SOLIDVIEW™

NOW YOU CAN BUILD IN SECONDS... WHAT USED TO TAKE FOREVER!

Lexidata's SOLIDVIEW™ is a revolutionary new technology for the display of solid objects. It is the fastest, most advanced system of its kind. Featuring a special combination of hardware and firmware, SOLIDVIEW achieves its superior performance by uniquely providing local hidden surface removal and local visible surface shading. The result is a solid modeling display system fast enough for interactive applications at an affordable cost.

Piping design (above) as displayed on the SOLIDVIEW solid modeling display system.

FEATURES

- Handles Hidden Surface Removal and Visible Surface Shading Locally
- Handles Piercing Objects Automatically
- Employs 3-D Graphics Primitives
- Supports Popular Shading Techniques
- Translucent Surface Mode

BENEFITS

- Offloads the Host Computer
- Requires No "Special" User Processing
- Provides a High-Level Interface to Application Programs; Reduces Host I/O
- Offers Flexibility for Varied Applications
- Allows Interior View of Object; Detects Surface Interference

LEXIDATA™

SOLIDVIEW REVOLUTIONIZES DISPLAY OF SOLID OBJECTS

Works with 3-D Data

In addition to a powerful 2-dimensional graphics command set, SOLIDVIEW accepts 3-dimensional data types, including 3-D polygons, vectors, horizontal segments, and points. Each data type can have constant or variable shading, color, or normal attributes. A powerful 3-D graphics command set displays these data types with hidden surfaces removed and visible surfaces shaded. Additional 3-D commands provide automatic sectioning, allow 3-D cutting planes, and provide visible surface picking and 3-D cursors.

Lexidata's SOLIDVIEW provides features which will improve any graphics application dealing with the display of 3-D objects. SOLIDVIEW increases system speed – reducing host overhead, software design time, and total system cost.

Local Hidden Surface Removal and Visible Surface Shading

To understand the power of SOLIDVIEW, compare the SOLIDVIEW approach with standard approaches used today. Using the traditional approach, object data from the host computer data base is processed completely by the host and output as pixels to the display processor. The host has to transform and clip the object into a viewing volume, remove hidden surfaces, and calculate pixel values for each visible surface. Usually the entire image has to be created at the host before pixel data is sent to the display processor. The result is a long delay before the first pixel is sent, and then the image appears one scan line at a time. The display processor acts only as a frame buffer to store the resultant image. This approach is heavily unbalanced and host intensive. Even after much computation by the host, the image appears slowly since it is transmitted pixel by pixel.

As shown in Figure 1, SOLIDVIEW provides a new, faster approach. While the host still performs the viewing transformation and volume clipping, the rest of the work is done by SOLIDVIEW. Working from 3-D polygon data that includes shades at the polygon's vertices, SOLIDVIEW removes hidden surfaces and displays visible surfaces with smooth shading interpolated from the polygon data. Since these operations are done in SOLIDVIEW's bipolar processor, they are completed at speeds far greater than a host computer could manage.

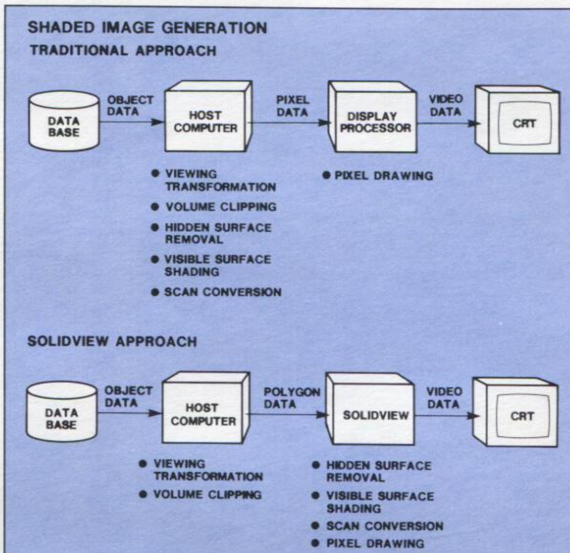

Figure 1. SHADED IMAGE GENERATION – Traditional solid modeling systems are host computer intensive and generally too slow for interactive applications. SOLIDVIEW provides local power which off-loads the host and balances the workload. The result is a faster, simpler, more powerful system that allows the user to create solid objects in a truly interactive environment.

Works with Host in Parallel to Increase Throughput

Besides providing local computing power, especially important for the display of solid objects, SOLIDVIEW does it in a way which provides for parallel processing with the host computer. As soon as the first polygon is ready for display, it is output to SOLIDVIEW and displayed immediately. Figure 2 compares SOLIDVIEW with conventional approaches, and shows quite graphically the time benefit achieved by this parallel processing.

As shown in Figure 2, SOLIDVIEW provides a completed picture in much less time, with less computation in the host, and the first part of the picture begins to appear almost immediately. SOLIDVIEW's combination of local computing power and parallel processing provide the speed needed to make interactive applications using solid objects a reality. And SOLIDVIEW does it at a price that is surprisingly low.

Cuts Host I/O Burdens

The polygon data structure used by SOLIDVIEW encodes information in a compact form. This results in reduced loading of the host I/O channels, which provides greater throughput, and allows other tasks to run without degradation.

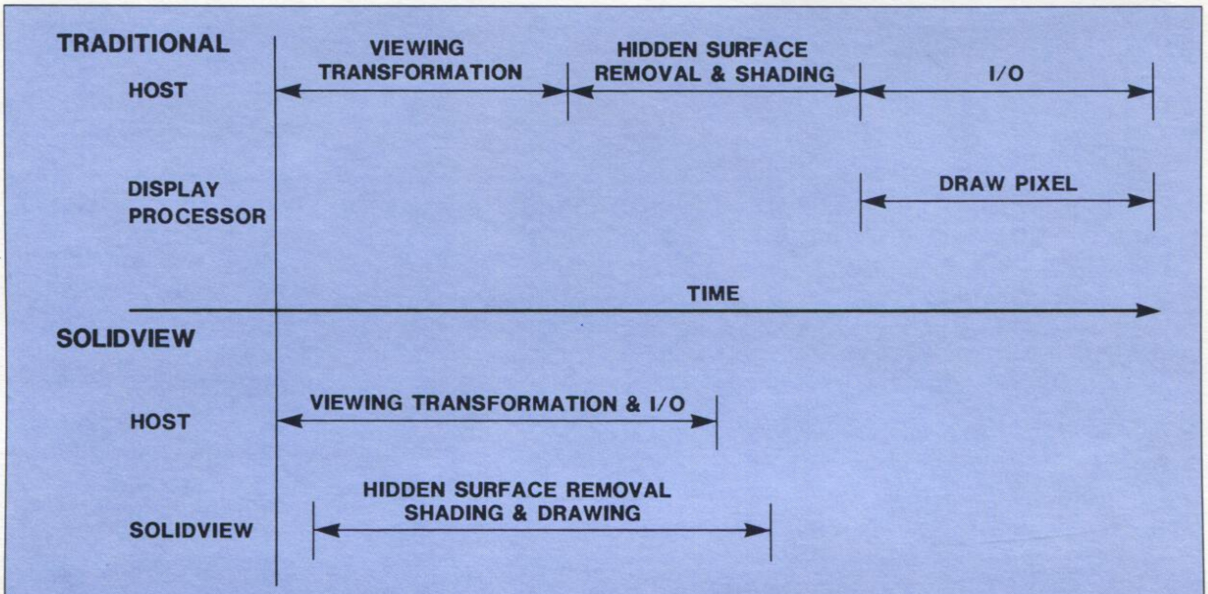

Figure 2. PERFORMANCE COMPARISONS: TRADITIONAL VS. SOLIDVIEW – Traditionally, the entire image is constructed at the host computer and then sent pixel by pixel to the display processor. Not only is this process slow, it is I/O and host intensive, and requires large amounts of memory. With SOLIDVIEW, the workload is shared evenly with the host. As soon as the first polygon is computed, it is sent to the SOLIDVIEW system where hidden surfaces are removed, visible surfaces are smoothly shaded, and the result is displayed. The response is virtually immediate, and the solid model builds in a visually useful sequence.

Cuts Data Storage Requirements in the Host

Since SOLIDVIEW can handle 3-D data directly, the host does not have to use hidden surface removal and visible surface shading algorithms which typically require pixel-by-pixel generation of the entire image before transmission to the display processor. Thus the need for host memory commensurate with the displayed image (often larger than 1 Mbyte) is eliminated. Smaller memory requirements eliminate a major host burden. For example, based on 10x10 pixels per polygon, the host requires 100 words of storage per polygon. With SOLIDVIEW, the same information is stored in only 18 words.

Supports Three Shading Techniques

1) Constant (Polyhedron) Shading for Maximum Speed

Each polygon is given constant shading. This requires less computation for the host and for SOLIDVIEW.

2) Intensity Interpolation (Gouraud) Shading for Smoothness

Polygons are transmitted by the host with intensity values for each vertex. The result is smooth shading with minimal host calculation. Since SOLIDVIEW removes hidden surfaces automatically, this operation can be performed on each polygon as soon as it is transmitted by the host. The result is parallel processing by the host and SOLIDVIEW, and the smoothly shaded solid object is typically complete in a few seconds.

Since each polygon is shaded as it is drawn, and subsequent polygons hide areas behind them, the order of drawing provides important information to the viewer. Thus, even the few seconds it takes to create the object provides important information.

3) Normal-Vector Interpolation (Phong) Shading for Flexibility

This approach provides the same smooth shading as the intensity interpolation, but the host provides two values to be interpreted for each vertex of the polygon. While this approach requires slightly more calculation by the host and by SOLIDVIEW, it now becomes possible to vary the light source position by changing the lookup table values. By pre-computing a sequence of lookup table values, and then transmitting them in rapid sequence to SOLIDVIEW, the light source can appear to move smoothly across the shaded image, providing additional detail about the part being viewed. This effect could easily be controlled interactively, again increasing viewer understanding of the image on the screen.

Lets You "See Inside" a Solid Model

As each polygon is sent to SOLIDVIEW, hidden surfaces are removed and visible surfaces of the new polygon hide the surface behind them. At the programmer's option, however, SOLIDVIEW can use a translucent pattern rather than an opaque shade. Since the translucent pattern only writes selected pixels, some of the information behind the added polygon is still visible. This can be thought of as a "screen door effect," and by proper specification of the translucent pattern, important details inside the solid model can remain visible even as other parts are added to the view. Once again, SOLIDVIEW off-loads a tedious task from the host and provides important viewing information to the user.

UNIQUE APPLICATIONS BECOME POSSIBLE WITH SOLIDVIEW

SOLIDVIEW's powerful features make new, interactive applications feasible for the first time. The following highlights just some of the possibilities.

3-D Cursor Interacts with Solid Objects

An interactive 3-D application requires a 3-D cursor, a 3-D input device, and fast visual response at the graphic display. Working with the host, SOLIDVIEW makes this possible. For example, to implement a 3-D cursor, the host might use an overlay plane to create a mesh pattern which pierces the solid model. As the cursor is moved in x, y, and z, SOLIDVIEW can instantly determine which parts of the cursor mesh are visible. The user receives instantaneous 3-D visual feedback as he works with the solid object on the display.

Creates Sectional Views Interactively

The host can define a cutting plane which SOLIDVIEW will incorporate as it determines which surfaces are visible. Only objects or parts of objects behind the cutting plane will be displayed by SOLIDVIEW. The host simply defines the cutting plane, and re-outputs the same polygon data that created the model. The rest is automatic, and the section view is complete in seconds. A new sectional view can be created from the same information simply by changing the cutting plane; the host does not get involved with the actual sectioning. With SOLIDVIEW the process is so fast and simple that interactive sectioning can be implemented easily.

Shows Piercing Objects with No Host Overhead

With SOLIDVIEW, the display of piercing objects is automatic, regardless of the data structure and algorithms used by the actual solid modeling software. SOLIDVIEW handles each pixel independently and shows piercing effects without host overhead or special consideration.

SUMMARY

SOLIDVIEW's local hidden surface removal and visible surface shading capabilities greatly reduce host computer loading and provide the speed needed for interactive solid modeling applications. SOLIDVIEW's special features add speed and flexibility to any application involving 3-D images, and its ability to work in parallel with the host provides instantaneous feedback for the user.

SOLIDVIEW COMMAND SUMMARY

A full command set is provided for the initialization, control, and loading of the SOLIDVIEW display processor. Additionally, SOLIDVIEW features a complete set of powerful 3-D graphics functions, as well as a number of 2-D commands.

INITIALIZATION

DSCFG	Sets up for hardware configuration.
DSCHAN	Selects display channels.
DSCLR	Clears display.
DSIWT	Waits for data channel input.
DSOPN	Opens display to commands.
DSCLS	Closes display to commands.
DSOWT	Waits for data channel output.
DSPLD	Loads program.

2-D & 3-D GRAPHICS FUNCTIONS

DSBLOC	Sends data block to display.
DSBLR	Gets data block from display.
DSCIR	Displays a 2-D circle.
DSDISP	Sets display parameters.
DSVEC	Displays a 2-D vector.
DSBUFF	Selects which of the buffers will be used for reading and writing.
DS3MOD	Sets 3-D drawing mode.
DSPATT	Loads an 8 x 8 translucency pattern.
DS3PNT	Draws a 3-D point.
DS3SEG	Draws a 3-D constant shaded segment.
DSSSEG	Draws a 3-D smooth shaded segment.
DS3POL	Draws a 3-D constant shaded polygon.
DSSPOL	Draws a 3-D smooth shaded polygon.
DS3VEC	Draws a 3-D constant shaded vector.
DSSVEC	Draws a 3-D smooth shaded vector.

TEXT FUNCTIONS

DSSAO	Sets alphanumeric display parameters.
DSTXT	Displays text.

PAN/ZOOM FUNCTIONS*

DSMOV	Displays movie.
DSMRG	Sets display margins.
DSZOM	Displays zoom.

HARDWARE CURSOR FUNCTIONS*

DSCER	Erases matrix cursor.
DSCLD	Loads matrix cursor.
DSCSL	Selects hardware cursor.
DSCXY	Sets cursor position.

PERIPHERAL CONTROL FUNCTIONS*

DSGKB	Gets data from keyboard.
DSITAB	Initializes data tablet.
DSRTAB	Reads tablet data.
DSSL	Sends light data to joystick.
DSG3D	Reads data from 3-D joystick.

LOOKUP TABLE FUNCTIONS

DSLLU	Ramps lookup table.
DSLRD	Reads lookup table block.
DSLWT	Writes lookup table block.

IMAGE FUNCTIONS

DSGET	Gets data from display.
DSLIM	Selects rectangular limits.
DSPUT	Sends data to display.

*These functions are not available on Model 34SV-1.

SOLIDVIEW CONFIGURATION SUMMARY

Model	34SV-1	34SV-2
Display resolution	640x512	640x512
Display refresh rate	25/30Hz Interlaced	25/30Hz Interlaced
Standard display memory	12 planes	12 planes
Standard depth buffer memory	12 planes	12 planes
Overlay memory	Not Available	4 planes
Color lookup table	12x8	12x8
Maximum number of simultaneously displayable colors	4096	4096
SCP peripheral interface processor/hardware cursor	Not Available	Standard
Pan/zoom controller	Not Available	Standard
Host interface	Parallel	Serial or Parallel
Chassis	8 slot	12 slot
Monitor (19″ color in-line LP)	Standard	Standard

SOLIDVIEW 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.

Average Polygon Write During Shading and Hidden Surface Removal
1000 polygons/second

Data Transfer Rate
Parallel interface – Up to one megaword (16 bits/word) per second from host computer in burst mode.
Serial interface – Switch selectable to rates up to 19.2K baud.

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

Printed in USA 6/83
No. 3950-101-PD