ISE-TR-83-35

GAMMA-NET: A LOCAL COMPUTER NETWORK COUPLED BY A HIGH SPEED OPTICAL FIBER RING BUS

– SYSTEM CONCEPT AND STRACTURE –

by

Yoshihiko Ebihara

Katsuo Ikeda

Tomoo Nakamura

Michihiro Ishizaka

Makoto Shinzawa

and

Kazuhiko Nakayama

April 14, 1983

INSTITUTE
OF
INFORMATION SCIENCES AND ELECTRONICS

UNIVERSITY OF TSUKUBA

GAMMA-NET: A Local Computer Network Coupled by a High Speed Optical Fiber Ring Bus
- System Concept and Structure -

Yoshihiko EBIHARA
Inst. of Information Sciences and Electronics, University of Tsukuba, Ibaraki, Japan

Katsuo IKEDA
Inst. of Information Sciences and Electronics, University of Tsukuba, Ibaraki, Japan

Tomoo NAKAMURA
Inst. of Information Sciences and Electronics, University of Tsukuba, Ibaraki, Japan

Michihiro ISHIZAKA
Mitsubishi Electric Co., Kamakura, Japan

Makoto SHINZAWA
Mitsubishi Electric Co., Kamakura, Japan

and

Kazuhiko NAKAYAMA
Inst. of Information Sciences and Electronics, University of Tsukuba, Ibaraki, Japan

The concept and structure of a local computer network are presented for a full scale high performance computer network coupled by an optical fiber ring bus. The design goals of the system are efficient resource sharing and improved RAS. Implementation issues of the system are also discussed, focusing on the optical fiber ring bus and the network operating system services.

Keywords: Computer network, network operating system, optical fiber, ring bus.

1. Introduction

Great progress has been made in expanding computer utilization over a wide range of social activities. As computers become more relied upon in society, the computing power required of them becomes greater and the quality of the processing becomes more critical. Likewise, the reliability, availability and serviceability (RAS) of those computer systems become more critical.

Augmenting the processing power and the storage capacity of a single computer system alone is not always an effective way of coping with the variety of today's computing requests. A distributed processing system, however, might meet these computing reqiurements, improve response time and reduce communication cost. A distributed processing system is especially superior to a single computer system in RAS, computing power growth potential, and the capability to add new functions as

Kleinrock[1], Zafiropulo[2], and Uyetani[3] have concluded.

Today's computer utilization shows the general tendency to connect geographically distributed computers and to share their valuable resources. This distributed processing is, also, an effective way to improve computing power and to share resources even when the distance between two processors is short; Wilkes and others stressed the significance of local area computer networks [4]-[6]. Groshe's law on cost performance doesn't apply to very large computers; it is more economical to build a computer complex, connecting two or more medium-scale systems, than to use a single large-scale computer to augment processing power.

At one time, a single large computing system was in service at the Science Information Processing Center (SIPC) of the University of Tsukuba for a large variety of processing of scientific, educational and administrative data, and served for research and development of the information science system [7]. The computing center had often experienced bottlenecks in this system because of local imbalances in processing power, and demands for computing were becoming more and more complex and diverse. In order to handle a large variety of requests to a single large-scale computer, it is necessary to raise its processing capacity greatly, and this is neither an economical nor an effective way to improve RAS. In view of these circumstances, the SIPC, in 1978, started to build a full-scale distributed processing system and named it the GAMMA-NET (General purpose And Multi-Media Annular NETwork) [8].

The GAMMA-NET was designed to serve multi-media data traffic that includes relatively short data transmissions, such as interactive TSS data communication of more than 200 terminals, and the transmission of large amounts of data ranging from several hundreds Kbytes to several megabytes, such as file transmissions to support an electronic mail system, an RJE system, or an image processing system. Also, a future goal for the GAMMA-NET is that it serve real-time data such as voice data communication. As a consequence of having to satisfy a large variety of traffic with needs of both volume and transmission speed, it was necessary to set up a high performance communication subsystem.

Of particular importance in the GAMMA-NET is the high-speed ring bus subsystem used to connect computers. This utilizes new LSI technology and optical fiber communication technology. Kobayashi has reviewed the trends of new technology, such as the LSI and the optical fiber [9]. LSI technology reduces the size and cost of arithmetic and logical processing units and memory units, and enables system designers to employ compact function elements. We have developed a firmware- and hardware-controlled Ring Bus Processor (RBP) which performs efficient data transmission. Optical fiber technology also provides high-speed, high-quality, and low-cost transmission lines.

Another major factor in the GAMMA-NET is the role of the Network Operating System (NOS), by which each Operating System (OS) of an individual subsystem is loosely coupled together by the help of interprocess communication facilities. Our interprocess communication facilities cover various types of data transmission, while the NOS developed by Tanaka et al. [10] is limited to message oriented communication. We employed the layered approach in installing the NOS rather than the no-layer

approach discussed by Spector [11]. Rowe and Birman reported on a loosely coupled system constructed by connecting UNIX systems [12]. The main reason why a loosely coupled networking scheme was adopted was to accomplish all of the network functions without major modification of the existing OS's, and to preserve the autonomy of each subsystem operation in the network environment. The NOS operates as the logical kernel of the system and consists of many processes on and around the Network Management System (NMS).

In the following the design concept and implementation issues of the GAMMA-NET are presented, focusing on the network architecture based on the optical fiber ring bus and the NOS services.

2. Design Principles of the GAMMA-NET

The basic design principles of the GAMMA-NET used to support an integrated computer network are described both from the physical and the logical aspects of system architectures.

From a physical point of view, the system is tightly coupled by a high-speed optical fiber ring bus through the RBP's, which are discussed later. This is because we first aimed at implementation of a high-speed communication subsystem which can transfer a file of large amounts, e.g. 1 Mbytes, in a few seconds. Second, in the system a file subsystem is considered to be supported by a back-end processor of which files can be accessed from remote functional subsystems. Thus, the ring bus is designed to be able to transfer data as fast as an I/O channel of a file subsystem, so as not to become a bottleneck in the system. Third, the GAMMA-NET is designed as state-of-the-art research to introduce a high-speed optical fiber bus, e.g. 100 Mbps (bps: bits per second), to develop a high-speed interface of a computer for connection to the bus, and to investigate the feasibility of each.

From a logical point of view, the existing subsystems are loosely coupled under the control of the native OS's, preserving the independent operations. This is because we first aimed to implement the system without greatly modifying the existing OS's to avoid significant impacts on the individual systems. Existing OS's within the system, which were built by different computer manufacturers, have distinct system architectures. Rewriting these heterogeneous OS's to fit a new network operating system would have been impractical. Our approach in introducing an NOS minimizes the amount of support software that must be developed to organize a computer network and allows clients to transfer to the new environment more easily. The drawback to this approach is that an existing program may be inefficient in resource utilization compared to an equivalent program specifically designed for the network. We believe, however, that the higher performance of the ring bus subsystem will compensate for this drawback. Second, computer reliability technology has been applied to single computers so far, rather than to network-wide activities. Each existing OS has established many recovery schemes. From these points we concluded that

the autonomy of each computer should be preserved to remain active or to restore the normal operation in case of network failure. In fact, the proposed NOS has intersystem interaction only by means of interprocess communication facilities. Such a loosely coupled network adequately preserves the autonomy of each OS. Third, it is very important to improve the RAS of the whole system. The NOS lies in one of the higher level application protocols in the hierarchy of the GAMMA-NET protocols, instead of being placed at the core of the system. Thus, the whole system will not crash even if the NOS fails.

3. GAMMA-NET Architecture

Fig.1 shows the hardware configuration of the system and Fig.2 illustrates the protocol structure based on the hierarchical control layers.

3.1 Hardware structure

3.1.1 Ring bus subsystem

Ring bus: The communication bus forms a ring. The feature of a ring bus is fully discussed by Wilkes [13], Pierce [14] and others, and is suitable for reducing overall system cost. The ring in our system employs dual 100 Mbps optical fiber lines. These lines are identical and there is no separate control. Almost all elements of the ring bus subsystem were designed to operate at 100 Mbps; however, the whole ring bus subsystem operates at 32 Mbps currently because some parts, mainly in the optical repeater (REP), were not available that could operate at 100 Mbps. The transmission channels of each bus are subdivided into nine subchannels as shown in Fig.3; one (Subchannel CH.0) is used for ring bus control and the others are used for data links, each with a transmission capacity of 320 KBPS (BPS: bytes per second). Multi-channel data transmission is allowed by combining up to five successive subchannels to expand the bandwidth to a maximum of 1.6 MBPS, according to the processing capacity of the communicating computers.

RBP: The ring bus subsystem consists of the ring bus, RBP's and a Ring Bus SuperVisor (RBSV). The RBP is a specially designed dedicated-processor for high-speed communication and is directly connected to a subsystem via an I/O channel. Most of the link level protocols adopted in an RBP are equipped with firmware and hardware. This relieves the GAMMA-NET subsystems from the task of managing the ring bus communication. The bit sequence of a packet is routed around the ring bus and only suffers a short delay at each RBP before being forwarded, just enough delay for discrimination of the header as to whether this packet was to be sent to this RBP or not. It is not completely read and stored in the manner of the the delay-buffered insertion as proposed by Liu et al. [15]. Thus, if an RBP is not transferring any

packet this delay is only 27 bits of time. An RBP is highly modularized, as described below, and has an REP for connection to the communication line.

The REP has its own power supply fed from a power line which is wired along with the optical fiber line. An RBP is connected to an REP through a Bus Interface Adapter (BIA). An RBP adapter connects a common interface of an RBP to an I/O channel interface of a computer. A Transmission Control Unit (TCU) executes the data link protocol and the physical level protocol through the BIA and the RBP adapter. The TCU executes error checks by CRC, acknowledgement of data reception and retransmission of erroneous packets. The TCU also operates to divide a message into 25-byte packets and transfers the packets to the ring bus, and receives packets and assembles them into a message. An RBP has an Attached Service Processor (ASP) which is a dedicated processor for diagnosis and performance measurement. The ASP watches the health of the RBP, reports the RBP status to the RBSV for system diagnosis and handles the switches in the REP to reconfigure the ring bus when it is ordered by the RBSV or the host processor. The ASP, also, puts time-stamps into the measurement record when events specified by the RBSV occur and reports these to the RBSV. A subsystem is connected directly to the RBP without any front-end-processor in order to decrease data transmission delay. Firmware implementation of data link control minimizes data transmission overhead; software support of the front-end-processor, as reported by Kawai et al. [16], may cause overhead to become a bottleneck in the ring bus subsystem. Major portions of the RBP are duplicated to guarantee reliable, high performance operation, and these operate concurrently under normal conditions.

RBSV: The RBSV not only monitors the health of the RBP's, by gathering the status of every RBP at fixed intervals of 250 milli-seconds but, also, orders an RBP to switch a line connection in the REP to reconfigure the system topology of the ring bus when it is necessary to do so. When some parts of the ring bus subsystem or the functional subsystems fail, the defective parts are cut off by bypassing them or by turning back the bus, so that the rest of the system can continue operations. The RBSV also collects performance measurement data on the ring bus and these data are processed by the unique NMS.

3.1.2 Functional subsystems

The GAMMA-NET has several subsystems, each of which is installed as a general purpose system but is specifically tuned to interactive processing (MELCOM 800III-2CPU, 8MB, 3MIPS), to interactive programming for CAI (MELCOM 700III, 4MB, 1.5MIPS), to batch processing (FACOM M200-2CPU, 12MB, 23MIPS), to administrative processing (FACOM M160, 6MB, 0.74MIPS), or to network management processing (MELCOM 70, 0.8MB, 0.6MIPS). Since a variety of processing capabilities is expected, subsystems are not identical and are designed for efficient function-oriented operations with good response. A Terminal Interface Processor (TIP) is one of the subsystems through which remote terminals, remote job entry devices and other peripheral devices are connected. Every device in this subsystem can be connected to any subsystem to utilize its resources and services.

3.2 Network operating system and hierarchical protocol system

3.2.1 Loosely coupled NOS

The GAMMA-NET system offers the NOS services to standardize and integrate the network access and services for users. Our motive for constructing a loosely coupled NOS is to reduce the degree of centralization of supervisory control and to make operation of computers as independent as possible, so as not to degrade RAS by the existence of the NOS. The crash of an NOS whose control is too centralized may damage the whole system, even though such an NOS may be desirable for better resource sharing. Therefore, an NOS should be weakly coupled with the existing OS's from the viewpoint of network management. The NOS proposed in the GAMMA-NET is a virtual network operating system and consists of supporting processes distributed within each subsystem. These processes are combined into one to form the NOS, which manages network operation and offers services to make the man-machine interface smoother. The processes in the NOS execute network functions, which are offered by the Network Management Protocol (NMP). The structure of the NOS is shown in Fig.4.

The processes of the NOS, located in the NMS, mainly collect and display system information, select and assign the most inactive subsystem to a user at a terminal, and search for files at the request of a user. These processes play an important role as the nucleus of the NOS. The other processes of an NOS, located at a subsystem other than the NMS, are to report local computer information to the NOS at the NMS subsystem. These complement the NOS nucleus. Both nucleus and complementary processes cooperate to achieve uniform network management as if a single network operating system were operating. Thus, a failure of any complementary process results only in the loss of information from the computer on which this process is running. The shut down of a nucleus process results in the loss of services, from which a user might be somewhat inconvenienced, however, the user can access any subsystem about which he has information. Thus, any process malfunction is not critical to the whole system operation. Moreover, the usefulness of the loosely coupled NOS is shown by the potential to add any computer to the system or to take one off from the system easily. Only implementation of local processes is required for a newly attached computer to participate in the NOS services.

3.2.2 Data link control layer

At this level, the Data Link control Protocol (DLP) specifies and offers services for the connection and disconnection of a data link, and for message transfers by the synchronization mechanism. The data link control procedure is divided into three phases - the connection phase, the data transfer phase and the disconnection phase, as shown in Fig.5. The data transfer phase is separated into a synchronization subphase and a data transfer subphase.

(1) Computer-RBP interface

In general, the I/O subchannels of a subsystem are used as follows.

1) A subsystem uses 32 I/O channels to communicate with an RBP. Subchannel #0 is used to control the RBP and the rest are used for data transfer; one subchannel supports one data link.

2) Assignment of subchannels #1-#31 are determined by the RBP at the time of data link establishment.

(2) Connection phase and disconnection phase

As an example, suppose that there are two subsystems, subsystem-A and subsystem-B, which are connected to the ring bus via RBP-A (Ra) and RBP-B (Rb), respectively, and a message is transmitted from A to B. The connection and disconnection procedures are illustrated in Fig.5. When subsystem-A issues a link request order, CONNECT LINK, via subchannel #0, RBP-A assigns one of the free subchannels, say #i, and sends back subchannel number #i as a new data link number to subsystem-A by means of an interrupt. Meanwhile, RBP-A transfers a data link connection command CL with the identification of RBP-A and the subchannel, Ra and #i, to the destination RBP (RBP-B of subsystem-B). Next, RBP-B assigns a free subchannel, say #j, and notifies both subsystem-B and RBP-A of the identification of RBP-B and subchannel number #j. By these procedures subsystem-A knows the assigned data link number as #i and that of subsystem-B as #j. The data link set up between RBP-A and RBP-B is identified by a subchannel number pair (i,j). A subsystem can handle a remote destination as if it were its own I/O peripheral device, as one data link is controlled by one I/O subchannel of the subsystem.

The disconnection procedure of the data link terminates the connection by exchanging disconnect commands DS's.

(3) Read/write synchronization of interprocess communication

A pair of processes between which a data link has been established communicates synchronously as explained below. Once again using the system example, a write request order WR issued from a process on subsystem-A and a read request order RD issued from the associated process on subsystem-B are matched at both RBP's to synchronize read/write operations. This order matching operation is called the rendezvous operation. An acknowledgement order STS from RBP-B is sent to confirm the order matching. After the matching between the related communicating RBP's is confirmed, RBP-A starts to transfer a message. If there is more than one pair of processes which have been linked together, the pair of processes which have completed the RD/WR matching first will be served first. Each of all the orders is sent via one free slot.

(4) Slot reservation for data transmission

Data transmission is performed in the slot reservation mode. To start data transmission a slot reservation request order WT must be issued. When RBP-A finds a free slot and sends the WT order to

RBP-B, the free slot number is saved in both RBP's. Successive transmission of data packets, preceded by the WT order, continues until the end of the data on the same reserved slot. The synchronous read/write operation is a basis for a highly efficient communication bus system. The rendezvous delay time, which is the time spent to match a read/write operation between communicating processes, is mainly the processing time of a user process during the intercommunication period or the result of an imbalance in the processing speed of subsystems. If communication processing overhead can be reduced and processors are interconnected with a network of high bandwidth and low latency time, the rendezvous delay time becomes about the same as traditional delay time, including latency time and propagation time. In the case of an asynchronous interprocess communication scheme, too much data transmission without carefully taking account of the partner's processing speed may result in the stuffing of receive buffers. Then, retransmitted packets caused by the busy status at the receiver's RBP may fill the ring bus, causing a ring bus overflow. Instead of an asynchronous interprocess communication scheme, the GAMMA-NET employs the synchronous interprocess communication scheme, which is more suitable for a high-speed ring bus. The GAMMA-NET scheme, however, differs from typical network communication schemes in two ways: First, message transmission starts at the time of completion of the read/write matching, rather than at the time of acknowledgement of the previous message transmission. Second, the data flow control works at the DLP level instead of the end-to-end level.

The transmission capacity of the synchronous interprocess communication scheme is limited, mainly, by the speed of the lower level control of communicating subsystems. This can be seen if one examines the data link operation. The rendezvous operation can be executed in parallel among more than one data link, and thus, avoids the performance degradation of the whole system.

The transmission of control orders at the DLP level is executed in the packet switching mode, while the transmission of data is executed in the message switching mode. In the packet switching operation, source and destination addresses and control information are placed at the head and tail of a packet for every transmission. The packet header overhead is considerable unless the ratio of the header length to the data length is small, as pointed out by Kleinrock [1]. If we apply the message switching scheme for the data transmission, we only need to pay the smaller overhead cost to reserve the fixed slot, without any addressing information on the data transmission. Moreover, implementation of the DLP becomes very simple, since the RBP controller needs to handle only the transmission and reception of data, once data transmission is started. This may cause additional packet retransmission overhead because a busy RBP that is currently transmitting data can not accept any orders from other RBP's, so the rejected orders must be transmitted again. However, there are very few such retransmissions in practical use because of the very short delay time of one I/O operation.

3.2.3 Network control layer

At this level, the Network Control Protocol (NCP) is prepared to manage a logical link between

communicating processes. The NCP provides two transmission modes according to the characteristics of data transmission. There is a multiplex mode to be used for relatively short data transmission, like TSS conversation, and a burst mode for bulk data transmission, like the transfer of a file. Currently the maximum message length is restricted to 32 KB for burst mode transmission and 2 KB for multiplex mode. These maximum message lengths were selected taking into account the minimum I/O buffer capacity of the existing computers. Comparison of the characteristics of the two transmission modes is shown in Table 1.

3.2.4 Function control layer

At the function control layer there are two kinds of protocols according to the kind of access method used: The File Access Protocol (FAP) and the Interprocess Communication Protocol (ICP). Standard library routines to utilize the interprocess communication facilities are prepared for higher level programming languages such as FORTRAN.

FAP: The FAP is prepared to access virtual files in any subsystem. This access method is general and powerful, and can be applied to almost all types of files, devices and processes. When a remote file is accessed, a logical link is established first between the remote file server process and the user process. Before opening the file, interactive negotiations verify file access properties such as the OS type, file organization, file structure and attributes, and file access authentication; then read, write and positioning operations are executed. A concept analogous to the FAP is adopted by DCNA, which is being developed by Nippon Telegraph and Telephone Public Corporation [17].

ICP: The ICP is prepared to standardize a logical link as an access method interface for efficient interprocess communication. This access method is used to simplify the interprocess communication procedure for clients. The need for the ICP is also discussed by Walden [18].

3.2.5 Application layer

At this level, application protocols are prepared for a number of applications to facilitate specific functions; the TSS protocol, the RJE protocol, the NMP and the File Transfer Protocol (FTP) are specified so far in the system. Application protocols are supported by the FAP or by the ICP.

TSS protocol: The TSS protocol is nothing but a virtual terminal protocol prepared for acquiring the TSS services offered by any subsystem from any terminal. It defines a standard code set, control characters and terminal control sequences. A pair of logical links used for the full duplex mode operation are established by request from a terminal.

RJE protocol: The RJE protocol is supported by the FAP and is prepared for remote batch processing. This offers standard procedures for job entry, job execution, output control and other service functions from a remote terminal.

NMP: The NMP is supported by the ICP and offers such functions as system information guidance,

automatic subsystem selection, load leveling, automatic file/attributes searching and statistical measurement of the system.

FTP: The FTP is supported by the FAP and provides such functions as retrieval and transfer of a file to and from remote subsystems.

Some other protocols are supported by the FAP or the ICP and supply applicable functions for users. They are used for special purposes among groups of network researchers.

4. Implementation of Network Services

4.1 Network management and network commands

A user can access the NMS to utilize the NOS services as if he is accessing a single large computer system. The system offers network commands for a TSS terminal user to get the NOS services in interactive processing. A command preceded with the "@" mark is a network command and, in general, is executed at the NMS. The user can enter a network command at any time while running a program on TSS service or on any other service, provided the terminal is ready to accept a command.

The basic flow of network command processing is described in the following. Let us suppose that a user is connected to a certain subsystem. When he needs the NOS services, he enters a network command by first typing "@". The TIP recognizes that this command should be processed at the NMS, establishes a connection to the NMS, and then transfers the command to the NMS. After completion of the specified services of the network command at the NMS, the user may terminate the connection by entering @END. Meanwhile, the previous connection to the subsystem remains active and the user may continue the processing on the connected subsystem.

4.2 Service facilities

The NOS supports the following network services.

(1) System information guidance

The NMS allows clients to access system information regarding the system status, the kinds of available resources and the subsystem types. Schedules of system operation, system documentation and a network newspaper are also included. Many network commands are provided to retrieve these resources.

(2) Automatic subsystem selection

A user may enter a network command, @SELECT [TSS] [host id], either to select a specific

subsystem by showing a subsystem name explicitly in the second argument or to select some subsystem automatically without specifying a subsystem name, whenever he begins TSS processing or switches access from one subsystem to another. When a TIP receives a command without a subsystem name, the TIP sets up a connection with the NMS and forwards the command to it. The NMS computes the loading factor by using the information in the system information file and the status information file, and selects the most inactive subsystem which can satisfy the request from the user. Then, the NMS sends back the subsystem name to the TIP and terminates the connection. Thus, the TIP is informed of the most suitable subsystem and establishes a connection between that selected subsystem and the user. When the NMS faiis to select a subsystem, the TIP can not set up a connection to it. In this case, the TIP notifies the user that the NMS can not select any subsystem and provides a prompt requiring the selection of a subsystem.

Automatic subsystem selection promotes more efficient sharing of network resources and better load leveling.

(3) Collection of system information

The System information Collecting Process (SCP) at the NMS demands system information in standard format every 30 sec. from the remote Information Support Process's (ISP) which reside in each of the subsystems. The remote ISP converts the native system information format into the network format specified by the NMP. The information received at the SCP is grouped and stored in files - the system information file, the resource information file and the status information file, as illustrated in Fig.6. The SCP handles the following three kinds of system information.

System information: the state of the ring bus subsystem (bypassed, looped-back or normal state), subsystem features (machine type, OS type and the type of special peripheral devices for image and Chinese character processing), a site-dependent constant, as well as the MIPS value and the memory capacity, etc.

Resource information: available software utilities (language processors, application utilities and the type of data bases, etc.). These contribute to better utilization of the system by and for clients.

Status information of a subsystem: the number of running jobs, the number of queued jobs, TSS logon counts, I/O request counts, etc.

These values are used to characterize machines with different architecture and speed. The most inactive subsystem is chosen by a load leveling algorithm, which takes into account some of these values. The SCP and the ISP's can communicate to each other by using the ICP. Malfunction of a remote ISP or the SCP causes situations where the contents of files may not be updated by new information and may be getting older and less valuable, but a client can still continue to access subsystems. He may feel inconvenienced by the poor quality and accuracy of the system information, but the situation will only compromise the reputation of the NMS's serviceability, and will not affect the whole network operation.

(4) Load leveling strategies

Leveling the workload among subsystems is of potential importance to the efficient utilization of a local computer system. Two deterministic and probabilistic load leveling policies were suggested by Chow et al. [19] and Hwang et al. [20]. A deterministic strategy assigns a job to an appropriate computer based on the current state of the network. A probabilistic one dispatches jobs in proportion to the processing speed of a computer. In general, the probabilistic approach is easier to implement but is only suitable for a static environment whose share of load is determined statistically, while the deterministic approach can optimize load balance against dynamic alteration of traffic as time passes.

Load leveling in the GAMMA-NET is implemented on the NMS with a deterministic load leveling policy. The algorithm is such that the NMS selects the most inactive subsystem for batch or remote entry jobs by the loading value that is calculated by dividing the figure of CPU speed (MIPS) by the number of jobs which are either running or waiting, and for TSS users by the loading value that is calculated by dividing the main memory capacity by the TSS logon counts. Each value is obtained from the data stored in the status information file and the system information file of the NMS.

The computation of loading values only takes into account a finite number of factors, and makes assumptions about the average CPU processing time and the amount of memory and file used. Therefore, it is merely an approximation of loading of a subsystem. Despite these limitations, balanced utilization of the system tends to provide higher system throughput, to minimize the average job response time, and to reduce the processor idle time.

(5) File search method

The automatic file search function is useful for terminal users as a means searching for files in the network. In this system a user is provided with the location and attributes of a file by entering a list command, @LIST [file name]. The TIP adds the user's file access privileges, implied by the user id and the password, to the list command before transferring it to the NMS. The user id and the password are registered on the TIP's authentication tables at the time of the logon session to the system. The access privileges are unique to the system and are common to all subsystems. The NMS, when it receives such a command, makes a connection to the desired subsystem, using interprocess communication facilities, and sends the list command to the connected subsystem where a local list program is activated to search for a file name designated by the command. The retrieved file attributes and the subsystem name are sent back to the TIP via the NMS as shown in Fig.7. Therefore, file attribute formats printed on the terminal are native to that of the local list program. This operation is repeated until all the available subsystems are searched.

Our off-the-shelf file referencing mechanism doesn't require complex implementation for file system operation, in contrast to that of building a new standard network file system.

Another feature to note is a catalogue command, @CATALOGUE, which is used to print out all of the names of the files which a user created at any subsystem. The control mechanism is the same as that of the list command as described above; the printed content is a set of file names and locations of subsystems.

(6) Collection and statistical measurement

An NMS operator can issue commands to collect measured data related to the ring bus communication traffic. The Statistical Process (SP) of the NMS activated by commands can order the RBSV to start/stop a measurement and collect the measured data. The general flow of statistical measurement is shown in Fig.6, where the ASP at RBP-i samples time-stamped events at every 60 micro sec.; the sampled data are transferred to the RBSV, then all of the sampled data are gathered and finally stored in the statistical files after statistical calculations have been performed. The more detailed measurement flow is discussed in our report [21].

4.3 Implementation of subsystem selection

Fig.8 illustrates the detailed flow of subsystem selection services for TSS processing. First, a terminal user presses the BREAK key and registers his user id and password to the TIP after the "LOGON PLEASE" message is issued from the TIP. Then, the TIP outputs a "@" mark which prompts a network command input. If the user enters a select command with no subsystem name as its argument, the TIP establishes a connection with the NMS and requests the optimal subsystem selection for that user. The Network management PRocess (NPR) activates the SELECT routine to calculate the loading value. After receiving the selected subsystem name from the NPR, the TIP terminates the connection with the NMS, and begins to establish a new connection with the selected subsystem. The TIP executes the logon procedure on the selected subsystem for the user. The user can use TSS services of the subsystem after the "!" mark is output, which indicates successful logon. The user may input a select command with a subsystem name when the NMS doesn't work or users want to use a specific subsystem.

5. Discussion

Some experimental results regarding the transfer characteristics of the RBS, which are measured by utilizing the statistical data collection and performance measurement facilities of the RBS, are shown. These include the processing time to connect and disconnect a data link, the rendezvous delay time of read/write matching, and the data trasmission speed and subchannel utilization of the multiplex mode and the burst mode.

Connection time: This is defined as the elapsed time between a link request order from a subsystem and an acknowledgement of the link establishment to the local subsystem, after receiving a connect accept command from the remote RBP. The connection procedures are illustrated in Fig. 5. For both the multiplex mode and burst mode, the processing times prove to have nearly the same processing time distribution, independent of the transmission modes. This distribution can be said to be of nearly uniform distribution. The mean connection time and the variance are 3.4 milli-seconds and 3.8, respectively.

Disconnection time: This is defined as the time required to terminate a data link connect by exchanging disconnect commands DS's. The mean disconnection time and the variance are 2.1 milli-seconds and 1.5, respectively.

Rendezvous delay time: Generally, this is defined as the time required to match a read/write operation at the initiating RBP. In this section, however, our primary concern is the sender's behavior, thus we defined it as the time between a write request order WR and an acknowledgement order STS. The mean rendezvous time and the variance in the multiplex mode are 22.5 milli-seconds and 942, respectively. And the mean rendezvous time and the variance in the burst mode are 19.5 milli-seconds and 603, respectively. The relatively larger variance seems to be caused due to the imbalance of processing power of subsystems in which the communicating processes reside.

Data transmission time: This is defined as the time between a slot reservation request order WT and the end of the data transmission. In the GAMMA-NET, the interactive data such as TSS data is transferred by the multiplex mode. The average length of data in the multiplex mode is 45 bytes, and the mean data transmission time is 0.6 milli-seconds. As a result, the effective transmission speed of the multiplex mode is estimated to be 1.95 KBPS by dividing the mean length of the data (45 bytes) by the mean rendezvous delay time (22.5 milli-seconds) plus the mean transmission time (0.6 milli-seconds).

Files are transferred by the burst mode in the GAMMA-NET. Currently the FTP supports only sequential files which occupy about 68 % of the existing files. The average length of data in the burst mode is about 34 Kbytes, and the mean transmission time is 120 milli-seconds. Consequently, the effective transmission speed of the burst mode is estimated to be about 243.7 KBPS by dividing the mean length of the data (34 Kbytes) by the mean rendezvous delay time (19.5 milli-seconds) plus the mean transmission time (120 milli-seconds). Since the maximum transmission capacity of an RBS subchannel is 320 KBPS, the mean values for the subchannel utilization are 0.76 and 0.006 for the burst mode and for the multiplex mode, respectively. From these figures, we can expect that one RBS subchannel can support about 166 (=1/0.006) terminals as a maximum capacity, provided that each communication is independent and does not interact with any other.

6. Conclusion

The concept and structure of the GAMMA-NET are presented above in detail. Several of the more important features of the system are summarized here.

(1) The off-the-shelf approach of the NOS implementation has saved significant development overhead, especially in such a full-scale computer network system. The NOS plays an important role in network management by using interprocess communication facilities. Also, the loosely coupled structure of the NOS is considered to be effective in improving the RAS of the whole system.

(2) The most capable and reliable ring bus subsystem has been developed by using high-speed optical fiber lines. To provide a well integrated communication network and to guarantee the best network performance, the subsystems are tightly coupled by the ring bus subsystem.

(3) In order to average an unbalanced workload or an unbalanced processor capacity within the GAMMA-NET, a deterministic load leveling scheme has been proposed for efficient network resource utilization.

(4) It has been shown that the implementation architecture of the GAMMA-NET is applicable for short-distance communication networks such as computer complex systems, and contributes well to the integrated development of a high performance network system.

(5) A commercial version of the ring bus subsystem [22], based on the RBS architectures of the GAMMA-NET system, is on the market and supplied by the Mitsubishi Electric Co. Since the GAMMA-NET has been developed as a proto-type system and has many experimental facilities and utilities, the newly marketed version is more simplified and the cost-performance is improved. Also, the NMS of this marketed system is provided as a software system that will be installed within a functional subsystem. In constrast, the NMS in the GAMMA-NET system is implemented on a separate subsystem which is dedicated exclusively to the NMS.

(6) Forming a network of distributed computers without major modification of their native OS's was the emphasized theme in this paper, because the standard access method of the native OS is designed to support a number of low-speed terminals and is not suited for bursts of high-speed data transmission. However, we think that it becomes necessary to upgrade the standard access method of native OS's for a very high speed communication subsystm. In the GAMMA-NET, files are transferred from the file system, via the standard access method, to the RBS. This architecture is still practical for file transmission, but,

for a communication subsystem with a speed of more than 100 Mbps, we recommend a system designer modify the native OS to transfer files directly from a file subsystem to the RBS. Further, the technical feasibility of a 1000 Mbps optical fiber communication subsystem is discussed by Kuo [23] and such a very high speed communication subsystem may be operational in the future.

(7) File access privileges, on the GAMMA-NET, are common throughout the subsystems. This policy of total freedom in accessing files throughout the system, so long as the system password and user name are known, is superior in the ease of system implementation. Also, file access is simplified for clients as opposed to a method in which file access privileges are different on each subsystem and are created and used independently. However, this strategy is inferior with regards to subsystem and file security since a subsystem and each file on that subsystem are accessible with the same user name and password. In other words, with the knowledge of a user's id and password all of his files may be easily accessed. Therefore, an adequate level of file access privilege assignments should be employed for the system in the future. For example, the TIP's authentication tables may be allowed to include plural user id's and passwords for a single user. The TIP, then, would handle all privilege verifications.

(8) The performance evaluation of the RBS, presented in the discussion, is limited to a certain aspect of the basic characteristics of the system. The remaining problems to be investigated include the processing time of end processors, the native OS's overhead, and the transfer capacity of the file systems.

Acknowledgements

This project has been jointly carried out by the SIPC of the University of Tsukuba and by the Mitsubishi Electric Corporation. The authors would like to express their appreciation to Professors Akira Sakaguchi and Yasuhiko Ogąwa for their discussion, and to all the other participants. Finally, this paper has been greatly improved by the insightful comments of Professor James W. Higgins of the University of Tsukuba.

References

[1] Kleinrock, L., Principles and Lessons in Packet Communication, *Proc. of the IEEE*, 66, 1320-1329, 1978.

[2] Zafiropulo, P., Performance Evaluation of Reliability Improvement Techniques for Single-Loop Communication System, *IEEE Trans. on Comm.*, COM-22, 742-751, 1974.

[3] Uyetani, A., Local Computer Networks, (in Japanese), *IECE*, 62, 1310-1316, 1979.

[4] Wilkes, M.V. and Wheeler, D.J., The Cambridge Digital Communication Ring, *Local Area Comm.*

Network Symp., Mitre Corp. and National Bureau of Standard, 1979.

[5] Wilkes, M.V., The Impact of Wide-Band Local Area Communication Systems on Distributed Computing, *COMPUTER*, 22-25, 1980.

[6] Metcalfe, R. and Boggs, D., Ethernet: Distributed packet switching for local computer networks, *Comm. of the ACM*, 19, 395-404, 1976.

[7] Nakayama, K. et al., On Line Retrieval at University of Tsukuba, *Proc. of the 3rd International Online Conf.*, 193-208, 1979.

[8] Ikeda, K. et al., Computer Network Coupled by 100 MBPS Optical Fiber Ring Bus -System Planning and Ring Bus Subsystem Description- , *COMPCON 80 Fall*, 159-165, 1980.

[9] Kobayashi, K., Computer, Communications and Man: The Integration of Computer Communications with Man as an Axis, *Computer Networks*, 237-250, 1981.

[10] Tanaka, H. and Moto-oka, T., Distributed File Management and Job Management of Network-Oriented Operating System, *JIPS*, 4, 18-25, 1981.

[11] Spector, A.Z., Performing Remote Operations Efficiently on a Local Computer Network, *Comm. of the ACM*, 25, 246-260, 1982.

[12] Rowe, L.A. and Birman, K.P., A Local Network Based on the UNIX Operating System, *IEEE Trans. on Soft. Eng.*, SE-8, 137-146, 1982.

[13] Wilkes, M.V., Communication using a digital ring, *PACNET Conf.* , 47-55, 1975.

[14] Pierce, J.R., Network for block switches of data, *Bell Syst. Tech. J.* , 51, 1133-1145, 1972.

[15] Lui, M.T. and Reames, C.C., Message communication protocol and operating system design for the Distributed Loop Computer Network (DLCN), *Proc. of the 4th Annual Symp. on Computer Architecture*, 193-200, 1977.

[16] Kawai, H. and Ebihara, Y. et al., Front End Processing User Protocol with Telephone Network, *ICCC-78*, 663-668, 1978.

[17] Toda, I., DCNA Higher Level Protocols, *IEEE Trans. on Comm.*, COM-28, 575-583, 1980.

[18] Walden, D., A System for Network, *Comm. of the ACM*, 15, 221-230, 1972.

[19] Chow, Y.C. and Kohler, W.H., Models for Dynamic Load Balancing in a Heterogeneous Multiple Processor System, *IEEE Trans. on Computers*, C-28, 354-361, 1979.

[20] Hwang, K. et al., A Unix-Based Local Computer Network with Load Balancing, *COMPUTER*, 55-66, 1978.

[21] Ebihara, Y., Ikeda, K. et al., Fault Diagnosis and Automatic Reconfiguration for a Ring Bus Subsystem, *submitted to Computer Networks*, 1983.

[22] Ishizaka, M. et al., A Link Level Protocol and its implementation in a Ring Network, *Proc. of the 6th Conf. on Local Computer Networks*, 43-51, 1981.

[23] Kuo, F. F., Design Issues for High Speed Local Network Protocols, *New Advances in Distributed Computer Systems*, 97-105, 1982.

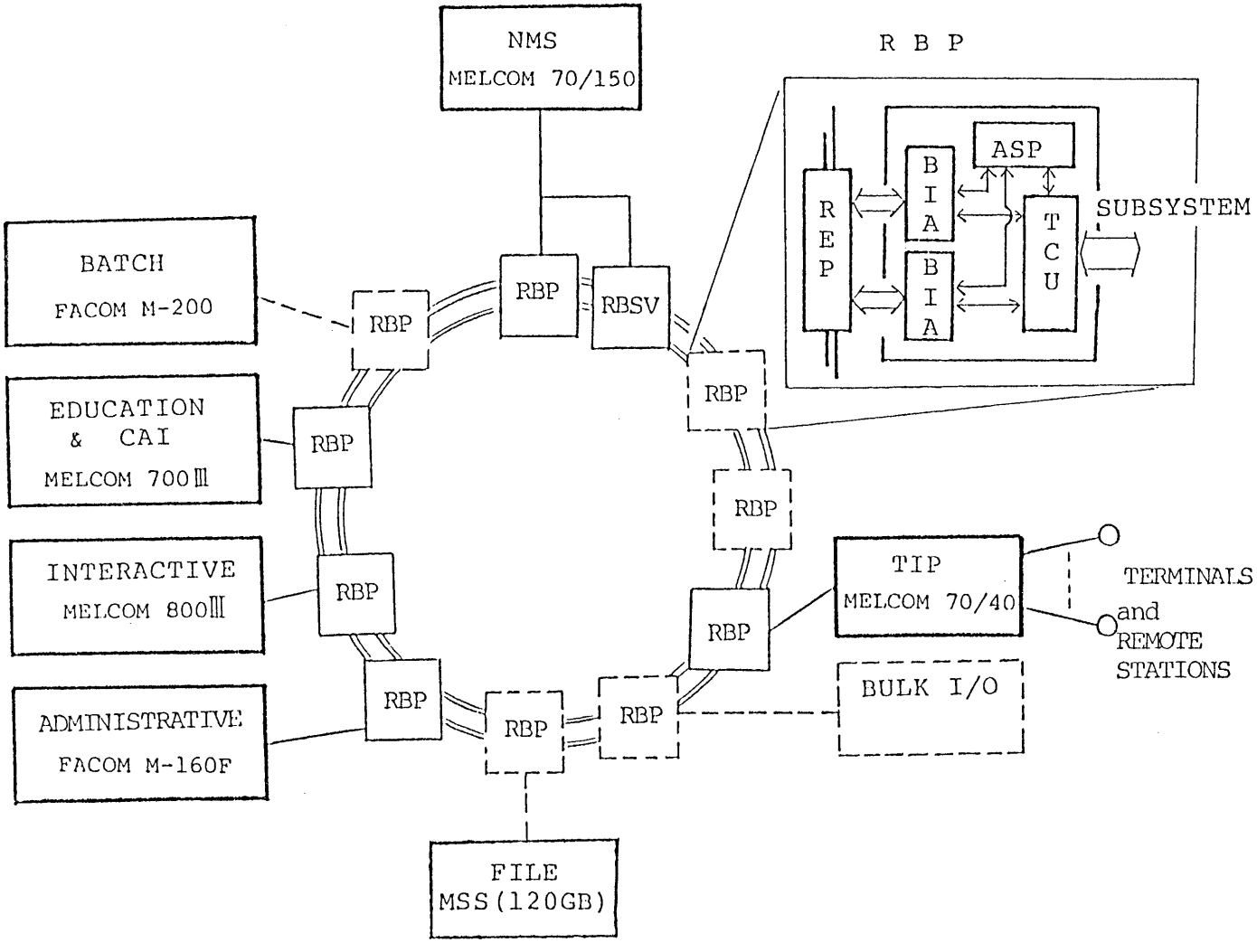


Fig.1 System configuration

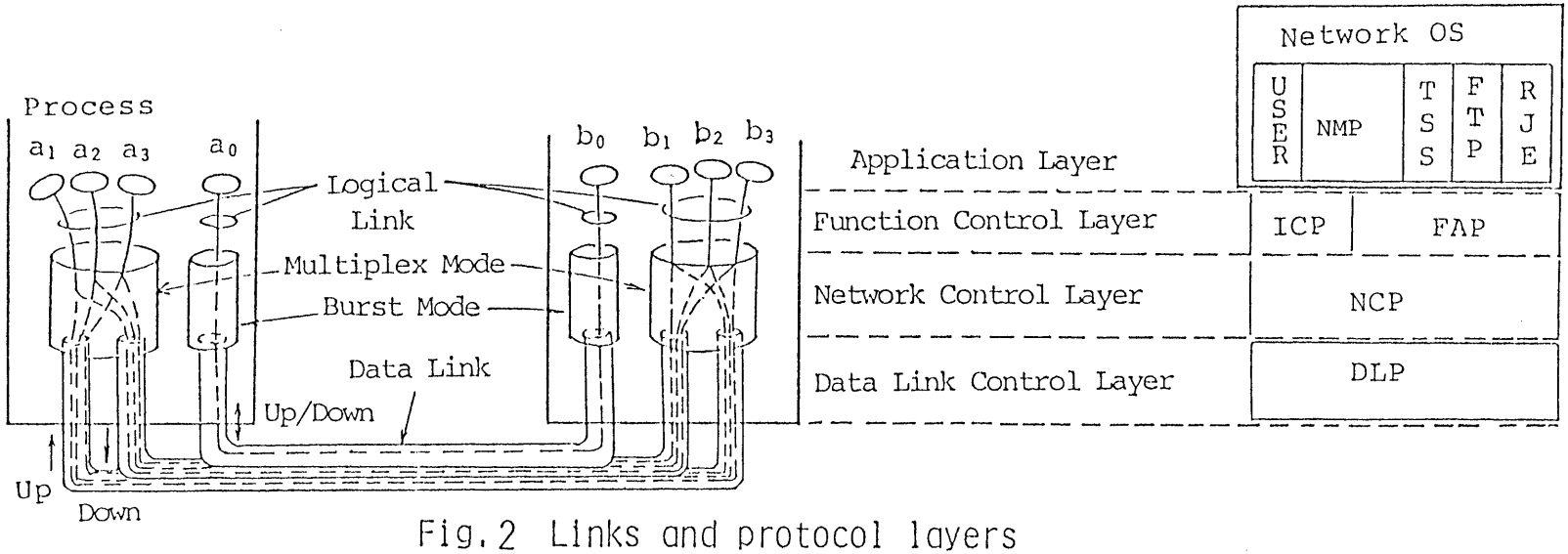
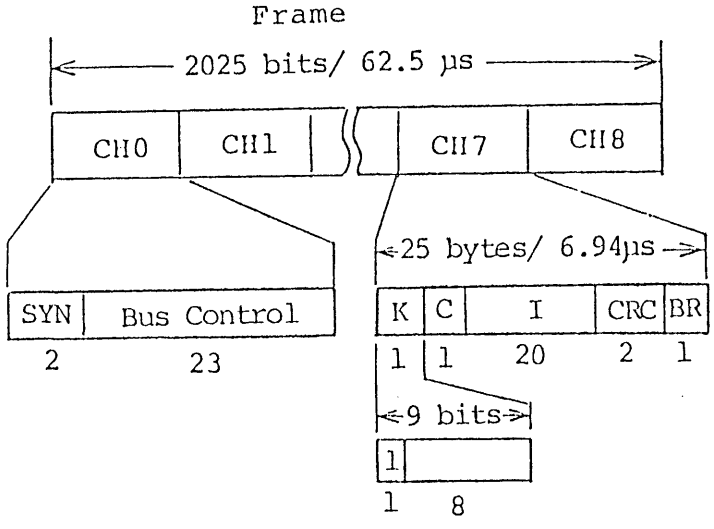


Fig. 2 Links and protocol layers



SYN : synchronous word
K : channel access key
C : command
I : information
CRC : cyclic redundancy check
BR : busy/response

Fig.3 Frame and packet structure

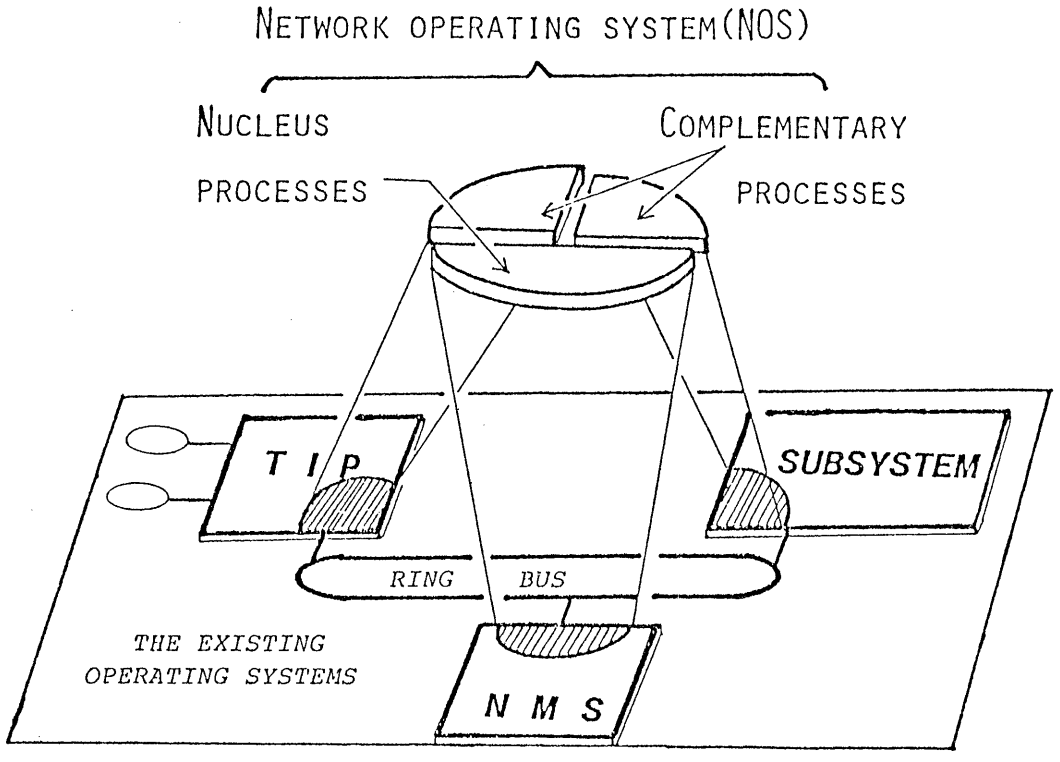


Fig. 4 Structure of network operating system

	A data link to logical links	Acknowledg-ment message	Mode	Application field	Flow control	Priority schedule
Burst mode	1 to 1	None	Half duplex with a data link	Large data; file transfer	None	None
Multiplex mode	1 to n Max. n=4096	None	Full duplex with a pair of data links	Short data; TSS,RJE	Option	High/low data message priority

Table 1 Characteristic comparison of burst/multiplex mode

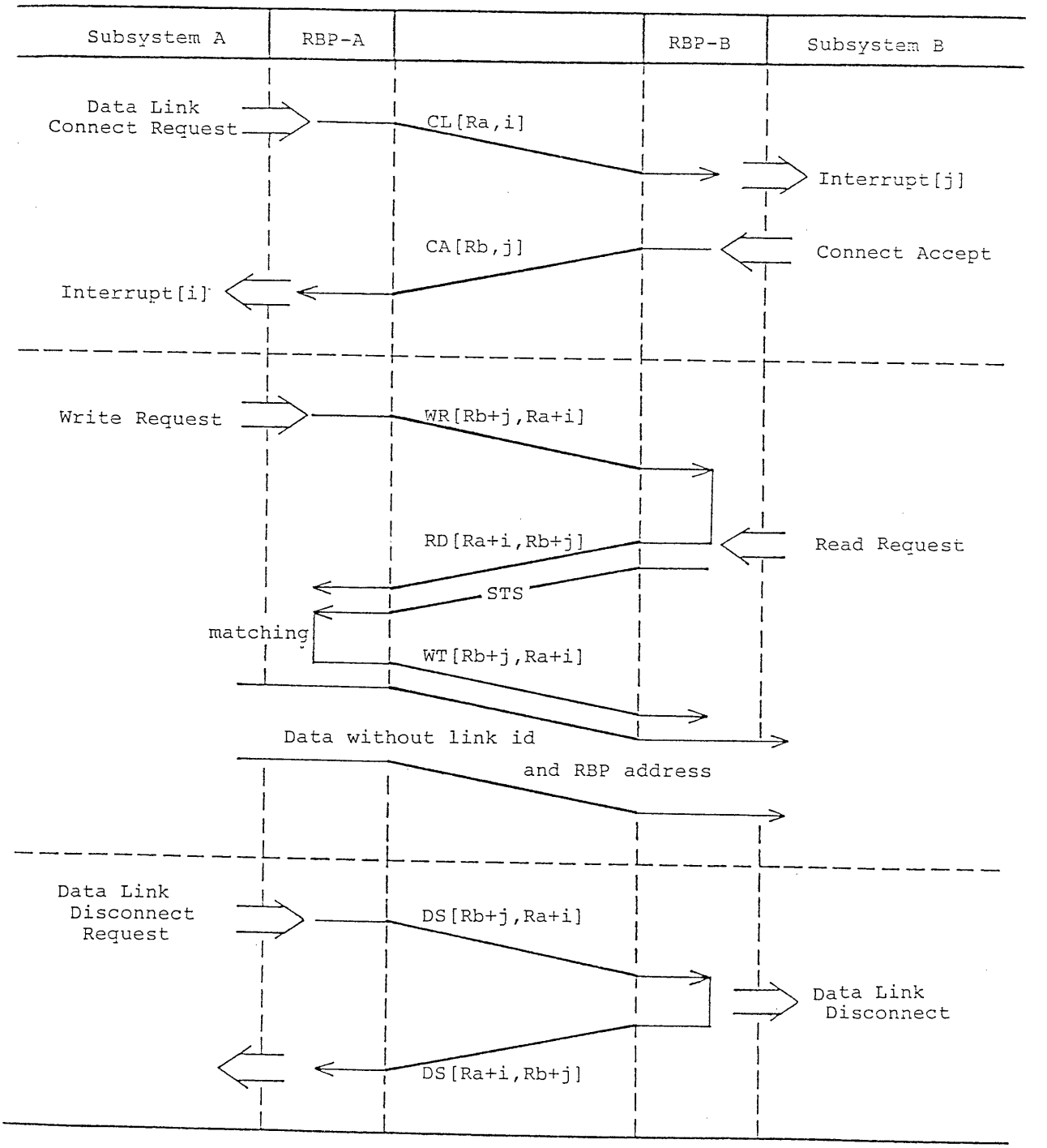


FIG. 5 DATA LINK CONTROL FLOW

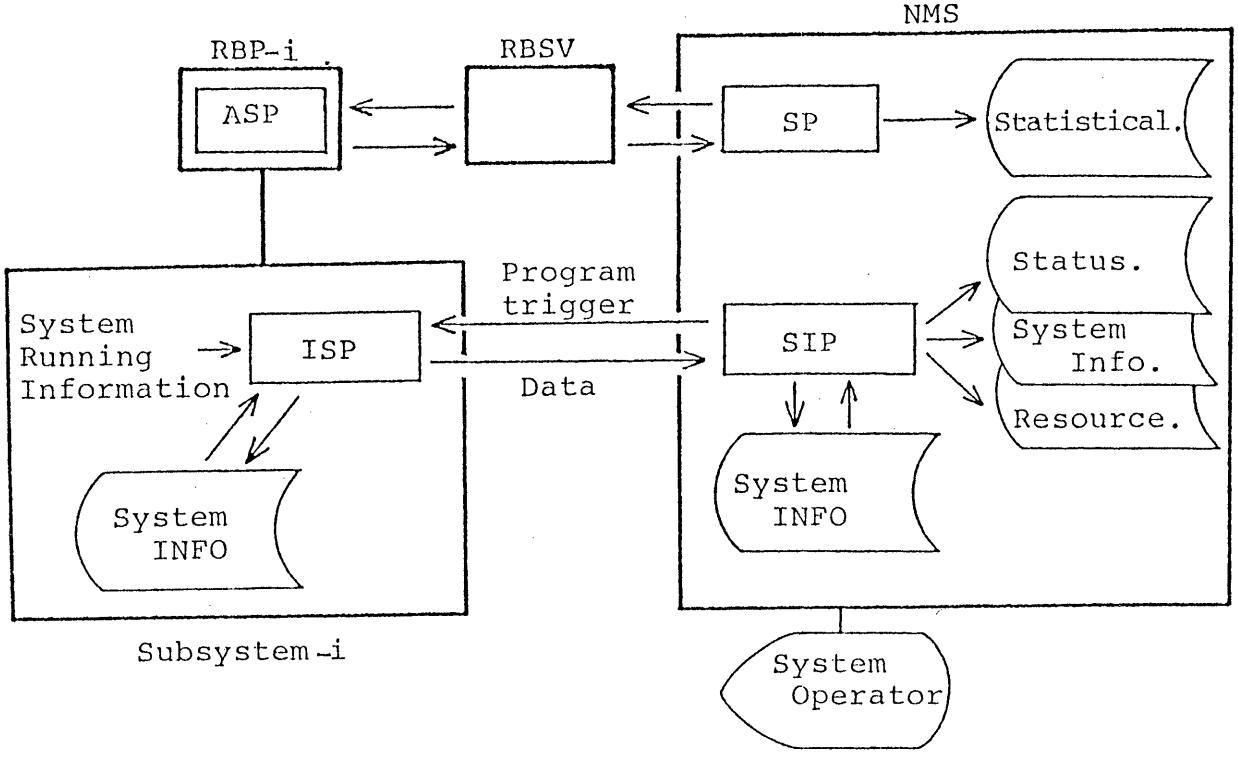


Fig.6 Collecting mechanism of the system information

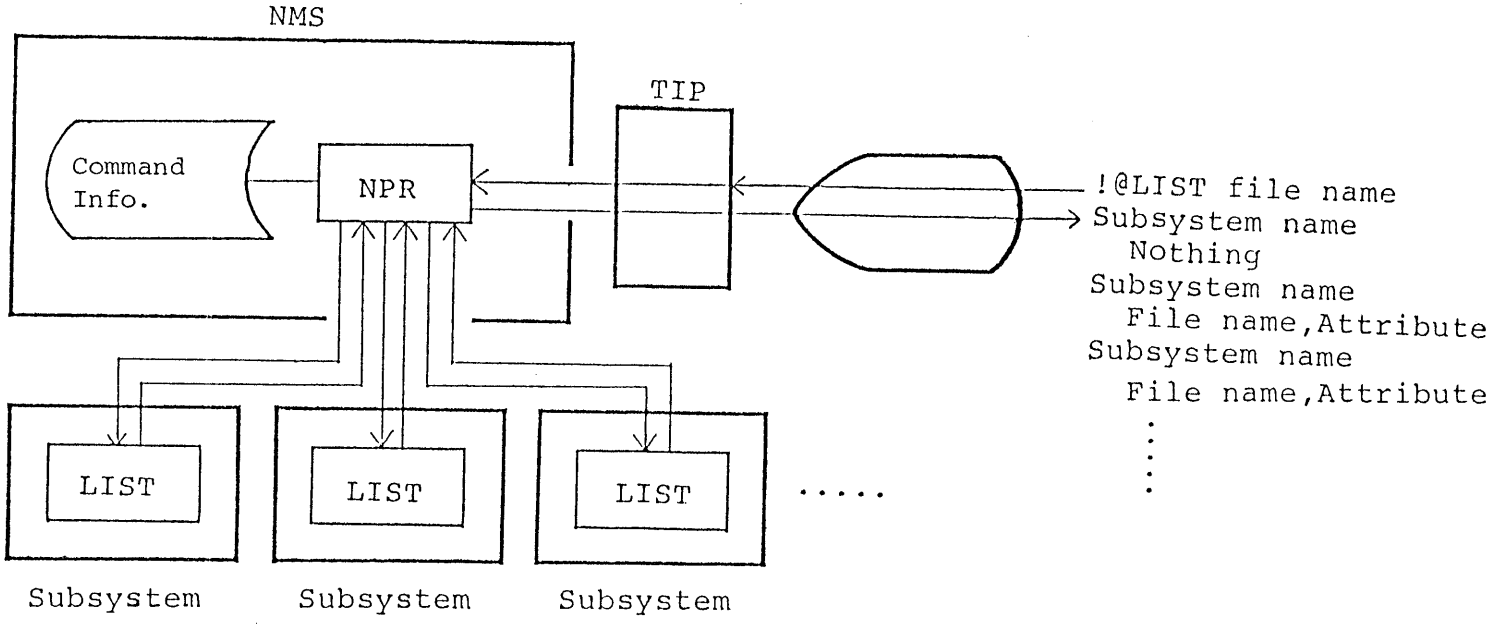


Fig. 7 Automatic file search control

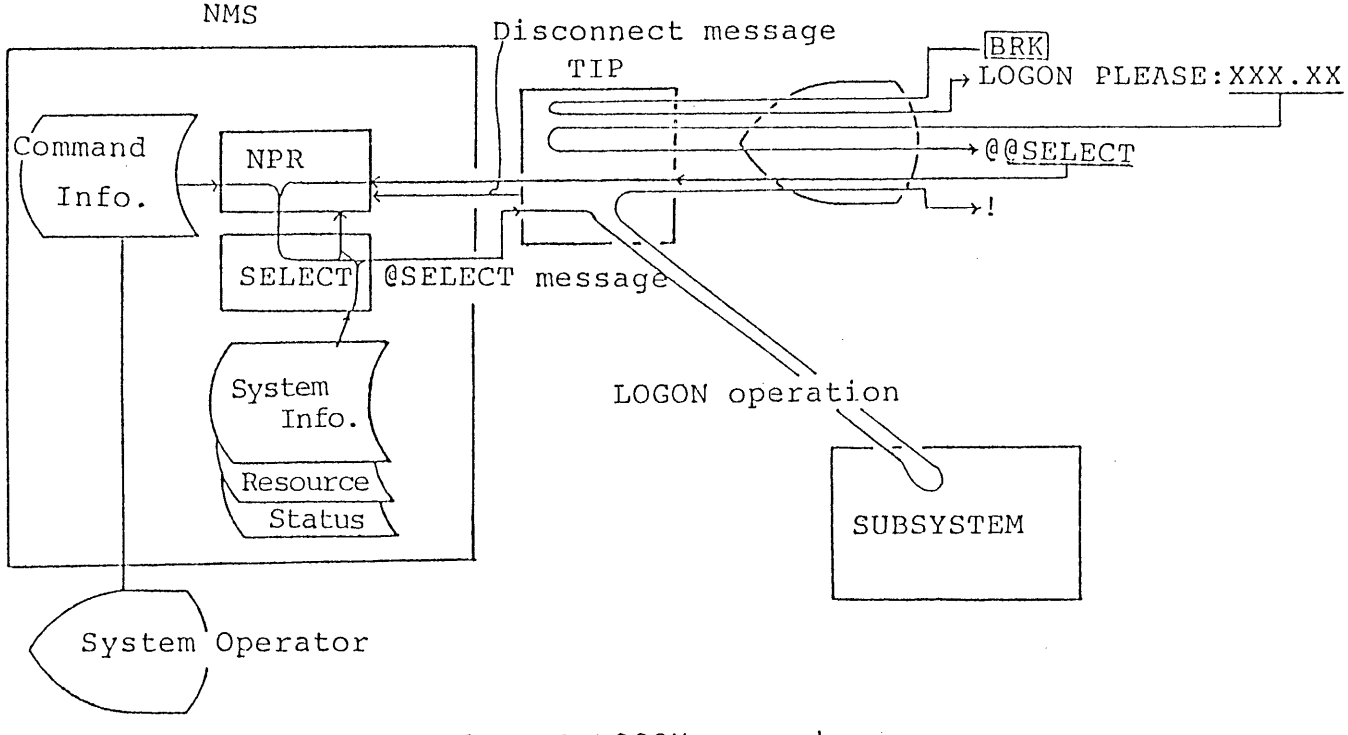


Fig. 8 TSS LOGON procedure

INSTITUTE OF INFORMATION SCIENCES AND ELECTRONICS
UNIVERSITY OF TSUKUBA
SAKURA-MURA, NIIHARI-GUN, IBARAKI 305 JAPAN

REPORT DOCUMENTATION PAGE	REPORT NUMBER ISE-TR-83-35

TITLE GAMMA-NET: A Local Computer Network Coupled by a High Speed Optical Fiber Ring Bus - System Concept and Structure -

AUTHOR(S) Yoshihiko Ebihara *
Katsuo Ikeda *
Tomoo Nakamura *
Kazuhiko Nakayama *
Michihiro Ishizaka **
Makoto Shinzawa **

* Inst. of Information Sciences and Electronics, Univ. of Tsukuba.
** Mitsubishi Electric Co.

REPORT DATE	NUMBER OF PAGES
April 14, 1983	26
MAIN CATEGORY	CR CATEGORIES
Communications	3.81

KEY WORDS

Computer network, Network operating system, Optical fiber, Ring bus.

ABSTRACT

The concept and structure of a local computer network are presented for a full scale high performance computer network coupled by an optical fiber ring bus. The design goals of the system are efficient resource sharing and improved RAS. Implementation issues of the system are also discussed, focusing on the optical fiber ring bus and the network operating system services.

SUPPLEMENTARY NOTES