

DISPLAY SYSTEM

12-3-84

INDEX

INTRODUCTION.....	1
FEATURES.....	1
PICTORIAL OVERVIEW:RESOURCE USAGE.....	2
PICTORIAL OVERVIEW:WORLD COORDINATE SYSTEM.....	3
DATA STRUCTURES.....	4
SYSTEM CONTROL VARIABLES.....	5
GLOSSARY OF FUNCTIONS.....	6
3/1 <u>ICON IDENTIFIERS</u>	9
SOURCE.....	9

INTRODUCTION - The Display System is a collection of routines found in the kernal and in an auxillary overlay (+ICON-OV) which are used to produce a view of the world through the main view screen of the starship and the terrain vehicle (excepting the rotating planet and the landing sequence which are handled in a separate overlay). Collision checking is also done within the Display System.

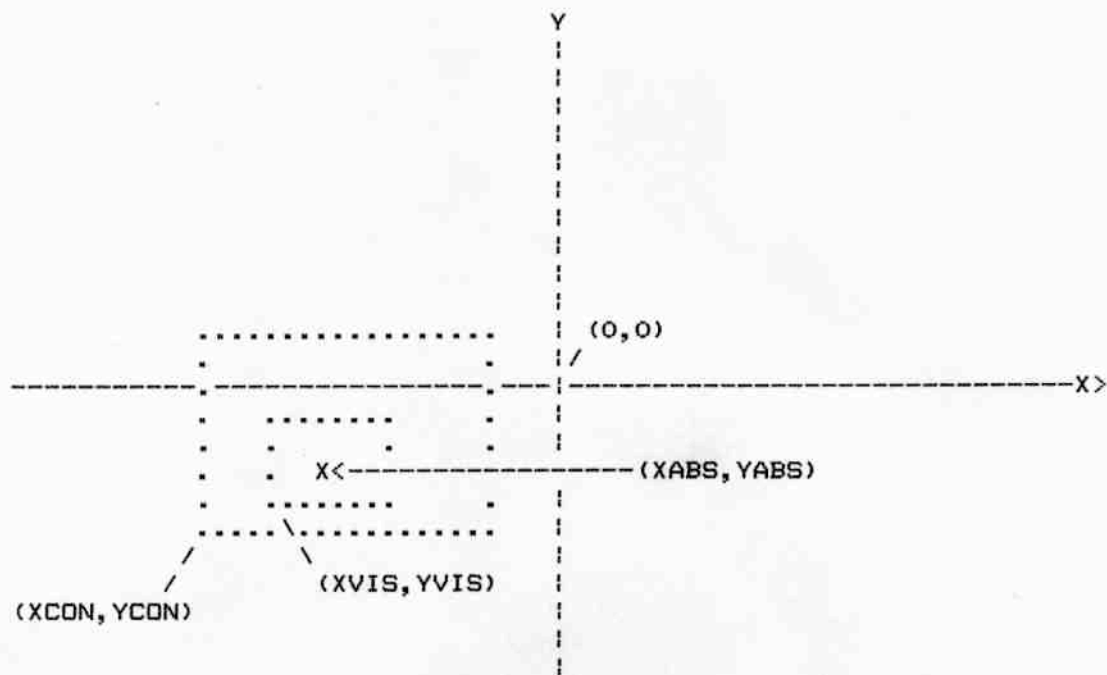
FEATURES - A summary of the Display System (DS) features follows:

1. DISPLAY WINDOW SELECTION - The location of the viewing window with respect to the world coordinate system can be selected.
2. AUTOMATIC ICON CHOICE GIVEN INSTANCE - given an instance address a routine selects the appropriate icon based on context and instance status (ie. heading, bio-status etc.) and installs it in the display list.
3. ICON DISPLAY WITH CLIPPING - only those icons are displayed that are considered visible.
4. SCALES OF MAGNIFICATION - there are 3 scales of icon sizes and cooresponding icon cell sizes.
5. SINGLE AND COLLECTIVE ICON FUNCTIONS - icons can be displayed, erased or moved indiviually or collectively.
4. VECTORED BACKGROUND DRAWING - a routine that draws the field upon which the icons are to be placed can be installed so that it is used each time the icons are updated.
5. COLLISION CHECKING - a routine is provided that will return a list of all icons at a given x,y coordinate.
6. INSTANCE SELECTION BY REGION - returns a list of all icons within a square region centered on a given x,y coordinate.
7. ICON>INSTANCE LOOKUP - returns the instance address given the icon display list number.
8. KERNAL RESIDENT FAST ARRAY WORDS

PICTORIAL OVERVIEW: RESOURCE USAGE

DISK	MEMORY
	=====
	GRAPHICS BUFFER (MAIN VIEW)
OHS FILE	------(USER MEMORY LIMIT)
	ICONLIST ARRAY
ICON IMAGE FILES	ICONIMAGE ARRAY
	MAIN VIEW BUFFER
	(11K SCRATCH SPACE)
	------(TOP OF CACHE)
	CACHE
	-----64K
	<=====> DISK BUFFER
+ICON-OV OVERLAY	+ICON-OV OVERLAY
	PAD(18 BYTES)
	-----KERNEL
	DISPLAY SYSTEM FARMS
	FUNCTION CODE
	=====

PICTORIAL OVERVIEW: WORLD COORDINATE SYSTEM



ADDENDUM TO DISPLAY SYSTEM DOCUMENTATION
PLANETARY COORDINATE RESTRICTIONS
12-11-84

RULES:

(see the source for the site selection overlay SITE-OV.CMP
for words using the following)

0 <= Planetary x coordinates <= 2303
0 <= Planetary y coordinates <= 959
XCON must be a multiple of 12
YCON must be a multiple of 20

Degree of LATITUDE is computed by:

YABS @ 480 - 10 53 */
if negative, south latitude
if positive, north latitude
if zero, equator

Degree of LONGITUDE is computed by

XABS @ 1152 - 10 64 */
if negative, east longitude
if positive, west longitude
if zero, prime meridian

DATA STRUCTURES

ICONLIST ARRAY

The ICONLIST array is a 9 x 300 byte array existing in high memory. The first row (0,0) of the array contains the array status information as follows:

```
ICONLIST--->| #ICONS | FREE-ICON | SET-ICON | ( un-used ) |
              0-----2-----4-----7
```

The remaining 299 rows contain the following information:

```
1-299 | ICON-X | ICON-Y | ICON-ID | ICON-CLR | ICON-IADDR |
        0-----2-----4-----5-----6-----9
```

Where:

ICON-X = x coordinate in world coordinate system
ICON-Y = y coordinate in world coordinate system
ICON-ID = identifier code for the icon (set by +ICON)
ICON-CLR = color override code (set by +ICON)
ICON-IADDR = instance address of cooresponding object.

ICONIMAGE ARRAY

The ICONIMAGE array is a 18x50 byte array existing in high memory. BLT image information for icons that are BLTs is contained in this array. This array is filled from disk files by the >1:nSCALE words. The format is as follows:

```
0-49 |color| image bits |color| image bits |
      0-----1-----10-----11-----18
```

```

*.CELL ..... Array cell plot routine vector
*COLORMAP ..... Colormap function vector (false coloration)
*.BACKGROUND ..... Background plot routine vector
*SET.BACKGROUND ..... Install-background-plot-routine-vector
                    routine vector ( this is a forward
                    reference required if the background plot
                    routine resides in and overlay.).
*.ICON-BACKGROUND .... plot background for single icon cell
                    routine vector ( similar to *.cell).
XCON YCON ..... World coordinates for lower left corner of
CONTOUR array on the planet's surface
XVIS YVIS ..... World coordinates for lower left corner of
visible portion of world at 1:4 scale.
XABS YABS ..... World coordinates of starship or terrain
vehicle.
XLLDEST YLLDEST ..... Lower left corner in display buffer
XLL YLL XUR YUR ..... Current region lower left and upper right
GLOBALSEED ..... Master terrain generation seed which is
fixed for a given planet.
SPHEREWRAP ..... Flag indicating that array cell address
computations should use spherical wrapping
when out of bounds.
SIGNEXTEND ..... Flag indicating that byte array fetch
operations should sign extend to word
length when placing them on the parm
stack.
CURRENT-SCALE ..... variable containing the current scale
(1,2,4 where 1 is full scale, 2 half etc)
CONTEXT-ID# ..... indication of the current spacial context
                    0 = planet surface
                    1 = orbit level
                    2 = system level
                    3 = hyperspace level
                    4 = encounter space
(SYSTEM) ..... instance address for current system
(ORBIT) ..... instance address for current orbit
(PLANET) ..... instance address for current planet
(ENCOUNTER) ..... instance address for current encounter
?NEBULA ..... indicator of whether in a nebula or not

```


GLOSSARY OF FUNCTIONS

ARRAY FUNCTIONS

```

LONGBYTEARRAY ..... ( xcols yrows seg <array name> -- \
                        define a long array)
                        ie. 9 9 5555 LONGBYTEARRAY SAMPLE1

<array name> ..... ( x y -- seg addr \ given column and
                        row return the segment and offset of
                        the cell) ie. 4 5 SAMPLE1 yeilds the
                        seg and offset for cell (4,5)

!OFFSETS ..... ( pfa-array -- \ stores row offset
                        lookup table at end of array. The
                        lookup table is used by all array cell
                        access operations.)

SETLARRAY ..... ( pfa-array -- \ sets array as the
                        current array for implied array
                        operations.)

FULLARRAY ..... ( -- x11 y11 xur yur \ get region
                        boundaries for full current
                        array)

SETREGION ..... ( x11 y11 xur yur -- \ set region
                        boundaries within an array for implied
                        region operations.)

FILLREGION ..... ( c -- \ fill current region of current
                        array with c)

.1X1CELL ..... ( x y -- \ plot a 1x1 pixel cell using
                        the current color model and array)
.1X2CELL ..... ( x y -- \ plot a 1x2 pixel cell)
.2X2CELL ..... ( x y -- \ plot a 2x2 pixel cell)
.4X4CELL ..... ( x y -- \ plot a 4x4 pixel cell)
.8X8CELL ..... ( x y -- \ plot a 8x8 pixel cell)

.REGION ..... ( -- \ plot the current region of the
                        current array into the current display
                        buffer using the current destination
                        origin and the current cell plot
                        routine using the current color model.
                        This routine is used to draw the
                        background contour map on the planet
                        surface.)
    
```


ICONLIST FUNCTIONS

```

INIT-ICONLIST ..... ( -- \ initialize ICONLIST array)

+ICON ..... ( -- \ install the current instance
into ICONLIST. This word is contained
in the +ICON-OV overlay)

+ICON-BOX ..... ( -- \ install the contents of the
current container into ICONLIST. This
word is contained in the +ICON-OV
overlay).

-ICON ..... ( i-addr -- \ delete the icon list
entry for instance i-addr)

?ICONS-AT ..... ( x y -- n n n .. n n cnt \ find icons
with the given x,y and leave a count)

?ICONS-LOCUS ..... ( x y radius -- n n n cnt \ find all
icons within radius of x y. Rough
distance computation uses {dx AND
dy}<radius)

.ICON ..... ( -- \ display the current icon)

.ICONLIST ..... ( -- \ display all visible icons in the
display list using the current
background and clipping parameters)

.STACKED-ICONS ..... ( n n n...n n cnt \ display the visible
icons on the stack)

MOVE-.ICON ..... ( iaddr -- \ erase icon for iaddr and
update blt parms, and redraw if
visible.)

ERASE-ICON ..... ( -- \ erase the current icon only if
it is visible, leaving any coincident
icons in tact)

@SET-ICON ..... ( -- n \ get the number of the current
icon in ICONLIST)

@#ICONS ..... ( -- n \ get the number of icons in the
list)

POINT>ICON ..... ( n -- \ set current icon pointer to
icon # n)

POINT>ICON=IADDR ..... ( iaddr -- \ set current icon pointer
to the entry with iaddr)

?VIS-ICON ..... ( -- t \ test if current icon is
visible)

```

ICON-ADDR (n -- seg offset \ compute cell
address relative to current icon at a
byte offset of n.)

MISC. FUNCTIONS

>1:1SCALE (-- \ set up for full scale icon
display)

>1:2SCALE (-- \ set up for half scale icon
display)

>1:4SCALE (-- \ set up for quarter scale icon
display)

.BACKGROUND (-- \ plot the entire background using
the current plot routine)

COLORMAP (val -- color-code \ false color based
on val and current color model)

!XYSEED (x y -- set fractal seed based on
location and GLOBALSEED)