DOUGLAS PAPER NO. 1814

X1.17

MAN, MACHINE, AND AUTOMATIC TEST OPERATIONS

BY
GEORGE F. MEISTER, JR.
S-IVB SYSTEMS DESIGN GROUP
SATURN GSE ELECTRICAL EQUIPMENT BRANCH
MISSILES & SPACE SYSTEMS DIVISION
DOUGLAS AIRCRAFT COMPANY, INC.

PRESENTED TO
BATTELLE MEMORIAL INSTITUTE
COLOMBUS, OHIO
14 NOVEMBER 1963

DOUGLAS MISSILE & SPACE SYSTEMS DIVISION

The establishment of man/machine relationships within automatic checkout systems is somewhat akin to the weather. Most engineers talk about it, but not too many of these same designers actually get involved, or do anything about it. This is probably brought about by the strange crossbreed of requirements extant for the type of engineer required to solve this problem. For a single individual to be qualified in this area requires a combination of digital systems engineer, operations engineer, programmer, and psychologist, or what is known in the industry as human engineer (implying I'm afraid that there are those of us who are not human). For an individual to be accomplished in each of these fields would require many years of training and experience. Therefore, the usual approach to the problem is to gather together a group of individuals, each experienced in his own area, who try to work together in order to solve the problem. Like all groups, however, the individuals vary in the amount of influence they have over the final group output. The result is that the output generated is roughly equivalent to what the strongest member of the group is capable of, or in more effective cases, what the two strongest members are capable of.

Automation is being attempted for the checkout and static firing of the individual stages of the Saturn vehicle. In addition, it is being attempted for the checkout and launch of the integrated vehicle itself. This attempt is based partly on the success achieved by military missile systems presently operational in the field, and partly on the success achieved by data gathering and parameter predicting systems also operational. Rapid advances in the state of computer design has also contributed to the appeal of automatic checkout systems. However, the communications problem present in the automatic systems now being designed and manufactured is vastly different than most of those previously encountered.

In the typical military missile system the equipment was designed for operating personnel who were considerably less technically qualified than the designers. It was assumed that the operating personnel had little, if any, specialized knowledge of the checkout system or the unit under test. Therefore, the number of displays and readouts provided were generally limited, and certainly not sufficient for any detailed analysis of the test operation. The number of controls and operator options were similarly limited.

The output of typical military systems generally consists of go-no go information, and in some cases, a printed output of the corrective action to be taken. The test procedures themselves are relatively simple and usually include only those operations which could not be implemented automatically. In the design of these systems, the operations engineer was sufficiently experienced to coordinate and, in a large measure, solve the man/machine communication problem. It is probable that some areas required consultation with programmers, digital designers, human engineers, etc. However, the majority of problems were solvable by reference to the overall operational problem.

In computer systems, per se, the input/output console, typewriter and line printer represent the communications link. In most cases the machines are programmed and operated by the same organization, and in many cases, by the same people. The "user" generally never sees the machine, nor is he aware of the actual operation except to know that his code sheets, or whatever the input form is, turns into the output required, such as reams of line printer outputs, etc. The equipment, therefore, is generally designed by and for programmers. Again certain areas arise that require consultation with other disciplines, but the primary influence is that of the programmer.

The integration of high speed digital computers into checkout and launch systems for space vehicles, along with the additional requirements generated due to the R&D nature of space programs, make both the go-no go and the programmer-oriented type of communication link insufficient for use in these systems. In order to understand the requirements for these new systems, an understanding of the type of personnel involved with the operation of these systems is necessary.

The first stumbling block in the creation of an effective communications link is met in the describing of the operational personnel, for there is no single description that will fit these people. Some are the engineers who are responsible for the successful conducting of the test program. Still others are programmers responsible for the programming and operation of the digital computer system itself. Still others are operational types who do not understand the technical aspects of the operation, but are responsible for meeting required schedules and specifications as part of the overall program plan.

The problem is further complicated by the fact that the operating personnel are generally experienced in the field of stage checkout. In fact, for years through the development of those systems we now take for granted such as Thor, Atlas, Redstone, etc., these same test engineers were the people who had to make hardware do the job regardless of the condition of the stage and checkout equipment when received at the operational site. These people have a justifiable pride in their ability and contributions to the success of these programs. The design engineer, on the other hand, has developed a subsystem in which he has a justifiable pride. If there is something wrong with that subsystem, the designer wants sufficient information to be able to firstly convince himself that it is his subsystem that has failed, and secondly, to tell what is probably wrong and what must be done to correct the fault.

To attack the problem of establishing a communications link which is suitable for use by the many types of operating personnel involved requires an understanding of the elements of such a link. The man/machine communications link contains four basic elements. These are:

1. Those controls exerted through the test program.
2. Those controls exerted during the test operation.
3. That information displayed to allow the operator to make decisions with respect to the test processing.
4. That information displayed to allow the operator to judge the operation of the checkout system and the unit under test.

Control Exerted Through the Test Program

Before defining the controls exerted through the test program, it is probably useful to define the test program itself and indicate the steps involved in its generation. The basic computer system is usually a medium-sized, relatively high-speed digital computer. A typical example is the Control Data Corporation 924-A, which is used in the S-IVB Checkout System. With this computer is a group of utility routines to handle such functions as input-output, tape searches, etc., and one or more executive routines. This executive routine normally interprets the test program inputs and causes the proper computer actions to execute the test program. It also monitors the processing of the test program and will generally take care of bookkeeping, processing interrupts, and timing.

The test program itself, then, must be a series of machine words which the executive routine will interpret correctly, and process according to the desires of the test engineer preparing the test program. The test program can best be thought of as a series of calling instructions which represent some required action on the part of the computer, and as a result, on the checkout system. These calling instructions may take many forms. Some of these are forms calling for discrete outputs, analog outputs, time delays, comparisons, measurements and displays. The precise list of calling instructions is a function of the computer and the checkout system.

The problem of man/machine communication is first encountered in the preparation of the test program. It is generally accepted that the person best able to define the test requirements for a subsystem to be tested is the designer of that subsystem. Therefore, the first problem is to establish a means of allowing that designer to express the test requirements of his subsystem in a manner which may be converted into a test program which is usable with the computer-controlled checkout system. The R&D nature of space programs puts an added constraint on the means of getting from designer to test program, in that the time to create and change test programs must be kept short to allow for the development processes that subsystem designs and subsystem test requirements pass through prior to being firmly established.

At present the most promising approach to the generation of test programs which seems to meet most of the requirements of the overall program is the use of a compiler/translater off-line to generate test programs. This compiler/translater accepts as an input either English language or quasi-English language and translates these inputs into the required series of calling instructions in machine language. Quite obviously, considerable gains can be made if the subsystem designer can express his test requirements in English language and engineering units with which he is familiar and have a computer translate these requirements into the proper test program without having to go through a second party. A by-product of this approach is the possibility of standardizing the language used by the design engineer such that his requirements can be understood not only by the personnel of his own company but also by those of other companies and organizations also working on the same overall program.

The first method of control through the test program is the establishment of the sequence of calling instructions in the test program. Some fairly obvious examples of this is the use of such statements as:

Set Command Relay #55

Measure Fuel Tank Pressure

Execute Digital Data Acquisition Subsystem Test

Apply 5.0 VDC to Telemetry Channel 24 Calibration Input

In the S-IVB program, the test requirements are generated in the form of flow diagrams using the types of statements indicated above. These flow diagrams are then transferred to standard computer coding sheets in a test language that has been defined for the checkout system. This language is very similar to the language used in the flow diagrams, with only minor changes made to simplify the actual compilation operation.

In addition to the action calling instructions noted above, a second type of calling instruction is also available to the test engineer. These are known as control operations. These consist of such operations as "Begin", "Halt", "If", and "Go To". Begin and Halt are fairly self explaining. The "If" statement is essentially a conditional jump statement. It is usually written in the flow diagram in a slightly different form. This form is:

Is the Fuel Tank Pressure Greater Than 35 PSIA?

At this point the flow diagram branches to two different paths depending upon the answer to the question. In order to put this statement into a form more compatible to the computer operation itself, the statement is transformed into a conditional jump in the form:

If the Fuel Tank Pressure is Greater Than 35 PSIA, Go To XXXX.

XXXX represents the procedure to be followed if the answer to the question is yes. The Go To statement, used here as a part of the conditional jump, can also be used in the test program as an unconditional jump.

Controls Exerted During the Test Operation

One of the most difficult phases of the system design of an automatic checkout system is that phase during which the general operating concepts are conceived and provisions are made to implement these concepts. It is quite obvious that the checkout system must be usable by the operating personnel, or it is of little value. However, as noted before, the operating personnel who use the checkout system in the type of R&D program represented by Saturn stage checkout, are of varied backgrounds and interests. The operating concepts, then, must in effect provide all things to all people. The test engineer wants a system with which he can get through the test successfully in spite of whatever unforeseen events might take place. The programmer wants a system which allows him to checkout and debug his programs as efficiently as possible. The operations engineer wants a system which he can understand, and which is simple enough that he can have a high level of confidence in its reliable operations. The subsystem designer wants a system that will implement his test requirements, and yet allow him to modify these as experience dictates.

The result of all of these requirements on the operating concepts of the checkout system is, as usual, a compromise. In general, the operating system should be kept simple, if for no other reason than the fact that the computer must process the inputs from the operator and act upon them. An extremely complicated system would put a large burden on the computer itself, and thus reduce its capability to successfully and reliably process the test program. The S-IVB operating system represents a compromise of all of the conflicting requirements put on such a system. In some respects it is more complicated than would be ideal, and some respects it is not as versatile as it could be. However, it does represent a reasonable approach to the problem, which, with actual experience in the field, should meet the needs of the checkout system and its operating personnel.

The operation of the S-IVB checkout system is controlled from a Digital Control Panel (See figure 1). In order to understand the operation of this panel, and how it exerts control over the computer system, it is necessary to go bact to the test program and look at certain elements of it not previously discussed which allow this on-line control. There are two such elements that are a part of the test program. These are the test statement number and the control characters.

The establishment of a test statement numbering scheme is, in itself, a complicated process. The programmers would like the test statement numbers to do all of the machines bookkeeping for it, and yet, the test engineer would like these same numbers to act as a road map to the test program. The test statement number itself has at least three functions. These are:

1. To act as an identifying tag for each test statement.
2. To act as a means of identifying a starting point for a particular operation.
3. To act as a means of indicating to the operator the current point in

the test program.

The third function will be discussed in the next section on Information Displayed to the Operator.

The second function, that of identifying a starting point for a particular operation, is relatively straight forward. By inserting a test statement number from the control panel with the keyboard shown, the operator can direct the computer to go to that point in the test and start processing it from there. Some concern must be given to controlling the number of places within a test program that the operator may start from. This will be discussed shortly under the general subject of control characters.

It is of considerable importance to the programmer that each test statement have some form of identification. Everytime an unconditional or conditional jump is processed within the test program, the location of the new statement must be indicated. In addition, the programmer must have some method of keeping the test statements in the right order and of referring to them when changes must be made to individual or sets of test statements. In most computing systems, a symbolic form of statement identification is used. Before running a program using this form of identification, an assembler must be used to assign discrete computer addresses to the symbolic forms. These assemblers, which are quite common in the computer field, are usually peculiar to each machine, or model of machine. A similar approach has merit for assigning test statement numbers. Instead of an assembler assigning memory locations, however, the same compiler/translator which processes the test language inputs, can also be used to assign test statement numbers. Interestingly enough, it turns out that an assembler must also be used in order to associate discrete memory locations with the test statement numbers.

In addition to the test statement itself, and the number assigned thereto, another character may be used as a part of the complete statement as input to the compiler. This character is referred to as a control character. Four control characters have been defined in the S-IVB system. These are:

1. H - Hold Point - a place in the test program where the system under test is in a quiescent state. It is not expected that the system will stay at this point for any extended period of time.
2. S - Stop Point - a place in the program which meets the requirements of a hold point but has the additional characteristic of being a place where the system may stay for an extended period of time.
3. E - Entry Point - a place in the test program where a test may be entered and executed independently of any part of the test program existing prior to the entry point.
4. N - Do Not Intervene - a statement or series of statements in the test program during which timing is sufficiently critical that it would jeopradize the successful processing of the test program if the computer were allowed to leave the test program in order to do bookkeeping or other non-related operations.

The control characters are added to the test flow diagram after it has been completed. Usually this is done by the programmer or test engineer in conjunction with the subsystem designer who created the original flow diagram. The establishment of control characters may, in some cases, cause revision to the flow diagram. For example, the test engineer may wish to designate an entry point in the test program, but finds that the operation is dependent upon some earlier step in the program. He must then revise the flow if he is to have the entry point desired. This may, of course, simply involve turning off and back on some function in the system under test.

While the Do Not Intervene inputs control the machine internally, and the Entry Point designations allow entry to start a test, most of the on-line operation makes use of the hold and stop points designated in the test program.

Three means of halting the computer operation are provided on the control panel. These are "Hold", "Stop", and "Emergency Stop" switches. The hold input causes the computer to go on to the next hold point and halt at that point. The stop input causes the computer to go to the next hold point and then back up to the last stop point. The emergency stop causes the computer to halt immediately and jump to a predefined emergency shutdown routine.

Through the use of these inputs the operator can either halt the test momentarily to investigate some event or circumstance prior to continuing (such as a voltage or pressure out of tolerance), or halt the test and reset it to a static condition while minor repairs are made or detailed investigation is carried out, or to get the unit under test into a safe condition and the system shut down to avoid what might become a serious condition. Once the test is halted, there must be, obviously, a means of starting it again. Two inputs are provided to do this. These are the "Resume" and "Clear" switches. The Resume input causes the computer to start processing the test program from the point at which it has halted. The Clear input causes the computer to reset the system to the state it was in prior to the beginning of the test program, and to process the next test designated by the operator.

It should be noted that both the Stop and Clear inputs imply a reset routine. It becomes increasingly clear with experience that a reset routine is not necessarily the test program in reverse, and that the reset routine is a constantly varying thing depending upon the status of the system and the current point in the test program. At present, the handling of resets in the S-IVB system is considerably more awkward than is desirable. It is hoped with further investigation and experience that a clean, and hopefully simple method can be established to generate these routines during the test operation.

The controls discussed so far represent a minimum requirement in order to allow the control of the test operation desired by the test engineer. However, the programmers involved do not have, with only these controls, the versatility of operation necessary to properly and efficiently checkout their programs. The programmer requires some control over the mode of operation of the computer with respect to the manner in which the test program is processed.

Four modes of operation can be defined within the structure of the test program. These are:

1. Processing of the test program as an entity without halt unless a fault is encountered or the operator intervenes.
2. Processing of the test program in relatively large increments with lengthy halts between increments.
3. Processing of the test program in relatively short increments with shorter halts between increments.
4. Processing of the test program one test statement at a time.

The first mode requires no implementation, since this is the normal mode of operation. The second mode is readily recognizable as, in effect, processing the test program from stop point to stop point. Therefore, an "Enable Conditional Stops" switch is incorporated on the control panel. This input causes the computer to treat each stop point as a halt instruction. The "Resume" switch is used to continue on to the next stop point.

The third mode can likewise be related to processing the test program from hold point to hold point. Therefore, an "Enable Conditional Holds" switch is incorporated on the control panel. This input will cause the computer to treat each hold and stop point as a halt instruction. Again the "Resume" switch causes the computer to continue. The fourth mode is not incorporated into the S-IVB system directly, but will be discussed later.

Although the above capabilities were incorporated into the system primarily with the programmer in mind, it can be seen that they also represent powerful tools to the test engineer during the actual test operation. Through the use of these switches the operator has the capability of slowing down the operation and in essence going through the test step by step. This form of operation allows the personnel connected with the test to keep current with the test operation and actually verify the operation as it is carried out.

One further capability seemed necessary during the evolution of the S-IVB operating system. It is assumed that none of the test programs will be perfect when they are first used. Nor will all of the possible contingencies have been considered and preprogrammed. Therefore, it is desirable to have the capability of interrupting a test program and jump to some other part of another program in order to accomplish some task that should have been, but was not, incorporated in the original test program. It is equally desirable to be able to return to the original test program at the point from which it was left. Therefore, a "Store Program Point" switch is incorporated on the control panel. This input causes the computer to store its current position in the test program and stand-by for a new starting point to be input by the operator. All of the other control previously described is now applicable to the new program being processed. By turning off this input, the computer is caused to return to the original test program at the point where the input was turned on.

Although this capability was originally incorporated to allow the establishment of a linkage between two test programs, several side benefits are also accrued. Since there is no restriction as to what test program may be jumped to in this mode, there is also no restriction that it necessarily be a test program per se. Therefore, two special routines are presently defined which may also be entered in this mode. One of these establishes typewriter communication between the operator and the machine, which allows a limited on-line modification of the test sequence or the generation of short routines to be executed on-line. The second causes the computer to go into an operating mode such that the test program is processed one test statement at a time dependent upon the operation of the "Resume" switch as the triggering factor. Therefore, even though, as noted before, the single test statement processing mode is not incorporated on the control panel directly, it is implemented in the system through this special capability. Obviously, other special routines can be written as required by future circumstances.

Information Displayed to Allow the Operator to Make Decisions

To a large extent, the information presented to the test operator on which he must base on the spot decisions is somewhat fixed by the controls he has available. For example, he must have an indication of the cause of any halts in the test operation. Therefore, indicators are provided showing the existence of a hold, stop, or emergency stop and whether that halt was generated by the computer or the operator. Also indicators are necessary to acknowledge operator inputs such as conditional holds or stops enabled and program point stored.

Secondly, the operator must have the benefit of the computer's preliminary analysis of any faults that appear in the system. This information can be presented on a CRT Display (which will be discussed in more detail in the next section), on the typewriter or line printer. Regardless of where it is displayed, all of the information available to the computer regarding the fault should be presented to the operator. The operator must then decide whether to continue on with the test, or to clear that test and process another. To specify that a test must be "scrubbed" whenever a fault is found is not reasonable, particularly in the early phases of factory checkout. Therefore, the operator must have the capability of testing around a fault as much as possible to find other faults before shutting down the system. Even in the case of "scrubbing" the test program being processed, the operator must decide which other, if any, test programs may be processed under the circumstances.

Information Displayed to Allow the Operator to Judge the Operation

This type of information covers that data displayed to the operating personnel both during and after the operation. A considerable amount of time can be saved in the overall test operation if sufficient information can be made available to the test operators during the processing of the test such that tests may be "bought-off" as soon after their completion as possible. It is highly desirable that the system configuration used during a test not be disturbed until the test is bought off. This is based upon the consideration that if a fault is found during post test analysis that requires rerunning the test, it is desirable that the system not have changed configuration between the two runnings of the same test program. However, it has been true for many years that more time is consumed setting up and tearing down for tests than is consumed in the actual test operation. The advent of automatic checkout systems does nothing but accent that fact. Therefore, the total time required at any given site to accomplish a testing program becomes primarily a consideration of set up time for the test operations. Reducing the interval between tests by timely displays to the operating personnel can mean a significant savings in time and money for an overall program.

In addition, the operating personnel can contribute to the test operation directly in proportion to their capability of understanding what is occurring during the processing of the test program. This capability is directly related to the amount and quality of the information presented by the automatic system. In the same vein, the automatic checkout system inevitably contains certain manual controls which allow the operating personnel to take over control of the system and put it in a safe condition should the computer or automatic system fail. In order to be able to perform this backup function intelligently, the operating personnel must be aware of what is taking place in the test program and what the state of the unit under test and the checkout system is at any time in the test operation. This is, in fact, a pretty large order for any display system.

The actual hardware used to provide the required information to the operating personnel will vary from system to system. However, it generally falls into four categories. These are:

1. Standard computer peripheral equipment such as typewriters, line printers, etc.
2. Standard recording devices such as strip charts, event recorders, etc.
3. Standard indicators such as meters, lights, etc.

4. Special display devices such as CRT displays.

The typewriter and line printer seem like logical choices for displaying data from the computer. However, each has a serious disadvantage. The typewriter is too slow to output any amount of meaningful data in a short period of time, and is certainly not fast enough to keep the operating personnel abreast of the test program operation. The line printer is much faster, but is difficult to read directly during an operation. In addition, both of these devices require a considerable amount of computer time to operate. The typewriter is useful for small amounts of information, and the line printer is useful for post test outputs. However, neither meets the requirements for on-line information display.

Standard recording devices such as strip charts and event recorders are useful to provide hard copy for post test analysis and a certain amount of on-line monitoring. However, these devices are best used in parallel with the computer system. Attempts to drive these devices leads to considerable requirements in terms of driving hardware, and in the case of analog values, prohibitive computer time requirements to drive the recording devices.

Standard indicators such as meters and lights are useful for monitoring certain critical functions. Again, these devices are best used in parallel with the computer system. In general, the use of these devices is, in effect, defined by the establishment of those manual controls necessary to backup the automatic system.

Because of the above considerations, a considerable amount of attention has been given to the incorporation of CRT type displays into the Saturn automatic checkout systems. Many different types of CRT displays have been, or are being, evaluated for the Saturn program. No effort will be made herein to describe any particular display system. However, certain ground rules should prevail for any such display system incorporated as part of an automatic checkout system. These ground rules are:

1. The display system should provide real time information to the operator in a form that may be readily understood without reference to supporting documentation during the operation. This implies that the display should have an output in engineering units with sufficient identification of parameters such that the information is recognizable to operating personnel familiar with the system under test.
2. The display system should have sufficient flexibility to allow changes to the number and arrangement of parameters displayed. Since one of the attributes of an automatic system is the capability of modifying the test program through changes to the software, and thus reduce the reaction time of the checkout system to changes in the unit under test, this same short reaction time should be achievable with the display system.
3. The display system should be such that it requires a minimum amount of computer time to process and update the information displayed. This implies a minimizing of the total computer requirements of the display system. These requirements may include such things as data processing, tape searches, computer

input-output, etc.

Two generations of CRT display systems presently exist. The first uses the CRT as an electronic line printer. Many models of this generation system exist in use today. This technique combines the readability of the typewriter with the speed of the line printer.

Generally, the line printing CRT display is used in conjunction with a program that essentially dumps a designated portion of memory into the CRT display system. It is capable of presenting real time parameter values by inserting these numbers into predesignated memory locations adjacent to other locations having parameter identification information prestored. While this approach does, essentially, meet the first two requirements for a display system (parameters are displayed in engineering units and the block of memory being dumped can be modified under program control), it falls woefully short of meeting the third criteria. It may be conservatively estimated that four words of identifying data is required for each word of meaningful information (in more complicated systems this ratio may go to ten-to-one or higher), the computer must process, store, and output four or more times as much data as would be necessary under more ideal conditions.

The second generation of CRT displays include some internal memory to alleviate the computer load. Most models of this generation are still in the development stage, although several are now in production. This internal memory varies with models from high speed core memory to slides or film strips. In this type of display the information is broken down into two distinct types, static and dynamic information. The first, static information, is made up of parameter identification and any other supporting information such as limits, etc., that is meaningful. The second, dynamic information, is the actual parameter values measured in real time by the checkout system. These two types of information are then married within the display unit, either within the memory, or optically, or electronically in the video portion of the display system.

In those display systems using core memory, the computer need only output the parameter values as long as the number of parameters being displayed in any test sequence does not exceed the memory capability supplied for static information. If the memory capability is exceeded, the computer must then also output that information as in the case of the CRT line printer. However, this static information need only be transmitted once, as compared to the first generation display system where it must be transmitted each time the parameter values are updated.

In those displays where slides or film strips are used, the static data is contained on the film and does not require any computer time to generate or transmit. This technique provides a large amount of internal storage at a very low cost and in a form compatible with video displays. This technique has the added advantage that the static information may be changed off line without requiring digital equipment of any kind.

In systems the size of those being developed for stage checkout on the Saturn program, a single display system seems to be adequate, although experience may show that two or even three such displays, each independent of the other, are required. As long as the third criteria is met by these display systems, that is that the computer requirements to process the display systems is kept to a minimum, it also is evident that a computer of the capability commensurate with that being used in the S-IVB system can handle both the display and test program processing simultaneously.

However, when the launch of the integrated vehicle is considered, or when future programs involving even bigger and more complicated systems are considered, it becomes obvious that multiple display systems are required due to the large increase in the amount of information required in a given time. It also appears that these displays will be of sufficient number that they must either operate essentially independently of the computer controlling the test operation, or that a second computer will be required to service the display systems. For the display system to operate essentially independently of the computer will require yet another generation of display systems to be developed. Considering the cost and time involved in the development of the second generation, it is more likely that a second computer servicing a group of the presently being developed second generation display systems will be incorporated in these larger systems.

The foregoing discussion of the establishment of a satisfactory man/machine relationship in automatic checkout systems may be summarized quite briefly. It must be realized that the establishment of the required communications between the operational personnel and the machine is one of the largest single tasks before the system designers of automatic checkout systems. The requirements imposed upon the system are far reaching and cannot be established too early in the system design. The definition of the process for generating test programs, the test language, the test statement numbering scheme, control characters, on-line controls, on-line indications, and display systems affects every facet of the hardware and software design. The computer is built into the system to help the man do a job. The man must be able to communicate with it and control its operation. Until such time as we are prepared to turn the entire operation over to computers (a situation which may or may not ever occur) it can be no other way.

DIGITAL CONTROL PANEL

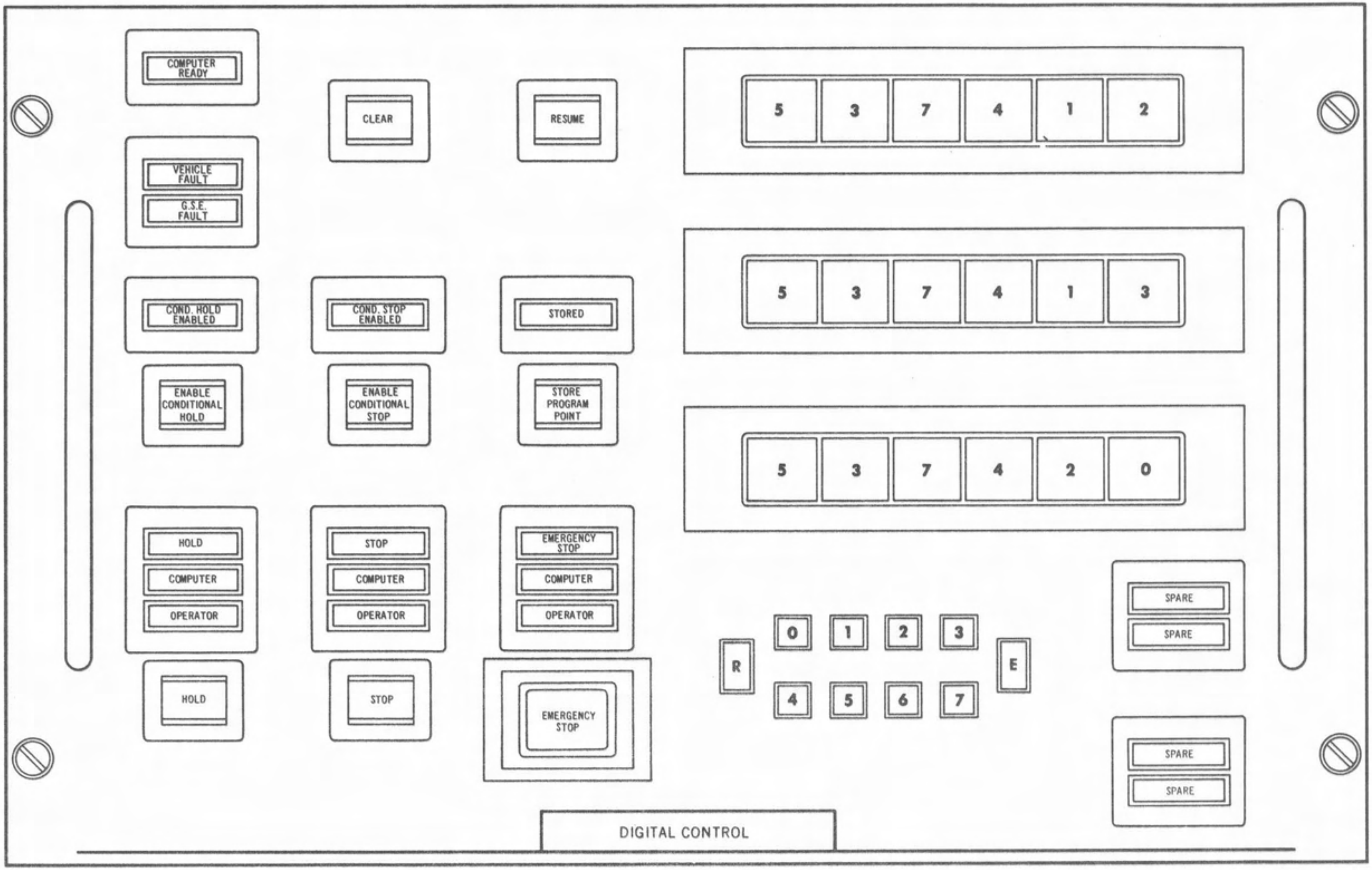

FIGURE 1