. The mode values 0, 1 and 2 correspond to the widths 16, 32 and 64.

1.10 SPRITESLIB

Statement: Sprite

Modes :

Syntax : Sprite

1.11 SPRITESLIB

SPRITESLIB

Overview

Command Index

GetaSprite

InFront

InFrontB

InFrontF

LoadSprites

SaveSprites

ShowSprite

Sprite

SpriteMode
