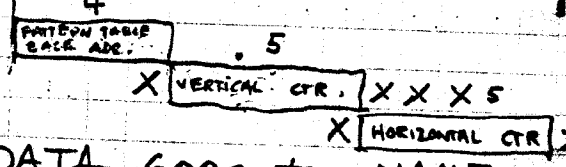
PATTERN NAME

Pattern NAME Address

(Scans though LIST of 768 Patterns by screen position (32 Columns 24 Rows, 32×24 = 768)

REG2

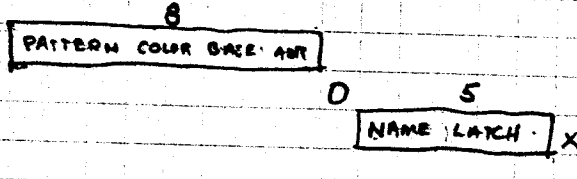


Pattern TABLE BASE Address (VR#2)
Vertical Counter (Row)
Horizonta Counter (Column)

DATA GOES to NAME LATCH

Color Address (Select one of 32 Possible color combinations. Note Pattern are Grouped by Color

Reg3

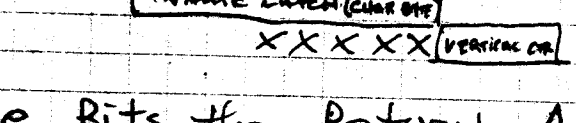


Pattern Color Base Address
Force 0
5 MSB's of NAME LATCH

Result goes to color Buffer to Await fetch of Pattern. Note 2 4 Bit codes. The 4 MS Bits are the "1" Color the 4 LSBits are the "0" color

Pattern Line Address (The NAME Give one of 256 diffent Pattern Shapes (8 Bytes of 8). The Vertical Count LSB's select obt which Byte off the Shape

Reg4



3 Bit of NAME TABLE BASE
8 Bits of NAME LATCH
3 LSB's of Vertical Counter

The Bits the Return Are Put in PATTERN SHIFT Register AND are shift out MSB First. Color Buffer is Load into Color LATCH.

		ADDRESS GENERATED	DATA STORED
N	①	PATTERN NAME ADR	8-BIT NAME CODE → NAME LATCH
C	②	COLOR ADR	8-BIT 2 COLORS → COLOR BUFFER
P	③	DOT PATTERN ADR (DOT RAM)	8-BIT DOT PATTERN → PATTERN SHIFT REG
			COLOR BUFFER → COLOR LATCH

Sprite Preprocessing (During + Before Line before line to be shown)

Reg 5 → **VERTICAL ATTRIBUTE POINTER** → 0 0

Sprite Attribute Base From Sprite Counter (Increment Each time)
Forced to Give Vertical Attribute in Attribute List

FIFO

- Fetch Vertical Position of Each sprite in turn. If Range test indicated sprite is to be shown on next line then push Sprite Counter Number onto FIFO stack. If sprite stack is full (5 ALREADY ON) then stop preprocessing. If Vertical Position = 208 dec then stop preprocessing (allows sizing sprite space)

Sprite Post Processing (At End of Line before showing)

(Vertical Fetch)

Reg 5 → 0 0

Sprite Attribute BASE
NEXT VALID LOCATION OUT of FIFO
Attribute Counter ↑ ie

Subtract Vertical Position MINUS SPRITE Vertical Position (to be used later) (ALREADY KNOWS it is IN RANGE).

GETS READY AS NEXT FETCH SEQ. TAKES PLACE

Fetch Horizontal Position

Reg 5 → 0 1

SAME FIFO As Vertical Attribute Counter

LOAD INTO Respective Sprite Horizontal Counter (1 for Each of 4 Possible sprites)
(COUNT is Stopped when loaded WAITS For Zero if TAG Bit 6 IS 0 WAIT for -32 if TAG Bit 6 is 1)

count down start time depends on TAG Bit 6

Fetch NAME of Sprite

1 0

same same same Attribute Counter

NAME Goes to NAME LATCH

Fetch TAG of Sprite

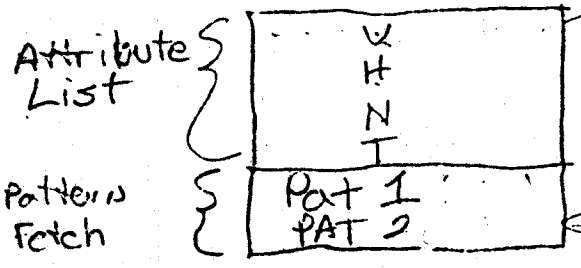
1 1

same same same Attribute Counter

Pull Next Sprite Number OUT of FIFO
LOAD 4 LSBs of TAG into Respective Sprite Color LATCH
MSB of TAG is Early CLOCK (or Early Countdown Bit)
3 Bits are UNUSED.

6	X	X	X	COLOR

EARLY COUNT DOWN



Vertical Pos (Attribute)
Horz Position
Sprite NAME
TAG (Early Clock; 4 Bits of Color)

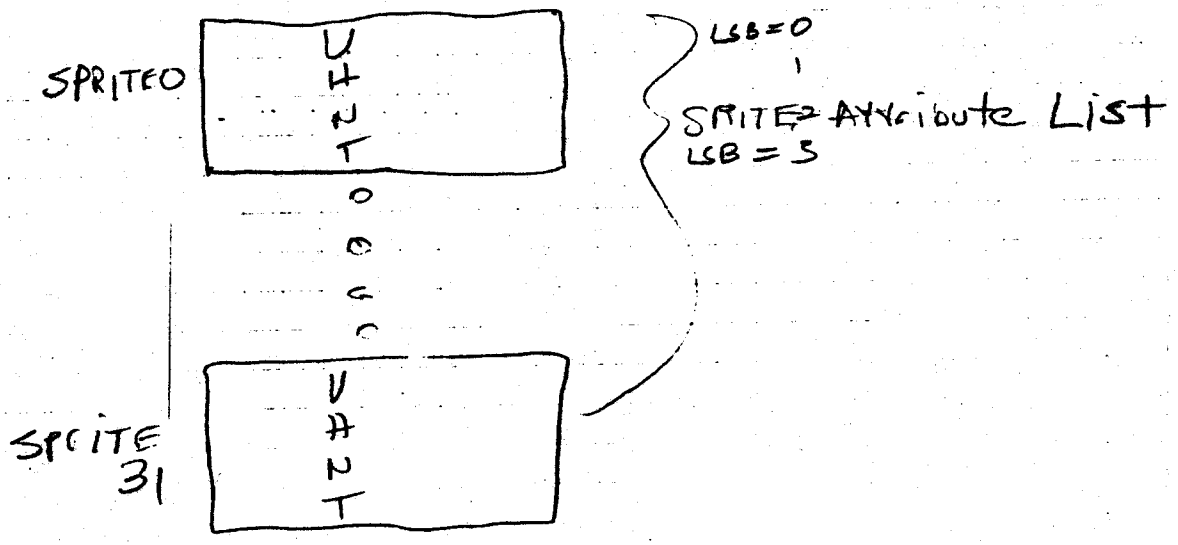
(Only Done If SIZE=1)

Preprocessing Of Sprites
Fetch Vertical position of Sprites Compare against Vertical COUNT (Note that the current Vertical Count is used even though shown on next line (ie sprites are shown 1 line lower than Vertical Position.
Preprocessing occurs during showing of current line. If sprite is in "RANGE" then the number of the sprite is put on a FIFO stack
The "RANGE" is 8 Lines wide for 8X8 sprites 16 Lines for 16X16 sprites and 32 lines for 32X32 sprites (size=MAG=1).
RANGE = (Vertical Count) - Sprite Vertical Attribute

For Size=0 MAG=0 RANGE to be shown is

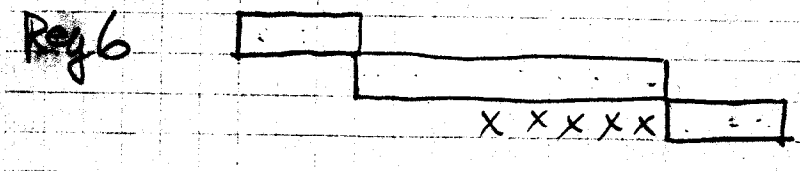
0	1
1	0
1	1

$0 \leq RANGE \leq 7$
$\leq RANGE < 16$
$\leq \quad < 15$
$\leq \quad \leq 31$



Sprite PATTERN Fetches

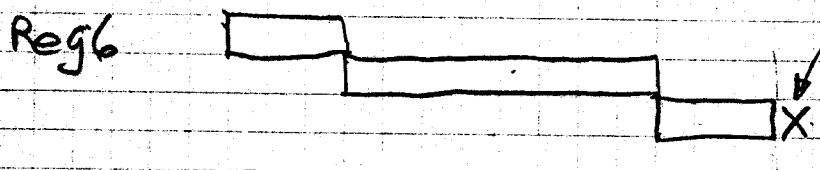
FOR MAG=0 SIZE=0



SPRITE PAT GEN BASE NAME LATCH
SUBTRACTION RESULT (RANGE) 3 LSB's

Fetch 1 Patern LINE STORE IN 1ST Sprite Shift Register

FOR MAG=1 SIZE=0



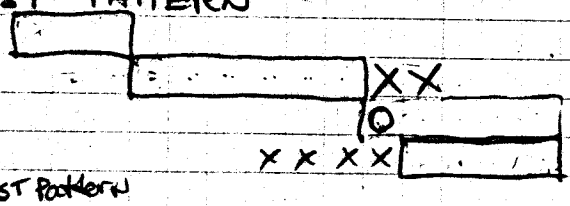
Subtraction Result Shift Right by 1 (Divide By 2)

In this Case SPRITE shift register shift at 1/2 DOT RATE

THIS GIVES 2x LENGTH

FOR MAG=0 SIZE=1

FIRST PATTERN



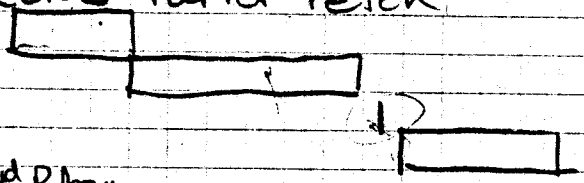
6 MSBs of NAME

4 LSBs of Subtraction (RANGE)

1ST Pattern

LOAD INTO 1ST shift Reg of Respective Sprite.

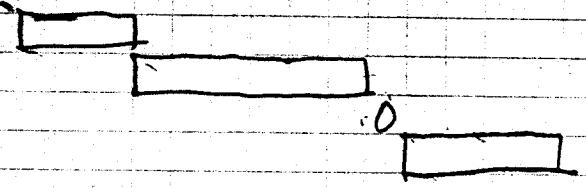
SECOND Patter Fetch



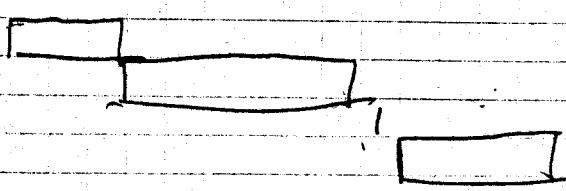
2nd Pattern

LOAD INTO 2nd Shift Reg. of Respective Sprite

FOR MAG=0 SIZE=0



Subtr. Result Right Shifted



FOR MAG1 Shift Registers Count at 1/2 RATE,