MCR-69-192 (Rev. 2)

8

FLIGHT PACKAGED ONBOARD CHECKOUT SYSTEMS

DEVELOPMENT UNIT

SOFTWARE DOCUMENTATION,

OVERALL SPECIFICATION - LEVEL 2

TOOL - TEST ORIENTED ONBOARD LANGUAGE SYSTEM

NAS 9-8000

March 18, 1970

Prepared for

NASA MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

MCR-69-192 (Rev. 2)

FLIGHT PACKAGED ONBOARD CHECKOUT SYSTEMS

DEVELOPMENT UNIT

SOFTWARE DOCUMENTATION,

OVERALL SPECIFICATION - LEVEL 2

TOOL - TEST ORIENTED ONBOARD LANGUAGE SYSTEM

NAS 9-8000

March 18, 1970

Approved

W. J. Wise, OCS Program Manager

Prepared for

NASA MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

FOREWORD

This document is submitted in accordance with Task 3.2a of Exhibit A to Contract NAS 9-8000 and constitutes partial compliance with paragraph 3.1.7 of Appendix D to the subject contract.

1.0 IDENTIFICATION NUMBER: P-TOL-000-00

2.0 NAME OF SYSTEM: TOOL - Test Oriented Onboard Language

3.0 PURPOSE:

The purpose of this system is to translate and edit test sequences, and to execute translated test sequences. Test translation consists of converting operator keyboard inputs, which are responses to displayed cues, into a test sequence data list. The operator may construct an entirely new test or, by utilizing the review capability, modify an existing test. Test execution is accomplished by a set of test element processors. These processors interpret the contents of the test sequence data list in order to perform the desired element operation.

4.0 LANGUAGE DESCRIPTION:

The OCS Test Oriented Onboard Language (TOOL) is a test language oriented to the task of spacecraft checkout.

4.1 Organization Structure

Figure 1 illustrates the test structure.

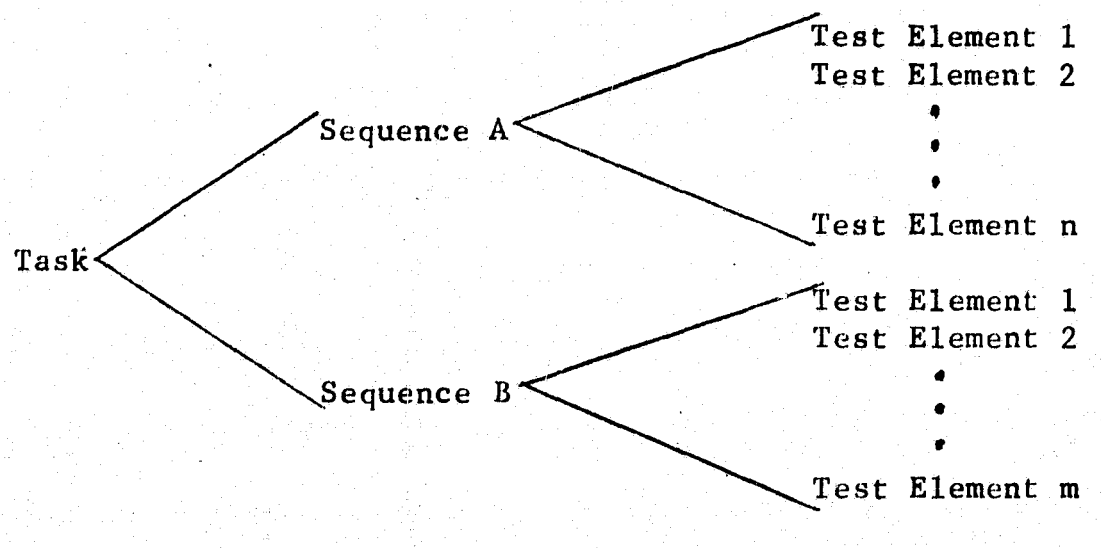


Figure 1

Test Element

The test element is the basic unit of the test language. Each element performs a specific operation such as commanding a switch closure, generating a stimulus, etc. Associated with each element is a set of modifiers. These modifiers serve to amplify the element to specify uniquely the required operation. An element may be uniquely defined by prefixing it with a number. The purpose of this prefix is to define reference points for branching and looping within a test sequence.

Test Sequence

A test sequence is a collection of test elements arranged in a logical order to accomplish a specific testing or control function. A test sequence can be dependent upon other sequences. Hence, one sequence can set up a test and another perform the test. The Go and No/Go logical paths through a test sequence shall be defined by properly linking sequences and elements together in an unambiguous manner. A checkout task could consist of a single sequence which contains calls to other sequences.

Tasks

Task is the name for the highest level of operation. The main sequence running on a level of concurrency is a Task. A Task may be a single sequence or a series of calls to other sequences. In this document when referring to levels of concurrency the words task and sequence are used interchangeably.

4.2 Sequence Translation

Sequence translation can be performed off-line using the CDC 160 A/G or on-line using the IBM 4 Pi EP. Translation consists of converting alphanumeric source input into a test sequence data list.

The Off-Line Translator (TLA) uses punched cards for input. The output consists of a test sequence data list, punched on paper tape or cards, and a printed hard copy.

The On-Line Translator (TOOL) uses keyboard entries for input. These entries are cued or requested by the translator in order to facilitate the operator's entry. Each entry is checked for validity and then converted into test sequence data. The test sequence data is placed directly into the test sequence data list, which resides in memory. The On-Line Translator also allows the reviewing and/or modification of existing test sequences.

4.3 Sequence Initiation

There are four methods by which a sequence can be initiated:

1. Manually: where the operator selects and initiates the sequence through the CADU keyboard.

2. Clock: where the sequence is initiated at a specified real time.

3. Monitor: where the sequence is repeatedly initiated at a specified cyclic rate.

4. Remote: where the sequence is initiated from a remote location.

4.4 Sequence Execution

A sequence can be executed at any one of four independent levels. These levels are designated A, B, C, and D. A sequence is assigned

to a level during execution initiation. Since these levels are independent, the test composer need not be concerned as to which level his sequence is to be executed.

Level Priority

Associated with each level is a priority counter. The contents of this counter determine the number of elements a level will execute before transferring control to the next level. The cycling of the levels is as follows:

E(A1),E(A2),...,E(An)
E(B1),E(B2),...,E(Bn)
E(C1),E(C2),...,E(Cn)
E(D1),E(D2),...,E(Dn)
E(An+1),E(An+2),...,E(An+n)
.
.
.
E(Dn+1),E(Dn+2),...,E(Dn+n)
E(A2n+1),E(A2n+2),...,E(A2n+n)
.
.
.

where E is an element processor; A, B, C, D denote the levels; and n is the priority count. If a particular level is not executing a sequence, it is removed from the element execution cycle.

Timing

Since all sequences share the same resources, element execution may have to wait because a required resource is not available. Also, the time between elements will vary depending upon the number of active levels.

Modes

A sequence can be executed in two different modes: continuous or segment. The primary difference between these modes is with the execution of the Display element. This element is normally executed only in the segment mode, and it always causes element execution to stop. Therefore, the "segments" in segment mode are defined by the Display element.

Read Monitor Status is a modified form of the segment mode. The difference being that in this mode only those sequences that have a change in monitor status are executed.

4.5 Resource Allocation

The purpose of resource allocation is to assign the available hardware and software resources to various functions only for the time that they are required. Thus, a particular resource, such as the Measurement Unit, is not tied up by a single sequence but passes from level to level as needed.

There are two categories of resources: hardware and software. The hardware resources are: SSU's, SGU's, MU's, MSU's, timers, and display. The software resources are the four levels of sequence execution. Level A is dedicated to Monitor sequence execution when they are enabled. Level B is dedicated to clock initiated sequences when they are enabled. When either of the above is disabled, the respective level becomes an allocatable resource.

Whenever an element requires a resource, it tries to allocate it. If the resource is available, the element is executed. If the resource is not available, control is transferred to the next active level. On the next level cycle, the element again tries to allocate the resource and the operation is repeated.

4.6 Control and Display Unit

4.6.1 Alphanumeric Display Unit (ADU)

The ADU is divided into four fields: level dedicated lines, work area, systems message line, and keyboard response line (see Figure 2).

LINE	ADU			LEVEL
1	SEQ NAME	- S -	MILESTONE	A
2	SEQ NAME	- S -	MILESTONE	B
3	SEQ NAME	- S -	MILESTONE	C
4	SEQ NAME	- S -	MILESTONE	D
5				
6				
7	WORK AREA			
8				
9				
10				
11	SYSTEM MESSAGES			
12	KEYBOARD RESPONSES			

Figure 2

4.6.1.1 Dedicated Level Lines

Each level has dedicated to it one of the first four lines. In each of these lines the first eight characters display the name of the sequence currently being executed; the tenth character is the level status; and the last ten characters are for milestone messages. The level status is:

R sequence is Running
D sequence is Displaying
S sequence is Suspended because of a delay in element execution or waiting for a hardware response
W sequence is Waiting for a resource (other than the ADU)
A sequence is Aborted due to an operator command
P sequence is Aborted due to a programming error
E sequence is Aborted due to a hardware error
M sequence is Aborted due to a math out of bounds error
N sequence needs the ADU

The milestone message is updated and displayed each time a Milestone element is executed. Initially, the milestone message "STARTED" is displayed. Any time a Display element cannot allocate the display resource, the milestone message associated with the level is replaced by "DISP REQ".

4.6.1.2 Work Area

These six ADU lines, along with the Microdata Assembly, are the display resource.

4.6.1.3 System Messages

Messages pertaining to the operation of the OCS System are displayed on this line.

4.6.1.4 Keyboard Responses

All inputs entered through the CADU alphanumeric keyboard are displayed on this line.

4.6.2 Microdata Assembly (MA)

This unit, along with the six line ADU work area, is the display resource.

4.6.3 Display Resource

The display resource is utilized by the Display element, the On-Line Translator, and those pushbutton programs that require cues.

4.7 Test Protection

Protection Keys

Associated with each sequence are five protection keys. The purpose of these keys is to prevent improper operation or alteration of the sequence. These protection keys are:

1. Review inhibit
2. Modify inhibit
3. Delete inhibit
4. Keyboard Execution inhibit
5. Call Execution inhibit

If the key is on, set to one, the function is inhibited. These keys are set during initial translation and cannot be altered.

Password

The function of the password is to allow only authorized personnel access to a sequence. If no password is specified, the sequence has public access.

4.8 Identification

Sequence

A sequence will be identified by an eight character alphanumeric name.

Switch Points

MSU and SSU switch points will be identified by two methods: a standard OCS nomenclature and a spacecraft identification.

1. The standard OCS nomenclature will be of the form: MSUn.xxx or SSUn.xxx where n is the unit number and xxx is the switch point number.

2. The spacecraft test point identification is a seven character alpha-numeric name.

Data Cells

Associated with each level is a set of ten data cells. The purpose of these data cells is to facilitate the handling of variables by the elements. Each data cell is capable of handling a 32 binary bit value and a four character unit identification. These data cells are identified by D0 through D9.

Data Constants

Data constants that are used in element execution are specified by a type code and enclosed in parenthesis. The type codes are:

X - hexadecimal
I - decimal integer
F - floating point decimal

If a type code is not specified, the default is floating point decimal (The maximum floating point number held in the computer is 9.999×10^{62}).
Examples: X(A13F) the hexadecimal number A13F (maximum four hex digits)
I(5) the decimal integer 5
F(2.5) the floating point decimal 2.5

An alphanumeric character string uses the dollar sign ($) for delimiters.

Example: $AMPLIFIER A01 TEST$

5.0 Element Description

The following pages are detailed descriptions of the TOOL elements arranged in alphabetical order. The element examples are for the Off-Line (TLA) Translator. Each example is presented first in the full verbal form and second in the abbreviated form. Page 53 is a table listing the elements in alphabetical order. Page 54 is a table listing of the elements in numerical order.

Mnemonic: AG

Element Code: X'06'

Introduction: This element defines the terminal boundary of a DO loop.

Examples:

P	ELEMENT	MODIFIER(S)
	AGAIN	
	AG	

Discussion: See DO element discussion.

Mnemonic: BG

Element Code: X'01'

Introduction: The purpose of this element is to define the beginning of a sequence.

Modifiers:
a. Name; up to eight characters
b. Type of Sequence (Optional)
   1. Manually initiated (default, assumed if no type given)
   2. Clock initiated (C), requires spacecraft time: day:hour:minute:second : ms
   3. Monitor (M), requires:
      Cycle time (2 sec to 24 hours) hour:minute:second

c. Password; up to four characters (optional); the user must know this password to access this sequence.
d. Protection keys: setting the key inhibits the function (optional`
   1. Review (R)
   2. Modify (M)
   3. Call Execute (E)
   4. Keyboard Execute (K)
   5. Delete (D)
e. Examples:

P	ELEMENT	MODIFIER(S)
	BEGIN	TEST01

This example uses all default options; it has public access and no protection.

P	ELEMENT	MODIFIER(S)
	BG	TIMETEST,C,4:6:0:0
	BEGIN	TIMETEST,CLOCK,4:6:0:0

These two examples show a clock initiated test to be run the fourth day on the sixth hour.

P	ELEMENT	MODIFIER(S)
	BEGIN	MONTEST,M,0:10:30,KASK,DELETE
	BG	MONTEST,MONITOR,0:10:30

These two examples show a monitor initiated test to be run every 10½ minutes; the first example uses a password "KASK" and is protected against deletion.

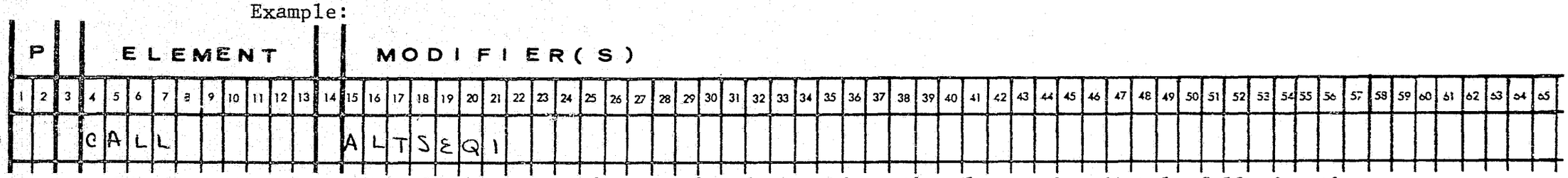
Mnemonic: CALL

Element Code: X'03'

Introduction: The purpose of this element is to call another sequence.

Modifier: Sequence name

Example:



Discussion: After the called sequence has completed execution, the element immediately following the CALL is executed. Four levels of calls may be used, that is, a CALL may call a sequence which contains a CALL to another sequence, which contains . . . and so on to four calls depth.

Element Code: X'OD'

Introduction: The purpose of this element is to set the various condition flags dependent upon the contents of a data cell. (The "EVALUATE" element also sets condition flags)
NOTE: If the "CHECK" element is to be used to check values supplied by a "MEASURE", keep in mind that the measured values are in the proper order of magnitude to have the units of Hz, microseconds, volts or milliamps associated with them. If the "Check" is against some other order of magnitude the value read by "MEASURE" must be adjusted using an "EVALUATE" element. (It is suggested that the new units associated with the data be entered at that time using the "Units of results" option in "EVALUATE". Example, it is desired to check a voltage signal against ± 50mv. First multiply (using "EVALUATE") the value read by "MEASURE" by 1000 to produce mv. Then check against ± 50.

Types: 1. Limits (L)

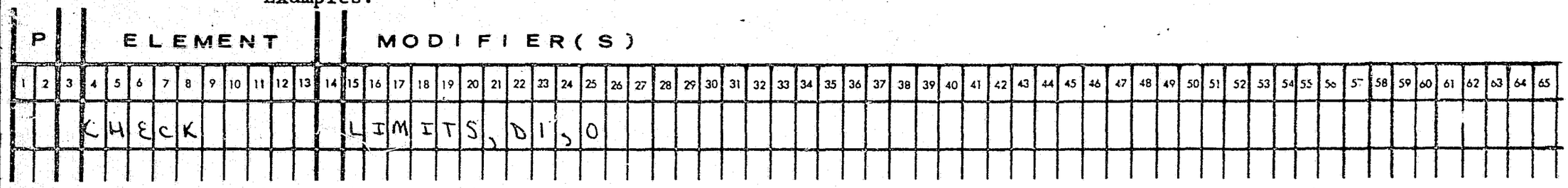
The contents of a data cell are compared algebraically with the specified limits. As a result of the compare, the following condition flags are set. These condition flags are to be used by the "IF" element. The check values may also be displayed (see "DISPLAY" element).

a. Over Limits (OV)
b. Under Limits (UN)
c. In Limits (IN) (equal to limits is considered in limits)
d. Out of Limits (OT)

Modifiers:

a. Upper limit value or data cell
b. Lower limit value or data cell
c. Data cell containing value to be checked: D0 through D9 (default is D0)

Examples:



In this example the contents of cell zero are compared to a high limit in data cell D1 and a low limit of zero.

P	ELEMENT	MODIFIER(S)
	CK	4,D3,1.01,D4

In this example cell D4 is compared to a high limit in D3 and a low limit of 1.01.

2. Logical (M)

The contents of a data cell are exclusively or'd with the specified mask to detect a match. As a result the following condition flags are set:

a. Equal (EQ); all resulting bits are zero
b. Not Equal (NE); not all bits are zero

Modifiers:

a. Data cell; D0 through D9 (default is D0)
b. Mask; up to 16 binary bits, right justified, or a data cell. (Left bits always zero)

Examples:

P	ELEMENT	MODIFIER(S)
	CK	LØGICAL,D1,X(104)

In this example the contents of cell D1 are exclusive or'd to the binary
0000 0000 0000 0000 0000 0001 0000 0100

P	ELEMENT	MODIFIER(S)
	CK	M,D2

In this example cell zero is Exclusive or'd to the binary mask in cell D2.

3. Contact (C)

The following condition flags are set by the contents of a data cell.

a. Open (OP); data cell equals zero
b. Closed (CL); data cell not equal to zero

Modifiers:

a. Data cell; D0 through D9 (default is D0)

Example:

P	ELEMENT	MODIFIER(S)
	CHECK	C,D3

Mnemonic: CL

Element Code: X'14'

Introduction: The purpose of this element is to clear a functional unit. This element deallocates the unit specified. "CLEAR ALL" deallocates all resources used by the level of concurrency executing the clear element.

Modifiers:
1. SSUn; where n is the unit number
2. MSUn; where n is the unit number
3. ALL (SGU, SSU, MSU, MU, Interrupt, ADU, and any other level started by this sequence)

Example:

P	ELEMENT	MODIFIER(S)
	CLEAR	SSU1
	CL	ALL

CONNECT

Mnemonic: CN

Element Code: X'07'

Introduction: The purpose of this element is to energize SSU switches. It allocates the appropriate SSU and SGU. WARNING, if the SGU has not been commanded off prior to issuing a "CONNECT" the selected point will receive whatever stimulus the SGU is producing.

Modifiers: SSU Unit and switch point numbers n.xxx or spacecraft I. D.

Examples:

P	ELEMENT	MODIFIER(S)
	CONNECT	SSU1.003
	CN	SSU2.003

These two examples connect the third switch point in the first and second switch units.

P	ELEMENT	MODIFIER(S)
	C2	CRAFTID

This example connects the switch point with the symbolic name of CRAFTID.

P	ELEMENT	MODIFIER(S)
	CN	SSU1.003

This example connects points 3 in switching unit 1.

Discussion: In an SSU only one non-latching switch can be connected at any one time, but any number of discrete switches can be connected. A non-latching switch will open upon the next SSU command, a latching switch will stay closed until it is disconnected or they are all cleared.
When a SSU switch is specified, a 5 ms delay in element execution will occur after the switch has been selected.

DELAY

Mnemonic: DL

Element Code: X'0F'

Introduction: The purpose of this element is to delay element execution for a specified period of time.

Modifier: Time Value: 5 ms up to 99 minutes; min: sec: ms

Examples:

P	ELEMENT	MODIFIER(S)
	DELAY	0:0:0:5
	DL	0:0:5:250

The first example delays execution 5 milliseconds. The second example delays execution 5¼ seconds.

Discussion: This element will insure that the minimum delay will be as specified; however, due to the concurrency of testing, the delay may exceed the given value. The level issuing the delay element will suspend operation while the delay is taking place.

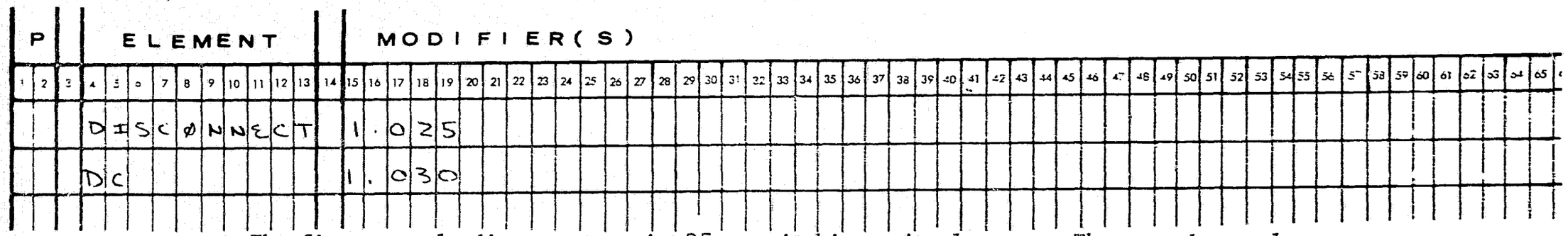
Mnemonic: DC

Element Code: X'08'

Introduction: The purpose of this element is to de-energize SSU switches but not to deallocate them.

Modifiers: SSU number and switch point on .xxx or spacecraft I.D. (if no switch point is specified, non-latching switch is assumed)

Examples:



The first example disconnects point 25 on switching unit 1. The second example disconnects point 30 on unit 1.

Discussion: A 5 ms delay in element execution will occur after all switches have been de-energized.

DISPLAY

Mnemonic: DP

Element Code: X'13'

Introduction: The purpose of this element is to give the sequence writer the capability to display information on the Alphanumeric Display Unit (ADU) or to select a Microdata Assembly (MA) frame. The writer has one of six lines of 21 characters each available on the ADU plasma in which to display his message.

The execution of the sequence will stop after the display is presented. If data is to be entered from the keyboard due to instructions given in the display, the completion of entering data (i.e., depression of the "Enter Data" button) will continue execution; otherwise, the "Run Continuous" or "Run Segment" button (depending upon which type of execution is being run) must be pushed to continue execution. If several display lines are to be presented at once on the ADU or if the MA is to be selected and lines are to be displayed on the ADU at the same time, the "&" (Ampersand) modifier must be used. This modifier will allow another display element to present its data without stopping between displays (the first "Display" element in a string always clears displays before presenting its information). If the "&" option is to be used and any "display" element in the string has the AMD option, all elements must have the AMD option.

When input is specified it must be entered in one of the forms shown on Page 6. This element will allocate the display resource (MA and ADU); if the resource is not available, it will wait until it can allocate the displays.

Types: 1. Message (M), this type allows the writer to present one line of information on the ADU or select an MA frame. If several lines are to be presented, the "&" modifier must be present to display all the lines at once.

Modifiers:

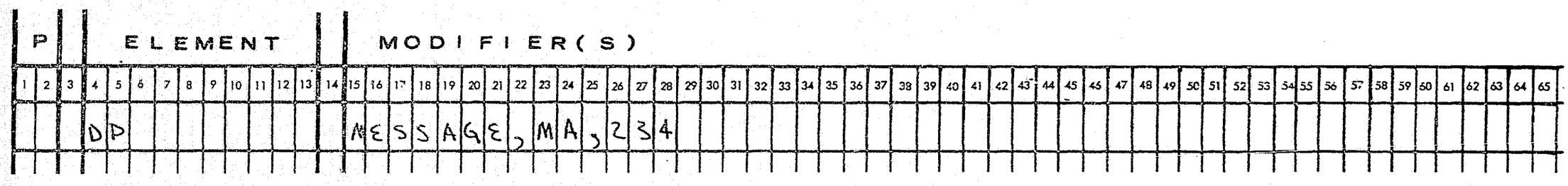
a. All Mode Display (AMD), optional. If this option is used the display will always be presented. If this option is not used, the display will only appear when the sequence is running in segment mode.

b. Input (I), optional (if the input option is used the "&" option may not be used)
   1) Data Cell; D0 through D9 (D0 is default case)
   2) Units, if required for identification of the type of units the Data Cell contains.
   If the message requests the operator to enter information through the keyboard, this "input" information tells where the keyboard input should be placed.
c. Device
   1) ADU, requires a line number 1 through 6 (default is line 1), or
   2) MA, requires a frame number, four octal digits.
d. Message; up to 20 characters inclosed by $ signs (valid only for ADU device).
e. "&" (optional); if the last character in the "Display" element is an & sign, the "Display" element immediately following will be displayed on the screen at the same time as the original Display(s). If the next element is not a "Display" element, the & will be ignored.

Examples:

P	ELEMENT	MODIFIER(S)
	DISPLAY	MESSAGE,AMD,ADU,1,$READ METER A1,ENTER$,&
	DP	M,AMD,I,D2,VOLTS,ADU,2,$VALUE IN VOLTS$

In this example the first display element writes a message in line five and links to the second element. The second writes the rest of the message on line two and expects a value to be entered which will be placed in Data Cell D2. The sequence will stop and wait for the input.



In this example the element selects an MA frame if the segment is being run in segment mode; if the segment is being run in the continuous mode, this element is not executed.

2. Parameter (P), this type will display on the ADU a floating point value from a data cell (and units, if desired).

   Modifiers:

   a. All Mode Display (AMD), Optional
   b. ADU line number from 1 to 6 (default is line 1)
   c. Data Cell, D0 through D9 (default is D0)
   d. Units, up to four characters, optional
   e. & (optional), causes link to next display before stopping

3. Results (R), this type will display on the ADU the results of the last "Check" element. All values will be displayed (ex.; if the last check was a check limit, the high, test and low values will be displayed on three lines).

   Modifiers:

   a. All Mode Display (AMD), optional
   b. ADU line number 1 through 6. The element may display more than one line; this will be the first line used. (Default is line 1.)
   c. & (optional) causes link to next display before stopping.

Examples:

P	ELEMENT	MODIFIER(S)
	DISPLAY	PARAMETER,AMD,D2,AMPS,&
	DP	R,AMD,2

The first line will display on line 1 the contents of Data Cell D2 and the word "AMPS." It will link to the second element which will display the last check values on line two down.

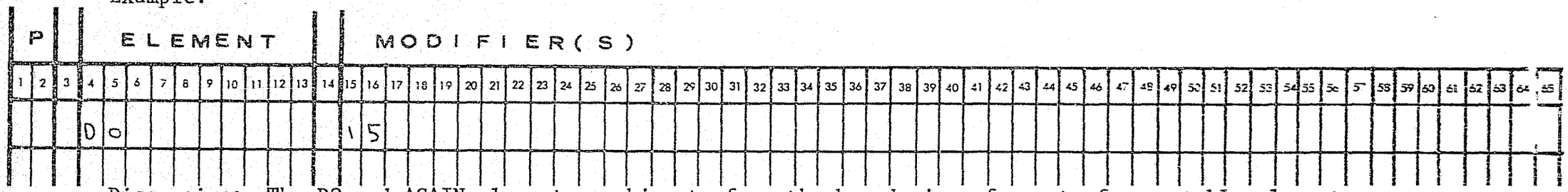
Mnemonic: DO

Element Code: X'05'

Introduction: The purpose of this element is to define the start of a repeatable set of elements and the repetition count.

Modifier: Repetition count: 0 to 255

Example:



Discussion: The DO and AGAIN elements combine to form the boundaries of a set of repeatable elements. This set of elements is called a "do loop." Associated with this do loop is a count which determines the maximum number of repetitions. When the DO element is executed, a counter is set up. Then the elements between the DO and AGAIN are executed. The AGAIN element decrements the counter by one. If the counter is greater than zero, a branch back to the element which immediately follows the DO is executed. If the counter is equal to zero, the element immediately following the AGAIN is executed.

No branching is allowed within a do loop. Hence, neither the CALL nor prefixed elements are allowed. Nesting of do loops is not allowed.

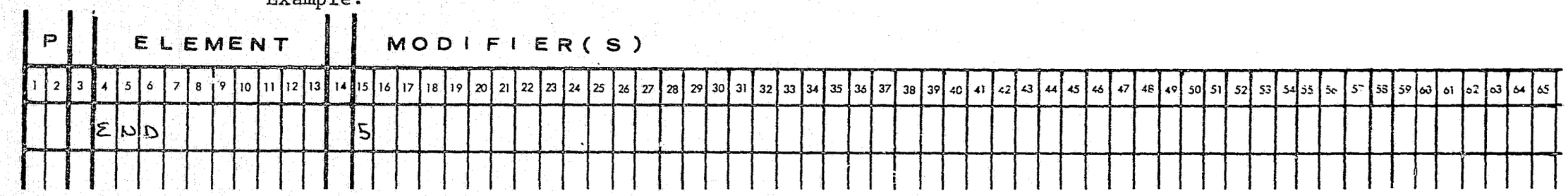
Mnemonic: END

Element Code: X'02'

Introduction: The purpose of this element is to define the end of a test sequence. It is not an executable element but simply a delimitor. If the "End" element is the end of the highest level sequence, it will deallocate all resources presently allocated to the level of concurrency through a "CLEAR ALL." Physically as well as logically the last element of every sequence must be an "END" element.

Modifier: One alphanumeric character label (optional) if no character is given "A" (normal end)

Example:



Discussion: In a monitor test the alphanumeric label is used to determine the monitor status. For initialization purposes the alphabetic "A" is considered the normal, or error free, end. Each time the monitor sequence is completed the modifier of the End element is compared to the modifier of the End element reached the previous time the monitor ran. If the modifiers are the same (ie, the same End element was reached both times) nothing happens; however, if they are not the same (that is, a different End element was encountered on the subsequent running) the Monitor Status Change light for the appropriate monitor is turned on.

EVALUATE

Mnemonic: EV

Element Code: X'0B'

Introduction: The purpose of this element is to perform operations with the contents of data cells. The algebraic condition flags will be set according to the resultant contents of the data cell and can be interrogated with the "IF" element. The general form of the Evaluate is: RESULTS = OPERAND1 OPERATOR OPERAND2. (The "Check" element also sets condition flags. The "Evaluate" will set conditions PLUS, MINUS or ZERO). A data cell may be loaded by equating to a fixed value as in the fifth example.

Modifiers:
1. Results; data cell D0 through D9
2. Operands
   a. Data cell D0 through D9
   b. Constant
3. Operators
   a. Addition +
   b. Subtraction -
   c. Multiplication X
   d. Division /
   e. And &
   f. Or '
   g. Exclusive Or "
4. Units for results (optional); up to four characters.

Examples:

P	ELEMENT	MODIFIER(S)
	EVALUATE	D3=D1+D2
	EV	D6=D5
	EVALUATE	D7=D6/F(100),OHMS
	EV	D8=D9&X(1F32),MASK
	EV	D9=2.06

Discussion: If no units are specified, the units associated with operand 1 are assigned to the results.

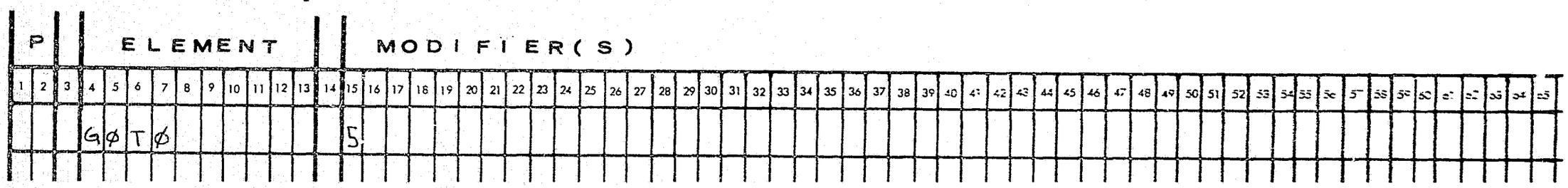
Mnemonic: GOTO

Element Code: X'04'

Introduction: This element unconditionally branches to a prefixed element within a test sequence.

Modifier: element identification number

Example:



Mnemonic: IF

Element Code: X'0E'

Introduction: This element is a conditional branch to a prefixed element within the sequence. If the specified flag is on, the branch is executed to the "Prefixed Element." If the flag is off, the next sequential element is executed.

Modifiers: 1. Flags
- a. Algebraic
  - 1) Over limits (OV)
  - 2) Under limits (UN)
  - 3) In limits (IN)
  - 4) Out of limits (OT)
  - 5) Plus (PL)
  - 6) Zero (ZE)
  - 7) Minus (MI)
- b. Logical
  - 1) Equal (EQ)
  - 2) Not Equal (NE)
- c. Contact Closure
  - 1) Open (OP)
  - 2) Closed (CL)

2. Alternate Element Prefix Number

Examples:

P	ELEMENT	MODIFIER(S)
	IF	EQUAL,16
	IF	E,16
	IF	OVER LIMITS,3
	IF	OV,3

Mnemonic: IT

Element Code: X'10'

Introduction: The purpose of this element is to provide a time dependent interrupt of element execution. Only one interrupt will be active per level of concurrency. If more than one interrupt is issued in a level all but the last will be ignored. When an interrupt timer times out the level of concurrency will be aborted. This interrupt timer is started with the intent of being turned off before it times out. If it does time out an unconditional branch to the end element is executed as soon as the present element has completed execution and the sequence is aborted.

Types: 1. Timer On (ON)
Modifier:
Time interval (T); 5 ms up to 79 minutes: min: sec: ms

Example:

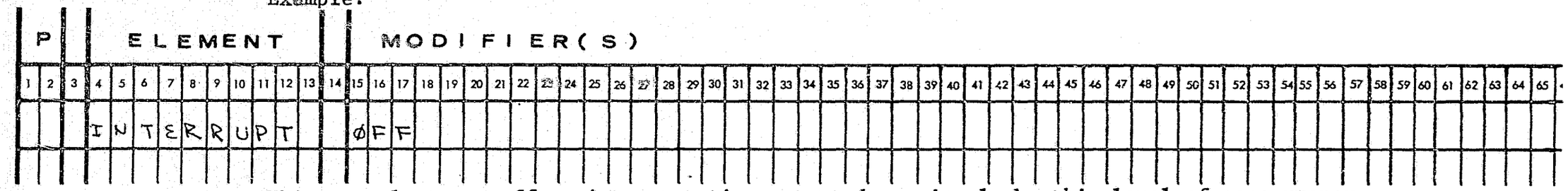
P	ELEMENT	MODIFIER(S)
	INTERRUPT	ØN,0:1:500
	IT	ØN,0:0:850

The first example will start a timer which will cause an exit after $1\frac{1}{2}$ seconds. The second will cause an exit after 850 ms.

2. Timer Off (OFF)

Modifiers: None

Example:



This example turns off an interrupt timer started previously by this level of concurrency.

MEASURE

Mnemonic: MU

Element Code: X'09'

Introduction: The purpose of this element is to control the operation of the Measurement Unit (MU) and Measurement Switching Unit (MSU). This element allocates the appropriate MU and MSU, performs the measurement and deallocates the units. The approximate expected value is used to range the hardware. If there is any question as to the magnitude of the approximate value expected, a lower than anticipated voltage should be used, a shorter than anticipated period should be used or a higher than anticipated frequency should be used. The system will automatically range to get an accurate value.
NOTE: If a voltage of less than 1.0V is specified a 5 ms delay occurs before measurements begin.

The values produced by the "MEASURE" element for analog signals are Hz, volts, milliseconds or milliamps. These units will be entered in the data cell in which the data is stored. Should the test writer desire to use the data in some other units the "EVALUATE" element should be used to change the order of magnitude. (It is suggested that the new units associated with the data be entered at this time using the "Units of results" option in "EVALUATE") Example, it is desired to hold the time measured in seconds rather than milliseconds. The data produced by "MEASURE" should be divided by 1000 using "EVALUATE" and the data will then reflect the proper order of magnitude for seconds.

Types: 1. DC Voltage (DC)
Modifiers:
a. MU and MSU identification; n.xxx or spacecraft I.D. name.
b. Data Cell for results; D0 through D9 (D0 is default case)
c. Voltage expected
Value; 10 mv to 40 v
d. Repetitive (optional), takes several samples (up to 63)
1) Time interval over which samples are to be taken; 500 us to 100 ms (by even 100 us steps); ms:us
2) Number of samples to be taken
3) Type of results
a) Average value (A)
b) Maximum value (M)

e. Examples:

P.	ELEMENT	MODIFIER(S)
	MEASURE	DC,1.003,D2,10MV
	MEASURE	DC,1.010,D2,1V,TIME 1:500,NUMBER 15,AVERAGE
	MU	DC,1.010,D2,1V,1:500,15,A

The first example measures DC voltage at MU 1 from MSU point 3 into Cell D2 with the expected value of 10 mv. The second and third example measure DC at MU 1, point 10 into Cell D2 with the expected value of near 1 volt. The measurements will be taken over a period of 1½ ms and the average of 15 samples will be recorded.

2. AC Voltage (AC)

Modifiers:

a. MU and MSU; n.xxx or spacecraft I.D. name
b. Data Cell; D0 through D9 (default is D0)
c. Peak Voltage expected
   Value; 50 mv to 40 v

d. Frequency expected
   Value; 0.1 Hz to 100KHz

NOTE: If the frequency is greater than 10 KHz, the maximum amplitude is limited to 10 volts peak. If the frequency is less than 10Hz for a peak measurement the repetitive mode is suggested to assure accurate results.

e. Type measurement wanted
   1) Peak (P)
   2) RMS (R)

f. Examples:

P	ELEMENT	MODIFIER(S)
	MEASURE	AC,1.002,100mV,100HZ,RMS,NUMBER 20,MAXIMUM
	MU	AC,1.002,100mV,100HZ,R,20,M . .

These examples both measure the RMS value of the AC voltage on point 2 of MU number 1. The measurement is taken 20 times and the maximum reading is recorded in Cell DO.

3. Frequency (F)

Modifiers:

a. MU and MSU; n.xxx or spacecraft I.D. name
b. Data Cell; DO through D9 (default is DO)
c. Peak voltage expected
   Value; 50 mv to 40 v
   NOTE: If the frequency is greater than 10KHz, the maximum amplitude is limited to 10 volts/peak.
d. Point of Measurement
   1) -10%
   2) -50%
   3) -25%
   4) Zero (Z)
   5) +10%
   6) +50%
   7) +25%
   8) D.C. Offset (averages signal to find zero point, valid only for repetitive signals)
e. Frequency
   Value; 0.1 Hz to 100 KHz

f. Examples:

P	ELEMENT	MODIFIER(S)
	MEASURE	FREQUENCY,1.020,D1,.707V,Z,5KHZ
	MU	F,1.020,.707V,6,5KHZ,10

These examples both measure frequency of point 20 on MU 1. The second example will take the average of 10 samples of a signal riding 50% above zero level.

4. Phase (P) , phase difference is expressed in time (not degrees or radians)

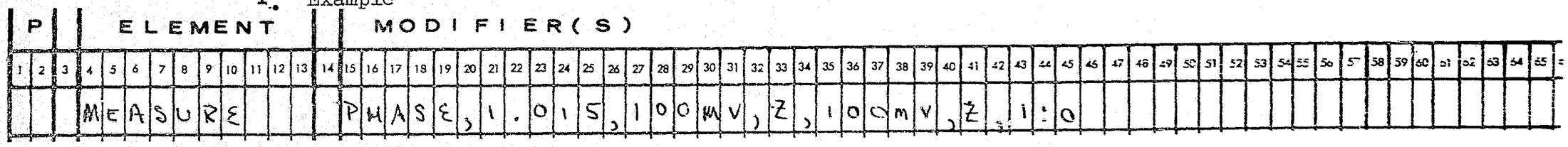
Modifiers:

a. MU and MSU; n.xxx or spacecraft I.D. name.
b. Data Cell; D0 through D9 (default is D0)
c. Peak voltage, first signal.
   Value; 50 mv to 40 v

NOTE: If the time interval is less than 100 us, the maximum amplitude is limited to 10 volts/peak.

d. Peak voltage, second signal.
   Value; 50 mv to 40 v
e. Interval expected
   Value; 1 μs to 1000 sec; sec: ms:us

f. Example



This example reads into Cell D0 the "phase" of point 15 which is a 100 mv signal with an expected phase difference of 1 ms. Note that point 15 must be one of the dual contact switch points so that there are two signals to compare for the phase difference measurement.

5. Interval (I)
Modifiers:
   a. MU and MSU; n.xxx or spacecraft I.D. name
   b. Data Cell; D0 through D9 (default is D0)
   c. Peak voltage expected on Pulse Train A, Value; 50 mv to 40 v
   d. Peak voltage expected on Pulse Train B, if necessary, Value; 50 mv to 40 v
   e. Cycle time expected, Value; 1 us to 1000 sec; sec: ms: us
   f. Input Type
      1) Single (S)
      2) Dual (D)
   g. Start measurement slope
      1) Positive (+)
      2) Negative (-)
   h. End measurement slope
      1) Positive (+)
      2) Negative (-)
      NOTE: If dual pulse train is selected, the start slope is from bus A and the end slope is from bus B.
   i. Point of Measurement of Pulse train A
      1) -10% 5) +10%
      2) -50% 6) +50%
      3) -25% 7) +25%
      4) Zero (Z) 8) D.C. Offset (averages signal to find zero point, valid only for repetitive signals)
   j. Point of Measurement of Pulse Train B, if necessary
      1) -10% 5) +10%
      2) -50% 6) +50%
      3) -25% 7) +25%
      4) Zero (Z) 8) D.C. Offset (averages signal to find zero point, valid only for repetitive signals)

e. Examples:

P	ELEMENT	MODIFIER(S)
	MEASURE	INTERVAL,1.022,D0,500MV,0:60:0,S,POSITIVE,NEGATIVE,Z
	MU	I,1.022,500MV,0:60:0,SINGLE,+,-,Z
	MU	I,1.022,500MV,0:60:0,S,+,+,Z

The first two examples read into Cell D0 the time that the signal on point 22 is in the positive state. The third example reads the length of the entire pulse.

6. Totalizing, Pulse Count (T), Note: The maximum count is 10,000
   Modifiers:
   a. MU and MSU; n.xxx
   b. Data Cell; D0 through D9 (default is D0)
   c. Peak voltage
      Value; 100 mv to 40 v

   d. Time interval over which count is to take place (T); 1 ms to 100 min, min: sec: ms
   e. Examples:

P	ELEMENT	MODIFIER(S)
	MEASURE	TOTAL,1.025,D3,3V,0:3:250

In this example the MU will count the pulses coming in on point 25 during a $3\frac{1}{4}$ second period.

7. Contact Closure (C) (A 1 result indicates contact closed, A 0 indicates contact open)
   Modifiers:
   a. MU and MSU; n.xxx
   b. Data Cell; D0 through D9 (default is D0)
   c. Examples:

P	ELEMENT	MODIFIER(S)
	MEASURE	CONTACT,1.006,D1
	MU	C,1.006

The second example places the contact mask into Cell D0.

8. Digital Data (DD)
   Modifiers:
   a. MU and MSU; n.xxx or spacecraft I.D. name
   b. Data Cell; D0 through D9 (default is D0)
   c. Types
      1) dual input (D)
      2) single input (S)

   d. Example:

P	ELEMENT	MODIFIER(S)
	MEASURE	DIGITAL,1.020,D9,DUAL
	MU	DD,1.020,D9,S

9. Voltage Verify (VV), used to verify that the Measurement Unit if functioning properly.
   Modifier:
   a. Data Cell for results.
   b. MU number.
   (The voltage read should be 6.3 volts ± 5%)

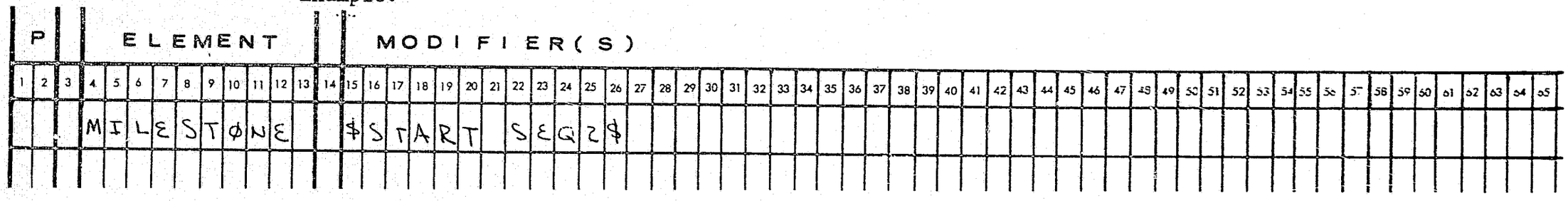
MILESTONE

Mnemonic: MI

Element Code: X'16'

Introduction: The purpose of this element is to enable the sequence writer to display a milestone message on the ADU on the status line associated with each level of concurrency. The message may be up to 10 characters in length and its purpose is to give the operator information on the status of a running sequence. (As the display resource is not required for execution of this element, the message is always displayed immediately when it is encountered.) The sequence writer should take into account human reaction time when setting up milestones as each message must remain long enough to be recognized by the operator.

Example:



Mnemonic: RD

Element Code: X'0C'

Introduction: To be specified

RECORD

Mnemonic: RC

Element Code: X'11'

To Be Specified

REPEAT FLAG

Mnemonic: RF

Element Code: X'15'

Introduction: This element defines a repeatable point in the element data file.

Modifiers: None

Example:

P	ELEMENT	MODIFIER(S)
	REPEATFLAG	
	RF	

Discussion: This element is used in conjunction with the REPEAT pushbutton.

START

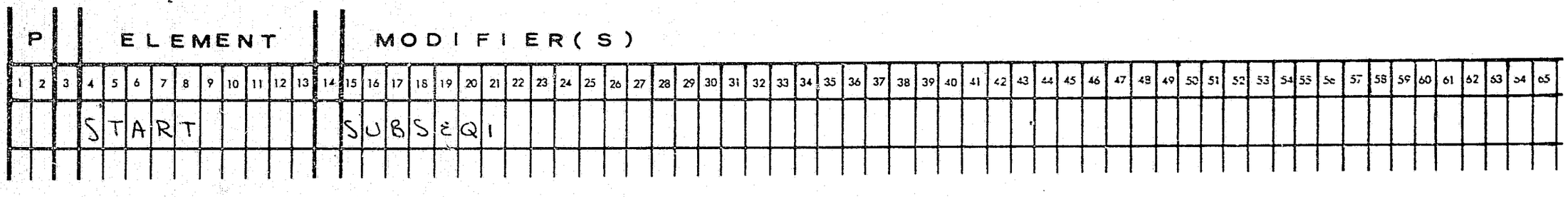
Mnemonic: STRT

Element Code: X'17'

Introduction: This element will start another sequence on another level. If no levels are available, the element will wait until one is available.

Modifier: The name of an existing sequence.

Example:



This example will start a sequence named "SUBSEQ1" on another level of concurrency.

STIMULATE

Mnemonic: SGU

Element Code: X'0A'

Introduction: The purpose of this element is to control the operation of the Stimulus Generator Unit. The "Connect" element controls the point to be stimulated. There are two techniques for using the "Stimulate" element. The first is to issue a "CONNECT" and then a "STIMULATE" using the "Automatic Stimulate" option. The stimulus will start immediately and if it is a continuous type signal will remain on until a "Stimulate Off" or "Stimulate Clear" is issued. The second technique to connect and use a "Stimulate" of types 4 thru 8 without the AS option to select the proper type of stimulation.

When the actual stimulation is desired, a"Stimulate On" should be issued and stimulation will occur with no delay. (It is acceptable to change either the "Stimulate" or "Connect" element at any time the SGU is in the "OFF" state). The digital option is an exception to the above in that it does not use a "CONNECT".

Types: 1. On (ON); this command begins, stimulation in the manner prescribed by a previous "Stimulate" element.
SGU Number
Example:

P	ELEMENT	MODIFIER(S)
	STIMULATE	ØN
	SGU	ØN

NOTE: If a "Stimulate ON" is issued before a "Stimulate" element which sets up the parameters is issued, an error results and no stimulation takes place.

2. Off (OFF); this command ends stimulation and returns all outputs to a null or zero condition.
SGU Number

Examples:

P	ELEMENT	MODIFIER(S)
	STIMULATE	ØFF
	SGU	ØFF

3. Clear (CL); this command will turn off all stimulus and deallocate the SGU and SSU. It should be used when the sequence has no further immediate needs for the stimulus resource.

Example:

P	ELEMENT	MODIFIER(S)
	SGU	CLEAR

Discussion: To allow for SGU setup, there is a 5 ms built-in delay after the element is executed. SGU Clear is an exception to this. In D/A conversions the delay is necessary only during the initial setup, that is, prior to the first stimulation

4. Waveform (WF)

Modifiers:

a. Automatic Stimulate (AS), optional. This is used in place of "Stimulate On" when desired.

b. Types
   1) Sine (SI)
   2) Square (SQ)
   3) Triangle (TR)

c. Frequency
   1) Constant; .1 Hz to 100 KHz
   2) Variable; Data cell D0 through D9

d. Amplitude
   1) 0v to 40v peak if frequency $\leq$ 10 KHz (for sine wave the value may be entered as RMS, maximum
      0v to 10v peak if frequency $>$10 KHz value in high range is 28.28V, in low range 7.07 volts)
   2) Variable; Data cell D0 through D9

e. SGU Number

f. Examples:

P	ELEMENT	MODIFIER(S)
	STIMULATE	WAVEFORM,SINE,500HZ,32V
	SGU	WF,AS,SI,500HZ,28.28VRMS

Both of these examples stimulate with 500 Hz, 32 volt peak sine wave, the second starts stimulation automatically. (using SGU #1)

5. DC (DC)

Modifiers:

a. Automatic Stimulate (AS), optional. This is used in place of "Stimulate On" when desired.

b. Types
   1) Voltage (V)
      a) Constant; 0 to $\pm$ 40v
      b) Variable; data cell D0 through D9

2) Current (C)
   a) Constant 0 to ± 250 ma
   b) Variable data cell D0 through D9

c. SGU Number

d. Examples:

P	ELEMENT	MODIFIER(S)
	STIMULATE	DC,VOLTAGE,4.03
	SGU	DC,CURRENT,D4
	SGU	DC,V,D2

6. Pulse (P)

Modifiers:

a. Automatic Stimulate (AS), optional. This is used in place of "Stimulate On" when desired.

b. Pulse description; ms;µs
   1) Pulse width (W)
   2) Pulse spacing (S) (Pulse width plus pulse spacing equals period)
   NOTE: The width and spacing must be within the same time interval ranges. These ranges are:
      a) 5 µs to 100 µs
      b) 100 µs to 1 ms
      c) 1 ms to 10 ms
      d) 10 ms to 100 ms
      e) 100 ms to 1 sec.

c. Pulse Count (C)
   1) 1 to 1000 pulses
   2) Continuous (CTS)

d. Amplitude
   1) Voltage (V)
      a) Constant
         (1) 0v to ± 40v if interval range ≥ 100 μs
         (2) 0v to ± 10v if interval range < 100 μs
      b) Variable; data cell D0 through D9
   or 2) Current (MA)
      a) Constant
         (1) 0ma to ± 250 ma if interrupt range ≥ 100 μs
         (2) 0ma to ± 133 ma if interrupt range < 100 μs
      b) Variable; data cell D0 through D9
e. SGU Number
f. Examples:

P	ELEMENT	MODIFIER(S)
	STIMULATE	PULSE,WIDTH 5:0,SPACE 7:0,COUNT 500,25 VOLTS
	SGU	P,5:0,7:0,500,25V

7. D/A Conversion (DA); this will output analog signals either as a single value or in a series of steps. The D/A conversion always starts immediately; therefore, if any corresponding measurement is to be taken this must be set up beforehand.

   Modifiers:
   a. Interval between changes in output (T); up to 100 ms in even 100 μs steps; ms:μs
   b. Count of total number of steps (C); max. 255
   c. Value at each step
   d. SGU Number

   Note: All of the steps must be in the same range. The ranges are: 0 to 1 volt, 1 to 10 volts, 10 to 40 volts.

e. Example:

P	ELEMENT	MODIFIER(S)
	STIMULATE	D/A CONVERSION,TIME 10:0,COUNT 5,STEP1 1V,STEP2 2V;
		STEP3 3V,STEP4 2V,STEP5 1V
	SGU	DA,10:0,5,1V,2V,3V,2V,1V

This example will send out a series of 5 steps going in order from 1 to 3 volts and back to 1 volt. Each voltage level will be held for 10 ms. (using SGU Number 1)

8. Digital (DG), this information is placed on the 12 digital data lines immediately and will stay there until it is changed. The "CONNECT" element is not used before this type of stimulate.
   a. Data
      1) Variable; D0 through D9, 12 least significant bits
      2) Constant; 12 binary bits (Enter as 3 Hex Digits)
      3) Off; to clear the digital data to zeros
   b. SSU Number
   c. Examples:

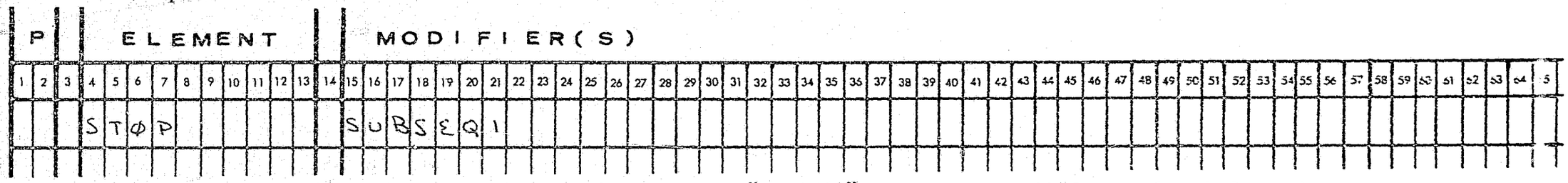
P	ELEMENT	MODIFIER(S)
	STIMULATE	DIGITAL,D1
	SGU	DG,X(20B)
	SGU	DG,OFF

Mnemonic: STOP

Element Code: X'18'

Introduction: This element will terminate a sequence which is running on a different level of concurrency. It will deallocate the level.

Example:



This example will stop the sequence "SUBSEQ1" running on a different level from the one issuing the stop element.

TRANSFER

Mnemonic: TF

Element Code: X'12'

To Be Specified

ALPHABETIC ELEMENT LIST

NAME	MNEM	CODE	DESCRIPTION
AGAIN	AG	06	Terminate DO Loop
BEGIN	BG	01	Beginning of Sequence
CALL	CALL	03	Call Sequence
CHECK	CK	OD	Check Contents of Data Cell
CLEAR	CL	14	Clear Unit
CONNECT	CN	07	Connect SSU switch
DELAY	DL	OF	Delay Element Execution
DISCONNECT	DC	08	Disconnect SSU Switch
DISPLAY	DP	13	ADU or MA Display
DO	DO	05	Start DO Loop
END	END	02	End of Sequence
EVALUATE	EV	OB	Evaluate Data
GOTO	GOTO	04	Unconditional Branch to "Prefixed Element"
IF	IF	OE	Conditional Branch to "Prefixed Element"
INTERRUPT	IT	10	Interrupt Element Execution
MEASURE	MU	09	MU Control
MILESTONE	MI	16	ADU Information Presentation
READ	RD	OC	Read Data S.C.T. & PCM
RECORD	RC	11	Record Data
REPEAT FLAG	RF	15	Repeatable Elements Flag
START	STRT	17	Starting a Sequence on Another Level
STIMULATE	SGU	OA	SGU Control
STOP	STOP	18	Stopping a Sequence on Another Level
TRANSFER	TF	12	Transmit Data

NUMERICAL ELEMENT LIST

CODE	MNEM	NAME	DESCRIPTION
01	BG	BEGIN	Beginning of Sequence
02	END	END	End of Sequence
03	CALL	CALL	Call Sequence
04	GOTO	GOTO	Unconditional Branch to "Prefixed Element)
05	DO	DO	Start DO Loop
06	AG	AGAIN	Terminate DO Loop
07	CN	CONNECT	Connect SSU Switch
08	DC	DISCONNECT	Disconnect SSU Switch
09	MU	MEASURE	MU Control
OA	SGU	STIMULATE	SGU Control
OB	EV	EVALUATE	Evaluate Data
OC	RD	READ	Read Data
OD	CK	CHECK	Check Contents of Data Cell
OE	IF	IF	Conditional Branch to "Prefixed Element"
OF	DL	DELAY	Delay Element Execution
10	IT	INTERRUPT	Interrupt Element Execution
11	RC	RECORD	Record Data
12	TF	TRANSFER	Transmit Data
13	DP	DISPLAY	ADU or MA Display
14	CL	CLEAR	Clear Unit
15	RF	REPEAT FLAG	Repeatable Element Flag
16	MI	MILESTONE	ADU Information Presentation
17	STRT	START	Starting a Sequence on Another Level
18	STOP	STOP	Stopping a Sequence on Another Level

6.0 System Structure

The Test Oriented Onboard Language (TOOL) System consists of three subsystems: Interrupt, On-Line Translator, and Test Execution. Figure 3 is a block diagram of this structure. The Interrupt Subsystem is triggered by hardware interrupts of either the external or priority class. Both of these classes of interrupts are handled by the Executive Interrupt Subsystem (P-ECS-000-00). Thus, the TOOL Interrupt Subsystem is executed as a part of the Executive Interrupt Subsystem. The other two subsystems are executed as scheduled systems under control of the Executive System Scheduling Subsystem (P-ECS-000-00). The On-Line Translator Subsystem is of higher priority than the Test Execution Subsystem in the scheduler.

6.1 Subsystem Structure

Each subsystem is separated into a set of functions. Each function is implemented through a set of processor, routine, and subroutine programs.

6.1.1 Interrupt Subsystem, P-TOL-100-00

The purpose of this subsystem is to decode and process interrupts from the Control and Display Unit (CADU), Measurement Unit (MU), and Stimulus Generator Unit (SGU).

6.1.1.1 CADU Interrupt Function

This function decodes and processes the interrupts generated by the CADU control pushbuttons. Associated with each pushbutton is a PB Processor program. In general, each PB Processor performs a processing operating such as updating system status; and then activates one of the TOOL scheduled subsystems.

6.1.1.2 MU Interrupt Function

The purpose of this function is to process the MU measurement complete interrupt.

6.1.1.3 SGU Interrupt Function

The purpose of this function is to process SGU interrupts.

6.1.2 On-Line Translator Subsystem, P-TOL-200-00

This subsystem is a single entry in the Executive Scheduler, and of higher priority than the Test Execution Subsystem. This priority arrangement allows the translation and test execution to be performed concurrently. Since the On-Line Translator does allocate the display resource, a test sequence will have to wait if a Display element is executed.

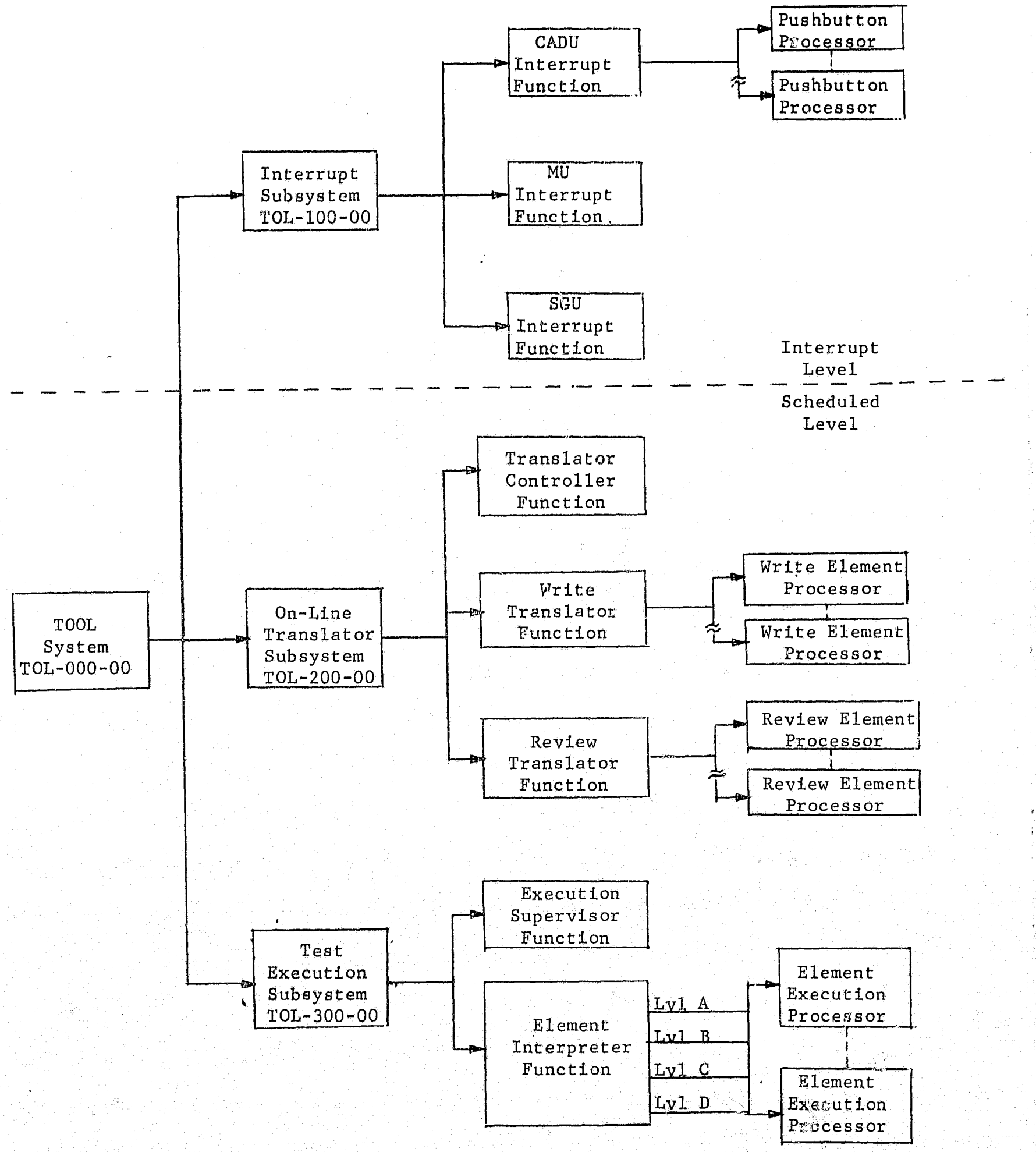


Figure 3
TOOL System Structure Block Diagram

6.1.2.1 Translator Controller Function

This function controls the operations of test translation and modification. In general, CADU keyboard inputs are checked for proper mode operation, and then routed to either the Write or Review processors for further validation checks.

6.1.2.2 Write Translator Function

This function translates CADU keyboard inputs into data for the test sequence data list. This translation is implemented through a set of Write Element Processors. Each processor first checks the input for validity, translates the data, and then presents the next sequential cue.

6.1.2.3 Review Translator Function

This function translates data from the test sequence data list into a readable display for operator information. This function is implemented through a set of Review Element Processors. Each processor translates an element contained in the test sequence data list into a display message, and then presents a cue describing the allowed options.

6.1.3 Test Execution Subsystem, P-TOL-300-00

This subsystem consists of an Execution Supervisor and a set of Element Execution Processors. This entire subsystem is a single entry in the Executive Scheduler.

6.1.3.1 Execution Supervisor Function

The purpose of the Supervisor is to supervise test execution by cycling through the four levels of test execution. This cycling consists of checking the level status, branching to the proper element processor, and maintaining the priority counter. When all four levels are idle, the Supervisor deactivates the Test Execution system to allow lower priority tasks to be executed.

6.1.3.2 Element Interpreter Function

This function consists of a set of processors which perform the operation specified by the test language element. In general, these processors allocate the required resource, perform the operation, and then deallocate the resource. If a resource is not available, the element processors set the wait status in the associated level. The Supervisor then proceeds to the next level. On the next level cycle an attempt is again made to obtain the required resource. If still not available control is again relinquished. When the resource becomes available, the element processor resets the wait status and continues execution.

7.0 OPERATION ACTION

Not Applicable

8.0 SYSTEM GROUND RULES

Not Applicable