

---

# The J300 Family of Video and Audio Adapters: Architecture and Hardware Design

**The J300 family of video and audio adapters provides a feature-rich set of hardware options for Alpha-based workstations. Unlike earlier attempts to integrate full-motion digital video with general-purpose computer systems, the architecture and design of J300 adapters exploit fast system and I/O buses to allow video data to be treated like any other data type used by the system, independent of the graphics subsystem. This paper describes the architecture used in J300 products, the video and audio features supported, and some key aspects of the hardware design. In particular, the paper describes a simple yet versatile color-map-friendly rendering system that generates high-quality 8-bit image data.**

**Kenneth W. Correll  
Robert A. Ulichney**

The overall architectural design goal for the J300 family of video and audio adapters was to provide the hardware support necessary to allow the integration of broadcast video into workstations. The three primary objectives were as follows: (1) digitized video data should be treated the same as any other data type in the system; (2) the video and the graphics subsystem designs should be completely independent of each other; and (3) any hardware designed should be low cost.

Digital has implemented the J300 architecture in three products: Sound & Motion J300, FullVideo Supreme JPEG, and FullVideo Supreme.<sup>1</sup> The Sound & Motion J300 (referred to in this paper simply as the J300) was the first product designed with this architecture and is the primary focus of this paper. The FullVideo Supreme JPEG and FullVideo Supreme products are based on the same design database as the J300. They differ from the J300 in the bus supported (they support the peripheral component interconnect [PCI] bus) and the lack of audio support. Additionally, the FullVideo Supreme product does not include hardware compression/decompression circuitry.

The J300 brings a wide range of video and audio capabilities to machines based on Digital's TURBOchannel I/O interconnect. Analog broadcast video can be digitized, demodulated, and rendered for display on any graphics device. The J300 provides hardware video compression and decompression to accelerate applications such as videoconferencing. The J300 supports analog broadcast video output from either compressed or uncompressed video files. Audio support includes a general-purpose, digital signal processor (DSP) to assist in the real-time management of the audio streams and for advanced processing, such as compression, decompression, and echo cancellation. Audio input and output capabilities include stereo analog I/O, digital audio I/O, and a headphone/microphone jack. Analog audio can be digitized to 16 bits per sample at a rate of up to 48 kilohertz (kHz).

This paper begins with an overview of some terminology commonly used in the field of broadcast video. The paper then presents the evolution and design of the J300 architecture, including several key enabling

technologies and the logical video data paths available. Next follows a discussion of the hardware design phase of the project and the trade-offs made to reconcile expectation and implementation. Detailed descriptions are devoted to specific areas of the design, including the video I/O logic, the AccuVideo rendering path, and the video and audio direct memory access (DMA) interfaces.

## Video Terminology Overview

Three fundamental standards are in use worldwide for representing what is referred to in this paper as broadcast video: the National (U.S.) Television System Committee (NTSC) recommendation, Phase Alternate Line (PAL), and Séquentiel Couleur avec Mémoire (SECAM). The standards differ in the number of horizontal lines in the display, the vertical refresh rate, and the method used for encoding color information. North America and Japan use the 525-line, 60-hertz (Hz) NTSC format; PAL is used in most of Europe; and SECAM is used primarily in France. Both the PAL and SECAM standards are 625-line, 50-Hz systems.<sup>2</sup>

All three television standards split an image or a frame of video data into two fields, referred to as the even and the odd fields. Each field contains alternate horizontal lines of the frame. The vertical refresh rate cited in the previous paragraph is the field rate; the frame rate is one-half of that rate.

Unlike computer display systems that use red, green, and blue (RGB) signals to represent color information, PAL and SECAM use a luminance-chrominance system, which has the three parameters  $Y$  (the luminance component), and  $U$  and  $V$  (the two chrominance components). NTSC uses a variation of YUV, where the  $U$  and  $V$  components are rotated by 33 degrees and called  $I$  and  $Q$ . YUV is related to RGB by the following conversion matrix:<sup>3</sup>

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ U &= -0.169R - 0.331G + 0.500B \\ V &= 0.500R - 0.419G - 0.081B \end{aligned}$$

All the different standards limit the bandwidth of the chrominance signal to between one-quarter and one-third that of the luminance signal. This limit is taken into account in the digital representation of the signal and results in what is called 4:2:2 YUV, where, for every four horizontally adjacent samples of  $Y$ , there are two samples of both  $U$  and  $V$ . All three components are sampled above the Nyquist rate in this format with a significant reduction in the amount of data needed to reconstruct the video image.

Various modulation techniques transform the separate  $Y$ ,  $U$ , and  $V$  components into a single signal, typically referred to as composite video. To increase the fidelity of video signals by reducing the luminance-chrominance cross talk caused by modulation, the

S-Video standard has been developed as an alternative. S-Video, which refers to separate video, specifies that the luminance signal and the modulated chrominance signal be carried on separate wires.

The J300 includes hardware support for the Joint Photographic Experts Group (JPEG) compression/decompression standard.<sup>4</sup> JPEG is based on the discrete cosine transform (DCT) compression method for still-frame color images. DCT is a widely accepted method for image compression because it provides an efficient mechanism to eliminate components of the image that are not easily perceived by casual inspection.

## Design History and Motivation

Digital arrived at the J300 adapter design after considering several digital video playback architectures. The Jvideo advanced development project, the implementation of one of the alternatives, was instrumental in achieving the design goals.

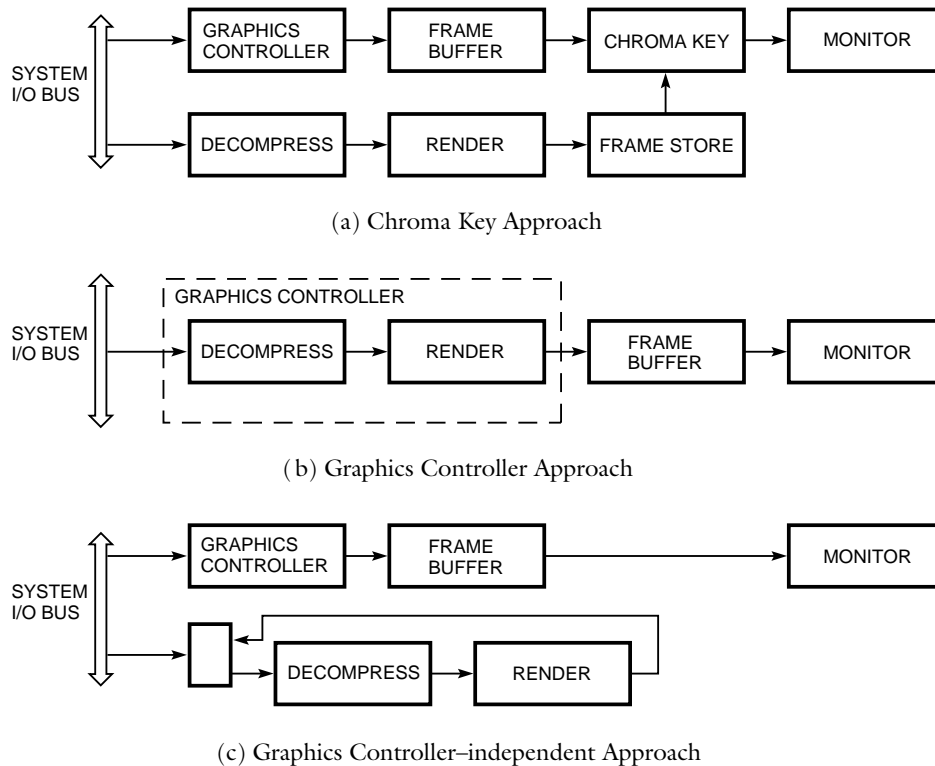
### Architectural Alternatives and Objectives

In January 1991, several Digital engineering organizations collaborated to define the architecture of a hardware seed project that could be used to explore a workstation's capability to process video data. The participants felt that the key technologies required to explore the goal of integrating computers and broadcast video were available. These enabling technologies were

1. The TURBOchannel high-speed I/O bus, which was a standard on Digital workstations
2. The anticipated acceptance of the JPEG compression/decompression standard and single-chip implementations that supported that standard
3. The development of a rendering system (now called the AccuVideo system) that could map YUV input values into an 8-bit color index using any number of available colors with very good results

We evaluated the three alternative approaches shown in Figure 1 for moving compressed video data from system memory, for decompressing and rendering the data, and, finally, for moving the data into the frame buffer.

The chroma key approach, shown in Figure 1a, differs little from previous work done at Digital and was the primary architecture used by the industry. Several variations of the exact implementation are in use, but, basically, the graphics device paints a designated color into sections of the frame buffer where the video data is to appear on the display. A comparator located between the graphics frame buffer and the display device looks at the serial stream of data coming from the graphics frame buffer and, when the data matches the chroma key (stored in a register), inserts the video data. As shown in Figure 1a, this approach



**Figure 1**  
Digital Video Playback Architectures

relies on a special connection between the video decompression block and the output of the graphics device. While this approach off-loads the system I/O bus, it treats video data differently from other data types to be displayed. In particular, the X Window System graphical windowing environment has no knowledge of the actual contents of the video window at any given time.

The graphics controller approach, shown in Figure 1b, integrates the decompression technology with the graphics accelerator. Although this approach has the potential of incurring the lowest overall system cost, it fails in two important aspects. First, it does not expose the windowing system to the video data. Second, since the graphics controller and video logic are integrated, the user must accept the level of graphics performance provided. No graphics upgrade path exists, so upgrading would require another product development cycle. Including the video logic across the range of graphics devices is not desirable, because such a design forces higher prices for users who are not interested in the manipulation of broadcast video.

The third approach, shown in Figure 1c, is much more radical. It places the responsibility of moving each field of video data to and from the decompression/rendering option squarely on the system. The system I/O bus must absorb not only the traffic generated by the movement of the compressed video to the decompression hardware but also the movement of the

decompressed video image from the accelerator back to system memory and back again over the same bus to the graphics option.

Accepting the third alternative architecture allowed us to meet the three important objectives for the project:

1. The workstation should be able to treat digitized video data the same as any other data type.
2. The inclusion of video capabilities in a workstation should be completely independent of the graphics subsystem used.
3. Any hardware option should be low cost.

The original design goals included audio I/O, even though the processing power and bandwidth needed for audio were far below those required for video. Since users who want video capability usually require audio capability as well, audio support was included so that users would have to buy only one option to get both audio and video. This design reduced the number of bus slots used.

#### ***The Jvideo Advanced Development Project***

Jvideo was the name given to the advanced development hardware seed project. Actual design work started in February 1991; power on occurred in September 1991. Jvideo has since become a widely used research tool.

**Table 1**  
The Nine Video Flow Paths

Input	Output			
	Analog	Compressed	Uncompressed	Dithered
Analog	...	A → C	A → U	A → D
Compressed	C → A	...	C → U	C → D
Uncompressed	U → A	U → C	...	U → D

Jvideo was an important advanced development project for several reasons. First, it was the vehicle used to verify the first two project objectives. Second, it was the first complete hardware implementation of the rendering circuit, thus verifying the image quality that was available when displaying video with fewer than 256 colors. Finally, it was during the development of Jvideo that the DMA structure and interaction with the system was developed and verified.

### J300 Features

This section describes the various video paths supported in the J300 and presents videoconferencing as an example of video data flow. The AccuVideo filter-and-scale and dithering system designs used in the J300 are presented in detail.

#### Video Paths

Table 1 summarizes the nine fundamental video paths that the J300 system supports. The input to the J300 can come from an external analog source or from the system in compressed or uncompressed form. The outputs include analog video and several internal formats, i.e., JPEG compressed, uncompressed, or dithered. Dithering is a technique used to produce a visually pleasant image while using far less information than was available in the original format.

A conceptual flow diagram of the major components of the J300 video system is shown in Figure 2. Physically, the frame store and the blocks to its left make up the video board. All the other blocks except for JPEG compression/decompression are part of the J300 application-specific integrated circuit (ASIC).

(The J300 Hardware Implementation section provides details on this ASIC.)

Both the upscale prior to the analog out block and the downscale after the analog in block scale the image size independently in the horizontal and vertical directions with arbitrary real-value scale factors. The filter-and-downscale function is handled by the Philips chip set, as described in the J300 Hardware Implementation section. The upscale block is a copy of the Bresenham-style scale circuit used in the filter-and-scale block.

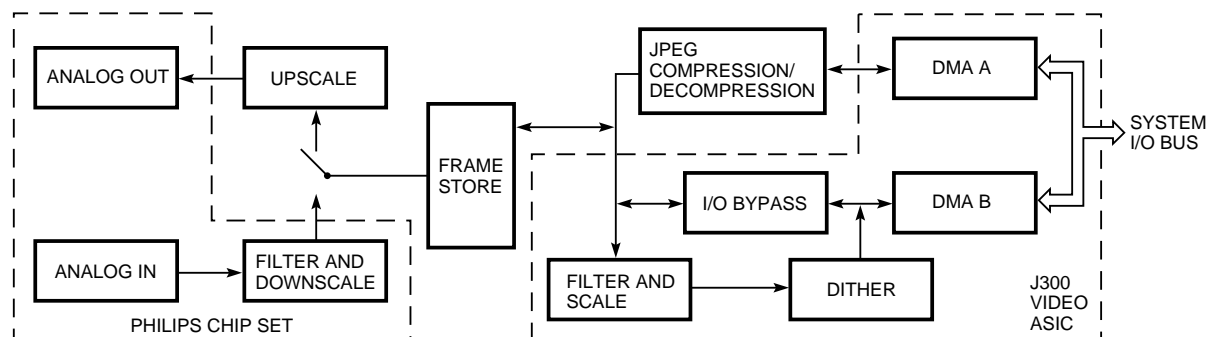
The Bresenham-style scale circuit is extremely simple and is described in "Bresenham-style Scaling," along with an interesting closed-form solution for finding initial parameters.<sup>5</sup> The filter-and-scale block is part of the J300 rendering system. The J300 supports arbitrary scaling for either enlargement or reduction in both dimensions. We carefully selected a few simple, three-element horizontal filters to be used in combination with scaling; the filters were small enough to be included in the J300 ASIC. The J300 supports three sharpening filters that are based on a digital Laplacian:<sup>6</sup>

$$\begin{aligned} \text{Low sharpness} & \quad \begin{pmatrix} -1/2 & 2 & -1/2 \end{pmatrix} \\ \text{Medium sharpness} & \quad \begin{pmatrix} -1 & 3 & -1 \end{pmatrix} \\ \text{High sharpness} & \quad \begin{pmatrix} -2 & 5 & -2 \end{pmatrix} \end{aligned}$$

The J300 also supports two low-pass or smoothing filters:

$$\begin{aligned} \text{Low smoothing} & \quad \begin{pmatrix} 1/4 & 1/2 & 1/4 \end{pmatrix} \\ \text{High smoothing} & \quad \begin{pmatrix} 1/2 & 0 & 1/2 \end{pmatrix} \end{aligned}$$

Sharpening is performed before scaling for enlargement and after scaling for reduction. Smoothing is always performed before scaling (as a band limiter) for reduction and after scaling (as an interpolator) for enlargement.



**Figure 2**  
J300 Video Flow

The second part of video rendering occurs in the dither block. The AccuVideo Rendering section provides details on this block.

The I/O bypass skips over the video rendering blocks when undithered uncompressed output is required. When uncompressed digital video is used as input, the I/O bypass is also used. DMA B thus passes dithered or uncompressed output and uncompressed input.

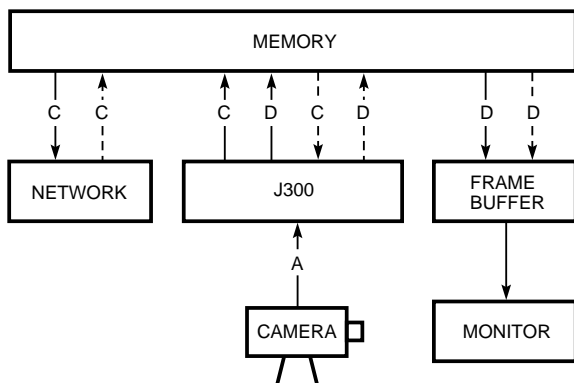
Compressed input and compressed output are passed through DMA A. The JPEG compression/decompression block handles all compression of output and decompression of input. The combination of the two DMA channels allows high data rates because both channels are often used in parallel.

### Videoconferencing Application

A good illustration of the video data flow in J300 is a videoconferencing application. Figure 3 shows the flow of analog (A), compressed (C), and dithered (D) video data to and from memory in a system on a network. The application software controls the flow of data between memory and the display and network devices. The J300 hardware must perform two fundamental operations:

1. Capture the local analog signal, compress the data, and send it to memory, and in parallel, dither the data and send it to memory. The solid arrows in Figure 3 denote the compress, send, and view paths.
2. Receive a remote compressed video stream from memory, decompress and dither the data, and send it back to memory. The dashed arrows in Figure 3 denote the receive, decompress, and view paths.

Figure 3 demonstrates the unique graphics controller independence of the J300 architecture, as shown in Figure 1c. In assessing the aggregate video data traffic, it is important to keep in mind that the



Notes: Dashed arrows represent the receive, decompress, and view paths (C→D). Solid arrows represent the compress, send, and view paths (A→C, A→D). The symbols A, C, and D stand for analog, compressed, and dithered data.

**Figure 3**  
Videoconferencing Application

dithered data is 8 bits per pixel, and the compressed data is approximately 1.5 bits per pixel. For example, consider a videoconference with 11 participants, where each person's workstation screen displays the images of the other 10 participants, each in a 320-by-240-pixel window and with a refresh rate of 20 Hz. The bus traffic required for each window is twice the compressed image size plus twice the decompressed image size, i.e.,  $(2 \times 320 \times 240 \times 1.5) \div 8$  bytes +  $(2 \times 320 \times 240)$  bytes = 182.4 kilobytes (kB) per window. The total bandwidth would be 182.4 kB  $\times$  11 windows  $\times$  20 Hz = 40.1 megabytes (MB) per second, which is well within the achievable bandwidth of both TURBOchannel and PCI buses.

These two operations through the J300 conceptual flow diagram of Figure 2 are shown explicitly in Figure 4 for the capture, compress, and dither paths, and in Figure 5 for the decompress and dither path. In Figure 4, video data is captured through the analog in block and buffered in the frame store block. The frame store then sends the data in parallel to the JPEG compression/decompression path, and to the filter, scale, and dither path, each of which sends the data to its own dedicated DMA port.

In Figure 5, compressed data enters DMA A, is JPEG decompressed using the frame store as a buffer, and is sent to the filter, scale, and dither path, where it is output through DMA B.

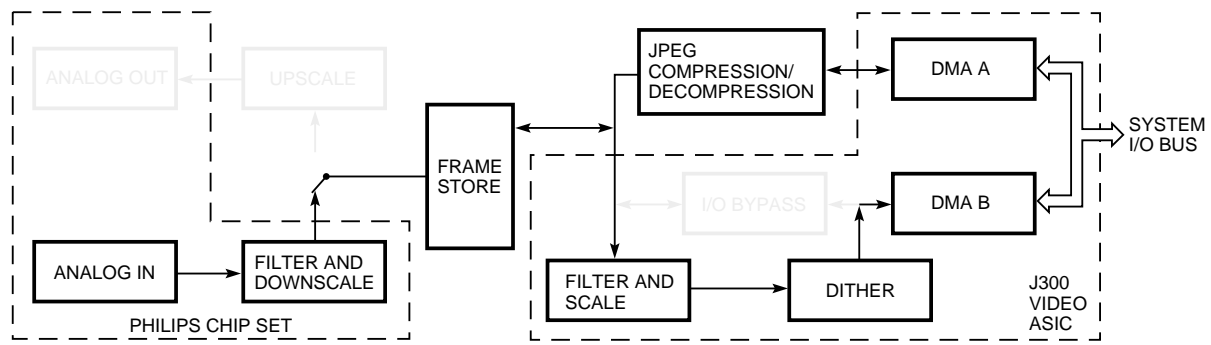
Figures 4 and 5 illustrate three of the nine possible video paths shown in Table 1. It is straightforward to see how the other six paths flow through the block diagram of Figure 2.

### AccuVideo Rendering

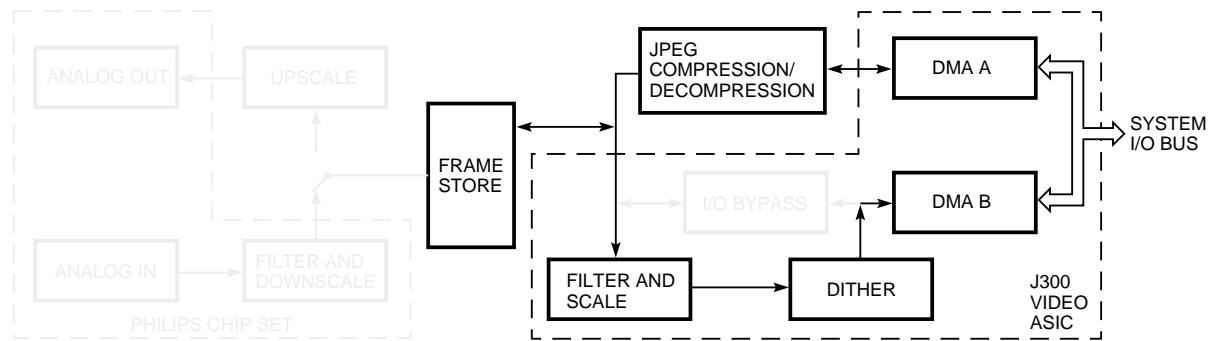
Digital's AccuVideo method of video rendering is used in the J300 and in other products.<sup>7,8</sup> J300 rendering is represented in Figure 2 by the filter-and-scale block and by the dither block. The following features are supported:

- High-quality dithering
- Selectable number of colors from 2 to 256
- YUV-to-RGB conversion with controlled out-of-bounds mapping
- Brightness, contrast, and saturation control
- Color or gray-scale output
- Two-dimensional (2-D) scaling to any size
- Sharpening and smoothing control

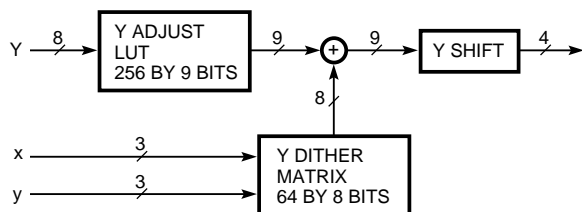
The algorithm for mean-preserving multilevel dithering is described by Ulichney in "Video Rendering."<sup>9</sup> Mean preserving denotes that the macroscopic average in the output image is maintained across the entire range of input values. Figure 6 depicts the version of the dithering algorithm used for the single component *Y* in the J300 prototype, Jvideo.



**Figure 4**  
Capture, Compress, and Dither Paths



**Figure 5**  
Decompress and Dither Path



**Figure 6**  
Dither Components of the Jvideo Prototype

To quantize with a simple shift register and still maintain mean preservation, a particular gain that happens to have a value between 1 and 2 must be imparted to the input.<sup>9</sup> This gain is included in the adjust look-up table (LUT), thus adding a bit to the data width of the input value to the ditherer.

In the case of the  $Y$  (luminance) component, the effect of brightness and contrast can be controlled by dynamically changing and loading the contents of this adjust LUT. Saturation control is a contrast-like mapping controlled on the  $U$  and  $V$  adjust LUTs.

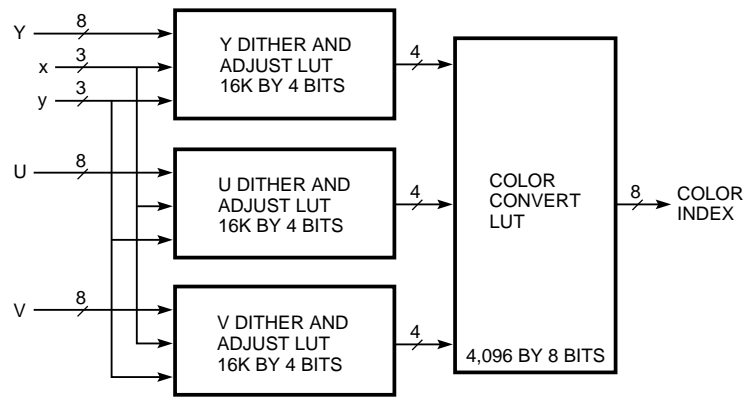
The least significant bits of the horizontal and vertical address ( $x, y$ ) of the pixel index the dither matrix. In the Jvideo prototype, we used an 8 by 8 recursive tessellation array.<sup>7</sup> Because the size of the array was so small, all the components in Figure 6 could be

encapsulated with a single 16K-by-4-bit random-access memory (RAM). This implementation is not the least expensive, but it is the easiest to build and is quite appropriate for a prototype.

Figure 7 illustrates the Jvideo dither system. The number of dither levels and associated color adjustment are designed in software and loaded into each of the 16K-by-4-bit LUTs for  $Y$ ,  $U$ , and  $V$ . Each component outputs from 2 to 15 dithered levels. The three 4-bit dithered values are used as a collective address to a color convert LUT, which is a 4K-by-8-bit RAM.

Loaded into this LUT is the conversion of each YUV triplet to one of  $N$  RGB index values. The generation of this LUT incorporates the state of the display server's color map at render time. Although this approach is much more efficient than a direct algebraic conversion known as dematrixing, an arbitrarily complex mapping of out-of-range values can take place because the table is built off line. Another paper in this issue of the *Journal*, "Software-only Compression, Rendering, and Playback of Digital Video," presents details on this approach.<sup>7</sup>

Perhaps the central characteristic of AccuVideo rendering is the pleasing nature of the dither patterns generated. We are able to obtain such patterns because we incorporate dither matrices developed using the void-and-cluster method.<sup>10</sup> These matrices are 32 by



**Figure 7**  
Jvideo Dither System

32 in extent. Although surprisingly small for the complexity and seamlessness of the patterns produced, this size requires 10 bits of display address information for indexing.

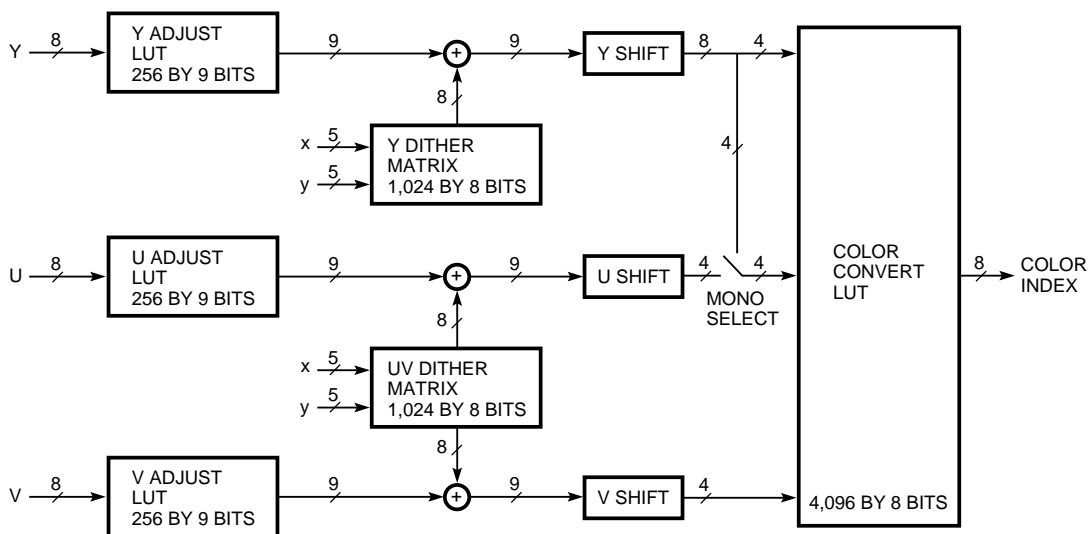
While very simple to implement, the single LUT approach used in the Jvideo system shown in Figure 7 becomes unattractive for a matrix of this size because of the large memory requirement. Eight bits of input plus 10 bits of array address requires a 256K-bit RAM for each color component; Jvideo's 8 by 8 dither matrix called for a more cost-effective 16K-bit RAM.

The dither system design used in the J300 is shown in Figure 8. The design is quite simple, requiring only RAM and three adders. We restricted the number of *U*- and *V*-dithered levels to always be equal. Such a restriction allows the sharing of a single dither matrix RAM. The paper "Video Rendering" provides details on

the relationship between the number of dithered levels for each component, the number of bits shifted, the normalization of the dither matrix values, the gain embedded in the adjust LUT, and the bit widths of the data paths.<sup>9</sup> Note that the decision to use RAM instead of read-only memory (ROM) for the adjust LUTs, dither matrices, and color convert LUT permits complete flexibility in selecting the number of dithered colors.

When the video source is monochrome, or whenever a monochrome display is desired, a Mono Select mode allows the *Y* channel to be quantized to up to 8 bits.

The algorithm used in the software-only version of AccuVideo exactly parallels Figure 8.<sup>7</sup> "Integrating Video Rendering into Graphics Accelerator Chips" describes variations of this architecture for other products.<sup>8</sup> One design always renders the same number of colors without adjustment, in favor of very low



**Figure 8**  
J300 Dither System

cost. Another performs YUV-to-RGB conversion first, to allow dithering to more than 256 colors. Note that with this design, for large numbers of output colors, the memory required for the back-end color convert LUT design would be prohibitive.

### **J300 Hardware Implementation**

Implementing the J300 hardware design entailed making trade-offs to keep down the costs. This section presents the major trade-offs and then discusses the resulting video and audio subsystem designs, the built-in I/O test capabilities, and the Verilog hardware description language design environment used.

#### ***Design Trade-offs***

In August 1991, the Jvideo hardware design team presented to engineering management several cost-reducing design alternatives with the goal of turning Jvideo into a product. Alternatives ranged from retaining the basic design (which would require a short design time and would result in the fastest time to market) to redesigning the board with minimal cost as the driving factor (which meant putting as much logic as possible into the J300 ASIC). Management accepted the latter proposal, and design started in January 1992.

The major design trade-offs involved in reducing module cost centered around three portions of the design: the accelerator chip, the pixel representation, and the dither circuit. The design team evaluated different JPEG hardware compression/decompression accelerators in terms of availability, performance, cost, and schedule risk. While various manufacturers claimed to have cheaper parts available within our design schedule constraints, the CL550 chip from C-Cube Microsystems, the same chip used in the Jvideo system, had reasonable performance and known idiosyncrasies. The designers decided to use one CL550 chip instead of two, as was done in Jvideo. This meant that in videoconferencing applications, the chip would have to be programmed to compress the incoming image and then reprogrammed to decompress the other images. The turnaround time of the programming required to implement the design change plus the compression time together accounted for the performance penalty that the product would pay for including only one CL550.

To understand the impact on performance of using just one CL550 chip, consider that all 700 registers in the chip would have to be reloaded when changing the chip from compression to decompression and vice versa. Given a register write cycle of 250 nanoseconds, the penalty is 175 microseconds. We estimated the time to compress an image as the number of pixels in the uncompressed image (the CL550 does occasion-

ally stall during compression or decompression, but we ignored this fact for these calculations) times the period of the pixel rate. For an image size of 320 by 240 pixels and a pixel clock period of 66.67 nanoseconds, the time used for compression is 5.12 milliseconds. If the desired overall frame rate of all images on the screen is 20 Hz, then approximately 11 percent of the available time is given to compression ( $(5.12 \text{ milliseconds} + 0.35 \text{ milliseconds}) \div 50 \text{ milliseconds}$ ). We judged this decrease in decompression performance reasonable, since approximately 30 percent of the early estimated cost of materials on the J300 was the CL550 and the associated circuits.

The second major area of savings came with the decision to use the 4:2:2 YUV pixel representation in the frame store, the CL550, and the input to the rendering logic. This approach reduced the width of the frame store and external data paths from 24 to 16 bits with no loss of fidelity in the image. The trade-off associated with this decision was that the design precluded the ability to directly capture video in 24-bit RGB unless the ASIC included a full YUV-to-RGB conversion. The main thrust of the product was to accelerate image compression and decompression on what was assumed to be the largest market, i.e., 8-bit graphics systems, by using the AccuVideo rendering path. Since 24-bit RGB can be obtained from 4:2:2 YUV pixel representation (which can be captured directly) with no loss of image fidelity, we considered this hardware limitation to be minor.

The third area of trade-offs revolved around the implementation of the dither circuit and how much of that circuitry the ASIC should include. The rendering system on Jvideo was implemented entirely with LUTs, a method that is inexpensive in terms of the random logic needed but expensive in terms of component cost. Early on, the design team decided that including the 4K-by-8-bit color convert LUT inside the ASIC was not practical. Placing the LUT outside the ASIC required using a minimal number of pins, 28, and using a readily available 8K-by-8-bit static random-access memory (SRAM) allowed the unused portion of the RAM to store the dither matrix values. Such a design reduced the amount of on-chip storage required for dither matrix values to 32 by 8 bits.

The impact of requiring dither matrix value fetches on a per-line basis added to the interline overhead 32 accesses for the new dither matrix values or 16 pixel clocks. The impact of the 16 added clocks on a line basis depends on the resultant displayed image size. If the displayed images are small, the impact is as much as 10 percent (for a 160-by-120-pixel image). It is uncommon, however, for someone to view video on a workstation at that resolution. At a more common displayed size of 640 by 480, the amount of overhead decreases to 3 percent.



### Video Subsystem Design

The major elements of the video subsystem design are the ASIC, which is designed in the Verilog hardware description language, the Philips digital video chip set, and the compression/decompression circuitry. This section discusses the ASIC design and some aspects of the video I/O circuit design.

**The J300 ASIC** The J300 ASIC design included not only the video paths discussed earlier in the section J300 Features but also all the control for the video I/O section of the design, all video random-access memory (VRAM) control, the CL550 interface, access to the diagnostics ROM, arbitration with the audio circuit for TURBOchannel access, and the TURBOchannel interface. Figure 9 shows a block diagram of the J300 ASIC. Only the DMA section of the design is discussed further in this paper.

The DMA interface built into the ASIC is designed to facilitate the movement of large blocks of data to or from system memory with minimal interaction from the system. The chip supports two channels: the first is used for CL550 host port data (compressed video and register write data); the second is used for pixel data flowing to or from the rendering circuit. Once started, each channel uses its map pointer register to access successive (*address, length*) pairs that describe the physical memory to be used in the operation. (The map pointer register points to the scatter/gather map

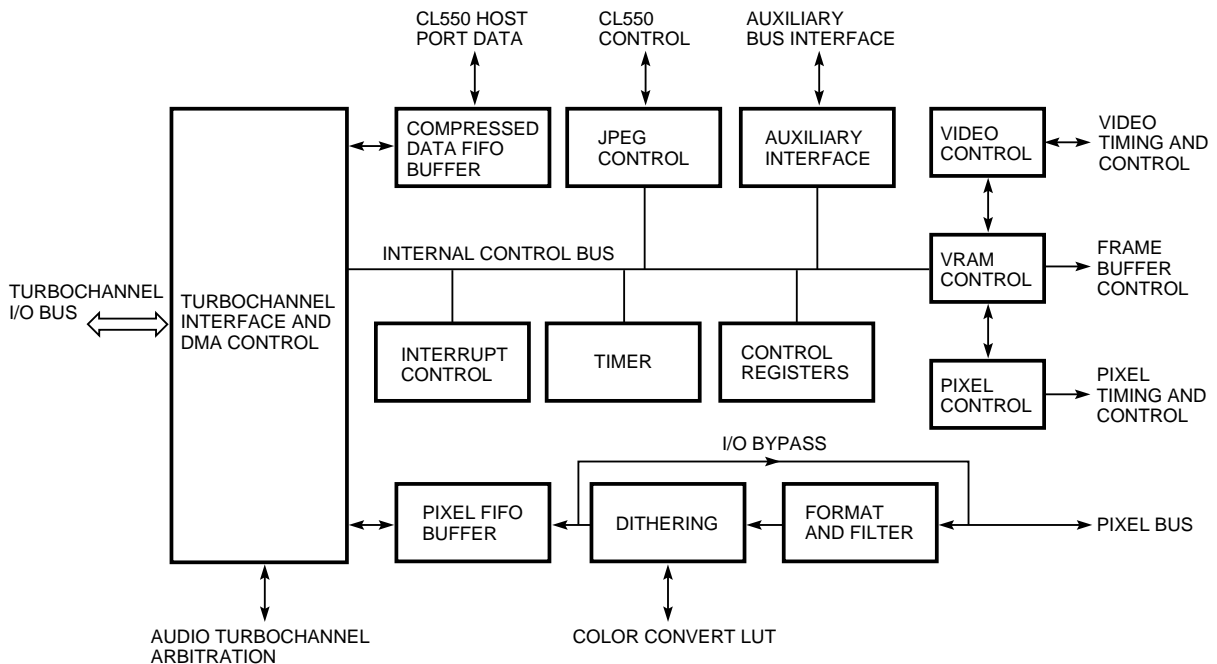
in system memory to be used.) The ASIC fills or empties the first buffer and then automatically fetches the next (*address, length*) pair in the scatter/gather map and so on until the operation is complete. When a compressed image is transferred into system memory, the exact length of the data set is unknown until the ASIC detects the end-of-image marker from the CL550. In this case, system software can read a length register to find out exactly how much data was transferred.

There is no restriction on the number of (*address, length*) pairs included in each scatter/gather map. New pairs can be assigned to each line of incoming video such that deinterlacing even and odd video fields can be accomplished as the data is moved into system memory.

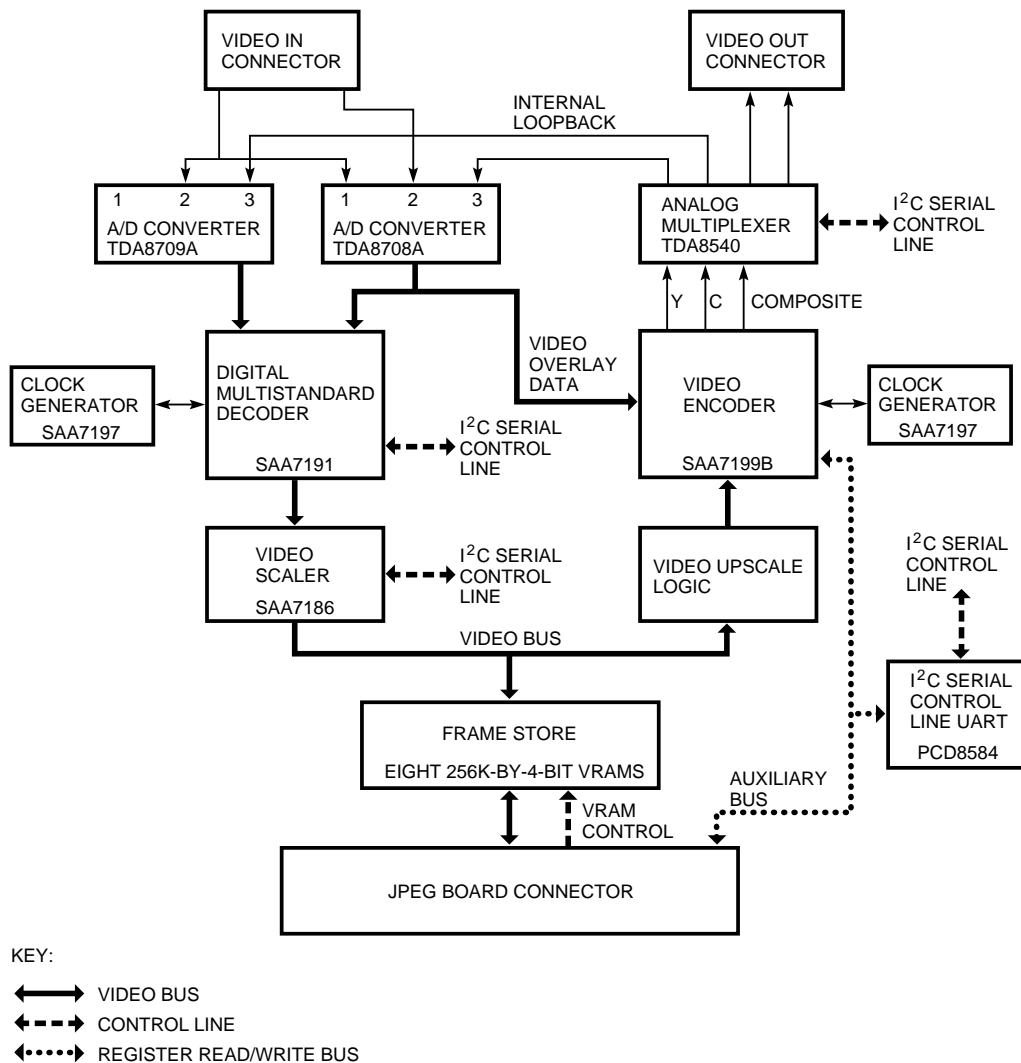
Since only the map pointer register needs to be updated between operations, system software can set up multiple buffers, each with its associated scatter/gather map, ahead of time.

**Video Input and Output Logic** The J300 video I/O circuit, shown in Figure 10, was designed using Philips Semiconductors' digital video chip set. Explanation of some aspects of the design follows.

The J300 uses the Philips chip set to digitize and decode input video. The chip set consists of the TDA8708A and the TDA8709A, as the analog-to-digital (A/D) converters, and the SAA7191, as the Digital MultiStandard Decoder (DMSD). This chip



**Figure 9**  
J300 ASIC Block Diagram



**Figure 10**  
J300 Video I/O

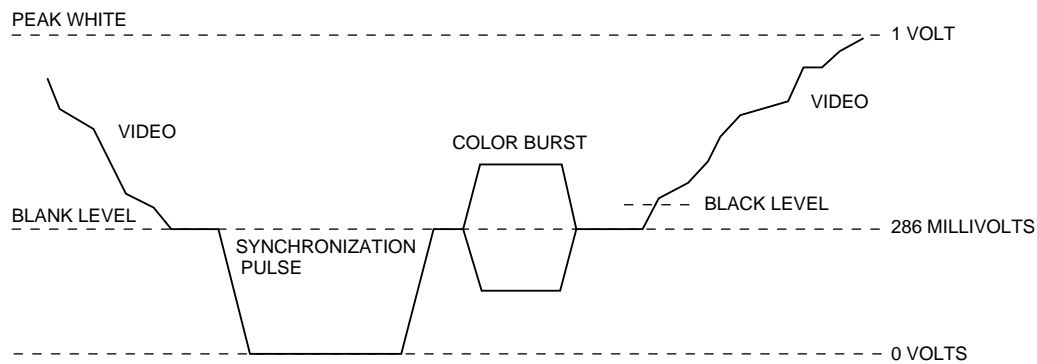
set supports NTSC (M), PAL (B, G, H, D), and SECAM (B, G, H, D, K, K1) formats.<sup>2</sup> It also supports square pixels, where the sampling rate is changed to 12.272727 megahertz (MHz) for the NTSC format and to 14.75 MHz for the PAL and SECAM formats. In addition, the J300 uses the SAA7186, a digital video scaler chip that can scale the input to an arbitrary size and perform horizontal and vertical filtering.

The A/D converters digitize the incoming video signal to 256 levels. A video signal is composed of negative-going synchronization pulses, a color burst (to aid in decoding color information), and positive-going video.<sup>11</sup> As an aid to visualizing this, Figure 11 illustrates a simplified version of the drawing presented in the Color Television Studio Picture Line Amplifier Output Drawing.<sup>11</sup> The level before and after the syn-

chronization pulses is referred to as blank level. Black level may or may not be the same as blank, depending on the standard. Video signals are 1 volt peak to peak.

The first stage included in the A/D converters is a three-to-one analog multiplexer. We used this circuit to allow two composite signals to be attached at the same time to support S-Video while allowing the third input to be used as an internal loop-back connection. The TDA8708A chip is used for composite video and for the luminance portion of S-Video. The TDA8709A chip is used only for the chrominance portion of S-Video.

The A/D converters contain an automatic gain control (AGC) circuit, which limits the A/D range. The bottom of the synchronization pulse is set at 0, and blank level is set at 64. Given these settings, peak white



**Figure 11**  
Depiction of Video Signal Terminology

corresponds to a value of 224. If the input video level tends to exceed 213, a peak white gain control loop is activated, which lowers the internal gain of the video. The SAA7191 processes the luminance, and the resulting range in the  $Y$  value is 16 for black and approximately 220 for white. As recommended by CCIR Report 601-2, there is room built into the two A/D converters and the DMSD to allow for additive noise that might be present in the distribution of video signals.<sup>3</sup>

The J300 video I/O design includes a video scaler so that the incoming video can be scaled down and filtered prior to compression. There are two primary reasons for this scaling. First, scaling reduces the amount of data to be processed, which results in a smaller compressed version of the image. Second, scaling removes any high-frequency noise in the image, which results in higher compression ratios. Unfortunately, if the user wishes to compress and also to view the incoming video stream, the video will more than likely be scaled again in the rendering circuit in the ASIC.

The J300 output video encoding circuit uses Philips' SAA7199B chip as the encoder. This component is followed by a low-pass filter and an analog multiplexer (Philips' TDA8540 chip), which functions as a 4 by 4 analog cross-point switch. The SAA7199B video encoder accepts digital data in a variety of formats, including 4:2:2 YUV. The SAA7199B processes the chrominance and luminance according to which standard is being encoded, either NTSC or PAL (B, G). The input range of the SAA7199B is compliant with CCIR Report 601-2 for YUV:  $Y$  varies from 16 to 235;  $U$  and  $V$  vary from 16 to 240. The analog multiplexer allows either the composite or S-Video output of the SAA7199B to be connected to the output connector. The switch also allows the video signals to be routed to the input circuit for an internal loop-back connection.

The J300 video I/O design initially included a frame store because the CL550 could not guarantee that compression of a field of video would be completed before the next field started. Even if the J300 scaled and filtered the video data prior to compression, some temporary storage was needed. We included eight 256K-by-4-bit VRAMs in the design for this storage.

In the mode where only the even field is being captured (which could be part of reducing the size of the final image from 640 by 480 pixels to 320 by 240 pixels), the J300 does not know when the system will request the next field of incoming video. VRAMs organized as 768 by 682 by 16 bits allow room to store two fields of either NTSC or PAL video. The incoming video stream continually alternates between these two buffers. The system then has the option of requesting the field that will provide the minimum input latency or the last complete field stored. Requesting the field with the minimum input latency creates the possibility that the compression and rendering operations will stall waiting for the finish of the video field being processed.

In another mode of operation, the memory is configured as a 1,024-by-512-by-16-bit buffer. This configuration is used when compressing or decompressing still images up to 1,024 pixels wide. Another use of the frame store organized in this way is for deinterlacing. In deinterlace mode, the even and odd fields are recombined to form one image. Deinterlacing allows capture of a full NTSC frame, but of only 512 lines of a PAL or SECAM frame. This restriction is due to the nature of the shift register cycles implemented in the VRAMs. A side effect of using this deinterlace mode when compressing the input is that the temporal effects of combining the two fields generate what the CL550 considers to be a large amount of high-spatial-frequency components in the image, thus resulting in poorer compression.

### Audio Subsystem Design

The designers believed that the J300 design should include audio capabilities that complemented the video capabilities. Consequently, the design incorporates an analog codec (the CS4215 from Crystal Semiconductors) and a digital audio codec (the MC56401 from Motorola Semiconductors). These two chips provide all the audio I/O specified in the design. They communicate to the rest of the system by means of a serial digital interface.

To provide audio capabilities such as compression, decompression, and format conversion, the J300 includes a general-purpose DSP (DSP56001 from Motorola Semiconductors) with 8K by 24 bits of external RAM. This DSP can communicate to the audio codecs through an integrated port. It also handles the real-time nature of that interface by using a portion of the RAM to buffer the digital audio data.

The J300 offers the same type of DMA support for audio data as for video data. The audio interface controller ASIC, along with the DSP, provides support for four independent DMA streams. These streams correspond to the four possible sources or sinks of audio data: analog audio in, analog audio out, digital audio in, and digital audio out. The left channel of the analog audio connection can also be routed to the headphone/microphone jack. Figure 12 shows a block diagram of the audio portion of the J300.

### Testability of I/O Sections

In the early stages of design, we were aware that built-in test features were needed to facilitate debugging and to reduce the amount of special audio- and video-specific test equipment required in manufacturing. Consequently, one J300 design goal was to include

internal and external loop-back capability on all major I/O circuits. This goal was achieved with the exception of the digital audio circuit.

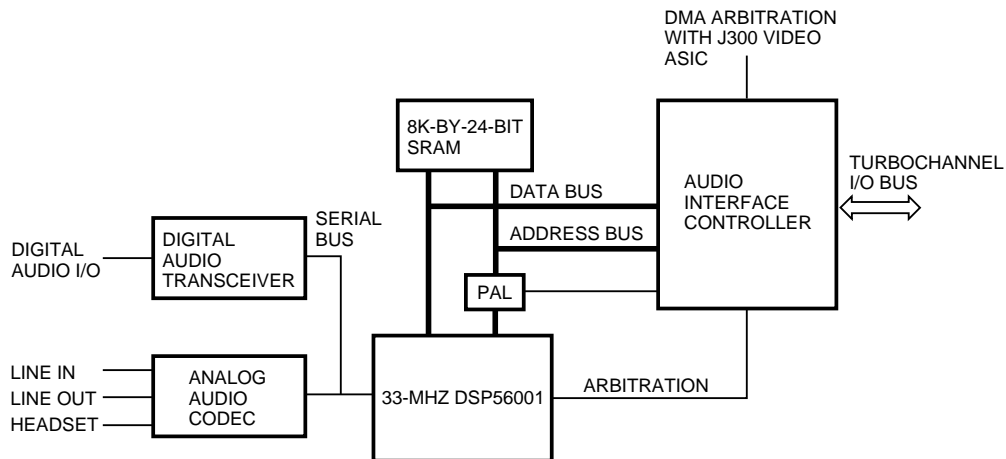
The video encoder can be programmed in test mode to output a flat field of red, green, or blue. This signal was used in internal and external loop-back. A comparison of the values obtained against known good values gives some level of confidence with regard to the video I/O stage. The designers accomplished external loop-back by using a standard S-Video cable.

The analog audio codec has internal loop-back capability, and a standard audio cable can be used for external loop-back tests. External loop-back tests of the headphone/microphone jack required a special adapter.

Even with this degree of internal and external loop-back capability, the goal was to be able to perform much more rigorous testing without the need of special instrumentation. Tests were developed that used two J300 systems to feed each other data. One J300 system output video data in NTSC or PAL formats of different test patterns, and the other J300 interpreted the signals. The designers used the same technique for both the digital and the analog audio codecs. This method provided a high degree of system coverage with no additional specialized test instruments.

### Hardware Design Environment

The ASIC was designed completely in a hardware description language called Verilog, using no schematic sheets. At first, we simulated pieces of the design, building simple Verilog models for all the devices in the J300. We simulated complex chips such as the video scaler and the CL550 as data sources or sinks, reading data from or writing data to files in



**Figure 12**  
J300 Audio Block Diagram

memory. This approach limited the video data that could be compressed or decompressed to samples where both versions already existed. In all cases, the I/O ports on devices modeled included accurate timing information. Verilog includes the capability to incorporate user-defined routines written in the C programming language that can be compiled into a Verilog executable. The J300 design team took advantage of this capability by writing an interface that took TURBOchannel accesses from a portion of shared memory and used them to drive the Verilog model of the TURBOchannel bus. In that way, the designers could write test routines in C, compile them, and run them against the Verilog model of the ASIC and of the rest of the board design.

The Verilog model proved to be useful in developing manufacturing diagnostics and was used to some extent for driver and library code development. It was a very effective tool for the hardware designers, because much of the test code written during the design phase was used to bring up the hardware in the lab and later as example code for library development. Use of the Verilog model for software development was not as extensive as was hoped, however. The requirement to have a Verilog license available each time a model was invoked limited the number of users. There were enough licenses for hardware development, but few were left for software development. Another reason the software development team did not rely on using the Verilog model was that even though the model provided an accurate simulation of the hardware, the model was also very slow.

### Concluding Remarks

With its Sound & Motion J300, FullVideo Supreme JPEG, and FullVideo Supreme products, Digital has achieved its goal of designing a hardware option that allows the integration of video into any workstation. The adapter performance on different platforms depends on many factors, chief among which are the efficiency of the bus design (either TURBOchannel or PCI), the amount of other traffic on the bus, and the design of the graphics device. As the performance of systems, particularly graphics devices, increases, the bottleneck in the J300 design becomes the pixel frequency through the J300 ASIC. For this reason, any future adapter designs should incorporate a higher pixel frequency.

The J300 family of products was the first to offer Digital's proprietary AccuVideo rendering technology, affording a high-quality yet low-cost solution for low-bit-depth frame buffers. Rendering video to 8 bits per pixel in combination with a high-speed bus allowed an architecture that is independent of the graphics subsystem.

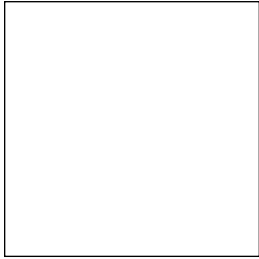
### Acknowledgments

The hardware engineering team included Rudy Stalzer, Petar Antonios, Tim Hellman, and Tom Fitzpatrick. Victor Bahl wrote the audio and video drivers, and Davis Pan wrote the code used by the DSP. Paul Gauthier contributed to the test routines available at power-up. Nagi Sivananjaiah wrote the diagnostics, and Lance Berc and Jim Gettys made major architectural contributions.

### References

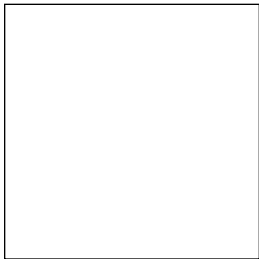
1. P. Bahl, "The J300 Family of Video and Audio Adapters: Software Architecture," *Digital Technical Journal*, vol. 7, no. 4 (1995, this issue): 34–51.
2. *Characteristics of Television Systems*, CCIR Report 624-2 (Geneva: International Radio Consultative Committee [CCIR], 1990).
3. *Encoding Parameters of Digital Television for Studios*, CCIR Report 601-2 (Geneva: International Radio Consultative Committee [CCIR], 1990).
4. *Information Technology—Digital Compression and Coding of Continuous-tone Still Images: Requirements and Guidelines*, ISO/IEC 10918-1:1994 (E) (Geneva: International Organization for Standardization/International Electrotechnical Commission, 1994).
5. R. Ulichney, "Bresenham-style Scaling," *Proceedings of the ISE&T Annual Conference* (Cambridge, Mass., 1993): 101–103.
6. R. Ulichney, *Digital Halftoning* (Cambridge, Mass.: The MIT Press, 1987).
7. P. Bahl, P. Gauthier, and R. Ulichney, "Software-only Compression, Rendering, and Playback of Digital Video," *Digital Technical Journal*, vol. 7, no. 4 (1995, this issue): 52–75.
8. L. Seiler and R. Ulichney, "Integrating Video Rendering into Graphics Accelerator Chips," *Digital Technical Journal*, vol. 7, no. 4 (1995, this issue): 76–88.
9. R. Ulichney, "Video Rendering," *Digital Technical Journal*, vol. 5, no. 2 (Spring 1993): 9–18.
10. R. Ulichney, "The Void-and-Cluster Method for Generating Dither Arrays," *ISE&T/SPIE Symposium on Electronic Imaging Science and Technology*, San Jose, Calif., vol. 1913 (February 1993): 332–343.
11. *Industrial Electronics Tentative Standard No. 1, Color Television Studio Picture Line Amplifier Output Drawing* (Arlington, Va.: Industrial Electronics Association, November 1977). This publication is intended as a companion document to *Electrical Performance Standards—Monochrome Television Studio Facilities*, EIA-170 (Arlington, Va.: Industrial Electronics Association, November 1957).

## Biographies



### **Kenneth W. Correll**

After receiving a B.S.E.E. from the University of Washington in 1978, Ken Correll worked at Sperry Flight Systems for eight years, designing cockpit display systems. He joined Digital in 1986, focusing on the specification and design of advanced development projects concerned with the integration of live video and computer systems; he received a patent for some of this work. In 1990, Ken moved to Massachusetts and began work on the Jvideo and J300 projects. Since then he has been involved in the design of graphics ASICs in the Graphics and Multimedia Hardware Engineering Group within the Workstations Business Segment.



### **Robert A. Ulichney**

Robert Ulichney received a Ph.D. from the Massachusetts Institute of Technology in electrical engineering and computer science and a B.S. in physics and computer science from the University of Dayton, Ohio. He joined Digital in 1978. He is currently a senior consulting engineer with Digital's Cambridge Research Laboratory, where he leads the Video and Image Processing project. He has filed several patents for contributions to Digital products in the areas of hardware and software-only motion video, graphics controllers, and hard copy. Bob is the author of *Digital Halftoning* and serves as a referee for a number of technical societies, including IEEE, of which he is a senior member.