The 5th International Symposium on Robotics in Construction June 6-8, 1988 Tokyo, Japan

DEVELOPMENT OF A PROCESS SIMULATION PROGRAM (ORBIC-1), FOR A BUILDING CONSTRUCTION ROBOTICS SYSTEM

A.Sakaguchi*, T.Shiokawa**, K.Hamada,**
Y.Yamawaki*, K.Izumi*, T.Wakisaka**

Ohbayashi Corporation
* System Development and Computing Center
** Institute of Engineering Technology Center
3, 2-chome, Kanda Tsukasa-cho, Chiyoda-ku,
Tokyo 101, Japan

ABSTRACT

A construction process simulation program which is called ORBIC-1, is developed. ORBIC-1 consists of Building Model Data Generation, Construction Process Simulation of Conglomerate Robots, Simulation Result Graphic Display and Simulation Result Analysis subsystems. The results of the simulation process are displayed in 3-D color animation graphics and also analyzed from a construction schematic point of view, and several kinds of plotter drawings of the construction schedule charts are produced.

INTRODUCTION

In constructing a so-called flexible manufacture system (FMS) in the building construction field wherein building construction robotics systems and autoconveying machines are combined and controlled by a computer, it is important to simulate these within the computer beforehand for checking the movement of each building construction robotics system and the process status.

With such a background, as one of the preliminary studies for development of a new construction system assuming robotization of building construction, we have developed a computer simulation program of the building construction process by a robotics system, which is reported hereinafter.

1. System Overview

This system was developed, in preparation for development of building construction robotics systems which will increase from now on, for the purpose of the "establishment of a computer simulation technique for building construction robotics systems" which is an essential technical element for development of building construction robotics systems.

The building construction robotics system process simulation program "ORBIC-1", which was developed this time, automatically simulates the component assembly process when standardized structural modular components (column, girder, slab components, etc.) are assembled at a construction site using various automatic assembling and conveying machines, which are networked according to the execution scheme for a model building.

The primary simulation parameters include the types, numbers and setting positions of the assembly machines, the moving sequences and speed of the machines, the building components' distribution sequence and those feeding interval times, etc.

Based on the given simulation parameters, a component conveying network among the respective automatic assembling and conveying machines is constructed within the program, and a process is simulated in which each building component distributed at an arbitrary time interval is carried through each assembly machine according to the network and located at a predetermined position in the final building.

As a result of the simulation, information such as the operation status of each assembly machine, the movement of the assembly machine, the building components' flow and the operating time is obtained.

The simulation result is displayed as animated 3-D color graphic images for easy verification and examination of the robotics building construction system. In addition, in order to understand the total construction period and examine in detail the operation status of each assembly machine and the operation flow, the process analysis of the simulation result is performed, and various construction progress schedule charts are produced.

2. The System Functions

2.1 Functional Configuration of the System

The system consists of , as shown in Figure 2.1, four subsystems in total, namely, a building model data generation subsystem, a building construction process simulation subsystem, a simulation result graphic display subsystem and a simulation result analysis subsystem. These subsystems are integrated and controlled by a subsystem execution control program and through which the engineer can access to all subsystems assisted with the Japanese-Kanji menu screens.

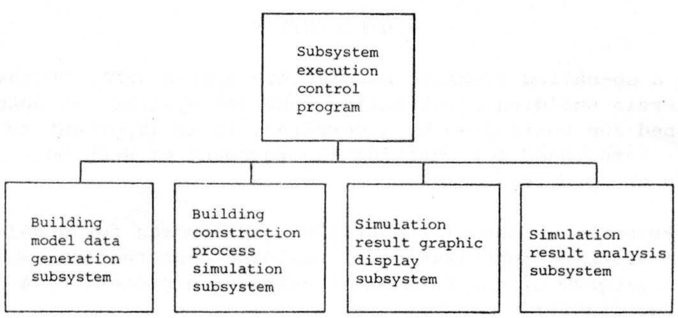


Figure 2.1 Functional Configuration Diagram of the System

2.2 The System Processing

The relation between the primary data and the processing of each subsystem is shown in Figure 2.2.

(Explanation of the Figure)

[Building Model Data Generation Subsystem]

First, based on the building primary shape definition data (number of floors, floor height, span and components' placement data for each floor), the placement data of the building components such as columns, girders and slab components of an arbitrary floor is automatically generated and completed via modification from the graphic display screen.

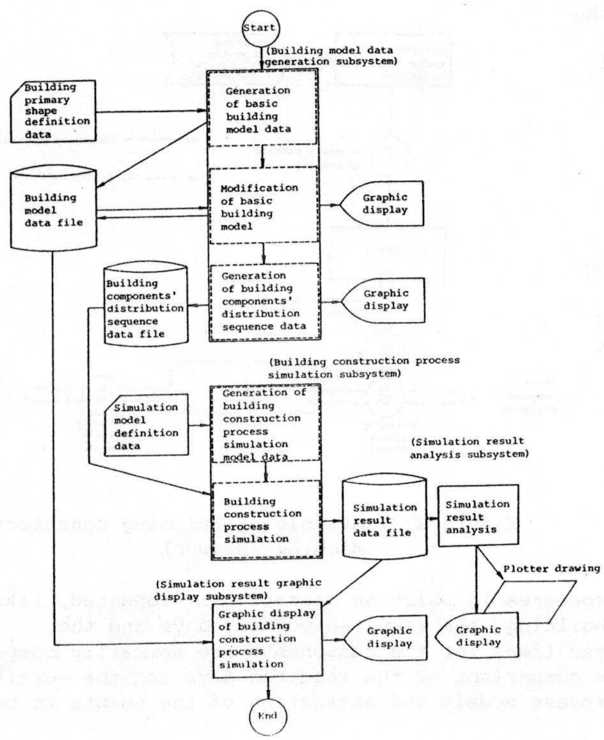


Figure 2.2
Relation between Processing and Data Flow of Building Construction Process Simulation System for Building Construction Robotics System

Then, for simulation, the sequence of distributing the building components (corresponding to the assembly sequence) to the assembly machines and the distribution interval time are specified for the individual building components, creating the building components' distribution sequence data file.

[Building Construction Process Simulation Subsystem]

Based on the simulation model definition data (such as the location of building construction and conveying machines, and the moving of sequence definition data of the machines either for each component type or for particular component), a building construction machines network model is constructed.

By feeding the building components to the constructed simulation model from the above-mentioned building components' distribution sequence data file in accordance with the distribution time interval, possessed by each building component data, the simulation is begun.

The fed building components are conveyed to the final placement location in the building through each machine in accordance with the building construction machines network. At this time, each machine carries a building component in accordance with the aforementioned moving sequence definition data defined for each building component type. The building component is conveyed between the respective machines by synchronizing the times when the machines are ready for passing and receiving the building component.

[Simulation Result Graphic Display Subsystem]

The positional data of all the machines and conveyed components is converted into configuration data at the times of event occurrence in a building construction process and at the further subdivided times between the events. These time-series data are displayed with 3-D color graphics images, providing animated monitoring of the simulation process. Here, checking of the simulation process, examination of the movement of the assembly robots and the like are performed by the functions such as changing the viewpoint for 3-D display of the building model and the assembly machines, stopping the animated screen at an arbitrary time (stop motion), magnifying/reducing an arbitrary portion on the display screen (zooming). In order to notify the occurrence of interference, the system stops the screen temporarily and changes the color tone of interfering components or machines automatically.

[Simulation Result Analysis Subsystem]

In the analysis subsystem, based on the data computed in the simulation subsystem such as the times when the respective machines act, and the times when the components are assembled, information for comparing and evaluating the validity of the assumed process model (such as the component configuration, the assembly sequence, the machines network, and the capability of the machines) is produced and displayed. At the present time, the following three kinds of outputs are produced.

1) Time chart: The types and time history of the operation, the waiting time and the operating time of the machines between the specified times are computed, and from the status of the occurrence of the waiting time and the like by the machines, a clue to the improvement in the movement sequence of the machines, the combination of the machines and the operation speed of the machines is found, in order to optimize the operation.

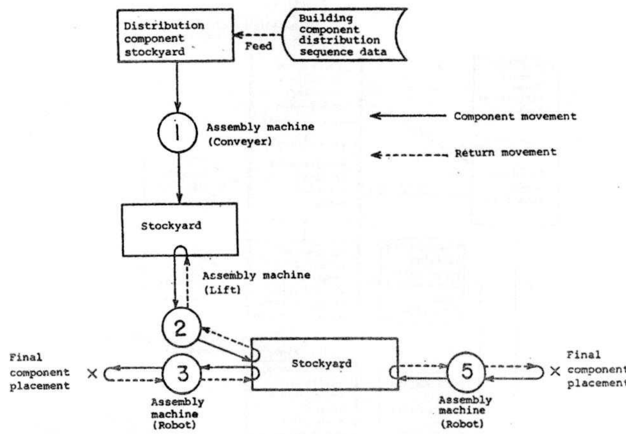


Figure 2.3 Example of Building Construction Machines Network

2) Cycle process chart: For the processes in which an operation is repeated, like the working process of each floor of a building, the required working days and the relations between the required process times for the components are summarily computed and graphically displayed. From the comparison of the required days and the working processes, validation between the process models and extraction of the points to be improved are carried out.

3) Total process chart: Approximate processes of the whole building are computed from the result of 2) and graphically displayed. From the approximate processes, the degree of the effect of each process model on the whole building is provided, and comparison with the conventional building construction process model can be done.

2.3 Program Size

The programs are all written by the FORTRAN language except the partial assembler language portions. The program size is currently total about 18,500 steps for the four subsystems, and about 10,000 steps for the program (display handler[1]) for supporting the development of a graphics processing program, including the interactive graphics processing portion.

The simulation programs basically have no limitations to the number of the assembly machines to be handled, the number of the building components to be distributed, the number of the basic assemblage operations of the machines and the like, due to its incore memory management feature. However, they are constructed so that they can be accomodated by changing a single memory size[2].

3. Simulation Model

This simulation is carried out by the Event-Driven Discrete Deterministic Simulation method.

The feature of this method is that the simulation continues while the event point occurring next is being sought. Also, if the initial condition of the simulation and the input data are known, the simulation result including the simulation process can be determined and reproduced. For this reason, this scheme is considered to be one of the most effective simulation techniques, which can provide more specific and accurate practical estimated data for the attainment of the robotic building construction system in the future.

3.1 Structural Elements of the Simulation Model

The primary structural elements of the simulation model are shown below.

(1) Entities

These are the motive force of this simulation, and the most noteworthy item is the movement of the conveying machines including the building components to be conveyed on the machines network and the assemblage robots.

(2) Attributes

As the parameters representing the attributes of a building component and the movement of a machine, there are machine performance data indicating the operation speed and the operating space ranges of the machine, and data such as the position and the posture of the building component and the machine.

(3) Activities

The elements which control the processing flow of the simulation include, for each event cycle of the simulation, determination of sending and receiving of the building components between the machines, determination of synchronization (or cooperation) between the machines, and the moving vector components' computation for the next travel between current and next moving position, positional computation and interference check computation of the machines.

(4) Event

A change in the system state is caused by a certain machine in operation reaching the next operation change point (called event point). The program proceeds with the simulation while controlling the movement sequence of these events along the right flow of time.

(5) Clocks

In order to conduct the event control, the main clock for controlling the time of the whole simulation and the event clock for indicating the predicted time required to reach the next operation event point for each conveying machine are used in addition to the aforementioned elements. When the time indicated by this event clock becomes zero, it indicates that this particular machine has just reached the next operation event point. The control of this event clock plays an important role in the simulation.

3.2 Processing Flow of the Simulation

The flow of the basic processes of the simulation is shown in Figure 3.1.

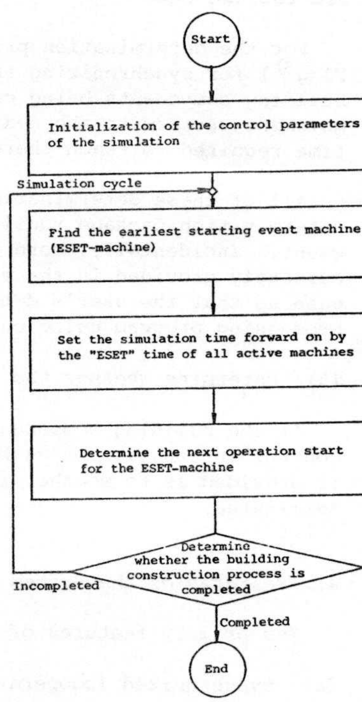


Figure 3.1 Processing Flow of the Simulation

[Description of the Processing]

(1) Initialization of the simulation control parameters

Reset of the main clock and initialization of the simulation control parameters of all the machines on the assembly machines network are executed.

(2) Find the earliest starting event machine (ESET-machine)

Here, among the event clock values (the predicted time values for reaching the next operation event) for the individual machines, the one showing the smallest value is called the Earliest Starting Event Time (ESET), and the machine having this ESET value (called ESET-machine) is found out from the entire system.

However, the machines which are temporarily stopping their operations (i.e. in waiting mode) because of failure of the preliminary condition between the machines related to each other are excluded from this process.

(3) Set the simulation time forward by the amount of the ESET time

The movement of all the machines in active mode in the system are advanced by the ESET value found in process (2).

At the same time, the main clock of the simulation is also counted up by the ESET value. As a result, the current event clock value of the machine found in the processing (2) becomes "0.", and these of the other machines all become "0." or greater. As a result of this process, the ESET-machine has reached the next new operation event point.

(4) Determine the next operation start for the ESET machine

The type of the current event of the machine which was found in (2) is examined, and determination of the next operation start and computation of various activity entries are carried out.

For the determination process, setting of the signal flag (HS flag: Hand-Shake Flag[3]) for synchronizing the machines, determination of receiving and passing of the building components being conveyed on the basis of them, and computations of the movement vectors to the next new operation event point and of the predicted traveling time required to reach there (the event clock value) are performed.

All of these determinations are provided within the "user built-in type determination package routine" corresponding to the type of the particular operation event. Incidentally, more than 40 event determination control function routines are currently provided in the main body of the simulation program for the sake of user's ease so that the user's determination package routine can produce a determination processing program using only these control function routines.

(5) Determine whether the building construction process simulation is completed.

If the building components distribution to all the machines is completed, or if the control instruction of SAVE/RESTART during the simulation is recognized, determination is provided as to whether the next simulation cycle is to be proceeded with or terminated.

4. Features of the System

The primary features of the system will now be described.

(a) Synchronized (cooperative) control between machines

In order to provide the cooperative operation control between machines, the hand-shake scheme generally used in data communication system is used.

(b) Carrying-in and -out controls of a stock-type machine

For providing a sequence of controls "carrying-in -> temporary storage -> carrying-out of the building components" of a stock-type machine (e.g. lift or stockyard) such as collectively conveying a plurality of building components at a time or temporarily storing them, control is provided using queues[4] on an order-of-arrival basis (FIFO: a First-in, First-out).

(c) Interference check between building components and machines

Interference check between the assembly robot's hand in operation and the components being conveyed and the building components already put in place is provided. Also, an interference check is provided between the robots in operation.

As to the method of the interference check, in order to reduce the computation load, check is provided by a simple minimax test as to whether the interference check zones (interference volumes) arbitrarily predefined for the assembly robot's hand and each building component interfere with each other or not, as shown in Figure 4.1.

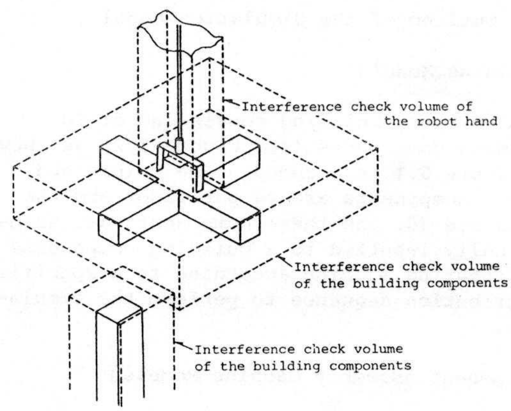


Figure 4.1 Interference Check between Building Components

(d) Temporary SAVE/RESTART of the simulation

The processing can be temporarily saved during the simulation, and the simulation can be restarted on a later date from the point of time when it was saved without disturbing the process flow from the simulation start time.

Particularly, these functions are useful when examining certain building component, trying to find out its adequate moving route, with different movement sequence definition data so as to avoid interference without repeating simulation from the beginning.

(e) Expressing the parameters of the assembly machines

In the simulation subsystem, only the position of a robot hand effector (a device for grasping building components which is attached to the robot hand portion, and operates together with the robot arm) expressed by each assembly machine reference coordinate system is obtained. On the other hand, in the simulation result graphic display subsystem, based on the positional parameters of this robot hand effector, the respective arm joint parameters (amounts of movement and rotation of a joint) are derived within a "user build-in type computing routine" which uses a previously obtained equation for analyzing the robot arm mechanism.

Generally, the expression of the parameters of a robot arm is given by converting all the respective arm shapes into a whole coordinate system using the "A"-matrix[5] which was used in the analyzing mechanism and the joint parameters. However, in this program, using PHIGS[6] as the graphic display software tool, the respective robot arm shapes are expressed by a collective graphics data unit (called structure) which is one of its graphic data organization characteristics. Forming the graphic data unit structures the same as the actual arm connecting structures, the joint parameters of each arm can be directly given to PHIGS, and the coordinate conversion process can be carried out using the arithmetic processing circuit of the graphic terminal side. As a result, the computation load to the host computer can be reduced, and an animated display of the simulation result of each event point can be provided substantially in real time.

5. Applied Example

An example is given wherein a building model consisting of the columns, girders and slabs made of PC as shown in Figure 5.1 was assumed and the simulation was applied to it.

5.1 Outline of the Simulation Model

[Building Model]

A 40-story building consisting of PC building components (see Figure 5.2) as shown in Figure 5.1 is assumed. The column and girder components are 44 per floor and the slabs are 56, and these components are automatically inputted to a building components distributing machine according to a specified distribution sequence to perform the simulation.

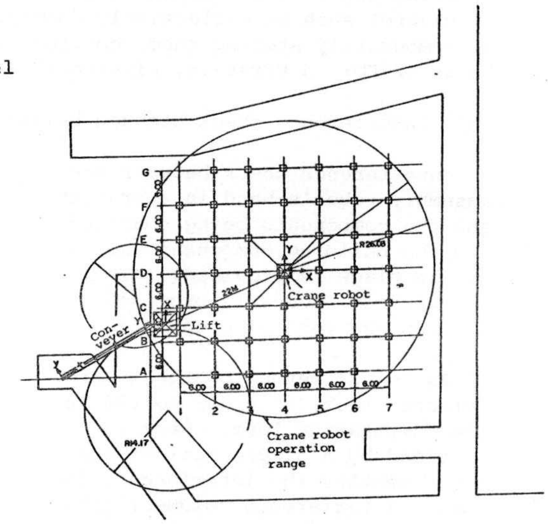


Figure 5.1
Outline of the Building Model and Construction Site

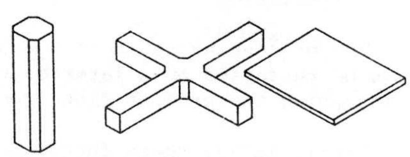
[Component Assembly Machine Models]

The outline of assembly machine models is shown in Table 5.1.

[Assembly Machines Network]

Each machine path, from the distribution (or feeding) of PC components through their final placement location in the building, consists of the following seven machine models.

Building component distributing machine -> Component conveyer -> Ground stockyard -> Component unloading lift -> Floor stockyard -> Crane robot -> Final setting positions of building components.



Column component   Girder component   Slab component

Figure 5.2
Building component shapes

Table 5.1 Outline of Assembly Machine Models

Types of assembly machine	Number	Number of defined path control points	Storage/conveying capacity of components
Building component distributing machine	1	1	1
Component conveyer	1	2	1
Facility for temporarily stocking components (stockyard)	2	1	2 - 3
Component unloading lift	1	2	2
Crane robot	1	7	1

[Crane Robot Mechanism]

As the crane robot, a 4-joint cylindrical coordinates arm robot as shown in Figure 5.3 was assumed.

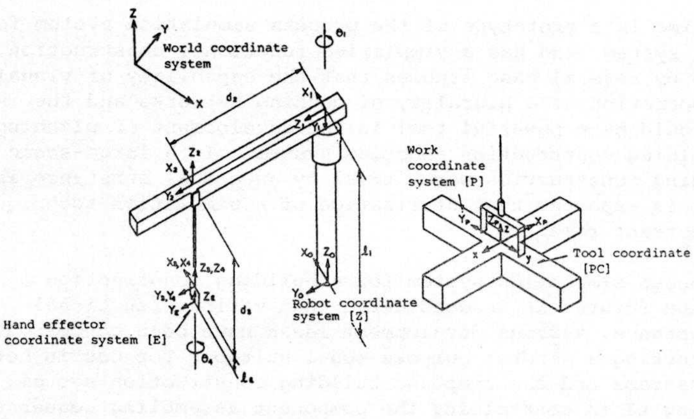


Figure 5.3 Crane Robot Mechanism

5.2 Simulation Result

As to the parameters related to the assembly machines, the moving speeds of the conveyer and the lift were assumed to be 0.5 m/sec, and the robot crane was assumed to have an arm expansion and contraction speed of 0.3 m/sec, a pivoting speed of 5°/sec and a hoisting speed of 0.5 m/sec. Also, as the time required for the motions of receiving and passing the components between the machines, a value of 10 - 15 sec is incorporated into each operation.

The assembly simulation result of the column components on the 4th floor and the girder and slab components of the 5th floor level of the same building model is shown in Table 5.2.

Table 5.2 Actual Operating Time of Each Machine

Machine name	Actual operating time (sec)	Actual operating Efficiency (%)
Component distributing machine	1439	11
Conveyer	5183	39
Lift	4535	34
Crane robot	13253	99

The time required to assemble a total of 144 components was 13,367 seconds (about 3 hours and 43 minutes), and the average assembly time required per component was about 11 minutes.

As a result of the specification that one computation period between events be 5 seconds or less, the positional computations were done at about 3,700 scenes (corresponding to the number of the output screens by animated display) in the whole process, and the number of simulation cycles was 3,204.

The processing of the simulation result was executed on an IBM 3090, and the computing time was about 135 seconds (CPU) and the processing time was about 37 minutes (ELAPS).

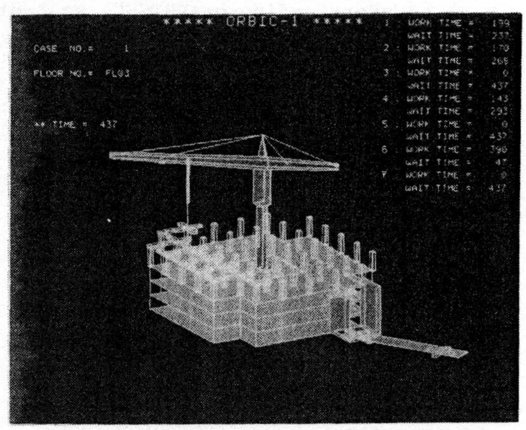


Figure 5.4 Graphic Display of the Simulation Result

Figure 5.4 shows a scene of 3-D animated display representation of the simulation result by a graphic display.

6. Conclusion

The system developed this time is a prototype of the process simulation system for building construction robotics system, and has a simplified functional construction. However, it was proven assured by several case studies that the capability of visually confirming the operational cooperation of a plurality of machine networks and the building construction status would be a powerful tool in the development or planning of a new process method for a building construction robotics system. In a large-scale production line, such as building construction work, trial by an actual substance is effectively impossible, and it is expected that utilization of a simulation technique of this kind would play an important role.

In order to make such a process simulation system for a building construction robotics system effective in the future, it is considered that various functional expansions are needed. For instance, various development ideas have been conceived, such as a technique for constructing a general purpose model suitable for use in both the PC building construction systems and the compound building construction system. An optimization technique utilizing AI in controlling the component assembling sequence and the composing of each building construction machines network is being considered. Also, a simulation technique including not only the operation of each machine but also a sensor function, and a technique for automatically producing and transferring the teaching data of each machine from the simulation result, and further research and development in this field are expected.

References

1) S. Harrington, "COMPUTER GRAPHICS", McGraw-Hill, 1983

2) E.L. Wilson and M. I. Hoit, "A Computer Adaptive Language for the Development of Structural Analysis Program", Computer & Structure, Vol. 19 pp321-338, PERGAMON PRESS, 1984

3) E. Yano, "A Microcomputer Interface That Can Be Made", pp134-142, Japan Broadcasting Press Association, 1984

4) J. Trenblay and P. G. Sorenson, "An Introduction to Data Structure with Applications", pp217-223, McGraw-Hill, 1984

5) R. P. Paul, T. Yoshikawa (Translator), "A Robot Manipulator", Corona Inc., 1981

6) "Understanding graPHIGS", sc33-8102, IBM, 1985