FAIRCHILD AIMS TO MOVE HIGH-END GRAPHICS TO PCs

It has a new concept in frame-buffer architecture that can be implemented with garden-variety DRAMs, yet provides the speed needed for 3-d image drawing

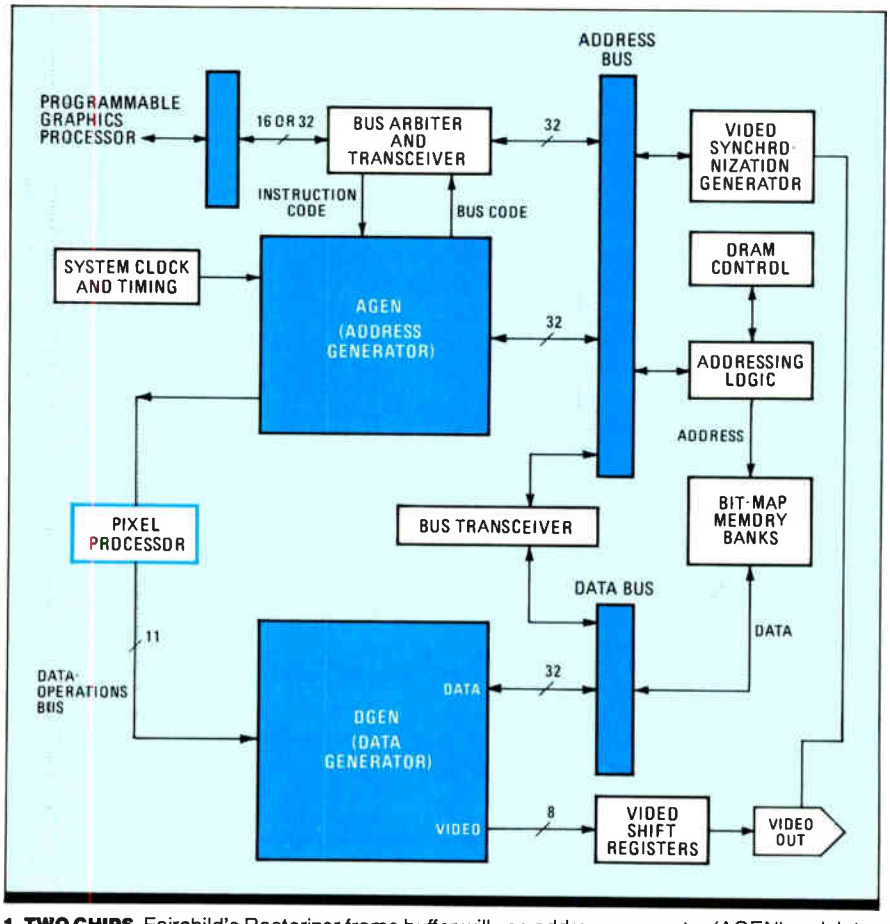
by Samuel Weber

Desktop computer graphics, long mired in the flatness of a two-dimensional world, is on the threshold of the third dimension thanks to a new concept in frame-buffer architecture. Fairchild Semiconductor Corp. has devised a low-cost, reconfigurable frame-buffer architecture having sufficient bandwidth to provide a fast enough read/write access time for 3-d graphics on personal computers. Called the Rasterizer, the Fairchild hardware dramatically slashes the time needed by the image memory to refresh the screen of a cathode-ray tube. The Cupertino, Calif., company is now working on the two integrated circuits that are the key to the new architecture.

The Rasterizer's architecture will be inexpensive enough for personal computers because it can be implemented with garden-variety dynamic random-access memories. Such a frame buffer, capable of performing primitive 3-d graphics, will cost no more than current 2-d implementations, Fairchild says. However, it's also possible to use faster RAMs for even higher performance, and so the new architecture can be applied to graphics applications on all levels, not just to PCs.

But it is the reconfigurability that gives the Rasterizer its speed: the new ICs can dynamically reconfigure the RAMs that make up the frame buffer to the most efficient structure for each type of line or shape. When populated with 100-ns DRAMs, the Rasterizer can operate at a rate of 15 million pixels/s for drawing vectors (screen refresh included). This is an average of 7.27 pixels/s per write cycle and a peak of 16. If static RAMs replace DRAMs, the average vector-drawing rate rises to between 20 and 25 million pixels/s.

Translating pixel drawing rates to wire-frame vector-writing rates, the Rasterizer populated



1. TWO CHIPS. Fairchild's Rasterizer frame buffer will use address-generator (AGEN) and data-generator (DGEN) chips: one AGEN per system and one DGEN for each memory plane.

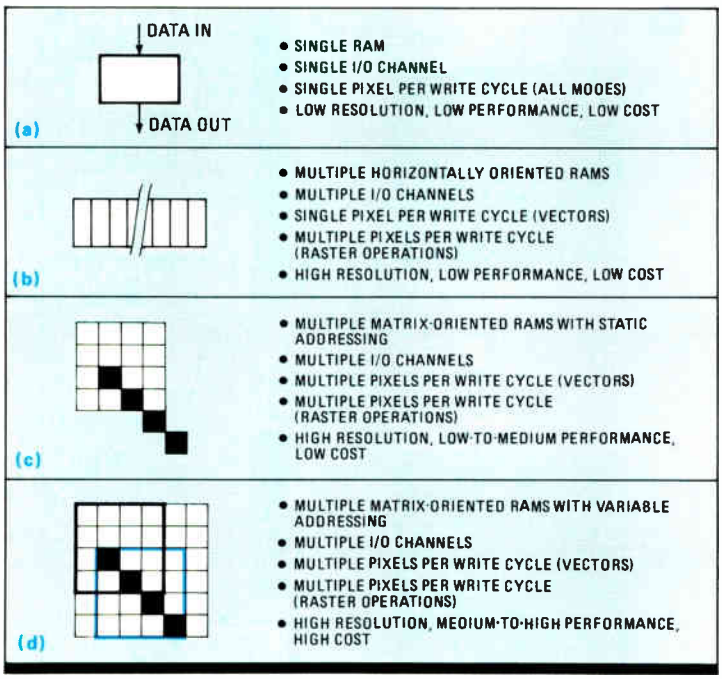
with garden-variety 100-ns DRAMs can write vectors to the screen at a rate of 1 million/s. The speed of the typical high-end 3-d systems is between 100,000 and 200,000 vector/s.

The Rasterizer architecture also speeds through other graphics operations: 80 million pixels/s for a bit-block transfer, 80 million pixels/s for filled polygons, and 45,000 characters/s for text. These are average, not peak speeds—peak speeds range from double to triple the average figures.

In computer graphics, video-output devices perform such back-end operations as digital-to-analog conversion, selecting colors from the available pallette, and display management. In all this, the key to high performance is in the frame buffer and its control. The frame buffer, also known as a bit map or image memory, stores all pixel information needed to describe images on standard raster-display CRTs.

The immediate impact of the Rasterizer frame-buffer design will be felt on wire-frame 3-d graphics drawn on a screen with vectors that are simply straight lines and faceted shading, says Michael L. Fowler, advanced product planning manager at Fairchild's digital and analog division in South Portland, Me. Many 3-d CAD/CAM applications ranging from electrical schematics to mechanical drafting use wire-frame images rather than Gouraud or Phong shading.

A wire frame describes the edges and envelope of an object's geometry. It is essentially a collection of polygons whose boundaries are defined by vectors—the entire area within a polygon need not be filled in. "But the chips we are planning for this architecture will provide hardware and software interfaces or 'hooks' for implementing displays with more realistic shading techniques, such as Gouraud or Phong shading," he says.



2. FOUR GENERATIONS. The first frame buffer was just a single RAM (a). Next step (b) was to gang RAMs together to generate multiple pixels during each write cycle. The third generation (c) opened the way to efficient writing of vectors while the fourth generation (d) speeds the vector writing process for 3-D graphics.

The Rasterizer hardware (see fig. 1) will span a range of applications, from low-cost personal computers to sophisticated high-end work stations. It implements four fixed graphics functions applicable to 2-d and 3-d systems: vector, polygon, bit-blt (bit block transfer) and char (characters). These functions are hard-wired to provide high-speed operation.

The heart of the Rasterizer concept is two very large-scale ICs, the address generator (AGEN) and data generator (DGEN). The AGEN is the element that controls frame buffer addressing. In the simplest Rasterizer configuration, a single AGEN controls all the frame-buffer memory planes, and a single DGEN modifies frame-buffer data and composes this data for video output. Fairchild plans to implement these chips in 1.5-μm advanced CMOS; they should be available early next year.

For high-performance graphics applications, the new Fairchild frame buffer can be populated with video random-access memories, but for the majority of microcomputer graphics, standard dynamic RAMs with 100-ns access times are adequate. For the highest-performance graphics systems, the Rasterizer can use fast CMOS static RAMs, now offering 25-ns access times.

Even with the fastest RAMs, most low-cost frame buffers used in 2-d graphics subsystems are inefficient at drawing 3-d wire frames. The way in which these buffers store 2-d pixel data, a method called word mode, works best for generating horizontal lines. "But lines are rarely horizontal; in most cases, they cut across the screen at an angle," Fowler points out. "So the frame buffer must read out data for as many horizontal lines as a vector cuts through before completing a single vector."

The data for the entire horizontal line must be read even though just a single pixel of information is required on each line. In effect, this amounts to drawing a single pixel for each write cycle. But such an operation slows the system's writing speed, making conventional frame buffers inefficient in 3-d applications.

To speed the writing process, two approaches compete: reorganizing the storage of data in the frame buffer or speeding up the graphics hardware. A reorganized buffer scheme is attractive because it hikes the rate of obtaining the data needed to draw vectors—so it increases the bandwidth efficiency of the graphics systems without the costly tactic of adopting sophisticated hardware.

A reorganized frame buffer usually can optimize a graphics system for the relatively low bandwidth needed for 3-d wire-frame graphics. Because wire-frame graphics involves envelopes

rather than filled-in areas, it can be handled by an effective bandwidth in the range of 10 to 20 million pixels/s. Of course, for maximum writing speed, both a reorganized frame buffer and enhanced hardware can be adopted.

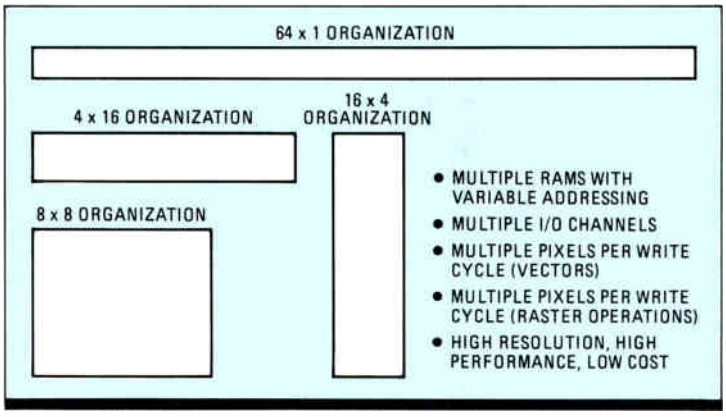
Today's complex frame-buffer designs are a far cry from the first crude attempts to use a single memory chip to store information for graphics displays. This lone DRAM (see fig. 2a) is, in reality, a graphics memory with a single input/output channel and an output of a single pixel per write cycle. It has extremely poor resolution, and its speed is so slow as to prevent the actual display of information on a CRT.

The second generation of graphics memories was a chain of DRAMs (see fig. 2b). In this multiple-I/O-channel scheme, multiple pixels can be written to the screen during each write cycle. This configuration is suitable only for writing pixels horizontally. Resolution increases with the number of DRAMs in the chain—but system performance remains relatively poor since this scheme can still draw only one pixel per line scan. For vector operations, second-generation frame buffers can generate about 1 to 2 million pixels/s.

Third-generation architecture is identified with the concept of a 2-d cell: a matrix of DRAMs configured for graphics (see fig. 2c). In a 2-d system, a cell can be a 4-by-4 array of DRAMs, an 8-by-8 array, or an asymmetrical arrangement, and it can generate data for multiple pixels during each write cycle. Each RAM chip within a cell is assigned to a specific pixel location on the screen, a technique called static addressing. The cell concept permits high-performance images to be produced at low system cost: a third-generation frame buffer operates at rates in the range of 6 million pixels/s. But this type of cellular graphics is not well-suited to such raster operations as refresh, bit-blt, and polygons because it, too, can draw only one pixel per line scan.

The cell concept is taken a step further in fourth-generation frame-buffer architecture (Fig. 2d), where the address of each RAM chip is variable. This allows a given cell to be assigned to various pixel locations on the screen. The fourth-generation architecture can write vectors to the screen at typical speeds of 10 to 20 million pixels/s. The drawback—particularly for microcomputer-based systems—is its high cost, a result of the overhead needed to support variable addressing of DRAMs within a cell.

All of the benefits embodied in third- and fourth-generation architectures are incorporated in the Rasterizer fifth-generation frame buffer (see fig. 3). However, this architecture goes further in terms of optimum flexibility. Through a combination of the Rasterizer's chip-set design and its built-in algorithms, the cell orientation can be dynamically configured four different ways, depending on the type of vector to be drawn. To draw horizontal vectors, a 4-by-16 array of RAMs is the most efficient since it delivers the maximum number of pixels (16) per write cycle. Similarly, to draw vertical vectors, a 16-by-4 array optimizes performance by drawing 16 vertical pixels per write cycle. And when vectors must be drawn at an angle—cutting through horizontal lines—the 8-by-8 array provides the best results. Finally, the word mode, which was lost in the previous two generations, is reinstituted in the Rasterizer in the form of a 64-by-1 array of RAMs. □



3. RECONFIGURABLE. The fifth-generation Rasterizer frame buffer dynamically organizes memory into various array sizes.

For more information, circle 481 on the reader service card.

HOW FAIRCHILD FOUND A NICHE IN GRAPHICS ICs

When planners at Fairchild Semiconductor Corp. decided to jump into the graphics market about 2½ years ago, they recruited Michael L. Fowler to define a graphics strategy for the company. Fowler's first conclusion was a negative one: it was already too late for the Cupertino, Calif., company to come into the market with a graphics engine, unless it could bring something special to the party.

So he set about talking to graphics systems designers to find out what they wanted. "The thing almost everyone wanted most was flexibility of the system interface," he says. "They wanted to be able to match the graphics engine easily to the specific system they were designing." His recommendation was to concentrate on optimizing the performance of the frame buffer.

It took two years to come up with the reconfigurable architecture embodied in the Rasterizer. When it's fully implemented, Fowler says, system designers will be able to achieve a wide range of graphics performance levels—up to 10 times that of currently available sytems, without any need for special high-speed memories.

Fowler, 34, has a BSEE from Southwestern Louisiana University, He had almost six years of graphics-design experience behind him—as a work-station system designer, then a hardware design manager at now-defunct Phoenix Computer Graphics in Lafayette, La., and at Day Telecommunications in Raleigh, N. C. He is now advanced product planning manager for Fairchild's digital and analog unit.

MICHAEL L. FOWLER