A Real Time Operating System for the Saturn V Launch Computer Complex

XIV.7

[redacted]

IBM No. 67-U60-0010

7/66

A REAL TIME OPERATING SYSTEM FOR THE SATURN V LAUNCH COMPUTER COMPLEX

by

Frank R. Palm

July 1966

INTERNATIONAL BUSINESS MACHINES CORPORATION
Post Office Box 1250, 150 Sparkman Drive
Huntsville, Alabama 35807

The first Saturn V launch will mark a significant milestone in the United States' effort to place man on the moon. To ensure a safe mission, extensive prelaunch vehicle checkout of various vehicle systems is required. On most previous missile systems, checkout was accomplished manually with the vehicle on the pad many months prior to launch. As vehicle systems became more complex, automation was necessary to ensure safe, accurate, and efficient vehicle checkout. On the uprated Saturn I, the Saturn IB, some automatic checkout has been accomplished, and will be further extended on the Saturn V.

Personnel involved with checkout often have many years of experience in checking out missile systems; therefore, their formulated ideas in regard to accomplishing checkout — especially in the early stages — leaves little room for reliance on a computer system. Consequently, automation is done in steps. In each step the test engineer is allowed absolute control at any time over the system. The checkout system has the capability to analyze and react to various vehicle conditions and also allows the operator to override these decision.

To perform Saturn Vehicle automatic checkout, the National Aeronautics and Space Administration selected a three-computer complex consisting of two RCA 110A Computers and a DDP 224 Computer. The three computers in the system are connected in series by data links. The master computer — an RCA 110A — located in the Launch Control Center, is referred to as the Launch Control Center Computer (LCCC). This computer is connected to the other RCA 110A, located on the Mobile Launcher, and is referred to as the Mobile Launcher Computer (MLC). The master computer is also connected to a DDP 224, the Display Control Computer (DCC).

The DCC is utilized to operate the display system which consists of 15 display consoles — each of which has a refresh memory. The engineer, by viewing these display consoles and other equipment panels, is kept informed of vehicle status. From these consoles and panels, the engineer can control the checkout.

The LCCC schedules and executes vehicle test programs and monitors system parameters.

A REAL TIME OPERATING SYSTEM FOR THE SATURN V LAUNCH COMPUTER COMPLEX: PALM-3

The Mobile Launcher Computer handles direct communication with the vehicle.

In order to utilize this computer system, IBM has written a versatile operating system to handle the communication and overall system supervision, allowing fast and efficient vehicle checkout with the test engineer having complete authority.

The operating system has three main functions:

(1) Handle the intercommunication between the test engineer and the vehicle. This involves commands transferred from the test engineer through two or three computers to the vehicle and the reverse path.

(2) Monitor vehicle data. System parameters are checked against limits, and out-of-tolerance conditions are reported to the test engineer.

(3) Provide capability to execute vehicle test programs as specified from a display console.

The operating system description and some of the computer characteristics are as follows. The main checkout computers, the RCA 110's, have a 32,768-word memory and the following input/output devices: a

32,768-word drum, 5 magnetic tape drives, a card reader, card punch, line printer, data link, a Digital Data Acquisition System, and discrete input/output channels. The Digital Data Acquisition System reads vehicle system parameters into the memory of either of the checkout computers. The discrete input/output equipment is used by the LCCC to read switch position status and to issue the outputs which turn Off/On lights. The discrete I/O equipment in the MLC sends signals to the vehicle and receives discrete signals from the vehicle. The computers also have a priority interrupt feature with four levels of interrupt. Each RCA 110A has three interval timers which are available to the programmer. A binary number, placed into one of the counters, is counted down at a one millisecond rate. On a zero count the timer causes an interrupt on the assigned priority level. These timers are used to time vehicle reactions and to eliminate any program hang-up caused by an unexpected hardware condition.

Here is a list of the major operating systems components.

The Interrupt Processor — This is one of the most important programs in the Operating System. It controls the various interrupting conditions, receiving control as soon as an interrupt occurs. It determines what has caused the interrupt and which program will handle it. It also performs some routine bookkeeping consisting of saving and restoring some program registers. Because there is a certain amount of inflexibility in the assignment of interrupts, this program makes possible a pseudo level of processing. Often an interrupt occurs and some portion of the processing is done on the interrupting level; however, if the processing is lengthy, the user program requests of the interrupt processor to be put on the pseudo level. The interrupt processor then services all pending lower-level interrupts and returns control to the requesting program on level zero. The originally interrupted program has not been restarted and will not be until the pseudo level is accomplished. Because the system is running on level zero and other interrupting condition may occur, the interrupt processor takes care of recursive reentry problems which might occur while this pseudo level is executing by allowing only one such function to be in execution at a given time. If a second function is requested, the interrupt processor sets a flag which is interrogated at the conclusion of the first function on the pseudo level.

Test Program Control — Since one of the primary functions of the checkout system is to perform vehicle and subsystem tests, the operating system contains a Test Program Control routine. This program is the primary interface with the test engineer at the display console. It analyzes his key-word input to determine test legality and if his console identification number is allowed to execute that test. Test programs are run one at a time; however, the test program control provides test engineers the capability to request execution of more than one program. These are scheduled on a first-come, first-served basis. One display console, containing a master identification to the program, can control all test program execution. It may: a) cancel any program from the stack, b) terminate any program, and c) request immediate execution of any other program. Individual consoles can also cancel or terminate those requests which have been executed from that console.

Input/Output Control System — The Input/Output Control System program services the tapes, drum, printer, card reader, and card punch. It performs the normal function of reading and writing of these devices doing the normal error checking. This program allows the input/output processing to be done in parallel with the computing. Since an input/output device failure can delay the system, this program is designed to enable the system operation to continue if a particular device fails. For example, if a request is made to write information on a tape for post-processing and the tape is inoperative, the program will switch to a second tape without data loss. If this second tape is also inoperative, the program will indicate to the user program that the requested operation is complete and the data will be lost. It is preferable that the entire system stay up and control be maintained, rather than bring the system down due to loss of an input/output device. This philosophy allows the test engineer or test conductor to determine when the system must be brought down, but the operating system continues to execute as long as possible.

Digital Data Acquisition System Executive — The Digital Data Acquisition System (DDAS) is used to access the vehicle system parameters. On a cyclic basis this system accesses analog and discrete indications from the vehicle. The analog parameters are converted to digital and stored in the DDAS memory. Words from this memory can be read into either of the computers. A separate executive is written to handle the reading of this device since it has several different modes of operation which are peculiar to the hardware. Data can be read in as a straight memory-to-memory transfer; e.g., a program may request to read "location X from the DDAS memory into location Y" of the RCA 110A computer memory. In another mode, data can be read in such a way that what is presented represents the latest available based on the DDAS scan rate.

The Data Link Executive — The operating system contains a data link executive which supervises the flow of information between the LCCC and the MLC. This data link executive schedules transmissions from one computer to the other, tests hardware for failure conditions, and sends appropriate command words to ensure that transmissions arrive correctly. The system consists of two data links, of which one is a redundant path. When one of these data links becomes inoperative, the data link executive switches to the other link and continues processing without loss of information.

Monitors — The operating system provides two types of monitor programs. One type is used solely for display information.

The parameters monitored by this program may be predefined or dynamically defined during system operation. A test engineer at the console may have the monitor program read system parameters on a periodic basis. These parameters may be checked for out-of-tolerance conditions and displayed on the display console. The other type of monitor function is an executive which allows the execution of special-purpose subroutines on a time-oriented basis. This useful tool allows execution of small monitor subroutines even while other tests are being performed. The program has a predefined polling sequence of these subroutines and a capability to run special routines out of sequence. No subroutine is executed until it is requested from a test program, from a test engineer at a display console, or from another subroutine. Once in execution, the routine remains in execution until it is terminated by any one of these functions.

Discrete Executive — One of the most important programs in the operating system is the one that handles the communication flow from the test engineer's Electrical Support Equipment panel to the computers and from the computers to the vehicle. From these panels the test engineer controls outputs to the vehicle, permits or inhibits test program control of outputs to the vehicle, and receives vehicle status as reflected on his panel lights. In keeping with the philosophy of the test engineer having total system control, most of the system switches are three-position switches. Since most of the same discretes can be issued by both test programs and switch position changes, the three-position switch is utilized to give the test engineer a combination of automation and absolute manual control. The switch normally controls one output function which can be issued by a test program only when the switch is in the center position. The operating system not only responds to the switch position changes by issuing the proper commands to the vehicle, but also maintains the inhibit status of each output to the vehicle. Whenever a test program elects to issue a discrete, it accesses this discrete through the discrete executive. In the executive, the inhibit status of that function is checked and, if inhibited, no command is issued to the vehicle.

The normal sequence of events when a switch position changes is as follows:

(1) The switch position change occurs.

(2) An interrupt occurs at the LCCC.

(3) This interrupt is interpreted by the interrupt processor and determined to be a discrete input.

(4) Control is transferred to the discrete executive.

(5) The discrete executive accesses a status table and a history table so that it can determine which input has changed; it accesses an action table where response information is stored for each discrete input.

(6) The normal response is to transmit a message utilizing the data link executive and the data link hardware to the MLC.

(7) When the message arrives at the MLC, the discrete executive issues the appropriate command to the vehicle. An almost identical reverse path exists for information which comes from the vehicle back to the test engineer's electrical support equipment panel. The vehicle status changes which causes an interrupt in the MLC. After processing by the interrupt processor, the discrete executive, the data link executive, and the discrete executive in the LCCC, the appropriate command is issued; and a light on the panel is turned off or on.

Display Programs — International Business Machines Corporation writes the programs in the LCCC which interfaces with the display computer. These are primarily the data link executive for data link transmissions and a display service routine which handles program requests for display of information on the display console. It is also possible for programs to request "Display Descriptions." These consist of predefined background data and conversion information. The background data is displayed on the display console, and the conversion information is utilized by the display computer to scale the raw binary data as it is sent to the display computer for display on the screen.

System Recovery — The overall system must be reliable, and since the MLC is unmanned during launch operation, it is important that the operating system provide not only the capability to operate when input/output devices fail, but also that it have the ability to recover from system malfunctions. The operating system in the MLC contains a bootstrap loader which is kept on the operating system tape and the drum. Both of these loaders can be accessed by a load drum or load tape switch remotely from the Launch Control Center, allowing the system to be reloaded if either the drum or tape is operative.

Utility Executive — The operating system also provides the standard utility type programs that are necessary in any computer system — such as loaders, dump routines, snapshot routines and memory alter routines.

Most of the utility capability can be controlled from the display console, and output can be to the console, to tape or to the printers. The operating system also contains programs to read the various clock systems that are input to the computer.

Reliability and Safety Considerations — One frequently-asked question is: Why are test programs executed in the remote computer instead of at the computer closest to the vehicle? In the uprated Saturn I, the Saturn IB series, the test programs were executed at the computer which was located near the launch pad. However, the LCCC can be maintained, if necessary, during a countdown hold. Since the MLC is not manned, no maintenance is possible during a countdown. The goal on the Saturn V system is to pull all possible functions back to the LCCC. The initial version of the operating system contains a limited amount of system reliability and recovery capability. This capability can be increased on future shots.

The discretes via the switches are one of the most positive means of control which the test engineer has over the vehicle status. If the test engineer loses the status of the discrete inputs or outputs (due to some hardware or software condition), a discrete re-initialization is necessary.

This may have been caused by data link hardware failure, a program bug, or a computer failure. If loading of the operating system is necessary, it is accomplished first. Once the operating system is running correctly, the test engineer has several options which he can exercise. As soon as the operating system begins to support the vehicle, the status of the switches and the discrete status of the vehicle are compared. Any differences that exist are printed and displayed. The test engineer can change switch positions to agree with the vehicle status, or he can give the Go-command which causes the operating system to issue all discrepancy conditions in the state specified by the switch.

One of the most serious problems in implementing the system has been documentation. Because of the severe schedules and the large number of users, it has been a problem keeping a master set of listings and flow charts up to date. To accomplish this, we have implemented a method of utilizing the same input for both the assembler and automatic flow charter. This symbolic input, along with the binary assembled information, is maintained in a compressed format on a disc. From this file of information, a user may request a system tape, a listing and/or flow chart of any program in the system.

A REAL TIME OPERATING SYSTEM FOR THE SATURN V LAUNCH COMPUTER COMPLEX: PALM-15

Checkout of the programs is accomplished on a breadboard facility at NASA's Marshall Space Flight Center. This facility contains the three computer complex, some of the actual vehicle equipment, and simulators for some of the equipment.

The following illustrations were used as slides in a presentation of this paper at the Real Time Systems Seminar, Houston, Texas, November 1966.

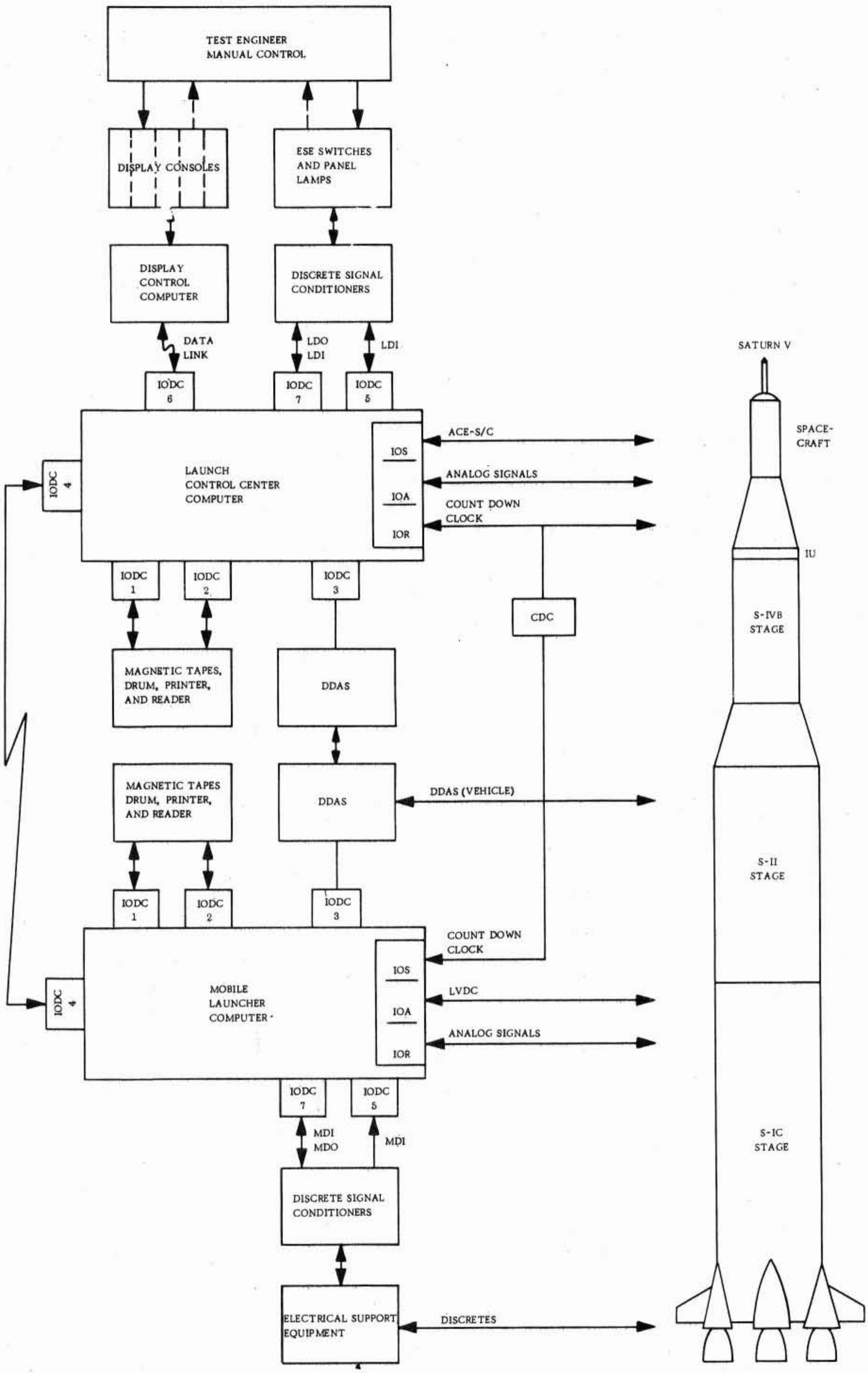
MISSION REQUIREMENTS

1 - TEST ENGINEER - VEHICLE COMMUNICATION

- SWITCH SERVICING
- VEHICLE STATUS DISPLAYS

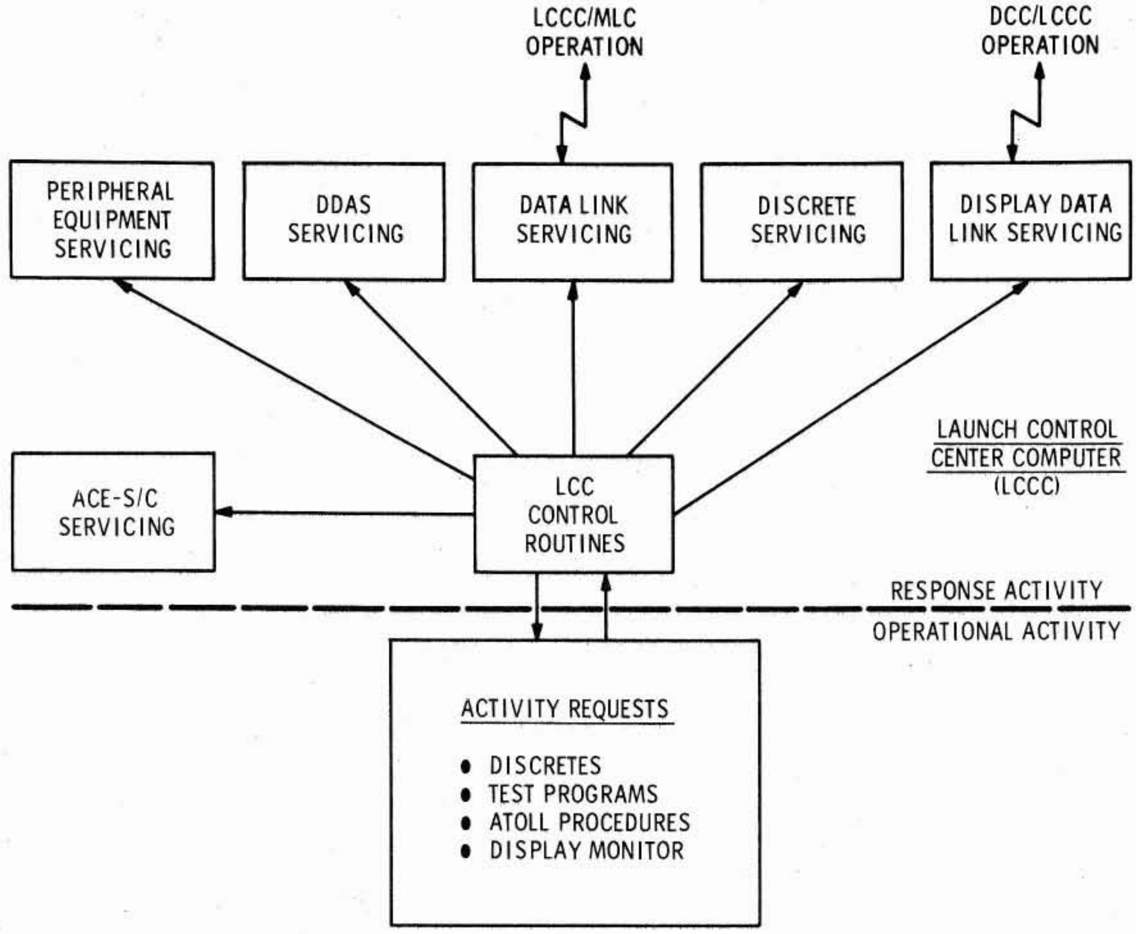
2 - MONITOR CAPABILITY

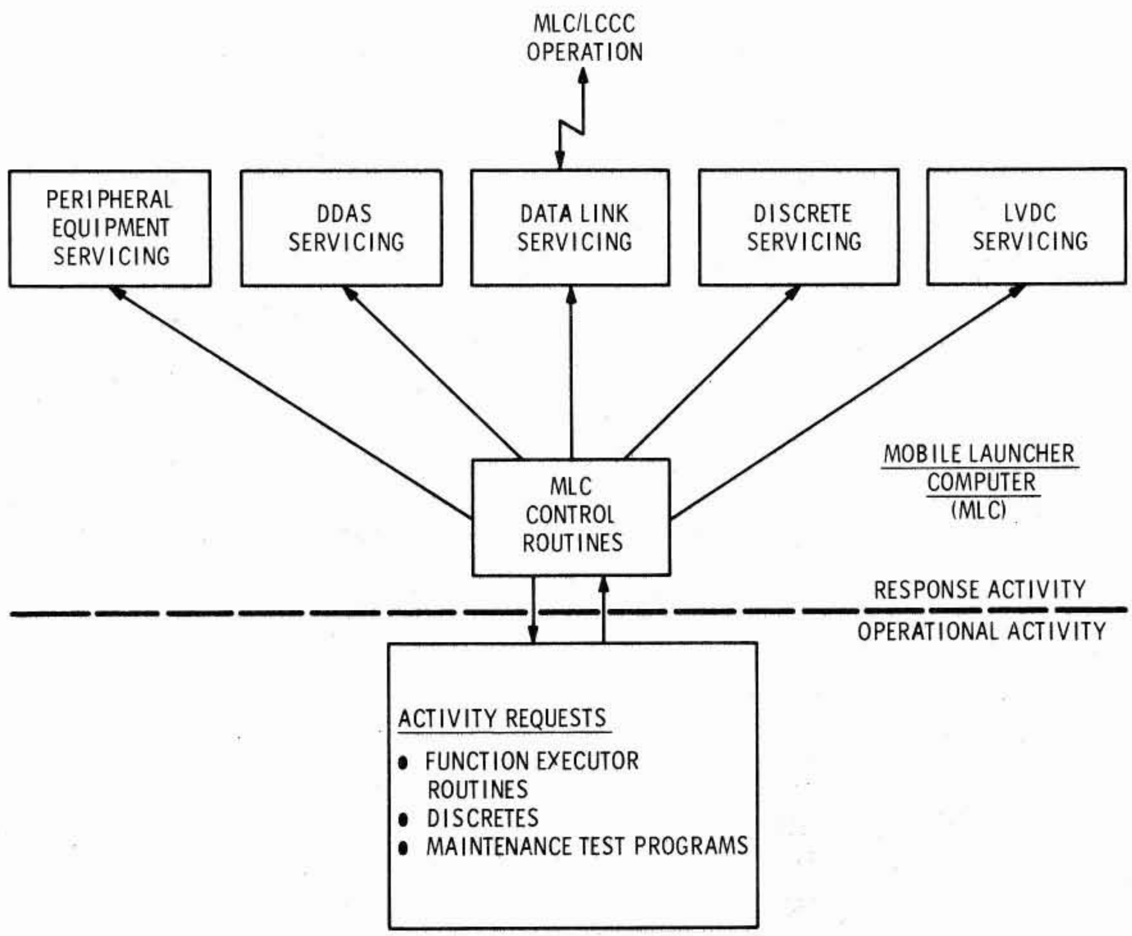
3 - AUTOMATED TEST PROGRAMS

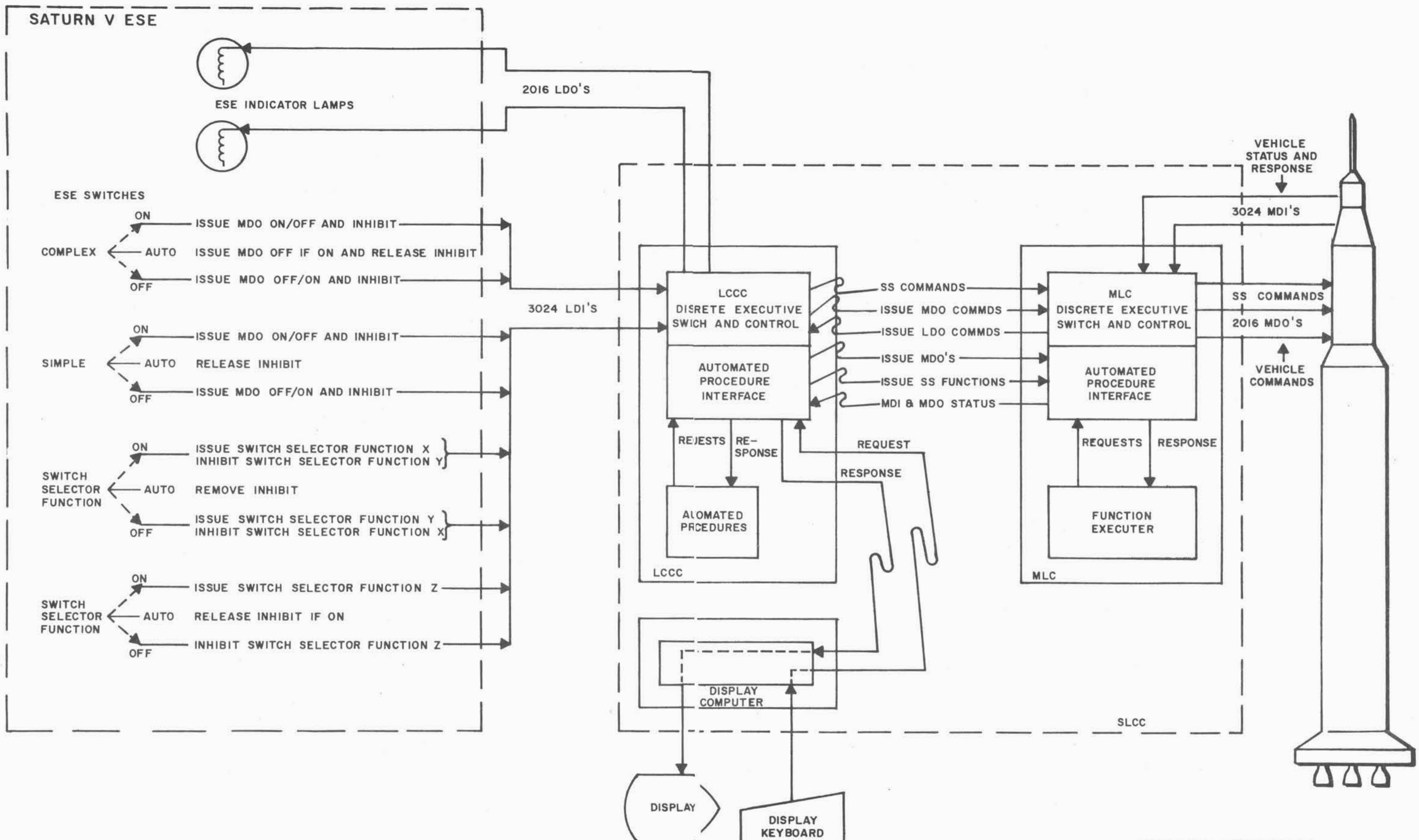


TEST ENGINEER MANUAL CONTROL
DISPLAY CONSOLES
ESE SWITCHES AND PANEL LAMPS
DISPLAY CONTROL COMPUTER
DISCRETE SIGNAL CONDITIONERS
DATA LINK
LDO
LDI
LDI
IODC 6
IODC 7
IODC 5
LAUNCH CONTROL CENTER COMPUTER
IODC 4
IOS
IOA
IOR
ACE-S/C
ANALOG SIGNALS
COUNT DOWN CLOCK
IODC 1
IODC 2
IODC 3
CDC
MAGNETIC TAPES, DRUM, PRINTER, AND READER
DDAS
MAGNETIC TAPES DRUM, PRINTER, AND READER
DDAS
DDAS (VEHICLE)
IODC 1
IODC 2
IODC 3
MOBILE LAUNCHER COMPUTER
IODC 4
IOS
IOA
IOR
COUNT DOWN CLOCK
LVDC
ANALOG SIGNALS
IODC 7
IODC 5
MDI
MDO
MDI
DISCRETE SIGNAL CONDITIONERS
ELECTRICAL SUPPORT EQUIPMENT
DISCRETES
SATURN V
SPACE-CRAFT
IU
S-IVB STAGE
S-II STAGE
S-IC STAGE

SUMMARY OF OPERATING SYSTEM FUNCTIONS







DISCRETE EXECUTIVE
DATA FLOW CHART