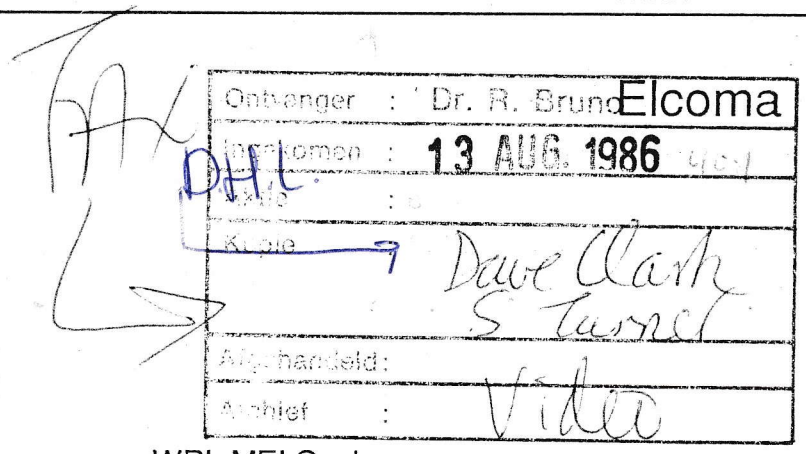
IMSC International Microelectronics Support Centre

CIMA
4 à 16, av. du Général-Leclerc
92260 Fontenay-aux-Roses
France

Electronic Components and Materials

Tél.: (1) 47 02 75 45

Télex 280 746F



From: JC SIX

To: MM

ABE	WRL MEI Osaka
KIMURA	MEC Nagaoka
TOYODA	"
HIRAYAMA	"
BRUNO	HIS Eindhoven
van LUIJT	"
LUIJKS	"
AKKERMANS	Elcoma Eindhoven
v. d. HOVEN	"
TAILLADE	HIS Fontenay
BRAUNE	IMSC Fontenay

Date: July 22th 1986

Subject: Functional specification VSR

The following document is a new version of the functional specification issued by june 1986. It has been enhanced with an extensive description of the DYUV coding and decoding scheme in appendix.

The functionality of the chip has not changed.

> Some minor modifications have been included in order to avoid problems in the utilization of the device.

> HIS made a software simulation of the DYUV channel which concluded to the necessity of changing the diagram of the DYUV converter.

> Several parts have been described in a better way.

This version replaces the previous one.

This document and the appendix are **company confidential.**

BEST REGARDS

JC SIX

VIDEO SYNTHESIZER

The Video Synthesizer is a function to be used in conjunction with a Video Controller function able to generate two 7.5 Million Bytes per second data flows. The Video Synthesizer chip (VSR) will generate one composit video image via 3 analog outputs driving directly the R, G and B channels of an RGB monitor or a TV set equiped with a video input interface.

The chip processes several images in parallel and finally combines them making overlays or sums of pixel values on a pixel per pixel basis. The control of the device is made using control bytes passed via the pixel input ports during control sequences at retrace period.

The two input flows can be processed in various ways to generate from one to three images depending on the mode. Overlay on TV images is controlled by the chip.

FEATURES

- Up to15 MHz Pixel rate
- 6-bit/color Direct Analog RGB outputs
- Configurable 256-Entry Color-Look-Up-Table
- Dual channel Real-time Delta YUV decoding
- Real-time RUN-LENGTH decoding
- Up to 3 simultaneous images
- Up to 3 levels of overlay
- Test on transparent pixel for overlays
- Wipes and fading controller
- Dynamic reload during retrace
- Hard copy output
- CMOS technology
- Surface Mounted Device package

APPLICATIONS

- Consumer Video Display Processor
- CD-Interactive Video Processor
- Multi-image CRT controllers

PACKAGE

- 44-pin Plastic Leaded Chip Carrier Package (PLCC)

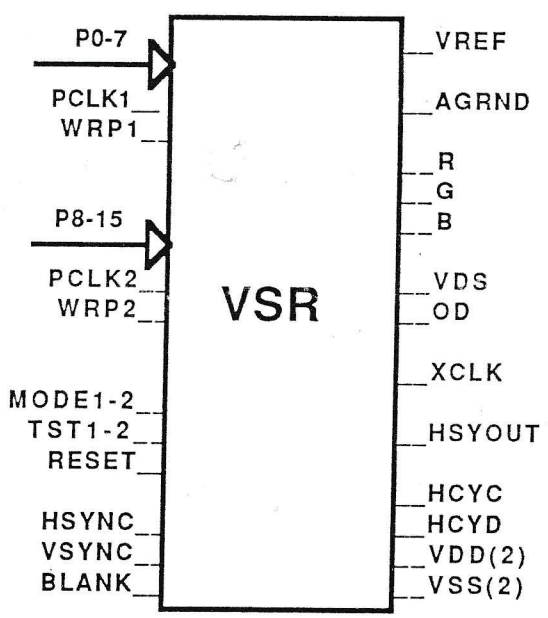


Figure 1: LOGICAL PIN DESCRIPTION

PHILIPS

IMSC ELCOMA 68000 SUPPORT GROUP
4-16 Avenue du Général Leclerc
92260 Fontenay-aux-Roses FRANCE

July 1986

BLOCK DIAGRAM

The following diagram shows the various elements of the video synthesizer. The two input buffers resynchronize the incoming data flows clocked by PCLK1 and PCLK2. Then an input multiplexor routes the 16 bits to the various processing paths. It is controlled by the mode register.

A run-length decoding function is then performed. On each of the two channels, various transformations can be applied for Direct RGB, RGB through a Color Look-Up Table (CLUT) or Delta YUV to RGB conversion. The results from each channel is then routed to the output multiplexer which is controlled real-time by the mode register, the color comparators and the overlay controller.

Finally, the results of each channel is applied to a set of 6-bit D/A converters generating a current contribution to produce the analog voltage levels.

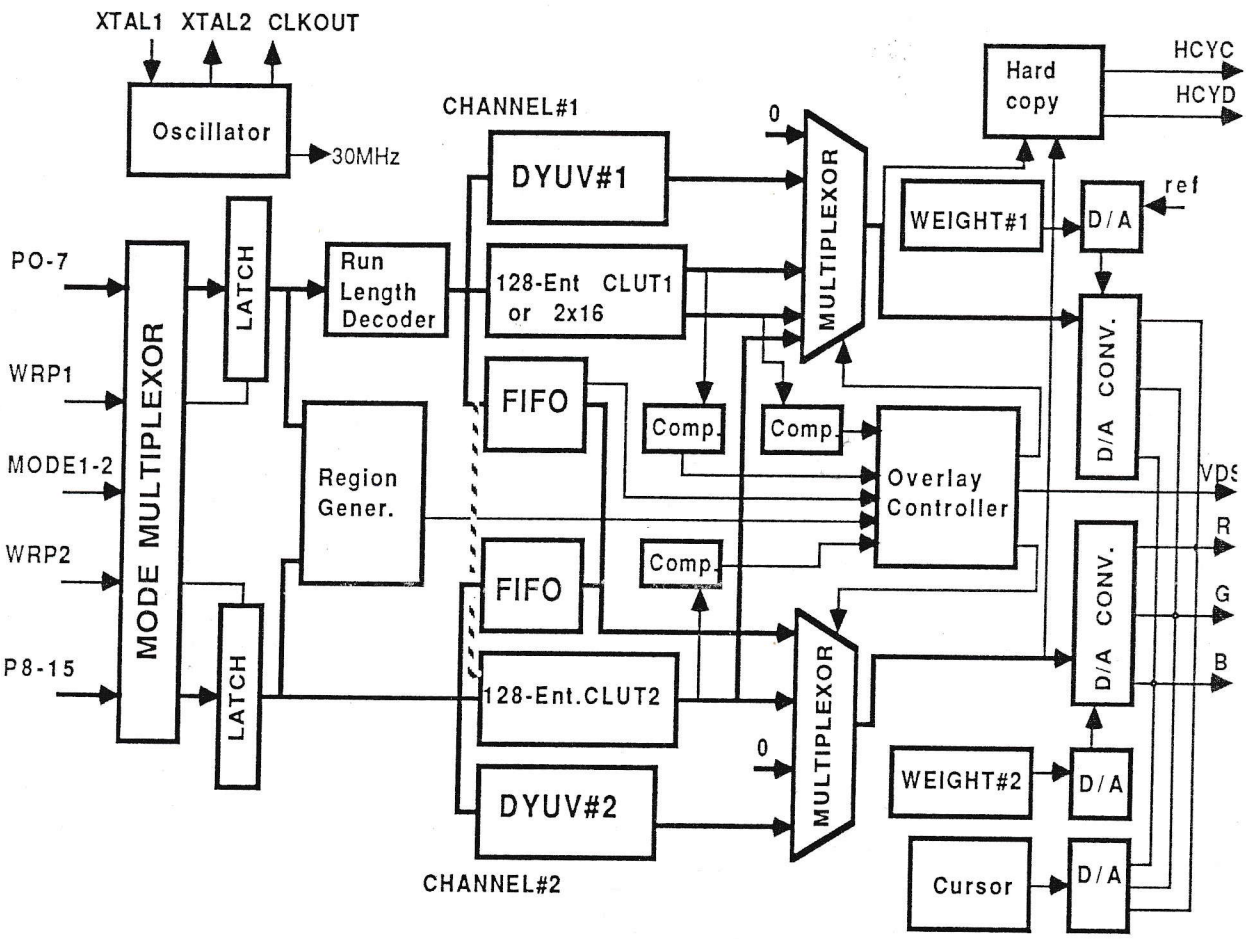


Figure 2: VIDEO SYNTHESIZER BLOCK DIAGRAM

PHILIPS
IMSC ELCOMA 68000 SUPPORT GROUP
4-16 Avenue du Général Leclerc
92260 Fontenay-aux-Roses FRANCE

PIN ARRANGEMENT

P0-P7	I	8-bit Pixel bus for Channel #1.
PCLK1	I	Strobe input for Channel #1. P0-P7 will be clocked on Low to High edge.
P8-P15	I	8-bit Pixel bus for Channel #2.
PCLK2	I	Strobe input for Channel #2. P8-P15 will be clocked on Low to High edge.
WRP1-2	I	Active Low control active inputs. When High Data is considered as image content. When Low Data is considered as control information.
$\overline{\text{BLANK}}$	I	Blank control line from the video controller. RGB outputs will be zeroed when blank is active (Low).
$\overline{\text{HSYNC}}$	I	Active Low Horizontal synchronisation input from video controller.
$\overline{\text{VSYNC}}$	I	Active Low Vertical synchronisation input from video controller.
$\overline{\text{HSYOUT}}$	O	Active Low Horizontal synchronisation delayed output for Monitor or TV set.
XCLK1	I	Clock Oscillator input (for crystal max 30 MHz), also external clock input
XCLK2	O	Clock Oscillator output (for crystal)
CLKOUT	O	Buffered Clock output for the Video Controllers

CRT INTERFACE

R, G, B	O	Analog color outputs.
VREF	I	Reference voltage for analog signals
AGND		Analog ground
$\overline{\text{VDS}}$	O	Digital output to control an external switch for TV overlays.
$\overline{\text{OD}}$	I	Active Low Output Disable input for R,G,B and VDS signals.

MISCELLANEOUS

VDD (2)		Power supply (5V)
VSS (2)		Power and digital ground
TST	I	Test input pin
$\overline{\text{RESET}}$	I	Active Low Reset input signal for chip initialisation.
MODE1-2	I	Mode input pins
HCYD	O	Digital serial data output for screen hard copy
HCYC	O	Clock output for screen hard copy

FUNCTIONS

The chip can process the data flows in various ways by having several pixel paths to display images using various coding schemes. The main operations are:

- Direct RGB images
- RGB through a Color Look-Up Table
- Delta YUV TO RGB converter
- Run Length decoder

The RGB flows are then combined by the mixing functions.

DECODING FUNCTIONS

- Direct RGB images

This mode is mainly intended for computer graphic images.
The color of the pixel is directly defined by a bit pattern giving the levels of Red, Green and Blue to load into the Digital to Analog converters.
A 15-bit pattern is used, each D to A converter receives 5 bits. The last bit (MSB) is used as a transparency bit.

- RGB through CLUT

It is an indirect definition of the color of a pixel. The pattern is used to point at one address in a fast RAM which gives the values to load into the D to A converters. The full possibility of the chip is then reached by generating one color out of 2^{18} (262144) colors.

Several modes can be used:

> An 8-bit pattern is used to address one out of 256 entries
> A 7-bit pattern is used to address one out of 128 entries
> A 4-bit pattern is used to address one out of 16 entries.

In each case, the output is 3 times 6 bits for Red, Green and Blue. The input rate can be 15 MHz when used in high resolution mode (768 pxl/line). The Video Controller function can be such that a 15 MBytes/S rate is generated using several devices in parallel.

Run-Length decoder

The RGB through CLUT 7bits mode can be used in conjunction with a RUN-LENGTH decoder allowing for very compact representation of images where the values of the pixel is not changing often during a video line. The coding scheme uses the eigth bit (MSB) of each byte as a flag.

> If the MSB is zeroed, the seven remaining bits define the color of a "single pixel" of the given color.

> If the MSB is set, the seven remaining bits define the color of a series of pixels and the following byte gives the number of identical pixels to be generated.

If the count value is zero, the current video line is completed with pixels of the given color. In any case, the run-length stops at the end of the current line.

- Delta YUV modulation

This mode is mainly useful for the reproduction of high quality natural images. It uses rather low data rates for a high quality image thanks to several data compression mechanisms. A very high quality can be obtained with only 100 KBytes per image.

The incoming code is a series of values to be used to calculate the values to be propagated to the D/A converters during one video line. It uses simultaneously two different principles: the YUV coding and the Delta modulation principles.

The YUV coding scheme defines the luminance information (Y) independently of the Color information (U and V). At the image generation stage, the U and V components are generated from the R, G and B values by the following unique set of formulas:

$$Y = R \times 0.299 + G \times 0.587 + B \times 0.114$$
$$U = (B-Y) \times 0.564$$
$$V = (R-Y) \times 0.713$$

Then the coefficients defined in the CCIR recommendation and a DC shift are applied giving the following values.

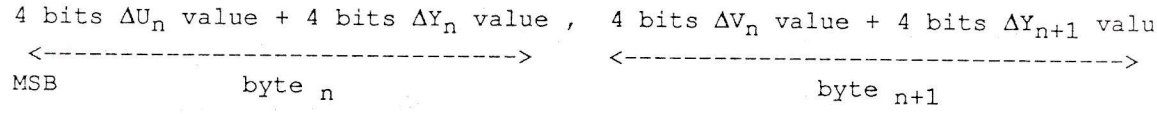
$$Yc = 219/255 \times (R \times 0.299 + G \times 0.587 + Bx0.114) + 16$$
$$Uc = ((B-Y) \times 0.564 \times 224/255) + 128$$
$$Vc = ((R-Y) \times 0.713 \times 224/255) + 128$$

The resolution of the human eye being lower for chrominance information than for luminance information, makes it possible to use a different resolution for U and V information than for Y information. The coding scheme implemented in the Video Synthesizer uses a U and V resolution which is half the Y resolution. It gives a compression factor of 12 to 8.

The Delta modulation allows for ancther compression factor of 2 by defining the value of one element relatively to its predecessors in the current video line. An initial absolute value must be given prior to start the accumulation to obtain the Y U V values.

The Delta YUV coded information will be converted real-time into RGB information by the DYUV-RGB converters of the Video Synthesizer.

For each line, an absolute YUV value is given first , then a series of Delta values organized in the following way:



> A first stage will transform Delta YUV pairs of bytes in a Delta YUV 4,4,4 word by assembling the U and V values propagated in two consecutive elements.

> A second stage will transform the 12-bit Delta YUV 4,4,4 word in a 24-bit word absolute YUV code (8 absolute Y, 8 absolute U and 8 absolute V) by adding non linear differential elements to the current absolute values.

This second stage will generate a value to be added to the current absolute value using a look-up table in ROM.

The amplitude of the signal is defined in accordance with the CCIR recommendation i.e. the Black level is level 16 and White level is level 235. This safety margin and a clipping mechanism are used to avoid

unpredicted wrap-around situations.

The prediction scheme is according to the following table:

Received Code	Output Delta	
0	0	
1	1	
2	4	
3	9	
4	16	
5	27	
6	44	
7	79	
8	128	
9	177	(-79 modulo 256)
10	212	(-44 " ")
11	229	(-27 " ")
12	240	(-16 " ")
13	247	(- 9 " ")
14	252	(- 4 " ")
15	255	(- 1 " ")

> A third stage will interpolate the U and V values to generate the same rate than the Y rate.

> A fourth stage will convert the YUV 8,8,8 word in a RGB 6,6,6 word by multiplying the Y, U and V values by coefficients written in a ROM area prior to add the various components to regenerate R G and B values.

The compensation for the CCIR ratio (255/219) on the Y value and the ratio on U and V values and the compensation for DC shifting being included, the coefficients to be applied are given by the following equations:

$$R = Y + 1.371 \times (V-128)$$
$$G = Y - 0.337 \times (U-128) - 0.698 \times (V-128)$$
$$B = Y + 1.733 \times (U-128)$$

> Finally, after clipping, RGB values will be applied to the D/A converters.

The incoming data rate in DYUV mode is limited to 7.5 Megabytes per second per channel i.e one byte every 133nS. After the first stage, the U and V rates are half of this value but an interpolation is made in the third stage to generate the correct intermediate values to obtain Y, U and V values every 133nS. The RGB conversion will give 7.5 Mpixels/S rate at the D/A converter level.

Two DYUV to RGB converters can operate in parallel so that two natural images can be combined to compose the final image.

A more complete description of the DYUV coding and decoding scheme is developped in an appendix of this document.

The following functional diagram shows one of the DYUV to RGB converters:

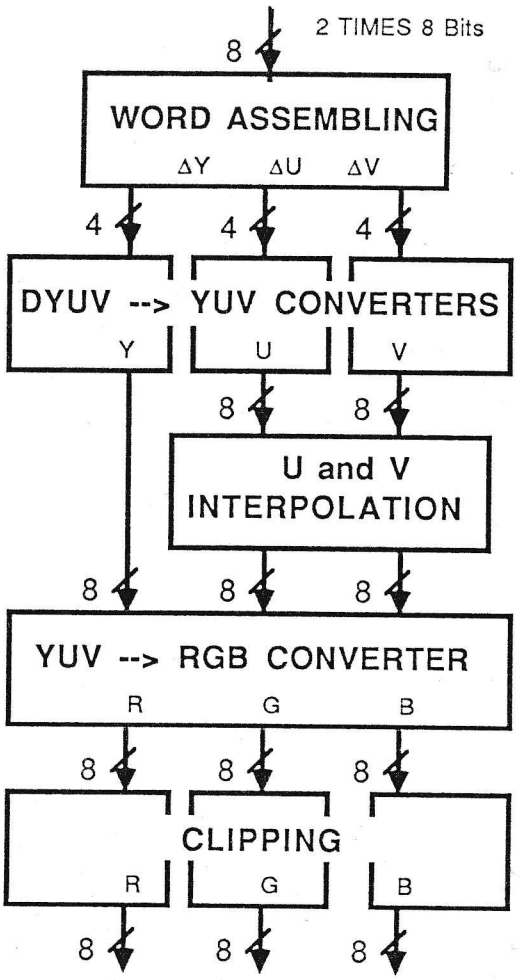


Figure 3 DYUV decoder block diagram

MODES

The previous technics will be used by the Video Synthesizer in different ways to generate the final image. Various configurations are programmable by loading the mode register of the chip.
The two 8-bit incoming flows can be combined or separated into several independant channels.

Channel# 1 defines basically the foreground image(s) and can be programmed for:

1 > RGB 8 bits CLUT
2 > RGB 4 bits CLUT1 + RGB 4bits CLUT1'
3 > 7 bits RGB CLUT1
4 > DYUV 8 bits

Channel# 2 defines basically the background image and can be programmed for

A > 7bits RGB CLUT2
B > DYUV 8 bits
C > RGB 15 bits Direct +transparency bit (with 8 bits from Channel #1)

Channel#1 can use bytes coming through ports P0-7 or P8-15 while Channel#2 uses bytes coming from the other port.

When the chip is driven using two Video controllers (BPCRT2s) , the high resolution mode (768 pixels / line) generates only 4 bits per incoming byte. As a result, channel #1 will generate only one image in mode 2 and Channel#2 will generate only 16 colors in mode A. The possibility to generate an 8 bit flow using two 4 bit outputs coming from two Video controllers (BPCRT2s) is also given by routing P0-3 and P8-11 to channel#1. The various configurations are controlled via SW1 and SW2 bits in the Mode register.

The following table summarizes the possible combinations.

Ch#1 / Ch#2	OFF	CLUT8	CLUT 4+4	CLUT7	CLUT 7or7	DYUV
OFF	O	O	O	O	O	O
CLUT7	O	X	O	O	X	O
DYUV	O	O	O	O	O	O
RGB15	O	X	X	X	X	X

O=POSSIBLE
X= NOT POSSIBLE

Figure 4: Mode table

The MODE register groups the various bits defining the decoding mode of Channel#1, Channel #2, the source of information for each channel and the Most significant bits of the CLUT address to reload.

The M10, M11 and M12 bits define the mode for channel#1.

M12	M11	M10		
0	0	0	Disable (output =0)	MODE 0
0	0	1	CLUT 8	MODE 1
0	0	0	CLUT 4+4	MODE 2
0	1	1	CLUT 7	MODE 3
1	0	0	CLUT 7or7	MODE 3'
1	0	1	DYUV	MODE 4

The M20, M21 and M22 bits define the mode for Channel#2.

M22	M21	M20		
0	0	0	Disable (output =0)	MODE 0
0	0	1	RGB15 (this mode turns Ch#1 off)	MODE C
0	1	0	N.U.	
0	1	1	CLUT 7	MODE A
1	0	0	N.U.	
1	0	1	DYUV	MODE B

The CLUT4+4 mode uses the addresses 0 to 15 of the CLUT1 with A6=0 for the entry defined by P0-3 and the same address with A6=1 (CLUT1') for the entry defined by P4-7.

The CLUT 7 or 7 mode is a CLUT 7 mode in which the CLUT1 or the CLUT2 can be used for Channel#1. The Clut Select bit (CS) determines if CLUT 1 (if CS=0) is used or if it is CLUT2. This mode is not usable with the

CLUT7 mode selected on Ch#2.

The SW1 and SW2 bits are used to swap or rerout the information presented on port P0-7 to Channel#2 and information presented on P8-15 to Channel#1:

If SW2=0	SW1=0	Normal (P0-7 -> Ch#1 and P8-15 -> Ch#2)
	SW1=1	Swapped (P0-7 -> Ch#2 and P8-15 -> Ch#1)
If SW2=1	SW1=0	Merged mode (P0-3 and P8-11 -> Ch#1 rest is don't care)
	SW1=1	Swapped merged mode (P0-3 and P8-11 -> Ch#2 rest is don't care)

Notice that the SW2 bit is considered to be zero during the control sequences so P0-7 or/and P8-15 are used to transfer DCA control information. The SW bits can be changed during horizontal retrace but in order to avoid problems, they are only effective at the beginnig of the next display period.

The CS bit defines which part of the CLUT is used in the CLUT7 or7 mode.

The RL enables the Run Length decoder of channel#1.

The CU2 bits controls the way the cursor is superimposed.

The A6 and A7 bits are the MSBs of the address of the CLUT entry to be reloaded.

The RE bit is defines if one or two regions are used (0= one region, 1= two regions)

The SI (superimpose) bit enables when set, the VDS signal for the overlay on TV image.

The layout of the MODE register is:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M22	M21	M20	M12	M11	M10	SW2	SW1	RL	CU2	CS	RE	SI	X	A7	A6

MIXING FUNCTIONS

The previously generated images are then combined to define the final image by overlaying up to 5 different planes which are:

The cursor plane
The first Foreground plane (FG1)
The second Foreground plane (FG2) , if available
The Background plane (BG)
The TV plane (TV)

The first function consists of defining what is the position of each plane on the Z axis from the front to the back. The second function is to define transparent areas in the planes to allow to see the planes behind. This is made by the overlay priority controller using signals generated by the Cursor controller, the color key controller and the region controller.

The relative position of each plane on the Z axis can be programmed by the PRIORITY/TRANSPARENCY register. Plane(s) defined by channel #1 can be set in front or behind the plane defined by channel#2. When two planes are defined by Channel#1, the priority can be programmed.

The cursor plane has always the first priority when it is validated. The TV plane is always on the last position.

The backdrop plane is the TV plane if implemented externally. If there is no provision for that, a black level will be generated. The SI bit being zeroed, allows for forcing to black level if there is no hardware installed externally whatever the result of the overlay controller can be.

OVERLAY PRIORITY CONTROLLER

Three bits (P1 to P3) of the Priority/Overlay register are used to define the position of the various planes on the Z axis. As only 3 planes are involved, 6 different possibilities are reserved:

From front to back, the sequence can be:

P1	P2	P3	
0	0	0	CUR, FG1, FG2, BG, TV
0	0	1	CUR, FG2, FG1, BG, TV
0	1	0	CUR, FG1, BG, FG2, TV
0	1	1	CUR, FG2, BG, FG1, TV
1	0	0	CUR, BG, FG1, FG2, TV
1	0	1	CUR, BG, FG2, FG1, TV

Notice that FG2 is only available if channel#1 is in mode 2 (CLUT4+4) which means that bit P3 determining the priority between FG1 and FG2 is active only in that case. In the same way, if the mode C is entered, only the BG plane is available so P1 to P3 are don't care.

The following diagrams display some of the possible modes.

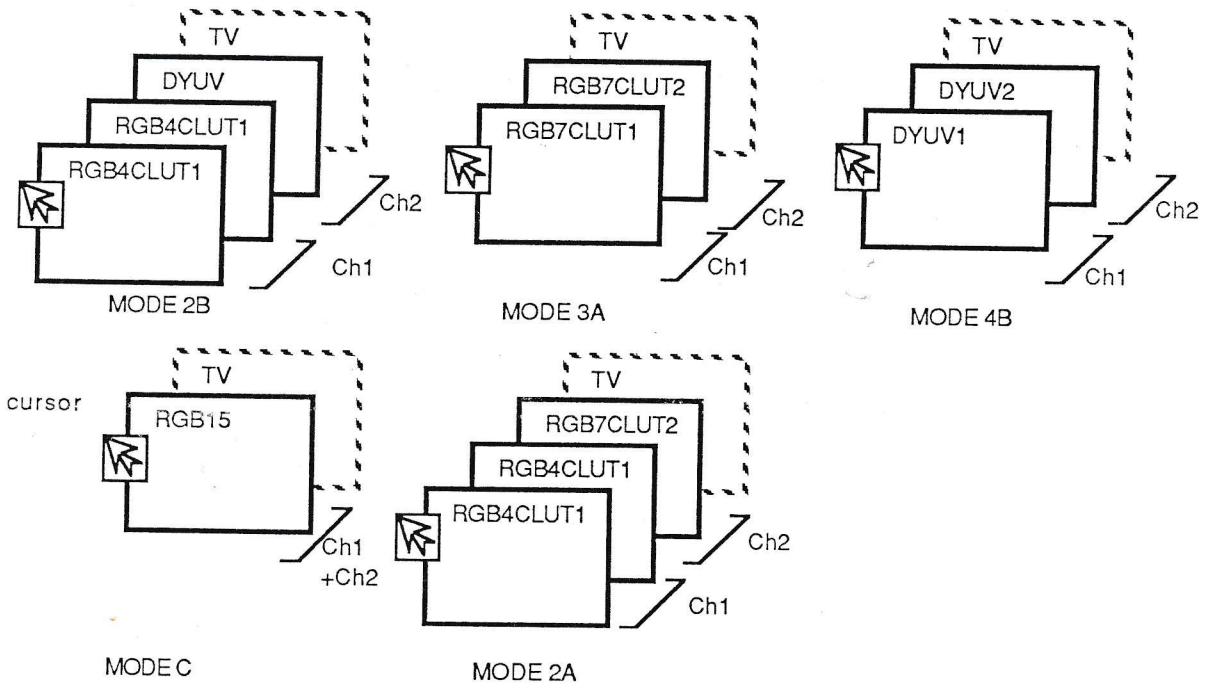


Figure 5: Modes

TRANSPARENCY CONTROLLER

The transparency of each plane can be controlled by various methods to allow for display of the planes with a lower priority.

Using the Overlay controller it is also possible to devalidate a plane so that it will not be displayed

whatever the transparency or the region bits are.

For each available image plane, the control of the transparency can be made using several mechanisms:

---> A comparison between the actual value of the output code (direct RGB or CLUT) and a transparent color register is made. Each bit comparison can be devalidated by a mask register. It is so possible to generate a transparency control information by matching on one given color, on a series of colors or even on a given bit.

The COLOR KEY CONTROLLER does that task.

---> The most significant bit of the incoming word (mode RGB 555).

---> The region bits generated from the REGION controller can be used.

If transparency control is applied to the last plane when visible and if the SI bit is set, the Video synthesizer controls an output pin to execute overlay on a TV image (when the system includes the necessary hardware externally).

For each plane (Cursor, FG1, FG2, BG) a series of bits of the Priority/Overlay register are defining the source of information controlling the transparency.

The following possibilities are available.

For FG1 the T10, T11, T12 T13 bits define:

T13	T12	T11	T10	
0	0	0	0	The plane is all transparent (disable)
0	0	0	1	Color key = 1 -> pixel is transparent
0	0	1	0	Transparent bit = 1 -> pixel is transparent
0	0	1	1	Region#1 bit =1 -> pixel is transparent
0	1	0	0	Region#2 bit =1 -> pixel is transparent
0	1	0	1	Region#1 bit or Color key = 1 -> pixel is transparent
0	1	1	0	Region#2 bit or Color key= 1 -> pixel is transparent
0	1	1	1	N.U.
1	0	0	0	The plane has no transparent area
1	0	0	1	Color key = 0 -> pixel is transparent
1	0	1	0	Transparent bit = 0 -> pixel is transparent
1	0	1	1	Region#1 bit =0 -> pixel is transparent
1	1	0	0	Region#2 bit =0 -> pixel is transparent
1	1	0	1	Region#1 bit or Color key = 0 -> pixel is transparent
1	1	1	0	Region#2 bit or Color key = 0 -> pixel is transparent
1	1	1	1	N.U.

The bits T20, T21,T22 and T23 play the same role for transparency control of FG2 and the bits T30, T31, T32 and T33 are used for BG plane.
Notice that the Txx bits are don't care when the plane is not active, depending on the mode.
The CUR bit of the Priority/Overlay register enable the cursor on top of the other planes.

CUR = 0 -> the cursor is disable (transparent)
CUR = 1 -> the cursor is enable

The Priority/Overlay register has the following layout:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P3	P2	P1	T33	T32	T31	T30	T23	T22	T21	T20	T13	T12	T11	T10	CUR

COLOR KEY CONTROL

For each of the 3 possible planes (FG1, FG2 and BG) , the transparency can be controlled by a comparison of the 18-bit output information (before multiplexors) to the content of a transparent color register. In order to allow for comparisons on a set of colors, an 18-bit MASK register can be applied to disable bits before comparison.

Three comparators are available for FG1, FG2 and BG planes. The comparators are not activated if the DYUV converter is selected on the relevent channel. The result of the comparison can then be used by the overlay controller.

The layout of the TRANSPARENT COLOR and MASK registers is:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R5	R4	R3	R2	R1	R0	X	X	G5	G4	G3	G2	G1	G0	X	X	B5	B4	B3	B2	B1	B0	X	X

Each bit is compared to the equivalent bit coming out of the decoder chain, the result is then ORed with the content of bit in the MASK register. The results are then ANDed to give one bit which equals one if all of the selected bits in the MASK corresponds to an equality between the output and the transparent color register. The MASK registers have the same layout than the Transparent color registers.

REGION CONTROLLER

The transparency of the various planes can also be controlled by using a coding mechanism defining transition points during each video line.

During the horizontal retrace period, the Video Synthesizer can store information made of series of 24-bit words defining the horizontal position of transitions or changes to be made in the mode of display. The DCA mechanism of the display controller can reload for each line (if necessary) up to 8 blocks (8 words) which enable complicated transition shapes. The display controllers can be used to generate one or two region control bits which can be used by the overlay controller.

Each 24-bit word is used to code the horizontal position (number of pixels in double resolution mode from the left blanking) (10bits RL0-9), which operation is selected and various extra information for local weight control.

At the beginning of each line, the first word is compared to the pixel counter. When an equality is found, the operation is performed and the comparison is performed on the second word and so on uptill the last word or the end of the line. Because two region bits can be generated simultaneously, two comparators are implemented. In this case, region registers 1 to 4 are used for region#1 and registers 5 to 8 are used for region#2.

By changing the content of the DCA, the user can generate a lot of complex contour shapes. During vertical retrace period, the software can update the content of the DCA in order to generate moving transitions between images. Any wipe effect can thus be generated. The shape, displacement, transition duration are fully under software control.

The region bits can be allocated to the control of each of the 3 possible planes (cursor plane being excluded) and can be used in combination with an eventual color keying mechanism applied to that plane.

Each of the Region registers (8 times 24 bits) is composed by four bits describing an action to be performed, by 10 bits (RL=Region Location) defining where the operation must occur during the line and 7 optional bits used as parameter for certain actions.

The various possible actions are:

CH3	CH2	CH1	CH0		
0	0	0	0	END of changes for the line	
1	0	0	0	RESET a Region bit	parameter is Region bit number (PA6)
1	0	0	1	SET a Region bit	parameter is Region bit number (PA6)
0	1	0	0	Change the weight of Channel#1	parameter is the new weight (PA5-0)
0	1	1	0	Change the weight of Channel#2	" " "
1	1	0	0	Reset region bit and change weight Channel#1	
1	1	1	0	Reset region bit and change weight Channel#2	
1	1	0	1	Set region bit and change weight Channel#1	
1	1	1	1	Set region bit and change weight Channel#2	

The format of the Region registers is:

23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
CH3-CH2-CH1-CH0 X X X PA6-PA5-PA4-PA3-PA2-PA1-PA0 RL9-RL8-RL7-RL6-RL5-RL4-RL3-RL2-RL1 RL0

The region registers can be allocated each to the definition of two regions as far as the number of transitions during one line does not exceed a total of 8. Notice that the region registers are active for every lines unless they are modify with a NOP (opcode 0000) or if RL bits set an address greater than the possible pixel count. For example, region#1 can be a rectangle used to define transparent area in Foreground#1 in DYUV mode (it "costs" two registers), when at the same time two other registers can define a circle for transparency of Background plane on TV image.

The following diagram displays the mechanism:

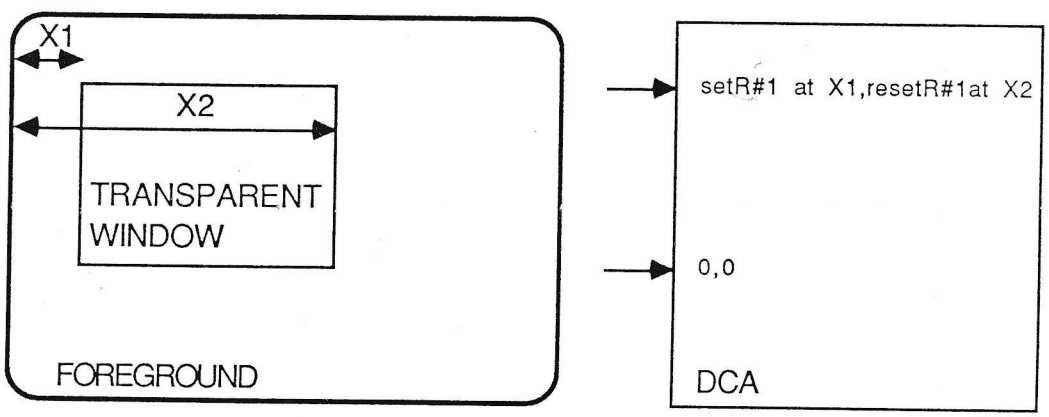


Figure 6 : Region mechanism

In this example, the rectangular window uses Region #1. It is declared in the DCA at the line before the first line where the window must occur. Then the region registers will be valid for the following lines.
Clearing the registers will end the window at the given line. Only writing 4 long words in DCA in necessary to install or move the window.

FADING / DISSOLVE CONTROLLER

The reference value of the Red, Green and Blue Digital to Analog converters of each channel can be controlled independently using the WEIGHT1 and WEIGHT2 Registers. These 6-bit registers are loaded with a coefficient acting directly as a multiplying factor on the actual R, G and B values before the addition of the two channels. An all-zeros value will give no contribution for the channel (no current), a all-ones value give a contribution equal to the nominal value of the color outputs.

The WEIGHT registers have the following layout:

byte#3								byte#4								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
x	x	x	x	x	x	x	x	w5	w4	w3	w2	w1	w0	x	x	(LSBs are don't care)

These registers are reloadable during the vertical or the horizontal retraces or via the region mechanism.

The Wipes and the Weight control commands can so be used concurrently during a line to allow for reload of the weight registers in between contour runs providing so the possibility to change the fading factor during the same video line.

CURSOR CONTROL

A cursor shape can be superimposed on top of the previously described composit image.

The position of the cursor is defined by the content of X and Y coordinate registers, the shape is defined by an on-chip 16 x 16 bits RAM area and the color is defined by the cursor color register (direct RGB 4bits).
Eight current sources controlled by the cursor generator contribute then to the RGB output instead of the normal D/A converters.

All cursor control registers are accessible using the reload mechanism during vertical or horizontal retrace periods.

The CURSOR X register defines the number of pixels of double resolution between the left border of the screen (BLANK signal) and the leftmost pixel of the square surrounding the cursor shape. The CURSOR Y register defines the number of lines between the upper part of the screen (Vertical Blank is regenerated internally) and the upper part of the square surrounding the cursor shape. The 0,0 coordinates correspond to the upper left corner of the full screen whatever display mode is used (with or without borders).

The CURSOR PATTERN register is a collection of bit to be set to allow the relevent pixel in the cursor shape to be active (non transparent). The pixel will take the color defined in the CURSOR COLOR register if the bit is 1 and will be transparent if the bit is zero.

The CURSOR COLOR register defines 1 color out of 16 with the following meaning:

c3	c2	c1	c0		
0	0	0	0	= Black	R, G and B levels to be fixed
0	0	0	1	= Red	"
0	0	1	0	= Green	"
0	0	1	1	= Yellow	"
0	1	0	0	= Blue	"
0	1	0	1	= Magenta	"
0	1	1	0	= Cyan	"
0	1	1	1	= White	"
1	0	0	0	= Grey	"
1	0	0	1	= Bright Red	"
1	0	1	0	= Bright Green	"
1	0	1	1	= Bright Yellow	"
1	1	0	0	= Bright Blue	"
1	1	0	1	= Bright Magenta	"
1	1	1	0	= Bright Cyan	"
1	1	1	1	= Bright White	"

All zeros in the CURSOR PATTERN registers or a zero in the CUR bit of the Priority register will erase the cursor from the screen.
The MSBs of the CURSOR PATTERN registers define the leftmost bits of the pattern. The low register addresses are allocated to the first lines of the pattern.

The CURSOR COLOR register has the following layout:

	byte#3								byte#4								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
x	x	x	x	x	x	x	x	c3	c2	c1	c0	x	x	x	x	(LSBs are don't care)

The Cursor Mode bit (CU2) defines if the cursor replaces the overlayed pixels or if its contribution is added to the overlayed pixels. If CU2 bit equals 0 then the inputs of the two multiplexors are zeroed during cursor active pixels. They are not zeroed if CU2=1.

CU2 bit is situated in the MODE register.

CONTROL REGISTERS

The Video Synthesizer has several control registers and a Color Look-Up Table which are loadable by the Display Processor during vertical retrace period (for image initialisation) or during the horizontal retrace period. Up to 16 sequences of 4 Bytes each can be accepted during one horizontal retrace period.

Each Channel can be used to control the on-chip registers. When by mistake, the two ports gives conflicting information i.e. addressing the same register at the same time, priority is given to Channel# 1.

REGISTER MAP

Address Hex	Register name
0 to 3F	CLUT 0 to 63 (18 bits) (out of 256 entries, the MSBs of address are in the MODE register)
40	MODE (16 bits)
41	PLANE PRIORITY/OVERLAY (16 bits)
42 to 44	TRANSPARENT COLOR REGISTER #1, # 2, #3 (18 bits)
45 to 47	MASK COLOR REGISTER # 1, # 2 ,#3 (18 bits)
48 and 49	DYUV ABSOLUTE START VALUE # 1, # 2, (24 bits)
4A and 4B	WEIGHT # 1, # 2 (6 bits)
50 to 58	REGION Registers 1 to 8 (24 bits)
60	CURSOR X (10 bits)
61	CURSOR Y (10 bits)
62	CURSOR COLOR (4 bits)
70 to 7F	CURSOR PATTERN 1 to 16 (16 bits)

As the Video Controller fetches automatically its data from memory (Dynamic and Image Control Areas), the CPU or the Coprocessor can considere the Video Synthesizer as a set of long words to load with an address (7 bits) + a 16 or a 24 bit data information .

When the application uses sub-screens, the Video Synthesizer can be dynamically reloaded in the middle of the screen to change for another mode.

Some examples of the control sequences used to reload various resources of the Video Synthesizer:

RELOAD CLUT ENTRIES

```
X  C  A5 A4 A3 A2 A1 A0   R5 R4 R3 R2 R1 R0 X  X  G5  G4  G3 G2 G1 G0  X  X  B5 B4 B3 B2 B1 B0  X  X
<----------------------><---------------------><---------------------><---------------------->
        BYTE 1                  BYTE2                 BYTE3                 BYTE4
```

C=0, A0-A5 = Address of CLUT entry, R5-R0 = RED Value, G5-G0 = Green Value, B5-B0 = Blue Value

The MSB of the address (A6 and A7) must be loaded in advance in the MODE register.

RELOAD CONTROL REGISTERS

```
X   C   A5 A4 A3 A2 A1 A0   D23  - - opt  - -   D16  D15 - - - - - - - - - - - - - - - - - - - - - - - - - - - - D0
<----------------------><---------------------><---------------------><---------------------->
        BYTE 1                  BYTE2                 BYTE3                 BYTE4
```

C=1, the bits C,A5-A0 gives the address of one of the control registers

RELOAD of Absolute DYUV start value for Channel#1

X 1 0 0 1 0 0 0 Y7- - - - - - - - - - - - Y0 U7 - - - - - - - - - - - - U0 V7 - - - - - - - - - - - - V)

BYTE 1 BYTE2 BYTE3 BYTE4

C=1, the bits C and A5-A0 gives the address of the register (48 Hexa in this example)

....ETC

INPUT INTERFACE

The Video Synthesizer has a programmable input interface to allow for various application configurations. The MODE1-2 pins are used to preset the interface mode. They work complementarily to the SW bits to control the input multiplexor.

Several modes are possible:

> 1 Video Controller (BPCRT2)
> 2 Video Controllers (BPCRT2)
> 1 Enhanced Video Controller (With multiplexed 8 bit bus)

The following diagram shows the input signal when used with 2 Video controllers:

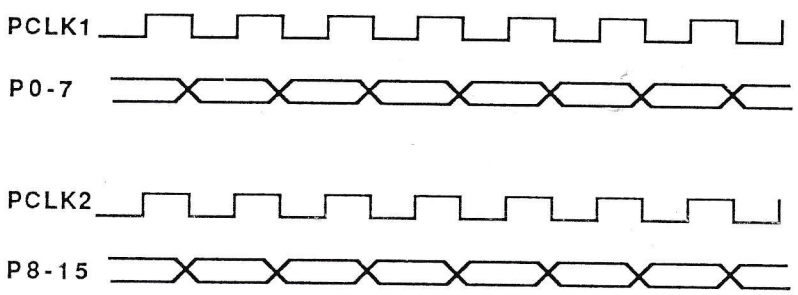


Figure 7: Typical input signals when driven by 2 BPCRT2s

The Mode register is also used to provide the possibility to swap the channels i.e. routing the P0-7 inputs to channel#2 and P8-15 to Channel# or to combine two nibbles coming from the two BPCRT2 in one byte for Channel#1 or #2.

The BPCRT2 can generate a border color in its reduced screen mode. The VSR must propagate this information and effectively generate the border while the BPCRT2 generates control information. If the WRP signal changes when BLANKN is still high, the last pixel transfered must be repeated to the output. When at the beginning of the next line, the BLANKN signal goes high, the VSR must generate again the same border color if the WRPN signal is still active.

ANALOG RGB INTERFACE

Two sets of three independent 6-bit wide D/A converters are necessary. The reference voltage is controlled via a a fourth D/A per channel. The outputs are current sources contributing to generate the R, G and B voltages, The outputs must be preferably standard video compatible (1V 75 ohms). Some reference voltage pin, analog ground pins are necessary to assure clean and stable signal generation.

For overlay on TV images, it could be interesting to have them three statable.

During the blanking period, the output must be set to level 16 because TV sets are using this level for alignment.

The following diagram shows the principle of the output stage.

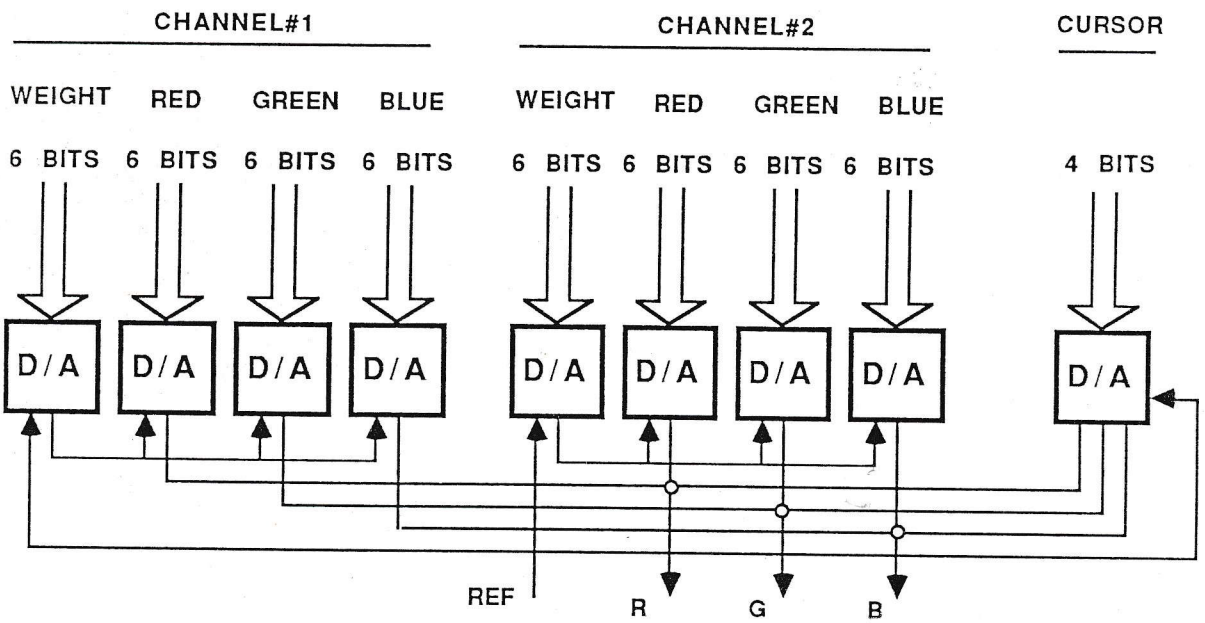


Figure 8: Digital to Analog output stage

SCREEN HARD-COPY INTERFACE

A serial output interface is provided for execution of paper copies of the composite image. The interface will serialize the value of each pixel (18 bits) for each channel.

The output mechanism allows to serialize the content of the two 18-bit flows available at the outputs of the multiplexers of Channel#1 and Channel#2.

During each video line, when the display reaches the X value of the Cursor, the two 18-bit values are loaded in shift registers prior to be serialized through the Hard Copy Data pin clocked by the Hard Copy Clock pin.

During one field, all the pixels belonging to a column of the screen will be output. By changing the CURSOR X value it is so possible to scan the full image or a part of it.

The two images being serialized out, it can be possible to print both of them or to combine them to get the final image.

The format of the output is:

R5 ..... R0, G5.....G0, B5.....B0 of Channel#1 then, R5....R0,G5....G0,B5.....B0 for Channel#2 Then the Weights of D/A #1 and D/A#2 must be serialized.

A 384x280 pixel image will be available in 384 x 20mS so 7,7 Seconds or 360 x 240 images will be available in 360 x 16.6mS so 6 seconds.

RESET MECHANISM

The Video Synthesizer uses a reset pin for start up condition. During reset condition, the Mode and the Priority/Overlay register are cleared disabling the chip.
The active low reset pin is to be connected to the RESOUT pin of the Video controller.

TEST

For test mode, the content of the output of the multiplexor must be available. The logic implemented for the Harcopy will be controlled from outside via HCYC as input for getting information out and for input of information for test of D/A converters.
One pin (TST) is used to put the chip in test mode so changing the function of several secondary pins.

POWER CONSUMPTION

The Video Synthesizer is supplied from a 5 Volts (±5%) power supply. The total power dissipation must not exceed 1 Watt.
The Reference voltage for the D/A converters is stabilized outside the chip.

BASIC APPLICATION

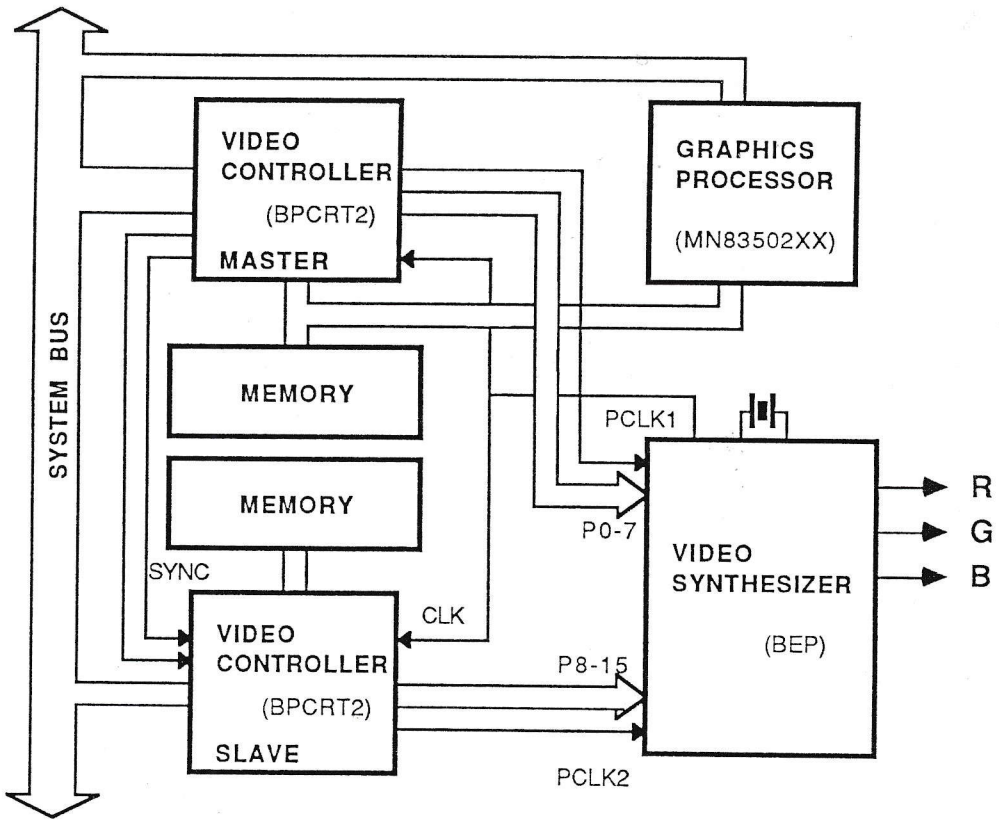


Figure 9 : Basic application