A PSEUDO THREE DIMENSIONAL VIDEO GRAPHICS SYSTEM

DESIGNER: M. FILIPAK

TABLE OF CONTENTS:

SYSTEM FEATURES:

Pseudo three dimensional display, ie, 2D objects in 3D space,

High resolution 764 pixel/line by 246 line screen (NTSC) where one pixel is equal to 1/4 color clock (0.025" on a 25" television),

256 levels of depth into the screen with automatic display priority,

Display priority can be changed with a single CPU store (as opposed to a fixed priority which requires all objects to be reshuffled),

The 764x244x256 display is within a 2048x1024x512 virtual space to simplify scrolling along x, y and z,

Two pixel position resolution resulting in 382 positions across the visible portion of the screen (for a 25" television this translates to .05" per increment of position),

Two pixel resolution in color resulting in 382 color changes across the visible portion of the screen,

Single pixel resolution in intensity resulting in 764 intensity changes across the visible portion of the screen,

Eighty character text on a standard NTSC broadcast television,

2763 color/lum combinations in the palette (an average of 8-1/2 luminance shades of 326 basic colors) and additionally 16 separately controllable intensity levels for each pixel for a total of 44,208 color/lum/intensity combinations,

Programmable palette gives the graphics designer full control over all aspects of pixel color (ie, hue, luminance & intensity) in an easily understood manner,

Fully interlaced repeat field displays for compatability with videodisc and other electronic media,

The composite video is generated synthetically and in baseband so that the signal is ready to be injected into the channel 2/3 moduator without color sub-carrier quadrature modulators, ratioing circuits or color phase delay lines thereby reducing parts count and cost, component complexity, quality assurance overhead, frequency alignment overhead, failure rates & color drift between samples and over time,

A single system clock frequency adjustment at the end of the assembly line simultaneously aligns the color burst frequency, the color phase circuity, the color sub-carrier frequency & the scan, line and field counters (in essence, everything except the channel 2/3 modulator and the audio sub-carrier), Automatically detects presence of external video input and synchronizes to its signal (for videodisc, etc.) & displays it when not displaying objects or, if no external video, displays background color,

Sprite type graphics objects defined by position (x,y,z) and height,

Up to 18,432 independent (visible) sprites (49,152 virtual sprites ready for scrolling into the visible screen) which can be used as either motion sprites or playfield sprites without differentiation on the hardware level allowing for maximum flexability in programming,

Four classes of sprites (color, intensity, mixed & special),

Twenty-three types of sprites with the data densities and bandwidth required for each optimized for broadcast television systems (NTSC, PAL & SECAM),

Sprites can be grouped together to form large pseudo 3D objects which can then be repositioned with only three CPU stores,

Sprites can be laminated one upon another to add detailed sections to otherwise low detail areas,

Playfield sprites can easily be used to create a 3D playfield with up to 256 levels of foreground/background objects,

Sprites are generated and regenerated without CPU involvement, without matrix transforms and without peripheral or internal math packs,

Anti-aliasing designed into objects by the graphic artist in a stright forward, easily understood and predictable manner, Distributed processing system architecture with the graphics subsystem separate from the main system allowing the CPU to run at full speed without wait states or halts,

The system is expandable (it allows for the future addition of a fully 3 dimensional video subsystem) and it is upgradable (a future fully 3 dimensional video subsystem can be added with minimal redesign and in minimal time to play old or new games),

Custom chips utilize hardwired logic (not microcoded) allowing relatively low clock rate permitting larger chip geometry resulting in increased yield and

Custom graphics chip designed using standard cell technology with spares on chip which can be used as needed to further increase yield.

SYSTEM LIMITATIONS/DRAWBACKS:

Extremely high horizontal resolution results in non-square pixels which are approximately 0.061" high by 0.026" wide (on a 25" television) slightly complicating circle drawing routines and the like,

Rotations must be accomplished by the CPU (by rotating the graphics),

Zooming (or shinking) of sprites moving toward (or away from) the screen must be accomplished by the CPU (by zooming or shrinking the graphics),

True perspective positioning must be done by the CPU and

The hardwiring of logic in the custom chips (as opposed to micro-coded logic) will make any modifications to these chips difficult, time consuming and costly.

THE MAIN SYSTEM:

1, a sixteen bit CPU;

2, 917K words of system memory space consisting of:

16K words of Operating System ROM for system operation, interrupt processing, input/output management (contoller routines, sound routines, videodisc handlers, playcable loaders, etc.), graphics routines & software signiture,

1/2K words of Operating System EEPROM (electrically eraseable programmable read only memory) for game parameter store which can remember high scores, skill level, game progress, etc. so that games can be resumed after power has been turned off for periods of up to ten years,

16K words of memory mapped I/O,

885K words of mixed media/system RAM (media can be ROM as in our present products or media subsystem consisting of videodisc reader, playcable loader, etc.),

64K words of addressing space to allow for a future fully three dimensional subsystem and

15.7M words of spare addressing space for expansion and

3, VIVIAN, a custom chip to perform the following functions:

dynamic system power-up configuration using CPU micro-code store to allow one basic architecture to handle a broad mix of memory and I/O configurations and speeds,

memory map control (gate signals) and

memory timing control (multiplex signals).

THE VIDEO SUBSYSTEM:

1, 48K words of graphics RAM containing sprite graphics and parammeters from the CPU is overlayed into the CPU address space by VIVIAN;

2, from one to three PENNYs, a custom chip, each of which performs the following functions:

simultaneous generation of 32 sprites using the parameters and data stored in graphics RAM,

reuse of each sprite generator up to 82 times per screen (assuming all are single line sprites),

programmable grouping of sprites to form larger pseudo 3D objects in x,y,z,

automatic visual prioritization (in z) for all sprites within each chip with transparency pixels allowing any sprites 'behind' to show through,

support of a virtual space over eight times larger than display space to ease program maintenance of sprite positions,

support of a display space larger than the actual screen to simplify simultaneous x,y scrolling,

output of a four bit color (index) per pixel on the fly, with a color index of zero designating transparency, for use by the palette RAM and

output of a four bit intensity per pixel on the fly, with an intensity of zero designating full (stored) intensity for use by the HEATHER chip;

3, 4K words of color palette RAM containing chrominance and luminance selection data from the CPU to the HEATHER chip with an intermediate pixel by pixel lookup supplied on the address lines from the PENNY chips color outputs (it is overlayed into the CPU address space by VIVIAN for initial color stores) and

4, HEATHER, a custom chip, which performs the following functions:

Generates the baseband composite video signal,

Syncronizes to and displays an external video signal (when all sprites are showing transparent) and

Generates the various system clocks.

SYSTEM MEMORY MAP:

```
           ______________
FFFFFF:  | UNASSIGNED    |
         |               |
         .               .
         . 15.7M TOTAL   .
         .               .
         |               |
100000:  |_______________|
         | FUTURE 64K    |
         | 3D GRAPHICS   |
         . SUBSYSTEM     .
         .               .
         .               .
         |               |
0F0000:  |_______________|
         | UNASSIGNED    | \
         | RAM           |  |
         |_______________|  |
0EC000:  |_4K_PALETTE____|  |
         | 16K PENNY 2   |  |
         | GRAPHICS RAM  |  |
         |               |  |
0E8000:  |_______________|   \  GRAPHICS
         | 16K PENNY 1   |   /  SUBSYSTEM
         | GRAPHICS RAM  |  |
         |               |  |
0E4000:  |_______________|  |
         | 16K PENNY 0   |  |
         | GRAPHICS RAM  |  |
         |               |  |
0E0000:  |_______________| /
         | SYSTEM RAM    |
         |       |       |
           |      v       |
           .              .
           .  869K TOTAL  .
           .              .
           |      ^       |
           |      |       |
           | SYSTEM MEDIA |
  00C000:  |______________|
           | 16K I/O MAP  |
           |              |
           |              |
 008000: |______________|
         | 16K EEPROM   |
       |            |
       |              |
  004000:  |______________|
           | 16K OS ROM   |
           |              |
           |              |
  000000:  |______________|
  SYSTEM BLOCK DIAGRAM:                            PAGE 8 OF 33
  +-------+                    +--------+
  |  CPU  |                    | VIVIAN |===>STORE/RECALL (TO EEPROM)
  |    A/D|<============>H====>|        |
  |     MC|===========>H======>|        |==>H MEMORY MAPPED SELECTS
|  dtack|<-----------H-H-----|          |       H
  +-------+      H H     +-------+        H    +-------+
                      H H     MEM MANAGER  H==>|C GATE |===> MEDCTRL
  +------------+       H H<====================>|A/D    |<==> MEDA/D
  | DYNAMIC  cs|<------H-H------------------H   +-------+
  | RAM     A/D|<=======>H      +-------+   H   +--------+
  |          MC|<=====>H H=====>|       |======>|A  ROM   |      16Kx16
  |            |       H=======>| LATCH |   H-->|cs       |      SYSTEM
  +------------+       H H      +-------+   H   |         |         ROM
    SYSTEM RAM         H H<=====================|D        |
                       H H                  H   +--------+
                       H H      +-------+   H   +-------------+
                       H H=====>|       |======>|A  EEPROM    | 1/2Kx16
                       H=======>| LATCH |   H-->|cs SHADOWED  |  SYSTEM
                       H H      +-------+   H   |   RAM       |  EEPROM
                       H H<====================>|D _          |
                       H H------------------H-->|r/w          |
  +-------+            H H  store/recall=======>|C            |
  | PENNY |            H H      +-------+   H   +-------------+
  |  INTEN|=[B,8]=>H   H H<====>|       |<--H   +----------+
  |  COLOR|==[11,8]=>H H=======>| MUX/  |   H   |  DYNAMIC |    16Kx16
  |    A/D|<===================>| GATE  |<=====>|A/D RAM   |   GRAPHIC
  |     MC|====================>|       |======>|MC        |       RAM
  +-------+        H H H H      +-------+   H   +----------+
  +-------+        H H H H                  H
  | PENNY |        H H H H      +-------+   H
  |  INTEN|=[7,4]=>H H H H<====>|       |<--H   +----------+
  |  COLOR|===[7,4]=>H H=======>| MUX/  |   H   |  DYNAMIC |    16Kx16
  |    A/D|<===================>| GATE  |<=====>|A/D RAM   |   GRAPHIC
  |     MC|====================>|       |======>|MC        |       RAM
  +-------+        H H H H      +-------+   H   +----------+
  +-------+        H H H H                  H
  | PENNY |        H H H H      +-------+   H
  |  INTEN|=[3,0]=>H H H H<====>|       |<--V   +----------+
  |  COLOR|===[3,0]=>H H=======>| MUX/  |   |   |  DYNAMIC |   16Kx16
  |    A/D|<===================>| GATE  |<==|==>|A/D RAM   |  GRAPHIC
  |     MC|====================>|       |===|==>|MC        |      RAM
  +-------+        H H H H      +-------+   |   +----------+
                   H H H H      +-------+   |
                   H H H H=====>|       |<--+     +----------+
                   H H H=======>| MUX/  |========>|A  STATIC |   4Kx16
                   H H======12=>| LATCH |<==>H<==>|D _  RAM  | PALETTE
                   H      +5V-->| /GATE |----H--->|r/w       |     RAM
   A = ADDRESS     H            +-------+    H    +----------+
   C = CONTROL     H                         H     +---------+
   D = DATA        H                         H=16=>| HEATHER |   COMPOSITE
  MC = MEM_CTRL    H===========================12=>|         |--> VIDEO TO
       (r/w, cas, ras, & ale)     extvideo-------->|         |   MODULATOR
                       crystal--------->|        |==> SYSTEM
                                                   +---------+ CLOCKS
```

SPRITE POSITIONING SPACE - VIRTUAL MEMORY:

```
                       ......................................
                   . '                                       `.
              .  '                    511                      `.
         . '                           :                         `.
    . '                                :                           `.
.'. . . . . . . . . . . . . . . . . . .:. . . . . . . . . . . . . . .`.
.                      :               |                    :         .
.                      :               |                    :         .
.                      :               |                    :         .
.                      :               |                    :         .
.                      :               |                    :         .
.                      :              -Y                    :         .
.                      :               |  255_____________  :         .
.                      :               |  /               \ :         .
.                      :               | Z                 \:         .
.                      :               |/                   \         .
.      -1023 -------------- -X --------*-------- X --------+-+-- 1023 .
.                      :              /|                   | |        .
.                      :             / |                 763 920      .
.                      :           -Z  |      visible      | |        .
.                      :           /   |                   | |        .
.                      :          /    Y       space       | |        .
.                      :         /     |                   | |        .
.                      :      -255     |                   | |        .
.                      :               +--243--------NTSC--+ |        .
.                      :               +--290-----------PAL--+        .
.                      :...............|....................:         .
.                     '                |                    `         .
.                   .                  |                     .        .
.                  '                   |                              .
.                .                     |                      `       .
.               '                      |                       .      .
.             .                      -511                             .
.            '                                                  `     .
```

SPRITE DEFINED:

A sprite is an object which is generated, and positioned on the
television screen without requiring the cpu to handle its data.
It can consist of mixed transparent/non-transparent pixels.
There is no restriction on pixel transparency.

```
|     height    |    |           | x,y,z +---- width -----+
|     width     |    |           |      \ |               |
|     x,y,z     |    |           |       \V               V
|               |    |           |        +----------------+<---+
| ///////////// |    |+----------+|       | ////////////// |    |
| /sprite/data/ |======>>| percept |======>>| ////sprite//// |  height
| ///////////// |    |+----------+|       | ////////////// |    |
|               |    |           |        +----------------+<---+
 graphics memory      PENNY chips         television screen
```

PERCEPT DEFINED:

A percept is a hardware sprite generator which can be used to
generate one or more sprite incarnations. There are 32 percepts
per PENNY numbered from 0 to 31. Each percept has an initial
link. The initial link points to the attribute list for the
first sprite incarnation. Each sprite thereafter contains a link
to the next sprite incarnation for that percept in link list
fashion. Sprites cannot be reincarated on the same line and the
link list must proceed in the order of television scanning (top
to bottom) without overlap between sprites from the same percept.

PENNY CHIP PHYSICAL ORGANIZATION:

```
+------------------------------------------------+
| +--------------------------------------------+ |
| |                 INTERFACE                  | |
| +--------------------------------------------+ |
| +--+ +--+ +--+ +--+ +----+ +--+ +--+ +--+ +--+ |
| |  | |  | |  | |  | |    | |  | |  | |  | |  | |
| +--+ P--+ +--+ +--+ | C  | +--+ P--+ +--+ +--+ |
| +--+ +E-+ +--+ +--+ | O  | +--+ +E-+ +--+ +--+ |
| |  | | R| |  | |  | | N  | |  | | R| |  | |  | |
| +--+ +--C +--+ +--+ | T  | +--+ +--C +--+ +--+ |
| +--+ +--+E+--+ +--+ | R  | +--+ +--+E+--+ +--+ |
| |  | |  | P  | |  | | O  | |  | |  | P  | |  | |
| +--+ +--+ +T-+ +--+ | L  | +--+ +--+ +T-+ +--+ |
| +--+ +--+ +-S+ +--+ |    | +--+ +--+ +-S+ +--+ |
| |  | |  | |  | |  | |    | |  | |  | |  | |  | |
| +--+ +--+ +--+ +--+ +----+ +--+ +--+ +--+ +--+ |
| +--------------------------------------------+ |
| |                 INTERFACE                  | |
| +--------------------------------------------+ |
+------------------------------------------------+
     GRAPHICS MEMORY MAP FOR ONE PERCEPT SHOWING FIRST SPRITE:

   <<<<<<<<< PERCEPT PARAMETERS page 12 >>>>>>>>>
     <<          F   D                         0 >>
     <<         +-+-+---------------------------+>>
     << PERC:   |S|E|          LINK             |>>----+ Skip next, percept
     <<    +-+-+---------------------------+>>    |  Enable & LINKage to
     <<<<<<<<<<<<<<<<<<<<<<<>>>>>>>>>>>>>>>>>>>>>>>    |  first sprite.
+------------------------------------------------------+
|    <<<<<<<<< SPRITE PARAMETERS page 14 >>>>>>>>>>
|    <<          F   D         8              0 >>
|    <<         +-------------+-----------------+>>
+--> << LINK:   |   FORMAT    |      YOFF       |>>      FORMAT & Y-OFFset
     <<         +-+-+---------+-----------------+>>       Color priority,
     <<         |C|M|           DISP            |>>----+  Multi-data &
     <<         +-+-+-----------+---------------+>>    |  DISPlay adrs
     <<         |      HGT      |     ZOFF      |>>    | HeiGhT & Z-OFFset
     <<         +-+-+-----------+---------------+>>    |
     <<         |I|R|           DATA            |>>---+| Invert, Reflect
     <<         +-+-+---------+-----------------+>>   ||  & DATA adrs
     <<         |     MAP     |     XOFF/2      |>>   || MAP mode & X-OFFset
     <<         +-+-+---------+-----------------+>>   ||
     <<         |S|E|           LINK            |>>   || Skip next, percept
     <<         +-+-+---------------------------+>>   ||  Enable & LINKage to
     <<<<<<<<<<<<<<<<<<<<<<<>>>>>>>>>>>>>>>>>>>>>>>   ||  next sprite, etc.
+-----------------------------------------------------+|
|    <<<<<<<<<<<< SPRITE DATA page 14 >>>>>>>>>>>>>    |
|    <<                                          >>    |
|    <<         +-------------------------------+>>    | always two words
+--> << DATA:   |                               |>>    | per scan line in
     <<         +--------- sprite data ---------+>>    | Multi-data
     <<         |                               |>>    | displayed per
     <<         +-------------------------------+>>    | FORMAT mode
     <<         |             etc.              |>>    |
     <<                                          >>    |
     <<<<<<<<<<<<<<<<<<<<<<<>>>>>>>>>>>>>>>>>>>>>>>    |
+-------------------------------------------------------+
|    <<<<<<<<< DISPLAY PARAMETERS page 26 >>>>>>>>>
|    <<          F          9                  0 >>
|    <<         +-----------+--------------------+>>
+--> << DISP:   |  STENCIL  |       YPOS         |>>      STENCIL mode
     <<         +-----------+-+------------------+>>       & Y-POSition
     <<         |  LAMINATE |%|       ZPOS       |>>      LAMINATE mode
     <<         +-----------+-+------------------+>>       & Z-POSition
     <<         |% % % % % %|      XPOS/2        |>>      X-POSition
     <<         +-----------+--------------------+>>
     <<<<<<<<<<<<<<<<<<<<<<<>>>>>>>>>>>>>>>>>>>>>>>
```

PERCEPT PARAMETERS:

S & E (Skip next switch & percept Enable) DEFINED:

```
            F E
           +-+-+----------------------------+
   PERC:   |S|E|                            |
           +-+-+----------------------------+
```

The skip next switch causes the next sprite to be skipped and resumes processing with the succeeding sprite which is linked from the skipped sprite as though the skipped sprite were processed normally.

The percept enable allows the next link to be processed. If not enabled, the percept is terminated and no further sprites will be generated by it until the next screen.

SPRITE PARAMETERS:

```
           F   D         8                0
           +-+-+---------+-----------------+
  LINK:    |   FORMAT    |      YOFF       |     FORMAT & Y-OFFset
           | page 17-20  |    page 23      |
           +-+-+---------+-----------------+      Color priority sw,
           |C|M|           DISP            |      Multi-data switch
           |pg 14                          |      & DISPlay adrs
           +-+-+-----------+---------------+
           |      HGT      |     ZOFF      |     HeiGhT & Z-OFFset
           |    page 23    |   page 23     |
           +-+-+-----------+---------------+
           |I|R|           DATA            |     Invert, Reflect
           |pg 24-25                       |      & DATA adrs
           +-+-+---------+-----------------+
           |     MAP     |     XOFF/2      |     MAP mode & X-OFFset
           |   page 16   |    page 16      |
           +-+-+---------+-----------------+
           |S|E|           LINK            |     Skip next, percept
           |pg 13                          |      Enable & LINKage to
           +-+-+---------------------------+      next sprite, etc.

           +-------------------------------+
  DATA:    |                               |     always two words
           |                               |     per scan line in
           +--------- sprite data ---------+     Multi-data
           |                               |     displayed per
           |    see FORMAT pages 17-20     |     FORMAT mode
           +-------------------------------+
           |             etc.              |
```

M (Multi-data switch) DEFINED:

```
             F
            +-+------------------------------+
  LINK:     |M|                           |
            +-+------------------------------+
```

The multi-data switch identifies this sprite as a multiple line sprite. As a multi-data sprite there must be two words of data for the sprite for each of HGT lines to be displayed. As a single data sprite (M=0) only one line of data (two words) is evaluated and that line is repeated HGT number of times down the screen.

FUNCTIONAL PRIORITIZATION:

There are 8192 levels of functional prioritization in each PENNY. The functional priority is divided between software controllable and hard-ware fixed priority fields. Soft priority has field significence over hard priority so that the programmer always has control of functional prioritization of sprites. Only when two sprites are coincident in space with the same soft priority does hard priority determine which sprite has functional priority. When portraying a three-dimensional space, soft priority is merely the z-coordinate of the sprite and so the terms, 'soft priority', 'z-coordinate' and 'z-level' can be used interchangeably.

SOFT PRIORITY (z-level)	HARD PRIORITY (percept number)	
00000000	00000	Highest PENNY functional priority
00000000	00001	
:	:	
11111111	11111	Lowest PENNY functional priority

```
                 .----------------------.
                .  |           PERCEPT 31 |
               .      |        lowest | HARD PRIORITY
             .        |      .     |    ---------------
          .--------------------------.       |     Within a soft
         .---------------------------. |       |   priority level,
      .----------------------------. | |       |   hard priority is
      |                  PERCEPT 0 | | |       |   determined by
      |                    highest | | |       |   percept number
      |                            | | |       |   (32 levels).
      |                            | | |       |
      |    pixel(x,y,z,percept)    | | |       |
      |  /                         | | |_______|
      | o                          | | |::::::::
      | |                          | | |::::::'
```

```
| |                          | | |::::'
| |                          | |_|:::
| |                          |_|::' :
|_|__________________________|::'---:-----------.
::|:::::::::::::::::::::::::::'     :   Z = 255 |
|    :           .     |          :    lowest |    SOFT PRIORITY
|    :        .        |      :   .     |    --------------
|    :     .----------------------:---.       |    Soft priority is
|    :  .-------------------------:-. |       |    determined by
|    :----------------------------: | |       |    z-coordinate
|    |                      Z = 0 | | |       | (256 levels).
|    |                    highest | | |       |
|    |                            | | |_______|
|    |                            | | |::::::::
|    |                            | | |::::::'
|    |                            | | |:::::'
|    |                            | | |::::
|    |                            | | |::'
|    |                            | |_|:'
|    |                            | |::
|    |                            |.|'
|    |____________________________|:'
|    ::::::::::::::::::::::::::::::
  |                               PALETTE PRIORITY
  |  color(x,y) PENNY 0  .----------. chrom(x,y)   ----------------
  +---------------------->|         +-----------+   System priority
  |  color(x,y) PENNY 1  |  COLOR  | lum(x,y)  |   is determined by
  | -------------------->|         +---------+ |   color palette
  |  color(x,y) PENNY 2  | PALETTE | iselect | |   mapping.
  | -------------------->|         +-------+ | |
  |               `---------'       | | |
|                    +------------+ | |
|                    |         | |
|  inten(x,y) PENNY 0  .----V----.         | |
  +---------------------->|         |       | |
   inten(x,y) PENNY 1  |  INTEN  |.--V-V--.
 ---------------------->|     +----->|   |
   inten(x,y) PENNY 2  |   MUX     |    | VIDEO     |
  -------------------->|      |      |   +----> VIDEO
                `---------'      |  GEN    |
EXTERNAL VIDEO SIGNAL ----------------->|     |
                          `-------'
```

SPRITE PARAMETERS (continued):

MAP (MAP mode) DEFINED:

```
             F           9
            +-------------+-----------------+
LINK+1:     |     MAP     |                 |
            +-------------+-----------------+
```

There are 128 map functions for each view as follows:

XOFF (X-OFFset) DEFINED:

```
                           8               0
            +-------------+-----------------+
LINK+1:     |             |     XOFF/2      |
            +-------------+-----------------+
```

XOFF is the unsigned binary displacement added to XPOS to determine
the x-coordinate of the sprite. See XPOS for more details (page 26).

SPRITE PARAMETERS (continued):

FORMAT (FORMAT mode) DEFINED:

```
             F           9
            +-------------+-----------------+
LINK+2:     |   FORMAT    |                 |
            +-------------+-----------------+
```

The FORMAT mode determines the way in which sprite data is interpreted and displayed on the screen.

```
SPRITE SELECTION GUIDE

CAN BE SELF-STENCILING
| CAN BE SELF-LAMINATING
| | AUTOMATIC EDGE SMOOTHING
| | | GRAPHICS DETERMINED BY: Data# (# = NUMBER OF BITS PER ELEMENT)
| | | |  COLOR DETERMINED BY: Format_code#, Graphics#, Zero#
| | | |  |  INTENSITY DETERMINED BY: Format_code#, Graphics#, Zero#
| | | |  |  |  GRAPHICS ELEMENT TYPE: Fixed-length, Run-length, Area
| | | |  |  |  |      ELEMENT SIZE IN PIXELS
| | | |  |  |  |      |      NUMBER OF ELEMENTS PER SPRITE
| | | |  |  |  |      |      |   MAX SPRITE SIZE IN PIXELS
| | | |  |  |  |      |      |   | + indicates offset(s), also
| | | |  |  |  |      |      |   |      FORMAT CODE
| | | |  |  |  |      |      |   |      |
V V V V  V  V  V      V      V   V      V     NAME & PAGE
- - - -- -- -- -  --------  --  ---  -------  ----------------------

Color only

X X - D4 G4 -- F      2      8   16  0000010  DAZZLER
X X - D1 F4 -- F    3(av)   32   96  100cccc  AIR BRUSH
X X X D8 G4 -- R   2 to 32   4  128  0000100  SMALL CARTOON
X X X D8 G4 -- R  8 to 128   4  512  0000110  LARGE CARTOON

Intensity only

  - - - D0 -- -- F      5      2   10+ 0001000  EDGE SMOOTHING
- - - D4 -- G4 F      1      8   8   0001010  DETAIL
- - - D4 -- G4 F      1      8   8   0001011  DETAIL REVERSE
- - - D8 -- G4 R   2 to 32   4  128  0001100  SMALL SHADE
  - - - D8 -- G4 R   2 to 32   4  128  0001101  SMALL REVERSE
- - - D8 -- G4 R  8 to 128   4  512  0001110  LARGE SHADE
  - - - D8 -- G4 R  8 to 128   4  512  0001111  LARGE REVERSE
- - - D1 -- F4 F      3     32   96  110iiii  TEXTURE
```

```
Mixed color & intensity

X X X D4 G4 F4 F      2      8   16  001iiii  SMALL EVENT
X X X D8 G4 F4 R   2 to 32   4  128  010iiii  MEDIUM EVENT
X X X D8 G4 F4 R  8 to 128   4  512  011iiii  LARGE EVENT

Non-priority Special Effects

X X X D7 -- F4 A  4 to 512   2 1024+ 111iiii  DUAL DISSOLVE

Priority Special Effects

X X X D1 -- -- F      6     32  192  0000011  SCREEN
X X X D8 -- -- R   2 to 32   4  128  0000101  MEDIUM GHOST
X X X D8 -- -- R  8 to 128   4  512  0000111  LARGE GHOST
X X X D8 -- -- A  4 to 512   2 1024+ 0000001  DUAL GHOST VIEWPORT


- - - -- -- -- -  --------  --  ---  101____  (future use)
- - - -- -- -- -  --------  --  ---  0000000  (future use)
  - - - -- -- -- -  --------  --  ---  0001001  (future use)
```

SPRITE FAMILY TREE

```
     DAZZLER   AIR BRUSH  SMALL CARTOON  LARGE CARTOON  SURFACE  Color only
         |          |           |            |            |
         |          |           |            |            |
       DETAIL    TEXTURE   SMALL SHADE  LARGE SHADE       |       Intensity
  DETAIL REVERSE    |    SMALL REVERSE  LARGE REVERSE     |           only
         |          |           |            |            |
         |          |           |            |            |
  SMALL EVENT     |     MEDIUM EVENT  LARGE EVENT       |   Mixed color &
         |          |           |            |            |       intensity
         |          |           |            |            |
       |          |           |            |        DISSOLVE     Priority
       |          |           |            |            |         special
       |          |           |            |            |
       |       SCREEN    SMALL GHOST  LARGE GHOST   VIEWPORT Non-priority
       |          |           |            |            |         special
         |          |           |            |            |
         |          |           |            |            |
     HIGH      MEDIUM      MEDIUM        LOW            |
    RESOLUTION  RESOLUTION  RESOLUTION   RESOLUTION      |
        \________/             \__________/             |
             |                      |                   |
          FIXED                    RUN               AREA
         LENGTH                  LENGTH            SPRITES
         SPRITES                 SPRITES            /
             \______________________|________________/
                                    |
```

DAZZLER SPRITE

Optimized for high resolution poly-chromatic detail, this sprite
has 16 pixels of color graphics with color determined by the
graphics data and with full intensity (no edge smoothing).

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  | COLOR | COLOR | COLOR | COLOR | COLOR | COLOR | COLOR | COLOR |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left --------------- 2 pixels per nibble ---------------> right
```

```
FORMAT  COLOR  color & intensity  comment
------- -----  -----   --------- ---------------------------
0000010  0000  0000      0000    pixels transparent, full intensity
   "   else        "    pixels displayed
   "     0001  0001      "      color-1, full intensity
   "     0010  0010      "      color-2, full intensity
   "    :   :       "        :
   "     1111  1111      "      color-15, full intensity
```

Results of storing a line of 01234567(hex) followed by a line
  of 76543210(hex)

```
datum pt
       \
      *-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |   |/1/|/2/|/3/|/4/|/5/|/6/|/7/| number = color number,
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ otherwise, transparent
      |/7/|/6/|/5/|/4/|/3/|/2/|/1/|   |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ---->|   |<---- 2 pixels
```

AIR BRUSH SPRITE

Optimized for high resolution mono-chromatic detail, this sprite has 96 pixels of color graphics with color determined by either the FORMAT parameter directly (as a color register) or by priority (as determined by the compliment of the sum of Z-POS plus Z-OFF) and with full intensity (no edge smoothing).

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F E D C B A 9 8 7 6 5 4 3 2 1 0 F E D C B A 9 8 7 6 5 4 3 2 1 0
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|C|
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   left ---------- 2 pixels per  &  4 pixels per ----------> right
         even numbered bit     odd numbered bit

FORMAT  Z[7,4]  C  color & intensity    comment
------- ------ --- -----   --------- ---------------------------
100%%%%  %%%%   0  0000      0000    pixels transparent
              1        "         pixels displayed
1000000          "          "           priority determines color
   "       "           "           (15 colors plus transparent)
   "  0000   "  0000        "            transparent
   "  0001   "  0001        "            color-1, full intensity
   "    :  "   :       "                   :
   "  1111   "  1111        "            color-15, full intensity
100else          "          "           FORMAT determines color
           "           "           (15 colors)
1000001  %%%%   "  0001           "           color-1, full intensity
1000010   "     "  0010           "           color-2, full intensity
    :  "  "   :       "                   :
1001111   "     "  1111           "           color-15, full intensity
```

Result of storing a line of 010101... followed by a line of 101010...

```
       +------ data word (1 or 2)
```

```
                 | +---- data bit (F through 0)          /// = colored,
                 | |                          otherwise, transparent
                 V V
          datum pt 1-F   1-E   1-D   1-C   1-B   1-A    ...  2-1   2-0
                 \  |    / \    |    / \    |    / \          |    / \
                *-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
                |   |///////|   |///////|   |///////| ... |   |///////|
                +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
                |///|       |///|       |///|       | ... |///|       |
                +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
             ---->|   |<---- 2 pixels

                Note the similarities and differences between this sprite
                and its kin, the TEXTURE & SCREEN sprites.
```

SMALL CARTOON SPRITE

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/2-1*:   C1  |L2/2-1*:   C2  |L3/2-1*:   C3  |L4/2-1*:   C4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right
```

FORMAT	Cn	color & intensity		comment
0000100	0000	0000	0000	pixels transparent, full intensity
"	else	"	pixels displayed	
"	0001	0001	"	color-1, full intensity
"	0010	0010	"	color-2, full intensity
"	:	:	"	:
"	1111	1111	"	color-15, full intensity

```
  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////.../////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+    -------   -----
|<----- L1 ------>|<----- L2 ------>|        2        0
  2 to 32 pixels    2 to 32 pixels      4        1
  all color C1      all color C2        6           2
                                   etc.
```

This sprite automatically generates left & right edge smoothing for each element, see EDGE SMOOTHING SPRITE (page) for details.

LARGE CARTOON SPRITE

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
```

```
    F       B       7       3     0 F       B       7       3     0
   +-------+-------+-------+-------+-------+-------+-------+-------+
   |L1/8-1*:   C1  |L2/8-1*:   C2  |L3/8-1*:   C3  |L4/8-1*:   C4  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
    left ---------------- L pixels per byte ----------------> right

FORMAT  COLOR  color & intensity  comment
------- -----  -----   --------- ---------------------------
0000110  0000  0000      0000    pixels transparent, full intensity
   "   else        "   pixels displayed
   "     0001  0001      "     color-1, full intensity
   "     0010  0010      "     color-2, full intensity
   "    :   :        "        :
   "     1111  1111      "     color-15, full intensity

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////.../////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         8        0
  8 to 128 pixels   8 to 128 pixels          16        1
  all color C1       all color C2      24           2
                                    etc.

This sprite automatically generates left & right edge smoothing
for each element, see EDGE SMOOTHING SPRITE (page ) for details.
```

SURFACE SPRITE

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F E       A 9               1 0 F             8               0
  +-+-------+-+---------------+---+-------------+-----------------+
  |S: XSLO  |S:    YSLO*      |   OFFSET/2-1*   :   WIDTH/2-1*    |
  +-+-------+-+---------------+-----------------+-----------------+
EDGE SMOOTHING SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F                 6           0 F   D                 4       0
  +-----------------+-------------+---+-----------------+----------+
  |  LEFTOFFSET/2*  |     WIDTH/2*    |  RIGHTOFFSET/2* ||
  +-----------------+-----------------+-----------------+----------+

Typical two lines of edge smoothing:

                 Left edge          Right edge
                (laminated by      (laminated by
               STENCIL sprite)      LAMINATE sprite)
datum pt          ________          __________
 /               /          \      /        \
*                  --+---          ---+--
<------- LEFTOFFSET ------->|<------ WIDTH ------>|<- RIGHTOFFSET ->
                 --+-==+---          ---+-----+--
<----- LEFTOFFSET ----->|<----- WIDTH ----->|<---- RIGHTOFFSET ---->
                 --+---              ---+--
                  /              \
Detail of edges        /                      \
             +-+-+-+-+-+          +-+-+-+-+-+
             | | | | | |          | | | | | |
             +-+-+-+-+-+          +-+-+-+-+-+
     LEFTOFFSET ----->|<------- WIDTH ------->|<----- RIGHTOFFSET
  2 to 1024 pixels    |        2 to 1024 pixels   |    2 to 1024 pixels
                 |                      |
          0 ______  |    _______________    |  ______
  Intensity 4 ______\ |   /_______________\   | /______
          8 ______ \|  / _______________ \  |/ ______
    Level   C ______\ \_/ /_______________\ \_/ /______
          F _______\_X_/_________________\_X_/_______

      Intensity is interpolated   RIGHTOFFSET is only needed if
```

Document Source: atarimuseum.com

for each pixel from the int-
ensities of the sprites on
either side of the edge. If
either stenciling sprite does
not have priority up to its
edge, then the edge will not
be visible & will not inter-
fere with the priority sprite
(which is generating its own
edges).

sprite is to be reflectable in
which case LEFTOFFSET plus
WIDTH plus RIGHTOFFSET must be
a constant for all lines.

DETAIL SPRITE

Optimized for high resolution multi-intensity detail, this sprite has 8 pixels with intensity determined by the graphics data. It is very useful for creating font characters for up to 95 columns of text (using 95 percepts). It must be laminated to a color sprite to be apparent.

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left --------------- 1 pixel per nibble ----------------> right
```

```
datum pt
      \
     *-+-+-+-+-+-+-+-+INTEN = 0000 ==> full intensity
     | | | | | | | | |      0001 ==> 15/16 intensity
     +-+-+-+-+-+-+-+-+       :      :
     | | | | | | | | |      1111 ==> 1/16 intensity
     +-+-+-+-+-+-+-+-+
   ---->| |<---- 1 pixel
```

DETAIL REVERSE SPRITE

This sprite is functionally identical to the DETAIL sprite except that the values of INTEN stored are internally complimented before be written on the screen resulting in an inverse video (but not complimentary color) display. It is very useful for creation of a cursor. Any font character can be transformed into that same character displayed over the cursor with a single bit set in its FORMAT field (ie, change 0001010 to 0001011).

```
  <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
  F       B       7       3     0 F       B       7       3     0
 +-------+-------+-------+-------+-------+-------+-------+-------+
 | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN | INTEN |
 +-------+-------+-------+-------+-------+-------+-------+-------+
  left --------------- 1 pixel per nibble ----------------> right

datum pt
      \
     *-+-+-+-+-+-+-+-+INTEN = 0000 ==> 1/16 intensity
     | | | | | | | | |      0001 ==> 2/16 intensity
     +-+-+-+-+-+-+-+-+       :      :
     | | | | | | | | |      1111 ==> full intensity
     +-+-+-+-+-+-+-+-+
  ---->| |<---- 1 pixel
```

Document Source: atarimuseum.com

```
SMALL SHADE SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/2-1*:   I1  |L2/2-1*:   I2  |L3/2-1*:   I3  |L4/2-1*:   I4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         2        0
  2 to 32 pixels    2 to 32 pixels      4        1
  all intensity I1  all intensity I2          6           2
                                  etc.

           In = 0000 ==> full intensity
                0001 ==> 15/16 intensity
                 :
                1111 ==> 1/16 intensity


SMALL REVERSE SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/2-1*:   I1  |L2/2-1*:   I2  |L3/2-1*:   I3  |L4/2-1*:   I4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         2        0
  2 to 32 pixels    2 to 32 pixels      4        1
  all intensity I1  all intensity I2          6           2
                                   etc.

            In = 0000 ==> 1/16 intensity
                 0001 ==> 2/16 intensity
                  :
                 1111 ==> full intensity
```

```
LARGE SHADE SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/8-1*:   I1  |L2/8-1*:   I2  |L3/8-1*:   I3  |L4/8-1*:   I4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         8        0
  8 to 128 pixels   8 to 128 pixels          16        1
  all intensity I1  all intensity I2         24           2
                                   etc.

           In = 0000 ==> full intensity
                0001 ==> 15/16 intensity
                  :
                1111 ==> 1/16 intensity



LARGE REVERSE SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/8-1*:   I1  |L2/8-1*:   I2  |L3/8-1*:   I3  |L4/8-1*:   I4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right
  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|////////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         8        0
  8 to 128 pixels   8 to 128 pixels          16        1
  all intensity I1  all intensity I2         24            2
                                    etc.

           In = 0000 ==> 1/16 intensity
                0001 ==> 2/16 intensity
                 :
                1111 ==> full intensity
```

TEXTURE SPRITE

Optimized for high resolution mono-intensity detail, this sprite has 96 pixels of intensity graphics with intensity determined by either the FORMAT parameter directly (as an intensity register) or by priority (as determined by the compliment of the sum of Z-POS plus Z-OFF) and with transparent color. It must be laminated to a color sprite to be apparent.

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F E D C B A 9 8 7 6 5 4 3 2 1 0 F E D C B A 9 8 7 6 5 4 3 2 1 0
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|I|
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   left ---------------- 3 pixels per bit -----------------> right
```

```
FORMAT  Z[7,4]  I  color & intensity    comment
------- ------ --- -----   --------- ----------------------------
110%%%%  %%%%   0   none     0000    unmodified (full) intensity
              1    "             pixel intensity modified
1100000          "    "               priority determines intensity
    "      "    "                (16 intensities)
    "  0000   "    "        0000     full intensity
    "  0001   "    "        0001     15/16 intensity
    "   :  "    "            :               :
    "  1111   "    "        1111     1/16 intensity
110else          "    "               FORMAT determines intensity
            "    "                (15 intensities)
1100001     %%%%   "    "        0001     15/16 intensity
1100010      "     "    "        0010     14/16 intensity
    :   "   "    "            :               :
1101111      "     "    "        1111     1/16 intensity
```

Result of storing a line of 010101... followed by a line of 101010...

```
        +------ data word (1 or 2)
        | +---- data bit (F through 0)  /// = intensity modified
        | |                       otherwise, full inten.
        V V
datum pt  1-F   1-E   1-D   1-C   1-B   1-A   ...   2-1   2-0
       \   |     |     |     |     |     |           |     |
      *-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
      |     |/////|     |/////|     |/////| ... |     |/////|
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
      |/////|     |/////|     |/////|     | ... |/////|     |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
   ---->|     |<---- 3 pixels

      Note the similarities and differences between this sprite
      and its kin, the AIR BRUSH & SCREEN sprites.
```

Document Source: atarimuseum.com

```
SMALL EVENT SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |   C   |   C   |   C   |   C   |   C   |   C   |   C   |   C   |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left --------------- 2 pixels per nibble ---------------> right

FORMAT  Z[7,4]   C  color & intensity   comment
------- ------ ---- -----   --------- ----------------------------
001%%%%  %%%%  0000  0000     0000    pixels transparent, full intensity
             0001  0001      pixels displayed in color-1
0010000          "     "                priority determines intensity
    "      "     "               (16 intensities)
    "  0000 "     "       0000        full intensity
    "  0001 "     "       0001        15/16 intensity
    "   :  "     "        :       :
    "  1110 "     "       1111        1/16 intensity
001else          "     "                FORMAT determines intensity
           "     "               (15 intensities)
0010001     %%%% "     "       0001        15/16 intensity
0010010      "  "     "       0010        14/16 intensity
    :  "   "     "        :       :
0011111   ""     "       1111        1/16 intensity
             0010  0010      pixels displayed in color-2
           :     :                  intensity determined as above
           :     :                    :
             1111  1111      pixels displayed in color-15
                              intensity determined as above

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |   |   |   |   |   |   |   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |   |   |   |   |   |   |   |
```

Document Source: atarimuseum.com

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
MEDIUM EVENT SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F       B       7       3     0 F       B       7       3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/2-1*:   C1  |L2/2-1*:   C2  |L3/2-1*:   C3  |L4/2-1*:   C4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right

FORMAT  Z[7,4]  Cn  color & intensity   comment
------- ------ ---- -----   --------- ---------------------------
010%%%%  %%%%  0000  0000     0000    pixels transparent, full intensity
             0001  0001      pixels displayed in color-1
0100000          "     "                priority determines intensity
    "       "     "               (16 intensities)
    "  0000 "     "          0000    full intensity
    "  0001 "     "          0001    15/16 intensity
    "   :   "     "           :        :
    "  1111 "     "          1111    1/16 intensity
010else          "     "                FORMAT determines intensity
            "     "               (15 intensities)
0100001      %%%% "     "          0001    15/16 intensity
0100010       "   "     "          0010    14/16 intensity
    :   "   "     "           :        :
0101111   ""     "          1111    1/16 intensity
             0010  0010      pixels displayed in color-2
            :     :                intensity determined as above
            :     :                     :
             1111  1111      pixels displayed in color-15
                             intensity determined as above

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|////////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
```

```
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         2        0
  2 to 32 pixels    2 to 32 pixels      4        1
  all color C1      all color C2        6           2
                                   etc.
```

This sprite automatically generates left & right edge smoothing
for each element, see EDGE SMOOTHING SPRITE (page) for details.

```
LARGE EVENT SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F     C B     8 7     4 3     0 F     C B     8 7     4 3     0
  +-------+-------+-------+-------+-------+-------+-------+-------+
  |L1/8-1*:   C1  |L2/8-1*:   C2  |L3/8-1*:   C3  |L4/8-1*:   C4  |
  +-------+-------+-------+-------+-------+-------+-------+-------+
   left ---------------- L pixels per byte ----------------> right

FORMAT  Z[7,4]  Cn  color & intensity   comment
------- ------ ---- -----   --------- ----------------------------
011%%%%  %%%%  0000  0000     0000    pixels transparent, full intensity
             0001  0001      pixels displayed in color-1
0110000          "     "               priority determines intensity
    "      "     "               (16 intensities)
    "  0000 "     "          0000    full intensity
    "  0001 "     "          0001    15/16 intensity
    "   :   "     "           :        :
    "  1111 "     "          1111    1/16 intensity
011else          "     "               FORMAT determines intensity
            "     "               (15 intensities)
0110001     %%%% "     "           0001    15/16 intensity
0110010      "   "     "           0010    14/16 intensity
    :   "   "     "           :        :
0111111   ""     "          1111    1/16 intensity
             0010  0010      pixels displayed in color-2
            :     :                intensity determined as above
            :     :                     :
             1111  1111      pixels displayed in color-15
                             intensity determined as above

  datum pt  Two typical elements out of four total elements.
 /
*-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+
|///////...//////|\\\\\\\\...\\\\\\| ...  For L =   Store
```

```
+-+-+-+-+   +-+-+-+-+-+-+-+   +-+-+-+      -------   -----
|<----- L1 ------>|<----- L2 ------>|         8        0
  8 to 128 pixels   8 to 128 pixels          16        1
  all color C1       all color C2      24           2
                                   etc.
```

This sprite automatically generates left & right edge smoothing
for each element, see EDGE SMOOTHING SPRITE (page) for details.

```
DISSOLVE SPRITE

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F           9             2   0 F     C             5         0
  +-----------+-------------+-----+-----+-------------+-----------+
  | LEFTOFF*  |  LEFTAREA*  | CENTOFF*  | RIGHTAREA*  | RIGHTOFF* |
  +-----------+-------------+-----------+-------------+-----------+
 left --------------



SCREEN SPRITE

This sprite does not have any color or intensity associated with
it's graphics.  It functions as a programmable switch to make the
sprites 'behind' it within the same PENNY, invisible.  Sprites
'in front' of it or from other PENNYs are not affected.

   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F E D C B A 9 8 7 6 5 4 3 2 1 0 F E D C B A 9 8 7 6 5 4 3 2 1 0
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|S|
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   left ---------------- 6 pixels per bit -----------------> right

      S = 0 ==> screen off, pixels generated by this PENNY behave
              normally.
          1 ==> screen on, all pixels generated by this PENNY
              which have a lower priority dissappear.

Result of storing a line of 010101... followed by a line of 101010...

           +------ data word (1 or 2)         /// = screen on
           | +---- data bit (F through 0)     else, screen off
          | |
          V V
datum pt     1-F         1-E         1-D      ...      2-0
       \      |           |           |                 |
      *-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
      |           |///////////|           | ... |///////////|
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
      |///////////|           |///////////| ... |           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+

      Note the similarities and differences between this sprite
      and its kin, the AIR BRUSH & TEXTURE sprites.
```

DUAL GHOST VIEWPORT SPRITE

```
   <----- Sprite data word 1 ----> <---- Sprite data word 2 ----->
   F           9             2   0 F     C             5         0
  +-----------+------------+-----+-----+------------+-----------+
  |    LEFTOFF*    |    LEFTAREA*    |   CENTOFF*  |   RGTAREA*  |
  +-----------+------------+-----------+------------+-----------+
```

USING THE AUTOMATIC COLOR GENERATION OF COLOR SPRITES:

```
       COLOR __          range of z __
               \                      \
._________________________________
|\             1111              \   224,255
| \_______________________________\
| |\             1110              \   192,223
| | \_______________________________\
| | |\             1101              \   160,191
| | | \_______________________________\
| | | |\             1100              \   128,159
| | | | \_______________________________\
| | | | |\             1011              \    96,127
| | | | | \_______________________________\
| | | | | |\             1010              \    64,95
| | | | | | \_______________________________\
| | | | | | |\             1001              \    32,63
| | | | | | | \_______________________________\
\ | | | | | | |\             1000              \     0,31
 \| | | | | | | \_______________________________\
  \ | | | | | | |                               |
   \| | | | | | |                               |
    \ | | | | | |                               |
     \| | | | | |                               |
      \ | | | | |                               |
       \| | | | |                               |
        \ | | | |       television screen       |
         \| | | |                               |
          \ | | |                               |
           \| | |                               |
            \ | |                               |
             \| |                               |
              \ |                               |
               \|_______________________________|
```

Color is generated automatically, as a function of the on-the-fly z-coordinate, but will affect ONLY this sprite's color and NOT the color of laminated sprites. The color is processed by the palette RAM in the normal way so that the final result is dependent upon the colors selected by the programmer for COLOR = '1xxx'. Note that this scheme gives 8 color-levels with each color-level encompassing 32 depth-levels in z.

USING THE AUTOMATIC INTENSITY ADJUST OF INTENSITY SPRITES:

```
     value            resulting     range
    of INTEN __        intensity __  of z __
               \                   \        \
.________________________________
|\_____________1111______________\  1/16   240,255
||\_____________1110______________\  2/16   224,239
|||\_____________1101______________\  3/16   208,223
||||\_____________1100______________\  4/16   192,207
|||||\_____________1011______________\  5/16   176,191
||||||\_____________1010______________\  6/16   160,175
|||||||\_____________1001______________\  7/16   144,159
||||||||\_____________1000______________\  8/16   128,143
|||||||||\_____________0111______________\  9/16   112,127
||||||||||\_____________0110______________\  10/16   96,111
|||||||||||\_____________0101______________\  11/16  80,95
||||||||||||\_____________0100______________\  12/16  64,79
|||||||||||||\_____________0011______________\  13/16  48,63
||||||||||||||\_____________0010______________\  14/16  32,47
\||||||||||||||\_____________0001______________\  15/16  16,31
 \||||||||||||||\_____________0000______________\  16/16    0,15
  \||||||||||||||                               |  (full)
   \|||||||||||||                               |
    \||||||||||||                               |
     \|||||||||||                               |
      \||||||||||                               |
       \|||||||||                               |
        \||||||||       television screen       |
         \|||||||                               |
          \||||||                               |
           \|||||                               |
            \||||                               |
             \|||                               |
              \||                               |
               \|______________________________|
```

INTENsity factor is generated automatically, as a function of the on-the-fly z-coordinate, for use in adjusting the intensity of ALL sprites which use this sprite for laminating. Note that this scheme gives 16 intensity-levels with each intensity-level encompassing 16 depth-levels in z.

YOFF (Y-OFFset) DEFINED:

```
                     8               0
           +-------------+-----------------+
LINK+2:    |             |      YOFF       |
           +-------------+-----------------+
```

YOFF is the unsigned binary displacement added to YPOS to determine the y-coordinate of the sprite. See YPOS for more details (page 26).

HGT (HeiGhT) DEFINED:

```
            F             8
           +---------------+---------------+
LINK+3:    |      HGT      |               |
           +---------------+---------------+
```

Height is a number which defines the vertical height in scan lines for the sprite. If the Multiline switch is true then each of HGT number of lines in the sprite will be distinct as determined by each of HGT number of lines of sprite data. If the Multiline switch is false then each of HGT number of lines in the sprite will be identically the first (and only) line of sprite data. Remember that a line of sprite data is always two words long.

ZOFF (Z-OFFset) DEFINED:

```
                       7             0
           +---------------+---------------+
LINK+3:    |               |     ZOFF      |
           +---------------+---------------+
```

ZOFF is the unsigned binary displacement added to ZPOS to determine the z-coordinate of the sprite. See ZPOS for more details (page 26).

SPRITE PARAMETERS (continued):

I & R (Invert & Reflect switches) DEFINED:

```
             F E
            +-+-+----------------------------+
 LINK+4:    |I|R|                            |
            +-+-+----------------------------+
```

Invert and reflect work on sprite data and do just as the names imply. Care and thought must be exercised in determining the X,Y,Z-POSitions/OFFsets for inverted and/or reflected data especially when topiaries are involved. The following rules apply for topiaries:

I	R	operation	comment
0	0	datum view	signs of YAW, XINC & ZINC used as stored, T/FWIDL used & DATA points to the first line of sprite data.
0	1	reflected view	signs of YAW & XINC automatically reversed, T/FWIDR used, DATA points to the first line of sprite data & X,Y,ZOFF must be adjusted.
1	0	inverted view	signs of XINC & ZINC automatically reversed, T/FWIDL used, DATA points to the last line of sprite data & X,Y,ZOFF must be adjusted.
1	1	inverted and reflected view	signs of YAW & ZINC automatically reversed, T/FWIDR used, DATA points to the last line of sprite data & X,Y,ZOFF must be adjusted.

As an aid for remembering the rules above, visualize the top view of the pyramid below and understand the two alternative methods of constucting and using the basic surface which are found on the next

```
page.

        X,Y,ZPOS = X_PYRAMID,Y_PYRAMID,Z_PYRAMID_APEX
          |                               |
          v                               v
          *     o       ---               *     +       ---
               /|\       |                     /|\       |
              /-|-\      |                    //|\\      |
             /--|--\    600ft                ///|\\\     |
            /---|---\    |                  ////|\\\\    |
           /----|----\   |                 /////|\\\\\   |
ground -- +-----+-----+ ---               +-----o-----+ 600ft
level                                      \\\\\|/////   |
                                            \\\\|////    |
            FRONT VIEW                       \\\|///     |    TOP VIEW
                                              \\|//      |
                                               \|/       |
                                                +       ---
let incremental of x,y,z = 100ft          |__ 600ft __|
                                          |           |
```

SPRITE PARAMETERS (continued):

I & R (Invert & Reflect switches) DEFINED (continued):

---- TWO ALTERNATIVE REPRESENTATIONS OF THE BASIC SURFACE ----

BASIC SURFACE WITH YAW = -45, XINC = 0 & ZINC = -1						BASIC SURFACE WITH YAW = -45, XINC = -1 & ZINC = 0					
S:YAW	S:XINC	S:ZINC	TWIDL	CWID	TWIDR	S:YAW	S:XINC	S:ZINC	TWIDL	CWID	TWIDR
1 8	0 P3,0	1 P3,1	P6,6	P8,1	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,1	P6,6
1 8	0 P3,0	1 P3,1	P6,5	P8,2	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,2	P6,5
1 8	0 P3,0	1 P3,1	P6,4	P8,3	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,3	P6,4
1 8	0 P3,0	1 P3,1	P6,3	P8,4	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,4	P6,3
1 8	0 P3,0	1 P3,1	P6,2	P8,5	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,5	P6,2
1 8	0 P3,0	1 P3,1	P6,1	P8,6	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,6	P6,1
1 8	0 P3,0	1 P3,1	P6,0	P8,7	P6,0	1 8	1 P3,1	0 P3,0	P6,0	P8,7	P6,0

NOTE: Pm,n is an assembler instruction to assemble a polycode number of width 'm' and of value 'n'. For example, P8,6 assembles to 01111111 (binary) which is the eight bit polycode representation of six.

------- VIEW OF BASIC SURFACE GENERATED BY EACH METHOD --------

```
            . - - +                                     + - - -
           /:    /|                                    /|     '
      600ft :   //|                                   //|    '
      below :  ///|                                  ///|   '
       gnd  : ////|                                 ////|  '
            ://///|                                /////| '
            +-----o                               +-----o'
           /     /                               /     /
 ground level  600ft above gnd         ground level  600ft above gnd
```

--- SURFACES GENERATED FROM BASIC SURFACE BY INVERSION & REFLECTION ---

```
+-- X,Y,ZOFF +-- X,Y,ZOFF        +-- X,Y,ZOFF +-- X,Y,ZOFF
|   = 0,0,12 |   = 6,0,6         |   = 6,0,6  |   = 0,0,0
v            v                   v            v
. - - +      + - - .             + - - -       - - - +
:    /|      |\    :            /|     '       `     |\
:   //|      |\\   :           //|    '         `    |\\
:  ///|      |\\\  :          ///|   '           `   |\\\
: ////|      |\\\\ :         ////|  '             `  |\\\\
://///|      |\R\\\:        /////| '               ` |\R\\\
+-----o      o-----+       +-----o                  `o-----+

+-- X,Y,ZOFF +-- X,Y,ZOFF  +-- X,Y,ZOFF              +-- X,Y,ZOFF
|   = 0,6,6  |   = 6,6,0   |   = 0,6,6               |   = 6,6,0
v            v             v                         v
+-----o      o-----+       +-----o                   o-----+
:\\\I\|      |/IR//:        \\\I\|`                 '|/IR//
: \\\\|      |//// :         \\\\| `               ' |////
:  \\\|      |///  :          \\\|  `             '  |///
:   \\|      |//   :           \\|   `           '   |//
:    \|      |/    :            \|    `         '    |/
` - - +      + - - '             + - - `       ' - - +
```

DISPLAY PARAMETERS:

```
            F          9                 0
           +-----------+-------------------+
  DISP:    |  STENCIL  |       YPOS        |     STENCIL mode
           |  page 26  |      page 27      |      & Y-POSition
           +-----------+-+-----------------+
           |  LAMINATE |%|      ZPOS       |     LAMINATE mode
           |  page 27  | |     page 27     |      & Z-POSition
           +-----------+-+-----------------+
           |% % % % % %|      XPOS/2       |     X-POSition
           |           |     page 27       |
           +-----------+-------------------+
```

STENCIL (STENCIL mode) DEFINED:

```
            F        A
           +-----------+-------------------+
DISP:      |  STENCIL  |                   |
           +-----------+-------------------+
```

There are 33 stencil modes for each DISPlay as follows:

```
0%%%%% - all sprites referencing this DISPlay are self-stenciling
         (ie, their own stencils determine their display).
1nnnnn - all sprites referencing this DISPlay are stenciled
         according to the stencil of the sprite currently being
         generated by percept 'nnnnn'.
```

With stenciling on, the resultant display will be the topological intersection of the stenciled sprite and the stenciling sprite, thus:

```
STENCILED +++
 SPRITE  +++++
        +++****ooooooo                  ++++ooooooo      RESULTING
       ++++*****oooooo                  +++++oooooo   SCREEN DISPLAY
      +++++****++ oooo      =====>      ++++   oooo
     ++++++****+++oooo      =====>      ++++   oooo
    +++++++****+++*ooo                  ++++   +ooo
   ++++++++****+++**oo                  ++++   ++oo
           oooo   oooo                oooo   oooo
           ooooooooooo  STENCILING    ooooooooooo
           ooooooooooo    SPRITE      ooooooooooo
```

Stenciling is performed irrespective of z-position. Stenciling is performed in x & y only.

Stenciling is different from masking. A mask defines the area where a sprite IS NOT to appear, however, a stencil defines the area where a sprite IS to appear. Masks are data intensive and they are useless except as masks. Stencils are data conservative since only the area of interest is reproduced in the data and a stencil can also be used as a graphic sprite for display purposes.

DISPLAY PARAMETERS (continued):

XPOS, YPOS & ZPOS (X-POSition, Y-POSition & Z-POSition) DEFINED:

```
                        9                  0
           +-----------+-------------------+
DISP:      |           |       YPOS        |
           +-----------+-+-----------------+
           |             |      ZPOS       |
           +-----------+-+-----------------+
           |           |      XPOS/2       |
           +-----------+-------------------+
```

These signed binaries are the x,y & z values of course or group positioning to which XOFF, YOFF & ZOFF are added to arrive at the sprite's true x, y & z positions on the screen. Both these parameters and their associated offsets are full resolution and full screen so that they can be individually used to perform complete positioning. The presence of the offsetting ability, though, makes group motion and/or screen scrolling much easier.

For example, suppose that there are a number of playfield objects in static relationship with each other. If they are all offset from a single position contained in this parameter group, and they all point to this parameter group with their DISPlay pointers, then they can be simultaneousy scrolled in X and/or Y and/or Z with one to three stores to these locations. Further, suppose that a moving object consists of a number of other sprites layered and/or arranged together. With appropriate offsets from a group position stored here, that moving object can be repositioned with from one to three stores. A combination of moving objects and scrolling playfield objects can be managed with two or more DISPlay parameter groups.

LAMINATE (LAMINATE mode) DEFINED:

```
             F         A
            +-----------+-------------------+
DISP+1:     |  LAMINATE |                   |
            +-----------+-------------------+
```

There are 33 laminate modes for each DISPlay as follows:

```
0%%%%% - all sprites referencing this DISPlay are self-laminating
         (ie, their own laminates determine their display).
1nnnnn - all sprites referencing this DISPlay are laminated
         according to the laminate of the sprite currently being
         generated by percept 'nnnnn'.
```

Laminating provides a surface upon which graphic sprites must project their data. The laminating sprite can either be a planer or a surface. Laminating sprites are the only participants in automatic z-prioritization. A graphic sprite must be laminated to appear on the screen. A sprite can laminate itself either by setting the LAMINATE mode to its own percept number or by setting the LAMINATE mode to zero (so that all sprites referencing the display parameters laminate themselves).

```
 DISPLAY PRIORITIZATION HARDWARE ALGORITHM:

   +---------------------------+
   | wait for next color clock <--------------------------------------+
   +---------------------------+                                      |
   |       clear flags         |                                      |
   | set INTEN = COLOR = 0000  |                                      |
   +-------------v-------------+                                      |
                 |                                                    |
     ------------V--------------  no                                  |
    < non-zero data to output ? >------------------------------------>+
     ------------v--------------                                      |
                 | yes                                                |
     ------------V-------------  yes          +------------+          |
    < self-laminating sprite ? >--------------> update X,Z |          |
     ------------v-------------               +-----v------+          |
                 | no                               |                 |
       ----------V----------  no                    |                 |
      < master's flag set ? >-----------------------|---------------->+
       ----------v----------              ----------V-----------      |
                 | yes +----------+  yes <   lowest Z of all    > no  |
                 +<----+ set flag <------< laminates in block ? >---->+
                 |     +----------+       ----------------------      |
      -----------V-----------  no                                     |
     < self-stencil sprite ? >-------------------+                    |
      -----------v-----------                    |                    |
                 | yes                 ----------V----------  no      |
           +-----V----+               < master's flag set ? >-------->+
           | set flag |                ----------v----------          |
           +-----v----+                          | yes                |
          |                      |               |
                 +<------------------------------+                    |
                 |                                                    |
 ----------------V----------------  yes  +------------------+         |
< is sprite an intensity sprite ? >------> set INTEN = data |         |
 ----------------v----------------       +--------v---------+         |
                 | no                             |                   |
                 +<-------------------------------+                   |
                 |                                                    |
    -------------V--------------  no                                  |
   < is sprite a color sprite ? >------------------------------------>+
    -------------v--------------                                      |
                 | yes                                                |
      -----------V-------------                                       |
     <  lowest numbered color  > yes     +------------------+         |
     < percept in this block ? >---------> set COLOR = data +-------->+
      -----------v-------------          +------------------+         |
                 | no                                                 |
                 +--------------------------------------------------->+
```

PALETTE MEMORY MAP:

```
   +--------------------------- COLOR[11,8] ; from PENNY2
   |+-------------------------- COLOR[7,4]  ; from PENNY1
   ||+------------------------- COLOR[3,0]  ; from PENNY0
   |||
   |||   F         A         5       1 0 <--- bit
   VVV  +---------+---------+-------+---+
0EC000: |   LUM   |   P1    |  P0   |ISS| This is the background color
        +---------+---------+-------+---+ or external video switch.
        +---------+---------+-------+---+
0EC00k: |   LUM   |   P1    |  P0   |ISS| This is PENNY0 color-k.
        +---------+---------+-------+---+
        +---------+---------+-------+---+
0EC0j0: |   LUM   |   P1    |  P0   |ISS| This is PENNY1 color-j.
        +---------+---------+-------+---+
        +---------+---------+-------+---+
0ECi00: |   LUM   |   P1    |  P0   |ISS| This is PENNY2 color-i.
        +---------+---------+-------+---+
        +---------+---------+-------+---+
0ECijk: |   LUM   |   P1    |  P0   |ISS| Combinations of colors can
        +---------+---------+-------+---+ be used to determine visual
            ^         ^        ^     ^    priority between PENNY's or
            |         |        |     |    for color mixing, shading,
         LUMinance  Phaser1  Phaser0 |    special effects, etc.
                                     |
Intensity Source Select -------------+
```

00 ==> intensity determined by intensity sprite from PENNY0
01 ==> intensity determined by intensity sprite from PENNY1
10 ==> intensity determined by intensity sprite from PENNY2
11 ==> full intensity commensurate with stored luminance & phasers

TWO METHODS OF GENERATING COLORS:

1, Select values for LUM, P1 & P0 from the color charts on the next
   four pages or

```
2, Calculate values for LUM, P1 & P0 as follows:

        ;; given desired amount of intensity of primary colors

        off = 0 =< R,G,B =< 23.9 = brightest

        ;; store LUMinance

        LUM = INTEGER[.30*R+.59*G+.11*B+.49]

        ;; store Phaser1

        TEMP = .877*(.70*R-.59*G-.11*B)
        P1 = INTEGER[TEMP-.49]  ; if TEMP < 0
        P1 = INTEGER[TEMP+.49]  ; if TEMP >= 0

        ;; & store Phaser0

        TEMP = .493*(.89*B-.30*R-.59*G)
        P0 = INTEGER[TEMP-.49]MOD16   ; if TEMP < 0
        P0 = INTEGER[TEMP/2+.49]MOD16 ; if TEMP >= 0
```

```
   legal NTSC ----> *m <--- minimum allowable LUM for this color point
   color point       n <--- maximum allowable LUM for this color point

                                NTSC COLOR CHART (FIRST QUADRANT)

              V A L U E S   O F   P 0
---------------------------------------------------
0         1         2         3         4         5

+         +         +         +         +         +   16 |
|                                                        |
|                                                        |
+         +         +         +         +         +   15 |
|                                                        |
|                                                        |
*9        +         +         +         +         +   14 |
|9                                                       |
|                                                        |
*8        *9        *10       +         +         +   13 |
|9         9         10                                  |
|                                                        |
*7        *8        *9        *10                 +   12 |
|10        10        10        11  PURE                  |
|                            +---+ MAGENTA               |
*7        *8        *8       |*9 |                +   11 | V
|12        12        11       | 11|                       | A
|                            +---+                       | L
*6        *7        *8        *9        +         +   10 | U
|12        13        13        12                        | E
|                                                        | S
*6        *6        *7        *8        +         +   9  |
|14        13        14        12                        | O
|                                                        | F
*5        *6        *7        *7        *8        +   8  |
|15        15        15        12        8               | P
|                                                        | 1
```

```
*5        *5        *6        *7        *8        +   7  |
|16        16        16        12         8                |
|                                                          |
*4        *5        *5        *6        *7        +   6  |
|17        17        15        12         8                |
|                                                          |
*3        *4        *5        *6        *6        +   5  |
|18        18        16        12         7                |
|                                                          |
*3        *4        *4        *5        *6        +   4  |
|20        20        16        12         8                |
|                                                          |
*2        *3        *4        *5        *5        +   3  |
|20        20        16        12         8                |
|                                                          |
*2        *2        *3        *4        *5        +   2  |
|22        20        16        12         8                |
|                                                          |
*1        *2        *3        *3        *4        +   1  |
|23        20        16        11         8                |
|                                                          |
*0 -------*1 -------*2 -------*3 -------*4 -------+   0  |
 24        20        16        12         8
```

CROSS POINTS (+) ARE ILLEGAL NTSC COLORS...BUT THEY
CAN BE USED FOR MONITORS. MAXIMUM AND MINIMUM
LUM VALUES DO NOT APPLY TO MONITORS EITHER.

NTSC COLOR CHART (SECOND QUADRANT)

```
                         V A L U E S   O F   P 0
            ----------------------------------------------------
            6    7    8    9   10   11   12   13   14   15     0

     | 16   +    +    +    +    +    +    +    +    +    +    +
     |                                            |
     |                                            |
     | 15   +    +    +    +    +    +    +    +    +    +    +
     |                                            |
     |                                            |
     | 14   +    +    +    +    +    +   PURE  *9   *9   *9   *9
     |                                    RED   9    9    9   |9
     |                                            |
     | 13   +    +    +    +    +    +    *9   *8   *8   *8   *8
     |                                     10   10   8    9   |9
     |                                        +---+     |
     | 12   +    +    +    +    +    *11  *9  |*7 | *7   *7   *7
     |                                11   11 | 11|  9    9   |10
     |                                        +---+     |
   V | 11   +    +    +    +    +    *11  *9   *7   *6   *6   *7
   A |                                12   12   12   10   10  |12
   L |                                            |
   U | 10   +    +    +    +    *13  *11  *9   *7   *6   *6   *6
   E |                           13   13   13   13   12   12  |12
   S |                                            |
     |  9   +    +    +    +    *13  *11  *9   *7   *5   *5   *6
   O |                           14   14   14   14   14   13  |14
   F |                                            |
     |  8   +    +    +    *15  *13  *11  *9   *7   *5   *5   *5
   P |                      15   15   15   15   15   15   14  |15
   1 |                                            |
     |  7   +    +    +    *15  *13  *11  *9   *7   *5   *4   *5
     |                      16   16   16   16   16   16   15  |16
     |                                          |
     |  6   +    +    *17  *15  *13  *11  *9   *7   *5   *4   *4
     |                 18   18   18   18   18   18   18   17  |17
     |                                          |
     |  5   +    +    *17  *15  *13  *11  *9   *7   *5   *3   *3
     |                 19   19   19   19   19   19   19   18  |18
     |                                          |
     |  4   +    *19  *17  *15  *13  *11  *9   *7   *5   *3   *3
     |            20   20   20   20   20   20   20   20   20  |20
     |                                          |
     |  3   +    *19  *17  *15  *13  *11  *9   *7   *5   *3   *2
     |            21   21   21   21   21   21   21   21   21  |20
     |          +---+                           |
     |  2       |*19| *17  *15  *13  *11  *9   *7   *5   *3   *2
     |     PURE | 22|  22   22   22   22   22   22   22   22  |22
     |   YELLOW +---+                           |
     |  1        *19  *17  *15  *13  *11  *9   *7   *5   *3   *1
     |            21   22   22   23   23   23   23   23   23  |23
     |                                          |
     |  0   +----*19 -*17 -*15 -*13 -*11 -*9 --*7 --*5 --*3 --*0
                  21   21   22   22   22   23   23   23   23   24
```

```
  |  0   +----*19 -*17 -*15 -*13 -*11 -*9  --*7 --*5 --*3 --*0
  |            21   21   22   22   22   23   23   23   23  |24
  |                                           |
  | 31   +    *19  *17  *15  *13  *11  *9   *7   *5   *3   *2
  |            20   21   21   21   22   22   23   23   23  |23
  |                                           |
  | 30   +    *19  *17  *15  *13  *11  *9   *7   *5   *3   *3
  |            20   20   20   21   21   22   22   23   23  |23
  |                                           |
  | 29   +    *19  *17  *15  *13  *11  *9   *7   *5   *4   *4
  |            19   20   20   20   21   21   22   22   22  |22
  |                                           |
V | 28   +    +    *17  *15  *13  *11  *9   *7   *5   *5   *5
A |                 19   19   20   20   21   21   21   21  |22
L |                                           |
U | 27   +    +    *17  *15  *13  *11  *9   *7   *6   *6   *6
E |                 18   19   19   20   20   20   20   21  |21
S |                                           |
  | 26   +    +    *17  *15  *13  *11  *9   *7   *7   *7   *7
O |                 18   18   19   19   19   19   19   20  |20
F |                                           |
  | 25   +    +    *17  *15  *13  *11  *8   *8   *8   *8   *8
P |                 17   18   18   18   18   18   19   19  |19
1 |                                           |
  | 24   +    +    *17  *15  *13  *11  *10  *10  *10  *10  *10
  |                 17   17   17   18   18   18   19   19  |19
  |                                           |
  | 23   +    +    +    *15  *13  *11  *11  *11  *11  *11  *11
  |                      16   17   17   18   18   18   19  |19
  |                                           |
  | 22   +    +    +    *15  *13  *12  *12  *12  *12  *12  *12
  |                      16   16   16   17   17   18   18  |18
  |                         +---+                   |
  | 21   +    +    +    *15 |*13| *13  *13  *13  *13  *13  *13
  |                      15 | 15|  16   16   16   17   17  |18
  |                         +---+                   |
  | 20   +    +    +    *15  *14  *14  *14  *14  *14  *14  *14
  |                      15   15   15   15   16   16   16  |17
  |                 PURE                       |
  | 19   +    +    GREEN     +    +    *15  *15  *15  *15  *15
  |                                     15   15   15   16  |16
  |                                            |
  | 18   +    +    +    +    +    +    +    +    +    +    +
  |                                            |
  |                                            |
  | 17   +    +    +    +    +    +    +    +    +    +    +

         6    7    8    9   10   11   12   13   14   15    0
         ---------------------------------------------------
                      V A L U E S   O F   P 0
```

NTSC COLOR CHART (THIRD QUADRANT)

```
*0 -------*1 -------*2 -------*3 -------*4 -------+    0 |
|24        20        16        12        8               |
|                                                        |
*2        *2        *2        *2        *3        *5  31 |
|23        20        16        12        8         5      |
|                                      +---+             |
*3        *3        *3        *3       |*3 |      *5  30 |
|23        20        16        12      | 8 | PURE  5      |
|                                      +---+ BLUE         |
*4        *4        *4        *4        *4        *5  29 |
|22        20        16        12        8         5      |
|                                                        |
*5        *5        *5        *5        *5        +   28 | V
|22        20        16        12        8               | A
|                                                        | L
*6        *6        *6        *6        *6        +   27 | U
|21        20        16        12        8               | E
|                                                        | S
*7        *7        *7        *7        *7        +   26 |
|20        20        16        11        7               | O
|                                                        | F
*8        *8        *8        *8        +         +   25 |
|19        19        15        11                        | P
|        +---+                                           | 1
*10      |*10|      *10       *10       +         +   24 |
|19      | 20|       16        12                        |
|        +---+ PURE                                      |
*11       *11  CYAN *11       *11       +         +   23 |
|19        20        16        12                        |
|                                                        |
*12       *12       *12       *12       +         +   22 |
|18        19        16        12                        |
|                                                        |
*13       *13       *13       +         +         +    21 |
|18        18        16                                   |
|                                                         |
*14       *14       *14       +         +         +    20 |
|17        18        16                                   |
|                                                         |
*15       *15       *15       +         +         +    19 |
|16        17        16                                   |
|                                                         |
+         +         +         +         +         +    18 |
|                                                         |
|                                                         |
+         +         +         +         +         +    17 |

0         1         2         3         4         5
---------------------------------------------------
              V A L U E S   O F   P 0          +----------+
                                      | IF THIS |
                     NTSC COLOR CHART (FOURTH QUADRANT) |IS SQUARE|
                                      | DRAWING |
              PREPARED BY MARK FILIPAK           | CAN BE  |
                                      | SCALED  |
                                      +----------+
```