VIDEO PROCESSING FOR DLP™ DISPLAY SYSTEMS

Vishal Markandey, Todd Clatanoff, Greg Pettitt

Digital Video Products, Texas Instruments, Inc.
P.O.Box 655012, MS 41, Dallas, TX 75265

ABSTRACT

Texas Instruments' Digital Light Processing™ (DLP™) technology provides all-digital projection displays that offer superior picture quality in terms of resolution, brightness, contrast, and color fidelity. This paper provides an overview of the digital video processing solutions that have been developed by Texas Instruments for the all-digital display. The video processing solutions include: progressive scan conversion, digital video resampling, picture enhancements, color processing, and gamma processing. The real-time implementation of the digital video processing is also discussed, highlighting the use of the Scanline Video Processor (SVP) and the development of custom ASIC solutions.

INTRODUCTION

The Digital Micromirror Device™ (DMD™) is a binary spatial light modulator developed at Texas Instruments. The DMD consists of an array of movable micromirrors functionally mounted over a CMOS SRAM. Each mirror is independently controllable and is used to modulate reflected light, mapping a pixel of video data to a pixel on display. A DMD mirror is controlled by loading data into the memory cell located below the mirror. The data electrostatically controls the mirror's tilt angle in a binary fashion, where the mirror states are either +10 degrees (ON) or -10 degrees (OFF). Light reflected by the ON mirrors is then passed through a projection lens and onto a screen.

The DMD forms the heart of Digital Light Processing (DLP) all-digital display systems. DLP systems are being developed in various forms, suitable for applications such as conference room projectors, institutional projectors, home theater, standard television, high definition displays, and motion pictures. These systems are characterized by:

- All digital display: DLP display systems are completely digital in that, except for the A/D conversion at the front end, all data processing and display are digital.
- Progressive display: DMD displays are inherently progressive, displaying complete frames of video. For interlace inputs such as most current video inputs, this necessitates an interlace to progressive scan conversion. Progressive scan display also has the advantage of improving display quality by removing interlace artifacts such as flicker.
- Square pixels, fixed display resolution: DMD display resolution is fixed by the number of mirrors on the DMD. This, combined with the 1:1 aspect ratio of the pixels, requires resampling of various input video formats to fit onto the DMD array.
- Digital color creation: Spectral characteristics of the color filters and the lamp are coupled to digital color processing in the system.
- Digital display transfer characteristic: DMD displays exhibit a linear relationship between the gray scale value used to modulate the micromirrors and the corresponding light intensity. A "degamma" process is performed as part of the video processing, prior to DMD display, to compensate for video signal gamma and prepare the data for DMD display.

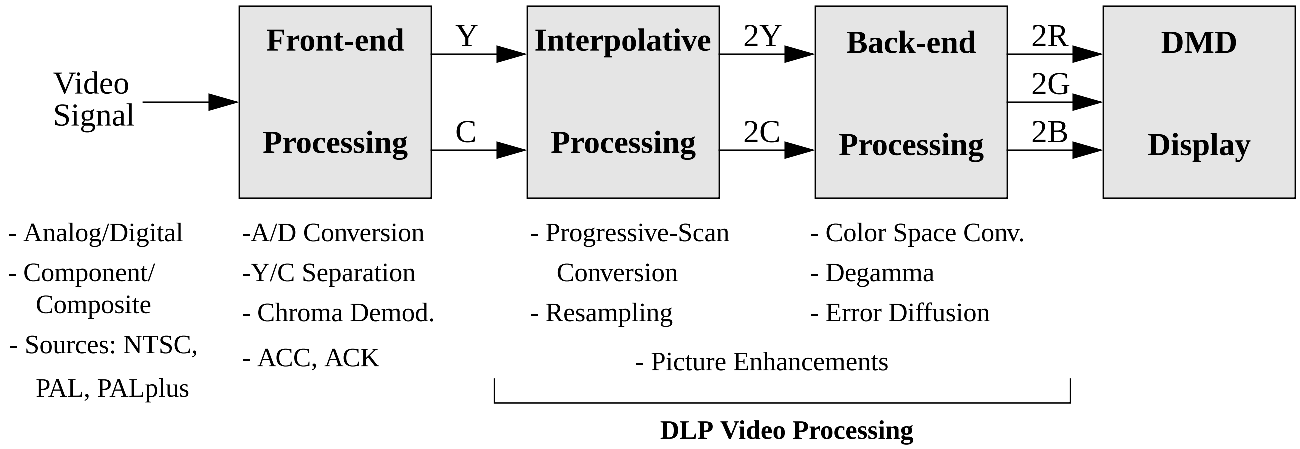
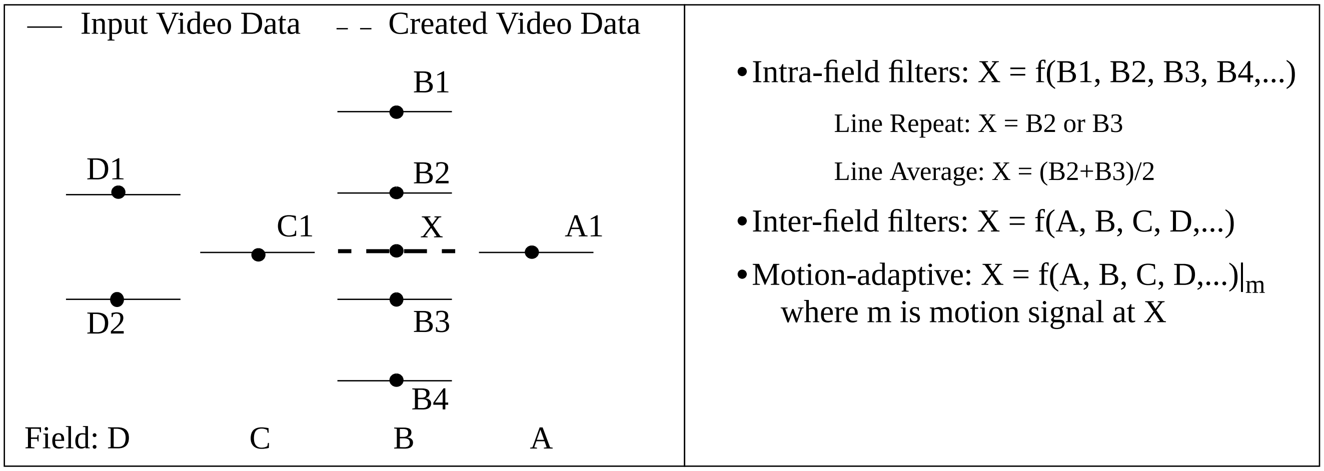


Figure 1: Video Processing For DLP Display Systems

The requirements described above are addressed in the Video Processing part of DLP display systems. Figure 1 shows the video processing functional block diagram of a typical DLP display system. The first block, labeled “front-end processing” performs signal decoding operations. The input signal, if composite, is subjected to Y/C separation and chroma demodulation. Additional operations include Automatic Chroma Control (ACC), and Automatic Color Killer (ACK). The signal is digitized - depending on the front end, the A/D conversion may occur either before or after Y/C separation. Digital Y/C data is output from this block and is subjected to “Interpolative Processing”, where the data undergoes interlace to progressive scan conversion, resampling, and picture enhancements. Picture enhancements include luminance and chrominance sharpening, and noise reduction. The enhanced, progressively scanned Y/C data is then passed through a color space conversion to obtain RGB data. This data is subjected to a degamma operation to remove the gamma imposed on the signal at transmission. During the degamma operation, error diffusion can be used as a means of subjectively improving the non-linear digital remapping process. The linearized, progressive RGB data is then reformatted into bit plan level data which is used to drive the DMD display using a Pulse Width Modulation (PWM) technique[1]. The following sections will describe interlace-to-progressive-scan conversion, resampling, picture enhancements, and degamma/error diffusion.

PROGRESSIVE-SCAN CONVERSION

The progressive nature of DMD display requires interlace data (such as NTSC and PAL video) to be converted to progressive form before it can be displayed on the DMD. Progressive-scan conversion is the process of creating new scan lines between existing lines, to provide double the vertical sampling rate of interlaced scanning. Progressive-scan conversion has the additional advantage of reducing interlace scanning artifacts such as interline flicker, raster line visibility, and field flicker. These artifacts are particularly objectionable for larger display systems where projection displays find their main application. Progressive-scan conversion is a well-studied field [1,2,3,4,5] with algorithms ranging from simple line repeat to complex variations of motion adaptive interpolation.



(a) **(b)**

Figure 2: Progressive-Scan Conversion

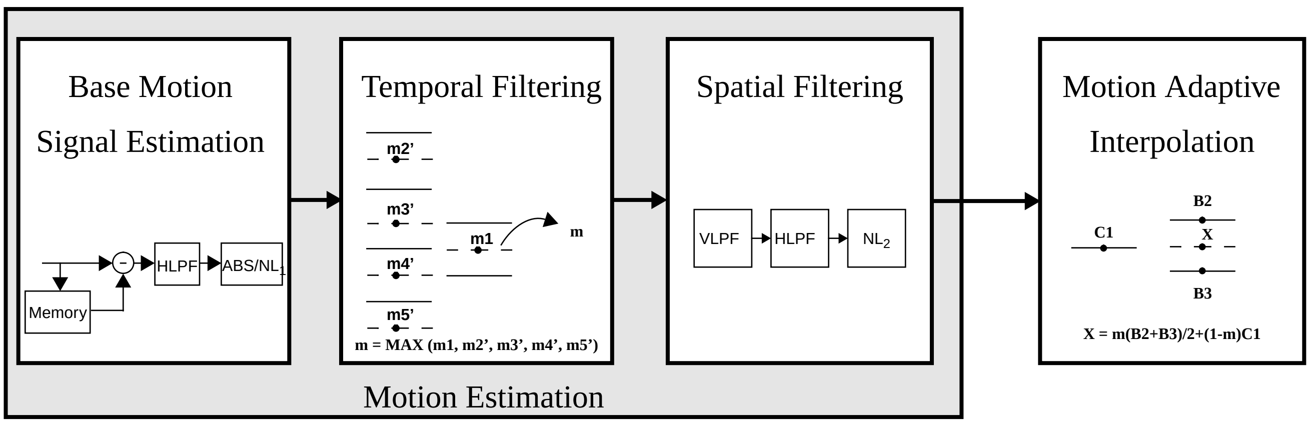


Figure 3: Motion Adaptive Progressive-Scan Conversion

The basic problem addressed in progressive-scan conversion is shown in Figure 2a. The solid lines show the input interlace signal and the dashed line shows the spatial location where the data is to be created by interlace to progressive-scan conversion. The data at X can be created by various proscan techniques listed in Figure 2b:

Intra-field filters use data from the same field - the filters can range from multi-tap FIR filters to simple line average or line repeat. The latter two are indicated separately in Figure 2b because of their common use. The major advantage of intra-field filters is that, unlike other progressive-scan techniques, they do not use multiple fields of data and hence do not require field data storage, translating into lower system implementation cost. The penalty paid is in terms of performance, intra-field filters being characterized by horizontal line flicker, still area flicker, and low resolution. In view of such performance issues, this class of progressive-scan conversion is primarily useful in low end applications. Various filters in this class were designed and tested for DLP display - it was concluded that considering the limited performance of the class as a whole and its low cost target applications, multi-tap filters offer little advantage over the cheaper line average/line repeat techniques.

The combination of data from multiple fields offers the potential of improving the performance of progressive-scan conversion by removing the artifacts noted above. A major concern in the design of such techniques is the sensitivity of the technique to the presence of motion or changes in the picture. The use of inter-field data in moving parts of the video can cause the appearance of “tearing” artifacts in the picture. On the other hand, the use of intra-field data in still parts of the video can cause artifacts such as loss of resolution, and flicker.

Inter-field filters that combine data from multiple fields in a pre-determined fashion are offered by some vendors. The combination of data from multiple fields can be realized in either a linear or non-linear fashion. Linear combinations are inherently not adaptive to motion and hence must be biased in favor of intra-field filtering to avoid “tearing” artifacts. This however leads to enhancing other artifacts such as loss of resolution and flicker. Non-linear combinations, such as median filters, and others, offer the potential of data adaptation and are an area of active research [4,5]. Various filters in this class were designed and tested for DMD display - it was concluded that since the filters are data adaptive rather than motion adaptive, they can display the artifacts mentioned above depending on data values.

Motion adaptive techniques that compute motion-weighted combination of inter and intra-field data offer the potential of overcoming the artifacts associated with other techniques noted above. Motion detection is a critical step in these techniques, because failure to detect motion can result in the use of inter-field data in moving parts of the video, where it can cause the appearance of “tearing” artifacts. On the other hand, oversensitive motion detection can cause the motion detector to be triggered by noise, resulting in intra-field data in still parts of the picture. This can lead to noticeable loss of resolution in the video. Thus, there is a need to balance the algorithm’s motion sensitivity with the ability to provide good resolution. We have developed motion detection techniques that explicitly addresses this issue and provides good picture quality in both moving and still picture areas.

Figure 3 shows a functional block diagram of motion adaptive progressive-scan conversion. A base motion signal is estimated for location **X** of Figure 2a, by performing temporal data differencing. This signal is subjected to a non-linear clipping function, and temporal and spatial filtering are applied to shape the motion signal. Temporal filtering produces the maximum motion value from among values in the current and past field. Following the temporal filter are spatial (vertical and horizontal) low-pass filters. These filters are used for smoothing and spreading the motion signal, such that the filtered signal is adequately expanded around moving edge boundaries. Insufficient spreading would result in noticeable “tearing” artifacts, caused from combining inter-field data in moving picture areas.

After completion of all filtering and non-linear operations, the motion signal is normalized to values ranging between 0 and 1. The normalized motion signal, m, is then used as a weighting factor between the inter- and intra-field components. For large motion values, interpolation is biased towards using intra-field components, which is required due to the variation between the successive video fields. In this case, use of inter-field data in the presence of motion would result in visible motion artifacts due to the lack of correlation between the current and previous field. However, for small motion values, a high degree of temporal correlation exists between adjacent video fields, which necessitates heavily weighting the inter-field component.

The general motion adaptive progressive-scan conversion technique described above forms the basis for a family of algorithms that we have developed. These algorithms cover a wide range of performance and associated cost. Each algorithm was developed for a specific base motion estimation technique - with reference to Figure 2a, motion estimation at **X** can be performed by computing a difference between field B and C (field differencing), B and D (frame differencing), or combination of A and C with B and D (multiple field differencing). For each such base motion estimation, the temporal/spatial filters, as well as the motion adaptive interpolation were optimized. Table 1 provides a listing of the algorithms with estimated ASIC gate counts that benchmark their complexity, external memory requirements, as well as comments on performance. Algorithms from this family are used in various DMD display applications representing various performance/cost trade-offs.

Algorithm	ASIC Size (gates)	Field Memory	Performance
Line Repeat	0	0	Poor: Flicker, Jaggies
Line Average	16k	0	Poor: Blurriness, Flicker
Intra-field FIR	46k	0	Poor-Fair: Blurriness, Flicker
M.A. Field Diff.	47.7k	1	Fair: High Res. Loss
M.A. Frame Diff.	86.9k	2	Good
M.A. Temporal	90.9k	3	Best

Table 1: Progressive-Scan Algorithms

DIGITAL RESAMPLING

Digital resampling (or scaling) is required for resizing video data to fit the DMD's pixel array, expanding letterbox video sources (various display feature modes), and maintaining correct aspect ratio for square pixel DMD display. Digital resampling can be used to maximize the active pixel area in a fixed-resolution optical configuration by appropriately resizing the video data to fit within the active optical window. Various display modes, such as “cinema”, “panorama”, etc., can be offered where the user can select a picture resizing option for letterbox source material. Also, digital input data may be available in non-square pixel form (e.g. CCIR601 component video with 720 active pixel per line). This data is digitally resampled to square pixel form for display on the square pixel DMD array.

A typical DLP display system uses digital resampling in both horizontal and vertical directions. The resampling filters are optimized for DMD display high spatial frequency display characteristics and also for the degree of computational complexity allowed in real-time applications. Table 2 shows an example case, for a DLP projection system based on DMD resolution of 800x600 pixels. Scaling requirements are shown for various input sources including NTSC, PAL, PALplus, and computer graphics. In this table, an

Source (F_S = 14.75 MHz)	Horz. Resample	Vert. Resample
NTSC	-	5:6
NTSC Letterbox	9:10	3:4
PAL	-	-
PAL 16:9	6:7	8:7
PALplus	6:7	6:7
VGA	4:5	4:5
SVGA	-	-

Table 2: Resampling Factors For DLP Projection Display

M:L resampling filter would imply converting M input samples into L output samples. In developing resampling filters for DLP display systems, a broad range of filter design algorithms was examined. The algorithms included nearest neighbor, linear interpolation, cubic splines, and custom designed FIR filters

[6, 7]. Algorithms were evaluated based on performance and implementation complexity. Performance evaluation include characterization of resampling artifacts such as aliasing and blurriness on DLP display. The result is a filter design methodology producing filters optimized for DLP display that can implemented within the constraints of DLP system architectures (the Video Processing Systems section describes system architecture implementation).

Vertical resampling filters are realized by convolving the input video lines with a discrete filter kernel. Filter kernel coefficients are optimized for performance and implementation cost, as noted above. Figure 4 shows the discrete convolution process used in 3:4 vertical resampling. In this figure, output lines o_0-$o_3$are created by convolving the input lines with a multi-tap filter kernel. This process results in output lines which are derived from the summation of weighted input lines, where the weighting corresponds to samples from the filter kernel. A similar approach is used for horizontal resampling.

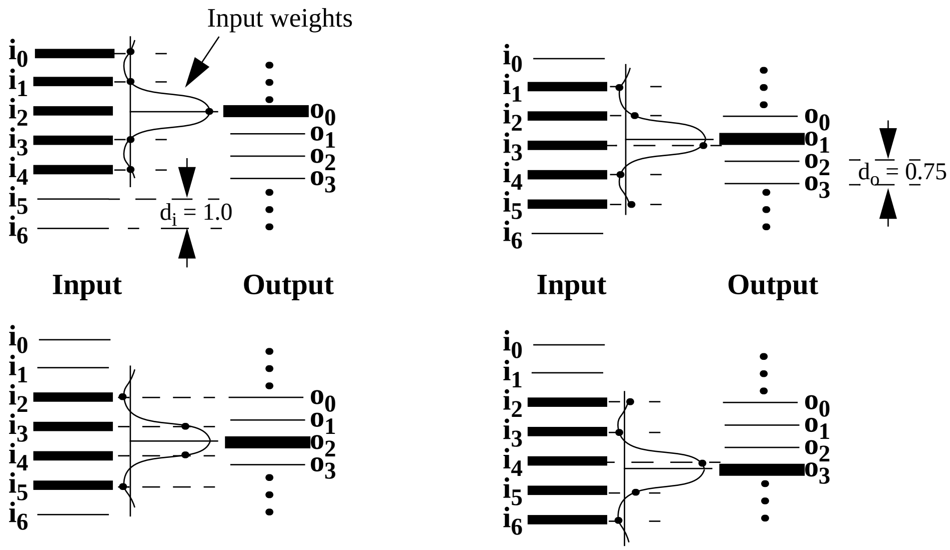


Figure 4: Discrete Convolution Process Used For 3:4 Vertical Resampling

Frequency responses for example cases (3:4 vertical and 9:10 horizontal resampling filters) are shown in Figure 5. The optimized filter frequency response is compared to the frequency response of both the ideal and bi-linear filters. It can seen that the optimal filter will result in less aliasing and blurriness in the scaled image as compared to the bi-linear filter.

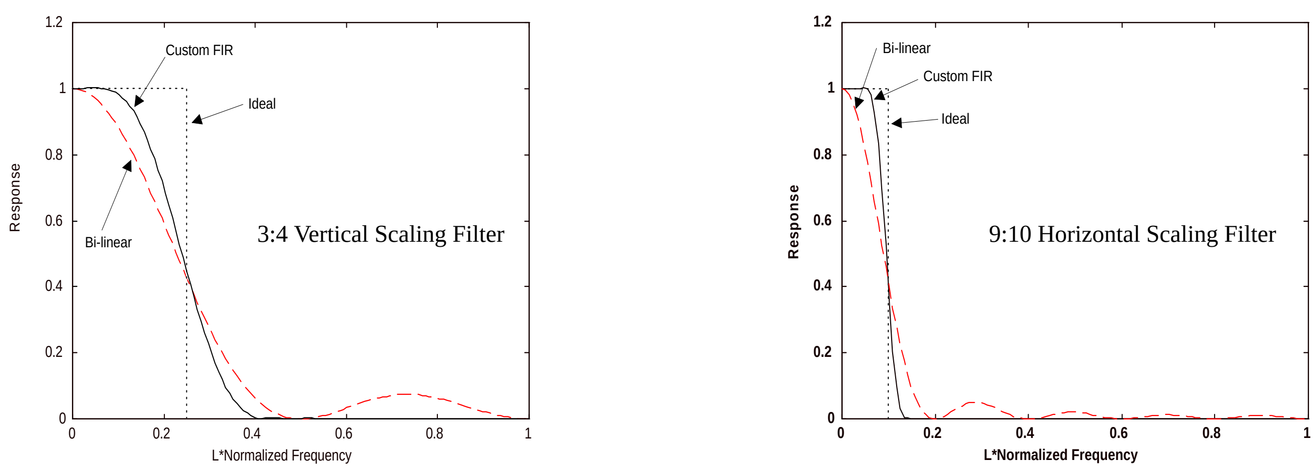


Figure 5: Frequency Response of Resampling Filters

VIDEO ENHANCEMENT

Enhancement algorithms are used to increase the subjective display quality. These algorithms include luminance and chrominance sharpening, noise reduction, dynamic range expansion, color correction, as well as the more traditional picture controls such as brightness, contrast, color saturation, hue, and color temperature selection. Luminance and chrominance sharpening algorithms are used to enhance mid-to-high image frequencies. Figure 6 shows an implementation of luminance sharpening. The sharpness filter

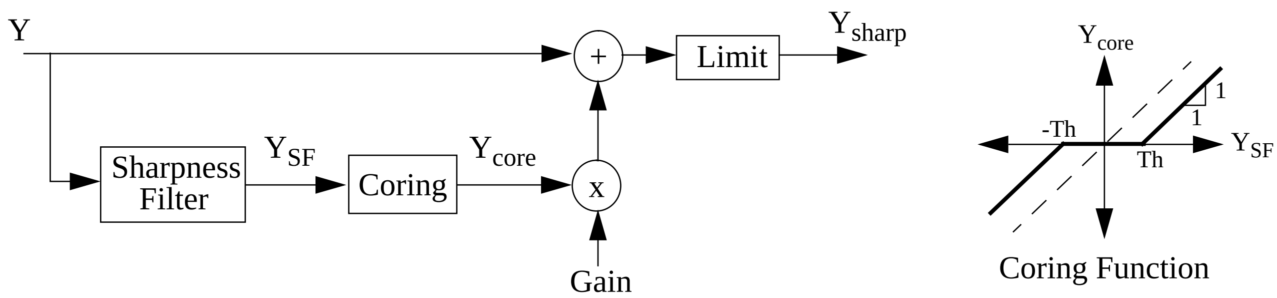


Figure 6: Luminance Enhancement Implementation

is realized using one or two-dimensional filtering, the filters being bandpass filters, highpass filters, or combinations thereof. Coring is applied to the filtered sharpness component, Y_{SF}, in order to reject signal variations attributed to noise. Figures 7 and 8 show the effect of luminance and chrominance sharpening for a 1-D linear edge transition. Notice that in Figure 7, the 1-D luminance signal is enhanced by 'peak-

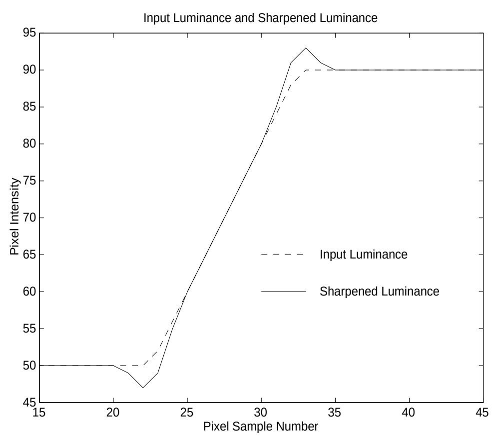


Figure 7: Luminance Enhancement

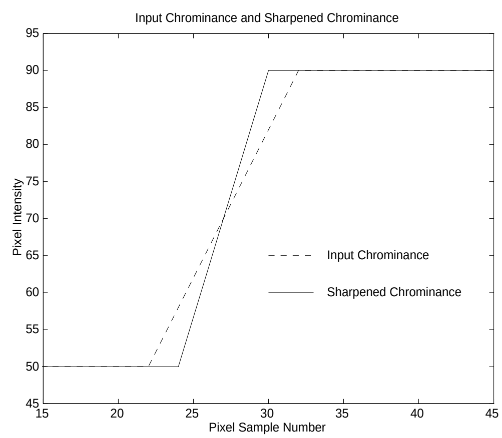


Figure 8: Chrominance Enhancement

ing' the signal at the edge transition points. Peaking the signal increases the apparent contrast of the edge, thereby creating the effect of a higher frequency transition. Luminance sharpening can also be implemented using 2-D filters, where enhancement takes place both horizontally and vertically.

Figure 8 shows an example of chrominance sharpening enhancement. In this case, chrominance data, which has much lower bandwidth than luminance, is enhanced by increasing the slope of the edge transition without introducing the ringing effects used in luminance sharpening. Use of ringing effects to enhance the chrominance signal would lead to undesired color aberrations around edges. The use of luminance and chrominance sharpening, as well as other enhancement/picture control algorithms, greatly increases the subjective quality of the processed video, leading to more visually pleasing video imagery.

COLOR SPACE CONVERSION

Color conversion provides the ability for video data conversion from luminance and color difference encoding to an RGB data format. The color conversion process utilizes a general 3x3 matrix multiplied by a 3x1 input vector producing the 3x1 RGB resultant vector. The general nature of the 3x3 matrix multiply used for color conversion allows for a wide range of video level mappings from the input luminance and color difference format. Through adjustment of the coefficient values, the input video levels can be mapped into the appropriate range of RGB values. Additionally, the 3x3 color conversion matrix can be merged with other matrix operators providing implementations of contrast, hue, color saturation picture controls. The expanded color conversion matrix has the following form

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \underbrace{\begin{bmatrix} CC1 & CC2 & CC3 \\ CC4 & CC5 & CC6 \\ CC7 & CC8 & CC9 \end{bmatrix}}_{\text{Color Correction}} \cdot \underbrace{\begin{bmatrix} CV1 & CV2 & CV3 \\ CV4 & CV5 & CV6 \\ CV7 & CV8 & CV9 \end{bmatrix}}_{\text{Color Conversion}} \cdot \underbrace{\begin{bmatrix} CT & 0 & 0 \\ 0 & SR & 0 \\ 0 & 0 & SB \end{bmatrix}}_{\text{Contrast and Color Saturation}} \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}}_{\text{Hue Control}} \cdot \begin{bmatrix} Y \\ R-Y \\ B-Y \end{bmatrix}$$

The color correction 3x3 matrix can be used for complete adaptation of the color gamut to the required color reproduction, or for selection of the white color temperature using only the diagonal elements. Following the matrix multiplication, the output RGB values must be rounded to the required bit resolution and limited to match the output dynamic range. The dynamic output range limitation is necessary due to the input luminance and color difference color space having the ability to express values that are not realizable RGB values.

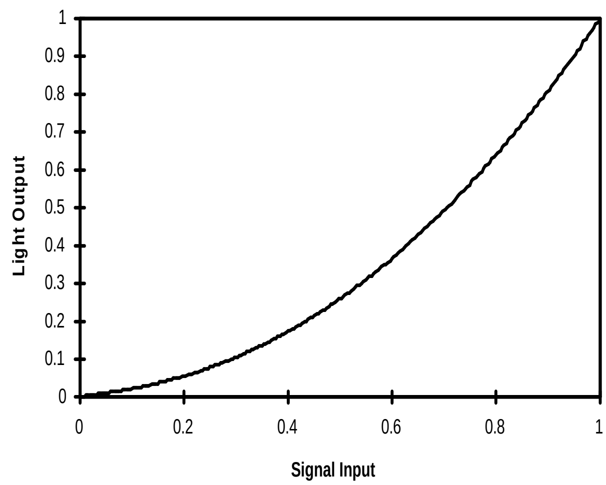
GAMMA PROCESSING

Video signals typically have gamma correction applied to them to compensate for the nonlinear signal-to-light characteristics of the CRT receiver. Gamma correction provides the additional benefit of increased signal-to-noise ratio of low level signals during analog transmission. Gamma correction, applies at the camera, maps linear light intensity to a non-linear voltage signal.

The transfer function of a CRT display system inverts the gamma correction resulting in a linear output response. DMD displays, unlike CRT displays, have a linear signal-to-light transfer function. The linear response of the DMD requires that gamma correction be removed from the input signal before display. The inverse gamma response, as known as “degamma” is realized in DLP display video processing. Degamma response is given by the following equation and is shown in Figure 9.

$$R = \begin{cases} R'_{709}/4.5 & R'_{709} \le 0.0812 \\ \left(\dfrac{R'_{709} + 0.099}{1.099}\right)^{1/0.45} & R'_{709} > 0.0812 \end{cases}$$

The degamma response is applied to the input signal using a RAM based lookup table (LUT), with individual LUTs used for red, green, and blue channels. The degamma response function is mapped into the RAM LUT using the desired number of output bits. The RAM based LUT structure provides additional data mapping benefits by performing brightness and contrast picture control adjustments as well as color temperature adjustment. The brightness control is accomplished through the addition of an offset



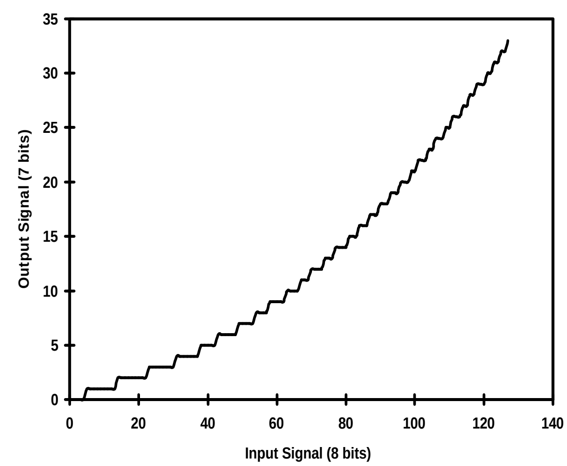


Figure 9: Degamma Response

Figure 10: 7-bit Degamma Response

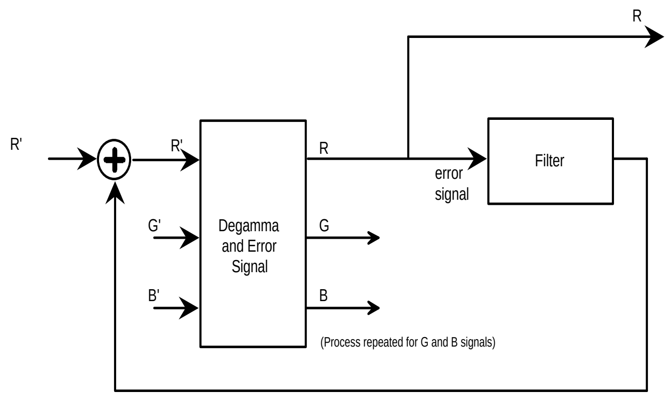
factor to the degamma function. The contrast picture control modifies the LUTs using a scale factor, and color temperature adjustment is implemented using individual scale factor adjustments for each of the R, G, B LUTs.

Additional, the flexibility of the LUTs can be used to perform non-linear mapping of the video. The mid-range values of the image can be stretched to provide higher detail contrast using S-curves mapping functions. The S-curve mapping function and degamma transfer function are combined to form a composite transfer function.

The precision of the LUT used in the degamma process can have great impact on the quality of the displayed image. As the output bit resolution is decreased, the low intensity portion of the degamma function shows increasing areas of poor quantization. This poor quantization results in the appearance of objectionable contours in the low intensity portions of the image. Figure 9 shows the degamma response curve, while Figure 10 shows the degamma curve using 7 bits of output resolution. In Figure 10, only the first 128 input code words are shown, clearly showing poor quantization at lower intensity levels. The contouring effect could be reduced through increased degamma output resolution, but such a solution can add a great deal of cost to the display product.

An alternate solution is the use of error diffusion. Error diffusion solutions seek to replace the low frequency contour areas of the image with a higher frequency mask signal. This higher frequency signal masks the contour effects to the observer, while maintaining the overall appearance of the continuous signal. Error diffusion distributes the mask signal to neighboring pixels through a diffusion filter. The mask signal is based on the difference between the ideal signal and the degamma mapping table output. Figure 11 shows a functional block diagram of the error diffusion process. The input signal is modified by the mask signal from neighboring pixels. The displayable bit resolution of the modified signal is passed on for display. The remaining signal is passed on to neighboring pixels. This error diffusion process is recursively repeated for the entire image frame.

The process of passing the process error to neighboring pixels is controlled by the diffusion filter. The diffusion filter can be designed to diffuse the error signal in many ways. Figure 12 shows an example diffusion filter. In this example, the error is diffused from four neighboring pixels to the current sample. A weighted combination of the error signals is summed into the current pixel's intensity.



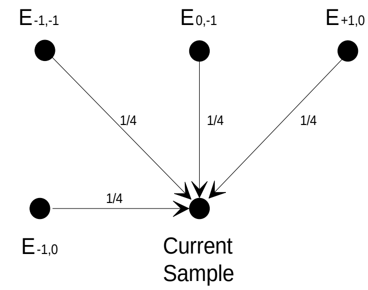


Figure 11: Error Diffusion Process

Figure 12: Error Diffusion Filter Example

VIDEO PROCESSING SYSTEMS

An example DLP video processing system architecture is shown in Figure 13. The complete system consists of front-end signal decoding, an SVP2 video processor, a programmable timing controller, field memories, a video processing ASIC (Data Path ASIC), DMD timing FPGA, and DMD display. The video processing algorithms described in this paper are implemented using the second generation Scan-line Video Processor (SVP2) [8] and the Data Path ASIC, both developed at Texas Instruments.

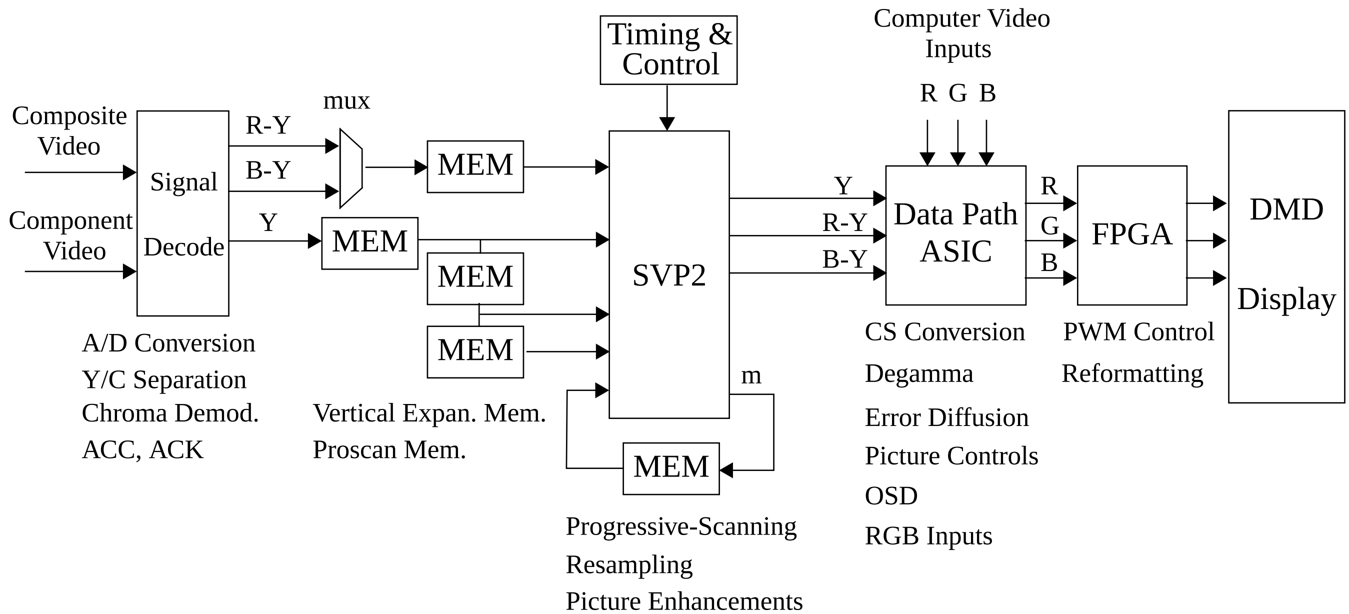


Figure 13: Video Processing Architecture for DMD Display System

The SVP2 is a fully programmable video processor capable of performing 5.7 billion 8-bit additions/second. The SVP2 uses a SIMD (Single Instruction, Multiple Data) architecture, comprised of a one dimensional array of 1-bit Processing Elements (PEs). Each PE, corresponding to one pixel on a line of video data, consists of a 1-bit ALU, two Register Files (RF0 and RF1), four working registers, and a Left/Right communication bus. Data is serially loaded into the PEs using the 48-bit wide Data Input Register (DIR). Processed data is transferred to the 32-bit wide Data Output Register (DOR) and output in a word-serial fashion. Independent clocks control the concurrent operations of data input by DIR, processing by PEs, and output by DOR.

The Data Path ASIC (DPA) has been designed specifically to support the DLP architecture. The DPA supports video signal inputs and an on screen display (OSD) signal. The DPA output provides RGB pixel databus for interface to the data reformater. The DPA provides a general 3x3 matrix multiplier for color space conversion, RGB LUTs for gamma processing, contrast, brightness, and color temperature picture controls, error diffusion processing, an auxiliary data channel for picture-in-picture (POP), picture-out-of-picture (POP), and computer graphics inputs, and OSD injection and control.

The video processing technology described in this paper is used in various forms in various DLP systems. For example, a DLP system meant primarily for computer signal display may not require features (e.g. resampling modes) specific to television. Various DLP systems incorporating this video processing technology to varying degrees are being developed for a wide range of applications including: conference room projectors, home theater, standard television, and institutional projectors. Table 3 lists some recent forums where DLP based projection displays incorporating our video processing technology have been demonstrated.

Display Application	Collaborations	Demonstrations
Conference Room Projector	nView / TI	INFOCOM 1995 (Dallas) COMDEX 1995 (Las Vegas) European Media Day 1995 (Brussels) SATIS 1995 (Paris)
Home Theater	Runco / TI Vidikron / TI	CEDIA 1995 (Dallas) European Media Day 1995 (Brussels)
Standard Television	Nokia / TI	IFA 1995 (Berlin) JES 1995 (Osaka) COMDEX 1995 (Las Vegas) CES 1996 (Las Vegas)
Institutional Projector	TI	JES 1995 (Osaka)

Table 3: DLP Video Processing Demonstrations

SUMMARY

DLP projection displays have been demonstrated for use in applications ranging from standard television to institutional projection systems. Video processing technology has been developed to address video signal conversion and enhancement for DLP display. The digital video processing solutions have been developed to maintain the superior quality of the DMD based display.

REFERENCES

[1] V. Markandey, T. Clatanoff, R. Gove, K. Ohara, “Motion Adaptive Deinterlacer For DMD (Digital Micromirror Device) Based Digital Television”, IEEE Transactions on Consumer Electronics, Vol. 40, No. 3, August 1994.

[2] M. Karlsson, et al., “Evaluation of Scanning Rate Up Conversion Algorithms; Subjective Testing of Interlaced to Progressive Conversion”, IEEE Transactions on Consumer Electronics, Vol. 38, No. 3, August 1992.

[3] M. Yugami, K. Ohara, A. Takeda, “EDTV with Scan-Line Video Processor”, IEEE Transactions on Consumer Electronics, Vol.38, No.3, August 1992.

[4] S. Naimpally, et. al., “Integrated Digital IDTV Receiver with Features”, IEEE Transactions on Consumer Electronics, Vol. 34, No. 3, August 1988.

[5] T. Doyle, M. Looymans, “Progressive Scan Conversion Using Edge Information”, Signal Processing of HDTV, II, Proc. of the Third Intl. Workshop on HDTV, August 1989.

[6] R. Crochiere, L. Rabiner, “Multirate Digital Signal Processing”, Prentice-Hall, 1983.

[7] G. Wolberg, “Digital Image Warping”, IEEE Computer Society Press, 1990.

[8] K. Nakayama, et al., “EDTV-II Decoder by SVP2 (the 2nd generation Scan-line Video Processor)”, IEEE Transactions on Consumer Electronics, Vol 41, No. 3, August 1995.