IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, May 9-10, 1991

# A High Performance Parallel Computing System for Imaging and Graphics

**H.W. Park**, Ph.D., **T. Alexander**, Ph.D., **S.H. Moon***, MSEE, **and Y. Kim**, Ph.D.
Image Computing Systems Laboratory
Department of Electrical Engineering, FT-10
University of Washington
Seattle, WA 98195
* from the Samsung Advanced Institute of Technology, Korea

**Abstract**

The applications of imaging and computer graphics are becoming more important and widespread. In view of the proliferation of the application areas of imaging and graphics, several computer architectures have been developed to perform image processing or computer graphics. Most systems have been optimized to either imaging or graphics, even though some application areas require both imaging and graphics. In order to satisfy the increasing number of these applications and to achieve high performance for both imaging and graphics, we have designed and are implementing a system called UWGSP4 (University of Washington Graphics System Processor #4).

UWGSP4 consists of four parts: a parallel vector processor, a shared memory system, an interconnection network, and a graphics subsystem. The parallel vector processor was provided mainly for imaging and general purpose computing, whereas the main functions of the graphics subsystem are to generate realistic three-dimensional images and to draw the display screen. Among the four parts, the shared memory and the interconnection network were designed to fully support the parallel vector processor and graphics subsystem so as to yield high sustained performance.

This paper discusses the architecture of UWGSP4, presents simulated performance figures for several imaging and graphics algorithms, and explores some application areas.

## 1 Introduction

During the last decade, the application areas of imaging and computer graphics have proliferated and have received a lot more attention. Some of the application areas require both high-speed imaging and computer graphics operation capabilities. For example, the fast Fourier transform (FFT), which is an important image processing tool, has been used in computer graphics in generating landscape and ocean scenes of startling realism [1]. On the other hand, computer graphics has been used in the volume rendering of medical computerized tomographic (CT) images to produce realistic three-dimensional images of the human body [2].

In conjunction with the proliferation of application areas, technology advances and developments in the case of VLSI technology, parallel and pipelined architectures, image display resolution, and interactive graphical user interfaces have provided near real-time image processing and 3-D graphics rendering. Also, technology developments make it possible for high-speed image processing and computer graphics operations to be performed in a single platform.

In consideration of the trends in technology development and application areas, the UWGSP (University of Washington Graphics System Processor) series of imaging and graphics workstations have been developed at the University of Washington since 1986. UWGSP1, developed in 1986, was a first generation fixed point imaging and graphics subsystem interfaced to an IBM PC/AT host. UWGSP1 consisted of a TMS34010 graphics system processor (GSP) closely coupled with a TMS 32020 digital signal processor (DSP), and provided a flexible, low-cost (less than $7000 total cost) and medium performance (it takes about 3 seconds to perform 3 x 3 convolution on a 512 x 512 image) imaging platform. Since fixed point 16-bit arithmetic is not sufficient for many image processing applications, we developed a second generation (UWGSP2) in 1988, which used a floating point processor (Texas Instruments 74ACT8837). The imaging performance of UWGSP2 was about twice that of UWGSP1. UWGSP3 [3], developed in the beginning of 1990, utilized a multiprocessor configuration for imaging and graphics, and consisted of a TMS34020 GSP and four TMS34082 floating point coprocessors that can be configured into either a pipelined or SIMD (single instruction, multiple data streams) mode depending on the algorithm. The peak performance of UWGSP3 is 160 MFLOPS, with a sustained performance of about 30 to 40 MFLOPS.

UWGSP4, which is introduced in this paper, is a superworkstation which enables image processing and computer graphics to be performed with very high speed. UWGSP4 was designed as a powerful image processing system capable of operating on large images in real time, as well as a powerful graphics workstation capable of generating realistic images in real time or near real time.

The following sections of this paper describe the architecture of UWGSP4, and some simulated performance figures for several imaging and graphics algorithms.

## 2 System Architecture of UWGSP4

Figure 1 shows the overall block diagram of the UWGSP4 which consists of four parts: a parallel vector processor, a shared memory, an interconnection network, and a graphics subsystem. The parallel vector processor and the graphics subsystem are the main computing parts for imaging and graphics processing, respectively, whereas the shared memory and the interconnection network provide the computing parts with high-bandwidth memory access and data transfer. The reason why two distinct computation parts are used to perform the image processing and graphics operations separately is that there is a difference between their operations and data flow; i.e., the vector

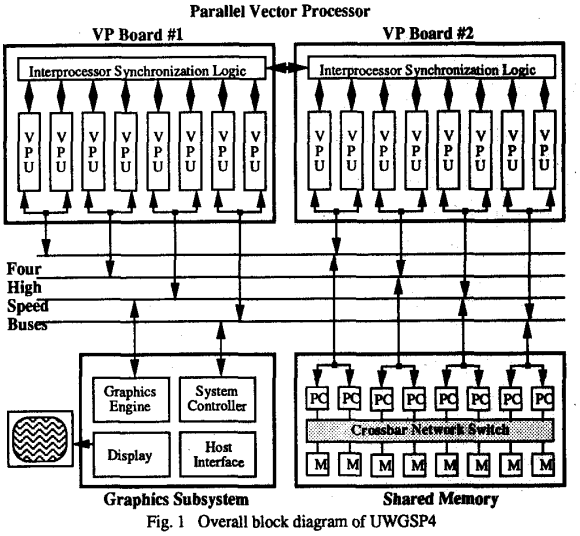


Fig. 1 Overall block diagram of UWGSP4

CH2954-6/91-0000-223 $1.00 © 1991 IEEE

processor can be fully utilized for image processing since most image processing requires same operation on a large amount of data set such as matrix operation, but computer graphics cannot fully utilize vector operations and requires mostly pipelined operations.

For most image processing operations such as image transforms and convolution, the image can be divided into several sub-image blocks and multiple processors can perform the operations on the sub-image blocks in parallel to speed up the processing. Therefore, multiple vector processor architecture, where the multiple vector processors are running in parallel, is a good solution for image processing. Our parallel vector processor consists of 16 vector processing units each of which can operate independently or in synchronism with the others. Since most image processing can be performed by multiple vector processors in parallel fashion, high sustained computation rates are expected in our UWGSP4. All communications between vector processing units are performed through the shared memory.

Since most of the limitations in parallel computing systems arise from the memory access and data transfer bandwidth, the shared memory and the interconnection network should be designed to maximize data transfer and memory access bandwidth. In UWGSP4, the interconnection network between processing units and memory modules of shared memory consists of a 8 x 8 crossbar network connected to four high-speed buses. A crossbar network is one of the best interconnection networks from the viewpoint of bandwidth and availability [4]. Since the depth of the crossbar network is four bytes for data plus some control signals (a total of 40 bits) and the cycle rate of the crossbar network is 40 MHz, the total data transfer rate of our crossbar network is 1,280 Mbytes/sec.

Since 16 vector processing units are connected to the shared memory via four high-speed buses, each high-speed bus provides four vector processing units with a communication path to shared memory by time-sharing. In order to meet the sustained data transfer rate of 1,280 Mbytes/sec, the high-speed bus runs at a speed of 80 MHz for which an ECL (emitter-coupled logic) bus interface is used.

The shared memory uses a 32-way interleaving scheme in order to achieve a 1,280 Mbytes/sec memory access bandwidth with standard DRAMs. Vector data or 2-D array data can be accessed with a single load or store instruction, so that the maximum memory access bandwidth (1,280 Mbytes/sec) can be sustained.

The graphics subsystem supports the host interface, the graphics processing, and an image display. The graphics processing section consists of a parallel-pipelined architecture capable of providing a high graphics performance (about 200,000 Gouraud shaded polygons/sec). The image display section supports 24-bit full color images, and double frame buffers are incorporated to support smooth animation. The host interface provides a 20 Mbytes/sec data transfer rate between UWGSP4 and a host computer.

For the implementation of this system, four kinds of ASIC (application specific IC) chips were designed and three different large-size multilayer PCB's (printed circuit board) were also designed for parallel vector processor, shared memory and graphics subsystem. One of the ASICs serves as the master controller for the vector processing unit. Two other ASICs were designed for controlling the shared memory, and the last ASIC is used in the graphics subsystem. Also, in order to increase chip density on the board, many chips in surface mountable packages will be used. Due to the large board size, fast clock, and high chip density, power distribution, clock skew, and cooling problems were considered and estimated.

The following sections describe each part in detail.

## 3 Parallel Vector Processor

Figure 2 shows the block diagram of a single vector processing unit which is the main computation engine in UWGSP4. The complete parallel vector processor consists of 16 such vector processing units over two boards as shown in Fig.1. Each vector processing unit

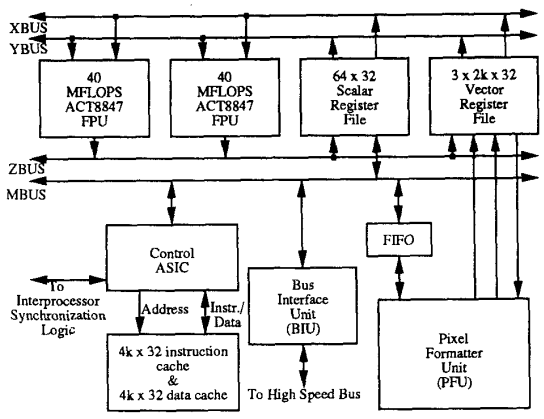


Fig. 2 Block diagram of a single vector processing unit

consists of two floating point units (FPUs), a set of scalar and vector register files, an application specific integrated circuit (ASIC) for control and instruction issue, a pixel formatter unit (PFU) for pixel handling, a unified instruction and data cache, and a bus interface unit (BIU) for interface to the high-speed bus.

The BIU provides the signal conversion between the standard TTL-level interface for vector processing units and the ECL-level interface to the high-speed bus.

All integer and floating point operations for imaging are performed by 74ACT8847 CMOS floating point processor chips from Texas Instruments. Each vector processing unit has two FPUs which operate in an alternating fashion with a 20 MHz two-phase clock. Each FPU possesses a full set of arithmetic and logic instructions. The arithmetic logic unit (ALU) and the multiplier within the floating point processor can operate independently or be used simultaneously when performing pipelined multiply-accumulates. The pipelined multiply-accumulation provides peak computation performance of 40 MFLOPS for each FPU, so that one vector processing unit provides a peak computing rate of 80 MFLOPS.

The master control ASIC of the vector processing unit fetches and interprets instructions, and controls the FPUs so as to execute the desired arithmetic and logical operations. It is implemented using a 20,000 gate CMOS standard-cell custom IC. In particular, this ASIC contains vector data addressing hardware which allows the vector register file to be accessed with different addressing patterns. The vector data addressing hardware permits most imaging algorithms (e.g., FFT, convolution, matrix multiplication and addition, and so on) to be performed in a single vector operation, so that it is not necessary for processor to spend its time for address calculation in vector operation.

Sixty four four-ported 32-bit scalar registers and three 2048-word vector register files have been provided for each vector processing unit. During scalar execution, the control ASIC manipulates the four ports to move data to and from the two FPUs and the data cache. Each vector register has two ports for separate read and write. UWGSP4 provides high-speed register-to-register vector operations as follows: the control ASIC loads two of the vector register files with arrays of input operands, and then causes the FPUs to perform the desired computations on the arrays to generate a third array, which is stored into the third register file. Thus, only two vector register files can be loaded from shared memory; the third register file can be used to store computed vector results to shared memory.

For high-speed scalar operation and instruction issue, 8 kbytes of instruction and data cache is provided, where each of instruction and data caches has 4 kbytes. The placement policies used for the instruction and data caches are direct mapping and a modified set-associative mapping systems, respectively.

When image data is stored in the shared memory and displayed

on the screen, image pixel data usually consists of 8-bit or 16-bit unsigned integer values, whereas most image computation should be done in floating point for computational accuracy. Therefore, conventional processors need to convert and format image pixel data into a form that is suitable for computations, as well as to carry out the reverse operations on the computed data before the data is transferred to the shared memory. In UWGSP4, this function is carried out by a special pixel formatter unit (PFU) which is implemented with a field programmable gate array (FPGA). The floating point processors are not interrupted for pixel formatting and data conversion between integer and floating point values, so that they can be fully utilized for the main computation. This PFU can also transfer data stored in the third vector register file to the other register file so that the results can be used as source operands.

For the interprocessor synchronization and cache coherence, a combination of a token-ring scheme in hardware and software test-and-set or fetch-and-add operations are provided in the vector processing unit. For example, if a vector processing unit wants to access a shared variable from the shared memory, it has to obtain a token from the interprocessor synchronization logic before accessing the shared memory.

## 4 Shared Memory and Interconnection Network

The shared memory and interconnection network supplies data for processors with high data bandwidth so that processors can provide their maximum performance. Figure 3 shows the block diagram of the shared memory and crossbar network of UWGSP4. The shared memory and interconnection network consist of four high-speed buses, four bus interface units (BIUs), eight port controllers (PCs), an 8 x 8 x 40-bit crossbar, eight memory controllers (MCs), and eight four-way (total 32-way) interleaved memory modules.

The BIUs interface the high-speed buses to the port controllers; each BIU interfaces one high-speed bus to two port controllers. All shared memory components, except the high-speed buses and BIUs, operate with a 40 MHz clock cycle, whereas the high-speed buses use an 80 MHz clock cycle. Therefore, two port controllers can communicate with one high-speed bus in an alternating fashion without any conflict. The BIUs interface the TTL-level logic in the shared memory to the ECL bus.

The port controllers were designed using 12,000 gate CMOS gate-array ASICs. The port controllers translate memory access commands from the vector processing units into simple commands which can be executed by the memory controller, and control the crosspoint

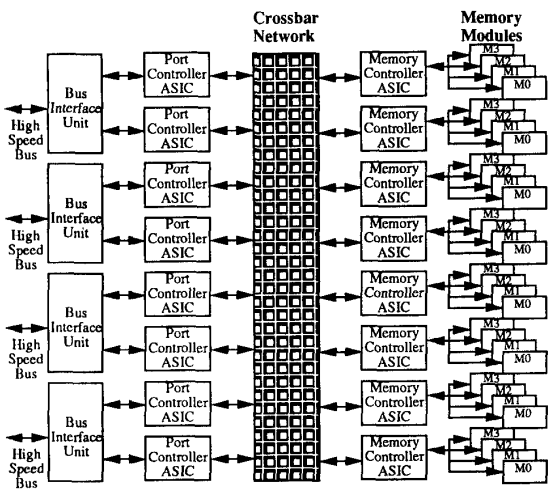


Fig. 3 Block diagram of the shared memory and crossbar network

switches in conjunction with the memory controllers. The vector processing units can access a scalar data item, a column or row vector, or 2-D array data with a single command. This command is converted by the port controller to one or more commands, each of which is sent to a memory controller for row vector accessing. The converted commands are sent to the appropriate memory controller sequentially through the crossbar network. After a command has been sent to a memory controller, data is sent from the vector processing unit to the memory controller via the port controller (during a write access), or is received from the memory controller and are sent to the vector processing unit (during a read access). Eight port controllers permit a maximum of eight vector processing units to access the shared memory simultaneously; every two vector processing units share one port controller. The port controller was designed for up to four processors to be connected to it. Most image computing operations such as FFT and image convolution are not I/O-bound, but CPU-bound. Therefore, 16 vector processing units can efficiently access the shared memory without severe conflict in the memory access.

After a row vector accessing command is transferred from the port controller to the memory controller, the memory controller generates the physical address of each data word. Each memory controller controls four interleaved DRAM memory modules, and accesses vector data from the memory modules at an access rate of 160 Mbytes/sec. Also, DRAM refresh is carried out by the memory controllers. The memory controller was designed as an 8,000 gate CMOS gate-array ASIC.

An interconnection network between eight port controllers and eight memory controllers is provided by an 8 x 8 crossbar network. The crossbar network is implemented with discrete Advanced Schottky TTL transceiver chips controlled by the port controllers and memory controllers. This crossbar network is arbitrated by the port controllers and memory controllers.

In order to increase memory bandwidth and to match the relatively slow memory to the processor speed, a memory interleaving scheme has been used [5]. In UWGSP4, 32-way memory interleaving is used to achieve a 1,280 Mbytes/sec memory bandwidth. All memory accesses are 32-bit word-based, but any vector processing unit can selectively write to any specific byte within a word by using byte mask information sent from the vector processing unit. The shared memory provides a maximum memory space of 1 Gbyte (256 Mwords).

## 5 Graphics Subsystem

Main functions of the graphics subsystem are to maintain and draw the display screen, to generate realistically-shaded three-dimensional images from scene descriptions, and to interface with a host computer. Figure 4 shows the graphics subsystem which consists of two independent polygon processing pipelines, four bit-blit interpolator (BBI) ASICs, a Z-buffer and a double-buffered frame buffer, a TMS34020 graphics processor and system controller, an overlay buffer, a cursor generator, RAMDACs and a host interface.

General three-dimensional graphics operations cannot fully utilize the parallel vector processor since most graphics operations are small-size matrix operations (3 x 3 or 4 x 4) or scalar operations. Therefore, pipelined architectures were designed for optimum graphics operations. The front-end graphics processing is performed by the two polygon processing pipelines. The operations provided by the pipelines include geometric transformation, back-face culling, illumination, clipping, projection and slope calculation. The polygon processing pipelines consist of nine Intel i80860 CPUs operating in a parallel-pipelined fashion. One of the nine CPUs, called the head processor, communicates with the shared memory to extract the actual polygon data by traversing an object hierarchy, and distributes polygon processing jobs to the two pipelines. Each pipeline is composed of four CPUs (corresponding to four pipeline stages) and performs transformation, lighting, clipping, and span generation. The

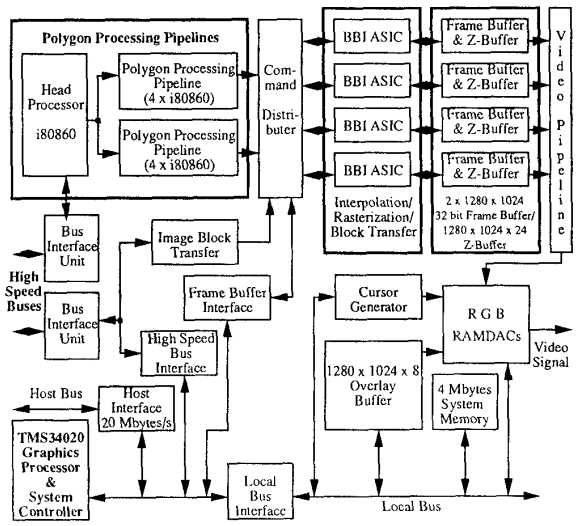


Fig. 4 Block diagram of the graphics subsystem

head processor dynamically assigns polygon rendering jobs to each pipeline stage to balance the load among eight CPUs. The results of these operations are fed to four BBI ASICs, where scan conversion is performed in conjunction with hidden surface removal.

The four ASICs, each of which is a 20,000 gate CMOS standard-cell custom IC, are capable of drawing and filling image pixels at the rate of 40 million pixels per second. They also control the memory in the frame buffer and Z-buffer, and carry out screen refresh functions under the supervision of the system controller. Another feature incorporated into the BBI chips is the ability to support a multi-windowed display with minimal overhead, based on an extended Z-buffer concept. The pipelined architecture of the polygon processing pipelines and four BBIs is capable of handling over 200,000 100-pixel polygons per second with Gouraud shading. Other special functions provided by the BBIs are transparency, antialiasing, and texture mapping.

For realistic animation, UWGSP4 was designed with a double frame buffer which consists of two 1280 x 1024 pixel buffers built using video RAMs (VRAMs), with each pixel being 32 bits deep to support 24-bit true color. The other eight bits in the 32 bits of frame buffer are used for transparency, antialiasing and alpha blending [6]. The Z-buffer, whose size is 1280 x 1024 x 24 bits, is implemented in VRAMs as well to significantly speed up three-dimensional rendering algorithms.

The system controller of the TMS34020 maintains the host interface, acts as a central controller for the UWGSP4 system, and controls the 1280 x 1024 x 8 bit overlay buffer. A 4 Mbyte local memory is provided for storing programs and data associated with the system controller. The data transfer rate between the host computer and UWGSP4 through the host interface is 20 Mbytes/sec.

Among four high-speed buses, the graphics subsystem uses only two high-speed buses. One interface with the high-speed bus is used to load graphics data from the shared memory to the polygon processing pipelines. The other interface is used to transfer the image data, program, and any control information between the shared memory and the host or the graphics subsystem.

## 6 Performance Estimation and Conclusions

Our goal for the UWGSP4 is to achieve a peak imaging performance of 1,280 MFLOPS and a graphics rate of 200,000 polygons/sec with Gouraud shading, and also to support a high sustained performance of more than 60% of the peak performance.

Initial simulations show that a 5 x 5 convolution can be performed on a 512 x 512 image in under 30 msec. The simulations also predict that UWGSP4 will be capable of carrying out a 2-D FFT at rates of over 600 MFLOPS, or equivalently, under 33 msec for a 512 x 512 image. For comparison, these computing rates for 5 x 5 convolutions and 2-D FFT are about 10 times and 100 times faster than the CMU Warp computer, respectively: a 10-cell Warp can perform a 5 x 5 convolution in 284 msec and a 2-D FFT in 2.5 sec on 512 x 512 images [7].

Our simulation results for the graphics performance are over 200,000 Z-buffered, Gouraud shaded 100-pixel polygons per second, and 250,000 3-D shaded 100-pixel lines per second. Also, some special graphics functions such as texture mapping and transparency are provided by the hardware for high speed processing. In comparison, the Silicon Graphics 4D/240GTX system provides about 100,000 Z-buffered, lighted quadrilaterials per second [8] and the Titan Supercomputer can render up to 200,000 Gouraud shaded triangles/sec [9].

In conclusion, the UWGSP4 is an integrated system with very high computational performance for both imaging and computer graphics application areas, such as scientific visualization, high performance military and medical diagnostic workstations, CAD/CAM, realistic animation, and so on. In particular, advanced medical applications require high speed image processing such as contrast enhancement, arithmetic/logical operations, rotation, zoom, window/level, image compression/decompression, cine display, and image analysis, together with compute-intensive graphics operations such as 3-D reconstruction and volume rendering. Also, military applications need advanced image processing and analysis as well as graphics operations such as vector map handling and terrain modeling and 3-D display.

## References


[1] H.O. Peitgen and D. Saupe (Eds.), The Science of Fractal Images, Springer-Verlag, New York, 1988.

[2] D.R. Ney, E.K. Fishman, D. Magid, and R.A. Drebin, "Volumetric Rendering of Computed Tomography Data: Principles and Techniques," IEEE Computer Graphics and Applications, vol. 10, pp. 24-32, 1990.

[3] K.S. Mills, G.K. Wong, and Y. Kim, "A High Performance Floating-Point Image Computing Workstations for Medical Applications," SPIE Medical Imaging IV, vol. 1232, pp. 246-256, 1990.

[4] K. Hwang and F.A. Briggs, Computer Architecture and Parallel Processing, McGraw-Hill, Inc., 1984.

[5] T. Cheung and J.E. Smith, "A Simulation Study of the CRAY X-MP Memory System," IEEE Trans. on Computer, vol. C-35(7), pp. 613-622, 1986.

[6] J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, Computer Graphics: Principles and Practice (second edition), Addison-Wesley Publishing Co., 1990.

[7] M. Annaratone, E. Arnould, T. Gross, H.T. Kung, M. Lam, O. Menzilcioglu, and J.A. Webb, "The Warp Computer: Architecture, Implementation, and Performance," IEEE Trans. on Computer, vol. C-36(12), pp. 1523-1538, 1987.

[8] K. Akeley, "The Silicon Graphics 4D/240GTX Superworkstation," IEEE Computer Graphics & Applications, vol. 9(4), pp. 71-83, 1989.

[9] B.S. Borden, "Graphics Processing on a Graphics Supercomputer," IEEE Computer Graphics & Applications, vol. 9(4), pp. 56-62, 1989.