SCREEN ANIMATION GENERATOR

Copyright 1989 ALBEGA Software

SCREEN ANIMATION GENERATOR

Introduction

This new software for the LYNX micro-computer has been written to enable owners to easily generate their own graphic screens, with active sprites if required.

At the time of writing (Mar '89), this booklet relates to Vers 1.1, further enhancements are planned.
Also supplied with this software are several examples of what effects can be achieved. These can be loaded from the disk by using the ESC L sequence.

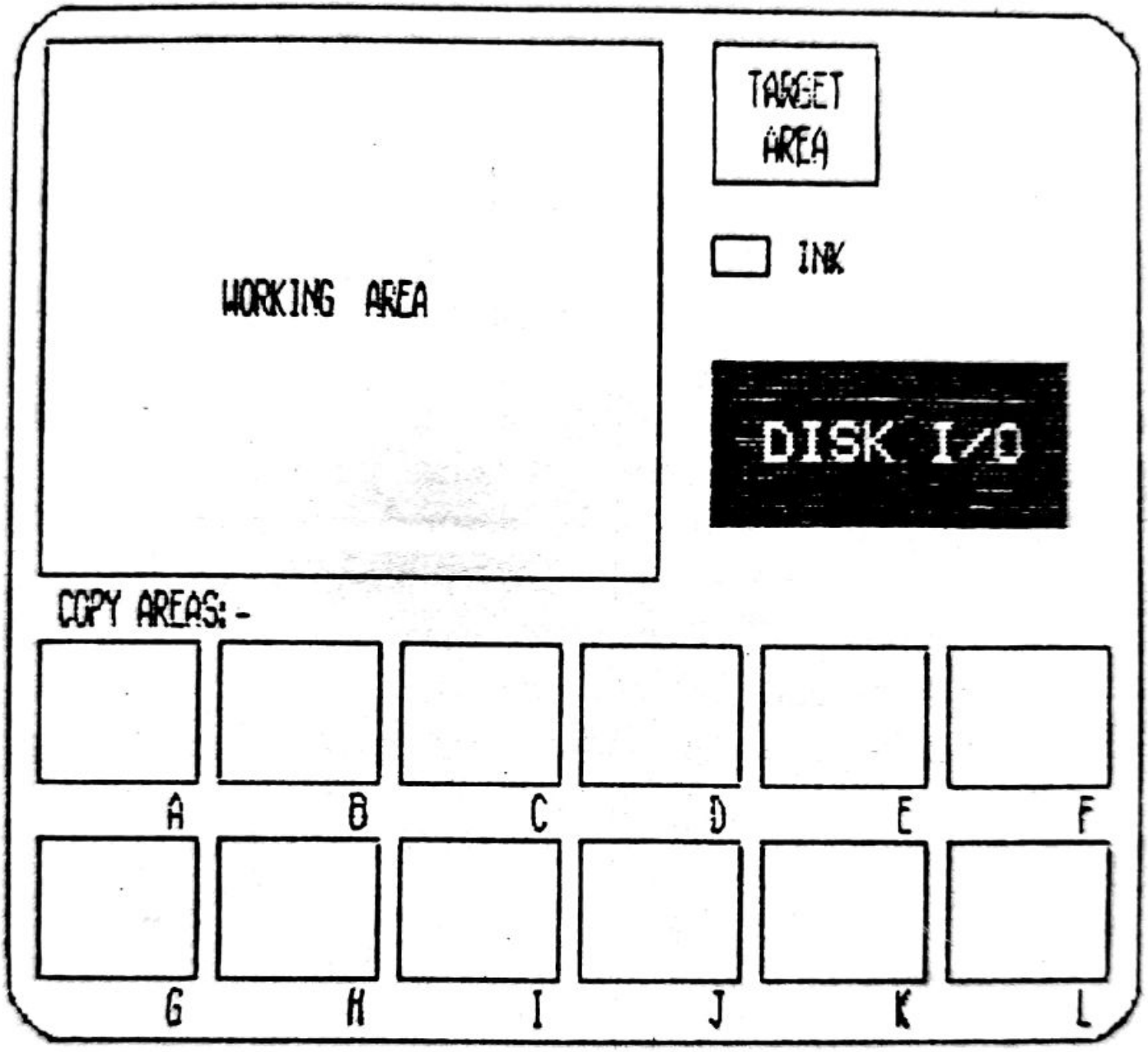
Unlike Vers. 1.0, which used a BASIC loader program, this utility is entirely machine code. Hence to use this software, use the disk instruction:-

EXT MLOAD "ANIM"

A representation of the screen is shown on a later page.

NB. This utility will currently only operate on a standard LYNX 96K up to Issue 3. Versions for Issue 4 and the 128K LYNX will follow at a later date. Due to the screen access routines, any modifications to the Banks 2 and 3 may cause failure of the software to operate correctly.

THE ANIMATION SCREEN



The above drawing represents the screen display. The screen is divided into different areas :-

1) The WORKING AREA
2) The TARGET AREA
3) The COPY AREAS
4) The DISK I/O area.

1) WORKING AREA

This is shown as a large blue grid representing a magnified view of the adjacent TARGET AREA. The WORKING AREA shows a section of screen memory which is FOUR bytes wide and effectively FOUR bytes deep. These are then subdivided down to BIT level. Thus the WORKING AREA represents a screen area of 32x32 pixels.
The cursor is shown as a small square surrounding each of the pixel areas. The cursor is simply moved by the arrow keys. Using either SHIFT key with these will move the cursor rapidly 8 pixels in the selected direction.
Any change to the WORKING AREA is immediately transferred to the TARGET AREA.

On start up, all screen areas are blank, with the cursor sited at the top LH corner of the WORKING AREA.

The current INK colour is shown also adjacent to the WORKING AREA. The INK colour can simply be changed by pressing the corresponding number key:-

0 gives BLACK
1 gives BLUE
2 gives RED
3 gives MAGENTA
4 gives GREEN
5 gives CYAN
6 gives YELLOW
7 gives WHITE

When the required pixel is outlined by the cursor, pressing the SPACE-BAR will set that pixel in both the WORKING AREA and the TARGET AREA. Using the SPACE-BAR with either SHIFT key enables contiguous lines to be drawn.

Pressing 'F' will fill that byte area in which the cursor currently resides. Thus a block of 8x8 pixels can be rapidly filled. Using 'F' with either SHIFT key will fill the entire WORKING AREA in the current INK colour. The TARGET AREA is also correspondingly filled.

See ESC 'B'.

2) TARGET AREA

As the TARGET AREA is the actual screen area represented by the WORKING AREA, only a few further points need be added.

When animation is called for, only the TARGET AREA is activated, this is covered by the function ESC 'A'.

Transfer of the design in the TARGET AREA, is initiated by the ESC 'P' or PUT command by specifying the COPY AREA letter code.

Any of the 12 COPY AREAS can be transferred to the TARGET AREA; which also provides the enlarged image in the WORKING AREA. This is done by means of the ESC 'G' or GET function, specifying the COPY AREA code letter.

3) COPY AREAS

Provision is provided for up to 12 COPY IMAGES to be created. Each IMAGE is identified by a code letter 'A-L'.

Not all of the IMAGES need be designed as provision is now provided to set the animation sequence over any range, set by the ESC 'A' function.

Any COPY IMAGE can be recalled at any time for further work, via the ESC 'G' function and either returned to that allocation or any other via the ESC 'P' function.

During the animation sequence, the COPY IMAGES are cycled through in order, appearing in the TARGET AREA, but in the direction A-L (or whatever) and not in reverse. This must be borne in mind when designing a sequence. Reverse action can be created by designing each of the COPY IMAGES in reverse order.

4) DISK I/O AREA

This is a transient area, and is not normally visible. It only appears when disk access is required, either for Loading or Saving a file.

After disk access is finished, then the area reverts to the red surround.

See functions for details of disk access.

FUNCTIONS

All functions are accessed by pressing the ESCape key together with the appropriate function letter. However certain functions have sub-functions and in these cases, the ESCape key is not required.

All functions including sub-functions require the RETURN key for input. In the case of sub-functions, double action is required of the RETURN key to escape the function mode.

Function letters:-

A Animate, currently 4 sub-functions.
B Background colour.
C Colour change.
G Get copy.
H Help, not yet implemented.
L Load, disk function.
M Mirror image.
P Put copy.
Q Quit, exit program.
S Save file, disk function.
T Traverse image.

Function 'A'nimate.

Once a series of COPY IMAGES have been made, these can be shown as animation in the TARGET AREA. Press ESC 'A' to enter animation mode. At this point, four sub-functions become available:-

A Auto or animate the set sequence, set by the L and T sub-functions.

L Last frame. User choice on which is to be the last frame in a sequence by pressing either the LEFT ARROW key or RIGHT ARROW key. Use the RETURN key to lock in the final frame before animation.

S Single step allows the user to step through the IMAGES one at a time. The letter code of each IMAGE currently addressed is also shown.

T Timer adjustment, allows speed of the animation to be set. Range is 0-8, where 0 is the fastest sequence of cycling the IMAGES and 8 is the slowest. Like the sub-function 'L', use the arrow keys to select followed by RETURN to enter.

Function 'B'ackground colour
This is similar to the previously mentioned WORKING AREA 'F'ill command. Whereas 'F' operates on an 8x8 area, and SHIFT 'F' fills the full 32x32 area, these only relate to the current screen being worked on.

ESC 'B' is a GLOBAL fill in the current colour and will flood all display areas.

By choosing BLACK (0), this feature can be used as an erase feature to start a new sequence of screen segments.

*_*_*_*_*_*

Function 'C'

Pressing ESC 'C' allows the user to change an existing colour to any other of their choice.
The sequence is: press ESC 'C', the system then waits for input, select the current colour number followed by the new colour number. This will change all the original pixel colours on both WORKING AREA and TARGET AREA.

Function 'G'et copy from COPY AREA

Pressing ESC 'G' will activate the transient DISK I/O area and the system will wait for a single letter input in the range A-L. The corresponding COPY IMAGE will then be copied to both the WORKING AREA and the TARGET AREA for further processing.
When the user has finished with amending the image, it can either be returned to the original COPY IMAGE position or sent to any other position via the ESC 'P' function.

*_*_*_*_*_*

Function 'H'elp

This will be the on-screen help function but at this time (Mar '89), this has not been implemented.

*_*_*_*_*_*

Function 'L'oad file from disk

This function activates the transient DISK I/O area on the screen.
Pressing ESC 'L' will then wait for input from the user. Unlike LYNX DOS, no quotes are required before or after the file name.
It is recommended that the user keeps a record of file names seperately as at present, any error in name for whatever reason, will result in the software faulting back to BASIC.
It is planned to trap this irritation out in later versions.

*_*_*_*_*_*

Function 'M'irror image

Pressing ESC 'M' puts the system into a mirror or flip mode and then waits for input from the user. The only keys which affect the displayed image are the ARROW keys. LEFT or RIGHT will mirror the image symmetrically about the centre line. Similarly, the UP or DOWN keys will mirror the image in the vertical sense.

*_*_*_*_*_*

Function 'P'ut image to COPY AREA

This is the reverse to the 'G'et function. ESC 'P' will wait for the user to specify the COPY IMAGE letter code, A-L. This transfers the current TARGET AREA image to the selected COPY AREA position. Note, neither the TARGET AREA image nor the WORKING AREA image is cancelled, so further changes can be implemented.

*_*_*_*_*_*

Function 'Q'uit system

Pressing ESC 'Q' will return the user to BASIC. If "ANIM" is required again, this will have to be freshly loaded in from disk again, as 'Q'uiting causes corruption in the software.

*_*_*_*_*_*

Function 'S'ave image file to disk
After completing the image sequence, as long as a disk is ready in the drive, using ESC 'S' will store the 12 COPY IMAGES held currently in the DATA STORE to disk.
NB. Failure to have the disk ready in the drive will result in an error code being generated, the software will then return the user to BASIC, with the resulting loss of the images.
It is not necessary to erase similar titled files, as the system will overwrite these on disk. This is useful when working on previously recorded data, but care is needed with file names in case previous files are over-written. This is why it is important for the user to keep documentation on the file names recorded and what they relate to.

*_*_*_*_*_*

Function 'T'raverse image

Pressing ESC 'T', will put the user into Traverse mode. It directly affects the TARGET AREA image only and allows the image to be incremented in position by means of the ARROW keys.
Using the ARROW keys only will move the image in one pixel steps.
Using a SHIFT key with the ARROW keys, will cause the image to shift by 8 pixels.
After the new position has been selected, pressing RETURN will transfer the TARGET image to the WORKING AREA, where "repair" work can be carried out.
NB. If the image has been shifted by this method, screen edge data is destroyed and cannot be recovered, so care is necessary in moving the image in the TARGET AREA.

-

TECHNICAL NOTES

The current image file is stored in the DATA STORE, from address &0000 to &1380 (4992 dec.). Each image occupies &0180 bytes, stored in order, BLUE-RED-GREEN.