F. W. Thoburn

A Transmission Control Unit for High-speed Computer-to-computer Communication

Abstract: An integral part of a process control and testing system used in several IBM plants is the transmission control unit (TCU). This paper discusses the design of the TCU, which provides the communication link between as many as 512 satellite computers and a central computer. It uses a microprogrammed polling scheme to establish connection with the satellites that need service and it permits messages of unrestricted length to be transmitted in either direction at the rate of 2.5×10^6 bits/sec over coaxial cables. With the aid of noise suppression circuits, it operates in the electrically noisy factory environment at an error rate of less than 10^{-8}.

Introduction

A system used in several IBM manufacturing plants for automated process control and testing has been described by Stuehler.[1] He has discussed the organization of the system in which a number of "satellite" computers scattered throughout a plant site are connected via a transmission control unit (TCU) to either or both of two duplexed "central" computers. This paper is concerned with the design of the TCU itself.

In any system where data is transmitted between a computer and a multiplicity of terminals, some means must be provided to multiplex the signals traveling in either direction between the terminals and the computer. The logical structure of the multiplexor depends on a variety of elements in the data transmission environment. Primary among these elements are the desired transmission rate, the distance between terminals and the computer, the number of terminals in the network, the frequency with which the terminals need service by the computer and the amount of data processing done by the terminals themselves.

In the environment with which we are concerned the crucial element is the desired speed of transmission. The desired speed is approximately that at which the satellite computer would be able to access the central computer data-base files if these files were directly attached to the satellite computer itself. This means a speed in the megabit per second range.

The author is located at the IBM Systems Manufacturing Division plant in San Jose, California.

Closely related to this requirement for high speed is the fact that the connection between the satellites and the central computer would be made via private lines over relatively short distances—no more than one mile.

A primary example of high-speed, short-haul transmission is the Bell System's T1 carrier system.[2] This system uses a pair of 22-gauge twisted wires with repeaters at 6000-ft intervals to transmit pulse-code modulated data over 24 time-multiplexed voice channels at an aggregate of 1.544×10^6 bits/sec.

The solution to the multiplexor design problem presented by the IBM manufacturing environment was influenced to a considerable extent by experience with a previously designed system called COMATS,[3] which had been installed in IBM's San Jose plant in 1966. It used a modified IBM 1460 computer as the central system and its satellite processors were based on the IBM 1440 computer. Multiplexing was accomplished by a hardware interrupt scanner and data transmission between the central system and the satellites occurred at a rate of 1.25×10^6 bits/sec.

The advent and availability of IBM's third-generation computers, including the System/360, the 1130 and the 1800 systems led to an examination of methods by which COMATS could take advantage of their increased computing power. Of particular concern in the multiplexor design was the desire to increase the transmission rate, to use a hardware scheme for polling the satellites, and to provide suitable interfacing circuitry between the third-generation satellites and central computers.

The approaches considered for up-dating the COMATS multiplexor included 1) redesigning the interfaces while retaining the existing multiplexor circuitry, 2) redesigning the existing multiplexor to make use of then new SLT[4] circuit family and 3) using a completely new multiplexing scheme. The third alternative seemed to offer the greatest long-term advantage; however, no commercially available unit was discovered that provided all the desired capabilities at an acceptable price. The approach finally chosen was to modify an existing IBM 2841 Storage Control Unit.[5] The 2841 is intended for use in controlling access to a disk or strip file or a slow-speed drum storage unit. It is a microprogrammed[6] device that connects to a standard I/O Selector Channel[7] of the IBM System/360; its control programs are contained in a transformer read-only store (TROS).[8] Since the 2841 is not intended for connection to remote files, it was necessary to completely rewrite its control programs and to alter some of its hardware features.

The result of this effort is a transmission control unit with the following major features:

- It can establish a connection between either of two central computers and any one of up to 512 satellites.
- It connects to satellites that are arranged in a "star" configuration.
- It uses a programmed polling scheme to determine whether or not a particular satellite needs service.
- It permits data to be transmitted in a "burst" multiplexing mode.
- It causes data to be transmitted in either direction at a rate of 2.5×10^6 bits/sec over one coaxial cable per satellite.

The alternatives to these features and the reasons for the choices made are discussed in the next section. The following sections describe the operation of the device—in particular, the polling and "handshaking" methods, the data modulation scheme and the transmission circuits.

Major design decisions

The most important considerations affecting the logical structure of the TCU involved choices to determine the number of satellites that could be connected to the system, the method of connecting the satellites, how communications would be initiated, the multiplexing method, the transmission speed and the means for achieving a given level of reliability.

- *Number of terminals*

The maximum number of satellites that could be connected to the TCU was chosen to be 512 in 8 modular groups of 64. This number was selected because of experience with COMATS, which had grown from an initial terminal capacity of 64 to 100 and then to 256 within a period of four years. Each level of growth required redesign of the multiplexor. This pattern of growth prompted a structure for the TCU that would permit a plant to expand its use of the system without having to replace the TCU every few years.

- *Terminal connection method*

The two most common arrangements for connecting terminals to a multiplexor are the loop and the star configurations.[9] In a loop arrangement the terminals are connected serially at points on a line that starts and ends at the unit controlling transmission. In a star configuration each terminal is connected directly to the unit controlling transmission, sometimes in a way that permits several terminals to transmit data on the same line.

One recent example of a loop arrangement is seen in the IBM 2790 system,[10,11] which is used primarily for the transfer of manually entered job-status messages from up to 100 terminals around a plant floor to a central computer. The system transmits data around the loop at an aggregate rate of 0.5×10^6 bits/sec. When the loop is made up of 22-gauge twisted-pair wires, repeaters are used at 1000-ft intervals, and when shielded twisted pairs are used, repeaters are inserted at 4500-ft intervals.

The star configuration was chosen for the TCU because the distance over which the data must travel between a given terminal and the control unit is, on the average, less than in a loop; and in the environment for which the unit was to be designed, repeaters would not be necessary. The star configuration would also make relocation of the terminals easier.

- *Method for initiating communication*

Communication between a terminal and a transmission control unit is commonly established either by an interrupt or a polling scheme. If an interrupt scheme is used, the terminal sends a message to the control unit whenever it needs service. The message is likely to specify the type of service needed, and when the message is decoded, the control unit notifies the terminal and then begins the data transfer procedure.

When polling is used, the control unit sends an inquiry message to the terminal to find out if that terminal needs service. A positive reply to the inquiry causes the data transfer procedure to begin, whereas a negative reply causes the control unit to poll another terminal.

Although an interrupt scheme has the potential for more efficient use of the system's facilities, a polling scheme was felt to be preferable for use in the electrically noisy environment of the factory. Experiments in attempting to use an interrupt scheme with COMATS showed that, with a large number of satellites, interrupts occurred frequently enough that cross-talk at the control unit would produce misleading messages.

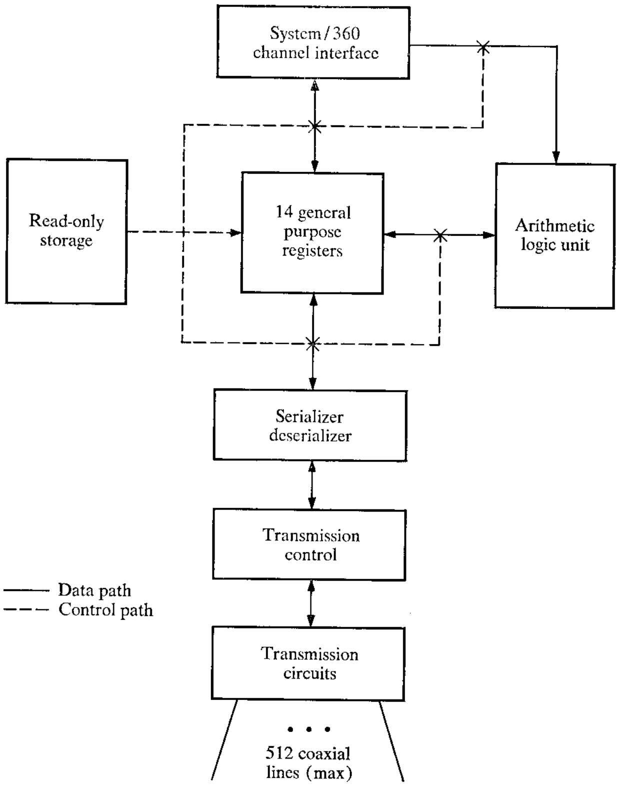


Figure 1 Block diagram of TCU.

• *Multiplexing methods*

In time-division multiplexing the length of time required to transmit a specified unit of data from or to all terminals in the system is called a *frame*. Among the many ways of specifying the unit of data to be transmitted in each frame are the following: 1) one bit per frame, 2) one byte per frame (in System/360 there are 9 bits per byte), 3) a block of some given number of bytes per frame and 4) a burst of an unrestricted number of bytes per frame.

Any form of time-division multiplexing requires some means at the receiver to associate the bits in the serial data stream with the proper terminal. The application considered here calls for the demultiplexed bits to be entered into a central computer channel after they are received at the TCU. The data entry procedure requires a "handshaking" routine each time data from a different satellite are presented to the central computer. If the data are presented at the rate of a bit per frame or even a byte per frame, the handshaking could easily take more time than is actually spent in transmitting the data. Since the length of any given message associated with a satellite is unpredictable, it is also not practical to base a frame length on a specified block of bytes. These considerations led to the use of burst-mode multiplexing in which a satellite gains exclusive use of the central computer channel for as long as needed to transmit the message.

• *Transmission speed*

The previously discussed design decisions resulted in a polled-star configuration with burst-mode multiplexing. Since this method of providing access to the central computer "locks-out" all other satellites for the duration of the data burst (which can be on the order of several thousand bytes) it is necessary to transmit the data at very high rates in order to achieve fast response to all satellite requests.

The particular rate was chosen to be compatible with satellite memory access rates. Potential users of the system indicated that the 1130 system would be the most commonly used satellite. This system is available with memory access-time options of either 2.2 or 3.6 μsec/two-bytes. There were no obvious reasons to restrict users to the 2.2 μsec option, so the fastest permissible rate would be 1.8 μsec/byte or 5×10^6 bits/sec. However, transmission of large bursts of data at memory cycle speed is hazardous because even slight differences between the transmission clock rate and the memory clock rate can cause loss of some data. Therefore the data transmission rate had to be slower than 1.8 μsec/byte. The solution was to transmit at 3.6 μsec/byte, using every other memory cycle for transmission, and, thus, permitting the satellite to do some processing concurrently with transmission.

Of course, it would be possible to achieve higher data rates if enough cables were installed to permit data to be transmitted parallel-by-byte between the satellite and the TCU. This was economically prohibitive, however, so serial-by-bit transmission via a single coaxial cable was chosen. Besides providing good reliability when used in conjunction with noise suppression circuits, the coaxial cable has a larger bandwidth capacity than twisted pairs.

• *Reliability*

Experiments with a prototype system showed that its error rate was less than one error bit per 10^8 bits transmitted. This rate was low enough to indicate that special error detection/correction schemes were not neccessary. The parity bit in each byte is sufficient for error detection. When an error is detected, the message is retransmitted.

TCU operation

A block diagram of the TCU is shown in Fig. 1. One of the blocks indicates a set of 14 general-purpose registers, each containing 9 bit positions. This choice of register size makes it compatible with the byte size of the System/360—8 data bits and 1 parity bit. While the 9-bit mode of transmission is the normal one for the TCU, provision is also made for transmission in a 10-bit mode which makes the TCU compatible with COMATS. The instruction sets of the COMATS computers use 8-bit BCD characters. In its 10-bit transmission mode, the TCU uses the extra two bits for synchronization.

The System/360 channel interface shown in Fig. 1 is the conventional way of attaching any I/O device to the System/360 channel. Once the TCU gains access to a channel (see the companion paper on PCOS by Calva[12]), a channel command word from a program stored in the central computer can be executed. The set of 15 commands is listed in Table 1.

The microprogram stored in the transformer read-only store (TROS) controls the flow of signals among the various registers in the TCU. Each elementary operation in the microprogram is defined by a 48-bit word. A word is read from the TROS every 500 nsec.

On receiving a START POLLING command from the central computer, the TCU sequentially polls each satellite to determine whether or not that satellite needs service. The coaxial cable for each satellite is terminated at the TCU by a circuit card that contains an "enable-disable" latch. During the polling operation the latch corresponding to the satellite specified by the TCU's 9-bit satellite address register is tested. If the latch is in the "disable" state, the address register is incremented and the latch for the next satellite is tested. When the TCU finds a latch in the "enable" state, it sends a one-byte poll order to the satellite. Three reactions to this order can ensue. 1) The satellite does not need service and therefore sends a one-byte negative response to the TCU. 2) The satellite is currently operating off-line, in which case the TCU waits 50 μsec for a response before incrementing the address register. 3) The satellite requires service and therefore sends a one-byte positive response back to the TCU along with from 0 to 16 bytes of "request code."

The request code and the satellite address are immediately presented to the central computer for processing. The TCU then sets the latch for this satellite to the "disable" state so that it will not be polled again until it has been serviced by the central computer. The address register is then incremented and the whole process is repeated for the next satellite while the central computer is processing the request code.

When the central computer receives a request code from the TCU it stores it in the queue of satellites requesting service. Whenever the central computer is ready to act on a request, it sends a command to the TCU. If a read operation is to be performed a SELECT CONTROLLER command must first be issued to identify the satellite to be serviced. For a write operation the first two bytes of data identify the satellite to be serviced.

On receiving a READ command, the TCU sends a one-byte read order to the satellite and waits for a response. If the response is positive, it is immediately followed by data which is transferred serially-by-bit to the TCU where it is reassembled into bytes and transferred parallel-by-bit to the central computer. If there is no response at all after 50 μsec or if the response is negative, the operation is stopped and the central computer is immediately notified of these conditions. The satellite may also issue a "wait" response, which means that it is not ready now, but will be ready to transmit data within the next 10 msec.

Receipt of a WRITE command by the TCU means that the satellite had requested data from the central computer.

Table 1 Command set issued by central computer to TCU.

Command	*Explanation of purpose*
SELECT CONTROLLER	Specifies the 9-bit address of the satellite computer with which the central computer wishes to communicate.
MASK CONTROLLER	Causes the enable-disable latch to be set to a specified state for a satellite address given by the central computer. This can be used to indicate the addresses for which no satellite exists in the application of the system and hence those addresses that need not be polled by the TCU.
START POLLING	Causes the TCU to begin searching for latches in the enable state which indicates that a satellite should be polled to determine whether or not it needs service.
READ REQUEST CODE	Causes the TCU to transfer a satellite's request for service to the central computer.
READ FROM CONTROLLER	Causes the TCU to notify a specified satellite that the central computer is ready to receive data in aswer to the satellite's request for service.
WRITE TO CONTROLLER	Causes the TCU to notify a specified satellite that the central computer is ready to send the requested data.
CPU CALL	Causes the TCU to notify a specified satellite that the central computer wants to send it some unsolicited data.
POLL CONTROLLER	Permits the central computer to command the TCU to determine whether a specified satellite needs service.
STOP POLLING	Prohibits the TCU from proceeding with its automatic polling operation.
RESET TCU	Cancels all operations in th TCU.
SENSE TCU	Causes TCU to present six bytes of sense data. This command is used in diagnostic routines.
READ MASK	Permits the central computer to find out the status of a specified enable-disable latch.
TEST I/O	Causes the TCU to transfer status to the I/O channel.
SWITCH CABLE	Causes the coaxial cable from all satellites to be switched to another TCU.
NO OP	A command provided for programming convenience.

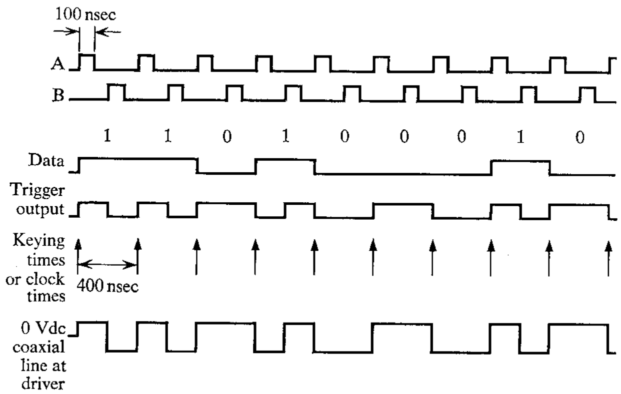
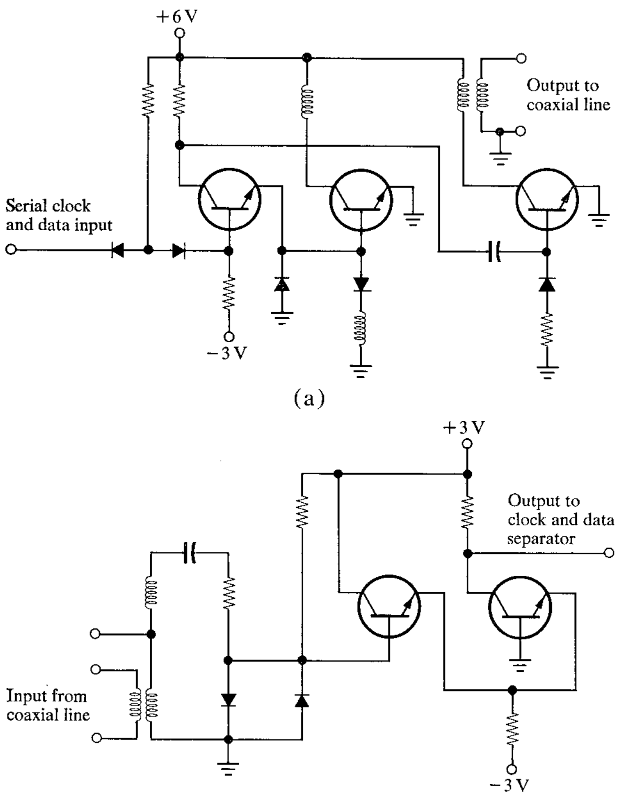


Figure 2 Timing diagram for generation of transmitted data signal.

Figure 3 TCU transmitter and receiver circuits. (a) transmitter, (b) receiver.



The TCU opens the channel for the addressed satellite by issuing a write order and waiting for a response of the same types as are given to a read order.

There is another form of write operation that is useful in some applications. This is initiated by the CPU-CALL command, which is used if ever the central computer wishes to send unsolicited data to the satellite.

To signify the end of a data transfer operation an End-of-Block character is sent at the end of the data stream. The receiving unit, TCU or satellite, acknowledges this by returning a Block Received byte. In the applications that have been implemented so far, the average length of a data block is about 2000 bytes.

After the transfer of data in either a read or write operation has been accomplished, the TCU sets the latch back to the "enable" state and resumes the polling operation until it receives another command from the central computer.

Data modulation

The modulation scheme selected for the system has some of the characteristics of frequency-shift keying modulation[13] in that pulses of different widths are used to represent the binary ONES and ZEROS. The binary ONE is represented by a full cycle of the higher-frequency signal and the binary ZERO is represented by a half cycle of the lower-frequency signal. Thus, the bit cells are of equal length for both ONES and ZEROS and the average dc value for any data pattern is zero.

Data is encoded for transmission in the following way. The transmitter contains two pulse generators, call them A and B, each operating at 2.5 MHz. The pulses are 100 nsec wide and occur at 400 nsec intervals. Signal B is delayed 200 nsec with respect to A. The data in the transmitter is ANDed with signal B and the output of the AND gate is ORed with signal A. The output of the OR serves as the input to a trigger circuit whose output constitutes the transmitted data stream. Thus signal A always produces a transition of the trigger output every 400 nec. When the data bit is a ONE, a transition of the trigger output occurs 200 nsec after the transition caused by signal A. Figure 2 illustrates this operation for a representative sequence of data bits. To synchronize the receiving and transmitting circuits and to guard against false starts due to noise, it is required that transmission begin with at least two ZEROS. The signal transitions produced by this sequence establish the clock times that serve as references during decoding. After the synchronization has been established, a data cell containing a ONE is transmitted as a "start bit" which sets an "allow data" latch in the receiver. In decoding the transmission a data cell is called a ONE if a transition is detected at a time between two consecutive clock transitions, and it is called a ZERO if no transition is detected.

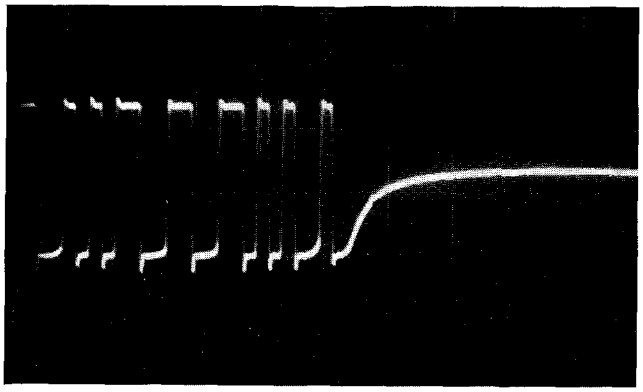
The line driver and receiver circuits that form the interface between the coaxial cable and the TCU are shown in Fig. 3. Transformer coupling is used at both ends of the line to provide isolation and to reduce the effects of common-mode noise and ground-level shift. The receiving end of the line is terminated in a resistive load equal to the characteristic impedance of the line. This minimizes reflections from the receiving end of the line. During transmission the driver end of the line is terminated in the low output-impedance of the driver. This, combined with the characteristic load impedance of the receiver, provides the system with high immunity from noise. The need for such precautions can be appreciated if one is aware that the cabling may pass through an area of the factory containing such high-noise-producing devices as arc welders. Figure 4 gives a typical example of how the pulse stream may be distorted during its passage through a long cable.

At very short cable distances, or when the driver and receiver are connected together for single-wire operation, a trailing transient caused by the discharge of energy stored in the driver transformer can cause spurious signals on the cable. This effect is overcome by requiring a 15 μsec turnaround time between sending and receiving.

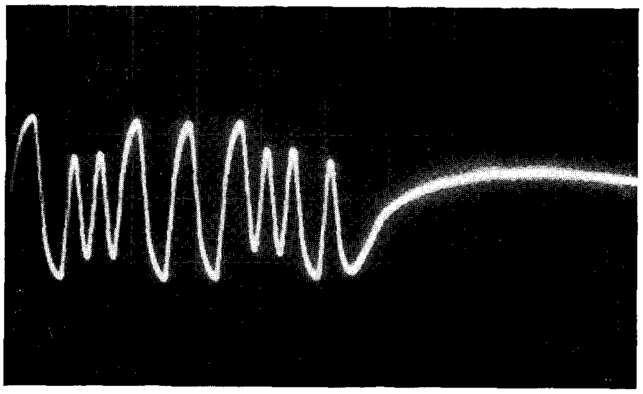
Summary

The system developed for automated testing and process control in some of IBM's manufacturing plants imposed a stringent set of requirements on the device needed to provide the computer-to-computer connection. 1) It had to perform the multiplexing function of establishing a channel between a central computer and any one of several hundred remote satellite computers. 2) It had to permit large bursts of data to be transferred over the channel at megabit rates with a minimum of cabling. 3) It had to provide a means of transferring the data with high reliability in a noisy environment.

The transmission control unit described in this paper satisfies the first requirement by a microprogrammed polling and handshaking scheme that relieves the central computer of the need to use its own storage for this purpose. The TCU satifies the second requirement by using an efficient, synchronous method of two-frequency modulation and by using a serializer-deserializer unit that is shared by all the satellites. And it satisfies the third requirement by using a standard coaxial cable and specially designed driver and receiver circuits that suppress the effects of the noise.



(a)



(b)

Figure 4 Effect of transmission on pulse shape. (a) Pulse sequence produced at transmitter; scale: 1 μsec/div and 5 V/div. (b) sequence obtained at receiver approximately 1 mile away; scale: 1 μsec/div and 1 volt/div.

Acknowledgments

Many people worked on the design and development of the TCU. The author gratefully acknowledges the contributions of all of them, especially A. H. Rall and R. V. Watkins who worked on all phases of the project.

References

1. J. E. Stuehler, "An Integrated Manufacturing Process Control System: Implementation in IBM Manufacturing," *IBM J. Res. Develop.* **14,** 605 (1970), this issue.
2. K. E. Fultz and D. B. Penick, "The Tl Carrier System," *Bell Syst. Tech. J.* **44,** 1405 (1965).
3. J. E. Stuehler and R. V. Watkins, "A Computer Operated Manufacturing and Test System," *IBM J. Res. Develop.* **11,** 452 (1967).
4. E. M. Davis, et al, "Solid Logic Technology: Versatile. High-performance Microelectronics," *IBM J. Res. Develop.* **8,** 102 (1964).
5. IBM System/360 Component Description—2841 and Associated DASD, IBM Form No. GA26-5988.
6. S. G. Tucker, "Microprogram Control for System/360," *IBM Syst. J.* **6,** 222 (1967).
7. IBM System/360 I/O Interface Channel to Control Unit, IBM Form No. A222-6843.
8. D. M. Taub and B. W. Kington, "The Design of Transformer (Dimond Ring) Read-only Stores," *IBM J. Res. Develop.* **8,** 443 (1964).
9. J. Martin, *Teleprocessing Network Organization,* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1970.
10. R. M. Duncan, "A New In-plant Data Communication System," *Telecommunications* **4,** No. 2, 32 (1970).
11. R. O. Hippert, Jr., "IBM 2790 Digital Transmission Loop," *IBM J. Res. Develop.* **14,** 662 (1970), this issue.
12. J. Calva, "PCOS: A Process Control Extension to Operating System/360," *IBM J. Res. Develop.* **14,** 620 (1970), this issue.
13. F. B. Bramhall and J. E. Broughtwood, "Frequency Modulated Carrier Telegraph System," *Trans. AIEE* **61,** 36 (1942).

Received November 26, 1969

Revised July 17, 1970