2 November 1981

MEMORANDUM FOR: Members, Technical and Scientific Facilities Working Group

25X1 FROM: [redacted] Chairman

SUBJECT: Working Group Information

1. Background

As you are aware, the IHSA has requested your participation in a 2-3 week effort to help them identify the direction the Agency should pursue in the areas of Technical and Scientific Facilities that will be required in the '85-'89 timeframe.

The IHSA has been directed to develop a strategic plan for the Agency's Information Handling systems by the end of August 1982. To accomplish this task within the time constraints, they have devised a four phase approach. In the first phase, the current one, a series of user oriented working groups will discuss and clarify the goals or objectives of Information Handling from their perspectives. The second phase will require other working groups consisting primarily of IH providers to address how the goals identified in the first phase may be implemented within technical, budgetary and other resource realities. The last two phases are concerned with preparing and coordinating the draft and final versions of the Strategic Plan. A schedule for the Strategic Plan development is attached for your information.

2. IHSA Point Papers

The IHSA has prepared three discussion papers that can be used as a basis for focusing our views and structure our product. In addition to the background information provided there are a number of specific questions which will require our

[redacted] 25X1

response. Because of the limited amount of time within this phase you are requested to give as much thought as reasonably possible to these questions and to the formulation of goals prior to our gathering on 9 November. Any initial quantification of facilities required or documentation supporting specific contentions will be appreciated.

3. Some Guidelines

a. The time frame for goal implementation is 1985 through 1989.

b. Our focus should be on required facilities and goals, not on solutions or implementation methods.

c. The fundamental thrust of our pursuit is to ascertain (as specifically as possible) the technical and scientific capabilities we require for our efficient functioning during the target time period.

d. Recommend specific changes to existing systems/capabilities are not our concern.

e. It is not fruitful in this pahse of the effort to pursue "general goodness" concerns, such as commonality and interoperability, because these are aspects of IHS implementation planning. They will be major concerns at that time.

f. In addition to the questions and goals posed, any additional topics which you believe should be addressed in this area will be welcomed.

4. I look forward to an informative and productive association. If you have any questions or observations concerning this matter before our meeting on November 9th, please contact me [redacted]

25X1

[redacted]

25X1

Attachments:
As Stated

AGENDA

Technical and Scientific Facilities

Monday, 9 November 1981

Room 2E-29

0900 - Welcome and Introduction
Working Group Chairman, [redacted] NFAC 25X1

0915 - Discussion of the Computer Aided Design and Interactive Graphics Point Paper. Comment and Review

1015 - Break

1030 - Presentation on Economic Modeling and Modeling Issues
25X1 [redacted] NFAC

1130 - Lunch

1300 - Discussion of Modeling and Mathematical Analysis Point Paper. Comment and Review

1415 - Break

1430 - Presentation on Array Processor Assessment
25X1 [redacted] ODP

1530 - Discussion of Special Machinery Point Paper. Comment and Review

1630 - Adjourn

Tuesday, 10 November 1981

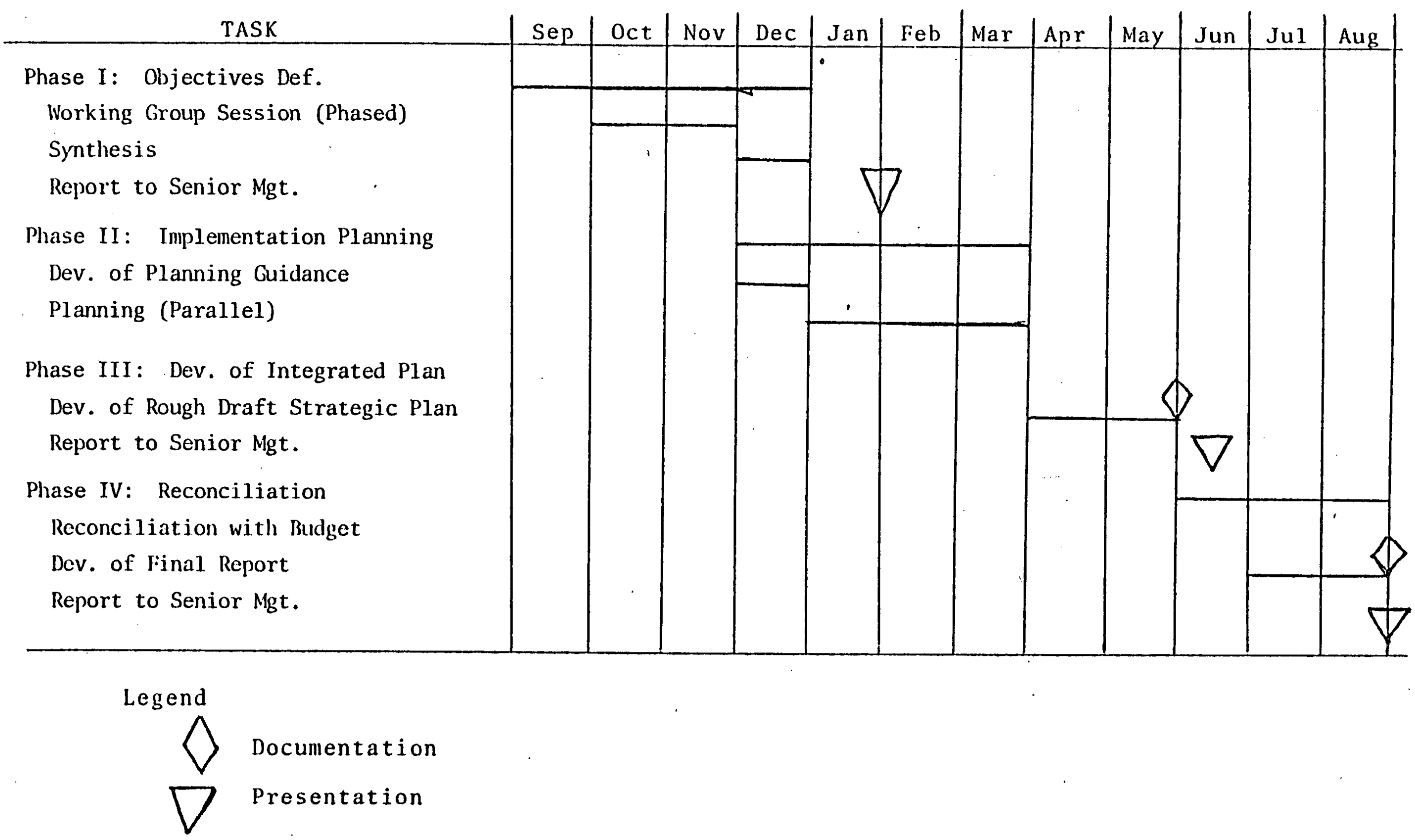
Room 4F-31

0945 - Discussion of Tentative Goals and Objectives. Instructions for Working Group
25X1 Chairman, [redacted]

25X1

Approved For Release 2003/08/13 : CIA-RDP84B00890R000400060009-1

PLAN FOR IHSs STRATEGIC PLAN



Approved For Release 2003/08/13 : CIA-RDP84B00890R000400060009-1

OVERVIEW PAPER

Technical and Scientific Facilities

I. Background

In this era of increased automation, the Agency increasingly relies on "computerized" methods for meeting its objectives. This is especially apparent in the areas which recieve high visibility due to commercial advances and areas which are particularly germane to the Agency's mission. Office automation, information dissemination, and security all come to mind and we see general functions or systems like "Word Processors" and SAFE systems being used and/or implemented.

However, the areas of specific user needs, while often less visible, are also impacted by technical advances. The Scientific and Technical Facilities within the Agency which are available to users must be assessed as to their adequacy and use. If we are to take advantage of these advances in our analytical process we must be aware of them and define our objectives and goals in using them.

II. Scope

This area has been broken down into three subjects:

- -- Computer Aided Design and Interactive Graphics
- -- Modeling and Mathematical Analysis
- -- Special Machinery

Although there is not a clear break between them, they are believed to encompass a large portion of the functionalities perceived to be needed to support the Agency's analytical mission. Additional topics may be included if they are deemed relevant.

III. Approach

Attached are point papers covering each area. They are intended to provide background information as well as to focus on issues pertinent to the Agency. These papers will be used as "talking papers" in the working group and it is hoped they will elicit the views of the various users of these facilities.

Within each paper are relatively specific questions concerning each area. To the extent possible the questions are intended to quantify specific user needs in the specific areas. It should be emphasized that this need assessment is not a commitment and does not guarantee a capability. The goals and objectives we have to establish will be used as general criteria toward which the Agency can proceed.

Approved For Release 2003/08/13 : CIA-RDP84B00890R000400060009-1

SECRET

Attached also are copies of the description of a high speed parallel processor and the description of a parallel processing technique used for graphic purposes. These are included for your information with respect to the subject matter and to illustrate that these areas are not trivial matters. Needs for these facilities will dictate that the Agency acquire and maintain the expertise necessary to efficiently implement them as we move from the "office automation" implementation era into the era of sophisticated technical processing.

IV. Top Level Questions

The governing question for this issue area is the magnitude and character of the technical processing requirement foreseen for the 1985-1989 time period. If it grows as significantly as the internal needs and external technological advances indicate it might, then accommodating it is going to require a sharply focused response. Such a response will involve investment, in new types of hardware, architectural innovation, new software, and special personnel resources. The implementation issues are complex, and would be addressed in the next phase of this strategic planning.

The more specific aspects of this concern are:

-- What scientific and mathematical facilities do the Agency require to meet analytical needs in the 1985-1989 timeframe? What is their scope, expected use, and relative level of need?

-- Given that there is a need for specialized scientific and mathematical facilities, what steps should be taken to acquire the expertise to define, acquire, install and maintain them? What is the funding, planning, timing, and environmental requirements necessary to meet these needs?

-- In rough terms, what computer capabilities (power, speed, systems, hardware, etc.) are required to meet these needs? What qualitative aspects in terms of growth, size of user community and changed capabilities are perceived?

Technical and Scientific Facilities

Computer-Aided Design and Interactive Graphics

I. Scope

Computer-Aided Design (CAD) is that field of computer application which supports the analyst (user) by automating the creation, storage, retrieval, display, and manipulation of variables within the design process. For purposes of this paper this is limited to physical variables, such as structure, equipment, and personnel, as opposed to non-material entities such as money, data, and policy. The result of CAD is normally a visual display or an input to some other process, for example, assembly drawings for manufacture, or resource allocations for project plans.

Interactive graphics comprises the computer-based capability to graphically display the results of analysis and to manipulate them to meet the user's needs. Included is the capability to present overlay information on a map or chart base and standard graphical presentation of information, such as bar and pie charts, graphs, and tabulations. Often information that is presented is derived from files and/or data bases which are then manipulated as a result of the graphically displayed data. Data manipulation techniques that are used in creating these displays are often complex (regression analysis, orbit smoothing, etc.).

The Agency's requirement for a CAD capability is as yet undefined. An effort has been initiated by ODP to examine CAD concepts and functional requirements. That effort is expected to culminate in the establishment of firm specific functional requirements for particular CAD systems.

However, an overall Agency capability in which specific CAD requirements can be implemented is not being planned. The fact that the Agency does have functions that can be supported by a CAD capability and the fact that advances are being made in CAD technology dictate that we examine this field to determine the future environment that is required.

II. Status

Applications in these areas can be characterized into the following general areas:

-- Interactive Graphics

This capability provides the user with the facility to display statistical information graphically (graphs, charts, tables, etc.). A VDT (or VDU) interface permits rapid manipulation of these displays. Often the source of data for input to these displays is a file or data base of specific information.

-- Building Description/Maintenance

Systems of this type aid in the physical planning process including space layout, utility movement (power, water, etc.), engineering, etc. These systems are useful in designing changes to existing structure internals in that changes can be assessed as to impact, cost, etc.

-- Cartographics

This aspect of CAD is concerned with the generation, storage and use of map information. It can be used to study/manipulate physical and political entities and is often used in conjunction with other variables and factors, water, economics, etc.

-- Engineering

This field offers a wide range of CAD capabilities in very diverse areas. Fundamentally, the computer is employed in the design process aiding especially in the spacial relationship of various components. This can be used in publication planning, individual component design, chip design, etc. These techniques can also be used in the "backward engineering" of systems, processes, and components to ascertain their particular constituents.

-- Process Control and Management

These systems permit the planning of moving processes (traffic, heat, work, etc.). Further, they provide for determination of supply definition, resource allocation, capacity limits, etc. As such they can be used to plan and follow production systems allowing management to change and reallocate resources as needed.

Some commercial applications of CAD are now quite sophisticated. In the heavy manufacturing area there are systems which provide "complete" automation. These permit the user to design equipment using a computer terminal and then use the results as input to the manufacturing process where various equipment is "driven" by computer commands. This is Computer Aided Manufacturing (CAM) and often the two areas are referred to as one (CAD/CAM). Manufacturers in portions of the shipbuilding and aircraft industries are currently using this technique.

B. Agency Status

There is a limited application of CAD and interactive graphic capabilities within the Agency at this time. Several interactive graphics applications exist, notably the TACK and TAD systems. These utilize map data bases and display pertinent information with them in an interactive mode. Other specific graphic applications exist, among them:

- RAMIS has a graphic filing and display capability, although not interactive and many systems present graphic output reporting.

- NPIC uses a CAD system developing some order of battle information. Using observed data from imagery they use computerized techniques to ascertain previously unknown variables.

- OCO's Cartography division uses computers in designing various products. These use computerized base maps in some cases to provide the basis of other displays. Graphical, text, and briefing materials are also pre pared using the GENIGRAPHICS computer aided design system, which will later be linked to VM as a data source.

Some project oriented offices have used CAD in hardware development to a limited extent.

III. Issues

There appears to be a great potential for use of CAD and interactive graphics in the Agency's normal functioning. Interactive graphics can be used in the analysis area in presenting the effects of changes in dependent variables. Cartographic capabilities can also be integrated into this process thus displaying visually the status of and effects on political/ military/etc. situations.

CAD and interactive graphics present significant processing requirements that are comparable to scientific and technical processing. Even today, with our relatively minimal use of such facilities we can see the effects of the unique machine loading that they represent. Three-dimensional figures are manipulated quite slowly on our general purpose data processing machinery. Two-dimensional figures are enhanced, expanded, or rotated rather slowly.

The mathematical operations involved are chiefly large matrix operations and fourier transforms. These are the types of vector arithmetic that point towards array and parallel processors, just as in modeling and scientific analyses. Thus a significant automated graphics requirement is likely to push us towards special scientific capabilities, just as a significant growth in modeling and scientific analyses will. Attachment A, from the March 1981 issue of IEEE's Spectrum, describes some of the current thinking with respect to complex graphics implementation on array processors.

1. What is the Potential of CAD in Backward Engineering of Foreign Systems and in Engineering of New Agency Systems?

Backward engineering is the process of determining a systems design configuration and internal functioning from external observations. CAD could provide the same assistance to backwards engineering by analysts dealing with weapons systems, military hardware, and production processes as it does to manufacturers in developing and producing the system.

2. What is the Potential of Interactive Graphics to Support Cartographic Functions?

Interactive graphics support could also be used more extensively in the publication production area. Maps, charts, and reports figures, etc., today are largely produced using "cut and paste" techniques. Publication production using completely automated techniques is broadly applied in the commercial arena to reduce costs and improve the quality of the product and should be pursued within the Agency. Additionally, graphical representation can be used to monitor the poduction process itself, thus giving management an insight into, and better control of, each aspect of production.

3. What is the Potential of CAD to Support Resource Management Functions in the Agency?

Computer support can also be used in the resource management area to graphically depict relative needs and availability of expertise, expendibles, and processing functions. PERT-based techniques are integrated with resource allocation and scheduling functionalities in the CAD environment. Their application, especially to rather complex projects, would aid management in ascertaining, allocating and scheduling resource needs.

In the area of physical design, implementation, and maintenance many CAD tools exist. In the physical plant area the Agency could benefit from a coordinated CAD system to track all aspects of the physical composition of buildings. Security, maintenance and logistical components would benefit from an integrated CAD system depicting the location of all power and communications lines, safes, phones, people, etc. This could be a shared system between the various components and would greatly facilitate the building maintenance and change process.

We are experiencing the same problems that forced the application of the CAD techniques in the commercial environment: inadequate numbers of qualified people to deal with developing and updating drawings in an increasingly complex environment and a records validity problem associated with the compounding of the normal human error rate in that same increasingly complex environment. The cost of obtaining additional personnel to cope manually with the consequences of the geometrically increasing complexity of the environment are greater than automating. Automating, furthermore, postures the organization to continue to cope with ever increasing complexity.

IV. Questions

With respect to CAD and interactive graphic capabilities what are the perceived functions which they could support and what are their respective relative importance?

- Support in the analytical process

  Statistical presentations
  Mapping displays
  Relative force displays
  Economic trending

- Production/Management Control

- Engineering Support (including backward engineering)

- Publication Design and Production

- Building Logistics

- Communications Systems

To what extent is a "general" CAD and/or interactive graphic capability needed to support relatively small and unique applications?

What is the need for the Agency to acquire a body of expertise in the CAD area?

Technical and Scientific Facilities **WORKING PAPER**

Modeling and Mathematical Analysis

I. Scope

The concern of the strategic planning for modeling and mathematical analysis capabilities is the need for specific, required functionalities to provide computer assistance to analysts in assessing external phenomenon. The application of these tools is expected to steadily and rapidly increase because they provide the capability to integrate the effects of very large arrays of observables or consider a multitude of "cases" describing a particular entity, function, or system. When appropriate, they permit the continuing cycling of cases necessary to arrive at a "best" solution.

Modeling can be used in analytical processes ranging from economic and political to hardware and behavior. Of principal concern to this assessment are the "larger" models, either in terms of input data or complexity and scope of the analytic processes. Many of the large models have significant on-line data base and applications software storage implications.

II. Status

A. Current

Currently the Agency uses modeling techniques in many intelligence production cycles. They are used primarily in NFAC by the various production offices. By nature they are analytic support tools and therefore usually tailored by a user to his specific needs. The NFAC Five Year ADP Plan (20 May 1981) cites many current modeling "systems" and "tools" among them:

- Economic

  These models cover a broad range of economies and are used in the assessment of individual country economies as well as regional assessments. Included are models which follow energy production and use. The TROLL system is a major tool in this environment.

- Political

  Models here are used to assess social change, attitudes, political instability, etc. Techniques used include scaling, Bayesian analysis, cross-impact studies, election forecasting, and social simulation models.

- Strategic

  Assessment of Soviet and Chinese military costs and expenditures are produced. Force strengths are calculated and refined for trade-off studies (NATO vs Warsaw Pact, for example).

- Systems

  Modeling is used extensively in defining and assessing particular systems and hardware. Aircraft, ships, antennas, etc. are examined and particular variables are used to determine others. These, in general, are specific models created for a particular device, system, or activity.

- Resources

  Natural and human caused phenomenon are studied using models including crop production, meteorological effects, transportation, fuels, and metals. The recently developed Meteorological Agronomical Geographic Analysis System (MAGAS) is used for many applications in this area.

Except for laboratory environments, such as those that are found in facilities like OSO's signal analysis center, our modeling and scientific analysis are done on general purpose computer machinery. Primarily, this machinery and the system software that drives it are designed to do data processing. Such machinery currently adequately supports the need, but principally because of the relatively low level of scientific processing compared to to the level of general data processing.

Because the current machinery is designed principally for general data processing, it suffers from several disadvantages in doing scientific processing. One is that it has inadequate precision for such functions as inverting very large matrices or integrating equations to determine trajectories with great precision. As a consequence, such machinery is normally used in a double precision mode for such functions. Double precision typically slows down the general purpose computer by significant factors. It is thus, relatively inefficient in doing scientific processing. A second problem is that the machinery is not designed for the basic characteristics of scientific processing versus general data processing. Scientific processing lends itself to a significant degree of parallelism in processing. If advantage is taken of this characteristic, much more efficient processing can be done.

Some hardware that is commercially available is designed to support scientific analyses, and does a much more efficient job than general data processing machinery with existing applications packages.

Principally this superior performance is achieved through greater precision, which avoids the problems of double precision, and through a basic parallelism. The latter is provided by such devices as separate memory and data caches, and an ability to parallel I/O and arithmetic processing.

To take advantage of array processors, however, new software may be required. Such software has to define vector operations so that array processors can be directed to perform parallel operations. This means that if we see ourselves moving in this direction there is much more required than just installing special machinery. New software is required, and the higher order language employed is a significant factor in the effectiveness of the implementation. FORTRAN, for example, supports parallel processing. The new DoD language, Ada, is specifically designed to support parallel processing, which is a major requirement in the embedded systems for which it was developed. This language, and its support environment, are expected to be fully debugged and mandated on U.S. Army systems beginning on 1 January 1983. This timing may be a bit optimistic, but reflects the commitment to and support of this development by DoD. The most pessimistic estimates of the availability of a "clean" Ada and its environment are two years later.

B. Trends

There is a strong indication that the use of models, modeling techniques, and mathematical analysis will increase in the future. Developments in the private sector will provide more such tools and methods and make them more widespread and understood.

The appreciation by analysts of the capabilities of analytic and modeling tools is rapidly growing. Increasingly, they are found essential to synthesizing a vast array of data and extracting from it precious information concerning capabilities, problems, and even intentions. What has frequently been a barrier to these applications is the ability of the analyst to interface first with the IHS via a VDT, and then his ability to understand and operate the model. With the academic training that technical students now all receive in computational techniques, the increasing presence of such trained individuals on our staff, and the changing attitude by older analysts towards such tools, this barrier is rapidly diminishing. So many individual analytic successes are now associated with the application of such tools that there is a general appreciation that their exploitation is a big factor in superior job performance. This is an individual operational factor that is going to be a powerful driver in the increased applications of models in the future.

III. Issues

Modeling and mathematical analysis techniques can and do impact computing resources greatly. Sometimes this impact is observed by the fact that the resource demands of most models are generally relatively light over the periodic accounting periods. When they are running, however, their demands may be quite high. In fact, operating restrictions are frequently placed against models because of their potential to disrupt other IHS resource users. If the applications of modeling and mathematical analysis are seen as growing significantly in the planning period--particularly relative to data processing--then specific architectural and organizational upgrades are indicated.

Issue 1. What are the specific areas in which modeling and mathematical analysis may be used?

Specific areas in which these tools may be used are of interest. Especially important is the definition of analytical areas wherein they are not being significantly used today, but are likely to be exploited in the future. This will allow the Agency to procure functional capabilities of a general nature which can then be used for specific applications.

Issue 2. What expertise is required to take advantage of these techniques?

Generally, these are complex and detailed areas of discipline. The time and effort required to understand and use them is great and leads to specialization. Comments have been made by Agency experts that is takes a very large amount of time to become fully knowledgeable and capable in using some of the more sophisticated models. Time periods in the range of one to three years have been mentioned. Such time periods present major career and organizational problems to the Agency. The Agency needs to figure out how to deal with the expertise problem. What expertise do we need and for what, how diffuse or concentrated should it be, and how do we make the necessary concentration attractive in a career sense?

Issue 3. What problems are there in acquiring and using these special techniques?

Models and analytical methods can present special problems in their procurement, installation and operation. They may require a special DBMS or language, or they may be developed to run on different machinery than we have. They may require special expertise to operate and maintain. An assessment of current problems in these areas is needed in planning the future environment.

IV. Questions

A. Resources

What changes/increases are envisioned in the areas of modeling and mathematical analysis? Quantify, to the extent possible, the number of models and mathematical analysis "systems" envisioned and quantify the amount of computer resources required to meet these needs. (See Table 1)

Have the costs in procuring these functionalities been incorporated in other long range planning?

B. Analytic Tools and Techniques

What new/changed techniques are envisioned? Specifically, in what analytical areas are changes envisioned and what new techniques do they require? For example, is more emphasis needed in statistical modeling vice trajectory analysis? Prioritize needs if possible.

C. Problems

What problems have been encountered in acquiring and using these techniques in the past and what are envisioned in the future? Specifically, cover procurement, installation and use problems.

Why does it take so long to develop a thorough understanding and facility in some models; are we actually developing the professional's analytic skills to the level of the model? Are the models poorly documented? Are the models poorly structured so that it is very difficult to undertand them, even given a good understanding of the theory? Do the models have unnecessarily complex input data and control requirements or do they have too many options for the typical Agency application?

D. Services

What need exists for the management of model and mathematical analysis capabilities--their procurement, maintenance and use? Consider their sources, costs, and expected uses.

TABLE 1

Systems/Functions Impacted by Technical and Scientific Facilities

System/Function	Description/Comment
TADS	Telemetry Analysis Display System; Response time and precision needed for Vector Arithmetic and Fast Fourier Transformation
SCAM II	Soviet Cost Analysis Model; Larger data base and precision needed
TROLL	General Economic Model; Increased size (to 10,000 equations) and speed required; Possible use of Array Processor
CHALLENGE	Oil Reservoir Model; Large core and CPU requirement; CRAY vector processor used in development
MAGAS	Manipulation of Meterological Data; Multiple Instruction Multiple Data (MIMD) capability desired; Display capability important
RADAR	RADAR Signature Data Analysis; Tenfold throughput increase desired; High precision required
TRAJ	Trajectory Analysis; Higher throughput desired; High I/O rates
SOSAG	Nuclear Weapons Simulation Model; High CPU use
Networking Models	ORD projects - JAWS (anti-satellite model), Soviet Transportation, Refinery, Hydrology, Agriculture, CW; CRAY-1 considered
Hardware/Systems Models	MVS, PRIME, TRACE, TAPEST, KADRE; Models to simulate/define hardware/systems; Possible Backward Engineering need; Large CPU potential

Systems/Functions Impacted by Technical and Scientific Facilities

System/Function	Description/Comment
Energy	Models needed to trace/simulate/follow world and regional energy production and use
Economic	Specific country/region models required; often time-phased tied to production cycles; See also TROLL
Warfare	Several models exist; often tied to warfare types (conventional, nuclear, naval, etc.)
Political	Models assessing political situations, potentials; election forecasting, threat analysis, stability assessment included
Commodity	Modeling and tracing of grain, metals, drugs, etc.; includes supply/demand analysis, status, production, etc.
CAD/CAM	Design, graphical presentation, data manipulation, etc.; specific areas of use desired

Technical and Scientific Facilities

SPECIAL MACHINERY

I. Scope

For our purposes, special machinery is considered to be of two classes:

- Class I - General purpose computers (generally mini's) dedicated to special applications, such as signal processing.
- Class 2 - Machines which are not general purpose but are in fact special machines optimized for a specific functionality, such as array processors.

In following sections status information and planning issues are presented.

II. Status

A. Class I

An inventory conducted in March 81 determined that the Agency had 71 minicomputers, most being used for specialized applications. The count by directorate was:

25X1

DDS&T	[redacted]
DDA	
NFAC	
DDO	

25X1 [redacted]

DDA uses their minicomputers primarily for information systems related to logistics, personnel, and medical applications; other uses include text editing and composition, data entry, and training. The ODP initiative to offer GIMS on a minicomputer is noteworthy; this provides the flexibility for GIMS applications to reside either on a central or dedicated facility without conversion difficulties.

DDO uses their minicomputers for data entry, CRAFT, language translation, and compartmented information storage and retrieval.

NFAC applications include plotters, digitizers, library automation, high speed text search, crisis management, and data analysis.

The reasons for using dedicated facilities rather than central computing facilities typically include:

- o The special nature of the application (e.g., signal analysis or text search)
- o Extraordinary requirement for responsiveness
- o Sensitivity of information

Acquisitions of dedicated computers are carefully evaluated and controlled by the requesting organizations, OL, ODP, and others. Their operation in performance of the dedicated function for which they were acquired is not a strategic planning concern.

The chief planning concern relative to such machinery is the consequence of applications' growth vis-a-vis the limited power they possess. As the demands for computing power of their resident appplications expand, a classic problem is created. More powerful machinery is needed, but the applications can only consume a portion of the total the new resource would make available. What is more, the more powerful equipment usually requires a higher level of operational support, and of a higher skill level. Cost effectiveness in resource management thus points towards migration of the expanded application to shared central facilities.

B. Class II

Machines of special functionality which are most frequently discussed are data base machines and array or parallel processors. The use of such special machinery could well imply significant changes in the architecture of our current centers. Thus, the value of the inclusion of such machinery has to justify the required investment.

1. Data Base Machines

Data base machines have generally been of interest to information service providers, moreso than to users. Perhaps this is because the thrust of data base machines is to move the basic data base management functions from the current software implementation to a hardware implementation, rather than provide new functionality at the user interface.

The principal benefits to accrue to the user from data base machines are improved performance and reliability. One of the most likely performance payoffs is probably to be derived through the greater efficiency in whole text search. A significant improvement in this area might result in easing or removal of existing constraints concerning personnel and daily access to the functionality.

Data base machines have been a subject of research for years and commercially available products are now beginning to appear. Britton Lee Inc. offered one of the earlier machines (IDM), a device which is being considered for use in CAMS II. Storage Technology Inc. has also announced a product (VSS).

ODP indicated in the Working Group on Information Handling Facilities that they will be studying the applicability of this technology. User supplied incentives to use the technology should be in the form of data base needs (numbers of data bases, sizes, and access rates), also an input to Information Handling Facilities Working Group.

2. Array Processors

Throughout the NFAC Long Range Plan for ADP there are references to increased computational requirements. One office specifically identifies array processors as a solution to their computational needs. Whether or not array processor technology is an appropriate solution is surfaced as an issue in the next section.

For our purpose, the term "array processor" means a single peripheral processor attachable to a general-purpose host computer so that the tandem combination provides a much higher numerical calculation capability than the host alone. An array processor might be viewed as an intermediate step in providing a high performance computational capability; the ultimate, and of course a much more expensive solution would be a supercomputer of the CRAY or CDC 205 class. For certain applications involving sophisticated computations on large arrays, an array processor can be very cost effective compared to doing the same computations on a large general purpose machine. Signal processing is one application, and indeed the Agency is in the process of installing a Floating Point (AP-120A) machine for that application. It has also been suggested that the TADS configuration might well be augmented with an array processor to achieve the additional power needed.

Some of the large telemetry and modeling applications within the Agency may also be suited to such machinery. ODP has researched the need to a limited extent and some perceived needs for an Array Processor do exist.

The attached manufacturer's literature (Attachment B) provides some of the best succinct, clear discussion of the different types of vector processing machinery we have seen. It provides a discussion of the technological features involved and their implications. Hopefully it will provide some insight into the complexity of issues relevant to the planning that will have to be resolved should there be indicated a need for such machinery. There are likely to be substantial organizational and programmatical impacts to user organizations in addition to the obvious architectural concerns relevant to their inclusion by service providers. For those interested in investigating the characteristics of array processors in greater depth, the September 1981 issue of the IEEE's Computer magazine is dedicated to the subject.

III. Issues and Questions

1. Does the projected growth in the modeling and mathematical analysis environment point to a likely need for special scientific machinery?

There are two principal factors which would indicate a future need for special scientific machinery, such as array or parallel processors: the quantity of the work and the unique demands of the work.

The quantity assessment derives from growth projections relative to existing models and functionalities currently resident on the central systems, for example the TROLL model and TADS and from the migration of processing from laboratory environments, such as OSO's, into a centralized environment. The latter occurs as the nature of the required processing becomes more routine, more production-like, and the power of the required processor increases. When the required power increases it is usually true that single applications can only partially utilize the more powerful machinery. For acceptable cost effectiveness, the environment then becomes one of shared machinery usage in a centralized environment.

The second factor, the unique demands of the work, derives from requirements for special machinery capabilities to handle the intended processing. For example, NFAC's TROLL economic model can currently handle up to 1800 simultaneous equations. NFAC has determined that it needs to expand this capability to be able to handle approximately 10,000 simultaneous equations. Such an order of magnitude increase creates signficant computational problems. The number of operations in matrix inversion, for example, goes up geometrically with the number of equations, as does the required precision. Such requirements may exceed the capabilities of available data processing machinery, forcing the acquisition of special machinery.

2. How great a problem is the Agency's thin strength in scientific computation professionals in the central environment?

Today almost all of the development and maintenance on scientific software is done by contractors. Development and maintenance requiring special access is frequently done on the same Agency equipment that is being used for analysis production, e.g., TADS and some TACK applications.

One consequence of the dependency on contractors, and the lack of a separate development environment forced by funding limitations, is that we are unable to run very sensitive data on the machines because of unavoidable contractor access. Granted that we would always like to have a greater depth of scientific computation expertise, the issue is the priority for it.

3. Is there adequate justification for the major software and programmatic undertakings associated with development of a scientific computation environment?

The implications of a special scientific environment go well beyond special processors. Although array processors, as opposed to parallel processors, can process existing FORTRAN code, taking full advantage of such capabilities implies special software. Even in FORTRAN, the software should be specifically written to execute vector processing. For a major program in this area, shifting to languages specifically designed to support such functions, such as Ada, is probably indicated.

Because of the specialized skills required to operate, adapt, and enhance software in this area, there would have to be a cadre of Agency professionals with expert knowledge. This would not be an insignificant personnel requirement on the part of both user and provider organizations.

With a significant workload requiring scientific-type processing security concerns might well dictate the need for separate production and development environments on a continuing basis. The former would be Agency-only; the contractors would have access to the development environment, as well as Agency personnel. This split would make it possible to run extremely sensitive data, now precluded where such a split does not exist.

Attachment A

Fast graphics use parallel techniques

Designers of computer graphics systems exploit parallel processing to provide the speed needed for interactive performance

The main strength of computer graphics is in its ability to exploit the massive parallel processing capacity of human vision—the capacity to perceive almost instantly complex visual patterns. However, until recently, the graphics displays have generally been prepared serially by the computer hardware. Because of the large volume of information to be computed in three-dimensional graphics, this has meant long processing times, a problem not critical for scientific and entertainment applications, but serious for the interactive, real-time systems used in computer-assisted design and in real-time simulations. In these systems, the main trend now is toward the development of parallel processing hardware that can dramatically decrease processing times.

Some large computer graphics systems using parallel processors are already operating, and many others are being developed. While parallel processing systems are currently limited to highly expensive simulators, VLSI designs under development could conceivably bring the cost of sophisticated computer graphics down to the price range of personal computers. Given the rapidly growing demand for interactive graphics and the suitability of VLSI circuits for parallel processing of increasing sophistication, it seems likely that this field will become one of the first to use VLSI technology in a big way. (See "The technologist's own 'super computer'," *Spectrum*, September 1980, pp. 48–51.)

At the same time, new methods are being developed to enter rapidly the large amounts of data in interactive graphics, and these methods, combined with faster hardware, will enlarge the already widespread applications of computer-assisted design.

Complex scenes broken into sections

When used in computer graphics, most parallel processing systems handle complex scenes or images by breaking them into sections. Each section, with its many similar calculations, can be generated in isolation from every other section. In this way special-purpose parallel processors, each handling only a part of the final image, can work far faster than serial processors.

The greatest premium on speed of image generation is in real-time simulations, where extremely complex images must be updated 30 times a second in response to the actions of the user. Such systems are used for the training of jet pilots and other military personnel. It is not surprising therefore that such simulation systems have been among the first computer graphics systems to use parallel processing.

An example is the Computrol system developed by Advanced Technology Systems of Fairlawn, N.J. The system can produce 30 full-color images a second, each using up to 30 000 edges or 10 000 light points. Many moving objects can be displayed, while fog, clouds, textures, and transparent objects can be modeled realistically (Fig. 1). The system permits fairly rapid generation of new data bases—a new airport can be programmed in a few man-days, for example. Computrol is to be used in the F-18 Weapons Tactics Trainer, being built by Hughes Aircraft for the Navy for operation in 1982. It will simulate maneuvering aircraft, terrain, gunfire, and missiles and will be equipped to train two pilots simultaneously. The pilots will be able to maneuver against each other in simulated missions. Similar systems are also used for training the crews of tanks, ships, and commercial aircraft.

Eric J. Lerner Contributing Editor

The detailed architecture of the Computrol system is treated as confidential by Advanced Technology Systems—in fact, most current simulator designs are kept confidential, a practice that has hindered progress in this field. However, the general design of the system has been published, and it gives a good idea of the concepts applied.

The Computrol hardware consists of eight subsystems or blocks. An off-line terminal is used for creating the "world" within which the simulation operates and is connected with a minicomputer that controls the modeling process. A conventional CPU and its associated main memory contain the data base for the simulated world and control the movement of objects through it in three dimensions. Three specialized units are concerned with converting the three-dimensional world into a two-dimensional graphic representation on a CRT screen. The frame processor projects the three-dimensional world into the appropriate two-dimensional field of view and simultaneously converts the objects into edge-based descriptions—that is, the edges define the borders between differently colored patches. The raster processor calculates the intercepts between these edges and each scanline on the CRT. Finally, the pixel processor takes the intercepts of the visible edges, together with shading data, and generates the color and intensity of each pixel on the scanline.

The main parallel processing features are in specialized processors. The frame processor uses parallel arithmetic units to perform the calculations that transform world coordinates in the data base to eye coordinates centered on the apparent viewpoint of the trainee. Similarly, the raster processor has parallel circuits to calculate the intercepts, and each subsection of the pixel processor does identical calculations for a CRT subsection.

Applications to CAD

While the current application of parallel processing to computer graphics is mainly limited to simulations, the same techniques would be of great use in computer-assisted design if the hardware could be made sufficiently cheap. This has become an increasingly urgent necessity, since CAD systems are now using three-dimensional techniques that, with existing hardware, tend to slow processing radically and prevent easy interaction with the user.

/0300-0054500.75©1981 IEEE

Until recently computer-assisted design has been applied mostly in the two-dimensional world of electronic design, but now it is expanding rapidly into three-dimensional applications in mechanical engineering and architecture. A typical commercial package, Synthavision, developed by the Mathematical Applications Group Inc. of Elmsford, N.Y., gives the user the ability to create any arbitrary solid, to manipulate it and view it from any angle, and to obtain its volume, weight, center of gravity, moments of inertia, and other geometrical characteristics.

General Electric has begun using Synthavision in the design of mechanical components, such as gear trains. Similar systems are being used by the Oak Ridge National Laboratory in Tennessee and the Lawrence Livermore National Laboratory in California in the design of complex magnets for controlled fusion experiments. Synthavision is also used in computer animation applications. In some cases, the computer-assisted design of components has been supplemented by computer movie simulation of what happens to the components under stress, thus automating both the design and test-evaluation procedures.

Synthavision and most similar systems use built-up complex solids by using a set of primitive shapes, such as cylinders, spheres, cubes, and wedges, as well as arbitrary shapes that can be parametrically specified. The solids appear on the CRT screen as if illuminated from a specified angle, and the user can adjust the reflective characteristic of the object's surface to simulate diffuse or specular reflectance.

Such techniques are also coming into use in construction engineering and landscaping. A group at the University of Massachusetts has developed a program called Ecosite that enables the user to construct land forms, to be used to reform surfaces that have been disrupted by strip coal mining. The system allows a designer to create landfill shapes that will blend naturally into the surrounding topography.

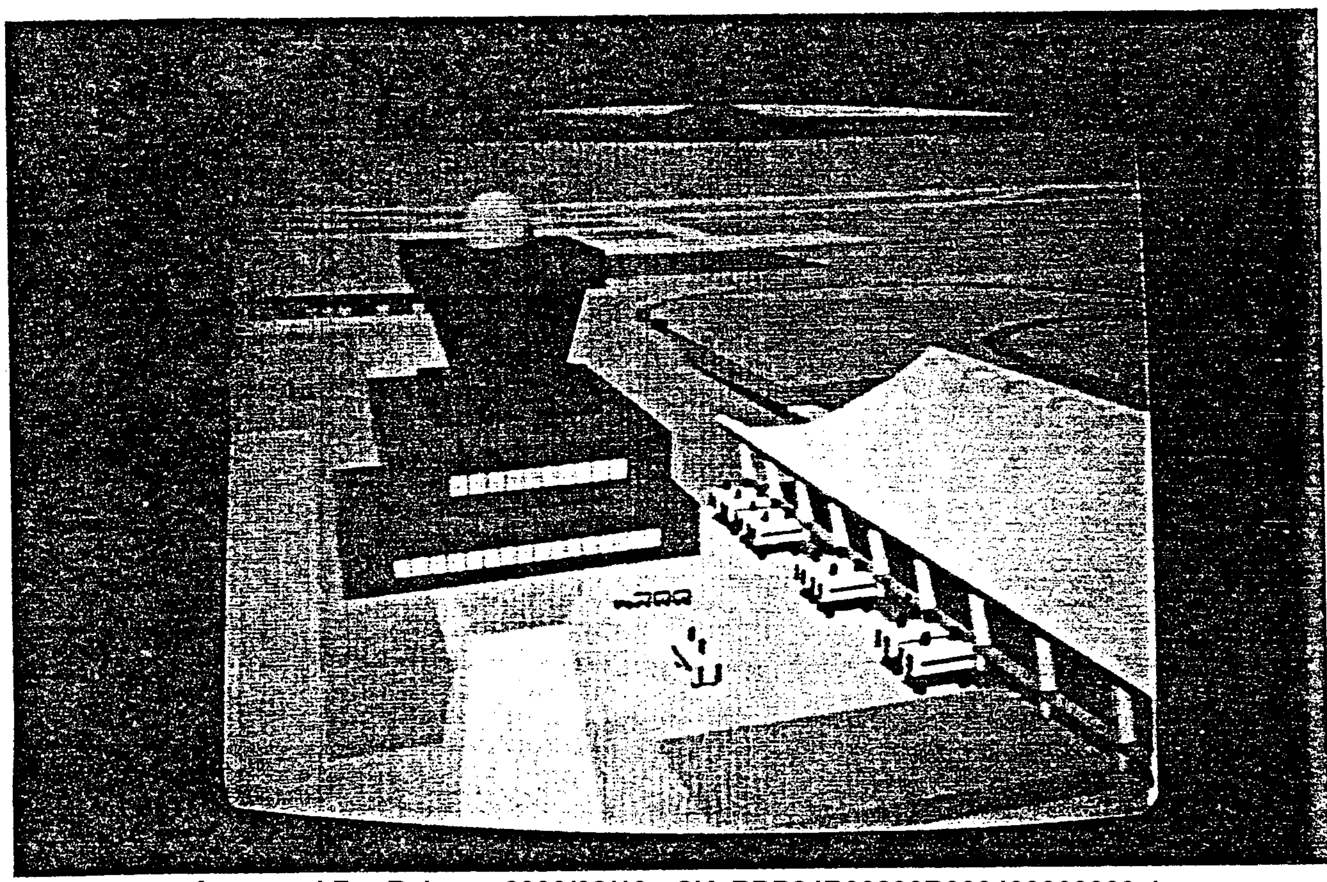
These existing systems, now implemented on conventional computers, would benefit enormously from the higher speed and interactive modes that would become available with the perfection of parallel processing systems now being designed.

A variety of methods of using parallel processors are under development to speed various aspects of computer graphics generation. Much of the work is focused on the hidden surface problem: the elimination of those portions of objects that are obscured by other objects. One particular software approach to this problem, the Z-buffer method, is especially suited to implementation by parallel processors, since the determination of what surface is visible is done independently for each pixel. The Z-buffer (described in "The computer graphics revolution," *Spectrum*, February 1981, pp. 35–39) is a buffer for each pixel of the image that allows only the nearest-object pixel to be entered.

One architecture proposed by Frederic Parke of Case Western Reserve University implements the *Z*-buffer method by splitting the image into regions and feeding the calculated surfaces in each region to separate parallel processors. Each parallel processor then determines the appropriate intensity and color values for each pixel in its area, loading them into the appropriate Z-buffers as it does so. Since only the closest pixels in each raster point will be allowed into the *Z*-buffer, the resulting image will automatically show only the surfaces that should be visible.

This system has a few limitations. Like all *Z*-buffer systems, it has difficulty in dealing with "aliasing"—the tendency of computer graphics systems to turn diagonal lines into staircases

[1] Complex scenes like this are generated by a flier training simulation system called Computrol, developed by Advanced Technology Systems. Computrol uses custom-designed parallel-processing hardware to produce 30 frames of three-dimensional simulation a second. The system costs approximately $2.5 million.



because of the finite dimensions of pixels. Also, the system becomes inefficient if all the objects are concentrated in a few regions, because most of the processors will then be idle.

A second architecture, developed by Henry Fuchs of the University of North Carolina, uses a central broadcast controller to distribute the data on each object to each processor. The data is broken up not by contiguous regions, but according to an interlace pattern, so that each processor handles pixels scattered over the whole of the image. This eliminates the problem of having some processors idle if the objects are concentrated in a certain area. However, this architecture turns out to be considerably slower, in general, than the split regions approach.

Mr. Parke at Case Western Reserve has proposed a hybrid architecture in which the input data is first split into a small number of regions and then distributed within each region, as in the broadcast approach. In this way the disadvantages of both approaches are minimized.

VLSI designs sought to cut costs

The best way to decrease the cost of parallel processing hardware is through VLSI. This approach is being pursued by James Clark and associates at Stanford University. He has designed, and is in the process of fabricating, a highly parallel VLSI computer graphics system consisting of a "geometry engine" and smart image memory.

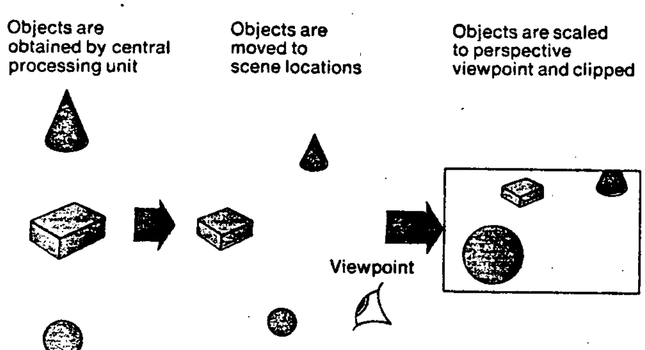
The geometry section of the processor consists of 12 identical geometry engine chips, each containing about 55 000 transistors. The processor performs the basic operations common to practically all computer graphics operations—transformations, clipping, and scaling of two- and three-dimensional polygons. It can perform about four million arithmetical operations a second, processing 900 polygons or 3500 edges every 1/30th of a second.

Each chip has a basically simple architecture, consisting of an arithmetic logic unit, three registers, and a stack, all working together to form four identical functional units. The 12-chip system consists of 1344 copies of a single bit-slice layout.

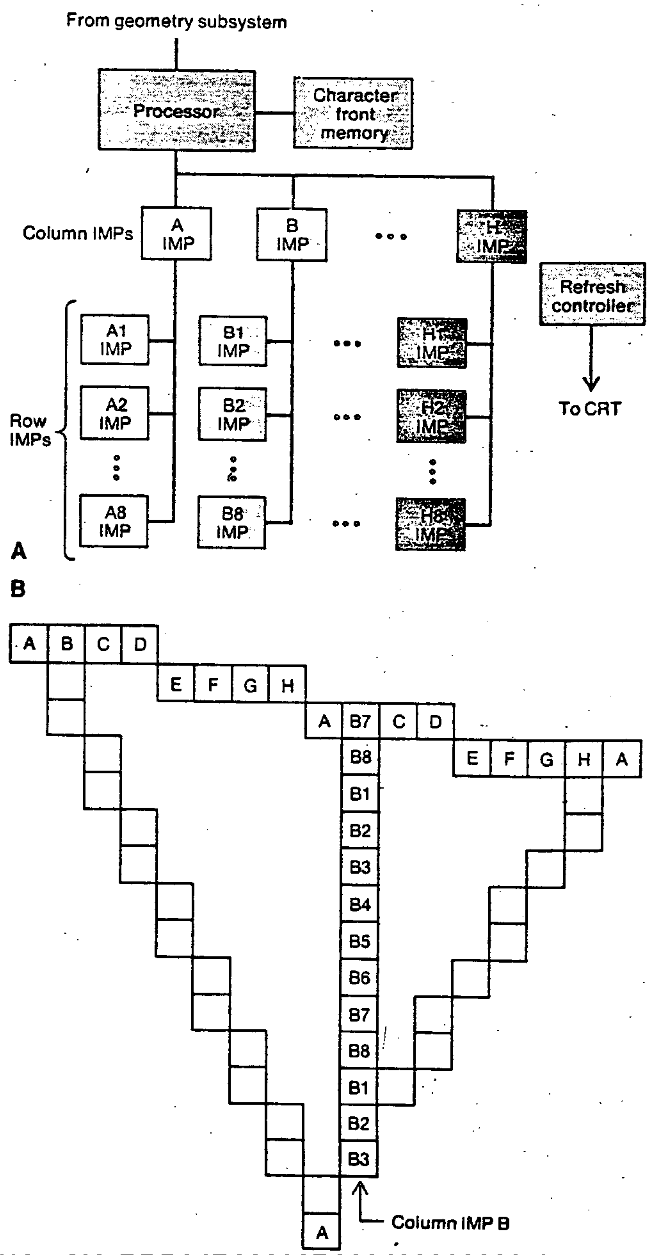
In operation, the geometry unit first receives the coordinates of polygons from a central processor and transforms them into the coordinates centered on the viewer. Four of the chips perform this operation by a combination of 4×4 matrix multiplications and vector dot products that accomplish the necessary rotations, translations, and projections to place the polygons in their locations in image space (Fig. 2). Since each chip has four identical subunits, 16 multiplications are being performed simultaneously in parallel.

The transformed polygons are then passed to the clipping subsystem, which determines what part of each polygon is within the field of view. For the three-dimensional case, the field of view is defined by five or six planes that bound the volume visible through the viewport of the image space. Each chip clips the polygons for one of the bounding planes and then passes it on to the next chip. Each geometry engine compares the coordinates of

[2] The geometry system developed by James Clark and colleagues at Stanford University uses parallel processing in a VLSI architecture to carry out procedures common to nearly all computer graphics. A scene consisting of lines, points, and polygons is first rotated and translated to correspond to the viewing position of the user. The scene is then "clipped" to eliminate those portions outside the viewer's field of view. Then the scene is scaled to fit within the viewing area of a CRT screen. The resulting polygonal coordinates are then passed to a smart image memory.



[3] The memory system for a 1024-by-1024-element display being developed at Stanford University contains eight column image memory processors (IMPs) and 64 row IMPs that convert sections of incoming polygons into alterations of specific pixels in the scan lines of the output display (A). Each row IMP is linked to one or more 16-kb memories, and the system has an output of 160 million bits per second. In operation (B), the edges of the sample triangle shown are scan-converted by the column IMPs and the interior pixels, by the row IMPs. The letters indicate which column IMP has converted each pixel, and the numbers indicate the row IMPs.



Approved For Release 2003/08/13 : CIA-RDP84B00890R000400060009-1

the end points of the polygon edges with the plane equation of the boundary surface. If both coordinates are outside the boundary, the edge is rejected; if both are inside, the edge is passed on to the next chip. If only one coordinate is inside—that is, the edge intersects the boundary—the chip finds the point of intersection. It does this by logarithmic search for the intersection point. Each of the four subunits of the chip computes one coordinate of the midpoint of the edge and determines if that midpoint falls outside or inside the boundary plane. If it falls outside, then the midpoint of the line connecting the inside end point with the original midpoint is then calculated and tested and the cycle repeated until the desired precision of the intersection point is achieved.

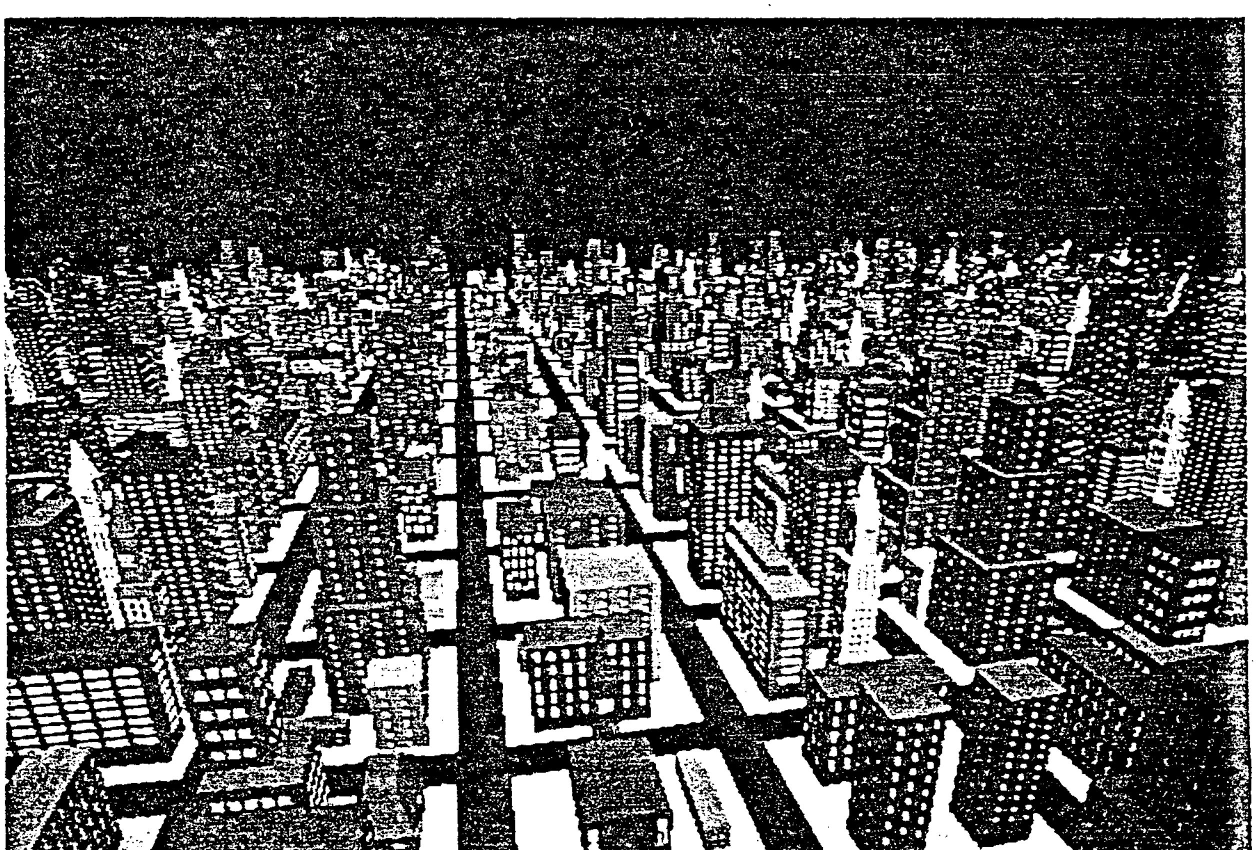
Finally, once the clipping operation is completed, the dimensions of the polygons are scaled to the size of the image viewport —that is, the farther away the objects are, the more the boundary area must be scaled down to the dimensions of the viewing screen for correct perspective. Two chips—one for the x, y scaling and the other for the z, or depth, scaling—perform the division of the coordinates simultaneously and feed the finished results to the smart image memory.

The second part of the computer graphics design is the image memory, a high-performance system for scan conversion. This is the process of determining which pixels on the screen correspond to the calculated images. The image memory is composed of a parent processor, an array of image memory processors, or IMPs, and a refresh controller. The IMP array consists of eight column IMPs and 64 row IMPs, each of the latter being responsible for 16 000 pixels of a 1024 × 1024 array (Fig. 3). The IMPs are connected with the CRT pixels through a two-level hierarchical busing structure, with interleaved processors along each bus. The interleaving is such that for any 8 × 8 array of pixels on the screen, each pixel is controlled by one processor. Thus, as in the Fuch's broadcast scheme at the University of North Carolina, each processor controls pixels scattered across the screen rather than concentrated in a single contiguous area. Each of the IMPs is a single LSI chip that contains two main functions, a linear difference engine and a memory interface processor.

In operation, the geometry engine passes the characteristics and locations of the elementary polygons to the parent processor, a standard microprogrammed chip. The parent processor prepares the polygons for scan conversion and broadcasts the resulting data to all of the column IMPs. The polygons are represented by the coordinates of their vertices. Each column IMP (C-IMP) uses its linear difference engine function to calculate what part of the line falls within the column controlled by its row IMPs and sends this information to the R-IMPs that cover that part of the column. The R-IMPs, in turn, use their linear difference engines to calculate which individual pixels should be altered and send this information to the memory interface processor for storing. At regular intervals, the refresh controller sends a signal to each of the memory interface processors to obtain updates of the new pixel values and uses them to form the new image on the CRT.

The edges of the polygon are thus converted by the C-IMPs to their new values, while the interior is converted by the R-IMPs. This architecture is being modified to implement the shading and coloring algorithms of most polygon systems. The modified architecture will obtain the shading values for the interior of the polygon by interpolating between the values for the edges. In ad-

[4] Procedural modeling employs a set of rules to generate more complex objects from simpler ones and combines both specific data entry and computer-generated repetitions. An example graphic, developed by Charles Csuri and associates at Ohio State University, illustrates how entire city blocks might appear.



dition, the system can be extended to the sort of broadcast Z-buffer hidden surface remover described by Dr. Fuchs.

The Clark architecture shows a conceptual similarity to that of the system developed by Advanced Technology Systems, with the column IMPs performing similar functions to the raster processors and the row IMPs analogous to the pixel processors.

Alternative ideas explored

A number of alternative ideas being developed use parallel processing to speed certain special functions useful in computer graphics. One example is a two-dimensional shift register designed by George Chaikin of the Goddard Space Flight Center office in New York City and Carl Weiman of General Electric.

This device is intended to speed the calculations involved in two common graphic transformations: scale changes, or "zooming," and rotation. In a conventional system, these transformations require many individual calculations to change the coordinates of each pixel in the image. Instead, Mr. Weiman and Mr. Chaikin have proposed using a hard-wired polar logarithmic transformation data channel to convert rotation and scale transformations into translation motions on a shift register. The data channel would connect pixels in the image plane arranged in a polar logarithmic pattern to those in the shift register arranged in a rectangular pattern.

In other words, a circle in the image plane is always mapped into a vertical line in the register, and a radius in the image plane is mapped into a horizontal line in the register. Rotation of the image is achieved when each pixel register is commanded to shift its content to its neighbor above or below. Scale conversion is achieved when each register is commanded to shift its content to its neighbor on the left or right. A single command can thus perform the work of many coordinate calculations.

While parallel processing will markedly speed computer graphics processing, the most efficient use of such savings in time will necessitate faster methods of data entry. Complex design problems in three dimensions often produce difficult problems simply in getting the design concepts into the computer in the first place. A number of techniques for data input and structuring are both easing data entry and simultaneously making some processing tasks more efficient.

Two of the most important techniques are the related approaches of procedural and hierarchical modeling. In procedural modeling, a set of laws is used to generate more complex objects from simpler ones. The hierarchical approach breaks down complex objects into simpler components or simpler representations with less detail.

Using procedural techniques, which combine both specific data entry and computer-generated repetitions where necessary, Charles Csuri and co-workers at Ohio State University have developed a system for the design of buildings by use of standardized components. Computer graphics can then be used to "construct" entire city blocks of a variety of such standardized buildings (Fig. 4). The results give one a realistic view of how the buildings would look in a city. An entire downtown area of two thousand buildings was designed with this system in less than two weeks. Such modularized techniques may have important applications in West Europe, where modularized building is far more common than in the United States. It was, in fact, such techniques that made possible construction of the elaborate data bases in systems such as Computrol.

Hierarchical techniques take into account the fact that as an object becomes more distant, less detail appears, and thus it is a waste of computing power to calculate distant objects to the same precision as nearby ones. In a hierarchical data base, a single object may be represented by a number of representatives, each having greater detail than the previous ones and the more detailed ones being used for when the object is closer to the viewer. Hierarchical representations also speed such processes as hidden surface elimination, since if it is found that an entire object will be obscured in a given image, elaboration of that object to finer degrees of detail will be obviated. Thus, once it is found that one building lies entirely behind another, the exterior windows, doors, and so on in the building will not be calculated at all.

Degree of detail required can be varied according to circumstances. Thus, moving objects can be calculated in less detail than motionless ones, or whole scenes can be less rigorously imaged if the field of view is moving rapidly. Another advantage of hierarchical data sets is in the reduction of memory storage requirements for graphics processors. A working set of images can be formed, consisting only of the images that have, in recent frames of the sequence, been resolvable. This working set can be kept in a fast access memory and only slowly changed or replenished as the field of view changes.

A third technique of importance in efficient data entry is the use of piecewise continuous surfaces, or splines, for defining continuously curved surfaces. In many CAD applications, the manipulation of sculptured surfaces, such as ship hulls, is extremely important, yet with even the fastest processing capabilities, point-by-point entry of such curves is very time-consuming. Spline surfaces simplify the creation of such sculptured surfaces on a computer graphics system.

A B-spline, a widely used type, consists of a network of points, each having associated with it a set of vectors that define the directions of curvature of the surface at the point. The combination of points and vectors can be used to produce smoothly curving surfaces that can have almost arbitrary characteristics. One can modify the surfaces by selecting a given point and either moving it or changing the associated vectors.

The combination of faster input algorithms and the increased speed and decreased cost of parallel-processing hardware will rapidly make very powerful CAD graphics systems widely available. In the next few years, such systems will be becoming a standard tool in engineering.

For further reading

A description of the Computrol system is given by Sam Ranjbaran and Ron Swallow in "Graphics of Complex Images in Training," *Second IEEE Workshop in Picture Data Description and Management,* 1980.

Frederic Parke describes two approaches to parallel processing in "Simulation and Expected Performance Analysis of Multiple Processor Z-Buffer Systems," *SIGGRAPH 1980*, pp. 48–53.

James Clark's designs for VLSI computer graphics systems are outlined in "A VLSI Geometry Processor for Graphics," *Computer*, July 1980, pp. 59–69, and also in "Distributed Processing in a High Performance Smart Image Memory," *Lambda*, Fourth Quarter 1980, pp. 40–45.

Hierarchical data organization is discussed by Steven Rubin and Turner Whitted in "A 3-Dimensional Representation for Fast Rendering of Complex Scenes," *SIGGRAPH 1980*, pp. 110–117.

Robert Marshall *et al*, outline a system for procedural data generation in "Procedure Models for Generating Three-Dimensional Terrain," *SIGGRAPH 1980*, pp. 154–159. The use of B-splines is examined by David Rogers and Steven Satterfield in "B-Spline Surfaces for Ship Hill Design," *SIGGRAPH 1980*, pp. 211–217. ◆

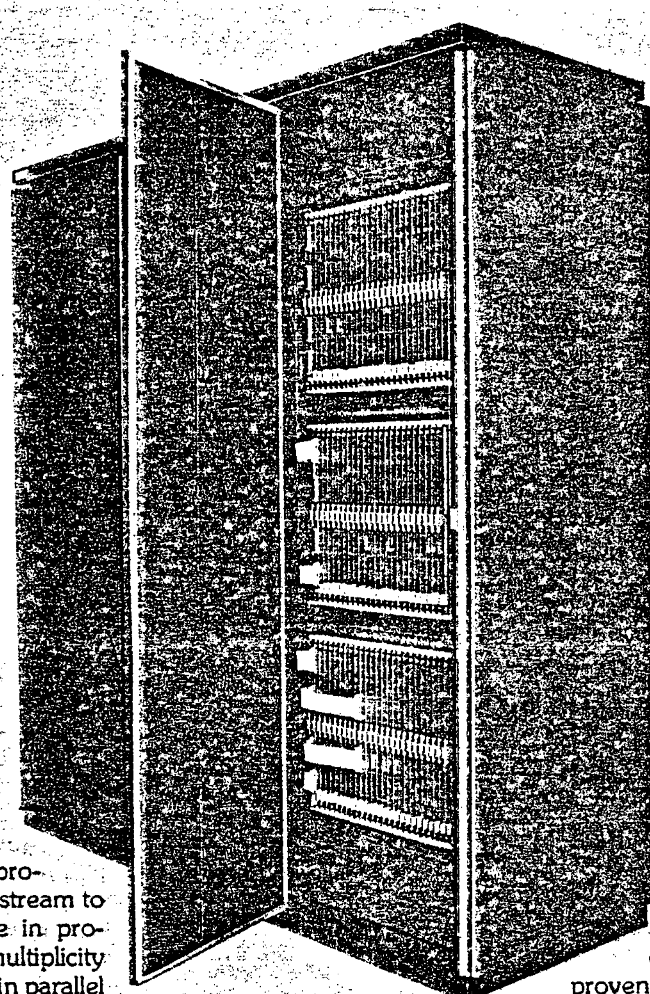
Attachment B

Heterogeneous Element Processor

Denelcor, Inc.

Tomorrows' Computer Is Here . . . Today

Denelcor's Heterogeneous Element Processor (HEP) is a large-scale (64-bit) high-speed digital computer whose architecture makes all other supercomputer architecture obsolete. HEP provides a totally new computing environment: high-speed, parallel processing of heterogeneous data elements. HEP has been designed for use in scientific and/or commercial applications which can effectively utilize processing speeds of ten million to 160 million instructions per second. HEP achieves this throughput because of its design which implements the Multiple Instruction Stream Multiple Data Stream (MIMD) architectural concept for the first time in a commercially available system.



HEP makes available to the user up to 1,024 independent instruction streams or processes, each with its own data stream to be used concurrently for use in programming applications. This multiplicity of instruction streams running in parallel enables and encourages breaking the application into its component parts for parallel processing. Other features of the HEP design provide the synchronization necessary to facilitate cooperation between concurrent processes, and eliminates the precedence delays which often occur when parallel processing is attempted using more conventional data processing equipment. An equal number of Supervisory Processes are available for processing the privileged functions necessary to the support of the User Processes for a total of 2,048 independent instruction streams.

The many capabilities of the HEP hardware are fully supported by HEP System Software so that the potential performance of the system is realized with relative ease. Using the available System Software, programming HEP is very similar to programming a conventional system, and only minimal additional programmer training is required.

In addition to the obvious design goals of fast throughput and the ability to solve very large and complex problems, HEP is designed for ease of operation and to be highly effective across the full range of general-purpose computing applications.

HEP hardware is modular and field expandable.

HEP achieves its high speed performance through advanced architectural concepts rather than through unproven "leading edge technology" electronic components. This provides the user benefits in economy and reliability.

HEP Parallel Fortran is designed for maximum similarity to existing languages, with logical extensions as necessary to implement the advanced features of HEP.

HEP is designed for ease of maintenance in the event of hardware malfunction. Maintainability features are an integral part of the hardware design, including an on-board maintenance diagnostic system which implements an Interactive Maintenance Language for diagnostic purposes.

Heterogeneous Element Processor

Denelcor, Inc.

Evolution of Computer Architecture

The earliest computers executed a single instruction at a time, using a single piece of data. The architecture of these machines, called SISD (for Single Instruction, Single Data Stream) computers, was straight-forward, and well suited to the technology of the times. As technology advanced and computer users required greater performance, SISD machines were made faster and faster, using newer and better components and designs. But a fundamental problem remained. Although the execution of a computer instruction is physically composed of several parts — instruction fetch, operand fetch, execution and result store — the SISD computer could only perform one of these at a time, since each step depended on the completion of the previous one. Thus, three-fourths of the expensive hardware stood idle at any given time, waiting for the rest of the hardware to finish operation.

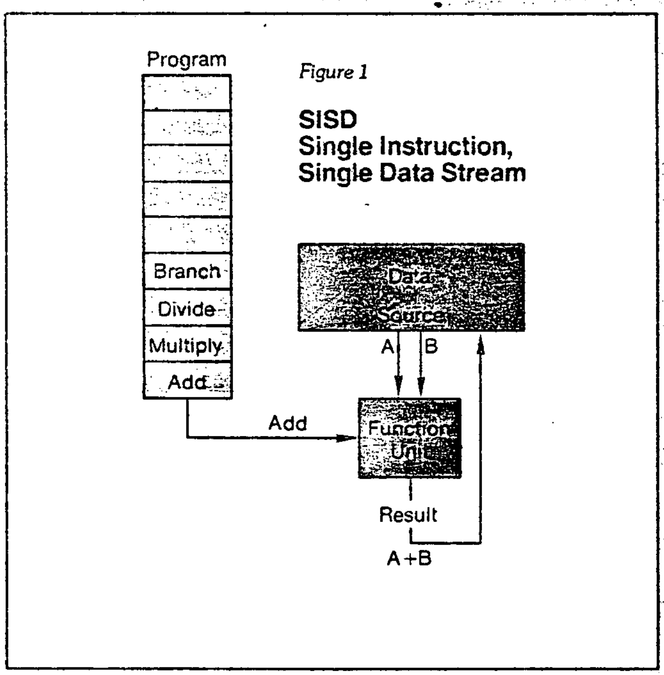


Figure 1

SISD Single Instruction, Single Data Stream

SISD designers attempted to remedy this by a technique called "look-ahead", in which instruction fetch for the next instruction was overlapped with some portion of the execution of the current instruction. This provided some performance improvement. However, digital computer programs, particularly those written in higher level languages, contain large numbers of test and branch instructions, in which the choice of the next instruction depends on the outcome of the current instruction. In such cases, "look-ahead" offers no speedup, and introduces substantial complexity to make sure that the partial execution of an incorrect next instruction does not contaminate the computation.

Another approach to increasing the speed of computation was to make multiple copies of portions of the SISD hardware. In this approach, called SIMD (for Single Instruction, Multiple Data Stream), the operand fetch, execution and result store portions of the hardware were replicated, so that the execution of a single instruction caused several values to be fetched, computed upon and the answers stored. For certain problems, this provided a substantial performance improvement. With sufficient hardware, entire vectors of numbers could be operated upon simultaneously. However, as with "look-ahead" SISD machines, the occurence of test and branch instructions, among others, required the machine to wait for the total completion of the instruction before proceeding. The test and branch itself could make no use of the replicated hardware.

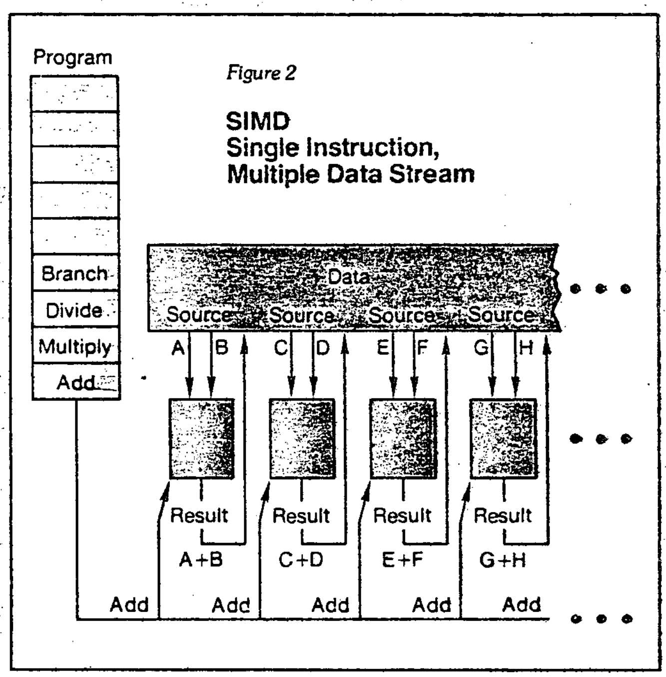


Figure 2

SIMD Single Instruction, Multiple Data Stream

In addition, two new problems were created by the SIMD architecture. Substantial portions of most programs are not vector-oriented. The computation of iteration variables and array subscripts is a scalar problem, for which SIMD offers no speedup, and the collection of operands across arrays is an addressing problem which many SIMD architectures do not handle. As a second problem, if an SIMD computer has a fixed quantity of replicated execution modules (adders, etc.), and if the length of the vector which the user wishes to operate on differs from the vector length of the machine, performance suffers and software complexity increases. The cost of computation remains high since the hardware is often not fully utilized.

Heterogeneous Element Processor

Denelcor, Inc.

Evolution of Computer Architecture

Continued difficulties with the implementation of high performance, cost effective computation using single instruction machines have led to the development of a new concept in computer architecture.

This concept, called MIMD (for Multiple Instruction, Multiple Data Stream) architecture, achieves high performance at low hardware cost by keeping all processor hardware utilized executing multiple parallel programs simultaneously. For example, while an add is in progress for one process, a multiply may be executing for another, a divide for a third; or similar functions may be executing simultaneously, such as multiple adds or divides. In MIMD architectures, cooperating programs are often called "processes". Independent programs may contain one or several processes.

processes. Since this arbitration of the state of memory locations is handled by hardware and without affecting the execution of unrelated instructions, the communication delay is short and the overhead is small.

MIMD computers may be used to execute either SISD or SIMD programs. SISD programs are just MIMD programs with no inter-program communication. Execution of multiple identical MIMD programs is equivalent to execution of an SIMD program.

In the SIMD case, MIMD computers may match the vector lengths exactly, while using remaining resources for unrelated computation. Thus, high efficiency may be maintained even through scalar portions of the code. But the major application of MIMD computers lies in problems of

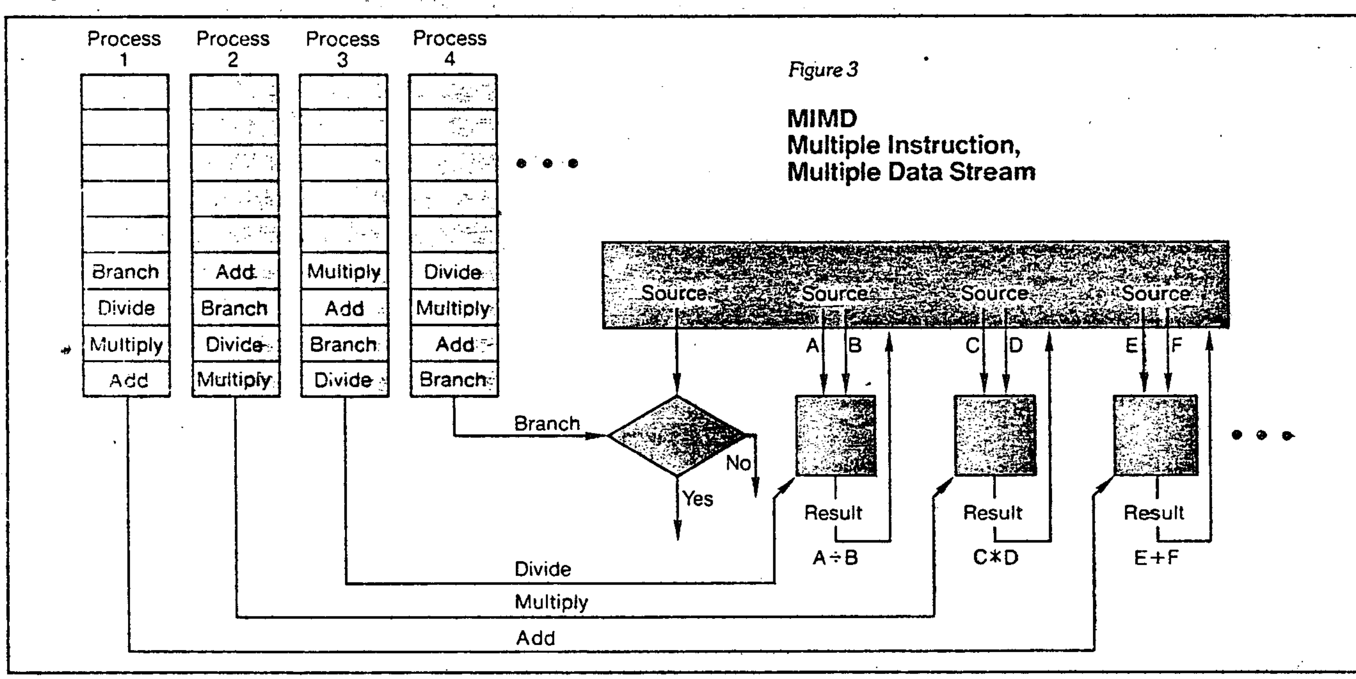


Figure 3

MIMD
Multiple Instruction, Multiple Data Stream

sufficient complexity that straightforward vector computation is not feasible. In these cases, which include continuous simulation and complicated partial differential equation solutions, MIMD architecture offers the only possible method of achieving significant parallelism. Denelcor's Heterogeneous Element Processor system is the only commercially available MIMD computer.

Because the multiple instructions executed concurrently by an MIMD machine are independent of each other, execution of one instruction does not influence the execution of other instructions and processing may be fully parallel at all times.

Successful MIMD architectures (figure 3) also provide low-overhead mechanisms for inter-process communication. In these architectures, data locations may contain not only a value but a state. Processes may synchronize by waiting for input locations to have the "full" state. Result storage may wait for output locations to attain the "empty" state resulting from consumption of their contents.

Heterogeneous Element Processor

Deneicor, Inc.

HEP Architecture

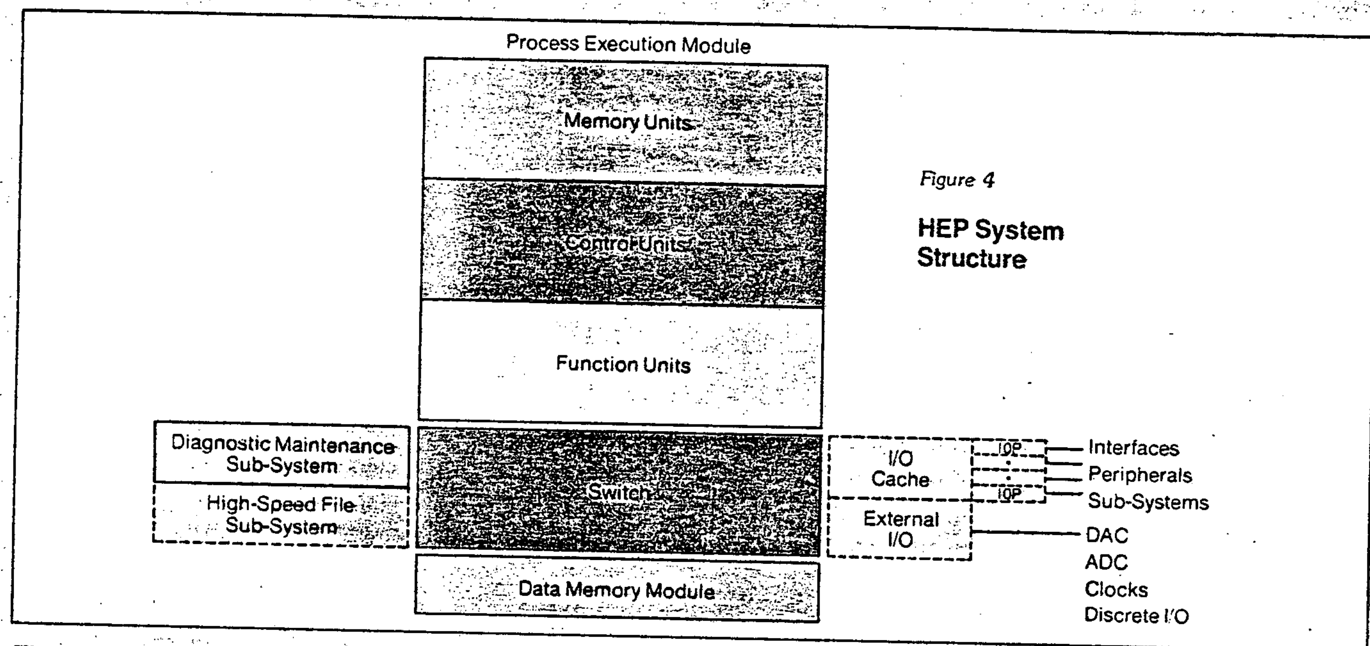


Figure 4

HEP System Structure

The HEP computer system consists of process execution modules (PEMs), data memory modules and support processors interconnected by a high-speed data switch network. All data memory modules are accessible by all PEMs. Thus, processes executing in parallel in one or several PEMs may cooperate by reading and writing shared information in the data memories. Parallel processes synchronize and pass information back and forth using the full/empty attribute of each data memory location. HEP instructions may automatically wait for an input data memory location to be full before execution, and leave the location empty after execution. Instructions may also wait for an output location to be empty before execution and leave it full after execution. This communications discipline allows processes to conveniently and unambiguously pass information to other processes while executing. The full/empty attribute ensures that reads and writes of inter-process variables will alternate and no information will be lost. For locations used exclusively within a process, the full/empty attribute is ignored and memory may be accessed conventionally.

Both normal and synchronized memory access are available to the Fortran programmer as well as the assembly programmer. Software modules in both Fortran and assembler programs may be distributed across several PEMs to achieve increased throughput. In general, design of a parallel program is not affected by whether the program will run in one or several PEMs.

In HEP, creation and termination of parallel processes in an MIMD program is a hardware capability directly available to the programmer. Processes are created or terminated in 100 nanoseconds by execution of a single HEP instruction. Thus, processes may be created at any point in a program where additional parallelism is required, and terminated as soon as their function is accomplished. Up to 64 user processes may exist simultaneously within each PEM in a HEP system.

In order to efficiently manipulate data, each PEM contains 2048 internal general purpose registers. PEMs automatically detect and flag normal arithmetic errors (overflow, underflow, etc.) and may generate traps on occurrence of these errors. Programs in a HEP system are protected from each other and relocated in memory by a set of relocation/protection registers in each PEM. This allows multiprogramming in a HEP system with full isolation of one user from the next.

All data and instruction words in a HEP are 64 bits long, although PEM data memory reference instructions allow partial word and byte addressing. The memory bandwidth in a HEP system is 20 million words/second per PEM, including the data switch network. Each PEM executes up to 10 million instructions per second. The architecture of the switch network allows up to 128 memory modules of up to one million words each and up to 16 PEM's. This range of system configurations results in speeds up to 160 million instructions per second on 64 bit data and memory

Heterogeneous Element Processor

Denelcor, Inc.

HEP Architecture & Software

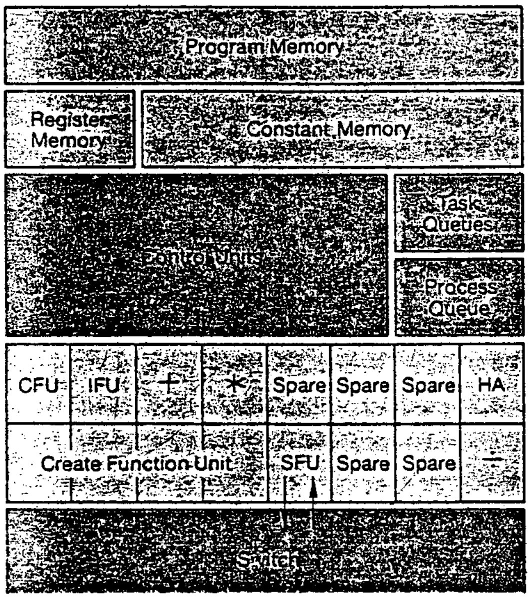


Figure 5

Process Execution Module Structure

HEP systems may include high-speed real-time I/O devices connected to the data switch network. These devices operate at memory speeds up to 80 million bytes/second. Normal I/O devices are connected to the HEP system through support processors. Thus, standard commercial I/O devices and controllers may be used for routine I/O functions. All standard I/O devices are accessible through the HEP operating system and Fortran I/O.

HEP Software

HEP systems support a batch operating system with Fortran and assembler programming languages. The HEP operating system provides input and output spooling, batch job scheduling, and full operator control of the system.

HEP Fortran is an extended ANSI Fortran IV with added parallel capabilities. The Fortran programmer has access to all standard Fortran formatted and unformatted I/O capabilities. In addition to the relaxation of syntax common to many Fortran compilers, HEP Fortran provides the programmer with the means for explicit parallel programming. A math library is also available which generates parallelism in the evaluation of known functions.

The HEP Assembly Language allows the user to access all of the capabilities of the system in an efficient manner. HEP Assembly Language subroutines may be included in a Fortran job to improve the efficiency of certain heavily used sections of code. Assembly Language programs have direct access to all hardware capabilities, including the direct creation and termination of arbitrary processes.

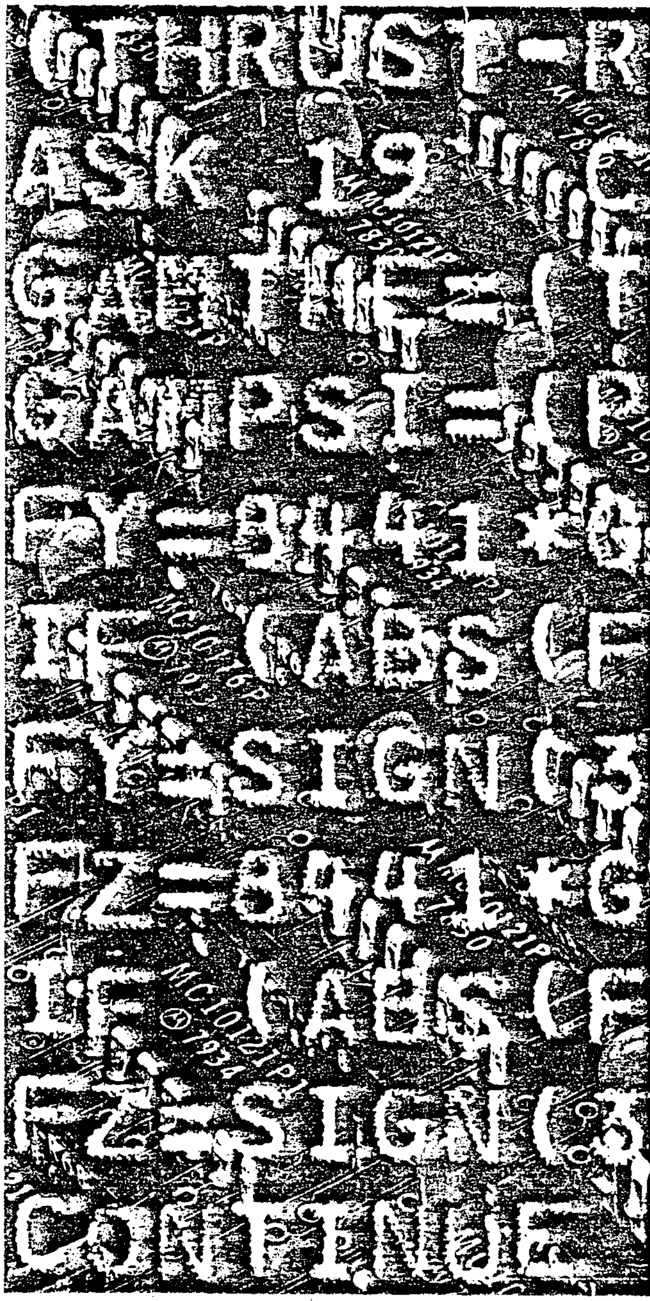
The HEP Link-Editor binds programs and subroutines into processes, tasks, and jobs. The input is from either HEP Fortran or HEP Assembler. The output is HEP machine executable code which is input to the HEP Loader at execution time. The HEP Link-Editor runs as a user job in the HEP PEM.

The HEP File System provides a large volume, high data rate I/O capability via the HEP Switch to a HEP System with multiple Process Execution Modules (PEM). Sequential access to information stored in multiple moving-head disk files is provided to the system at data rates from 80 megabytes per second (the maximum input rate for the switch), to approximately 1 megabyte per second (the rotating storage data rate). Random access to information is provided with comparable bandwidth, depending on the logical file size and the access patterns.

The HEP Interactive Maintenance Language (IML) provides a sophisticated yet easy-to-use language for debugging the HEP System. It is used in conjunction with maintenance hardware, wither test slots in the HEP main frame or off-line test fixtures. The language is procedure-oriented, thus permitting complex functions to be coded into higher order procedures.

Heterogeneous Element Processor
Denelcor, Inc.

HEP Applications



The most obvious area of HEP application is the multiprogramming of ordinary SISD algorithms. This application does not use the inter-process communications features of HEP, but fully utilizes its computing capacity. Since HEP's parallel architecture allows more complete utilization of its hardware, the cost effectiveness of HEP multiprogramming is higher than for other machines of comparable performance. Another benefit of HEP's effectiveness at conventional computation is that it can easily run all jobs at a facility — not just those which are sufficiently large or important to be written in parallel.

The application for which HEP was originally designed was the solution of systems of ordinary differential equations, such as those describing flight dynamics problems. In these problems, a substantial system of dissimilar equations must be solved, often in real-time. Many of the functional relationships in the equations are empirically derived and must be repetitively evaluated by multi-dimensional interpolation in lookup tables. Historically, such problems could only be solved, with limited precision and great expense, using analog computers. The HEP MIMD architecture is the first commercially available digital technology capable of effectively addressing these problems.

Another application area well suited to HEP is the solution of partial differential equations describing continuous media. These equations, which occur in fluid dynamics and heat transfer problems, are typically modelled using a grid of lattice points within the continuous medium. The behavior at a point is a function of the values at its neighbors. The HEP's architecture allows these problems to be solved with full parallelism, even in the presence of irregular or time-varying lattice geometry, or with complex functional relationships between lattice points.

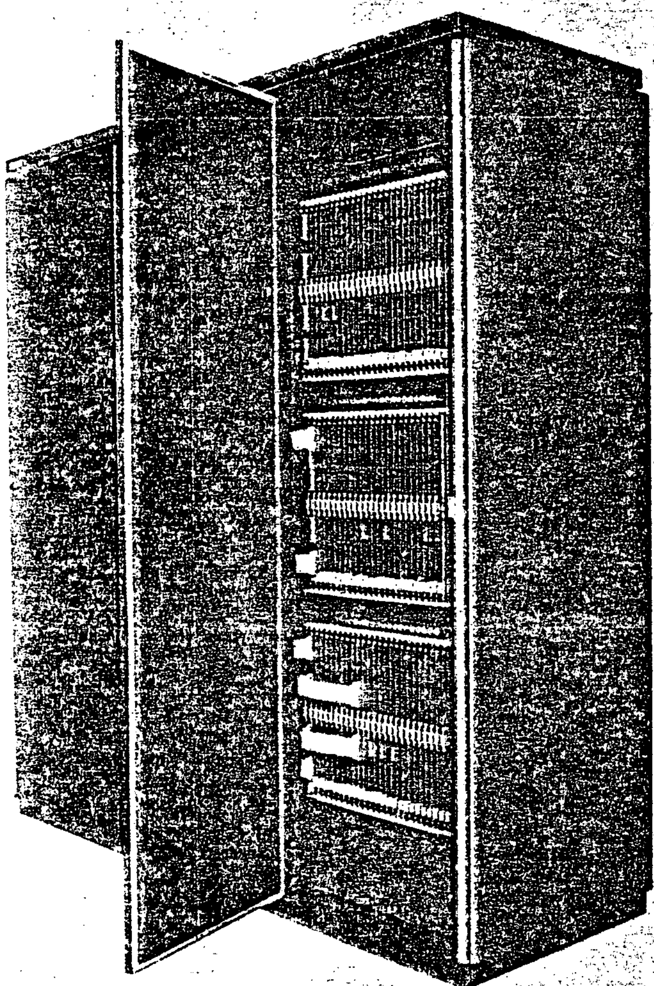
A fourth, and very general, application area for HEP is that class of problems for which a large number of discrete elements must be modeled or computed upon. Examples of such problems are tree and table searches, multiparticle physics problems, electric power distribution, and fractional distillation simulations. In all cases, complex behavior at a number of sites must be modeled, and interaction between the sites is critical to the result. Such problems are easily solved on the HEP.

The computing requirements for each of these applications are different. To effectively supply the range of capabilities needed, the HEP system is available in several configurations.

Heterogeneous Element Processor

Denelcor, Inc.

HEP Summary



HEP's building block architecture offers total flexibility enabling the user to start with the exact amount of computer power needed. As computing requirements grow, HEP's field-expandability allows the user to easily and economically add hardware and software modules to accommodate the largest of applications. These advanced features clearly place HEP in the forefront of digital computer technology and provide strong competition for existing computer systems, both scalar and vector.

The evolution of HEP is a natural result of Denelcor's on-going commitment to meet the market needs with state-of-the-art, high-quality systems.

- 10 Million to 160 Million Instructions per Second, Scalar or Vector.
- 2,048 to 32,768 General Purpose, 64-bit Registers
- 262,000 to one Billion Bytes Memory Capacity
- Parallel Computing in Fortran
- Fail-Soft Architecture
- Unbounded I/O Rates
- Real-Time Synchronized

Denelcor, Inc.
3115 East 40th Avenue
Denver, Colorado 80205
303-399-5700
TWX: 910-931-2201

Approved For Release 2003/08/13 : CIA-RDP84B00890R000400060009-1