

DECLASSIFIED  
E.O. 13526  
Authority  
By LS NAWA, Date 11-18

INITIAL HARVEST CONTROL PROGRAM

Parameter Specification, Insertion, & Checking

June 1960

CONFIDENTIAL

DECLASSIFIED  
 E.O. 13526  
 Authority  
 By L.S. HAWA, Date 12-15

13 Jun 60  
 Initial HCP

PARAMETER SPECIFICATION, INSERTION AND CHECKING

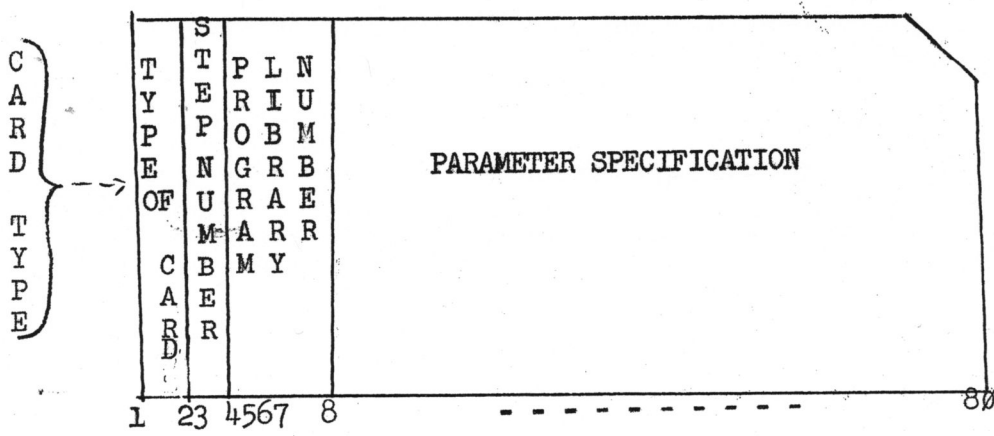
I. PARAMETER SPECIFICATION

A. Request Language

1. Program parameters will be specified on Parameter Cards which accompany the request for each problem program. These cards have the following format:

- (CARD TYPE)
- Col. 1 : Type of Card ("X", indicating a Parameter Card)
  - Col. 2-3 : Step number of this program within the job containing the program request.
  - Col. 4-7 : Program Library Number for this program
  - Col. 8-80 : Parameter Specification

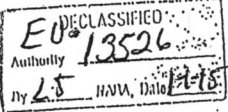
Parameters are specified by symbolic name of parameter and absolute or symbolic parameter itself.



2. The symbolic parameter name or symbolic parameter itself can be any combination of no more than eight alpha-numeric characters of which the first is alphabetic so long as provision for the symbol is made in the Symbol Translator Table of the problem program into which the parameter is to be inserted.

Absolute parameters are allowed which conform to the parameter description that is part of the corresponding symbolic parameter name entry in the Symbol Translator Table.

3. There are two categories of parameters which can be specified in the parameter description: numeric data which is to be converted to binary numbers and character data which is to be converted card-column by card-column through a standard character conversion.



Numeric data may be punched in the Parameter Cards in any of three different forms: binary, octal, or decimal. The numbers are converted to binary numbers of a specified (by the parameter description) field length which cannot exceed 64 bits. If the numbers punched on the Parameter cards are preceded explicitly by a plus or minus sign, a one-bit sign byte becomes part of the resulting binary number; otherwise the resulting binary number is considered to be unsigned. Both integer and mixed numbers (integer and fraction) can be represented; if the octal or decimal form of numeric data is used and if the numbers have a fraction/part then the fraction length in bits of the converted fraction must be specified in the parameter description.

Character data will be converted card-column by card column from the original coding (initially tape BCD) to a fixed eight-bit HCS (HARVEST Character Set) coding.

4. More than one numeric or symbolic parameter can accompany a symbolic parameter name as long as they are each of the same type (binary, octal, decimal or symbolic) and have the same field length. These parameters are separated by commas on the punched cards. The programmer can, through the parameter description, specify that certain exact or maximum number of numbers must or can accompany a given symbolic parameter name or can indicate that as many as are punched on the card(s) will be accepted by the program.

The programmer can, through the parameter description, specify an exact or maximum number of columns of character data which must or can be accepted or indicate that as many as are specified on the parameter card (s) can be accepted. The program can specify the number of columns per card which are to be converted.

5. When there is space and it is desired, more than one parameter (symbolic parameter name plus parameter(s)) can be punched on one card.

When more than one card is required to specify a symbolic parameter name and its accompanying parameter(s) then the second card has the symbolic parameter name repeated and then a +1 appended to it, the third card a +2, etc.

A single numeric or symbolic parameter cannot have a part of one number or symbol on one card and the remainder on the next.

The programmer can specify the number of columns per card which are to be converted; if this is not the actual number of columns to be converted which are punched on a given card, the actual number of columns is enclosed in parens following the symbolic parameter name and appended +n if any.

DECLASSIFIED  
 E.O. 13526  
 Authority  
 By L.S. NAVA, Date 11-18

6. All blanks in the parameter specification field of a Parameter Card are ignored except for those in columns to be converted in character data.

The symbolic parameter name and its associated parameter(s) are separated by an equal (=) character.

When a symbolic parameter is used, the letter S enclosed in parens is punched on the Parameter Card following the symbolic parameter name.

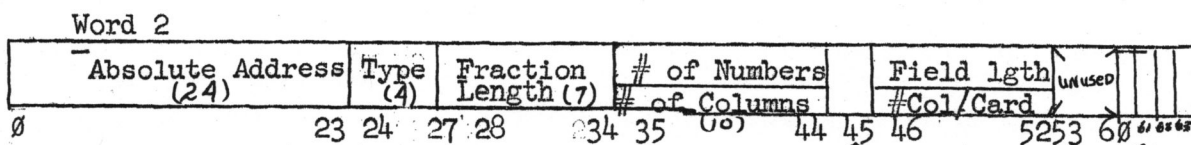
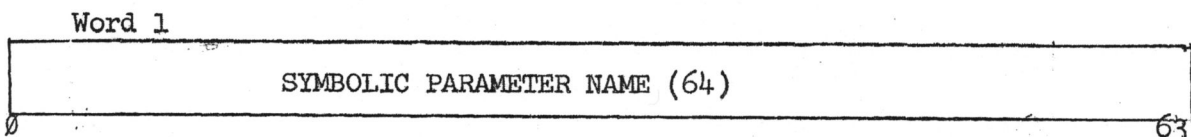
7. Symbolic parameters can indicate, through the Symbol Translator Table, any memory area containing the desired parameter, or can indicate a parameter which is a part of the Symbol Translator Table itself.

B. Symbol Translator Table

1. The Symbol Translator Table for a particular program is constructed by HASTII from PLE (Parameter Location Entry), SPE (Symbolic parameter Entry), and SPEI (Symbolic Parameter Entry immediate pseudo-op statements). Each statement causes a two-word\*entry to be made in the table. The table will be placed near the end of programmer available memory immediately preceding the Symbol File Table.

\* may be three, if names can be up to 16 chars. in HASTI.

2. A parameter location entry (PLE) in the Symbol Translator Table has the following format:



↑ Exact/Max.

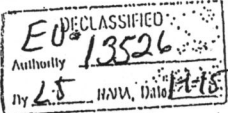
63: Input  
 62: Error  
 61: Must

a. Symbolic Parameter Name - the HCS - character coded configuration of the name; for a name of less than eight characters, the word contains blanks (0000 0000<sub>2</sub>) in the rightmost character positions.

b. Absolute Address - the starting bit address of memory area into which the parameter(s) is (are) to be inserted.

c. Type - indicates whether the parameter is to be a binary, decimal, or octal number to be converted to a binary number or a group of characters to be converted column-by-column of the Parameter Card.

d. Fraction Length - indicates the number of bits to be used as the fraction part of the resulting binary fraction when an octal or decimal number with a fraction part is specified. The maximum acceptable value of this field is 64<sub>10</sub>.



e. (1) # of Numbers - When a binary, octal, or decimal numeric parameter is specified, this indicates the exact or maximum number of numbers which must or can accompany the symbolic parameter name; a zero value for this field indicates that as many numbers as are punched on the card (s) will be accepted by the program.

(2) # of Columns - When a character parameter is specified, this indicates the exact or maximum number of columns of character data which must or can be accepted by the program; a zero value for this field indicates that as many as are specified on the parameter card (s) will be accepted by the program. The maximum value for this field is  $1023_{10}$ .

f. EX/Max - When the # of Numbers or # of Columns field is non-zero, this bit indicates whether the number is the exact number of numbers or characters which the program must have or whether the number is the maximum number of numbers or characters which the program can accept. When the # of numbers or # of characters field is zero this bit is disregarded.

g. (1) Field Length - When a binary, octal, or decimal numeric parameter is specified, this indicates the field length in bits of the resulting binary number; the maximum acceptable value for this field is  $64_{10}$ .

(2) # of Cols./Card - When a character parameter is specified, this field indicates the number of to-be-converted character columns to be placed on each card containing this parameter, unless the number of columns are specified on the Parameter Card itself following the Symbolic Parameter Name; the maximum acceptable value for this field is  $73_{10}$ .

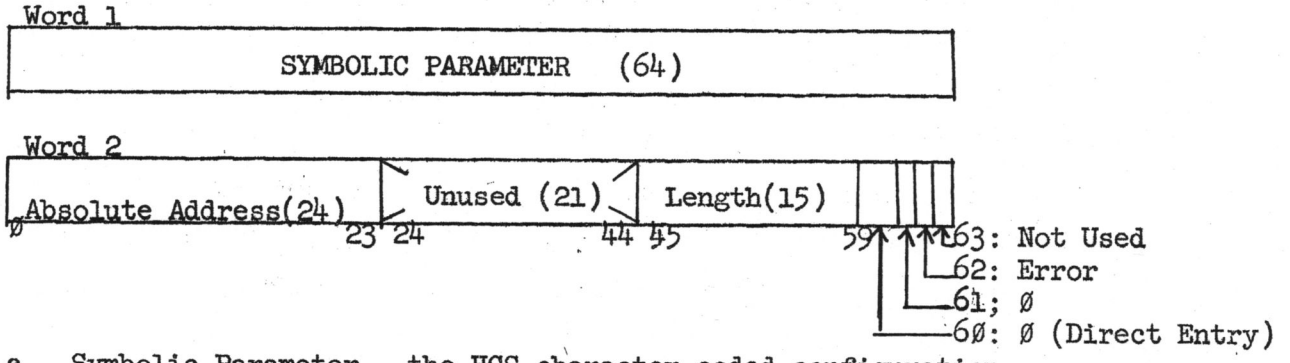
h. MUST? - Indicates whether this parameter has to be inserted in the program for it to function properly (i.e., that it must be specified on parameter cards accompanying the request for the program).

i. ERROR - The program can indicate that this parameter was the one which the program checked in its Parameter Check Routine and found unacceptable.

j. INPUT? - The Job File Generator Program sets this bit to one if it inserts this particular parameter. It is used to let the system know quickly if any required parameters have not been specified by the Parameter Cards.

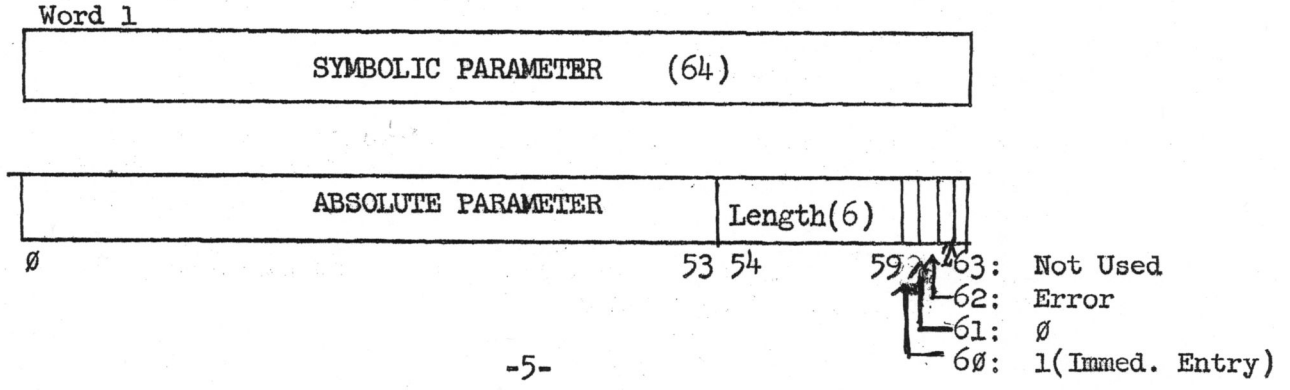
DECLASSIFIED  
 E.O. 13526  
 Authority  
 By 15 NAVA, Date 11-15

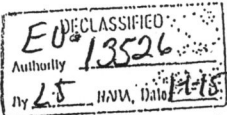
3. A symbolic parameter entry (SPE) in the Symbol Translator Table has the following format:



- a. Symbolic Parameter - the HCS-character coded configuration of the symbolic representation of the parameter; for a name of less than eight characters, the word contains blanks (0000 0000<sub>2</sub>) in the rightmost character positions.
- b. Absolute Address - the starting bit address of the memory area from which the parameter is obtained.
- c. Length - the length in words and bits of the parameter; the maximum length is 511.63<sub>10</sub> words.
- d. Direct/Immediate - a zero bit in this field indicates that this table entry in direct rather than immediate.
- e. Not Available - this bit position corresponds to the "MUST?" field in a PLE entry in the table; it is always zero.
- f. Error - the program can indicate that this parameter is associated with its reason for rejecting the parameters in its Parameter Check Routine.
- g. INPUT? - this bit is set to one by the Job File Generator Program when this table entry is referenced by a symbolic parameter on a Parameter Card.

4. A symbolic parameter entry immediate (SPEI) in the Symbol Translator Table has the following format:





- a. Symbolic Parameter - Same as for SPE
- b. Absolute Parameter - the parameter itself, starting with bit position zero and extending for the specified number of bits.
- c. Length - the length in bits of the immediate parameter; the length can be from 1 to  $54_{10}$  bits.
- d. Direct/Immediate - a one bit in this field indicates that this table entry is immediate rather than direct.
- e. Not available, Error, INPUT? - same as for SPE.

### C. Pseudo-Op Formats

1. The format for the HAP II pseudo-op PLE (Parameter Location Entry) is as follows:

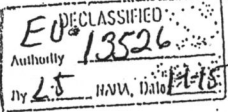
SYMBOLIC PARAMETER NAME PLE, SYMBOLIC LOCATION, TYPE(FRACTION LENGTH),  
# OF NUMBERS OR # OF COLUMNS (EXACT/MAXIMUM,  
FIELD LENGTH OR # OF COLS/CARD, MUST)

- a. Symbolic Parameter Name - the alpha-numeric configuration which is to be used on the Parameter Card(s) to identify this parameter; it is also the name which the programmer will use when referring to this entry in the Symbol Translator Table (e.g. in the program's Parameter Check Routine to set the "ERROR"(bit).
- b. PLE - the pseudo-op
- c. Symbolic Location - indicates the starting bit address of the memory area into which the parameter(s) is (are) to be inserted.
- d. The <sup>way the</sup> remaining fields are to be described in assembly language has not been determined.

2. The format for the HAP II pseudo-op SPE (Symbolic Parameter Entry) is as follows:

SYMBOLIC PARAMETER SPE, SYMBOLIC LOCATION, LENGTH

- a. Symbolic Parameter - same as for PLE pseudo-op description.
- b. SPE - the pseudo-op
- c. Symbolic Location - indicates the starting bit address of the memory area from which the parameter is to be obtained at Job File Generation time.
- d. Length - the length in words and bits of the parameter; the maximum length is  $511.63_{10}$  words.



3. The format for HAP II pseudo-op SPEI (symbolic parameter entry immediate) is as follows:

SYMBOLIC PARAMETER SPEI, SYMBOLIC LOCATION, LENGTH

- a. Symbolic Parameter - same as for PLE pseudo-op description
- b. SPEI - the pseudo-op
- c. Symbolic Location - indicates the starting bit address of the memory area from which HAP II will obtain the parameter to place in the SPEI entry in the Symbol Translator Table.
- d. Length - the length in bits of the parameter; the maximum length is  $54_{10}$  bits.

D. Machine Language Modifications

1. Absolute (machine language), one-use changes can be made to a program (e.g., for "gimmicks") by specifying an absolute parameter location in place of the symbolic parameter name field on a Parameter Card that accompanies the request for a program.

When an absolute parameter location is specified, there is no parameter description already available (as there is for a symbolic parameter name in the Symbol Translator Table) and therefore the parameter description must accompany the absolute parameter location on the Parameter Card.

2. The forms in which the absolute parameter locator and the parameter description can be specified are as follows:

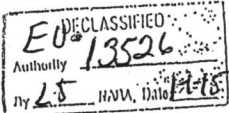
a. The starting bit address of the area in which the parameter is to be placed will be of the form: word address, period, bit address within the word (3872.37, 24376.0); the period and a bit address of zero must be explicitly specified.

b. If the address is preceded by a "D" in parens, the number is decimal; otherwise it is assumed to be octal.

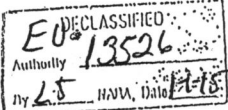
c. (1) If the parameter itself is symbolic, the address is followed by an "S" in parens; the specification of a symbolic parameter must be on one Parameter Card.

(2) If the parameter is composed of characters which are to be converted card-column by card-column, the address is followed by Type, "E", and the # of columns of characters to be converted on this card; these two items are separated by a comma and enclosed in parens. The specification of a character parameter may require more than one card, in which case a +1 is appended to the address of the second card (this precedes the parameter description), a +2 for the third card, etc.





(3) If the parameter is numeric, the address is followed by type (B, O, or D for binary, octal, or decimal), the field length in bits (including sign, if any), and the fraction length in bits; these three items are separated by commas and enclosed in parens. If the numbers have no fractional part, the fraction length field may be specified to be zero, may be left blank, or may be omitted completely. The specification of the numbers which accompany the absolute parameter location may require more than one card, in which case a +1 is appended to the address for the second card, a +2 for the third card, etc., and the parameter description need not be repeated.



## II. PARAMETER INSERTION AND CHECKING

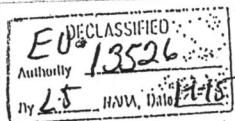
### A. Parameter Record Processing

1. When the request for a particular program is read from the REQUEST FILE by the JOB FILE GENERATOR program, the requested program is found on the PROGRAM LIBRARY and the record containing the Library Program Ident, Lower Memory, and the Symbol Translator and Symbol File Tables is read into memory. First, all Input/Output Statement (IOS) records accompanying the Program Request are processed, then all parameter records are processed.
2. Each parameter is translated into absolute location and absolute parameter. All parameters translated have the associated "INPUT?" bit in the program's Symbol Translator Table set to one. If an entry cannot be found in the Symbol Translator Table which corresponds to the symbolic parameter name or symbolic parameter, the program and the job containing the request for the program are rejected with appropriate logging on the JOB FILE LIST Tape. Each parameter that is to be inserted into Lower Memory is inserted at time of translation; each parameter that is to be inserted into Upper Memory is, after translation, saved in a work area of the JOB FILE GENERATOR program.
3. After all parameters have been translated, a comparison is made of "INPUT?" and "MUST" bits of each entry in the Symbol Translator Table. If any entry has "INPUT?" bit set to zero and "MUST" bit set to one, the program and the job containing the request for the program are rejected with appropriate logging on the JOB FILE LIST Tape.

If all "MUST" parameters have been translated, then Lower Memory is written on a work tape. Then the Upper Memory record of the program is input from the PROGRAM LIBRARY and the already translated Upper Memory parameters are inserted.

### B. Parameter Checking by the Problem Program

1. Control is then given to the JOB FILE GENERATOR's Check Control routine. This routine writes the main part of the JOB FILE GENERATOR on a work tape, inputs Lower Memory from the work tape on which it was written into its proper assembly area, and then branches to the program's Parameter Check Routine.
2. Parameter checking and/or "house-keeping" for the program can be done at this time. No input/output or interrupt facility is available to the program at this time. If the program finds something wrong with its parameters which is associated with a particular entry in its Symbol Translator Table, the program may tag the entry as incorrect by setting the "ERROR" bit in the entry to one. After completion of its Parameter Check Routine, the program returns control to the JOB FILE GENERATOR's Check Control routine indicating a rejection or acceptance of the parameters which have been inserted.



3. If a parameter acceptance is indicated on the JOB FILE GENERATOR's Check Control routine by a problem program's Parameter Check Routine, the program is written on the JOB FILE, the JOB FILE GENERATOR is input from the work tape on which it was written, and control is returned to it.

4. If a parameter rejection is indicated to the JOB FILE GENERATOR's Check Control routine by the program's Parameter Check Routine, the program and the job containing the request for the program are rejected with appropriate logging on the JOB FILE LIST Tape.

Prepared by Richard T. Burch, MPRO-104