APPROVED FOR PUBLICATION

AAEC/TM576

2 [illegible]

AAEC/TM576

AUSTRALIAN ATOMIC ENERGY COMMISSION
RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

THE A.A.E.C. COMPUTER NETWORK

by

D.J. RICHARDSON

November 1970

APPROVED FOR PUBLICATION

ISBN 0 642 99387 4

AUSTRALIAN ATOMIC ENERGY COMMISSION
RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

THE A.A.E.C. COMPUTER NETWORK

by

D. J. RICHARDSON

ABSTRACT

The central computing system at the Australian Atomic Energy Commission's Research Establishment is currently based on an IBM360 model 50 computer. Twelve smaller digital computers and one analogue computer are in use at various points within a half mile radius of the central computer installation.

This paper describes plans to link a number of these computers into a network connected to the central computer, and outlines the philosophy behind the setting up of this network.

Note: This paper has been submitted to a journal. Further details can be obtained from the author or the Director of the Research Establishment.

National Library of Australia card number and ISBN 0 642 99387 4

CONTENTS

Page

1. INTRODUCTION

The central computing system at the Australian Atomic Energy Commission is currently based on an IBM360 model 50 computer with 512K bytes of core storage, one multiplexor channel, and two selector channels. Standard I/O equipment consists of a model 2540 card reader/punch, a model 2671 paper tape reader, and a model 1403 line printer attached to the multiplexor channel. A 2314 disk drive facility is attached to the first selector channel, and two 2400 series 9 track magnetic tape units are attached to the second selector channel.

Scientific computations form 90 percent of the central computer's work-load; the remaining 10 percent are commercial type applications. Two distinct types of job occur in the scientific area. By far the larger number of these are short running, of from five to ten minutes average duration, written in Fortran IV, and needing perhaps 100K of core storage. These are written by physicists, engineers and chemists within the Research Establishment to aid them in the processing and interpreting of experimental and theoretical data. The remaining scientific jobs are based on very large programs, called nuclear codes, which, because of their size and the time taken to write them, are usually shared between Atomic Energy Establishments in different countries. These jobs are fewer in number, but run for from one to fifteen hours on the central computer, and require most of the available core storage. They are mostly CPU bound jobs, and are used in reactor design calculations. Approximately 50 percent of the central computer's time is spent in the running of these large nuclear code jobs.

In addition to the central computer, a number of smaller digital computers are in use at the Research Establishment – all within a half mile radius of the central computer installation. Some of these small computers are used for the control of experiments, and for the collection and editing of data from such experiments. Other small computers are used to provide conversational Fortran-type facilities to teletype terminal users. These conversational Fortran-type facilities fill a need between that of desk calculation usage and central computer usage. At present, the size of program that they can run is limited to about 50 statements.

An EAI690 hybrid analogue-digital computer, which includes an EAI640 digital computer and a medium sized analogue computer with 30 integrators, is used for simulation studies and the analysis of reactor systems.

At present, those computers which control experiments use paper tape for data output, and this paper tape output is later submitted to the central computer for further detailed analysis.

With this background in mind, the possibility of linking the various digital computers together into some form of network was considered. The following facilities were sought:

1. Each computer in the network should be able to call upon the services of the central computer for the direct, detailed analysis of edited output from the experiments under its control. This would eliminate the handling of paper tape output, leading to faster, more reliable data processing.
2. Each computer in the network should be able to access libraries of their own programs, stored in the central computing system.

It was envisaged, for example, that a program in one of the small computers controlling an experiment would be able to cope with the normal running of that experiment and also, within its storage limitations, with a number of foreseeable unusual eventualities. There would, however, be certain foreseeable eventualities that the program could detect but, because of space limitations, could not itself cope with. In this event, the currently running program would select and summon a suitable replacement program, handle the eventuality, and then recall the original program, replacing all or part of its own core with these programs as the need arose.

3. The conversational Fortran facilities available in some of the small computers should be made available at other points in the network, and should be extended to include access to similar but more extensive facilities, such as CALL360, on the central computer.

4. Facilities for remote job entry to the central computer should be available through the network. This would probably involve the provision of card readers and printers for certain computers within the network.

5. Nuclear codes and other long-running central computer programs should be able to be monitored through the network. Interactive displays would be used for this purpose.

6. All of the above activities should be able to occur with the minimum of wait time, and without interference to each other.

2. NETWORK DESIGN CONSIDERATIONS

Various forms of network were considered in seeking to meet the above requirements.

Because of the size and complexity of the IBM360 operating system it was thought desirable that the network should appear to the IBM360 as a standard extension of the IBM360 hardware, able to be specified at operating system generation time. This was to minimize the need for special systems programming within the operating system, and to enable users of the central facility to send and access data over the network using the standard IBM job control language.

To meet this requirement, a PDP9L computer was chosen to act as a buffer and control computer between the IBM360 computer and the desired computer network, and a generalized computer to computer interface between the IBM360 computer and the PDP9L computer was designed (Richardson, 1969). This interface enables up to 128 different devices or destinations to be accessed through the PDP9L. Each device appears as a standard IBM device to the IBM360 computer.

Installation of the PDP9L computer and the interface to the IBM360 computer was completed in December 1969. In addition to this interface, two 7-track magnetic tape units, a CALCOMP plotter, a card reader, a line printer, an ASR33 teletype unit and a fast paper tape punch were attached to the PDP9L computer. These devices are currently available as additional I/O devices to the IBM360 computer, using the interface to the PDP9L.

With the PDP9L able to act as a buffer between the IBM360 and the other site computers, the main constraint on the form of linkage of these computers was removed, the PDP9L being able to match the I/O requirements of the IBM360 to any reasonably chosen network signalling system in the remainder of the network.

The following questions arose:

1. Should data transfers within the network be synchronous or asynchronous?
2. Would 2400 baud telephone lines suffice, or should higher capacity private lines be used?
3. What error detection facilities would be needed? Should automatic error correction as well as automatic error detection be included in the network facilities?
4. How closely knit and interdependent should the network be? How free would each individual computer within the network be to pursue its own interests and ignore the network?

With all computers situated within a half mile radius of the central computer installation, signal propagation time was not a significant factor. Also, since the cycle times of individual computers differed, synchronous data transfers would require extensive buffering or matching to the slower computer in each case. For these reasons, asynchronous transfer of data, on a call and answer basis, was chosen.

Government regulations forbad the use of the local PABX network. As the major cost of the cabling thus required was in the laying of the cables, high capacity lines were chosen. Three other factors also influenced this choice. If one of the network computers required a change of program to cope with some unusual circumstance in controlling an experiment, the half-minute delay if 2400 baud lines were used could be critical. Also, if low capacity lines were used, time-slicing a number of teletype users with other blocked data transfers over the network would result in unacceptable teletype response times for the teletype users. A final point was that any interactive display would be severely restricted in its responses without high capacity network communication.

For error handling, parity checking on each byte transferred over the network, with the detection and notification of single bit errors to each computer involved, was considered sufficient. Users desiring greater protection would be able to include checksums within their data blocks.

In choosing the actual form of linking for the various computers, a prime consideration was to avoid the possibility of a local computer breakdown affecting the entire network. A corollary of this consideration was that any computer should be able to decide not to use the network, and that failure to respond to signals from the remaining computers within the network should be readily detected by them and should not affect their own inter-communications.

To satisfy each of the above requirements, a communication path independent of the computers to be attached to it was devised (Ellis, 1970). This communication path, or Data Way as it has been called, is described in the following section.

3. THE NETWORK DATA WAY

The network Data Way design consists of a controller and a multiwire cable having various entry points for the attachment to it of computers and certain specialized peripheral devices. A 24 bit parallel data bus is proposed with a parity bit for each eight bit byte. Five pairs of symmetrical, asynchronous control lines govern the data flow and status signalling over the Data Way.

Two types of control unit may be attached to the Data Way, an open and a restricted type. An open type control unit is one that will accept all possible Data Way signals, and this type will be used for the attachment of the normal class of small digital computer. Restricted type control units will be used for the attachment of single byte operating devices such as teletypes and paper tape equipment to the Data Way.

The Data Way will operate as a party line. Any two digital computers attached to open type control units on the Data Way may communicate with each other if the Data Way is not currently in use. A number of fixed signalling sequences are available. When one control unit wishes to communicate with another control unit on the Data Way, an Initial Selection Sequence is entered. A 24 bit address and function word is presented to the Data Way. This address and function word contains the desired 8 bit control unit address, the function to be performed (8 bits), and, if the addressed control unit is an open type control unit, the 8 bit address of the calling control unit.

The eight command bits specifying the function to be performed are:

Bit 0	Unassigned
1	Sense
2 & 3	Control
2	Write
3	Read
4	Variation
5	Variation
6	Variation
7	Variation

These bits occupy bits 8–15 of the address and function word.

A restricted control will be able to communicate with only one nominated control unit. The return address therefore need not be sent to restricted control units, and bits 16–23 of address and function words sent to these units will be used to contain immediate data.

If the Data Way is not busy, and correct responses are received from the called control unit, a Data Transfer Sequence may follow the Initial Selection Sequence. This will occur if the requested command was not an immediate command, and both control units desire to send or receive data. Otherwise, an Ending Sequence will follow immediately. Data will be transferred on a call and answer basis, and transfer will therefore be asynchronous. Sustained data transfer rates of the order of 50,000 bytes per second appear feasible.

Either control unit may terminate the data transfer, and an Ending Sequence will then follow. Ending status will be available to each control unit as an eight bit byte, right justified within a 24 bit Data Way word. The proposed bit assignments for this status byte are:–

Bit	
0	Attention
1	Status Modifier
2	Unassigned
3	Busy
4	Unassigned
5	Device End
6	Device Error
7	Data Way Error.

Status Modifier will be used by those control units unable to provide current status on request.

If the Data Way is busy, an indication of this will be returned to the originating control unit. Subsequently, the original control unit will be informed when the Data Way ceases to be busy, and another Initial Selection Sequence may then be attempted.

If the Data Way is available, but no response is received from the requested control unit, the Data Way controller will issue a time-out disconnect after approximately 1 msec. Any computer may therefore ignore the network without unduly impeding other Data Way communications.

Data transfers into and out of each digital computer on the Data Way will use asynchronous data channels whenever possible, otherwise program controlled data transfers will need to be used.

Problems of simultaneity arise when more than one control unit attempts an Initial Selection Sequence at the same time. Two methods of detecting this case will be used. First, if the OR-ed address selections have incorrect parity, both selections will be rejected by the Data Way controller, and each control unit must repeat its initial selection sequence. Second, initiating control units will check the address on the bus lines with the address that they selected before continuing with the Initial Selection Sequence. Once a valid address selection has been made, all other control units are locked out until the Ending Sequence has completed.

4. IMPLEMENTATION OF THE NETWORK

It is planned to implement the network in a number of stages. The first two such stages are illustrated in Figures 1 and 2.

In Stage 1, a small NOVA computer will be connected to the central facility's PDP9L computer. This NOVA computer contains 12K 16 bit words of storage and has five ASR33 teletype terminals attached to it. It is currently being used to develop a multi-terminal Fortran-like conversational compiler for site users (Sanger 1970), which will replace a smaller, single terminal facility at present available on 4K PDP8 computers.

The NOVA computer was chosen as the first stage Data Way connection to the PDP9L computer because of its proximity to this computer. This will allow Data Way concepts to be tried out without waiting for or relying on any underground cabling.

The NOVA computer link will be used to send NOVA programs from the IBM360 computer direct to the NOVA computer, using the IBM360-PDP9L link and the Data Way. These programs are at present compiled on the IBM360, sent to the PDP9L, punched out on the fast paper tape punch, and then manually fed into the NOVA computer through one of its ASR33 teletype stations.

During Stage 2 of the network, four of the five NOVA teletype units will be transferred to restricted control units on the Data Way. This will enable the party line concepts of the Data Way to be checked out, and will also provide more widely available access to the NOVA's conversational compiler facility.

Three other presently independent systems will then be connected to the Data Way, using open-type control units. These systems are:

1. A PDP15–PDP7 complex, being used to monitor and gather data from a linear accelerator system.
2. A PDP8 being used to control a crystal diffractometer, and
3. An EAI690 analogue-digital computing system being used for reactor design studies and systems simulation.

The PDP15-PDP7 complex will use the network to send edited data to the central computing facility for detailed analysis by pre-stored IBM360 programs. Later uses of the network by this complex will include remote job entry for the IBM360 from a card reader and printer to be attached to the PDP15, and interactive monitoring of long running IBM360 jobs with display and light pen equipment.

The crystal diffractometer PDP8 will use the network to provide access to libraries of diffractometer control programs. The NOVA's conversational compiler facility may also be used by simulating a NOVA terminal from the PDP8 teletype unit.

The EAI690 system will link to the network through its EAI640 digital computer. The main use of the network will likewise be to access libraries of programs stored on the IBM360 disks. Another network use being considered is to allow the IBM360 to undertake digital calculations for the analogue computer during simulation studies when such calculations prove too complex for the EAI640 computer.

With many control units seeking to use the Data Way at the same time, questions of priorities may need to be resolved. When the Data Way becomes available after having been busy, all control units will be notified together. It is hoped that inherent differences in interrupt response times within programmable computers will minimise the resultant conflict. A pleasing feature of the Data Way is the inherently faster response times of restricted control units under these conditions, giving automatic priority to teletype and other unit byte peripheral usage. It may be necessary to incorporate differing hardware delays within restricted control units to sequence their individual re-selection attempts.

5. COMMUNICATION WITH THE CENTRAL COMPUTING FACILITY

The link between the IBM360 and the PDP9L computers permits up to 128 different devices and destinations to be accessed through the PDP9L by the IBM360 computer.

A uniform addressing scheme has been adopted for use throughout the network. Each 8 bit address has the following format:

$$\overset{\overset{\text{Spare}}{\downarrow}}{X} \qquad 1 \qquad \overbrace{X \quad X \quad X}^{\text{Computer}} \qquad \overbrace{X \quad X \quad X}^{\text{Device}}$$

Devices attached to computers on the Data Way will have addresses in which the last three bits are not all zero. In general, computers attached to the Data Way will have addresses with zeros in these last three bit positions, and open control units will ignore the contents of these three bit positions.

Communication initiated by the IBM360 is straightforward from the point of view of the IBM360, since the PDP9L and link ensure that all destinations through the link appear like normal IBM devices to the IBM360 computer. The PDP9L maintains up-to-date sense bytes and status information for each possible link address.

At the present time, the IBM360 can communicate with all PDP9L devices and with the PDP9L core itself. Two part data transfers are used to send PDP9L programs from the IBM360 to the PDP9L core. A Primary Write of 24 bytes is first used to send information about the following program: the program size, location in core and entry address. A simple I/O waiting loop for the PDP9L is included with these 24 bytes. Following the Primary Write, a Secondary Write is used to transmit the actual program to the PDP9L. Distinctive IBM360 write command codes have been allocated to Primary and Secondary Writes.

These concepts will extend readily to communications involving the Data Way. A single group of PDP9L subroutines will handle all Data Way communications, since the only variable in talking to different Data Way open control units is the control unit address itself. The computers associated with Data Way addresses will be responsible for the signalling to the PDP9L of changes in the sense bytes and status data maintained in the PDP9L for these addresses. This will avoid the PDP9L unduly occupying the Data Way in seeking this information.

Communication with the IBM360 initiated from within the network is not quite so straightforward however, since the IBM360 itself must initiate all actual I/O data transfers. IBM360 I/O devices may, however, signal their status asynchronously to the central processing unit and this provision will be used to signal the IBM360 when a point in the network wishes to communicate with the central computer.

Provision exists within the IBM360 operating system to recognize ‘attention’ status signals sent from console typewriters, the operating system then issuing a read command to the signalling console to accept input messages. Similar software sequences will be used to issue a special diagnostic read when an ‘attention’ signal is received over the PDP9L link.

When an open control unit on the Data Way wishes to communicate with the IBM360, a Diagnostic Write command will be issued to the PDP9L computer associated with the IBM360 computer. On receipt of the Diagnostic Write command, which will be identifiable by the presence of a special variation bit in the Data Way write command, the PDP9L will read up to 24 diagnostic bytes from the originating computer into a special PDP9L reception area associated with that originating computer. When convenient to do so, the PDP9L will then signal ‘attention’ over the link to the IBM360 for the address associated with the originating Diagnostic Write command. When the Diagnostic Read is received from the IBM360, exactly 24 bytes of information will be sent to the IBM360, commencing with those bytes received from the originating computer.

If the first diagnostic byte is non-blank, the first eight bytes will indicate the name of a program within a library associated with the corresponding network destination address. This program will then be retrieved and sent to the destination address, using the Primary and Secondary write concepts outlined above. Because of space limitations within the PDP9L computer, the Secondary Write transmissions will be blocked so as not to exceed 800 bytes in any one block.

If the first network diagnostic byte received by the IBM360 is blank, the format of the remaining 23 bytes will depend on prior arrangements with the originating computer. These bytes may, for example, request a specific analysis program and instruct the IBM360 to read a specific number of records or one file of information from some specified network address associated with the originating network computer, and to return output to another specified address. In this way, Diagnostic Writes need not be associated one for one with each record transmitted over the Data Way.

6. NETWORK CONTROL

Whenever the facilities of one of the network computers is to be made available to other network computers, that computer must contain programs able to communicate with the rest of the network. Likewise, when one computer wishes to use facilities available elsewhere in the network, that computer must also contain network communication programs. Care must be taken when programs are transmitted over the network and subsequently activated that this ability to communicate is not inadvertently lost. Each such program must contain or leave unimpaired the ability to communicate with the network.

To the extent that a particular computer within the network dedicates its facilities to the network and is used by the network, that computer may be said to control the network. In this sense, the IBM360-PDP9L complex will exercise considerable control over the network.

It is envisaged that libraries of programs for each of the network computers will be stored within the central computing facility, and that small boot-strap routines, available on each computer, will enable local operators to use their teletypes to select their required programs. Subsequent network usage and the degree of operator intervention required will, of course, depend upon the programs selected.

7. FUTURE EXPANSION

Once Stage 2 of the network has been completed, additional local computer systems will be attached to the Data Way and the computer network as the need arises. Some of these computer systems will contain card readers and printers which will then be available for remote job entry to the IBM360. A display and light pen type facility has been ordered for attachment to the NOVA computer. This will enable visual communication concepts for use on the network to be developed.

In the software area, efficient ways of making the central computing system's facilities available on a conversational basis are being investigated. As a first step, it is hoped to make IBM's CALL 360 system available to network users. Local digital computers would act as line concentrators and editors for their associated teletype or similar units. Communication with the IBM360 and the CALL 360 program would then be on a line by line rather than a character by character basis.

The possibility of remote computer installations wishing to connect to the A.A.E.C. network has been considered. Synchronous data transmission would be needed in this case if a reasonable data transmission rate were to be maintained. This would probably be achieved, if needed, by using small digital computers as buffers, with synchronous block transmission between them.

8. COMPARISON WITH OTHER NETWORKS

Three main forms of multi-computer networks are in use: line switched systems, message switched systems, and fully interconnected systems. Line switched systems are usually noisier than the other two forms, and are not often used in high capacity systems.

Message switched systems include destination information with each block of data transmitted, and these blocks normally have some fixed format. Message blocks may pass through a number of different computers before reaching their ultimate destination, each computer in the chain choosing the most appropriate current path for forwarding the message to the next computer. The Australian Post Office is at present setting up a multi-computer message switched network, to be called the Common User Data Network. Extensive use will be made of disk and magnetic tape storage to cope with temporary delays in the network. Considerable duplication of equipment is being made in the main centres to give high reliability.

The ARPA network in the United States (Roberts and Wessler 1970) is an excellent example of a medium to high capacity, message switched network. The computers in this network are interconnected to give at least two possible paths between every two network points. Message switching is purely core to core with no peripheral storage equipment involved. A 24 bit cyclic checksum is included with each 1000 bit block of data for error detection only.

The proposed A.A.E.C. computer network is a fully interconnected system from the point of view of the Data Way. There is at present no path duplication within the Data Way, the reliability of the system depending upon the reliability of the Data Way controller. Destination information is implicit in the establishment of the desired communication paths, and no restriction other than a maximum blocksize is placed on the form of the data transmitted. The form of the network allows standard software to be used within the central computer for most network communication. This will be a big factor in maintaining compatibility with successive releases of operating system software.

9. CONCLUSION

The proposed A.A.E.C. computer network will allow computers in the network to communicate with each other and to have ready access to the facilities of the IBM360 central computer, with data transfer rates around 50,000 bytes per second. Teletype terminals and other unit byte peripheral devices will connect directly to the network, and interactive conversational compiler facilities will be available at these terminals.

10. REFERENCES

Ellis, P.J. (1970). – A Multiplexed Computer-Computer, Computer-Device Data Link. AAEC/E206.

Richardson, D.J. (1969). – A Generalized Computer to Computer Link for an IBM360 Computer. The Australian Computer Journal Vol. 1, No. 5, November, 1969.

Roberts, L.G. and Wessler, B.D. (1970). – Computer Network Development to Achieve Resource Sharing. Computer Communications and Display Conference, Sydney, 1970.

Sanger, P.L. (1970). – ACL-NOVA: A Multi-user Conversational Compiler System for NOVA Computers. In course of publication.

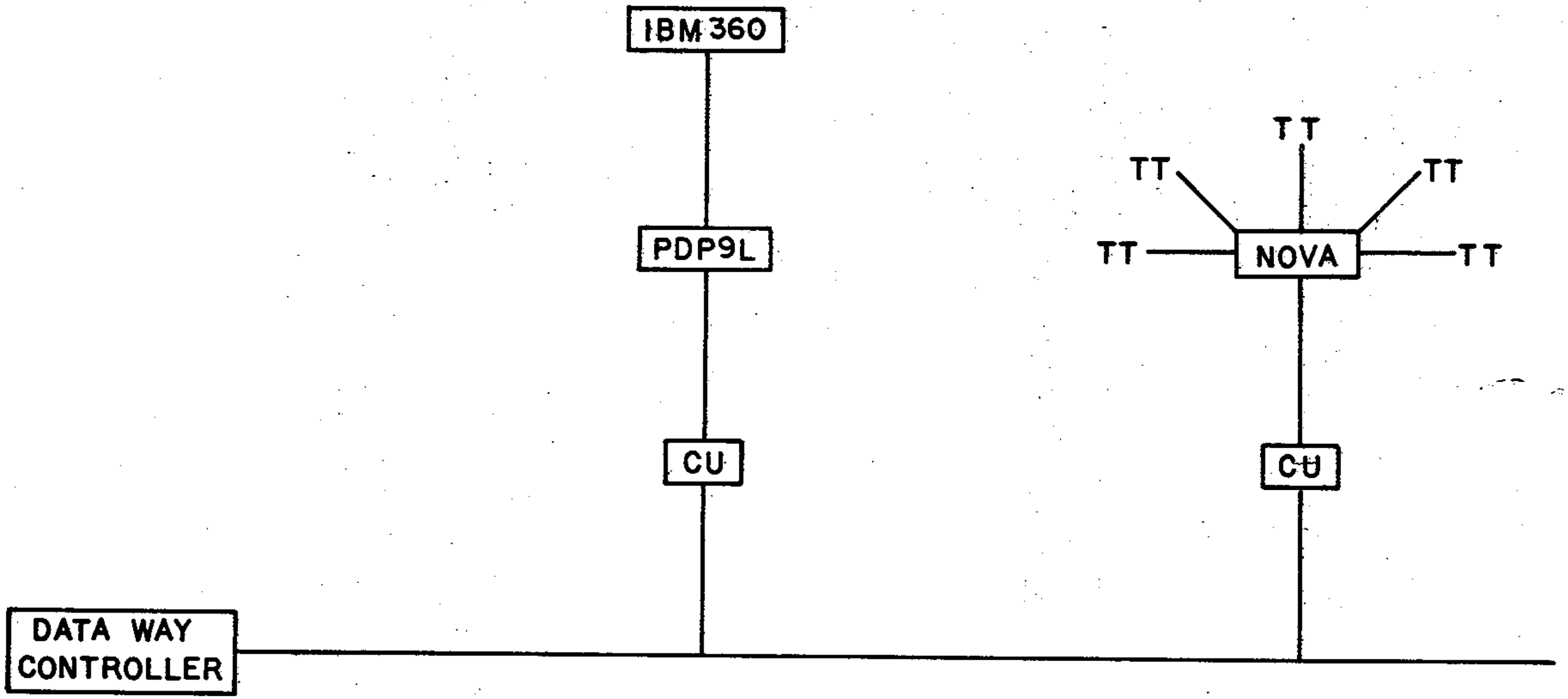

FIGURE 1. IMPLEMENTATION OF THE NETWORK: STAGE 1

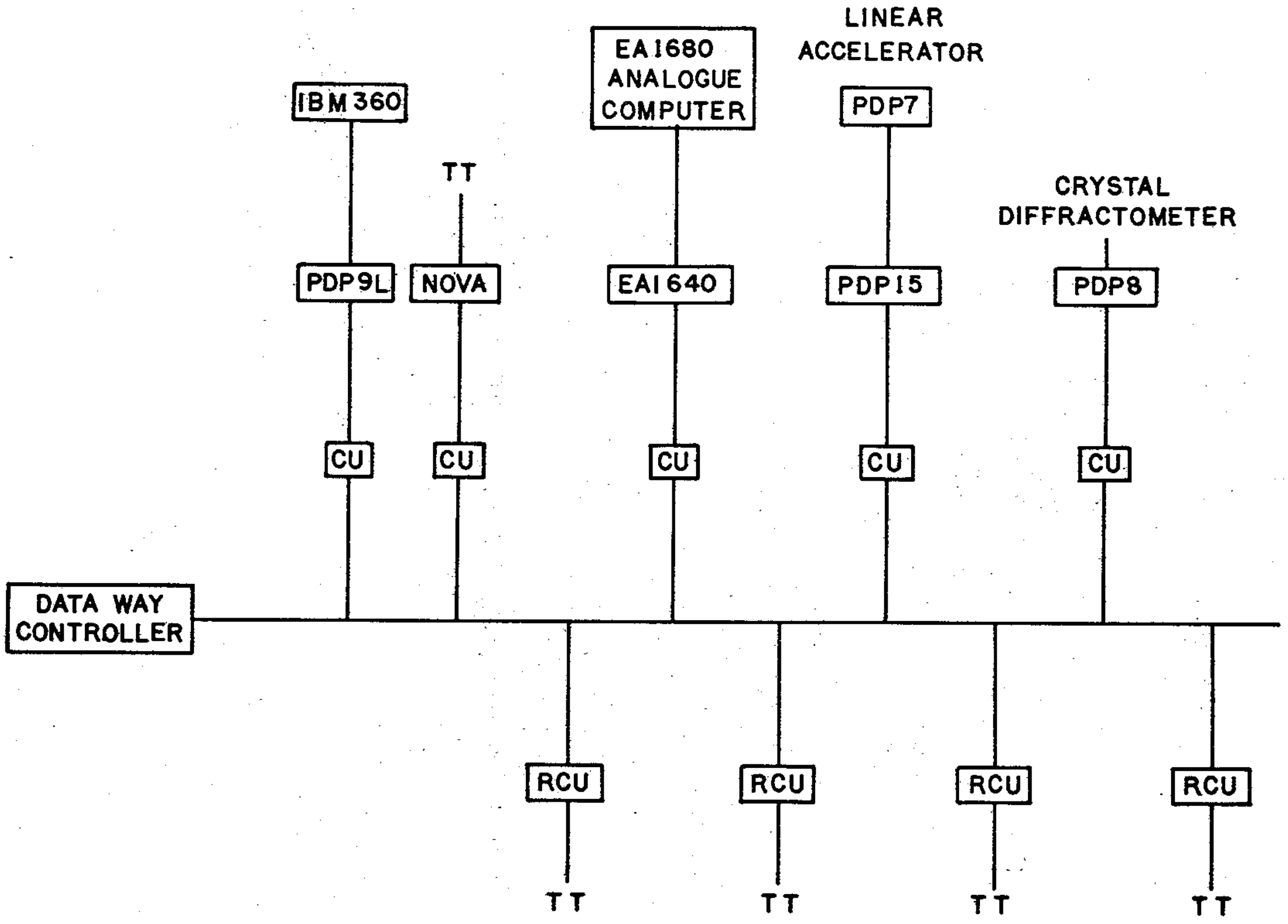

FIGURE 2. IMPLEMENTATION OF THE NETWORK: STAGE 2