ICOT Technical Report: TR-001

Toward a New Generation Computer Architecture

TR-001

Research and Development Plan for Computer Architecture in the Fifth Generation Computer Project

by

Shunichi Uchida

July, 1982

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

Abstract

This paper outlines the research and development plan for the Fifth Generation Computer Project from the view point of computer architecture. The architectural goal of this project is to develop the basic technology to build a highly parallel processor which supports a logic programming language. As intermediate goals, a parallel inference machine, a knowledge base machine and a sequential inference machine are considered.

In the paper, an approach to the goal is introduced as well as motivations of the planning and its technological background.

1. Introduction

Since the first generation computer systems appeared, about a quarter of a century has passed. Now computers are into their fourth generation, which is characterized by VLSIs and computer networks.

Present computer systems are used in all kind of applications and have become indispensable tools in our society. In recent years, computers have found applications in, among others, office automation (OA), computer-aided design or manufacturing (CAD/CAM), and home computing.

In these new applications, computers are increasingly used by non-computer specialists and thus new requirements are arising in addition to conventional requirements, such as faster processing speeds and more memory space.The onus is on contemporary computer science to produce new technology to fulfill these requirements.

The computer systems of today, however, are still based on the von Neumann model of computation, although all kinds of improvements have been made since the first generation. Essentially, all that has happened is that software systems have been extended to cope with new and more sophisticated applications. The software systems used in current computers have become too large and complicated to maintain good productivity, reliability and maintainability. And hardware systems based on conventional architectures are also approaching their limits in terms of improvements in computing power and memory space to support these complicated software systems.

Recent research in software engineering, artificial intelligence, computer architecture, and VLSI technology seems to indicate that now is the time to move on from the old von Neumann model of computers.

Research on artificial intelligence has come up with methods of expressing a variety of knowledge and how it can be used through research on natural language understanding systems, consultation systems (QA systems), expert systems, speech and picture recognition systems, intelligent robots, and so forth. These research results point to the possibility of intelligent computers which can be used as assistants to human workers.

Software engineering gives us methods of producing and maintaining large and complex software systems as well as specifications for the computer architecture which becomes the nucleus of the software systems. The concept of data abstraction and new programming languages based on well formed mathematical foundations have had a great influence on the other fields of computer science.[13]

Research on computer architecture has been stimulated by the research on software engineering and innovations in VLSI technology. Software engineering indicates the functions or features which the hardware systems should have to support new high-level languages. One example is the support of functionality, a very important feature for parallel architectures such as dataflow machines.

Generally, the hardware mechanisms to support these features require vast quantities of hardware components, as can be seen in dataflow or object-oriented architectures. VLSI technology can make these architectures practical. VLSIs will probably be the basis for implementing the new architectures which will support new programming languages with better performance and will make many artificial intelligence systems practical in real use.

In the light of the above developments, a national project, called the fifth generation computer project, was planned in Japan. In this plan, the fifth generation computer is characterized by functions oriented to knowledge information processing. Its machine language will be based on a logic programming language like PROLOG called FGKL (Fifth Generation Kernel Language). Its architecture will be based on a parallel computation model like the dataflow machine which will be implemented in VLSIs.

This project will span 10 years, divided into three stages: a 3-year initial stage, a 4-year intermediate stage and a 3-year final stage. The initial stage started in April 1982.

This paper outlines the plan and its research subjects from the viewpoint of computer architecture. Section 2 briefly explains motivations of the planning. Section 3 describes research goals and the research and development schedule. Section 4 gives details of several important research subjects closely related to computer architecture.

2. Motivations of Planning

Project planning began by discussing what kind of research should be done in information processing to prepare for the requirements anticipated to arise in the 1990s. Three committees were organized in JIPDEC (Japan Information Processing Development Center) at the suggestion of MITI (Ministry of International Trade and Industry) in 1979 to make a proposal for the Japanese national project.

Future computer systems were investigated and discussed from two different viewpoints: one was social needs, the other, the technology to be available in the 1990s. The main topics discussed are summarized as follows.

2.1 Social Needs

In the society of the future, computers will be more and more important for our daily life. Computers will find applications in fields where they are not used today. In Japan, computers main use has been in secondary industries to raise manufacturing productivity. This made it possible to produce high-quality, low-priced industrial goods, such as cars, TV sets, and more computers.[31]

However, in other fields, such as offices, engineering design, agriculture and fisheries, medical care, education, public service, and government, productivity is still very low, resulting in very high costs. Japanese society is aging at an unprecedented rate. People 65 years of age or older will make up 12% of the entire population in the 1990s. Thus, computers will have to be introduced in primary and tertiary industries to conserve resources and energy. These demands, and the new fields where computers are expected to play a very important role are summarized in Table 1.[33]

In the light of these demands, knowledge information processing came up as an important target. One of the most difficult problems that present computers suffer from is unfriendly man-machine interfaces which prevent non-specialist users from using computers easily. This problem has been studied in such research fields as pattern recognition and artificial intelligence.

In Japan, research in these fields was promoted by a previous national project, called the research and development of the Pattern Information Processing System (PIPS). This project suggested the importance of knowledge information processing functions which support systems for speech, picture and image recognition more efficiently.

Finally, we concluded that the realization of intelligent computers is not a dream but a practical research target which is worth conducting as a national project. And we named the computer system to be developed as the goal of the project the fifth generation computer system.

2.2 New Computer Technology

The technological background for future computer systems, in this case, the fifth generation computer system (FGCS), will come from such research fields as artificial intelligence, software engineering, . computer architecture, computer networking, and VLSI technology.

Recent research topics which we considered very important in constructing an image of the FGCS are as follows.

1) Artificial Intelligence

- Methodology to express knowledge, and to infer with knowledge as seen in expert systems, and consultation systems, natural language understanding systems, intelligent robots.

- Research on high-level logic to formalize mechanisms of inference and organization of knowledge, as shown in theorem proving, and research on modal logic.

- New languages and computational models to implement experimental systems. Examples are LISP, PLANNER, PROLOG, NETL, Actor theory, and Frame theory.

- Many experimental systems for natural language understanding, consultation, speech and picture recognition, and so forth.

2) Software Engineering

- New programming languages and computational models, such as functional programming, logic programming, concurrent programming, and so on.

- Proposals for new machine models, such as reduction machines and functional machines.

- New concepts of programming, such as abstract data type, modularization, and hierarchical organization.

- Program verification and synthesis, including research on the semantics of programs, specification, description and so on.

- Research on programming environment, including personal computing and networking.

- Research on data base systems, including distributed data base systems.

3) Computer Architecture

- Software oriented computer architectures to support high-level languages such as LISP, PROLOG, Smalltalk, and Ada. Examples of these architectures are a dataflow architecture and an object-oriented architecture.

- Parallel architectures to build high-speed computers for symbol manipulations and numerical calculations, such as a dataflow architecture, and architectures of various array processors and special purpose machines.

- Data base machine architectures, such as a relational data base machine architecture, including hardware components to implement relational algebraic operations.

- Distributed architectures to support computer networking, including personal computers, local networks, and various hardware components which support the implementation of good programming environments like a bit-map display, picture input/output devices, and so on.

4) VLSI Technology

- VLSI architectures to make full use of the potential of VLSI technology, the key technology in realizing new computer architectures.

- VLSI CAD systems including methods of describing computer architectures and logic circuits as well as methods of organizing architecture design databases.

- New circuit design methods, such as gate arrays and PLAs, as well as full custom VLSI circuits to realize short turnaround time and high reliability.

- New devices, such as GaAs and Josephson junction devices, including magnetic memory devices.

For each of the above research fields, we went over present research status and forecast innovations in the next 10 years to make images of the computer systems which would appear in 1990s. Differences in viewpoint resulted in several images of the computer emerging. The image deriving from the viewpoint of computer architecture was different from that deriving from the viewpoint of software technology and artificial intelligence.

After long discussions, we arrived at a composite images and made a final plan in summer 1981. This plan was announced at the international conference on fifth generation computers held October 1981 in Tokyo. In this plan, new computer architectures, such as a dataflow architecture and an object-oriented architecture play a very important role in harmonizing the requirements from software technology with those from distributed architecture and VLSI technology.

3. Outline of the FGCS Project

3.1 Functional View of the Research Goal

The goal of the FGCS project is to develop the basic technology which will be necessary to build the computer systems of the 1990s and to develop a prototype fifth generation computer system.

The FGCS is oriented to a knowledge information processing system and will be capable of accommodating intelligent conversation and inference functions, employing knowledge bases. The functions of FGCS may be roughly classified as follows:

1) Problem solving and inference
2) Knowledge base management
3) Intelligent interfaces

These functions will be realized by making individual software and hardware systems. A conceptual image from the programming viewpoint is shown in Fig. 1. In this figure, the modeling (software) system is the project's ultimate goal for software research. The machine (hardware system) is the ultimate goal for hardware research.

The upper half of the modeling system circle corresponds to the problem solving and inference function, the lower half to the knowledge base management function. The portion that overlaps the human system circle on the left corresponds to the intelligent interface function.

The interface between the software and hardware systems will be the kernel language, and the hardware system will directly execute the kernel language.

The figure also illustrates that the intelligent interface function relies heavily on the two former groups of functions, and that the software and hardware systems are biased decisively towards the human system by significant enhancement of logic level of the hardware system.

The hardware system will have the hardware mechanisms to efficiently support the high-level kernel language and knowledge representation language and provide users with enough processing power to make the application systems practical.

The goal of the software research on the problem solving and inference function is a co-operative problem solving system. In such a system, a single problem will be solved by two or more problem solving systems co-operating with each other. An example is a medical diagnosing system where a diagnosis is made in the same way as a physician and a surgeon might co-operate to make a diagnosis.

For this purpose, it will be necessary to develop a meta inference system that performs inferences on the inference processes and also on the knowledge posessed by the individual problem solving systems. The meta inference system will have to have not only deductive inference functions but also such higher level functions as common sense reasoning and inductive inference. These functions will be realized as software systems in the kernel language. Corresponding to the basic deductive inference processes, the kernel language will be syllogistic, and classified in a logic programming language of which a current example is PROLOG.

The goal of software research on the knowledge base management function is to establish knowledge information processing technology. This includes research on a knowledge representation system, a knowledge base design and maintenance support system, a knowledge acquisition system, and a distributed knowledge base management system.[24]

These systems will finally be integrated into a co-operative problem solving system. One particularly important research item will be a semi-automated knowledge acquisition system equipped with a certain level of learning functions.

For the basis of implementing the knowledge base management function, the interface to the relational database function which includes a consistency testing function will have to be realized in the kernel language.

The goal of research on the intelligent interface function is to develop basic software and hardware systems which enable the user to communicate with the computer in natural languages, speech, graphics and picture.[25] The natural language understanding system will cover a basic vocabulary of up to 10,000 words and up to 2,000 grammatical rules and attain a 99 % accuracy in syntactic analysis.

For speech processing, speech input and output systems will be studied. The goal for speech input is to accept continuous speech in Japanese standard pronounciation by multiple speakers, and to have a vocabulary of 50,000 words with 95% recognition rate for individual words. Processing speed will be within 3 times real-time speech.

To attain these goals, research and development will be conducted for hardware systems, such as dedicated hardware processors and high performance interfaces for efficiently processing signal level speech or picture.

The goal of hardware research on the problem solving and inference functions is to develop hardware inference mechanisms, that is, an inference machine.[32] An important function of the inference machine is supporting the kernel language, attaining enough processing speed to make the application systems practical.

The maximum speed is expected to be 100M to 1G LIPS. LIPS stands for Logical Inference Per Second and 1 LIPS denotes one syllogistic inference operation per second (one invocation of a clause). 1 LIPS corresponds to 100 to 1,000 IPS (Instructions Per Second) for current computers.

To realize such performance, the key research and development item will be a high-level parallel architecture which efficiently supports symbol manipulations and uses high-speed devices. The research will concentrate on a new architecture, based on dataflow and object oriented architectures, suited to a parallel inference mechanism.

The goal of hardware research on the knowledge base management function is to develop hardware mechanisms, that is, a knowledge base machine. An important function of the knowledge base machine will be the capability to store a very large amount of data which may consist of instanciated facts and rules for inference and the capability to efficiently search, sort and update the data.[27]

The requirements for the hardware system to support the high-level knowledge base function are not so clear-cut; however, a relational data base system will probably be the basis of the machine. The research will concentrate on a relational data base machine architecture including parallel and pipeline processing mechanisms, and a memory hierarchy system with a virtual memory mechanism.

Memory capacity will probably be 100 to 1,000 GB and the average time to retrieve a piece of knowledge data for answering a question, a few seconds.

In the final stage of the project, the software systems will be integrated in a prototype software system known as the basic software system. The hardware systems will also be integrated as a prototype FGCS system which will probably include about 1,000 processing elements and be implemented in VLSIs.

On these software and hardware systems, basic application systems will be developed. The goals of the basic application systems are listed in Table 2.

The conceptual organization of the prototype FGCS is illustrated in Fig.-2 and Fig.-3. These figures also show the layers of the software and hardware systems and the position of the languages which interface the hardware and software systems.

For the actual use of the FGCS, however, several models will be developed to meet requirements for various performance levels for each of the three functions. For example, a knowledge base enhanced model will be used to manage large-scale knowledge data. Small models will be used as future personal computers.

These models of the FGCS will have the kernel language as their common machine language and will be used as nodes of network systems. Thus, the software and hardware systems will be organized to form distributed systems.

In addition to the research subjects described above, there is one more important class of research subject: development support systems. This includes the development of a sequential inference machine and research on VLSI technology and systems architecture.[26][28]

The sequential inference machine (SIM) will be developed to support and accelerate the development of the software system by providing researchers with a good programming environment. The SIM will be a firmware-based personal computer system supporting an extended PROLOG language (called FGKL version 0.) with a local network system to interconnect many SIMs.

The plan is to complete the design and implementation of the SIM in the initial stage so that SIMs can be used from the intermediate stage.

The research on VLSI technology will mainly focus on an intelligent VLSI CAD system, including methods of describing computer architectures, hardware modules, and logic circuits as well as techniques of organizing an architecture design data base. Another important role of this subject will play is to support the implementation of various experimental systems, such as the SIM, an experimental dataflow machine, and an experimental relational data base machine.

The research on systems architecture is intended to provide the researchers with software and hardware development support systems which will consist of large scale computers, SIMs, several experimental systems, special purpose processors, and so forth. An important research item here is the development of a high-speed local network system which will enable us to efficiently carry out experiments which will combine experimental software systems and hardware systems in various ways. This research network system will probably develop into a knowledge information processing network system on which software systems, such as a co-operative problem solving system, will be implemented.

3.2 Schedule of Research and Development

The research subjects in this project involve many unknowns and thus the research and development will have to progress by trial and error. Plans must change, depending on the level of achievement in each research subject.

This project will span 10 years, divided into an initial stage (3 years), intermediate stage (4 years), and final stage (3 years). At the end of the initial and intermediate stages, research achievements will be evaluated and details of the research plan for the next stage determined. The points of emphasis for each stage are as follows.

In the initial stage, the purpose of research is to accumulate the achievements of the past and evaluate them to construct new frameworks toward knowledge information processing. Thus, many research items will be studied independently and candidates for each research subject will be screened to determine the main research subjects for the intermediate stage.

In the intermediate stage, the research items of the initial stage will converge on two research subjects: an inference machine subsystem and a knowledge base machine subsystem.

The inference machine subsystem will support the version 1 kernel language which will include functions for describing parallel processing. Thus, its architecture will be based on a parallel architecture, such as a dataflow architecture. This machine is called the parallel inference machine (PIM). The PIM will contain about 100 processing elements.[29]

The knowledge base machine subsystem will support the knowledge representation language to be developed in the intermediate stage. Its architecture will include mechanisms to support relational algebraic operations and some new operations for handling instanciated facts and rules (axioms).

In this stage, part of the VLSI CAD system will be used to support the implementation of the hardware modules for these experimental machines.

In the final stage, a total system will be developed by restructuring and combining the research achievements of two subsystems developed in the intermediate stage. Software systems, such as the basic application systems, will also be run on this system to evaluate its performance.

The research subjects of the initial stage are listed in Table-3.

4. Research Plan for Computer Architecture

This section outlines the research and development plan for computer architecture and some of the research subjects which will be studied in the initial stage.

4.1 Overview of Architecture Research

The most important requirements for research on the fifth generation computer architecture are listed below.

1) Support of the fifth generation kernel language (FGKL)
2) Support of modular programming
3) High processing power for symbol manipulations
4) Support of a large knowledge base
5) Support of distributed processing and personal computing
6) Implementation in VLSIs

Recent research achievements in computer architectures show that these requirements are mutually interrelated. Many researchers are trying to develop a new basis constructed by unifying various theories and experiences.

In the same way, this project must aim at establishing a new framework as the basis for the new generation of computers. Fig.-4 illustrates one approach to achieving the goal from these requirements.

Each requirement will be discussed from such view-point as programming language, parallel processing, distributed processing, and VLSI implementation.

4.2 Programming Languages

The basis for the programming language for the FGCS was selected by considering the following points.

1) Description of the processing of inference and representation of knowledge
2) Support of modular programming
3) Possibility of program verification
4) Eligibility for data base handling
5) Description of parallel processing
6) Description of distributed processing

We surveyed and evaluated these points to compare candidates for the kernel language for the FGCS.

Point 1 is related to such language features as procedure invocation (subroutine call) by pattern matching, non-deterministic processing (like search of a game tree), and declarative description (not procedural).

These features correspond to the basic deductive inference mechanisms often observed in the mechanisms of theorem provers. These features first turned up in a programming language called PLANNER [7] which introduced back-tracking as a mechanism for realizing non-deterministic processing. PLANNER, however, was implemented in LISP and its semantics (computational model) were not always clear.

These features next occurred in PROLOG. The first PROLOG interpreter was made in France. It was not particularly efficient because it was implemented in FORTRAN. One of the unique features of PROLOG is that its semantics are very clear because it is based on a procedure of theorem proving for first order logic (Horn clause). When PROLOG was implemented in assembler on DEC-10, its superiority over other languages became more obvious. Now, PROLOG is considered to be the most appropriate language for implementing high-level inference systems, such as common sense reasoning and inductive inference systems.[1]-[4]

Points 2 and 6 are closely related when observed from an architectural viewpoint. Concerning point 2, the concept of data abstraction should be a very important basis for programming methodology, especially modular programming.

One way of supporting data abstraction is to introduce an object-oriented computation model, the basis for such languages as SIMULA, CLU, and Smalltalk.[10][11][16] This model also applies for the description of distributed processing. As the purpose of modular programming is to make isolated program modules and to express computation by message-passing among these modules, it is naturally related to distributed processing and distributed architectures.[38] The most generalized model is probably the Actor model, which includes a dataflow model as its subset.[8]

Hardware architectures and mechanisms, such as object-oriented architecture, capability mechanisms, and dynamic type checking mechanisms should be researched in parallel with these language features.

As the point 3, a good mathematical basis for programming languages is a must for understanding the meaning of programs. For program verification and synthesis, functional languages and logic programming languages are appropriate because they have clear and formal models.[13]

As point 4, this project concentrates on the interface between the kernel language and the relational database. The machine language level is usually set to relational algebraic operations, such as selection,restriction and join. Functional languages and logic programming languages are will extend naturally to interface with relational algebra.

Point 5 is important not only for describing many problems naturally but also for building high-performance machines to employ parallel architectures. An important example is the relation of functional languages to dataflow architecture. The functionality of programs must be assured by programming languages which make full use of parallel architecture, especially for symbol manipulations. For this purpose, functional languages and logic programming languages are again the most promising.

After long discussions on the above points, we determined that the kernel language (FGKL) would be a logic programming language.We selected PROLOG as first stepping stone toward the FGKL. As the present PROLOG is implemented on conventional sequential machines and support of modular programming is not always sufficient, several extentions will have to be made to fulfill this project's requirements.

Three versions (0 to 2) will be designed at the beginning of the initial, intermediate and final stages, respectively. The extensions will aim to include language features such as the following.[24]

- For version 0,
1) Mechanism for structuring programs
2) Interface to the relational database

- For version 1,
3) Introduction of functions for parallel execution
4) Mechanism for supporting abstract data type handling
5) Meta level control functions to control parallel execution

- For version 2,
6) Extension of meta level control functions such as intelligent backtracking, exception handling, control of focus, and multi-world model.

These versions of the FGKL correspond to the hardware systems to be developed in each stage. Version 0 corresponds to the sequential inference machine (SIM) which will be the firmware-based personal computer. Version 1 corresponds to the parallel inference machine (PIM) which will be based on a dataflow architecture. Version 3 corresponds to the final prototype system which will be developed at the end of this project.

4.3 Parallel Processing and Distributed Processing

The important research goals for computer architecture through the initial and intermediate stages are summarized as follows;

1) Research on parallel computational models and parallel machine organizations for a logic programming language (kernel language version 1, or parallel PROLOG)

2) Research on the mechanisms to support modular programming by introducing an object-oriented architecture

3) Design and implementation of an experimental model dataflow machine to collect various data for designing a highly parallel processor for symbol manipulation and for implementing hardware modules using semi-custom or custom LSIs and VLSIs

4) Development of a small-scale relational data base machine which can be used for research tools connected to the sequential inference machine (SIM)

5) Development of SIMs (extended PROLOG machines) as research tools.

Development of a parallel inference machine will require both top-down and bottom-up approaches. The top-down approach will start from research on parallel execution models of PROLOG and reach research for the dataflow machine.

A number of researchers have been working on parallel execution models of PROLOG. Execution of a PROLOG program can be thought of as an and-or-tree search. And-nodes correspond to the terms in each clause, or-nodes to a group of clauses each of which has the same clause head. To execute PROLOG programs in parallel, two styles of processing, multiprocessing[34] and pipelining, may be applied. Pipelining is often called stream processing.[6] Multiprocessing can be applied to the or-nodes, pipelining to the and-nodes. In these processing mechanisms, the techniques studied in dataflow machine research will probably be commonly applicable. Examples are such mechanisms as lazy evaluation, procedure invocation, associative search, and structure memory with garbage collector. This kind of research has just started and there remain a lot of interesting problems to be solved.[17]-[23]

The bottom-up approach will start from research on machine components, such as processing elements or structure memories for a dataflow machine. The research on object-oriented architecture will also proceed along more or less same lines. The performance or capacity of each machine component will be very important in determining the overall machine structure. The techniques of implementing components in LSIs or VLSIs will also be studied through experimental development. In addition, hardware and software simulators will be made to evaluate overall machine structures.

This project plans to develop an experimental dataflow machine containing about 10 processing elements. An image of this machine is illustrated in Fig.-5.[29] Some basic components, such as processing elements with associative memory mechanisms and a connection network will also be experimentally built.

Research on an object-oriented architecture will be done in parallel with research on the kernel language. To describe concurrent processes, the concept of an object is very convenient. This concept has been implemented in several languages, like SIMULA, Smalltalk, and ADA.[16] For PROLOG, the introduction of this concept has been studied, for example, in T-PROLOG, which is developed in Hungary.[5] The hardware mechanisms for supporting the objects or abstract data types have been studied in such machines as ALTO, and System 38.[10][14]

In this project, this research plans to develop memory mechanisms which support object creation and deletion, dynamic type checking, and associative search to map logical names to physical addresses. These mechanisms may initially be used in the SIM. After that, they will be included in the PIM.

Research on the knowledge base machine will start with research on a relational database machine. This research will especially focus on development of the memory hierarchy system. This system is very important in making the relational database machine practical. This system will include large capacity silicon disks (100 to 400 MB) as well as a high speed interface to large capacity (about 10 GB) magnetic disks. This machine will also include 4 to 8 processing elements to perform some relational algebraic operations such as join and projection. An image of this machine is shown in Fig.-6.[27]

The intention in developing the relational database machine in the initial stage is not just as an experiment but also as a research tool which will be necessary from the intermediate stage. This small-scale database machine will become the basis for the large parallel database machine to be developed in the intermediate stage, because of the memory hierarchy system.

In this large machine, new hardware mechanisms, such as a highly parallel sort and search processor based on a stream, based dataflow control will be introduced.

The development of the sequential inference machine (SIM) is one of the most important goals of the initial stage. The most important requirement of this machine is to provide researchers with a good programming environment including a fast interpreter for FGKL version 0 (extended PROLOG) and a large memory space. The SIM will employ a fast microprogrammable processor with several dedicated hardware modules to enhance stack-operations. Execution speed should be about 10 times faster than a present large-scale computer (about 500 KLIPS). A large memory space is very important for debugging PROLOG programs and thus, a memory space of more than 16 MW is target. The organization of this machine is shown in Fig.-7.[32]

This machine will include several hardware stacks and some hardware modules for hashing to improve its execution speed. For smooth man-machine communication, peripherals will include a bit-map display, Japanese character input/output devices, picture and speech input/output devices and local network interfaces. In addition to these devices, an interface mechanism to the relational database machine will be developed so that it can be used as an external database for the extended PROLOG interpreter. How to include the relational database in the PROLOG language processor poses one of the most interesting research questions.

As the SIM is to be used as a personal computer, research on distributed processing will also be done to develop a research computer network system which will enable researchers to carry out experiments efficiently.

4.4 VLSI CAD System

In this project, research on VLSI CAD system has two major purposes. One is to develop an intelligent VLSI CAD system. This research is an FGCS application. It will cover high-level hardware description languages and methods to convert hardware descriptions to low-level circuit descriptions using various hardware design skills. An image of this CAD system is illustrated in Fig.-8.[28]

The other purpose is to provide researchers with design tools to develop the various hardware modules and semi-custom or custom LSIs and VLSIs required to build many experimental machines. For the design tools, a hardware description language to specify module functions will be very important for good documentation, because several manufacturers will co-operate in making many modules, thus necessitating a standard language.. Most manufacturers in Japan have their own CAD systems which cover gate-level circuit design, pattern layout, and chip making.

Under the above conditions, research will focus on a high-level CAD system which includes items like the following.

1) Hardware descriptions such as specification description and functional description

2) Methods of translating high-level descriptions (functional description and structural description) into concrete circuit descriptions

3) Methods of simulating and verifying the described hardware circuits

4) Accumulation of design data to build an architecture design database

5) Development of a personal logic design station which will be made on a sequential inference machine with a small-scale database machine

Research on high-level hardware description and verification shares a common background with research on an intelligent programming system which will include research on specification writing, program verification and algorithm banks. Thus, the achievements in software research will be introduced into this area.[36] For example, a logic programming language can function as the basis for hardware description languages more appropriate for verification and translation.[37]

5. Conclusion

This paper briefly outlined the FGCS project and its architecture research plans. Research on computer architecture should be done in parallel with research on programming languages. In this project, the language features will determine the external specifications of the hardware systems. For example, FGKL version 0 will determine the primitive machine structure of the SIM, and FGKL version 1, the structure of the PIM. Programming languages impose top-down requirements, VLSI imposes bottom-up requirements.

Bottom-up requirements include a regular structure and adequate modularity in hardware design and a powerful CAD system to cope with the complexity of the circuits to be designed. Research on these items will not get far without appropriate targets. In this project, this research will be done by producing hardware modules for experimental machines, such as a dataflow machine and a relational database machine.

The new generation of computers which will appear in the 1990s will have to satisfy both top-down and bottom-up requirements. The final prototype system developed at the end of this project will be one of the first examples of the new generation of computers.

Acknowledgments

This plan was arrived at through investigating various fields of computer science. The author would like to express his gratitude to all the researchers who contributed to the investigation. The author also gratefully acknowledges the valuable suggestions of Mr.K.Fuchi, director of the research laboratory of ICOT, and Dr.K.Furukawa and Mr.T.Yokoi of ICOT and ETL.

References

1. David H.D.Warren: IMPLEMENTING PROLOG -compiling predicate logic programs, vol.1, D.A.I. Research Report No.39, Univ. of Edinburgh, May 1977.

2. David H.D.Warren: IMPLEMENTING PROLOG -compiling predicate logic programs, vol.2, D.A.I. Research Report No.40, Univ. of Edinburgh, May 1977.

3. David H.D.Warren: LOGIC PROGRAMMING and COMPILER WRITING, D.A.I. Research Report No.44, Univ. of Edinburgh, Sep. 1977.

4. L.M.Pereira, F.C.N.Pereira and D.H.D.Warren: User's Guide to DEC system-10 PROLOG, Sept. 1978.

5. I.Futo and J.Szeredi: T-PROLOG: A very high level simulation system — General Information Manual, SZ. K. I. 1011 Budapest I. Iskola Utca 8, 1981.

6. K.L.Clark and S.Gregory: A relational language for parallel programming, Res. Rep. DOC 81/16, Dept. of Computing, Imperial College of Science and Technology, July 1981.

7. C.Hewitt: PLANNER: A language for manipulating models and proving theorems in a robot, MIT/AI Memo No.168, Aug. 1970.

8. H.G. Baker, Jr: Actor systems for real-time computation, MIT/LCS/TR-197, Mar. 1978.

9. G.J.Sussman,et al: Computer-aided evolutionary design for digital integrated systems, MIT/AI Memo No.256, May 1979.

10. C.P.Thacker,et al: Alto:A personal computer, CSL-79-11, PERC Xerox, Aug.1979.

11. A.Goldberg and A.Kay: Teaching SMALLTALK, SSL-77-2, PERC Xerox, Jun.1977.

12. D.W.Clark:The Dorado: A High-Performance Personal Computer, CSL-81-1, PERC Xerox, Jan.1981.

13. P.Wegner(Ed): Research directions in software technology, MIT Press, 1979.

14. G.J.Myers: Advances in computer architecture, John Wiley and Sons, 1978.

15. A.Snyder: A machine architecture to support an object-oriented language, MIT/LCS/TR-209, MIT, 1979.

16. J.Rattner,et al: Ada determines architecture of 32-bit microprocessor, Electronics, p.119-126, Feb. 24, 1981.

17. J.B.Dennis: The varieties of dataflow computers, Proc. 1st Int'l Conf. on Distributed Computer Systems, p.430-439, 1979.

18. W.Ackerman: Dataflow languages, Proc. NCC, p.1087-1095, 1978.

19. W.Ackerman: A structure memory for dataflow computer, MIT/LCS/TR-186, MIT, 1977.

20. Arvind,et al: An asynchronous programming language and computing machine, Tech.Rep.No.114A, UCI, 1979.

21. P.C.Treleaven: Exploiting program concurrency in computing systems, COMPUTER, p.42-50, Jan. 1979.

22. A.Plas,et al: LAU system architecture: A parallel data-driven processor based on single assignment, Proc. Int'l Conf. on parallel processing, p.293-302, 1976.

23. A.L.Davis: A data flow evaluation system based on the concept of recursive locality, Proc. NCC, p.1079-1086, 1979.

24. K.Furukawa,et al: Problem solving and inference mechanisms, Proc. of Int'l Conf. on the Fifth Generation Computer Systems, Oct. 1981.

25. H.Tanaka,et al: Intelligent man-machine interface mechanisms, ibid.

26. T.Yokoi,et al: Logic programming and dedicated super personal computer, ibid.

27. M.Amamiya,et ai: New architectures for knowledge base mechanisms, ibid.

28. K.Sakamura,et al: VLSI and system architecture, ibid.

29. H.Tanaka,et al: Preliminary research on data flow machine and data base machine as the basic architecture of fifth generation computer, ibid.

30. K.Fuchi: Aiming for knowledge information processing system, ibid.

31. H.Karatsu: What is required of the 5th generation computer -Social needs and its impact, ibid.

32. S.Uchida, et al: New architectures for inference mechanisms, ibid.

33. Fifth generation computers, JIPDEC Report, Summer 1980, JIPDEC, 3-5-8 Shiba-koen, Minato-ku, 105, Tokyo.

34. H.Aida:On the parallel prolog system PARALOG (in Japanese), Proc. of PROLOG Conf. in Tsukuba, p.19-27, Mar. 1982.

35. M.Amamiya and R.Hasegawa: Dataflow machine and functional programming (in Japanese), AL81-84,PRL81-63, IECE of Japan, Dec. 1981.

36. W.E.Cory and W.M.vanCleenput:Development in verification of design correctness –Tutorial–, Proc. of 17th DA Conf. P.156-163,Jun. 1980.

37. S.Uchida and T.Higuchi:Architecture description and logic simulation using PROLOG (in Japanese), CA-44-5, IPS of Japan, Feb. 1982.

38. M.Tokoro and T.Takizuka;On the semantic structure of information –A proposal of the abstract storage architecture–, Proc. of 9th Annual Sympo. on Computer Architecture, Apr. 1982.

39. P.C.Treleaven and I.G.Lima:Japan's fifth generation computer systems project, Res.Rep. of Univ. of Newcastle upon Tyne, May. 1982.

Table 1 Social Needs and Computer Applications

Social background	Need	Specific need	Computer application
Low productivity areas Public services Clerical work Government Distribution	Increase in productivity of indirectly related fields	Increase in productivity of planning surveys	DSS
		Increase in productivity of clerical processing	OA
	Increase in productivity of service sector	Immediate "window" services	OA
		Immediate general services	VLDB
International competition Slowdown in international competition Energy and material conservation Decline in ability to export (Friction in international trade)	Knowledge intensification	Sophistication of machines and devices	ROBOT
		Diversification	CAD
	Improvement of technology, improvement capability	Supply of technical information	IR
		Mechanization of design work	CAD
		Rationalization of experimental work	Lab Auto
	Energy and material conservation	Energy conservation control	ROBOT
		Changes in industrial structure	Promotion of information industry
Changes in population structure Aging of population (Fewer young workers) Friction from rising level of education Decline in desire to work Increase in educational and training costs	Aging of population	Increased medical expense	ME
		Increased welfare burden	CAI IR
		Health maintenance	HOME COMPUTER
	Emergence of new generation	Rising interest in fashions	PERSONAL COMPUTER
		Aversion to unpleasant jobs	ROBOT CAD IR
Informationalization of society Medicine Education Countermeasures to increased population density Security costs Generation gap Regional gaps International gaps	Improvement of quality and economy of social system services	Systematization of medicine	ME
		Individualization in education	CAI
		Systematization of calamity prevention	ROBOT
		Systematization of crime prevention	IR
	Response to increasing interest in fashions	Individualization of applications	PERSONAL COMPUTER
	Communication gap	Individualization in education	COMPUTER NETWORK
		Generation gap	CAI
		International gap	LT
Informationalization of life Communication gap centering on the individual Energy and material conservation Increase in educational and training costs for the individual	Improvement of convenience	Composite mode reservation	COMPUTER NETWORK
		Modernization of mail and electric communications	TELEMAIL, TELETEXT
	Informationalization of the home	Systematization of home education	CAI
		Hobbies	HOBBY COMPUTER
	Automation of the home	Sophistication of household appliances	HOME COMPUTER
		Automation of equipment	HOME COMPUTER

Key

DSS	DECISION SUPPORT SYSTEM	IR	INFORMATION RETRIEVAL
OA	OFFICE AUTOMATION	Lab Auto	LABORATORY AUTOMATION
VLDB	VERY LARGE DATA BASES	ME	MEDICAL ELECTRONICS
ROBOT	(incl. CAM:COMPUTER AUTOMATED MANUFACTURING)	CAI	COMPUTER ASSISTED INSTRUCTION
CAD	COMPUTER ASSISTED DESIGN	LT	LANGUAGE TRANSLATION

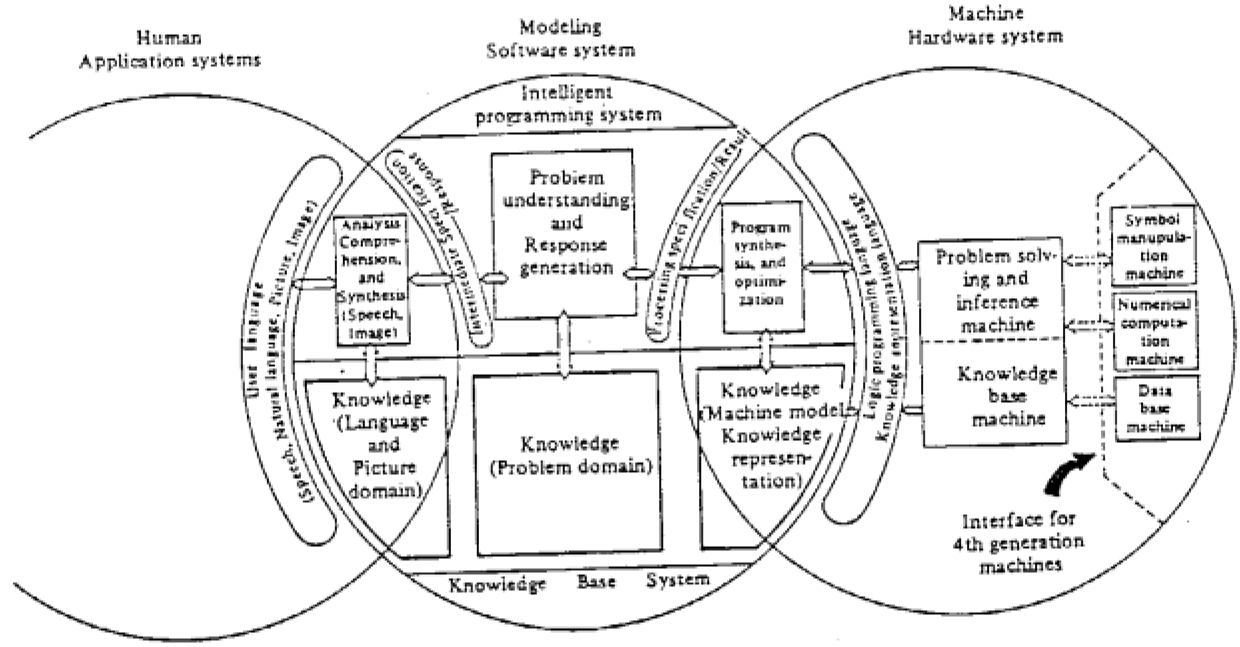


Fig.-1 Conceptual diagram of a fifth generation computer system as viewed from the standpoint of programming

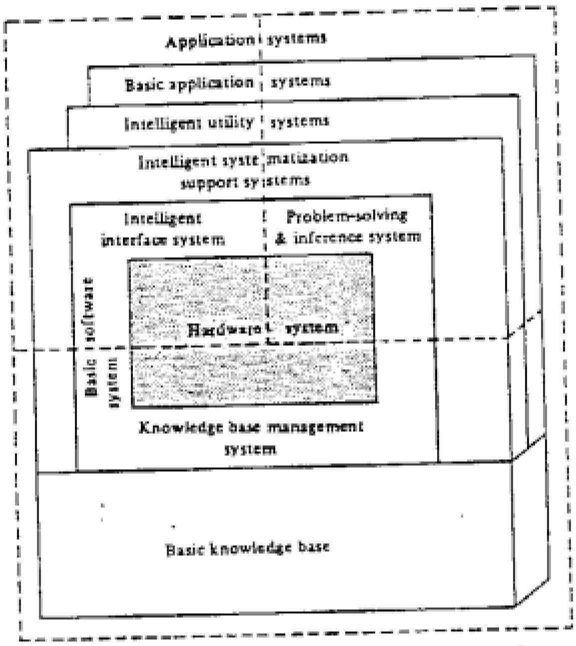


Fig.-2 Conceptual diagram of the composition of a fifth generation computer software system

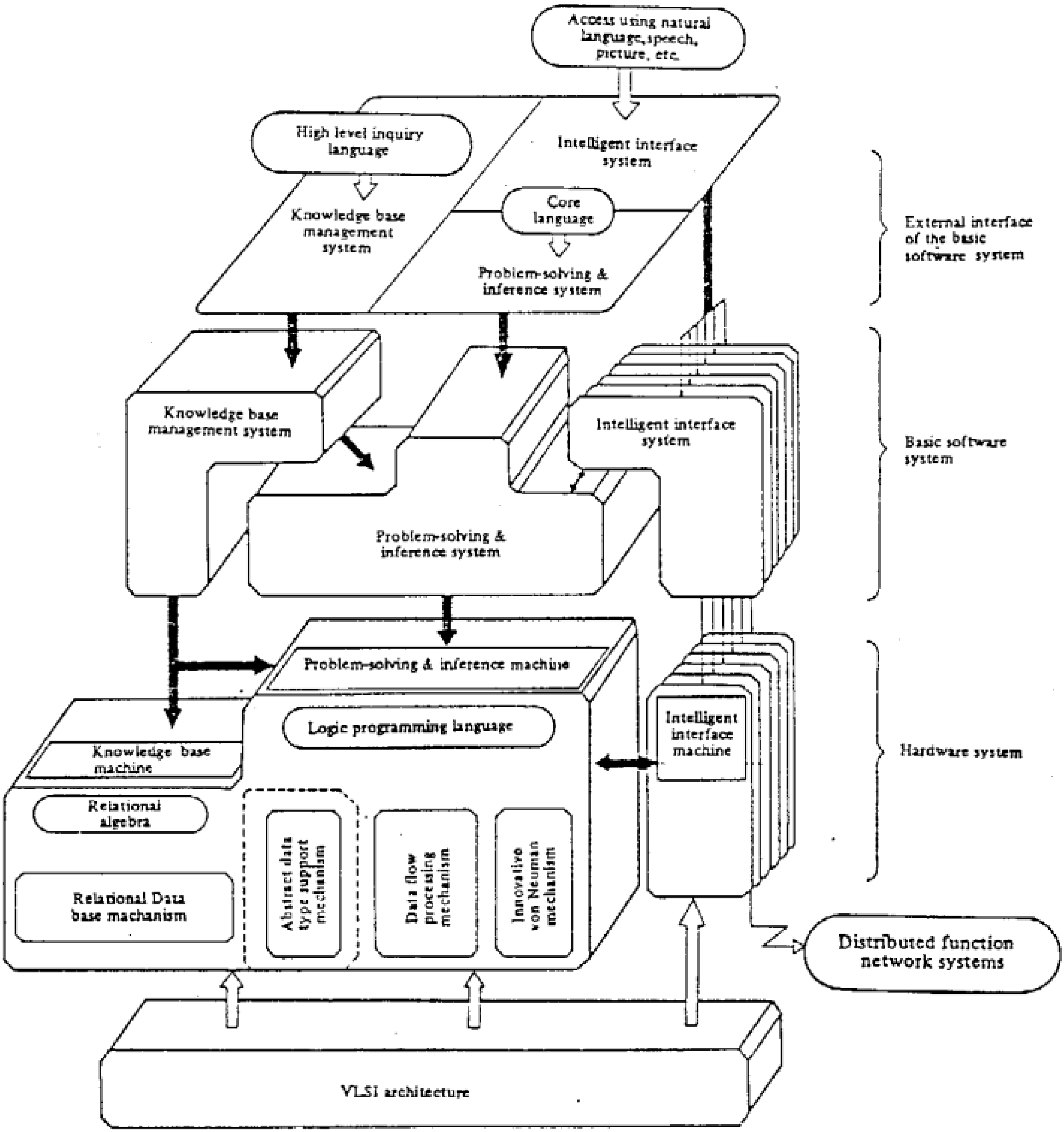


Fig.-3 A conceptual diagram of FGCS

Table 2 Subjects and targets for basic applications systems

Machine translation system

- Translations among multiple languages
- Vocabulary size: 100,000 words
- Machine to guarantee a 90% accuracy, with the remaining 10% to be processed through intervention by man.
- System to be an integrated system where computers participate in the individual stages ranging from text editing to printing and of translations.
- Total costs involved to remain at 30% or lower than those of translation by man.

Consultation systems

- Specimen applications
    - Medical diagnosis
    - Natural language comprehension
    - Mechanical equipment CAD (computer aided design)
    - Computer user consultation
    - Computer systems diagnosis
- Number of objects: 5000 or more
- Inference rules: 10,000 or more
- Semi-automated knowledge acquisition
- Interfaces with system: Natural languages and speech
- Vocabulary size: 5000 words or more

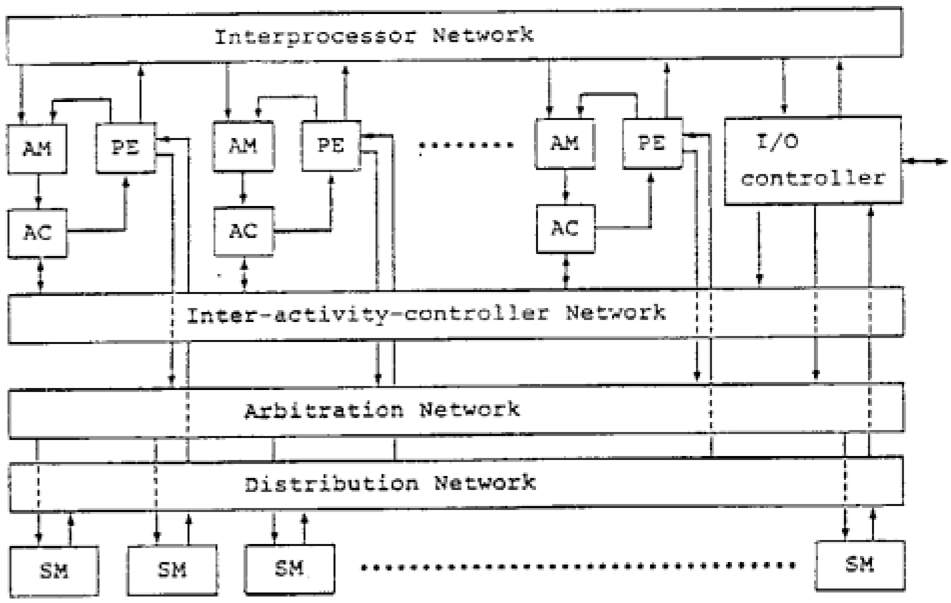
Table 3. Research and development subjects in the initial stage

Subject	Research and development subject (- denotes a technical item for the initial stage)	
Problem-solving and inference system	Problem-solving and inference mechanism	- 5G kernel language - Co-operative problem-solving mechanism - Parallelization of inference mechanisms
	Problem-solving and inference machine	- Data flow machine - Abstract data type support mechanism - Parallel type inference mechanism
Knowledge base system	Knowledge base mechanism	- Knowledge representation system - Large-scale knowledge base system - Distributed knowledge base management system
	Knowledge base machine	- Relational database machine - Parallel type relational operation and knowledge operation hardware mechanism - Knowledge base basic hardware mechanism
Intelligent interface system		- Natural language processing - Speech processing - Graphics and image processing - High-performance interface equipment
Development support system	Pilot model for software development	- Sequential inference machine hardware system - Sequential inference machine software system
	Techniques of integration in VLSIs and systems architecture	- Intelligent VLSI-CAD system - Software and hardware development support systems
Basic applications systems	Machine translation systems	
	Consultation systems	
	Intelligent programming systems	- Modular programming system - Meta/specification description language and verification systems - Automated program synthesis and algorithm bank

```
1. Inference     ---> Logic        ---> Parallel -----I
   Functions          Programming       inference     I
                                        Model         I
                                                      I
2. High-speed    ---> Parallel     ---> Dataflow -----I
   Computation        and               Machine       I
   (Symbol            Pipeline                        I-> Paralle --I
     Manipulation)                                    I   Inference I
                                                      I   Machine   I
3. Modular       ---> Abstract   --I                  I             I
   Programming        Data Type,   I                  I             I
                      Capability   I-> Object --------I             I
                                   I   Oriented                     I
4. Distributed   ---> Concurrent---I   Machine                      I
   Processing         Processing,                                   I---> FGCS
                      Message                                       I     Prototype
                      Passing                                       I     System
                                                                    I
5. A large       ---> Relational ---> Relational --I                I
   Capality           Database        DBM          I                I
   Knowledge Base                                  I-> Knowledge---I
                                                   I   Base         I
                                                   I   Machine      I
6. Software      ---> Personal   -----> Sequential---I              I
   Development        Computer,       Inference                     I
   Tool               Local           Machine    ---I  Super -------)->( 1 chip    )
                      Network                       I-> Personal  I    ( Inference )
                                                    I   Computer  I    ( Machine   )
7. Intelligent   ---> Speech I/O,---> Special   ----I             I
   Man-machine        Picture I/O     Purpose                     I
   Interface                          Processors ------------------I

8. VLSI          ---> VLSI-CAD,  ----> Development  -------------> Intelligent
   Technology         Architecture      Support,                    VLSI CAD
                      Design DB        Hardware                     System
                                       Description
```

Fig.- 4. An Approach to the Fifth Generation Computer System



AM : activity memory module
AC : activity controller
PE : processing element
SM : structure memory module

Fig.-5. An image of the experimental dataflow machine

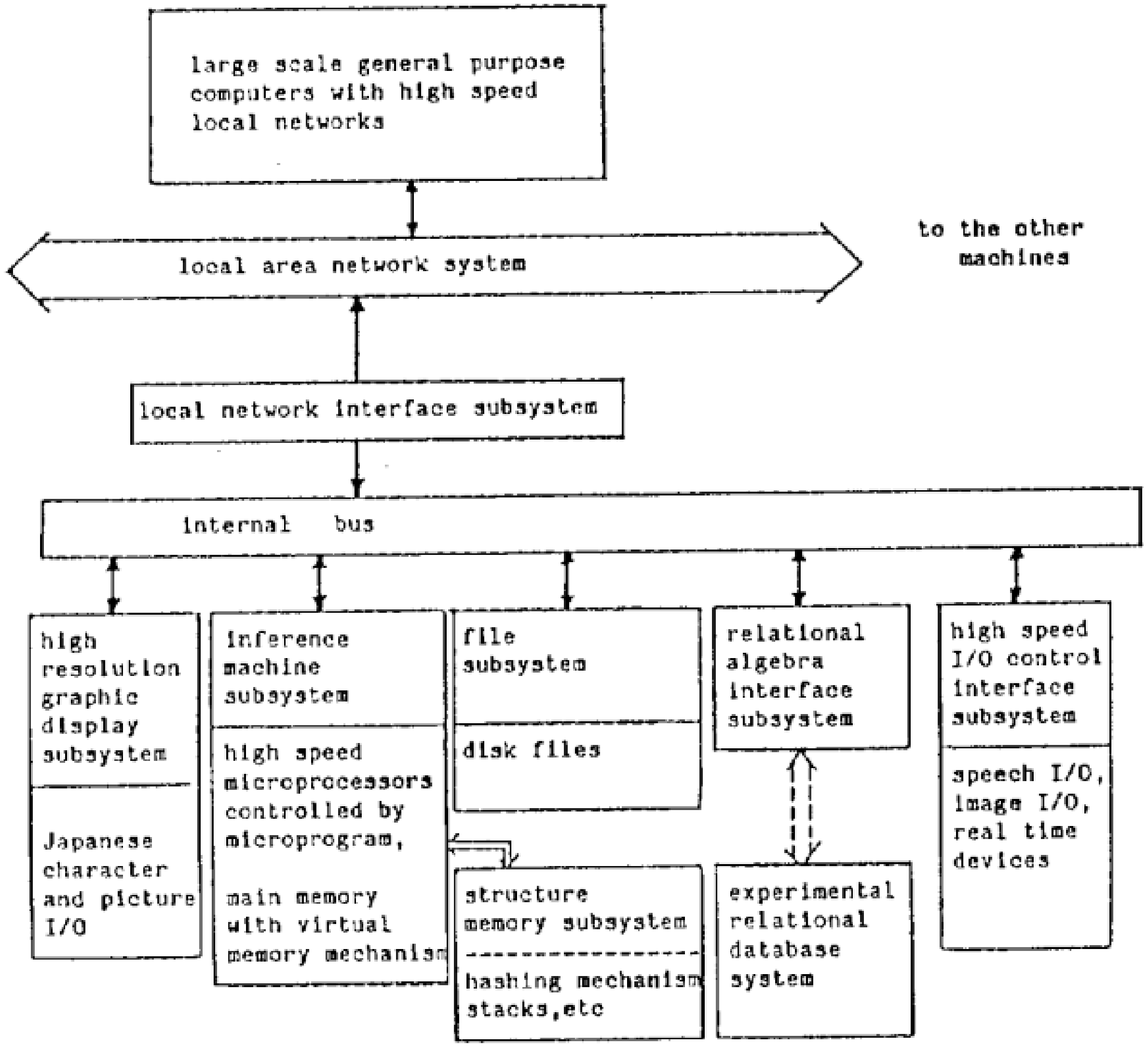


Fig.-7. An organization of the sequential inference machine

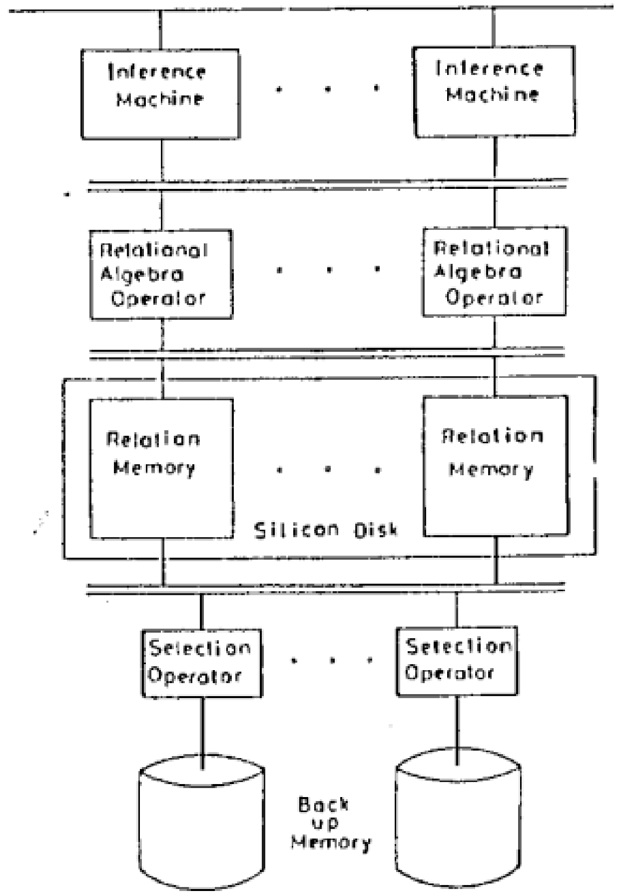


Fig.-6 An image of the small scale relational data-base machine connected to the sequential inference machines

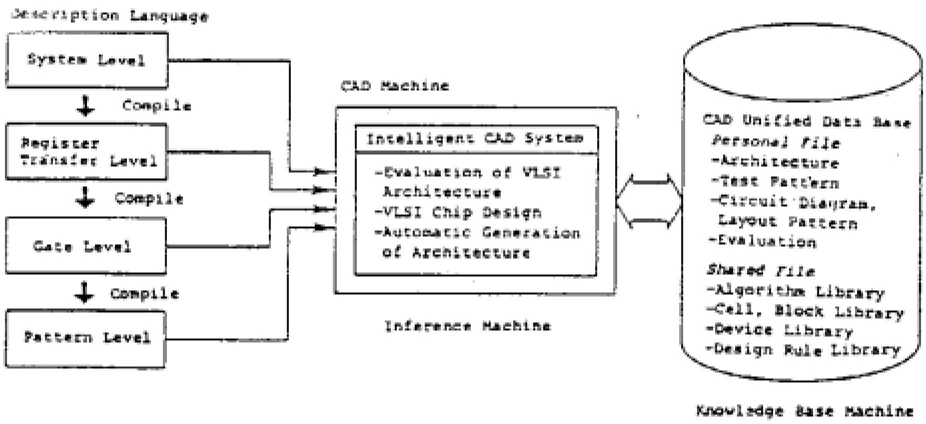


Fig.-8. An image of the intelligent CAD system