QBLITLIB

Conversion program

COLLABORATORS

	TITLE : QBLITLIB		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Conversion program	October 9, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

QBLITLIB

1.1 Overview of QBLITLIB

```
                              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 QBLITLIB

```
Statement: Queue
--------------------------------------------------------------------------------
Modes  :
Syntax : Queue Queue#,Max Items

The Queue command creates a queue object for use with the QBlit and
UnQueue commands. What is a queue? Well, queues (in the Blitz 2 sense)
are used for the purpose of multi-shape animation. Before going into
what a queue is, let's have a quick look at the basics of
animation.

Say you want to get a group of objects flying around the screen. To
achieve this, you will have to construct a loop similar to the
following:

Step 1: Start at the first object
Step 2: Erase the object from the display
Step 3: Move the object
Step 4: Draw the object at it’s new location on the display
Step 5: If there are any more objects to move, go on to the next object and then ↩
   go to step 2, else...
Step 6: go to step 1
```

Step 2 is very important, as if it is left out, all the objects will leave trails behind them! However, it is often very cumbersome to have to erase every object you wish to move. This is where queues are of use.

Using queues, you can ’remember’ all the objects drawn through a loop, then, at the end of the loop (or at the start of the next loop), erase all the objects ’remembered’ from the previous loop. Lets have a look at how this works:

```
Step 1: Erase all objects remembered in the queue
Step 2: Start at the first object
Step 3: Move the object
Step 4: Draw the object at it’s new location, and add it to the end of the queue
Step 5: If there are any objects left to move, go on to the next object, then go ↩
   to step 3; else...
Step 6: Go to step 1
```

This is achieved quite easily using Blitz 2’s queue system. The UnQueue command performs step 1, and the QBlit command performs step 4.
.
Queues purpose is to initialize the actual queue used to remember objects in. Queue must be told the maximum number of items the queue is capable of remembering, which is specified in the Max Items parameter.

1.3 QBLITLIB

```
Statement: QBlit
--------------------------------------------------------------------------------
Modes  :
Syntax : QBlit Queue#,Shape#,X,Y[,Excessonoff]
```

QBlit performs similarly to Blit, and is also used to draw a shape onto the currently used bitmap. Where QBlit differs, however, is in that it also remembers (using a queue) where the shape was drawn, and how big it was. This allows a later UnQueue command to erase the drawn shape.

Please refer to the Queue command for an explanation of the use of queues.

The optional Excessonoff parameter works identically to the Excessonoff parameter used by the Blit command. Please refer to the Blit command for more information on this parameter.

1.4 QBLITLIB

```
Statement: UnQueue
--------------------------------------------------------------------------------
Modes  :
Syntax : UnQueue Queue#[,BitMap#]

UnQueue is used to erase all ’remembered’ items in a queue. Items are
placed in a queue by use of the QBlit command. Please refer to Queue
for a full explanation of queues and their usage.

An optional BitMap# parameter may be supplied to cause items to be
erased by way of ’replacement’ from another bitmap, as opposed to the
normal ’zeroing out’ erasing.
```

1.5 QBLITLIB

```
Statement: QBlitMode
--------------------------------------------------------------------------------
Modes  :
Syntax : QBlitMode BLTCON0

QBlitMode allows you to control how the blitter operates when QBlitting
shapes to bitmaps. Please refer to BlitMode for more information on
this command.
```

1.6 QBLITLIB

```
Statement: FlushQueue
--------------------------------------------------------------------------------
Modes  :
Syntax : FlushQueue Queue#

FlushQueue will force the specified queue object to be ’emptied’,
causing the next UnQueue command to have no effect.
```

1.7 QBLITLIB

```
                .--------------------------------------------------------------------------
|                                  QBLITLIB                                 |
`---------------------------------------------------------------------------'


                     Overview
                                               Command Index
```