IMAGEVIEW™

A LOW-COST, EASY-TO-USE IMAGING SUB-SYSTEM

Lexidata's IMAGEVIEW™ is a powerful, low-cost imaging sub-system designed to off-load the host computer for processing, manipulating, and displaying image and pixel data. IMAGEVIEW offers functionality that until now has only been available on high-priced systems.

CT body scan (above) as displayed on the IMAGEVIEW image processing and display system.

FEATURES

- Affordable Combination of Hardware and Firmware
- Virtual Image System
- High-Level FORTRAN Host Command Library
- Wide Variety of Imaging Functions
- Display Processing Capability Using Lookup Tables
- Region of Interest Processing

BENEFITS

- Lowers OEM System Cost; Provides Fast, Flexible Sub-System
- Simplifies Management of Image and Pixel Data
- Speeds Program Development
- Provides Flexibility to Satisfy Specific Application Requirements
- Offers Real-Time Functionality That is Non-Destructive to Stored Data
- Provides "Selected Area" Capability for Specialized Applications

LEXIDATA™

SYSTEM OVERVIEW

The IMAGEVIEW system is based on a high-speed (224 nsec cycle time), microprogrammed display processor. The processor receives commands and data via a high-speed, 16-bit parallel DMA interface. The IMAGEVIEW Command Library is the interface between the programmer and the IMAGEVIEW display processor – its function is to format commands using the Virtual Image definitions stored in the host, and to check for errors in the programmer input. Since all of the computation-intensive pixel operations are performed locally in the IMAGEVIEW system, the host is free to handle other tasks.

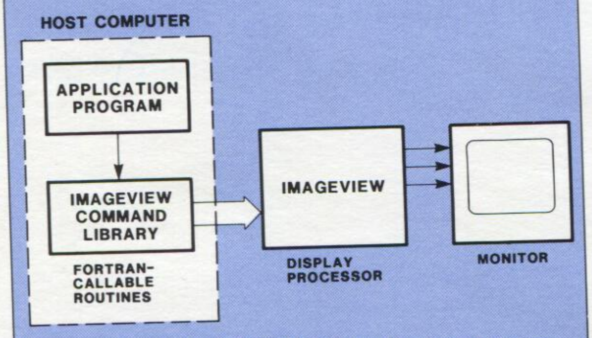

Figure 1. SYSTEM OVERVIEW – Application programs running on the user's host computer interface with IMAGEVIEW's easy-to-use command library, which formats and dispatches commands to the powerful display processor.

FAST, VERSATILE HARDWARE

Microprogrammed for Flexibility

While other systems provide real-time functionality using expensive specialized hardware, IMAGEVIEW provides the speed needed for interactive applications through a microprogrammed, Lexidata-proprietary bipolar processor. The key to this speed is the dual-ported memory architecture of the processor, which allows constant access to the display memory, even while data is being read out to the monitor. Since imaging applications require constant pixel manipulation, the IMAGEVIEW system provides a unique combination of speed (through fast hardware) and functionality (through firmware).

Variety of Configurations

IMAGEVIEW is a self-contained, rack-mountable unit, available in a wide variety of resolutions and configurations. Memory resolutions include 640x512 and 1280x1024. With optional memory, the 640x512 systems can display 4096 colors simultaneously; the 1280x1024 systems can display 256 colors. Both systems have an available palette of 16.7 million colors.

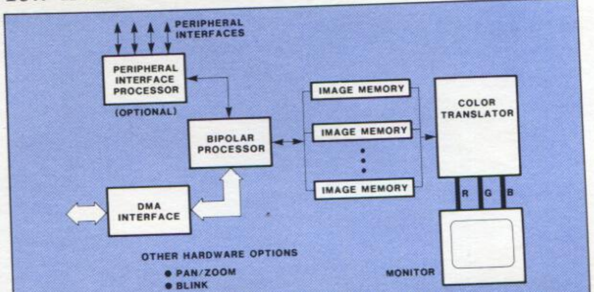

Figure 2. HARDWARE OVERVIEW – IMAGEVIEW's high-speed bipolar processor has direct access to image memory, and is microprogrammed to provide a wide array of features.

Options include a peripheral interface processor, which handles the available interactive devices – trackball, joystick or data tablet; a blink controller; and a pan/zoom controller for instantaneous, non-destructive zoom and smooth panning. A variety of color or greyscale monitors are available.

INDUSTRY STANDARD FUNCTIONS

IMAGEVIEW offers a wide variety of industry-standard functions that have previously been available only on high-priced imaging systems. Arithmetic and Boolean functions operate on the numerical value of the selected image. Logical functions allow the user to compare the selected image with another image or against a range of values.

FUNCTIONS	EXAMPLES	
Arithmetic	ADD	AVERAGE
Boolean	EXOR	AND
Logical	THRESHOLD	COMPARE
Spatial	COPY	INVERT
Statistical	HISTOGRAM	RANGE
Array	GRADIENT	VERTICAL DIFFERENCE

Figure 3. IMAGEVIEW Functions and Examples

Spatial functions modify the position of an image or selected pixels within an image, and statistical functions return data about the pixel values contained in an image. Array functions operate on groups of adjacent pixels. IMAGEVIEW also includes the high-speed text, vector and polygon fill functions which have become synonymous with Lexidata products.

VIRTUAL IMAGES™

Simplifies Use of Image Memory

IMAGEVIEW features a revolutionary method of partitioning the display memory into a series of independent workspaces called Virtual Images. This easy-to-use, high-level approach to data management enables the user to access the display memory with maximum efficiency. Applications which require the processing and display of several images will find this particularly useful. For example,

- The memory could be organized into several adjoining Virtual Images. This would allow side-by-side comparison of different images such as an original next to several processed images.
- The memory could be organized by planes. Several planes could be used for storing displayed images while the remainder are used for storing non-displayed images such as work images, characters, or frequently used symbols.

Allows User-Definable Workspaces

Using an IMAGEVIEW control function, the user defines Virtual Images as 5-word arrays stored on the host computer. This is accomplished by specifying the consecutive memory planes which the Virtual Image will occupy, the x,y position of the Virtual Image origin within the available memory, and the dimensions of the desired Virtual Image. Once defined, the Virtual Image is referenced by name; the user does not need to work with the absolute position of the image in memory. When performing operations on Virtual Images, the IMAGEVIEW Command Library uses the data stored in the host array to format commands which are checked for errors and then sent to the display processor. Since all of the computation-intensive pixel operations are performed locally in the IMAGEVIEW system, the host is free to handle other tasks.

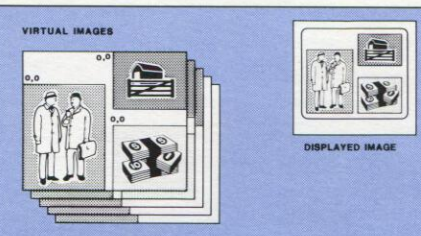

Figure 4. VIRTUAL IMAGES – IMAGEVIEW features a new, easy-to-use approach to image data management. The user can define any number of independent workspaces, called Virtual Images, and perform a variety of IMAGEVIEW functions directly on these separate images. Each of the images has its own name, origin, coordinate system, and number of bits. Any number of Virtual Images can be defined, since storage of the 5-word array needed for each one is done on the host. This allows greater utilization of available image memory, and reduces costs by simplifying applications software development.

Overlapping Virtual Images Are Handled Automatically

Processing can be carried out even when the source and destination Virtual Images overlap. IMAGEVIEW automatically keeps track of its position in memory to insure that data is processed in the proper order. This prevents loss of data and makes processing on overlapping Virtual Images a powerful and easy-to-use feature.

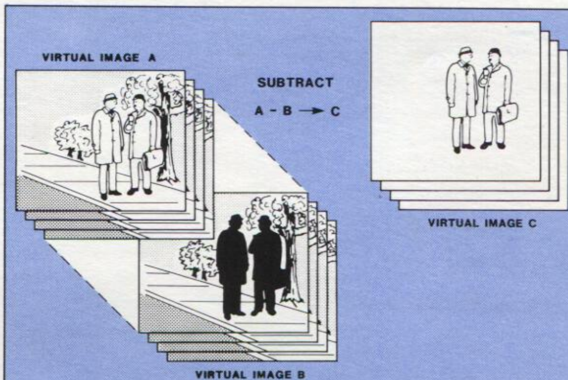

Figure 5. SUBTRACTION OF TWO VIRTUAL IMAGES – In this example, the goal is to remove the background. Here, Virtual Image A is the complete scene. Virtual Image B is the same, with the shadow of the men indicating a pixel value of zero. The subtraction yields zero everywhere except for the images of the men. The result is stored in Virtual Image C. Virtual Images A and B are unchanged.

VIRTUAL LOOKUP TABLES™

Virtual Lookup Tables™ extend the power of the Virtual Image concept. Just as Virtual Images allow the available image memory to be partitioned and accessed as needed, Virtual Lookup Tables allow the actual high-speed lookup table to be partitioned in a similar way. IMAGEVIEW handles all the processing to allow the user to work with these "multiple" lookup tables independently. The same 5-word array that defines the Virtual Image also defines its associated Virtual Lookup Table.

DISPLAY PROCESSING

IMAGEVIEW uses high-speed lookup tables (LUTs) for color translation and display processing. LUTs provide a real-time, interactive approach to certain imaging functions, since they operate directly on the data being presented to the display monitor. IMAGEVIEW offers two parallel sets of processing commands – those that work directly on the Virtual Image in memory (image processing), and those that work only on the Virtual Lookup Table associated with that Virtual Image (display processing). These

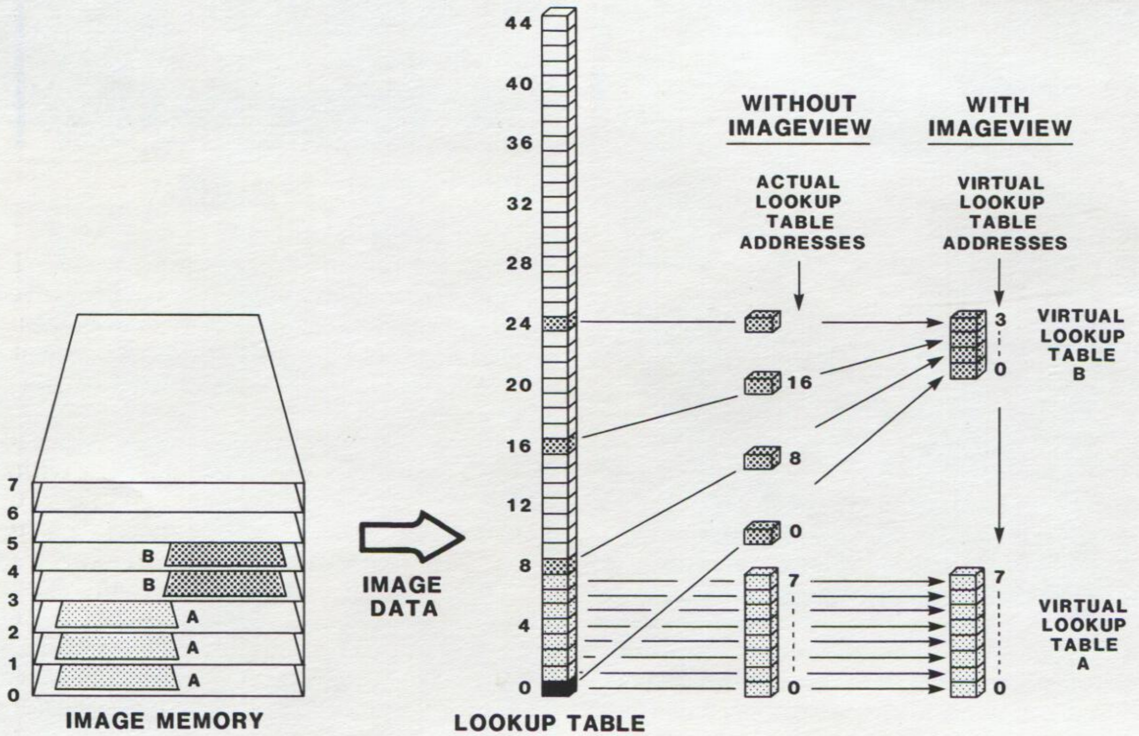

Figure 6. THE VIRTUAL LOOKUP TABLE APPROACH – Here, two Virtual Images are defined; Image A is on planes 0-2, and Image B is on planes 3-4. Loading the lookup table for Image A is straightforward, since Image A maps to sequential values in the lookup table. Without Virtual Lookup Tables, loading the lookup table for Image B would be very tedious, since Image B maps to non-sequential values (0, 8, 16, 24).

With Virtual Lookup Tables, however, IMAGEVIEW handles this second case as simply as the first, since a Virtual Lookup Table always appears to the user as a *sequential* list of addresses, starting from zero. The programmer need not be concerned with the actual addresses in the lookup table itself.

parallel functions achieve the same visual effect, but for different purposes.

IMAGEVIEW's display processing capability is ideal for interactive applications. For example, the user can threshold the displayed image interactively through the use of a trackball. Using display functions, an interactive search can be performed for the desired range of threshold values, without affecting the pixel values in the Virtual Image. Once the desired result is achieved, the image memory can be easily updated for storage or further processing, since each display processing function has a corresponding image processing function.

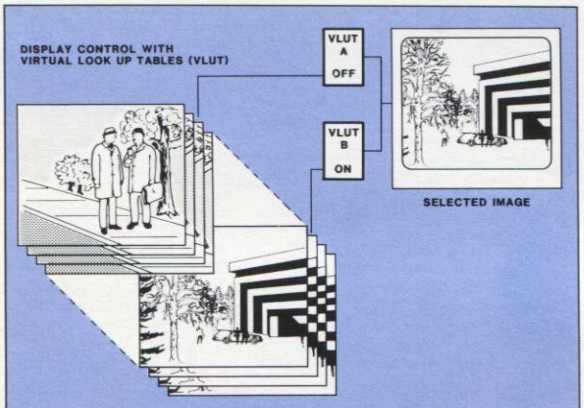

Figure 7. VIRTUAL LOOKUP TABLES SIMPLIFY DISPLAY CONTROL – When different Virtual Images occupy separate image planes, Virtual Lookup Tables (VLUTs) can be used to quickly select which image is displayed. A VLUT is turned "on" by loading it with the desired values. A VLUT is turned "off" by loading it with zeros. IMAGEVIEW's Virtual Lookup Table functions handle this task with ease.

Display Functions for Real-Time Processing

Display processing is also useful in applications where, for example, the user wants to compare several images with a stored template. The template is loaded in the upper planes with the "subtract" display function selected. The images to be compared are then downloaded into the lower planes. The difference between the image and the template is instantly displayed.

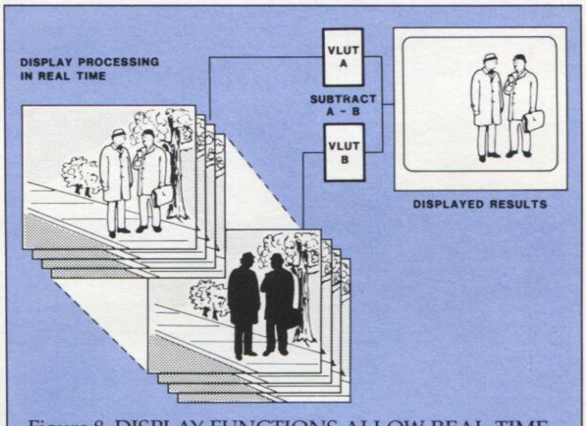

Figure 8. DISPLAY FUNCTIONS ALLOW REAL-TIME PROCESSING – Display functions provide powerful processing capability using only the lookup table values. Here, a subtraction is performed between two images by subtracting their Virtual Lookup Tables, leaving both images in memory unchanged. This example also points out how display processing functions parallel the image functions, since in Figure 5 the same result was achieved, but in that case, image memory was affected.

REGION OF INTEREST PROCESSING

Another powerful feature of IMAGEVIEW is the capability to process only a portion of an image, if desired. The user may designate a Region of Interest (ROI) and subsequent operations will only be performed within that area.

Region of Interest processing is achieved by associating another Virtual Image with the image to be processed. The ROI image then serves as a mask which indicates which pixels are to be processed. When a function is requested during ROI processing, only the pixels in the source image whose corresponding pixels in the ROI image have a specified value will be processed. All others will be left alone. The ROI image need not be "right above" the image to be processed; it can be resident anywhere in the system, since IMAGEVIEW will automatically "align" the image and ROI origins during processing. Furthermore, the ROI image can be a single shape, many shapes, or merely selected pixels.

ROI processing is ideal for interactive applications, since the user can "cut out" portions of a picture, or outline an area of an image to be enhanced. Since it works on a pixel-by-pixel basis, a region of any shape can be specified and regions made up of noncontiguous pixels can also be used.

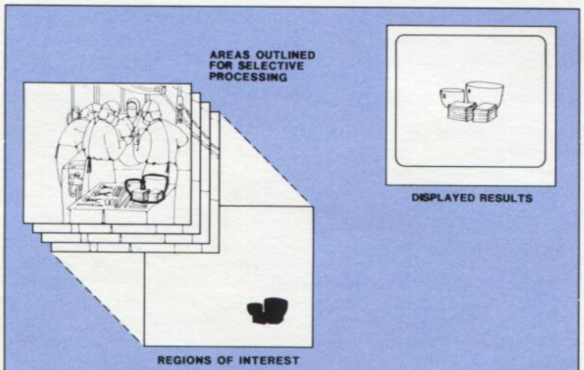

Figure 9. REGION OF INTEREST PROCESSING – This example shows just one of many ways that Region of Interest (ROI) processing can be used. In this case, the outline of the bowl and sponges is defined in the ROI plane. Then, with ROI processing enabled, a supersampling operation is performed, enlarging the defined image.

DOWNLOADABLE FONTS USING ROI PROCESSING

ROI processing is particularly useful for applications requiring downloadable fonts or logos. Each character can be set up as a binary-valued ROI image. Unlike traditional systems which merely copy the characters to the work area as a binary value, IMAGEVIEW allows the user to write characters into the work area in multiple colors. Thus, a single-bit font can be projected into several planes in one step, yielding a color character.

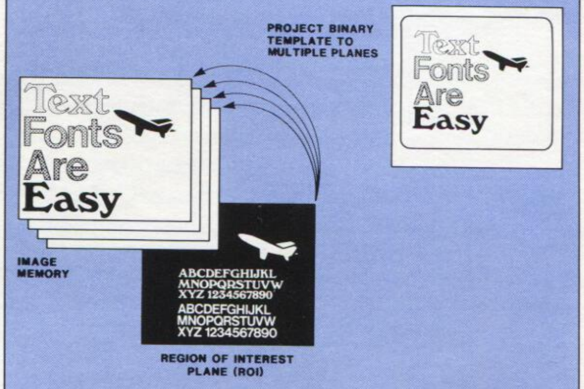

Figure 10. TEXT FONTS USING ROI PROCESSING – The ROI plane is used to store a binary "template" of each character. To place these characters into image memory, the user calls the "initialize" function and specifies the desired color. The result is a character with the same font as the one in the ROI plane, but in any color desired.

SPECIALIZED INPUT/OUTPUT FUNCTIONS

Most applications depend on efficient transfer of data between image memory and the host. Since different applications require different access methods, IMAGEVIEW provides a range of Input/Output (I/O) functions to satisfy most any application.

For example, in computing an image transform, the data must be accessed by rows, then by columns. This can be accomplished easily with standard IMAGEVIEW functions. Operations which require data from a "neighborhood" of pixels, such as image convolutions, can be performed easily by using block mode access.

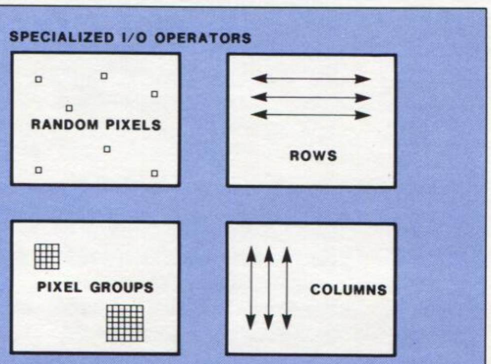

Figure 11. INPUT/OUTPUT ACCESS MODES – IMAGEVIEW's wide array of input/output modes simplifies application and algorithm development.

Input Processing Capability

IMAGEVIEW's I/O functions allow arithmetic and Boolean functions to be performed between the existing data in a Virtual Image and data being written in. This is useful when the Virtual Image is used for storing cumulative results such as the running average or differences between successive frames. Input processing eliminates unnecessary pixel access as well as the need to provide temporary storage for the input image.

SUMMARY

IMAGEVIEW provides a powerful base for the imaging system builder at a price/performance level unmatched in today's imaging marketplace. Its revolutionary Virtual Image approach to data management offers flexibility that allows the available memory to be used with maximum efficiency. IMAGEVIEW's powerful command set provides the depth of functionality needed for today's imaging applications.

IMAGEVIEW CONFIGURATION SUMMARY

Model 34IV –	**L081X-030/025**	**L081X-060/050**	**L081Z-030/025**	**L081Z-060/050**	**H041X-030/025**	**H041Z-030/025**
Display resolution	640x512	640x512	640x512	640x512	1280x1024	1280x1024
Display refresh rate	25/30Hz interlaced	50/60Hz non-interlaced	25/30Hz interlaced	50/60Hz non-interlaced	25/30Hz interlaced	25/30Hz interlaced
Standard image memory	8 planes	8 planes	8 planes	8 planes	4 planes	4 planes
Maximum image memory	12 planes	12 planes	12 planes	12 planes	8 planes	8 planes
Optional overlay memory	4 planes	4 planes	4 planes	4 planes	1 plane	1 plane
Color Lookup Table	12x8	12x8	12x8	12x8	8x8	8x8
Maximum number of simultaneously displayable colors	4096	4096	4096	4096	256	256
Pan/zoom controller	Not available	Not available	Standard	Standard	Not available	Standard
SCP peripheral interface processor/hardware cursor	Option	Option	Option	Option	Option	Option
Blink controller	Option	Option	Not available	Not available	Standard	Standard
Chassis	8 slot	8 slot	8 slot	8 slot	12 slot	12 slot

IMAGEVIEW COMMAND SUMMARY

All of the following IMAGEVIEW commands work directly with Virtual Images. Each Virtual Image is defined by a 5-word array stored on the host. The array name is the name of the Virtual Image, and also defines the name of the Virtual Lookup Table (VLUT) associated with that Virtual Image. Image processing functions work with pixels within a Virtual Image, while display functions operate on the Virtual Lookup Table associated with that Virtual Image. Display functions provide a convenient, non-destructive, real-time processing capability, since they only affect the values stored in the high-speed lookup tables. Thus, a display function can be used to quickly view the effect of a particular operation before committing it to pixel memory. Each display processing function has a parallel image processing function.

CONTROL FUNCTIONS

IMSET	Specifies the hardware configuration.
IMIMAG	Generates Virtual Image description.
IMROI	Enables/disables ROI processing.

INPUT/OUTPUT

Virtual Image access

IMRLIN	Reads sequential data (rows/columns) from an image.
IMWLIN	Writes sequential data (rows/columns) from an image.
IMRLMT	Defines a read window on an image.
IMWLMT	Defines a write window on an image.
IMRSEQ	Reads sequential data from a window.
IMWSEQ	Writes sequential data into a window.
IMRRAN	Provides random pixel reads on an image.
IMWRAN	Provides random pixel writes on an image.

VLUT access

ILREAD	Reads a block of data from an image.
ILWRIT	Writes a block of data into an image.

IMAGE PROCESSING

Arithmetic operations on Virtual Images

IMADD	Adds two images together.
IMSUB	Subtracts one image from another.
IMMULT	Multiplies two images together.
IMDIV	Divides one image with another.
IMAVG	Averages two images together (adds the two images together and then divides by 2).
IMADC	Adds a constant to an image.
IMSBC	Subtracts a constant from an image.
IMMLTC	Multiplies an image with a constant.
IMDVC	Divides an image with a constant.
IMINIT	Initializes an image with a constant.

Boolean operations on Virtual Images

IMOR	Performs an OR operation between two images.
IMAND	Performs an AND operation between two images.
IMEXOR	Performs an XOR operation between two images.
IMORC	Performs an OR operation between an image and a constant.
IMANC	Performs an AND operation between an image and a constant.
IMEXC	EX-ORs an image with a constant.
IMNEG	Negates an image (1's complement).

Logical operations on Virtual Images

IMBOOL	Compares an image with a comparison window. Constant values are written into the output image for true and false comparisons.
IMCMPR	Compares two images. A constant value is written into the output image for each true comparison and the original value for each false comparison.
IMTRSH	Compares an image with a comparison window. A constant value is written into the output image for true comparisons and the original pixel value for each false comparison.

Spatial operations on Virtual Images

IMCOPY	Copies one image into another.
IMMOVE	Copies one image into another and writes a constant value into the old image.
IMRVRS	Reverses and copies one image into another.
IMINVR	Inverts and copies one image into another.
IMTRSP	Transposes and copies one image into another.
IMSWAP	Swaps one image with another.
IMSAMP	Magnifies and minifies an image by integer amounts.

Array operations on Virtual Images

IMHDIF	Computes the horizontal pixel difference between pixels on adjacent lines.
IMVDIF	Computes the vertical pixel difference between pixels on a line.
IMGRAD	Computes the magnitude of the gradient.

DISPLAY PROCESSING

Arithmetic operations on VLUTs

ILADD	Adds two VLUTs together.
ILSUB	Subtracts one VLUT from another.
ILMULT	Multiplies two VLUTs together.
ILDIV	Divides one VLUT with another.
ILAVG	Averages two VLUTs together (adds the two VLUTs together and then divides by 2).
ILADC	Adds a constant to a VLUT.
ILSBC	Subtracts a constant from a VLUT.
ILMLTC	Multiplies a VLUT with a constant.
ILDVC	Divides a VLUT with a constant.
ILINIT	Initializes a VLUT with a constant.
ILRAMP	Initializes a VLUT as a series of linear segments.

Boolean operations on VLUTs

ILOR	Performs an OR operation between two VLUTs.
ILAND	Performs an AND operation between two VLUTs.
ILEXOR	Performs an XOR operation between two VLUTs.
ILORC	Performs an OR operation between a VLUT and a constant.
ILXOR	Performs an XOR operation between a VLUT and a constant.
ILEXC	EX-ORs a VLUT with a constant.
ILNEG	Negates a VLUT (1's complement).

Logical operations on VLUTs

ILBOOL	Compares a VLUT with a comparison window. Constant values are written into the output VLUT for true and false comparisons.
ILCMPR	Compares two VLUTs. A constant value is written into the output VLUT for each true comparison and the original value for each false comparison.
ILTRSH	Compares a VLUT with a comparison window. A constant value is written into the output VLUT for true comparisons and the original pixel value for each false comparison.

STATISTICAL

Operations on Virtual Images

IMRANG	Determines the range of the gray levels within an image.
IMHIST	Generates histogram of gray level occurrence.
IMSTAT	Obtains statistical data on pixels within an image.

OTHER COMMANDS

The following graphic and system commands are not specific to IMAGEVIEW functionality, but provide important additional capability. These functions work in screen coordinates.

Initialization

DSOPN	Opens channel to the display.
DSPLD	Restarts the display.
DSCHAN	Enables plane to accept text and graphics data.
DSCLS	Closes channel to the display.
DSCLR	Clears the screen.
DSOWT	Wait for data channel output.
DSIWT	Wait for data channel input.

Graphics functions in screen coordinates

DSBLOC	Send a block of graphics commands.
DSBLR	Read a block of graphics data.
DSVEC	Draw vector.
DSSFIL	Polygon seed fill.
DSDISP	Set write mode.

Text functions in screen coordinates

DSSAO	Set alphanumeric display parameters.
DSTXT	Display text.

Pan/zoom functions in screen coordinates*

DSMRG	Set display margins.
DSZOM	Set zoom parameters.
DSMOV	Display movie.

Blink controller functions*

DSBLIN	Set blink rate.
DSBCTL	Set blink mode.

Hardware cursor functions in screen coordinates*

DSCLD	Load matrix cursor.
DSCER	Erase matrix cursor.
DSCXY	Position cursor.
DSCSL	Select hardware cursor type.

Peripheral control functions in screen coordinates*

DSGXY	Read trackball/joystick position.
DSGKB	Get data from keyboard.
DSSL	Send light data to joystick/trackball.
DSITAB	Set tablet parameters.
DSRTAB	Read tablet data.

*Supported with optional hardware only.

IMAGEVIEW SYSTEM SPECIFICATIONS

Alphanumerics
An alphanumeric character font is provided which supports upper case text, numerals, and punctuation. The font size is 5x7 pixels in a 7x9 field.

Cursor
Size and shape of non-destructive cursor is user-loadable within 64 pixel by 64 pixel matrix. Full screen cross-hair is also selectable.

Data Transfer Rate
Up to one megaword (16 bits/word) per second from host computer in burst mode.

Power Requirements
115 VAC ± 10% 47-63Hz (3 wire)
230 VAC ± 10% 47-63Hz (3 wire)
600 watts average.
Requirements vary depending on configuration size.

Environmental Requirements
Operating Temperature
10 to 40 degrees C
Storage Temperature
−35 to 70 degrees C
Operating Relative Humidity
10% to 90% (non-condensing)
Storage Relative Humidity
10% to 90% (non-condensing)
Altitude
8,000 ft.
Acoustic Noise Level
The acoustic noise level shall not exceed the NC-60 noise criteria curve.

Chassis Dimensions
8-slot: 5.25″ high x 19″ wide x 27″ deep.
12-slot: 8.75″ high x 19″ wide x 27″ deep.

Weight
8-slot: 40-70 lbs. including power supply.
12-slot: 60-100 lbs. including power supply.

LEXIDATA

CORPORATE WORLD HEADQUARTERS

Lexidata Corporation
755 Middlesex Turnpike
Billerica, MA (USA) 01865
(617) 663-8550
TWX: 710-347-1574

U.S. offices:

Boston
Chicago
Houston
Ithaca
Los Angeles
New York
Raleigh
San Francisco
Washington, D.C.

EUROPEAN HEADQUARTERS

Lexidata, LTD
Weaver House
Station Road
Hook, Basingstoke
Hampshire
RG27 9JY
UNITED KINGDOM
(44) (025672) 4334
TWX: 851/858364 LEXUK G

International offices:

Australia
France
Israel
Japan
Netherlands
Switzerland
United Kingdom
West Germany

LEXIDATA FRANCE SARL
30, Rue du Morvan SILIC 557
94643 RUNGIS Cédex
Tél. (1) 686 56 71
Telex 204 683 LEXI-FR

Product specifications subject to change without notice due to continuing research and development.
™LEXIDATA is a trademark of the Lexidata Corporation.
™IMAGEVIEW, Virtual Images, and Virtual Lookup Tables are trademarks of the Lexidata Corporation.
Cover application photo courtesy of Dr. Samuel J. Dwyer III, Dept. of Diagnostic Radiology, University of Kansas.

Printed in USA 6/83
No. 3950-301-PD