



Quaestiones Informaticae

Vol. 3 No. 1

February, 1984

Quaestiones Informaticae

An official publication of the Computer Society of South Africa and
of the South African Institute of Computer Scientists

'n Amptelike tydskrif van die Rekenaarvereniging van Suid-Afrika en
van die Suid-Afrikaanse Instituut van Rekenaarwetenskaplikes

Editor: Prof G. Wiechers

Department of Computer Science and Information Systems
University of South Africa
P.O. Box 392, Pretoria 0001

Editorial Advisory Board

PROFESSOR D. W. BARRON
Department of Mathematics
The University
Southampton SO9 5NH
England

PROFESSOR K. GREGGOR
Computer Centre
University of Port Elizabeth
Port Elizabeth 6001
South Africa

PROFESSOR K. MACGREGOR
Department of Computer Science
University of Cape Town
Private Bag
Rondebosch 7700
South Africa

PROFESSOR M. H. WILLIAMS
Department of Computer Science
Herriot-Watt University
Edinburgh
Scotland

MR P. P. ROETS
NRIMS
CSIR
P.O. Box 395
Pretoria 0001
South Africa

PROFESSOR S. H. VON SOLMS
Department of Computer Science
Rand Afrikaans University
Auckland Park
Johannesburg 2001
South Africa

DR H. MESSERSCHMIDT
IBM South Africa
P.O. Box 1419
Johannesburg 2000

PROFESSOR P. C. PIROW
Graduate School of Business
Administration
University of the Witwatersrand
P.O. Box 31170
Braamfontein 2017
South Africa

Subscriptions

Annual subscriptions are as follows:

	SA	US	UK
Individuals	R6	\$7	£3,0
Institutions	R12	\$14	£6,0

Circulation manager

Mr E Anderssen
Department of Computer Science
Rand Afrikaanse Universiteit
P O Box 524
Johannesburg 2000
Tel.: (011) 726-5000

Quaestiones Informaticae is prepared for publication by Thomson Publications South Africa (Pty) Ltd for the Computer Society of South Africa and the South African Institute of Computer Scientists.

Very High Speed Graphics Work-Station

J Jablonski and J H Dijkman
Council for Scientific and Industrial Research Pretoria

Abstract

The design of a high speed graphics workstation is presented. A specifically designed Graphics System Interface combines standard computer peripherals to produce a sophisticated man machine workstation. A review is given on a software support package and certain special features are emphasized. Finally some performance figures are presented.

Introduction

As the computer aided presentation of drawings and designs has become a fact of life, interactive graphics systems have become an indispensable design tool. A graphics workstation consists of one or more display devices coupled to graphics input and/or output devices which together can be used to support an interactive graphics session.

Description of the workstation

The graphics workstation for which the GSI interface and GMASP software support package were developed consists of the following devices:

- Tektronix GMA102A display monitor
- Summagraphics Bit Pad One digitizing tablet
- Keyboard with loudspeaker

Both types of the display techniques, that is storage and 'write thru' are implemented in the GMA102A. The display monitor is equipped with vector and character generators. The visible address area of the display is 4096 units in X and 3072 units in the Y direction. The character generator is capable of drawing any of 128 ASCII characters in one of 4 sizes. The vector generator facilitates drawing of lines in any one of the 16 available line patterns. A hard copy unit may also be attached to the display unit.

The user's program controls the display unit by sending it a block of 16 bit words called a display list (Fig. 1). This block contains control information and the data necessary to create the desired picture. The end of the display list is indicated by a control word in which an EOF (end-of file) marker is set. As each control word sets the display mode (storage or 'write thru'), it is possible to create a picture in such a way that some parts of it are drawn in 'write thru' while the rest is drawn in storage mode.

The Bit Pad One digitizing tablet is used primarily as an input device for coordinate data or as a device to select specific parts of the displayed picture. The active area of the tablet comprises 2048 points in both the X and the Y directions. The Bit Pad may also be equipped with a pen stylus or a four-button cursor. The digitizing tablet transmits coordinate pairs in three modes, i.e. point mode, stream mode and switched stream mode. The sampling rate may be selected programmatically from 1 to 200 coordinate pairs per second and for each coordinate pair five bytes of data are transmitted (Fig. 2).

The Bit Pad tablet and keyboard with built in loudspeakers provide a convenient way for the user to interact with the system.

The graphics system interface

The Graphics System Interface (GSI) was designed as an interface between the 3200 series Perkin Elmer processors and the complete workstation, but in point of fact it contains six functional parts, each of them being a separate interface. These parts are as follows: Multiplexor Bus Interface (MBI), EDMA Bus Interface, GMA display monitor Interface (DI), Bit Pad One Interface (BI), Keyboard Interface (KI) and Sound Generator (SG). The GSI is connected to two busses, i.e. Extended Direct Memory Access Bus and Multiplexor Bus (Fig. 3). Direct Memory Access Bus is used by both the GMA and the Bit Pad interfaces. The reason for this was the way in which the GMA receives graphic data from the processor, i.e. by means of a block data transfer. Another reason for implementing the DMA transfer was to be able to use the GMA in refreshed 'write thru' mode while at the same time reading coordinate data from the Bit Pad. This feature gives the programmer the facility to place a cursor on the screen and to move this cursor under Bit Pad control without program intervention.

The Multiplexor Bus Interface creates the communication path between the user's program and the remaining interfaces. From the point of view of Perkin Elmer processors, the whole workstation is seen as a single device with one device address assigned to the whole GSI. The MBI fulfills all the requirements of the Perkin Elmer I/O system by allowing the system to execute the I/O commands which are used to enable/disable interrupts, check status of the GSI and to write or read the registers of the GSI. Two I/O commands are required to write or to read a GSI register (except the Status Register). The first command selects the desired register and the second command writes to or reads from that register.

In addition to the other functional circuits, the MBI contains an Interface Mode Register and a Status Register. The contents of the Interface Mode Register determines which devices and which modes of operation will be activated by the start command. At the termination of a device operation an I/O interrupt is created. The source of the interrupt is identified by the contents of the Status Register.

The Keyboard Interface accepts an odd parity 8 bit ASCII code. Any of the three keys originally labelled 'Here is', 'Break' and 'Repeat' when depressed simultaneously with any other key sets one of three flag bits in the keyboard data register. This facility extends the range of the keyboard.

The Sound Generator can produce a single tone of fixed duration selected from 8 frequencies in each of 4 octaves.

This frequency range extends from 130 Hz (approximately C below middle C) to 1 976 Mz (approximately B in the third octave above middle C).

The DMA interface contains circuits which perform the DMA bus acquisition sequence and control the data transfers according to the Perkin Elmer Extended Direct Memory Access Bus specification. Strap options are provided in order to adapt the

GRAPHIC CONTROL WORD
VECTOR DEFINITION
VECTOR DEFINITION
VECTOR DEFINITION
ALPHA CONTROL WORD
CHARACTER CODE
CHARACTER CODE
CHARACTER CODE
CHARACTER CODE
EOF MARKER

FLAG BYTE
X BYTE
X BYTE
Y BYTE
Y BYTE

FIGURE 1 Display list

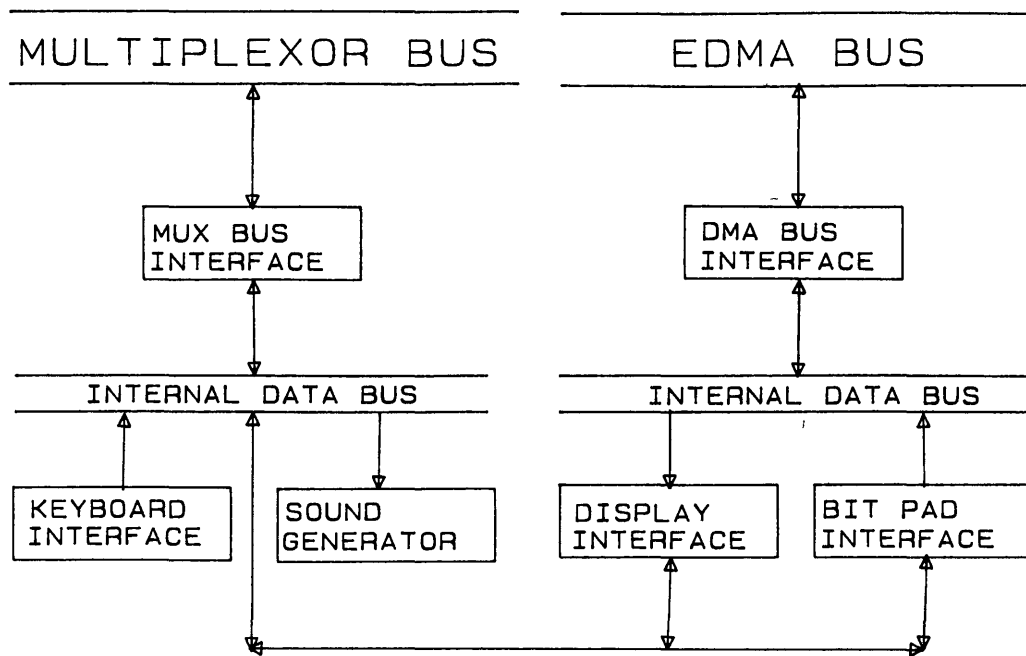


FIGURE 3 Block diagram of the graphics system interface

DMA interface to the specific requirements of the different Perkin Elmer processors

The EDMA specification allows 3 modes of data transfer, they are halfword, fullword and burst mode. The EDMA bus acquisition sequence may be followed by a one 16 bit data transfer in halfword mode or by two data transfers in fullword mode or by eight data transfers in burst mode. For the GMA display interface only read burst mode was incorporated. For the Bit Pad interface the fullword mode would have been the best, but as this mode is not implemented on the 3240 processor the write halfword mode was chosen instead.

The Bit Pad Interface (BI) contains three registers which are loaded by the user's program before the start command initializes the GSI operation. These three registers form the Mode Register, the Start Address Register and the Buffer Length Counter. The contents of the Mode Register determines the sampling rate and selects the operating mode of the Bit Pad. The Start Address Register contains the memory address where data from the tablet will be transferred. The Buffer Length Counter contains the number of coordinate pairs that are required to be transferred to the main memory. There are two modes in which BI may operate — digitize mode and repeat mode. Bit Pad interface is only ready to transmit both X and Y data to EDMA bus after receiving the required 5 bytes of data from the tablet (1 byte — flag data, 2 bytes — X data, 2 bytes — Y data). Thereafter, two halfword transfers are executed in order to send one coordinate pair.

In repeat mode the X,Y data is always sent to the same pair of halfwords in main memory whose address is determined by the Start Address Register. In digitize mode consecutive coordinates are transferred to the consecutive locations in the buffer whose start address and length are determined by the Start Address Register and the Buffer Length Counter. In both modes the flags created by the cursor may be used to terminate the tablet operation. By reading the Buffer Length Counter at the end of a data transfer the programmer is able to obtain the number of coordinate pairs transferred to the buffer during the operation.

The Interface Mode Register of GSI controls the way in which the display list is transferred from the computer to the GMA. Two operating modes of the display interface are possible, that is stop mode and repeat mode. Before activating the interface operation both registers of the DI should be set. The Display Control Register creates signals to control hardware features

such as non store, view, erase, make copy, centre and inhibit.

The Start Address Register is loaded with the memory address at which the display list starts. The start command sets the EDMA request signal and then the display list transfer begins. The display list is transferred in blocks of eight 16 bit words into the FIFO (first-in first out) memory which is used in order to cater for the different timing requirements. The FIFO is 16 words long and therefore only capable of storing two bursts. The FIFO is read by the GMA at a rate limited by the GMA instruction execution time. After 8 words have been read from FIFO, the DMA request signal is set and the next burst is written into the FIFO memory. One burst is loaded into the FIFO in 2 us, while the execution time of the quickest GMA instruction, which is a 'move absolute', takes longer than 3 us. Bursts are read from main memory until the EOF marker is detected at the input to the FIFO. The GMA operation is terminated as soon as the EOF marker is detected at the output of the FIFO. If DI is operating in stop mode when the EOF marker is encountered, then DI activity ceases and a Multiplexor Bus interrupt is generated. In repeat mode the DMA request signal is set instead of the Multiplexor Bus interrupt. The EDMA request restarts the display list transfer from the original address contained in the Start Address Register. Under these circumstances the transfer may only be terminated by resetting the repeat bit in the Interface Mode Register programmatically so that the last transfer of the display list is done in stop mode.

As all device interfaces are independent of one another they may operate in any combination. A particularly useful combination is that of operating both the tablet and display interfaces in repeat mode. By placing the two memory locations containing coordinate data from the bit pad tablet inside the display list and using them as a current absolute position of the picture defined in this segment of the display list, the programmer has a means by which he can create a moveable cursor of any shape (Fig. 4). Once the display list transfer has been activated by the program, the cursor will follow the position of the Bit Pad stylus without program intervention.

Workstation Software Support Package

A subroutine package (GMASP) creates a structured communication path between a program and the graphics workstation. GMASP provides a means of creating display lists on a high level and these are then transmitted to the workstation as a unit.

The package can be used at two levels of programming. Level 1 is a basic 'systems' level where display modes are set and communication with the interface is accomplished by means of supervisor calls via the operating system driver. The higher level 2 of the package is responsible for the administration of a buffered set of display lists.

Level 1 provides no means of setting up display lists but supplies the user with sufficient facilities to enable him to communicate with the graphics workstation at a system programmer's level. Error logging to communicate errors to the user is not done. There are, however, routines to return the status of the interface after every transfer. Typical functions at this level are:

- (a) Establishing the input/output channel assigned to the GMA,
- (b) Controlling the modes of the display: Bit Pad, keyboard and tone generator,
- (d) Reading the current status of the interface,
- (c) Setting modes for automatic cursor control.

Level 2 provides the main advantages of the package by utilizing the special features provided by the interface. It handles the creation and transfer of display lists to the interface. These display lists are buffered into a pool of uniquely numbered display list 'buffers'. In using this level the programmer specifies a block of memory in his program for use by the package and storage management is done automatically. In this way the programmer can limit the amount of memory used for display lists. Procedures are provided to create, build and update display lists in a structured fashion. To achieve this, the level 2 procedures are logically divided into 3 groups:

- general procedures: These are procedures that control the definition of display lists, global mode settings specific to display lists as well as the use and ultimate release of these display lists.
- building procedures: These constitute the graphic primitives such as move, draw, character output, cursor address specification, etc.
- entry procedures: Entry points may be defined in display lists which may later be modified so as to create different instances of a display entity.

One of the main features provided by the interface is that cursor tracking is done automatically and that cursor definition is left to the programmer. The programmer specifies a special 'entry point' in a display list (commonly called the cursor address) which is then automatically updated by the Bit Pad via the interface so that the cursor position changes as the Bit Pad stylus is moved. This is all done without any program intervention. Apart from the automatic tracking facility, the programmer may also define numerous forms of cursors. A typical example of this is to define a whole graphic entity as a cursor and then to 'drag' this entity across the display as if it were a cursor. Upon termination of the cursor tracking (that is, when the stylus is depressed), the program can easily change the mode of the 'cursor' into storage mode, thereby fixing an image of that entity on the display.

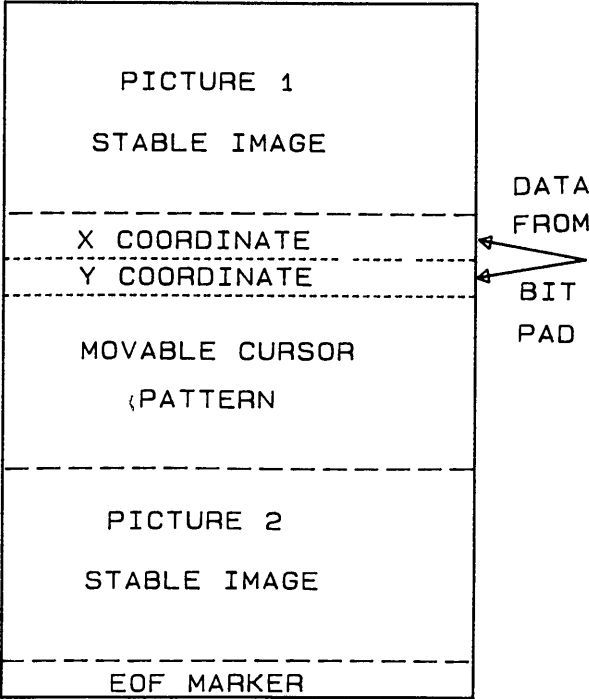


FIGURE 4 Display list with cursor

Two aspects which are particularly useful in complex graphic applications, such as in cartography, are the high speed of the interface and its direct access to a large portion of main memory. By using these features, very complex cartographic images can be generated at a very high speed with little effort, by having the complete specification defined in one display list.

The only disadvantage of the package is that garbage collection must be done in order to make optimal use of the memory available for display lists. For this purpose, an optimizing garbage collector was written in order to administer the use of this memory. The memory pool provided by the program gets more and more fragmented as display lists are released for re-use and others of different lengths defined. When this fragmentation becomes critical, all the display lists are moved towards low memory in order to allow that the free space to be compacted into one contiguous block at the end of the pool. The following two conditions will invoke garbage collection:

- (a) The total amount of memory available is sufficient for the definition of a new display list, but is fragmented in such a way that no individual fragment fully contains the display list.
- (b) The definition of a variable length display list where all the available memory must be available for the building of such a display list.

Extensive error checking and reporting is done at level 2 in order to ensure that the programmer gets sensible feedback.

PICTURE	DMA TRANSFER TIME	DISPLAY TIME
ONE LINE OF (136) SMALLEST CHARACTERS	$36 \times 10^{-6} \text{ s}$	$12.2 \times 10^{-3} \text{ s}$
THE SET OF 204 SHORT VECTORS (EACH 20 UNITS LONG)	$52 \times 10^{-6} \text{ s}$	$1.1 \times 10^{-3} \text{ s}$

TABLE 1 Display and data transfer times

regarding the use of memory, display modes etc. This aspect together with the other procedures supplied at this level, enables the program to communicate with the GMA Graphics Workstation efficiently at a high level.

Conclusion

In this configuration of the graphics workstation and Graphics System Interface the capabilities of devices being used limit the transfer to and from the workstation. In case of the GMA102A, the execution time of the quickest GMA instruction, that is, 'move absolute', is larger than 3 μ s while the rate of the DMA transfer is 8 Mb/s. One burst of eight GMA instructions of any type is read from memory in a time that is

shorter than the execution of any instruction. Loading of the computer output bandwidth is thus limited to $200/(3 \times 8) = 8\%$ or less of the available capacity. This prevents operation of the display from impacting other users of the system. The capability of using the bandwidth of the display device to a maximum allows images consisting of many thousands of vectors to be drawn within a very acceptable response time constraint. Further improvement of the GSI is possible by building a fast processor into it. This processor could build display lists for polygons, circles etc. and be responsible for operations like zooming, rotation, translation etc., according to data received from the host computer.

Table 1 gives some indications of the display times for pictures displayed in 'write-thru' mode.

References

- [1] PERKIN ELMER Corporation. 3220 Processor installation and maintenance manual. Perkin Elmer publication number H29 695R08. Oceanport, New Jersey 07757. June 1979.
- [2] PERKIN ELMER Corporation. Input/Output interface design specification. Perkin Elmer publication number 40 009. Oceanport, New Jersey 07757.
- [3] PERKIN ELMER Corporation. Universal Direct Memory Access interface for 32 bit processors. Perkin Elmer publication number H29 728. Oceanport, New Jersey 07757. October 1979.
- [4] PERKIN ELMER Corporation. Universal Logic Interface instruction manual. Perkin Elmer publication number 29 311R01. Oceanport, New Jersey 07757. April 1975.
- [5] SUMMAGRAPHICS Corporation. Bit Pad One User's Manual. Fairfield, Connecticut 06430, USA. August 1981.
- [6] TEKTRONIX Inc. GMA102A storage display monitor instruction manual. Manual part number 070 3635 00. Beaverton, Oregon 97077, USA. August 1980.
- [7] TEKTRONIX Inc. GMA series option 42/43 vector/character instruction manual. Manual part number 070 2549 01. Beaverton, Oregon 97077, USA. May 1980.
- [8] Dreyer, J. J., le Roux, Dijkman, J. H. & Koen, A. J. GMASP Users Manual. SWISK 33. CSIR, NRIMS, Pretoria. January 1983.
- [9] J. Jablonski, P. P. Roets. Graphics System Interface (GSI) for Perkin Elmer 3200 Computer Maintenance and Installation Manual. I464. CSIR, NRIMS, Pretoria. February 1983.

Notes for Contributors

The purpose of this Journal will be to publish original papers in any field of computing. Papers submitted may be research articles, review articles, exploratory articles of general interest to readers of the Journal. The preferred languages of the Journal will be the congress languages of IFIP although papers in other languages will not be precluded.

Manuscripts should be submitted in triplicate to: Prof. G. Wiechers at:

Department of Computer Science
University of South Africa
P.O. Box 392
Pretoria 0001
South Africa

Form of manuscript

Manuscripts should be in double-space typing on one side only of sheets of A4 size with wide margins. The original ribbon copy of the typed manuscript should be submitted. Authors should write concisely.

The first page should include the article title (which should be brief), the author's name, and the affiliation and address. Each paper must be accompanied by a summary of less than 200 words which will be printed immediately below the title at the beginning of the paper, together with an appropriate key word list and a list of relevant Computing Review categories.

Tables and figures

Illustrations and tables should not be included in the text, although the author should indicate the desired location of each in the printed text. Tables should be typed on separate sheets and should be numbered consecutively and titled.

Illustrations should also be supplied on separate sheets, and each should be clearly identified on the back in pencil with the Author's name and figure number. Original line drawings (not photoprints) should be submitted and should include all relevant details. Drawings, etc., should be submitted and should include all relevant details. Drawings, etc., should be about twice the final size required and lettering must be clear and "open" and sufficiently large to permit the necessary reduction of size in block-making.

Where photographs are submitted, glossy bromide prints are required. If words or numbers are to appear on a photograph, two prints should be sent, the lettering being clearly indicated on one print only. Computer programs or output should be given on clear original printouts and preferably not on lined paper so that they can be reproduced photographically.

Figure legends should be typed on a separate sheet and placed at the end of the manuscript.

Symbols

Mathematical and other symbols may be either handwritten or typewritten. Greek letters and unusual symbols should be identified in the margin. Distinction should be made between capital and lower case letters between the letter O and zero; between the letter l, the number one and prime; between K and kappa.

References

References should be listed at the end of the manuscript in alphabetical order of author's name, and cited in the text by number in square brackets. Journal references should be arranged thus:

1. ASHCROFT, E. and MANNA, Z. (1972). The Translation of 'GOTO' Programs to 'WHILE' Programs, in *Proceedings of IFIP Congress 71*, North-Holland, Amsterdam, 250-255.
2. BÖHM, C. and JACOPINI, G. (1966). Flow Diagrams, Turing Machines and Languages with only Two Formation Rules, *Comm. ACM*, 9, 366-371.
3. GINSBURG, S. (1966). *Mathematical Theory of context-free Languages*, McGraw Hill, New York.

Proofs and reprints

Galley proofs will be sent to the author to ensure that the papers have been correctly set up in type and not for the addition of new material or amendment of texts. Excessive alterations may have to be disallowed or the cost charged against the author. Corrected galley proofs, together with the original typescript, must be returned to the editor within three days to minimize the risk of the author's contribution having to be held over to a later issue.

Fifty reprints of each article will be supplied free of charge. Additional copies may be purchased on a reprint order form which will accompany the proofs.

Only original papers will be accepted, and copyright in published papers will be vested in the publisher.

Letters

A section of "Letters to the Editor" (each limited to about 500 words) will provide a forum for discussion of recent problems.

Hierdie notas is ook in Afrikaans verkrygbaar.

Quaestiones Informaticae



Contents/Inhoud

Very High Speed Graphics Work-Station*	1
J Jablonski and H J Dijkman	
Documenting a Database System with a Database*	5
D A Hunter	
Program Design as the Tool of Preference in Computer Literacy Education, Experience, Incentives, Implications*	9
J M Richfield	
A Virtual Multiprocessor Computer Design*	15
T Turton	
A Context Sensitive Metalanguage for Intelligent Editors*	21
S M Kaplan	
Concurrency: An Easier Way to Program*	25
C S M Mueller	

*Presented at the Third South African Computer Symposium held on
14th to 16th September, 1983.