J. E. Stuehler

An Integrated Manufacturing Process Control System: Implementation in IBM Manufacturing*

Abstract: An integrated manufacturing process control system has been developed and implemented in several of IBM's manufacturing facilities to control process and test equipment used in the manufacture of IBM products. The system architecture consists of a central, on-line, IBM 360 Data Processing System operating under OS/360 which communicates via high-speed data channels to satellite IBM 1800, 1130 and System/360 processors.

These satellite processors control various types of process and test equipment. The central system serves as a common data bank and an input/output device for the satellite processors. In addition, the central system performs data analysis and management reporting on information obtained from the manufacturing floor. This paper discusses the general system requirements and specifications along with the hardware and software required to implement those requirements and specifications. Also discussed are problems which were encountered after initial development and plans for future development.

Introduction

IBM manufacturing facilities in both the United States and Europe have installed computer systems of essentially identical design to aid in the control of many types of manufacturing processes. The basic structure of the system is depicted in Fig. 1. One or two central computer systems (IBM System/360) are attached to several satellite computers (IBM 1130, 1800 and/or 360 processors) via a high-speed transmission control unit (multiplexor).

The satellite computers attach to, and control, various types of manufacturing process and test equipment. The central computer system serves as a data bank, processor and shared input/output device for the satellite computers. It provides for storage and analysis of process data. The central computer minimizes the cost and size of the satellite computers by performing tedious calculations, providing the facilities of a large data base, and reducing input/output requirements. When used, the second central system provides backup, additional capacity, and a better response in a duplexed mode of operation.

This system structure was developed so that one basic design could serve several IBM facilities, thus reducing the hardware and software development costs that would be incurred if each facility were required to develop its own manufacturing process control system(s). In addition, the common design was able to draw upon the combined resources of several locations in order to make optimum use of critical skills.[1] Another major advantage of the common control system is in minimizing the cost of transferring products for manufacture from one location to another.

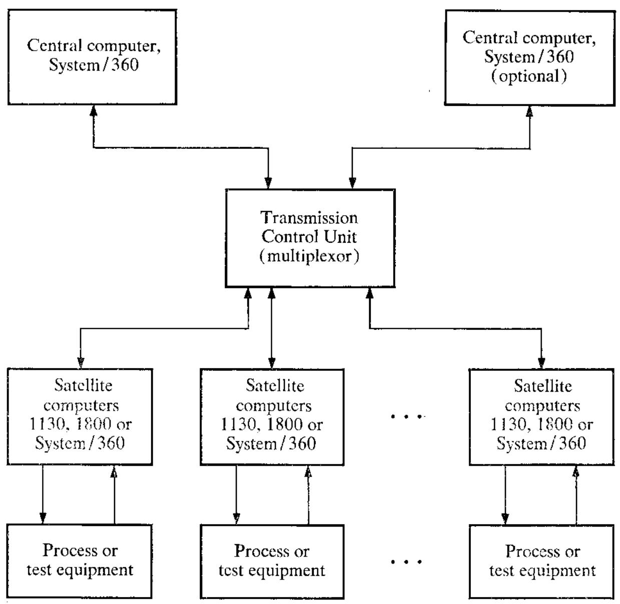

Figure 1 Manufacturing Process Control System used in IBM.

J. E. Stuehler is with the IBM Systems Manufacturing Division in Boulder, Colorado 80301.

* A different version of this paper was presented at the 1970 Fall Joint Computer Conference.

This paper describes the requirements for, and development of, the common system. Also included are some of the problems encountered and the oversights, now corrected, that occurred during development and implementation. Other papers in this issue discuss the transmission control unit,[2] the real-time operating system[3] and representative applications of the system.[4a,4b]

History

The earliest major manufacturing process control system to be implemented in IBM was the system known as COMATS (Computer Operated Manufacturing and Test System).[5] This system consisted of a pair of duplexed 1460 Data Processing Systems attached to numerous satellite processors through a high-speed multiplexor. The satellite processors were specially developed computers which provided the control required to test most of the disk storage products for which the system was designed. The system was conceived to reduce the costs associated with developing, building and maintaining special test equipment to do a similar job.

After COMATS became operational, it was obvious that many other manufacturing facilities in IBM had control requirements that could be met by a similar approach. However, each IBM plant manufactures, in general, products requiring its own particular test and process control philosophy. COMATS, as implemented, would not meet all needs. Therefore, in order to save each manufacturing location the cost of developing a unique system, a system versatile enough to meet the needs of most plants was developed.

The manufacturing environment

The types of manufacturing processes in IBM extend from the fabrication of microelectronic components through the assembly and testing of complex electronic processors; the products range from tiny precision machined parts through large electromechanical data processing input/output equipment. The processes required to produce these and other IBM products can, however, generally be classified among five categories.

The first is a process that produces a large quantity of a single type of electrical or mechanical component such as a magnetic disk or core, or a tape or disk head. The computer may be used here to control the mechanized operations. In addition, a great deal of process optimization is possible when the computer is used to collect and analyze data to determine the effect of each process variable.

The second type of process produces many "customized" variations of a single product. Examples include integrated circuits and printed circuit boards and cards. The computer can play an important role here by optimally controlling the process and test equipment. However, the computer must also supply information to the manufacturing process to "customize" the product. This requires obtaining and storing large amounts of engineering information about how each component is to be made and tested.

The third type of manufacturing process produces mechanical and electrical assemblies and subsystems. Product examples include central processing units and input/output equipment such as printers, tape drives, displays, etc. The process consists of assembly and test operations. Most of the assembly operations are difficult to mechanize. However, the computer may be used to give assembly instructions. An important role the control computer can play is in testing.[6] The computer can supply customized diagnostic programs for assemblies under test, provide the control logic required to test electromechanical input/output devices, and retrieve and analyze test results to provide processed output for use in correcting assembly problems and in controlling quality.

The fourth type of manufacturing process is general machining of mechanical components. The computer may be used to control the machining and measurement equipment. It is possible for the computer to feed back measurement data to machine tools to control and optimize the process. In addition, the computer is required to convert engineering information into tool instructions for each unique part to be machined.

The fifth type of process actually resides in development rather than manufacturing. An important step in the development "process" is in proving the product to be manufacturable. In order to experiment with process variables, a flexible and easily programmable control system is required. Furthermore, test data analysis is important to determine the effects of varying process parameters.

The manufacturing environment may be further characterized by considering that a plant may employ more than one of the above types of processes. Furthermore, a manufacturing process may be spread out in several buildings at distances of up to a mile apart. The processes continuously change to allow the introduction of new products or changes to existing ones. In general, each change must occur rapidly in order to keep pace with development and market requirements. Another trait of the manufacturing environment is that entire manufacturing processes are sometimes transferred totally from one location to another to balance work loads. Often a manufacturing process will be installed in two or more facilities for increased production and/or emergency production.

Selection of a system approach

With an understanding of the manufacturing environment, the requirements for a common control system can be identified. The most significant of these include sensor-based input/output capability, extensive information handling and storing capability, modularity with the ability to easily and rapidly install new applications, the ability to mix and transfer all types of applications, and certainly not least important is the requirement for economy.

As pointed out by Kinberg and Landeck,[7] the satellite computer system concept[8–12] best implements the above requirements. The satellite computers interface to process and test equipment through sensor-based input/output. These satellite computers may be transferred from one location to another and additional satellites may easily be added to expand an existing system. The central processor can provide extensive data analysis and large data banks, while minimizing the need for such capability at the satellite.

Two other possible approaches to developing a manufacturing process control system[13] were considered and eliminated for the following reasons. A single, large central computer system would not provide the power, versatility and modularity required. This is because of the number of different types of control applications that exist in an IBM plant, and because each such application normally undergoes frequent change which would be difficult to cope with on a single computer system without affecting other applications. The use of a separate control computer for each process would not be adequate because of the expense in providing data banks and information analysis capabilities. In addition, the cost of duplicated input/output equipment and redundant programming would be much greater than with the satellite approach.

Development and implementation

• *Central computer system*

The IBM System/360 was considered to be the most practical system for use as the central computer. The primary considerations were growth capability plus the existence of many types of input/output equipment and commercially available programs. Only third-generation data processing equipment was considered in order to provide state-of-the-art experience and motivation for the skilled programmers who would be needed to design the applications programs and implement an operating system. Other IBM data processing systems were considered which were generally lower in cost than the 360. These systems may have satisfied the needs of some types of processes where large data banks and a great deal of data analysis were not required (example: testing and process development). However, they had limited growth capability compared to the 360; and they did not have the larger analysis and input/output capabilities required in the other process applications. Therefore, at the expense of possibly "over computerizing" some very few locations, the System/360 was selected to maintain commonality.

• *Satellite computers*

It would be desirable to use a single type of satellite computer so that familiarity with the device would minimize programming and maintenance costs. However, because of the diversity of control requirements at the process level, this was not possible without greatly increasing the average cost of each control application. The amount of control logic required for process or test equipment is inversely proportional to the logic capability of the product being produced or tested. That is, more "intelligence" is required to test components than is required to test input/output devices. Similarly, more intelligence is required to test input/output devices than is required to test systems. Thus, in general, IBM 1800 and 360 systems are used to control process and test equipment which produce components (process types one and two) while the lower cost and lower powered 1130 system is used on products having higher intelligence (process types three and four). The 1800 systems are interfaced to process/test equipment through digital and analog input/output channels. The 360 and 1130 systems are interfaced via special hardware connected to the standard channels.

• *System response*

To keep the cost of the satellite computers low, it is necessary to minimize the data processing requirements (amount of core and speed) and the input/output equipment required at the satellite. Thus, the satellite computer will be heavily dependent on the central computer for these services. However, when the satellite computer requires data or programs from the central computer, or is required to send data to the central computer, it cannot wait for a long period of time because the process or test equipment may also have to wait (consequently requiring more production equipment and higher implementation costs). Ideally, the satellite computer should be able to send or receive data from the central system as fast as the satellite could access its own files if it had them. Thus, a system design specification for simple data/program transfer was established at 500 msec 95% of the time with the central system handling 3600 interrupts/hour (an average of one interrupt per second). This specification is the length of time the satellite computer must wait from the time it requests a program, or some data, from the central system (or requests the central system to take data) until the program or data (an average of 2,000 bytes) have entered the satellite (or the satellite has sent 2,000 bytes of data). This specification places severe requirements on the communications system be-

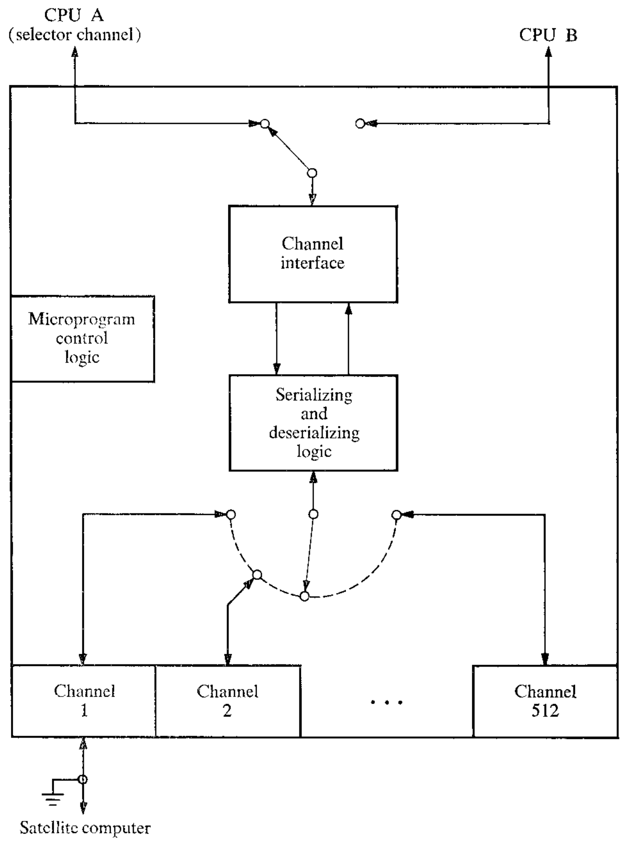

Figure 2 Transmission Control Unit (TCU).

tween the satellite and central computers as well as on the central computer's operating system.

• *Communications system*
A multiplexor was required to allow communications between the satellite and central computers. The requirements of such a device included:

Modularity in the number of attachable satellites to allow growth (a maximum of several hundred satellite computers in some facilities was considered reasonable while other facilities might never have more than a dozen).

Distance capabilities for communicating up to one mile were required to be able to cover a plant site.

Channel bandwidth for transmission in the megabit/second range was needed to minimize satellite waiting time. That is, the channel bandwidth should be of the same order of magnitude as the central CPU channel and file transmission rates.

Serial-by-bit transmission was required to minimize cabling costs.

High reliability of transmission was required to allow operation in an electronically noisy factory environment.

No commercially available multiplexors were found to be suitable under the above requirements. A Transmission Control Unit (TCU)[2] was, therefore, developed; this unit is shown schematically in Fig. 2. The TCU is basically a microprogrammed, solid-state switch that provides polling and allows communication between either of two central 360 computers and any of up to 512 satellite computers (modular in groups of 64). The TCU also provides serializing and transmission logic which allows transmission of serial-by-bit data over a single coaxial cable at a rate of 2.5×10^6 bits/second. The number of bits in error is less than one bit for every 10^8 bits transmitted. The TCU communicates with a satellite computer via a transmission adapter.

• *Central computer operating system*
Some of the more important characteristics of an operating system in the central system include the following:

Response time to accept an interrupt from the TCU and begin processing should be of the order of a few milliseconds to provide the response required by the satellite computer which may be waiting.

Input/output support for many different types of devices is required to allow the system to be applied effectively in the different process environments. That is, one environment may require small, fast-access files while in another environment slower access to large quantities of information is required. Differences may also be found in the requirements for graphic terminals, printers, tapes (for history), etc.

Multiprogramming is required to allow one or more satellite computers to receive service while file accessing or other input/output operations are pending for another satellite computer. This capability greatly reduces the waiting time due to queues for the satellite computers.

Support software such as compilers, assemblers, analysis routines, etc is required to minimize programming costs and the need for programmers.

Modularity is required in order to minimize the price the small user must pay in core-storage overhead which is required to obtain the sophistication needed by the larger user (examples include number of levels of multiprogramming, requirements for compilers, concurrent operation of peripheral input/output devices, types and amounts of input/output equipment, etc.).

Two approaches were considered to implement the above major requirements. One was to develop a special operating system and the other was to implement a commercially available one. The special operating system would be better from the standpoint of response and amount of core required since it could be customized to perform well in these areas. The disadvantages were that a great deal of development work would be necessary to provide the versatility required to support the varied input/output requirements at each using location. Furthermore, advantage could not be taken of already available compilers and utility programs designed to operate under a commercially available operating system. For these reasons, the decision was made to use a commercially available operating system.[14]

The best system available to provide multiprogramming capability was the IBM Operating System/360 (OS/360) which was augmented by a "Real-time Control Program (RTCP)" to provide a real-time multiprogramming environment for supporting the satellite computers. This combination of OS/360, the RTCP and an input/output appendage to support the TCU is referred to as the Process Control Operating System (PCOS).[3] The core map of the central computer is illustrated in Fig. 3. TCU communications and TCU-detected errors are handled by the TCU appendage. All TCU interrupts are passed to the RTCP which invokes either a core- or disk-resident service module (real-time program) to handle the interrupt. The service module may (if required) initiate a background program to perform analysis on information the satellite computer has sent. The service modules always have priority in utilizing the central system resources, which allows a fast response to a satellite request for service.

Normally, the time required to enter a core-resident service module after the TCU posts an interrupt to the central computer is less than 25 msec (if the service module is not active).

A drawback of using OS/360 as compared to a special purpose operating system is in the amount of core required. A Model 40 with 128K bytes of core storage is the smallest system in the 360 line that can effectively run OS/360. This again is an expense to some locations which might have begun with a System/360 Model 30 as the central computer if a "special" operating system had been developed.

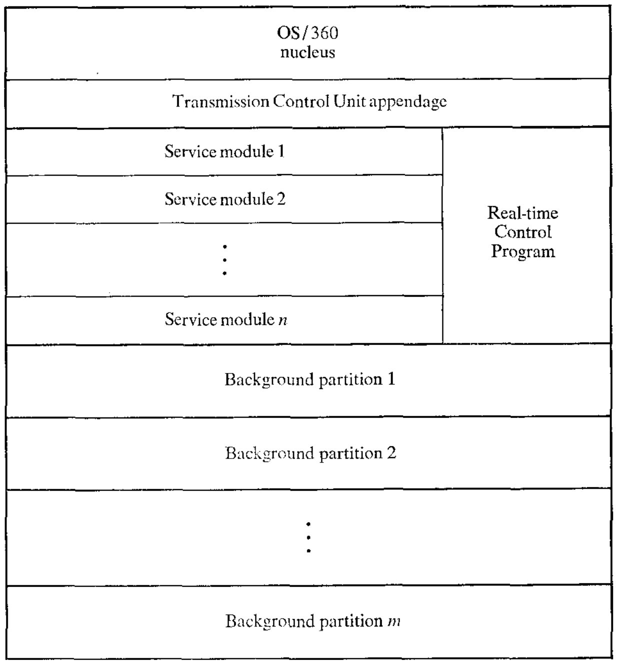

Figure 3 Process Control Operating System (PCOS).

Considerations in retrospect

Experience has revealed two problems that have now been solved but were not originally anticipated. One was a hardware design problem and the other a software problem. After the TCU specifications had been determined, any location planning on the first usage of a type of satellite computer (i.e., 1130, 1800 or 360) had the responsibility to develop the unique transmission logic adapter between that type of satellite computer and the coaxial cable which connected to the TCU. It was later learned that each designing location had developed an interface completely different from the others. As a consequence, common software in the central system could not be used to communicate with every type of satellite computer. This was because each type of transmission logic adapter presented different status indicators to the TCU or responded differently to TCU commands. Furthermore, since each adapter was designed differently, an engineering change placed in the TCU might affect some adapters adversely while not affecting others.

This problem was solved by developing a common transmission logic interface which could be used by every type of satellite computer. The common adapter was then uniquely interfaced to a particular type of satellite computer as illustrated in Fig. 4. Now all satellite computers look the same to the TCU and central system software.

The software problem had to do with the definition (or lack of definition) of the service modules which reside under the RTCP. Most system users were developing "application" service modules to be unique to a given application. As an example, a particular tester would require one or more unique service modules to completely support it; and these application service modules could support no other application. Other locations, however, began development of "system" service modules. A single system service module could provide service to two or more applications. An example might be a system service module to retrieve information and store it on a file for any application, whereas each application service module contained its own retrieval logic. The benefit of system service modules is that these routines, which can be common, would have to be developed only once. However, there is an initially high development cost for each such service module since they must offer a great deal of versatility.

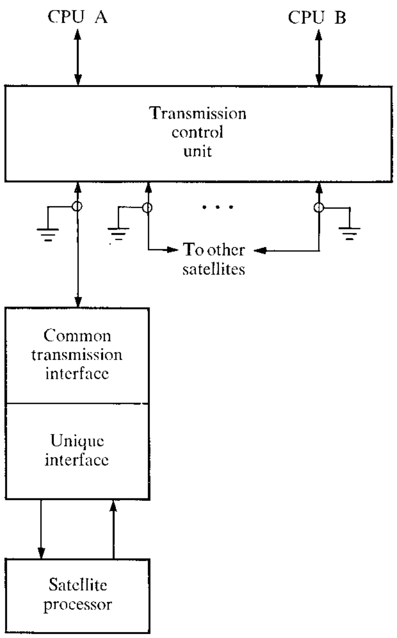

Figure 4 Transmission Logic Scheme.

The problem presented by having these two philosophies was that applications developed to be run on a system using system service modules could not easily be transferred to another location without also transferring the system service modules or rewriting the application programs. This problem was solved by defining and developing a set of system service modules that are considered an integral part of the operating system. Being able to change this philosophy of operating system function attests to the need for a great deal of operating system versatility. It is not unreasonable to expect that other conceptual changes will be made in the future based on knowledge not yet gained.

Current system usage

Architectural, as well as hardware and software specifications for the system were developed by representatives from each user location in 1967. Development responsibility for portions of the hardware and software were assumed by several of IBM's manufacturing locations. In 1968, the components of the system were brought together at a single IBM manufacturing facility in Boulder, Colorado for system testing. The system is installed and operating in eight domestic and two European IBM manufacturing plants and plans currently exist for implementation at several other plants. The need for a versatile systems approach can be attested to by looking at how the common process control system is applied at several IBM locations. At two locations, 30 to 50 satellite 1130 computers are being used for testing electromechanical input/output devices. Both of these locations utilize the central system as the satellite's input/output device (storing programs, reporting, etc.). Furthermore, the central system performs test data and defect analysis and reporting for the quality engineering group. Two other locations use the system heavily for controlling processes producing magnetic components. Here some satellite computers (e.g., 1800's) are used to control process variables while other satellite computers (e.g., 1130's) control test equipment. The central computer is used to store and correlate test results with process variables, thus allowing process optimization. As soon as enough data can be accumulated, process models can be designed and installed in the central system to better control the processes. Two other locations use the system primarily for supplying test programs to central processing units undergoing test. Here the central system stores and supplies large diagnostic programs to satellite computers. The central system collects test and diagnostic data for engineering analysis. One of these two locations uses its system to give assembly instructions via display units to assembly personnel working on complex electronic subassemblies. One location uses its system to test complex integrated circuit memory modules. Here the central system must supply test data to satellite computers which control test equipment. Again, the central system receives the data, analyzes reports and stores relevant test data received from the satellite computer. Other IBM locations have combinations of the above applications installed on their systems. The number of satellite computers ranges from half a dozen at one location to over 50 at another. Central system configurations at the various locations include a single System/360 Model 40, a pair of Model 50's, and a pair of Model 65's. Table 1 gives a summary of system usage in IBM that is up-to-date at the time this paper is published.

System performance

It is impossible to generalize any system performance criterion because of the differences in system configuration implemented by each using location (i.e., types of input/output, size of processor used, features of operating system utilized, etc.). However, a "representative" location has installed 35 satellite 1130 computers on a System/360 Model 40 central system with 256K of core. They utilize the multiprogramming with a fixed number of tasks (MFT) version of OS/360 and use a 2314 Disk Storage Unit as their bulk file. The observed response of their system closely follows the results of the simulation of their system, which is depicted in Fig. 5. This curve shows that 1890 messages per hour can be handled with a response of 300 msec (90% of the time). The system performance specification of 3600 interrupts/hour being handled within 500 msec is shown to be met. Other data show that with 1890 interrupts/hour, the system is 20% utilized (with no background processing). Input/output is one percent

Table 1 Summary of system usage.

User Location	*Date on-line*	*Central system, type & number of units*	*Satellites, type & number of units*	*Terminals, type & number of units*	*Applications*
Boulder, Colorado	3rd Q, 1968	IBM System/360, Model 50 (1)	IBM 1401 (1) IBM 1440 (1) IBM 1130 (6) IBM 1800 (1)	IBM 2260 (12)	Assembly & test of magnetic tape drives; monitoring & control of magnetic recording head manufacturing processes; test of circuit cards and analysis of test data; remote entry of job status data; numerical-control processing; motor testing.
San Jose, California	3rd Q, 1968	IBM System/360, Model 40 (2)	IBM 1130 (35) IBM 1800 (1) IBM System/360, Model 40 (1)	IBM 2260 (33) IBM 2740/41 (10) Special terminals (60)	Monitoring & control of magnetic recording head manufacturing processes; storage disk manufacturing; assembly & test of disk drives & control units, production control & defect reporting, management data reporting.
Kingston, New York	4th Q, 1968	IBM System/360, Model 65 (2)	IBM 1130 (3) IBM 1800 (1) IBM System/360, Model 65 (28)* IBM System/370, Model 165 (20)*	IBM 2260 (16)	Assembly & test of computer central processing units; test of circuit cards, test data generation, simulation, quality reporting, field diagnostic support.
Raleigh, North Carolina	4th Q, 1968	IBM System/360, Model 40 (1)	IBM 1440 (5) IBM 1130 (4)	—	Test of data communication terminals; test of keyboards for data entry; remote entry for job status data.
Rochester, Minnesota	1st Q, 1969	IBM System/360, Model 40 (1)	IBM 1130 (43)	—	Assembly & test of optical character readers and input/output equipment; inspection of mechanical components; test of circuit cards.
Mainz, Germany	3rd Q, 1969	IBM System/360, Model 40 (1)	IBM 1440 (2) IBM 1130 (12)	Special terminals (34)	Monitoring & control of magnetic recording head manufacturing processes; assembly & test of computer central processing units.
Burlington, Vermont	1st Q, 1970	IBM System/360, Model 50 (2)	IBM 1130 (3) IBM 1800 (9)	IBM 2260 (100)	Test of monolithic memory; monitoring and control of memory manufacture.
Vimercate, Italy	1st Q, 1970	IBM System/360, Model 40 (1)	IBM 1130 (10)	—	Assembly & test of computer central processing units & input/output equipment; test of circuit cards.
East Fishkill, New York	1st Q, 1970	IBM System/360, Model 65 (1)	IBM 1130 (1) IBM 1800 (2)	—	Test & data analysis of integrated circuits.
Boca Raton, Florida	2nd Q, 1970	IBM System/360, Model 40 (1)	IBM 1130 (16)	—	Assembly & test of computer central processing units & input/output devices; test of circuit cards.

* The Models 65 and 165 under remote systems test appear as satellites to the central system.

utilized. The service modules (application programs) servicing the interrupts had run times ranging from one to ten msec with an average of seven msec. There was an average of two file accesses per interrupt.

The future

With a common system structure established in a number of IBM manufacturing plants, it is now possible to develop additional common system services and applications.

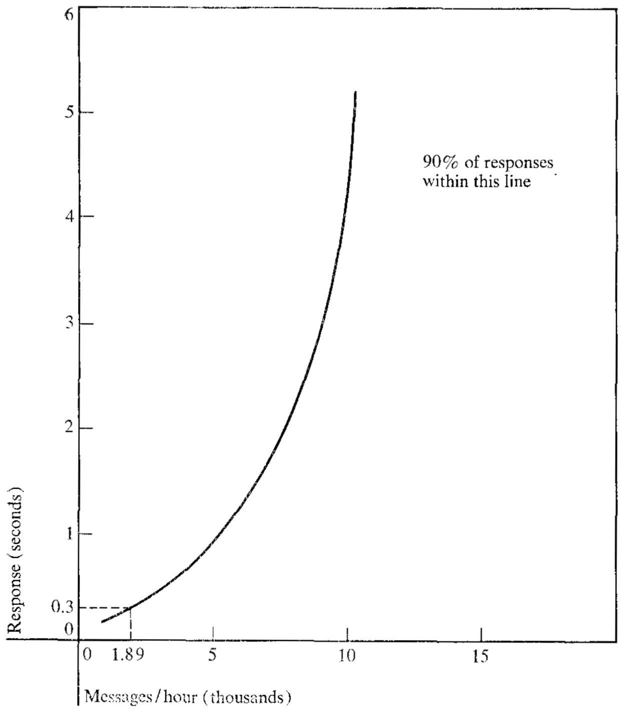

Figure 5 System Simulation.

A concept currently being considered is a high-level Process and Test Language (PTL) which, through the use of the central system's resources for compilation, will allow an engineer with minimum programming experience to efficiently and rapidly apply a satellite computer to control process or test equipment.[15] The test or process engineer can develop the control logic using macro statements in high-level languages he can easily learn.[16] Similar statements can be used to send process/test information to the central computer where programmer-written routines can analyze, store and report on the information.

Projected for the future are satellite-computer-driven machine tools with numeric control processors and post processors resident at the central computer and accessible by the satellite. It will be possible for machine-parts programmers to directly enter macroprogram statements in the satellite computer or a terminal device to make a new part or modify an old one. These statements will be sent to the central computer for checking and compilation. The result of the compilation will return to the satellite computer to cause execution by an on-line machine tool.

Process automation programs which reside in the central computer will be developed to directly accept raw development (engineering) information.[17] These programs will then interpret the design information and send process information to satellite computers which will be used to control process and test equipment. Among other benefits, the use of such programs will allow a very fast response to product/process engineering changes.

A common quality-assurance analysis programming system which will reside in the central computer is being considered. This program will allow the quality control engineer to easily use complex statistical methods in analyzing process information when a new application is installed on the system. It is possible that the quality control engineer will, in the future, be directly entering high-level (macro) statements in a language like PTL into satellite computers. These statements would specify data he would want collected and indicate to the central system the kind of analysis to be performed on the data.

Also important for the future is more sophisticated use of data management techniques. Such techniques will allow a user to directly access system data banks to retrieve test and/or process data without the aid of a programmer. The user will then be able to specify statistical programs to operate upon the data and methods for presenting the final output.

The process control central system will be interfaced to other manufacturing information systems that process information pertaining to production, warehousing, maintenance and in-process inventories. Information to be passed to those systems from the process control system includes production yields, equipment down time, units in process, units tested, etc. By completing the tie of process control systems to manufacturing information systems, one can achieve true plant automation.[18,19]

Summary

The justification for undertaking a common manufacturing process control approach in IBM was to reduce redundant hardware and software development and implementation costs at each IBM location. This has been accomplished. In addition, it has been possible, by pooling ideas, to develop a system superior to that which a single location could have developed. This pooling of ideas and knowledge has ultimately led to a process control solution which, in general, reduces the costs of installing new manufacturing processes and of modifying existing ones.

Although the system as described in this paper is installed and operating at several IBM facilities, it appears to be only a first step toward automation. Almost daily, new ideas by one or more users of the system suggest how new functions could be added to the system to either further reduce the implementation cost of new applications or to improve the quality of products being manufactured under control of the system. It appears that these ideas, which are born as a result of experience in using the system, are the real justification for the common system approach since many users can now benefit from a single idea and development.

Acknowledgments

It would be lengthy for the author to acknowledge all individuals who contributed directly or indirectly to the success of the system. However, the key hardware developers were R. Watkins, F. Thoburn and T. Rall all of San Jose, California and K. Cisewski of Rochester, Minnesota. The primary software developers were M. Mauldin and T. Reilly of Kingston, New York along with R. Henry, San Jose, California and J. Calva, Rochester, Minnesota. R. Boydston and C. Connoy of San Jose, California developed the system simulator mentioned in this paper. Special acknowledgment goes to W. Moore of Harrison, New York (IBM's System Manufacturing Division Headquarters) for the excellent job he performed in initiating and coordinating the entire project.

References

1. W. Anderson, "Controlling Processes with Computers," *Automation* **16,** No. 1, 70 (1969).
2. F. W. Thoburn, "A Transmission Control Unit for High-speed Computer-to-computer Communication," *IBM J. Res. Develop.* **14,** 614 (1970), this issue.
3. J. R. Calva, "PCOS: A Process Control Extension to Operating System/360," *IBM J. Res. Develop.* **14,** 620 (1970), this issue.

4a. R. B. Fischer, C. M. Cox and C. Holdinsky, "Computer-aided Testing and Fabrication of Magnetic Tape Heads," *IBM J. Res. Develop.* **14,** 633 (1970), this issue.

4b. M. R. Radio, et al., "Precision Automatic Measuring of *X-Y* Coordinates," *IBM J. Res. Develop.* **14,** 641 (1970), this issue.

5. J. E. Stuehler and R. V. Watkins, "A Computer Operated Manufacturing and Test System, "*IBM J. Res. Develop.* **11,** 452 (1967).
6. J. E. Stuehler, "Hardware-Software Trade Offs in Testing," *IEEE Spectrum* **5,** No. 12, 51 (1968).
7. C. Kinberg and B. W. Landeck, "Integrated Manufacturing Systems: Architectural Considerations," *IBM J. Res. Develop.* **14,** 589 (1970), this issue.
8. N. Gaines, et al, "Union Carbide Integrates Multicomputer Process Control," *Instrumentation Technology* **14,** No. 3, 49 (1967).
9. J. Waugh and A. Yonda, "NSRL On Line Computer System," *IEEE Transactions on Nuclear Science* **NS-15,** 129 (1968).
10. R. L. Horst, "The Justification of Digital Process Control," *Modern Data Systems* **1,** No. 8, 20 (1968).
11. V. Loskutov, "Computation Technology in Automatic Control Systems," *Mekhanizatsiya i Avtomatizatsiya Proizvodstva* **18,** No. 3, 44 (1964); translation FTD-HT-23-835-67, Foreign Technology Division U. S. Air Force, Wright-Patterson AFB, Ohio.
12. C. Bond, "A Digital-Computer System for Industry," *Industrial Electronics* **4,** No. 5, 221 (1966).
13. J. E. Stuehler, "The Devoted, Shared or Satellite Approach for Computer Control of Manufacturing Processes?" *Proceedings of Western Electronic Show and Convention* 8–1 (1969).
14. J. Spooner, "Real Time Operating System for Process Control," *Instrument Society of America Conference Reprint* D1-1DAHCOD(67), 1967.
15. E. Johnson and J. McCarthy, "Development of Software Systems for Automated Test Equipment," *Proceedings of Western Electronic Show and Convention* 21–2 (1969).
16. Special Issue on Process Control Computer Programming Languages, *IEEE Transactions on Industrial Electronics and Control Instrumentation* **IECI-15,** No. 2 (1968).
17. R. Boedecker, "The Computerized Factory," *Assembly Engineering* **9,** No. 6 (1966).
18. N. Chiantella, "The Systems Approach to Plant Automation," *ASTME Vectors* **3,** 4 (November, 1968).
19. F. Gut, "Operations Control Systems," *Automation* **16,** No. 1, 55 (1969).

Received November 5, 1969