Copyright 1995, IS&T

SLANT/PRISM Convertible Structured Color Processor MN5515

*Teruo Fumoto, Katsuhiro Kanamori, Osamu Yamada, Hideto Motomura, Hiroaki Kotera and Makoto Inoue**
Matsushita Research Institute Tokyo, Inc., Japan
**Matsushita Electronics Corporation, Japan*

Overview

Color image reproduction in hard copy becomes more precious by recent progress of color input/output devices and image processing technologies. Especially in printing devices, its imaging characteristics in color reproduction deeply depend on printing devices and so various technologies of image processing are developed individually. Recent approach of device independent color reproduction or idea of color management systems needs common color processing method. We developed a new color processor LSI that has 3D-LUT and newly designed SLANT-PRISM interpolator suitable for those requirements and useful for various kinds of color conversion and color management system.

Processor Architecture

3D-LUT and Interpolator Method

Color conversion used in printing devices or color management system is thought as conversion from original color space to another color space. For these conversion, output color data (C',M',Y',K') are represented by functions of input color data (C,M,Y) as the following equations.

$$
\begin{aligned}
C' &= Func\_c(C,M,Y) \\
M' &= Func\_m(C,M,Y) \\
Y' &= Func\_y(C,M,Y) \\
K' &= Func\_k(C,M,Y)
\end{aligned}
\tag{1}
$$

Func_x (x = c,m,y,k) are defined individually according to the devices and usually complex to realize precious color reproduction. 3D-LUT for all input data set (C,M,Y) is perfect solution for those complex functions. But it needs much memory, so reduced 3D-LUT with 3D-interpolation is a practical approach to realize 3D-LUT method like figure 1.[1,2] The memory size necessary for practical 3D- LUT has been estimated and 8 or 16 division in each axis were proved to be acceptable.[3,4]

PRISM and SLANT-PRISM Interpolator

3D interpolation is useful to calculate the precise output from coarse LUT data. 3D-LUT and PRISM interpolator was firstly implemented onto single chip LSI (MN5511)[5], but has little problem for black generation in CMY input color space, then SLANT-PRISM interpolator been designed to improve the black generation and gray signal reproduction.[6]

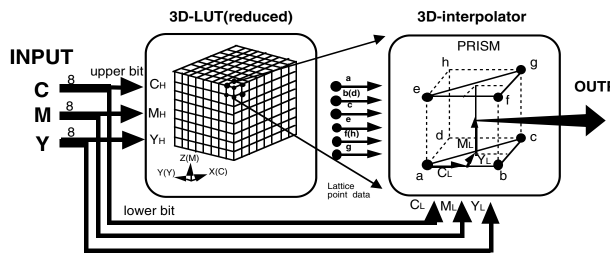

Figure 1. 3D-LUT and interpolation method

Figure 2(a) shows a basic idea of PRISM interpolator and Figure 2(b) shows a new SLANT-PRISM interpolator. The difference between these is in its interpolation unit structure.

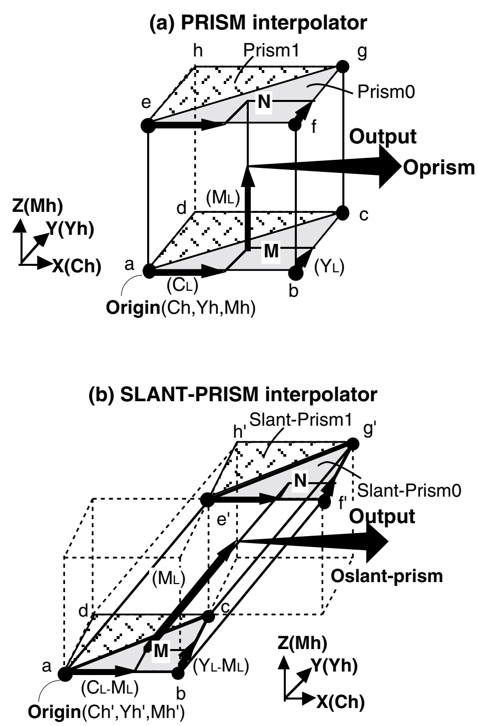

Figure 2. PRISM and SLANT-PRISM interpolator

The operation of PRISM and SLANT-PRISM interpolation are succeeded by the following process. Input color signal (C,Y,M) are divided into upper bit (Ch,Yh,Mh) and lower bit (Cl,Yl,Ml).

$$C = Ch+Cl,\ Y = Yh+Yl,\ M = Mh+Ml \quad (2)$$

Ch, Yh and Mh are correspond to X, Y and Z axis of 3D-LUT respectively as shown in Figure 2. In PRISM case an address of origin of unit cube is (Ch,Yh,Mh), but in case of SLANT-PRISM an address of origin (Ch',Yh',Mh') is decided by the following equations.

if (Ml>Cl) Ch' = Ch-1 else Ch' = Ch
if (Ml>Yl) Yh'= Yh-1 else Yh' = Yh (3)
Mh ' = Mh

The interpolated output data $\mathbf{O}_{\mathbf{prism}}$ or $\mathbf{O}_{\mathbf{slant\text{-}prism}}$ are calculated using 6 lattice point data according to the following equations, where [a], [b], [c], [d], [e], [f], [g], [h], [e'], [f'], [g'] and [h'] are the data of the lattice point a,b,c,d,e,f,g,h, e',f',g' and h' in 3D-LUT table respectively, [M] and [N] are the output data of M and N in lower and upper triangle plane respectively.

in case of PRISM interpolator:
If (Cl>Yl) then 6 lattice points are a,b,c,e,f and g.; Prism0
If (Cl≤Yl) then 6 lattice points are a,c,d,e,g and h.; Prism1

$$\mathbf{O}_{\mathbf{prism}} = [\mathbf{M}] + ([\mathbf{N}] - [\mathbf{M}])\,(Ml)\,/L \quad (4)$$

; (where L is length between lattice points)

in case of Prism0

$$[\mathbf{M}] = [a] + Cl([b] - [a])/L + Yl([c] - [b])\,/L$$
$$[\mathbf{N}] = [e] + Cl([f] - [e])/L + Yl([g] - [f])\,/L \quad (5)$$

in case of Prism1

$$[\mathbf{M}] = [a] + Cl([c] - [d])\,/L + Yl([d] - [a])/L$$
$$[\mathbf{N}] = [e] + Cl([g] - [h])\,/L + Yl([h] - [e])/L \quad (6)$$

in case of SLANT-PRISM interpolator:
If ((Cl-Ml)>(Yl-Ml)) then 6 lattice points are a,b,c,e,f and g.; Slant-Prism0
If ((Cl-Ml)≤(Yl-Ml)) then 6 lattice points are a,c,d,e,g and h.;Slant-Prism1
where, if (Cl-Ml)<0 then (Cl-Ml)=(Cl-Ml)+L
If (Yl-Ml)<0 then (Yl-Ml) = (Yl-Ml) + L

$$\mathbf{O}_{\mathbf{slant\text{-}prism}} = [\mathbf{M}] + ([\mathbf{N}] - [\mathbf{M}])\,(Ml)\,/L \quad (7)$$

in case of Slant-Prism0

$$[\mathbf{M}] = [a]+(Cl-Ml)([b]-[a])/L+(Yl-Ml)([c]-[b])/L$$
$$[\mathbf{N}] = [e']+(Cl-Ml)([f']-[e'])/L+(Yl-Ml)([g']-[f'])/L \quad (8)$$

in case of Slant-Prism1

$$[\mathbf{M}] = [a]+(Cl-Ml)([c]-[d])\,/L+(Yl-Ml)([d]-[a])\,/L$$
$$[\mathbf{N}] = [e']+(Cl-Ml)([g']-[h'])/L+(Yl-Ml)([h']-[e'])/L \quad (9)$$

Comparing (4)(5)(6) and (7)(8)(9), only small differences exist between PRISM and SLANT-PRISM interpolation formula. Those differences are addresses between efgh and e'f'g'h' and interpolation coefficients between Cl,Yl and (Cl-Ml),(Cl-Ml).

3D-LUT Configuration

To calculate above equation fast, parallel access to 6 lattice points of 3D-LUT is developed and pipeline calculation of PRISM or SLANT-PRISM interpolation is adopted. All memories of 3D-LUT are divided into 6 sub-memories in order to pick up 6 data of lattice points once at a time. Figure 3 shows a configuration of 3D-LUT and each lattice points belongs to which sub-memories. By these 3D-LUT configuration, any 6 data of lattice point of PRISM or SLANT-PRISM can be gotten from independent 6 sub-memories once at a time. Sub-memory assignment of M0,M1,M2 or M3,M4,M5 to 3D-LUT are cyclical along both X and Y direction, and M0,M1,M2 are assigned to the plane where Z is even and M3,M4,M5 are assigned to the plane where Z is odd. And interpolation formula of [4,5,6] or [7,8,9] are divided to plural pipeline stages to get high through put. By these configuration, this 3D-LUT with 3D-interpolator put an output data synchronously with input pixel clock after pipeline delay.

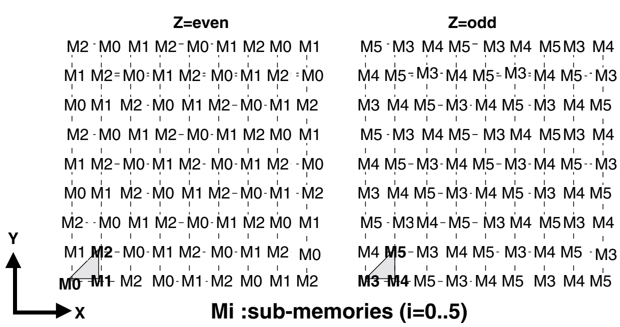

Figure 3. A configuration of 3D look-up table

LSI Design

A new interpolator structured with PRISM to/from SLANT-PRISM convertible and the parallel memories are implemented onto a single chip LSI. 3D-LUT is composed of internal RAM on chip and interpolation is done by hardware quickly and changeable PRISM to/from SLANT-PRISM by internal register. Input color data (C,M,Y) is converted by 3D-LUT and interpolation hardware, and processed output data are fed out synchronizing with input pixel clock after 10 pipeline delays.

LSI Configuration

Figure 4 shows a block diagram of single chip color processor LSI (MN5515). Each units operate synchronizing with input pixel clock (GCLK).

Gin separates upper 3 bits (Ch,Mh,Yh) and lower 5 bits (Cl,Ml,Yl) from input (C,M,Y) signal with 8 bit each. C,M,Y signal must be supplied with synchronous clock (GCLK).

AdrGen generates 3D-LUT addresses of 6 sub-memories of M0,M1,M2,M3,M4,M5 for 6 lattice points.

Copyright 1995, IS&T

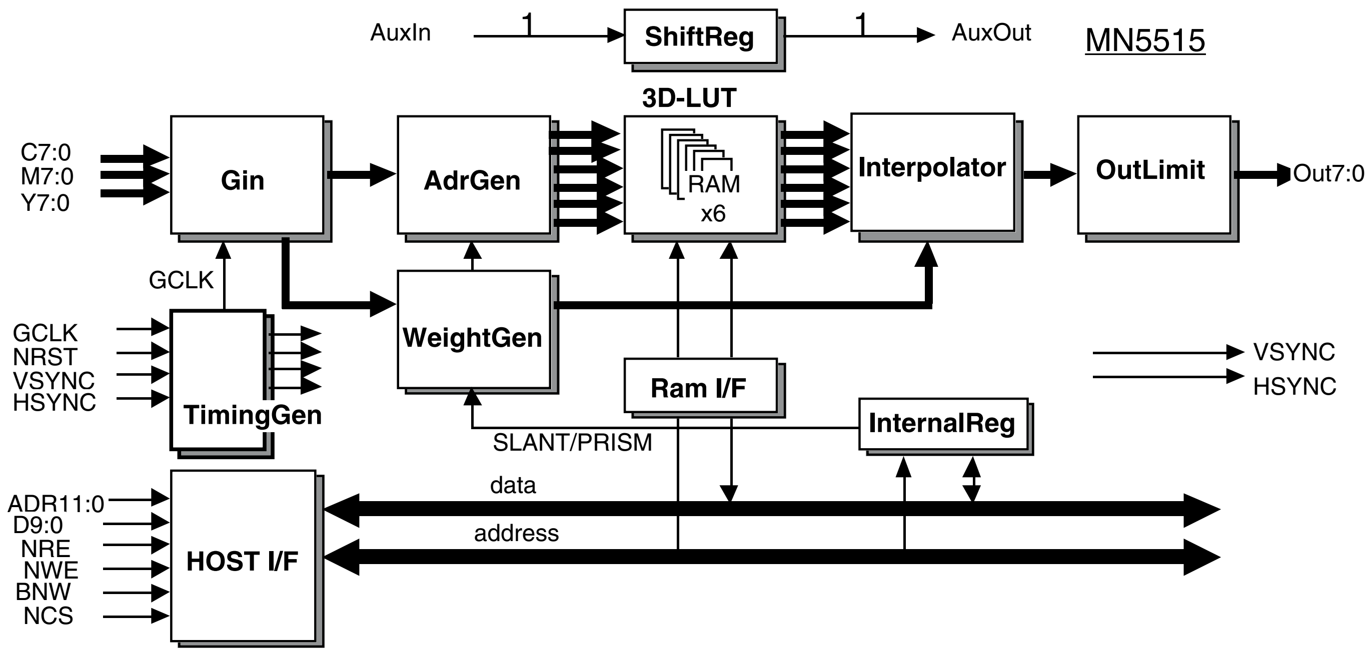

Figure 4. Block diagram of LSI (MN5515)

Each 6 addresses are generated separately and in parallel, so 6 independent address generators are equipped in this unit. These address generators generate respective addresses according to the interpolation modes of PRISM or SLANT-PRISM, and convert from three dimensional addresses (X,Y,Z) of 3D-LUT to linear addresses (Ai) of sub-memory of Mj (j=0,1..,5).

$$\begin{aligned} j &= (X+Y)\%3 + (Z\%2)*3 \\ Ai &= Kxy*(Z/2) + Kx*Y + (X/3) \end{aligned} \tag{9}$$

where Kx=3 and Kxy=Kx*9 in MN5515.

This linear address (Ai) are used to addresses of 6 sub-memories of 3D-LUT (M0..M5) in parallel and 6 lattice points data are picked up once at a time. Output data from 3D-LUT (M0..M5) are arranged to the lattice points of a,b,c,e,f,g or a,c,d,e,g,h under cyclic phenomena in figure 3.

WeightGen calculates three interpolation coefficient for PRISM or SLANT-PRISM. And these coefficient are similar to each other as described above, so those coefficient are changed according to the interpolation mode in this unit.

Interpolator works according to PRISM or SLANT-PRISM formula (4)..(9) where only interpolation coefficients are different. These coefficients are already settled by WeightGen unit, and 6 lattice points data are picked up using addresses from AdrGen according to PRISM or SLANT-PRISM respectively, so this interpolation unit operates to PRISM and SLANT-PRISM in common. This units calculates the difference between lattice points data according to X and Y axis at first, and weighted sums with each coefficient at both bottom and top triangle in XY plane next, and finally interpolation according to Z axis in pipeline process.

OutLimit clips data over 255 and below 0, then output data is limited between 0 and 255 (8 bit).

InternalReg sets interpolation mode, width of 3D-LUT to signed 8 or 9 or 10 bits, offset data of 3D-LUT and software reset.

HOST I/F supports asynchronous CPU bus timing. The data of 3D-LUT and all internal register are written from host CPU through this unit.

ShiftReg supplies the pipeline delay to auxiliary input equal to color conversion (10 GCLK) and output auxiliary signal.

Table 1 shows specification summary of MN5515. This LSI operates from DC to 16 MHz with 24 bits color and generates one output with 8 bits at once. If application needs 24 bits color at once, three LSIs are needed.

Table 1. Specification Summary of MN5515

Maximum speed (MHz)	16MHz
input data width	8bit
output data width	8bit
LUTsize (lattice points)	9×9×9=729points
LUT data width	1 0bit
Host interface width	8bit/16bit
pipeline delay	10 clock
interface logic	CMOS
power supply voltage	5V
power consumption	500mW
package	100QFP
3-D Interpolation	Prism/Slant

Remarkable Features

The first remarkable feature of this LSI is that the feature of this processor is decided by the data of the internal 3D-LUT. Internal 3D-LUT (composed by SRAM) are easy to use for hardware system designer, and 6 independent address generators for 6 sub-memories for 3D-LUT in common LSI chip are also easy to design LSI. The access speed to internal 3D-LUT can be extremely higher than that of external RAM.

The second remarkable feature is the selectivity of interpolation mode. PRISM interpolation is suitable for YCC color space (one luminous and two chromatic), and SLANT-PRISM interpolation is suitable for primary color space (RGB or CMY). In SLANT-PRISM interpolation mode, slanted Z axis is equal to gray axis (R=G=B or C=M=Y), so processed output signal when input is gray signal is not affected by the lattice points of color signal when primary color space (RGB or CMY) is used for input color space. And oppositely in PRISM interpolation mode, Z axis is equal to Y(luminous) signal and monochrome signal is not affected by lattice points of color signal.

Applications

Black Generation

As the data of four color printer, black signal (K) is used and generated from primary color signal (CMY or RGB). Except for conventional black generation method like Full Black or UCR Black, new black generation method based on chromaticity are proposed.[7] SLANT-PRISM interpolator has an advantage for full black, UCR black generation and above new method as compared with PRISM interpolator, because interpolation error is almost zero in SLANT-PRISM. This feature is mainly due to the fact that SLANT-PRISM region is faced to the differential discontinuity plane (C=M, M=Y and Y=C plane) of minimum calculation used for black generation.[6]

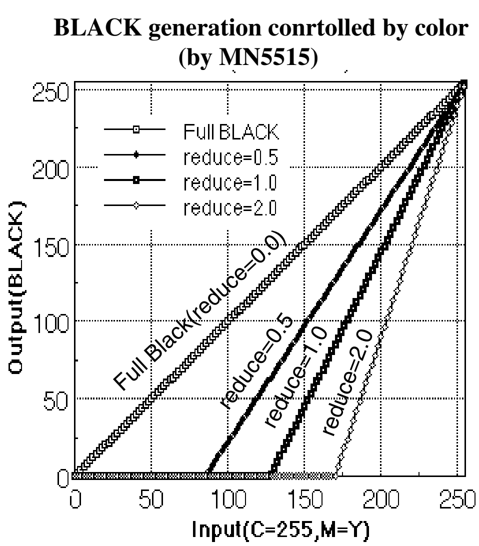

Figure 5. Black generation reduced by ratio of color/gray

Figure 5 shows an example of black signal generation adjusted by color/gray ratio by this color processor algorithm, where we assumed;

color = max(C,M,Y)-min(C,M,Y)
black = min(C,M,Y) (10)
controlled Black = black*(1- reduce*color/black)

In this case color/black value is used to control black generation based on chromaticity. The proposed color processor is easy to implement this feature and especially SLANT-PRISM interpolator is much better for these applications.

Color Management System

A color management and its standardization are the hot topics at present.[8] The best solution is coming now. The new processor will make it possible to convert the extremely large image data in very short time.

Secondary Color Adjustment

A secondary color (Rs,Gs,Bs,Cs,Ms,Ys) control of 6 color components is easy to consider for human perception, and sometimes this control method is used in professional color scanner or image processing system. Secondary color is detected or calculated comparing primary color as following equation.

$$\begin{aligned}
Cs &= C - \min(M,Y), && \text{where } Cs \geq 0 \\
Ms &= M - \min(C,Y), && \text{where } Ms \geq 0 \\
Ys &= Y - \min(C,M), && \text{where } Ys \geq 0 \\
Rs &= \max(M,Y) - C, && \text{where } Rs \geq 0 \\
Gs &= \max(C,Y) - M, && \text{where } Gs \geq 0 \\
Bs &= \max(C,M) - Y, && \text{where } Bs \geq 0 \\
Bk &= \min(C,M,Y)
\end{aligned} \tag{11}$$

And adjusted signal (C’,M’,Y’) is calculated by followings.

(C’,M’,Y’)=(C,M,Y)+**K**(Cs,Ms,Ys,Rs,Gs,Bs,Bk)
where **K** is adjusting coefficient matrix (3*7). (12)

These equation has a maximum or minimum calculation and those function are suitable for SLANT-PRISM interpolation because of the same reason described in black generation. Figure 6(b) shows secondary color detection characteristic by this color LSI of SLANT-PRISM algorithm when input color path are from 0 to 192 in Figure 6(a). This separation characteristic is just equal to that of calculation. Secondary color control can be easily realized by this single chip color processor LSI.

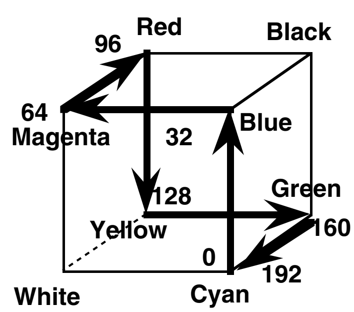

Figure 6(a). Input color path in Figure 6 (b)

Copyright 1995, IS&T

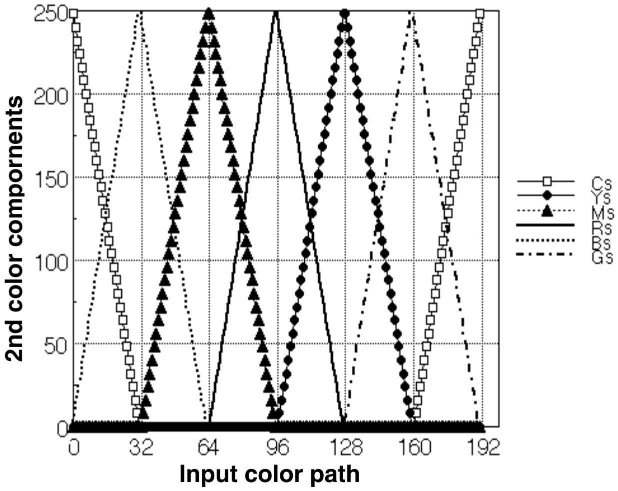

Figure 6 (b). Secondary color extraction by MN5515

Conclusion

Single chip color processor LSI with PRISM and SLANT-PRISM interpolator and internal 3D-LUT have been developed. 3D-LUT and interpolation method become more common for recent color management system. This color processor LSI will be fit to color masking, black generation, color adjusting and recent color management systems. PRISM interpolation is suitable for YCC color space and SLANT-PRISM interpolation is suitable for primary (C,M,Y or R,G,B) color space. PRISM and SLANT-PRISM interpolator can be changed to fit suitable application. We are planning to develop how to decide 3D-LUT data for various color applications including color management systems.

References

1. K. Kanamori, T. Fumoto and H. Kotera, "A color Transformation Algorithm using PRISM interpolation" *IS&T's 8th International Congress on Advances in NIP Technologies*, (1992).
2. K. Kanamori, H. Kotera et al, "Fast color processor with programmable interpolation by small memory (PRISM)" *Journal of Electric Imaging* Vol. **2** July (1993).
3. J. M. Kassen, W. E. Plouffe and S. I. Nin "A tetrahedral interpolation technique for color space conversion*", Device-Independent Color Imaging and Imaging Systems Integration*, *SPIE* vol. **1909**,(1993).
4. Raja Balasubramanian "Color Transformations for Printer Color Correction", *IS&T's 2nd Color Conference* (1994).
5. K. Kanamori, O. Yamada, H. Motomura, T. Fumoto, H. Kotera, M. Inoue, "Single Chip Color Processor MN5511 using 3D LUT and PRISM Interpolation" *IS&T's 9th International Congress on Advances in NIP Technologies*, (1993).
6. T. Fumoto, K. Kanamori, O. Yamada, H. Motomura and H. Kotera, "The Color Data Conversion Method Using 3-Dimensional LUT and Interpolation, *IS&T's 10th International Congress on Advanced NIP Technologies*, (1994).
7. H. Ogatsu, S. Kita and K. Murai, "A Flexible GCR based on CIEL*A*B*", *IS&T/SPIE Symposium on Electric Imaging Science and Technology* (1995).
8. Gerald M. Murch, "Cross-Platform Image Fidelity", *IS&T's 2nd Color Conference* (1994).