Westinghouse

ENGINEER

SEPTEMBER 1961

Westinghouse
ENGINEER
SEPTEMBER 1961

Volume 21 / Number 5

TABLE OF CONTENTS

Cover Design The modern high-speed digital computer, the "brain" of a digital control system, does all of its arithmetic with only two digits—1 and 0. The computer is programmed to "think" from a flow diagram of the process being controlled. These elements are used by artist Dick Marsh to form this month's cover design.

Published bimonthly (January, March, May, July, September and November) by Westinghouse Electric Corporation, Pittsburgh, Pennsylvania.

Subscriptions:

United States and possessions	$2.50 per year
All other countries	$3.00 per year
Single copies	$0.50 each

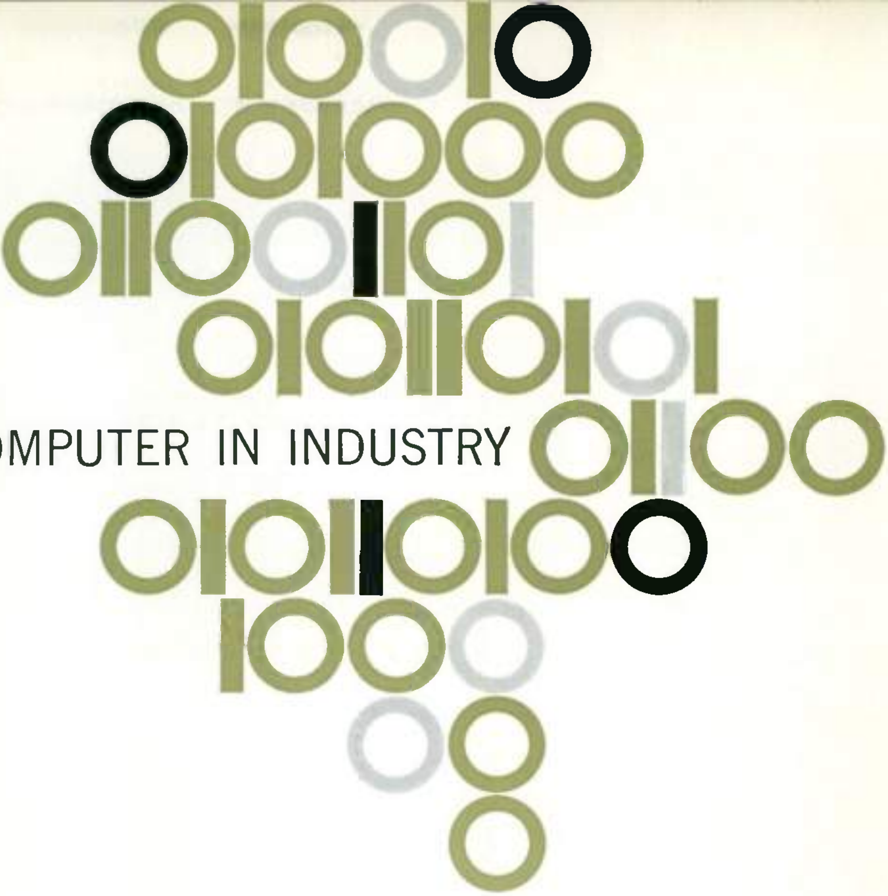
Mailing Address:
Westinghouse ENGINEER
P.O. Box 2278
3 Gateway Center
Pittsburgh 30, Pennsylvania

Microfilm:
Reproductions of the magazine by years are available on positive microfilm from University Microfilms, 313 N. First Street, Ann Arbor, Michigan.

The following terms, which appear in this issue, are trademarks of the Westinghouse Electric Corporation and its subsidiaries:
Prodac, Pulsetter

The Westinghouse ENGINEER is printed in the United States by The Lakeside Press, Chicago, Illinois.

THE CONTROL COMPUTER IN INDUSTRY



The modern digital computer, which has already caused revolutions in business data processing and engineering design, is now creating a similar revolution in industrial control.

The control computer's prime purpose, as its name indicates, is *control*. It has the built-in ability to control the operation of a process—to evaluate operating conditions, to make necessary adjustments or changes, to detect and correct problems, and to otherwise supervise and control the operation. It is, in effect, the director of an entire process.

The control computer has different objectives, is based on different concepts, and requires different approaches than computers whose primary function is information processing. For this reason, we have devoted this entire issue to industrial control computers, with subjects ranging from basic principles to practical application. The coverage is not meant to be all-inclusive, but merely to paint a broad picture of the concepts and application of the digital control computer.

In compiling, reviewing, and editing the material for this issue, we were particularly impressed by two aspects of control computer application. One is the tremendous potential of computer control; the other is the extensive and careful planning that lies behind a successful computer application.

Both of these aspects are related to the nature of the control computer. On the one hand it is an extremely flexible device; it can be "trained" to do many varied and complex control assignments. But on the other hand, the "training" itself is not simple. The computer can do only what it has been told to do, and this means that its program and its memory must contain the best information attainable about the process to be controlled. Even in old and well-tested processes, this frequently requires a thorough study of the process itself as a necessary first step to computer application. Much of the information necessary simply was never required for conventional control.

Overall, the control computer is not "just another control system." While many present installations are based largely on duplicating human actions in the operation of a process, the capabilities of the computer range far beyond this. A second and potentially valuable step is to tailor the process to the computer's abilities rather than to human operation. This is now being done in some applications.

All this assumes, of course, that the process lends itself to control by computer, and many do not. For the present, every potential application must be thoroughly evaluated, and the net benefit determined. Some benefit—such as improvement of product quality or quantity—must be apparent, or the computer is simply not practical.

As evidenced in the following pages, the control computer is already a part of the industrial scene. But there is little doubt that the early applications are merely forerunners of much more extensive installations in the near future.

THE EDITORS

BIT-BY-BIT ARITHMETIC... By a Digital Computer

The decimal number system has been the basis of man's mathematical progress since he first learned to distinguish between one and ten fingers. It was therefore quite natural that man's first digital computers were created in his own image, if for no other reason than to facilitate his communication with the computer. However, he soon found as he progressed in the art that the decimal system is not well suited to large, complicated high-speed computers. Most electronic elements can be most reliably operated in only one of two states—*on* or *off*—which when adapted to a number system means that the computer must be restricted to two digits, 0 and 1. Hence, the binary number system, about as clumsy for human use as Roman numerals, has become the mathematical base for all modern high-speed digital computers. Computers built with ultra-high speed switching devices can do extremely tedious binary arithmetic at incredible speeds.

binary arithmetic

Any number can be expressed in the form:

$$N=d_nR^n+\cdots+d_3R^3+d_2R^2+d_1R^1+d_0R^0$$

where N is the number, d is the digit, R is the *base* or *radix* of the number system, and n is the power of the radix. The number is commonly written as:

$$N=d_n\cdots d_3d_2d_1d_0$$

Hence, in the familiar decimal system where 10 is the radix,

$$173=1\times10^2+7\times10^1+3\times10^0$$

Note that the position of each digit in a decimal number corresponds to a power of 10.

In the binary system, where 2 is the radix, this same value is written:

$$10101101=1\times2^7+0\times2^6+1\times2^5+0\times2^4+1\times2^3+1\times2^2+0\times2^1+1\times2^0$$

To those unaccustomed to the binary system, the value of a binary number is difficult to visualize. To determine the decimal equivalent of any binary number (or vice versa), a power of 2 can be associated with the position of each binary digit in a binary number. Thus:

n	7	6	5	4	3	2	1	0
2^n (decimal value)	128	64	32	16	8	4	2	1
Binary 173	1	0	1	0	1	1	0	1

Hence, the decimal value of the binary number 10101101 is $128+0+32+0+8+4+0+1=173$.

Using the decimal equivalence shown above, any decimal number can be converted to binary form. For example, the binary equivalents of the decimal digits are:

Decimal	*Binary*	*Decimal*	*Binary*
0	0	5	101
1	1	6	110
2	10	7	111
3	11	8	1000
4	100	9	1001

The arithmetical rules for handling binary numbers are similar to those for decimal numbers—except that everything is done with only two digits. The rules for adding and subtracting binary digits are:

$0+0=0$ $\quad$ $0-0=0$
$1+0=1$ $\quad$ $1-0=1$
$0+1=1$ $\quad$ $1-1=0$
$1+1=0$ carry 1 $\quad$ $0-1=1$ borrow 1

Multiplication is accomplished in the same fashion as in decimal arithmetic—by addition and shifting. For example:

$$\begin{array}{r}1001\\ \times101\\ \hline 1001\\ 0000\\ 1001\\ \hline 101101\end{array}\qquad(9\times5=45)$$

Similarly, division is accomplished through subtraction and shifting:

$$\begin{array}{r}1001\\ 101\overline{)101101}\\ 101\\ \hline 101\\ 101\\ \hline\end{array}\qquad(45\div5=9)$$

Complements and Subtraction—In actual practice, many digital computers perform all forms of arithmetic—addition, subtraction, multiplication, and division—with only an adding process. This is done primarily to avoid duplication of digital circuitry. Therefore, the common subtraction procedure must be rearranged so that it can be accomplished with basically an addition process. One convenient method used in binary computers is called the *one's complement* system. Although the following description is not a rigorous explanation of the one's complement method, it does illustrate the basic process performed by a computer.

If the binary number 00101 is in a five-place counter, the one's complement of this number is:

$$11111-00101=11010.$$

Note that the one's complement can be formed by merely interchanging 1's and 0's in the counter. This step is easily accomplished in computer circuits. The subtraction process can then be accomplished in two steps: (1) The *complement* of the number to be subtracted is added to the number to be subtracted from; (2) the carry beyond the counter is added to the least order digit of the above sum. This last step is called *end-around carry.* Consider the example, $11-8=3$:

$$\begin{array}{rl}01011 & (11)\\ +10111 & \text{(one's complement of binary 8)}\\ \hline 100010 & \\ +\llcorner\!\!\longrightarrow1 & \text{(end-around carry)}\\ \hline 00011 & (3)\end{array}$$

This manipulation is easily accomplished with digital circuitry by merely forming the complement of the number to be subtracted, adding, and bringing the overflow (carry beyond five-place counter) around and adding to the least significant place in the counter.

Consider the example, $8-11=-3$:

$$\begin{array}{rl} 01000 & (8) \\ +10100 & \text{(one's complement of binary 11)} \\ \hline 11100 & \text{(no carry beyond 5 places)} \\ -11111 & \\ \hline -00011 & (-3) \end{array}$$

Again, this manipulation is easily accomplished with digital circuitry. When the computer senses no end-around carry for the second step, it merely interchanges 1's and 0's (forms a one's complement) and assigns a negative sign.

binary numbers in a computer

In the digital computer, the ones and zeros required for binary number representation are formed in computer elements where voltage is on or off, magnetization is in one direction or the other, or a pulse is present or absent. Numbers can be transferred within the machine in *serial* or *parallel* form. In serial form, a number is transmitted as a series of pulses through a single conductor, each pulse (or absence of pulse) representing a *bit* (1 or 0) in the binary number. In parallel form, a separate conductor carries each bit in the binary number simultaneously. For the same degree of component sophistication, the parallel form is obviously the more rapid way of handling numbers, and most modern high-speed computers employ parallel systems. In the majority of the examples that follow, the parallel form is assumed.

NOR logic

The NOR computer element, the basic element used in the Westinghouse Prodac control computer, can be used to form any of the arithmetic or logic functions required in a digital computer.

A NOR element has an output only if all input signals are zero. If any input exists, there is no output. This basic element can be constructed with a PNP transistor (Fig. 1). The transistor is employed as a simple two-position switch; with no input, the transistor output is a high-impedance so that the output is essentially at output supply voltage. This normal supply voltage can be used to drive succeeding stages of NOR elements. If an input voltage is applied, the transistor switches from its normal state of cutoff to a state of saturation, with an extremely low impedance output so that the output voltage is essentially at ground level, or zero.

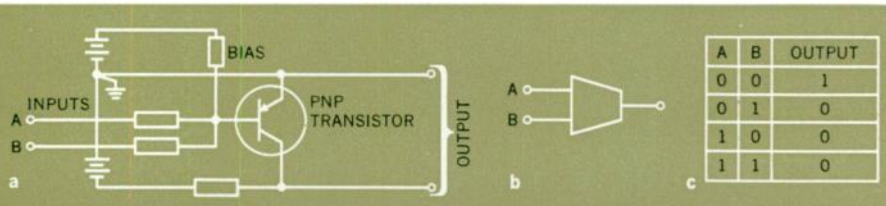
In general, the NOR logic device has more than two inputs so that more complicated decisions can be made. In this particular circuit, a nonconnected input is a zero input to the NOR element. This single logic device can be used in combinations to form the necessary logic functions (Fig. 2).

circuits for digital arithmetic

The arithmetic unit of a digital computer is made up of several types of digital circuits, all of which act under the direction of a central control to accomplish the various arithmetical actions called for.

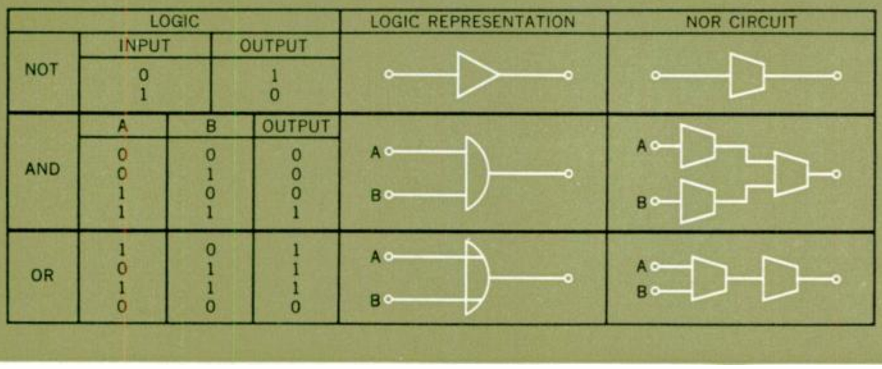
Flip-Flop Circuits—The flip-flop circuit is basic to all binary computer designs; as suggested by the name, the circuit is stable in either of two states. It will remain in one state until pulsed to switch to the other state. This bi-stable nature allows the flip-flop circuit to be used for information storage, or to remember numbers.

A flip-flop circuit constructed of NOR elements is shown in Fig. 3. In a typical operation, the *set* terminal is the input to the device. The *reset* terminal is used to clear the device and set the *output* at zero. If a pulse appears at the input, signifying a one-bit, the *output* terminal will be activated, forming a one-bit output. If the complement of the input number is desired (for example, as previously indicated in the subtraction process), it will appear at the *complement output* terminal. A flip-flop circuit is used to



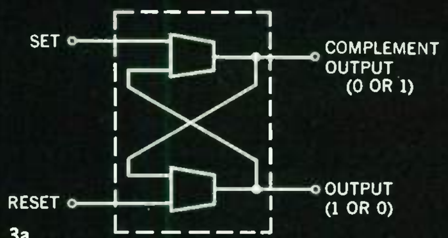
A	B	OUTPUT
0	0	1
0	1	0
1	0	0
1	1	0

Fig. 1 (a) A NOR element can be illustrated with PNP transistor. (b) Symbol for NOR element. (c) Table defining NOR element logic.

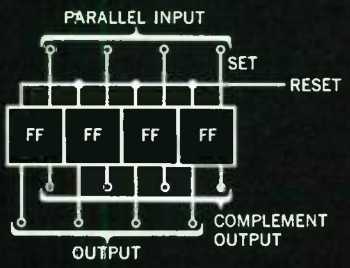


LOGIC			
	INPUT		OUTPUT
NOT	0		1
	1		0
	A	B	OUTPUT
AND	0	0	0
	0	1	0
	1	0	0
	1	1	1
OR	1	0	1
	0	1	1
	1	1	1
	0	0	0

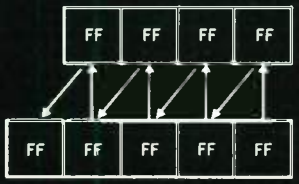
Fig. 2 Typical logic functions are shown along with methods for obtaining these functions with NOR elements. (Unconnected inputs to NOR elements are considered as zero inputs.)



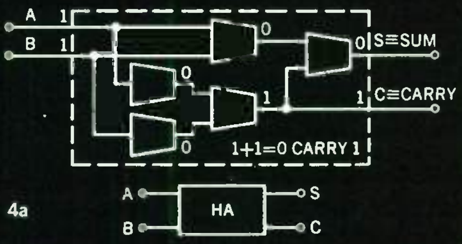
3a



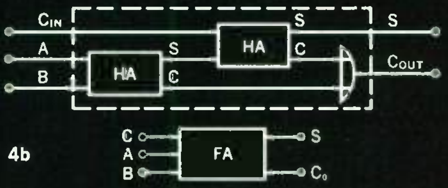
3b



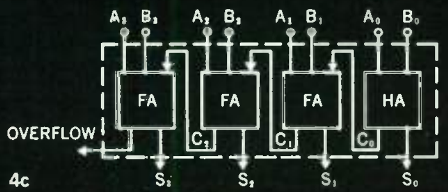
3c



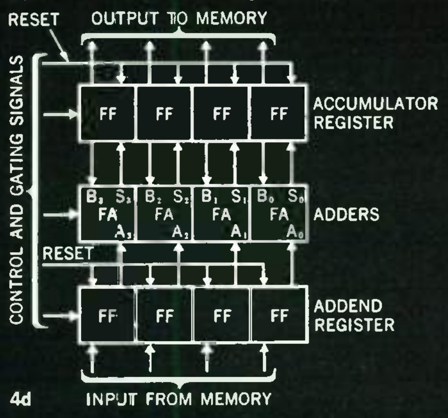
4a



4b



4c



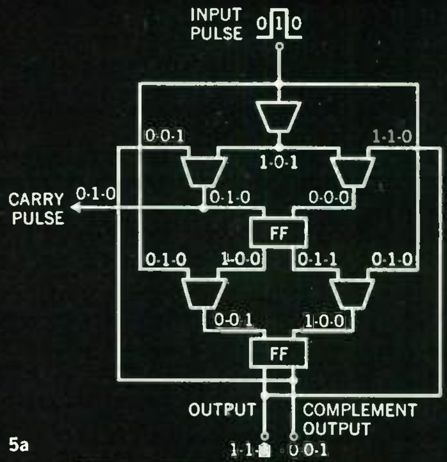
4d

Fig. 3 (a) A basic flip-flop element can be formed from two NOR elements. An input at the *set* terminal produces an output at the *output* terminal; an input at the *reset* terminal produces an output at the *complement output* terminal (and eliminates the output at the *output* terminal). (b) A four-bit parallel flip-flop register for holding binary numbers. Both the number and the number's complement can be read from the register. (c) Two flip-flop registers can be used for shifting.

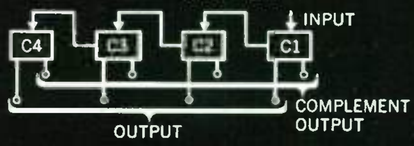
Fig. 4 (a) A basic half-adder circuit formed from NOR elements. This circuit will perform basic binary addition and carry. The addition 1+1=0 carry 1 is shown. (b) A full adder, which can accept carry from the previous less significant digit position adder, can be formed from two half adders. (c) A four-bit parallel adder with overflow. (d) In this simplified arithmetic unit, two flip-flop registers hold the numbers to be added. The numbers are fed from the *accumulator* and *addend* registers into the *adders*, and the sum transferred back to the accumulator register.

Fig. 5 (a) A counter can be formed from NOR elements, used as gates, and two flip-flop elements. The input pulse can be thought of as 0 in the initial condition, 1 during the pulse, and a return to 0. The initial condition of the NOR elements and flip-flop elements is indicated by the left-hand digit at each location; during the pulse input (1), element conditions are illustrated by the middle digits; after the pulse input (0), element conditions are shown by the right-hand digits. In the example shown, the counter is assumed to have an initial 1 output, so that the input pulse changes the output to 0 and produces a carry pulse. (b) Counter elements are arranged to produce a binary output count of input pulses. (c) The relative positions of the four counters is shown for 12 input pulses.

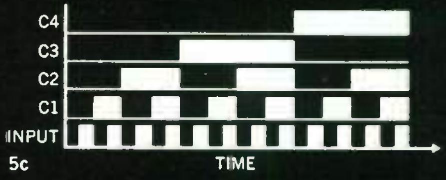
Fig. 6 (a) The hysteresis loop of a magnetic core is such that the core tends to stay magnetized in either one direction or the other. (b) Each magnetic core has essentially four windings: the *x*- and *y*-selection windings and the inhibit winding are input windings, and the sense winding is an output winding. (c) A core memory plane consists of a group of cores mounted on *x-y* selection wires (insulated). The sense winding and the inhibit winding (not shown) are threaded through all cores in the plane. (d) In the usual computer memory system, the sense winding from each plane produces one bit; the planes are stacked to produce memory words of several bits. Hence, the *x-y* selection wires select a column of cores.



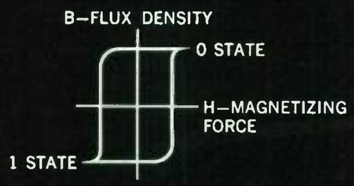
5a



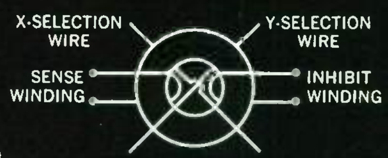
5b



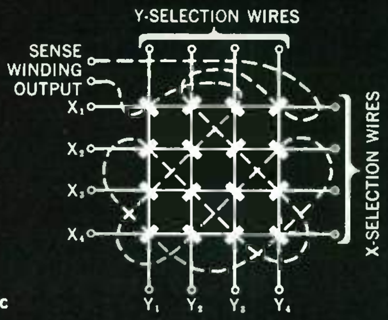
5c



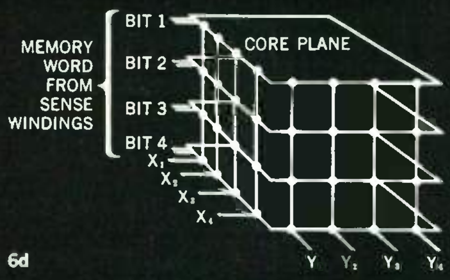
6a



6b



6c



6d

store (remember) each bit in a word (Fig. 3b). Besides its use in the arithmetic unit for temporarily holding numbers being operated upon, flip-flop registers are used in many *buffering* stages throughout the computer for holding numbers between various computer operations until needed.

The shifting mechanism needed for performing multiplication and division can also be accomplished with flip-flop registers. Using two registers (Fig. 3c), a number to be shifted is transferred in parallel from one register to a second, and then transferred in parallel back into the original register shifted one place over. This operation can be repeated as many times as necessary to get a number shifted the desired number of places.

Half-Adders and Full-Adders—Adder circuitry performs basic addition in a digital computer. One method for adding two binary bits simultaneously to form a sum and a carry bit is shown in Fig. 4a. This is a *half-adder*, made up of NOR elements, and will perform basic bit addition and carry. A *full adder*, which can handle input carry bits, is composed of two half-adders, as shown in Fig. 4b. A full adder is required for each bit, and for parallel adding would be connected as shown in Fig. 4c.

A simplified version of an arithmetic unit for adding would consist of two flip-flop registers (an accumulator and an addend register) and an adder, as shown symbolically in Fig. 4d. These elements would be controlled from a central control unit through gating devices. In a typical addition process, one of the numbers to be added would be held in the accumulator; the second number would be placed in the addend register. When the control signals the arithmetic unit to add, the numbers in the accumulator and addend registers are fed into the adding element, the addition process performed, and the sum returned to the accumulator.

An actual arithmetic unit would need provisions for subtraction, additional registers for performing the shifting functions needed in multiplication and division operations, and provision for carrying number sign.

counters

Basically a counter translates pulses in serial form into a binary total. A counter with two flip-flop circuits and additional NOR elements used as gates is shown in Fig. 5a. If a 0 output is assumed, an input pulse changes the output to a 1; the next input pulse changes the output to 0, and produces a carry output pulse. These devices, when connected as shown in Fig. 5b, count input pulses in a binary fashion.

Counters are often used in digital circuitry as frequency dividers. For example, in Fig. 5c, 12 input pulses would produce 6 output pulses in the first counting stage, 3 pulses in the second counting stage, etc.

Another typical counter application is the *sequencing register*, where the binary number in the register is increased by one each time a pulse enters from the computer timing clock.

computer memory

All digital computers must have the ability to store information. In control computers especially, this stored information must be immediately available. Magnetic-core storage systems have by far the widest use in high-speed computers because of the extremely rapid and random access possible. Other widely used forms of memory are magnetic drums and discs, where magnetized spots on a magnetic surface provide the stored information. However, since reading and writing heads must be mechanically positioned over the area where information is stored, access to these forms of memory is slow by modern computer standards.

Magnetic-Core Storage—Magnetic-core memory systems employ many small toroidal magnetic cores, each having a square hysteresis loop characteristic as shown in Fig. 6a. The flux level in the core can be in one of two states—fully saturated in one direction or the other. Thus, the core is in effect a "flip-flop" element. When magnetized in one direction, it is in the "1" state; in the opposite direction, the "0" state. An arbitrary relationship is illustrated in Fig. 6a. A core can be flipped from one state to the other only if the full magnetizing current is applied to the input windings. If it is flipped, the flux level in the core changes, to produce a pulse (by transformer action) in the output winding.

Each magnetic core has four windings (Fig. 6b)—X and Y *selection* lines, a *sense* winding, and an *inhibit* winding. In a typical *coincident-current* arrangement (the most common technique used in large computer memory systems), cores are held in a plane array on X and Y selection lines (Fig. 6c). A sense winding and an inhibit winding are threaded through all cores in the plane. To select a particular core in the plane, for example X_2Y_2, a pulse current of half value is applied to terminals X_2 and Y_2, so that only core X_2Y_2 will receive the full pulse current necessary to change its state.

A complete core memory system consists of many parallel planes of cores, each plane representing a single bit of a binary number (Fig. 6d). Thus, instead of a single core, an X_2Y_2 column of cores is selected. The sense winding from each plane provides the bit information from the selected core in the plane.

To obtain bit information from a core in each plane, half current is applied to the X and Y selection lines. If the selected XY core is in the "1" state, the core magnetization is flipped, and a pulse is generated in the sense winding. If the core is already in the "0" state, no pulse is generated. These binary bits from the core plane sense windings are stored in parallel in a memory buffer register. Since reading the information transforms all read cores to the "0" state, the information must be re-entered into the core if the number is to be retained in storage for further use. This re-entry is accomplished by pulsing the X and Y selection wires with half currents in reverse, which flip each selected XY core back to a "1." To prevent a "1" from being stored where an original "0" existed, a positive half-value current is applied to the inhibit winding of each plane where a 0 originally existed. The generation of inhibiting voltage is directed by the memory buffer register where the readout number is being held.

To place a new word in memory, a similar sequence is employed. The number to be stored is first placed in a memory buffer register. A half current is applied to the X and Y selection lines, putting all cores in the selected word location in the "0" state. A negative half current is then applied to the X and Y selection lines, and a positive half current is put in the inhibit winding of each plane where a "0" is desired, as directed by the buffer register.

Address Matrix—The memory address matrix is the link between the computer control and the computer memory. The control can call for stored information in the memory by sending an address (in the form of a binary number) to the memory address matrix. The address matrix decodes the address number and thereupon energizes the desired X and Y selection lines in the memory. One method for accomplishing this function can be illustrated with a *diode decoding matrix*.

The basic logic element in a typical address matrix is shown in Fig. 7a. A voltage output occurs only if a negative voltage input exists at both A and B. This logic element (negative logic AND) is connected in an array of crossed insulated wires, as shown in Fig. 7b. Matrix input is supplied from both output and complement output of a flip-flop register. As illustrated in the accompanying table, any given X output line can be energized by supplying a correctly coded input. With such an X- and Y-selection matrix (Fig. 7c), the desired X and Y selection lines in the memory can be energized.

computer control

The computer control unit (in simplified form) consists essentially of a clock, a sequencer, an instruction-decoding matrix, and control circuitry.

The clock (a high-frequency oscillator) supplies a continuous chain of pulses; these pulses are used to gate (or lead) signals through the computer with a proper time relationship to each other. This allows the computer processes to proceed in an orderly fashion without regard for variations in the speed of individual NOR elements.

The sequencer (Fig. 8) in its simplest form can be thought of as a form of counter that counts clock pulses, and at a predetermined count provides action-initiating signals to the control circuitry. The sequencer goes through a pattern of instruction signals (P_0 to P_4 in Fig. 8) and then resets itself to repeat the sequence.

The instruction-decoding matrix decodes binary instructions from the stored memory for the control circuitry. The control circuitry accepts signals from the instruction decoding matrix (or the sequencer), and initiates the proper control signals to the other elements in the computer. This over-simplified explanation can be illustrated by tracing through a typical *add instruction cycle* in the computer shown in Fig. 9.

Add Instruction Cycle—Assume that one of the numbers to be added is already in the *accumulator register* of the arithmetic unit from a previous instruction cycle.

The first pulse (P_0) from the control sequencer is applied through the control circuitry to the *instruction counter*, causing the number in the instruction counter to advance one binary digit.

The second pulse (P_1) from the sequencer causes a new instruction to be transferred to the *instruction register*. This is accomplished in the following sequence: (1) The number in the instruction counter, which is a new memory address of the form $A_xB_xA_yB_y$ is transferred to the *memory address matrix*. (2) The *memory* content (an instruction word) specified by the address matrix is read out to the *memory output*. (3) Memory output gates guide the transfer of the instruction word from the memory output to the instruction register. This instruction word obtained from memory consists of two parts: The first part of the word is another memory address (again of the form $A_xB_xA_yB_y$), and the second part of the word is a coded instruction for the control circuitry (for example, an add instruction might be 1010, a subtracting instruction 0101, etc.).

The third sequencer pulse (P_2) causes the *addend register* of the arithmetic unit to be loaded. This is accomplished as follows: (1) The address portion of the instruction in the instruction register is transferred to the memory address matrix. (2) The contents of the memory location specified by the address matrix is read out to the memory output. (3) Memory output gates guide the transfer of the information from the memory output to the addend register of the arithmetic unit.

The fourth sequencer pulse (P_3) causes the control instruction part of the word in the instruction register to be transferred to the control's *instruction-decoding matrix*, where the instruction is decoded and tells the control circuitry to issue instructions to the *arithmetic unit* to prepare for an add operation.

The fifth pulse (P_4) from the sequencer initiates the add sequence in the arithmetic unit. The contents of the addend and accumulator registers are fed into the adding circuitry, and the sum placed in the accumulator, ready for the next cycle. The P_4 pulse also resets the sequencer for the next instruction cycle.

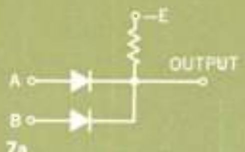
computer input and output

The principles described thus far apply to any type of digital computer—data processing or control. A distinct difference between data processing computers and control computers comes in getting information in and out of the computer. Data process computers are often referred to as *off-line* computers because information can be fed into the computer, processed, and read back out at the convenience of the computer operator. However, a control computer must keep in exact step with the process it is controlling. It is therefore an *on-line* computer, and must be able to accept information directly from the process, make the necessary computations, and issue appropriate control instructions back to the process with a minimum of delay.

Information from most processes is initially obtained in analog form—a voltage from a thermocouple, a shaft position, a pressure measurement, etc.—and must be converted into digital form. Two types of analog-to-digital converters are illustrated in Fig. 10. One counts clock pulses while a saw-tooth voltage wave is rising to the analog value; the other uses a diode matrix to select the binary value corresponding to each position (bar) of the commutator. After the necessary control computations are made by the digital computer, control instructions are issued back to the process. Methods for issuing on-off control instructions to the process, or generating an analog control voltage are illustrated in Fig. 11: A method for obtaining contact closure outputs is shown in Fig. 11a; Fig. 11b shows how these same relays could produce an analog voltage output.

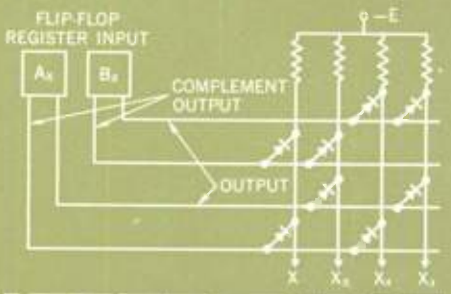
digital process control

The operation of a digital computer in controlling a process can be illustrated by considering the simple hypothetical



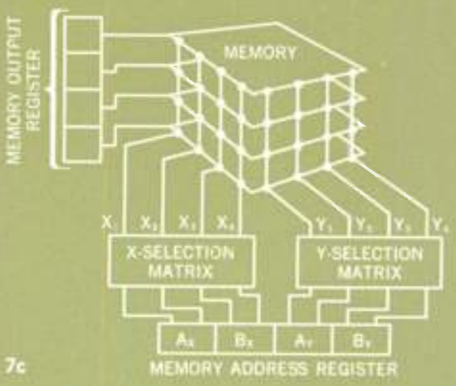
A	B	OUTPUT
0	0	0
0	1	0
1	0	0
1	1	1

7a

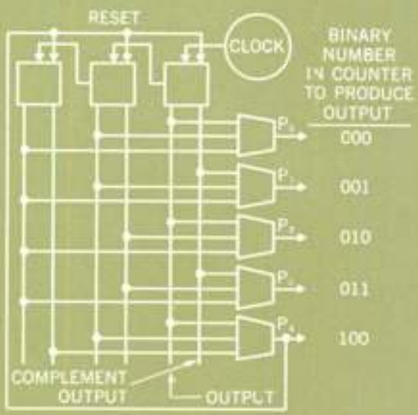


A		B		X_1	X_2	X_3	X_4
OUTPUT	COMP. OUTPUT	OUTPUT	COMP. OUTPUT				
0	1	0	1	1	0	0	0
1	0	0	1	0	1	0	0
0	1	1	0	0	0	1	0
1	0	1	0	0	0	0	1

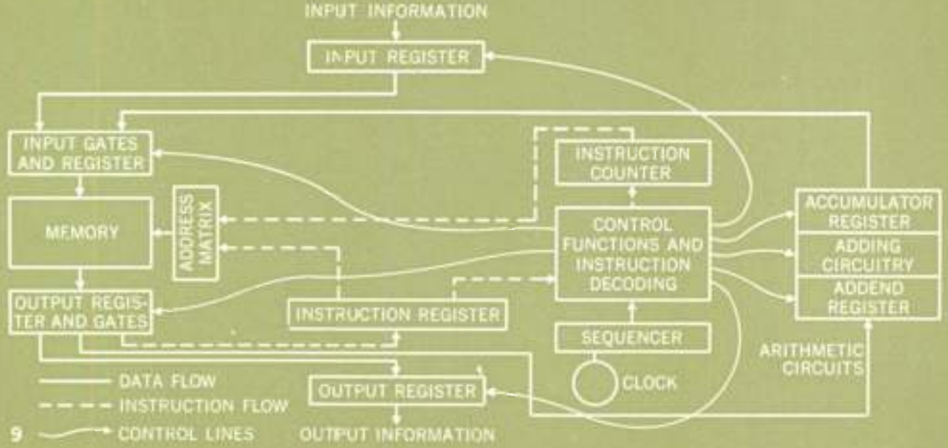
7b



7c



8



9

Fig. 7 (a) Diode circuit for producing negative logic AND. Both inputs *A* and *B* must be present to produce output, as shown in the table. (b) The diode matrix is one form of decoding matrix. As shown in the accompanying table, an *x*-output wire can be energized by providing an *AB* input word. (c) This functional diagram illustrates the method of selecting words from memory. The memory address word is placed in a memory address register (some form of flip-flop register); this register energizes the *x*- and *y*-selection matrices, which cause the word to be selected from memory and placed in a memory output register.

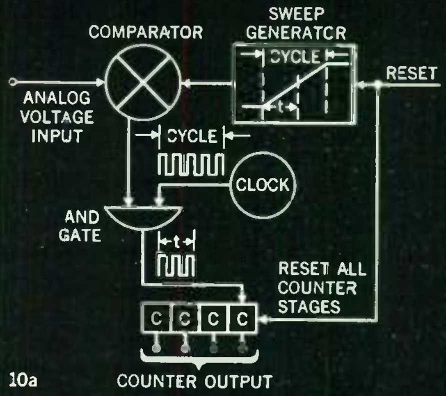
Fig. 8 The principle of a sequencer can be illustrated with a counter (Fig. 5) and NOR gating elements. Since the NOR element has an output only when all inputs are zero, P_0 output will exist for count of 000, P_1 for 001, etc. The NOR output pulses provide the control with a regular, repeatable sequence of pulses.

Fig. 9 Diagram of a computer, using elements that have been described.

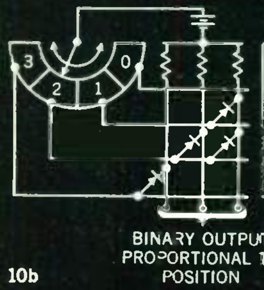
Fig. 10 (a) The operation of an analog-to-digital converter can be illustrated with this simple example. The analog voltage input is compared with a sweep-generator voltage, and the comparator produces an output until the sweep generator voltage equals the analog input, at which time comparator output stops. This stops the clock pulses from reaching the counter, so that the binary number in the counter is proportional to the analog input voltage. The count starts again when the sweep generator and counter are reset. (b) An analog-to-digital position converter can be illustrated with this simple diode matrix encoder. The digital output is shown in the table.

Fig. 11 (a) The output from a digital computer can be applied to relay operating coils. These coils can be used to energize closing and opening relays on process equipment, or (b) the coils can control switches in a digital-to-analog converter, to generate an analog voltage.

Fig. 12 This hypothetical control situation is used to illustrate a digital control function. The problem is to keep fluid temperature constant by varying fluid flow. The temperature (*x*) is used to control the valve position (*y*) through the assumed relationship, $y = Ax + B$.

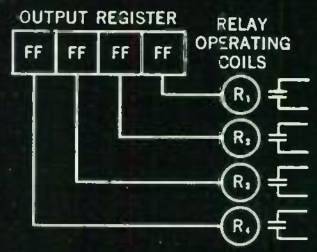


10a

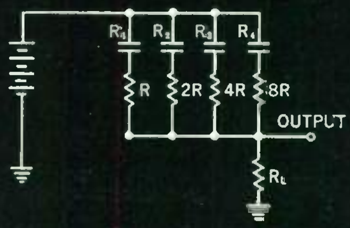


INPUT POSITION	OUTPUT
0	000
1	001
2	011
3	100

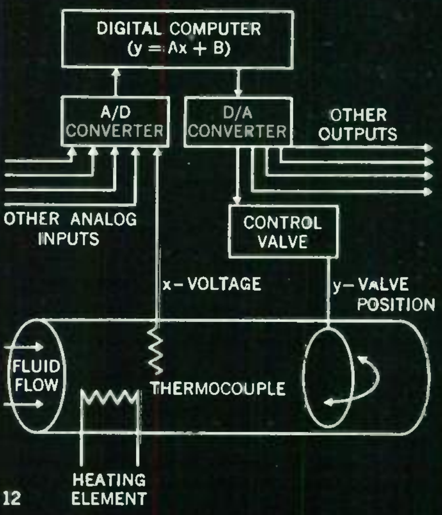
10b



11a



11b



12

control operation shown in Fig. 12. This control problem assumes that a fluid is being heated as it flows through a pipe. Fluid flow (and therefore temperature) is regulated by a controlled valve. For the purposes of this example, the relationship between fluid temperature and flow is assumed to be:

$$y = Ax + B$$

where y is the desired value of fluid flow in terms of a voltage to the controlled valve, x is the measured temperature of the fluid, and A and B are constants stored in the computer memory.

In this simplified situation, the computer will obtain a value of x from an analog-to-digital converter, perform the indicated arithmetic, and feed the calculated control voltage back through a digital-to-analog converter to the valve. The process can be accomplished as follows:

(1) To actuate the analog-to-digital converter (ADC), the control unit obtains a control word from memory and supplies it to the ADC unit, telling it where to place the read value of x in memory. One ADC unit will handle a number of inputs so that the control word must provide the ADC unit with additional instructions, such as how to set up the multiplexing, select the ADC gain range, etc.

(2) The arithmetic operations involved are performed by the computer:

- (a) The value of x is taken from memory and placed in the multiplier register of the arithmetic unit.
- (b) An order to multiply is given (somewhat similar to the *add* cycle previously described). The value A is taken from memory and transferred to the arithmetic unit, and the product Ax formed in the accumulator register.
- (c) An add instruction cycle is performed (as previously described), in which B is taken from memory and added to Ax. The result ($y = Ax + B$) is now in the accumulator register.

(3) The computed value of y in our simple computer (Fig. 9) is read from the accumulator register into memory, from memory to the output register, and from the output register to a digital-to-analog converter, which forms the analog voltage to control the valve.

(4) In a typical control computer, the next instruction would instruct the analog-to-digital converter to select another value in the process, and another control operation would be performed.

A large control computer can make hundreds of checks and calculations per second, each similar to the control operation described. Analog input units would scan the various sensing points throughout the process being controlled and convert these analog indications into digital signals for computer use. Incoming signals would be multiplexed—connected one at a time for a small fraction of a second—so that one analog-to-digital converter can handle hundreds of inputs.

The computer periodically compares these analog input quantities with upper and lower limits, or makes simplified calculations such as those described, and then takes appropriate actions.

error detection

A large-scale control computer contains thousands of transistors, diodes, or other types of static devices. And though the probability of success of each individual component may be near unity (0.99999···), the probability of success of the total will be reduced by the relationship:

$$P_0 = P_1 \times P_2 \times P_3 \times P_4 \times \cdots \times P_n$$

where P_0 is the overall probability of success, and P_n is the probability of success of each component. Therefore, means must be built into a computer to detect a failed device, and to locate it quickly. This self-checking feature is particularly important in control computers.

Parity Check—The parity check is one of the most common methods used for checking for correct transmission of information throughout the computer. A typical system is to assign one extra bit, called the ***parity-check bit***, to each word. If the word contains an even number of 1's, the parity-check bit will also be a 1; if the word contains an odd number of 1's, the parity-check bit will be a 0. Hence, the total word, including the parity check bit, will always have an odd number of 1's; any single bit error in transmission will destroy this parity, causing appropriate action to be taken. The computer may be programmed to try to correct itself, or to notify the operator of the error.

Diagnostic Routine—Diagnostic routines are usually provided for each unit of the computer system so that computer operation can be periodically checked, or checked in the event of an error detection. Diagnostic routines can be permanently stored in the computer memory so that the computer can switch itself to the appropriate routine. These programs can be designed so that the operator can quickly locate any malfunctioning unit.

A large control computer will have many additional built-in checking and fail-safe features to insure quick detection of an error, and a minimum of down time.

computer or control system?

Digital control ranges from single-variable control using digital devices through card-programmed sequencing control up to control systems using stored-program digital control computers. The distinction between an advanced digital control system, and a digital ***computer*** control system is sometimes not clear. In fact, a considerable overlap exists.

Perhaps the most fundamental difference is that a control computer is programmable within the memory of the computer, whereas a control system without a computer is not. This means that computer control can be altered merely by changing the computer instruction program within the stored memory of the computer, whereas in a computerless control system, actual wiring changes in circuitry are required. Therefore, the designer applying a computer can rely on flow charting and programming techniques, whereas the designer of a control system must apply the digital equipment as components, using these components to solve individual switching and logical functions, as required in the application. The distinction between a control computer and a control system will probably tend to become even less clear as their application progresses.

Westinghouse
ENGINEER
Sept 1961

This article was staff written by M. M. Matthews. We would like to thank the following persons for their help: Dr. Ruth Goodman and Dr. Terry Jeeves, Research Laboratories; Loren R. Collins and John R. Ball, New Products Laboratories; and Richard T. Byerly and Paul E. Lego, ASE&A Department.

DIGITAL CONTROL FUNCTIONS

E. P. ROSS and F. G. WILLARD
Systems Control Department
Westinghouse Electric Corporation
Buffalo, New York

The advent of solid-state switching devices has so broadened the capabilities of digital control as to make it seem like a new art. Actually, many traditional control devices are digital in nature—the light switch, for example, and industrial relays and contactors. But although these electromechanical devices are extremely useful, all have limitations for extensive use in control systems because of their bulk, power requirements, limited life, and slow response.

Solid-state devices, being relatively free of these limitations, have freed engineers to design vastly more complex functions than could be incorporated in earlier controls.

These modern digital controls are extremely varied in type and scope, ranging from simple position regulators to complex control systems using computers. However, they have in common several major functions, which are used in different combinations to achieve the desired system.

basic digital control functions

Instrumentation—This is the detection of quantities, qualities, and events pertaining to the process under control and the apprising of the control system of these quantities, qualities, and events. There are two basic classes.

Proportional instrumentation measures such quantities as speed, voltage, current, temperature, position, pressure, and flow. Sometimes the quantity is converted directly into digital form, as when a pulse generator and counter are used together to indicate the position of a rotating shaft.

In other cases, the instrumentation first converts the quantity into an analog electrical signal—for example, a pyrometer output. An analog-to-digital converter then converts the signal to digital form, usually by comparing the analogous quantity with a set of reference standards (such as discrete voltage levels) in the circuitry of the converter and identifying the standard that most nearly equals the analogous quantity.

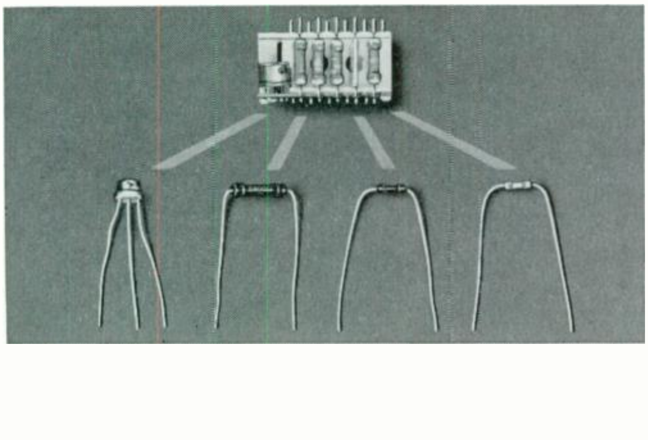
The quality of an instrument employing analog-to-digital conversion can be described in terms of its proportionality or linearity, the stability of its reference standards, the sensitivity of its comparator, and the number of different comparisons recognizable. Speed of response is important also because most such instruments compare reference-standard combinations in sequence, each comparison requiring a finite time.

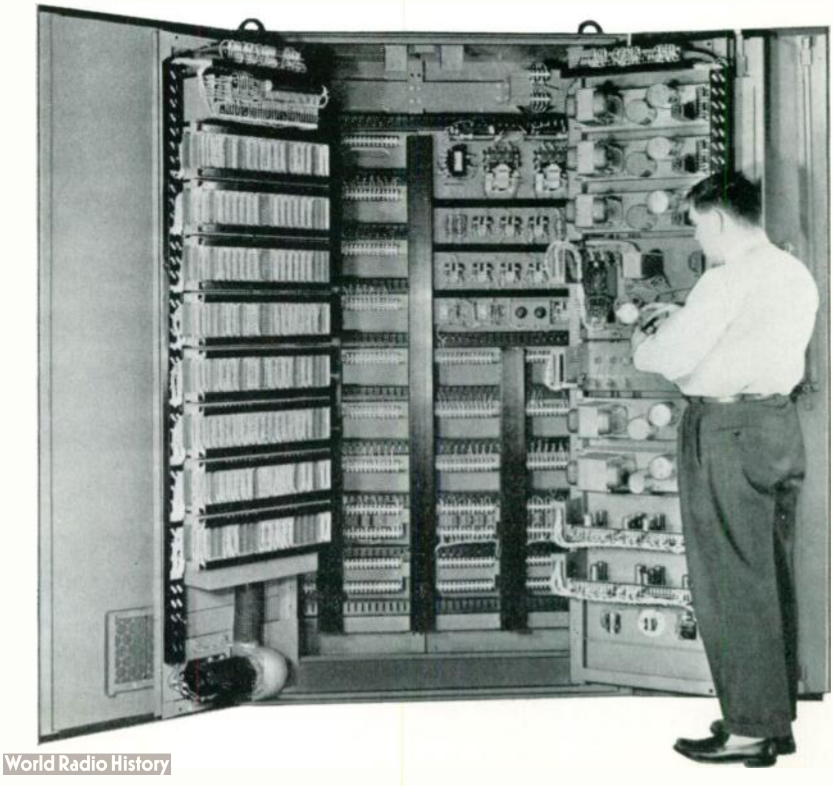
Status instrumentation merely provides a yes-or-no statement of some quality, quantity, or event, so it is simpler than proportional instrumentation. Examples are detection of over-speed, proximity, under-voltage, surface-finish defects, and workpiece breakage. Status instrumentation converts data to a readily measured analogous quantity, compares this quantity with a single reference standard, and provides a signal yes or no (implying comparison or noncomparison).

Information Input and Storage—Information *input* is the provision, by an operator, of data that modifies the control system's effects on the process under control. The data includes statements of desired process quantities, selection of operational modes, notice of manual intervention, and de-

Below This NOR element, consisting of a solid-state switch and input resistors, is a basic building block in digital control equipment. Functional modules made by combining standardized NOR elements, capacitors, and resistors form such components as counters, shift registers, timers, and gates.

Right A typical digital control unit, showing how the functional modules are combined in accessible arrays. Such control units, often used in groups, make up programmers, regulators, data loggers, and computers.





scription of initial material quantities. Pushbuttons, switches, punched-card or tape readers, and telephone dials are typical kinds of input apparatus.

The primary requirement for information input apparatus is that it be compatible with a human operator and with the control system. It must transmit information, directly from the operator or by means under his control, to the control system with minimum likelihood of error. Because the operator often has to focus his attention on the process during the transmission, the input means must be simple and foolproof.

Storage is the saving of input information and instrumentation signals for use some time later than their input. It is accomplished by such devices as electronic flip-flops, magnetic core memories, and magnetic drums.

Storage apparatus must copy information faithfully and hold it reliably until it is needed. Since the information commonly consists of one or more sets of on-or-off voltage signals, storage devices that are inherently capable of two stable states are usually selected and one such device used for each on-or-off signal.

Logic and Sequencing—Logic is the recognition of combinations of events that are pertinent to the control of the process. Its primary requirement is the ability to evaluate incoming on-or-off signals and from them to derive an on-or-off response in accordance with a predetermined rule or logical equation. Such rules can be formulated in terms of the logical connectives AND, OR, and NOT. For example, the rule might be "Provide an *on* output if (and only if) inputs W OR X, AND NOT Y, are *on*." Alternatively, the formulation may be in terms of the connective NOR, where the same example becomes "Provide an *on* output if (and only if) neither input Y NOR Z is *on*, where input Z is *on* if neither W NOR X is *on*." Since the word "NOR" appears twice in this example, the function can be performed with a circuit consisting of two NOR elements.

Protective interlocking of the process machinery, first-come-first-served interlocking of various operator inputs, detection of the mutual completion of several simultaneous process events, and recognition of various operating modes are typical jobs performed by the logic function.

Sequencing is the control of the order and time at which process events are made to occur. It consists of a memory to note which events have occurred, time delays to establish the duration of events, and means for selecting the next event required. The memory devices usually are transistor flip-flops, ring counters, and shift registers; special transistorized RC-circuit timers commonly make up the time delays. Successive events are selected by a programmed chain of operations, performed by such devices as ring counters and shift-register chains, that may be modified by feedback information.

When instrumentation is provided to detect completion of an event, feedback often replaces the time delay and the sequence control is then said to be asychronous. The steel-mill sequence control is an example. At some point in the sequence, drive motors are started to move a hot ingot some distance along the mill tables. A hot-metal detector, so placed as to "see" the ingot when it has been moved far enough, notifies the sequence controller that the event (ingot movement) is completed, and the controller then stops the table drive motors and proceeds to the next event.

Typical sequencing tasks are start-up sequences for large machines, motion sequences for auxiliary process drives, process coordination, and modeling of material flow through a process. ("Modeling" is electrical simulation of a process to keep the control system aware of the stage of the process.)

Logic and sequencing is the area in which digital control is most outstanding. The complex decision-making ability of large groupings of identical simple circuit devices is so great that a control system can now be built to operate any process for which adequate instrumentation exists.

Arithmetic—This function consists of the mathematical operations of addition, subtraction, multiplication, division, counting, comparison, numerical integration, differentiation, and correlation. Arithmetic plays at least a minor role in most digital control systems, since process quantities are usually among the controlled variables.

For example, digital regulators require subtraction (e.g., to determine difference between a drive feedback position signal and a reference signal), comparison (e.g., to compare an actual quantity with a desired quantity), and counting (e.g., to count the number of events that have occurred). Division is used in speed-control systems that divide a master frequency to produce a controlled frequency.

In most systems, the arithmetic operations are synthesized or approximated by counters, adding circuits, and shift registers. The basic circuits are often constructed of the same devices used for logic and sequencing.

Arithmetic operations are also performed by digital computers, but a distinction is usually made between computers and digital control systems that include the arithmetic function. In the control system, the arithmetic operations, the order in which they occur, the source of numbers upon which operations occur, and the destination of results are fixed. Furthermore, the primary purpose of performing the arithmetic operation is to control a portion of the process directly. A computer performs arithmetic operations in a more general and flexible manner.

Digital Regulation—This is the regulation of a drive by digital means to provide faster and more accurate regulation than can be achieved by analog means. Speed and position regulation probably are the most common, but any quantity for which digital instrumentation exists can be controlled.

Most digital regulators are basically similar in that they incorporate the functions of digital position instrumentation, subtraction, decoding, and analog signal generation. Their principles can be illustrated by describing the position regulators used by Westinghouse in steel mill control.

In these devices the position instrumentation (say for a screwdown drive) consists of a position transducer that produces a pulse train as a function of shaft angle, and a counting device that integrates pulses to provide a number directly proportional to position. The resolution is limited only by the capacity of the counter, each additional count stage doubling the resolution. Subtraction, performed by a binary subtraction circuit, determines the difference between actual and desired position. The difference indicates both direction and magnitude of motion required to bring the controlled drive to the desired position. A digital-to-analog converter decodes the difference signal and generates an analog signal to supply intelligence to the controlled

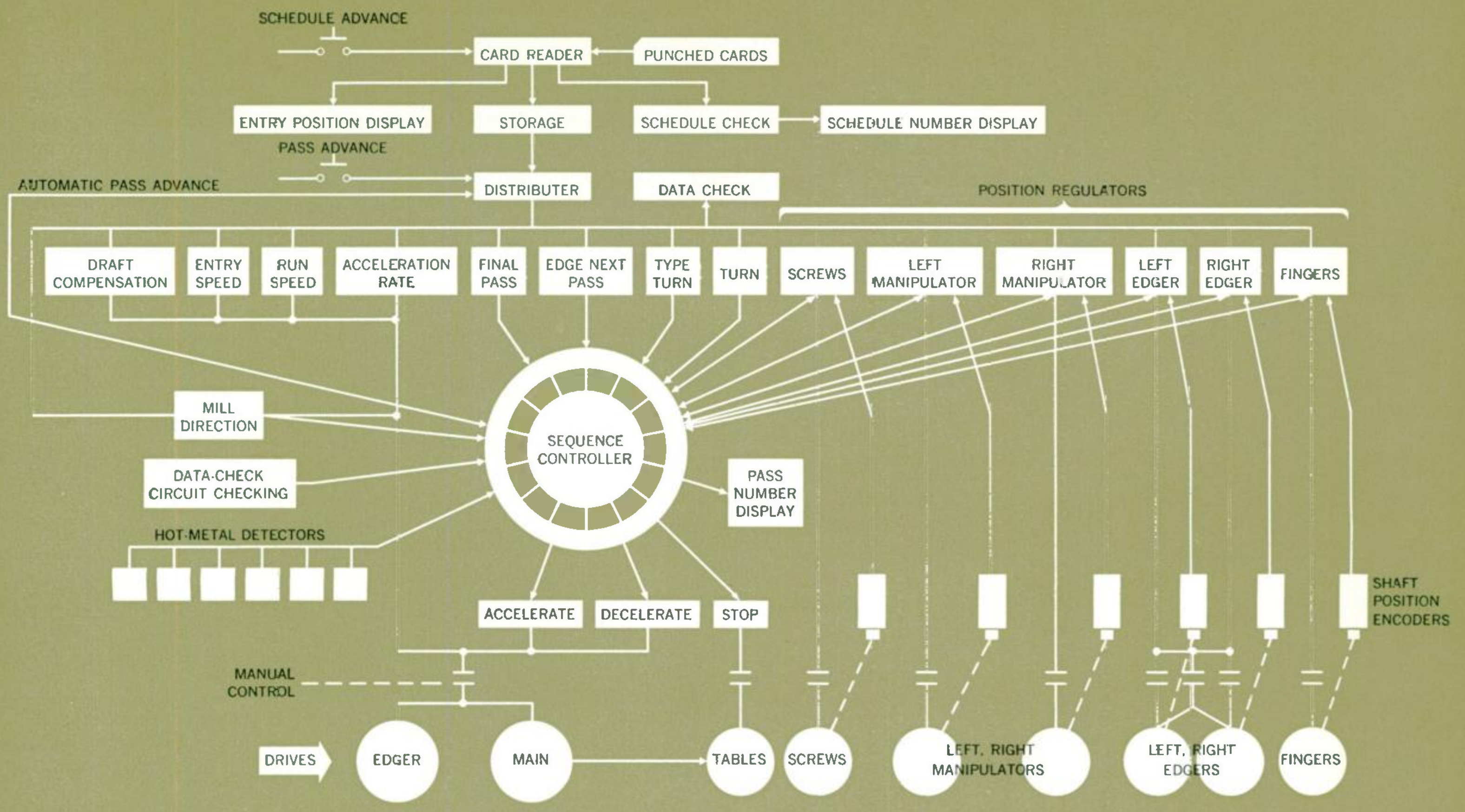


Fig. 1 Schematic diagram of automatic rolling mill.

drive. The drive then moves at maximum velocity in the proper direction and slows at maximum deceleration to arrive on position in the shortest time possible within the capacity of the machines.

Many of the unique capabilities of digital control systems stem from this function. Digital regulators have great precision, excellent response, and near-perfect repeatability. They make it possible to meet many performance requirements that would be too stringent for other controls.

Information Output—This is the communication of information to people. Some digital systems (called data-accumulation systems) produce only information. All others, even those intended primarily to control processes directly, form many useful information by-products. Therefore, the systems require means for presenting data in understandable form.

The two broad classes of information output are alarm signals and descriptive signals. The former are often audible or visible on-or-off signals that occur only when a malfunction is detected. Such signals require operator attention, so they often are simply set by the system and reset by the operator.

Descriptive signals are generally more complex in information content. They are displayed for immediate operator attention, stored as documents for later perusal, or both. The data usually is in raw form, unprocessed, but some of the more sophisticated data-accumulation systems correlate and summarize data. The degree of data processing and type of output apparatus vary with the application.

In general, information output ultimately becomes a process feedback, either through immediate operator reaction or by its influence on managerial decisions regarding the process. The latter use of information output offers the possibility of totally coordinated plant production control.

applications

Digital control systems using static components are now employed in many industrial applications. Some of these are automatic conveyor systems, automatic blast-furnace loading systems, stockhouse programmers, missile erectors, metal-forming presses, automatic fabrication and assembly lines, data-logging systems, ingot buggy and associated equipment programming and positioning systems, flying shear and product length systems, and rolling mills.

This list, while typical of present applications, by no means exhausts the possibilities. The advantages of digital control techniques and static components are now available for any system from the simplest to integrated systems of a scope beyond anything now in existence.

Brief descriptions of two static digital control systems illustrate how the basic functions are applied.

Rolling Mill Automatic Control—Some of the most comprehensive digital control systems are those for automatic rolling mills. A few of these, such as the one diagrammed in Fig. 1, approach computer-controlled mills in the number and complexity of functions performed.

This system controls all mill functions automatically from the moment an ingot or slab enters until the product is delivered from the reversing mill. Input data, supplied by punched cards, includes number of passes, roll openings and sideguard settings for each pass, mill speeds for each pass, and ingot-turning data.

The system applies the basic digital control functions in the following ways:

(a) *Instrumentation*—Hot-metal detectors and shaft position encoders supply information for drive positioning control.

(b) *Information Input and Storage*—A magnetic core storage system receives information read from punched cards. Many auxiliary transistor flip-flop arrays in the distributor and in the sequence controller store data used in performing the various sequences and subsequences.

(c) *Logic and Sequencing*—The sequence controller includes an overall sequence register for each slab, a sequence register for the functions performed in each pass, and a number of subsequences including data accuracy check, roll positioning sequences, rolling sequence, and ingot turning sequence.

(d) *Arithmetic*—Arithmetic functions are continually performed by the digital position regulators in subtracting drive feedback position from reference position and also in such operations as calculating sideguard positions for ingot-turning sequences.

(e) *Digital Regulation*—Digital position regulators control roll position and sideguard position. Special high-response analog speed regulators or voltage regulators for the individual drives provide systems compatible with digital control.

(f) *Information Output*—Digital displays show ingot entry position, product schedule number, pass number, and any other information desired.

Sectional Machine Speed Control—The Pulsetter control for sectional machines such as those used for making paper is diagrammed in Fig. 2. It is completely digital and provides much better performance than any analog system. Speed regulation is approximately 0.01 percent for the entire machine and 0.025 percent for each section. Machine speed adjustments can be made in increments of 0.01 percent and section speed difference adjustments in increments of 0.025 percent.

The speed reference (*information input*) to each section is a pulse train of standard frequency from a master crystal reference, as modified by the operator's adjustment of the master pulse-rate divider (*arithmetic*). Each section control contains another pulse-rate divider (*arithmetic*) for speed difference adjustment. The frequency from this divider is the section speed reference, which is compared by the digital speed circuits to the frequency from the section pulse generator (*instrumentation*). The frequency difference is converted to an analog signal that goes to the magnetic amplifier regulator and also to the digital position circuits. These circuits integrate the error pulses (*arithmetic*) and provide an analog signal to the magnetic-amplifier regulator that is actually a roll position error. This makes the average speed error zero (*regulation*).

Operator displays (*information output*) are digital, permitting supervision of machine performance with much greater ease and accuracy than is possible with analog displays. Data logging of such variables as machine speeds is practical, and central automatic control of the mill is compatible with the machine control itself.

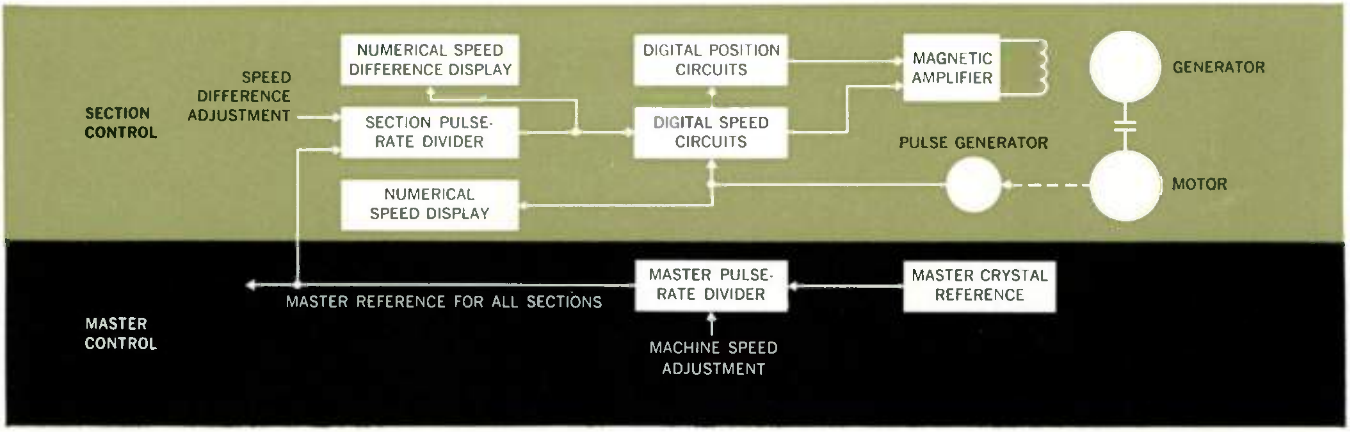
system coordination

Nearly all practical systems are combinations of digital, analog, and mechanical subsystems in which the characteristics of each affect the requirements of the others. Therefore, all the subsystems must be matched with each other to form a well integrated system. This important fact is sometimes overlooked because digital control is somewhat more flexible than analog control with respect to subsystems, creating a tendency to think of it as completely isolated.

The increasing use of digital control is steadily widening the circle of engineers who understand digital techniques, applications, and number systems. As this understanding spreads and deepens, it leads to ever more sophisticated digital control systems and more effective use of them.

Westinghouse
ENGINEER
Sept. 1961

Fig. 2 The Pulsetter speed control for sectional machines exploits the accuracy and repeatability afforded by digital techniques to regulate machine speed and section speed difference.



APPLICATION OF DIGITAL CONTROL COMPUTERS

R. O. Decker
Advanced Systems Engineering
and Analytical Department
Westinghouse Electric Corporation
East Pittsburgh, Pennsylvania

The successful application of a digital control computer to achieve automation of an industrial process requires an intelligent system engineering approach. A digital control computer can perform one function, or several, and determining exactly what is needed for a particular process or plant is a critical step. This means that the first effort must be a preliminary study to establish the technical and economic feasibility of computer control. This, in turn, requires a thorough knowledge of the process itself, and an equally complete knowledge of the functions that can be performed by a control computer.

functions performed by a digital control computer

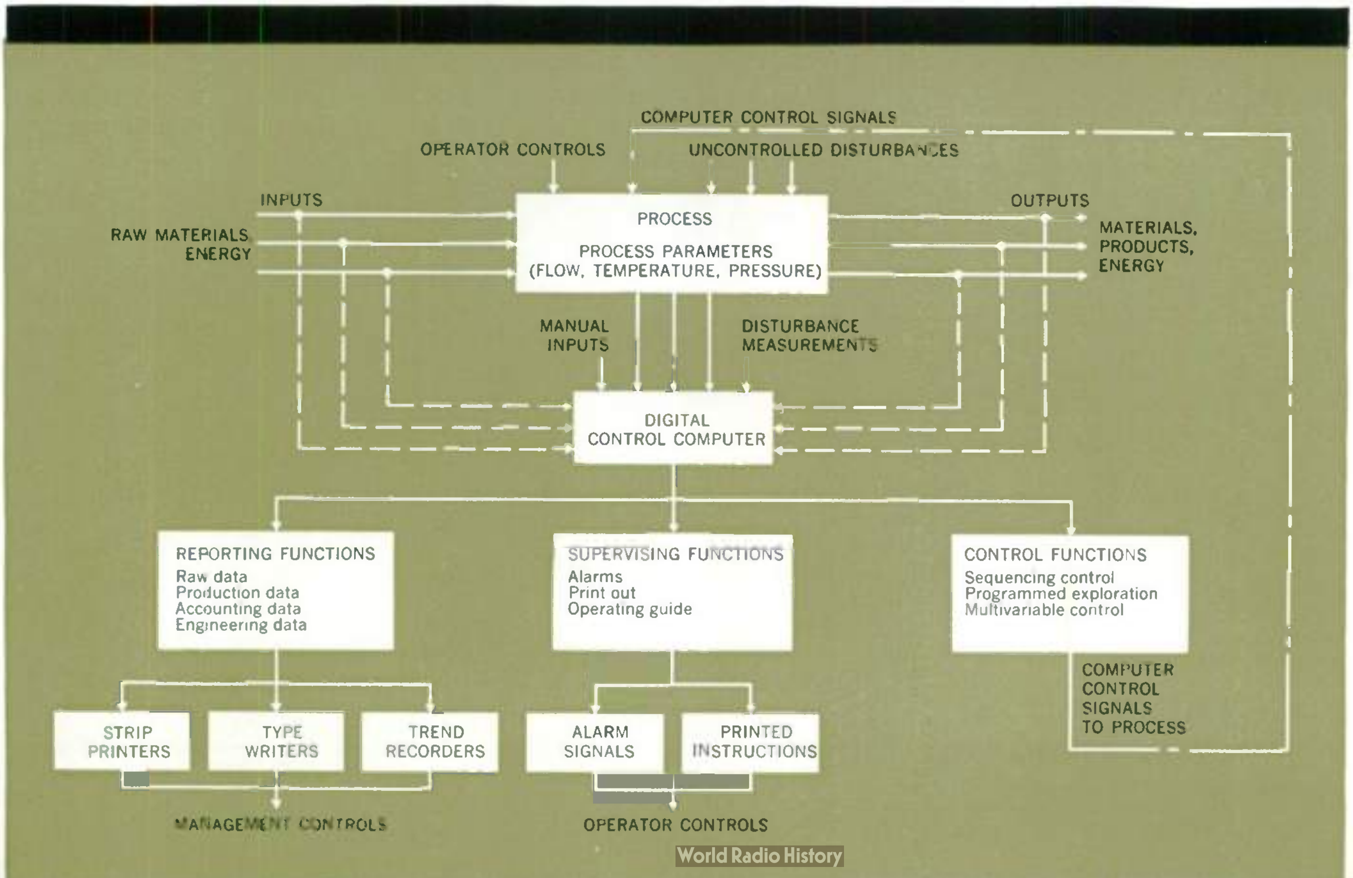
A digital control computer, its relation to a process, and the three major types of functions performed in the system are shown in Fig. 1. The inputs to the computer are measurements of physical input variables, measurement of plant or process parameters, measurements of certain physical disturbances, and measurements of physical output variables.

The three types of functions are categorized by the nature of the action taken by the process control computer. If the output is in the form of typed or printed records to be examined at some later date, the computer performs a *reporting* function. If the output is in the form of signals to the operator, the computer performs a *supervising function*. If the output is in the form of signals transmitted to control the process, the computer performs a *control function*.

In the *reporting function* (see Fig. 2), the computer presents tabulated information ranging from raw data to complex process engineering data. Raw data presentation includes periodic printout of process flows, temperatures, and pressures. Also, special printout on operator demand is provided. Little or no programming is required for this operation. By additional programming, the raw data can be "smoothed" and calculations performed to present production data. This includes totals, trends, efficiencies, operating costs, and operating profits. Accounting data presentation includes facts for inventory records, ordering information, raw materials scheduling, the preparation of billing documents, and preparation of input data documents for other machine accounting systems. The presentation of engineering data includes correlation of input-output information, analysis of overall operation, data for improving existing process control, and for building a mathematical model of the process.

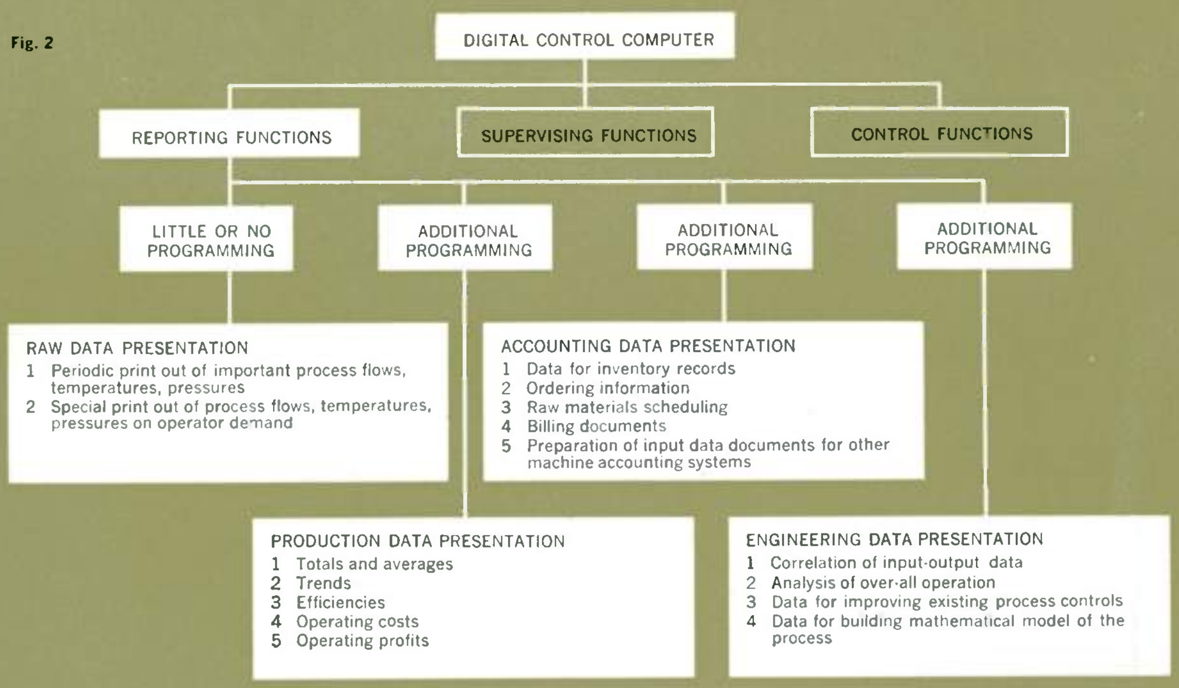
Examples of these functions can be found in various steel and electric utility applications. In one automatic steam

Fig. 1 Relation of digital control computer to a process.



THREE FUNCTIONS OF DIGITAL CONTROL COMPUTERS

Fig. 2



RAW DATA PRESENTATION

1 Periodic print out of important process flows, temperatures, pressures
2 Special print out of process flows, temperatures, pressures on operator demand

ACCOUNTING DATA PRESENTATION

1 Data for inventory records
2 Ordering information
3 Raw materials scheduling
4 Billing documents
5 Preparation of input data documents for other machine accounting systems

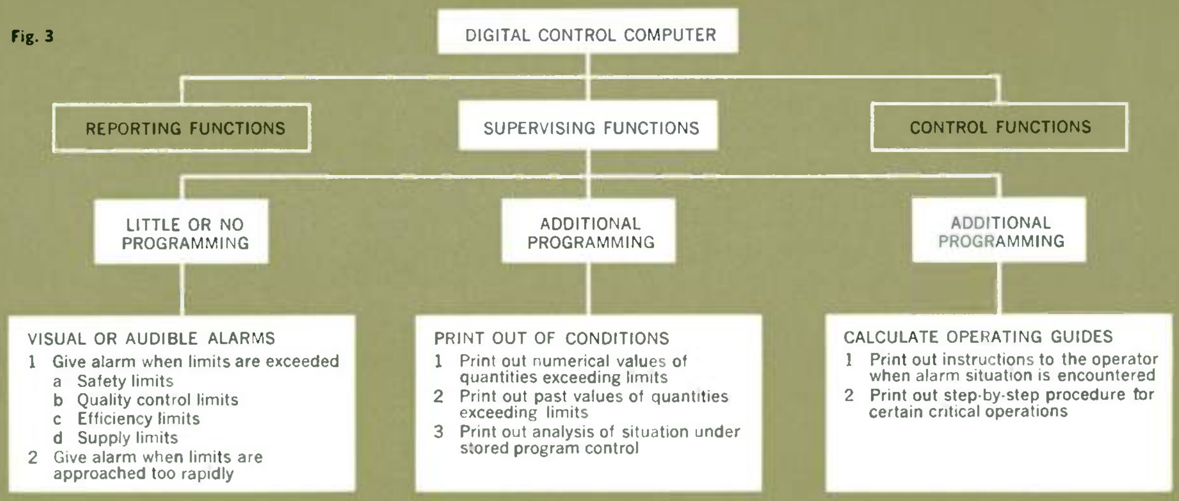
PRODUCTION DATA PRESENTATION

1 Totals and averages
2 Trends
3 Efficiencies
4 Operating costs
5 Operating profits

ENGINEERING DATA PRESENTATION

1 Correlation of input-output data
2 Analysis of over-all operation
3 Data for improving existing process controls
4 Data for building mathematical model of the process

Fig. 3



VISUAL OR AUDIBLE ALARMS

1 Give alarm when limits are exceeded
  a Safety limits
  b Quality control limits
  c Efficiency limits
  d Supply limits
2 Give alarm when limits are approached too rapidly

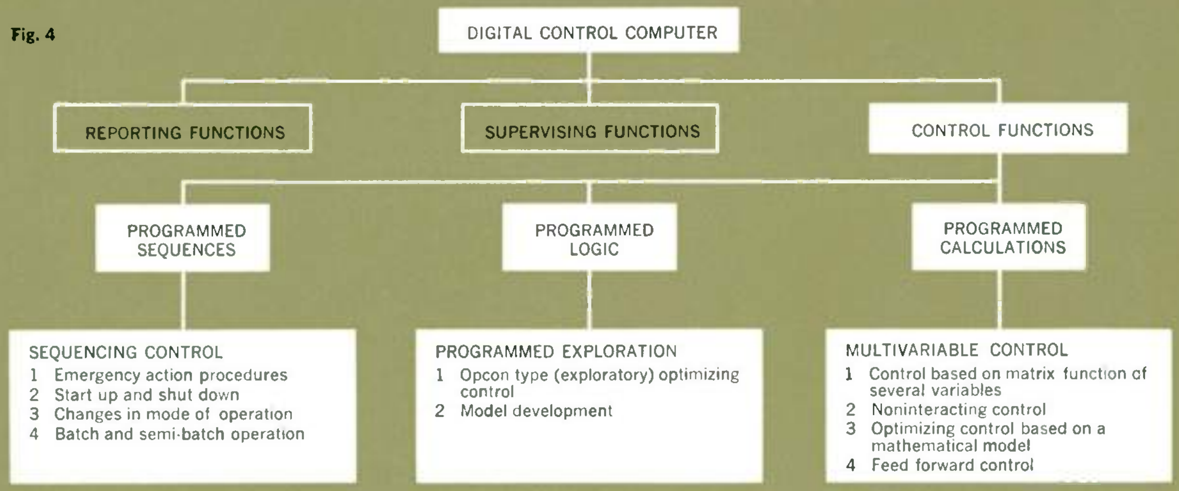
PRINT OUT OF CONDITIONS

1 Print out numerical values of quantities exceeding limits
2 Print out past values of quantities exceeding limits
3 Print out analysis of situation under stored program control

CALCULATE OPERATING GUIDES

1 Print out instructions to the operator when alarm situation is encountered
2 Print out step-by-step procedure for certain critical operations

Fig. 4



SEQUENCING CONTROL

1 Emergency action procedures
2 Start up and shut down
3 Changes in mode of operation
4 Batch and semi-batch operation

PROGRAMMED EXPLORATION

1 Opcon type (exploratory) optimizing control
2 Model development

MULTIVARIABLE CONTROL

1 Control based on matrix function of several variables
2 Noninteracting control
3 Optimizing control based on a mathematical model
4 Feed forward control

plant, efficiencies will be calculated every fifteen minutes and a warning signal will be given to the operator if efficiencies are below specified limits. In a computer-controlled tinning line, calculations of incentive pay are made from production data. In this same application, a shipping ticket, which includes quality classification, is prepared by the computer for each coil as it is produced. An example of analysis of overall operation is found in the electric utility industry, where heat rate calculations are made, giving the power output per ton of fuel input.

In the *supervising function* the computer acts as an important link between the process and the process operator (see Fig. 3). In the supervising mode, the computer rapidly scans input voltages representing physical variables and compares each scanned voltage with the correct voltage stored as a digital number in the computer memory. In many cases, the scanned quantity represents open and closed states of valves, on or off condition of motors, etc.

The fact that a quantity is above or below a reference value can be recognized, and visual or audible alarms given when limits are exceeded. These limits may be imposed for reasons of safety, for quality control, for loss of efficiency, for low levels of product inventory, or for unsuitable values of the input material flow.

Special printout programs can also be initiated, giving the numerical values of the deviation from desired levels of any abnormalities. Under certain conditions, such as process equipment failure, a history of events preceding the failure is desirable. The computer can be programmed to preserve readings on a logarithmic scale over the previous ten hours. A great advantage of the stored-program digital computer over a simple data logger-scanner is its capabilities when an abnormality is encountered; here the computer can be programmed to enter an analysis mode of operation, in which it examines the surrounding conditions and prints out an analysis. For example, a high temperature at a point in the system can be indicated by an alarm; in addition to simple alarming, the computer can be programmed to interrupt its normal scanning cycle and scan temperatures and pressures of adjacent process areas.

The *control functions* of a process control computer are the most advanced operations, although the sequencing control operation is relatively simple (see Fig. 4). Sequencing control involves logical decisions in carrying out emergency actions, in starting up and shutting down a process, or in changing the mode of operation of a process. Sequencing control has been applied in the steel industry to control the sequenced operation of a rolling mill. In the electric utility industry the starting and stopping procedures of turbine generator operation are examples of sequencing control.

Programmed exploration involves the changing of controlled process inputs in discrete steps and recording the resultant changes in the process outputs. This type of exploration can be used for two purposes: to carry out an experimental type of optimizing control; or to develop a mathematical model of the process.

Multivariable control is a level of control beyond that provided by single-loop analog controllers. For example, the controlled quantity may be represented by several output variables; the desired quality may not be measurable directly but may be a computed function of temperatures, pressures, densities, or other variables.

In non-interacting control, the interactions among several input and several output variables are greatly reduced or eliminated. Such a control might be designed so that a change in one input variable causes a change in several output variables. To change only one output variable and hold the others constant, additional input variables would have to be changed at the same time.

Optimizing control based on a mathematical model of the process involves the solution of equations to find a maximum or a minimum value. The mathematical model may be either linear or nonlinear. To date, such controls have been based on a steady-state model of the process.

Feedforward control anticipates the effects of disturbances occurring "upstream" and makes adjustments to compensate for the effects of these disturbances.

application of digital control computers

The order in which these three main functions are listed represents the evolution of computer applications. At first the only function entrusted to digital devices was the accumulation of information from the process and the logging of this information for the operator's use and for permanent record. Soon engineers discovered that digital computers were able to do much more than scan and log data. The small additional cost of a digital computer over a data logging system was justified on the basis of the additional supervising function that a digital computer could offer. Thus the computer was allowed to assist the operator in the many decisions. At this stage, however, action was taken only by the operator.

With the accelerated improvement of components (transistors, core memories) and design techniques, digital computers have proven themselves fast and reliable enough for the control function.

In almost all applications the stability of the control scheme must be proved before the computer control system is installed. Stability can be proved either analytically or (in the case of more complicated systems) experimentally. The experimental studies are performed on a simulated model, usually on an analog computer.

If the control scheme is feasible and inherently stable, the next step is to convert it into a sequence of logical operations that can be programmed into a digital computer. This sequence of computer logical operations is represented by flow charts. The system analyst who charts the problem need not be familiar with all the intricate programming details of the computer, but needs a general understanding of its capabilities. In general, the flow charts for a given problem are the same regardless of the computer used.

The translation of information in the flow charts to a sequence of instructions is done by skilled programmers. The magnitude of programming effort in an average computer application is great and the cost and time for programming and "debugging" should not be overlooked.

Often the experimental data available from the process is not sufficient to establish exact relationships between variables. In this event, the computer can be used to accumulate and correlate data from the process after it is in-

Figs. 2, 3, 4 Functions of a control computer—reporting, supervising, controlling.

stalled. When sufficient data has been accumulated, improvements can be made in the control program and the computer can go into closed-loop operation.

The preceding steps are common to many computer control applications, although each application has different characteristics. In some applications the system operation and relationships between variables are well defined and therefore the design of a control scheme is relatively simple. In others, great preliminary study effort must be devoted to defining the relationship between variables. Some applications are of the simple sequencing type while others use fast dynamic control. The success of any computer-control application depends on the system engineer's thorough knowledge of the system, his familiarity with what a digital computer can do, and his ability to recognize similarities and differences with other applications.

system engineering and digital computer control

System engineering, as applied to industrial digital computer control systems, can be defined as the economic and scientific coordination of production machines, digital computers, related input-output devices, transducers, and instrumentation. System engineering is usually performed by a team, because plant operating methods, equipment performance, operator's capabilities, control techniques, and many economic factors must be considered in applying a computer to automate a plant or process.

Thus, the team must include engineers with detailed information about plant operations. In some instances, this information can be supplied by industry engineers from the computer manufacturer; in many instances, the information must be supplemented by the customer's engineers, who become full or part-time members of the team.

Also, the team must include engineers who understand the limitations and capabilities of the plant equipment and its control, including the necessary instrumentation and transducers to couple the computer to the process. The plant equipment may include steam boilers, turbines, generators, motors, rolling mill stands, or other types of process equipment. The control equipment may include valve actuators, motor starters, and feedback controls.

The system engineering manpower requirements may be quite large in a new application where a detailed study must be performed. The chart in Fig. 5 shows the typical steps and the time required for the system engineering and application aspects of computer control to a complex plant or process where a detailed study must be performed.

The first step is to determine the feasibility of achieving specified technical and economic objectives. In many instances, this first step includes the determination of technical and economic objectives. The first step must provide answers to two questions:

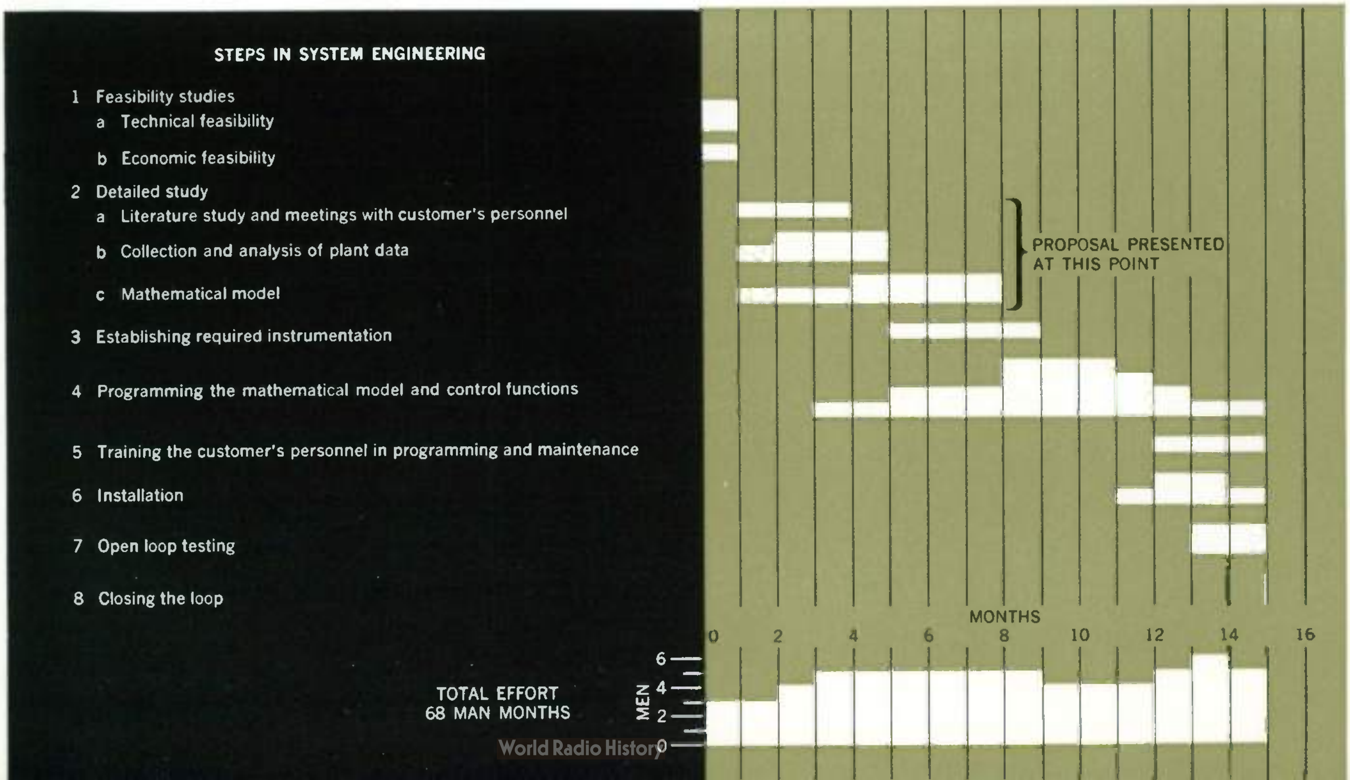
(a) Can computer control be applied to the process?
(b) Can the computer and other instrumentation be justified economically?

The second step is to undertake a detailed study of the process, including a study of pertinent technical literature. This detailed study leads into the development of a mathematical model based upon equations describing energy and materials balances, and other process characteristics. Many relationships involved in the development of the mathematical model are empirical and must be obtained from a large amount of accurately collected plant data.

The remaining steps are indicated on the chart. During open-loop testing (step 7), all computer programs are checked; that is, the computer accepts signals from the plant's instruments and processes the signals but the computer outputs are recorded, not fed to the sub-loop control elements. As the programs are being checked out, the mathematical model is improved on the basis of accumulated data. The final step is to close the loop for automatic computer control.

Westinghouse ENGINEER Sept. 1961

Fig. 5 Steps in application of control computer.



DESIGN OF A CONTROL COMPUTER

W. W. Ramage, *Supervising Engineer*
Logic Design Section
New Products Department
Westinghouse Electric Corporation
Cheswick, Pennsylvania

The digital control computer is a direct descendant of the scientific computer and the data processor, and brings to industrial control all the inherent capabilities of such devices. Especially important are the computer's ability to store and use large amounts of process information, and to coordinate a large number of controlled functions.

The basic elements of the digital control computer are the same as those of the scientific computer. These elements include a storage medium, or *memory*, for both program and data; a *control* unit for coordinating the sequence of operations within the computer itself; an *arithmetic* or logical unit for performing specified manipulations on the data; and *input* and *output* units to provide the necessary communications with other equipment or operators.

The differences between the two types of computers involve a shift in emphasis among the basic elements because of the different natures of their applications. The scientific computer is an *off-line*, or nonreal-time device; i.e., there is no critical time relationship between the various inputs nor between inputs and outputs. The control computer, on the other hand is an *on-line*, or real-time device. Here the computer is an integral part of a process in which a definite time relationship must be maintained between the input, which frequently contains information from the process, and the output, which controls the process directly.

Communications between the computer and operators are different in on-line and off-line applications. In problem-solving applications, the operator must be able to enter problems and associated data and to receive in suitable form the results of such problems, along with any notification of malfunction of either the computer or the program.

The on-line control operator, however, is less concerned with program or problem entry (which is already permanently within the computer) but needs to know the present status of the process as understood by the computer, and any unusual or emergency conditions within either the process or the computer. These requirements demand more indicating types of outputs, such as lights, alarms, and line printers, than the off-line processes.

One other difference between the two types is the need to interrupt or change programs on short notice with an on-line controller.

As plant or process conditions change, different programs or sequences of programs may be necessary; sometimes these must interrupt a program already running. For example, the control computer must recognize emergency conditions within the plant, evaluate the situation, and determine and initiate corrective action.

Prodac computer system

The Westinghouse Prodac (Programmed Digital Automatic Control) computer system, illustrated in Fig. 1, is designed to encompass a wide range of control applications. The system is centered around the storage mediums and the control circuits used in transferring information within the system. Since the storage unit contains both data and program instructions, all subunits must be able to communicate with it. This requires a central control function —i.e., a "traffic cop"—that can direct the flow of information into and out of storage and between the various subunits. This function is performed by the *information transfer system.*

The main path or highway between units is a common trunk circuit capable of transferring 30 binary bits of information in parallel between any of the units on the system. Any unit desiring to transfer information within the system must first request permission and allow the transfer control circuits to accomplish the transfer when the trunk is available.

The other units included depend upon the application; however, all systems will contain a *system control unit* and an *arithmetic unit* in some form. The system control unit can be considered the director of the system, since its chief function is to sequence the instructions associated with any given problem and to assign tasks to the other units as they are called for within the problem. The system control unit is capable of performing certain types of instructions by itself without activating other units, but its principal function is to keep the various problems within the system progressing in a coordinated and efficient manner. Associated with the control unit is an *indexing unit* and *priority selector.* These units are used primarily for instruction modification and program interruption or switching, where necessary.

The *arithmetic unit* performs all arithmetic operations required in the problem at hand. The present unit performs its operations in parallel to increase the speed of operation, and carries out 35 different types of arithmetic commands. This unit does not initiate requests to the information transfer system when operands are required; instead, the control unit requests that the necessary information be transferred whenever arithmetic operations are to be performed.

Other units in the system may be simple storage registers, shown as *buffer registers* in Fig. 1, which are used for buffering information between the system and other devices; or they may be fairly complex units, such as the *peripheral units* in Fig. 1, capable of sequencing through a set of instructions under their own control and performing a limited set of commands. An example is the *analog input logic and control unit.* These more complex units are similar to the system control unit in that they obtain instructions from the storage and perform the specified

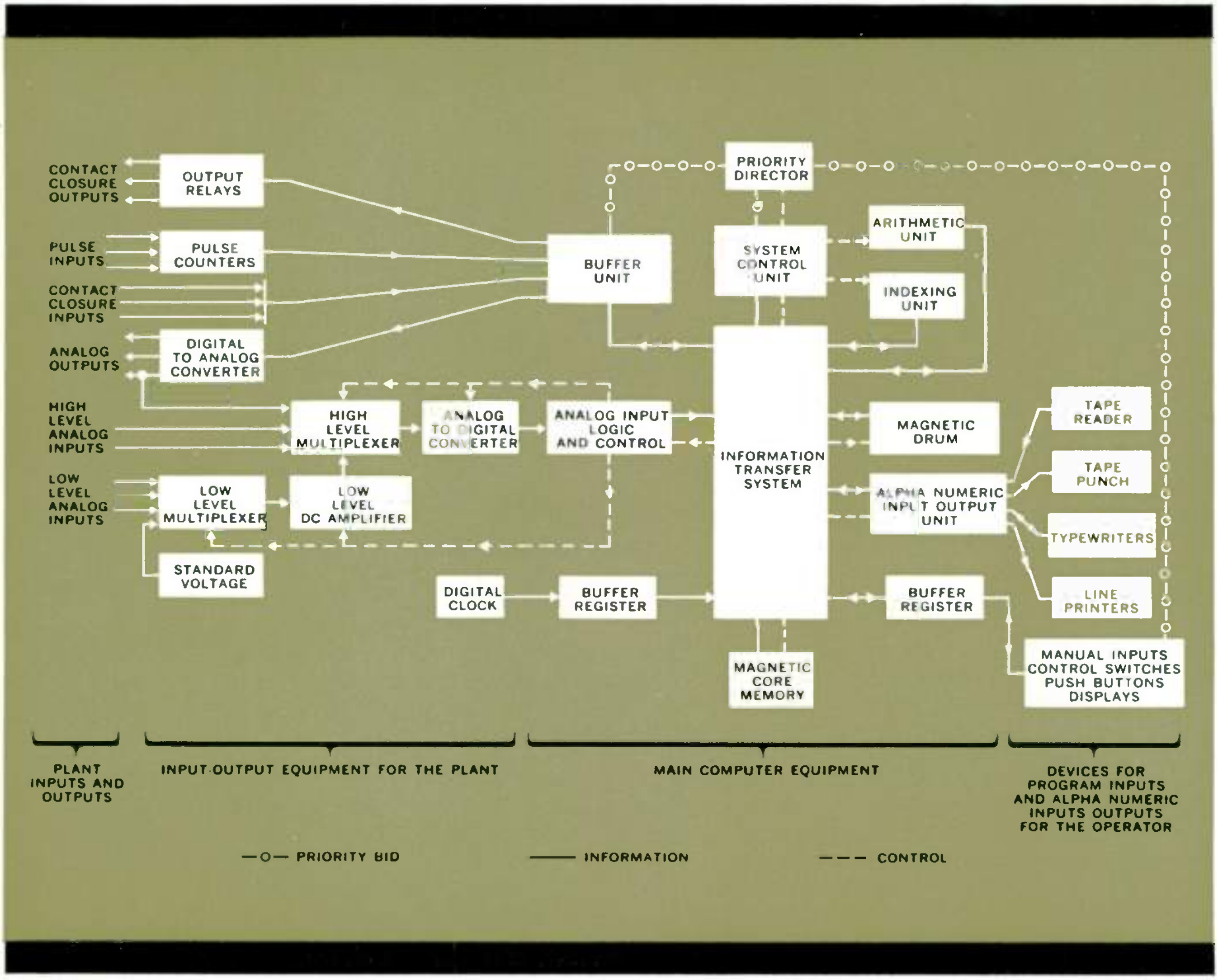


Fig. 1 The general design of the Prodac control computer is shown in this diagram.

command; however, the peripheral units can communicate only to and from the storage and perform only fairly simple and specialized tasks.

As an example of the manner in which information is processed, consider the case of examining a particular analog input point, such as a temperature reading, to check whether it falls within the desired range. The program to accomplish this operation is contained in the storage. The system control unit is sequencing through the instructions, and at some point in the program receives an instruction to check the temperature range. To do this, it directs the analog input unit to connect to a specified input (representing the temperature reading), to convert the input signal to a digital representation, and to store the result in a specified storage location.

The system control accomplishes this by requesting the information transfer unit to move the specified command to the analog input unit; then since the input operation is relatively long in time (100 times the computer instruction time), the system control initiates instructions or switches to another program until the operation is completed.

When the analog input unit has finished its assignment it signals the system to that effect. The system control may at that time obtain an instruction to place the stored temperature indication in the arithmetic unit and compare it with a stored limit.

The system control now requests the information transfer system to move the input value to the arithmetic unit's accumulator, and, with another request, to move the stored limit value to the addend register; the latter instruction requests the arithmetic unit to subtract the limit value from the input in its accumulator. This request to the arithmetic unit is over separate command wires as shown in Fig. 1.

The next instruction obtained by the system control will ask it to sense the result in the accumulator and then base its future program upon that result. If the result indicates the temperature was within the proper range, the system control requests the next instruction in its normal program; however, if the result indicates the temperature was outside the desired range, the system control switches to another program designed to take appropriate action.

The ultimate limiting capacity of the system might appear to be in the rate at which information transfers can be handled by the information transfer unit. The present unit, if kept continually busy, would process over 37 000 transfers per second; however, since such a rate would cause excessive waiting times by units with a low priority rating, a figure of 30 000 transfers per second is more realistic. Since many of the units attached to the computer system will be relatively low-speed input-output devices, a large number of units could be attached without overburdening the system.

units characteristic of control

For on-line computer control certain elements or functions of the computer must be included primarily because of the control application.

The necessity for sensing and setting a large number of external conditions led to the development of a *buffer unit* capable of accommodating 3584 input and output points. As shown in Fig. 1, this unit, like the arithmetic unit, has a set of special control lines linking it with the system control unit. These control lines allow the system to directly address any one point or group of points (14 or 28 points in a group), thereby making the information contained in these points as accessible as information held in the computer storage.

When operating in the sensing mode, i.e., as an input device, the buffer unit switches at computer speeds and introduces no additional delays to the system. However, when operating in the set mode, i.e., as an output device, specific delays may be associated with particular points or groups of points to allow time for setting relatively slow output devices such as relays. If the output device is capable of operating at computer speeds, no additional delay is necessary.

When an output device requiring delay is set by the computer, the buffer unit assumes a busy state for the required time interval and will not accept further instructions. This busy condition is displayed to the control unit, which can, in effect, temporarily bypass further instructions relating to the buffer unit.

Like the arithmetic unit, this unit makes no requests to the information transfer unit, but allows the system control to request all information transfers, and receives from the system control directions about handling any information received. For example, the buffer unit may receive 28 bits of information from the data trunk, and the command from the system control may direct it to set all outputs of a particular group to zero when the corresponding bit of the received information is a one, and to otherwise leave the outputs unchanged.

The buffer unit can be used for an almost unlimited number of purposes, such as setting external switches to hold information for displays, for controlling digital-to-analog converters and other devices, and for reading information from such devices.

Among the peripheral units necessary for control requirements is an *analog input unit*. This unit is capable of receiving an instruction over the data trunk and, once initiated, to select one of 2048 input points, amplify the associated voltage if necessary, convert it to an equivalent binary representation, and then request the information transfer unit to move this binary information to a specified location within the storage medium. If a large number of points must be scanned rapidly, the unit can be made to request its next instruction at the same time. When operating in this manner the unit is a separate simple stored-program computer that can basically carry out only one type of instruction; but, in so doing, it relieves the system control of the time required to obtain the data. This unit, like all peripheral units, assumes a busy state when carrying out instructions and cannot be given new instructions until the current task is completed. This busy condition is displayed to the system control so that it can skip any instructions relating to the unit while it is busy.

The *priority selector*, a subunit of the system control unit, furnishes a means of switching programs because of external conditions or emergencies. This equipment provides the controls that cause crucial information of the current program—that is, information contained in the accumulator, index registers, etc.—to be saved in the storage medium and replaces information in these registers for the next program to be processed. Upon completion of the interrupting program, the priority selector reinstates the previous program at the place where it was interrupted by replacing the previously stored information in the proper registers.

The priority selector also provides the system control with the ability to temporarily suspend a program to wait for some relatively slow input or output device, and run any other program that might be waiting to be processed. The original program is automatically reinstated when the slow device has operated.

This feature allows the system to use several programs during otherwise idle periods and thereby increase the effective speed of the computer.

Other actions associated with control applications, such as the need for special displays, alarms, etc., are accomplished by the buffer unit, or one of the buffer registers. These requirements differ for every application; the particular application requirement is matched to the computer system requirements through these buffering devices.

reliability considerations

Computer reliability has received much attention in circuit and logical design for many years. However, with the advent of the computer within an on-line process loop, reliability becomes more important, because the consequences of a failure are greater. With on-line process control, reliability considerations for the hardware are more stringent, and also must encompass all the computer operating programs.

A common belief is that once a program has been checked out and has performed successfully under test conditions, it should not cause errors in the operation. However, it is virtually impossible to completely check out all possible input combinations and sequences, or even to foresee all the conditions that may occur in an operation. Hence, the design of both hardware and programs must include techniques for detecting unanticipated conditions or faults and, if possible, a means of recovering from them.

Reliability considerations in Prodac systems can be covered under the three general classifications of *logical design features*, *circuit and mechanical design features*, and *programming design.* Within the logical design area, several features are built into the control computer to provide facilities for detecting and, in many cases, analyzing a system failure.

All information stored or transferred within the system is handled as half words of 14 binary bits with a parity check bit, and every movement of information over the data trunk is checked for correct parity. Invalid parity may not necessarily signify an error condition, since units such as the arithmetic unit do not maintain parity within their operations.

The information transfer unit, therefore, contains circuits to remember which units should provide parity correct information and which units may not. The units expected to satisfy parity include all storage devices, such as drums and magnetic cores, and most input-output units, such as those handling printers, paper tape, analog converters, etc. Any failure by these units to satisfy parity would be detected and an error signal returned to the unit initiating the request to transfer. Action taken when an error is detected depends upon the program design.

Each peripheral unit on the system has certain built-in checks to determine whether logic is functioning properly. In most cases this amounts to a check on the proper start and finish of the sequencers or to specific checks during a sequence to determine validity of operation. These checks differ between units depending upon the function being accomplished. In addition, units contain a "time out" feature such that, if the time of operation (unit in a busy state) exceeds a specified maximum, an error signal is generated indicating a malfunction. The action to be taken for any error generated in a peripheral unit comes under program control.

Each peripheral unit contains in its control information one bit that either initiates or suppresses a program-interrupt request through the priority selector if an error is detected within that unit. This feature allows peripheral equipment errors to be brought under program control. At certain times when the peripheral unit is operating in step with the controlled process, a failure would require immediate notification, analysis, and alternate action if necessary. Under these conditions the unit would be instructed to request a system program interrupt upon error. In other cases, the program being sequenced by the system control unit may be of higher priority than the function being performed by a peripheral unit; for example, during printout of statistical data it might not be desirable to interrupt the system when an error is detected. When interruption is not requested, the program that initiated the peripheral unit must check for error-free operation at the completion of the unit's operation. If any error had occurred during operation, one bit of the control word would have been set to indicate that fact and this condition can be sensed under program control. Any action then taken as a result of error is completely dependent upon the program design.

Errors occurring in the system control unit, arithmetic unit, and the buffer unit require immediate attention, and therefore no provision is made to prevent an interrupt. The system control unit asks for an interruption of program when an error is detected, and, in most cases, this switches the program to a diagnostic and recovery procedure. However, in those cases where errors occur when the system is operating on a diagnostic or other high priority program the request for interrupt cannot be recognized, because the system is presumably already operating at its highest level. In these instances the system control unit continues to repeat the instruction to attempt a continuation of the program for the case of transient errors. The system control unit contains circuitry for detecting invalid operation codes, parity errors on information transfers, and a time out for failure to accomplish a requested information transfer.

One further consideration in the logical design for reliability purposes is arithmetic overflow. Arithmetic overflow occurs when numbers with too many digits are generated in arithmetic operations or when information is shifted too far in the accumulator. In an operating program, the data must be properly scaled so that overflow conditions do not occur; hence, any unanticipated overflow must be caused by invalid data, machine failure, or other causes. Overflows that are not sensed and reset before the next attempted arithmetic instruction will cause a request for interrupt in the same manner as other errors.

With regard to the circuit and mechanical design, the reliability of the computer was based on "worst case" design considerations on all circuits, with ample (usually 100 percent) derating of all components. All transistors are 100 percent tested as individual components, again in subassemblies (logical elements), and again in conjunction with printed circuit board tests. A typical printed board is shown in Fig. 2; the mounted subassemblies contain inverter amplifiers used in forming the logic circuits contained on the board.

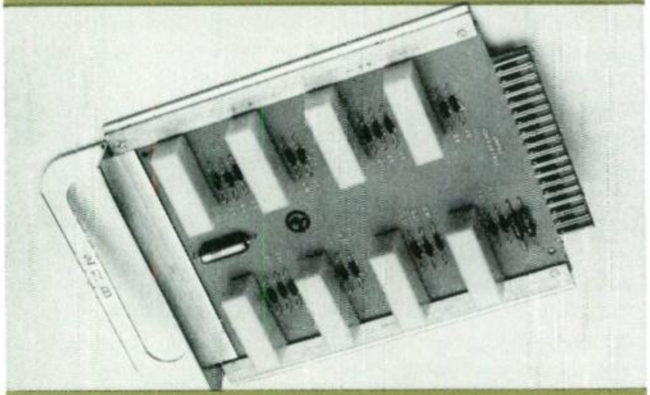
In arranging the physical layout of the computer, the number of connectors was held to a minimum consistent with the conflicting requirements of replacement and manufacturing practice. However, in general, there are at most four removable types of connections in traversing a signal path from a transistor in one cabinet to a transistor in another cabinet. All back-panel wiring is accomplished by wire-wrapping technique and intercabinet connections through pre-formed cables, which are classified and separately formed according to the types of signals involved.

For ease in testing, the computer maintenance panels are located close to the circuit terminations of the associated equipment. Thus, a maintenance man can reach, from one position, the circuits under test while operating the circuits from the maintenance panel. A typical maintenance panel is shown in Fig. 3.

Another aspect of reliability is the part played by programming in the overall success of any computer application. Intelligent programming of operational programs can do much to anticipate nonallowable conditions and to provide the means to recover from them, but many conditions (errors or unforeseen circumstances) are impossible to anticipate; such occurrences generally require diagnosis.

Fig. 2 Top and Center are photos showing a typical printed board used in the control computer and the arrangement of such boards in a computer cabinet. Printed portion of the circuit is on the reverse side of this board.

Fig. 3 (Bottom) This interior view of the computer shows a maintenance panel; similar boards for checking malfunctions are a part of each section of the computer.



The Prodac computer system is designed to permit these diagnostic procedures to be programmed and hence carried out automatically. For example, one peripheral unit is an auxiliary storage unit containing drums. If an error is detected during a transfer of information between the drum and the core storage, the unit can be told to make a high level bid through the priority selector to a diagnostic routine. A preliminary diagnosis of the cause of failure can be made by examining the addressable registers of the unit itself; if warranted, certain sensitive points within the unit itself (state of sequencer, etc.) can be checked by connecting these points, which are already available on exit terminals, through inputs in the buffer unit. The diagnostic program may now be able to ascertain that the transfer failed because of a parity error, sequencer error, overloaded data trunk, or for some unknown cause, and its course of action may be simply to record the failure and try the transfer again. If the transfer cannot be made to work, the diagnostic program will protect against use of bad information by the program that originally initiated the transfer; it accomplished this by placing warning flags in those locations into which the data should have been transferred, and also printing out, or signalling, the existence of an uncorrected fault.

The degree of sophistication and possibilities in this area depends to a large extent on the application, and the storage provided for such routines.

future trends

Today's control computer differs from its predecessors only in the relative emphasis of certain of its functions; however, as more is learned about the applications, the difference in hardware may become greater.

Several characteristics of control computers may well become significantly different in the future. The marriage of analog and digital techniques that appears to be inherent in control applications may expand from simple measuring and converting to the inclusion of more elaborate techniques, such as integrating, where analog techniques are quite efficient.

As more is learned about the types of programs characteristic of control, a greater difference may arise in the built-in instructions of these machines. Already a need has arisen for more logical and single-bit types of commands and for a square-root arithmetic instruction in control applications. More versatile and perhaps more powerful input and output commands appear to be another area for further change.

In the area of storage, the control computers seem to require even larger memories than off-line computers, although they are capable of working with a shorter word length (generally half of that used in scientific computers).

One present development in this area is the large, relatively fast access, semipermanent device that provides economical storage where the stored information need not be subject to modification during operation.

The control computer of today has capabilities far beyond those of conventional control. The control computer of tomorrow will undoubtedly extend these capabilities to control a larger number of industrial processes, and eventually many combinations of processes.

COMPUTERS FOR STEAM STATION CONTROL

J. W. SKOOGLUND
Electric Utility Engineering
Westinghouse Electric Corporation
East Pittsburgh, Pennsylvania

R. B. SQUIRES
Systems Control and Instrumentation
Westinghouse Electric Corporation
East Pittsburgh, Pennsylvania

Automatic control in modern steam plants has reached a high degree of sophistication. This condition has resulted from continuous and orderly extensions in plant apparatus design and in plant control technology. In the past few years, control technology and hardware have taken major steps forward with the development of the high-speed, solid-state digital control computer. Nine control computers have been purchased by utilities for automation of steam power plants in this country. Many other companies who purchase units of 100 mw or larger are undertaking studies to determine if control computers should be applied to their next unit, and if so, to what extent.

levels of automatic control

Power plant control has been refined through the years from relatively uncoordinated and physically separated controls to the completely centralized automatic systems now being installed. Although each plant has a somewhat different control scheme, plant control can be listed under six general levels:

Decentralized Control—Early plants had several separate control locations—one each for boiler, turbine, generator, and major auxiliaries. These units were started and supervised from separate locations with instrumentation at each local control position. A number of simple variable recorders were used in conjunction with local operation of various valve actuators.

Semicentralized Control—In this approach, now in general use, a single control room serves the entire unit—boiler, turbine, generator, and major auxiliaries. A roving operator supplements the man in the control room. Startup can be achieved only by the combined actions of these men and other plant personnel, so that actions at various actuators and local control boards are still required. Multipoint recorders and miniaturization of controls are employed.

Centralized Control—By taking advantage of improved technology and devices, almost all local control locations can be moved into the control room. The furnace is provided with flame protection and automatic ignition systems. Drains and other valving are moved to the operating floor whenever possible. Operating aids are provided by bearing temperature monitors and an analog heat rate or unit performance monitoring system.

Centralized Control with a Digital Data Logger—To the plant described above a digital data logger is added and the analog bearing temperature and performance monitoring functions replaced with digital computer programs. Calculations of heat rate and efficiencies can be complex or simple depending on the computer equipment used. Startup is possible from the control room once clearances have been obtained from the roving operator.

Digital Computer Monitoring of Centralized Control—In addition to data-logging functions, the digital computer monitors the sequence of startup and shutdown. Instructions to the operator as to what functions to perform next are printed out by the computer from its memory. This means that complete startup and shutdown sequences must be internally programmed in the computer. This level of automation reduces operator decisions during the crucial periods of startup and shutdown.

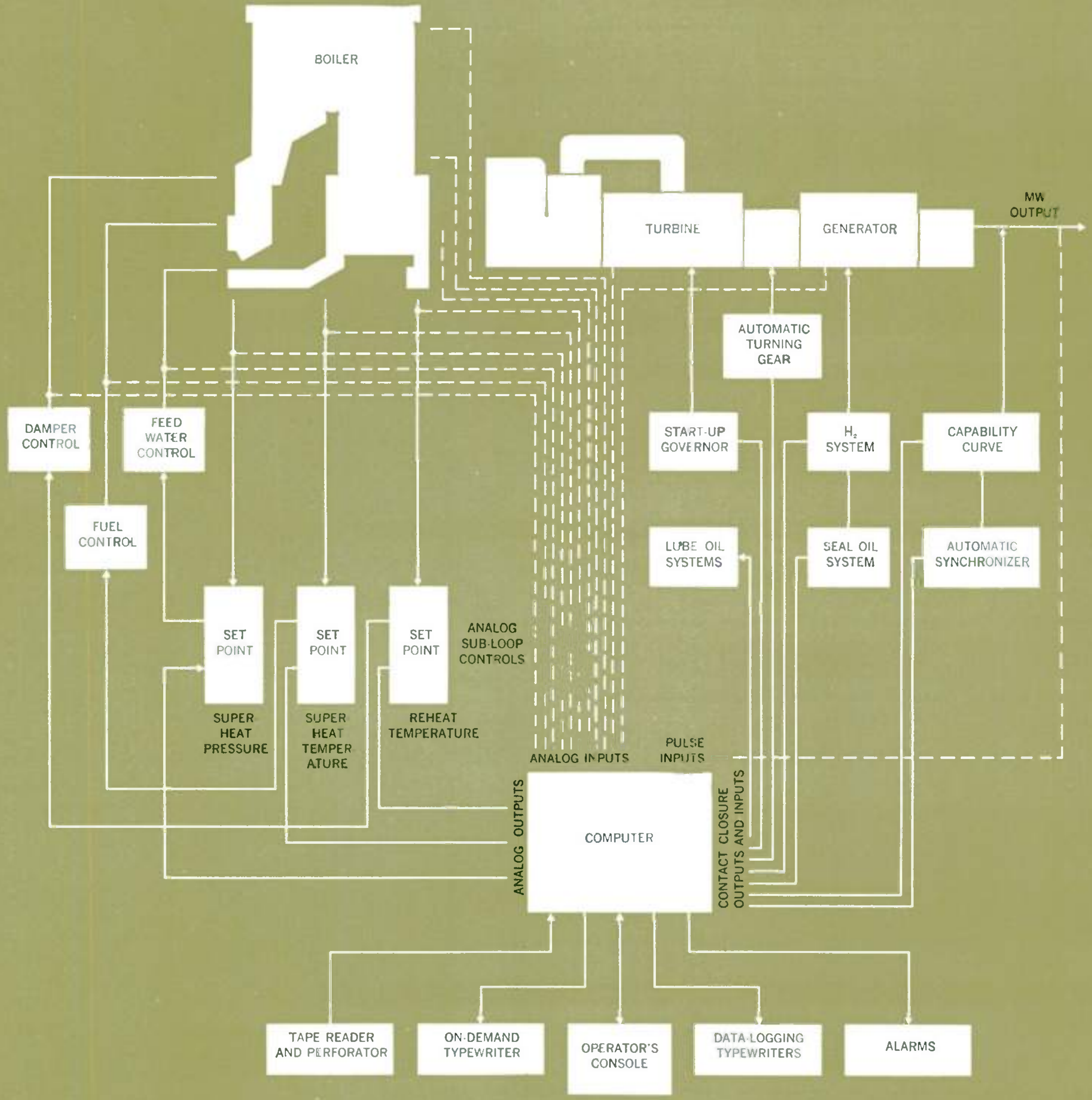
Digital Computer Control—The computer assumes responsibility and exercises all of the operator functions during startup and shutdown. The various station services are checked for availability and the computer initiates all actions to bring the unit up to full load and down again. During normal operation, the computer optimizes controller set points, and continually performs scanning, alarming, and process calculations. The programs for these functions are stored within the computer.

In future plants these levels of automation undoubtedly will be extended. These extensions will increasingly emphasize the design of plants specifically for automation. Almost certainly the computer will assume a greater role in dynamic control of the plant. More of the control and protective functions, which presently are duplicated by the computer and by subloops, will be transferred to the computer. When the reliability of the computer system has been demonstrated, the choice of using the computer or a subloop will be one of speed and economics: Can the control or protective function be added to the computer without using too much computer time; and is programming and memory expense less than subloop hardware costs?

factors motivating automation

Application studies and evaluations of steam plant automation are complex, involved, and quite difficult to carry through. It is difficult to get an accurate measure of benefits and improvements and to assign the right dollar evaluation to them. Three principal difficulties are encountered in this area.

First is the perennial problem in evaluating any control system whose objective is to do a "better" control job. Evaluation requires getting an accurate picture of the errors, failures, and shortcomings of the *existing* control system and operating procedure. For example, a frequent statement is that operators make mistakes, such as tripping the wrong circuit breaker, under the stress of emergency conditions. But what mistakes? What are the consequences? Another complaint is that operators do not keep oxygen trimmed to the right level or do not blow soot as frequently as they should. How bad are these errors and how many dollars are they worth? Answers to some of these questions lie buried in operating logs and chart records—



INTEGRATION OF COMPUTER INTO PLANT

Although the control computer replaces some recording devices and assumes a major part of the operator's duties in sequencing plant equipment and logging data, it is used in the power plant by superimposing above the normal plant control system. This is illustrated above where the plant is shown in block diagram form.

Dotted lines represent computer inputs from the many transducers, limit switches and other sensing devices. These represent the communication channels through which the computer determines the current plant status. Solid lines from the computer indicate outputs, which provide set-point adjustment of analog subloop controls as well as the primary intelligence for a number of motor starters and power-operated valves. Generally, off-on functions are controlled directly and analog feedback control loops are controlled indirectly through set-points.

In some cases it may be necessary to bypass a subloop control during startup or shutdown to obtain stable operation should the subloop be beyond its operating range. Under normal operation the plant is under the full-time subloop control with the computer making periodic checks and readjustment of set-points to optimize efficiency or assure safe and stable operation.

others do not exist at all. In short, a major problem in evaluating "better" control is developing a sound picture of the shortcomings of *existing* controls.

Assuming the shortcomings of existing controls are known, the second difficulty is determining to what extent the control computer can remove these shortcomings. Without a mass of field experience, estimates of the degree of improvement must be based on engineering studies liberally spiced with engineering judgment and insight.

Once estimates are obtained on performance improvements the third difficulty is converting these to a dollar value. For example, one of the principal potential benefits of automation is improved reliability. Everyone agrees that improved reliability has economic worth. But the measurement of reliability and the assignment of a dollar value is certainly less than an exact science.

While these difficulties and uncertainties are somewhat disconcerting, this does not remove the necessity of making the best possible economic evaluation with the facts that are available if a sound plant automation policy is to be developed.

The principal areas of potential performance improvement and economic benefits are:

1. Fuel savings due to closer performance monitoring and closer control of plant variables.
2. Improved reliability.
   a. Higher availability—reduced power replacement costs.
   b. Reduced forced outage rate—reduction in installed reserve.
   c. Reduced danger of catastrophe.
3. Reduced manpower requirements.

computer requirements

Although the steam power plant can be generally classified as a process, comparison with other industries shows that power plant requirements are more stringent for digital control computers. Because of the characteristics and complexity of the plant and its equipment, the requirements for speed, memory, control, accuracy, and reliability are greater than for the typical industrial process. The requirements of the plant affect each part of the computer, such as analog inputs, contact inputs and outputs, computation, and memory.

Analog Inputs—Analog input signals bring to the computer information on temperature, pressure, and flows; the operator previously obtained this information from indicating and recording instruments. All analog inputs must be converted to dc signals: temperatures are converted directly by thermocouples or resistance temperature detectors; pressures and flows (differential pressures) are converted indirectly by electrical strain gages or differential transformers actuated by the motion of diaphragms or bourdon tubes. Alternating-current inputs of voltage, current, and power must be rectified and filtered. Typically, a steam plant might have 1200 analog inputs with more than 70 percent of these being temperature measurements.

The average and maximum rates at which analog inputs must be scanned and compared to limits depend on the number of inputs and on how much the variable can change before that point is scanned again. Most of the variables in a power plant are tied in with the relatively slow thermal process in the boiler, but some points, such as turbine speed during startup, must be scanned as often as once every two seconds. In a large installation, the average scan rate should be 30 to 50 points a second. The rate for a particular measurement may change because of plant conditions. For example, the differential temperatures in boiler tubes and turbine parts are critical at startup and must be scanned more often than under constant load conditions.

Contact Input and Output—The digital inputs represented by the contacts of pressure switches, control con-

COMPUTER SUBROUTINE FOR CHECKING HYDROGEN PURITY

To illustrate the logic used in the development of the computer program for plant control, consider a typical flow chart for monitoring generator hydrogen purity.

In the first block the question is asked if the generator has a pressure greater than 5 psi. A "no" answer exits down and the subroutine is completed, that is, purity is not checked below 5 psi. A "yes" answer to the first question exits to the right and leads to a check of hydrogen purity to determine if it is greater than 70 percent. If not, this condition is alarmed as indicated by the triangular block (1) and the "hydrogen word" is turned on. This word is an internal indication in the computer memory that hydrogen purity is in question. Proper hydrogen pressure is checked next by determining if it is within a ±2 psig band about one of the three possible nominal settings (30, 45 or 58 psig). Should it not be within any of these limits, the hydrogen pressure alarm (2) is set.

Next, the hydrogen annunciator panel is checked for any alarms and a computer alarm (3) is set if any are detected. The program then ends.

The example given has taken only one of several possible paths through this subroutine. If hydrogen purity had been greater than 70 percent, the rate of purity change would have been checked. If the rate is greater than 1 percent in last 10 minutes an alarm (1) is set and a path similar to that described before is again followed. If rate of purity change is less than 1 percent in 10 minutes an exit down is made and a check is made to see if the hydrogen side seal oil pump is running. If running, purity is compared with 95 percent and the computer exits to the right if above 95 percent. The computer "hydrogen word" is turned off if on. Upon low purity the low hydrogen purity alarm (5) is set.

Had the seal oil pump not been running, this condition would have been alarmed (4) and purity compared to 90 percent rather than 95. (Purity is about 5 percent lower without the hydrogen side seal oil pump in service). After a purity alarm, the computer "hydrogen word" is checked; if off, a further check of purity is made. Should it be less than 85 percent the unit is tripped (6), unless the computer is in the monitor-only mode.

tacts, limit switches, etc., in the plant indicate the status of equipment to be controlled and monitored. Scanning is done once a second. The digital outputs, represented by the relay contacts operated by the computer, indicate the actions taken by the computer in controlling the plant equipment. Typically, there will be about 400 digital inputs and 600 outputs.

Pulse inputs, such as from watthour meters, go into external counters, which add them for a short time, after which the contents of the counters are cleared into the computer memory.

The computer controls the plant equipment by the closure of mercury-contact relays, which are operated 28 at a time with any combination of close and open operations. Each relay can also be set individually. The relay drive circuits are multiplexed and connected to groups of relays in turn, so that the relays are energized only long enough to pick up. If this is not long enough for the plant equipment being controlled, polarized relays are used; these relays maintain their position until their reset coils are energized by the computer.

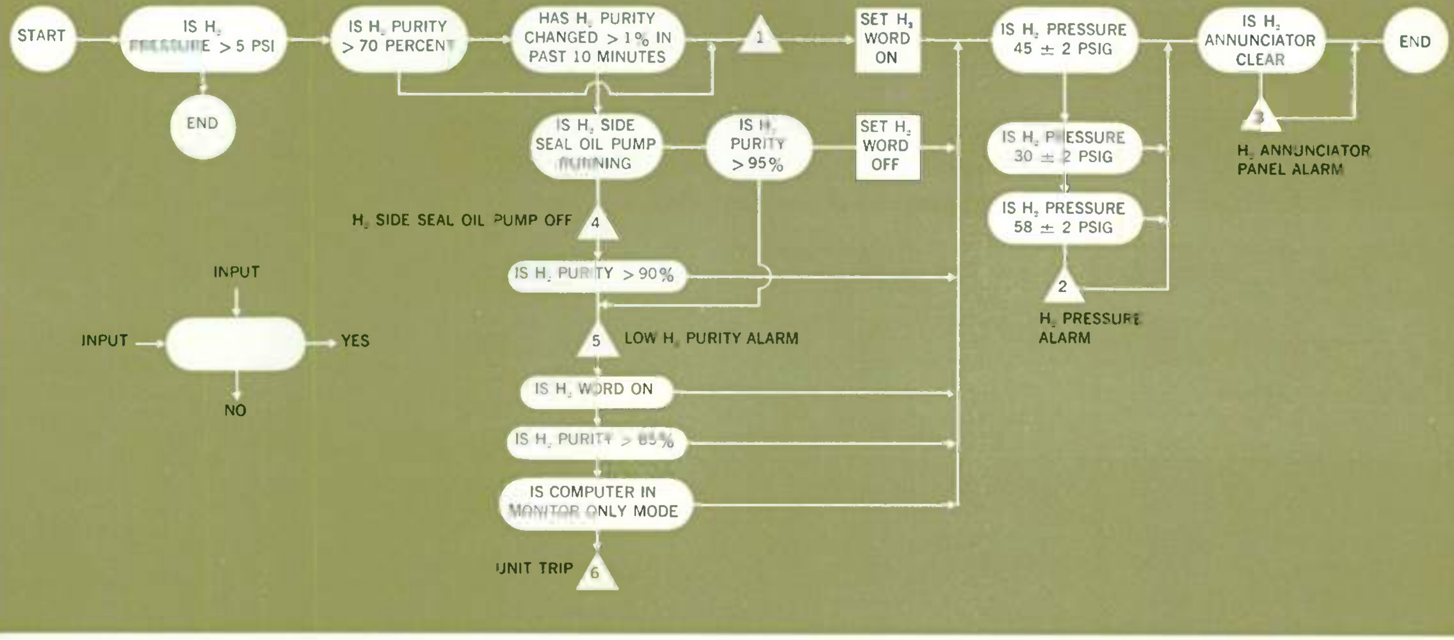
Analog Output—The computer controls the set points of analog controllers for quantities such as steam pressure by setting analog output voltages. Each output must have its own power supply to secure the necessary isolation. Each output magnitude is set by 10 mercury relays representing a 10-bit binary number. This represents an accuracy of 1 part in 1024. The output is checked by feeding it back into the analog input unit and making appropriate corrections. About 40 analog outputs are normally provided.

Computation and Memory—The speed of computing is dictated in general by scan cycles, and modest speed usually suffices. However, when the computer is controlling equipment directly, as in startup, or is participating in control during dynamic optimization, the requirements for computer speed increase markedly. For this reason, the Westinghouse Prodac computer uses a high-speed random-access core memory as a working storage with access time so short that it poses no problems.

The other requirement for memory—that of size—is dictated by the amount of control and sequencing needed, and the complexity of contingency actions. Auxiliary drum storage provides most of the total memory requirements. The amount of high-speed core memory is determined by the amount of high-speed control action required, and by the speed of information transfer from the auxiliary drum storage. A well-designed system can bring information from drum to core storage at a rate faster than it can be used in the arithmetic and control operations, so that required core storage is minimized. For the Prodac system in power plant control, the required storage capacity—largely in drum—is 50 000 to 100 000 words, depending upon the degree of automation and plant complexity.

A most important consideration is that this amount of memory must be filled with the instructions of the stored program, which are written by programmers. The cost of programming an instruction substantially exceeds the cost of the memory location that it fills, so that programming costs represent a sizable investment.

Priority Direction—The control of an automated steam plant requires the simultaneous consideration of many sequences. During startup, for example, boiler firing rate is being increased while the turbine is accelerating. While a number of analog input points are being scanned for alarming, an off-normal logging and alarm printing may also be going on. These programs each involve a number of control actions and numerous time delays imposed by controlled equipment. Therefore, a system is needed to select programs to do the most important things first. The Prodac computer accomplishes this by means of an internal *priority director*, which automatically selects the program with the highest priority level. The priority director provides



for storing the register contents of one program and bringing in the contents of a higher priority program. This feature saves the memory space and programming time that would be required to insert these instructions each time the running program was interrupted.

Accuracy—Accuracy of the control computer in an automatic plant is limited by accuracy of the primary sensors. The Prodac computer can measure an analog voltage to ±0.15 percent, but once in digital form the precision can be one part in millions if required. The 0.15 percent accuracy is better than most sensors, so that sensor accuracy is an area for future improvements. However, where variables are needed to close accuracy, the computer can apply corrections based on the many other parameters that the computer has simultaneously available, so that the fundamental requirement for sensors is repeatability rather than absolute accuracy or linearity.

Coordination with Analog Subloops—Although the computer performs many of the operator functions and may replace some of the existing plant control systems, a number of analog subloops remain. Some of these loops control feedwater flow, fuel and air flow, steam temperature, generator voltage, and turbine load. Normally the computer does not control these quantities directly, but indirectly through set-point adjustment. However, the computer does provide backup for the analog subloops should they fail to function properly. The control computer first detects a malfunction by scanning and computation. Next, a computer program is called from memory to take over the subloop control. This digital computer control program must be carefully established so that the operation of the system under these conditions is at least as stable and reliable as that of the normal analog control system.

The control computer must also recognize any malfunction that is so serious that the computer cannot take over and maintain control. In this case, all or part of the plant should be shut down.

plant transient analysis—improved control

In programming the control computer, every detail of plant operation under normal and emergency conditions must be transformed into computer instructions. Although the control computer could be programmed to duplicate present manual operation, this would not utilize the full computer capabilities. To obtain the full benefits of automation, the complete process should be analyzed for optimum operation under computer control. Before the computer is operated in the control mode—during startup, shutdown or running—it is imperative that the action of the control computer be studied and compared with plant dynamics to assure stable and safe operation. This is especially true if the computer is used in place of, or complementary to, conventional boiler control equipment.

The effect of the computer's performance characteristics on plant response must be determined. A number of methods to study the response of the power plant under transient conditions have been developed and are being further refined. These methods require the development of the proper process equations of the power plant so that plant operation can be represented mathematically. Coefficients and constants for these equations are developed from an examination of the type of power plant equipment involved, and from design data on the various equipments. Knowing these parameters, and using the process equations, the operation or response of the plant can be represented. The response of the control computer can also be simulated. By combining the representations of the plant and computer, the problems can be studied under a great number of different and varying plant control conditions.

The development of power plant automation must include development of programs for the control computer and a system of analysis for plant study. Development of system analysis may be considered as part of the overall programming effort for the computer and is a vital part of the Westinghouse program for automatic power plant control. Through studies of this type the actual requirements of the digital control computer can be established, as well as some of the required characteristics of the power plant equipment. In some cases a rearrangement of the plant cycle will greatly facilitate plant automation. Since the cost of plant analysis alone will be many thousands of dollars, analytical techniques must be refined, to minimize the effort on any one program.

Improved Plant Control—One reason for developing a mathematical method to simulate plant operation was to establish and prove out various computer programs; an equally important use is the development of new types of control systems. With simulation techniques, a new scheme for *non-interacting* boiler control has been developed. Noninteracting control is a control system in which changes in one plant variable can be made with a minimum of interaction upon the other variables. With conventional control systems an error must exist between the measured variable and set-point before corrective action is taken. During rapid load changes the result can be excessive excursions in temperature and pressure. With noninteracting control, the interrelationship and effect of one plant variable upon another is represented as mathematical transfer functions and their effect can be predetermined. In this way a corrective action in changing, say, steam flow also results in simultaneous corrective signals being sent to the fuel valve, fans and the desuperheating spray. The net result is much closer control of plant variables under both steady-state and, more important, transient conditions.

PARTIAL OUTLINE OF TURBINE GENERATOR AUTOMATIC STARTING LOGIC

This logic illustrates the sequence of operations necessary to progress from turning gear status to governing valve control with approximately 10 percent load. The first block represents a computer command and the result is the unit rolling off turning gear to the pre-set speed of 325 rpm. The remaining blocks in the top line represent questions that when satisfied exit to the right, and when not satisfied exit to the bottom. A negative answer, or exit to the bottom, initiates a detailed program which will satisfy the original inquiry. The time used to satisfy the questions shown, depending upon unit and plant conditions, could be as little as one-half hour or extend to several hours.

OPERATIONS BY THE COMPUTER DURING STARTUP:

1 Monitor preliminary conditions
2 Open drain valves
3 Test emergency air and seal oil systems
4 Check turbine oil pressure and place on turning gear
5 Start circulator
6 Start condensate, gland seal, and vacuum systems
7 Apply generator field
8 Set set-points on combustion control
9 Start fuel pumps
10 Test turbine oil pumps
11 Test emergency trip
12 Reset trips
13 Ready boiler
14 Position boiler dampers and purge
15 Light-off burners
16 Raise temperature to 500 degrees F
17 Start exciter and synchronize on turning gear (for cross-compound units only)
18 Roll turbine to 325 rpm
19 Bring turbine to 3600 rpm
20 Synchronize and load to capacity of starting control
21 Shift control to governing valves
22 Raise load to 10 percent
23 Put main boiler feedpumps in service
24 Raise steam pressure to normal
25 Raise steam temperature
26 Raise load
27 Start remaining auxiliaries and burners
28 Raise to desired load

OPERATIONS BY THE COMPUTER DURING NORMAL CONDITIONS:

1 Monitor plant conditions
  Scan analog quantities, alarm if out of limits
  Scan contact-closures, alarm if not correct
  Compare calculated quantities with equivalent measurements to check limits or sensors
2 Show selected quantities as a function of time on printer or trend recorder
3 Detect and analyze emergencies, and take corrective action
4 Prepare reports, log measured and calculated values
5 Make performance tests and calculate generator, boiler, heater, and auxiliaries efficiencies, compare to calculated "bogey" values
6 Optimizing plant performance—Constantly move closer to best performance by adjusting: excess air, superheat and reheat temperatures, burner configuration, gas recirculation or burner tills, dampers, turbine valve throttling, use of auxiliaries

By holding closer limits on plant variables, and minimizing rapid temperature and pressure changes, equipment is worked in a less severe manner and overall reliability is increased. By operating closer to plant design limits, additional operating economies are obtained.

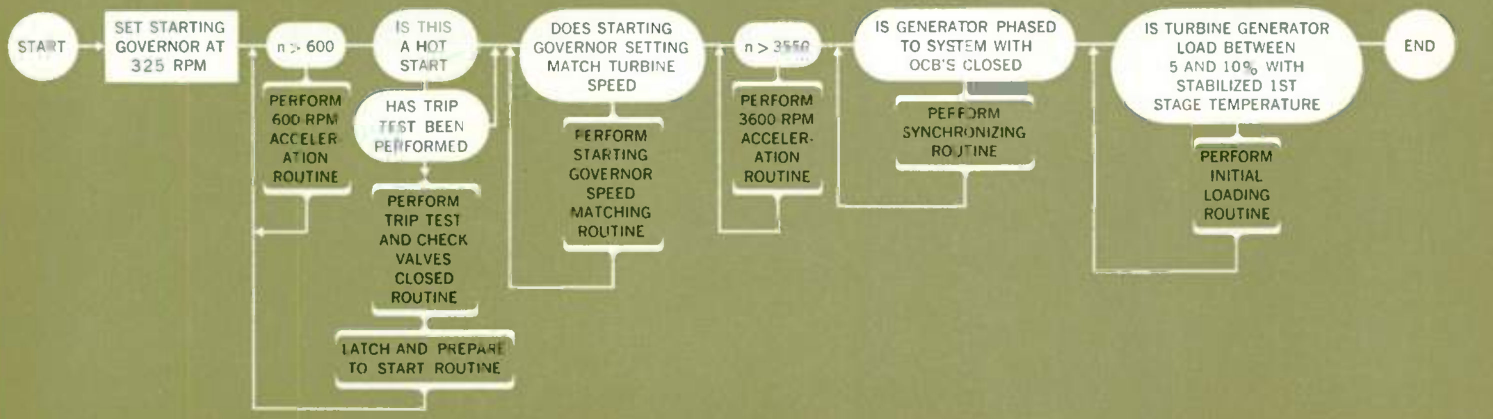
automated system of the future

Under emergency conditions, when plant loads must be increased rapidly to make up for a generation deficiency, corrective action can be obtained with computer control more rapidly than is now possible; this corrective action can reduce the likelihood of large system shutdowns caused by insufficient generation. Eventually, steam generating plants with no on-shift operators will evolve. There will be, of course, maintenance and supervisory personnel on duty during the day to check proper operation, but at night the plant will operate unattended. This has been realized today with hydroelectric and gas turbine plants—the steam plant is only larger and more complex.

The automated power system of the future will contain a master computer at the central dispatching office to automatically determine which power plants will be operated to meet the varying daily loads in the most economic manner, and will signal to the plant control computers in each station to start up and shut down units. A number of companies are now studying the application of the digital control computer to the economic dispatch problem. In addition these computers can do more than the present analog systems by calculating the most economic manner to start up and shut down units through the day. It is a small additional step to send output signals to the individual plants for completely automatic control.

It is possible to imagine many types and levels of automation in the power system of the future, but the automatic power plant is progressing at such a rapid rate that the state of the art even five years from today is difficult to predict. However, by incorporating the most advanced thinking on each job as it progresses, and at the same time continuing a long range development program looking to the future, industry can be assured that development will progress as rapidly as possible, consistent with reliable and safe operation.

Westinghouse ENGINEER Sept. 1961



COMPUTER CONTROL FOR THE STEEL INDUSTRY

E. H. BROWNING, *Manager*
Metal Working Section
Industrial Engineering Department
East Pittsburgh, Pennsylvania

J. H. COURCOULAS, *Manager*
Computer and Data Processing Control Section
Systems Control Engineering Department
Buffalo, New York

Steel production capacity in the United States has risen from about 80 million tons to 150 million tons in the past 20 years. The amount of finished steel produced per manhour also has increased about twofold in that period, and the quality of steel products and the number of products available have increased. The steel industry's use of some 35 billion kilowatt-hours of electric energy a year, compared with its consumption of about half that 20 years ago, reveals the main reason for these vast technological advances—effective use of electric drives and control.

Obviously, much has happened since the days when workmen fed steel by hand into steam-driven rolling mills. The steel industry has been quick to adopt advanced operating techniques, culminating now in control systems employing on-line stored-program digital computers.

control development

The introduction of electrical prime movers near the turn of the century was the first step in application of electric power and control. These prime movers did more than merely replace the steam engines previously used: they added flexibility. Next came regulated electrical drive systems, for rolling mills and processing lines, that further increased speed, capacity, and flexibility.

These were *evolutionary* steps. The developments of the past five years can better be described as *revolutionary*, for there has literally been a technological explosion in drive and control systems. One new development has followed on the heels of another, and the end is not in sight.

The introduction five years ago of Prodac card-programmed digital automatic rolling-mill controls began the technological explosion. These were applied as complementary controls for the highly developed adjustable-voltage dc drive systems that already had gained general acceptance in the industry. The initial intent was to improve, through automation, the operation of primary reversing mills such as blooming, slabbing, reversing-roughing, and plate mills. A number of highly successful installations have been made and, today, consideration of programmed or more advanced controls for new rolling mills has become the rule.

Programmed controls are essentially directing centers that govern the overall operation of a process. They can be applied to any process for which definite step-by-step procedures can be set down in a logical and sequential order. Digital programmed controls were applied first to primary reversing rolling mills because all the operations required of this type of mill were understood.

A typical reversing mill with digital programmed control is diagrammed in Fig. 1. The card reader receives a punched business card containing instructions (rolling schedules prepared off-line before the rolling process is begun). The operator presses a pushbutton and the information from the card is read into the programmed controller. When the raw material is ready to enter the process equipment, pressing another pushbutton initiates the fully automatic rolling cycle. Hot-metal detectors locate the metal with respect to the mill and its auxiliary apparatus; their signals are used for sequencing so the bar can be passed through the mill the predetermined number of times required to reduce it to the desired size.

One subsystem of such a mill is diagrammed in Fig. 2. Information (desired roll opening) from the punched card is placed in a memory unit in the controller. The actual position of the rolls is sensed by a pulse-type position transducer, whose output signal is fed to a reversible binary counter. When the operator initiates the rolling cycle, the reference signal from the memory is read into a digital difference detector; simultaneously, the output of the reversible binary counter giving the roll separation is also read into the difference detector. The difference detector compares the two signals by numerical subtraction, and the difference is supplied through a digital-to-analog converter into a magnetic amplifier for the screwdown drive system. The screwdown drive motor is driven until the difference signal becomes zero and the actual roll separation corresponds to the reference value.

Programmed control has also proved to be a valuable tool for governing the blast-furnace process. It regulates the stockhouse material-handling operations and charges the raw materials in the proper sequence and quantity.

Programmed control has helped the steel industry increase production, use processing equipment more efficiently, improve product quality, and extend management control over its processes. Useful though it is, however, programmed control requires a considerable amount of off-line effort to establish and prepare the operational data used for command input. This suggested the next step forward—on-line stored-program digital computer control systems in which input information is minimized and direct use is made of data collected from the process.

Fig. 1 (Top Left) Simplified diagram of a card-programmed reversing mill. The programmed controller automatically controls horizontal roll openings for each pass, edging-roll openings, mill speed, speed matching of edging rolls and tables to the mill, descaling sprays, acceleration rates, entry speeds, and sideguard, manipulator, and finger operation.

Fig. 2 (Top Right) Screwdown control subsystem for the mill diagrammed in Fig. 1.

Fig. 3 (Center) A stored-program control computer (lower zone) directs the conventional drive and control equipment (middle zone) for a reversing plate mill (upper zone).

Fig. 4 (Bottom) Prodac on-line computer control equipment for a reversing plate mill.

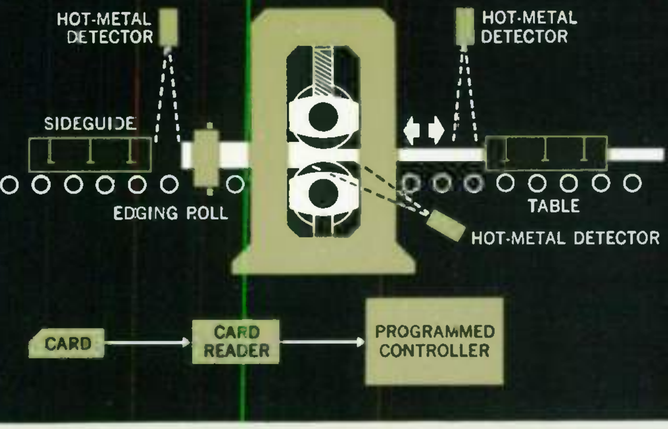


Fig. 1

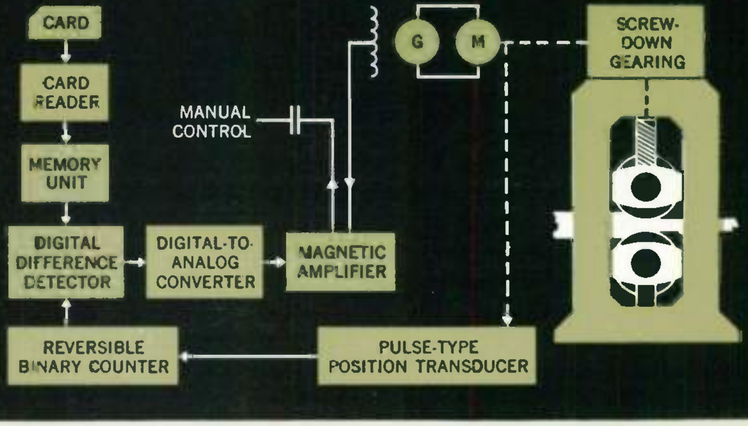


Fig. 2

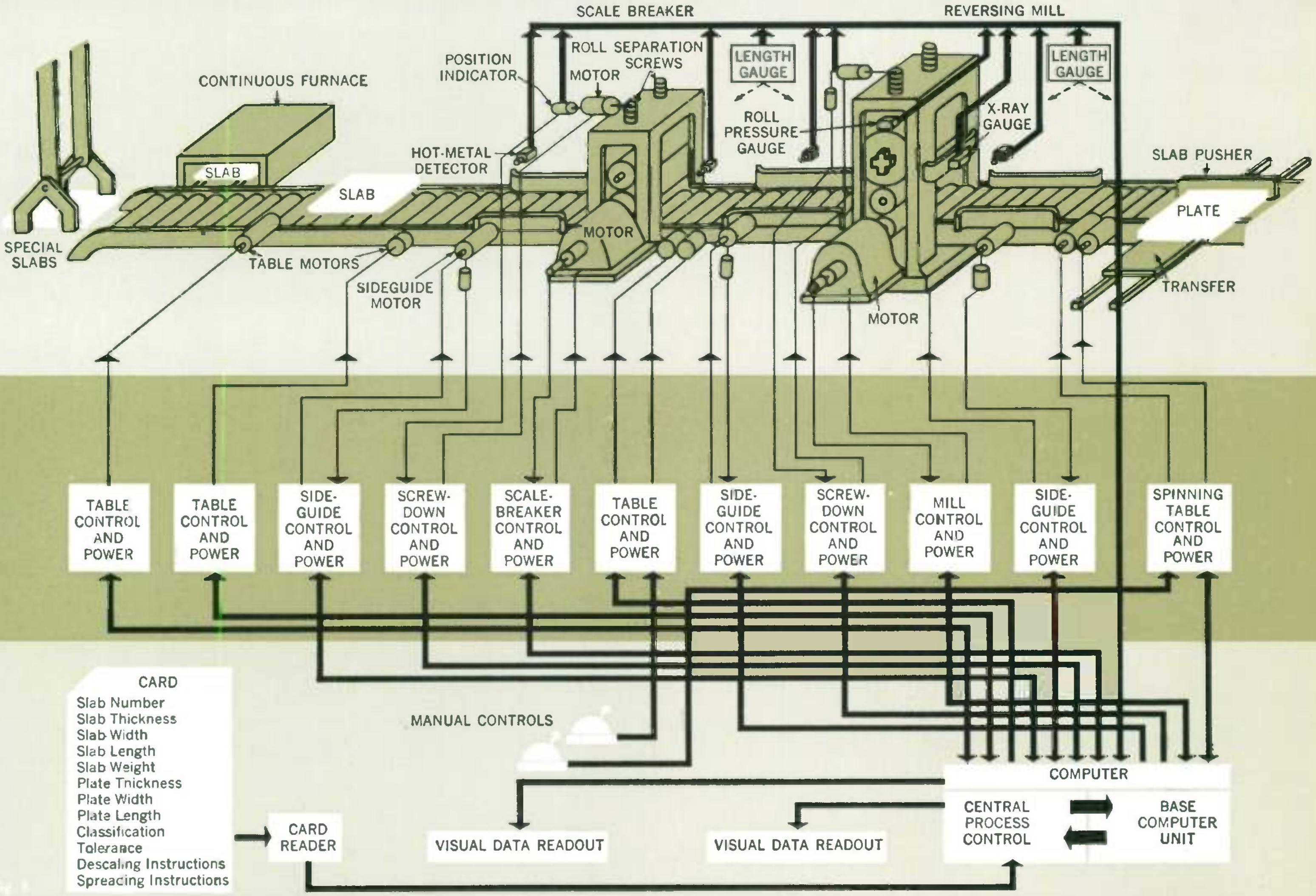


Fig. 3

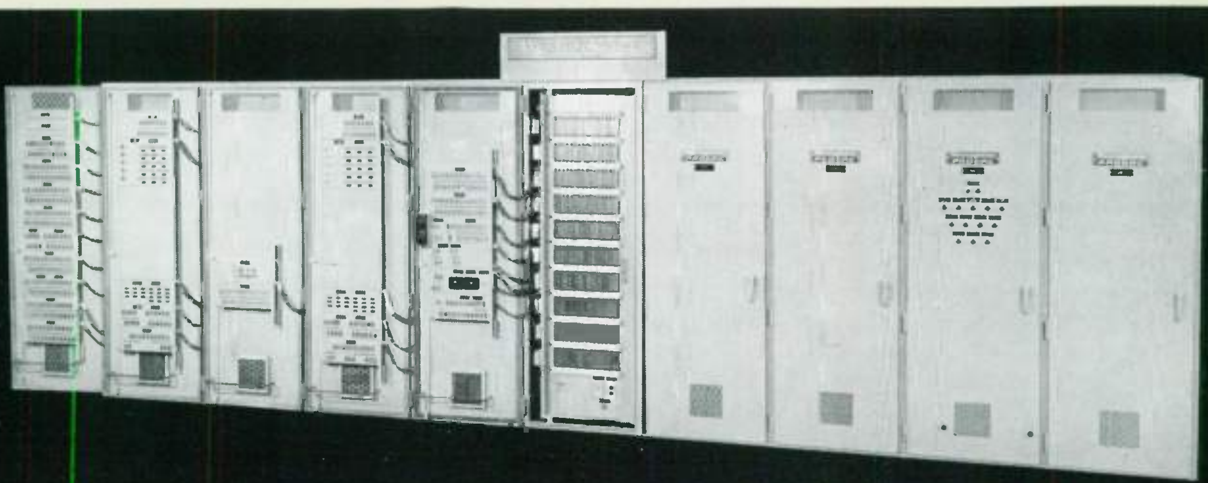


Fig. 4

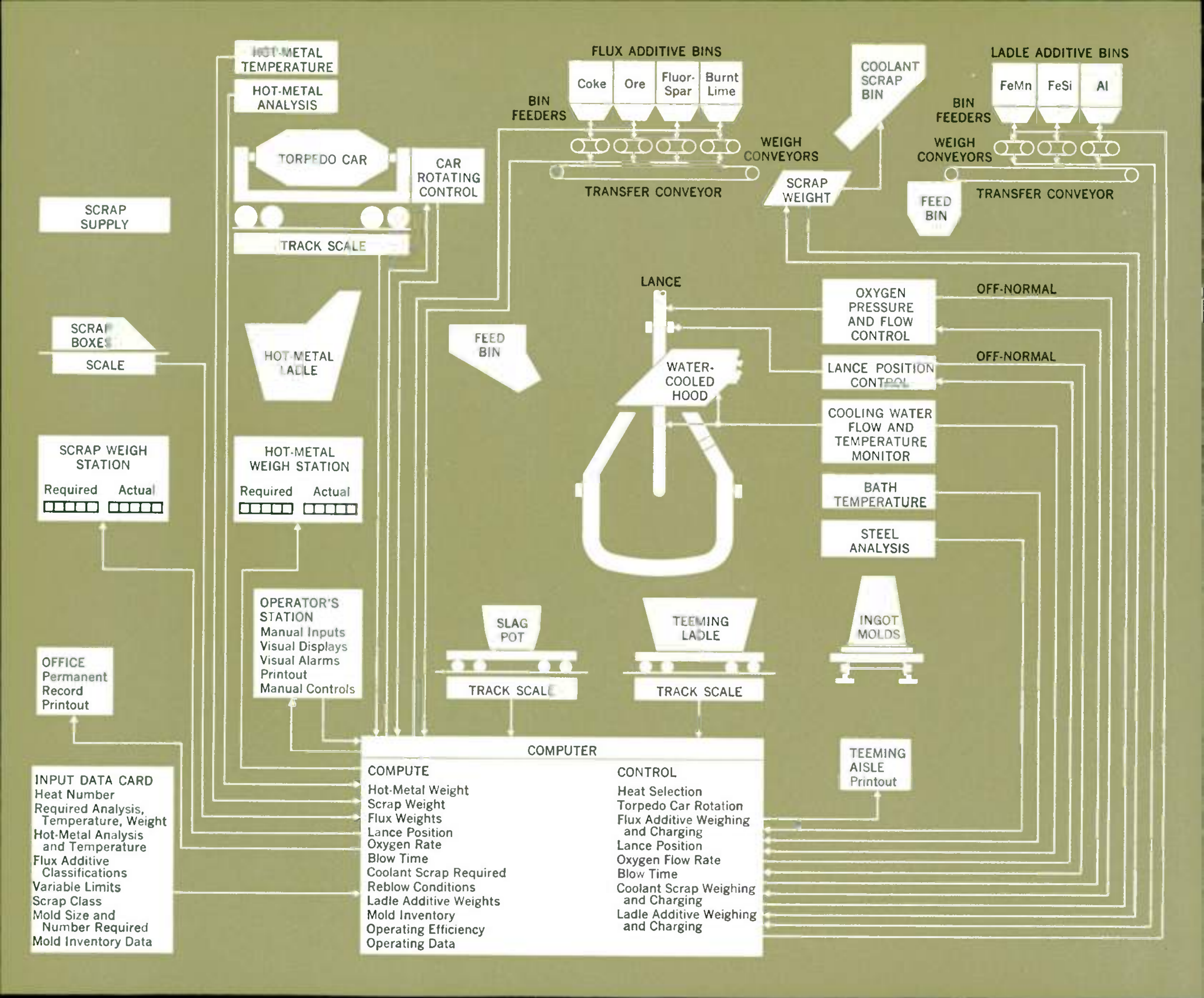


Fig. 5 Basic oxygen converter process controlled by an on-line stored-program computer.

on-line computer control

Rolling Mills—Prodac on-line stored-program digital control computers for four rolling mills have been completed or are under construction. They are two plate mills (one 134-inch, one 160-inch/210-inch), a hot strip mill reversing rougher (56-inch), and a blooming, slabbing, and plate mill (46-inch x 114-inch—39-inch and 60-inch x 160-inch).

The general organization of a reversing plate-mill system is illustrated in Fig. 3. The mill receives slabs ranging from 4 inches to 24 inches thick, 48 inches to 75 inches wide, and 54 inches to 264 inches long, weighing up to 70 000 pounds. A series of reversing passes through the mill rolls reduces the slabs to plate 3/16 inch to 15 inches thick and up to 200 inches wide and 125 feet long.

A heated slab is brought onto the approach tables and passed through the scale breaker to remove oxide scales and to make the initial reduction. The slab then is turned 90 degrees, passed broadside through the reversing mill a number of times, again turned 90 degrees, and finish processed for the required number of passes.

The top zone in Fig. 3 shows the mill processing equipment, its driving motors, and the sensing devices that feed information back to the computer control system and to the conventional regulators. The sensing devices include screw and sideguide position indicators, hot-metal detectors for locating the metal in relation to the mill, pressure gauges for indicating rolling forces, and x-ray and length gauges for determining dimensions of the plate.

The middle zone in Fig. 3 shows the conventional automatic regulated controls used with rolling mills of this type. They include dc adjustable-voltage drive systems, regulators, and protective devices. The bottom zone contains the Prodac on-line computer that governs the process operation. It includes information input equipment, the computer, and the necessary output devices for command, set-point adjustment, data transmission, and data display.

Input information is contained on a separate card for each slab. It includes slab dimensions, desired plate dimensions, alloy classification, and a few other instructions. Permanent process information is stored in the computer memory. The computer draws upon these two sources of information to determine the number of passes, the drafting practice (reduction of slab cross section in each pass), and the point at which the slab is to be turned.

Drafting practice is limited by safe roll force values, torque availability, maximum permissible reduction (inches) per pass, and maximum percent draft empirically allowed. These limitations and the importance of achieving correct plate thickness and shape in the fewest number of passes make roll force the most important consideration in the operation. The proper draft per pass must be computed and set automatically to achieve the roll force that, for a given product width and roll crown, produces a flat plate. This is a complicated procedure that can be expressed mathematically, so it is suitable for solution by an on-line digital computer. Slab temperature affects the roll force required to produce a given reduction, so the computer predicts the temperature of the slab at each pass and establishes the relationship between the roll force required and the reduction to be made during the pass.

The stored program enables the computer to change the number of passes and the reduction of metal in each pass once a schedule has been started if the force and torque measured at the rolls vary significantly from the computer predictions. The mill thus rolls each plate in the fewest possible number of passes, minimizing process time.

During passes when the slab or plate thickness is greater than two inches, the roll force and roll separation settings are used with the constants identifying the mill's modulus of elasticity to establish the proper roll opening. When the plate thickness is two inches or less, x-ray gauge signals provide actual thickness information and the computer compares the calculated thickness with the gauge signals. Any errors resulting from roll wear, changes in temperature of mechanical parts, or other effects are righted by automatic correction of the equations in the computer's stored program.

The Prodac control computer for a reversing plate mill is shown in Fig. 4. Similar controls will be applied to tandem hot and cold rolling mills.

Basic Oxygen Converter—A computer control system has been developed for the basic oxygen converter steel-making process because the growth possibilities of this process offer an ideal opportunity for computer application.

The basic oxygen converter process is somewhat similar to the older Bessemer converter process. (See Fig. 5.) Steel scrap, molten iron, and fluxing materials such as coke, lime, and fluorspar are charged into the furnace vessel. An oxygen lance is then lowered into the vessel and oxygen at supersonic speed is blown onto the surface of the charge. The charge is reduced to steel after 15 or 20 minutes of blowing, and the heat is tapped from the furnace into a teeming ladle. Ladle additives are put in to provide the final steel analysis desired, and the steel is teemed into ingot molds.

A complete process study was made, using data from operating plants and from theoretical analyses. The investigators determined the effects of metallurgical and chemical reactions and wrote mathematical expressions to define the quantitative charging of materials and the metallurgical, chemical, and energy relationships that exist throughout the making of a heat of steel. A model of the process was set up on an analog computer, and a digital computer system was developed to control the model.

The result is a system of operation under computer control. Charging the furnace, blowing the charge, and making the additions necessary to reach a desired end temperature and analysis are accomplished with a high degree of automation. By eliminating human error, this control can considerably reduce the number of heats that fail initially to fall within specified tolerances of analysis and quantity.

With a small amount of specific input information and its generalized stored program, the control computer can calculate the combination of variables that will produce the desired end-point temperature, analysis, and weight. These variables include blowing time and the weights of scrap, hot metal, and fluxing materials charged. The system includes programmed control of many functions such as positioning of equipment, initiation of operations, and weighing of charges and additives. Moreover, the system can log process and production data. Inventory control and scheduling of ingot molds, on-line process-cost accounting, and other production procedures are feasible.

advantages of computer control

The computer's high-speed computation and memory capabilities permit process control with a minimum amount of input information from the operator. This minimizes the time and expense required for off-line instruction preparation. Also, a computer control system can interpret data on process and production variables for process studies and comparative performance evaluations. The data can be presented in a form that is compatible with in-plant accounting systems.

The computer also checks the process operation constantly and minimizes waste of time that comes about through human error. The process line is kept working, so material is more continuously in process than is possible in a manually operated process. Probably most important is the control computer's ability to compute the step-by-step procedures required for optimizing production. This minimizes the time and power requirements and insures high product quality and duplication of characteristics.

Thus, the computer control with a proper stored program based on careful study of process requirements and attendant production equipment capabilities uses the processing equipment to its fullest capacity and yet assures staying within safe limits. This prolongs equipment life and minimizes down-time caused by damaging the equipment through overloading.

All this places management in close control of a process when a computer is applied. Management has full control of the stored program and also has the means for checking operation by examining the output data.

Experience has shown that the advantages realized are commensurate with the additional capital investment required. In the general case of a steel rolling mill, computer control usually can be added to the process equipment for one percent or less of the total plant cost. (Total plant cost includes the cost of buildings, processing equipment, electrical drive system, and equipment installation.)

equipment integration

A control computer must be compatible with the mechanical and electrical drive systems of the governed process. Particular attention must be given to defining system inertias, friction effects and variations, mechanical backlash, required accuracies, response times of electrical and mechanical subsystems, and degrees of forcing required to achieve the correct operating times.

The conventional protective devices and subsystems must still be effective. Provision must be made for complete manual takeover, and manual operation must be as effective as it is with conventional equipment.

Computer control requires more sensing devices than does conventional control. For example, a rolling-mill control needs pulse-type position transducers for feeding back digital signals of roll separation and sideguard position. The control must calculate the number of passes required and the drafting practice to be followed, so it needs reliable signals of roll force. These are supplied by load cells that sense the stretching of mill housings. X-ray gauge and width gauge signals provide information for upgrading operations and modifying the information in the stored program. Scale indications of material weight, hot-metal detector outputs giving locational data, and various signals of conditional interlocking permit proper sequencing.

A steel-mill control computer usually has two main sections—base computer unit and central process control. The base computer unit contains the priority director, control unit, arithmetic unit, memory, and power supply; it is selected on the basis of required speed and memory capacity. The central process control contains the input and output units, buffer sections, analog-to-digital and digital-to-analog conversion equipment, and a maintenance panel.

Punched-card input has been preferred, but tape and selector-switch inputs are just as suitable. Cards or tapes can be prepared at the process location or off-line.

Steel-mill processes require a high degree of service continuity to minimize costly downtime, so adequate fault-checking means must be provided. Diagnostic and sampling programs provide fixed maintenance patterns.

analysis and planning required

Before a computer control system can be applied, designers must acquire the fullest possible knowledge of the process. The interrelationships of process variables and their effects on product quality, production efficiency, and costs must be identified; so must the part played by human operators and conventional automatic controls. The physics, mechanics, chemistry, and other theoretical aspects of the process must be established and expressed in mathematical equations, and this theoretical knowledge must be related to field test data. Correlating the quantitative and qualitative data then makes it possible to construct a functional program spelling out the intended use of the computer and detailing its functions. Only then can a system incorporating computer control be designed. After the system has been installed, the initial stored program usually is reevaluated on the basis of test data and upgraded.

The investigations undertaken before applying computer controls to a reversing primary mill are typical. Application, design, and research engineers reviewed existing basic theoretical data and developed new theoretical approaches to the laws of plastic deformation of metal undergoing reduction in a rolling mill. Such factors as the effects of temperature on the metal and the mill, roll force as it in turn affects stretching of a mill housing, oil films in bearings, and analysis of the metal being rolled were defined by mathematical expressions. The engineers also conducted extensive tests on operating mills and in laboratories.

The investigation required about 45 man-months. This includes preparation of functional computer programs but does not include actual programming time for the specific computer employed. Procedures were initiated after start-up to collect operational test data for evaluation and possible modification of the initial stored program.

upgrading existing installations

This article has been devoted mainly to application of computer controls in new installations. Existing installations can also be upgraded by adding computer control.

This problem must be approached through studies and field tests that assure a complete understanding of the process and its operation. Then the compatibility of the existing equipment and arrangement with computer control systems must be evaluated. The existing mechanical and electrical process systems, subsystems, equipment, and procedures often must be modified to assure the best possible conditions for computer control. Finally, a decision must be made as to what degree of automatic control should be applied—data accumulation, programmed control, or on-line computer control. The decision depends, for the most part, on the aims of the process operators, the improvement in process operation that can be expected, and the capital expenditure considered reasonable.

the future

Integration of in-line steel-mill processes offers excellent possibilities for computer control. A central computer could control, as a unit process, such separate and essentially batch-type processes as the operation of soaking pits, a primary mill, a reheating furnace, and a hot strip mill.

Expanding this idea of centralized computer control of related processes introduces the concept of controlling many unrelated processes. Eventually, individual computer control of processes may be supplanted by a central computer working on a sharing basis. For example, the operator of three processing lines could use one computer of the proper speed and capacity to control all three lines economically on a sharing basis through priority assignments. More efficient accounting and information-handling procedures, in conjunction with process control, can be expected as computers take over control of material flow, inventory, wage and incentive pay, and cost accounting for the process.

FUTURE OF COMPUTER CONTROL

W. R. HARRIS, *Manager*
Industrial Engineering Department

E. L. HARDER, *Manager*
Advanced Systems Engineering
and Analytical Department
Westinghouse Electric Corporation
East Pittsburgh, Pennsylvania

In a general sense, the control computer is merely another step in a continuing and long standing trend toward more automatic control. But the amazing versatility and effectiveness of the computer has led to results undreamed of a few years ago, and to future control possibilities far beyond the scope of today's devices. Moreover, the computer has led to a new insight into the functions of previous control devices, and introduced some concepts of information processing and communication that will have far-reaching effects on industrial process control.

All processes and systems have two essential ingredients: One is the *work handling* of the physical material, or the physical service itself, such as steel making, transportation, or automobile fabrication; the other is the *information* necessarily associated with these operations, which in turn must be processed. Associated with this is the business information related to the processes of industry.

The concept of two separate and distinct processes, mechanical work and information processing, was not fully recognized in early control systems. In fact, the information processing operations were usually buried within the machinery, and hardly distinguishable from the power-handling elements. Signals were combined in the fields of a machine, differentiated in a damping transformer, or subtracted in a nulling device; all these are analog computer operations.

However, when a system such as a feedback control is analyzed mathematically, certain signals such as speeds, positions, accelerations, levels, etc., are found to have numerical values ascribed to them, which are acted upon according to fixed mathematical laws. These quantities are information, even though they are processed by the mechanisms of the machinery. As machines and processes have become more complex, however, the information processing or intelligence function has emerged as a separate organ, generally known as a control computer.

Almost without exception, control computers use the digital concept. Analog quantities are still important in machines because most of the inherent relationships following the laws of nature are, and always will be, analog in nature. However, digital representation is not only finding a large place in the processing of information but is beginning to find applications in many areas previously considered the domain of analog devices. For example, there are digital speed regulators and position controllers.

role of the computer

In considering the application of computers to industrial processes, the role of the computer in relation to other control and power devices is important. The generalized picture in Fig. 1 shows some of these relations.

The computer issues commands to the feedback and sequencing control systems. It uses the information about the process gathered from a variety of sensing devices and compares this information with the input data that describes what is desired from the process. The computer then calculates—based on the mathematical model stored in its memory—adjustments to be made or sequencing to be followed to optimize the process in cost, quality, or output.

The control computer also provides interesting potentiality in multivariable control. When each variable is controlled separately, as has been the case in the past, it is difficult to correlate action during transients and to prevent undesirable interaction. However, with a control computer, and the mathematical relationships between each input and output, it is possible to solve from these mathematical relationships the combination of inputs to be controlled and the method of controlling them to achieve the desired effect on one single output. This is known as "noninteracting" control and has been referred to earlier in this series of articles.

The incorporation of these complex relations into the computer has possibilities for control action that would be either impossible or very expensive with independent controls. Also, through predetermined or on-line calculated sequences of settings the control computer can radically change the strategy of control and the distribution of functions between the various subsystems in the computer loop. The superior performance obtainable through this flexibility may be an important part of the economical justification for control computers in the future.

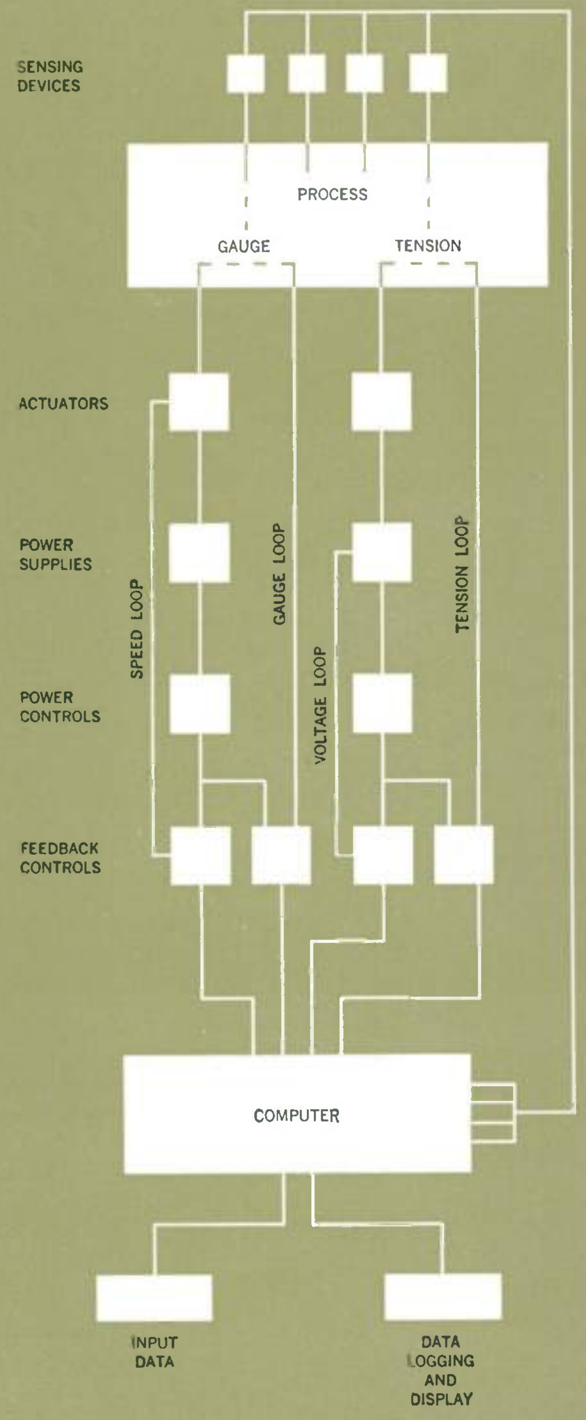
The computer, except in isolated cases, does not take the place of conventional control devices and feedback systems. It is applied in addition to them. The real function of the computer is to extend man's sensory perceptions and his intelligence into productive equipment and to provide closer "management control" of the productive operation.

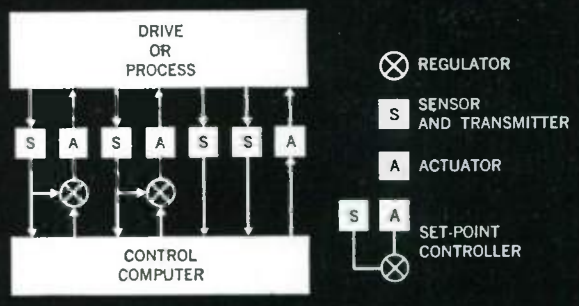
In fact, instead of taking the place of other control and feedback systems, computer control increases the demand for such systems and may impose more stringent performance specifications upon them.

sensing devices

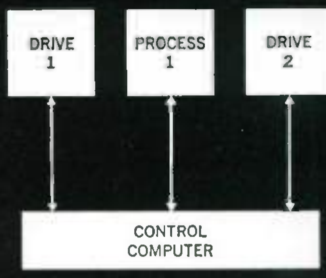
Currently, one of the most critical elements in the control computer system is the sensing device. Increasing numbers of such devices will be required to operate "on-line" to provide process intelligence to the control system.

As automatic control is extended beyond present installations, sensing devices can become a major problem and perhaps a stumbling block to further progress. For example, in electric utility applications, a reliable and satisfactory flame detector is important and much further development is needed. In one of the steel mill processes, a more

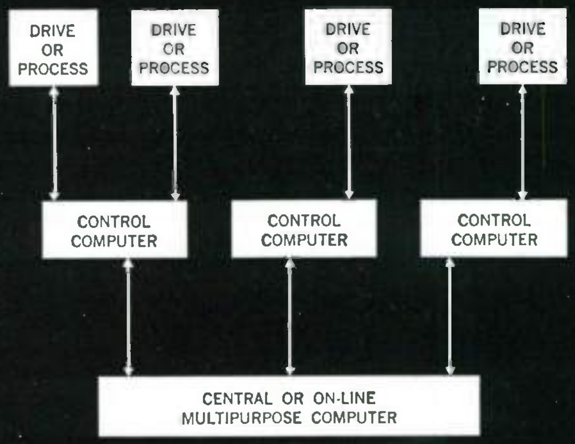




a SINGLE COMPUTER—PROCESS



b MULTIPLE PROCESSES



c COMMUNICATING COMPUTERS

thorough job could be done if sensing devices for continuously measuring the temperature of molten steel, and for continuous analysis of the steel, were available. On card-programmed mills in the steel industry, major effort was required to obtain a hot metal detector that could reliably see through clouds of steam and flying mill scale and operate successfully in the environment surrounding a slabbing mill.

Two major problems exist in the area of sensing devices and instrumentation: First, an extremely broad spectrum of sensing devices with widely differing characteristics is needed for industrial applications; second, the devices are generally needed in limited quantities, which makes their development and manufacture expensive.

Sometimes the computer programmer can circumvent the lack of a sensing device. If the process can be predicted mathematically, the computer can be programmed to calculate the state of the process after a specified time, or calculate the time required for the process to arrive at a specified state. Several of the applications currently being engineered function in this manner. In one of these, a continuous means of measuring the temperature of the molten bath is not available, nor is instrumentation for sensing the analysis of the material. However, through tests of similar processes, a mathematical model has been prepared and programmed into the computer. The computer calculates the amount of coolant scrap to be added and the blow time so that at the end of the process, the correct analysis of metal is supplied at the proper temperature. However, a more thorough job could be done if suitable sensing devices were available.

systems engineering and programming

Systems engineering is a vital element of the new computer techniques. The functions of system engineering are to analyze the problem, plan the control approach, prepare the mathematical model, specify the required components, and coordinate them into a system that will give the performance desired.

The beginning of systems engineering was to tailor individual feedback systems to the user's process and productive requirements. The general technique—which has been used so successfully by American industry—was to separately regulate quantities such as speed, current, temperature, and pressure, so that by keeping all variables constant the output product would be uniform.

The situation is quite different with the new computer controls. Individual feedback units are now only subsystems of a broadened control loop. More and more of the process is being included in this loop, which is now under the guidance of a master computer-controller. This treatment is widening and complicating systems engineering tremendously. For example, as mentioned in a previous article, the plate mill computer control application required extensive studies and consumed approximately 45 man-months of effort. This does not include programming time, which took 12 additional man-months.

Fig. 1 (Left) This shows some general relationships between the computer, control and power devices, and process.

Fig. 2 (Right) Some of the possible combinations of computers and communications systems. Many existing computer applications are similar to the top diagram.

economic considerations

A study of the literature on computer applications shows that surprisingly little has been disclosed about economic justification based on actual plant performance. Until applications are better known and computers more widely applied, proprietary considerations will undoubtedly prevent complete disclosure of findings.

A review of the considerable number of computer and programming applications made in the steel industry shows that the computer cost is one percent or less of the total investment in the process. Therefore, a very modest return on total process investment attributed to the computer is ample justification for the installation. Economic returns can be realized through reduced cost of production, better productivity, more efficient use of raw material, improved quality of product, or maintenance of closer tolerances.

A frequent by-product of computer study is a better understanding of the process and a better arrangement of productive equipment or control. Gathering material for a mathematical model of a process requires an extremely close look at all of the surrounding parameters; this knowledge itself can lead to substantial improvements. In many cases these improvements have been attributed to the computer, but, to be realistic, any study group should subtract that part of the gain that is realized by improvement in the process itself.

To balance the cost of a computer application against anticipated returns, several elements of installation cost must be evaluated. They are:

1. Cost of computer hardware.
2. Cost of system engineering.
3. Cost of programming.
4. Cost of input and output data handling equipment.
5. Sensing devices, which may involve special problems.

Experience has shown that the total cost of the computer installation may be anywhere from 1½ to 4 times the cost of the computer hardware. The ratio depends upon the complexity and difficulty of systems engineering and programming, the amount of input and output data-handling equipment required, and the availability of sensing devices. Good examples of widely different requirements in input and output equipment are the electric utility power plant installations and the plate mill installations in the steel industry. These applications use virtually the same central computer, but a power plant may have more than 1200 inputs and a plate mill from 40 to 100 inputs.

For the greatest economy in any application, the computer must be closely integrated with conventional control and feedback systems and with heavy apparatus. Rather than simply applying a computer, all parts of the installation must be considered together so that the arrangement and the functioning of the entire process are coordinated. Furthermore, the equipment must be designed to meet industrial environmental and reliability requirements. The most successful method of evaluating a computer application is to form a team of specialists to function throughout the final design and installation of the computer.

advances in computers and communication

All of the preceding factors are changing rapidly as technical and economic bottlenecks are broken by new computer developments.

In the past 15 years, the unit cost of calculation has been lowered by some 30 000-to-1. A calculation that took two weeks on a desk calculator and cost about $300, now costs about one cent on a modern high powered computer. Each decade of increase in the calculations-per-dollar has expanded the economic range of applicability of the computer. With higher speeds, more efficient use of components, greater reliability, and much more flexible and varied arrangements for applying the digital technology, this trend should continue. The steady stream of research successes—thin films, cryogenics, tunnel diodes, and molecular electronics—make it almost certain that the digital equipment for each job will be available in more suitable and economical form in the next several years. Most system engineers are counting on this.

examples of systems

Naturally, economics will dictate a building-block approach with flexible elements that can be mass produced. The flexibility available in these building blocks is growing by leaps and bounds, so there will be few limitations to the designer of a complex system.

Some of the interconnected arrangements of computers and communications that are already being considered or actually in use are indicated in Fig. 2. A single control computer functioning in connection with a single drive or process is shown in Fig. 2a. Many of the applications described in the preceding series of articles fall in this general category.

One very obvious extension is a single computer to control both the process variables and the drive variables associated with a single operation; a second small extension is to have a single computer control several or even a large number of separate drives within a plant. In fact, in one recent application, a computer is being studied for the control of several hundred distinct small drives with individual characteristics. This is illustrated in Fig. 2b.

The diversity of problems that a control computer might be called upon to solve may, in some cases, result in interconnection of one or more control computers with a high-powered central computer, as illustrated in Fig. 2c. The local computer does the scanning of the few to several thousand points involved in the process. It controls all of the displays and alarms, provides most of the operator indications for the process, and closes all control loops in which high continuity requirements exist. However, for certain difficult but less frequently needed or less critical computations requiring the services of a higher powered computer, the control computer is in communication with a high-powered central machine. When the central computer is available, the control computer feeds in data, has the complex calculation made, and receives the answers for direct use, or for updating its own mathematical model.

This is but a single illustration of dozens of possible arrangements now being considered for combining process or satellite computers with a central computer. At the other extreme, the central computer, if on-line, might perform practically all of the calculations, with the control computer associated with the process acting as an interpreter, monitor, scanner, data logger, and operating console and performing other functions closely associated with the process control.

In more geographically extended systems, the availability of data communication circuits and of on-line computers that can be tied directly to these circuits is rapidly changing the structure of many business operations. Complete mechanical processing of orders for shipping stocks is already an accomplished fact over extensive wire networks. The execution of the resulting warehouse collection and shipping operations is in various stages of consideration or actuality at present. The extension of this concept to include simple and later more complex manufacturing and process control is certainly under consideration. The existence of on-line computers with communication links should lead to early appearance of systems like Fig. 2c.

Communication channels will also be an important factor in the maintenance of complex systems equipment, particularly of the digital type, inasmuch as most of the diagnostic procedures for isolating trouble involve data input and data output to and from the system; such data could be remotely examined by experts or even automatically if we look far enough into the future. In fact, there is a considerable amount of automatic trouble isolation in the diagnostic and marginal checking programs of both our military and commercial systems today.

conclusion

The systems approach to the overall data processing requirements of an industrial establishment involving scientific and business computing, as well as control computers, will inevitably lead to certain advantageous interconnections. Quite obviously, the business data processor is in need of data on all phases of the operation as input to its various programs. This includes raw data from personnel and manufacturing operations, incoming supplies and outgoing shipments, as well as much more refined data. Some of this same data is used in the control computer for controlling the processes, and some of the production guide quantities required for the on-line operation of the process will also be useful in the business data processing, and vice versa. Consequently, it would be expected that some of the manual links through which this data now gets from one to the other would be closed, as computer interconnections and random access storage systems for buffer storages become better developed. In the meantime, more direct mechanical data gathering will be achieved through scattered input devices, with communication circuits for both control and business computers and advantageous interconnections of these developing. However, inevitably, this initial approach to the problem will give way to overall system design, considering the entire information processing complex and the physical power and material handling of the plant operation as a whole.

While progress must be paced by solid improvement in process knowledge, instrumentation, and economy, the possibilities in improved product quality, higher production, and greatly improved efficiency of business data handling present a great challenge to the engineers and business firms engaged in this development. There is room for much wisdom and much brilliance in applying the new overall systems concept, in harnessing this new technology most effectively to aid in the physical and mental work of producing all of our necessities and luxuries.

DIGITAL DICTIONARY

ACCESS TIME The time required to obtain information from, or place information into memory

ACCUMULATOR A register in the arithmetic unit of a digital computer in which the results of arithmetic and logical operations are formed.

ADDRESS A number that identifies either a location in memory or a computer register.

ALPHA-NUMERIC CODE A code used to express the letters of the alphabet numerically.

ANALOG COMPUTER A computer in which mathematical operations (such as multiplication or integration) are carried out using continuously variable physical quantities (usually voltage).

ARITHMETIC UNIT That part of the digital computer that performs the arithmetic and logic operations.

BINARY NUMBER A number expressed in powers of 2, as contrasted to a decimal number expressed in powers of 10.

BIT *B*inary dig*IT* (0 or 1). The figures used to form binary numbers.

BUFFER A register that holds information temporarily, often to compensate for a difference in rate of flow of information from one part of the computer to another.

CODE A system of symbols and rules for expressing information in a form suitable for computer operation (noun). The process of writing a computer program in machine language (verb).

CONTROL UNIT That part of the digital computer that controls the sequence and execution of instructions.

CORE MEMORY, MAGNETIC A combination of miniature ferrite toroidal cores arranged so that magnetic flux levels can be used to represent stored binary numbers.

DIAGNOSTIC ROUTINE A routine, stored within the computer memory, that the computer or operator can call upon to test for a malfunction within the computer.

DIGITAL COMPUTER A computer that operates with discrete variables, or digits. This computer can solve complex problems with the four basic arithmetic operations—addition, subtraction, multiplication, and division.

DRUM MEMORY A rotating cylinder coated with magnetic material on which binary information is stored in the form of magnetic spots. May also be a rotating *disk*.

FEEDBACK CONTROL A control system that monitors a controlled variable and supplies corrective information in the form of an *error signal*, which is the difference between the desired value of the controlled variable (*set point*) and the actual value of the controlled variable (*feedback signal*). Sometimes called *closed loop* system. May involve analog or digital devices or both.

FLIP-FLOP A device with only two stable states. It remains in one state until signaled to change.

GATE A device that has an output only when certain input conditions are satisfied. *AND*, *OR*, and *NOR* elements are examples of gates.

INDEX REGISTER A computer register that can automatically modify an instruction. It facilitates programming and reduces execution time and the number of instructions.

INSTRUCTION A word, which, when decoded by the control unit of the computer, initiates a specific computer operation.

LOGIC The science of the formal principles of reasoning. Logic is applied in a digital computer for decision making and arithmetic operations. *Logic functions* involve combinations of the basic logical operations, such as *AND*, *OR* and *NOT*.

MODEL, MATHEMATIC A set of mathematical relationships by which system behavior is described.

NOR ELEMENT A gate with multiple inputs and one output that is energized only if all inputs are zero. All other logic functions can be formed from combinations of this element.

ON-LINE OPERATION The type of computer operation where input information is fed directly from the measuring devices into the computer. Results are obtained from current values of operating data in time to allow effective control action. This type of operation is also known as *real time operation*. If the computer operates independently of the time base of its inputs, it is said to be in *off-line operation*.

OPEN LOOP A control system in which the controlled process is controlled from some arbitrary reference, without regard to response of the controlled process. No feedback is used to provide corrective action.

OPTIMIZATION The procedure by which the process is continuously or periodically adjusted to the best set of operating conditions.

OVERFLOW The generation of a quantity beyond the capacity of the register that is to receive the result.

PARITY CHECK A method for checking correctness of information transfer within a computer.

PERIPHERAL EQUIPMENT Input and output devices that work in conjunction with a computer.

PROGRAMMING Preparation of a planned sequence of steps that the computer must take to solve a problem. The programmer may or may not *code* these steps (translate them into detailed computer instructions).

PULSE GENERATOR An oscillator circuit used to generate pulses on a regular repetitive basis.

PULSE TRAIN A group of pulses transmitted in sequence.

RANDOM ACCESS MEMORY A memory system for which the access time is constant regardless of the location being addressed. A *magnetic core memory* is a random access memory.

REGISTER A group of flip-flops that hold binary information.

RING COUNTER A counter consisting of a number of flip-flops connected in a circular pattern so that each input pulse turns the next flip-flop on while all other flip-flops are turned off.

ROUTINE A set of sequential instructions to direct the computer in performing a single operation or a series of operations.

SCAN RATE The rate at which a control computer periodically checks a controlled quantity.

SEQUENCE REGISTER Typically, a counter that is pulsed or reset following the execution of an instruction to form a new memory address, which locates the next instruction.

SERIAL OPERATION Type of operation within a digital computer where all bits of a word are handled sequentially rather than simultaneously. The latter case is *parallel operation*.

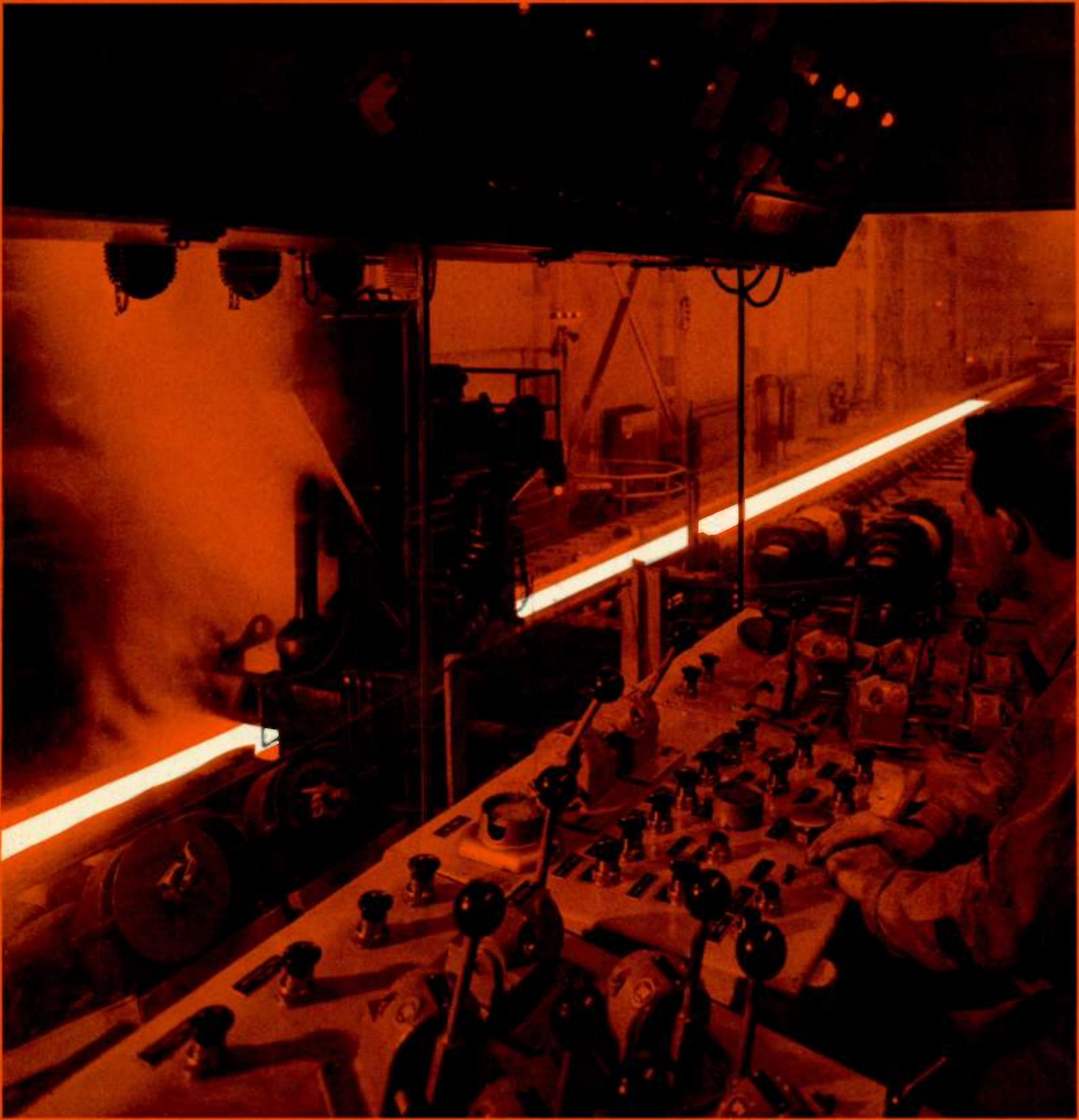
SET POINT In a feedback control loop, the set point determines the desired value of the quantity being controlled.

SHIFT REGISTER A register whose content can be shifted.

SUBLOOP A feedback control loop that is part of a more complex control system.

SUBROUTINE A computer routine, stored in the memory, to which a master routine may transfer to perform such operations as obtaining a square root. This avoids complicating a master routine with repetitive routines.

WORD A group of bits constituting an arithmetic quantity or a computer instruction.



This automatic reversing roughing mill is controlled by a Prodac system. A punched card contains all the information necessary to process an incoming slab. The operator presses two buttons to initiate the mill operation, which then proceeds automatically. The mill is located in Jones and Laughlin's Aliquippa Works.