ADVANCED COMPUTER PROGRAMMING

A Case Study of a Classroom Assembly Program

F. J. Corbató J. W. Poduska J. H. Saltzer

The M.I.T. Press
Massachusetts Institute of Technology
Cambridge, Massachusetts, 1963

Library of Congress Catalog Card Number: 63-20529

Printed in the United States of America

PREFACE

The present book is a case study of an assembler-compiler program. It is intended to be an advanced programming text for college students, system programmer trainees, and anyone trying to acquire a general understanding of system programming techniques. We feel that laboratory exercise is an important vehicle for teaching the techniques discussed in this volume. Therefore, the translator program example used must be written in an existing language of an existing computer. We consequently have chosen the FAP language of the IBM 7090 computer to describe the translator program. Other reasons for this particular choice are given in Chapter 1. Any loss of generality is partially offset by the fact that the 7090 is currently the most widely used large-scale computer in the world and one to which many colleges and universities have access.

The motivation for the present work began with the large gap between the usual beginning digital computer programming course and the sophisticated system programming techniques of interest in programming research and development. It was felt that too many students were uncritically using the existing programming systems and were overawed by the apparent complexities in such programs as the original FØRTRAN compiler.

In order to serve as an introduction to system programming and to convince the student that the principles of translators are relatively few and basically simple, a Classroom Assembly Program named CAP was written. It was first used in November 1960, in the M. I. T. course 6.251, Digital Computer Programming Systems. Since then, an execution monitor program has been added for the convenience of both students and instructors.

Course 6.251, where CAP has been used, is a one-semester introductory course of 12 units (3 contact hours per week, 9 hours preparation.) The course begins with study of an algebraic language such as FØRTRAN or MAD. The next section covers a machine language such as FAP. The third section is devoted to the study of the CAP assembler-compiler. During the semester, the course attempts to present most important contemporary ideas about computer programming. Many of these ideas are then illustrated in the CAP exercise.

Specifically, CAP has been used as follows: Students after studying the translator have been expected to make specified improvements and changes to it, using 6 to 8 computer runs for debugging purposes. (More ambitiously, the students could have written CAP from the specifications, but insufficient computer access prevented this for even the better students.)

For each of the eight semesters that CAP has been taught, the student enrollment, which has been gradually increasing, has been a cross section of the more than twenty departments at M. I. T. Thus we conclude that the <u>average</u> student is able to grasp and enjoy the basic principles of a translator program when it is appropriately presented.

The reader is assumed to be able to program in the FAP machine language sufficiently well to know how to look up features of the FAP assembler or of the 7090 computer in the

IBM published reference manuals.*† He is assumed also to be acquainted with the Binary Symbolic Subroutine (BSS) linkage and relocation used in the IBM FØRTRAN Monitor System (described in the FAP Reference Manual).*

The book is organized into two major divisions, the description of CAP (five chapters) and the appendices containing listings of the CAP assembler. The compiler part of the program is considered to be advanced material, and the text advises the beginning reader which parts may be safely skipped over.

The appendices include listings of both the assembler-compiler program and of the execution monitor program. The listing of the assembler-compiler is essential to an understanding of the text. The execution monitor listing, while not so important, is included for two reasons. First, an advanced student may make the execution monitor a further case study in advanced programming techniques. Second, it is included for completeness, for the instructor who may wish to adapt it to his needs. It should be noted that the execution monitor program does make use of a few specific features of the current M.I.T. FØRTRAN Monitor System and 7090 computer.

Acknowledgment should be given to the efforts of the many teaching assistants who have labored to make the use of CAP effective. Particular mention is made of Neil Haller for his work on the early stages of CAP and introducing the first version of the execution monitor program, and of Neil Barta for his preliminary description of the UPDATE feature of FAP, from which a major part of Chapter 5 is adapted. We also are especially appreciative of the useful comments on the present manuscript made by Neil Barta and Thomas Hastings.

The programs described in this book were developed at the M.I.T. Computation Center, Cambridge, Massachusetts.

Cambridge, Mass.
May, 1963

F. J. Corbató
J. W. Poduska
J. H. Saltzer

---

* Reference Manual, FØRTRAN Assembly Program (FAP), IBM Publication C28-6235 (September, 1962).

† Reference Manual, IBM 7090 Data Processing System, IBM Publication A22-6528-4 (March, 1962).

CONTENTS

Chapter 1

INTRODUCTION

In an age of increasing complexity, the reader may reasonably ask why he should want to learn the innermost structure of a digital computer programming system. For the day of the renaissance man is indeed past; the intricacies of present-day knowledge as well as the limitation on time for comprehension, of necessity, allow a person to be a specialist in but a limited number of areas. The answers will vary, but it is inescapable that digital computers have already during their short presence become an immensely important device in modern society. As for the future implications, the only issue of debate is whether or not computers are bringing a second industrial revolution as the steam engine heralded the first. Examples of the penetration of computers into our daily activities abound; to name but a few: banking, payroll processing, production and inventory control, income tax processing, satellite orbit computation and tracking, numerically controlled machine tools, airline reservation systems, and military defense communication networks.

Because digital computers have become important, it is inevitable that the accompanying system programs will grow in importance too. For computers reach a high level of effectiveness only when the programming systems allow the ultimate user of the system to program directly—albeit often unknowingly by that name—and thereby avoid intermediary programmers. The development of these direct usage languages is presently limited by the ease and rapidity that suitable translation programs can be written. These translation programs, are variously named problem oriented language processors, compilers, or assembly programs, depending on the language level at which they meet the user. Today, more and more, a computer is incomplete without an accompanying programming system of considerable sophistication.

Moreover, computer systems are still rapidly evolving in many directions: The detailed circuit technology is still making great strides, the logical design is changing to include multiconsoles and multiprocessors, and the programming systems are being enlarged to include larger roles such as the time-shared operation of the computer. It is important in this highly fluid state of affairs that others in addition to the system programming specialists have an understanding of programming systems. What is needed for the optimum use of computers in the future is that responsible individuals within computer-affected organizations understand the problems and general techniques of programming systems to the same extent that the problems and techniques of computer hardware are now understood. For without knowledgeable and critical guidance there will be not only many costly abuses of computers but there will be little vision and few ideas for new computer applications.

To give the reader insight into contemporary programming systems, the following chapters will present a case study of the inner structure of a combination assembler-compiler program. The program is called CAP, an acronym for Classroom Assembly Program, and it contains many of the typical features of present-day translators. The case study technique will prove helpful since there are many interrelated factors to consider and

discuss. As well as acquiring an inner knowledge of a translator, the reader of CAP will acquire three additional benefits, namely:

1. The study of detailed programming techniques.
2. How to read and study a large program.
3. How to organize a large program.

For several reasons the CAP program has been written in the FAP symbolic machine language of the IBM 7090 computer. A machine language representation has been specifically chosen because of its concreteness and lack of ambiguity for the reader. This reason is especially pertinent when one considers that one of the principal objectives of the study of CAP is to remove the mystery of system programming and to establish a feet-on-the-ground attitude in the reader. Finally, the FAP language, rather than SØS, for example, has been used in order to have its powerful subprogram feature which allows separate translation and rigid independence of program segments—a feature which greatly assists the initial understanding of a large program.

CAP is weaker than the usual translators, such as FAP, in that it has only subsets and examples of various special features and does not have the machinery for separately translatable subprograms. CAP differs from FAP in style, too, in that it is more elegantly written (that is, in terms of simplicity, brevity, and clarity) and highly organized with many subprograms. The CAP style is in contrast to that of many translator programs in active use where extreme short-cuts have been used in the interest of minimizing operating speed. (Often the short-cuts used are analogous to those for reducing the cost of commercial television receiver and frequently shortsighted from a maintenance point of view.) The basic techniques used in translators remain the same, however, so that CAP is a valid program from which to learn. One feature of CAP that merits comment is that although intermediate tapes are simulated, the program fits entirely in core memory and is independent of intermediate storage devices. Present-day translation programs have frequently overlooked the speed advantages of remaining entirely in core memory particularly while translating short subprograms which should be the major use when a translator allowing subprograms is utilized.

Finally, before proceeding with the remaining chapters, discussion is in order on how to study CAP. Past experience with many students indicates that the following advice is useful:

1. Obtain an understanding of what CAP does from the point of view of a user.
2. Determine the specifications of CAP as a program.
3. Determine the specifications of subroutines PASS1, and PASS2.
4. Starting in PASS1, study the specifications of the successive programs in the hierarchy of subprogram usage. (Omit the compiler.)
5. Starting at the top of the hierarchy, study how each subprogram meets its specifications. Review steps 2 to 4 sufficiently often that you are always sure of <u>what</u> a program is supposed to do before considering <u>how</u> it does it.
6. Remember that all subprograms can only communicate by means of their calling sequences because they are separately translated.
7. When studying, it is a great advantage to know that a program has been debugged. Nevertheless, there will always be sections of program which appear not to work correctly. After spending a reasonable amount of time, if no progress is made, avoid getting bogged down by jotting down on a pad the uncertain point for later discussion with others.
8. The compiler can be studied easily after the basic CAP is understood.
9. The advanced student can improve his program analysis abilities, by studying the execution monitor program, although it is given largely for reference purposes.

Chapter 2

CAP USER'S REFERENCE MANUAL

2.1 The CAP Language

Before we begin to study how the CAP assembly program works, we should pause to determine exactly what job it is intended to do. We can perhaps get the best picture of this job if we examine the user's reference manual for the CAP language. This reference manual is the subject of the present chapter. The brevity of the reference manual is at once an indication of the simplicity of the CAP language and of the assembly program itself.

2.2 Card Format

CAP instructions are typed one to a card as shown in Figure 2.1. Columns 1 to 6 are known as the symbolic location field and may contain a symbol or blanks. Columns 7 and 12 are always blank, leaving room for a three or four letter operation code in the operation field, columns 8 to 11. The variable field begins in column 13 and terminates at the

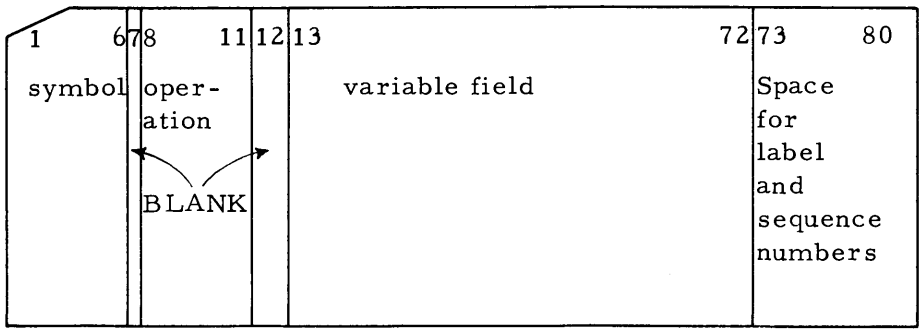


Figure 2.1. Format of CAP symbolic cards.

first blank column, or column 73. An arbitrary comment may follow this first blank column. This comment will be ignored by the assembly program as will the sequence number field, columns 73 to 80.

Symbolic Location Field

This field may contain a symbol, a string of one to six characters, at least one of which is nonnumeric, and none of which are the following eleven special characters:

+ - * / , = . ' ( ) $

A symbol may be defined only by its appearance in the symbolic location field of some instruction card.

Operation Field

This field may contain a mnemonic associated either with one of thirty-four 7090 instructions or one of five pseudo-operations. The allowed 7090 instruction mnemonics are

ACL	ANA	CAL	CHS	CLA	CLS	CØM	FAD
FDP	FMP	FSB	LAC	LAS	LBT	LDQ	LGL
LGR	LXA	ØRA	PBT	RQL	SLW	STØ	STQ
SXA	TIX	TMI	TPL	TQP	TRA	TSX	TZE
XCA	XCL						

The instructions LAC, LXA, SXA, and TSX are assembled with a tag of 4. The instruction TIX is assembled with a tag of 4 and a decrement of 1.

The allowed pseudo-operation mnemonics are

REM INT ØCTL CØMP END

The effect of these pseudo-operations is explained in a later section.

Variable Field (Operations)

The variable field specifies the address of an operation. It may contain an expression consisting of a string of symbols and decimal integers connected by the break and grouping characters:

\+ – * ( )

All multiplications must be made explicit by the use of the asterisk even if one of the operands is a parenthetical expression. The variable field is evaluated in signed 35 bit integer arithmetic. If the result is negative, it is two's complemented before the final step in which the answer is taken modulo 2^{15}. The result is combined with the specified operation code by a logical "OR".

Sequence Number Field

Columns 73 to 80 may be used for labeling and sequence numbering and are ignored by the CAP assembly program.

2.3 Pseudo-Operations

REM The REM pseudo-operation is used to introduce an arbitrary remark into the assembly listing. Card columns 1 to 80 will be printed and the card will be otherwise ignored by the assembler. If a symbol appears in columns 1 to 6, it will be ignored.

INT INT is a data-generating pseudo-operation. The variable field of the INT pseudo-op consists of signed decimal integers separated by commas and terminating at the first blank column. For each decimal integer, a word is assembled with the decimal integer

inserted in the left half of the word. A comma with no integer following it will cause a word of all zeros to be assembled. A decimal integer may be preceded by a minus sign and must be of absolute value less than 2^{17}. A symbol, if any, appearing in the symbolic location field will be defined to be the location of the first integer assembled. Succeeding integers will be placed in succeeding locations in core storage.

ØCTL The twelve characters in card columns 13 to 24 are taken to be octal digits and are used to form a 12 digit octal word in core storage in the next location to be assigned by the assembler. If the characters appearing in columns 13 to 24 are not octal digits, an incorrect word will be generated and no error indication will be made. A symbol in the symbolic location field will be defined to be the location of the generated word.

CØMP The CØMP pseudo-op specifies that the entire variable field, columns 13 to 72, is taken to be an arithmetic statement which is to be compiled, in much the same manner as in FØRTRAN or MAD. Blanks are ignored and commas may be used to indicate tagging. The arithmetic statement must consist of a symbol followed by an equal sign and followed by an arithmetic expression. This expression may consist of symbols connected by the break and grouping characters:

\+ − * / ( )

Numbers in the expression will be taken as symbols referring to memory locations. The indicated arithmetic expression will be compiled in floating point arithmetic, and a list of the instructions compiled will appear on the CAP assembly listing. If a symbol appears in the symbolic location field of the CØMP card, its value will be the location of the first compiled instruction.

END This pseudo-op marks the physical end of the program and defines the entry point to the program to be the value of the expression in the variable field. If a symbol appears in the symbolic location field, it is given the value of the first location not used by the program.

2.4 Use of CAP

CAP is a package of subroutines which is called by

```
      .
      .
TSX         $CAP,4
      .
      .
```

The AC should contain the location in core storage into which the first instruction of the symbolic program is to be assembled. When CAP is finished it will leave in the AC the entry point to the program. The sense register (SI) will be nonzero if any assembly errors were noted by CAP.

2.5 Output of CAP

The CAP assembler has two outputs, a printed assembly listing, and a binary machine program. The listing consists of one or more printed lines for each instruction card in the symbolic input deck. This line contains the 80 columns of the original card, the

12 digit octal word which CAP has assembled as well as the octal location in which the instruction has been placed, and pertinent coded error indications. In the case of CØMP pseudo-ops, the CØMP card will be printed and followed by a list of the instructions generated by the compiler in the format described earlier. The assembly listing is written on an output tape for later printing. The binary machine program is left in core storage beginning at the location specified by the program which called CAP.

2.6 Restrictions and Error Indications

1. No more than 100 symbols may be defined. If this restriction is exceeded, further symbols are ignored and a comment is printed at the beginning of the assembly listing, and SI bit 17 will be turned on.

2. All operation codes must be among those listed earlier in this chapter. If an illegal operation code is encountered, it will be treated as zero, SI bit 34 will be turned on, and the letter "O" will be printed on the assembly listing next to the offending instruction.

3. All symbols appearing in variable fields and CØMP statements must be defined. If an undefined symbol is encountered, it will be given value zero, SI bit 35 will be turn on, and the letter "U" will be printed next to the offending instruction.

4. The variable field of an INT pseudo-operation must contain only decimal integers, preceded by plus or minus signs and commas. If an illegal character is encountered, that word will be assembled as zero, SI bit 33 will be turned on, and the letter "E" will be printed on the assembly listing next to the offending pseudo-op.

5. No more than 200 separate elements and break characters may appear in a CØMP statement. If this restriction is exceeded, the CØMP statement is skipped, and SI bit 14 will be turned on.

6. No more than 125 nested parentheses may appear in an arithmetic expression in a variable field. If this restriction is exceeded, an incorrect value may be computed and SI bit 15 or 16 will be turned on, depending on the nature of the parentheses count.

The following two restrictions occur when CAP is run under the Classroom Execution Monitor described in Chapter 5:

7. No more than 150 cards may appear in the symbolic program.

8. The symbolic program must not assemble into more than 256 binary machine instructions or require more than 300 card images to be written on the collation tape.

Chapter 3

THE CAP ASSEMBLER

3.1. How Does an Assembler Work?

In this chapter we shall examine in detail the workings of CAP and of assembly programs in general. While references to the exact coding of CAP are specific to this assembly program, the general discussion and flow charts are common to most assembly programs for most computers.

The purpose of any assembly program is to translate the symbolic cards describing a machine language program into that machine language program. For convenience, this translation can be considered to consist of two operations: First, the mnemonic codes representing machine operations must be replaced by the binary machine codes representing those same operations, and these binary codes must be assigned locations in core storage. Second, the symbolic variable field of each instruction must be evaluated in terms of the symbols appearing in the symbolic location fields of other instructions, and the resulting address must be inserted in the instruction. Consider the following program, written in the CAP language:

```
        CAL  BITS     GET CØUNT.
        SLW  WØRD     SAVE.
HERE    TRA  HERE     STØP.
BITS    INT  6        BIT CØUNT.
WØRD    INT  0        STØRAGE FØR BIT CØUNT.
```

In order to translate the first instruction, CAL BITS, we need to know two things. First, what is the binary machine code corresponding to the mnemonic CAL? Second, what is the value of the address part of the instruction, that is, what is the value of the symbol BITS? The first question can be answered by reference to a table of operation mnemonics and machine codes, an essential part of any assembler. The second question, however, requires knowledge of which symbolic card has the label BITS. This knowledge can be gained only be going completely through the symbolic deck once to determine the location value of each symbol.

We see, then, that the assembly program must go through the symbolic cards twice. The first pass through the symbolic cards is required to assign each instruction to a place in core storage and thereby to define the value of the symbol, if any, appearing in its symbolic location field. Then, on the second pass through the cards, it is possible to evaluate the variable field of each instruction on the basis of the symbols defined on the first pass.

We may expect, therefore, that CAP will exhibit a basic structure consisting of two passes through the input symbolic card deck. In fact, since CAP is coded in the form of independent subroutines, we shall find that this two-pass structure is handled by two subroutines, named, conveniently, PASS1 and PASS2. These two subroutines are called by

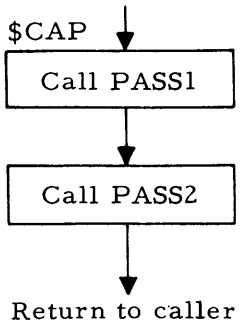


Figure 3.1. Flow diagram of subroutine CAP.

another single subroutine named CAP. (The reader should note that the name CAP will hereafter be used both for the entire assembly program and for the subroutine which calls PASS1 and PASS2. The meaning of any particular usage should be clear from context.) Let us examine a flow diagram of the subroutine CAP, in Figure 3.1.

The CAP subroutine is called by the sequence

```
      .
      .
CAL   ØRG
TSX   $CAP,4
      .
      .
```

in the main subprogram. (See listings of MAIN and CAP in Appendix 1.) ØRG specifies the location in core storage at which the machine language program assembled by CAP is to start. Subroutine CAP then gives this information as an argument to subroutines PASS1 and PASS2, which perform the two passes through the symbolic card deck mentioned earlier.

Note that subroutines PASS1 and PASS2 upon encountering errors turn on bits in the sense register (SI); subroutine CAP therefore clears the SI before calling each subroutine, and saves its contents upon return. The main program, upon return from CAP, could determine if the assembly was successful by examining the SI, although it does not do this.

3.2 Pass One, Symbolic Definitions

It is stated earlier that the purpose of the first pass is to assign each instruction a place in core storage and thereby define all symbols appearing in location fields of the symbolic program. The procedure involved in doing this is, as might be expected, quite straightforward. First, an instruction location counter (ILC) is set to contain the location where the first instruction is to be assembled, which is the origin of the machine language program being generated by CAP. Then, a card is read. If it is not a pseudo-operation, the symbol, if any, appearing in the symbolic location field is defined, the card is put away in a place at which it can be found by pass two, and the ILC is incremented by one. The process is then repeated for the next card. If a pseudo-operation is encountered, some special processing may have to occur. For example, when the END card is encountered, pass one should terminate rather than continue reading cards. A flow diagram of pass one is shown in Figure 3.2.

If we examine the coding of the loop in subroutine PASS1, we find that it takes very few instructions, primarily because the difficult jobs are relegated to subroutines. For example, the box labeled "Read card" is handled by a subroutine named READ1. The entire operation, of determining whether there is a symbol to define and defining it to be the value of the ILC, is handled by another subroutine SYMSTØ. Similarly subroutine WCT1 handles the problem of saving the card for the second pass. If we believe that these subroutines work as their calling sequences specify, the understanding of pass one is greatly simplified.

In fact, the physically largest section of subroutine PASS1 is devoted to processing the pseudo-operations, even though this processing is perhaps the least important function of pass one. Let us examine what must be done when pseudo-operations are encountered. Perhaps the simplest procedure occurs for the pseudo-operation REM. In this case the loop is re-entered after skipping the operations of symbol definition and increasing the

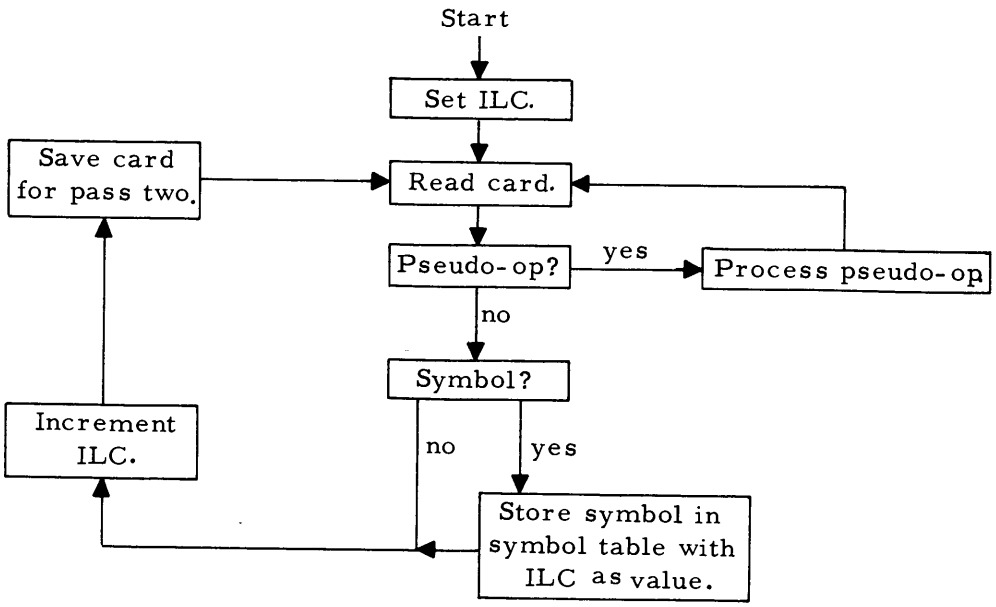


Figure 3.2. Flow diagram of the first assembly pass.

ILC. The only procedure of interest is saving the REM card for pass two. (See Figure 3.3, a flow diagram including pseudo-op processing.)

In the case of the ØCTL pseudo-operation during pass one, the only concern is the number of words of storage required (one in this case) and the definition of any symbol appearing in its symbolic location field. Therefore, it can be handled exactly like the ordinary operation codes, that is, by defining the symbol and increasing the ILC by one.

If an INT pseudo-operation appears, the same considerations apply as before. However, the variable field of the INT may specify that several words be generated. (See INT description in Chapter 2.) The variable field always specifies that at least one word should be generated. If there are to be additional words, for each extra word there will be a comma in the variable field. Therefore, the assembler may learn how many words will be generated simply by counting the number of commas in the variable field and adding one. Remember that the only concern of pass one is counting the number of registers used by the source program and defining symbols. The procedure used when an INT is encountered is, then, to test for and define the symbol in its symbolic location field, and to count the number of commas in its variable field. The subroutine CØMMA performs this last step, and also adds one plus the number of commas to the ILC. The loop is then re-entered for the next card.

The operation of the CØMP pseudo-operation will not be explained in detail here except to say that the symbol, if any, in columns 1 to 6 is defined, the card is saved for pass two, and a subroutine CØMPØP is called to process the pseudo-operation variable field. CØMPØP causes the generation of the instruction sequence required to carry out the computation indicated in the variable field and increases the ILC appropriately. The operation of subroutine CØMPØP is not essential to an understanding of pass one or the rest of CAP. A full discussion of the subroutine may be found in Chapter 4.

We come finally to the END pseudo-operation. When this card is encountered, pass one is complete except for certain simple terminal procedures. The subroutine ENDØP must first be called to finish off the work of the CØMPØP subroutine by making space at the end of the program for the temporary storage locations required by all the compiled

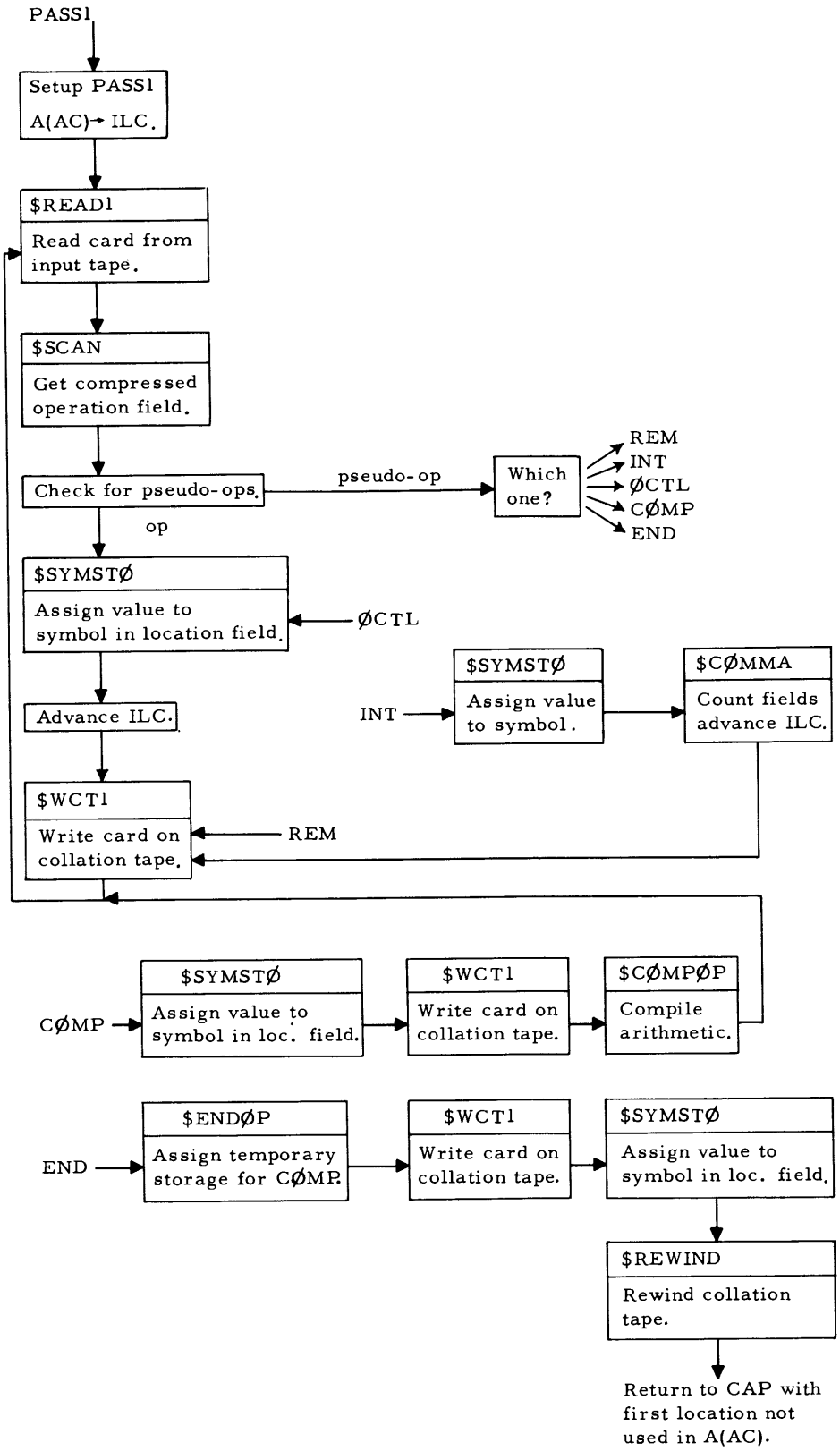


Figure 3.3. Flow diagram of subroutine PASS1.

instruction sequences. Then, the symbol, if any, in columns 1 to 6 of the END card is defined and the card saved for pass two. Since pass one is now finished the value of the ILC, which is now equal to the first location not used by the object program being assembled, is placed in the AC, and subroutine PASS1 returns to the program which called it.

3.3 The Collation Tape

It has been mentioned several times earlier that pass one must put the symbolic card images away in a place where pass two will be able to find and process them. While in principle it would be possible for pass two to backspace the input tape (or the operator to reload the card reader with the symbolic program), in practice it is much simpler for pass one to write the card images on a second tape, the collation tape. Pass one then ends by rewinding this collation tape, and pass two can begin again with the first card in the symbolic input program.

It is worthwhile noting, also, that when small symbolic programs (say, less than 150 cards) are being assembled, there is no reason why a collation tape is necessary, as there is enough room in the core storage of a 7090 to hold all the card images at once.

A common alternate procedure for larger programs is to collect a buffer of, say, 150 cards, then write the entire buffer on a collation tape at once. While the tape write takes place, the assembly program can be processing more input cards and storing them in a second buffer.

Still another method uses two collation tapes, collating half the input cards on one, then starting a rewind so that when pass two begins there will be no wait for tape positioning. The second half of the program is collated on the second tape, which is rewound at the end of pass one, and which will be properly positioned about halfway through pass two when it is needed.

If no collation tape is used, it is still convenient for pass one to call a subroutine to store the cards; the subroutine simply inserts them into a core memory buffer rather than writing a collation tape. Similarly, pass two uses a complementary subroutine which locates and transmits the core buffer rather than reading back from a collation tape.

3.4 Pass Two, Symbolic Evaluation

When all symbols have been defined by pass one, it is possible to finish the assembly by processing each card image in order, and determining values for its operation code and for its variable field. The purpose of pass two, it will be remembered, is to evaluate the operation code and variable field of each card, to assemble the binary machine word required to represent the instruction, and to print an assembly listing containing the original card and the octal equivalent of the machine word generated. Again, the basic procedure is straightforward, although pass two is a little more complicated than pass one. The ILC is again set to start at the origin specified by the program which called CAP.

The main loop of pass two then operates as follows: First, a card is read from the collation tape. If the card does not refer to a pseudo-operation, the operation code is evaluated by comparing it to entries in the operation table. The numeric code of the machine instruction corresponding to the given mnemonic is obtained from this table. Then, the variable field is evaluated. These two results are combined by a logical "OR" and inserted in core storage at the location specified by the ILC. (An alternate procedure might be to store the instruction in an output buffer for punching.) A line is printed

on the assembly listing containing the card image and the octal equivalent of the word that was inserted in core storage. Finally, the ILC is increased, and the loop repeated for the next card.

The main loop of subroutine PASS2 takes but a few instructions, as most of the difficult jobs are handed down to subroutines to perform. The cards are read from the collation tape by subroutine READ2, and the assembly listing is printed by an internal subroutine PRNT1. The most difficult job, evaluation of the variable field on the basis of the symbols defined in pass one, is handled by subroutine VAREVL.

As in pass one, the physically largest section of coding in pass two is that involved in processes not strictly important for an understanding of how pass two works, that is, processing the pseudo-operations, and printing the assembly listing. The pseudo-operations are handled as special cases as they were in pass one, by performing some simple operations and re-entering the main loop at a strategic point. Let us examine them again, one at a time, to learn how they each fit into pass two.

The REM Pseudo-operation again is the simplest of the pseudo-ops. The REM card is printed on the assembly listing, and the loop re-entered at the point where the next card is read. (See Figure 3.4). A slightly different print subroutine is used, as no octal word was generated for the REM pseudo-op and nothing need be printed in the columns normally used for printing the octal word.

The CØMP pseudo-operation is handled exactly like the REM pseudo-operation in pass two, since all compilation operations were finished in pass one. (See Chapter 4 for details on the CØMP pseudo-operation.)

The INT pseudo-operation is taken care of very simply by calling a subroutine INTØP to evaluate the variable subfields and to insert the results in core storage. The INT card is printed on the assembly listing along with the first machine word generated.

The ØCTL pseudo-operation is handled on the spot by PASS2 as an example of in-line coding. A BCD-binary conversion is performed, the result inserted in core storage, and the ØCTL card printed on the assembly listing.

As a last step for each of the above pseudo-operations, the pass two loop is re-entered at an appropriate place. In the case of the END pseudo-operation, however, the loop terminates. The variable field of the END card is evaluated by subroutine VAREVL, and this value is saved (and printed) as the entry point to the assembled machine program. Pass two is now complete. The error flags, if any, are placed in the SI, and PASS2 returns to the program which called it.

A comment on the error flags in subroutine PASS2 is in order at this point. Whenever an undefined symbol is encountered in a variable field by subroutine VAREVL, or an illegal operation code by PASS2, or an INT error by subroutine INTØP, an appropriate bit in the sense indicator register is turned on. The subroutine used to print out the assembly listing examines the SI and prints any error flags next to the instruction being processed. The SI is then set to zero before the next instruction is processed. In addition, one cell is kept throughout pass two which contains the logical combination ("OR") of all the error bits of individual instructions. It is this last cell that is placed in the SI when pass two is finished.

3.5 VAREVL, Evaluation of the Symbolic Variable Field

We now come to the problem of evaluating the symbolic variable field of each instruction; a problem often considered to be the essence of the assembly process. At first glance, given that the values of the symbols which might appear in a variable field have been defined during pass one, we might think that this evaluation would be quite easy. In fact, if we were asked to carry out such an evaluation we would have no difficulty working out the answer in a short time. However, the algorithm needed for the evaluation is

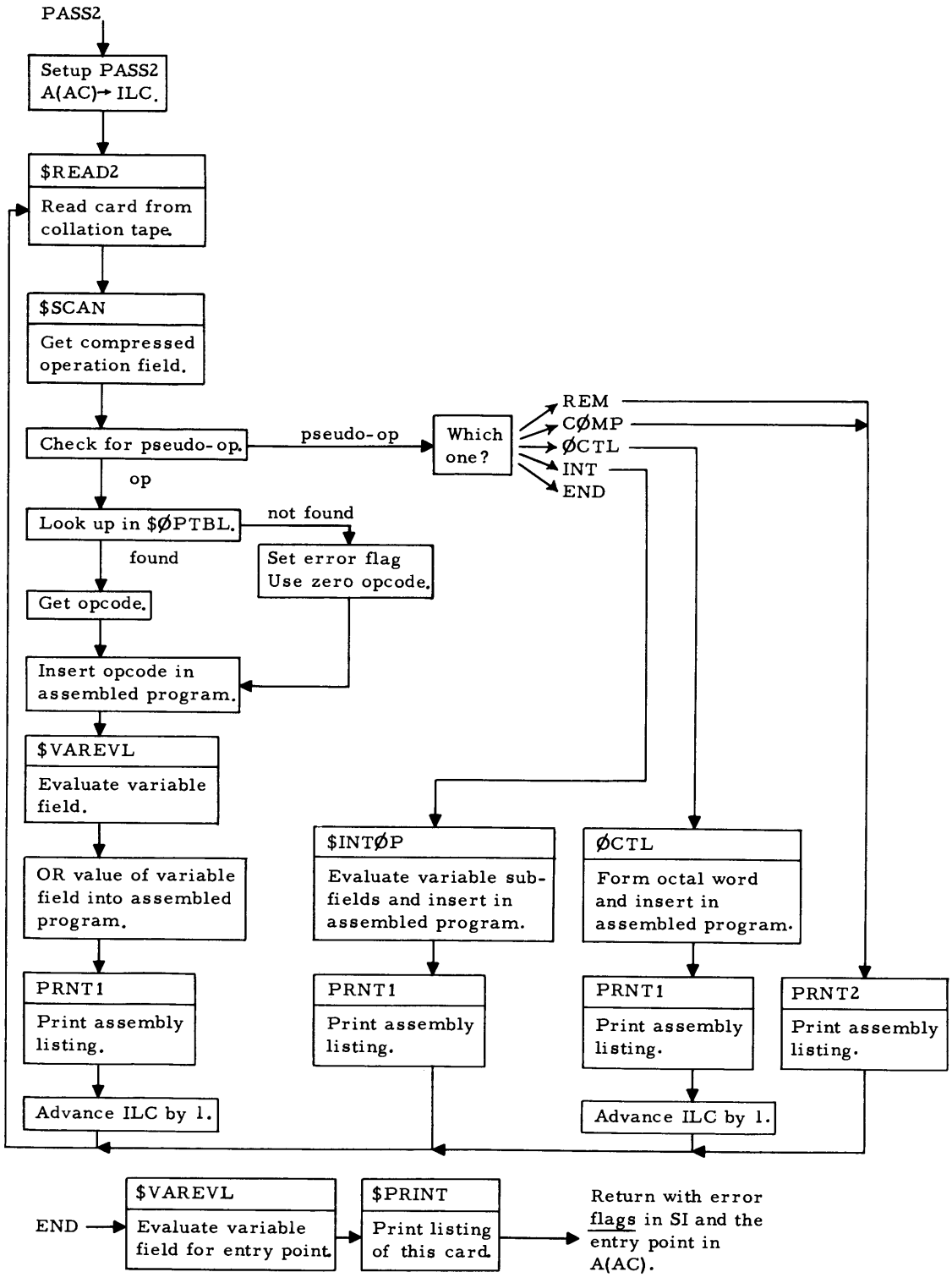


Figure 3.4. Flow diagram of subroutine PASS2.

surprisingly complicated, because of the existence of an implied order of operations in the mind of the person writing the expression. Consider, for example the following CAP symbolic instruction:

```
X    CAL   ALPHA+4* BETA
```

where ALPHA and BETA are symbols which appear in the symbolic location fields of cards elsewhere in the program. In evaluating the expression "ALPHA+4* BETA", the multiplication must be carried out before the addition operation, or else an answer will be obtained which is different than the one intended by the writer of the expression. Although this order of precedence is a usual convention in mathematical notation, it must be systematically observed by the assembler when evaluating the expression.

Let us examine a moderately complicated expression and see what sort of combinations of symbols may appear. After figuring out what procedure is used in each of these cases, a general procedure will begin to emerge which can be formalized into an algorithm for the evaluation procedure.

Let us take, as an example, the symbolic expression

```
+4* ABC-ALPHA+S* 2
```

and assume that ABC, ALPHA, and S are defined symbols. We first observe that a symbolic expression can be characterized as a string of elements (symbols or decimal integers) separated by break characters and terminated by a blank column. The allowed break characters represent the binary operations of addition (+), subtraction (-), and multiplication (*), and the unary plus and minus sign. For the moment, the ability to handle parenthetical expressions will be ignored. The unary plus at the beginning of the expression, if not provided by the programmer, is automatically inserted as a first step of evaluation.

To formalize the scan of this expression, let us create three windows which can be moved across the expression in such a way that the center window always shows us an element, and the left and right windows show us the break characters on the corresponding left and right sides of that element. For example, if the windows were placed on the above expression as far to the left as possible, we would obtain:

```
[+] [4] [*]  ABC-ALPHA+S* 2
```

What does this combination of operands imply? First, the plus sign on the left signals that we are starting to evaluate a term. The asterisk on the right signals that there are more things to come in this term, so the saving of the element in the center for a future multiplication is all we can do. The element is saved in a location named "term" ready for reference later.

Now, move the windows to the right until the next element falls in the center. We obtain

```
+4 [*] [ABC] [-]  ALPHA+S*2
```

Again examining the left and right break characters to decide what should be done, we argue as follows: The asterisk on the left tells us to multiply the old value of the term by the value of the present element. This result may be returned to the storage location "term". The minus sign on the right signals that the term has come to an end, and that the value stored away in "term" should be added into the "sum" register for this expression.

Now, move the window to the right again. This time, we obtain

```
+4* ABC [-] [ALPHA] [+]  S*2
```

The left window exhibits a minus sign signaling the start of a new term, a negative one at that. Therefore, we may store away the negative of the value of the present element in the location "term". The plus sign on the right again signals the end of the term, and that the value of the term should be added to the "sum" register.

Moving the window once more, we obtain

+4* ABC-ALPHA [+] [S] [*] 2

This combination of operators is identical to that found at the beginning of the expression so that we may follow the same procedure. First, on the basis of the plus sign we store away the value of the present element since we are starting a new term. Second, since the * indicates that there is more to come in this term, we must wait until later elements are brought into consideration.

Finally, with the window in its next and last position, we have

+4* ABC-ALPHA+S [*] [2] [ ]

This time the situation is similar to one encountered before, except for the lack of an operator in the right window. The left break character again requires us to multiply the value of the term collected so far by the value of the present element. The blank appearing in the right window tells us to add the term into the "sum" register and stop, as the evaluation of the symbolic expression is complete.

Although, this procedure seems complicated, let us see if we can develop a flow diagram describing the algorithm. The procedure has the following characteristics: After moving the window, we first examine the break character in the left window, do something about it, then examine the break character on the right. After processing on the basis of this right break character, we move the window and repeat the same series of steps. This procedure is formalized in the flow diagram in Figure 3.5. If we follow the

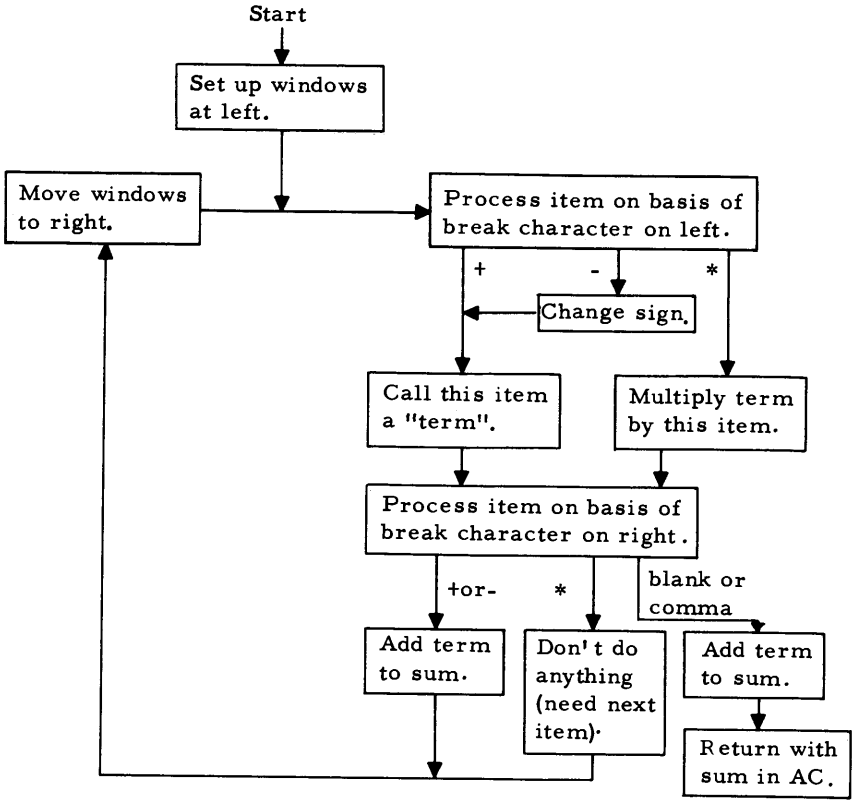


Figure 3.5. Flow diagram of subroutine EVAL.

flow diagram through for the expression examined previously, we see that it carries out each of the operations described. This flow diagram describes the operation of the subroutine EVAL, which is internal to the subprogram VAREVL. An important procedure which is implicit in this flow diagram is that of evaluating the item appearing in the center window. If the element is a decimal integer, a decimal-to-binary conversion must be made. On the other hand, if the element is a symbol, its value must be looked up. This lookup procedure is done by the subroutine SYMGET which acts as a complement to the subroutine SYMSTØ used during pass one.

How EVAL is Called

EVAL is an internal subroutine of the subprogram VAREVL. The subprogram VAREVL itself simply sets up EVAL and calls it properly; when EVAL has finished evaluating the expression, VAREVL handles the operation of reducing the answer to a core memory location. (See Figure 3.6, a flow diagram of VAREVL.)

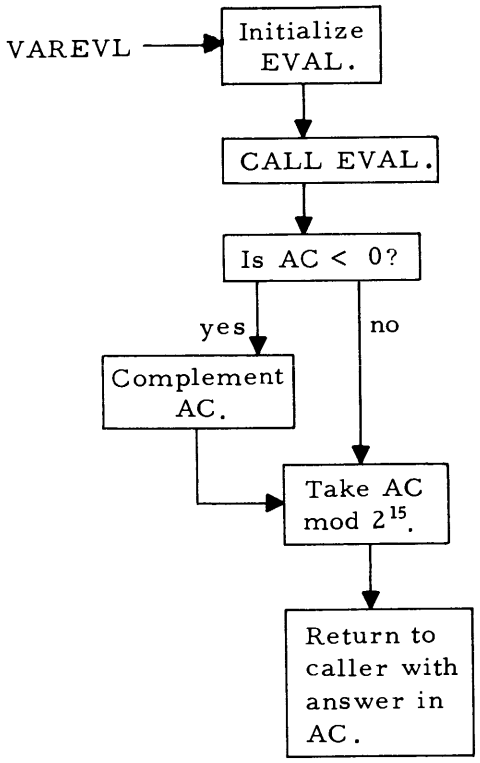


Figure 3.6. Flow diagram of VAREVL.

Making EVAL an internal subroutine of VAREVL allows EVAL to be defined recursively. That is, if the occasion should arise that EVAL needs to have a subexpression evaluated, it can call on subroutine EVAL to do the job. One might expect to get into difficulty with this procedure, since when EVAL is called recursively, it will change many registers and temporary results. We will see that this difficulty is circumscribed by picking out critical temporary results and saving them in a special way.

In terms of the picture described above, a parenthetical expression may be considered to be an element which appears in the center window. Whenever the center window is determined to contain a parenthetical expression as an element, the element is evaluated by calling the subroutine most able to handle the evaluation of an expression, namely subroutine EVAL. In order to call EVAL, it is necessary to save away temporary results, such as the values of the "term" and "sum" registers that have been collected so that those registers may be used by EVAL for the subexpression evaluation. Then, when EVAL is finished evaluating the subexpression, the "term" and "sum" registers are restored; the evaluation of the original expression continues, using for the value of the element in the center window the answer obtained by EVAL on the recursive call.

Since the parenthetical expression may itself contain another nested parenthetical expression, EVAL must be very careful how it saves away its temporary results, as a second saving of temporary results might destroy the first set.

To handle this problem, two subroutines named SAVE and UNSAVE are used by EVAL. These two subroutines manipulate a last-in, first-out storage array called a push-down list. Each time subroutine SAVE is called, an item or block of items is stored in the list. When subroutine UNSAVE is called, the last item or block stored in the list is retrieved. Successive calls to UNSAVE retrieve items stored by earlier calls to SAVE.

EVAL, then, saves temporary results in the push-down list before calling itself, and retrieves the results later. If the expression requires repeated recursion, the pushdown list will save and restore the temporary variables in the proper order.

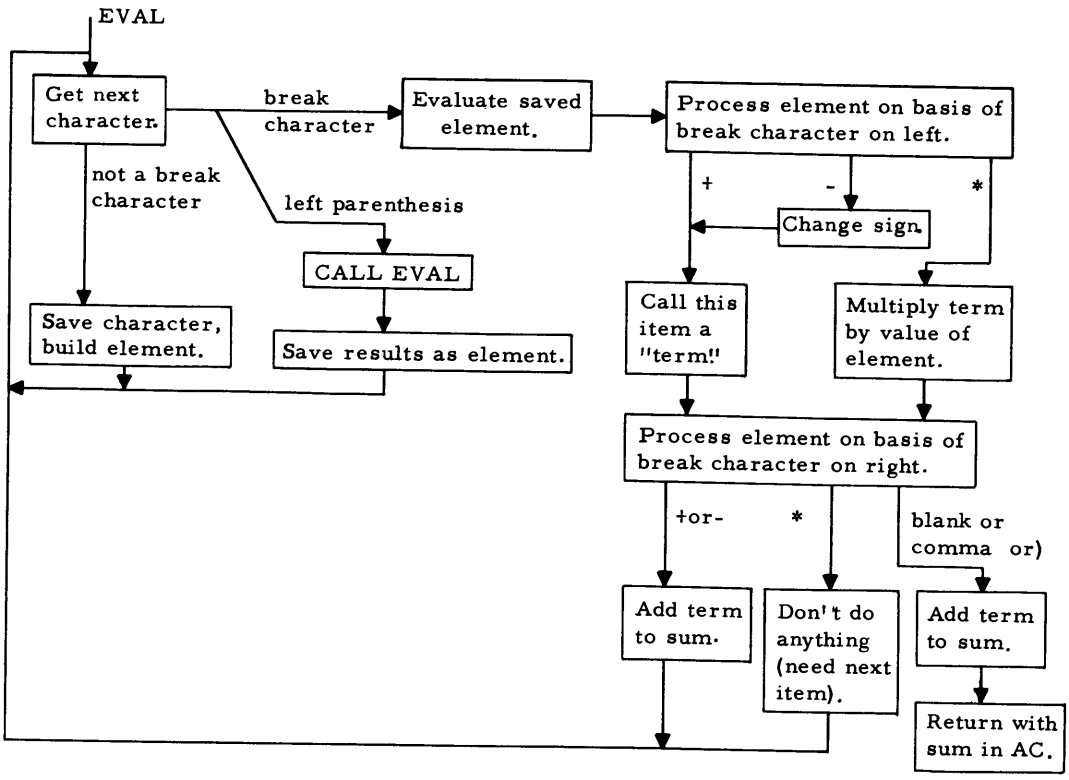


Figure 3.7. Flow diagram of EVAL with recursive capabilities.

Figure 3.7 is a flow diagram of EVAL with the ability to handle parenthetical expressions added. The recursive ability of EVAL is not essential to the understanding of the general expression evaluation procedure; it should be ignored in early study by assuming that no parentheses are encountered.

3.6 Subprogram Calling Sequences and Definitions

In this section, the calling sequences and a thumbnail description of each of the utility subroutines used in CAP are described. For reference, the same information about subroutines CAP, PASS1, PASS2, and VAREVL is reproduced here.

Primary Subroutines

CAP CAP is called by

```
..
CAL     ØRG
TSX     $CAP,4
..
```

Subroutine CAP causes the symbolic program written on cards and appearing on the input tape to be assembled in core storage starting at the location specified by the address portion of the accumulator.

PASS1 PASS1 is called by

```
..
CAL     ØRG
TSX     $PASS1,4
..
```

Subroutine PASS1 performs the first pass of an assembly program over the symbolic cards on the input tape, writes them on a pseudo-collation tape, and defines symbols; assuming that the symbolic program is to start at the location specified by the address portion of the accumulator. If errors are found they are noted in the SI. PASS1 uses index register one to contain the complement of the ILC.

PASS2 PASS2 is called by

```
..
CAL     ØRG
TSX     $PASS2,4
..
```

Subroutine PASS2 performs the second pass of an assembly program by reading the symbolic cards appearing on the collation tape. The program is assembled in core storage starting at the location specified by the address portion of the AC, and an assembly listing is prepared on the output tape. PASS2 uses index register one to contain the complement of the ILC. If errors are found they are noted in the SI.

VAREVL subroutine VAREVL is called by

```
..
TSX     $VAREVL,4
PZE     BUFF
..
```

where BUFF is the location of a 14 word buffer containing a symbolic card image. VAREVL will evaluate the variable field starting with the first character of BUFF+2 and continuing to the first blank, comma, or column 73. If any undefined symbols are encountered, SI bit 35 will be turned on.

Input and Output Subroutines

Both PASS1 and PASS2 call several input-output routines to handle tape manipulations. These I/Ø subroutines are

READ1 Read Input Tape, called by

```
..
TSX     $READ1,4
PZE     BUFF
.
.
BUFF BSS    14
..
```

The 80 columns of a symbolic card are read from the input tape into the fourteen word buffer at BUFF. Note that 80 characters do not quite completely fill the buffer; the last 4 positions may contain arbitrary characters.

WCT1 Write Collation Tape, called by

```
..
TSX     $WCT1,4
PZE     BUFF
.
.
BUFF BSS    14
..
```

The fourteen word BCI buffer is written on the intermediate tape.

REWIND Rewind Collation Tape, called by

```
        ..
        TSX     $REWIND,4
        ..
```

The intermediate tape is marked with an end of file and rewound.

READ2 Read collation tape, called by

```
        ..
        TSX     $READ2,4
        PZE     BUFF
        .
        .
BUFF    BSS     14
        ..
```

Fourteen words of the intermediate tape are read into the buffer at BUFF. READ2 checks that the collation tape has been rewound.

PRINT Write on output tape for off-line printing, called by

```
        ..
        TSX     $PRINT,4
        PZE     A,0,n
        ..
```

The n word line image starting in location A is written on the output tape (tape A3). The first character of A (normally blank) is used for carriage control. PRINT counts the lines of output and stops after 300.

Symbol Table Subroutines

For forming and searching a symbol table a subroutine package with entries SYMSTØ and SYMGET is used.

SYMSTØ The sequence

```
        ..
        TSX     $SYMSTØ,4
        ..
```

will cause the BCD characters in the AC to be scanned (blanks removed), right justified, and inserted in a symbol table together with its value, the complement of IR1. If the symbol is blank, it is ignored and no entry is made in the table.

SYMGET The sequence

```
        ..
        TSX     $SYMGET,4
        ..
```

will cause the value of the symbol in the AC (assumed to be scanned and right justified) to be looked up in the symbol table. If the symbol is defined, the value is returned in the AC. If undefined, zero is returned in the AC and SI bit 35 is set on.

Utility Subroutines

CAP also uses a package of utility programs which includes SCAN, CØMMA, SAVE, and UNSAVE.

SCAN SCAN is called by

```
..
TSX     $SCAN,4
..
```

on return, the BCD word in the AC is compressed to the right, with blanks removed and leading positions filled with zeros.

CØMMA Subroutine CØMMA is called by

```
..
TSX     $CØMMA,4
PZE     BUFF
..
```

CØMMA counts the number of commas plus one starting with the first character in BUFF+2 and ending with the first blank or column 73. The count is subtracted from index register one. SAVE and UNSAVE manipulate items in a pushdown list.

SAVE SAVE is called by

```
..
TSX     $SAVE,4
PZE     A,0,n
..
```

the n words in registers, A, A + 1, ..., A + n - 1 are placed at the top of the pushdown list and the other items in the list are pushed down n places. (Note that the pushdown effect is achieved by pointers, not by actually moving all the previous entries in the list down in core memory.)

UNSAVE UNSAVE is called by

```
..
TSX     $UNSAVE,4
PZE     A,0,n
..
```

The top n items in the pushdown list are read into locations A, A + 1, ..., A + n - 1 and the other items in the list are pushed up n places.

The pushdown list has a maximum depth of 500 locations. Any attempt to exceed this depth is ignored and SI bit 15 is set. Attempts to retrieve more items than have been stored are ignored and SI bit 16 is set.

Subroutine INTØP is used to evaluate variable fields of the INT pseudo-op during pass two.

INTØP INTØP is called by

```
TSX     $INTØP,4
PZE     BUFF
```

where BUFF+2 is the address of the first location of the buffer containing the variable field. INTØP scans the variable field and converts each decimal subfield (as delineated by commas) to a binary number; shifts the number obtained into the decrement; and stores it in the next location in the program being assembled, assuming that index register one contains the complement of the ILC. INTØP then increments the ILC and repeats the operation for the next subfield.

Subroutine ENDØP is used at the end of pass one to reserve temporary storage for CØMP pseudo-ops.

ENDØP ENDØP is called by

```
TSX     $ENDØP,4
```

Control returns to the caller after ENDØP changes the C(IR1) by the proper amount and enters the symbol TEM into the symbol table.

ØPTBL The first word in $ØPTBL is a control word containing in its address the location of the first item in the operation table and in its decrement the length of the operation table; the rest of ØPTBL consists of pairs of entries, a right-justified BCD mnemonic paired with the binary machine code for that mnemonic.

Subroutine CØMPØP and the subroutines it calls are described in Chapter 4.

Chapter 4

THE COMPILER OF CØMP PSEUDO-OPERATIONS

In this chapter we will examine in detail the operation of the set of subprograms which compile arithmetic for CØMP pseudo-operations. The material under discussion is of an advanced nature and not essential to an understanding of the CAP assembly program. A beginning reader may skip this chapter, as the material in the sequel will not make reference to the compiler. The reader is assumed to be familiar with an algebraic language such as FØRTRAN, ALGØL, or MAD.

4.1 Why a Compiler?

Compilers exist to free the programmer from worry about coding details while working with algebraic calculations. The compiler can take care of the coding details, and the programmer need only concentrate on setting up the proper equations.

The primary reason for including a compiler in CAP is educational. We shall see the close similarity between the internal processes of assemblers and compilers; some of the mystery as to how compilers work will thereby disappear.

Another reason for including a compiler is to provide a contrast with the macro-operation processors found in many present-day assembly programs. A compiler is an often overlooked alternative and provides a flexibility of expression which the macro-processor cannot obtain.

4.2 What Does a Compiler Do?

The point of the compiler is very simple. If the programmer writes on a card a statement

```
CØMP    Y = ALPHA + BETA
```

the program which results is identical to that which would have resulted if the programmer had instead given the instructions

```
CLA     ALPHA
FAD     BETA
STØ     Y
```

We see, then, that the purpose of the compiler is to generate a program to perform the algebraic computation indicated by the symbols and break characters in the variable field of the CØMP statement.

There are several algorithms available to perform the compilation. In the CAP compiler, a nonrecursive procedure contrasts with the recursive procedure used for evaluating expressions in subroutine VAREVL, discussed in Chapter 3. We will see that the algorithm is a collection of simple, straightforward ideas combined in such a way as to produce a sophisticated result.

4.3 Relation of CØMP to CAP

We recall that when the CAP assembler encounters a CØMP pseudo-operation during pass one, it calls a subroutine named CØMPØP.

CØMPØP and the collection of subroutines which it calls compile the symbolic machine instructions in the CAP language required to carry out the computation called for by the CØMP statement. The compiler writes these symbolic instructions on the collation tape in the same format as CAP language symbolic instructions which the programmer writes and the order in which they are to be performed. The compiler increases the ILC by by the number of instructions compiled, and returns control to subroutine PASS1 to continue the first assembly pass. By writing symbolic cards on the collation tape during pass one, the compiler thereby discharges its responsibility; the symbolic instructions on the collation tape will be assembled by the second assembly pass as would instructions provided by the programmer himself.

4.4 Precedence

The language available to the CØMP programmer allows the use of addition, subtraction, multiplication, and division—with parentheses as grouping characters. Since the programmer will wish to attach an order of precedence to these operations, the compiler must take that order into account when creating the symbolic program. The order of precedence used is the following:

parenthetical expressions

multiplication and division

addition and subtraction

This precedence table corresponds to the table commonly assumed by mathematicians. It states, for example, that in the expression

A + B/C

the division is to be carried out before the addition.

4.5 The Spread Field; CØMPØP

The subroutine called to compile CØMP pseudo-operations is CØMPØP. CØMPØP operates in two passes. In the first pass, it scans the variable field of the CØMP card, ignoring blanks, and separates the symbols and break characters one to a word in a buffer known as the spread field. For example, if the variable field contains

SUM = G1 + G2 + G3/SIX

pass one of CØMPØP would produce a spread field containing in successive locations

SUM
=
G1
+
G2
+
G3
/
SIX

Later scans may now search the spread field for break characters with a simple search loop. Symbols which are longer than six characters are permissible. They will be broken up and stored in successive words in the spread field. Since the comma is not a break character, the sequence of characters ABC,1 will be considered to be a single symbol and stored appropriately. When compiled as the address of an instruction, this symbol could represent a tagged address.

All scans of the spread field will ignore a zero appearing within the spread field. The value of this property will become clear later when we see how the spread field is modified as the expression is compiled. An alternative procedure with similar flexibility is to place successive items of the spread field in a string pointer list.

Having re-expressed the arithmetic statement to be compiled in a form easier to work with, subroutine CØMPØP proceeds with the actual compilation. A scan is made for a parenthetical expression which is in some sense "innermost." That is, it is to contain no parenthetical expressions. The procedure for finding such an "innermost" expression is as follows: Scan the spread field starting at the top for left and right parentheses, leaving markers behind at the left parentheses, and stopping at the first right parenthesis. The last left parenthesis marker and the position of the right parenthesis define an "innermost" parenthetical expression. A subroutine named EXPR is now called, with arguments consisting of the pointers to the left and right ends of the parenthetical expression, and the location of the beginning of the spread field. Subroutine EXPR will compile the symbolic CAP language program necessary to compute the expression within the parentheses and will write this symbolic program on the collation tape. EXPR will then modify the spread field by replacing the left parenthesis, the entire expression within the parentheses, and the right parenthesis with zeros. The last instruction in the symbolic CAP language program generated by EXPR will be an instruction to store the result of the computation in a temporary storage location. The symbolic name of this temporary storage location is inserted directly in the spread field by EXPR in one of the locations formerly occupied by the parenthetical expression. The symbol TEM+nn will always fit into the space vacated by the original expression. This is one of the reasons for choosing to spread out the original expression into a spread field.

At this point, the "innermost" parenthetical expression is compiled. CØMPØP now starts over again, looking for a new "innermost" parenthetical expression in the modified spread field. Since the old expression, along with its parentheses, was replaced by a single symbol in the spread field, CØMPØP can scan for a new "innermost" parenthetical expression exactly as it did before. It is now clear why zero words are ignored within the spread field. Whenever the compiler writes instructions on the collation tape, it replaces the symbols and operators within the spread field leading to the compilation

of these instructions by zeros. Later scans of the spread field ignore the presence of the zero positions, as nothing more is to be compiled from the information that was once contained there.

CØMPØP iterates in the manner described; first locating an innermost parenthetical expression, and then calling upon EXPR to compile the expression. EXPR removes the expression from further consideration by modifying the spread field.

Eventually, CØMPØP will reach a situation in which the spread field contains no parenthetical expressions. Instead, it will contain a simple expression preceded by a symbol and an equal sign. In this case, subroutine EXPR is again called with parameters indicating the beginning and end of the simple expression and with an additional parameter specifying that the program compiled is to leave its result in the AC rather than in temporary storage. EXPR again generates symbolic instructions, writes them on the collation tape, and modifies the spread field by replacing all elements compiled by zeros. Upon returning to CØMPØP the compilation is nearly completed except for storage of the final result. Subroutine CØMPØP then generates the necessary STØ instruction to complete the compilation. Let us follow this procedure through for a moderately complicated expression. Consider the following CØMP pseudo-operation

CØMP Y = ((A+B)*(E-C*DL)+END)*F+L1

Figure 4.1 shows the spread field and instructions compiled in succeeding steps. Figure 4.2 is a flow diagram of CØMPØP.

Step 1. CØMPØP places the variable field in the spread field (Figure 4.1a) and scans for left and right parentheses, starting at the top, ending with the first right parenthesis. (See Figure 4.1b.) It then calls EXPR to compile this "innermost" expression. EXPR will write the instructions indicated as "step one" in Figure 4.1f, on the collation tape and modify the spread field to that shown in Figure 4.1c.

Step 2. CØMPØP scans again for left and right parentheses and calls EXPR to compile the expression found. EXPR writes on the collation tape the instructions indicated as "step two" in Figure 4.1f, and modifies the spread field to that shown in Figure 4.1d.

Step 3. One more scan for parenthetical expressions results in a call to EXPR and compilation of instructions indicated as "step three" in Figure 4.1f. EXPR modifies the spread field to appear as in Figure 4.1e.

Step 4. The scan for parentheses fails this time. CØMPØP calls EXPR to compile the remaining simple expression and specifies that the result of the computation be left in the AC. EXPR compiles the instructions labeled "step four."

Step 5. CØMPØP compiles an STØ instruction with a symbolic address consisting of that variable to the left of the equal sign. The compilation is now complete.

CØMPØP keeps track of parenthetical expressions by means of pointers to positions in the spread field. An alternative procedure is to push successive field items down in a push-down list searching for a right parenthesis. Then, the subroutine compiling the expression can retrieve items back to the last left parenthesis.

Note that we have not yet learned how EXPR compiles the symbolic arithmetic instructions and places them on the collation tape. We are analyzing the compiler from the "outside in" and are still at a stage where the organization of the compiler is the most important thing to be learned. Having established the procedure by which parentheses are handled, we are now ready to begin studying the details of instruction creation.

(a)	(b)	(c)	(d)	(e)		(f)	
Y	Y	Y	Y	Y		CLA	A
=	=	=	=	=	Step 1	FAD	B
(	lp→(	lp→(	lp→(	TEM+4		STØ	TEM
(	lp→(	TEM	TEM	0		LDQ	C
A	A	0	0	0		FMP	DL
+	+	0	0	0		STØ	TEM+1
B	B	0	0	0	Step 2	CLA	E
)	rp→)	0	0	0		FSB	TEM+1
*	*	*	*	0		STØ	TEM+2
(	(	lp→(	TEM+2	0		LDQ	TEM
E	E	E	0	0		FMP	TEM+2
-	-	-	0	0		STØ	TEM+3
C	C	C	0	0	Step 3	CLA	TEM+3
*	*	*	0	0		FAD	END
DL	DL	DL	0	0		STØ	TEM+4
)	)	rp→)	0	0		LDQ	TEM+4
+	+	+	+	0		FMP	F
END	END	END	END	0	Step 4	STØ	TEM+5
)	)	)	rp→)	0		CLA	TEM+5
*	*	*	*	*		FAD	L1
F	F	F	F	F	Step 5	STØ	Y
+	+	+	+	+			
L1	L1	L1	L1	L1			

Figure 4.1. Successive spread fields and resulting compilation for CØMP Y = ((A+B)*(E-C*DL)+END)*F+L1.

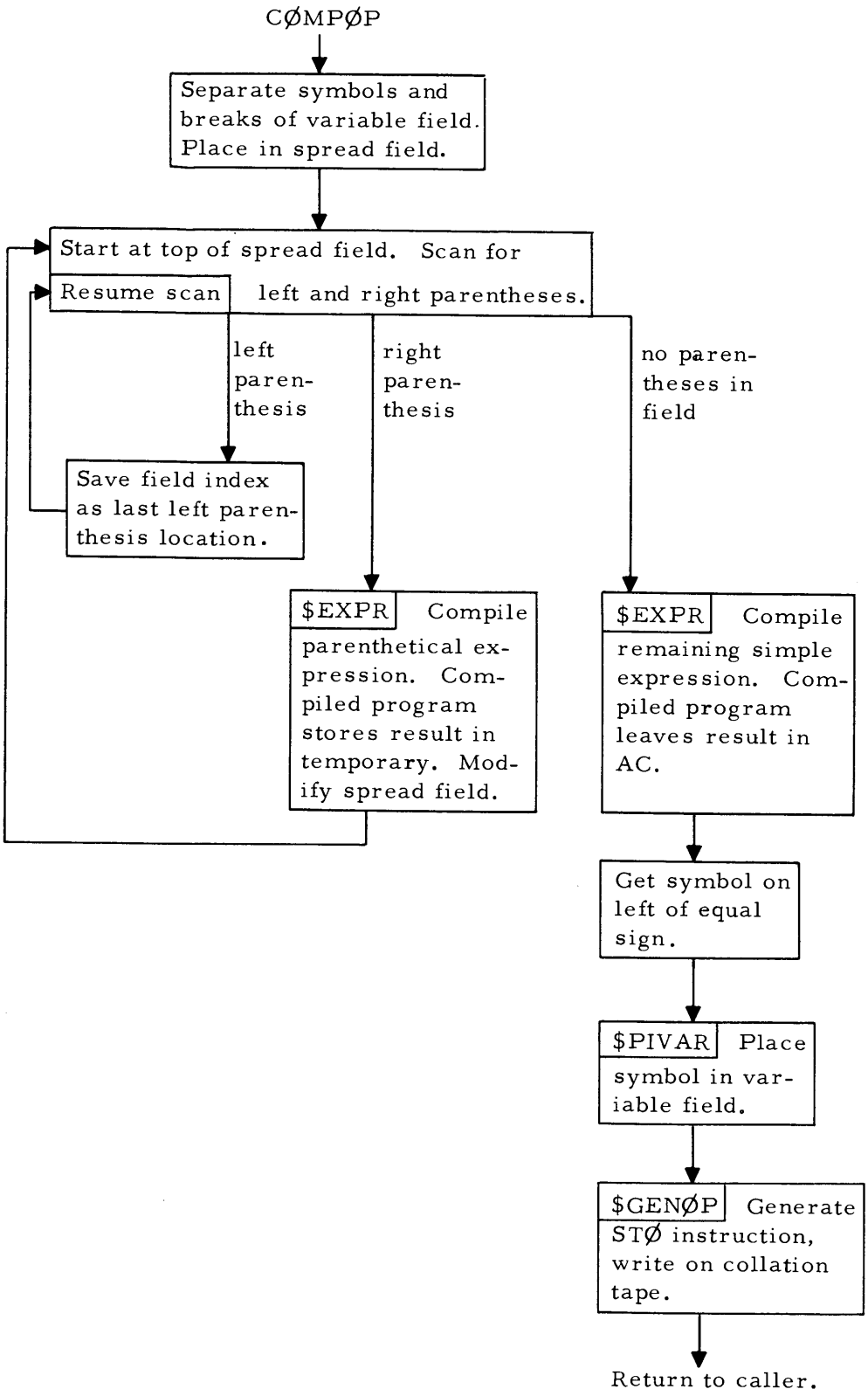


Figure 4.2. Flow diagram of subroutine CØMPØP.

4.6 Compilation of Individual Instructions

In the fifth step in the example above, subroutine CØMPØP had to compile the instruction STØ Y. To write this instruction on the collation tape, a package of subroutines is used which manipulate a collation tape buffer and write on the collation tape. The collation tape buffer is a 14-word buffer which is used to collect a symbolic card image.

The first subroutine in this package is PIVAR. (Place in variable field.) Its calling sequence is

```
TSX   $PIVAR,4
```

PIVAR takes the contents of the AC as a BCD word, and inserts that BCD word in the next available space in the variable field of the collation tape buffer. Columns 13 to 18 are filled in by the first call to PIVAR, columns 19 to 24 on the next, etc.

The last piece of information known about any instruction is always the operation code. Subroutine GENØP inserts the operation code and writes the collation tape buffer on the collation tape. Its calling sequence is

```
      .
      .
TSX   $GENØP,4
BCI   1,  opr
      .
      .
```

where "opr" is the operation mnemonic to be inserted in the operation field. GENØP inserts the instruction code into the operation field (columns 7 to 12) writes the entire collation tape buffer on the collation tape, and clears out the buffer with blanks, resetting PIVAR to store in columns 13 to 18. Thus the sequence required to generate the STØ Y instruction in step five, above, is

```
      .
      .
CAL     FLD,1        GET SYMBØL FRØM SPREAD FIELD.
TSX     $PIVAR,4     INSERT IN VAR FIELD.
TSX     $GENØP,4     GENERATE STØ ØP.
BCI     1,  STØ      ..
      .
      .
```

When it compiles instructions, subroutine EXPR also uses the subroutines PIVAR and GENØP.

4.7 Compilation of Simple Expressions; EXPR

Subroutine EXPR has the responsibility of compiling parentheses-free expressions. This responsibility includes the proper handling of precedence below the level of parenthetical expressions. EXPR handles precedence by making two passes over the symbolic expression; during the first pass, all terms (symbols connected by asterisks and slashes) are compiled leaving the expression in the form of a summation of individual elements (subroutine TERM compiles the terms). In the second pass over the expression, EXPR

compiles the necessary add and subtract instructions to complete the summation. Let us consider a typical spread field expression that EXPR is to compile. The expression comes from Step 2 of the previous example.

```
E
-
C
*
DL
```

In the first pass, EXPR locates terms containing more than one symbol. In the given expression, the second term falls into this category. Therefore, EXPR calls subroutine TERM with parameters pointing to the upper and lower boundaries of the term C*DL. Subroutine TERM compiles a program which computes the value of the term and inserts the answer into temporary storage. In this case the program written on the collation tape is

```
LDQ   C
FMP   DL
STØ   TEM
```

TERM will also modify the spread field by replacing the elements of the term with zeros, and inserting the name of the temporary storage location into the spread field in an appropriate place. When TERM finishes, the spread field will appear as follows:

```
E
-
TEM
0
0
```

Since there are no more terms in our sample expression, pass one of EXPR is complete, and pass two begins. In pass two, EXPR compiles and writes on the collation tape a program to perform the summation of the elements in the expression.

The second pass consists of the following steps, indicated in the flow diagram in Figure 4.3.

1. Scan the spread field from the top, looking for the end of the first symbol. If an initial minus sign is passed, set a switch.
2. Compile the instruction CLA or CLS (on the basis of the switch set in Step 1) with a symbolic address consisting of the symbol obtained in Step 1, using PIVAR and GENØP. Replace the operator and the symbol in the spread field with a zero.
3. Continue scanning the spread field for the end of the next symbol. Again, if an initial minus sign is passed, set a switch.
4. Compile the instruction FAD or FSB (on the basis of the switch set in Step 3) with a symbolic address consisting of the symbol obtained in Step 3, using PIVAR and GENØP. Replace the symbol and the operator in the spread field with a zero.
5. Repeat Steps 3 and 4 until the end of the expression is reached. Now, if requested, compile an instruction to store the result in a temporary location. The second pass is now complete, and the expression has been compiled.

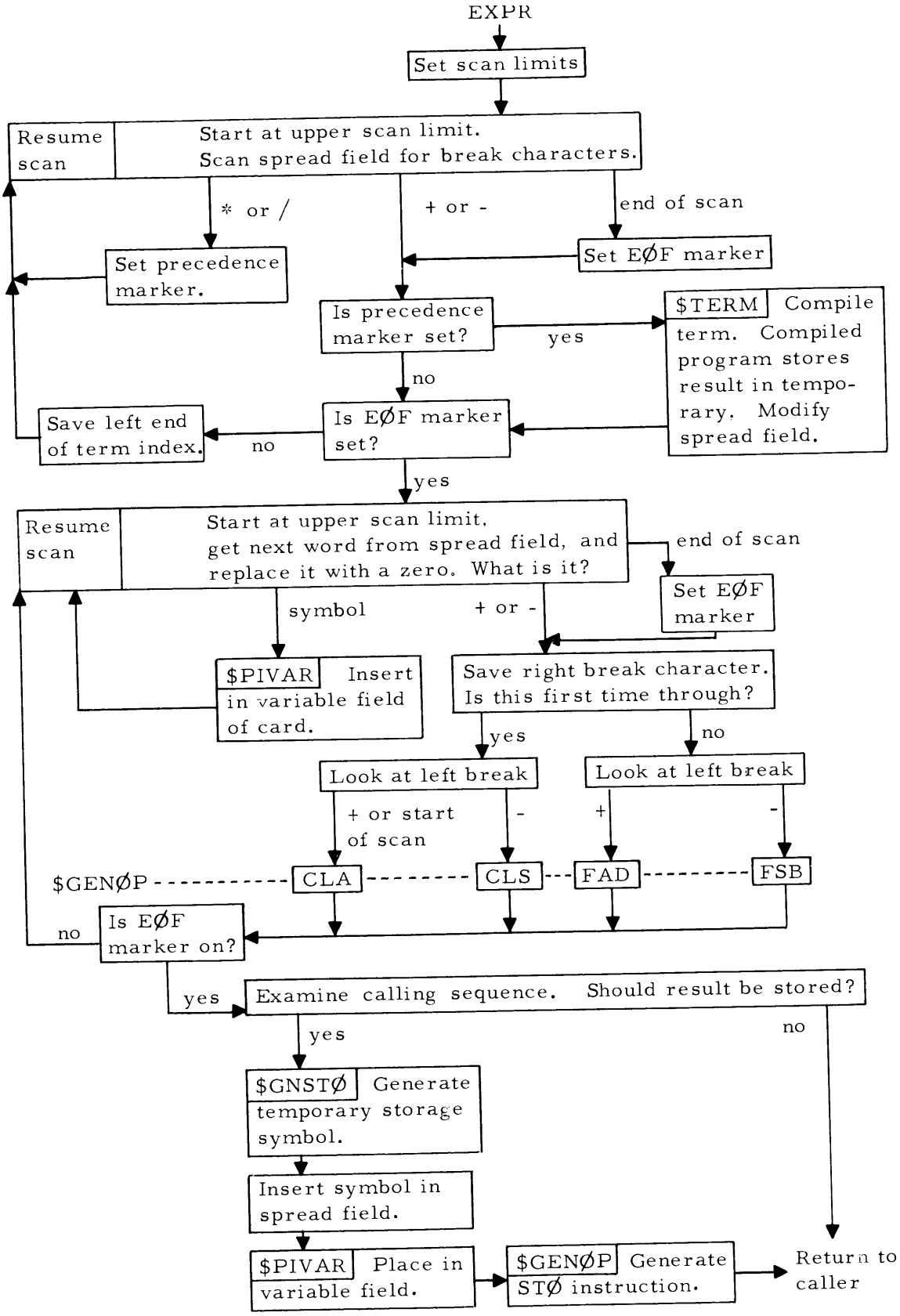


Figure 4.3. Flow diagram of subroutine EXPR.

4.8 Temporary Storage and Subroutine GNSTØ

The last step in subroutine EXPR was compilation of an instruction to store the AC in a temporary location. What symbolic address should be placed in the STØ instruction, and how can temporary storage be reserved? Subroutine GNSTØ provides this service. The calling sequence

```
TSX    $GNSTØ,4
```

will bring into the accumulator the symbol TEM+n where n is one less than the number of times GNSTØ has been called. Subroutine GNSTØ will also keep track of the total number of temporary locations used so that subroutine ENDØP can reserve space at the end of assembly pass one. The first call to GNSTØ brings back the symbol TEM; later calls produce symbols such as TEM+1, etc. The instruction

```
STZ*   $NSTØ
```

resets GNSTØ so that the next call starts again with the symbol TEM. Since separate CØMP statements are independent, they can use the same temporary storage locations, and CØMPØP resets NSTØ at the beginning of each new CØMP statement.

The sequence used by EXPR to compile the store instruction is, then,

```
      .
      .
TSX   $GNSTØ,4     GET TEMPØRARY SYMBØL.
SLW   FLD,1        INSERT IN SPREAD FIELD.
TSX   $PIVAR,4     PLACE IN VARIABLE FIELD.
TSX   $GENØP,4     GENERATE STØ ØP.
BCI   1, STØ
      .
      .
```

4.9 The Compilation of Terms; TERM

When EXPR encounters a term consisting of symbols connected by asterisks and slashes, it calls subroutine TERM to compile instructions which compute the value of the term and leave the result in temporary storage. Subroutine TERM performs this compilation by scanning the term in much the same manner as subroutine VAREVL (see Chapter 3) noting for each symbol the break character on its left and on its right. The break character on the left may be the beginning of the term, an asterisk, or a slash. The one on the right may be the end of the term, an asterisk or a slash. Thus a symbol may have one of nine pairs of break characters associated with it. Since the instructions compiled in each of the nine cases is different, a nine-way branch must be made for each symbol. The flow diagram in Figure 4.4 illustrates this nine-way branch. The scan of the term begins at the left (or top, in terms of the spread field).

Let us consider a simple term, and follow the operation of TERM through the flow diagram. Suppose TERM is to compile the following spread field:

```
C
*
D
*
E
/
F
```

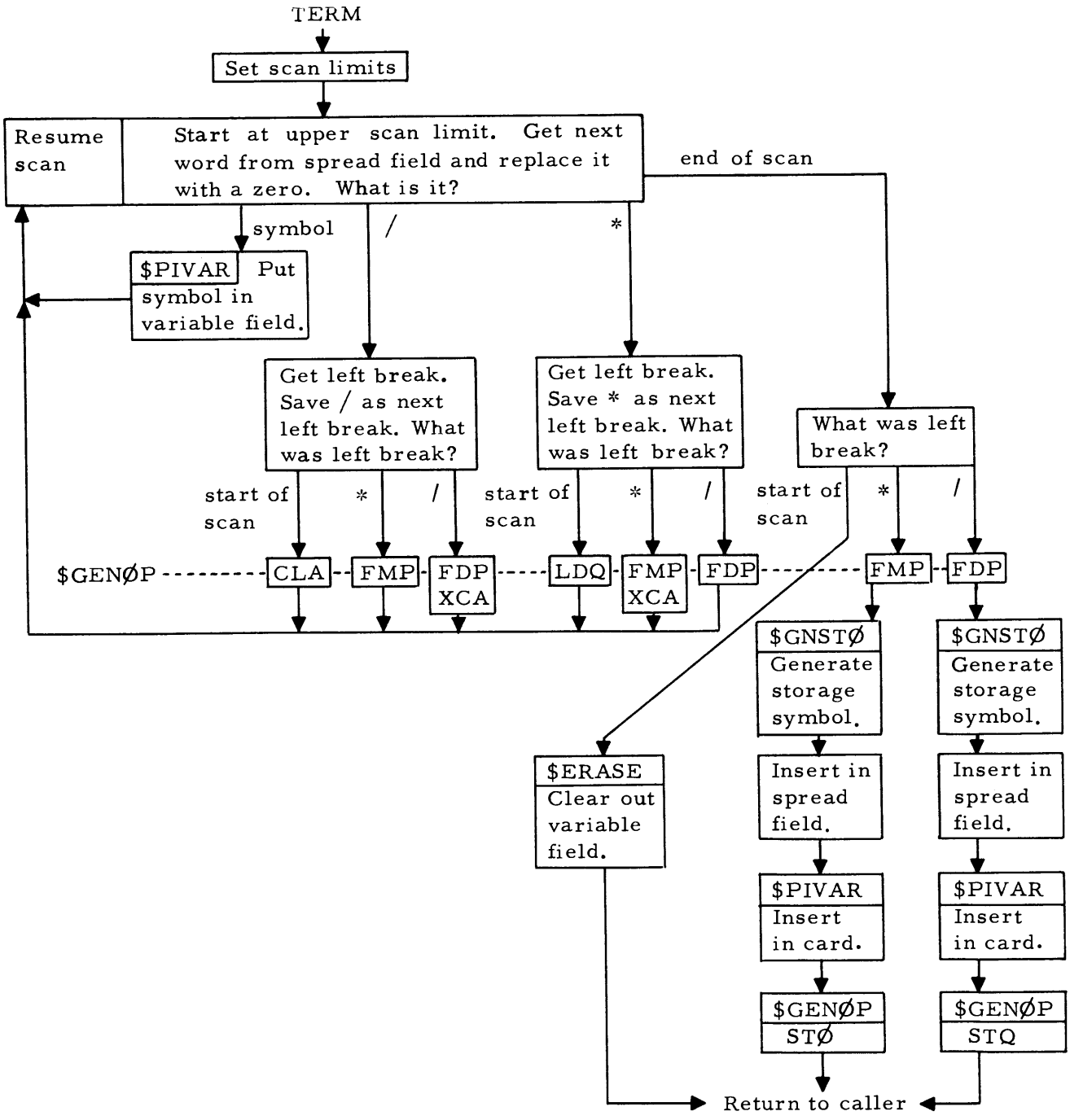


Figure 4.4. Flow diagram of subroutine TERM.

Upon scanning for the first symbol, we find that the left break is the beginning of the term, the right break an asterisk. Following the flow diagram, we see that the instruction LDQ C is compiled in preparation for the multiply operation. We may note that in this case, the compilation leaves the result in the proper register so that the next instruction FMP will operate correctly. If the right break character had been a slash, the instruction CLA C would have been compiled instead. We will see that the algorithm leaves the result in the proper register in all cases.

The scan now resumes. The next symbol has an asterisk on the left and an asterisk on the right. The asterisk on the left signals that we should compile the instruction FMP D; the asterisk on the right warns of a coming multiplication, so the result must be returned to the MQ with an XCA instruction.

Resuming the scan once more, we find that the third symbol has on the left an asterisk, on the right a slash. Again, the asterisk on the left signals that the instruction FMP should be compiled; however, the slash on the right indicates that the next operation will be division. Therefore, the result is left in the AC in proper position for the FDP instruction.

Returning to the scan for the fourth and final time, we find the symbol F surrounded by a slash on the left and the end of the term on the right. The slash calls for a division operation, so the instruction FDP F is compiled. The end-of-term break indicates that we are almost finished. A temporary storage location is generated by GNSTØ and the instruction STQ TEM is compiled. Note that if the last operation had been a multiplication, the last instruction would have been STØ TEM instead.

Now, compilation of the term is finished. Although it has not been mentioned before, the spread field was reset to zero during the scan, and, at the end, symbol TEM was placed back into the spread field. The final result of the compilation by TERM is as follows:

Spread field	Collation tape
TEM	LDQ C
0	FMP D
0	XCA
0	FMP E
0	FDP F
0	STQ TEM

4.10 Review

With the study of subroutine TERM, we have completed our examination of the compiler. A brief review of the essential points covered may help place those points in the proper perspective.

The compiler operates during the first assembly pass of CAP. The compiler places the instructions generated on the collation tape for processing by the second assembly pass just as though the programmer had provided them.

Subroutine CØMPØP coordinates the compilation. CØMPØP goes over the symbolic expression in two passes. During the first pass, it places the symbolic expression in the spread field — one symbol or break character to a memory location.

In the second pass it evaluates the expression from the innermost set of parentheses outward with the help of subroutine EXPR. Subroutine EXPR also operates in two passes. In the first pass, EXPR reduces the expression to a summation by calling on subroutine TERM to compile the instructions to compute the individual terms. The second pass of EXPR compiles the instructions needed to compute the resulting summation.

During all phases of the compilation, the compiler modifies the spread field as it generates instructions and places them on the collation tape. Subroutines GENØP, PIVAR, GNSTØ, and ERASE help put together symbolic instructions and write them on the collation tape.

When the compilation is finished, control returns to CAP to continue assembly pass one.

4.11 Calling Sequence of Compiler Subroutines

This section describes the calling sequences of each of the subroutines of the compiler and presents for easy reference a thumbnail sketch of the external characteristics of each subroutine.

CØMPØP Subroutine CØMPØP is called by

```
TSX     $CØMPØP,4
PZE     BUFF
```

where BUFF is the first location of a 14-word buffer containing the symbolic CØMP card. CØMPØP compiles the instructions necessary to perform the arithmetic specified by the variable field of the card in the buffer, writes these instructions on the collation tape, and increases the value of the ILC (assumed to be stored in complement form in index register one) by the number of instructions compiled.

EXPR Subroutine EXPR is called by

```
TSX     $EXPR,4
PZE     LI, T, RI
PZE     FLD
```

where FLD-LI is the address of the left break and FLD-RI is the address of the right break. EXPR takes a string of symbols connected by + - * or / and compiles the result in floating point. If T = 0, the result is placed in temporary storage. Otherwise, the result is in the AC. The spread field is modified accordingly.

TERM Subroutine TERM is called by

```
TSX     $TERM,4
PZE     LI, 0, RI
PZE     FLD
```

where FLD-LI is the address of the left break, and FLD-RI is the address of the right break. TERM takes a string of symbols connected by * or / and compiles the result in floating point. The compiled program places its result in temporary storage, and TERM modifies the spread field accordingly.

The following subroutines are used to form symbolic instructions:

PIVAR Subroutine PIVAR (place in variable field) is called by

```
TSX     $PIVAR,4
```

PIVAR takes the $C(AC)_{p,\ 1\text{-}35}$ as a BCD word and stores that word in the next available location in the collation tape buffer. On the first call to PIVAR, the next available location is the first word in the variable field position of the buffer.

ERASE Subroutine ERASE is called by

```
TSX     $ERASE,4
```

Subroutine ERASE clears the collation tape buffer, replacing all words with blanks, and resetting PIVAR so that on the next call it will start at the beginning of the variable field.

GENØP Subroutine GENØP is called by

```
TSX     $GENØP,4
BCI     1,  opr
```

where the letters "opr" are the symbolic operation code desired. GENØP will take the symbolic operation code in location 1,4 and insert it into the operation field of the collation tape buffer. It will then write the buffer on the collation tape and call subroutine ERASE to clear the buffer so that it may be used again.

GNSTØ Subroutine GNSTØ is called by

```
TSX     $GNSTØ,4
```

Subroutine GNSTØ returns to the caller after placing in the AC $_{p,1-35}$ a symbol of the form TEM+n where n is one less than the number of times that GNSTØ has been called. Entry point NSTØ will contain this number; and if NSTØ is reset to zero, n will be reset to zero for the next call to GNSTØ. GNSTØ keeps track of the largest n ever encountered and leaves it in a location where it is accessible to subroutine ENDØP for purposes of assigning temporary storage at the end of the first assembly pass of CAP.

Chapter 5

CAP AS A LABORATORY EXERCISE

CAP finds application both in the classroom and in the laboratory. In the laboratory the student modifies or improves the assembler, for example, by adding pseudo-operations to make the CAP language more flexible or by improving the internal operations of the assembler. Appendix C contains a list of suggested modifications.

This chapter is divided into two parts to correspond, roughly, to material of greater interest to the student and to his instructor, respectively. No clear line can be drawn between these interests, of course, as the instructor will wish to read the entire chapter and an advanced student will find much of interest in the second part.

5.1 The CAP Laboratory

The CAP assembly program was written with expansion in mind. Thus, although there might be simpler ways to perform some of the operations called for in the original CAP language, extension of these operations might be difficult if a simpler, less general, approach had been used in the original coding. There are also several examples throughout CAP of points onto which additional coding may be easily attached. An analogy would be the complicated highway interchange with one blocked exit at a point where a new highway is to be built someday.

The suggested modifications represent changes which are at once useful, educational, and not too difficult, when the operation of the original assembler is well understood.

When CAP is used in the laboratory, the main program which calls CAP is replaced by an execution monitor program to aid in debugging the modifications. This execution monitor provides aid in case the modified assembly program gets into a loop or comes to a stop, and it provides a postmortem when the CAP assembly is finished.

Also, in the laboratory, the input-output subroutines are replaced with an I/Ø simulator package to speed up testing; this simulator provides as CAP input a symbolic test program for assembly and simulates the collation tape with a core buffer.

Extent of Laboratory Assignment

A typical laboratory assignment might be the following: The instructor selects a set of modifications totaling in value about 200 "points" as required modifications. (See Appendix C for point values.) The student then selects additional modifications worth about 100 points. The student is permitted eight or nine computer "runs" to attempt to get all 300 points of modifications working correctly.

Evaluation of the student's work is done on the basis of a brief written report describing the modifications attempted and the degree of success in achieving modification. Printed computer output should accompany the report as evidence of correct operation of the modified assembly program.

How CAP Is Modified

Two different procedures have been used to allow the modification of CAP. In the first and simpler procedure, the student makes a copy of the symbolic decks of all the subroutines basic to the assembler and, if desired, the compiler. He then makes changes to this deck of 1000 to 2000 cards and submits it for assembly by FAP and testing under the execution monitor.

If this procedure is used, the reader may wish to skip the next sections and proceed immediately with the discussion of testing of the modified assembly program (Section 5.3).

5.2 UPDATE

If a large class uses CAP as a laboratory exercise, the above procedure can lead to the processing of a very large number of cards. An alternate procedure involving the UPDATE feature of FAP can significantly cut down on the number of cards used. Under this procedure, the unmodified CAP subprograms are placed in symbolic form on a single UPDATE input tape for all students, and each student need only submit cards corresponding to the changes he wishes to make in the subprograms. The UPDATE pseudo-operations of the FAP language control the merging of the student's changes with the original symbolic programs and the assembly of the merged programs.

The UPDATE procedure has the disadvantage that the student must learn the UPDATE language in order to modify CAP. However, the advantages of a small input deck are significant both in time saved preparing input tapes for the computer and in added reliability of a smaller deck of cards.

All features of the UPDATE language necessary for the successful modification of CAP will be discussed here. The FAP Reference Manual contains additional information.*

The Use of UPDATE

Images of the cards submitted for a run are written ahead of run time on the System Input Tape by off-line card-to-tape equipment. When programs are assembled normally on the 7090 (without UPDATE), FAP reads the card images from the System Input Tape and processes them one at a time. When UPDATE is used, two more tapes are involved: the UPDATE Input Tape and the UPDATE Output Tape. In CAP, only the UPDATE Input Tape is used.

The UPDATE Input Tape contains the unaltered symbolic versions of the CAP subroutines as shown in the listings in Appendix A. The serialization in columns 73 to 80 on the lists is also on the UPDATE Input Tape and is used by FAP to determine the order of processing card images from the System Input Tape and the UPDATE Input Tape.

Because the UPDATE facility is a part of FAP the first card of any deck submitted using UPDATE must be

```
*  FAP
```

This card causes control to be transferred to FAP. FAP retains control until an END card is processed. It is important to keep in mind that the program assembled begins at the * FAP card on the System Input Tape and terminates with the first END card processed; this END card may be on either the System Input Tape or the UPDATE Input Tape. Assembly of another subprogram requires another * FAP card.

---

* Reference Manual, FØRTRAN Assembly Program (FAP), IBM Publication C28-6235 (September, 1962).

The use of four UPDATE pseudo-operations (UPDATE, DELETE, DELETE THRU, and SKIPTØ) will be described. UPDATE operations are FAP pseudo-operations and, as such, begin in column 8 of the card.

The UPDATE Pseudo-Operation

The UPDATE pseudo-operation specifies the use of the UPDATE feature of FAP. A card with UPDATE punched in the operation field follows the * FAP card. The variable field, beginning in column 16, specifies the details of the UPDATE run. The first subfield contains the logical tape number of the tape unit on which the UPDATE Input Tape has been mounted. In the following examples we will assume that the UPDATE Input Tape is mounted on logical tape drive 11. The other subfields of the UPDATE card specify features not used in CAP and should be left blank. Hence the first two cards in each CAP UPDATE assembly are

```
*  FAP
   UPDATE  11
```

Adding and Replacing Cards

Assembly, initiated by the * FAP and UPDATE cards, continues as card images of FAP instructions are read from the normal System Input Tape and the UPDATE Input Tape one at a time in serial order. A serialized card image on the System Input Tape is assembled before a card of equal or higher serialization but after a card of lower serialization on the UPDATE Input Tape. Whenever FAP encounters card images of equal serialization on the two tapes, the card image on the System Input Tape is assembled in place of the card image on the UPDATE Input Tape. If there is no serialization on the card image on the System Input Tape, the card image is immediately assembled. (See Figure 5.1, a flow diagram of UPDATE.)

More than nine cards can be inserted between two consecutive cards already on the UPDATE Input Tape by giving the first card to be inserted a serial number between the two cards on the UPDATE Input Tape. The remaining cards to be inserted at this point in the subprogram are not serialized.

Changes can be made in increasing order of serialization only.

Deleting Cards from Programs on the UPDATE Input Tape

To remove a card from a program on the UPDATE Input Tape, the DELETE pseudo-operation is used. When FAP reads a card from the System Input Tape that has DELETE in its operation field, cards are assembled from the UPDATE Input Tape until a card image with serialization equal to that of the DELETE card is found. FAP does not assemble this card image from the UPDATE Input Tape; normal updating and assembly continue with the next card from each tape.

If many consecutive cards are to be deleted from programs on the UPDATE Input Tape, the DELETE THRU pseudo-operation may be used. When FAP reads a card that has DELETE in its operation field and the letters THRU in the variable field, no more card images from the UPDATE Input Tape are assembled until a card of serialization higher than that of the DELETE THRU card is found on the UPDATE Input Tape.* FAP will then resume normal updating and assembly.

---

* As of May, 1962, the M. I. T. version of FAP requires THRU in columns 15 to 18; this differs from the FAP Reference Manual description of DELETE THRU.

To delete a block of cards from the middle of a program: First, insert a DELETE card with serialization of the first card in the block. This DELETE card should be followed by a DELETE THRU with serialization equal to the serial number of the last card to be deleted. DELETE THRU will delete a card of equal but not higher serialization. The input tapes should never be moved backward while updating a program.

The Necessary END Card

To insure proper operation of UPDATE, the last card of the input deck for each subprogram updated must be a serialized END card. The serialization of the END card in the input deck must be identical to that of the END card on the UPDATE Input Tape for the subprogram being updated.

Bypassing Assembly of Subprograms

The UPDATE Input Tape will be rewound before the job starts and we may assume that it is properly positioned to begin assembly of the first subprogram on the tape. The order of subprograms on the UPDATE Input Tape is specified in Figure 5.2. The order is the same as on the CAP listings.

The first

```
*  FAP
   UPDATE  11
```

would therefore, start assembly of subprogram CAP. At the end of this assembly the UPDATE Input Tape would be positioned ready to start assembly of the second subprogram. The next

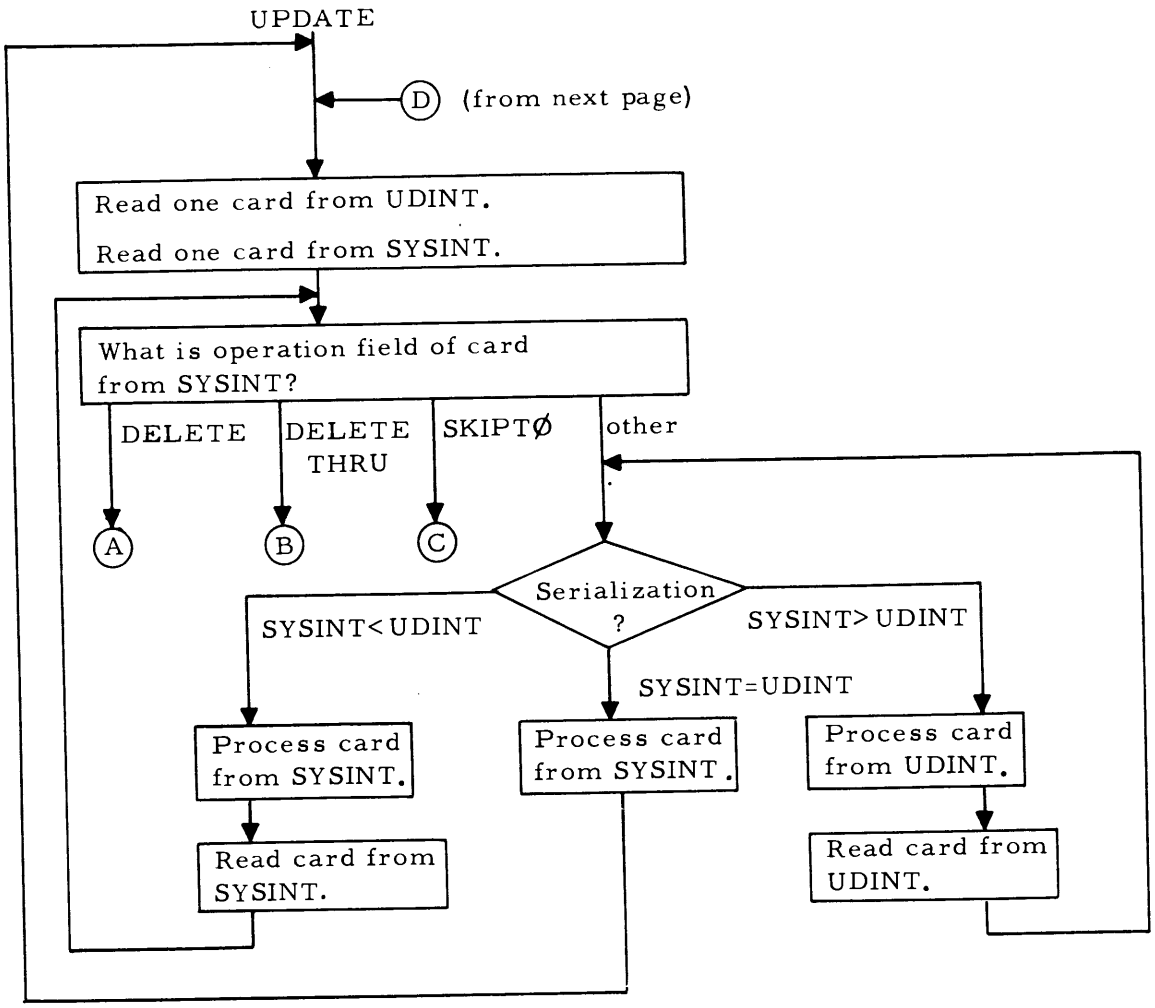
```
*  FAP
   UPDATE  11
```

would start assembly of PASS1, and so forth.

Most of the suggested alterations to CAP require changes to only a few of the subprograms. Therefore, it would be wasteful of machine time to assemble all of the CAP subprograms during each run. Assembly of subprograms not being modified on the UPDATE Input Tape may be omitted by proper use of the SKIPTØ pseudo-operations.

When FAP reads a card image from the System Input Tape with SKIPTØ in its operation field, assembly is suspended and the UPDATE Input Tape is read until a card image of serialization identical to the serialization of the SKIPTØ card is found. Normal updating and assembly commence with the card of identical serialization on the UPDATE Input Tape. A card of serialization higher than that of the SKIPTØ card will not terminate the SKIPTØ operation; the serializations must be identical. Thus, assembly of a subprogram can be avoided by using a SKIPTØ card serialized with the serial number of the first card in the next subprogram to be updated. Subprograms must be updated and assembled in the order that they appear on the UPDATE Input Tape; SKIPTØ cannot be used to move the UPDATE Input Tape backward.

It is good practice to include a SKIPTØ card in the input deck for every subprogram to be updated. If the UPDATE Input Tape is positioned ready to read the card specified by the SKIPTØ card, FAP will begin assembly with that card. Inclusion of the SKIPTØ cards in all input decks makes each subprogram independent of all others. The input cards for a particular subprogram may be removed from the complete input deck without



SYSINT - System Input Tape

UDINT - UPDATE Input Tape

SYSINT=UDINT - Serialization on card from SYSINT equals serialization on card from UDINT.

SYSINT<UDINT - Serialization on card from SYSINT is less than serialization on card from UDINT.

SYSINT>UDINT - Serialization on card from SYSINT is greater than serialization on card from UDINT.

Figure 5.1a. UPDATE flow diagram.

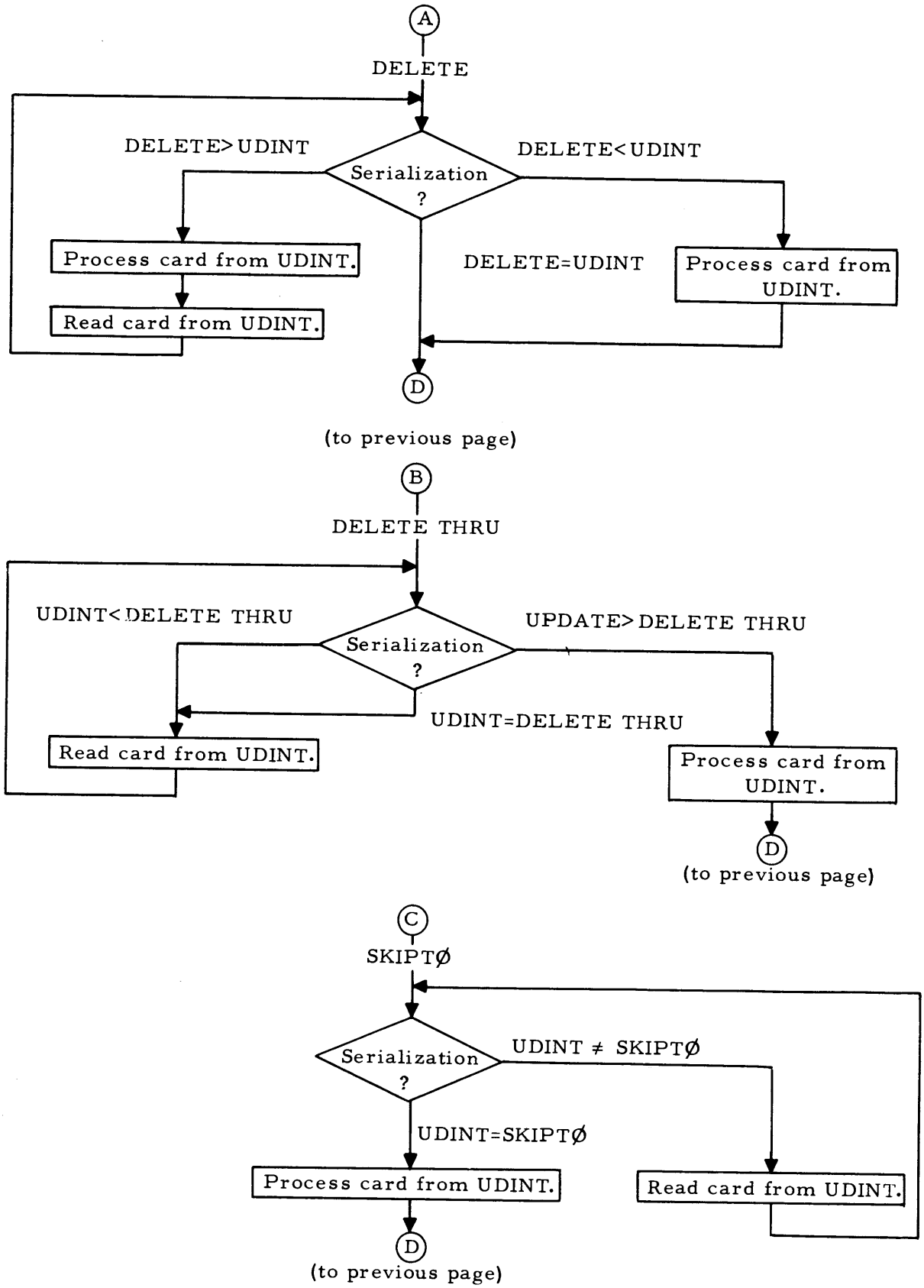


Figure 5.1b. UPDATE flow diagram.

Subprogram	Serialization of first card	Serialization of END card
CAP	CAP00010	CAP00320
PASS1	PAS10010	PAS10790
PASS2	PAS20010	PAS22210
VAREVL	VEVL0010	VEVL2220
ØPTBL	ØPTB0010	ØPTB0790
INTØP	INTP0010	INTP0990
UTILITIES	UTIL0010	UTIL1140
SYMSTØ	SYMS0010	SYMS0540
ENDØP	ENDP0010	ENDP1020
CØMPØP	CØMP0010	CØMP1860
EXPR	EXPR0010	EXPR1710
TERM	TERM0010	TERM1350

Figure 5.2. Order of subprograms on CAP UPDATE Input Tape.

the need to add a SKIPTØ card in the deck for the following subprogram. The first three cards for each subprogram to be updated should be

```
*  FAP
   UPDATE   11
   SKIPTØ
```

Column 73 ↓

SUBR0010

(Serial number of the first card in the subprogram to be updated.)

Remember that the UPDATE Input Tape contains the unaltered, symbolic version of the CAP subprograms as contained in the listing in Appendix A. When we submit a deck to update a CAP subprogram, it is the combination of that symbolic input deck and the unaltered symbolic program on the UPDATE Input Tape that is assembled. When new changes are made to a subprogram all previous desired changes to that subprogram must be included in the input deck.

5.3 How CAP Is Tested

If the modified version of CAP assembles successfully, it may be tested on the same computer run. To simplify this testing a special library tape is used with the FØRTRAN Monitor System. This library tape contains the execution monitor program and all of the subroutines of the CAP assembler in an unmodified, binary form. The student need only

assemble those subprograms of CAP for which changes are desired, and the library will provide the rest of the subroutines needed to complete CAP. The student must also provide a main program which calls the execution monitor program.

Once a subprogram has been modified, assembled, and checked out, it may be submitted on later runs in binary form; it need not be reassembled if no changes are to be made to it.

Let us suppose that a student has made a change to one subprogram, VAREVL, in his attempt to add division to the variable field operations. If he submits an assembly and a main program as an FMS job, the following steps will be carried out:

1. The FAP assembly will take place.

2. If the assembly is successful, the main program and the program just assembled, VAREVL, will be loaded into core memory.

3. The library will be searched for the rest of the CAP assembler and the execution monitor, and they will be loaded into core memory.

4. The CAP assembler, as modified, is then run under the execution monitor program. The input-output simulator will provide a symbolic test program for CAP to assemble. A typical symbolic program used to test CAP is shown in Appendix B.

5. When CAP finishes its assembly of the test program (or gets into a loop or stops because of the modifications), control of the computer returns to the execution monitor which prints out for debugging and comparison purposes, the following:

   a. The symbolic test program CAP worked on.

   b. The collation tape, if anything was written on it by subroutine WCT1. The collation tape is printed out in BCD.

   c. An octal postmortem of all programs which were submitted (in this case, only VAREVL and the main program).

   d. An octal postmortem of the region in core storage in which CAP was to have placed the assembled program.

In the case of the VAREVL test, it will be noted that the symbolic test program in Appendix B has in it several variable field division signs. Examination of the addresses assembled for these instructions will tell whether or not the modification worked correctly.

In case of difficulty, such as a program stop or loop, the collation tape dump is often most helpful if the stop occurred in pass one, since the tape will contain the last instruction processed correctly. Similarly, pass two loops or stops may be diagnosed by observing which instruction was the last processed and printed on the CAP assembly listing. For example, if the first instruction which does not appear on the CAP output listing is the first instruction in which division appears in the variable field, one might suspect the new VAREVL modification.

In connection with item five, listed earlier in this section, the execution monitor assumes CAP to be in an endless loop if it takes longer than five seconds to complete its assembly. The postmortem indicates the instruction location where the program was stopped. Adding one to this location will give the instruction which was next to be executed. A normal CAP assembly takes about one second on the IBM 7090 and the most complicated interaction of modifications should not extend this time by more than three seconds.

A typical CAP execution run is shown in Appendix B following assembly listings of the execution monitor subprograms. The format of the CAP assembly output and of the postmortem outputs can be seen there.

5.4 Tactics for Modifying CAP

Experience has shown that the following tactics can be helpful in making maximum use of the limited number of computer runs available for debugging modifications to CAP.

1. Some modifications are closely related to others; making the first modification allows the second to follow with but a few instructions.

2. All anticipated modifications should be submitted before the fourth or fifth run (if eight runs are available) to allow sufficient time for debugging.

3. Leave the addition of pseudo-operations which change the ILC (such as BSS) until later runs; debugging the simpler modifications in early runs. (If one of these pseudo-operations fails, the result is usually catastrophic.)

4. Observe that the point values attached to modifications are an indication of their relative difficulty. In particular, modifications to the compiler require an understanding of advanced material in Chapter 4 and should be avoided by the beginner.

5.5 The Instructor's Point of View

The material discussed in this section is of an advanced nature and may be skipped by the reader not interested in teaching CAP to a class.

The Execution Monitor

The execution monitor is a package of library subroutines called by a main program. The calling sequence to this monitor is

```
TSX     $TESTS,4
```

The main program listed in Appendix B, which contains the above instruction, may be assembled and given to the student in binary form for submission along with his modifications. The main program also contains three words of octal 7's which prevent the student from duplicating the binary cards on an IBM 026 keypunch. Without the octal 7's the 026 may duplicate the cards incorrectly but the 7's prevent all duplication, and thus they insure against the possibility of an incorrect binary main program. Note also that the execution monitor does not return to the main program which called it, it exits to the FØRTRAN Monitor System when finished testing CAP.

The execution monitor first prints a subprogram storage map of all binary and symbolic programs submitted by the student. This is done by reference to subroutine MØVIE) inserted at the time of loading by the BSS loader.* The storage map lists all subprograms found in MØVIE) from the beginning of core storage up to the subroutine TESTS, which is the first subprogram loaded from the library.

Depending on the status of sense switch one, either a core storage clock or a magnetic tape on channel B of the 7090 in combination with a data channel trap is used as a five-second timer. In the latter case, a scratch tape (tape B3 as the program is shown in Appendix B) is write selected and a sequence of data channel commands with a word count of 50000 and terminating with an IØCT command is given to channel B.

---

* Subroutine MØVIE) is a copy of the BSS loader table which has been moved to a position following the last subprogram loaded and given an entry point name by the BSS loader before beginning execution. This loader table consists of entry name and entry point pairs and permits a selective storage map and postmortem to be given.

Since the word transmission rate of a 729 mod IV magnetic tape is about 10,000 words per second, the data channel trap will occur in about 5 seconds if CAP has not completed its assembly and returned to the monitor by that time. This trap will restart the computer if it is at a program stop.

Other trap returns are also set by the execution monitor. A standard floating point trap interpreter is provided which changes underflow to zero and terminates the run on overflow. The select trap return is set up and the select trap enabled before calling CAP.

After these traps have been enabled, the execution monitor places in the AC the origin of the symbolic program that CAP is to assemble (50000_8) and calls CAP.

An I/Ø simulator package handles all calls for input and output from CAP. The input tape is simulated by a core storage buffer containing strings of card images. Subprogram PRØG is used as a buffer to hold these strings. The collation tape is also simulated using a core buffer.

Control eventually returns to the execution monitor; it returns either via the expected return from CAP, or via timer or select traps. The execution monitor prints an appropriate comment and gives a postmortem of relevant information. It then returns to the FØRTRAN Monitor System with a standard system load sequence.

Miscellaneous Details About the Laboratory

If a student has made a modification which is not tested in the symbolic test program contained in subprogram PRØG, a special input/output package is used which reads card images from the System Input Tape after the student's * DATA card. All other I/Ø operations are handled in exactly the same way as in the usual I/Ø simulator package.

Each student must have the UPDATE Input Tape rewound at the beginning of his job. This rewind may be accomplished in one of several ways; perhaps the simplest is the temporary modification of the FØRTRAN Monitor System to rewind the tape between jobs.* An alternative might be to require that each student use the REWIND pseudo-operation in his first FAP assembly.

Making an UPDATE Input Tape

The UPDATE Input Tape used for CAP may be made with the aid of the FAP UPDATE facility. In the following discussion, since the tape is being written, it will be referred to as an UPDATE Output Tape. When making an UPDATE tape from a card deck, only an output tape is specified on the UPDATE card. For example, if the tape being written is on logical drive 11, the FAP control card would be

```
UPDATE    ,11,,D
```

The D in the fourth subfield specifies that assembly is deleted, permitting the entire tape, including all subroutines, to be written with only one loading of FAP.

Since the third subfield is void, the output tape will be in blocked format. This blocked format is preferable to unblocked, as less time will be required to move the UPDATE tape when it is used later by a class. (FAP writes blocked records 16 cards to a block.)

Since assembly is deleted by the fourth subfield, regular END cards (in the subroutines being placed on the UPDATE Output Tape) will not stop FAP: the pseudo-operation ENDUP will. Following the last subprogram being placed on the UPDATE Output Tape, the UPDATE pseudo-operations ENDFIL and REWIND may be used to complete the tape.

If a student should attempt to SKIPTØ a serial number not on the UPDATE tape, FAP will stop with a comment and print the last card on the UPDATE tape. For this reason, a card with a distinctive comment such as "SKIPTØ ERRØR" may be inserted after the last subprogram written on the UPDATE tape.

---

* J. H. Saltzer, M. I. T. Computation Center Memo CC-204 (February, 1963).

Appendix A

LISTING OF THE CLASSROOM ASSEMBLY PROGRAM

This appendix consists of FAP listings of the complete Classroom Assembly Program. At the end of these listings is an assembly output produced by CAP, of a sample CAP language program. Certain conventions have been observed in these listings. The double asterisk (**) has been used as a zero element in the variable field of those instructions subject to program modification. Each subroutine begins with the pseudo-operation PCC to insure that all cards in the original subprogram appear on the listing. Since the listings are to be used as references for UPDATE modifications, the position of all control cards must be known.

Index to Appendix A

```
      MAIN PROGRAM FOR CAP.

                                PCC                                                    MAIN0010
                                COUNT   26                                             MAIN0020
                                LBL     MAIN          BINARY CARD LABEL.               MAIN0030

  TRANSFER VECTOR
00000   336225636447      .SETUP
00001   475131456360      PRINT
00002   232147606060      CAP
00003   256731636060      EXIT

00004   0074 00 4 00000           CALL    .SETUP        SETUP LIBRARY TIMER AND DUMP RETURNS.  MAIN0050
00005   1 00000 0 00007
00006   0 00004 0 00000
00007   0074 00 4 00001           TSX     $PRINT,4      PRINT 'BEGIN ASSEMBLY' COMMENT.        MAIN0060
00010   0 00007 0 00031           PZE     BEG,0,7       ..                                     MAIN0070
00011  -0500 00 0 00030           CAL     ORG           SET ORIGIN.                            MAIN0080
00012   0074 00 4 00002           TSX     $CAP,4        GO TO CAP.                             MAIN0090
00013  -0130 00 0 00000           XCL                   GET ENTRY POINT.                       MAIN0100
00014  -0763 00 0 00025           LGL     21            CONVERT ENTRY TO OCTAL-BCI.            MAIN0110
00015  -0754 00 0 00000           ZAC                   ..                                     MAIN0120
00016   0774 00 4 00005           AXT     5,4           ..                                     MAIN0130
00017   0767 00 0 00003           ALS     3             ..                                     MAIN0140
00020  -0763 00 0 00003           LGL     3             ..                                     MAIN0150
00021   2 00001 4 00017           TIX     *-2,4,1       ..                                     MAIN0160
00022  -0602 00 0 00047           ORS     RET+7         OR TO COMMENT.                         MAIN0170
00023   0074 00 4 00001           TSX     $PRINT,4      PRINT 'RETURN FROM CAP' COMMENT.       MAIN0180
00024   0 00011 0 00040           PZE     RET,0,9       ..                                     MAIN0190
00025   0 07400 4 00003           CALL    EXIT          RETURN TO FORTRAN MONITOR SYSTEM.      MAIN0200
00026   1 00000 0 00030
00027   0 00201 0 00000

                                                                                         MAIN0210
00030 +000000050000       ORG   OCT     50000         CAP PROGRAM ORIGIN.              MAIN0220
                                                                                         MAIN0230
00031   016060606060      BEG   BCI     7,1           TEST OF CAP, BEGIN ASSEMBLY.     MAIN0240
00032   606060606060
00033   632562636046
00034   266023214773
00035   602225273145
00036   602162622544
00037   224370336060
00040   006060606060      RET   BCI     9,0     RETURN FROM CAP, ENTRY POINT IS 00000.   MAIN0250
00041   606060606051
00042   256364514560
00043   265146446023
00044   214773602545
00045   635170604746
00046   314563603162
00047   600000000000
00050   336060606060

                                END                                                      MAIN0260
```

```
          MAIN PROGRAM FOR CAP.
                    POST PROCESSOR ASSEMBLY DATA

       51 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
        31     BEG     10
         2     CAP     12
        30     ORG     11
        40     RET     22,    24
         3    EXIT     25
         1   PRINT      7,    23
         0  .SETUP      4


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..     000003 IN HUNDREDTHS OF MINUTES.
```

```
                SUBROUTINE CAP, CLASS ASSEMBLY PROGRAM.

                                PCC                                                     CAP00010
                                COUNT   32                                              CAP00020
                                LBL     CAP             BINARY CARD LABEL.              CAP00040
                    00005       ENTRY   CAP             START CLASS ASSEMBLY PROGRAM.   CAP00050

  TRANSFER VECTOR
00000  472162620160      PASS1
00001  472162620260      PASS2
00002  475131456360      PRINT

  LINKAGE DIRECTOR
00003  000000000000
00004  232147606060

00005  0634 00 4 00020   CAP     SXA     RX4,4           SAVE IR4.                       CAP00060
00006  0601 00 0 00025           STO     ORG             SAVE ORIGIN.                    CAP00070
00007  0441 00 0 00034           LDI     =0              CLEAR ERROR FLAGS.              CAP00080
                                                                                        CAP00090
00010  0074 00 4 00000           TSX     $PASS1,4        GO TO PASS1.                    CAP00100
                                                                                        CAP00110
00011 -0054 00  000001           LFT     1               TEST FOR SYMBOL TABLE OVERFLOW. CAP00120
00012  0020 00 0 00022           TRA     STFL            YES, GIVE DIAGNOSTIC.           CAP00130
00013  0604 00 0 00026   STOK    STI     SIND            SAVE PASS1 FLAGS.               CAP00140
00014  0441 00 0 00034           LDI     =0              CLEAR INDICATORS FOR PASS2.     CAP00150
00015  0500 00 0 00025           CLA     ORG             GET ORIGIN.                     CAP00160
                                                                                        CAP00170
00016  0074 00 4 00001           TSX     $PASS2,4        GO TO PASS2.                    CAP00180
                                                                                        CAP00190
00017  0442 00 0 00026           OSI     SIND            FORM COMPLETE ERROR FLAGS.      CAP00200
00020  0774 00 4 00000   RX4     AXT     **,4            RESTORE IR4.                    CAP00210
00021  0020 00 4 00001           TRA     1,4             RETURN.                         CAP00220
                                                                                        CAP00230
00022  0074 00 4 00002   STFL    TSX     $PRINT,4        COMMENT.                        CAP00240
00023  0 00005 0 00027           PZE     WSTFL,0,5       ..                              CAP00250
00024  0020 00 0 00013           TRA     STOK            RETURN FOR PASS2 ANYWAY.        CAP00260
                                                                                        CAP00270
00025  0 00000 0 00000   ORG     PZE                     STORAGE FOR PROGRAM ORIGIN.     CAP00280
00026  0 00000 0 00000   SIND    PZE                     STORAGE FOR SENSE INDICATORS.   CAP00290
00027  006270442246      WSTFL   BCI     5,0SYMBOL TABLE SIZE EXCEEDED.                  CAP00300
00030  436063212243
00031  256062317125
00032  602567232525
00033  242524336060
                                                                                        CAP00310
                                END                                                     CAP00320

  LITERALS
00034  000000000000
```

```
                SUBROUTINE CAP, CLASS ASSEMBLY PROGRAM.
                   POST PROCESSOR ASSEMBLY DATA

        35 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
         5     CAP
        25     ORG      6,   15
        20     RX4      5
        26     SIND     13,  17
        22     STFL     12
        13     STCK     24
         0    PASS1     10
         1    PASS2     16
         2    PRINT     22
        27    WSTFL     23


NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..     000003 IN HUNDREDTHS OF MINUTES.
```

```
                PASS 1 OF CLASS ASSEMBLY PROGRAM.

                                                                                PAS10010
                        PCC                                                     PAS10020
                        COUNT   79                                              PAS10040
                        LBL     PASS1           BINARY CARD LABEL.              PAS10050
                00012   ENTRY   PASS1           FIRST ASSEMBLY PASS OF CAP.

  TRANSFER VECTOR
00000  662363016060     WCT1
00001  512521240160     READ1
00002  622321456060     SCAN
00003  627044626346     SYMSTO
00004  234644442160     COMMA
00005  234644474647     COMPOP
00006  254524464760     ENDOP
00007  512566314524     REWIND

  LINKAGE DIRECTOR
00010  000000000000
00011  472162620160

                                                                                PAS10060
00012  0634 00 4 00076  PASS1 SXA     RX4,4           SAVE IR4.                PAS10070
00013  0737 00 1 00000         PAC     0,1             IR1 IS -(ILC).           PAS10080
00014  0020 00 0 00017         TRA     RCD             SKIP TAPE WRITING FOR FIRST CARD.
                                                                                PAS10090
                                                                                PAS10100
00015  0074 00 4 00000  NEXT   TSX     $WCT1,4         WRITE CARD ON COLLATION TAPE.
00016  0 00000 0 00100          PZE     BUFF            ..                       PAS10110
                                                                                PAS10120
                                                                                PAS10130
00017  0074 00 4 00001  RCD    TSX     $READ1,4        READ IN NEXT CARD.       PAS10140
00020  0 00000 0 00100          PZE     BUFF            ..                       PAS10150
                                                                                PAS10160
00021 -0500 00 0 00101  CFLD   CAL     BUFF+1          GET OP-FIELD.            PAS10170
00022  0074 00 4 00002          TSX     $SCAN,4         COMPRESS TO RIGHT.       PAS10180
                                                                                PAS10190
00023  0774 00 4 00012          AXT     NPTBL,4         CHECK FOR PSEUDO-OP.     PAS10200
00024 -0340 00 4 00045          LAS     PTBL+NPTBL,4    COMPARE WITH TABLE.      PAS10210
00025  0020 00 0 00027          TRA     *+2             NO, SKIP.                PAS10220
00026  0020 60 4 00046          TRA*    PTBL+NPTBL+1,4  YES, EXIT.               PAS10230
00027  2 00002 4 00024          TIX     *-3,4,2         NO, INDEX AND TRY AGAIN. PAS10240
                                                                                PAS10250
00030 -0500 00 0 00100  OP     CAL     BUFF            NOT A PSEUDO-OP, ASSUME OP,
00031  0074 00 4 00003          TSX     $SYMSTO,4       GET SYMBOL, AND SAVE.    PAS10260
00032  1 77777 1 00015          TXI     NEXT,1,-1       ADVANCE ILC AND RETURN.  PAS10270

                                                                                PAS10280
                        SPACE   2                                               PAS10290
                        PSEUDO-OP TABLE AND TRANSFERS.                          PAS10300
                                                                                PAS10310
00033  000000512544     PTBL   BCI     1,000REM        REMARK CARD.             PAS10320
00034  0020 00 0 00045         TRA     REM             ..                       PAS10330
00035  000000314563            BCI     1,000INT        FORTRAN INTEGER.         PAS10340
00036  0020 00 0 00046         TRA     INT             ..                       PAS10350
00037  000046236343            BCI     1,000CTL        SIMPLE OCTAL.            PAS10360
00040  0020 00 0 00062         TRA     OCTL            ..                       PAS10370
00041  000023464447            BCI     1,00COMP        ARITHMETIC.              PAS10380
00042  0020 00 0 00053         TRA     COMP            ..                       PAS10390
00043  000000254524            BCI     1,000END        END CARD.                PAS10400
00044  0020 00 0 00065         TRA     END             ..
```

```
                PASS 1 OF CLASS ASSEMBLY PROGRAM.

                  00012   NPTBL EQU     *-PTBL        2*(NUMBER OF PSEUDO-OPS).                  PAS10410

                                SPACE   2                                                        PAS10420
                                        PSEUDO-OPS.                                              PAS10430
                                                                                                 PAS10440
00045  0020 00 0 00015   REM   TRA     NEXT          IGNORE REMARK, RETURN.                     PAS10450
                                                                                                 PAS10460
00046 -0500 00 0 00100   INT   CAL     BUFF          GET SYMBOL.                                PAS10470
00047  0074 00 4 00003         TSX     $SYMSTO,4     SAVE.                                      PAS10480
00050  0074 00 4 00004         TSX     $COMMA,4      GO COUNT COMMAS.                           PAS10490
00051  0 00000 0 00100         PZE     BUFF          ..                                         PAS10500
00052  0020 00 0 00015         TRA     NEXT          RETURN.                                    PAS10510
                                                                                                 PAS10520
00053 -0500 00 0 00100   COMP  CAL     BUFF          GET SYMBOL.                                PAS10530
00054  0074 00 4 00003         TSX     $SYMSTO,4     SAVE.                                      PAS10540
00055  0074 00 4 00000         TSX     $WCT1,4       WRITE COMP CARD ON COLLATION TAPE.         PAS10550
00056  0 00000 0 00100         PZE     BUFF          ..                                         PAS10560
00057  0074 00 4 00005         TSX     $COMPOP,4     GO COMPILE.                                PAS10570
00060  0 00000 0 00100         PZE     BUFF          ..                                         PAS10580
00061  0020 00 0 00017         TRA     RCD           RETURN FOR NEXT CARD.                      PAS10590
                                                                                                 PAS10600
00062 -0500 00 0 00100   OCTL  CAL     BUFF          GET SYMBOL.                                PAS10610
00063  0074 00 4 00003         TSX     $SYMSTO,4     SAVE.                                      PAS10620
00064  1 77777 1 00015         TXI     NEXT,1,-1     ADVANCE ILC AND RETURN.                    PAS10630
                                                                                                 PAS10640
00065  0074 00 4 00006   END   TSX     $ENDOP,4      GO TO RESERVE STORAGE AND LITERALS.        PAS10650
00066  0074 00 4 00000         TSX     $WCT1,4       WRITE END CARD ON COLLATION TAPE.          PAS10660
00067  0 00000 0 00100         PZE     BUFF          ..                                         PAS10670
00070 -0500 00 0 00100         CAL     BUFF          GET SYMBOL.                                PAS10680
00071  0074 00 4 00003         TSX     $SYMSTO,4     SAVE.                                      PAS10690
00072  0074 00 4 00007         TSX     $REWIND,4     REWIND COLLATION TAPE.                     PAS10700
00073  0754 00 1 00000         PXA     0,1           GET FIRST LOCATION NOT USED BY PROGRAM.    PAS10710
00074  0737 00 4 00000         PAC     0,4           RECOMPLEMENT.                              PAS10720
00075  0754 00 4 00000         PXA     0,4           PUT IN AC.                                 PAS10730
00076  0774 00 4 00000   RX4   AXT     **,4          RESTORE IR4.                               PAS10740
00077  0020 00 4 00001         TRA     1,4           RETURN TO CALLER.                          PAS10750
                                                                                                 PAS10760
00100                    BUFF  BSS     14            STORAGE FOR HOLLERITH CARD IMAGE.          PAS10770
                                                                                                 PAS10780
                                END                                                              PAS10790
```

PASS 1 OF CLASS ASSEMBLY PROGRAM.
POST PROCESSOR ASSEMBLY DATA

116 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

30	OP											
65	END	44										
46	INT	36										
17	RCD	14,	61									
45	REM	34										
76	RX4	12										
100	BUFF	16,	20,	21,	30,	46,	51,	53,	56,	60,	62,	67, 70
53	COMP	42										
15	NEXT	32,	45,	52,	64							
62	OCTL	40										
21	OFLD											
33	PTBL	24,	26,	45								
2	SCAN	22										
0	WCT1	15,	55,	66								
4	COMMA	50										
6	ENDOP	65										
12	NPTBL	23,	24,	26,	45							
12	PASS1											
1	READ1	17										
5	COMPOP	57										
7	REWIND	72										
3	SYMSTO	31,	47,	54,	63,	71						

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000005 IN HUNDREDTHS OF MINUTES.

```
                PASS 2 OF CLASS ASSEMBLY PROGRAM.

                                PCC
                                COUNT   221                                                     PAS20010
                                LBL     PASS2           BINARY CARD LABEL.                      PAS20020
                                ENTRY   PASS2           SECOND ASSEMBLY PASS OF CAP.            PAS20030
                  00010                                                                         PAS20050

  TRANSFER VECTOR
00000   464763224360      OPTBL
00001   512521240260      READ2
00002   622321456060      SCAN
00003   652151256543      VAREVL
00004   314563464760      INTOP
00005   475131456360      PRINT

  LINKAGE DIRECTOR
00006   000000000000
00007   472162620260

00010   0634 00 4 00136   PASS2 SXA     Rx4,4           SAVE IR4.                               PAS20060
00011   0737 00 1 00000         PAC     0,1             IR1 IS -(ILC).                          PAS20070
00012  -0500 60 0 00000         CAL*    $OPTBL          SETUP SEARCH FOR OP.                    PAS20080
00013   0771 00 0 00022         ARS     18              LENGTH TO ADDRESS.                      PAS20090
00014   0621 00 0 00037         STA     OPAXT           SAVE LENGTH.                            PAS20100
00015   0361 60 0 00000         ACL*    $OPTBL          FIRST+LTH.                              PAS20110
00016   0621 00 0 00040         STA     OPLAS           SAVE FOR LAS.                           PAS20120
00017   0621 00 0 00047         STA     OPFND           SAVE FOR PICKUP.                        PAS20130
00020   0441 00 0 00312         LDI     =0              CLEAR INDICATORS.                       PAS20140
00021   0600 00 0 00263         STZ     FLGSM           CLEAR TOTAL FLAGS.                      PAS20150
                                                                                                PAS20160
00022   0442 00 0 00263   NEXT  OSI     FLGSM           FORM COMPOSITE FLAGS.                   PAS20170
00023   0604 00 0 00263         STI     FLGSM           SAVE.                                   PAS20180
00024   0441 00 0 00312         LDI     =0              CLEAR INDICATORS.                       PAS20190
00025   0634 00 1 00257         SXA     LOC,1           INITIAL ILC FOR EACH CARD.              PAS20200
                                                                                                PAS20210
00026   0074 00 4 00001         TSX     $READ2,4        READ IN NEXT CARD.                      PAS20220
00027   0 00000 0 00274         PZE     BUFF            ..                                      PAS20230
                                                                                                PAS20240
00030  -0500 00 0 00275   OFLD  CAL     BUFF+1          GET COMPRESSED OP-FIELD.                PAS20250
00031   0074 00 4 00002         TSX     $SCAN,4         ..                                      PAS20260
                                                                                                PAS20270
00032   0774 00 4 00012         AXT     NPTBL,4         CHECK FOR PSEUDO-OP.                    PAS20280
00033  -0340 00 4 00070         LAS     PTBL+NPTBL,4    COMPARE WITH TABLE.                     PAS20290
00034   0020 00 0 00036         TRA     *+2             NO, SKIP.                               PAS20300
00035   0020 60 4 00071         TRA*    PTBL+NPTBL+1,4  YES, EXIT.                              PAS20310
00036   2 00002 4 00033         TIX     *-3,4,2         NO, INDEX AND TRY AGAIN.                PAS20320
                                                                                                PAS20330
00037   0774 00 4 00000   OPAXT AXT     **,4            OP, GET LENGTH OF OP-TABLE FOR SEARCH.  PAS20340
00040  -0340 00 4 00000   OPLAS LAS     **,4            COMPARE WITH CURRENT TABLE ENTRY.       PAS20350
00041   0020 00 0 00043         TRA     *+2             NO, SKIP.                               PAS20360
00042   1 77777 4 00047         TXI     OPFND,4,-1      FOUND, INDEX AND EXIT.                  PAS20370
00043   2 00002 4 00040         TIX     *-3,4,2         NO, INDEX AND TRY AGAIN.                PAS20380
                                                                                                PAS20390
00044   0055 00  000002         SIR     2               ILLEGAL OPCODE, SET FLAG.               PAS20400
00045  -0754 00 0 00000         PXD     0,0             TAKE ZERO FOR OP.                       PAS20410
00046   0020 00 0 00050         TRA     OPFND+1         SKIP PICKUP.                            PAS20420
                                                                                                PAS20430
00047  -0500 00 4 00000   OPFND CAL     **,4            PICKUP OPCODE FROM OP-TABLE.            PAS20440
```

```
                PASS 2 OF CLASS ASSEMBLY PROGRAM.

00050  0602 00 1 00000          SLW     0,1             INSERT IN ASSEMBLED PROGRAM.      PAS20450
                                                                                           PAS20460
00051  0074 00 4 00003          TSX     $VAREVL,4       GO EVALUATE VARIABLE FIELD.       PAS20470
00052  0 00000 0 00274          PZE     BUFF            ..                                PAS20480
00053 -0602 00 1 00000          ORS     0,1             OR TO WORD IN ASSEMBLED PROGRAM.  PAS20490
                                                                                           PAS20500
00054  0074 00 4 00142          TSX     PRNT1,4         GO PRINT ASSEMBLY LISTING.        PAS20510
                                                                                           PAS20520
00055  1 77777 1 00022          TXI     NEXT,1,-1       RETURN.                           PAS20530


                                SPACE   2                                                 PAS20540
                                        PSEUDO-OP TABLE AND TRANSFERS.                    PAS20550
                                                                                          PAS20560
00056  000000512544      PTBL   BCI     1,COOREM        REMARK CARD.                      PAS20570
00057  0020 00 0 00070          TRA     REM             ..                                PAS20580
00060  000000314563             BCI     1,000INT        FORTRAN INTEGER.                  PAS20590
00061  0020 00 0 00072          TRA     INT             ..                                PAS20600
00062  000046236343             BCI     1,000CTL        SIMPLE OCTAL.                     PAS20610
00063  0020 00 0 00076          TRA     OCTL            ..                                PAS20620
00064  000023464447             BCI     1,00COMP        ARITHMETIC.                       PAS20630
00065  0020 00 0 00111          TRA     COMP            ..                                PAS20640
00066  000000254524             BCI     1,000END        END CARD.                         PAS20650
00067  0020 00 0 00113          TRA     END             ..                                PAS20660
                00012    NPTBL  EQU     *-PTBL          2*(NUMBER OF PSEUDO-OPS).         PAS20670


                                SPACE   2                                                 PAS20680
                                        PSEUDO-OPS.                                       PAS20690
                                                                                          PAS20700
00070  0074 00 4 00173   REM    TSX     PRNT2,4         PRINT REMARK.                     PAS20710
00071  0020 00 0 00022          TRA     NEXT            RETURN.                           PAS20720
                                                                                          PAS20730
00072  0074 00 4 00004   INT    TSX     $INTOP,4        DO INTEGER CONVERSION.            PAS20740
00073  0 00000 0 00274          PZE     BUFF            ..                                PAS20750
00074  0074 00 4 00142          TSX     PRNT1,4         GO PRINT ASSEMBLY LISTING.        PAS20760
00075  0020 00 0 00022          TRA     NEXT            RETURN.                           PAS20770
                                                                                          PAS20780
00076 -0754 00 0 00000   OCTL   PXD     0,0             SIMPLE OCTAL, CLEAR AC.           PAS20790
00077  0774 00 2 00002          AXT     2,2             2 WORDS.                          PAS20800
00100  0774 00 4 00006   OLP    AXT     6,4             6 CHARACTERS PER WORD.            PAS20810
00101  0560 00 2 00300          LDQ     BUFF+2+2,2      GET WORD.                         PAS20820
00102 -0773 00 0 00003          RQL     3               BCI-OCTAL CONVERSION.             PAS20830
00103 -0763 00 0 00003          LGL     3               ..                                PAS20840
00104  2 00001 4 00102          TIX     *-2,4,1         COUNT CHARACTERS.                 PAS20850
00105  2 00001 2 00100          TIX     OLP,2,1         COUNT WORDS.                      PAS20860
00106  0602 00 1 00000          SLW     0,1             INSERT IN ASSEMBLED PROGRAM.      PAS20870
00107  0074 00 4 00142          TSX     PRNT1,4         GO PRINT ASSEMBLY LISTING.        PAS20880
00110  1 77777 1 00022          TXI     NEXT,1,-1       INDEX AND RETURN.                 PAS20890
                                                                                          PAS20900
00111  0074 00 4 00173   COMP   TSX     PRNT2,4         PRINT COMP AS A REMARK.           PAS20910
00112  0020 00 0 00022          TRA     NEXT            RETURN.                           PAS20920
                                                                                          PAS20930
00113  0074 00 4 00003   END    TSX     $VAREVL,4       EVALUATE VARIABLE FIELD.          PAS20940
00114  0 00000 0 00274          PZE     BUFF            ..                                PAS20950
00115  0601 00 0 00260          STO     EPNT            SAVE AS ENTRY POINT.              PAS20960
```

```
00116  0074 00 4 00204          TSX     FLAGS,4         GET FLAGS.                          PAS20970
00117  0602 00 0 00267          SLW     PBUFF           PLACE IN PBUFF.                     PAS20980
00120  0500 00 0 00260          CLA     EPNT            CONVERT ENTRY POINT.                PAS20990
00121  0074 00 4 00225          TSX     OCTA,4          ..                                  PAS21000
00122  0560 00 0 00312          LDQ     =0              CLEAR MQ.                           PAS21010
00123 -0765 00 0 00022          LGR     18              SHIFT TO POSITION.                  PAS21020
00124 -0501 00 0 00316          ORA     =H   000        INSERT BLANKS LEFT.                 PAS21030
00125  0602 00 0 00272          SLW     PBUFF+3         INSERT IN PBUFF.                    PAS21040
00126 -0130 00 0 00000          XCL                     GET RIGHT HALF.                     PAS21050
00127 -0501 00 0 00315          ORA     =H000           INSERT BLANKS RIGHT.                PAS21060
00130  0602 00 0 00273          SLW     PBUFF+4         INSERT IN PBUFF.                    PAS21070
00131 -0500 00 0 00317          CAL     =H              BLANK OUT REST OF PBUFF.            PAS21080
00132  0602 00 0 00270          SLW     PBUFF+1         ..                                  PAS21090
00133  0602 00 0 00271          SLW     PBUFF+2         ..                                  PAS21100
00134  0074 00 4 00005          TSX     $PRINT,4        GO PRINT.                           PAS21110
00135  0 00023 0 00267          PZE     PBUFF,0,19      ..                                  PAS21120
00136  0774 00 4 00000  RX4     AXT     **,4            RESTORE IR4.                        PAS21130
00137  0500 00 0 00260          CLA     EPNT            GET ENTRY.                          PAS21140
00140  0442 00 0 00263          OSI     FLGSM           GET TOTAL ERROR FLAGS.              PAS21150
00141  0020 00 4 00001          TRA     1,4             RETURN.                             PAS21160

                                SPACE   2                                                   PAS21170
                                PRINT ROUTINES.                                             PAS21180
                                                                                            PAS21190
00142  0634 00 4 00171  PRNT1   SXA     P1X4,4          SAVE IR4.                           PAS21200
00143  0074 00 4 00204          TSX     FLAGS,4         GET ERROR FLAGS.                    PAS21210
00144  0602 00 0 00267          SLW     PBUFF           PLACE IN PBUFF.                     PAS21220
00145  0535 00 4 00257          LAC     LOC,4           GET +(ILC).                         PAS21230
00146  0754 00 4 00000          PXA     0,4             ..                                  PAS21240
00147  0074 00 4 00225          TSX     OCTA,4          CONVERT OCTAL ADDRESS.              PAS21250
00150  0602 00 0 00270          SLW     PBUFF+1         PLACE IN PBUFF.                     PAS21260
00151  0534 00 4 00257          LXA     LOC,4           GET -(ILC).                         PAS21270
00152 -0500 00 4 00000          CAL     0,4             GET ASSEMBLED WORD.                 PAS21280
00153  0074 00 4 00236          TSX     OCTW,4          CONVERT OCTAL WORD.                 PAS21290
00154 -0600 00 0 00262          STQ     RHOCT           SAVE RIGHT HALF.                    PAS21300
00155 -0765 00 0 00022          LGR     18              SHIFT TO POSITION.                  PAS21310
00156 -0501 00 0 00316          ORA     =H   000        INSERT BLANKS LEFT.                 PAS21320
00157  0602 00 0 00271          SLW     PBUFF+2         PLACE IN PBUFF.                     PAS21330
00160 -0600 00 0 00272          STQ     PBUFF+3         ..                                  PAS21340
00161 -0500 00 0 00262          CAL     RHOCT           GET RIGHT HALF.                     PAS21350
00162  0560 00 0 00312          LDQ     =0              ZERO MQ.                            PAS21360
00163 -0765 00 0 00022          LGR     18              SHIFT TO POSITION.                  PAS21370
00164 -0130 00 0 00000          XCL                     PLACE IN AC.                        PAS21380
00165 -0501 00 0 00315          ORA     =H000           INSERT BLANKS RIGHT.                PAS21390
00166  0602 00 0 00273          SLW     PBUFF+4         PLACE IN PBUFF.                     PAS21400
00167  0074 00 4 00005          TSX     $PRINT,4        GO PRINT.                           PAS21410
00170  0 00023 0 00267          PZE     PBUFF,0,19      ..                                  PAS21420
00171  0774 00 4 00000  P1X4    AXT     **,4            RESTORE IR4.                        PAS21430
00172  0020 00 4 00001          TRA     1,4             RETURN.                             PAS21440
                                                                                            PAS21450
00173  0634 00 4 00202  PRNT2   SXA     P2X4,4          SAVE IR4.                           PAS21460
00174  0774 00 4 00005          AXT     5,4             BLANK OUT PBUFF TO PBUFF+4.         PAS21470
00175 -0500 00 0 00317          CAL     =H              ..                                  PAS21480
00176  0602 00 4 00274          SLW     PBUFF+5,4       ..                                  PAS21490
00177  2 00001 4 00176          TIX     *-1,4,1         ..                                  PAS21500
```

PASS 2 OF CLASS ASSEMBLY PROGRAM.

```
00200  0074 00 4 00005          TSX    $PRINT,4        GO PRINT.                          PAS21510
00201  0 00023 0 00267          PZE    PBUFF,0,19      ..                                 PAS21520
00202  0774 00 4 00000  P2X4    AXT    **,4            RESTORE IR4.                       PAS21530
00203  0020 00 4 00001          TRA    1,4             RETURN.                            PAS21540

                                SPACE  2                                                  PAS21550
                                BCI CONVERSION ROUTINES.                                  PAS21560
                                                                                          PAS21570
00204  0634 00 4 00223  FLAGS   SXA    FLX4,4          SAVE IR4.                          PAS21580
00205  0140 00 0 00206          TOV    *+1             TURN OFF OVERFLOW LIGHT.           PAS21590
00206 -0046 00 0 00000          PIA                    GET ERROR FLAGS.                   PAS21600
00207 -0765 00 0 00003          LGR    NFLGS           SHIFT TO MQ.                       PAS21610
00210 -0500 00 0 00314          CAL    =H000001        BLANK FOR CARRIAGE CONTROL.        PAS21620
00211  0774 00 4 00003          AXT    NFLGS,4         CONVERT FLAGS.                     PAS21630
00212  0162 00 0 00215          TQP    *+3             IF PLUS, NO FLAG.                  PAS21640
00213  0767 00 0 00006          ALS    6               INSERT FLAG.                       PAS21650
00214  0361 00 4 00267          ACL    TFLGS+NFLGS,4   ..                                 PAS21660
00215 -0773 00 0 00001          RQL    1               CHECK NEXT BIT.                    PAS21670
00216  2 00001 4 00212          TIX    *-4,4,1         INDEX.                             PAS21680
00217  0560 00 0 00317          LDQ    =H              FILL IN BLANKS ON RIGHT.           PAS21690
00220  0140 00 0 00223          TOV    *+3             ..                                 PAS21700
00221 -0763 00 0 00006          LGL    6               ..                                 PAS21710
00222 -0140 00 0 00221          TNO    *-1             ..                                 PAS21720
00223  0774 00 4 00000  FLX4    AXT    **,4            RESTORE IR4.                       PAS21730
00224  0020 00 4 00001          TRA    1,4             RETURN.                            PAS21740
                                                                                          PAS21750
00225  0634 00 4 00234   OCTA    SXA    OAX4,4          SAVE IR4.                          PAS21760
00226 -0765 00 0 00017          LGR    15              SHIFT TO MQ.                       PAS21770
00227 -0500 00 0 00313          CAL    =H00000         FIRST, A BLANK.                    PAS21780
00230  0774 00 4 00005          AXT    5,4             5 DIGITS.                          PAS21790
C0231  0767 00 0 00003          ALS    3               CONVERT TO BCI.                    PAS21800
00232 -0763 00 0 00003          LGL    3               ..                                 PAS21810
00233  2 00001 4 00231          TIX    *-2,4,1         ..                                 PAS21820
00234  0774 00 4 00000  OAX4    AXT    **,4            RESTORE IR4.                       PAS21830
00235  0020 00 4 00001          TRA    1,4             RETURN.                            PAS21840
                                                                                          PAS21850
00236  0634 00 4 00255   OCTW    SXA    OWX4,4          SAVE IR4.                          PAS21860
00237 -0130 00 0 00000          XCL                    PLACE IN MQ.                       PAS21870
00240 -0754 00 0 00000          PXD    0,0             FIRST HALF, CLEAR AC.              PAS21880
00241  0774 00 4 00006          AXT    6,4             6 DIGITS.                          PAS21890
00242  0767 00 0 00003          ALS    3               CONVERT.                           PAS21900
00243 -0763 00 0 00003          LGL    3               ..                                 PAS21910
00244  2 00001 4 00242          TIX    *-2,4,1         ..                                 PAS21920
00245  0602 00 0 00261          SLW    LHOCT           SAVE LEFT HALF.                    PAS21930
00246 -0754 00 0 00000          PXD    0,0             LAST HALF, CLEAR AC.               PAS21940
00247  0774 00 4 00006          AXT    6,4             6 DIGITS.                          PAS21950
00250  0767 00 0 00003          ALS    3               CONVERT.                           PAS21960
00251 -0763 00 0 00003          LGL    3               ..                                 PAS21970
00252  2 00001 4 00250          TIX    *-2,4,1         ..                                 PAS21980
00253 -0130 00 0 00000          XCL                    FORM COMPLETE RESULT.              PAS21990
00254 -0500 00 0 00261          CAL    LHOCT           ..                                 PAS22000
00255  0774 00 4 00000  OWX4    AXT    **,4            RESTORE IR4.                       PAS22010
00256  0020 00 4 00001          TRA    1,4             RETURN.                            PAS22020
```

```
                                SPACE   2                                                                PAS22030
                                        STORAGE AND CONSTANTS.                                           PAS22040
                                                                                                         PAS22050
00257  0 00000 0 00000   LOC    PZE                     -(ILC) BEFORE CONVERSIONS.                       PAS22060
00260  0 00000 0 00000   EPNT   PZE                     ENTRY POINT FROM END CARD.                       PAS22070
00261  0 00000 0 00000   LHOCT  PZE                     LEFT HALF OF OCTAL-BCI.                          PAS22080
00262  0 00000 0 00000   RHOCT  PZE                     RIGHT HALF OF OCTAL-BCI.                         PAS22090
00263  0 00000 0 00000   FLGSM  PZE                     TOTAL ERROR FLAGS.                               PAS22100
                                                                                                         PAS22110
                  00264   TFLGS  SYN     *             TABLE OF ERROR FLAGS.                            PAS22120
00264  000000000025             BCI     1,COCCOE      SI BIT 33.                                       PAS22130
00265  000000000046             BCI     1,000000      SI BIT 34.                                       PAS22140
00266  000000000064             BCI     1,00000U      SI BIT 35.                                       PAS22150
                  00003   NFLGS  EQU     *-TFLGS       NUMBER OF ERROR FLAGS.                           PAS22160
                                                                                                         PAS22170
00267                    PBUFF  BSS     19            PRINT BUFFER.                                    PAS22180
                  00274   BUFF   SYN     PBUFF+5       START OF CARD IMAGE BUFFER.                      PAS22190
                                                                                                         PAS22200
                                END                                                                      PAS22210
```

LITERALS

```
00312  000000000000
00313  000000000060
00314  000000000160
00315  000000606060
00316  606060000000
00317  606060606060
```

PASS 2 OF CLASS ASSEMBLY PROGRAM.
POST PROCESSOR ASSEMBLY DATA

320 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

113	END	67
72	INT	61
257	LOC	25, 145, 151
100	OLP	105
70	REM	57
136	RX4	10
274	BUFF	27, 30, 52, 73, 101, 114, 312
111	COMP	65
260	EPNT	115, 120, 137
223	FLX4	204
22	NEXT	55, 71, 75, 110, 112
234	OAX4	225
225	OCTA	121, 147
76	OCTL	63
236	OCTW	153
30	OFLD	
255	OWX4	236
171	P1X4	142
202	P2X4	173
56	PTBL	33, 35, 70
2	SCAN	31
204	FLAGS	116, 143
263	FLGSM	21, 22, 23, 140
4	INTOP	72
261	LHOCT	245, 254
3	NFLGS	207, 211, 214, 267
12	NPTBL	32, 33, 35, 70
37	OPAXT	14
47	OPFND	17, 42, 46
40	OPLAS	16
0	OPTBL	12, 15
10	PASS2	
267	PBUFF	117, 125, 130, 132, 133, 135, 144, 150, 157, 160, 166, 170, 176, 201, 312
5	PRINT	134, 167, 200
142	PRNT1	54, 74, 107
173	PRNT2	70, 111
1	READ2	26
262	RHOCT	154, 161
264	TFLGS	214, 264, 267
3	VAREVL	51, 113

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000009 IN HUNDREDTHS OF MINUTES.

```
                $VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.

                                PCC
                                COUNT   222                                                     VEVL0010
                                LBL     VAREVL      BINARY CARD LABEL.                          VEVL0020
                     00034      ENTRY   RVREVL      EVALUATE FIELDS BETWEEN COMMAS.             VEVL0030
                     00005      ENTRY   VAREVL      EVALUATE FIRST FIELD.                       VEVL0050
                                                                                                VEVL0060
  TRANSFER VECTOR
00000  627044272563      SYMGET
00001  622165256060      SAVE
00002  644562216525      UNSAVE

  LINKAGE DIRECTOR
00003  000000000000
00004  516551256543

00005   0634 00 4 00030  VAREVL SXA     RX4,4       SAVE IRS.                                   VEVL0070
00006   0634 00 2 00031        SXA     RX2,2       ..                                          VEVL0080
00007   0634 00 1 00032        SXA     RX1,1       ..                                          VEVL0090
00010  -0500 00 4 00001        CAL     1,4         GET BUFFER ADDRESS.                         VEVL0100
00011   0361 00 0 00254        ACL     =12         BUFF+12.                                    VEVL0110
00012   0621 00 0 00063        STA     LDQ         SAVE FOR PICKUP.                            VEVL0120
00013   0600 00 0 00245        STZ     TEOF        RESET EOF MARK.                             VEVL0130
00014   0600 00 0 00246        STZ     TCOM        RESET COMMA MARK.                           VEVL0140
00015  -0760 00 0 00141        SLT     1           TURN OFF LIGHT 1.                           VEVL0150
00016   0761 00 0 00000        NOP                 ..                                          VEVL0160
00017   0774 00 2 00012        AXT     10,2        COUNT 10 WORDS.                             VEVL0170
00020   0774 00 1 00007        AXT     7,1         COUNT 6 CHARACTERS.                         VEVL0180
00021   0560 60 0 00063        LDQ*    LDQ         GET FIRST WORD OF VARIABLE FIELD.           VEVL0190
00022  -0600 00 0 00252        STQ     MQ          SAVE FOR EVAL.                              VEVL0200
00023   0074 00 4 00052  GEVAL TSX     EVAL,4      GO EVALUATE FIELD.                          VEVL0210
00024   0734 00 4 00000        PAX     0,4         PLACE RESULT IN IR4.                        VEVL0220
00025   0120 00 0 00027        TPL     *+2         PLUS OR MINUS.                              VEVL0230
00026   0737 00 4 00000        PAC     0,4         MINUS, FORM 2S COMPLEMENT.                  VEVL0240
00027   0754 00 4 00000        PXA     0,4         FINAL RESULT IN A(AC).                      VEVL0250
00030   0774 00 4 00000  RX4    AXT     **,4        RESTORE IRS.                                VEVL0260
00031   0774 00 2 00000  RX2    AXT     **,2        ..                                          VEVL0270
00032   0774 00 1 00000  RX1    AXT     **,1        ..                                          VEVL0280
00033   0020 00 4 00002        TRA     2,4         RETURN WITH RESULT IN AC.                   VEVL0290

                                SPACE   2                                                       VEVL0300
                                RE-ENTRY TO EVALUATE MULTIPLE FIELDS.                           VEVL0310
00034  -0754 00 0 00000  RVREVL PXD     0,0         CLEAR AC.                                   VEVL0320
00035  -0520 00 0 00246        NZT     TCOM        CHECK FOR COMMA ENCOUNTERED.                VEVL0330
00036   0020 00 4 00001        TRA     1,4         NO, EXIT WITH ZERO.                         VEVL0340
00037   0600 00 0 00245        STZ     TECF        RESET EOF MARK.                             VEVL0350
00040   0600 00 0 00246        STZ     TCOM        RESET COMMA MARK.                           VEVL0360
00041  -0760 00 0 00141        SLT     1           TURN OFF LIGHT 1.                           VEVL0370
00042   0761 00 0 00000        NOP                 ..                                          VEVL0380
00043   1 00001 4 00044        TXI     *+1,4,1     DECREASE CALL LOCATION BY ONE.              VEVL0390
00044   0634 00 4 00030        SXA     RX4,4       SAVE IRS.                                   VEVL0400
00045   0634 00 2 00031        SXA     RX2,2       ..                                          VEVL0410
00046   0634 00 1 00032        SXA     RX1,1       ..                                          VEVL0420
00047   0774 00 1 00000  REX1   AXT     **,1        RESTORE IRS FOR POSITION IN FIELD.          VEVL0430
00050   0774 00 2 00000  REX2   AXT     **,2        ..                                          VEVL0440
                                                                                                VEVL0450
```

```
                $VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.

00051  0020 00 0 00023          TRA     GEVAL           GO EVALUATE THIS FIELD.                    VEVL0460

                                SPACE   2                                                          VEVL0470
                                EVALUATION SUBROUTINE, RECURSIVELY DEFINED.                        VEVL0480
                                                                                                   VEVL0490
                                                                                                   VEVL0500
00052  0634 00 4 00243   EVAL   SXA     EVX4,4          SAVE IR4.                                  VEVL0510
00053  0600 00 0 00241          STZ     SUM             INITIALIZE REGISTERS.                      VEVL0520
00054  0600 00 0 00242          STZ     TERM            ..                                         VEVL0530
00055  0600 00 0 00251          STZ     VAL             ..                                         VEVL0540
00056  0600 00 0 00250          STZ     SYM             RESET SYM.                                 VEVL0550
00057  0774 00 4 00025          AXT     NPL,4           SET LBKCH TO PLUS.                         VEVL0560
00060  0634 00 4 00244          SXA     LBKCH,4         ..                                         VEVL0570
00061  0020 00 0 00100          TRA     RSCAN           GO TO SCANER.                              VEVL0580
                                                                                                   VEVL0590
00062  0774 00 1 00006   SCAN   AXT     6,1             COUNT 6 CHARACTERS.                        VEVL0600
00063  0560 00 2 00000   LDQ    LDQ     **,2            PICKUP NEXT WORD.  ADDRESS IS BUFF+12.     VEVL0610
00064 -0754 00 0 00000   CHAR   PXD     0,0             CLEAR AC.                                  VEVL0620
00065 -0763 00 0 00006          LGL     6               GET CHARACTER.                             VEVL0630
00066 -0600 00 0 00252          STQ     MQ              SAVE MQ.                                   VEVL0640
00067  0774 00 4 00025          AXT     NBK,4           COMPARE WITH LIST OF BREAKS.               VEVL0650
00070 -0340 00 4 00136          LAS     TABBK+NBK,4     ..                                         VEVL0660
00071  0020 00 0 00073          TRA     *+2             NOT THIS ONE, SKIP.                        VEVL0670
00072  0020 00 0 00136          TRA     BKCH            BREAK FOUND, EXIT.                         VEVL0680
00073  2 00003 4 00070          TIX     *-3,4,3         NOT THIS ONE, INDEX AND TRY AGAIN.         VEVL0690
00074 -0765 00 0 00006          LGR     6               NOT A BREAK, BUILD SYMBOL.                 VEVL0700
00075 -0500 00 0 00250          CAL     SYM             ..                                         VEVL0710
00076 -0763 00 0 00006          LGL     6               ..                                         VEVL0720
00077  0602 00 0 00250          SLW     SYM             SAVE PARTIAL SYMBOL.                       VEVL0730
00100  0520 00 0 00245   RSCAN  ZET     TEOF            TEST FOR END-OF-FIELD.                     VEVL0740
00101  0020 00 0 00106          TRA     EOFB            YES, EXIT TO RPAR SECTION.                 VEVL0750
00102  0560 00 0 00252          LDQ     MQ              NO, RESTORE MQ.                            VEVL0760
00103  2 00001 1 00064          TIX     CHAR,1,1        COUNT CHARACTERS.                          VEVL0770
00104  2 00001 2 00062          TIX     SCAN,2,1        COUNT WORDS.                               VEVL0780
00105 -0625 00 0 00245          STL     TEOF            END-OF-FIELD REACHED, APPEND AS            VEVL0790
00106  0774 00 4 00003   EOFB   AXT     NRPAR,4         MANY RPAR AS NECESSARY.                    VEVL0800
00107 -0500 00 0 00255          CAL     =H000000        ..                                         VEVL0810
00110  0020 00 0 00136          TRA     BKCH            GO TO BREAK.

                                SPACE   2                                                          VEVL0820
                                TABLE OF BREAKS.                                                   VEVL0830
                                                                                                   VEVL0840
                        00111   TABBK   SYN     *                                                  VEVL0850
                                                                                                   VEVL0860
                                                                                                   VEVL0870
00111  000000000020     CPL    BCI     1,00000+        PLUS.                                      VEVL0880
00112  0020 00 0 00172          TRA     LPL             ..                                         VEVL0890
00113  0020 00 0 00210          TRA     RPL             ..                                         VEVL0900
00114  000000000040             BCI     1,00000-        MINUS.                                     VEVL0910
00115  0020 00 0 00175          TRA     LMI             ..                                         VEVL0920
00116  0020 00 0 00210          TRA     RMI             ..                                         VEVL0930
00117  000000000054             BCI     1,00000*        STAR.                                      VEVL0940
00120  0020 00 0 00200          TRA     LST             ..                                         VEVL0950
00121  0020 00 0 00215          TRA     RST             ..                                         VEVL0960
00122  000000000060             BCI     1,00000         BLANK.                                     VEVL0970
00123  0000 60 0 00123          HTR*    *               SHOULD NEVER GET HERE.
```

```
                $VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.

00124  0020 00 0 00237          TRA     BLANK           ..                              VEVL0980
00125  000000000073             BCI     1,00000,        COMMA.                          VEVL0990
00126  0000 60 0 00126          HTR*    *               SHOULD NEVER GET HERE.          VEVL1000
00127  0020 00 0 00231          TRA     RCOM            ..                              VEVL1010
00130  000000000074             BCI     1,00000(        LPAR.                           VEVL1020
00131  0000 60 0 00131          HTR*    *               SHOULD NEVER GET HERE.          VEVL1030
00132  0020 00 0 00216          TRA     LPAR            ..                              VEVL1040
00133  000000000034     CRPAR   BCI     1,00000)        RPAR.                           VEVL1050
00134  0000 60 0 00134          HTR*    *               SHOULD NEVER GET HERE.          VEVL1060
00135  0020 00 0 00225          TRA     RPAR            ..                              VEVL1070
                                                                                         VEVL1080
                  00025  NBK    EQU     *-TABBK         NUMBER OF BREAK CHARACTERS.     VEVL1090
                  00003  NRPAR  EQU     *-CRPAR         BREAK NUMBER OR RPAR.           VEVL1100
                  00025  NPL    EQU     *-CPL           BREAK NUMBER OF PLUS.           VEVL1110

                         SPACE   2                                                       VEVL1120
                                 BREAK CHARACTER SECTION.                                VEVL1130
                                                                                         VEVL1140
00136   0634 00 4 00247  BKCH   SXA     RBKCH,4         SAVE NUMBER OF RIGHT BREAK.     VEVL1150
00137  -0340 00 0 00256          LAS     =H000000(       CHECK FOR LPAR.                 VEVL1160
00140   0020 00 0 00142          TRA     *+2             NO, SKIP.                       VEVL1170
00141   0020 00 0 00216          TRA     LPAR            YES, GO TO IT.                  VEVL1180
00142   0520 00 0 00251          ZET     VAL             EXPRESSION, SYMBOL, OR NUMBER.  VEVL1190
00143   0020 00 0 00167          TRA     LBK             EXPRESSION, NO SYMBOL TO CONVERT. VEVL1200
00144  -0500 00 0 00250          CAL     SYM             SYMBOL OR NUMBER.               VEVL1210
00145  -0320 00 0 00257          ANA     =H              NUMBERS HAVE NO ZONE.           VEVL1220
00146   0100 00 0 00153          TZE     NUM             NUMBER, GO CONVERT.             VEVL1230
00147  -0500 00 0 00250          CAL     SYM             SYMBOL, GET VALUE.              VEVL1240
00150   0074 00 4 00000          TSX     $SYMGET,4       ..                              VEVL1250
00151   0601 00 0 00251          STO     VAL             SAVE VALUE.                     VEVL1260
00152   0020 00 0 00167          TRA     LBK             EXIT TO LBK.                    VEVL1270
                                                                                         VEVL1280
00153   0560 00 0 00250  NUM    LDQ     SYM             NUMBER, UNSIGNED.               VEVL1290
00154   0774 00 4 00006          AXT     6,4             COUNT 6 DIGITS.                 VEVL1300
00155  -0754 00 0 00000  NLOOP  PXD     0,0             CLEAR AC.                       VEVL1310
00156  -0763 00 0 00006          LGL     6               GET DIGIT.                      VEVL1320
00157   0601 00 0 00253          STO     DIG             SAVE.                           VEVL1330
00160   0500 00 0 00251          CLA     VAL             PROGRAMED 10*VAL.               VEVL1340
00161   0767 00 0 00002          ALS     2               4*VAL.                          VEVL1350
00162   0400 00 0 00251          ADD     VAL             4*VAL+VAL.                      VEVL1360
00163   0767 00 0 00001          ALS     1               2*(4*VAL+VAL)=10*VAL.           VEVL1370
00164   0400 00 0 00253          ADD     DIG             ADD THIS DIGIT.                 VEVL1380
00165   0601 00 0 00251          STO     VAL             SAVE PARTIAL RESULT.            VEVL1390
00166   2 00001 4 00155          TIX     NLOOP,4,1       COUNT DIGITS CONVERTED.         VEVL1400

                         SPACE   2                                                       VEVL1410
                                 LEFT BREAK SECTION                                      VEVL1420
                                                                                         VEVL1430
00167   0600 00 0 00250  LBK    STZ     SYM             LEFT BREAK, RESET SYM.          VEVL1440
00170   0534 00 4 00244          LXA     LBKCH,4         GET NUMBER OF LEFT BREAK.       VEVL1450
00171   0020 00 4 00137          TRA     TABBK+NBK+1,4   GO TO LEFT BREAK.               VEVL1460
                                                                                         VEVL1470
00172   0500 00 0 00251  LPL    CLA     VAL             +, TERM=VAL.                    VEVL1480
00173   0601 00 0 00242          STO     TERM            ..                              VEVL1490
```

```
                $VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.

00174  0020 00 0 00204          TRA     RBK             GO TO RIGHT BREAK.

00175  0502 00 0 00251   LMI    CLS     VAL             -, TERM=-VAL.
00176  0601 00 0 00242          STO     TERM            ..
00177  0020 00 0 00204          TRA     RBK             GO TO RIGHT BREAK.

00200  0560 00 0 00251   LST    LDQ     VAL             *, TERM=TERM*VAL.
00201  0200 00 0 00242          MPY     TERM            ..
00202 -0600 00 0 00242          STQ     TERM            ..
00203  0020 00 0 00204          TRA     RBK             GO TO RIGHT BREAK.


                         SPACE   2
                                 RIGHT BREAK SECTION.

00204  0534 00 4 00247   RBK    LXA     RBKCH,4         GET NUMBER OF RIGHT BREAK.
00205  0634 00 4 00244          SXA     LBKCH,4         THIS IS NEXT LEFT BREAK.
00206  0600 00 0 00251          STZ     VAL             RESET VAL.
00207  0020 00 4 00140          TRA     TABBK+NBK+2,4   GO TO RIGHT BREAK.

00210  0500 00 0 00241   RPL    CLA     SUM             +, SUM=SUM+TERM.
00211  0400 00 0 00242          ADD     TERM            ..
00212  0601 00 0 00241          STO     SUM             ..
00213  0600 00 0 00242          STZ     TERM            RESET TERM.
00214  0020 00 0 00100          TRA     RSCAN           RESUME SCAN.

               00210   RMI      SYN     RPL             -, SAME AS +.

00215  0020 00 0 00100   RST    TRA     RSCAN           *, RESUME SCAN.


                         SPACE   2
                                 LPAR, RPAR, AND EOF SECTION.

00216  0074 00 4 00001   LPAR   TSX     $SAVE,4         (, SAVE REGISTERS.
00217  0 00004 0 00241          PZE     SUM,0,4         ..
00220  0074 00 4 00052          TSX     EVAL,4          CALL SELF.
00221  0601 00 0 00251          STO     VAL             RESULT IS VAL.
00222  0074 00 4 00002          TSX     $UNSAVE,4       RESTORE REGISTERS.
00223  0 00004 0 00241          PZE     SUM,0,4         ..
00224  0020 00 0 00100          TRA     RSCAN           RESUME SCAN.

00225  0500 00 0 00241   RPAR   CLA     SUM             ), END OF EXPRESSION, GET SUM.
00226  0400 00 0 00242          ADD     TERM            ADD IN CURRENT TERM.
00227  0534 00 4 00243          LXA     EVX4,4          RESTORE IR4.
00230  0020 00 4 00001          TRA     1,4             RETURN TO CALLER.

00231 -0625 00 0 00246   RCOM   STL     TCOM            SET MARK FOR COMMA ENCOUNTERED.
00232 -0625 00 0 00245          STL     TEOF            SET MARK FOR EOF.
00233  0760 00 0 00141          SLN     1               EXTERNAL MARK FOR COMMA ENCOUNTERED.
00234  0634 00 1 00047          SXA     REX1,1          SAVE IRS FOR RE-SCAN.
00235  0634 00 2 00050          SXA     REX2,2          ..
00236  0020 00 0 00225          TRA     EOF             GO TO ECF SECTION.

00237 -0625 00 0 00245   BLANK  STL     TEOF            END-OF-FIELD.
00240  0020 00 0 00225          TRA     EOF             ..
```

```
                $VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.

                      00225   EOF   SYN     RPAR          SAME AS RPAR.                  VEVL2020
                                                                                         VEVL2030

                                    SPACE   2
                                            STORAGE AREA FOR SAVE.                       VEVL2040
                                                                                         VEVL2050
00241  0 00000 0 00000   SUM   PZE                    ..                                 VEVL2060
00242  0 00000 0 00000   TERM  PZE                    ..                                 VEVL2070
00243  0 00000 0 00000   EVX4  PZE                    ..                                 VEVL2080
00244  0 00000 0 00000   LBKCH PZE                    ..                                 VEVL2090
                                                                                         VEVL2100

                                    SPACE   2
                                            TEMPORARY STORAGE.                           VEVL2110
                                                                                         VEVL2120
00245  0 00000 0 00000   TEOF  PZE                    ..                                 VEVL2130
00246  0 00000 0 00000   TCOM  PZE                    ..                                 VEVL2140
00247  0 00000 0 00000   RBKCH PZE                    ..                                 VEVL2150
00250  0 00000 0 00000   SYM   PZE                    ..                                 VEVL2160
00251  0 00000 0 00000   VAL   PZE                    ..                                 VEVL2170
00252  0 00000 0 00000   MQ    PZE                    ..                                 VEVL2180
00253  0 00000 0 00000   DIG   PZE                    ..                                 VEVL2190
                                                                                         VEVL2200
                                    END                                                  VEVL2210
                                                                                         VEVL2220

  LITERALS
00254  000000000014
00255  000000000034
00256  000000000074
00257  606060606060
```

$VAREVL, SCAN AND EVALUATE VARIABLE FIELD OF CAP CARD.
POST PROCESSOR ASSEMBLY DATA

260 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

252	MQ	22, 66, 102
111	CPL	136
253	DIG	157, 164
225	EOF	236, 240, 241
167	LBK	143, 152
63	LDQ	12, 21
175	LMI	115
172	LPL	112
200	LST	120
25	NBK	67, 70, 136, 171, 207
25	NPL	57, 136
153	NUM	146
204	RBK	174, 177, 203
210	RMI	116, 215
210	RPL	113, 215
215	RST	121
32	RX1	7, 46
31	RX2	6, 45
30	RX4	5, 44
241	SUM	53, 210, 212, 217, 223, 225
250	SYM	56, 75, 77, 144, 147, 153, 167
251	VAL	55, 142, 151, 160, 162, 165, 172, 175, 200, 206, 221
136	BKCH	72, 110
64	CHAR	103
106	EOFB	101
52	EVAL	23, 220
243	EVX4	52, 227
216	LPAR	132, 141
231	RCOM	127
47	REX1	234
50	REX2	235
225	RPAR	135, 241
1	SAVE	216
62	SCAN	104
246	TCOM	14, 35, 40, 231
245	TEOF	13, 37, 100, 105, 232, 237
242	TERM	54, 173, 176, 201, 202, 211, 213, 226
237	BLANK	124
133	CRPAR	136
23	GEVAL	51
244	LBKCH	60, 170, 205
155	NLOOP	166
3	NRPAR	106, 136
247	RBKCH	136, 204
100	RSCAN	61, 214, 215, 224
111	TABBK	70, 111, 136, 171, 207
34	RVREVL	
0	SYMGET	150
2	UNSAVE	222
5	VAREVL	

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000010 IN HUNDREDTHS OF MINUTES.

```
                OPERATION TABLE FOR CAP.

                                PCC
                                COUNT   79                                              OPTB0010
                                LBL     OPTBL           BINARY CARD LABEL.              OPTB0020
                                ENTRY   OPTBL           ENTRY TO POINTER WORD.          OPTB0040
                    00002                                                               OPTB0050

  LINKAGE DIRECTOR
00000   000000000000
00001   464763224360

00002   0 00104 0 00003   OPTBL PZE     *+1,0,LTH       CONTROL WORD.                   OPTB0060
                                                                                        OPTB0070
00003   000000212343              BCI     1,000ACL        CAP MNEMONIC.                 OPTB0080
00004  +036100000000              OCT     036100C000C00   7090 INSTRUCTION.            OPTB0090
00005   000000214521              BCI     1,CCOANA                                      OPTB0100
00006  -032000000000              OCT     432000C000000                                 OPTB0110
00007   000000232143              BCI     1,000CAL                                      OPTB0120
00010  -050000000000              OCT     450000000000                                  OPTB0130
00011   000000233062              BCI     1,C00CHS                                      OPTB0140
00012  +076000000002              OCT     076000000C02                                  OPTB0150
00013   000000234321              BCI     1,C00CLA                                      OPTB0160
00014  +050000000000              OCT     050000000000                                  OPTB0170
00015   000000234362              BCI     1,000CLS                                      OPTB0180
00016  +050200000000              OCT     05C20C000000                                  OPTB0190
00017   000000234644              BCI     1,000COM                                      OPTB0200
00020  +076000000006              OCT     076000000006                                  OPTB0210
00021   000000262124              BCI     1,C0CFAD                                      OPTB0220
00022  +030000000000              OCT     030000000000                                  OPTB0230
00023   000000262447              BCI     1,000FDP                                      OPTB0240
00024  +024100000000              OCT     024100000000                                  OPTB0250
00025   000000264447              BCI     1,C00FMP                                      OPTB0260
00026  +026000000000              OCT     026000000000                                  OPTB0270
00027   000000266222              BCI     1,000FSB                                      OPTB0280
00030  +030200000000              OCT     03C200000C00                                  OPTB0290
00031   000000432123              BCI     1,C00LAC                                      OPTB0300
00032  +053500400000              OCT     053500400000                                  OPTB0310
00033   000000432162              BCI     1,000LAS                                      OPTB0320
00034  -034000000000              OCT     434000C000000                                 OPTB0330
00035   000000432263              BCI     1,CC0LBT                                      OPTB0340
00036  +076000000001              OCT     076000000001                                  OPTB0350
00037   000000432450              BCI     1,000LDQ                                      OPTB0360
00040  +056000000000              OCT     056000C000000                                 OPTB0370
00041   000000432743              BCI     1,C00LGL                                      OPTB0380
00042  -076300000000              OCT     476300000000                                  OPTB0390
00043   000000432751              BCI     1,000LGR                                      OPTB0400
00044  -076500000000              OCT     476500000000                                  OPTB0410
00045   000000436721              BCI     1,C00LXA                                      OPTB0420
00046  +053400400000              OCT     053400400000                                  OPTB0430
00047   000000465121              BCI     1,0000RA                                      OPTB0440
00050  -050100000000              OCT     450100000000                                  OPTB0450
00051   000000472263              BCI     1,000PBT                                      OPTB0460
00052  -076000000001              OCT     476000000C01                                  OPTB0470
00053   000000515043              BCI     1,000RQL                                      OPTB0480
00054  -077300000000              OCT     477300000C00                                  OPTB0490
00055   000000624366              BCI     1,C00SLW                                      OPTB0500
00056  +060200000000              OCT     060200000000                                  OPTB0510
00057   000000626346              BCI     1,000STO                                      OPTB0520
```

OPERATION TABLE FOR CAP.

```
00060 +060100000000            OCT     060100000000                                    OPTB0530
00061  000000626350            BCI     1,000STQ                                        OPTB0540
00062 -060000000000            OCT     460000000C00                                    OPTB0550
00063  000000626721            BCI     1,000SXA                                        OPTB0560
00064 +063400400000            OCT     063400400000                                    OPTB0570
00065  000000633167            BCI     1,000TIX                                        OPTB0580
00066 +200001400000            OCT     200001400C00                                    OPTB0590
00067  000000634431            BCI     1,C00TMI                                        OPTB0600
00070 -012000000000            OCT     412000000000                                    OPTB0610
00071  000000634743            BCI     1,000TPL                                        OPTB0620
00072 +012000000000            OCT     012000000000                                    OPTB0630
00073  000000635047            BCI     1,C00TQP                                        OPTB0640
00074 +016200000000            OCT     016200000000                                    OPTB0650
00075  000000635121            BCI     1,000TRA                                        OPTB0660
00076 +002000000000            OCT     002000000C00                                    OPTB0670
00077  000000636267            BCI     1,C00TSX                                        OPTB0680
00100 +007400400000            OCT     007400400000                                    OPTB0690
00101  000000637125            BCI     1,000TZE                                        OPTB0700
00102 +010000000000            OCT     010000000C00                                    OPTB0710
00103  000000672321            BCI     1,C00XCA                                        OPTB0720
00104 +013100000000            OCT     013100000000                                    OPTB0730
00105  000000672343            BCI     1,000XCL                                        OPTB0740
00106 -013000000C00            OCT     413000000000                                    OPTB0750
                                                                                       OPTB0760
                 00104   LTH   EQU     *-OPTBL-1     2*(NUMBER OF ALLOWED OPERATIONS). OPTB0770
                                                                                       OPTB0780
                               END                                                     OPTB0790
```

POST PROCESSOR ASSEMBLY DATA

107 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

```
  104    LTH     2,  107
    2  OPTBL   107
```

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000004 IN HUNDREDTHS OF MINUTES.

```
                INTOP, EVALUATE INT PSEUDO-OP.

                                PCC                                                                 INTP0010
                                COUNT   99                                                          INTP0020
                                LBL     INTOP           BINARY CARD LABEL.                          INTP0040
                     00002      ENTRY   INTOP           PSEUDO-OP 'INT' EVALUATOR.                  INTP0050

  LINKAGE DIRECTOR
00000  000000000000
00001  314563464760

00002  0634 00 4 00112   INTOP SXA     RX4,4           SAVE IRS.                                   INTP0060
00003  0634 00 2 00113         SXA     RX2,2           ..                                          INTP0070
00004  0634 00 1 00066         SXA     HX1,1           ..                                          INTP0080
00005 -0500 00 4 00001         CAL     1,4             GET BUFFER ORIGIN.                          INTP0090
00006  0361 00 0 00122         ACL     =12             FORM BUFF+12.                               INTP0100
00007  0621 00 0 00014         STA     SCAN            ..                                          INTP0110
00010  0600 00 0 00115         STZ     INT             CLEAR CONVERSION.                           INTP0120
00011  0600 00 0 00117         STZ     TER             RESET ERROR MARK.                           INTP0130
00012  0600 00 0 00120         STZ     TDG             RESET DIGIT MARK.                           INTP0140
00013  0774 00 2 00012         AXT     10,2            SCAN TEN WORDS.                             INTP0150
                                                                                                    INTP0160
00014  0560 00 2 00000   SCAN  LDQ     **,2            GET BUFFER WORD.  ADDRESS IS BUFF+12.       INTP0170
00015  0774 00 4 00006         AXT     6,4             SIX CHARACTERS.                             INTP0180
00016 -0754 00 0 00000         PXD     0,0             CLEAR AC.                                   INTP0190
00017 -0763 00 0 00006         LGL     6               GET CHARACTER.                              INTP0200
00020 -0340 00 0 00121         LAS     =10             CHECK FOR DIGIT.                            INTP0210
00021  0020 00 0 00036         TRA     CHAR            MUST BE CHARACTER.                          INTP0220
00022  0020 00 0 00054         TRA     ERROR           NO CHARACTER FOR CODE TEN.                  INTP0230
00023 -0625 00 0 00120         STL     TDG             DIGIT ENCOUNTERED, SET MARK.                INTP0240
00024  0601 00 0 00116         STO     DIG             SAVE DIGIT.                                 INTP0250
00025  0500 00 0 00115         CLA     INT             PROGRAMMED MULTIPLICATION OF INT BY TEN.    INTP0260
00026  0767 00 0 00002         ALS     2               4*INT.                                      INTP0270
00027  0400 00 0 00115         ADD     INT             4*INT+INT=5*INT.                            INTP0280
00030  0767 00 0 00001         ALS     1               2*(4*INT+INT)=10*INT.                       INTP0290
00031  0361 00 0 00116         ACL     DIG             ADD DIGIT, IGNORING SIGN.                   INTP0300
00032  0601 00 0 00115         STO     INT             SAVE.                                       INTP0310
00033  2 00001 4 00016   RSCAN TIX     SCAN+2,4,1      COUNT CHARACTERS.                           INTP0320
00034  2 00001 2 00014         TIX     SCAN,2,1        COUNT WORDS.                                INTP0330
00035  0020 00 0 00102         TRA     BLANK           END OF FIELD EQUIVALENT TO BLANK.           INTP0340
                                                                                                    INTP0350
00036  0774 00 1 00010   CHAR  AXT     NBK,1           COMPARISON LOOP, GET NUMBER OF BREAKS.      INTP0360
00037 -0340 00 1 00054         LAS     TABBK+NBK,1     COMPARE WITH TABLE.                         INTP0370
00040  0020 00 0 00042         TRA     *+2             NOT THIS ONE, TRY AGAIN.                    INTP0380
00041  0020 60 1 00055         TRA*    TABBK+NBK+1,1   BREAK FOUND, GO TO IT.                      INTP0390
00042  2 00002 1 00037         TIX     *-3,1,2         NOT THIS ONE, INDEX AND TRY AGAIN.          INTP0400
00043  0020 00 0 00054         TRA     ERROR           CANT FIND BREAK, ERROR.                     INTP0410
                                                                                                    INTP0420
00044  000000000020      TABBK BCI     1,00000+        BREAK TABLE, PLUS.                          INTP0430
00045  0020 00 0 00057         TRA     PLUS            ..                                          INTP0440
00046  000000000040            BCI     1,00CC0-        MINUS.                                      INTP0450
00047  0020 00 0 00062         TRA     MINUS           ..                                          INTP0460
00050  000000000073            BCI     1,00000,        COMMA.                                      INTP0470
00051  0020 00 0 00066         TRA     COMMA           ..                                          INTP0480
00052  000000000060            BCI     1,00000         BLANK.                                      INTP0490
00053  0020 00 0 00102         TRA     BLANK           ..                                          INTP0500
                     00010     NBK    EQU     *-TABBK         LENGTH OF BREAK TABLE.              INTP0510
                                                                                                    INTP0520
```

```
                INTOP, EVALUATE INT PSEUDO-OP.

00054  0055 00  000004   ERROR SIR     4          MARK INTOP ERROR.                          INTP0530
00055 -0625 00 0 00117         STL     TER        MARK ERROR IN THIS WORD.                   INTP0540
00056  0020 00 0 00033         TRA     RSCAN      RESUME SCAN.                               INTP0550
                                                                                           INTP0560
00057  0520 00 0 00120   PLUS  ZET     TDG        PLUS SIGN, ILLEGAL AFTER DIGIT.            INTP0570
00060  0020 00 0 00054         TRA     ERROR      NG, GO MARK ERROR.                         INTP0580
00061  0020 00 0 00033         TRA     RSCAN      OK, IGNORE PLUS.                           INTP0590
                                                                                           INTP0600
00062  0520 00 0 00120   MINUS ZET     TDG        MINUS SIGN, ILLEGAL AFTER DIGIT.           INTP0610
00063  0020 00 0 00054         TRA     ERROR      ..                                         INTP0620
00064  0502 00 0 00115         CLS     INT        IF LEGAL, CHANGE SIGN OF INT.              INTP0630
00065  0020 00 0 00032         TRA     RSCAN-1    ..                                         INTP0640
                                                                                           INTP0650
00066  0774 00 1 00000   COMMA AXT     **,1       FIELD MARK, STORE THIS WORD, AND,          INTP0660
00067  0500 00 0 00115         CLA     INT        PREPARE FOR NEXT WORD.                     INTP0670
00070  0767 00 0 00022         ALS     18         ..                                         INTP0680
00071  0520 00 0 00117         ZET     TER        TEST FOR ERROR IN THIS WORD.               INTP0690
00072 -0754 00 0 00000         PXD     0,0        YES, CONVERSION IS ZERO.                   INTP0700
00073  0600 00 0 00117         STZ     TER        RESET ERROR MARK.                          INTP0710
00074  0601 00 1 00000         STO     0,1        ..                                         INTP0720
00075  0600 00 0 00115         STZ     INT        ..                                         INTP0730
00076  0600 00 0 00120         STZ     TDG        RESET DIGIT MARK.                          INTP0740
00077  1 77777 1 00100         TXI     *+1,1,-1   ..                                         INTP0750
00100  0634 00 1 00066         SXA     HX1,1      ..                                         INTP0760
00101  0020 00 0 00033         TRA     RSCAN      ..                                         INTP0770
                                                                                           INTP0780
                 00066   HX1   SYN     COMMA      STORAGE FOR IR1 IS IN COMMA.               INTP0790
                                                                                           INTP0800
00102  0534 00 1 00066   BLANK LXA     HX1,1      END OF FIELD MARK, STORE THIS WORD,        INTP0810
00103  0500 00 0 00115         CLA     INT        AND PREPARE TO EXIT.                       INTP0820
00104  0767 00 0 00022         ALS     18         ..                                         INTP0830
00105  0520 00 0 00117         ZET     TER        TEST FOR ERROR IN THIS WORD.               INTP0840
00106 -0754 00 0 00000         PXD     0,0        YES, CLEAR CONVERSION.                     INTP0850
00107  0600 00 0 00117         STZ     TER        RESET ERROR MARK.                          INTP0860
00110  0601 00 1 00000         STO     0,1        ..                                         INTP0870
00111  1 77777 1 00112         TXI     *+1,1,-1   COUNT LAST WORD CONVERTED.                 INTP0880
                                                                                           INTP0890
00112  0774 00 4 00000   RX4   AXT     **,4       RESTORE IRS.                               INTP0900
00113  0774 00 2 00000   RX2   AXT     **,2       ..                                         INTP0910
00114  0020 00 4 00002         TRA     2,4        EXIT TO CALLER.                            INTP0920
                                                                                           INTP0930
00115  0 00000 0 00000   INT   PZE                STORAGE FOR CONVERSION.                    INTP0940
00116  0 00000 0 00000   DIG   PZE                STORAGE FOR DIGIT.                         INTP0950
00117  0 00000 0 00000   TER   PZE                MARK FOR ERROR THIS WORD.                  INTP0960
00120  0 00000 0 00000   TDG   PZE                MARK FOR DIGIT ENCOUNTERED THIS FIELD.     INTP0970
                                                                                           INTP0980
                               END                                                         INTP0990


  LITERALS
00121  000000000012
00122  000000000014
```

```
                INTOP, EVALUATE INT PSEUDO-OP.
                   POST PROCESSOR ASSEMBLY DATA

      123 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
       116     DIG     24,   31
        66     HX1      4,  100,  102
       115     INT     10,   25,   27,   32,   64,   67,   75,  103
        10     NBK     36,   37,   41,   54
       113     RX2      3
       112     RX4      2
       120     TDG     12,   23,   57,   62,   76
       117     TER     11,   55,   71,   73,  105,  107
        36    CHAR     21
        57    PLUS     45
        14    SCAN      7,   33,   34
       102   BLANK     35,   53
        66   COMMA     51,  102
        54   ERROR     22,   43,   60,   63
         2   INTOP
        62   MINUS     47
        33   RSCAN     56,   61,   65,  101
        44   TABBK     37,   41,   54


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000005 IN HUNDREDTHS OF MINUTES.
```

UTILITY PROGRAMS FOR CAP.

```
                        PCC                                                          UTIL0010
                        COUNT   114                                                  UTIL0020
                        LBL     UTIL            BINARY CARD LABEL.                   UTIL0040
  00024                 ENTRY   COMMA           COUNT COMMAS IN VARIABLE FIELD.      UTIL0050
  00052                 ENTRY   SAVE            ENTER WORDS IN PUSH-DOWN LIST.       UTIL0060
  00074                 ENTRY   UNSAVE          REMOVE WORDS FROM PUSH-DOWN LIST.    UTIL0070
  00002                 ENTRY   SCAN            ONE WORD EDITOR.                     UTIL0080
                                                                                     UTIL0090
                              $SCAN, REMOVE BLANKS FROM THE                          UTIL0100
                        WORD IN LAC, AND COMPRESS TO RIGHT.                          UTIL0110
                                                                                     UTIL0120

  LINKAGE DIRECTOR
C0000  000000000000
00001  234644442160

00002  0634 00 4 00021   SCAN  SXA     SCX4,4          SAVE IR4.                    UTIL0130
C0003 -0130 00 0 00000         XCL                     PLACE WORD IN MQ.            UTIL0140
00004  0774 00 4 00006         AXT     6,4             COUNT SIX CHARACTERS.        UTIL0150
00005  0600 00 0 00023         STZ     WORD            CLEAR COMPRESSED WORD.       UTIL0160
00006 -0754 00 0 00000   SLOOP PXD     0,0             CLEAR AC.                    UTIL0170
00007 -0763 00 0 00006         LGL     6               GET CHARACTER.               UTIL0180
00010 -0340 00 0 01107         LAS     =060            CHECK FOR BLANK.             UTIL0190
00011  0020 00 0 00013         TRA     *+2             NO, SKIP.                    UTIL0200
00012  0020 00 0 00017         TRA     RSCAN           YES, IGNORE.                 UTIL0210
00013 -0765 00 0 00006         LGR     6               NO BLANK, BUILD WORD.        UTIL0220
00014 -0500 00 0 00023         CAL     WORD            ..                           UTIL0230
00015 -0763 00 0 00006         LGL     6               ..                           UTIL0240
00016  0602 00 0 00023         SLW     WORD            SAVE PARTIAL WORD.           UTIL0250
00017  2 00001 4 00006   RSCAN TIX     SLOOP,4,1       COUNT CHARACTERS.            UTIL0260
00020 -0500 00 0 00023         CAL     WORD            GET COMPRESSED WORD.         UTIL0270
00021  0774 00 4 00000   SCX4  AXT     **,4            RESTORE IR4.                 UTIL0280
00022  0020 00 4 00001         TRA     1,4             RETURN WITH RESULT IN AC.    UTIL0290
                                                                                     UTIL0300
00023  0 00000 0 00000   WORD  PZE                     STORAGE FOR PARTIAL WORD.    UTIL0310
                                                                                     UTIL0320
                        END OF SCAN.                                                 UTIL0330
```

```
                        EJECT
                                                                                    UTIL0340
                              $COMMA, COUNT COMMAS IN VARIABLE                      UTIL0350
                        FIELD PLUS ONE TO FIRST BLANK OR COLUMN                     UTIL0360
                        72. COUNT IS SUBTRACTED FROM IR1.                           UTIL0370
                                                                                    UTIL0380
00024  0634 00 4 00047  COMMA SXA    COX4,4          SAVE IRS.                    UTIL0390
00025  0634 00 2 00050         SXA    COX2,2          ..                           UTIL0400
00026 -0500 00 4 00001         CAL    1,4             GET BUFFER ADDRESS.          UTIL0410
00027  0361 00 0 01106         ACL    =12             PLUS 12.                     UTIL0420
00030  0621 00 0 00033         STA    LDQ             STA IN PICKUP.               UTIL0430
00031  0774 00 4 00012         AXT    10,4            SCAN BUFF+2 TO BUFF+12.      UTIL0440
00032  0774 00 2 00006  CLP4   AXT    6,2             SIX CHARACTERS.              UTIL0450
00033  0560 00 4 00000  LDQ    LDQ    **,4            GET WORD.                    UTIL0460
00034 -0754 00 0 00000  CLP2   PXD    0,0             CLEAR AC.                    UTIL0470
00035 -0763 00 0 00006         LGL    6               GET CHARACTER.               UTIL0480
00036 -0340 00 0 01110         LAS    =HC0000,        CHECK FOR COMMA.             UTIL0490
00037  0020 00 0 00044         TRA    RCLP            NO, CANT BE BLANK.           UTIL0500
00040  1 77777 1 00044         TXI    RCLP,1,-1       YES, COUNT AND RESUME SCAN.  UTIL0510
00041 -0340 00 0 01107         LAS    =HC0000         CHECK FOR BLANK.             UTIL0520
00042  0020 00 0 00044         TRA    *+2             NO, SKIP.                    UTIL0530
00043  0020 00 0 00046         TRA    ECSCN           END OF COMMA SCAN.           UTIL0540
00044  2 00001 2 00034  RCLP   TIX    CLP2,2,1        COUNT CHARACTERS.            UTIL0550
00045  2 00001 4 00032         TIX    CLP4,4,1        COUNT WORDS.                 UTIL0560
                                                                                    UTIL0570
00046  1 77777 1 00047  ECSCN  TXI    *+1,1,-1        COUNT LAST BLANK OR E.O.F.   UTIL0580
00047  0774 00 4 00000  COX4   AXT    **,4            RESTORE IRS.                 UTIL0590
00050  0774 00 2 00000  COX2   AXT    **,2            ..                           UTIL0600
00051  0020 00 4 00002         TRA    2,4             RETURN.                      UTIL0610
                                                                                    UTIL0620
                        END OF COMMA.                                               UTIL0630
```

```
                    UTILITY PROGRAMS FOR CAP.

                                EJECT
                                          $SAVE AND $UNSAVE, PUSHDOWN LIST.                    UTIL0640
                                                                                               UTIL0650
                                                                                               UTIL0660
                                                                                               UTIL0670
00052  0634 00 4 00071   SAVE  SXA   SVX4,4        SAVE IRS.                                   UTIL0680
00053  0634 00 2 00072         SXA   SVX2,2        ..                                          UTIL0690
00054 -0500 00 4 00001         CAL   1,4           GET CONTROL WORD.                           UTIL0700
00055 -0734 00 2 00000         PDX   0,2           COUNT TC IR2.                               UTIL0710
00056  0754 00 2 00000         PXA   0,2           COUNT TC A(AC).                             UTIL0720
00057  0361 00 4 00001         ACL   1,4           (ADDRESS OF FIRST)+COUNT.                   UTIL0730
00060  0621 00 0 00062         STA   *+2           STA IN PICKUP.                              UTIL0740
00061  0774 00 4 00764   SCNT  AXT   SVN,4         CURRENT STORAGE COUNT TO IR4.               UTIL0750
00062 -0500 00 2 00000         CAL   **,2        GET WORD. **= BES OF CURRENT BLOCK            UTIL0760
00063  0602 00 4 01106         SLW   SBUFF+SVN,4   PLACE IN LIST.                             UTIL0770
00064  2 00001 4 00067         TIX   *+3,4,1       COUNT LIST.                                 UTIL0780
00065 -0055 00  000002         SIL   2             LIST EXCEEDED, SET INDICATOR,               UTIL0790
00066  0020 00 0 00071         TRA   SVX4          AND EXIT.                                   UTIL0800
00067  2 00001 2 00062         TIX   *-5,2,1       COUNT WORDS TRANSMITTED.                    UTIL0810
00070  0634 00 4 00061         SXA   SCNT,4        SAVE LIST COUNT.                            UTIL0820
00071  0774 00 4 00000   SVX4  AXT   **,4          RESTORE IRS.                                UTIL0830
00072  0774 00 2 00000   SVX2  AXT   **,2          ..                                          UTIL0840
00073  0020 00 4 00002         TRA   2,4           RETURN.                                     UTIL0850
                                                                                               UTIL0860
00074  0634 00 4 00117 UNSAVE SXA   UNSX4,4       SAVE IRS.                                   UTIL0870
00075  0634 00 2 00120         SXA   UNSX2,2       ..                                          UTIL0880
00076 -0500 00 4 00001         CAL   1,4           GET CONTROL WORD.                           UTIL0890
00077  0622 00 0 00115         STD   UNTXL         INSERT COUNT.                               UTIL0900
00100  0771 00 0 00022         ARS   18            COUNT TC A(AC).                             UTIL0910
00101  0361 00 4 00001         ACL   1,4           (ADDRESS OF FIRST)+COUNT.                   UTIL0920
00102  0621 00 0 00113         STA   UNSLW         STA IN STORE.                               UTIL0930
00103  0774 00 2 00001         AXT   1,2           SETUP FOR WORD COUNT.                       UTIL0940
00104  0534 00 4 00061         LXA   SCNT,4        LIST COUNT TO IR4.                          UTIL0950
00105  1 00001 4 00106  UNSLP  TXI   *+1,4,1       COUNT IN LIST.                              UTIL0960
00106 -3 00764 4 00111         TXL   *+3,4,SVN     IS LIST EXCEEDED.                           UTIL0970
00107 -0055 00  000004         SIL   4             YES, SET INDICATOR,                         UTIL0980
00110  0020 00 0 00117         TRA   UNSX4         AND EXIT.                                   UTIL0990
00111 -0500 00 4 01106         CAL   SBUFF+SVN,4   OK, GET WORD,                               UTIL1000
00112  0600 00 4 01106         STZ   SBUFF+SVN,4   AND CLEAR LIST.                             UTIL1010
00113  0602 00 2 00000  UNSLW SLW   **,2          INSERT IN CALLING PROGRAM.                  UTIL1020
00114  1 00001 2 00115         TXI   *+1,2,1       COUNT WORDS TRANSMITTED.                    UTIL1030
00115 -3 00000 2 00113  UNTXL TXL   UNSLW,2,**    COMPARE WITH BLOCK LENGTH.                  UTIL1040
00116  0634 00 4 00061         SXA   SCNT,4        SAVE LIST COUNT.                            UTIL1050
00117  0774 00 4 00000  UNSX4 AXT   **,4          RESTORE IRS.                                UTIL1060
00120  0774 00 2 00000  UNSX2 AXT   **,2          ..                                          UTIL1070
00121  0020 00 4 00002         TRA   2,4           RETURN.                                     UTIL1080
                                                                                               UTIL1090
                       00764  SVN   EQU   500     LENGTH OF SAVE LIST.                        UTIL1100
00122                  SBUFF BSS   SVN           LIST BUFFER.                                UTIL1110
                                                                                               UTIL1120
                                          END OF SAVE AND UNSAVE.                              UTIL1130
                                                                                               UTIL1140
                                END

  LITERALS
01106  000000000014
01107  000000000060
01110  000000000073
```

UTILITY PROGRAMS FOR CAP.
POST PROCESSOR ASSEMBLY DATA

1111 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

33	LDQ	30
764	SVN	61, 63, 106, 111, 112, 122, 1106
34	CLP2	44
32	CLP4	45
50	COX2	25
47	COX4	24
44	RCLP	37, 40
52	SAVE	
2	SCAN	
61	SCNT	70, 104, 116
21	SCX4	2
72	SVX2	53
71	SVX4	52, 66
23	WORD	5, 14, 16, 20
24	COMMA	
46	ECSCN	43
17	RSCAN	12
122	SBUFF	63, 111, 112
6	SLOCP	17
105	UNSLP	
113	UNSLW	102, 115
120	UNSX2	75
117	UNSX4	74, 110
115	UNTXL	77
74	UNSAVE	

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000006 IN HUNDREDTHS OF MINUTES.

```
$SYMSTO, AND $SYMGET, OPERATIONS WITH SYMBOL TABLE.

                        PCC                                                                     SYMS0010
                        COUNT   54                                                              SYMS0020
                        LBL     SYMSTO          BINARY CARD LABEL.                              SYMS0040
 00026                  ENTRY   SYMGET          ENTRY TO LOOK-UP VALUE OF SYMBOL.               SYMS0050
 00042                  ENTRY   PSMTBL          POINTER TO SYMBOL TABLE AND SIZE.               SYMS0060
 00003                  ENTRY   SYMSTO          ENTRY TO PLACE SYMBOL AND VALUE IN TABLE.       SYMS0070
                                                                                                SYMS0080
                                $SYMSTO, FORM SYMBOL TABLE.                                     SYMS0090
                                                                                                SYMS0100

  TRANSFER VECTOR
00000  622321456060      SCAN

  LINKAGE DIRECTOR
00001  000000000000
00002  627044272563

00003 -0340 00 0 00353  SYMSTO LAS     =H              CHECK FOR BLANK LOCATION FIELD.         SYMS0110
00004  0020 00 0 00006          TRA     *+2             NOT BLANK, SKIP.                        SYMS0120
00005  0020 00 4 00001          TRA     1,4             BLANK, DONT STORE, RETURN TO CALLER.    SYMS0130
00006  0634 00 4 00024          SXA     SSX4,4          SAVE IR4.                               SYMS0140
00007  0074 00 4 00000          TSX     $SCAN,4         COMPRESS SYMBOL TO RIGHT.               SYMS0150
00010 -0130 00 0 00000          XCL                     PLACE SYMBOL IN MQ.                     SYMS0160
00011  0754 00 1 00000          PXA     0,1             GET +(ILC).                             SYMS0170
00012  0737 00 4 00000          PAC     0,4             ..                                      SYMS0180
00013  0754 00 4 00000          PXA     0,4             ..                                      SYMS0190
00014 -0534 00 4 00042          LXD     PSMTBL,4        GET CURRENT COUNT OF TABLE.             SYMS0200
00015  1 00002 4 00016          TXI     *+1,4,2         MAKE ROOM FOR ONE MORE.                 SYMS0210
00016 -3 00310 4 00021          TXL     *+3,4,LSMTBL    CHECK FOR TABLE OVERFLOW.               SYMS0220
00017 -0055 00  000001          SIL     1               SYMTBL EXCEEDED, SET INDICATOR.         SYMS0230
00020  0020 00 0 00024          TRA     SSX4            GO TO RETURN.                           SYMS0240
00021 -0600 00 4 00353          STQ     SYMTBL,4        SAVE SYMBOL.                            SYMS0250
00022  0602 00 4 00354          SLW     SYMTBL+1,4      SAVE VALUE.                             SYMS0260
00023 -0634 00 4 00042          SXD     PSMTBL,4        SAVE TABLE COUNT.                       SYMS0270
00024  0774 00 4 00000   SSX4   AXT     **,4            RESTORE IR4.                            SYMS0280
00025  0020 00 4 00001          TRA     1,4             RETURN.                                 SYMS0290

                        SPACE   2                                                               SYMS0300
                                $SYMGET, LOOK UP SYMBOL AND GET VALUE.                          SYMS0310
                                                                                                SYMS0320
00026  0634 00 4 00040  SYMGET SXA     SGX4,4          SAVE IR4.                               SYMS0330
00027 -0534 00 4 00042          LXD     PSMTBL,4        GET TABLE COUNT.                        SYMS0340
00030 -0340 00 4 00353          LAS     SYMTBL,4        COMPARE WITH TABLE.                     SYMS0350
00031  0020 00 0 00033          TRA     *+2             NOT THIS ONE, SKIP.                     SYMS0360
00032  0020 00 0 00037          TRA     SYMFND          FOUND, EXIT.                            SYMS0370
00033  2 00002 4 00030          TIX     *-3,4,2         INDEX AND TRY AGAIN.                    SYMS0380
                                                                                                SYMS0390
00034 -0754 00 0 00000          PXD     0,0             NOT FOUND, VALUE IS ZERO.               SYMS0400
00035  0055 00  000001          SIR     1               SET UNDEFINED SYMBOL INDICATOR.         SYMS0410
00036  0020 00 0 00040          TRA     SGX4            GO TO EXIT.                             SYMS0420
                                                                                                SYMS0430
00037 -0500 00 4 00354  SYMFND CAL     SYMTBL+1,4      FOUND, GET VALUE.                       SYMS0440
00040  0774 00 4 00000   SGX4   AXT     **,4            RESTORE IR4.                            SYMS0450
00041  0020 00 4 00001          TRA     1,4             RETURN.                                 SYMS0460
```

```
                $SYMSTO, AND $SYMGET, OPERATIONS WITH SYMBOL TABLE.

                                SPACE   2                                                       SYMS0470
                                        STORAGE AND CONSTANTS.                                  SYMS0480
                                                                                                SYMS0490
                    00310   LSMTBL EQU      2*100           ROOM FOR 100 SYMBOLS.           SYMS0500
00042  0 00000 0 00353  PSMTBL PZE      SYMTBL,0,**     POINTER WORD TO SYMTBL.         SYMS0510
00353                   SYMTBL BES      LSMTBL          SYMBOL TABLE.                   SYMS0520
                                                                                                SYMS0530
                                END                                                             SYMS0540

  LITERALS
00353  606060606060
```

```
                    POST PROCESSOR ASSEMBLY DATA

      354 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
        0   SCAN     7
       40   SGX4    26,   36
       24   SSX4     6,   20
      310 LSMTBL    16,   42,  353
       42 PSMTBL    14,   23,   27
       37 SYMFND    32
       26 SYMGET
        3 SYMSTO
      353 SYMTBL    21,   22,   30,   37,   42


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000004 IN HUNDREDTHS OF MINUTES.
```

ENDOP AND OTHER SUBROUTINES USED BY COMP.

```
                        PCC                                                                 ENDP0010
                        COUNT   102                                                         ENDP0020
                        LBL     ENDOP       BINARY CARD LABEL.                              ENDP0040
00036                   ENTRY   PIVAR       ENTRY TO PLACE WORD IN VARIABLE FIELD.       ENDP0050
00045                   ENTRY   GENOP       ENTRY TO PLACE OP IN OP-FIELD AND WCT1.      ENDP0060
00053                   ENTRY   ILC         ENTRY LOCATION FOR SAVING CURRENT ILC.       ENDP0070
00071                   ENTRY   GNSTO       ENTRY TO GENERATE TEMPORARY STORAGE.         ENDP0080
00060                   ENTRY   ERASE       ENTRY TO ERASE VARIABLE FIELD.               ENDP0090
00122                   ENTRY   NSTO        ENTRY TO COUNT LOCATIONS FOR ERASABLES.      ENDP0100
00004                   ENTRY   ENDOP       END CARD PSEUDO-OP.                          ENDP0110
                                                                                            ENDP0120
                                ENTRY TO RESERVE STORAGE.                                   ENDP0130
                                                                                            ENDP0140

  TRANSFER VECTOR
00000  627044626346     SYMSTO
00001  662363016060     WCT1

  LINKAGE DIRECTOR
00002  000000000000
00003  473165215160

00004 -0520 00 0 00123  ENDOP NZT   MSTO          IF NO STORAGE ALLOCATED,                ENDP0150
00005  0020 00 4 00001         TRA   1,4          RETURN TO CALLER.                       ENDP0160
00006  0634 00 4 00016         SXA   ENDX4,4      STORAGE ALLOCATED, SAVE IR4.            ENDP0170
00007 -0500 00 0 00150         CAL   =HTEM        INSERT THIS SYMBOL IN SYMBOL TABLE.     ENDP0180
00010  0074 00 4 00000         TSX   $SYMSTO,4    ..                                      ENDP0190
00011  0074 00 4 00001         TSX   $WCT1,4      PUT REM CARD ON CT1.                    ENDP0200
00012  0 00000 0 00020         PZE   EBUFF        ..                                      ENDP0210
00013  0535 00 4 00123         LAC   MSTO,4       INCREASE ILC FOR STORAGE.               ENDP0220
00014 -0634 00 4 00015         SXD   *+1,4        ..                                      ENDP0230
00015  1 00000 1 00016         TXI   *+1,1,**     ..                                      ENDP0240
00016  0774 00 4 00000  ENDX4  AXT   **,4         RESTORE IR4.                            ENDP0250
00017  0020 00 4 00001         TRA   1,4          RETURN.                                 ENDP0260
                                                                                           ENDP0270
00020  606325446060     EBUFF BCI   9, TEM   REM  TEMPORARY STORAGE AREA BEGINS HERE.     ENDP0280
00021  605125446060
00022  632544474651
00023  215170606263
00024  465121272560
00025  215125216022
00026  252731456260
00027  302551253360
00030  606060606060
00031  606060606060         BCI   5,                                                     ENDP0290
00032  606060606060
00033  606060606060
00034  606060606060
00035  606060606060
                                                                                           ENDP0300
                                ENTRY TO FORM VARIABLE FIELD.                              ENDP0310
                                                                                           ENDP0320
00036  0634 00 4 00043  PIVAR SXA   PX4,4         SAVE IR4.                               ENDP0330
00037  0774 00 4 00013  PCNT  AXT   11,4         COUNT 10 WORDS WITH TNX.                ENDP0340
00040 -2 00001 4 00043         TNX   PX4,4,1      INDEX WORD COUNT.                       ENDP0350
00041  0602 00 4 00141         SLW   PBUFF+12,4   PLACE WORD IN BUFFER.                   ENDP0360
```

```
00042  0634 00 4 00037          SXA     PCNT,4          SAVE WORD COUNT.                    ENDP0370
00043  0774 00 4 00000  PX4     AXT     **,4            RESTORE IR4.                        ENDP0380
00044  0020 00 4 00001          TRA     1,4             RETURN.                             ENDP0390

                                SPACE   2                                                   ENDP0400
                                ENTRY TO INSERT OP-FIELD AND WCT1.                          ENDP0410
                                                                                            ENDP0420
00045  0634 00 4 00056  GENOP   SXA     GOPX4,4         SAVE IR4.                           ENDP0430
00046 -0500 00 4 00001          CAL     1,4             GET OP.                             ENDP0440
00047  0602 00 0 00126          SLW     PBUFF+1         INSERT OP-FIELD.                    ENDP0450
00050  0074 00 4 00001          TSX     $WCT1,4         WRITE COLLATION TAPE.               ENDP0460
00051  0 00000 0 00125          PZE     PBUFF           ..                                  ENDP0470
00052  0074 00 4 00060          TSX     ERASE,4         CLEAR PBUFF.                        ENDP0480
00053  0774 00 4 00000  ILC     AXT     **,4            INCREMENT ILC.                      ENDP0490
00054  1 77777 4 00055          TXI     *+1,4,-1        ..                                  ENDP0500
00055  0634 00 4 00053          SXA     ILC,4           SAVE CURRENT ILC.                   ENDP0510
00056  0774 00 4 00000  GOPX4   AXT     **,4            RESTORE IR4.                        ENDP0520
00057  0020 00 4 00002          TRA     2,4             RETURN.                             ENDP0530

                                SPACE   2                                                   ENDP0540
                                ENTRY TO ERASE PBUFF.                                       ENDP0550
                                                                                            ENDP0560
00060  0634 00 4 00067  ERASE   SXA     ERX4,4          SAVE IR4.                           ENDP0570
00061  0774 00 4 00013          AXT     11,4            RESET PCNT.                         ENDP0580
00062  0634 00 4 00037          SXA     PCNT,4          ..                                  ENDP0590
00063  0774 00 4 00016          AXT     14,4            LOAD BUFFER WITH BLANKS.            ENDP0600
00064 -0500 00 0 00145          CAL     =H              ..                                  ENDP0610
00065  0602 00 4 00143          SLW     PBUFF+14,4      ..                                  ENDP0620
00066  2 00001 4 00065          TIX     *-1,4,1         ..                                  ENDP0630
00067  0774 00 4 00000  ERX4    AXT     **,4            RESTORE IR4.                        ENDP0640
00070  0020 00 4 00001          TRA     1,4             RETURN.                             ENDP0650

                                SPACE   2                                                   ENDP0660
                                ENTRY TO GET NEXT TEMPORARY STORAGE SYMBOL.                 ENDP0670
                                                                                            ENDP0680
00071  0500 00 0 00122  GNSTO   CLA     NSTO            PLACE NUMBER OF LAST STORAGE.       ENDP0690
00072  0560 00 0 00122          LDQ     NSTO            IN AC AND MQ.                       ENDP0700
00073  0400 00 0 00143          ADD     =1              INCREMENT AND SAVE FOR NEXT.        ENDP0710
00074  0601 00 0 00122          STO     NSTO            ..                                  ENDP0720
00075  034C 00 0 00123          CAS     MSTO            CHECK FOR MSTO EXCEEDED.            ENDP0730
00076  0601 00 0 00123          STO     MSTO            YES, UPDATE MSTO.                   ENDP0740
00077  0761 00 0 00000          NOP                     EQUAL, IGNORE.                      ENDP0750
00100  0131 00 0 00000          XCA                     PLACE NSTO IN AC.                   ENDP0760
00101 -0100 00 0 00104          TNZ     *+3             CHECK FOR ZERO NSTO.                ENDP0770
00102 -0500 00 0 00150          CAL     =HTEM           ZERO, PICKUP CHARACTERS.            ENDP0780
00103  0020 00 4 00001          TRA     1,4             RETURN TO CALLER.                   ENDP0790
00104  0340 00 0 00144          CAS     =10             CHECK FOR ONLY ONE DIGIT.           ENDP0800
00105  0020 00 0 00112          TRA     TWODG           TWO DIGITS.                         ENDP0810
00106  0020 00 0 00112          TRA     TWODG           ..                                  ENDP0820
00107  0767 00 0 00006          ALS     6               ONE DIGIT, SHIFT TO POSITION.       ENDP0830
00110 -0501 00 0 00147          ORA     =HTEM+0         INSERT CHARACTERS.                  ENDP0840
00111  0020 00 4 00001          TRA     1,4             RETURN TO CALLER.                   ENDP0850
00112  0131 00 0 00000  TWODG   XCA                     TWO DIGITS, PLACE NSTO IN MQ AGAIN. ENDP0860
```

```
                ENDOP AND OTHER SUBROUTINES USED BY COMP.

00113 -0754 00 0 00000          PXD     0,C             CLEAR AC.                       ENDP0870
00114  0221 00 0 00144          DVP     =10             MOD 10.                         ENDP0880
00115 -0773 00 0 00006          RQL     6               FIRST DIGIT TO K5.              ENDP0890
00116 -0600 00 0 00124          STQ     DNSTO           SAVE.                           ENDP0900
00117 -0501 00 0 00124          ORA     DNSTO           FORM DECIMAL STORAGE NUMBER.    ENDP0910
00120 -0501 00 0 00146          ORA     =HTEM+00        INSERT CHARACTERS.              ENDP0920
00121  0020 00 4 00001          TRA     1,4             RETURN WITH RESULT IN AC.       ENDP0930

                                SPACE   2                                               ENDP0940
                                STORAGE AND CONSTANTS.                                  ENDP0950
                                                                                        ENDP0960
00122  0 00000 0 00000  NSTO    PZE                     CURRENT STORAGE COUNTER.        ENDP0970
00123  0 00000 0 00000  MSTO    PZE                     MAXIMUM STORAGE COUNTER.        ENDP0980
00124  0 00000 0 00000  DNSTO   PZE                     DECIMAL STORAGE COUNTER.        ENDP0990
00125                   PBUFF   BSS     14              STATEMENT BUFFER.               ENDP1000
                                                                                        ENDP1010
                                END                                                     ENDP1020

  LITERALS
00143  000CC0000001
00144  000000000012
00145  606060606060
00146  632544200C00
00147  632544200060
00150  632544606060
                POST PROCESSOR ASSEMBLY DATA

  151 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
      53    ILC     55
      43    PX4     36,   40
      67    ERX4    60
     123    MSTO     4,   13,   75,   76
     122    NSTO    71,   72,   74
      37    PCNT    42,   62
       1    WCT1    11,   50
     124    DNSTO  116,  117
      20    EBUFF   12
       4    ENDOP
      16    ENDX4    6
      60    ERASE   52
      45    GENOP
      71    GNSTO
      56    GOPX4   45
     125    PBUFF   41,   47,   51,   65
      36    PIVAR
     112    TWODG  105,  106
       0 SYMSTO    10


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000005 IN HUNDREDTHS OF MINUTES.
```

```
SUBROUTINE COMPOP, COMPILE ARITHMETICS FOR CAP.

                PCC                                                                    COMP0010
                COUNT   186                                                            COMP0020
                LBL     COMPOP          BINARY CARD LABEL.                             COMP0040
 00010          ENTRY   COMPOP          EVALUATE 'COMP' PSEUDO-OP.                     COMP0050
                                                                                       COMP0060
        *         $COMPOP IS CALLED BY,                                                COMP0070
        *                                                                              COMP0080
        *               TSX $COMPOP,4                                                  COMP0090
        *               PZE BUFF                                                       COMP0100
        *                                                                              COMP0110
        *       WHERE BUFF IS A 14 WORD BUFFER CONTAINING THE                          COMP0120
        *       HOLLERITH CARD IMAGE OF THE COMP STATEMENT.                            COMP0130
        *       COMPOP TAKES THE VARIABLE FIELD AS A FORTRAN                           COMP0140
        *       ARITHMETIC STATEMENT AND COMPILES IN FLOATING                          COMP0150
        *       POINT.  COMMAS ARE TREATED AS PART OF THE SYMBOL,                      COMP0160
        *       HENCE TAGGING IS ALLOWED.  BLANKS ARE IGNORED.                         COMP0170
        *                                                                              COMP0180
        *         COMPOP OPERATES IN 2 PASSES. PASS1 TAKES THE                         COMP0190
        *       CARD IMAGE APART SEPARATING SYMBOLS FROM OPERATION                     COMP0200
        *       CHARACTERS.  PASS2 EVALUATES EXPRESSIONS FROM THE                      COMP0210
        *       INNERMOST (...) PAIR OUTWARD.  SOME OPTIMIZATION                       COMP0220
        *       IS DONE BUT THERE ARE NO DIAGNOSTICS.                                  COMP0230

  TRANSFER VECTOR
00000   314323606060    ILC
00001   456263466060    NSTO
00002   255121622560    ERASE
00003   256747516060    EXPR
00004   473165215160    PIVAR
00005   272545464760    GENOP

  LINKAGE DIRECTOR
00006   000000000000
00007   234644474647

00010   0634 00 4 00210 COMPOP SXA     RX4,4           SAVE IRS.                      COMP0240
00011   0634 00 2 00211        SXA     RX2,2           ..                             COMP0250
00012   0754 00 1 00000        PXA     0,1             GET -(ILC).                    COMP0260
00013   0621 60 0 00000        STA*    $ILC            SAVE.                          COMP0270
00014  -0500 00 4 00001        CAL     1,4             GET CONTROL WORD.              COMP0280
00015   0361 00 0 00536        ACL     =12             BUFF+12.                       COMP0290
00016   0621 00 0 00032        STA     CAL1            STA IN PICKUP.                 COMP0300
00017   0600 60 0 00001        STZ*    $NSTO           ZERO NUMBER OF TEMPORARY STORAGE. COMP0310
00020   0600 00 0 00216        STZ     TEOF1           RESET EOF MARK.                COMP0320
00021  -0500 00 0 00534        CAL     =1              SETUP SYM.                     COMP0330
00022   0602 00 0 00215        SLW     SYM             ..                             COMP0340
00023   0074 00 4 00002        TSX     $ERASE,4        ERASE BUFFER.                  COMP0350
00024   0774 00 4 00310        AXT     LFLD,4          SETUP FLD COUNT.               COMP0360
00025   0634 00 4 00124        SXA     FCNT,4          ..                             COMP0370
00026   0140 00 0 00027        TOV     *+1             TURN OFF OVERFLOW LIGHT.       COMP0380
00027   0020 00 0 00030        TRA     CPAS1           GO TO PASS1.                   COMP0390
```

```
                                        SPACE   2                                                                COMP0400
                                                PASS 1 OF COMP, SEPARATE FIELD INTO SYMBOL AND BREAKS.           COMP0410
                                                                                                                 COMP0420
                                                                                                                 COMP0430
00030   0774 00 2 00012   CPAS1 AXT     10,2            COUNT 10 WORDS IN VARIABLE FIELD.                      COMP0440
00031   0774 00 1 00006   BSCN1 AXT     6,1             COUNT 6 CHARACTERS.                                    COMP0450
00032  -0500 00 2 00000   CAL1  CAL     **,2            GET WORD.                                              COMP0460
00033  -0340 00 0 00542         LAS     =H              CHECK FOR ALL BLANKS.                                  COMP0470
00034   0020 00 0 00036         TRA     *+2             NO, SKIP.                                              COMP0480
00035   0020 00 0 00063         TRA     RSCN1+2         ALL BLANK, IGNORE.                                     COMP0490
00036  -0130 00 0 00000         XCL                     PLACE IN MQ.                                           COMP0500
00037  -0754 00 0 00000   SCN1  PXD     0,0             CLEAR AC.                                              COMP0510
00040  -0763 00 0 00006         LGL     6               GET NEXT CHARACTER.                                    COMP0520
00041  -0600 00 0 00217         STQ     MQ              SAVE MQ.                                               COMP0530
00042  -0340 00 0 00540         LAS     =H000C0         CHECK FOR BLANK.                                       COMP0540
00043   0020 00 0 00045         TRA     *+2             NO, SKIP.                                              COMP0550
00044   0020 00 0 00061         TRA     RSCN1           YES, IGNORE.                                           COMP0560
00045   0774 00 4 00007         AXT     NBK,4           CHECK FOR BREAK.                                       COMP0570
00046  -0340 00 4 00075         LAS     TABBK+NBK,4     COMPARE WITH TABLE.                                    COMP0580
00047   0020 00 0 00051         TRA     *+2             NO, SKIP.                                              COMP0590
00050   0020 00 0 00075         TRA     BRK             BREAK FOUND, EXIT.                                     COMP0600
00051   2 00001 4 00046         TIX     *-3,4,1         INDEX, AND TRY AGAIN.                                  COMP0610
00052  -0765 00 0 00006         LGR     6               BUILD SYM.                                             COMP0620
00053  -0500 00 0 00215         CAL     SYM             ..                                                     COMP0630
00054  -0763 00 0 00006         LGL     6               ..                                                     COMP0640
00055  -0140 00 0 00060         TNO     *+3             IF SYMBOL FULL,                                        COMP0650
00056   0074 00 4 00123         TSX     STFLD,4         INSERT IN LIST,                                        COMP0660
00057  -0500 00 0 00534         CAL     =1              AND BEGIN NEW SYMBOL.                                  COMP0670
00060   0602 00 0 00215         SLW     SYM             SAVE PARTIAL SYMBOL.                                   COMP0680
00061   0560 00 0 00217   RSCN1 LDQ     MQ              RESTORE MQ.                                            COMP0690
00062   2 00001 1 00037         TIX     SCN1,1,1        COUNT CHARACTERS.                                      COMP0700
00063   2 00001 2 00031         TIX     BSCN1,2,1       COUNT WORDS.                                           COMP0710
00064  -0625 00 0 00216         STL     TEOF1           SET EOF MARK.                                          COMP0720
00065   0020 00 0 00075         TRA     BRK             GO PROCESS BREAK.                                      COMP0730
                                                                                                                 COMP0740
00066   000000000020    TABBK BCI     1,00000+        TABLE OF BREAK CHARACTERS.                             COMP0750
00067   000000000040          BCI     1,000C0-        ..                                                     COMP0760
00070   000000000C54          BCI     1,CCCC0*        ..                                                     COMP0770
00071   000000000061          BCI     1,C00C0/        ..                                                     COMP0780
00072   000000000074          BCI     1,00000(        ..                                                     COMP0790
00073   000000000C34          BCI     1,00000)        ..                                                     COMP0800
00074   000000000C13          BCI     1,00000=        ..                                                     COMP0810
                        00007   NBK   EQU     *-TABBK         SIZE OF TABLE.                                 COMP0820
                                                                                                             COMP0830
00075   0602 00 0 00220   BRK   SLW     LBRK            SAVE BREAK CHARACTER.                                  COMP0840
00076  -0500 00 0 00215         CAL     SYM             GET SYMBOL.                                            COMP0850
00077  -0340 00 0 00534         LAS     =1              CHECK FOR NO CHARACTERS.                               COMP0860
00100   0020 00 0 00102         TRA     *+2             YES, SKIP.                                             COMP0870
00101   0020 00 0 00110         TRA     NOSYM           NO, DONT STORE.                                        COMP0880
00102   0560 00 0 00542         LDQ     =H              SYMBOL, LEFT JUSTIFY.                                  COMP0890
00103  -0763 00 0 00006         LGL     6               ..                                                     COMP0900
00104  -0140 00 0 00103         TNO     *-1             ..                                                     COMP0910
00105   0074 00 4 00123         TSX     STFLD,4         PLACE IN FLD.                                          COMP0920
00106  -0500 00 0 00534         CAL     =1              BEGIN NEW SYMBOL.                                      COMP0930
00107   0602 00 0 00215         SLW     SYM             ..                                                     COMP0940
00110   0520 00 0 00216   NOSYM ZET     TEOF1           CHECK FOR EOF.                                         COMP0950
00111   0020 00 0 00134         TRA     EOF1            EOF, GO TO IT.
```

```
                SUBROUTINE COMPOP, COMPILE ARITHMETICS FOR CAP.

00112 -0500 00 0 00220          CAL     LBRK            NOT EOF, GET BREAK CHARACTER.           COMP0960
00113  0074 00 4 00123          TSX     STFLD,4         PLACE IN FIELD.                         COMP0970
00114 -0500 00 0 00220          CAL     LBRK            GET BREAK CHARACTER.                    COMP0980
00115  0322 00 0 00535          ERA     =H00000=        CHECK FOR =.                            COMP0990
00116 -0100 00 0 00061          TNZ     RSCN1           NO, RESUME SCAN.                        COMP1000
00117 -0500 00 0 00124          CAL     FCNT            YES, MARK TOP OF FIELD.                 COMP1010
00120  0361 00 0 00534          ACL     =1              ..                                      COMP1020
00121  0621 00 0 00221          STA     TFLD            ..                                      COMP1030
00122  0020 00 0 00061          TRA     RSCN1           RESUME SCAN.                            COMP1040
                                                                                                COMP1050
00123  0634 00 4 00130  STFLD   SXA     STX4,4          SAVE IR4.                               COMP1060
00124  0774 00 4 00310  FCNT    AXT     LFLD,4          GET CURRENT FLD INDEX.                  COMP1070
00125  0602 00 4 00534          SLW     FLD,4           INSERT WORD.                            COMP1080
00126 -2 00001 4 00132          TNX     XFLD,4,1        COUNT THIS WORD.                        COMP1090
00127  0634 00 4 00124          SXA     *-3,4           SAVE COUNT.                             COMP1100
00130  0774 00 4 00000  STX4    AXT     **,4            RESTORE IR4.                            COMP1110
00131  0020 00 4 00001          TRA     1,4             RETURN.                                 COMP1120
                                                                                                COMP1130
00132 -0055 00  000010  XFLD    SIL     10              COMP STATEMENT TOO LONG, MARK ERROR,    COMP1140
00133  0020 00 0 00210          TRA     RX4             AND RETURN.                             COMP1150
                                                                                                COMP1160
00134 -0500 00 0 00124  EOF1    CAL     FCNT            MARK BOTTOM OF FIELD.                   COMP1170
00135  0621 00 0 00222          STA     BFLD            ..                                      COMP1180
00136  0020 00 0 00137          TRA     CPAS2           GO TO PASS2.                            COMP1190


                                SPACE   2                                                       COMP1200
                                PASS 2 OF COMP, FIND AND EVALUATE EXPRESSIONS.                  COMP1210
                                                                                                COMP1220
00137  0534 00 1 00222  CPAS2   LXA     BFLD,1          SET CONTROL TXH.                        COMP1230
00140 -0634 00 1 00155          SXD     RSCN2+1,1       ..                                      COMP1240
00141  0534 00 1 00221  BSCN2   LXA     TFLD,1          GET TOP OF FIELD.                       COMP1250
00142  0634 00 1 00223          SXA     LLPAR,1         TREAT AS LAST (.                        COMP1260
00143  1 77777 1 00144          TXI     **+1,1,-1       INDEX TO FIRST WORD.                    COMP1270
00144 -0500 00 1 00534  SCN2    CAL     FLD,1           GET NEXT FLD WORD.                      COMP1280
00145  0100 00 0 00154          TZE     RSCN2           IF ZERO, IGNORE.                        COMP1290
00146 -0340 00 0 00541          LAS     =HC00C00(       CHECK FOR (.                            COMP1300
00147  0020 00 0 00151          TRA     **+2            NO, SKIP.                               COMP1310
00150  0020 00 0 00157          TRA     LPAR2           YES, GO TO IT.                          COMP1320
00151 -0340 00 0 00537          LAS     =H00000)        CHECK FOR ).                            COMP1330
00152  0020 00 0 00154          TRA     *+2             NO, SKIP.                               COMP1340
00153  0020 00 0 00161          TRA     RPAR2           YES, GO TO IT.                          COMP1350
00154  1 77777 1 00155  RSCN2   TXI     **+1,1,-1       COUNT WORDS.                            COMP1360
00155  3 00000 1 00144          TXH     SCN2,1,**       CHECK FOR EOF.                          COMP1370
00156  0020 00 0 00172          TRA     EOF2            EOF, GO TO IT.                          COMP1380
                                                                                                COMP1390
00157  0634 00 1 00223  LPAR2   SXA     LLPAR,1         SET LAST (.                             COMP1400
00160  0020 00 0 00154          TRA     RSCN2           RESUME SCAN.                            COMP1410
                                                                                                COMP1420
00161 -0634 00 1 00167  RPAR2   SXD     EXPW1,1         EXPRESSION, MARK BOTTOM FOR $EXPR.      COMP1430
00162  0600 00 1 00534          STZ     FLD,1           CLEAR FLD.                              COMP1440
00163  0534 00 1 00223          LXA     LLPAR,1         GET INDEX OF LAST (.                    COMP1450
00164  0600 00 1 00534          STZ     FLD,1           CLEAR FLD.                              COMP1460
00165  0634 00 1 00167          SXA     EXPW1,1         MARK TOP FOR $EXPR.                     COMP1470
00166  0074 00 4 00003          TSX     $EXPR,4         GO EVALUATE EXPRESSION.                 COMP1480
00167  0 00000 0 00000  EXPW1   PZE     **,0,**         ZERO TAG MEANS STORE INTERMEDIATE.      COMP1490
```

```
C0170   0 00000 C 00534          PZE     FLC             ..                                      COMP1500
00171   0020 00 0 00141          TRA     BSCN2           RESTART SCAN.                           COMP1510
                                                                                                 COMP1520
00172 -0634 00 1 00177   EOF2    SXD     EXPW2,1         FINAL EXPR, MARK BOTTOM.                COMP1530
00173   0534 00 1 00223          LXA     LLPAR,1         GET INDEX OF TOP.                       COMP1540
00174   0600 00 1 00534          STZ     FLC,1           CLEAR FLD.                              COMP1550
00175   0634 00 1 00177          SXA     EXPW2,1         MARK TOP.                               COMP1560
00176   0074 00 4 00003          TSX     $EXPR,4         EVALUATE EXPRESSION.                    COMP1570
C0177   0 00000 7 00000  EXPW2   PZE     **,7,**         NON-ZERO TAG MEANS LEAVE IN AC.         COMP1580
00200   0 00000 0 00534          PZE     FLC             ..                                      COMP1590
00201   0774 00 1 00310          AXT     LFLD,1          FORM FINAL STORAGE.                     COMP1600
00202 -0500 00 1 00534           CAL     FLD,1           GET WORD.                               COMP1610
00203   0100 00 0 00206          TZE     *+3             IF ZERO, SYMBOL DONE.                   COMP1620
00204   0074 00 4 00004          TSX     $PIVAR,4        PLACE IN VARIABLE FIELD.                COMP1630
00205   1 77777 1 00202          TXI     *-3,1,-1        INDEX FOR NEXT WORD.                    COMP1640
00206   0074 00 4 00005          TSX     $GENOP,4        FINAL OP IS STO.                        COMP1650
00207   606263466060             BCI     1, STO          ..                                      COMP1660
                                                                                                 COMP1670
00210   0774 00 4 00000  RX4     AXT     **,4            RESTORE IRS.                            COMP1680
00211   0774 00 2 00000  RX2     AXT     **,2            ..                                      COMP1690
C0212 -0500 60 0 00000           CAL*    $ILC            RESTORE ILC.                            COMP1700
00213   0734 00 1 00000          PAX     0,1             ..                                      COMP1710
00214   0020 00 4 00002          TRA     2,4             RETURN.                                 COMP1720


                                 SPACE   2                                                       COMP1730
                                         STORAGE AND CONSTANTS.                                  COMP1740
                                                                                                 COMP1750
C0215   0 00000 0 00000  SYM     PZE                     PARTIAL SYMBOL STORAGE.                 COMP1760
00216   0 00000 0 00000  TEOF1   PZE                     END OF FIELD MARK.                      COMP1770
00217   0 00000 0 00000  MQ      PZE                     MQ STORAGE.                             COMP1780
00220   0 00000 0 00000  LBRK    PZE                     LAST BREAK.                             COMP1790
C0221   0 00000 0 00000  TFLD    PZE                     TOP OF FLD.                             COMP1800
00222   0 00000 0 00000  BFLD    PZE                     BOTTOM OF FLD.                          COMP1810
00223   0 00000 0 00000  LLPAR   PZE                     LAST (.                                 COMP1820
                 00310   LFLD    EQU     200             LENGTH OF FLD.                          COMP1830
C0534                    FLD     BES     LFLD            ARITHMETIC FIELD BUFFER.                COMP1840
                                                                                                 COMP1850
                                 END                                                             COMP1860
```

LITERALS

```
00534   000000000001
00535   000000000013
00536   000000000014
00537   000000000034
00540   000000000060
00541   000000000074
00542   606060606060
```

SUBROUTINE COMPOP, COMPILE ARITHMETICS FOR CAP.
POST PROCESSOR ASSEMBLY DATA

543 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

217	MQ	41, 61
75	BRK	50, 65
534	FLD	125, 144, 162, 164, 170, 174, 200, 202
0	ILC	13, 212
7	NBK	45, 46, 75
211	RX2	11
210	RX4	10, 133
215	SYM	22, 53, 60, 76, 107
222	BFLD	135, 137
32	CAL1	16
134	EOF1	111
172	EOF2	156
3	EXPR	166, 176
124	FCNT	25, 117, 134
220	LBRK	75, 112, 114
310	LFLD	24, 124, 201, 224, 534
1	NSTO	17
37	SCN1	62
144	SCN2	155
130	STX4	123
221	TFLD	121, 141
132	XFLD	126
31	BSCN1	63
141	BSCN2	171
30	CPAS1	27
137	CPAS2	136
2	ERASE	23
167	EXPW1	161, 165
177	EXPW2	172, 175
5	GENOP	206
223	LLPAR	142, 157, 163, 173
157	LPAR2	150
110	NOSYM	101
4	PIVAR	204
161	RPAR2	153
61	RSCN1	35, 44, 116, 122
154	RSCN2	140, 145, 160
123	STFLD	56, 105, 113
66	TABBK	46, 75
216	TEOF1	20, 64, 110
10	COMPOP	

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000008 IN HUNDREDTHS OF MINUTES.

```
SUBROUTINE EXPR, EVALUATE EXPRESSION FOR COMPOP.

                        PCC                                                          EXPR0010
                        COUNT   171                                                  EXPR0020
                        LBL     EXPR            BINARY CARD LABEL.                   EXPR0040
  00006                 ENTRY   EXPR            ARITHMETIC EXPRESSION EVALUATOR.     EXPR0050
                                                                                     EXPR0060
                *         $EXPR IS CALLED BY,                                        EXPR0070
                *                                                                    EXPR0080
                *              TSX $EXPR,4                                           EXPR0090
                *              PZE LI,T,RI                                           EXPR0100
                *              PZE FLD                                               EXPR0110
                *                                                                    EXPR0120
                *       WHERE (FLD-LI) IS THE ADDRESS OF THE LEFT BREAK,             EXPR0130
                *       AND (FLD-RI) IS THE ADDRESS OF THE RIGHT BREAK.              EXPR0140
                *       EXPR TAKES A STRING OF SYMBOLS CONNECTED BY                  EXPR0150
                *       + - * OR / (S) AND COMPILES THE RESULT IN                    EXPR0160
                *       FLOATING POINT.  IF T=0, THE RESULT IS PLACED IN             EXPR0170
                *       TEMPORARY STORAGE, OTHERWISE, RESULT IS IN AC.  THE          EXPR0180
                *       SYMBOLIC FIELD IS MODIFIED ACCORDINGLY.                      EXPR0190
                *                                                                    EXPR0200
                *          $EXPR OPERATES IN TWO PASSES.  PASS1 USES $TERM           EXPR0210
                *       TO REDUCE THE EXPRESSION TO A SUMMATION.  PASS2              EXPR0220
                *       PERFORMS THE SUMMATION.  SOME OPTIMIZATION                   EXPR0230
                *       IS DONE.                                                     EXPR0240

  TRANSFER VECTOR
00000   632551446060    TERM
00001   473165215160    PIVAR
00002   272545464760    GENOP
00003   274562634660    GNSTC

  LINKAGE DIRECTOR
00004   000000000000
00005   256747516060

00006   0634 00 4 00164   EXPR  SXA     RX4,4         SAVE IRS.                    EXPR0250
00007   0634 00 2 00165         SXA     RX2,2         ..                           EXPR0260
00010   0634 00 1 00166         SXA     RX1,1         ..                           EXPR0270
00011  -0500 00 4 00001         CAL     1,4           GET CONTROL WORD.            EXPR0280
00012   0622 00 0 00054         STD     RSCN1+1       RIGHT BREAK INDEX, PASS 1.   EXPR0290
00013   0622 00 0 00116         STD     RSCN2+1       RIGHT BREAK INDEX, PASS 2.   EXPR0300
00014   0625 00 0 00171         STT     TAG           SAVE TAG FOR DECISION TO STORE. EXPR0310
00015   0621 00 0 00065         STA     TRMWD         SAVE LEFT BREAK INDEX FOR FIRST TERM. EXPR0320
00016   0734 00 1 00000         PAX     0,1           INDEX FOR LEFT BREAK.        EXPR0330
00017   1 77777 1 00020         TXI     *+1,1,-1      INDEX OF FIRST WORD IN FLD.  EXPR0340
00020   0634 00 1 00100         SXA     IWD1,1        SAVE FOR PASS2.              EXPR0350
00021  -0500 00 4 00002         CAL     2,4           INSERT FLD ADDRESSES.        EXPR0360
00022   0621 00 0 00035         STA     EPAS1         ..                           EXPR0370
00023   0621 00 0 00102         STA     SCN2          ..                           EXPR0380
00024   0621 00 0 00066         STA     BA1           ..                           EXPR0390
00025   0621 00 0 00104         STA     BA2           ..                           EXPR0400
00026   0402 00 0 00176         SUB     =1            ..                           EXPR0410
00027   0621 00 0 00160         STA     BA3           ..                           EXPR0420
00030   0600 00 0 00172         STZ     TEOF          RESET EOF MARK.              EXPR0430
00031   0600 00 0 00170         STZ     LBRK          RESET LEFT BREAK.            EXPR0440
00032   0600 00 0 00173         STZ     TSYM          RESET SYMBOL MARK.           EXPR0450
00033   0600 00 0 00174         STZ     TSTSL         RESET */ MARK.               EXPR0460
```

```
                SUBROUTINE EXPR, EVALUATE EXPRESSION FOR COMPOP.

00034  0020 00 0 00035          TRA     EPAS1           GO TO PASS 1 OF EXPR.                    EXPR0470


                                 SPACE   2                                                        EXPR0480
                                         PASS1, REDUCE TO SUMMATION.                              EXPR0490
                                                                                                  EXPR0500
00035 -0500 00 1 00000   EPAS1 CAL     FLD,1           GET NEXT WORD IN FLD.                    EXPR0510
00036  0100 00 0 00053          TZE     RSCN1           IF ZERO, IGNORE.                         EXPR0520
00037 -0340 00 0 00202          LAS     =H00000/        CHECK FOR /.                             EXPR0530
00040  0020 00 0 00042          TRA     *+2             NO, SKIP.                                EXPR0540
00041  0020 00 0 00057          TRA     STSL1           YES, EXIT.                               EXPR0550
00042 -0340 00 0 00201          LAS     =HC0000*        CHECK FOR *.                             EXPR0560
00043  0020 00 0 00045          TRA     *+2             NO, SKIP.                                EXPR0570
00044  0020 00 0 00057          TRA     STSL1           YES, EXIT.                               EXPR0580
00045 -0340 00 0 00200          LAS     =H00000-        CHECK FOR -.                             EXPR0590
00046  0020 00 0 00050          TRA     *+2             NO, SKIP.                                EXPR0600
00047  0020 00 0 00061          TRA     PLMI1           YES, EXIT.                               EXPR0610
00050 -0340 00 0 00177          LAS     =H00000+        CHECK FOR +.                             EXPR0620
00051  0020 00 0 00053          TRA     *+2             NO, SKIP.                                EXPR0630
00052  0020 00 0 00061          TRA     PLMI1           YES, EXIT.                               EXPR0640
00053  1 77777 1 00054   RSCN1 TXI     *+1,1,-1        COUNT WORDS.                             EXPR0650
00054  3 00000 1 00035          TXH     EPAS1,1,**      CHECK FOR EOF.                           EXPR0660
00055 -0625 00 0 00172          STL     TEOF            SET EOF MARK,                            EXPR0670
00056  0020 00 0 00061          TRA     PLMI1           AND TREAT AS +-.                         EXPR0680
                                                                                                  EXPR0690
00057 -0625 00 0 00174   STSL1 STL     TSTSL           * OR / MET, SET MARK.                    EXPR0700
00060  0020 00 0 00053          TRA     RSCN1           RESUME SCAN.                             EXPR0710
                                                                                                  EXPR0720
00061 -0520 00 0 00174   PLMI1 NZT     TSTSL           END OF TERM, CHECK FOR */.               EXPR0730
00062  0020 00 0 00070          TRA     NSTSL           NO, SKIP TERM.                           EXPR0740
00063 -0634 00 1 00065          SXD     TRMWD,1         SET RIGHT INDEX FOR TERM.                EXPR0750
00064  0074 00 4 00000          TSX     $TERM,4         GO TO TERM.                              EXPR0760
00065  0 00000 0 00000   TRMWD PZE     **,0,**         ..                                       EXPR0770
00066  0 00000 0 00000   BA1   PZE     FLD             ..                                       EXPR0780
00067  0600 00 0 00174          STZ     TSTSL           RETURN FROM TERM, RESET STSL.            EXPR0790
00070  0634 00 1 00065   NSTSL SXA     TRMWD,1         SET NEXT LEFT INDEX.                     EXPR0800
00071 -0520 00 0 00172          NZT     TEOF            TEST FOR EOF.                            EXPR0810
00072  0020 00 0 00053          TRA     RSCN1           NO, RESUME SCAN.                         EXPR0820
00073  0020 00 0 00074          TRA     EPAS2           YES, GO TO PASS2.                        EXPR0830


                                 SPACE   2                                                        EXPR0840
                                         PASS2, COMPUTE SUM.                                      EXPR0850
                                                                                                  EXPR0860
00074  0600 00 0 00173   EPAS2 STZ     TSYM            RESET SYMBOL MARK.                       EXPR0870
00075 -0500 00 0 00177          CAL     =HC0000+        SET LBRK=+.                              EXPR0880
00076  0602 00 0 00170          SLW     LBRK            ..                                       EXPR0890
00077 -0625 00 0 00175          STL     FIRST           SET FIRST ADDEND MARK.                   EXPR0900
00100  0774 00 1 00000   IWD1  AXT     **,1            GET INDEX OF FIRST WORD IN FLD.          EXPR0910
00101  0600 00 0 00172          STZ     TECF            RESET ECF MARK.                          EXPR0920
00102 -0500 00 1 00000   SCN2  CAL     FLD,1           GET NEXT WORD IN FLD.                    EXPR0930
00103  0100 00 0 00115          TZE     RSCN2           IF ZERO, IGNORE.                         EXPR0940
00104  0600 00 1 00000   BA2   STZ     FLD,1           ZERO FLD LOCATION.                       EXPR0950
00105 -0340 00 0 00200          LAS     =H00000-        CHECK FOR -.                             EXPR0960
00106  0020 00 0 00110          TRA     *+2             NO, SKIP.                                EXPR0970
00107  0020 00 0 00121          TRA     BRK             YES, EXIT.                               EXPR0980
```

```
                SUBROUTINE EXPR, EVALUATE EXPRESSION FOR COMPOP.

00110 -0340 00 0 00177          LAS     =H00000+        CHECK FOR +.                    EXPR0990
00111  0020 00 0 00113          TRA     *+2             NO, SKIP.                       EXPR1000
00112  0020 00 0 00121          TRA     BRK             YES, EXIT.                      EXPR1010
00113  0074 00 4 00001          TSX     $PIVAR,4        SYMBOL, PLACE IN VARIABLE FIELD.EXPR1020
00114 -0625 00 0 00173          STL     TSYM            SET SYMBOL MARK.                EXPR1030
00115  1 77777 1 00116  RSCN2   TXI     *+1,1,-1        COUNT WORDS.                    EXPR1040
00116  3 00000 1 00102          TXH     SCN2,1,**       CHECK FOR EOF.                  EXPR1050
00117 -0625 00 0 00172          STL     TEOF            SET EOF MARK,                   EXPR1060
00120  0020 00 0 00121          TRA     BRK             AND TREAT AS +-.                EXPR1070

                                SPACE   2                                               EXPR1080
                                ANALYZE BREAK.                                          EXPR1090
                                                                                        EXPR1100
00121 -0130 00 0 00000  BRK     XCL                     GET LAST BREAK,                 EXPR1110
00122  0500 00 0 00170          CLA     LBRK            AND STORE THIS ONE IN LBRK.     EXPR1120
00123 -0600 00 0 00170          STQ     LBRK            ..                              EXPR1130
00124  0520 00 0 00175          ZET     FIRST           CHECK FOR FIRST ADDEND.         EXPR1140
00125  0020 00 0 00137          TRA     FBRK            YES, GO TO IT.                  EXPR1150
00126  0322 00 0 00177          ERA     =H00000+        CHECK FOR +.                    EXPR1160
00127  0100 00 0 00131          TZE     FAD             YES, GO TO IT.                  EXPR1170
00130  0020 00 0 00134          TRA     FSB             NO, MUST BE -, GO TO IT.        EXPR1180
                                                                                        EXPR1190
00131  0074 00 4 00002  FAD     TSX     $GENOP,4        LBRK=+, OP=FAD.                 EXPR1200
00132  602621246060             BCI     1, FAD          ..                              EXPR1210
00133  0020 00 0 00153          TRA     EOF2            CHECK FOR EOF.                  EXPR1220
                                                                                        EXPR1230
00134  0074 00 4 00002  FSB     TSX     $GENOP,4        LBRK=-, OP=FSB.                 EXPR1240
00135  602662226060             BCI     1, FSB          ..                              EXPR1250
00136  0020 00 0 00153          TRA     EOF2            CHECK FOR EOF.                  EXPR1260

                                SPACE   2                                               EXPR1270
                                FIRST BREAK SECTION.                                    EXPR1280
                                                                                        EXPR1290
00137 -0520 00 0 00173  FBRK    NZT     TSYM            CHECK FOR SYMBOL.               EXPR1300
00140  0020 00 0 00153          TRA     EOF2            NO SYMBOL, IGNORE UNARY OP.     EXPR1310
00141  0600 00 0 00175          STZ     FIRST           FIRST ADDEND ENCOUNTERED.       EXPR1320
00142  0322 00 0 00177          ERA     =H00000+        CHECK FOR PLUS.                 EXPR1330
00143  0100 00 0 00145          TZE     CLA             YES, GO TO IT.                  EXPR1340
00144  0020 00 0 00150          TRA     CLS             NO, MUST BE -, GO TO IT.        EXPR1350
                                                                                        EXPR1360
00145  0074 00 4 00002  CLA     TSX     $GENOP,4        LBRK=+, OP=CLA.                 EXPR1370
00146  602343216060             BCI     1, CLA          ..                              EXPR1380
00147  0020 00 0 00153          TRA     EOF2            CHECK FOR EOF.                  EXPR1390
                                                                                        EXPR1400
00150  0074 00 4 00002  CLS     TSX     $GENOP,4        LBRK=-, OP=CLS.                 EXPR1410
00151  602343626060             BCI     1, CLS          ..                              EXPR1420
00152  0020 00 0 00153          TRA     EOF2            CHECK FOR EOF.                  EXPR1430

                                SPACE   2                                               EXPR1440
                                END-OF-FIELD AND RETURN SECTION.                        EXPR1450
                                                                                        EXPR1460
00153 -0520 00 0 00172  EOF2    NZT     TEOF            TEST FOR EOF REACHED.           EXPR1470
00154  0020 00 0 00115          TRA     RSCN2           NO, RESUME SCAN.                EXPR1480
```

```
                SUBROUTINE EXPR, EVALUATE EXPRESSION FOR COMPCP.

00155  0520 00 0 00171          ZET     TAG             CHECK TAG OF CONTROL WORD.          EXPR1490
00156  0020 00 0 00164          TRA     RX4             NON-ZERO, LEAVE IN AC, RETURN.      EXPR1500
00157  0074 00 4 00003          TSX     $GNSTO,4        ZERO, GENERATE TEMPORARY STORAGE.   EXPR1510
00160  0602 00 1 77777   BA3    SLW     FLD-1,1         INSERT IN FLD.                      EXPR1520
00161  0074 00 4 00001          TSX     $PIVAR,4        PLACE SYMBOL IN VARIABLE FIELD.     EXPR1530
00162  0074 00 4 00002          TSX     $GENOP,4        FINAL OP IS STO.                    EXPR1540
00163  606263466060             BCI     1, STO          ..                                  EXPR1550
00164  0774 00 4 00000   RX4    AXT     **,4            RESTORE IRS.                        EXPR1560
00165  0774 00 2 00000   RX2    AXT     **,2            ..                                  EXPR1570
00166  0774 00 1 00000   RX1    AXT     **,1            ..                                  EXPR1580
00167  0020 00 4 00003          TRA     3,4             RETURN TO CALLER.                   EXPR1590

                                SPACE   2                                                   EXPR1600
                                STORAGE AND CONSTANTS.                                      EXPR1610
                                                                                            EXPR1620
00170  0 00000 0 00000   LBRK   PZE                     LAST BREAK.                         EXPR1630
00171  0 00000 0 00000   TAG    PZE                     TAG OF CONTROL WORD.                EXPR1640
00172  0 00000 0 00000   TEOF   PZE                     EOF MARK.                           EXPR1650
00173  0 00000 0 00000   TSYM   PZE                     SYMBOL MARK.                        EXPR1660
00174  0 00000 0 00000   TSTSL  PZE                     */ MARK.                            EXPR1670
00175  0 00000 0 00000   FIRST  PZE                     FIRST ADDEND MARK.                  EXPR1680
               00000     FLD    EQU     **              DUMMY SYMBOL FOR FLD.               EXPR1690
                                                                                            EXPR1700
                                END                                                         EXPR1710

  LITERALS
00176  000000000001
00177  000000000020
00200  000000000040
00201  000000000054
00202  000000000061
```

SUBROUTINE EXPR, EVALUATE EXPRESSION FOR COMPOP.
POST PROCESSOR ASSEMBLY DATA

203 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

66	BA1	24					
104	BA2	25					
160	BA3	27					
121	BRK	107,	112,	120			
145	CLA	143					
150	CLS	144					
131	FAD	127					
0	FLD	35,	66,	102,	104,	160,	176
134	FSB	130					
166	RX1	10					
165	RX2	7					
164	RX4	6,	156				
171	TAG	14,	155				
153	EOF2	133,	136,	140,	147,	152	
6	EXPR						
137	FBRK	125					
100	IWD1	20					
170	LBRK	31,	76,	122,	123		
102	SCN2	23,	116				
172	TEOF	30,	55,	71,	101,	117,	153
0	TERM	64					
173	TSYM	32,	74,	114,	137		
35	EPAS1	22,	34,	54			
74	EPAS2	73					
175	FIRST	77,	124,	141			
2	GENCP	131,	134,	145,	150,	162	
3	GNSTC	157					
70	NSTSL	62					
1	PIVAR	113,	161				
61	PLMI1	47,	52,	56			
53	RSCN1	12,	36,	60,	72		
115	RSCN2	13,	103,	154			
57	STSL1	41,	44				
65	TRMWD	15,	63,	70			
174	TSTSL	33,	57,	61,	67		

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000008 IN HUNDREDTHS OF MINUTES.

```
SUBROUTINE TERM, EVALUATE SIMPLE TERMS FOR EXPR.

                PCC
                COUNT   135                                                     TERM0010
                LBL     TERM                                                    TERM0020
 00006          ENTRY   TERM            BINARY CARD LABEL.                      TERM0040
                                        EVALUATE TERM OF EXPRESSION.            TERM0050
        *                                                                       TERM0060
        *       $TERM IS CALLED BY,                                             TERM0070
        *                                                                       TERM0080
        *            TSX $TERM,4                                                TERM0090
        *            PZE LI,0,RI                                                TERM0100
        *            PZE FLD                                                    TERM0110
        *                                                                       TERM0120
        *       WHERE (FLD-LI) IS THE ADDRESS OF THE LEFT BREAK,                TERM0130
        *       AND (FLD-RI) IS THE ADDRESS OF THE RIGHT BREAK.                 TERM0140
        *       TERM TAKES A STRING OF SYMBOLS CONNECTED BY                     TERM0150
        *       * OR / (S) AND COMPILES THE RESULT IN FLOATING                  TERM0160
        *       POINT.  THE RESULT IS STORED IN TEMPORARY                       TERM0170
        *       STORAGE, AND THE SPREAD FIELD IS MODIFIED ACCORDINGLY.          TERM0180
        *       SOME OPTIMIZATION IS DONE.                                      TERM0190

  TRANSFER VECTOR
00000   473165215160    PIVAR
00001   272545464760    GENOP
00002   255121622560    ERASE
00003   274562634660    GNSTO

  LINKAGE DIRECTOR
00004   000000000000
00005   632551446060

00006   0634 00 4 00133  TERM  SXA     RX4,4           SAVE IRS.               TERM0200
00007   0634 00 2 00134        SXA     RX2,2           ..                      TERM0210
00010   0634 00 1 00135        SXA     RX1,1           ..                      TERM0220
00011  -0500 00 4 00001        CAL     1,4             GET CONTROL WORD.       TERM0230
00012   0622 00 0 00037        STD     RSCAN+1         INDEX OF RIGHT BREAK.   TERM0240
00013   0734 00 1 00000        PAX     0,1             INDEX OF LEFT BREAK.    TERM0250
00014   1 77777 1 00015        TXI     *+1,1,-1        INDEX OF FIRST WORD IN FLD.TERM0260
00015  -0500 00 4 00002        CAL     2,4             INSERT BUFFER ADDRESSES.TERM0270
00016   0621 00 0 00024        STA     SCAN            ..                      TERM0280
00017   0621 00 0 00026        STA     BA1             ..                      TERM0290
00020   0402 00 0 00140        SUB     =1              ..                      TERM0300
00021   0621 00 0 00117        STA     BA2             ..                      TERM0310
00022   0621 00 0 00127        STA     BA3             ..                      TERM0320
00023   0600 00 0 00137        STZ     LBRK            RESET LAST BREAK.       TERM0330

00024  -0500 00 1 00000  SCAN  CAL     FLD,1           GET NEXT WORD IN FLD.   TERM0340
00025   0100 00 0 00036        TZE     RSCAN           IF ZERO, IGNORE.        TERM0350
00026   0600 00 1 00000  BA1   STZ     FLD,1           ZERO FLD LOCATION.      TERM0360
00027  -0340 00 0 00142        LAS     =HC0000/        CHECK FOR /.            TERM0370
00030   0020 00 0 00032        TRA     *+2             NO, SKIP.               TERM0380
00031   0020 00 0 00063        TRA     SLASH           YES, GO TO IT.          TERM0390
00032  -0340 00 0 00141        LAS     =HC0000*        CHECK FOR *.            TERM0400
00033   0020 00 0 00035        TRA     *+2             NO, SKIP.               TERM0410
00034   0020 00 0 00041        TRA     STAR            YES, GO TO IT.          TERM0420
00035   0074 00 4 00000        TSX     $PIVAR,4        SYMBOL, PLACE IN VARIABLE FIELD.TERM0430
00036   1 77777 1 00037  RSCAN TXI     *+1,1,-1        COUNT WORDS.            TERM0440
00037   3 00000 1 00024        TXH     SCAN,1,**       CHECK FOR EOF.          TERM0450
                                                                               TERM0460
```

```
                SUBROUTINE TERM, EVALUATE SIMPLE TERMS FOR EXPR.

00040  0020 00 0 00105          TRA     EOF             EOF, GO TO IT.          TERM0470

                                SPACE   2                                       TERM0480
                                        BREAK=*.                                TERM0490
                                                                                TERM0500
                                                        STAR, GET LAST BREAK.   TERM0510
00041 -0130 00 0 00000   STAR   XCL                     THEN PLACE * IN LBRK.   TERM0520
00042 -0500 00 0 00137          CAL     LBRK            ..                      TERM0530
00043 -0600 00 0 00137          STQ     LBRK            IF LBRK=0, GO TO IT.    TERM0540
00044  0100 00 0 00050          TZE     ST1             CHECK, LBRK=*.          TERM0550
00045  0322 00 0 00141          ERA     =H00000*        YES, GO TO IT.          TERM0560
00046  0100 00 0 00053          TZE     ST2             NO, MUST BE /, GO TO IT.TERM0570
00047  0020 00 0 00060          TRA     ST3                                     TERM0580
                                                                                TERM0590
00050  0074 00 4 00001   ST1    TSX     $GENOP,4        LBRK=0, OP=LDQ.         TERM0600
00051  604324506060             BCI     1, LDQ          ..                      TERM0610
00052  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.            TERM0620
                                                                                TERM0630
00053  0074 00 4 00001   ST2    TSX     $GENOP,4        LBRK=*, OP=FMP.         TERM0640
00054  602644476060             BCI     1, FMP          ..                      TERM0650
00055  0074 00 4 00001          TSX     $GENOP,4        THEN, OP=XCA.           TERM0660
00056  606723216060             BCI     1, XCA          ..                      TERM0670
00057  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.            TERM0680
                                                                                TERM0690
00060  0074 00 4 00001   ST3    TSX     $GENOP,4        LBRK=1, OP=FDP.         TERM0700
00061  602624476060             BCI     1, FDP          ..                      TERM0710
00062  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.

                                SPACE   2                                       TERM0720
                                        BREAK=/.                                TERM0730
                                                                                TERM0740
                                                        SLASH, GET LAST BREAK.  TERM0750
00063 -0130 00 0 00000   SLASH  XCL                     THEN PLACE / IN LBRK.   TERM0760
00064 -0500 00 0 00137          CAL     LBRK            ..                      TERM0770
00065 -0600 00 0 00137          STQ     LBRK            IF LBRK=0, GO TO IT.    TERM0780
00066  0100 00 0 00072          TZE     SL1             CHECK, LBRK=*.          TERM0790
00067  0322 00 0 00141          ERA     =H00000*        YES, GO TO IT.          TERM0800
00070  0100 00 0 00075          TZE     SL2             NO, MUST BE /, GO TO IT.TERM0810
00071  0020 00 0 00100          TRA     SL3                                     TERM0820
                                                                                TERM0830
00072  0074 00 4 00001   SL1    TSX     $GENOP,4        LBRK=0, OP=CLA.         TERM0840
00073  602343216060             BCI     1, CLA          ..                      TERM0850
00074  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.            TERM0860
                                                                                TERM0870
00075  0074 00 4 00001   SL2    TSX     $GENOP,4        LBRK=*, OP=FMP.         TERM0880
00076  602644476060             BCI     1, FMP          ..                      TERM0890
00077  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.            TERM0900
                                                                                TERM0910
00100  0074 00 4 00001   SL3    TSX     $GENOP,4        LBRK=1, OP=FDP.         TERM0920
00101  602624476060             BCI     1, FDP          ..                      TERM0930
00102  0074 00 4 00001          TSX     $GENOP,4        THEN, OP=XCA.           TERM0940
00103  606723216060             BCI     1, XCA          ..                      TERM0950
00104  0020 00 0 00036          TRA     RSCAN           RESUME SCAN.
```

```
                SUBROUTINE TERM, EVALUATE SIMPLE TERMS FOR EXPR.

                                        SPACE   2                                                       TERM0960
                                                END-OF-FIELD AND RETURN SECTION.                        TERM0970
                                                                                                        TERM0980
C0105 -0500 00 0 00137   EOF    CAL     LBRK            EOF, GET LBRK.                                 TERM0990
00106  0100 00 0 00112          TZE     EOF1            IF LBRK=0, GO TO IT.                           TERM1000
00107  0322 00 0 00141          ERA     =H00000*        CHECK, LBRK=*.                                 TERM1010
00110  0100 00 0 00114          TZE     EOF2            YES, GO TO IT.                                 TERM1020
C0111  0020 00 0 00124          TRA     EOF3            NO, MUST BE /, GO TO IT.                       TERM1030
                                                                                                        TERM1040
00112  0074 00 4 00002   EOF1   TSX     $ERASE,4        LBRK=0, NO COMPILATION.                        TERM1050
00113  0020 00 0 00133          TRA     RX4             GO TO RETURN.                                  TERM1060
                                                                                                        TERM1070
00114  0074 00 4 00001   EOF2   TSX     $GENOP,4        LBRK=*, OP=FMP.                                TERM1080
C0115  602644476060             BCI     1, FMP          ..                                             TERM1090
00116  0074 00 4 00003          TSX     $GNSTO,4        GENERATE TEMPORARY STORAGE.                    TERM1100
00117  0602 00 1 77777   BA2    SLW     FLC-1,1         INSERT IN FLD.                                 TERM1110
C0120  0074 00 4 00000          TSX     $PIVAR,4        PLACE SYMBOL IN VARIABLE FIELD.                TERM1120
00121  0074 00 4 00001          TSX     $GENOP,4        FINAL OP IS STO.                               TERM1130
C0122  606263466060             BCI     1, STO          ..                                             TERM1140
00123  0020 00 0 00133          TRA     RX4             GO TO RETURN.                                  TERM1150
                                                                                                        TERM1160
00124  0074 00 4 00001   EOF3   TSX     $GENOP,4        LBRK=/, OP=FDP.                                TERM1170
00125  602624476060             BCI     1, FDP          ..                                             TERM1180
00126  0074 00 4 00003          TSX     $GNSTO,4        GENERATE TEMPORARY STORAGE.                    TERM1190
00127  0602 00 1 77777   BA3    SLW     FLC-1,1         INSERT IN FLD.                                 TERM1200
00130  0074 00 4 00000          TSX     $PIVAR,4        PLACE SYMBOL IN VARIABLE FIELD.                TERM1210
00131  0074 00 4 00001          TSX     $GENOP,4        FINAL OP IS STQ.                               TERM1220
00132  606263506060             BCI     1, STQ          ..                                             TERM1230
                                                                                                        TERM1240
00133  0774 00 4 00000   RX4    AXT     **,4            RESTORE IRS.                                   TERM1250
00134  0774 00 2 00000   RX2    AXT     **,2            ..                                             TERM1260
C0135  0774 00 1 C0000   RX1    AXT     **,1            ..                                             TERM1270
00136  0020 00 4 00003          TRA     3,4             RETURN TO CALLER.                              TERM1280

                                        SPACE   2                                                       TERM1290
                                                STORAGE AND CONSTANTS.                                  TERM1300
                                                                                                        TERM1310
00137  0 00000 0 00000   LBRK   PZE                     LAST BREAK CHARACTER.                          TERM1320
                00000    FLD    EQU     **              DUMMY SYMBOL FOR FLD.                          TERM1330
                                                                                                        TERM1340
                                        END                                                             TERM1350

  LITERALS
C0140  000C0000001
00141  000000000054
00142  000000000061
```

SUBROUTINE TERM, EVALUATE SIMPLE TERMS FOR EXPR.
POST PROCESSOR ASSEMBLY DATA

143 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

26	BA1	17										
117	BA2	21										
127	BA3	22										
105	ECF	40										
0	FLD	24,	26,	117,	127,	140						
135	RX1	10										
134	RX2	7										
133	RX4	6,	113,	123								
72	SL1	66										
75	SL2	70										
100	SL3	71										
50	ST1	44										
53	ST2	46										
60	ST3	47										
112	EOF1	106										
114	EOF2	110										
124	EOF3	111										
137	LBRK	23,	42,	43,	64,	65,	105					
24	SCAN	16,	37									
41	STAR	34										
6	TERM											
2	ERASE	112										
1	GENOP	50,	53,	55,	60,	72,	75,	100,	102,	114,	121,	124, 131
3	GNSTO	116,	126									
0	PIVAR	35,	120,	130								
36	RSCAN	12,	25,	52,	57,	62,	74,	77,	104			
63	SLASH	31										

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000006 IN HUNDREDTHS OF MINUTES.

```
CIOP . . . BUFFERED I/O PACKAGE FOR CAP.  NO TAPE ERROR CHECKING.

                                PCC                                                                    CIOP0010
                                COUNT   147                                                            CIOP0020
                                LBL     CIOP        BINARY CARD LABEL.                                 CIOP0040
                00003           ENTRY   READ1       READ ONE RECORD ON INPUT TAPE.                     CIOP0050
                00154           ENTRY   PRINT       WRITE ONE RECORD ON OUTPUT TAPE.                   CIOP0060
                00030           ENTRY   WCT1        WRITE ONE RECORD ON COLLATION TAPE.                CIOP0070
                00104           ENTRY   REWIND      END-FILE AND REWIND COLLATION TAPE.                CIOP0080
                00110           ENTRY   READ2       READ ONE RECORD FROM COLLATION TAPE.               CIOP0090
                                                                                                       CIOP0100
  TRANSFER VECTOR
00000  256731636060     EXIT

  LINKAGE DIRECTOR
00001  000000000000
00002  512521240160

00003  0634 00 2 00013  READ1 SXA     RX2,2       SAVE IR2.                                          CIOP0110
00004 -0500 00 4 00001         CAL     1,4         GET BUFFER ADDRESS.                                CIOP0120
00005  0621 00 0 00015         STA     INCARD      INSERT IN I/O COMMAND.                             CIOP0130
00006  0762 00 0 01202         RTDA    2           SELECT INPUT TAPE.                                 CIOP0140
00007  0540 00 0 00015         RCHA    INCARD      START DATA CHANNEL.                                CIOP0150
00010  0060 00 0 00010         TCOA    *           WAIT FOR DATA TO ARRIVE.                           CIOP0160
00011  0774 00 2 00016         AXT     WEOFI,2     GET ERROR COMMENT LOCATION.                        CIOP0170
00012  0030 00 0 00214         TEFA    ERROR       CHECK FOR ERROR.                                   CIOP0180
00013  0774 00 2 00000  RX2    AXT     **,2        RESTORE IR2.                                       CIOP0190
00014  0020 00 4 00002         TRA     2,4         RETURN TO CALLER.                                  CIOP0200
                                                                                                       CIOP0210
00015  0 00016 0 00000  INCARD IOCD    **,0,14     READ ONE 14 WORD RECORD AND STOP.                  CIOP0220
                                                                                                       CIOP0230
00016  0 00011 0 00017  WEOFI IOCD    *+1,0,9     OUTPUT COMMENT IN CASE OF ERROR.                   CIOP0240
00017  002545246046            BCI     9,END OF FILE REACHED WHILE READING CAP INPUT TAPE.            CIOP0250
00020  266026314325
00021  605125212330
00022  252460663031
00023  432560512521
00024  243145276023
00025  214760314547
00026  646360632147
00027  253360606060

                                                                                                       CIOP0260
                        WCT1 WRITES A 14 WORD BUFFER ON TAPE B3.  IF CONTROL                           CIOP0270
                        WORD HAS A NON-ZERO TAG, COUNT IS OBTAINED FROM DECREMENTCIOP0280
                        AND RECORD IS FILLED OUT WITH BLANKS.                                          CIOP0290
                                                                                                       CIOP0300
                        THIS ROUTINE IS BUFFERED, AND OUTPUT IS OVERLAPPED                             CIOP0310
                        WITH COMPUTATION.                                                              CIOP0320
                                                                                                       CIOP0330
00030  0634 00 4 00067  WCT1  SXA     WX4,4       SAVE IR5.                                          CIOP0340
00031  0634 00 2 00066         SXA     WX2,2       ..                                                 CIOP0350
00032 -0520 00 0 00223         NZT     FSTART      IS THIS THE FIRST CALL.                            CIOP0360
00033  0772 00 0 02203         REWB    3           YES, MAKE SURE TAPE REWOUND.                       CIOP0370
00034 -0625 00 0 00223         STL     FSTART      SET MARKER.                                        CIOP0380
00035 -0500 00 4 00001         CAL     1,4         GET CONTROL WORD.                                  CIOP0390
00036  0625 00 0 00225         STT     TAG         GET TAG.                                           CIOP0400
00037 -0520 00 0 00225         NZT     TAG         IF ZERO,                                           CIOP0410
```

```
      CIOP . . . BUFFERED I/O PACKAGE FOR CAP.  NO TAPE ERROR CHECKING.

00040 -0500 00 0 00263         CAL     =14B17          ASSUME 14 WORDS IN BUFFER.                    CIOP0420
00041 -0734 00 2 00000         PDX     0,2             SAVE COUNT FOR MOVE OPERATION.                CIOP0430
00042  0771 00 0 00022         ARS     18              MOVE COUNT TO ADDRESS.                        CIOP0440
00043  0361 00 4 00001         ACL     1,4             FORM END ADDRESS.                             CIOP0450
00044  0621 00 0 00054         STA     PCKUP           INSERT IN PICKUP INSTRUCTION.                 CIOP0460
00045  0754 00 2 00000         PXA     0,2             SAVE IR2 IN AC.                               CIOP0470
00046  0774 00 2 00072         AXT     WEOCT,2         GET ADDRESS OF ERROR COMMENT.                 CIOP0480
00047  0061 00 0 00047         TCOB    *               WAIT FOR PREVICUS WRITE TO FINISH.            CIOP0490
00050 -0760 00 0 02000         ETTB                    CHECK FOR END OF TAPE ON LAST WRITE.          CIOP0500
00051  0020 00 0 00214         TRA     ERROR           END OF TAPE ENCOUNTERED, GO COMPLAIN.         CIOP0510
00052  0734 00 2 00000         PAX     0,2             RESTORE IR2.                                  CIOP0520
00053  0774 00 4 00016         AXT     14,4            SET MOVE COUNTER.                             CIOP0530
00054 -0500 00 2 00000  PCKUP  CAL     **,2            MOVE DATA INTO OUTPUT BUFFER.                 CIOP0540
00055  0602 00 4 00262  SLW    SLW     WBUFF+14,4      INSERT.                                       CIOP0550
00056 -2 00001 4 00060         TNX     *+2,4,1         COUNT.                                        CIOP0560
00057  2 00001 2 00054         TIX     PCKUP,2,1       INDEX, AND GET NEXT WORD.                     CIOP0570
00060 -3 00001 4 00064         TXL     *+4,4,1         IS BUFFER FULL.                               CIOP0580
00061 -0500 00 0 00264         CAL     =H              NO, FILL IT OUT WITH BLANKS.                  CIOP0590
00062  0602 60 0 00055         SLW*    SLW             ..                                            CIOP0600
00063  2 00001 4 00061         TIX     *-2,4,1         TEST FOR BUFFER FULL.                         CIOP0610
00064  0766 00 0 02223         WTBB    3               SELECT COLLATION TAPE.                        CIOP0620
00065 -0540 00 0 00071         RCHB    IOWCT           START CHANNEL.                                CIOP0630
00066  0774 00 2 00000  WX2    AXT     **,2            RESTORE IRS.                                  CIOP0640
00067  0774 00 4 00000  WX4    AXT     **,4            ..                                            CIOP0650
00070  0020 00 4 00002         TRA     2,4             RETURN TO CALLER.                             CIOP0660
                                                                                                     CIOP0670
00071  0 00016 0 00244  IOWCT  IOCD    WBUFF,0,14      WRITE A 14 WORD RECORD AND STOP.              CIOP0680
                                                                                                     CIOP0690
                                                                                                     CIOP0700
00072  0 00011 0 00073  WEOCT  IOCD    *+1,,9                                                        CIOP0710
00073  002545246046            BCI     9,0END OF TAPE REACHED WHILE WRITING COLLATION TAPE.
00074  266063214725
00075  605125212330
00076  252460663031
00077  432560665131
00100  633145276023
00101  464343216331
00102  464560632147
00103  253360606060                                                                               CIOP0720
                                                                                                   CIOP0730
00104  0770 00 0 02203  REWIND WEFB    3               WRITE AN END OF FILE.                         CIOP0740
00105 -0625 00 0 00224         STL     TREW            SET MARK FOR TAPE REWOUND.                    CIOP0750
00106  0772 00 0 02203         REWB    3               NOW REWIND TAPE.                              CIOP0760
00107  0020 00 4 000C1         TRA     1,4             RETURN TO CALLER.                             CIOP0770
                                                                                                     CIOP0780
00110  0634 00 2 00122  READ2  SXA     R2X2,2          SAVE IR2.                                     CIOP0790
00111 -0520 00 0 00224         NZT     TREW            HAS COLLATION TAPE BEEN REWOUND.              CIOP0800
00112  0020 00 0 00125         TRA     NREW            NO, GO COMMENT.                               CIOP0810
00113 -0500 00 4 00001         CAL     1,4             GET CONTROL WORD.                             CIOP0820
00114  0621 00 0 00124         STA     IOTIN           INSERT BUFFER ADDRESS IN I/O COMMAND.         CIOP0830
00115  0762 00 0 02223         RTBB    3               READ SELECT COLLATION TAPE.                   CIOP0840
00116 -0540 00 0 00124         RCHB    IOTIN           START CHANNEL.                                CIOP0850
00117  0774 00 2 00142         AXT     WEOFC,2         GET ADDRESS OF ERROR COMMENT.                 CIOP0860
00120  0061 00 0 00120         TCOB    *               WAIT FOR BUFFER TO FILL.                      CIOP0870
00121 -0030 00 0 00214         TEFB    ERROR           HAS END OF FILE BEE N REACHED.                CIOP0880
00122  0774 00 2 00000  R2X2   AXT     **,2            ALL OK, RESTORE IR2.                          CIOP0890
00123  0020 00 4 00002         TRA     2,4             RETURN TO CALLER.
```

```
      CIOP . . . BUFFERED I/O PACKAGE FOR CAP.  NO TAPE ERROR CHECKING.

                                                                                                  CIOP0900
00124  3 00016 0 00000   IOTIN IORT    **,,14          READ ONE RECORD AND STOP.                  CIOP0910
                                                                                                  CIOP0920
00125  0074 00 4 00154   NREW  TSX     PRINT,4         FORGOT TO REWIND, COMMENT.                 CIOP0930
00126  0 00011 0 00131         PZE     WNREW,0,9       ..                                         CIOP0940
00127  0074 00 4 00104         TSX     REWIND,4        REWIND IT.                                 CIOP0950
00130  0020 00 0 00113         TRA     READ2+3         RETURN TO READ ROUTINE.                    CIOP0960
                                                                                                  CIOP0970
00131  005125212402      WNREW BCI     9,CREAD2 CALLED BEFORE REWIND, COLLATION TAPE REWOUND.     CIOP0980
00132  602321434325
00133  246022252646
00134  512560512566
00135  314524736023
00136  464343216331
00137  464560632147
00140  256051256646
00141  644524336060
                                                                                                  CIOP0990
00142  0 00011 0 00143   WEOFC IOCD     *+1,,9          OUTPUT COMMENT IF ERROR.                   CIOP1000
00143  002545246046            BCI     9,0END OF FILE REACHED WHILE READING COLLATION TAPE.       CIOP1010
00144  266026314325
00145  605125212330
00146  252460663031
00147  432560512521
00150  243145276023
00151  464343216331
00152  464560632147
00153  253360606060
                                                                                                  CIOP1020
00154  0634 00 2 00201   PRINT SXA     PX2,2           SAVE IR2.                                  CIOP1030
00155  0774 00 2 00204         AXT     WLNEX,2         GET ADDRESS OF ERROR COMMENT.              CIOP1040
00156  0500 00 0 00222         CLA     LNCNT           GET LINECOUNT.                             CIOP1050
00157  0402 00 0 00262         SUB     =1              LOWER BY ONE.                              CIOP1060
00160  0601 00 0 00222         STO     LNCNT           RETURN.                                    CIOP1070
00161 -0120 00 0 00214         TMI     ERROR           EXIT IF TOO MANY LINES.                    CIOP1080
00162 -0500 00 4 00001         CAL     1,4             GET CONTROL WORD.                          CIOP1090
00163  0622 00 0 00203         STD     PIO             INSERT COUNT IN I/O COMMAND.               CIOP1100
00164 -0734 00 2 00000         PDX     0,2             GET COUNT.                                 CIOP1110
00165  0771 00 0 00022         ARS     18              MOVE TO ADDRESS.                           CIOP1120
00166  0361 00 4 00001         ACL     1,4             FORM END ADDRESS.                          CIOP1130
00167  0621 00 0 00174         STA     GET             INSERT IN PICKUP.                          CIOP1140
00170  0754 00 2 00000         PXA     0,2             GET COUNT.                                 CIOP1150
00171  0361 00 0 00203         ACL     PIO             FORM BUFFER END ADDRESS.                   CIOP1160
00172  0621 00 0 00175         STA     GIVE            INSERT IN STORE.                           CIOP1170
00173  0060 00 0 00173         TCOA    *               WAIT FOR LAST PRINT TO FINISH.             CIOP1180
00174 -0500 00 2 00000   GET   CAL     **,2            MOVE DATA TO OUTPUT BUFFER.                CIOP1190
00175  0602 00 2 00000   GIVE  SLW     **,2            ..                                         CIOP1200
00176  2 00001 2 00174         TIX     *-2,2,1         ..                                         CIOP1210
00177  0766 00 0 01203         WTDA    3               WRITE SELECT OUTPUT TAPE.                  CIOP1220
00200  0540 00 0 00203         RCHA    PIO             START CHANNEL.                             CIOP1230
00201  0774 00 2 00000   PX2   AXT     **,2            RESTORE IRS.                               CIOP1240
00202  0020 00 4 00002         TRA     2,4             RETURN TO CALLER.                          CIOP1250
                                                                                                  CIOP1260
00203  0 00000 0 00226   PIO   IOCD    PBUFF,,**       WRITE ONE RECORD AND STOP.                 CIOP1270
                                                                                                  CIOP1280
00204  0 00007 0 00205   WLNEX IOCD    *+1,,7          OUTPUT COMMENT IF ERROR.                   CIOP1290
```

```
      CIOP . . . BUFFERED I/O PACKAGE FOR CAP.  NO TAPE ERROR CHECKING.

00205  004751462751            BCI     7,GPROGRAMMER OUTPUT EXCEEDS 300 RECORDS.          CIOP1300
00206  214444255160
00207  466463476463
00210  602567232525
00211  246260030000
00212  605125234651
00213  246233606060
                                                                                          CIOP1310
                                GENERAL ERROR ROUTINE.                                    CIOP1320
                                MAKE SPECIFIED COMMENT, THEN RETURN TO MONITOR VIA EXIT.  CIOP1330
                                                                                          CIOP1340
00214  0634 00 2 00216   ERRCR SXA     *+2,2           INSERT COMMAND LOCATION IN RCHA.   CIOP1350
00215  0766 00 0 01203         WTDA    3               WRITE SELECT OUTPUT TAPE.          CIOP1360
00216  0540 00 0 00000         RCHA    **              OUTPUT APPROPRIATE COMMENT.        CIOP1370
00217  0074 00 4 00000         CALL    EXIT            RETURN TO MONITOR FOR DUMP.        CIOP1380
00220  1 00000 0 00222
00221  0 10403 0 00001
                                                                                          CIOP1390
                                STORAGE AND CONSTANTS.                                    CIOP1400
00222  0 00000 0 00454   LNCNT PZE     300             MAXIMUM LINECOUNT.                 CIOP1410
00223  0 00000 0 00000  FSTART PZE                     FLAG FOR FIRST CALL TO WCT1.       CIOP1420
00224  0 00000 0 00000   TREW  PZE                     FLAG FOR COLLATION TAPE REWOUND.   CIOP1430
00225  0 00000 0 00000   TAG   PZE                                                        CIOP1440
00226                    PBUFF BSS     14              PRINTER OUTPUT BUFFER.             CIOP1450
00244                    WBUFF BSS     14              COLLATION TAPE OUTPUT BUFFER.      CIOP1460
                                END                                                       CIOP1470

  LITERALS
00262  000000000001
00263  000016000000
00264  606060606060
```

```
          CIOP . . . BUFFERED I/O PACKAGE FOR CAP.  NO TAPE ERROR CHECKING.
                     POST PROCESSOR ASSEMBLY DATA

       265 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
       174     GET   167
       203     PIC   163,  171,  200
       201     PX2   154
        13     RX2     3
        55     SLW    62
       225     TAG    36,   37
        66     WX2    31
        67     WX4    30
         0    EXIT   217
       175    GIVE   172
       125    NREW   112
       122    R2X2   110
       224    TREW   105,  111
        30    WCT1
       214   ERROR    12,   51,  121,  161
       124   IOTIN   114,  116
        71   IOWCT    65
       222   LNCNT   156,  160
       226   PBUFF   203
        54   PCKUP    44,   57
       154   PRINT   125
         3   READ1
       110   READ2   130
       244   WBUFF    55,   71
        72   WEOCT    46
       142   WEOFC   117
        16   WEOFI    11
       204   WLNEX   155
       131   WNREW   126
       223  FSTART    32,   34
        15  INCARD     5,    7
       104  REWIND   127


NO ERROR IN ABOVE ASSEMBLY.
    *TIME SPENT IN FAP..    000007 IN HUNDREDTHS OF MINUTES.
```

```
      TEST OF CAP, BEGIN ASSEMBLY.
                          CAP     REM  THE FOLLOWING ARE ALL LEGAL CAP INSTRUCTIONS.                    TSTCAP00
                                  REM  PROGRAM TO COUNT BITS IN AC.                                     TSTCAP01
                                                                                                        TSTCAP02
     50000  056000050016  COUNT   LDQ  ZERO            ZERO TEST CELLS                                TSTCAP03
     50001  460000050021          STQ  BITS            ..                                             TSTCAP04
     50002  053400450020          LXA  THSX            COUNT 36 BITS.                                 TSTCAP05
     50003  076000000001  LOOP    LBT                  BIT OR NO.                                     TSTCAP06
     50004  002000050013          TRA  NO              NO BIT.                                        TSTCAP07
     50005  060200050022  YES     SLW  WORD            BIT, SAVE AC,                                  TSTCAP08
     50006  450000050021          CAL  BITS            AND INCREMENT COUNT.                           TSTCAP09
     50007  036100050017          ACL  ONE             ..                                             TSTCAP10
     50010  060200050021          SLW  BITS            ..                                             TSTCAP11
     50011  450000050022          CAL  WORD            RESTORE AC.                                    TSTCAP12
     50012  476500000001          LGR  1               NEXT BIT.                                      TSTCAP13
     50013  200001450003  NO      TIX  LOOP            INDEX.                                         TSTCAP14
     50014  450000050021          CAL  BITS            GET COUNT.                                     TSTCAP15
     50015  002100070000   DONE   OCTL 002100070000    STOP WITH TRANSFER TO 70000 OCTAL.             TSTCAP16
                                  REM  STORAGE.                                                       TSTCAP17
     50016  000000000000   ZERO   OCTL 000000000000    TRUE ZERO.                                     TSTCAP18
     50017  000001000000  ONE     INT  1               INCREMENT OF ONE.                              TSTCAP19
     50020  434000000044  THSX    LAS  36              ADDRESS IS 36.                                 TSTCAP20
                                  REM  DATA.                                                          TSTCAP21
     50021  000000000000  BITS    INT  0               STORAGE FOR BIT COUNT.                         TSTCAP22
     50022  000000000000  WORD    INT  0               TEMPORARY STORAGE FOR AC.                      TSTCAP23
                                  REM   TEST OF CAP PSEUDO-OPS, AND FLAGS.                            TSTCAP24
0    50023  000000000010          ILCD 8               ILLEGAL OPCODE.                                TSTCAP25
U    50024  002000000000          TRA  UNDEF           UNDEFINED SYMBOL.                              TSTCAP26
E    50025  000001000000          INT  1,2,-7,13A3,9   ERROR IN INTOP.                               TSTCAP27
OU   50032  000000000000          WMW  AEN             ILLEGAL OPCODE AND UNDEFINED SYMBOL.           TSTCAP28
                                  COMP   NO = YES + LOOP
     50033  050000050005          CLA  YES
     50034  030000050003          FAD  LOOP
     50035  060100050013          STO  NO
                           COMP   COMP     COUNT = LOOP * YES * NO / DONE / ZERO / ONE                 TSTCAP29
     50036  056000050003          LDQ  LOOP
     50037  026000050005          FMP  YES
     50040  013100000000          XCA
     50041  026000050013          FMP  NO
     50042  024100050015          FDP  DONE
     50043  013100000000          XCA
     50044  024100050016          FDP  ZERO
     50045  013100000000          XCA
     50046  024100050017          FDP  ONE
     50047  460000050110          STQ  TEM
     50050  050000050110          CLA  TEM
     50051  060100050000          STO  COUNT
     50052  002000050036          TRA  COMP            USE OF SYMBOL DEFINED BY COMP.                 TSTCAP30
                                  COMP     COUNT,1 = WORD,2 - DONE,4                                  TSTCAP31
     50053  050000050022          CLA  WORD,2
     50054  030200050015          FSB  DONE,4
     50055  060100050000          STO  COUNT,1
                                  COMP     WORD = (BITS+THSX*(ONE+THSX*(ZERO+THSX*(DONE+THSX*NO)))) TSTCAP32
     50056  056000050020          LDQ  THSX
     50057  026000050013          FMP  NO
     50060  060100050110          STO  TEM
     50061  050000050015          CLA  DONE
     50062  030000050110          FAD  TEM
     50063  060100050111          STO  TEM+1
     50064  056000050020          LDQ  THSX
     50065  026000050111          FMP  TEM+1
     50066  060100050112          STO  TEM+2
```

```
50067   050000050016          CLA   ZERO
50070   030000050112          FAD   TEM+2
50071   060100050113          STO   TEM+3
50072   056000050020          LDQ   THSX
50073   026000050113          FMP   TEM+3
50074   060100050114          STO   TEM+4
50075   050000050017          CLA   ONE
50076   030000050114          FAD   TEM+4
50077   060100050115          STO   TEM+5
50100   056000050020          LDQ   THSX
50101   026000050115          FMP   TEM+5
50102   060100050116          STO   TEM+6
50103   050000050021          CLA   BITS
50104   030000050116          FAD   TEM+6
50105   060100050117          STO   TEM+7
50106   050000050117          CLA   TEM+7
50107   060100050022          STO   WORD
                      TEM     REM   TEMPORARY STORAGE AREA BEGINS HERE.
                50000 END     END   COUNT          FINALLY THE END.          TSTCAP33

    RETURN FROM CAP, ENTRY POINT IS 50000.
```

Appendix B

PROGRAMS TO ALLOW USE OF CAP IN THE LABORATORY

This appendix contains FAP assembly listings of subprograms of the execution monitor and the I/Ø simulator used when CAP is used as a laboratory exercise. The listings are followed by a typical student output when running under the execution monitor. This output includes a storage map, CAP assembly listing, and postmortem.

Index to Appendix B

```
                MAIN PROGRAM FOR CAP.

                                                                                MAIN0010
                                PCC                                             MAIN0020
                                COUNT   8                                       MAIN0030
                                LBL     MAIN            BINARY CARD LABEL.

  TRANSFER VECTOR
00000  632562636260      TESTS

                                                                                MAIN0050
00001  0074 00 4 00000          TSX     $TESTS,4        GO TO TESTS WITH INDEX. MAIN0060
                   00002        DUP     1,3             THESE CARDS CANNOT BE DUPLICATED.
                                                                                MAIN0070
00002 -3 77777 7 77777          SVN     -1,7,-1         ..
C0003 -3 77777 7 77777
00004 -3 77777 7 77777
                                                                                MAIN0080
                                END


                POST PROCESSOR ASSEMBLY DATA

      5 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
        0  TESTS     1


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000002 IN HUNDREDTHS OF MINUTES.
```

```
TESTS FOR CAP, SWITCH FOR INTERVAL TIMER.

                                PCC
                                COUNT   12
                                LBL     TESTS           BINARY CARD LABEL.                      TESTS010
                      00004     ENTRY   TESTS           INTERLUCE TO TESTS).                    TESTS020
                                                                                                TESTS030
                                                                                                TESTS050
                                                                                                TESTS060
  TRANSFER VECTOR
C0000  633163606060     TIT
00001  632562636234     TESTS)

  LINKAGE DIRECTOR
C0002  000000000000
00003  632562636260

00004  0760 00 0 00161  TESTS SWT     1               TEST SWITCH ONE FOR INTERVAL TIMER USE. TESTS070
00005 -0625 60 0 00000          STL*    $TIT            SWITCH ONE UP, USE CORE CLOCK.          TESTS080
00006  0772 00 0 01206          REWA    6               REWIND UPDATE INPUT TAPE.               TESTS090
00007  0021 60 0 00001          TTR*    $TESTS)         THEN GO DIRECTLY TO TESTS).             TESTS100
                                                                                                TESTS110
                                END                                                             TESTS120


                    POST PROCESSOR ASSEMBLY DATA

      10 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
        0    TIT      5
        4  TESTS
        1 TESTS)      7


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000002 IN HUNDREDTHS OF MINUTES.
```

TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.

```
                PCC
                COUNT   1409
                LBL     TESTS)          BINARY CARD LABEL.                          TT)00010
                                                                                    TT)00020
                                                                                    TT)00030
                                                                                    TT)00050
                                                                                    TT)00060
00007           ENTRY   TESTS)          PRIMARY NAME OF MONITOR.                    TT)00070
00702           ENTRY   BACK            RETURN POINT IN CASE OF ERROR.              TT)00080
00235           ENTRY   TIT             TEST LOCATION TO USE INTERVAL TIMER.        TT)00090
00663           ENTRY   LSTM            LEAVE-SELECT-TRAPPING MODE.                 TT)00100
01145           ENTRY   SX4             ENTRY TO SAVE IR4 FOR POST MORTEM.          TT)00110
01146           ENTRY   SVCON           ENTRY TO SAVE CONSOLE FOR POST MORTEM.      TT)00120
02407           ENTRY   WOT             WRITE-OUTPUT-TAPE, A3.                      TT)00130
02520           ENTRY   NPRINT          ON LINE PRINT UNDER CARRIAGE CONTROL.       TT)00140
01026           ENTRY   EPMR            ENTRY TO MARK ERROR POST MORTEM.            TT)00150
01260           ENTRY   OCT)            FULL WORD OCTAL-BCI CONVERTER.              TT)00160
01173           ENTRY   OCTADR          OCTAL ADDRESS TO BCI CONVERTER.             TT)00170
01233           ENTRY   COMADR          COMPLEMENT ADDRESS TO BCI CONVERTER.        TT)00180
01207           ENTRY   TABBLK          TABLE FOR DELETION OF BLANKS WITH CRQ.      TT)00190
01301           ENTRY   BCDTAB          ENTRY TO REMOVE ILLEGAL BCI CHARACTERS.
```

```
                                TTL     SECTION 1, SETUP LOWER CORE AND PRINT STORAGE MAP.      TT)10000

  TRANSFER VECTOR
00000   444665312534      MOVIE)
00001   513147606060      RIP
00002   232147606060      CAP
00003   474751462760      PPROG
00004   472363016060      PCT1

  LINKAGE DIRECTOR
00005   000000000000
00006   632562636234

00007 -0760 00 0 00007  TESTS) LTM                     JUST IN CASE.                          TT)10010
00010 -0760 00 0 00002         EFTM                    ..                                     TT)10020
00011 -0760 00 0 00010         LSNM                    ..                                     TT)10030
00012  0600 00 0 00005         STZ     5               RESET INTERVAL TIMER.                  TT)10040
00013  0760 00 0 00012         DCT                     TURN OFF DCT LIGHT.                    TT)10050
00014  0761 00 0 00000         NOP                     ..                                     TT)10060
00015  0140 00 0 00016         TOV     *+1             TURN OFF OVERFLOW LIGHT.               TT)10070
00016  0441 00 0 03125         LDI     =0              CLEAR INDICATORS.                      TT)10080
00017  0760 00 0 00140         SLF                     TURN OFF SENSE LIGHTS.                 TT)10090
00020  0774 00 4 00144         AXT     100,4           WIPE OUT LOWER CORE.                   TT)10100
00021  0600 00 4 00144         STZ     100,4           ..                                     TT)10110
00022  0600 00 4 00145         STZ     101,4           ..                                     TT)10120
00023  2 00002 4 00021         TIX     *-2,4,2         ..                                     TT)10130
00024 -0500 00 0 00026         CAL     *+2             SET THE INITIAL TRAP WITH AN ENABLE.   TT)10140
00025  0602 00 0 00015         SLW     13              ..                                     TT)10150
00026  0564 00 0 03127         ENB     =2              THIS MAY TRAP.                         TT)10160
00027 -0500 00 0 00167         CAL     C(1)            SETUP LOWER CORE, ETM RETURN.          TT)10170
00030  0602 00 0 00001         SLW     1               ..                                     TT)10180
00031 -0500 00 0 00170         CAL     C(2)            STR RETURN.                            TT)10190
00032  0602 00 0 00002         SLW     2               ..                                     TT)10200
00033 -0500 00 0 00171         CAL     C(7)            SET INTERVAL TIMER RETURN.             TT)10210
00034  0602 00 0 00007         SLW     7               ..                                     TT)10220
00035 -0500 00 0 00172         CAL     C(8)            FPT RETURN.                            TT)10230
00036  0602 00 0 00010         SLW     8               ..                                     TT)10240
00037 -0500 00 0 00173         CAL     C(13)           CHANNEL B TRAP RETURN.                 TT)10250
00040  0602 00 0 00015         SLW     13              ..                                     TT)10260
00041 -0500 00 0 00174         CAL     C(ST)           SELECT TRAP RETURN.                    TT)10270
00042  0602 00 0 40001         SLW     16385           =/40001.                               TT)10280
00043  0602 00 0 40002         SLW     16386           =/40002.                               TT)10290
00044  0764 00 0 01203         BSRA    3               REMOVE MONITOR COMMENT OF EXECUTION.   TT)10300
00045  0060 00 0 00045         TCOA    *               WAIT FOR DSC A.                        TT)10310
00046  0760 00 0 00005         IOT                     TURN OFF I/O CHECK LIGHT.              TT)10320
00047  0761 00 0 00000         NOP                     ..                                     TT)10330
00050 -0760 00 0 01000         ETTA                    TURN OFF EOT LIGHT.                    TT)10340
00051  0761 00 0 00000         NOP                     ..                                     TT)10350

                               SPACE   2                                                      TT)10360
                               PRINT STORAGE MAP UP TO TESTS.                                 TT)10370
                                                                                              TT)10380
00052 -0500 60 0 00000  MAP    CAL*    $MOVIE)         GET CONTROL WORD OF MOVIE).            TT)10390
00053  0771 00 0 00022         ARS     18              CALCULATE LAST ADDRESS + 1.            TT)10400
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                     SECTION 1, SETUP LOWER CORE AND PRINT STORAGE MAP.

                                                                                                                    TT)10410
00054  0361 60 0 00000          ACL*   $MCVIE)          ..                                                        TT)10420
00055  0621 00 0 00073          STA    SRCH             COMPLETE ADDRESSES.                                       TT)10430
00056  0621 00 0 00076          STA    SRCH+3           ..                                                        TT)10440
00057  0402 00 0 03127          SUB    =2               ..                                                        TT)10450
00060  0621 00 0 00106          STA    MORG+4           ..                                                        TT)10460
00061  0400 00 0 03126          ADD    =1               ..                                                        TT)10470
00062  0621 00 0 00104          STA    MORG+2           ..                                                        TT)10480
00063  0400 00 0 03127          ADD    =2               ..                                                        TT)10490
00064  0621 00 0 00102          STA    MORG             ..                                                        TT)10500
00065  0774 00 2 00002          AXT    2,2              SKIP FIRST TWO WORDS OF MOVIE).                           TT)10510
00066  0774 00 1 00036          AXT    NSYM,1           SET MAXIMUM NUMBER OF ENTRIES.                            TT)10520
00067  0074 00 4 02407          TSX    WOT,4            COMMENT.                                                  TT)10530
00070  0 00006 0 00141          PZE    XMAP01,0,6       ..                                                        TT)10540
00071  0074 00 4 02407          TSX    WOT,4            COMMENT, HEADINGS.                                        TT)10550
00072  0 00004 0 00147          PZE    XMAP02,0,4       ..                                                        TT)10560
00073  0500 00 2 00000   SRCH   CLA    **,2             GET CURRENT WORD.                                         TT)10570
00074  0402 00 0 03162          SUB    =0700000000000   CHECK FOR SEVEN PREFIX.                                   TT)10580
00075  0100 00 0 00102          TZE    MORG             IF SO, EXIT TO PRINT NAME AND ORIGIN.                     TT)10590
00076 -0500 00 2 00000          CAL    **,2             IF NOT, CHECK TRANSFER VECTOR.                            TT)10600
00077 -0320 00 0 03164          ANA    =0777400777777   MASK OUT LARGEST POSSIBLE DECREMENT.                      TT)10610
00100  0100 00 0 00102          TZE    MORG             IF ZERO, EXIT TO PROGRAM ORIGIN.                          TT)10620
00101  1 00002 2 00073          TXI    SRCH,2,2         NOT AN ORIGIN, BACK UP BY 2.                              TT)10630
00102 -0500 00 2 00000   MORG   CAL    **,2             PROGRAM ORIGIN, GET ABSOLUTE ORIGIN.                      TT)10640
00103  0621 00 1 03067          STA    ORIGIN+NSYM,1    STORE IN TABLE.                                           TT)10650
00104 -0500 00 2 00000          CAL    **,2             GET ENTRY POINT.                                          TT)10660
00105  0621 00 1 03125          STA    ENTRY+NSYM,1     STORE IN TABLE.                                           TT)10670
00106 -0500 00 2 00000          CAL    **,2             GET BCI NAME.                                             TT)10680
00107 -0100 00 0 00111          TNZ    *+2              IF ALL ZEROES,                                            TT)10690
00110 -0500 00 0 03163          CAL    =H(MAIN)         MUST BE MAIN PROGRAM.                                     TT)10700
00111  0602 00 1 03031          SLW    NAME+NSYM,1      STORE IN TABLE.                                           TT)10710
00112  0602 00 0 00154          SLW    XMAP03+1         INSERT IN COMMENT.                                        TT)10720
00113  0560 00 1 03067          LDQ    ORIGIN+NSYM,1    GET ORIGIN AND CONVERT TO OCTAL-BCI.                      TT)10730
00114  0074 00 4 01173          TSX    OCTADR,4         ..                                                        TT)10740
00115 -0600 00 0 00155          STQ    XMAP03+2         INSERT IN COMMENT.                                        TT)10750
00116  0560 00 1 03125          LDQ    ENTRY+NSYM,1     GET ENTRY POINT AND CONVERT TO OCTAL-BCI.                 TT)10760
00117  0074 00 4 01173          TSX    OCTADR,4         ..                                                        TT)10770
00120 -0600 00 0 00156          STQ    XMAP03+3         INSERT IN COMMENT.                                        TT)10780
00121  0074 00 4 02407          TSX    WOT,4            WRITE THIS COMMENT.                                       TT)10790
00122  0 00004 0 00153          PZE    XMAP03,0,4       ..                                                        TT)10800
00123 -0500 00 1 03031          CAL    NAME+NSYM,1      IF THIS NAME IS TESTS, QUIT.                              TT)10810
00124 -0340 00 0 03161          LAS    =HTESTS          ..                                                        TT)10820
00125  0020 00 0 00127          TRA    *+2              NO, CONTINUE.                                             TT)10830
00126  0020 00 0 00133          TRA    TMAP             YES, EXIT.                                                TT)10840
00127 -2 00001 1 00131          TNX    EXCM,1,1         NO, COUNT ENTRIES IN TABLE.                               TT)10850
00130  1 00004 2 00073          TXI    SRCH,2,4         SKIP OVER FOUR AND TRY AGAIN.                             TT)10860
00131  0074 00 4 02407   EXCM   TSX    WOT,4            COMMENT, SYMBOL TABLE EXCEEDED.                           TT)10870
00132  0 00004 1 00157          PZE    XMAP04,1,4       ..                                                        TT)10880
00133 -0634 00 1 01447   TMAP   SXD    LSUB,1           SAVE COUNT OF SYMBOL TABLE FOR POST MORTEM.               TT)10890
00134  0074 00 4 02407          TSX    WOT,4            COMMENT, END OF SYMBOL TABLE.                             TT)10900
00135  0 00004 0 00163          PZE    XMAP05,0,4       ..                                                        TT)10910
00136  0061 00 0 00136          TCOT   *                WAIT FOR TIMING CHANNEL.                                  TT)10920
00137  0140 00 0 00175          TOV    EVAL             TURN OFF OVF LIGHT,                                       TT)10930
00140  0020 00 0 00175          TRA    EVAL             AND GO TO EVAL.                                           TT)10940
                                                                                                                    TT)10950
00141  006264224751      XMAP01 BCI    6,0SUBPROGRAM STORAGE MAP FOLLOWS.
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                     SECTION 1, SETUP LOWER CORE AND PRINT STORAGE MAP.

00142  462751214460
00143  626346512127
00144  256044214760
00145  264643434666
00146  623360606060
00147  006060606060      XMAP02 BCI      4,0      NAME  ORIGIN ENTRY                    TT)10960
00150  452144256060
00151  465131273145
00152  602545635170
00153  006060606060      XMAP03 BCI      4,0                                            TT)10970
00154  606060606060
00155  606060606060
00156  606060606060
00157  006270442246      XMAP04 BCI      4,0SYMBOL TABLE EXCEEDED.                      TT)10980
00160  436063212243
00161  256025672325
00162  252425243360
00163  002545246046      XMAP05 BCI      4,0END OF STORAGE MAP.                         TT)10990
00164  266062634651
00165  212725604421
00166  473360606060
00167  0021 00 0 00262    C(1)  TTR      TRAPR          ETM TRAP RETURN.                TT)11000
00170  0021 00 0 00314    C(2)  TTR      STRR           STR RETURN.                     TT)11010
00171  0021 00 0 00524    C(7)  TTR      TIMR           INTERVAL TIMER RETURN.          TT)11020
00172  0021 00 0 00404    C(8)  TTR      FPTR           FPT RETURN.                     TT)11030
00173  0021 00 0 00533    C(13) TTR      SLR            STOP-LOOP RETURN.               TT)11040
00174  0021 00 0 00627    C(ST) TTR      IOTMR          SELECT TRAP RETURN.             TT)11050
                                                                                        TT)11060
                   02223  T     TAPENO   B3B            TAPE FOR TIMER.                 TT)11070
```

```
                                    TTL     SECTION 2, SET TRAPS, TIME AND EXIT TO CAP.          TT)20000
                                                                                                 TT)20010
00175  0074 00 4 02407    EVAL    TSX     WOT,4          EXECUTION COMMENT.                    TT)20020
00176  0 00007 0 00241             PZE     TCAP,0,7       ..                                    TT)20030
00177  0074 00 4 02407             TSX     WOT,4          BLANK LINE.                           TT)20040
00200  0 00001 0 03160             PZE     =H      ,0,1   ..                                    TT)20050
00201  0074 00 4 00001             TSX     $RIP,4         GO SET UP READ PROGRAMS.              TT)20060
00202  0520 00 0 00235             ZET     TIT            TEST FOR INTERVAL TIMER.              TT)20070
00203  0020 00 0 00207             TRA     ITIM           GO TO INTERVAL TIMER.                 TT)20080
00204  0766 00 0 02223             WTBT                   TAPE TIMER, SELECT TIMER TAPE.        TT)20090
00205 -0540 00 0 00237             RCHT    TIME           LOAD WITH TIME.                       TT)20100
00206  0020 00 0 00213             TRA     ESTM           GO TO SET TRAP.                       TT)20110
00207 -0500 00 0 03136    ITIM    CAL     =300           SET INTERVAL TIMER FOR FIVE SECONDS.  TT)20120
00210  0760 00 0 00006             COM                    ..                                    TT)20130
00211  0601 00 0 00005             STO     5              INSERT IN TRAP LOCATION.              TT)20140
00212  0600 00 0 00006             STZ     6              CLEAR LOCATION RETURN.                TT)20150
00213 -0760 00 0 00005    ESTM    ESTM                   SET TRAP.                             TT)20160
00214 -0500 00 0 00234             CAL     ORG            SET ORIGIN.                           TT)20170
00215  0074 00 4 00002             TSX     $CAP,4         GO TO CAP.                            TT)20180
00216  0621 00 0 00236             STA     EXEC           SAVE ENTRY POINT.                     TT)20190
00217 -0760 00 0 00007             LTM                    JUST IN CASE.                         TT)20200
00220  0634 00 4 01166             SXA     IR4,4          SAVE IR4.                             TT)20210
00221  0074 00 4 01146             TSX     SVCON,4        SAVE CONSOLE.                         TT)20220
00222  0560 00 0 00236             LDQ     EXEC           CONVERT ENTRY TO OCTAL-BCI.           TT)20230
00223  0074 00 4 01173             TSX     OCTADR,4       ..                                    TT)20240
00224 -0754 00 0 00000             PXD     0,0            SHIFT AND OR TO COMMENT.              TT)20250
00225 -0763 00 0 00036             LGL     30             ..                                    TT)20260
00226 -0602 00 0 00257             ORS     WEP+7          ..                                    TT)20270
00227 -0130 00 0 00000             XCL                    ..                                    TT)20280
00230 -0602 00 0 00260             ORS     WEP+8          ..                                    TT)20290
00231  0074 00 4 02407             TSX     WOT,4          COMMENT, ENTRY POINT.                 TT)20300
00232  0 00011 0 00250             PZE     WEP,0,9        ..                                    TT)20310
00233  0020 00 0 00702             TRA     BACK           NO PROVISION FOR EXECUTION FOR NOW.   TT)20320
                                                                                                 TT)20330
00234 +000000050000       ORG     OCT     50000          CAP PROGRAM ORIGIN.                   TT)20340
00235  0 00000 0 00000    TIT     PZE                    LOCATION TO TEST FOR INTERVAL TIMER.  TT)20350
00236  0 00000 0 00000    EXEC    PZE                    DUMMY TRA TO ASSEMBLED PROGRAM.       TT)20360
00237 -0 60650 0 17130    TIME    IOCP    -25000,0,25000 TIMED FOR ABOUT 5 SECONDS.            TT)20370
00240 -1 60650 0 17130            IOCT    -25000,0,25000 ..                                    TT)20380
00241  016060606060       TCAP    BCI     7,1            TEST OF CAP, BEGIN ASSEMBLY.          TT)20390
00242  606060606060
00243  632562636046
00244  266023214773
00245  602225273145
00246  602162622544
00247  224370336060
00250  006060606060       WEP     BCI     9,0            RETURN FROM CAP, ENTRY POINT IS 00000.TT)20400
00251  606060606060
00252  512563645145
00253  602651464460
00254  232147736025
00255  456351706047
00256  463145636031
00257  626000000000
00260  003360606060
```

```
                                    TTL     SECTION 3, TRAP RETURNS AND LSTM.                                TT)30000
                                                                                                             TT)30010
00261  0 00000 0 00000   QUIT     IOCD    0,0,0           COMMAND TO DISCONNECT CHANNEL.                    TT)30020


                                    SPACE   2                                                                TT)30030
                                            TRAPPING MODE RETURN.                                            TT)30040
                                                                                                             TT)30050
00262 -0760 00 0 00007   TRAPR    LTM                     RETURN FROM TRANSFER TRAP.                       TT)30060
00263  0634 00 4 01166            SXA     IR4,4           SAVE IR4.                                        TT)30070
00264  0074 00 4 01146            TSX     SVCON,4         SAVE CONSOLE.                                    TT)30080
00265 -0520 00 0 00000            NZT     0               IF LOCATION 0 NOT SET,                           TT)30090
00266  0020 00 0 00361            TRA     SEQR            MUST BE A WILD TRANSFER.                         TT)30100
00267  0560 00 0 00000            LDQ     0               GET ADDRESS OF TRANSFER INSTRUCTION.             TT)30110
00270  0074 00 4 01173            TSX     OCTADR,4        CONVERT TO OCTAL-BCD.                            TT)30120
00271 -0754 00 0 00000            PXD     0,0             ..                                               TT)30130
00272 -0763 00 0 00022            LGL     18              SHIFT AND OR TO COMMENT.                         TT)30140
00273 -0602 00 0 00307            ORS     XTRAP+5         ..                                               TT)30150
00274 -0130 00 0 00000            XCL                     ..                                               TT)30160
00275 -0602 00 0 00310            ORS     XTRAP+6         ..                                               TT)30170
00276  0074 00 4 02407            TSX     WOT,4           COMMENT.                                         TT)30180
00277  0 00012 1 00302            PZE     XTRAP,1,10      ..                                               TT)30190
00300 -0625 00 0 01026            STL     EPMR            SET TO GIVE ERROR POST MORTEM.                   TT)30200
00301  0020 00 0 00702            TRA     BACK            EXIT TO POST MORTEM SECTION.                     TT)30210
                                                                                                             TT)30220
00302  006351214562       XTRAP  BCI     9,0TRANSFER INSTRUCTION IN LOCATION 00000 HAS BEEN TRAPP         TT)30230
00303  262551603145
00304  626351642363
00305  314645603145
00306  604346232163
00307  314645600000
00310  000000603021
00311  626022252545
00312  606351214747
00313  252433606060               BCI     1,ED.                                                            TT)30240


                                    SPACE   2                                                                TT)30250
                                            STR RETURN.                                                      TT)30260
                                                                                                             TT)30270
00314 -0760 00 0 00007   STRR     LTM                     RETURN FOR TRAPPED STR.                          TT)30280
00315  0634 00 4 01166            SXA     IR4,4           SAVE IR4.                                        TT)30290
00316  0074 00 4 01146            TSX     SVCON,4         SAVE CONSOLE.                                    TT)30300
00317 -0520 00 0 00000            NZT     0               IF LOCATION 0 NOT SET,                           TT)30310
00320  0020 00 0 00361            TRA     SEQR            MUST BE A WILD TRANSFER.                         TT)30320
00321  0534 00 4 00000            LXA     0,4             GET A(STR)+1.                                    TT)30330
00322  1 77777 4 00323            TXI     *+1,4,-1        DECREMENT IT.                                    TT)30340
00323  0634 00 4 00324            SXA     *+1,4           ..                                               TT)30350
00324  0560 00 0 00000            LDQ     0               CHECK FOR PROGRAMMED STR.                        TT)30360
00325 -0754 00 0 00000            PXD     0,C             ..                                               TT)30370
00326 -0763 00 0 00003            LGL     3               ..                                               TT)30380
00327  0402 00 0 03131            SUB     =5              ..                                               TT)30390
00330  0100 00 0 00334            TZE     *+4             YES.                                             TT)30400
00331 -0500 00 0 00000            CAL     0               IF NOT, STOP OR LOOP.                            TT)30410
00332  0621 00 0 00610            STA     SADR                                                             TT)30420
```

```
00333  0020 00 0 00541          TRA     SLR1            ..                                      TT)30430
                                                                                                 TT)30440
00334  0754 00 4 00000          PXA     0,4             PROGRAMMED STR.                         TT)30450
00335 -0130 00 0 00000          XCL                     ..                                      TT)30460
00336  0074 00 4 01173          TSX     OCTADR,4        CONVERT TO OCTAL-BCD.                   TT)30470
00337 -0754 00 0 00000          PXD     0,0             ..                                      TT)30480
00340 -0763 00 0 00014          LGL     12              SHIFT AND OR TO COMMENT.                TT)30490
00341 -0602 00 0 00354          ORS     XSTR+4          ..                                      TT)30500
00342 -0130 00 0 00000          XCL                     ..                                      TT)30510
00343 -0602 00 0 00355          ORS     XSTR+5          ..                                      TT)30520
00344  0074 00 4 02407          TSX     WOT,4           COMMENT.                                TT)30530
00345  0 00011 1 00350          PZE     XSTR,1,9        ..                                      TT)30540
00346 -0625 00 0 01026          STL     EPMR            SET TO GIVE ERROR POST MORTEM.          TT)30550
00347  0020 00 0 00702          TRA     BACK            EXIT TO POST MORTEM SECTION.            TT)30560
                                                                                                 TT)30570
00350  006263516031      XSTR  BCI     9,0STR INSTRUCTION IN LOCATION 00000 HAS BEEN TRAPPED.  TT)30580
00351  456263516423
00352  633146456031
00353  456043462321
00354  633146456000
00355  000000006030
00356  216260222525
00357  456063512147
00360  472524336060

                               SPACE   2                                                       TT)30590
                               ENTRY FOR SEQUENCING ERROR.                                     TT)30600
                                                                                               TT)30610
00361  0074 00 4 02407   SEQR  TSX     WOT,4           COMMENT FOR BAD CALLING SEQUENCE.       TT)30620
00362  0 00017 1 00365          PZE     XSEQR,1,15      ..                                      TT)30630
00363 -0625 00 0 01026          STL     EPMR            SET TO GIVE ERROR POST MORTEM.          TT)30640
00364  0020 00 0 00702          TRA     BACK            EXIT TO POST MORTEM SECTION.            TT)30650
                                                                                                 TT)30660
00365  006351214562      XSEQR BCI     9,0TRANSFER TO LOWER CORE. PROBABLY IR4 NOT RESET, OR TA TT)30670
00366  262551606346
00367  604346662551
00370  602346512533
00371  604751462221
00372  224370603151
00373  046045466360
00374  512562256373
00375  604651606321
00376  276044316262          BCI     6,G MISSING FROM TRANSFER INSTRUCTION.                  TT)30680
00377  314527602651
00400  464460635121
00401  456226255160
00402  314562635164
00403  236331464533

                               SPACE   2                                                       TT)30690
                               FLOATING POINT TRAP RETURN.                                     TT)30700
                                                                                               TT)30710
00404 -0520 00 0 00000   FPTR  NZT     0               IF LOCATION 0 NOT SET,                  TT)30720
```

```
00405  0021 00 0 00425          TTR     TR8             MUST BE A WILD TRANSFER.                        TT)30730
00406  0604 00 0 00473          STI     SSI             SAVE THE INDICATORS.                            TT)30740
00407  0441 00 0 00000          LDI     0               GET SPILL CODE.                                 TT)30750
00410 -0054 00  000004          LFT     4               CHECK FOR OVERFLOW.                             TT)30760
00411  0021 00 0 00431          TTR     FPER            OVERFLOW, EXIT.                                 TT)30770
00412 -0054 00  000002          LFT     2               AC UNDERFLOW.                                   TT)30780
00413  0500 00 0 03125          CLA     =0              YES, RESET AC.                                  TT)30790
00414 -0054 00  000001          LFT     1               MQ UNDERFLOW.                                   TT)30800
00415  0560 00 0 03125          LDQ     =0              YES, RESET MQ.                                  TT)30810
00416  0441 00 0 00473          LDI     SSI             RESTORE ORIGINAL INDICATORS.                    TT)30820
00417  0634 00 4 00422          SXA     *+3,4           SAVE IR4.                                       TT)30830
00420  0534 00 4 00000          LXA     0,4             PICKUP 0.                                       TT)30840
00421  0634 00 4 00424          SXA     *+3,4           INSERT RETURN ADDRESS.                          TT)30850
00422  0774 00 4 00000          AXT     **,4            RESTORE IR4.                                    TT)30860
00423  0600 00 0 00000          STZ     0               CLEAR LOCATION 0.                               TT)30870
00424  0021 00 0 00000          TTR     **              RETURN TO CALLER.                               TT)30880
                                                                                                         TT)30890
00425 -0760 00 0 00007  TR8     LTM                     JUST IN CASE.                                   TT)30900
00426  0634 00 4 01166          SXA     IR4,4           SAVE IR4 FOR POST MORTEM.                       TT)30910
00427  0074 00 4 01146          TSX     SVCON,4         GO SAVE CONSOLE AND RESET TRAPS.                TT)30920
00430  0020 00 0 00361          TRA     SEQR            GO TO SEQR ROUTINE.                             TT)30930
                                                                                                         TT)30940
00431 -0760 00 0 00007  FPER    LTM                     JUST IN CASE.                                   TT)30950
00432  0634 00 4 01166          SXA     IR4,4           SAVE IR4.                                       TT)30960
00433  0441 00 0 00473          LDI     SSI             RESTORE INDICATORS.                             TT)30970
00434  0074 00 4 01146          TSX     SVCON,4         SAVE CONSOLE.                                   TT)30980
00435  0441 00 0 00000          LDI     0               GET SPILL CODE AND ADDRESS OF INSTRUCTION.      TT)30990
00436 -0056 00  000010          LNT     10              DIVISION ERROR.                                 TT)31000
00437  0020 00 0 00443          TRA     *+4             NO, SKIP.                                       TT)31010
00440  0074 00 4 02407          TSX     WOT,4           YES, COMMENT.                                   TT)31020
00441  0 00003 1 00474          PZE     WDVER,1,3       ..                                              TT)31030
00442  0020 00 0 00453          TRA     SPER            EXIT.                                           TT)31040
00443 -0056 00  000002          LNT     2               ACCUMULATOR OVERFLOW.                           TT)31050
00444  0020 00 0 00447          TRA     *+3             NO, SKIP.                                       TT)31060
00445  0074 00 4 02407          TSX     WOT,4           YES, COMMENT.                                   TT)31070
00446  0 00004 1 00477          PZE     WACO,1,4        ..                                              TT)31080
00447 -0056 00  000001          LNT     1               MQ OVERFLOW.                                    TT)31090
00450  0020 00 0 00453          TRA     SPER            NO, EXIT.                                       TT)31100
00451  0074 00 4 02407          TSX     WOT,4           YES, COMMENT.                                   TT)31110
00452  0 00005 1 00503          PZE     WMQO,1,5        ..                                              TT)31120
00453 -0500 00 0 00000  SPER    CAL     0               GET SPILL CODE.                                 TT)31130
00454 -0320 00 0 03141          ANA     =0C00017000000  MASK OUT JUNK.                                  TT)31140
00455 -0765 00 0 00025          LGR     21              CONVERT TO OCTAL.                               TT)31150
00456  0767 00 0 00003          ALS     3               SHIFT AND OR TO COMMENT.                        TT)31160
00457 -0763 00 0 00003          LGL     3               ..                                              TT)31170
00460 -0602 00 0 00522          ORS     XFPT+10         ..                                              TT)31180
00461 -0500 00 0 00000          CAL     0               GET ADDRESS OF TRAPPED INSTRUCTION.             TT)31190
00462  0402 00 0 03126          SUB     =1              ..                                              TT)31200
00463 -0130 00 0 00000          XCL                     ..                                              TT)31210
00464  0074 00 4 01173          TSX     OCTADR,4        CONVERT TO OCTAL-BCD.                           TT)31220
00465 -0130 00 0 00000          XCL                     OR TO COMMENT.                                  TT)31230
00466 -0602 00 0 00517          ORS     XFPT+7          ..                                              TT)31240
00467  0074 00 4 02407          TSX     WOT,4           COMMENT.                                        TT)31250
00470  0 00014 1 00510          PZE     XFPT,1,12       ..                                              TT)31260
00471 -0625 00 0 01026          STL     EPMR            SET TO GIVE ERROR POST MORTEM.                  TT)31270
```

```
00472  0020 00 0 00702          TRA     BACK          EXIT TO POST MORTEM SECTION.          TT)31280
                                                                                             TT)31290
00473  0 00000 0 00000    SSI   PZE                   TEMPORARY STORAGE FOR SI.             TT)31300
00474  002431653162       WDVER BCI     3,0DIVISION ERROR.                                  TT)31310
00475  314645602551
C0476  514651336060
00477  002123236444       WACO  BCI     4,0ACCUMULATOR OVERFLOW.                            TT)31320
00500  644321634651
00501  604665255126
00502  434666336060
00503  004464436331       WMQO  BCI     5,0MULTIPLIER-QUOTIENT OVERFLOW.                    TT)31330
00504  474331255140
00505  506446633125
00506  456360466525
00507  512643466633
00510  002643462163       XFPT  BCI     9,0FLOATING POINT SPILL OCCURRED IN LOCATION 00000, SPIL TT)31340
00511  314527604746
00512  314563606247
00513  314343604623
00514  236451512524
00515  603145604346
00516  232163314645
00517  600000000000
00520  736062473143
00521  436023462425             BCI     3,L CODE IS 00.                                     TT)31350
00522  603162600000
00523  336060606060

                                SPACE   2                                                   TT)31360
                                        INTERVAL TIMER RETURN.                              TT)31370
                                                                                            TT)31380
00524 -0760 00 0 00007   TIMR   LTM                    JUST IN CASE.                        TT)31390
00525  0634 00 4 01166          SXA     IR4,4         SAVE IR4 FOR POST MORTEM.            TT)31400
00526  0074 00 4 01146          TSX     SVCON,4       SAVE CONSOLE AND RESET TRAPS.        TT)31410
00527 -0500 00 0 00006          CAL     6             PICKUP TRAP LOCATION.                TT)31420
00530  0100 00 0 00361          TZE     SEQR          IF ZERO, MUST BE WILD TRANSFER.      TT)31430
00531  0621 00 0 00610          STA     SADR                                               TT)31440
00532  0020 00 0 00541          TRA     SLR1          GO TO ANALYZE.                       TT)31450

                                SPACE   2                                                   TT)31460
                                        TAPE TIMER RETURN.                                  TT)31470
                                                                                            TT)31480
00533 -0760 00 0 00007   SLR    LTM                    STOP-LOOP TRAPPED RETURN.            TT)31490
00534  0634 00 4 01166          SXA     IR4,4         SAVE IR4.                            TT)31500
00535  0074 00 4 01146          TSX     SVCON,4       SAVE CONSOLE.                        TT)31510
00536 -0500 00 0 00014          CAL     12                                                 TT)31520
00537  0100 00 0 00361          TZE     SEQR                                               TT)31530
00540  0621 00 0 00610          STA     SADR                                               TT)31540
00541 -0500 60 0 00610   SLR1   CAL*    SADR                                               TT)31550
00542 -0320 00 0 03152          ANA     =0377700000000  MASK IT.                          TT)31560
00543  0100 00 0 00571          TZE     PSTOP         IF HTR OR FOR, CALL IT A STOP.       TT)31570
00544 -0500 00 0 00610          CAL     SADR                                               TT)31580
00545  0402 00 0 03126          SUB     =1            REDUCE C(ILC) BY 1.                  TT)31590
```

```
00546   0621 00 0 00610          STA     SADR            ..                                      TT)31600
00547  -0500 60 0 00610          CAL*    SADR            GET TRAPPED INSTRUCTION.                TT)31610
00550  -0320 00 0 03165          ANA     =0777700000000  MASK IT.                               TT)31620
00551   0774 00 4 00004          AXT     4,4             COMPARE WITH TRUE STOPS.                TT)31630
00552  -0340 00 4 00610          LAS     PSTOPS+4,4      ..                                      TT)31640
00553   0020 00 0 00555          TRA     *+2             NO.                                     TT)31650
00554   0020 00 0 00571          TRA     PSTOP           TRUE STOP.                              TT)31660
00555   2 00001 4 00552          TIX     *-3,4,1         NO, INDEX.                              TT)31670
                                                                                                 TT)31680
00556   0560 00 0 00610  LOOP    LDQ     SADR                                                    TT)31690
00557   0074 00 4 01173          TSX     OCTADR,4        CONVERT TO OCTAL BCD.                   TT)31700
00560  -0754 00 0 00000          PXD     0,0             ..                                      TT)31710
00561  -0763 00 0 00030          LGL     24              SHIFT AND OR TO COMMENT.                TT)31720
00562  -0602 00 0 00617          ORS     XLCOP+6         ..                                      TT)31730
00563  -0130 00 0 00000          XCL                     ..                                      TT)31740
00564  -0602 00 0 00620          ORS     XLCOP+7         ..                                      TT)31750
00565   0074 00 4 02407          TSX     WOT,4           COMMENT.                                TT)31760
00566   0 00010 1 00611          PZE     XLOOP,1,8       ..                                      TT)31770
00567  -0625 00 0 01026          STL     EPMR            SET TO GIVE ERROR POST MORTEM.          TT)31780
00570   0020 00 0 00702          TRA     BACK            EXIT TO POST MORTEM SECTION.            TT)31790
                                                                                                 TT)31800
00571   0560 00 0 00610  PSTOP   LDQ     SADR            TRUE STOP, GET ADDRESS OF TRAPPED       TT)31810
00572   0074 00 4 01173          TSX     OCTADR,4        INSTRUCTION AND CONVERT TO OCTAL-BCD.   TT)31820
00573  -0754 00 0 00000          PXD     0,0             ..                                      TT)31830
00574  -0763 00 0 00036          LGL     30              SHIFT AND OR TO COMMENT.                TT)31840
00575  -0602 00 0 00625          ORS     XPSTOP+4        ..                                      TT)31850
00576  -0130 00 0 00000          XCL                     ..                                      TT)31860
00577  -0602 00 0 00626          ORS     XPSTOP+5        ..                                      TT)31870
00600   0074 00 4 02407          TSX     WOT,4           COMMENT.                                TT)31880
00601   0 00006 1 00621          PZE     XPSTOP,1,6      ..                                      TT)31890
00602  -0625 00 0 01026          STL     EPMR            SET TO GIVE ERROR POST MORTEM.          TT)31900
00603   0020 00 0 00702          TRA     BACK            EXIT TO POST MORTEM SECTION.            TT)31910
                                                                                                 TT)31920
00604   0220 00 0 00000  PSTOPS  DVH     -               LIST OF STOPS.                          TT)31930
00605   0224 00 0 00000          VDH     -,,-            ..                                      TT)31940
00606   0240 00 0 00000          FDH     -               ..                                      TT)31950
00607   0420 00 0 00000          HPR                     ..                                      TT)31960
00610   0 00000 0 00000  SADR    PZE                     ADDRESS OF STOP.                        TT)31970
00611   004751462221     XLOOP   BCI     8,0PROBABLE ENDLESS LCOP AROUND LOCATION 00000.         TT)31980
00612   224325602545
00613   244325626260
00614   434646476021
00615   514664452460
00616   434623216331
00617   464560000000
00620   000033606060
00621   004751462751     XPSTOP  BCI     6,0PROGRAM STOP AT LOCATION 00000.                      TT)31990
00622   214460626346
00623   476021636043
00624   462321633146
00625   456000000000
00626   003360606060
```

```
TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
      SECTION 3, TRAP RETURNS AND LSTM.

                                    SPACE   2                                                          TT)32000
                                            SELECT OR COPY TRAP RETURN.                                TT)32010
                                                                                                       TT)32020
                                                                                                       TT)32030
00627 -0760 00 0 00007  IOTMR LTM           JUST IN CASE.                                              TT)32040
00630  0634 00 4 01166        SXA   IR4,4    SAVE IR4.                                                  TT)32050
00631  0074 00 4 01146        TSX   SVCON,4  SAVE THE CONSOLE.                                          TT)32060
00632 -0760 00 0 00010        LSNM           JUST IN CASE.                                              TT)32070
00633  0534 00 4 40000        LXA   16384,4  GET (ADDRESS+1) OF TRAPPED INSTRUCTION.                    TT)32080
00634  1 77777 4 00635        TXI   *+1,4,-1 SET TO ADDRESS OF TRAPPED INSTRUCTION.                     TT)32090
00635  0754 00 4 00000        PXA   0,4      TO AC.                                                     TT)32100
00636 -0130 00 0 00000        XCL            TO MQ.                                                     TT)32110
00637  0074 00 4 01173        TSX   OCTADR,4 CONVERT TO OCTAL-BCD.                                      TT)32120
00640 -0754 00 0 00000        ZAC            INSERT IN COMMENT.                                         TT)32130
00641 -0763 00 0 00022        LGL   18       ..                                                         TT)32140
00642 -0602 00 0 00661        ORS   XIOBD+8  ..                                                         TT)32150
00643 -0130 00 0 00000        XCL            ..                                                         TT)32160
00644 -0602 00 0 00662        ORS   XICBD+9  ..                                                         TT)32170
00645  0074 00 4 02407        TSX   WOT,4    COMMENT.                                                   TT)32180
00646  0 00012 1 00651        PZE   XIOBD,1,10 ..                                                       TT)32190
00647 -0625 00 0 01026        STL   EPMR     SET TO GIVE ERROR POST MORTEM.                             TT)32200
00650  0020 00 0 00702        TRA   BACK     EXIT TO POST MORTEM SECTION.                               TT)32210

00651  002163632544     XIOBD BCI   9,0ATTEMPT TO USE ILLEGAL I/O INSTRUCTION IN LOCATION 00           TT)32220
00652  476360634660
00653  646225603143
00654  432527214360
00655  316146603145
00656  626351642363
00657  314645603145
00660  604346232163
00661  314645600000
00662  000000336060           BCI   1,000.                                                            TT)32230

                                    SPACE   2                                                          TT)32240
                                            ROUTINE TO LEAVE SELECT TRAPPING MODE.                     TT)32250
                                                                                                       TT)32260
                                                                                                       TT)32270
00663 -0600 00 0 00677  LSTM  STQ   SLSTM      SAVE MQ.                                               TT)32280
00664 -0760 00 0 00010        LSNM             JUST IN CASE.                                          TT)32290
00665  0560 00 0 40001        LDQ   16385      SAVE TRAP RETURN.                                      TT)32300
00666 -0600 00 0 00700        STQ   SLSTM+1    ..                                                     TT)32310
00667  0560 00 0 00701        LDQ   TLSTM      SET TRAP.                                              TT)32320
00670 -0600 00 0 40001        STQ   16385      ..                                                     TT)32330
00671  0766 00 0 01204        WTDA  4          SELECT INSTRUCTION.                                    TT)32340
00672  0540 00 0 00261        RCHA  QUIT       NIL COMMAND.                                           TT)32350
00673  0560 00 0 00700  RLSTM LDQ   SLSTM+1    RESTORE ORIGINAL TRAP.                                 TT)32360
00674 -0600 00 0 40001        STQ   16385      ..                                                     TT)32370
00675  0560 00 0 00677        LDQ   SLSTM      RESTORE ORIGINAL MQ.                                   TT)32380
00676  0020 00 4 00001        TRA   1,4        RETURN.                                                TT)32390
                                                                                                       TT)32400
00677  0 00000 0 00000  SLSTM PZE              MQ STORAGE.                                            TT)32410
00700  0 00000 0 00000        PZE              TRAP RETURN STORAGE.                                   TT)32420
00701  0021 00 0 00673  TLSTM TTR   RLSTM      SELECT TRAP RETURN.
```

```
                                    TTL     SECTION 4, WIND UP THIS RUN.                                    TT)40000
                                                                                                            TT)40010
00702 -0760 00 0 00007   BACK      LTM                     JUST IN CASE.                                   TT)40020
00703  0600 00 0 00005              STZ     5               RESET INTERVAL TIMER.                           TT)40030
00704 -0061 00 0 00706              TCNT    *+2             IF TAPE TIMER IN USE,                           TT)40040
00705 -0540 00 0 00261              RCHT    QUIT            RESET TAPE TIMER.                               TT)40050
00706  0074 00 4 00663              TSX     LSTM,4          RESET TRAP, JUST IN CASE.                       TT)40060
00707 -0520 00 0 00235              NZT     TIT             IF TAPE TIMER USED,                             TT)40070
00710  0772 00 0 02223              REWT                    REWIND TIMER TAPE.                              TT)40080
00711  0520 00 0 01026              ZET     EPMR            TEST FOR ERROR POST MORTEM.                     TT)40090
00712  0020 00 0 01401              TRA     PMR             YES, EXIT.                                      TT)40100
00713  0020 00 0 01403              TRA     PMR+2           POST MORTEM IN ANY CASE.                        TT)40110
                                                                                                            TT)40120
                                    SECTION TO GIVE FINAL EXIT TO MONITOR.                                  TT)40130
                                                                                                            TT)40140
00714  0074 00 4 02407   FINIS     TSX     WOT,4           END OF RUN COMMENT.                             TT)40150
00715  0 00002 1 01027              PZE     ENDRN,1,2       ..                                              TT)40160
00716  0760 00 0 00004              ENK                     CHECK CONSOLE KEYS.                             TT)40170
00717 -0130 00 0 00000              XCL                     ..                                              TT)40180
00720  0044 00 0 00000              PAI                     ..                                              TT)40190
00721 -0056 00  040000              LNT     040000          IF KEY 3 IS UP,                                 TT)40200
00722  0020 00 0 00765              TRA     DOOR            DONT CHECK KEYS AT ALL.                         TT)40210
                                                                                                            TT)40220
C0723  0056 00  000002   STOP      RNT     2               CHECK FOR STOP INSTRUCTION.                     TT)40230
00724  0020 00 0 00732              TRA     FILE            NO, CHECK FOR REQUEST FOR EOF ON A3.            TT)40240
00725  0074 00 4 02520              TSX     PRINT,4         YES, COMMENT.                                   TT)40250
00726 -0 00014 0 01031              MZE     XSTOP,0,12      ..                                              TT)40260
00727  0760 00 0 00004              ENK                     GET THE KEYS.                                   TT)40270
00730 -0130 00 0 00000              XCL                     PLACE IN AC,                                    TT)40280
00731  0044 00 0 00000              PAI                     PLACE IN SI.                                    TT)40290
                                                                                                            TT)40300
00732  0056 00  000004   FILE      RNT     4               CHECK FOR EOF ON A3 REQUEST.                    TT)40310
00733  0020 00 0 00742              TRA     CLOSE           NO, CHECK FOR CLOSE TAPE INSTRUCTION.           TT)40320
00734  0770 00 0 01203              WEFA    3               YES, WRITE EOF ON A3.                           TT)40330
00735  0074 00 4 02520              TSX     PRINT,4         COMMENT.                                        TT)40340
00736 -0 00014 0 01045              MZE     XFILE,0,12      ..                                              TT)40350
00737  0760 00 0 00004              ENK                     GET THE KEYS.                                   TT)40360
00740 -0130 00 0 00000              XCL                     PLACE IN AC,                                    TT)40370
00741  0044 00 0 00000              PAI                     PLACE IN SI.                                    TT)40380
                                                                                                            TT)40390
00742  0056 00  000001   CLOSE     RNT     1               CHECK FOR CLOSE OUTPUT TAPE INSTRUCTION.        TT)40400
00743  0020 00 0 00765              TRA     DOOR            NO, GO TO DOOR.                                 TT)40410
00744  0774 00 1 00012              AXT     10,1            SET TO COMMENT 10 TIMES.                        TT)40420
00745  0074 00 4 02407              TSX     WOT,4           YES, WRITE (END OF TAPE.) AND                   TT)40430
00746  0 00003 0 01077              PZE     ENDTP,0,3       ..                                              TT)40440
00747  0074 00 4 02407              TSX     WOT,4           SKIP TO CHANNEL FOUR.                           TT)40450
00750  0 00001 0 03147              PZE     =H4     ,0,1    ..                                              TT)40460
00751  0770 00 0 01203              WEFA    3               THEN AN EOF MARK.                               TT)40470
00752  2 00001 1 00745              TIX     *-5,1,1         ..                                              TT)40480
00753  0074 00 4 02407              TSX     WOT,4           COMMENT, ITS REALLY THE END.                    TT)40490
00754  0 00011 0 01102              PZE     RENDTP,0,9      ..                                              TT)40500
00755  0074 00 4 02407              TSX     WOT,4           SKIP TO CHANNEL FOUR.                           TT)40510
00756  0 00001 0 03147              PZE     =H4     ,0,1    ..                                              TT)40520
00757  0774 00 4 00012              AXT     10,4            TEN EOF'S ON A3.                                TT)40530
00760  0770 00 0 01203              WEFA    3               ..                                              TT)40540
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                     SECTION 4, WIND UP THIS RUN.

00761  2 00001 4 00760          TIX     *-1,4,1         ..                                  TT)40550
00762 -0772 00 0 01203          RUNA    3               THEN RUN OUTPUT TAPE.               TT)40560
00763  0074 00 4 02520          TSX     PRINT,4         COMMENT.                            TT)40570
00764 -0 00016 0 01061          MZE     XCLOSE,0,14     ..                                  TT)40580
                                                                                            TT)40590
00765  060C 00 0 03125   DOOR   STZ     =0              MAKE A ZERO LOCATION.               TT)40600
00766  0564 00 0 03125          ENB     =0              RESET THE TRAPS.                    TT)40610
00767 -0760 00 0 00007          LTM                     JUST IN CASE.                       TT)40620
00770  0074 00 4 00663          TSX     LSTM,4          ..                                  TT)40630
00771 -0760 00 0 00010          LSNM                    ..                                  TT)40640
00772  0760 00 0 00004          ENK                     GET CONSOLE KEYS.                   TT)40650
00773 -0130 00 0 00000          XCL                     PLACE IN AC.                        TT)40660
00774  0044 00 0 00000          PAI                     PLACE IN INDICATORS.                TT)40670
00775 -0056 00  C00001          LNT     1               CHECK IF OPERATOR WANTS TO STOP.    TT)40680
00776  0020 00 0 01002          TRA     DOOR1           NO.                                 TT)40690
00777  0074 00 4 02520          TSX     PRINT,4         YES, COMMENT.                       TT)40700
C1000 -0 00013 0 01113          MZE     SEXIT,0,11      ..                                  TT)40710
01001  0020 00 0 00765          TRA     DOOR            GO BACK TO CHECK KEYS.              TT)40720
                                                                                            TT)40730
01002  0022 00 0 01003   DOOR1  TRCA    *+1             TURN OFF RC LIGHT.                  TT)40740
01003  0774 00 1 00005          AXT     5,1             SET UP FOR 5 TRIES.                 TT)40750
01004  0772 00 0 01201          REWA    1               REWIND THE SYSTEMS TAPE.            TT)40760
01005  0762 00 0 01221          RTBA    1               SELECT.                             TT)40770
01006  0540 00 0 01024          RCHA    LOAD            LOAD SEQUENCE OF CHANNEL COMMANDS.  TT)40780
01007 -0054 00  000002          LFT     2               MISTART OR EXITM.                   TT)40790
01010  0020 00 0 01013          TRA     *+3             MISTART.                            TT)40800
01011  0762 00 0 01221          RTBA    1               EXITM, SKIP TWO RECORDS.            TT)40810
01012  0762 00 0 01221          RTBA    1               ..                                  TT)40820
01013  0060 00 0 01013          TCOA    *               WAIT FOR DSCA.                      TT)40830
01014  0022 00 0 01020          TRCA    *+4             EXIT FOR READ ERROR.                TT)40840
01015 -0500 00 0 01015          CAL     *               SET PREFIX OF 34 TO MZE.            TT)40850
01016  0630 00 0 00042          STP     34              ..                                  TT)40860
01017  0020 00 0 00001          TRA     1               THEN RETURN TO MONITOR.             TT)40870
01020  2 00001 1 01004          TIX     DOOR1+2,1,1     READING ERROR, INDEX AND TRY AGAIN. TT)40880
                                                                                            TT)40890
01021  0074 00 4 02520          TSX     PRINT,4         FIVE REDUNDANCY FAILURES, COMMENT.  TT)40900
C1022 -0 00017 0 01126          MZE     XRCA1,0,15      ..                                  TT)40910
01023  0020 00 0 00765          TRA     DOOR            GO TRY FIVE MORE TIMES.             TT)40920
                                                                                            TT)40930
01024 -0 00003 0 00000   LOAD   IOCP    0,0,3           CHANNEL COMMANDS.                   TT)40940
01025  1 00000 0 00000          TCH     0               ..                                  TT)40950
01026  0 00000 0 00000   EPMR   PZE                     TEST CELL FOR ERROR POST MORTEM.    TT)40960
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                      SECTION 4, WIND UP THIS RUN.

                                EJECT                                                   TT)40970
                                TITLE           SUPPRESS GENERATED OCTAL LISTING.       TT)40980
                                                                                        TT)40990
                                                COMMENTS.                               TT)41000
                                                                                        TT)41010
01027  002545246046       ENDRN BCI    2,0END OF RUN.                                  TT)41020
                                                                                        TT)41030
01031  004225706003       XSTOP BCI    9,0KEY 34 DOWN, STOP COMMAND WHILE CHECKING KEYS, PRESS  TT)41040
                                                                                        TT)41050
01042  626321516360              BCI    3,START TO CONTINUE.                            TT)41060
                                                                                        TT)41070
01045  004225706003       XFILE BCI    9,0KEY 33 DOWN, AN EOF HAS BEEN WRITTEN ON A3, PRESS STA  TT)41080
                                                                                        TT)41090
01056  516360634660              BCI    3,RT TO CONTINUE.                               TT)41100
                                                                                        TT)41110
01061  004225706003       XCLOSE BCI   9,0KEY 35 DOWN, TAPE A3 HAS BEEN CLOSED, CHANGE TAPE AND  TT)41120
                                                                                        TT)41130
01072  604751256262              BCI    5, PRESS START TO CONTINUE.                     TT)41140
                                                                                        TT)41150
01077  012545246046       ENDTP BCI    3,1END OF TAPE.                                 TT)41160
                                                                                        TT)41170
01102  006330255125       RENDTP BCI   9,0THERE REALLY ISN'T ANY MORE ON THIS TAPE - HONEST.  TT)41180
                                                                                        TT)41190
01113  004225706001       SEXIT BCI    9,0KEY 17 DOWN, STOP COMMAND BEFORE EXIT, PRESS START TO  TT)41200
                                                                                        TT)41210
01124  602346456331              BCI    2, CONTINUE.                                    TT)41220
                                                                                        TT)41230
01126  002631652560       XRCA1 BCI    9,0FIVE CONSECUTIVE REDUNDANCY FAILURES IN READING A1. P  TT)41240
                                                                                        TT)41250
01137  512562626062              BCI    6,RESS START FOR FIVE MORE TRIES.               TT)41260
                                                                                        TT)41270
                                DETAIL          RETURN TO NORMAL LISTING MODE.          TT)41280
```

```
                                        TTL     SECTION 5, SVCON, PMR, OCT, OCTADR, AND TABBLK.          TT)50000
                                                                                                         TT)50010
01145  0634 00 4 01166    SX4     SXA     IR4,4           ENTRY TO SAVE IR4 FOR PMR USE.                TT)50020


                                        SPACE   2                                                        TT)50030
                                                CONSOLE LIGHTS SAVING ROUTINE.                           TT)50040
                                                                                                         TT)50050
01146  0600 00 0 00005    SVCON   STZ     5               RESET INTERVAL TIMER.                         TT)50060
01147 -0061 00 0 01151            TCNT    *+2             IF TAPE TIMER IN USE,                         TT)50070
01150 -0540 00 0 00261            RCHT    QUIT            RESET TAPE TIMER.                             TT)50080
01151  0634 00 4 01153            SXA     *+2,4           SAVE IR4.                                     TT)50090
01152  0074 00 4 00663            TSX     LSTM,4          RESET TRAP.                                   TT)50100
01153  0774 00 4 00000            AXT     **,4            RESTORE IR4.                                  TT)50110
01154  0634 00 1 01164            SXA     IR1,1           SAVE IR1.                                     TT)50120
01155  0634 00 2 01165            SXA     IR2,2           SAVE IR2.                                     TT)50130
01156 -0600 00 0 01171            STQ     MQ              SAVE THE MQ.                                  TT)50140
01157  0602 00 0 01170            SLW     AC              SAVE THE AC.                                  TT)50150
01160  0765 00 0 00045            LRS     37              GET S,Q,P BITS.                               TT)50160
01161 -0600 00 0 01172            STQ     SVPQ            SAVE.                                         TT)50170
01162  0604 00 0 01167            STI     SIND            SAVE INDICATORS.                              TT)50180
01163  0020 00 4 00001            TRA     1,4             RETURN.                                       TT)50190
                                                                                                         TT)50200
01164  0 00000 0 00000    IR1     PZE                     CONSOLE STORAGE.                              TT)50210
01165  0 00000 0 00000    IR2     PZE                     ..                                            TT)50220
01166  0 00000 0 00000    IR4     PZE                     ..                                            TT)50230
01167  0 00000 0 00000    SIND    PZE                     ..                                            TT)50240
01170  0 00000 0 00000    AC      PZE                     ..                                            TT)50250
01171  0 00000 0 00000    MQ      PZE                     ..                                            TT)50260
01172  0 00000 0 00000    SVPQ    PZE                     ..                                            TT)50270


                                        SPACE   2                                                        TT)50280
                                                ADDRESS TO OCTAL CONVERTER, DELETE LEADING ZEROES.       TT)50290
                                                                                                         TT)50300
01173 -0763 00 0 00025    OCTADR  LGL     21              WORD IN MQ.                                   TT)50310
01174 -0754 00 0 00000            PXD     0,0             CLEAR AC.                                     TT)50320
01175  0634 00 4 01205            SXA     *+8,4           SAVE IR4.                                     TT)50330
01176  0774 00 4 00005            AXT     5,4             SET UP LOOP.                                  TT)50340
01177  0767 00 0 00003            ALS     3               INSERT 000 BETWEEN GROUPS OF THREE BITS.      TT)50350
01200 -0763 00 0 00003            LGL     3               ..                                            TT)50360
01201  2 00001 4 01177            TIX     *-2,4,1         ..                                            TT)50370
01202 -0130 00 0 00000            XCL                     ANSWER IN MQ.                                 TT)50380
01203 -0154 05 0 01207            CRQ     TABBLK,0,5      DELETE LEADING ZEROES.                        TT)50390
01204 -0773 00 0 00006            RQL     6               LAST ZERO.                                    TT)50400
01205  0774 00 4 00000            AXT     **,4            RESTORE IR4.                                  TT)50410
01206  0020 00 4 00001            TRA     1,4             RETURN.                                       TT)50420


                                        SPACE   2                                                        TT)50430
                                                TABLE FOR DELETING LEADING ZEROES.                       TT)50440
                                                                                                         TT)50450
01207  600000001207       TABBLK  VFD     06/60,15/0,15/TABBLK                                          TT)50460
01210  010000001221               VFD     H6/1,15/0,15/TABBLK+10                                        TT)50470
01211  020000001221               VFD     H6/2,15/0,15/TABBLK+10                                        TT)50480
```

```
01212  030000001221              VFD     H6/3,15/0,15/TABBLK+10                    TT)50490
01213  040000001221              VFD     H6/4,15/0,15/TABBLK+10                    TT)50500
01214  050000001221              VFD     H6/5,15/0,15/TABBLK+10                    TT)50510
01215  060000001221              VFD     H6/6,15/0,15/TABBLK+10                    TT)50520
01216  070000001221              VFD     H6/7,15/0,15/TABBLK+10                    TT)50530
01217  100000001221              VFD     H6/8,15/0,15/TABBLK+10                    TT)50540
01220  110000001221              VFD     H6/9,15/0,15/TABBLK+10                    TT)50550
01221  000000001221              VFD     H6/0,15/0,15/TABBLK+10                    TT)50560
01222  010000001221              VFD     H6/1,15/0,15/TABBLK+10                    TT)50570
01223  020000001221              VFD     H6/2,15/0,15/TABBLK+10                    TT)50580
01224  030000001221              VFD     H6/3,15/0,15/TABBLK+10                    TT)50590
01225  040000001221              VFD     H6/4,15/0,15/TABBLK+10                    TT)50600
01226  050000001221              VFD     H6/5,15/0,15/TABBLK+10                    TT)50610
01227  060000001221              VFD     H6/6,15/0,15/TABBLK+10                    TT)50620
01230  070000001221              VFD     H6/7,15/0,15/TABBLK+10                    TT)50630
01231  100000001221              VFD     H6/8,15/0,15/TABBLK+10                    TT)50640
01232  110000001221              VFD     H6/9,15/0,15/TABBLK+10                    TT)50650


                                 SPACE   2                                         TT)50660
                                         COMPLEMENT ADDRESS TO OCTAL-BCI CONVERTER. TT)50670
                                                                                   TT)50680
01233   0634 00 4 01256  COMADR SXA     COMRT,4         SAVE IR4.                 TT)50690
01234   0774 00 4 01251          AXT     COMSW+1,4       RESET SWITCH.             TT)50700
01235   0634 00 4 01250          SXA     COMSW,4         ..                        TT)50710
01236  -0130 00 0 00000          XCL                     ..                        TT)50720
01237  -0100 00 0 01242          TNZ     *+3             IF ZERO, SET.             TT)50730
01240   0560 00 0 03157          LDQ     =H     -0       ..                        TT)50740
01241   0020 00 0 01256          TRA     COMRT           AND EXIT.                 TT)50750
01242   0737 00 4 00000          PAC     0,4             ..                        TT)50760
01243   0754 00 4 00000          PXA     0,4             ..                        TT)50770
01244  -0765 00 0 00017          LGR     15              ..                        TT)50780
01245   0774 00 4 00005          AXT     5,4             ..                        TT)50790
01246   0767 00 0 00003          ALS     3               ..                        TT)50800
01247  -0763 00 0 00003          LGL     3               ..                        TT)50810
01250   0020 00 0 01251  COMSW   TRA     *+1             SWITCH AFTER FIRST NON-ZERO DIGIT. TT)50820
01251   0100 00 0 01254          TZE     *+3             ..                        TT)50830
01252  -0501 00 0 03157          ORA     =H     -0       ..                        TT)50840
01253  -0625 00 0 01250          STL     COMSW           SET SWITCH.               TT)50850
01254   2 00001 4 01246          TIX     *-6,4,1         ..                        TT)50860
01255  -0130 00 0 00000          XCL                     ..                        TT)50870
01256   0774 00 4 00000  COMRT   AXT     **,4            RESTORE IR4.              TT)50880
01257   0020 00 4 00001          TRA     1,4             RETURN.                   TT)50890


                                 SPACE   2                                         TT)50900
                                         FULL WORD BINARY TO OCTAL-BCI CONVERTER.  TT)50910
                                                                                   TT)50920
01260   0634 00 4 01276  OCT     SXA     *+14,4          WORD IN MQ.               TT)50930
01261   0774 00 4 00006          AXT     6,4             CONVERT LEFT HALF.        TT)50940
01262  -0754 00 0 00000          PXD     0,0             ..                        TT)50950
01263   0767 00 0 00003          ALS     3               ..                        TT)50960
01264  -0763 00 0 00003          LGL     3               ..                        TT)50970
01265   2 00001 4 01263          TIX     *-2,4,1         ..                        TT)50980
01266   0602 00 0 01300          SLW     *+10            ..                        TT)50990
```

```
01267  0774 00 4 00006          AXT     6,4             CONVERT RIGHT HALF.             TT)51000
01270 -0754 00 0 00000          PXD     0,0             ..                              TT)51010
01271  0767 00 0 00003          ALS     3               ..                              TT)51020
01272 -0763 00 0 00003          LGL     3               ..                              TT)51030
01273  2 00001 4 01271          TIX     *-2,4,1         ..                              TT)51040
01274 -0130 00 0 00000          XCL                     LOAD COMPLETED WORD.            TT)51050
01275 -0500 00 0 01300          CAL     *+3             ..                              TT)51060
01276  0774 00 4 00000          AXT     **,4            RESTORE IR4.                    TT)51070
01277  0020 00 4 00001          TRA     1,4             RETURN TO CALLER.               TT)51080
                                                                                         TT)51090
01300  0 00000 0 00000          PZE                     TEMPORARY STORAGE.              TT)51100
                 01260   OCT)   SYN     OCT             EXTERNAL NAME FOR OCT.          TT)51110


                                SPACE   2                                               TT)51120
                                TABLE FOR DELETING ILLEGAL BCI CHARACTERS.              TT)51130
                                                                                        TT)51140
01301  000000001301     BCDTAB VFD     H6/0,15/0,15/BCDTAB     00                      TT)51150
01302  010000001301            VFD     H6/1,15/0,15/BCDTAB     01                      TT)51160
01303  020000001301            VFD     H6/2,15/0,15/BCDTAB     02                      TT)51170
01304  030000001301            VFD     H6/3,15/0,15/BCDTAB     03                      TT)51180
01305  040000001301            VFD     H6/4,15/0,15/BCDTAB     04                      TT)51190
01306  050000001301            VFD     H6/5,15/0,15/BCDTAB     05                      TT)51200
01307  060000001301            VFD     H6/6,15/0,15/BCDTAB     06                      TT)51210
01310  070000001301            VFD     H6/7,15/0,15/BCDTAB     07                      TT)51220
01311  100000001301            VFD     H6/8,15/0,15/BCDTAB     10                      TT)51230
01312  110000001301            VFD     H6/9,15/0,15/BCDTAB     11                      TT)51240
01313  540000001301            VFD     H6/*,15/0,15/BCDTAB     12                      TT)51250
01314  130000001301            VFD     H6/=,15/0,15/BCDTAB     13                      TT)51260
01315  140000001301            VFD     H6/',15/0,15/BCDTAB     14                      TT)51270
01316  540000001301            VFD     H6/*,15/0,15/BCDTAB     15                      TT)51280
01317  540000001301            VFD     H6/*,15/0,15/BCDTAB     16                      TT)51290
01320  540000001301            VFD     H6/*,15/0,15/BCDTAB     17                      TT)51300
01321  200000001301            VFD     H6/+,15/0,15/BCDTAB     20                      TT)51310
01322  210000001301            VFD     H6/A,15/0,15/BCDTAB     21                      TT)51320
01323  220000001301            VFD     H6/B,15/0,15/BCDTAB     22                      TT)51330
01324  230000001301            VFD     H6/C,15/0,15/BCDTAB     23                      TT)51340
01325  240000001301            VFD     H6/D,15/0,15/BCDTAB     24                      TT)51350
01326  250000001301            VFD     H6/E,15/0,15/BCDTAB     25                      TT)51360
01327  260000001301            VFD     H6/F,15/0,15/BCDTAB     26                      TT)51370
01330  270000001301            VFD     H6/G,15/0,15/BCDTAB     27                      TT)51380
01331  300000001301            VFD     H6/H,15/0,15/BCDTAB     30                      TT)51390
01332  310000001301            VFD     H6/I,15/0,15/BCDTAB     31                      TT)51400
01333  540000001301            VFD     H6/*,15/0,15/BCDTAB     32                      TT)51410
01334  330000001301            VFD     H6/.,15/0,15/BCDTAB     33                      TT)51420
01335  340000001301            VFD     H6/),15/0,15/BCDTAB     34                      TT)51430
01336  540000001301            VFD     H6/*,15/0,15/BCDTAB     35                      TT)51440
01337  540000001301            VFD     H6/*,15/0,15/BCDTAB     36                      TT)51450
01340  540000001301            VFD     H6/*,15/0,15/BCDTAB     37                      TT)51460
01341  400000001301            VFD     H6/-,15/0,15/BCDTAB     40                      TT)51470
01342  410000001301            VFD     H6/J,15/0,15/BCDTAB     41                      TT)51480
01343  420000001301            VFD     H6/K,15/0,15/BCDTAB     42                      TT)51490
01344  430000001301            VFD     H6/L,15/0,15/BCDTAB     43                      TT)51500
01345  440000001301            VFD     H6/M,15/0,15/BCDTAB     44                      TT)51510
01346  450000001301            VFD     H6/N,15/0,15/BCDTAB     45                      TT)51520
```

```
01347  460000001301          VFD     H6/O,15/0,15/BCDTAB     46                              TT)51530
01350  470000001301          VFD     H6/P,15/0,15/BCDTAB     47                              TT)51540
01351  500000001301          VFD     H6/Q,15/0,15/BCDTAB     50                              TT)51550
01352  510000001301          VFD     H6/R,15/0,15/BCDTAB     51                              TT)51560
01353  540000001301          VFD     H6/*,15/0,15/BCDTAB     52                              TT)51570
01354  530000001301          VFD     H6/$,15/0,15/BCDTAB     53                              TT)51580
01355  540000001301          VFD     H6/*,15/0,15/BCDTAB     54                              TT)51590
01356  540000001301          VFD     H6/*,15/0,15/BCDTAB     55                              TT)51600
01357  540000001301          VFD     H6/*,15/0,15/BCDTAB     56                              TT)51610
01360  540000001301          VFD     H6/*,15/0,15/BCDTAB     57                              TT)51620
01361  600000001301          VFD     06/60,15/0,15/BCDTAB    60  NO BLANKS ALLOWED IN VFD.   TT)51630
01362  610000001301          VFD     H6//,15/0,15/BCDTAB     61                              TT)51640
01363  620000001301          VFD     H6/S,15/0,15/BCDTAB     62                              TT)51650
01364  630000001301          VFD     H6/T,15/0,15/BCDTAB     63                              TT)51660
01365  640000001301          VFD     H6/U,15/0,15/BCDTAB     64                              TT)51670
01366  650000001301          VFD     H6/V,15/0,15/BCDTAB     65                              TT)51680
01367  660000001301          VFD     H6/W,15/0,15/BCDTAB     66                              TT)51690
01370  670000001301          VFD     H6/X,15/0,15/BCDTAB     67                              TT)51700
01371  700000001301          VFD     H6/Y,15/0,15/BCDTAB     70                              TT)51710
01372  710000001301          VFD     H6/Z,15/0,15/BCDTAB     71                              TT)51720
01373  540000001301          VFD     H6/*,15/0,15/BCDTAB     72                              TT)51730
01374  730000001301          VFD     06/73,15/0,15/BCDTAB    73  NO COMMAS ALLOWED IN VFD.   TT)51740
01375  740000001301          VFD     H6/(,15/0,15/BCDTAB     74                              TT)51750
01376  540000001301          VFD     H6/*,15/0,15/BCDTAB     75                              TT)51760
01377  540000001301          VFD     H6/*,15/0,15/BCDTAB     76                              TT)51770
01400  540000001301          VFD     H6/*,15/0,15/BCDTAB     77                              TT)51780

                             SPACE   2                                                       TT)51790
                                     POST MORTEM SECTION.                                    TT)51800
01401   0074 00 4 02407  PMR  TSX     WOT,4           COMMENT ERROR IN OBJECT PROGRAM.        TT)51810
01402   0 00007 0 01474       PZE     PROER,0,7       ..                                      TT)51820
01403   0760 00 0 00141       SLN     1               SENSE LIGHTS ON MEANS POST MORTEM.      TT)51830
01404   0760 00 0 00142       SLN     2               ..                                      TT)51840
01405   0760 00 0 00143       SLN     3               ..                                      TT)51850
01406   0760 00 0 00144       SLN     4               ..                                      TT)51860
01407   0074 00 4 00003       TSX     $PPROG,4        DUMP SYMBOLIC PROGRAM.                  TT)51870
01410   0074 00 4 00004       TSX     $PCT1,4         DUMP COLLATION TAPE.                    TT)51880
01411   0074 00 4 01523       TSX     DCON,4          GO TO PRINT CONSOLE.                    TT)51890
                                                                                              TT)51900
01412   0774 00 2 00036  SLPM AXT     NSYM,2          SET UP SYMBOL TABLE SEARCH.             TT)51910
01413  -0500 00 2 03067       CAL     ORIGIN+NSYM,2   GET NEXT ORIGIN.                        TT)51920
01414   0621 00 0 01445       STA     COD1            PLACE IN CALLING SEQUENCE.              TT)51930
01415   0402 00 2 03070       SUB     ORIGIN+NSYM+1,2 COUNT IN ADDRESS.                       TT)51940
01416   0767 00 0 00022       ALS     18              PLACE IN DECREMENT.                     TT)51950
01417   0622 00 0 01445       STD     COD1            PLACE IN CALLING SEQUENCE.              TT)51960
01420   0560 00 2 03031       LDQ     NAME+NSYM,2     GET THE NAME OF THIS PROGRAM.           TT)51970
01421  -0754 00 0 00000       PXD     0,C             SHIFT AND OR TO COMMENT.                TT)51980
01422  -0763 00 0 00022       LGL     18              ..                                      TT)51990
01423  -0501 00 0 03154       ORA     =HOF 000        ..                                      TT)52000
01424   0602 00 0 01505       SLW     PMH+2           ..                                      TT)52010
01425  -0130 00 0 00000       XCL                     ..                                      TT)52020
01426  -0501 00 0 03137       ORA     =H000 F0        ..                                      TT)52030
01427   0602 00 0 01506       SLW     PMH+3           ..                                      TT)52040
                                                                                              TT)52050
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                      SECTION 5, SVCON, PMR, OCT, OCTADR, AND TABBLK.

01430  0560 00 0 03142          LDQ     =HOOCTAL        ASSUME DOUBLE SPACE FOR HEADING.          TT)52060
01431  0500 00 0 02022          CLA     PMCNT           GET PM LINE COUNT.                        TT)52070
01432  0402 00 0 03131          SUB     =5              AT LEAST FIVE LINES LEFT.                 TT)52080
01433  0120 00 0 01437          TPL     *+4             IF PLUS, OK, SKIP.                        TT)52090
01434  0560 00 0 03145          LDQ     =H1OCTAL        SET TO EJECT PAGE.                        TT)52100
01435  0500 00 0 03135          CLA     =59             RESET LINE COUNT.                         TT)52110
01436  0020 00 0 01440          TRA     *+2             SKIP ADDITION.                            TT)52120
01437  0400 00 0 03130          ADD     =3              NORMALLY TWO LINES FOR HEADING.           TT)52130
01440  0601 00 0 02022          STO     PMCNT           SAVE PM LINE COUNT.                       TT)52140
01441 -0600 00 0 01503          STQ     PMH             INSERT CARRAIGE CONTROL.                  TT)52150
01442  0074 00 4 02407          TSX     WOT,4           PRINT HEADING.                            TT)52160
01443  0 00005 0 01503          PZE     PMH,0,5         ..                                        TT)52170
01444  0074 00 4 01640          TSX     OCTDMP,4        EXIT TO GIVE POST MORTEM.                 TT)52180
01445  0 00000 0 00000   COD1   PZE     **,0,**         ..                                        TT)52190
01446  1 77777 2 01447          TXI     *+1,2,-1        INDEX.                                    TT)52200
01447  3 00000 2 01413   LSUB   TXH     SLPM+1,2,**     COUNT INSERTED BY MAP, NEXT POST MORTEM.  TT)52210
                                                                                                 TT)52220
01450  0560 00 0 03142          LDQ     =HOOCTAL        ASSUME DOUBLE SPACE FOR HEADING.          TT)52230
01451  0500 00 0 02022          CLA     PMCNT           GET PM LINE COUNT.                        TT)52240
01452  0402 00 0 03131          SUB     =5              AT LEAST FIVE LINES LEFT.                 TT)52250
01453  0120 00 0 01457          TPL     *+4             IF PLUS, OK, SKIP.                        TT)52260
01454  0560 00 0 03145          LDQ     =H1OCTAL        SET TO EJECT PAGE.                        TT)52270
01455  0500 00 0 03135          CLA     =59             RESET LINE COUNT.                         TT)52280
01456  0020 00 0 01460          TRA     *+2             SKIP ADDITION.                            TT)52290
01457  0400 00 0 03130          ADD     =3              NORMALLY TWO LINES FOR HEADING.           TT)52300
01460  0601 00 0 02022          STO     PMCNT           SAVE PM LINE COUNT.                       TT)52310
01461 -0600 00 0 01503          STQ     PMH             INSERT CARRIAGE CONTROL.                  TT)52320
01462  0074 00 4 02407          TSX     WOT,4           PRINT HEADING.                            TT)52330
01463  0 00007 0 01510          PZE     PMBP,0,7        ..                                        TT)52340
01464 -0500 00 0 00234          CAL     ORG             SET ORIGIN OF ASSEMBLED PROGRAM.          TT)52350
01465  0621 00 0 01467          STA     *+2             ..                                        TT)52360
01466  0074 00 4 01640          TSX     OCTDMP,4        ..                                        TT)52370
01467  0 00400 0 00000          PZE     **,0,256        ..                                        TT)52380
01470  0074 00 4 02407          TSX     WOT,4           END OF POST MORTEM, COMMENT.              TT)52390
01471  0 00004 0 01517          PZE     ENDPM,0,4       ..                                        TT)52400
01472  0760 00 0 00140          SLF                     TURN OFF SL AFTER POST MORTEM.            TT)52410
01473  0020 00 0 00714          TRA     FINIS           RETURN.                                   TT)52420
                                                                                                 TT)52430
01474  004622412523      PROER BCI     7,00BJECT PROGRAM ERROR, PROGRAM TERMINATED.              TT)52440
01475  636047514627
01476  512144602551
01477  514651736047
01500  514627512144
01501  606325514431
01502  452163252433
01503  004623632143      PMH   BCI     5,COCTAL DUMP OF 000000 FOLLOWS.                          TT)52450
01504  602464444760
01505  462660000000
01506  000000602646
01507  434346666233
01510  004623632143      PMBP  BCI     7,00CTAL DUMP OF ASSEMBLED PROGRAM FOLLOWS.               TT)52460
01511  602464444760
01512  462660216262
01513  254422432524
01514  604751462751
```

```
01515  214460264643
01516  434666623360
01517  002545246046        ENDPM BCI     4,0END OF POST MORTEM.                    TT)52470
01520  266047466263
01521  604446516325
01522  443360606060


                                   SPACE   2                                       TT)52480
                                           USE RESULTS OF SVCON TO DUMP CONSOLE LIGHTS.  TT)52490
01523   0634 00 4 01602    DCON    SXA     DCNX4,4         SAVE IR4.               TT)52500
01524   0560 00 0 01172            LDQ     SVPQ            S,Q,P.                  TT)52510
01525  -0754 00 0 00000            PXD     0,0             CLEAR AC.               TT)52520
01526  -0763 00 0 00001            LGL     1               ..                      TT)52530
01527   0767 00 0 00013            ALS     11              ..                      TT)52540
01530  -0763 00 0 00001            LGL     1               ..                      TT)52550
01531   0767 00 0 00013            ALS     11              ..                      TT)52560
01532  -0763 00 0 00001            LGL     1               ..                      TT)52570
01533  -0501 00 0 03155            ORA     =H 0,0,0        PUT IN COMMAS.          TT)52580
01534   0602 00 0 01612            SLW     HEAD02+2        INSERT.                 TT)52590
01535   0560 00 0 01170            LDQ     AC              CONVERT AC TO OCTAL.    TT)52600
01536   0074 00 4 01260            TSX     OCT,4           ..                      TT)52610
01537   0602 00 0 01615            SLW     HEAD03+2        INSERT.                 TT)52620
01540  -0600 00 0 01616            STQ     HEAD03+3        ..                      TT)52630
01541   0560 00 0 01171            LDQ     MQ              CONVERT MQ TO OCTAL.    TT)52640
01542   0074 00 4 01260            TSX     OCT,4           ..                      TT)52650
01543   0602 00 0 01621            SLW     HEAD04+2        INSERT.                 TT)52660
01544  -0600 00 0 01622            STQ     HEAD04+3        ..                      TT)52670
01545   0560 00 0 01167            LDQ     SIND            CONVERT SI TO OCTAL.    TT)52680
01546   0074 00 4 01260            TSX     OCT,4           ..                      TT)52690
01547   0602 00 0 01625            SLW     HEAD05+2        INSERT.                 TT)52700
01550  -0600 00 0 01626            STQ     HEAD05+3        ..                      TT)52710
01551   0560 00 0 01164            LDQ     IR1             CONVERT IR1.            TT)52720
01552   0074 00 4 01173            TSX     OCTADR,4        ..                      TT)52730
01553  -0600 00 0 01631            STQ     HEAD06+2        INSERT.                 TT)52740
01554   0560 00 0 01165            LDQ     IR2             CONVERT IR2.            TT)52750
01555   0074 00 4 01173            TSX     OCTADR,4        ..                      TT)52760
01556  -0600 00 0 01634            STQ     HEAD07+2        ..                      TT)52770
01557   0560 00 0 01166            LDQ     IR4             CONVERT IR4.            TT)52780
01560   0074 00 4 01173            TSX     OCTADR,4        ..                      TT)52790
01561  -0600 00 0 01637            STQ     HEAD08+2        INSERT.                 TT)52800
01562   0074 00 4 02407            TSX     WOT,4           PRINT CONSOLE.          TT)52810
01563   0 00004 0 01604            PZE     HEAD01,0,4      ..                      TT)52820
01564   0074 00 4 02407            TSX     WOT,4           ..                      TT)52830
01565   0 00003 0 01610            PZE     HEAD02,0,3      ..                      TT)52840
01566   0074 00 4 02407            TSX     WOT,4           ..                      TT)52850
01567   0 00004 0 01613            PZE     HEAD03,0,4      ..                      TT)52860
01570   0074 00 4 02407            TSX     WOT,4           ..                      TT)52870
01571   0 00004 0 01617            PZE     HEAD04,0,4      ..                      TT)52880
01572   0074 00 4 02407            TSX     WOT,4           ..                      TT)52890
01573   0 00004 0 01623            PZE     HEAD05,0,4      ..                      TT)52900
01574   0074 00 4 02407            TSX     WOT,4           ..                      TT)52910
01575   0 00003 0 01627            PZE     HEAD06,0,3      ..                      TT)52920
01576   0074 00 4 02407            TSX     WOT,4           ..                      TT)52930
                                                                                   TT)52940
```

```
                TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
                       SECTION 5, SVCON, PMR, OCT, OCTADR, AND TABBLK.

01577  0 00003 0 01632          PZE    HEAD07,0,3     ..                          TT)52950
01600  0074 00 4 02407          TSX    WOT,4          ..                          TT)52960
01601  0 00003 0 01635          PZE    HEAD08,0,3     ..                          TT)52970
01602  0774 00 4 00000   DCNX4  AXT    **,4           RESTORE IR4.                TT)52980
01603  0020 00 4 00001          TRA    1,4            RETURN.                     TT)52990
                                                                                  TT)53000
C1604  014746626360      HEAD01 BCI    4,1POST MORTEM OF CONSOLE.                  TT)53010
01605  444651632544
01606  604626602346
01607  456246432533
01610  006060606060     HEAD02 BCI    3,0      S,Q,P= 0,0,0                       TT)53020
01611  627350734713
01612  600073007300
01613  006060606060     HEAD03 BCI    4,C      AC =  000000000000                 TT)53030
01614  212360136060
01615  000000000000
01616  000000000000
01617  006060606060     HEAD04 BCI    4,0      MQ =  000000000000                 TT)53040
01620  445060136060
01621  000000000000
01622  000000000000
01623  006060606060     HEAD05 BCI    4,0      SI =  000000000000                 TT)53050
01624  623160136060
01625  000000000000
01626  000000000000
01627  006060606060     HEAD06 BCI    3,0      IR1=  000000                       TT)53060
01630  315101136060
01631  000000000000
01632  006060606060     HEAD07 BCI    3,0      IR2=  000000                       TT)53070
01633  315102136060
01634  000000000000
01635  006060606060     HEAD08 BCI    3,0      IR4=  000000                       TT)53080
01636  315104136060
01637  000000000000

                                SPACE  2                                          TT)53090
                                       OCTDMP, OCTAL DUMP WITH MNEMONICS.         TT)53100
                                                                                  TT)53110
                                                                                  TT)53120
01640   0634 00 4 02016  OCTDMP SXA    DX4,4          SAVE IRS.                   TT)53130
01641   0634 00 2 02017          SXA    DX2,2          ..                          TT)53140
01642   0634 00 1 02020          SXA    DX1,1          ..                          TT)53150
01643  -0500 00 4 00001          CAL    1,4            GET CALLING SEQUENCE.       TT)53160
01644  -0734 00 1 00000          PDX    0,1            COUNT TO IR1.               TT)53170
01645   0621 00 0 02023          STA    ILC            FIRST LOCATION.             TT)53180
01646   0771 00 0 00022          ARS    18             FIRST + N = LAST.           TT)53190
01647   0361 00 4 00001          ACL    1,4            COMPUTE END ADDRESS.        TT)53200
01650   0621 00 0 01660          STA    LOOP1          SET UP LOOP.                TT)53210
01651   0621 00 0 01664          STA    LOOP1+4        ..                          TT)53220
01652   0621 00 0 01742          STA    REG+2          ..                          TT)53230
01653   0560 00 0 02023    ADR   LDQ    ILC            GET ILC AND CONVERT.        TT)53240
01654   0074 00 4 01173          TSX    OCTADR,4       ..                          TT)53250
01655  -0773 00 0 00006          RQL    6              PUT BLANK AT END.           TT)53260
01656  -0600 00 0 02746          STQ    OUT+1          INSERT.                     TT)53270
01657   0774 00 2 00022  LOOP2 AXT    18,2           SET WORD COUNT.
```

```
01660 -0500 00 1 00000   LOOP1 CAL     **,1          GET WORD.                                          TT)53280
01661 -3 00017 2 01740          TXL     REG,2,15      IF FIRST WORD, CHECK FOR REPEATS.                  TT)53290
01662  0602 00 0 02024          SLW     LWORD         SAVE WORD.                                         TT)53300
01663  0774 00 4 00000          AXT     0,4           SET IR4.                                           TT)53310
01664 -0340 00 1 00000          LAS     **,1          CHECK FOR REPEATS.                                 TT)53320
01665  0020 00 0 01672          TRA     *+5           NOT SAME, EXIT.                                    TT)53330
01666  1 00001 4 01670          TXI     *+2,4,1       SAME, INDEX.                                       TT)53340
01667  0020 00 0 01672          TRA     *+3           NOT SAME, EXIT.                                    TT)53350
01670  2 00001 1 01664          TIX     *-4,1,1       INDEX AND TRY NEXT WORD.                           TT)53360
01671  1 77777 4 01672          TXI     *+1,4,-1      LAST WORD, INDEX FOR LINE AT BOTTOM.               TT)53370
01672  3 00022 4 01675          TXH     *+3,4,18      IF MORE THAN 18 REPEATS, COMMENT.                  TT)53380
01673 -0634 00 4 01674          SXD     *+1,4         LESS THAN SIX, RETURN TO NORMAL.                   TT)53390
01674  1 00000 1 01740          TXI     REG,1,**      ..                                                 TT)53400
01675  0634 00 4 02025          SXA     NREP,4        MORE THAN SIX, CONVERT NUMBER.                     TT)53410
01676  0560 00 0 02025          LDQ     NREP          PLACE IN MQ.                                       TT)53420
01677  0600 00 0 02026          STZ     DNREP         CLEAR DECIMAL NREP.                                TT)53430
01700  0774 00 4 00036          AXT     30,4          SET CONVERSION LOOP.                               TT)53440
01701 -0754 00 0 00000          PXD     0,0           CLEAR AC.                                          TT)53450
01702  0221 00 0 03133          DVP     =10           MOD TEN.                                           TT)53460
01703  0767 00 4 00036          ALS     30,4          SHIFT TO POSITION.                                 TT)53470
01704 -0602 00 0 02026          ORS     DNREP         INSERT.                                            TT)53480
01705  2 00006 4 01701          TIX     *-4,4,6       INDEX.                                             TT)53490
01706  0560 00 0 02026          LDQ     DNREP         DELETE LEADING ZEROES.                             TT)53500
01707 -0154 06 0 01207          CRQ     TABBLK,0,6    ..                                                 TT)53510
01710 -0600 00 0 02037          STQ     WREP+8        PLACE IN COMMENT.                                  TT)53520
01711 -0500 00 0 02024          CAL     LWORD         CONVERT WORD.                                      TT)53530
01712  0074 00 4 02047          TSX     (OPCD),4      ..                                                 TT)53540
01713  0602 00 0 02043          SLW     WREP+12       ..                                                 TT)53550
01714  0560 00 0 02024          LDQ     LWORD         ..                                                 TT)53560
01715  0074 00 4 01260          TSX     OCT,4         ..                                                 TT)53570
01716  0602 00 0 02044          SLW     WREP+13       ..                                                 TT)53580
01717 -0600 00 0 02045          STQ     WREP+14       ..                                                 TT)53590
01720  0560 00 0 03143          LDQ     =HC           ASSUME DOUBLE SPACE FOR COMMENT.                   TT)53600
01721  0500 00 0 02022          CLA     PMCNT         GET POST MORTEM LINE COUNT.                        TT)53610
01722  0402 00 0 03130          SUB     =3            REPEAT COMMENT TAKES THREE LINES.                  TT)53620
01723  0120 00 0 01726          TPL     *+3           SKIP IF IT FITS.                                   TT)53630
01724  0560 00 0 03146          LDQ     =H1           SET TO EJECT PAGE.                                 TT)53640
01725  0500 00 0 03134          CLA     =58           RESET PM LINE COUNT.                               TT)53650
01726  0601 00 0 02022          STO     PMCNT         SAVE LINE COUNT.                                   TT)53660
01727 -0600 00 0 02027          STQ     WREP          INSERT CARRIAGE CONTROL.                           TT)53670
01730  0074 00 4 02407          TSX     WOT,4         WOT REPEATS.                                       TT)53680
01731  0 00020 0 02027          PZE     WREP,0,16     ..                                                 TT)53690
01732  0074 00 4 02407          TSX     WOT,4         BLANK LINE.                                        TT)53700
01733  0 00001 0 03160          PZE     =H      ,0,1  ..                                                 TT)53710
01734 -0500 00 0 02023          CAL     ILC           INCREMENT ILC BY NREP.                             TT)53720
01735  0361 00 0 02025          ACL     NREP          ..                                                 TT)53730
01736  0621 00 0 02023          STA     ILC           ..                                                 TT)53740
01737  0020 00 0 01653          TRA     ADR           RETURN TO CONTROL LOOP.                            TT)53750
                                                                                                         TT)53760
01740  0074 00 4 02047   REG    TSX     (OPCD),4      NORMAL PATH, CONVERT WORD.                         TT)53770
01741  0602 00 2 02771          SLW     OUT+20,2      ..                                                 TT)53780
01742  0560 00 1 00000          LDQ     **,1          CONVERT TO OCTAL.                                  TT)53790
01743  0074 00 4 01260          TSX     OCT,4         ..                                                 TT)53800
01744  0602 00 2 02772          SLW     OUT+21,2      INSERT.                                            TT)53810
01745 -0600 00 2 02773          STQ     OUT+22,2      ..                                                 TT)53820
```

```
01746 -2 00001 1 01776          TNX     FIN,1,1         COUNT WORDS CONVERTED.                  TT)53830
01747  2 00003 2 01660          TIX     LOOP1,2,3       COUNT WORDS THIS LINE.                  TT)53840
                                                                                                 TT)53850
01750  0560 00 0 03160   PPM    LDQ     =H              ASSUME SINGLE SPACE FOR THIS LINE.      TT)53860
01751  0500 00 0 02022          CLA     PMCNT           GET LINE COUNT.                         TT)53870
01752  0402 00 0 03126          SUB     =1              COUNT 1 LINE.                           TT)53880
01753  0120 00 0 01756          TPL     *+3             SKIP IF IT FITS.                        TT)53890
01754  0560 00 0 03146          LDQ     =H1             SET TO EJECT.                           TT)53900
01755  0500 00 0 03135          CLA     =59             SET LINE COUNT.                         TT)53910
01756  0601 00 0 02022          STO     PMCNT           SAVE LINE COUNT.                        TT)53920
01757 -0600 00 0 02745          STQ     OUT             SET CARRIAGE CONTROL.                   TT)53930
01760  0074 00 4 02407          TSX     WOT,4           PRINT THIS LINE.                        TT)53940
01761  0 00024 0 02745          PZE     OUT,0,20        ..                                      TT)53950
01762 -0500 00 0 02746          CAL     OUT+1           UPDATE WORD COUNT.                      TT)53960
01763 -0320 00 0 03150          ANA     =0070707070777  ..                                      TT)53970
01764  0361 00 0 03144          ACL     =000707070C7600 ..                                      TT)53980
01765 -0320 00 0 03150          ANA     =0070707070777  ..                                      TT)53990
01766 -0130 00 0 00000          XCL                     ..                                      TT)54000
01767 -0154 04 0 01207          CRQ     TABBLK,0,4      REMOVE LEADING ZEROES.                  TT)54010
01770 -0773 00 0 00014          RQL     12              BACK TO PROPER POSITION.                TT)54020
01771 -0600 00 0 02746          STQ     OUT+1           INSERT.                                 TT)54030
01772 -0500 00 0 02023          CAL     ILC             UPDATE ILC.                             TT)54040
01773  0361 00 0 03132          ACL     =6              ..                                      TT)54050
01774  0621 00 0 02023          STA     ILC             ..                                      TT)54060
01775  0020 00 0 01657          TRA     LOOP2           ..                                      TT)54070
01776 -2 00003 2 02004   FIN    TNX     *+6,2,3         IF LINE NOT FULL,                       TT)54080
01777 -0500 00 0 03160          CAL     =H              FILL IN LINE WITH BLANKS.               TT)54090
02000  0602 00 2 02771          SLW     OUT+20,2        ..                                      TT)54100
02001  0602 00 2 02772          SLW     OUT+21,2        ..                                      TT)54110
02002  0602 00 2 02773          SLW     OUT+22,2        ..                                      TT)54120
02003  2 00003 2 02000          TIX     *-3,2,3         INDEX.                                  TT)54130
02004  0560 00 0 03160          LDQ     =H              ASSUME SINGLE SPACE FOR LAST LINE.      TT)54140
02005  0500 00 0 02022          CLA     PMCNT           GET LINE COUNT.                         TT)54150
02006  0402 00 0 03126          SUB     =1              COUNT 1 LINE.                           TT)54160
02007  0120 00 0 02012          TPL     *+3             SKIP IF IT FITS.                        TT)54170
02010  0560 00 0 03146          LDQ     =H1             SET TO EJECT.                           TT)54180
02011  0500 00 0 03135          CLA     =59             SET LINE COUNT.                         TT)54190
02012  0601 00 0 02022          STO     PMCNT           SAVE LINE COUNT.                        TT)54200
02013 -0600 00 0 02745          STQ     OUT             SET CARRIAGE CONTROL.                   TT)54210
02014  0074 00 4 02407          TSX     WOT,4           PRINT LAST LINE.                        TT)54220
02015  0 00024 0 02745          PZE     OUT,0,20        ..                                      TT)54230
02016  0774 00 4 00000   CX4    AXT     **,4            RESTORE IRS.                            TT)54240
02017  0774 00 2 00000   CX2    AXT     **,2            ..                                      TT)54250
02020  0774 00 1 00000   CX1    AXT     **,1            ..                                      TT)54260
02021  0020 00 4 00002          TRA     2,4             RETURN.                                 TT)54270
                                                                                                 TT)54280
02022  0 00000 0 00055   PMCNT  PZE     45              STORAGE FOR LINE COUNT.                 TT)54290
02023  0 00000 0 00000   ILC    PZE                     POST MORTEM STORAGE.                    TT)54300
02024  0 00000 0 00000   LWORC  PZE                     ..                                      TT)54310
02025  0 00000 0 00000   NREP   PZE                     ..                                      TT)54320
02026  0 00000 0 00000   CNREP  PZE                     ..                                      TT)54330
02027  006060606060      WREP   BCI     9,0                     .. FOLLOWING 00000              TT)54340
02030  606060606060
02031  606060606060
02032  606060606060
```

```
02033  606060606060
02034  606060606060
02035  333360264643
C2036  434666314527
02037  600000000000
02040  602325434362          BCI     7, CELLS ALL CONTAIN 0000000000000000 ..                TT)54350
02041  602143436023
C2042  464563213145
02043  600000000000
02044  000000000000
02045  000000000000
02046  603333606060


                              SPACE   2                                                       TT)54360
                                      (OPCD), FIND MNEMONIC OPERATION CODE.                   TT)54370
                                                                                              TT)54380
02047   0634 00 2 02131  (OPCD) SXA     AR2,2               SAVE IR2.                        TT)54390
02050   0634 00 1 02132        SXA     AR1,1               SAVE IR1.                        TT)54400
C2051   0774 00 1 00000        AXT     0,1                 SET IR1.                         TT)54410
02052  -0320 00 0 03165        ANA     =0777700000000      MASK OPCD.                       TT)54420
02053   0100 00 0 02125        TZE     OPFND               HTR EXIT.                        TT)54430
02054   0602 00 0 02135        SLW     OPBIN               SAVE OPCD.                       TT)54440
02055   0630 00 0 02134        STP     PRE                 SAVE PREFIX.                     TT)54450
02056   0520 00 0 02134        ZET     PRE                 IF NON-ZERO, TYPE A.             TT)54460
02057   0020 00 0 02117        TRA     TYPEA               ..                               TT)54470
02060   0534 00 2 02406        LXA     SIZE,2              SET IR2.                         TT)54480
02061   0020 00 2 02116        TRA     LOWER,2             BEGIN SEARCH.                    TT)54490
                                                                                              TT)54500
02062  -3 77540 1 02066  SRCH1 TXL     SRCH2,1,-NUM        CHECK RANGE.                     TT)54510
02063  -0500 00 1 02146        CAL     TABL,1              GET TABLE ENTRY.                 TT)54520
02064  -0320 00 0 03165        ANA     =0777700000000      MASK OPCD.                       TT)54530
02065  -0340 00 0 02135        LAS     OPBIN               COMPARE WITH OPBIN.              TT)54540
02066   1 77776 2 02115  SRCH2 TXI     RAISE,2,-2          BIGGER, GO TO RAISE INDEX.       TT)54550
02067   0020 00 0 02125        TRA     OPFND               FOUND, EXIT WITH INDEX.          TT)54560
02070   1 77776 2 02116        TXI     LOWER,2,-2          SMALLER, GO TO LOWER INDEX.      TT)54570
02071   1 00400 1 02062        TXI     SRCH1,1,+256        TABLE, POWER OF TWO INCREMENTS.  TT)54580
02072   1 77400 1 02062        TXI     SRCH1,1,-256        ..                               TT)54590
02073   1 00200 1 02062        TXI     SRCH1,1,+128        ..                               TT)54600
02074   1 77600 1 02062        TXI     SRCH1,1,-128        ..                               TT)54610
02075   1 00100 1 02062        TXI     SRCH1,1,+64         ..                               TT)54620
02076   1 77700 1 02062        TXI     SRCH1,1,-64         ..                               TT)54630
02077   1 00040 1 02062        TXI     SRCH1,1,+32         ..                               TT)54640
02100   1 77740 1 02062        TXI     SRCH1,1,-32         ..                               TT)54650
02101   1 00020 1 02062        TXI     SRCH1,1,+16         ..                               TT)54660
02102   1 77760 1 02062        TXI     SRCH1,1,-16         ..                               TT)54670
02103   1 00010 1 02062        TXI     SRCH1,1,+8          ..                               TT)54680
02104   1 77770 1 02062        TXI     SRCH1,1,-8          ..                               TT)54690
02105   1 00004 1 02062        TXI     SRCH1,1,+4          ..                               TT)54700
02106   1 77774 1 02062        TXI     SRCH1,1,-4          ..                               TT)54710
02107   1 00002 1 02062        TXI     SRCH1,1,+2          ..                               TT)54720
02110   1 77776 1 02062        TXI     SRCH1,1,-2          ..                               TT)54730
02111   1 00001 1 02062        TXI     SRCH1,1,+1          ..                               TT)54740
02112   1 77777 1 02062        TXI     SRCH1,1,-1          ..                               TT)54750
02113   0020 00 0 02117        TRA     TYPEA               NOT FOUND, TYPE A.               TT)54760
```

```
02114  0020 00 0 02117          TRA     TYPEA                   ..                      TT)54770
02115  0522 00 2 02115   RAISE  XEC     *,2                     RAISE INDEX.            TT)54780
02116  0522 00 2 02116   LOWER  XEC     *,2                     LOWER INDEX.            TT)54790
                                                                                       TT)54800
02117  0560 00 0 02135   TYPEA  LDQ     OPBIN                   TYPE A, PICKUP MNEMONIC.TT)54810
02120 -0754 00 0 00000          PXD     0,0                     ..                      TT)54820
02121 -0763 00 0 00003          LGL     3                       ..                      TT)54830
02122  0734 00 1 00000          PAX     0,1                     ..                      TT)54840
02123 -0500 00 1 02145          CAL     LSTA+7,1                PICKUP MNEMONIC.        TT)54850
02124  0020 00 0 02131          TRA     AR2                     GO TO EXIT.             TT)54860
                                                                                       TT)54870
02125 -0500 00 1 02146   OPFND  CAL     TABL,1                  OPERATION FOUND, PICKUP WORD.TT)54880
02126 -0320 00 0 03140          ANA     =0000000777777          MASK OUT JUNK.          TT)54890
02127  0767 00 0 00006          ALS     6                       SHIFT LEFT ONE CHARACTER.TT)54900
02130 -0501 00 0 03156          ORA     =H  000                 INSERT BLANKS.          TT)54910
02131  0774 00 2 00000   AR2    AXT     **,2                    RESTORE IRS.            TT)54920
02132  0774 00 1 00000   AR1    AXT     **,1                    ..                      TT)54930
02133  0020 00 4 00001          TRA     1,4                     RETURN TO CALLER.       TT)54940
                                                                                       TT)54950
02134  0 00000 0 00000   PRE    PZE                             PREFIX STORAGE FOR TYPE A TEST.TT)54960
02135  0 00000 0 00000   OPBIN  PZE                             STORAGE FOR BINARY OPCD.TT)54970

                                 SPACE   2                                             TT)54980
                                         TABLES FOR (OPCD).                            TT)54990
                                                                                       TT)55000
02136  606063674360      LSTA   BCI     1,  TXL                                        TT)55010
02137  606063456760             BCI     1,  TNX                                        TT)55020
02140  606062635160             BCI     1,  STR                                        TT)55030
02141  606044712560             BCI     1,  MZE                                        TT)55040
02142  606063673060             BCI     1,  TXH                                        TT)55050
02143  606063316760             BCI     1,  TIX                                        TT)55060
02144  606063673160             BCI     1,  TXI                                        TT)55070
02145  606047712560             BCI     1,  PZE                                        TT)55080
                                                                                       TT)55090
02146  000000306351      TABL   VFD     012/0000,H24/0HTR  HTR                         TT)55100
02147  002000635121             VFD     012/0020,H24/0TRA  TRA                         TT)55110
02150  002100636351             VFD     012/0021,H24/0TTR  TTR                         TT)55120
02151  002200635123             VFD     012/0022,H24/0TRC  TRCA                        TT)55130
02152  002400635123             VFD     012/0024,H24/0TRC  TRCC                        TT)55140
02153  003000632526             VFD     012/0030,H24/0TEF  TEFA                        TT)55150
02154  003100632526             VFD     012/0031,H24/0TEF  TEFC                        TT)55160
02155  004000634350             VFD     012/0040,H24/0TLQ  TLQ                         TT)55170
02156  004100313121             VFD     012/0041,H24/0IIA  IIA                         TT)55180
02157  004200633146             VFD     012/0042,H24/0TIO  TIO                         TT)55190
02160  004300462131             VFD     012/0043,H24/0OAI  OAI                         TT)55200
02161  004400472131             VFD     012/0044,H24/0PAI  PAI                         TT)55210
02162  004600633126             VFD     012/0046,H24/0TIF  TIF                         TT)55220
02163  005100313151             VFD     012/0051,H24/0IIR  IIR                         TT)55230
02164  005400512663             VFD     012/0054,H24/0RFT  RFT                         TT)55240
02165  005500623151             VFD     012/0055,H24/0SIR  SIR                         TT)55250
02166  005600514563             VFD     012/0056,H24/0RNT  RNT                         TT)55260
02167  005700513151             VFD     012/0057,H24/0RIR  RIR                         TT)55270
02170  006000632346             VFD     012/0060,H24/0TCO  TCOA                        TT)55280
02171  006100632346             VFD     012/0061,H24/0TCO  TCOB                        TT)55290
```

```
02172  006200632346          VFD     012/0062,H24/OTCO   TCOC                TT)55300
02173  006300632346          VFD     012/0063,H24/OTCO   TCOD                TT)55310
02174  007400636267          VFD     012/0074,H24/OTSX   TSX                 TT)55320
02175  010000637125          VFD     012/0100,H24/OTZE   TZE                 TT)55330
02176  011400236551          VFD     012/0114,H24/OCVR   CVR                 TT)55340
02177  012000634743          VFD     012/0120,H24/OTPL   TPL                 TT)55350
02200  013100672321          VFD     012/0131,H24/OXCA   XCA                 TT)55360
02201  014000634665          VFD     012/0140,H24/OTOV   TOV                 TT)55370
02202  016100635046          VFD     012/0161,H24/OTQO   TQO                 TT)55380
02203  016200635047          VFD     012/0162,H24/OTQP   TQP                 TT)55390
02204  020000444770          VFD     012/0200,H24/OMPY   MPY                 TT)55400
02205  020400654344          VFD     012/0204,H24/OVLM   VLM                 TT)55410
02206  022000246530          VFD     012/0220,H24/ODVH   DVH                 TT)55420
02207  022100246547          VFD     012/0221,H24/ODVP   DVP                 TT)55430
02210  022400652430          VFD     012/0224,H24/OVDH   VDH                 TT)55440
02211  022500652447          VFD     012/0225,H24/OVDP   VDP                 TT)55450
02212  024000262430          VFD     012/0240,H24/OFDH   FDH                 TT)55460
02213  024100262447          VFD     012/0241,H24/OFDP   FDP                 TT)55470
02214  026000264447          VFD     012/0260,H24/OFMP   FMP                 TT)55480
02215  030000262124          VFD     012/0300,H24/OFAD   FAD                 TT)55490
02216  030200266222          VFD     012/0302,H24/OFSB   FSB                 TT)55500
02217  030400262144          VFD     012/0304,H24/OFAM   FAM                 TT)55510
02220  030600266244          VFD     012/0306,H24/OFSM   FSM                 TT)55520
02221  032000214562          VFD     012/0320,H24/OANS   ANS                 TT)55530
02222  032200255121          VFD     012/0322,H24/OERA   ERA                 TT)55540
02223  034000232162          VFD     012/0340,H24/OCAS   CAS                 TT)55550
02224  036100212343          VFD     012/0361,H24/OACL   ACL                 TT)55560
02225  040000212424          VFD     012/0400,H24/OADD   ADD                 TT)55570
02226  040100212444          VFD     012/0401,H24/OADM   ADM                 TT)55580
02227  040200626422          VFD     012/0402,H24/OSUB   SUB                 TT)55590
02230  042000304751          VFD     012/0420,H24/OHPR   HPR                 TT)55600
02231  044000313162          VFD     012/0440,H24/OIIS   IIS                 TT)55610
02232  044100432431          VFD     012/0441,H24/OLDI   LDI                 TT)55620
02233  044200466231          VFD     012/0442,H24/OOSI   OSI                 TT)55630
02234  044400462663          VFD     012/0444,H24/OOFT   OFT                 TT)55640
02235  044500513162          VFD     012/0445,H24/ORIS   RIS                 TT)55650
02236  044600464563          VFD     012/0446,H24/OONT   ONT                 TT)55660
02237  046000432421          VFD     012/0460,H24/OLDA   LDA                 TT)55670
02240  050000234321          VFD     012/0500,H24/OCLA   CLA                 TT)55680
02241  050200234362          VFD     012/0502,H24/OCLS   CLS                 TT)55690
02242  052000712563          VFD     012/0520,H24/OZET   ZET                 TT)55700
02243  052200672523          VFD     012/0522,H24/OXEC   XEC                 TT)55710
02244  053400436721          VFD     012/0534,H24/OLXA   LXA                 TT)55720
02245  053500432123          VFD     012/0535,H24/OLAC   LAC                 TT)55730
02246  054000512330          VFD     012/0540,H24/ORCH   RCHA                TT)55740
02247  054100512330          VFD     012/0541,H24/ORCH   RCHC                TT)55750
02250  054400432330          VFD     012/0544,H24/OLCH   LCHA                TT)55760
02251  054500432330          VFD     012/0545,H24/OLCH   LCHC                TT)55770
02252  056000432450          VFD     012/0560,H24/OLDQ   LDQ                 TT)55780
02253  056200435131          VFD     012/0562,H24/OLRI   LRI                 TT)55790
02254  056400254522          VFD     012/0564,H24/OENB   ENB                 TT)55800
02255  060000626371          VFD     012/0600,H24/OSTZ   STZ                 TT)55810
02256  060100626346          VFD     012/0601,H24/OSTO   STO                 TT)55820
02257  060200624366          VFD     012/0602,H24/OSLW   SLW                 TT)55830
02260  060400626331          VFD     012/0604,H24/OSTI   STI                 TT)55840
```

```
02261  062100626321          VFD     012/0621,H24/0STA   STA          TT)55850
02262  062200626324          VFD     012/0622,H24/0STD   STD          TT)55860
02263  062500626363          VFD     012/0625,H24/0STT   STT          TT)55870
02264  063000626347          VFD     012/0630,H24/0STP   STP          TT)55880
02265  063400626721          VFD     012/0634,H24/0SXA   SXA          TT)55890
02266  064000622330          VFD     012/0640,H24/0SCH   SCHA         TT)55900
02267  064100622330          VFD     012/0641,H24/0SCH   SCHA         TT)55910
02270  070000234770          VFD     012/0700,H24/0CPY   CPY          TT)55920
02271  073400472167          VFD     012/0734,H24/0PAX   PAX          TT)55930
02272  073700472123          VFD     012/0737,H24/0PAC   PAC          TT)55940
02273  075400476721          VFD     012/0754,H24/0PXA   PXA          TT)55950
02274  076000476225          VFD     012/0760,H24/0PSE   PSE          TT)55960
02275  076100454647          VFD     012/0761,H24/0NOP   NOP          TT)55970
02276  076200512462          VFD     012/0762,H24/0RDS   RDS          TT)55980
02277  076300434362          VFD     012/0763,H24/0LLS   LLS          TT)55990
02300  076400226251          VFD     012/0764,H24/0BSR   BSR          TT)56000
02301  076500435162          VFD     012/0765,H24/0LRS   LRS          TT)56010
02302  076600665162          VFD     012/0766,H24/0WRS   WRS          TT)56020
02303  076700214362          VFD     012/0767,H24/0ALS   ALS          TT)56030
02304  077000662526          VFD     012/0770,H24/0WEF   WEF          TT)56040
02305  077100215162          VFD     012/0771,H24/0ARS   ARS          TT)56050
02306  077200512566          VFD     012/0772,H24/0REW   REW          TT)56060
02307  077400216763          VFD     012/0774,H24/0AXT   AXT          TT)56070
02310  077600622445          VFD     012/0776,H24/0SDN   SDN          TT)56080
02311  402100254563          VFD     012/4021,H24/0ENT   ESNT         TT)56090
02312  402200635123          VFD     012/4022,H24/0TRC   TRCB         TT)56100
02313  402400635123          VFD     012/4024,H24/0TRC   TRCD         TT)56110
02314  403000632526          VFD     012/4030,H24/0TEF   TEFB         TT)56120
02315  403100632526          VFD     012/4031,H24/0TEF   TEFD         TT)56130
02316  404200513121          VFD     012/4042,H24/0RIA   RIA          TT)56140
02317  404600473121          VFD     012/4046,H24/0PIA   PIA          TT)56150
02320  405100313143          VFD     012/4051,H24/0IIL   IIL          TT)56160
02321  405400432663          VFD     012/4054,H24/0LFT   LFT          TT)56170
02322  405500623143          VFD     012/4055,H24/0SIL   SIL          TT)56180
02323  405600434563          VFD     012/4056,H24/0LNT   LNT          TT)56190
02324  405700513143          VFD     012/4057,H24/0RIL   RIL          TT)56200
02325  406000632345          VFD     012/4060,H24/0TCN   TCNA         TT)56210
02326  406100632345          VFD     012/4061,H24/0TCN   TCNB         TT)56220
02327  406200632345          VFD     012/4062,H24/0TCN   TCNC         TT)56230
02330  406300632345          VFD     012/4063,H24/0TCN   TCND         TT)56240
02331  410000634571          VFD     012/4100,H24/0TNZ   TNZ          TT)56250
02332  411400232150          VFD     012/4114,H24/0CAQ   CAQ          TT)56260
02333  412000634431          VFD     012/4120,H24/0TMI   TMI          TT)56270
02334  413000672343          VFD     012/4130,H24/0XCL   XCL          TT)56280
02335  414000634546          VFD     012/4140,H24/0TNO   TNO          TT)56290
02336  415400235150          VFD     012/4154,H24/0CRQ   CRQ          TT)56300
02337  420000444751          VFD     012/4200,H24/0MPR   MPR          TT)56310
02340  426000642644          VFD     012/4260,H24/0UFM   UFM          TT)56320
02341  430000642621          VFD     012/4300,H24/0UFA   UFA          TT)56330
02342  430200642662          VFD     012/4302,H24/0UFS   UFS          TT)56340
02343  430400642144          VFD     012/4304,H24/0UAM   UAM          TT)56350
02344  430600646244          VFD     012/4306,H24/0USM   USM          TT)56360
02345  432000214521          VFD     012/4320,H24/0ANA   ANA          TT)56370
02346  434000432162          VFD     012/4340,H24/0LAS   LAS          TT)56380
02347  440000622244          VFD     012/4400,H24/0SBM   SBM          TT)56390
```

```
02350  450000232143            VFD     012/4500,H24/0CAL   CAL                 TT)56400
02351  450100465121            VFD     012/4501,H24/00RA   ORA                 TT)56410
02352  452000457163            VFD     012/4520,H24/0NZT   NZT                 TT)56420
02353  453400436724            VFD     012/4534,H24/0LXD   LXD                 TT)56430
02354  453500432423            VFD     012/4535,H24/0LDC   LDC                 TT)56440
02355  454000512330            VFD     012/4540,H24/0RCH   RCHB                TT)56450
02356  454100512330            VFD     012/4541,H24/0RCH   RCHD                TT)56460
02357  454400432330            VFD     012/4544,H24/0LCH   LCHB                TT)56470
02360  454500432330            VFD     012/4545,H24/0LCH   LCHD                TT)56480
02361  456400434731            VFD     012/4564,H24/0LPI   LPI                 TT)56490
02362  460000626350            VFD     012/4600,H24/0STQ   STQ                 TT)56500
02363  460100625131            VFD     012/4601,H24/0SRI   SRI                 TT)56510
02364  460200465162            VFD     012/4602,H24/00RS   ORS                 TT)56520
02365  460400624731            VFD     012/4604,H24/0SPI   SPI                 TT)56530
02366  462000624350            VFD     012/4620,H24/0SLQ   SLQ                 TT)56540
02367  462500626343            VFD     012/4625,H24/0STL   STL                 TT)56550
02370  463400626724            VFD     012/4634,H24/0SXD   SXD                 TT)56560
02371  464000622330            VFD     012/4640,H24/0SCH   SCHB                TT)56570
02372  464100622330            VFD     012/4641,H24/0SCH   SCHD                TT)56580
02373  470000232124            VFD     012/4700,H24/0CAD   CAD                 TT)56590
02374  473400472467            VFD     012/4734,H24/0PDX   PDX                 TT)56600
02375  473700472423            VFD     012/4737,H24/0PDC   PDC                 TT)56610
02376  475400476724            VFD     012/4754,H24/0PXD   PXD                 TT)56620
02377  476000446225            VFD     012/4760,H24/0MSE   MSE                 TT)56630
02400  476300432743            VFD     012/4763,H24/0LGL   LGL                 TT)56640
02401  476400226226            VFD     012/4764,H24/0BSF   BSF                 TT)56650
02402  476500432751            VFD     012/4765,H24/0LGR   LGR                 TT)56660
02403  477200516445            VFD     012/4772,H24/0RUN   RUN                 TT)56670
02404  477300515043            VFD     012/4773,H24/0RQL   RQL                 TT)56680
02405  477400216723            VFD     012/4774,H24/0AXC   AXC                 TT)56690
02406                  ENTBL    BSS     0                                       TT)56700
02406  0 77540 0 00022 SIZE     PZE     18,,-NUM                                TT)56710
                                                                               TT)56720
                       NB               ADDRESS OF SIZE IS (2*E+2),WHERE E IS A NUMBER SUCH THAT TT)56730
                                             (2**E).GE.(NUMBER OF ENTRIES IN TABLE).G.(2**(E-1)) TT)56740
                                        DECREMENT OF SIZE IS COMPLEMENT OF TABLE LENGTH.  TT)56750
                                                                               TT)56760
            00240      NUM   EQU       ENTBL-TABL    NUMBER IN OP TABLE.      TT)56770
```

```
                                TTL     SECTION 6, INPUT/OUTPUT SUBROUTINES.                                    TT)60000
                                        BUFFERED WOT, CALLING SEQUENCE,                                         TT)60010
                                                                                                                TT)60020
                                             TSX  $WOT,4        OR        TSX  $WOT,4                         TT)60030
                                             PZE  FIRST,T,LAST            PZE  FIRST,T,N                       TT)60040
                                                                                                                TT)60050
                                        IF T IS NON-ZERO OR IF SW5 IS DOWN, PRINT ALSO.                         TT)60060
                                        WRITES ONLY FIRST 22 WORDS ON TAPE.                                     TT)60070
                                                                                                                TT)60080
02407  0634 00 4 02442   WOT    SXA     WTX4,4          SAVE IR4.                                               TT)60090
02410  0060 00 0 02410          TCOA    *               WAIT FOR DSC A.                                         TT)60100
02411 -0760 00 0 01000          ETTA                    CHECK FOR EOT CHN A.                                    TT)60110
02412  0020 00 0 02452          TRA     ETA             EOT, EXIT.                                              TT)60120
02413 -0500 00 4 00001          CAL     1,4             GET CONTROL WORD.                                       TT)60130
02414  0625 00 0 02451          STT     WTTAG           SAVE TAG FOR DECISION TO PRINT.                         TT)60140
02415  0734 00 4 00000          PAX     0,4             FIRST TO IR4.                                           TT)60150
02416 -0634 00 4 02432          SXD     WTTXI,4         SAVE FOR BUFFERING.                                     TT)60160
02417 -0634 00 4 02422          SXD     WTTIX,4         SAVE FOR FINDING N.                                     TT)60170
02420 -0734 00 4 00000          PDX     0,4             N OR LAST TO IR4.                                       TT)60180
02421  1 00001 4 02422          TXI     *+1,4,1         PLUS 1.                                                 TT)60190
02422  2 00000 4 02424   WTTIX  TIX     *+2,4,**        IS IT N OR LAST.                                        TT)60200
02423  1 77777 4 02424          TXI     *+1,4,-1        N IN IR4.                                               TT)60210
02424 -3 00026 4 02426          TXL     *+2,4,22        N .LE. 22.                                              TT)60220
02425  0774 00 4 00026          AXT     22,4            N = 22.                                                 TT)60230
02426 -0634 00 4 02450          SXD     WTCOM,4         SAVE FOR DSC COMMAND.                                   TT)60240
02427  1 02715 4 02430          TXI     *+1,4,BUFF      FORM BUFF+N.                                            TT)60250
02430  0634 00 4 02436          SXA     WTSLW,4         SAVE FOR BUFFERING.                                     TT)60260
02431 -0534 00 4 02450          LXD     WTCOM,4         N IN IR4.                                               TT)60270
02432  1 00000 4 02433   WTTXI  TXI     *+1,4,**        FORM FIRST+N.                                           TT)60280
02433  0634 00 4 02435          SXA     WTCAL,4         SAVE FOR BUFFERING.                                     TT)60290
02434 -0534 00 4 02450          LXD     WTCOM,4         N IN IR4.                                               TT)60300
02435 -0500 00 4 00000   WTCAL  CAL     **,4            MOVE TO BUFFER.                                         TT)60310
02436  0602 00 4 00000   WTSLW  SLW     **,4            ..                                                      TT)60320
02437  2 00001 4 02435          TIX     *-2,4,1         ..                                                      TT)60330
02440  0766 00 0 01203          WTDA    3               SELECT OUTPUT TAPE.                                     TT)60340
02441  0540 00 0 02450          RCHA    WTCOM           LOAD TO WRITE OUT BUFFER.                               TT)60350
02442  0774 00 4 00000   WTX4   AXT     **,4            RESTORE IR4.                                            TT)60360
02443  0520 00 0 02451          ZET     WTTAG           WAS THE TAG ZERO.                                       TT)60370
02444  0020 00 0 02520          TRA     PRINT           NO, THEN PRINT ALSO.                                    TT)60380
02445  0760 00 0 00165          SWT     5               ZERO TAG, BUT CHECK SW5.                                TT)60390
02446  0020 00 4 00002          TRA     2,4             UP, DON'T PRINT.                                        TT)60400
02447  0020 00 0 02520          TRA     PRINT           DOWN, PRINT ALSO.                                       TT)60410
                                                                                                                TT)60420
02450  0 00000 0 02715   WTCOM  IOCD    BUFF,0,**       OUTPUT COMMAND, N IN DECREMENT.                         TT)60430
02451  0 00000 0 00000   WTTAG  PZE     0,0,0           STORAGE FOR TAG.                                        TT)60440
                                                                                                                TT)60450
02452 -0764 00 0 01202   ETA    BSFA    2               END OF TAPE A3, SET FOR RESTART.                        TT)60460
02453 -0764 00 0 02204          BSFB    4               REMOVE BINARY OUTPUT.                                   TT)60470
02454 -0764 00 0 02204          BSFB    4               ..                                                      TT)60480
02455  0774 00 1 00012          AXT     10,1            10 CLOSE COMMENTS.                                      TT)60490
02456  0074 00 4 02407          TSX     WOT,4           CLOSE COMMENT.                                          TT)60500
02457  0 00003 0 01077          PZE     ENDTP,0,3       ..                                                      TT)60510
02460  0074 00 4 02407          TSX     WOT,4           CHANNEL 4 SKIP.                                         TT)60520
02461  0 00001 0 03147          PZE     =H4     ,0,1    ..                                                      TT)60530
02462  0770 00 0 01203          WEFA    3               END OF FILE MARK.                                       TT)60540
```

```
02463   2 00001 1 02456          TIX     *-5,1,1         COUNT 10 TIMES.                                    TT)60550
02464  -0772 00 0 01203          RUNA    3               THEN RUN A3.                                       TT)60560
02465   0441 00 0 03125          LDI     =0              CLEAR INDICATORS.                                  TT)60570
02466   0760 00 0 01000          BTTA                    TURN OFF BOT LIGHT.                                TT)60580
02467   0761 00 0 00000          NOP                     ..                                                 TT)60590
02470   0074 00 4 02520          TSX     PRINT,4         COMMENT, AND STOP.                                 TT)60600
02471  -0 00022 0 02476          MZE     WEOTA3,0,18     ..                                                 TT)60610
                                                                                                            TT)60620
02472   0760 00 0 01000          BTTA                    CHECK FOR BEGINNING OF TAPE.                       TT)60630
02473   0020 00 0 00765          TRA     DOOR            BOT, GO RESTART.                                   TT)60640
02474  -0764 00 0 01203          BSFA    3               NOT BOT, BACKSPACE TILL THERE.                     TT)60650
02475   0020 00 0 02472          TRA     *-3             THEN TRY AGAIN.                                    TT)60660
                                                                                                            TT)60670
02476   002545246046     WEOTA3  BCI     9,0END OF TAPE MARK ENCOUNTERED ON TAPE A3, TAPE CLOSED.           TT)60680
02477   266063214725
02500   604421514260
02501   254523466445
02502   632551252460
02503   464560632147
02504   256021037360
02505   632147256023
02506   434662252433
02507   602330214527             BCI     9, CHANGE TAPE AND PRESS START TO RESTART THIS JOB.                TT)60690
02510   256063214725
02511   602145246047
02512   512562626062
02513   632151636063
02514   466051256263
02515   215163606330
02516   316260414622
02517   336060606060


                                 SPACE   2                                                                  TT)60700
                                         BUFFERED PRINT, CALLING SEQUENCE,                                  TT)60710
                                                                                                            TT)60720
                                              TSX  $PRINT,4      OR       TSX  $PRINT,4                     TT)60730
                                              PRE  FIRST,T,LAST           PRE  FIRST,T,N                    TT)60740
                                                                                                            TT)60750
                                         RECOGNIZES BLANK, 0, AND 1 AS CARRIAGE CONTROL (OTHER              TT)60760
                                         CARRIAGE CONTROLS DON'T WORK TOO WELL), OTHERWISE,                  TT)60770
                                         SINGLE SPACE. PRINTS ONLY 20 WORDS, AND T IS IGNORED.              TT)60780
                                         PRE=PZE, NORMAL RETURN (2,4), PRE=MZE, SPACE 0.1 PAGE,             TT)60790
                                         LIGHT UP CONSOLE, AND STOP (HPR -1,7,63). RESTART                  TT)60800
                                         RETURNS (2,4).                                                     TT)60810
                                                                                                            TT)60820
02520   0634 00 4 02642  PRINT  SXA     PR4,4           SAVE IRS.                                          TT)60830
02521   0634 00 2 02643          SXA     PR2,2           ..                                                 TT)60840
02522   0634 00 1 02644          SXA     PR1,1           ..                                                 TT)60850
02523   0600 00 0 00005          STZ     5               RESET INTERVAL TIMER.                              TT)60860
02524  -0061 00 0 02526          TCNT    *+2             IF TAPE TIMER IN USE,                              TT)60870
02525  -0540 00 0 00261          RCHT    QUIT            RESET TAPE TIMER.                                  TT)60880
02526  -0500 00 4 00001          CAL     1,4             GET CONTROL WORD.                                  TT)60890
02527   0734 00 4 00000          PAX     0,4             FIRST TO IR4.                                      TT)60900
02530  -0634 00 4 02541          SXD     PTXI,4          SAVE FOR PICKUP.                                   TT)60910
```

```
02531 -0634 00 4 02534        SXD     *+3,4           SAVE FOR FINDING N.                 TT)60920
02532 -0734 00 4 00000        PDX     0,4             N OR LAST TO IR4.                   TT)60930
02533  1 00001 4 02534        TXI     *+1,4,1         PLUS 1.                             TT)60940
02534  2 00000 4 02536        TIX     *+2,4,**        IS IT N OR LAST.                    TT)60950
02535  1 77777 4 02536        TXI     *+1,4,-1        N IN IR4.                           TT)60960
02536 -3 00024 4 02540        TXL     *+2,4,20        IS N .LE. 20.                       TT)60970
02537  0774 00 4 00024        AXT     20,4            N = 20.                             TT)60980
02540  0634 00 4 02614        SXA     WDCNT,4         SAVE WDCNT.                         TT)60990
02541  1 00000 4 02542  PTXI  TXI     *+1,4,**        COMPUTE LAST+1.                     TT)61000
02542  0634 00 4 02616        SXA     LDQ,4           INSERT IN PICKUP.                   TT)61010
02543  0534 00 4 02614        LXA     WDCNT,4         N IN IR4.                           TT)61020
02544  0522 00 0 02616        XEC     LDQ             PICKUP FIRST WORD.                  TT)61030
02545 -0754 00 0 00000        PXD     0,C             GET FIRST CHARACTER,                TT)61040
02546 -0763 00 0 00006        LGL     6               FOR CARRIAGE CONTROL.               TT)61050
02547 -0340 00 0 03126        LAS     =1              COMPARE WITH 1.                     TT)61060
02550 -0500 00 0 03127        CAL     =2              C.C. .G. 1, SINGLE SPACE.           TT)61070
02551  0020 00 0 02552        TRA     *+1             C.C. .E. 1, NEW PAGE.               TT)61080
02552  0737 00 2 00000        PAC     0,2             C.C. .E. 0, DOUBLE SPACE.           TT)61090
02553 -0500 00 2 02660        CAL     SPRS,2          PICKUP SPR,                         TT)61100
02554  0602 00 0 02633        SLW     SPRA            AND INSERT AFTER SELECT.            TT)61110
                                                                                          TT)61120
02555  0774 00 4 00005        AXT     5,4             SET FOR FIVE CHARACTERS, FIRST WORD. TT)61130
02556 -0500 00 0 03151        CAL     =020000000000   COLUMN MARK, SKIP 1ST COLUMN.       TT)61140
                                                                                          TT)61150
02557  0774 00 1 00030  HRI   AXT     24,1            CLEAR CARD IMAGE.                   TT)61160
02560  0600 00 1 02715        STZ     CARDIM+24,1     ..                                  TT)61170
02561  0600 00 1 02716        STZ     CARDIM+25,1     ..                                  TT)61180
02562  2 00002 1 02560        TIX     *-2,1,2         ..                                  TT)61190
02563  0774 00 2 00001        AXT     1,2             SET MARK FOR LEFT HALF.             TT)61200


                              SPACE   2                                                   TT)61210
02564  0602 00 0 02657  PRLP  SLW     PRCOL           SAVE COLUMN MARKER.                 TT)61220
02565 -0754 00 0 00000        PXD     0,0             GET NEXT CHARACTER.                 TT)61230
02566 -0763 00 0 00006        LGL     6               ..                                  TT)61240
02567  0767 00 0 00001        ALS     1               DOUBLE IT,                          TT)61250
02570  0734 00 1 00000        PAX     0,1             AND PLACE IN IR1.                   TT)61260
02571 -0500 00 0 02657        CAL     PRCOL           GET COLUMN MARKER.                  TT)61270
                                                                                          TT)61280
02572 -3 00037 1 02606  ZONE  TXL     DIG,1,31        IF NO ZONE, SKIP.                   TT)61290
02573 -3 00137 1 02575        TXL     *+2,1,95        IGNORE BLANK.                       TT)61300
02574 -3 00140 1 02612        TXL     RLOOP,1,96      ..                                  TT)61310
02575  3 00077 1 02600        TXH     *+3,1,63        CHECK FOR 12 ZONE.                  TT)61320
02576 -0602 00 2 02714        ORS     CARDIM+23,2     YES, OR IT IN.                      TT)61330
02577  1 77740 1 02605        TXI     TZEDG,1,-32     REMOVE ZONE, AND CHECK DIGIT.       TT)61340
02600  3 00137 1 02603        TXH     *+3,1,95        CHECK FOR 11 ZONE.                  TT)61350
02601 -0602 00 2 02712        ORS     CARDIM+21,2     YES, OR IT IN.                      TT)61360
02602  1 77700 1 02605        TXI     TZEDG,1,-64     REMOVE ZONE AND CHECK DIGIT.        TT)61370
02603 -0602 00 2 02710        ORS     CARDIM+19,2     OR IN 0 ZONE.                       TT)61380
02604  1 77640 1 02605        TXI     *+1,1,-96       REMOVE ZONE, AND CHECK DIGIT.       TT)61390
02605 -3 00000 1 02612  TZEDG TXL     RLOOP,1,0       IGNORE 0 DIGIT WITH A ZONE.         TT)61400
                                                                                          TT)61410
02606 -3 00022 1 02611  CIG   TXL     *+3,1,18        CHECK FOR (8-N) CHARACTER.          TT)61420
02607 -0602 00 2 02670        ORS     CARDIM+3,2      YES, OR IN 8.                       TT)61430
02610  1 77760 1 02611        TXI     *+1,1,-16       REMOVE 8.                           TT)61440
```

```
02611 -0602 00 3 02710         ORS     CARDIM+19,3   OR IN DIGIT.                          TT)61450
                                                                                           TT)61460
02612  0771 00 0 00001   RLOOP ARS     1             SHIFT COLUMN MARKER.                  TT)61470
02613  2 00001 4 02564         TIX     PRLP,4,1      COUNT CHARACTERS PER WORD.            TT)61480
02614  0774 00 4 00000   WDCNT AXT     **,4          THIS WORD DONE, RESTORE WORD COUNT.   TT)61490
02615 -2 00001 4 02625         TNX     PNOW,4,1      INDEX, AND EXIT WHEN COUNT EXHAUSTED. TT)61500
02616  0560 00 4 00000   LDQ   LDQ     **,4          PICKUP NEXT WORD.                     TT)61510
02617  0634 00 4 02614         SXA     WDCNT,4       SAVE WORD COUNT.                      TT)61520
02620  0774 00 4 00006         AXT     6,4           RESTORE CHARACTER COUNT.              TT)61530
02621 -0100 00 0 02564         TNZ     PRLP          RETURN IF MORE TO THIS HALF.          TT)61540
02622 -3 00000 2 02625         TXL     PNOW,2,0      IF RIGHT HALF DONE, GO PRINT.         TT)61550
02623 -0500 00 0 03153         CAL     =0400000000000  SET COLUMN MARKER.                  TT)61560
02624  1 77777 2 02564         TXI     PRLP,2,-1     SET IR2 FOR RIGHT HALF AND RETURN.    TT)61570
                                                                                           TT)61580
02625  0774 00 1 00030   PNOW  AXT     24,1          MOVE CARDIM TO BUFF,                  TT)61590
02626  0060 00 0 02626         TCOA    *             WHEN DSCA IS FREE.                    TT)61600
02627 -0500 00 1 02715         CAL     CARDIM+24,1   ..                                  TT)61610
02630  0602 00 1 02745         SLW     BUFF+24,1     ..                                  TT)61620
02631  2 00001 1 02627         TIX     *-2,1,1       ..                                  TT)61630
                                                                                           TT)61640
02632  0766 00 0 01361         WPRA                  SELECT PRINTER,                       TT)61650
02633  0000 60 0 02633   SPRA  HTR*    *             SPACE CARRIAGE.                       TT)61660
02634  0540 00 0 02664         RCHA    PRCOM         AND LOAD WITH THIS LINE.              TT)61670
02635 -3 00001 4 02642         TXL     PR4,4,1       C(IR4) .LE. 1 IFF WDCNT EXHAUSTED.    TT)61680
02636 -0500 00 0 02663         CAL     NOSPC         MORE TO DO, SET SPR.                  TT)61690
02637  0602 00 0 02633         SLW     SPRA          ..                                    TT)61700
02640 -0500 00 0 03153         CAL     =0400000000000  GET COLUMN MARKER.                  TT)61710
02641  0020 00 0 02557         TRA     HRI           GO BACK AND DO IT AGAIN.              TT)61720
                                                                                           TT)61730
02642  0774 00 4 00000   PR4   AXT     **,4          RESTORE IRS.                          TT)61740
02643  0774 00 2 00000   PR2   AXT     **,2          ..                                    TT)61750
02644  0774 00 1 00000   PR1   AXT     **,1          ..                                    TT)61760
02645  0500 00 4 00001         CLA     1,4           GET CONTROL WORD AGAIN.               TT)61770
02646  0120 00 4 00002         TPL     2,4           IF PLUS, NORMAL EXIT.                 TT)61780
                                                                                           TT)61790
02647  0766 00 0 01361         WPRA                  STOP COMMAND, SPACE PRINTER.          TT)61800
02650  0760 00 0 01364         SPRA    4             ..                                    TT)61810
02651  0760 00 0 01363         SPRA    3             ..                                    TT)61820
02652  0502 00 0 03125         CLS     =0            SET S BIT TO ONE.                     TT)61830
02653  0760 00 0 00006         COM                   SET AC TO ALL ONES.                   TT)61840
02654  0560 00 0 03166         LDQ     =0777777777777  LIGHT UP MQ.                        TT)61850
02655  0420 00 7 77777         HPR     -1,7          STOP.....                             TT)61860
02656  0020 00 4 00002         TRA     2,4           AND RETURN...                         TT)61870
                                                                                           TT)61880
                                       STORAGE AND CONSTANTS FOR PRINT.                    TT)61890
                                                                                           TT)61900
02657  0 00000 0 00000   PRCOL PZE                   COLUMN MARKER STORAGE.                TT)61910
02660  0760 00 0 01364   SPRS  SPRA    4             CARRIAGE CONTROL,   0                 TT)61920
02661  0760 00 0 01361         SPRA    1             MAINTAIN THIS       1                 TT)61930
02662  0761 00 0 00000         NOP                   ORDER.              2                 TT)61940
02663  0760 00 0 01371   NOSPC SPRA    9             RIGHT HALF SPRA.                      TT)61950
02664  0 00030 0 02715   PRCOM IOCD    BUFF,0,24     PRINT COMMAND.                        TT)61960
                                                                                           TT)61970
               02520    NPRINT SYN     PRINT         EXTERNAL NAME FOR PRINT.              TT)61980
```

```
                        SPACE   2                                                       TT)61990
                                STORAGE AREA.                                           TT)62000
                                                                                        TT)62010
02665                 CARDIM BSS    24          CARD IMAGE BUFFER FOR PRINT.          TT)62020
02715                  BUFF  BSS    24          OUTPUT BUFFER FOR WOT.                TT)62030
02745                  OUT   BSS    22          OUTPUT BUFFER FOR POST MORTEM.        TT)62040
                                                                                        TT)62050
              00036     NSYM  EQU    30          30 NAMES ALLOWED IN MOVIE).           TT)62060
02773                  NAME  BSS    NSYM        SUBROUTINE NAME TABLE.                TT)62070
03031                ORIGIN BSS    NSYM        SUBROUTINE ORIGIN TABLE.              TT)62080
03067                 ENTRY BSS    NSYM        SUBROUTINE ENTRY POINT TABLE.         TT)62090
                                                                                        TT)62100
                        END                                                             TT)62110
```

LITERALS

```
03125  000000000000
03126  000000000001
03127  000000000002
03130  000000000003
03131  000000000005
03132  000000000006
03133  000000000012
03134  000000000072
03135  000000000073
03136  000000000454
03137  000000602646
03140  000000777777
03141  000017000000
03142  004623632143
03143  006060606060
03144  007070707600
03145  014623632143
03146  016060606060
03147  046060606060
03150  070707070777
03151  200000000000
03152  377700000000
03153  400000000000
03154  462660000000
03155  600073007300
03156  606000000060
03157  606060604000
03160  606060606060
03161  632562636260
03162  700000000000
03163  744421314534
03164  777400777777
03165  777700000000
03166  777777777777
```

TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
POST PROCESSOR ASSEMBLY DATA

3167 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

2223	T	136, 175, 204, 205, 704, 705, 710, 1147, 1150, 2524, 2525
1170	AC	1157, 1535
1171	MQ	1156, 1541
1653	ADR	1737
2132	AR1	2050
2131	AR2	2047, 2124
2	CAP	215
2606	DIG	2572
2020	DX1	1642
2017	DX2	1641
2016	DX4	1640
2452	ETA	2412
1776	FIN	1746
2557	HRI	2641
2023	ILC	1645, 1653, 1734, 1736, 1772, 1774
1164	IR1	1154, 1551
1165	IR2	1155, 1554
1166	IR4	220, 263, 315, 426, 432, 525, 534, 630, 1145, 1557
2616	LDQ	2542, 2544
52	MAP	
240	NUM	2062, 2406, 2407
1260	OCT	1301, 1536, 1542, 1546, 1715, 1743
234	ORG	214, 1464
2745	OUT	1656, 1741, 1744, 1745, 1757, 1761, 1762, 1771, 2000, 2001, 2002, 2013, 2015
1503	PMH	1424, 1427, 1441, 1443, 1461
1401	PMR	712, 713
1750	PPM	
2644	PR1	2522
2643	PR2	2521
2642	PR4	2520, 2635
2134	PRE	2055, 2056
1740	REG	1652, 1661, 1674
1	RIP	201
533	SLR	173
473	SSI	406, 416, 433
1145	SX4	
235	TIT	202, 707
425	TR8	405
250	WEP	226, 230, 232
2407	WOT	67, 71, 121, 131, 134, 175, 177, 231, 276, 344, 361, 440, 445, 451, 467, 565, 600 645, 714, 745, 747, 753, 755, 1401, 1442, 1462, 1470, 1562, 1564, 1566, 1570, 1572, 1574, 1576 1600, 1730, 1732, 1760, 2014, 2456, 2460
702	BACK	233, 301, 347, 364, 472, 570, 603, 650
2715	BUFF	2427, 2450, 2630, 2664
1445	COD1	1414, 1417
167	C(1)	27
170	C(2)	31
171	C(7)	33
172	C(8)	35
1523	DCON	1411
765	DOOR	722, 743, 1001, 1023, 2473
1026	EPMR	300, 346, 363, 471, 567, 602, 647, 711

```
 213   ESTM   206
 175   EVAL   137,  140
 131   EXCM   127
 236   EXEC   216,  222
 732   FILE   724
 431   FPER   411
 404   FPTR   172
 207   ITIM   203
1024   LOAD  1006
 556   LOOP
2136   LSTA  2123
 663   LSTM   706,  770, 1152
1447   LSUB   133
 102   MORG    60,   62,   64,   75,  100
2773   NAME   111,  123, 1420
2025   NREP  1675, 1676, 1735
  36   NSYM    66,  103,  105,  111,  113,  116,  123, 1412, 1413, 1415, 1420, 2773, 3031, 3067, 3125
1260   OCT)  1301
   4   PCT1  1410
1510   PMBP  1463
2625   PNCW  2615, 2622
2564   PRLP  2613, 2621, 2624
2541   PTXI  2530
 261   QUIT   672,  705, 1150, 2525
 610   SADR   332,  531,  540,  541,  544,  546,  547,  556,  571
 361   SEQR   266,  320,  430,  530,  537
1167   SIND  1162, 1545
2406   SIZE  2060
1412   SLPM  1447
 541   SLR1   333,  532
 453   SPER   442,  450
2633   SPRA  2554, 2637
2660   SPRS  2553
  73   SRCH    55,   56,  101,  130
 723   STOP
 314   STRR   170
1172   SVPQ  1161, 1524
2146   TABL  2063, 2125, 2407
 241   TCAP   176
 237   TIME   205
 524   TIMR   171
 133   TMAP   126
 477   WACO   446
 503   WMQO   452
2027   WREP  1710, 1713, 1716, 1717, 1727, 1731
2442   WTX4  2407
 510   XFPT   460,  466,  470
 350   XSTR   341,  343,  345
2572   ZONE
 742  CLOSE   733
1256  COMRT  1233, 1241
1250  COMSW  1234, 1235, 1253
 173  C(13)    37
 174  C(ST)    41
1602  DCNX4  1523
```

```
2026  DNREP   1677, 1704, 1706
1002  DOOR1    776, 1020
1517  ENDPM   1471
1027  ENDRN    715
1077  ENDTP    746, 2457
2406  ENTBL   2407
3067  ENTRY    105,  116
 714  FINIS   1473
 627  IOTMR    174
1660  LOOP1   1650, 1651, 1747
1657  LOOP2   1775
2116  LOWER   2061, 2070
2024  LWORD   1662, 1711, 1714
2663  NOSPC   2636
2135  OPBIN   2054, 2065, 2117
2125  OPFND   2053, 2067
2022  PMCNT   1431, 1440, 1451, 1460, 1721, 1726, 1751, 1756, 2005, 2012
   3  PPROG   1407
2657  PRCOL   2564, 2571
2664  PRCOM   2634
2520  PRINT    725,  735,  763,  777, 1021, 2444, 2447, 2470, 2665
1474  PROER   1402
 571  PSTOP    543,  554
2115  RAISE   2066
2612  RLOOP   2574, 2605
 673  RLSTM    701
1113  SEXIT   1000
 677  SLSTM    663,  666,  673,  675
2062  SRCH1   2071, 2072, 2073, 2074, 2075, 2076, 2077, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2110, 2111
              2112
2066  SRCH2   2062
1146  SVCON    221,  264,  316,  427,  434,  526,  535,  631
 701  TLSTM    667
 262  TRAPR    167
2117  TYPEA   2057, 2113, 2114
2605  TZERO   2577, 2602
2614  WDCNT   2540, 2543, 2617
 474  WDVER    441
2435  WTCAL   2433
2450  WTCOM   2426, 2431, 2434, 2441
2436  WTSLW   2430
2451  WTTAG   2414, 2443
2422  WTTIX   2417
2432  WTTXI   2416
1045  XFILE    736
 651  XIOBD    642,  644,  646
 611  XLOOP    562,  564,  566
1126  XRCAL   1022
 365  XSEQR    362
1031  XSTOP    726
 302  XTRAP    273,  275,  277
1301  BCDTAB  1301, 1302, 1303, 1304, 1305, 1306, 1307, 1310, 1311, 1312, 1313, 1314, 1315, 1316, 1317, 1320, 1321
              1322, 1323, 1324, 1325, 1326, 1327, 1330, 1331, 1332, 1333, 1334, 1335, 1336, 1337, 1340, 1341, 1342
              1343, 1344, 1345, 1346, 1347, 1350, 1351, 1352, 1353, 1354, 1355, 1356, 1357, 1360, 1361, 1362, 1363
              1364, 1365, 1366, 1367, 1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1400
```

TESTS) FOR CAP, MONITOR FOR CLASS ASSEMBLY PROGRAM.
POST PROCESSOR ASSEMBLY DATA

```
2665 CARDIM  2560, 2561, 2576, 2601, 2603, 2607, 2611, 2627
1233 COMADR
1604 HEAD01  1563
1610 HEAD02  1534, 1565
1613 HEAD03  1537, 1540, 1567
1617 HEAD04  1543, 1544, 1571
1623 HEAD05  1547, 1550, 1573
1627 HEAD06  1553, 1575
1632 HEAD07  1556, 1577
1635 HEAD08  1561, 1601
   0 MOVIE)    52,   54
2520 NPRINT  2665
1173 OCTADR   114,  117,  223,  270,  336,  464,  557,  572,  637, 1552, 1555, 1560, 1654
1640 OCTDMP  1444, 1466
3031 ORIGIN   103,  113, 1413, 1415
 604 PSTOPS   552
1102 RENDTP   754
1207 TABBLK  1203, 1207, 1210, 1211, 1212, 1213, 1214, 1215, 1216, 1217, 1220, 1221, 1222, 1223, 1224, 1225, 1226
             1227, 1230, 1231, 1232, 1707, 1767
   7 TESTS)
2476 WEOTA3  2471
1061 XCLOSE   764
 141 XMAP01    70
 147 XMAP02    72
 153 XMAP03   112,  115,  120,  122
 157 XMAP04   132
 163 XMAP05   135
 621 XPSTOP   575,  577,  601
2047 (OPCD)  1712, 1740
```

NO ERROR IN ABOVE ASSEMBLY.
*TIME SPENT IN FAP.. 000054 IN HUNDREDTHS OF MINUTES.

```
BUFFERED INPUT/OUTPUT PROGRAMS FOR CAP.

                        PCC                                                                 RIP00010
                        COUNT   278                                                         RIP00020
                        LBL     RIP             BINARY CARD LABEL.                          RIP00030
 00012                  ENTRY   RIP             SETUP READ OPERATIONS.                      RIP00050
 00071                  ENTRY   WCT1            WRITE-COLLATION-TAPE-1.                     RIP00060
 00147                  ENTRY   REWIND          REWIND COLLATION TAPE.                      RIP00070
 00157                  ENTRY   READ2           READ COLLATION TAPE.                        RIP00080
 00243                  ENTRY   PRINT           BUFFERRED PRINT PROGRAM.                    RIP00090
 00277                  ENTRY   PPROG           DUMP SYMBOLIC PROGRAM.                      RIP00100
 00347                  ENTRY   PCT1            DUMP COLLATION TAPE.                        RIP00110
 00027                  ENTRY   READ1           READ INPUT TAPE.                            RIP00120
                                                                                            RIP00130
                                RIP SETS UP READ1 AND PPROG.                                RIP00140
                                                                                            RIP00150

  TRANSFER VECTOR
00000   475146276060    PROG
00001   626704606060    SX4
00002   626523464560    SVCON
00003   664663606060    WOT
00004   254744516060    EPMR
00005   222123426060    BACK
00006   436263446060    LSTM
00007   222324632122    BCDTAB

  LINKAGE DIRECTOR
00010   000000000000
00011   513147606060

00012   0634 00 4 00122 RIP     SXA     RX4,4           SAVE IR4.                           RIP00160
00013  -0500 60 0 00000         CAL*    $PROG           GET CONTROL WORD.                   RIP00170
00014  -0734 00 4 00000         PDX     0,4             COUNT TO IR4.                       RIP00180
00015  -3 04064 4 00017         TXL     *+2,4,LPROG     MAXIMUM NUMBER OF CARDS IN PROG.    RIP00190
00016   0774 00 4 04064         AXT     LPROG,4         ..                                  RIP00200
00017   0754 00 4 00000         PXA     0,4             COUNT TO A(AC).                     RIP00210
00020   0361 60 0 00000         ACL*    $PROG           GET END ADDRESS.                    RIP00220
00021   0621 00 0 00040         STA     PEND            SAVE IN READ1.                      RIP00230
00022   0621 00 0 00312         STA     PPEND           SAVE IN PPROG.                      RIP00240
00023   0634 00 4 00037         SXA     PLTH,4          SAVE IN READ1.                      RIP00250
00024   0634 00 4 00310         SXA     PPLTH,4         SAVE IN PPROG.                      RIP00260
00025   0534 00 4 00122         LXA     RX4,4           RESTORE IR4.                        RIP00270
00026   0020 00 4 00001         TRA     1,4             RETURN.                             RIP00280


                        SPACE   2                                                           RIP00290
                                READ1 TAKES NEXT 14 WORDS OF PROG AND STORES IN BUFF.       RIP00300
                                                                                            RIP00310
00027   0520 00 0 00057 READ1   ZET     TECFI           CHECK FOR END OF PROGRAM.           RIP00320
00030   0020 00 0 00051         TRA     EOFI            END REACHED, COMMENT.               RIP00330
00031   0634 00 4 00122         SXA     RX4,4           SAVE IRS.                           RIP00340
00032   0634 00 2 00123         SXA     RX2,2           ..                                  RIP00350
00033  -0500 00 4 00001         CAL     1,4             GET CONTROL WORD.                   RIP00360
00034   0361 00 0 10611         ACL     =14             COMPUTE BUFFER+14.                  RIP00370
00035   0621 00 0 00041         STA     SLW1            INSERT.                             RIP00380
00036   0774 00 4 00016         AXT     14,4            SET WORD COUNT.                     RIP00390
00037   0774 00 2 00000 PLTH    AXT     **,2            SET PROGRAM INDEX.                  RIP00400
```

BUFFERED INPUT/OUTPUT PROGRAMS FOR CAP.

```
00040 -0500 00 2 00000   PEND  CAL     **,2          COPY CARD INTO BUFFER.                          RIP00410
00041  0602 00 4 00000   SLW1  SLW     **,4          ..                                              RIP00420
00042  2 00001 2 00044         TIX     *+2,2,1       IF END OF PROG, SET TEOFI.                      RIP00430
00043 -0625 00 0 00057         STL     TECFI         ..                                              RIP00440
00044  2 00001 4 00040         TIX     PEND,4,1      COUNT WORDS.                                    RIP00450
00045  0634 00 2 00037         SXA     PLTH,2        SAVE PROGRAM INDEX.                             RIP00460
00046  0534 00 4 00122         LXA     RX4,4         RESTORE IRS.                                    RIP00470
00047  0534 00 2 00123         LXA     RX2,2         ..                                              RIP00480
00050  0020 00 4 00002         TRA     2,4           RETURN.                                         RIP00490
                                                                                                     RIP00500
00051  0522 60 0 00001   EOFI  XEC*    $SX4          SAVE IR4.                                       RIP00510
00052  0074 00 4 00002         TSX     $SVCON,4      SAVE CONSOLE.                                   RIP00520
00053  0074 00 4 00003         TSX     $WOT,4        COMMENT, EOF ON INPUT.                          RIP00530
00054  0 00011 0 00060         PZE     WEOFI,0,9     ..                                              RIP00540
00055 -0625 60 0 00004         STL*    $EPMR         SET ERROR INDICATOR.                            RIP00550
00056  0020 60 0 00005         TRA*    $BACK         EXIT TO POST MORTEM.                            RIP00560
                                                                                                     RIP00570
00057  0 00000 0 00000   TEOFI PZE                   END OF FILE MARK ON INPUT TAPE.                 RIP00580
00060  002545246046      WEOFI BCI     9,0END OF FILE REACHED WHILE READING CAP INPUT TAPE.            RIP00590
00061  266026314325
00062  605125212330
00063  252460663031
00064  432560512521
00065  243145276023
00066  214760314547
00067  646360632147
00070  253360606060
```

```
                         SPACE   2                                                                   RIP00600
                                 WCT1 TAKES 14 WORDS AND WRITES THEM ON COLLATION                     RIP00610
                                 TAPE BUFFER. IF TAG OF CONTROL WORD IS NON-ZERO,                     RIP00620
                                 THEN N (IN DECREMENT OF CONTROL WORD) WORDS                          RIP00630
                                 OF BUFFER AND 14-N BLANK WORDS GO TO COLLATION                       RIP00640
                                 TAPE BUFFER.                                                        RIP00650
                                                                                                     RIP00660
00071  0520 00 0 00135   WCT1  ZET     TECTC         CHECK FOR EOT ON CT1.                           RIP00670
00072  0020 00 0 00126         TRA     EOTC          EOT REACHED, COMMENT.                           RIP00680
00073  0634 00 4 00122         SXA     RX4,4         SAVE IRS.                                       RIP00690
00074  0634 00 2 00123         SXA     RX2,2         ..                                              RIP00700
00075  0634 00 1 00124         SXA     RX1,1         ..                                              RIP00710
00076 -0500 00 4 00001         CAL     1,4           GET CONTROL WORD.                               RIP00720
00077  0625 00 0 00134         STT     TAG           SAVE TAG.                                       RIP00730
00100  0771 00 0 00022         ARS     18            D(AC) TO A(AC).                                 RIP00740
00101 -0520 00 0 00134         NZT     TAG           IF TAG=0, A(AC)=14.                             RIP00750
00102 -0500 00 0 10611         CAL     =14           COUNT IS 14.                                    RIP00760
00103  0734 00 2 00000         PAX     0,2           COUNT TO IR2.                                   RIP00770
00104  1 00001 2 00105         TXI     *+1,2,1       RAISE COUNT FOR TEST.                           RIP00780
00105  0754 00 2 00000         PXA     0,2           PLACE IN A(AC).                                 RIP00790
00106  0361 00 4 00001         ACL     1,4           COMPUTE, BUFFER+COUNT+1.                        RIP00800
00107  0621 00 0 00112         STA     WCAL          INSERT.                                         RIP00810
00110  0774 00 4 00016         AXT     14,4          SET RECORD COUNT.                               RIP00820
00111  0774 00 1 10150   WLCT  AXT     LCT1,1        MAXIMUM NUMBER OF RECORDS ON CT1.               RIP00830
00112 -0500 00 2 00000   WCAL  CAL     **,2          MOVE BUFFER TO CT1.                             RIP00840
00113  2 00001 2 00115         TIX     *+2,2,1       COUNT BUFFER.                                   RIP00850
00114 -0500 00 0 10614         CAL     =H            BLANKS IF BUFFER DONE.                          RIP00860
```

```
00115  0602 00 1 10571          SLW     CT1,1           WCT.                                                    RIP00870
00116  2 00001 1 00120          TIX     *+2,1,1         COUNT WORDS CT1.                                        RIP00880
00117 -0625 00 0 00135          STL     TEOTC           END OF CT1, SET TEOTC.                                  RIP00890
00120  2 00001 4 00112          TIX     WCAL,4,1        COUNT WORDS IN RECORD.                                  RIP00900
00121  0634 00 1 00111          SXA     WLCT,1          SAVE CT1 INDEX.                                         RIP00910
00122  0774 00 4 00000  RX4     AXT     **,4            RESTORE IRS.                                            RIP00920
00123  0774 00 2 00000  RX2     AXT     **,2            ..                                                      RIP00930
00124  0774 00 1 00000  RX1     AXT     **,1            ..                                                      RIP00940
00125  0020 00 4 00002          TRA     2,4             RETURN.                                                 RIP00950
                                                                                                               RIP00960
00126  0522 60 0 00001  EOTC    XEC*    $SX4            SAVE IR4.                                               RIP00970
00127  0074 00 4 00002          TSX     $SVCON,4        SAVE CONSOLE.                                           RIP00980
00130  0074 00 4 00003          TSX     $WOT,4          COMMENT, EOT CT1.                                       RIP00990
00131  0 00011 0 00136          PZE     WEOTC,0,9       ..                                                      RIP01000
00132 -0625 60 0 00004          STL*    $EPMR           SET ERROR INDICATOR.                                    RIP01010
00133  0020 60 0 00005          TRA*    $BACK           EXIT TO POST MORTEM.                                    RIP01020
                                                                                                               RIP01030
00134  0 00000 0 00000  TAG     PZE                     CONTROL WORD FOR WCT1.                                  RIP01040
00135  0 00000 0 00000  TEOTC   PZE                     END OF TAPE MARK ON COLLATION TAPE.                     RIP01050
00136  002545246046     WEOTC   BCI     9,0END OF TAPE REACHED WHILE WRITING CAP COLLATION TAPE.                RIP01060
00137  266063214725
00140  605125212330
00141  252460663031
00142  432560665131
00143  633145276023
00144  214760234643
00145  432163314645
00146  606321472533


                                SPACE   2                                                                       RIP01070
                                        REWIND SETS UP READ2 AND PCT1. READ2 AND                                RIP01080
                                        PCT1 CHECK THAT REWIND HAS BEEN CALLED.                                 RIP01090
                                                                                                               RIP01100
00147 -0625 00 0 00156  REWIND  STL     TREW            CT1 REWOUND.                                            RIP01110
00150 -0500 00 0 00111          CAL     WLCT            GET CT1 INDEX IN A(AC).                                 RIP01120
00151  0767 00 0 00022          ALS     18              PLACE IN D(AC).                                         RIP01130
00152  0622 00 0 00175          STD     TLCT            SAVE IN READ2.                                          RIP01140
00153  0622 00 0 00403          STD     PTLCT           SAVE IN PCT1.                                           RIP01150
00154  0140 00 0 00155          TOV     *+1             TURN OFF ACOVL.                                         RIP01160
00155  0020 00 4 00001          TRA     1,4             RETURN.                                                 RIP01170
                                                                                                               RIP01180
00156  0 00000 0 00000  TREW    PZE                     BEGINNING OF TAPE MARK ON COLLATION TAPE.               RIP01190


                                SPACE   2                                                                       RIP01200
                                        READ2 READS ONE RECORD OF THE COLLATION                                 RIP01210
                                        TAPE BUFFER INTO BUFFER OF CONTROL WORD.                                RIP01220
                                                                                                               RIP01230
00157  0634 00 4 00122  READ2   SXA     RX4,4           SAVE IRS.                                               RIP01240
00160  0634 00 2 00123          SXA     RX2,2           ..                                                      RIP01250
00161 -0520 00 0 00156          NZT     TREW            TEST FOR REWIND.                                        RIP01260
00162  0020 00 0 00204          TRA     NREW            NOT REWOUND, COMMENT.                                   RIP01270
00163  0520 00 0 00231          ZET     TEOFC           CHECK FOR EOF ON CT1.                                   RIP01280
00164  0020 00 0 00223          TRA     EOFC            EOF REACHED, COMMENT.                                   RIP01290
00165  0774 00 2 10150  RLCT    AXT     LCT1,2          MAXIMUM NUMBER OF CARDS ON CT1.                         RIP01300
```

```
00166 -C500 00 4 00001          CAL     1,4             GET CONTROL WORD.                       RIP01310
C0167  0361 00 0 10611          ACL     =14             COMPUTE BUFFER+14.                      RIP01320
C0170  0621 00 0 00173          STA     *+3             INSERT.                                 RIP01330
00171  0774 00 4 00016          AXT     14,4            SET WORD COUNT.                         RIP01340
00172 -0500 00 2 10571          CAL     CT1,2           MOVE TO BUFFER.                         RIP01350
C0173  0602 00 4 00000          SLW     **,4            ..                                      RIP01360
00174  1 77777 2 00175          TXI     *+1,2,-1        COUNT CT1.                              RIP01370
00175  3 00000 2 00177   TLCT   TXH     *+2,2,**        CHECK FOR FILE MARK.                    RIP01380
00176 -0625 00 0 00231          STL     TEOFC           FILE MARK.                              RIP01390
00177  2 00001 4 00172          TIX     *-5,4,1         COUNT WORDS.                            RIP01400
00200  0634 00 2 00165          SXA     RLCT,2          SAVE CT1 INDEX.                         RIP01410
00201  0534 00 4 00122          LXA     RX4,4           RESTORE IRS.                            RIP01420
00202  0534 00 2 00123          LXA     RX2,2           ..                                      RIP01430
C0203  0020 00 4 00002          TRA     2,4             RETURN.                                 RIP01440
                                                                                                RIP01450
00204  0074 00 4 00006   NREW   TSX     $LSTM,4         CT1 NOT REWOUND, RESET TRAP.            RIP01460
00205  0074 00 4 00003          TSX     $WOT,4          COMMENT.                                RIP01470
C0206  0 00011 0 00212          PZE     WNREW,0,9       ..                                      RIP01480
00207 -0760 00 0 00005          ESTM                    RE-ENTER TRAP.                          RIP01490
00210  0074 00 4 00147          TSX     REWIND,4        REWIND CT1.                             RIP01500
00211  0020 00 0 00165          TRA     RLCT            RETURN.                                 RIP01510
                                                                                                RIP01520
00212  005125212402     WNREW BCI     9,CREAD2 CALLED BEFORE REWIND, COLLATION TAPE REWOUND.  RIP01530
C0213  602321434325
00214  246022252646
00215  512560512566
C0216  314524736023
00217  464343216331
00220  464560632147
00221  256051256646
00222  644524336060                                                                           RIP01540
                                                                                               RIP01550
00223  0522 60 0 00001   EOFC   XEC*    $SX4            SAVE IR4.                               RIP01560
00224  0074 00 4 00002          TSX     $SVCON,4        SAVE CONSOLE.                           RIP01570
C0225  0074 00 4 00003          TSX     $WOT,4          COMMENT, EOF ON CT1.                    RIP01580
00226  0 00011 0 00232          PZE     WEOFC,0,9       ..                                      RIP01590
00227 -0625 60 0 00004          STL*    $EPMR           SET ERROR INDICATOR.                    RIP01600
C0230  0020 60 0 00005          TRA*    $BACK           EXIT TO POST MORTEM.                    RIP01610
                                                                                               RIP01620
00231  0 00000 0 00000   TEOFC PZE                     END OF FILE MARK ON COLLATION TAPE.     RIP01630
00232  002545246046      WEOFC BCI     9,OEND OF FILE REACHED WHILE READING COLLATION TAPE.
C0233  266026314325
00234  605125212330
00235  252460663031
00236  432560512521
C0237  243145276023
C0240  464343216331
00241  464560632147
00242  253360606060

                                SPACE   2                                                      RIP01640
                                        PRINT COUNTS LINES AND CALLS $WOT.                     RIP01650
                                                                                               RIP01660
                                                                                               RIP01670
C0243  0634 00 4 00122   PRINT  SXA     RX4,4           SAVE IR4.                               RIP01680
00244  0500 00 0 00267          CLA     LNCNT           COUNT LINES.
```

```
C0245   0402 00 0 10610          SUB     =1              ..                                  RIP01690
00246   0601 00 0 00267          STO     LNCNT           ..                                  RIP01700
00247  -0120 00 0 00261          TMI     LNEX            CHECK FOR LNCNT EXCEEDED.           RIP01710
00250  -0500 00 4 00001          CAL     1,4             SET CONTROL WORD FOR WOT.           RIP01720
C0251   0621 00 0 00255          STA     CWOT            ..                                  RIP01730
00252   0622 00 0 00255          STD     CWOT            ..                                  RIP01740
00253   0074 00 4 00006          TSX     $LSTM,4         RESET TRAP.                         RIP01750
00254   0074 00 4 00003          TSX     $WOT,4          OK, GO TO WOT.                      RIP01760
00255   0 00000 0 00000   CWOT   PZE                     ..                                  RIP01770
00256  -0760 00 0 00005          ESTM                    RESET TRAP.                         RIP01780
00257   0534 00 4 00122          LXA     RX4,4           RESTORE IR4.                        RIP01790
00260   0020 00 4 00002          TRA     2,4             RETURN.                             RIP01800
                                                                                             RIP01810
00261   0522 60 0 00001   LNEX   XEC*    $SX4            LNCNT EXCEEDED, SAVE IR4.           RIP01820
00262   0074 00 4 00002          TSX     $SVCON,4        SAVE CONSOLE.                       RIP01830
00263   0074 00 4 00003          TSX     $WOT,4          COMMENT.                            RIP01840
00264   0 00007 1 00270          PZE     WLNEX,1,7       ..                                  RIP01850
00265  -0625 60 0 00004          STL*    $EPMR           SET ERROR INDICATOR.                RIP01860
00266   0020 60 0 00005          TRA*    $BACK           EXIT TO POST MORTEM SECTION.        RIP01870
                                                                                             RIP01880
00267   0 00000 0 00454   LNCNT  PZE     300             MAXIMUM LINES OUTPUT.               RIP01890
00270   004751462751      WLNEX  BCI     7,0PROGRAMMER   OUTPUT EXCEEDS 300 RECORDS.        RIP01900
00271   214444255160
00272   466463476463
00273   602567232525
00274   246260030000
00275   605125234651
00276   246233606060

                                 SPACE   2                                                   RIP01910
                                         PPROG PRINTS UP TO 150 CARDS OF $PROG.              RIP01920
                                                                                             RIP01930
00277   0634 00 4 00122   PPROG  SXA     RX4,4           SAVE IRS.                           RIP01940
00300   0634 00 2 00123          SXA     RX2,2           ..                                  RIP01950
00301   0634 00 1 00124          SXA     RX1,1           ..                                  RIP01960
00302   0074 00 4 00003          TSX     $WOT,4          INITIAL COMMENT.                    RIP01970
00303   0 00010 0 00337          PZE     PHEAD,0,8       ..                                  RIP01980
00304   0074 00 4 00003          TSX     $WOT,4          BLANK LINE.                         RIP01990
00305   0 00001 0 10614          PZE     =H      ,0,1    ..                                  RIP02000
00306  -0500 00 0 00007          CAL     $BCDTAB         GET ADDRESS OF TABLE.               RIP02010
00307   0621 00 0 00313          STA     CRQ1            ..                                  RIP02020
00310   0774 00 1 00000   PPLTH  AXT     **,1            PROGRAM WORD COUNT.                 RIP02030
00311   0774 00 2 00016          AXT     14,2            BUFFER WORD COUNT.                  RIP02040
00312   0560 00 1 00000   PPEND  LDQ     **,1            GET CURRENT WORD.                   RIP02050
00313  -0154 06 0 00000   CRQ1   CRQ     **,0,6          DELETE ILLEGAL CHARACTERS.          RIP02060
00314  -0600 00 2 10610          STQ     PBUFF+15,2      PLACE IN BUFFER.                    RIP02070
00315  -2 00001 2 00317          TNX     *+2,2,1         COUNT WORDS IN BUFFER.              RIP02080
00316   1 77777 1 00312          TXI     *-4,1,-1        COUNT WORDS IN PROG.                RIP02090
00317   0560 00 0 10614          LDQ     =H              PICKUP STANDARD CARRIAGE CONTROL.   RIP02100
00320   0500 00 0 00336          CLA     PPCNT           DECREMENT LINE COUNT.               RIP02110
00321   0402 00 0 10610          SUB     =1              ..                                  RIP02120
00322   0120 00 0 00325          TPL     *+3             CHECK FOR OVERFLOW.                 RIP02130
00323   0500 00 0 10612          CLA     =59             OVERFLOW, RESTORE LINE COUNT-1.     RIP02140
00324   0560 00 0 10613          LDQ     =H1             SET TO EJECT.                       RIP02150
00325   0601 00 0 00336          STO     PPCNT           SAVE REMAINING LINE COUNT.          RIP02160
```

```
00326 -0600 00 0 10571          STQ     PBUFF           INSERT CARRIAGE CONTROL.                    RIP02170
00327  0074 00 4 00003          TSX     $WOT,4          PRINT THIS CARD.                            RIP02180
00330  0 00017 0 10571          PZE     PBUFF,0,15      ..                                          RIP02190
00331  2 00001 1 00311          TIX     PPLTH+1,1,1     COUNT RECORDS.                              RIP02200
00332  0534 00 4 00122          LXA     RX4,4           RESTORE IRS.                                RIP02210
00333  0534 00 2 00123          LXA     RX2,2           ..                                          RIP02220
00334  0534 00 1 00124          LXA     RX1,1           ..                                          RIP02230
00335  0020 00 4 00001          TRA     1,4             RETURN.                                     RIP02240
                                                                                                     RIP02250
00336  0 00000 0 00072   PPCNT  PZE     58              60 LINES PER PAGE - 2 FOR HEADING.          RIP02260
00337  016060606060      PHEAD  BCI     2,1                                                         RIP02270
00340  606060606060
00341  604746626360             BCI     6, POST MCRTEM OF SYMBOLIC PROGRAM.                         RIP02280
00342  444651632544
00343  604626606270
00344  442246433123
00345  604751462751
00346  214433606060

                                SPACE   2                                                           RIP02290
                                PCT1 PRINTS UP TO 300 RECORDS OF CT1.                               RIP02300
                                                                                                    RIP02310
                                                                                                    RIP02320
00347  0634 00 4 00122   PCT1   SXA     RX4,4           SAVE IRS.                                   RIP02330
00350  0634 00 2 00123          SXA     RX2,2           ..                                          RIP02340
00351  0634 00 1 00124          SXA     RX1,1           ..                                          RIP02350
00352  0074 00 4 00003          TSX     $WOT,4          INITIAL COMMENT.                            RIP02360
00353  0 00010 0 00411          PZE     CHEAD,0,8       ..                                          RIP02370
00354  0074 00 4 00003          TSX     $WOT,4          BLANK LINE.                                 RIP02380
00355  0 00001 0 10614          PZE     =H      ,C,1    ..                                          RIP02390
00356 -0500 00 0 00007          CAL     $BCDTAB         GET ADDRESS OF TABLE.                       RIP02400
00357  0621 00 0 00365          STA     CRQ2            ..                                          RIP02410
00360 -0520 00 0 00156          NZT     TREW            TEST FOR REWIND.                            RIP02420
00361  0074 00 4 00147          TSX     REWIND,4        NO, REWIND.                                 RIP02430
00362  0774 00 1 10150          AXT     LCT1,1          SET MAXIMUM RECORD COUNT, COLLATION TAPE.   RIP02440
00363  0774 00 2 00016   PCTL   AXT     14,2            BUFFER WORD COUNT.                          RIP02450
00364  0560 00 1 10571          LDQ     CT1,1           GET CURRENT WORD.                           RIP02460
00365 -0154 06 0 00000   CRQ2   CRQ     **,0,6          DELETE ILLEGAL CHARACTERS.                  RIP02470
00366 -0600 00 2 10610          STQ     PBUFF+15,2      PLACE IN BUFFER.                            RIP02480
00367  1 77777 1 00370          TXI     *+1,1,-1        COUNT WORDS ON CT1.                         RIP02490
00370  2 00001 2 00364          TIX     *-4,2,1         COUNT WORDS IN BUFFER.                      RIP02500
00371  0560 00 0 10614          LDQ     =H              PICKUP STANDARD CARRIAGE CONTROL.           RIP02510
00372  0500 00 0 00410          CLA     PCCNT           DECREMENT LINE COUNT.                       RIP02520
00373  0402 00 0 10610          SUB     =1              ..                                          RIP02530
00374  0120 00 0 00377          TPL     *+3             CHECK FOR OVERFLOW.                         RIP02540
00375  050C 00 0 10612          CLA     =59             OVERFLOW, RESTORE LINE COUNT-1.             RIP02550
00376  0560 00 0 10613          LDQ     =H1             SET TO EJECT.                               RIP02560
00377  0601 00 0 00410          STO     PCCNT           SAVE REMAINING LINE COUNT.                  RIP02570
00400 -0600 00 0 10571          STQ     PBUFF           INSERT CARRIAGE CONTROL.                    RIP02580
00401  0074 00 4 00003          TSX     $WOT,4          PRINT THIS RECORD.                          RIP02590
00402  0 00017 0 10571          PZE     PBUFF,0,15      ..                                          RIP02600
00403  3 00000 1 00363   PTLCT  TXH     PCTL,1,**       CHECK FOR EOF ON CT1.                       RIP02610
00404  0534 00 4 00122          LXA     RX4,4           RESTORE IRS.                                RIP02620
00405  0534 00 2 00123          LXA     RX2,2           ..                                          RIP02630
00406  0534 00 1 00124          LXA     RX1,1           ..                                          RIP02640
00407  0020 00 4 00001          TRA     1,4             RETURN.
```

```
00410  0 00000 0 00072   PCCNT PZE    58              60 LINES PER PAGE - 2 FOR HEADING.   RIP02650
00411  016060606060      CHEAD BCI    2,1                                                  RIP02660
00412  606060606060                                                                        RIP02670
00413  604746626360             BCI    6, POST MORTEM OF COLLATION TAPE.                   RIP02680
00414  444651632544
00415  604626602346
00416  434321633146
00417  456063214725
00420  336060606060

                 04064   LPROG EQU    14*150          150 CARDS ON INPUT TAPE.             RIP02690
                 10150   LCT1  EQU    14*300          300 RECORDS MAX ON COLLATION TAPE.   RIP02700
                                                                                           RIP02710
                                                                                           RIP02720
00421  512523465124             BCI    9,RECORD NO. 1, COLLATION TAPE, NOTHING HAS BEEN WRITTEN RIP02730
00422  604546336001
00423  736023464343
00424  216331464560
00425  632147257360
00426  454663303145
00427  276030216260
00430  222525456066
00431  513163632545
00432  604645602363             BCI    5, ON CT1.                                           RIP02740
00433  013360606060
00434  606060606060
00435  606060606060
00436  606060606060
10571                   CT1   BES    LCT1-14         COLLATION TAPE BUFFER.               RIP02750
10571                   PBUFF BSS    15              PRINT BUFFER.                        RIP02760
                                                                                           RIP02770
                               END                                                         RIP02780
```

LITERALS

```
10610  000000000001
10611  000000000016
10612  000000000073
10613  016060606060
10614  606060606060
```

BUFFERED INPUT/OUTPUT PROGRAMS FOR CAP.
POST PROCESSOR ASSEMBLY DATA

10615 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

REFERENCES TO DEFINED SYMBOLS

10571	CT1	115, 172, 364
12	RIP	
124	RX1	75, 301, 334, 351, 406
123	RX2	32, 47, 74, 160, 202, 300, 333, 350, 405
122	RX4	12, 25, 31, 46, 73, 157, 201, 243, 257, 277, 332, 347, 404
1	SX4	51, 126, 223, 261
134	TAG	77, 101
3	WOT	53, 130, 205, 225, 254, 263, 302, 304, 327, 352, 354, 401
5	BACK	56, 133, 230, 266
313	CRQ1	307
365	CRQ2	357
255	CWCT	251, 252
223	EOFC	164
51	EOFI	30
126	EOTC	72
4	EPMR	55, 132, 227, 265
10150	LCT1	111, 165, 362, 421, 10571
261	LNEX	247
6	LSTM	204, 253
204	NREW	162
347	PCT1	
363	PCTL	403
40	PEND	21, 44
37	PLTH	23, 45
0	PROG	13, 20
165	RLCT	200, 211
41	SLW1	35
175	TLCT	152
156	TREW	147, 161, 360
112	WCAL	107, 120
71	WCT1	
111	WLCT	121, 150
411	CHEAD	353
267	LNCNT	244, 246
4064	LPROG	15, 16, 421
10571	PBUFF	314, 326, 330, 366, 400, 402
410	PCCNT	372, 377
337	PHEAD	303
336	PPCNT	320, 325
312	PPEND	22
310	PPLTH	24, 331
277	PPROG	
243	PRINT	
403	PTLCT	153
27	READ1	
157	READ2	
2	SVCCN	52, 127, 224, 262
231	TEOFC	163, 176
57	TEOFI	27, 43
135	TEOTC	71, 117
232	WEOFC	226
60	WEOFI	54

```
                BUFFERED INPUT/OUTPUT PROGRAMS FOR CAP.
                   POST PROCESSOR ASSEMBLY DATA

      136  WEOTC    131
      270  WLNEX    264
      212  WNREW    206
        7  BCDTAB   306,  356
      147  REWIND   210,  361


NO ERROR IN ABOVE ASSEMBLY.
    *TIME SPENT IN FAP..     000012 IN HUNDREDTHS OF MINUTES.
```

```
      SYMBOLIC PROGRAM TO TEST CAP.

                                 PCC                                                                  PROG0010
                                 COUNT   150                                                          PROG0020
                                 LBL     PROG          BINARY CARD LABEL.                             PROG0030
                      00002      ENTRY   PROG                                                         PROG0050

  LINKAGE DIRECTOR
00000  000000000000
00001  475146276060

00002  0 01724 0 00003   PROG  PZE     *+1,0,LTH     CONTROL WORD.  START,,LENGTH.                    PROG0060
                               TITLE                 DELETE GENERATED OCTAL FROM LISTING.           PROG0070
00003  232147606060            BCI     9,CAP    REM  THE FOLLOWING ARE ALL LEGAL CAP INSTRUCTIO     PROG0080
00014  456233606060            BCI     5,NS.          TSTCAP00                                       PROG0090
00021  606060606060            BCI     9,       REM  PROGRAM TO COUNT BITS IN AC.                   PROG0100
00032  606060606060            BCI     5,             TSTCAP01                                       PROG0110
00037  234664456360            BCI     9,COUNT  LDQ  ZERO          ZERO TEST CELLS                  PROG0120
00050  606060606060            BCI     5,             TSTCAP02                                       PROG0130
00055  606060606060            BCI     9,       STQ  BITS          ..                               PROG0140
00066  606060606060            BCI     5,             TSTCAP03                                       PROG0150
00073  606060606060            BCI     9,       LXA  THSX          COUNT 36 BITS.                   PROG0160
00104  606060606060            BCI     5,             TSTCAP04                                       PROG0170
00111  434646476060            BCI     9,LOOP   LBT                BIT OR NO.                       PROG0180
00122  606060606060            BCI     5,             TSTCAP05                                       PROG0190
00127  606060606060            BCI     9,       TRA  NO            NO BIT.                          PROG0200
00140  606060606060            BCI     5,             TSTCAP06                                       PROG0210
00145  702562606060            BCI     9,YES    SLW  WORD          BIT, SAVE AC,                    PROG0220
00156  606060606060            BCI     5,             TSTCAP07                                       PROG0230
00163  606060606060            BCI     9,       CAL  BITS          AND INCREMENT COUNT.             PROG0240
00174  606060606060            BCI     5,             TSTCAP08                                       PROG0250
00201  606060606060            BCI     9,       ACL  ONE           ..                               PROG0260
00212  606060606060            BCI     5,             TSTCAP09                                       PROG0270
00217  606060606060            BCI     9,       SLW  BITS          ..                               PROG0280
00230  606060606060            BCI     5,             TSTCAP10                                       PROG0290
00235  606060606060            BCI     9,       CAL  WORD          RESTORE AC.                      PROG0300
00246  606060606060            BCI     5,             TSTCAP11                                       PROG0310
00253  606060606060            BCI     9,       LGR  1             NEXT BIT.                        PROG0320
00264  606060606060            BCI     5,             TSTCAP12                                       PROG0330
00271  454660606060            BCI     9,NO     TIX  LOOP          INDEX.                           PROG0340
00302  606060606060            BC1     5,             TSTCAP13                                       PROG0350
00307  606060606060            BCI     9,       CAL  BITS          GET COUNT.                       PROG0360
00320  606060606060            BCI     5,             TSTCAP14                                       PROG0370
00325  244645256060            BCI     9,DONE   OCTL 002100070000  STOP WITH TRANSFER TO 70000     PROG0380
00336  604623632143            BCI     5, OCTAL.      TSTCAP15                                       PROG0390
00343  606060606060            BCI     9,       REM  STORAGE.                                       PROG0400
00354  606060606060            BCI     5,             TSTCAP16                                       PROG0410
00361  712551466060            BCI     9,ZERO   OCTL 000000000000  TRUE ZERO.                      PROG0420
00372  606060606060            BCI     5,             TSTCAP17                                       PROG0430
00377  464525606060            BCI     9,ONE    INT  1             INCREMENT OF ONE.                PROG0440
00410  606060606060            BCI     5,             TSTCAP18                                       PROG0450
00415  633062676060            BCI     9,THSX   LAS  36            ADDRESS IS 36.                   PROG0460
00426  606060606060            BCI     5,             TSTCAP19                                       PROG0470
00433  606060606060            BCI     9,       REM  DATA.                                          PROG0480
00444  606060606060            BCI     5,             TSTCAP20                                       PROG0490
00451  223163626060            BCI     9,BITS   INT  0             STORAGE FOR BIT COUNT.           PROG0500
00462  606060606060            BCI     5,             TSTCAP21                                       PROG0510
00467  664651246060            BCI     9,WORD   INT  0             TEMPORARY STORAGE FOR AC.        PROG0520
```

```
        SYMBOLIC PROGRAM TO TEST CAP.

00500   606060606060        BCI     5,                      TSTCAP22                                                PROG0530
00505   606060606060        BCI     9,      REM     TEST OF CAP PSEUDO-OPS, AND FLAGS.                              PROG0540
00516   606060606060        BCI     5,                      TSTCAP23                                                PROG0550
00523   606060606060        BCI     9,      ILCD 8                  ILLEGAL OPCODE.                                 PROG0560
00534   606060606060        BCI     5,                      TSTCAP24                                                PROG0570
00541   606060606060        BCI     9,      TRA  UNDEF              UNDEFINED SYMBOL.                               PROG0580
00552   606060606060        BCI     5,                      TSTCAP25                                                PROG0590
00557   606060606060        BCI     9,      INT  1,2,-7,13A3,9      ERROR IN INTOP.                                 PROG0600
00570   606060606060        BCI     5,                      TSTCAP26                                                PROG0610
00575   606060606060        BCI     9,      WMW  AEN                ILLEGAL OPCODE AND UNDEFINE                     PROG0620
00606   246062704422        BCI     5,C SYMBOL.             TSTCAP27                                                PROG0630
00613   606060606060        BCI     9,      COMP    NO = YES + LOOP                                                 PROG0640
00624   606060606060        BCI     5,                      TSTCAP28                                                PROG0650
00631   234644476060        BCI     9,COMP  COMP    Z = A * B * C / D / E / FLAG                                    PROG0660
00642   606060606060        BCI     5,                      TSTCAP29                                                PROG0670
00647   606060606060        BCI     9,      TRA  COMP               USE OF SYMBOL DEFINED BY CO                     PROG0680
00660   444733606060        BCI     5,MP.                   TSTCAP30                                                PROG0690
00665   606060606060        BCI     9,      COMP    COUNT,1 = LOOP01,2 - START,4                                    PROG0700
00676   606060606060        BCI     5,                      TSTCAP31                                                PROG0710
00703   606060606060        BCI     9,      COMP    DZ = (E+Z*(D+Z*(C+Z*(B+Z*A)))) + FOO                            PROG0720
00714   606060606060        BCI     5,                      TSTCAP32                                                PROG0730
00721   606060606060        BCI     9,      TIX  COUNT+3*(((NO-YES)*Z+1)*(LOOP-COUNT)+8*(ZE                         PROG0740
00732   514640244645        BCI     5,RO-DONE)+8)           TSTCAP33                                                PROG0750
00737   606060606060        BCI     9,      REM  TEST OF PROPOSED MODIFICATIONS TO CAP.                             PROG0760
00750   606060606060        BCI     5,                      TSTCAP34                                                PROG0770
00755   446443606060        BCI     9,MUL   INT  0                  MULTIPLY DEFINED SYMBOL.                        PROG0780
00766   606060606060        BCI     5,                      TSTCAP35                                                PROG0790
00773   446443606060        BCI     9,MUL   INT  0                  MULTIPLY DEFINED SYMBOL.                        PROG0800
01004   606060606060        BCI     5,                      TSTCAP36                                                PROG0810
01011   236060606060        BCI     9,C     PZE  -32+BITS           PZE CODE.                                       PROG0820
01022   606060606060        BCI     5,                      TSTCAP37                                                PROG0830
01027   246060606060        BCI     9,D     MZE  NO+65              MZE CODE.                                       PROG0840
01040   606060606060        BCI     5,                      TSTCAP38                                                PROG0850
01045   606060606060        BCI     9,      SLW  $+1                $ FOR THIS LOCATION                             PROG0860
01056   606060606060        BCI     5,                      TSTCAP39                                                PROG0870
01063   606060606060        BCI     9,      CLA  2*$-1              $ TEST.                                         PROG0880
01074   606060606060        BCI     5,                      TSTCAP40                                                PROG0890
01101   606060606060        BCI     9,      MTH  C/3                DIVISION IN ADDRESS                             PROG0900
01112   606060606060        BCI     5,                      TSTCAP41                                                PROG0910
01117   716060606060        BCI     9,Z     PON  (YES-COUNT)/3+F    PON WITH DIVISION IN ADDR                       PROG0920
01130   256262602151        BCI     5,ESS ARITHMETIC.       TSTCAP42                                                PROG0930
01135   606060606060        BCI     9,      STO  (C-FLAG)/2         DIVISION WITH NEGATIVE ANSW                     PROG0940
01146   255133606060        BCI     5,ER.                   TSTCAP43                                                PROG0950
01153   606060606060        BCI     9,      SLW  -8*BITS+$*1+7*Z    USE OF $ AS A SYMBOL.                           PROG0960
01164   606060606060        BCI     5,                      TSTCAP44                                                PROG0970
01171   606060606060        BCI     9,      ARF  ($-C)/5+Q          $,/, AND ILLEGAL OPCODE                         PROG0980
01202   606060606060        BCI     5,                      TSTCAP45                                                PROG0990
01207   546051254421        BCI     9,* REMARK CARD WITH * IN COLUMN 1.                                             PROG1000
01220   606060606060        BCI     5,                      TSTCAP46                                                PROG1010
01225   224345426060        BCI     9,BLNK          BLNK            BLANK OPCODE.                                   PROG1020
01236   606060606060        BCI     5,                      TSTCAP47                                                PROG1030
01243   745313546160        BCI     9,($=*/ NOP  UNDEF              S, O, AND U FLAGS.                              PROG1040
C1254   606060606060        BCI     5,                      TSTCAP48                                                PROG1050
C1261   506060606060        BCI     9,Q     HOL 5THIS IS HOLLERITH INFO.                                            PROG1060
01272   606060606060        BCI     5,                      TSTCAP49                                                PROG1070
01277   606060606060        BCI     9,      CLA* FLAG               USE OF FLAGGED INSTRUCTION.                     PROG1080
```

```
        SYMBOLIC PROGRAM TO TEST CAP.

01310  606060606060          BCI   5,                                                    PROG1090
01315  264321276060          BCI   9,FLAG   STO  =1139          TSTCAP50                PROG1100
01326  606060606060          BCI   5,                                                    PROG1110
01333  606060606060          BCI   9,       CLA  =1139          TSTCAP51 LITERAL.       PROG1120
01344  606060606060          BCI   5,                                                    PROG1130
01351  605460606060          BCI   9, *     LAC* =5597.0        TSTCAP52 SAME LITERAL.  PROG1140
01362  606060606060          BCI   5,                                                    PROG1150
01367  606060606060          BCI   9,       TNX  Z,3,5          TSTCAP53 E, S, AND F FLAGS. PROG1160
01400  606060606060          BCI   5,                                                    PROG1170
01405  606060606060          BCI   9,       FAD  (YES-COUNT)*(NO-LOOP)+START,2+3*(NO-LOOP) TSTCAP54 PROG1180
01416  606060606060          BCI   5,                                                    PROG1190
01423  247160606060          BCI   9,CZ     OCT  17,13,-44,Q,13,,1  TSTCAP55 OCT PSEUDO-OP. PROG1200
01434  606060606060          BCI   5,                                                    PROG1210
01441  606060606060          BCI   9,       FDP  FLAG           TSTCAP56 A DIVIDE INSTRUCTION. PROG1220
01452  606060606060          BCI   5,                                                    PROG1230
01457  606060606060          BCI   9,       STO  MUL+UNDEF      TSTCAP57 FOLLOWED BY A STORE. PROG1240
01470  606060606060          BCI   5,                                                    PROG1250
01475  626321516360          BCI   9,START  EQU  COUNT+3*(NO-LOOP) TSTCAP58 PROPER USE OF EQU. PROG1260
01506  606060606060          BCI   5,                                                    PROG1270
01513  216060606060          BCI   9,A      TRA  START          TSTCAP59 USE OF SYMBOL DEFINED BY EQ PROG1280
01524  643360606060          BCI   5,U.                                                  PROG1290
01531  473160606060          BCI   9,PI     EQU  END-COMP       TSTCAP60 PHASE ERROR.  PROG1300
01542  606060606060          BCI   5,                                                    PROG1310
01547  226060606060          BCI   9,B      TRA  PI             TSTCAP61 USE OF SYMBOL WITH PHASE ER PROG1320
01560  514651336060          BCI   5,ROR.                                                PROG1330
01565  256060606060          BCI   9,E      BSS  5              TSTCAP62 PROPER BSS.   PROG1340
01576  606060606060          BCI   5,                                                    PROG1350
01603  266060606060          BCI   9,F      BSS  LOOP-COUNT     TSTCAP63 PROPER SYMBOLIC DEFINITION PROG1360
01614  462660226262          BCI   5,OF BSS.                                             PROG1370
01621  434646470001          BCI   9,LOOP01 BSS  CALL-F         TSTCAP64 IMPROPER BSS, PHASE ERROR. PROG1380
01632  606060606060          BCI   5,                                                    PROG1390
01637  232143436060          BCI   9,CALL   CALL COUNT,NO,C,YES TSTCAP65 LEGAL CALL MACRO. PROG1400
01650  606060606060          BCI   5,                                                    PROG1410
01655  232143430260          BCI   9,CALL2  CALL ABLE,BAKER+2,CHRLY-5*(NO-LOOP) TSTCAP66 PROG1420
01666  606060606060          BCI   5,                                                    PROG1430
01673  602646466060          BCI   9, FOO   CLA  $              TSTCAP67 CHECK ILC AFTER BSS AND CAL PROG1440
01704  433360606060          BCI   5,L.                                                  PROG1450
01711  602545246060          BCI   9, END   END  COUNT+1*UNDEF  TSTCAP68 FINALLY THE END. PROG1460
01722  606060606060          BCI   5,                                                    PROG1470
                             DETAIL                             TSTCAP69 RETURN TO NORMAL MODE. PROG1480
                 01724  LTH  EQU   *-1-PROG                    LENGTH OF PROG.        PROG1490
                             END                                                         PROG1500


                    POST PROCESSOR ASSEMBLY DATA

     1727 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
      1724    LTH     2, 1727
         2    PROG  1727


 NO ERROR IN ABOVE ASSEMBLY.
     *TIME SPENT IN FAP..    000007 IN HUNDREDTHS OF MINUTES.
```

```
      1.17 MINUTES ELAPSED SINCE START OF JOB
SUBPROGRAM STORAGE MAP FOLLOWS.
     NAME  ORIGIN ENTRY
     CAP      144   151
     PASS1    201   213
     PASS2    317   327
     VAREVL   637   644
     OPTBL   1117  1121
     INTOP   1226  1230
     SCAN    1351  1353
     SYMSTO  2462  2465
     ENDOP   3036  3042
     COMPOP  3207  3217
     EXPR    3752  3760
     TERM    4155  4163
     (MAIN)  4320  4321
     TESTS   4325  4331
END OF STORAGE MAP.
```

TEST OF CAP, BEGIN ASSEMBLY.

```
                                CAP     REM  THE FOLLOWING ARE ALL LEGAL CAP INSTRUCTIONS.                TSTCAP00
                                        REM  PROGRAM TO COUNT BITS IN AC.                                 TSTCAP01
      50000    056000050016    COUNT   LDQ  ZERO            ZERO TEST CELLS                            TSTCAP02
      50001    460000050021            STQ  BITS            ..                                         TSTCAP03
      50002    053400450020            LXA  THSX            COUNT 36 BITS.                             TSTCAP04
      50003    076000000001    LOOP    LBT                  BIT OR NO.                                 TSTCAP05
      50004    002000050013            TRA  NO              NO BIT.                                    TSTCAP06
      50005    060200050022    YES     SLW  WORD            BIT, SAVE AC,                              TSTCAP07
      50006    450000050021            CAL  BITS            AND INCREMENT COUNT.                       TSTCAP08
      50007    036100050017            ACL  ONE             ..                                         TSTCAP09
      50010    060200050021            SLW  BITS            ..                                         TSTCAP10
      50011    450000050022            CAL  WORD            RESTORE AC.                                TSTCAP11
      50012    476500000001            LGR  1               NEXT BIT.                                  TSTCAP12
      50013    200001450003    NO      TIX  LOOP            INDEX.                                     TSTCAP13
      50014    450000050021            CAL  BITS            GET COUNT.                                 TSTCAP14
      50015,   002100070000    DONE    OCTL 002100070000    STOP WITH TRANSFER TO 70000 OCTAL.         TSTCAP15
                                        REM  STORAGE.                                                   TSTCAP16
      50016    000000000000    ZERO    OCTL 000000000000    TRUE ZERO.                                 TSTCAP17
      50017    000001000000    ONE     INT  1               INCREMENT OF ONE.                          TSTCAP18
      50020    434000000044    THSX    LAS  36              ADDRESS IS 36.                             TSTCAP19
                                        REM  DATA.                                                      TSTCAP20
      50021    000000000000    BITS    INT  0               STORAGE FOR BIT COUNT.                     TSTCAP21
      50022    000000000000    WORD    INT  0               TEMPORARY STORAGE FOR AC.                  TSTCAP22
                                        REM   TEST OF CAP PSEUDO-OPS, AND FLAGS.                        TSTCAP23
0     50023    000000000010            ILCD 8               ILLEGAL OPCODE.                            TSTCAP24
U     50024    002000000000            TRA  UNDEF           UNDEFINED SYMBOL.                          TSTCAP25
E     50025    000001000000            INT  1,2,-7,13A3,9   ERROR IN INTOP.                           TSTCAP26
OU    50032    000000000000            WMW  AEN             ILLEGAL OPCODE AND UNDEFINED SYMBOL.       TSTCAP27
                                        COMP   NO = YES + LOOP                                          TSTCAP28
      50033    050000050005            CLA  YES
      50034    030000050003            FAD  LOOP
      50035    060100050013            STO  NO
                                COMP    COMP   Z = A * B * C / D / E / FLAG                             TSTCAP29
      50036    056000050143            LDQ  A
      50037    026000050145            FMP  B
      50040    013100000000            XCA
      50041    026000050114            FMP  C
      50042    024100050115            FDP  D
      50043    013100000000            XCA
      50044    024100050146            FDP  E
      50045    013100000000            XCA
      50046    024100050132            FDP  FLAG
      50047    460000050154            STQ  TEM
      50050    050000050154            CLA  TEM
      50051    060100050121            STO  Z
      50052    002000050036            TRA  COMP            USE OF SYMBOL DEFINED BY COMP.             TSTCAP30
                                        COMP   COUNT,1 = LOOP01,2 - START,4                             TSTCAP31
      50053    050000050150            CLA  LOOP01,2
      50054    030200050142            FSB  START,4
      50055    060100050000            STO  COUNT,1
                                        COMP   DZ = (E+Z*(D+Z*(C+Z*(B+Z*A)))) + FOO                     TSTCAP32
      50056    056000050121            LDQ  Z
      50057    026000050143            FMP  A
      50060    060100050154            STO  TEM
      50061    050000050145            CLA  B
      50062    030000050154            FAD  TEM
      50063    060100050155            STO  TEM+1
      50064    056000050121            LDQ  Z
      50065    026000050155            FMP  TEM+1
```

```
      50066   060100050156          STO  TEM+2
      50067   050000050114          CLA  C
      50070   030000050156          FAD  TEM+2
      50071   060100050157          STO  TEM+3
      50072   056000050121          LDQ  Z
      50073   026000050157          FMP  TEM+3
      50074   060100050160          STO  TEM+4
      50075   050000050115          CLA  D
      50076   030000050160          FAD  TEM+4
      50077   060100050161          STO  TEM+5
      50100   056000050121          LDQ  Z
      50101   026000050161          FMP  TEM+5
      50102   060100050162          STO  TEM+6
      50103   050000050146          CLA  E
      50104   030000050162          FAD  TEM+6
      50105   060100050163          STO  TEM+7
      50106   050000050163          CLA  TEM+7
      50107   030000050153          FAD  FOO
      50110   060100050137          STO  DZ
      50111   200001440517          TIX  COUNT+3*(((NO-YES)*Z+1)*(LOOP-COUNT)+8*(ZERO-DONE)+8)      TSTCAP33
                                    REM  TEST OF PROPOSED MODIFICATIONS TO CAP.                      TSTCAP34
      50112   000000000000  MUL     INT  0              MULTIPLY DEFINED SYMBOL.                     TSTCAP35
      50113   000000000000  MUL     INT  0              MULTIPLY DEFINED SYMBOL.                     TSTCAP36
0     50114   000000047761  C       PZE  -32+BITS       PZE CODE.                                    TSTCAP37
0     50115   000000050114  D       MZE  NO+65          MZE CODE.                                    TSTCAP38
U     50116   060200000001          SLW  $+1            $ FOR THIS LOCATION                          TSTCAP39
U     50117   050000077777          CLA  2*$-1          $ TEST.                                      TSTCAP40
OU    50120   000000000000          MTH  C/3            DIVISION IN ADDRESS                          TSTCAP41
0     50121   000000050154  Z       PON  (YES-COUNT)/3+F  PON WITH DIVISION IN ADDRESS ARITHMETIC.   TSTCAP42
      50122   060100077762          STO  (C-FLAG)/2     DIVISION WITH NEGATIVE ANSWER.               TSTCAP43
U     50123   060200030657          SLW  -8*BITS+$*1+7*Z  USE OF $ AS A SYMBOL.                     TSTCAP44
OU    50124   000000000014          ARF  ($-C)/5+Q      $,/, AND ILLEGAL OPCODE                      TSTCAP45
0     50125   000000050115  * REMARK CARD WITH * IN COLUMN 1.                                       TSTCAP46
0     50126   000000050126  BLNK        BLNK            BLANK OPCODE.                                TSTCAP47
OU    50127   000000000000  ($=*/ NOP   UNDEF           S, O, AND U FLAGS.                           TSTCAP48
OU    50130   000000000000  Q       HOL 5THIS IS HOLLERITH INFO.                                     TSTCAP49
0     50131   000000050132          CLA* FLAG           USE OF FLAGGED INSTRUCTION.                  TSTCAP50
      50132   060100031043  FLAG    STO  =1139          LITERAL.                                     TSTCAP51
      50133   050000031043          CLA  =1139          SAME LITERAL.                                TSTCAP52
OU    50134   000000000000   *      LAC* =5597.0        E, S, AND F FLAGS.                           TSTCAP53
0     50135   000000050121          TNX  Z,3,5          TAG AND DECREMENT FIELD.                     TSTCAP54
      50136   030000050212          FAD  (YES-COUNT)*(NO-LOOP)+START,2+3*(NO-LOOP)                   TSTCAP55
0     50137   000000000021  DZ      OCT  17,13,-44,Q,13,,1  OCT PSEUDO-OP.                         TSTCAP56
      50140   024100050132          FDP  FLAG           A DIVIDE INSTRUCTION.                        TSTCAP57
U     50141   060100050113          STO  MUL+UNDEF      FOLLOWED BY A STORE.                         TSTCAP58
0     50142   000000050030  START   EQU  COUNT+3*(NO-LOOP)  PROPER USE OF EQU.                       TSTCAP59
      50143   002000050142  A       TRA  START          USE OF SYMBOL DEFINED BY EQU.                TSTCAP60
0     50144   000000000126  PI      EQU  END-COMP       PHASE ERROR.                                 TSTCAP61
      50145   002000050144  B       TRA  PI             USE OF SYMBOL WITH PHASE ERROR.              TSTCAP62
0     50146   000000000005  E       BSS  5              PROPER BSS.                                  TSTCAP63
0     50147   000000000003  F       BSS  LOOP-COUNT     PROPER SYMBOLIC DEFINITION OF BSS.           TSTCAP64
0     50150   000000000002  LCOP01  BSS  CALL-F         IMPROPER BSS, PHASE ERROR.                   TSTCAP65
0     50151   000000050000  CALL    CALL COUNT,NO,C,YES LEGAL CALL MACRO.                            TSTCAP66
OU    50152   000000000000  CALL2   CALL ABLE,BAKER+2,CHRLY-5*(NO-LOOP)                               TSTCAP67
U     50153   050000000000   FOO    CLA  $              CHECK ILC AFTER BSS AND CALL.                TSTCAP68
                             TEM    REM  TEMPORARY STORAGE AREA BEGINS HERE.
U                 50000      END    END  COUNT+1*UNDEF  FINALLY THE END.                             TSTCAP69

     RETURN FROM CAP, ENTRY POINT IS 50000.
```

```
        POST MORTEM OF SYMBOLIC PROGRAM.

CAP     REM  THE FOLLOWING ARE ALL LEGAL CAP INSTRUCTIONS.              TSTCAP00
        REM  PROGRAM TO COUNT BITS IN AC.                               TSTCAP01
COUNT   LDQ  ZERO            ZERO TEST CELLS                            TSTCAP02
        STQ  BITS            ..                                         TSTCAP03
        LXA  THSX            COUNT 36 BITS.                             TSTCAP04
LOOP    LBT                  BIT OR NO.                                 TSTCAP05
        TRA  NO              NO BIT.                                    TSTCAP06
YES     SLW  WORD            BIT, SAVE AC,                              TSTCAP07
        CAL  BITS            AND INCREMENT COUNT.                       TSTCAP08
        ACL  ONE             ..                                         TSTCAP09
        SLW  BITS            ..                                         TSTCAP10
        CAL  WORD            RESTORE AC.                                TSTCAP11
        LGR  1               NEXT BIT.                                  TSTCAP12
NO      TIX  LOOP            INDEX.                                     TSTCAP13
        CAL  BITS            GET COUNT.                                 TSTCAP14
DONE    CCTL 002100070000    STOP WITH TRANSFER TO 70000 OCTAL.         TSTCAP15
        REM  STORAGE.                                                   TSTCAP16
ZERO    OCTL 000000000000    TRUE ZERO.                                 TSTCAP17
ONE     INT  1               INCREMENT OF ONE.                          TSTCAP18
THSX    LAS  36              ADDRESS IS 36.                             TSTCAP19
        REM  DATA.                                                      TSTCAP20
BITS    INT  0               STORAGE FOR BIT COUNT.                     TSTCAP21
WORD    INT  0               TEMPORARY STORAGE FOR AC.                  TSTCAP22
        REM   TEST OF CAP PSEUDO-OPS, AND FLAGS.                        TSTCAP23
        ILCD 8               ILLEGAL OPCODE.                            TSTCAP24
        TRA  UNDEF           UNDEFINED SYMBOL.                          TSTCAP25
        INT  1,2,-7,13A3,9   ERROR IN INTOP.                            TSTCAP26
        WMW  AEN             ILLEGAL OPCODE AND UNDEFINED SYMBOL.       TSTCAP27
        COMP   NO = YES + LOOP                                          TSTCAP28
COMP    COMP   Z = A * B * C / D / E / FLAG                             TSTCAP29
        TRA  COMP            USE OF SYMBOL DEFINED BY COMP.             TSTCAP30
        COMP   COUNT,1 = LCOP01,2 - START,4                             TSTCAP31
        COMP   DZ = (E+Z*(D+Z*(C+Z*(B+Z*A)))) + FOO                     TSTCAP32
        TIX  COUNT+3*(((NO-YES)*Z+1)*(LOOP-COUNT)+8*(ZERO-DONE)+8)      TSTCAP33
        REM  TEST OF PROPOSED MODIFICATIONS TO CAP.                     TSTCAP34
MUL     INT  0               MULTIPLY DEFINED SYMBOL.                   TSTCAP35
MUL     INT  0               MULTIPLY DEFINED SYMBOL.                   TSTCAP36
C       PZE  -32+BITS        PZE CODE.                                  TSTCAP37
D       MZE  NO+65           MZE CODE.                                  TSTCAP38
        SLW  $+1             $ FOR THIS LOCATION                        TSTCAP39
        CLA  2*$-1           $ TEST.                                    TSTCAP40
        MTH  C/3             DIVISION IN ADDRESS                        TSTCAP41
Z       PON  (YES-COUNT)/3+F  PON WITH DIVISION IN ADDRESS ARITHMETIC.  TSTCAP42
        STO  (C-FLAG)/2      DIVISION WITH NEGATIVE ANSWER.             TSTCAP43
        SLW  -8*BITS+$*1+7*Z  USE OF $ AS A SYMBOL.                     TSTCAP44
        ARF  ($-C)/5+Q       $,/, AND ILLEGAL OPCODE                    TSTCAP45
* REMARK CARD WITH * IN COLUMN 1.                                       TSTCAP46
BLNK        BLNK             BLANK OPCODE.                              TSTCAP47
($=*/   NOP  UNDEF           S, O, AND U FLAGS.                         TSTCAP48
Q       HOL 5THIS IS HOLLERITH INFO.                                    TSTCAP49
        CLA* FLAG            USE OF FLAGGED INSTRUCTION.                TSTCAP50
FLAG    STO  =1139           LITERAL.                                   TSTCAP51
        CLA  =1139           SAME LITERAL.                              TSTCAP52
 *      LAC* =5597.0         E, S, AND F FLAGS.                         TSTCAP53
        TNX  Z,3,5           TAG AND DECREMENT FIELD.                   TSTCAP54
        FAD  (YES-COUNT)*(NO-LOOP)+START,2+3*(NO-LOOP)                  TSTCAP55
DZ      OCT  17,13,-44,Q,13,,1  OCT PSEUDO-OP.                          TSTCAP56
        FDP  FLAG            A DIVIDE INSTRUCTION.                      TSTCAP57
```

```
        STO   MUL+UNDEF       FOLLOWED BY A STORE.                    TSTCAP58
START   EQU   COUNT+3*(NO-LOOP)  PROPER USE OF EQU.                    TSTCAP59
A       TRA   START           USE OF SYMBOL DEFINED BY EQU.           TSTCAP60
PI      EQU   END-COMP        PHASE ERROR.                            TSTCAP61
B       TRA   PI              USE OF SYMBOL WITH PHASE ERROR.         TSTCAP62
E       BSS   5               PROPER BSS.                             TSTCAP63
F       BSS   LOOP-COUNT      PROPER SYMBOLIC DEFINITION OF BSS.      TSTCAP64
LCOP01  BSS   CALL-F          IMPROPER BSS, PHASE ERROR.              TSTCAP65
CALL    CALL  COUNT,NO,C,YES  LEGAL CALL MACRO.                       TSTCAP66
CALL2   CALL  ABLE,BAKER+2,CHRLY-5*(NO-LOOP)                          TSTCAP67
 FOO    CLA   $               CHECK ILC AFTER BSS AND CALL.           TSTCAP68
 END    END   COUNT+1*UNDEF   FINALLY THE END.                        TSTCAP69
```

```
        POST MORTEM OF COLLATION TAPE.

CAP     REM  THE FOLLOWING ARE ALL LEGAL CAP INSTRUCTIONS.          TSTCAP00
        REM  PROGRAM TO COUNT BITS IN AC.                           TSTCAP01
COUNT   LDQ  ZERO          ZERO TEST CELLS                          TSTCAP02
        STQ  BITS          ..                                       TSTCAP03
        LXA  THSX          COUNT 36 BITS.                           TSTCAP04
LOOP    LBT                BIT OR NO.                               TSTCAP05
        TRA  NO            NO BIT.                                  TSTCAP06
YES     SLW  WORD          BIT, SAVE AC,                            TSTCAP07
        CAL  BITS          AND INCREMENT COUNT.                     TSTCAP08
        ACL  ONE           ..                                       TSTCAP09
        SLW  BITS          ..                                       TSTCAP10
        CAL  WORD          RESTORE AC.                              TSTCAP11
        LGR  1             NEXT BIT.                                TSTCAP12
NO      TIX  LOOP          INDEX.                                   TSTCAP13
        CAL  BITS          GET COUNT.                               TSTCAP14
DONE    OCTL 002100070000  STOP WITH TRANSFER TO 70000 OCTAL.       TSTCAP15
        REM  STORAGE.                                               TSTCAP16
ZERO    OCTL 000000000000  TRUE ZERO.                               TSTCAP17
ONE     INT  1             INCREMENT OF ONE.                        TSTCAP18
THSX    LAS  36            ADDRESS IS 36.                           TSTCAP19
        REM  DATA.                                                  TSTCAP20
BITS    INT  0             STORAGE FOR BIT COUNT.                   TSTCAP21
WORD    INT  0             TEMPORARY STORAGE FOR AC.                TSTCAP22
        REM   TEST OF CAP PSEUDO-OPS, AND FLAGS.                    TSTCAP23
        ILCD 8             ILLEGAL OPCODE.                          TSTCAP24
        TRA  UNDEF         UNDEFINED SYMBOL.                        TSTCAP25
        INT  1,2,-7,13A3,9 ERROR IN INTOP.                          TSTCAP26
        WMW  AEN           ILLEGAL OPCODE AND UNDEFINED SYMBOL.     TSTCAP27
        COMP   NO = YES + LOOP                                      TSTCAP28
        CLA  YES
        FAD  LOOP
        STO  NO
COMP    COMP   Z = A * B * C / D / E / FLAG                         TSTCAP29
        LDQ  A
        FMP  B
        XCA
        FMP  C
        FDP  D
        XCA
        FDP  E
        XCA
        FDP  FLAG
        STQ  TEM
        CLA  TEM
        STO  Z
        TRA  COMP          USE OF SYMBOL DEFINED BY COMP.           TSTCAP30
        COMP   COUNT,1 = LOOP01,2 - START,4                         TSTCAP31
        CLA  LOOP01,2
        FSB  START,4
        STO  COUNT,1
        COMP   DZ = (E+Z*(D+Z*(C+Z*(B+Z*A)))) + FCO                 TSTCAP32
        LDQ  Z
        FMP  A
        STO  TEM
        CLA  B
        FAD  TEM
        STO  TEM+1
        LDQ  Z
```

```
        FMP   TEM+1
        STO   TEM+2
        CLA   C
        FAD   TEM+2
        STO   TEM+3
        LDQ   Z
        FMP   TEM+3
        STO   TEM+4
        CLA   D
        FAD   TEM+4
        STO   TEM+5
        LDQ   Z
        FMP   TEM+5
        STO   TEM+6
        CLA   E
        FAD   TEM+6
        STO   TEM+7
        CLA   TEM+7
        FAD   FOO
        STO   DZ
        TIX   COUNT+3*(((NO-YES)*Z+1)*(LOOP-COUNT)+8*(ZERO-DONE)+8)      TSTCAP33
        REM   TEST OF PROPOSED MODIFICATIONS TO CAP.                     TSTCAP34
MUL     INT   0               MULTIPLY DEFINED SYMBOL.                   TSTCAP35
MUL     INT   0               MULTIPLY DEFINED SYMBOL.                   TSTCAP36
C       PZE   -32+BITS        PZE CODE.                                  TSTCAP37
C       MZE   NO+65           MZE CODE.                                  TSTCAP38
        SLW   $+1             $ FOR THIS LOCATION                        TSTCAP39
        CLA   2*$-1           $ TEST.                                    TSTCAP40
        MTH   C/3             DIVISION IN ADDRESS                        TSTCAP41
Z       PON   (YES-COUNT)/3+F  PON WITH DIVISION IN ADDRESS ARITHMETIC.  TSTCAP42
        STO   (C-FLAG)/2      DIVISION WITH NEGATIVE ANSWER.             TSTCAP43
        SLW   -8*BITS+$*1+7*Z  USE OF $ AS A SYMBOL.                     TSTCAP44
        ARF   ($-C)/5+Q       $,/, AND ILLEGAL OPCODE                    TSTCAP45
* REMARK CARD WITH * IN COLUMN 1.                                        TSTCAP46
BLNK            BLNK          BLANK OPCODE.                              TSTCAP47
($=*/   NOP   UNDEF           S, O, AND U FLAGS.                         TSTCAP48
Q       HOL 5THIS IS HOLLERITH INFO.                                     TSTCAP49
        CLA*  FLAG            USE OF FLAGGED INSTRUCTION.                TSTCAP50
FLAG    STO   =1139           LITERAL.                                   TSTCAP51
        CLA   =1139           SAME LITERAL.                              TSTCAP52
 *      LAC*  =5597.0         E, S, AND F FLAGS.                         TSTCAP53
        TNX   Z,3,5           TAG AND DECREMENT FIELD.                   TSTCAP54
        FAD   (YES-COUNT)*(NO-LOOP)+START,2+3*(NO-LOOP)                  TSTCAP55
DZ      OCT   17,13,-44,Q,13,,1  OCT PSEUDO-OP.                          TSTCAP56
        FDP   FLAG            A DIVIDE INSTRUCTION.                      TSTCAP57
        STO   MUL+UNDEF       FOLLOWED BY A STORE.                       TSTCAP58
START   EQU   COUNT+3*(NO-LOOP)  PROPER USE OF EQU.                      TSTCAP59
A       TRA   START           USE OF SYMBOL DEFINED BY EQU.              TSTCAP60
PI      EQU   END-COMP        PHASE ERROR.                               TSTCAP61
B       TRA   PI              USE OF SYMBOL WITH PHASE ERROR.            TSTCAP62
E       BSS   5               PROPER BSS.                                TSTCAP63
F       BSS   LOOP-COUNT      PROPER SYMBOLIC DEFINITION OF BSS.         TSTCAP64
LOOP01  BSS   CALL-F          IMPROPER BSS, PHASE ERROR.                 TSTCAP65
CALL    CALL  COUNT,NO,C,YES LEGAL CALL MACRO.                           TSTCAP66
CALL2   CALL  ABLE,BAKER+2,CHRLY-5*(NO-LOOP)                             TSTCAP67
 FOO    CLA   $               CHECK ILC AFTER BSS AND CALL.              TSTCAP68
 TEM    REM   TEMPORARY STORAGE AREA BEGINS HERE.
 END    END   COUNT+1*UNDEF   FINALLY THE END.                           TSTCAP69
```

```
POST MORTEM OF CONSOLE.

     S,Q,P= 0,0,0

     AC =  000000050000

     MQ =  606060600500

     SI =  000000000007

     IR1=   27624

     IR2=       1

     IR4=   73226

OCTAL DUMP OF CAP    FOLLOWS.
      144   TTR 002100000213  TTR 002100000327  TTR 002100007767  HTR 000000000000  TIX 232147606060  SXA 063400400164
      152   STO 060100000171  LDI 044100000200  TSX 007400400144  LFT 405400000001  TRA 002000000166  STI 060400000172
      160   LDI 044100000200  CLA 050000000171  TSX 007400400145  OSI 044200000172  AXT 077400473226  TRA 002000400001
      166   TSX 007400400146  HTR 000005000173  TRA 002000000157  HTR 000000050000  HTR 000000000000  TCO 006270442246
      174   MZE 436063212243  TIX 256062317125  TNX 602567232525  TIX 242524336060  HTR 000000000000

OCTAL DUMP OF PASS1  FOLLOWS.
      201   TTR 002100007615  TTR 002100007553  TTR 002100001353  TTR 002100002465  TTR 002100001375  TTR 002100003217
      207   TTR 002100003042  TTR 002100007673  HTR 000000000000  MZE 472162620160  SXA 063400400277  PAC 073700100000
      215   TRA 002000000220  TSX 007400400201  HTR 000000000301  TSX 007400400202  HTR 000000000301  CAL 450000000302
      223   TSX 007400400203  AXT 077400400012  LAS 434000400246  TRA 002000000230  TRA 002060400247  TIX 200002400225
      231   CAL 450000000301  TSX 007400400204  TXI 177777100216  HTR 000000512544  TRA 002000000246  HTR 000000314563
      237   TRA 002000000247  HTR 000046236343  TRA 002000000263  HTR 000023464447  TRA 002000000254  HTR 000000254524
      245   TRA 002000000266  TRA 002000000216  CAL 450000000301  TSX 007400400204  TSX 007400400205  HTR 000000000301
      253   TRA 002000000216  CAL 450000000301  TSX 007400400204  TSX 007400400201  HTR 000000000301  TSX 007400400206
      261   HTR 000000000301  TRA 002000000220  CAL 450000000301  TSX 007400400204  TXI 177777100216  TSX 007400400207
      267   TSX 007400400201  HTR 000000000301  CAL 450000000301  TSX 007400400204  TSX 007400400210  PXA 075400100000
      275   PAC 073700400000  PXA 075400400000  AXT 077400477624  TRA 002000400001  TNX 602545246060  TNX 602545246060
      303   TIX 234664456320  PZE 015464452425  TIX 266060263145  TIX 214343706063  TXH 302560254524  TXH 336060606060
      311   TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 636263232147  PZE 061160606060

OCTAL DUMP OF PASS2  FOLLOWS.
      317   TTR 002100001121  TTR 002100007703  TTR 002100001353  TTR 002100000644  TTR 002100001230  TTR 002100007767
      325   HTR 000000000000  MZE 472162620260  SXA 063400400455  PAC 073700100000  CAL 450060000317  ARS 077100000022
      333   STA 062100000356  ACL 036160000317  STA 062100000357  STA 062100000366  LDI 044100000631  STZ 060000000602
      341   OSI 044200000602  STI 060400000602  LDI 044100000631  SXA 063400100576  TSX 007400400320  HTR 000000000613
      347   CAL 450000000614  TSX 007400400321  AXT 077400400012  LAS 434000400407  TRA 002000000355  TRA 002060400410
      355   TIX 200002400352  AXT 077400400104  LAS 434000401226  TRA 002000000362  TXI 177777400366  TIX 200002400357
      363   SIR 005500000002  PXD 475400000000  TRA 002000000367  CAL 450000401226  SLW 060200100000  TSX 007400400322
      371   HTR 000000000613  ORS 460200100000  TSX 007400400461  TXI 177777100341  HTR 000000512544  TRA 002000000407
      377   HTR 000000314563  TRA 002000000411  HTR 000046236343  TRA 002000000415  HTR 000023464447  TRA 002000000430
      405   HTR 000000254524  TRA 002000000432  TSX 007400400512  TRA 002000000341  TSX 007400400323  HTR 000000000613
      413   TSX 007400400461  TRA 002000000341  PXD 475400000000  AXT 077400200002  AXT 077400400006  LDQ 056000200617
      421   RQL 477300000003  LGL 476300000003  TIX 200001400421  TIX 200001200417  SLW 060200100000  TSX 007400400461
      427   TXI 177777100341  TSX 007400400512  TRA 002000000341  TSX 007400400322  HTR 000000000613  STO 060100000577
      435   TSX 007400400523  SLW 060200000606  CLA 050000000577  TSX 007400400544  LDQ 056000000631  LGR 476500000022
      443   ORA 450100000635  SLW 060200000611  XCL 413000000000  ORA 450100000634  SLW 060200000612  CAL 450000000636
      451   SLW 060200000607  SLW 060200000610  TSX 007400400324  HTR 000023000606  AXT 077400477616  CLA 050000000577
      457   OSI 044200000602  TRA 002000400001  SXA 063400400510  TSX 007400400523  SLW 060200000606  LAC 053500400576
      465   PXA 075400400000  TSX 007400400544  SLW 060200000607  LXA 053400400576  CAL 450000400000  TSX 007400400555
      473   STQ 460000000601  LGR 476500000022  ORA 450100000635  SLW 060200000610  STQ 460000000611  CAL 450000000601
      501   LDQ 056000000631  LGR 476500000022  XCL 413000000000  ORA 450100000634  SLW 060200000612  TSX 007400400324
      507   HTR 000023000606  AXT 077400477405  TRA 002000400001  SXA 063400400521  AXT 077400400005  CAL 450000000636
```

```
  515   SLW 060200400613  TIX 200001400515  TSX 007400400324  HTR 000023000606  AXT 077400477371  TRA 002000400001
  523   SXA 063400400542  TOV 014000000525  PIA 404600000000  LGR 476500000003  CAL 450000000633  AXT 077400400003
  531   TQP 016200000534  ALS 076700000006  ACL 036100400606  RQL 477300000001  TIX 200001400531  LDQ 056000000636
  537   TOV 014000000542  LGL 476300000006  TNO 414000000540  AXT 077400477343  TRA 002000400001  SXA 063400400553
  545   LGR 476500000017  CAL 450000000632  AXT 077400400005  ALS 076700000003  LGL 476300000003  TIX 200001400550
  553   AXT 077400477340  TRA 002000400001  SXA 063400400574  XCL 413000000000  PXD 475400000000  AXT 077400400006
  561   ALS 076700000003  LGL 476300000003  TIX 200001400561  SLW 060200000600  PXD 475400000000  AXT 077400400006
  567   ALS 076700000003  LGL 476300000003  TIX 200001400567  XCL 413000000000  CAL 450000000600  AXT 077400477306
  575   TRA 002000400001  HTR 000000027624  HTR 000000050000  PZE 000500000000  HTR 000000000000  HTR 000000000007
  603   HTR 000000000025  HTR 000000000046  HTR 000000000064  TNX 606460606060  TNX 606060606060  TNX 606060606060
  611   TNX 606060600500  HTR 000000606060  TNX 602545246060  TNX 602545246060  TIX 234664456320  PZE 015464452425
  617   TIX 266060263145  TIX 214343706063  TXH 302560254524  TXH 336060606060  TNX 606060606060  TNX 606060606060
  625   TNX 606060606060  TNX 606060606060  TNX 636263232147  PZE 061160606060  HTR 000000000000  HTR 000000000060
  633   HTR 000000000160  HTR 000000606060  TNX 606060000000  TNX 606060606060

OCTAL DUMP OF VAREVL FOLLOWS.
  637   TTR 002100002510  TTR 002100001423  TTR 002100001445  HTR 000000000000  STR 516551256543  SXA 063400400667
  645   SXA 063400200670  SXA 063400100671  CAL 450000400001  ACL 036100001113  STA 062100000722  STZ 060000001104
  653   STZ 060000001105  MSE 476000000141  NOP 076100000000  AXT 077400200012  AXT 077400100007  LDQ 056060000722
  661   STQ 460000001111  TSX 007400400711  PAX 073400400000  TPL 012000000666  PAC 073700400000  PXA 075400400000
  667   AXT 077400477346  AXT 077400200001  AXT 077400127624  TRA 002000400002  PXD 475400000000  NZT 452000001105
  675   TRA 002000400001  STZ 060000001104  STZ 060000001105  MSE 476000000141  NOP 076100000000  TXI 100001400703
  703   SXA 063400400667  SXA 063400200670  SXA 063400100671  AXT 077400100002  AXT 077400200012  TRA 002000000662
  711   SXA 063400401102  STZ 060000001100  STZ 060000001101  STZ 060000001110  STZ 060000001107  AXT 077400400025
  717   SXA 063400401103  TRA 002000000737  AXT 077400100006  LDQ 056000200627  PXD 475400000000  LGL 476300000006
  725   STQ 460000001111  AXT 077400400025  LAS 434000400775  TRA 002000000732  TRA 002000000775  TIX 200003400727
  733   LGR 476500000006  CAL 450000001107  LGL 476300000006  SLW 060200001107  ZET 052000001104  TRA 002000000745
  741   LDQ 056000001111  TIX 200001100723  TIX 200001200721  STL 462500001104  AXT 077400400003  CAL 450000001114
  747   TRA 002000000775  HTR 000000000020  TRA 002000001031  TRA 002000001047  HTR 000000000040  TRA 002000001034
  755   TRA 002000001047  HTR 000000000054  TRA 002000001037  TRA 002000001054  HTR 000000000060  HTR 000060000762
  763   TRA 002000001076  HTR 000000000073  HTR 000060000765  TRA 002000001070  HTR 000000000074  HTR 000060000770
  771   TRA 002000001055  HTR 000000000034  HTR 000060000773  TRA 002000001064  SXA 063400401106  LAS 434000001115
  777   TRA 002000001001  TRA 002000001055  ZET 052000001110  TRA 002000001026  CAL 450000001107  ANA 432000001116
 1005   TZE 010000001012  CAL 450000001107  TSX 007400400637  STO 060100001110  TRA 002000001026  LDQ 056000001107
 1013   AXT 077400400006  PXD 475400000000  LGL 476300000006  STO 060100001112  CLA 050000001110  ALS 076700000002
 1021   ADD 040000001110  ALS 076700000001  ADD 040000001112  STO 060100001110  TIX 200001401014  STZ 060000001107
 1027   LXA 053400401103  TRA 002000400776  CLA 050000001110  STO 060100001101  TRA 002000001043  CLS 050200001110
 1035   STO 060100001101  TRA 002000001043  LDQ 056000001110  MPY 020000001101  STQ 460000001101  TRA 002000001043
 1043   LXA 053400401106  SXA 063400401103  STZ 060000001110  TRA 002000400777  CLA 050000001100  ADD 040000001101
 1051   STO 060100001100  STZ 060000001101  TRA 002000000737  TRA 002000000737  TSX 007400400640  HTR 000004001100
 1057   TSX 007400400711  STO 060100001110  TSX 007400400641  HTR 000004001100  TRA 002000000737  CLA 050000001100
 1065   ADD 040000001101  LXA 053400401102  TRA 002000400001  STL 462500001105  STL 462500001104  PSE 076000000141
 1073   SXA 063400100706  SXA 063400200707  TRA 002000001064  STL 462500001104  TRA 002000001064  HTR 000000050000
 1101   HTR 000000000000  HTR 000000077116  HTR 000000000014  HTR 000000001077  HTR 000000000000  HTR 000000000014
 1107   HTR 000000000000  HTR 000000000000  TNX 602631450000  HTR 000000000001  HTR 000000000014  HTR 000000000034
 1115   HTR 000000000074  TNX 606060606060

OCTAL DUMP OF OPTBL  FOLLOWS.
 1117   HTR 000000000000  MZE 464763224360  PZE 000104001122  HTR 000000212343  ACL 036100000000  HTR 000000214521
 1125   ANA 432000000000  HTR 000000232143  CAL 450000000000  HTR 000000233062  PSE 076000000002  HTR 000000234321
 1133   CLA 050000000000  HTR 000000234362  CLS 050200000000  HTR 000000234644  PSE 076000000006  HTR 000000262124
 1141   FAD 030000000000  HTR 000000262447  FDP 024100000000  HTR 000000264447  FMP 026000000000  HTR 000000266222
 1147   FSB 030200000000  HTR 000000432123  LAC 053500400000  HTR 000000432162  LAS 434000000000  HTR 000000432263
 1155   PSE 076000000001  HTR 000000432450  LDQ 056000000000  HTR 000000432743  LGL 476300000000  HTR 000000432751
 1163   LGR 476500000000  HTR 000000436721  LXA 053400400000  HTR 000000465121  ORA 450100000000  HTR 000000472263
 1171   MSE 476000000001  HTR 000000515043  RQL 477300000000  HTR 000000624366  SLW 060200000000  HTR 000000626346
 1177   STO 060100000000  HTR 000000626350  STQ 460000000000  HTR 000000626721  SXA 063400400000  HTR 000000633167
 1205   TIX 200001400000  HTR 000000634431  TMI 412000000000  HTR 000000634743  TPL 012000000000  HTR 000000635047
 1213   TQP 016200000000  HTR 000000635121  TRA 002000000000  HTR 000000636267  TSX 007400400000  HTR 000000637125
 1221   TZE 010000000000  HTR 000000672321  XCA 013100000000  HTR 000000672343  XCL 413000000000
```

```
OCTAL DUMP OF INTOP  FOLLOWS.
      1226   HTR 000000000000  TXH 314563464760  SXA 063400401340  SXA 063400201341  SXA 063400101314  CAL 450000400001
      1234   ACL 036100001350  STA 062100001242  STZ 060000001343  STZ 060000001345  STZ 060000001346  AXT 077400200012
      1242   LDQ 056000200627  AXT 077400400006  PXD 475400000000  LGL 476300000006  LAS 434000001347  TRA 002000001264
      1250   TRA 002000001302  STL 462500001346  STO 060100001344  CLA 050000001343  ALS 076700000002  ADD 040000001343
      1256   ALS 076700000001  ACL 036100001344  STO 060100001343  TIX 200001401244  TIX 200001201242  TRA 002000001330
      1264   AXT 077400100010  LAS 434000101302  TRA 002000001270  TRA 002060101303  TIX 200002101265  TRA 002000001302
      1272   HTR 000000000020  TRA 002000001305  HTR 000000000040  TRA 002000001310  HTR 000000000073  TRA 002000001314
      1300   HTR 000000000060  TRA 002000001330  SIR 005500000004  STL 462500001345  TRA 002000001261  ZET 052000001346
      1306   TRA 002000001302  TRA 002000001261  ZET 052000001346  TRA 002000001302  CLS 050200001343  TRA 002000001260
      1314   AXT 077400127665  CLA 050000001343  ALS 076700000022  ZET 052000001345  PXD 475400000000  STZ 060000001345
      1322   STO 060100100000  STZ 060000001343  STZ 060000001346  TXI 177777101326  SXA 063400101314  TRA 002000001261
      1330   LXA 053400101314  CLA 050000001343  ALS 076700000022  ZET 052000001345  PXD 475400000000  STZ 060000001345
      1336   STO 060100100000  TXI 177777101340  AXT 077400477367  AXT 077400200001  TRA 002000400002  HTR 000000000000
      1344   HTR 000000000000  HTR 000000000000  HTR 000000001252  HTR 000000000012  HTR 000000000014

OCTAL DUMP OF SCAN   FOLLOWS.
      1351   HTR 000000000000  TIX 234644442160  SXA 063400401372  XCL 413000000000  AXT 077400400006  STZ 060000001374
      1357   PXD 475400000000  LGL 476300000006  LAS 434000002460  TRA 002000001364  TRA 002000001370  LGR 476500000006
      1365   CAL 450000001374  LGL 476300000006  SLW 060200001374  TIX 200001401357  CAL 450000001374  AXT 077400477430
      1373   TRA 002000400001  HTR 000000254524  SXA 063400401420  SXA 063400201421  CAL 450000400001  ACL 036100002457
      1401   STA 062100001404  AXT 077400400012  AXT 077400200006  LDQ 056000400315  PXD 475400000000  LGL 476300000006
      1407   LAS 434000002461  TRA 002000001415  TXI 177777101415  LAS 434000002460  TRA 002000001415  TRA 002000001417
      1415   TIX 200001201405  TIX 200001401403  TXI 177777101420  AXT 077400477527  AXT 077400200116  TRA 002000400002
      1423   SXA 063400401442  SXA 063400201443  CAL 450000400001  PDX 473400200000  PXA 075400200000  ACL 036100400001
      1431   STA 062100001433  AXT 077400400764  CAL 450000201104  SLW 060200402457  TIX 200001401440  SIL 405500000002
      1437   TRA 002000001442  TIX 200001201433  SXA 063400401432  AXT 077400476723  AXT 077400200011  TRA 002000400002
      1445   SXA 063400401470  SXA 063400201471  CAL 450000400001  STD 062200001466  ARS 077100000022  ACL 036100400001
      1453   STA 062100001464  AXT 077400200001  LXA 053400401432  TXI 100001401457  TXL 700764401462  SIL 405500000004
      1461   TRA 002000001470  CAL 450000402457  STZ 060000402457  SLW 060200201104  TXI 100001201466  TXL 700004201456
      1467   SXA 063400401432  AXT 077400476717  AXT 077400200010  TRA 002000400002  HTR 000000000000  HTR 000000000000

                                .. FOLLOWING   498 CELLS ALL CONTAIN  HTR 000000000000 ..

      2457   HTR 000000000014  HTR 000000000060  HTR 000000000073

OCTAL DUMP OF SYMSTO FOLLOWS.
      2462   TTR 002100001353  HTR 000000000000  TNX 627044272563  LAS 434000003035  TRA 002000002470  TRA 002000400001
      2470   SXA 063400402506  TSX 007400402462  XCL 413000000000  PXA 075400100000  PAC 073700400000  PXA 075400400000
      2476   LXD 453400402524  TXI 100002402500  TXL 700310402503  SIL 405500000001  TRA 002000002506  STQ 460000403035
      2504   SLW 060200403036  SXD 463400402524  AXT 077400477506  TRA 002000400001  SXA 063400402522  LXD 453400402524
      2512   LAS 434000403035  TRA 002000002515  TRA 002000002521  TIX 200002402512  PXD 475400000000  SIR 005500000001
      2520   TRA 002000002522  CAL 450000403036  AXT 077400476771  TRA 002000400001  PZE 000106003035  HTR 000000000000

                                .. FOLLOWING   129 CELLS ALL CONTAIN  HTR 000000000000 ..

      2727   HTR 000000254524  HTR 000000050164  HTR 000000632544  HTR 000000050154  HTR 000000264646  HTR 000000050153
      2735   PZE 002321434302  HTR 000000050152  HTR 000023214343  HTR 000000050151  MZE 434646470001  HTR 000000050150
      2743   HTR 000000000026  HTR 000000050147  HTR 000000000025  HTR 000000050146  HTR 000000000022  HTR 000000050145
      2751   HTR 000000004731  HTR 000000050144  HTR 000000000021  HTR 000000050143  TCO 006263215163  HTR 000000050142
      2757   HTR 000000002471  HTR 000000050137  HTR 000000000054  HTR 000000050134  HTR 000026432127  HTR 000000050132
      2765   HTR 000000000050  HTR 000000050130  TSX 007453135461  HTR 000000050127  HTR 000022434542  HTR 000000050126
      2773   RFT 005451254421  HTR 000000050125  HTR 000000000071  HTR 000000050121  HTR 000000000024  HTR 000000050115
      3001   HTR 000000000023  HTR 000000050114  HTR 000000446443  HTR 000000050113  HTR 000000446443  HTR 000000050112
      3007   HTR 000023464447  HTR 000000050036  HTR 000066465124  HTR 000000050022  HTR 000022316362  HTR 000000050021
      3015   HTR 000063306267  HTR 000000050020  HTR 000000464525  HTR 000000050017  HTR 000071255146  HTR 000000050016
      3023   HTR 000024464525  HTR 000000050015  HTR 000000004546  HTR 000000050013  HTR 000000702562  HTR 000000050005
      3031   HTR 000043464647  HTR 000000050003  PZE 002346644563  HTR 000000050000  TNX 606060606060
```

```
OCTAL DUMP OF ENDOP  FOLLOWS.
      3036   TTR 002100002465  TTR 002100007615  HTR 000000000000  MZE 473165215160  NZT 452000003161  TRA 002000400001
      3044   SXA 063400403054  CAL 450000003206  TSX 007400403036  TSX 007400403037  HTR 000000003056  LAC 053500403161
      3052   SXD 463400403053  TXI 177770103054  AXT 077400477512  TRA 002000400001  TNX 606325446060  TNX 605125446060
      3060   TNX 632544474651  TIX 215170606263  MZE 465121272560  TIX 215125216022  TIX 252731456260  TXH 302551253360
      3066   TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060
      3074   SXA 063400403101  AXT 077400400013  TNX 600001403101  SLW 060200403177  SXA 063400403075  AXT 077400474365
      3102   TRA 002000400001  SXA 063400403114  CAL 450000400001  SLW 060200003164  TSX 007400403037  HTR 000000003163
      3110   TSX 007400403116  AXT 077400427667  TXI 177777403113  SXA 063400403111  AXT 077400474363  TRA 002000400002
      3116   SXA 063400403125  AXT 077400400013  SXA 063400403075  AXT 077400400016  CAL 450000003203  SLW 060200403201
      3124   TIX 200001403123  AXT 077400474670  TRA 002000400001  CLA 050000003160  LDQ 056000003160  ADD 040000003201
      3132   STO 060100003160  CAS 034000003161  STO 060100003161  NOP 076100000000  XCA 013100000000  TNZ 410000003142
      3140   CAL 450000003206  TRA 002000400001  CAS 034000003202  TRA 002000003150  TRA 002000003150  ALS 076700000006
      3146   ORA 450100003205  TRA 002000400001  XCA 013100000000  PXD 475400000000  DVP 022100003202  RQL 477300000006
      3154   STQ 460000003162  ORA 450100003162  ORA 450100003204  TRA 002000400001  HTR 000000000010  HTR 000000000010
      3162   HTR 000000000000  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060
      3170   TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060
      3176   TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060  TNX 606060606060
      3204   TNX 632544200000  TNX 632544200060  TNX 632544606060  HTR 000000000001  HTR 000000000012  TNX 606060606060

OCTAL DUMP OF COMPOP FOLLOWS.
      3207   TTR 002100003111  TTR 002100003160  TTR 002100003116  TTR 002100003760  TTR 002100003074  TTR 002100003103
      3215   HTR 000000000000  TIX 234644474647  SXA 063400403417  SXA 063400203420  PXA 075400100000  STA 062160003207
      3223   CAL 450000400001  ACL 036100003745  STA 062100003241  STZ 060060003210  STZ 060000003425  CAL 450000003743
      3231   SLW 060200003424  TSX 007400403211  AXT 077400400310  SXA 063400403333  TOV 014000003236  TRA 002000003237
      3237   AXT 077400200012  AXT 077400100006  CAL 450000200315  LAS 434000003751  TRA 002000003245  TRA 002000003272
      3245   XCL 413000000000  PXD 475400000000  LGL 476300000006  STQ 460000003426  LAS 434000003747  TRA 002000003254
      3253   TRA 002000003270  AXT 077400400007  LAS 434000403304  TRA 002000003260  TRA 002000003304  TIX 200001403255
      3261   LGR 476500000006  CAL 450000003424  LGL 476300000006  TNO 414000003267  TSX 007400403332  CAL 450000003743
      3267   SLW 060200003424  LDQ 056000003426  TIX 200001103246  TIX 200001203240  STL 462500003425  TRA 002000003304
      3275   HTR 000000000020  HTR 000000000040  HTR 000000000054  HTR 000000000061  HTR 000000000074  HTR 000000000034
      3303   HTR 000000000013  SLW 060200003427  CAL 450000003424  LAS 434000003743  TRA 002000003311  TRA 002000003317
      3311   LDQ 056000003751  LGL 476300000006  TNO 414000003312  TSX 007400403332  CAL 450000003743  SLW 060200003424
      3317   ZET 052000003425  TRA 002000003343  CAL 450000003427  TSX 007400403332  CAL 450000003427  ERA 032200003744
      3325   TNZ 410000003270  CAL 450000003333  ACL 036100003743  STA 062100003430  TRA 002000003270  SXA 063400403337
      3333   AXT 077400400253  SLW 060200403743  TNX 600001403341  SXA 063400403333  AXT 077400474464  TRA 002000400001
      3341   SIL 405500000010  TRA 002000003417  CAL 450000003333  STA 062100003431  TRA 002000003346  LXA 053400103431
      3347   SXD 463400103364  LXA 053400103430  SXA 063400103432  TXI 177777103353  CAL 450000103743  TZE 010000003363
      3355   LAS 434000003750  TRA 002000003360  TRA 002000003366  LAS 434000003746  TRA 002000003363  TRA 002000003370
      3363   TXI 177777103364  TXH 300253103353  TRA 002000003401  SXA 063400103432  TRA 002000003363  SXD 463400103376
      3371   STZ 060000103743  LXA 053400103432  STZ 060000103743  SXA 063400103376  TSX 007400403212  PZE 000256000306
      3377   HTR 000000003743  TRA 002000003350  SXD 463400103406  LXA 053400103432  STZ 060000103743  SXA 063400103406
      3405   TSX 007400403212  PZE 000253700307  HTR 000000003743  AXT 077400100310  CAL 450000103743  TZE 010000003415
      3413   TSX 007400403213  TXI 177777103411  TSX 007400403214  TNX 606263466060  AXT 077400477520  AXT 077400200116
      3421   CAL 450060003207  PAX 073400100000  TRA 002000400002  HTR 000000000001  HTR 000000003274  HTR 000000000000
      3427   TNX 606060606060  HTR 000000000307  HTR 000000000253  HTR 000000000307  TIX 247160606060  HTR 000000000000

                                  .. FOLLOWING   198 CELLS ALL CONTAIN  HTR 000000000000 ..

      3743   HTR 000000000001  HTR 000000000013  HTR 000000000014  HTR 000000000034  HTR 000000000060  HTR 000000000074
      3751   TNX 606060606060

OCTAL DUMP OF EXPR   FOLLOWS.
      3752   TTR 002100004163  TTR 002100003074  TTR 002100003103  TTR 002100003127  HTR 000000000000  TIX 256747516060
      3760   SXA 063400404136  SXA 063400204137  SXA 063400104140  CAL 450000400001  STD 062200004026  STD 062200004070
      3766   STT 062500004143  STA 062100004037  PAX 073400100000  TXI 177777103772  SXA 063400104052  CAL 450000400002
      3774   STA 062100004007  STA 062100004054  STA 062100004040  STA 062100004056  SUB 040200004150  STA 062100004132
      4002   STZ 060000004144  STZ 060000004142  STZ 060000004145  STZ 060000004146  TRA 002000004007  CAL 450000103743
      4010   TZE 010000004025  LAS 434000004154  TRA 002000004014  TRA 002000004031  LAS 434000004153  TRA 002000004017
      4016   TRA 002000004031  LAS 434000004152  TRA 002000004022  TRA 002000004033  LAS 434000004151  TRA 002000004025
```

```
     4024   TRA 002000004033  TXI 177777104026  TXH 300253104007  STL 462500004144  TRA 002000004033  STL 462500004146
     4032   TRA 002000004025  NZT 452000004146  TRA 002000004042  SXD 463400104037  TSX 007400403752  PZE 000256000253
     4040   HTR 000000003743  STZ 060000004146  SXA 063400104037  NZT 452000004144  TRA 002000004025  TRA 002000004046
     4046   STZ 060000004145  CAL 450000004151  SLW 060200004142  STL 462500004147  AXT 077400100306  STZ 060000004144
     4054   CAL 450000103743  TZE 010000004067  STZ 060000103743  LAS 434000004152  TRA 002000004062  TRA 002000004073
     4062   LAS 434000004151  TRA 002000004065  TRA 002000004073  TSX 007400403753  STL 462500004145  TXI 177777104070
     4070   TXH 300253104054  STL 462500004144  TRA 002000004073  XCL 413000000000  CLA 050000004142  STQ 460000004142
     4076   ZET 052000004147  TRA 002000004111  ERA 032200004151  TZE 010000004103  TRA 002000004106  TSX 007400403754
     4104   TNX 602621246060  TRA 002000004125  TSX 007400403754  TNX 602662226060  TRA 002000004125  NZT 452000004145
     4112   TRA 002000004125  STZ 060000004147  ERA 032200004151  TZE 010000004117  TRA 002000004122  TSX 007400403754
     4120   TNX 602343216060  TRA 002000004125  TSX 007400403754  TNX 602343626060  TRA 002000004125  NZT 452000004144
     4126   TRA 002000004067  ZET 052000004143  TRA 002000004136  TSX 007400403755  SLW 060200103742  TSX 007400403753
     4134   TSX 007400403754  TNX 606263466060  AXT 077400474373  AXT 077400200001  AXT 077400100307  TRA 002000400003
     4142   TIX 264646606060  HTR 000000700000  HTR 000000004072  HTR 000000004067  HTR 000000000000  HTR 000000000000
     4150   HTR 000000000001  HTR 000000000020  HTR 000000000040  HTR 000000000054  HTR 000000000061
OCTAL DUMP OF TERM   FOLLOWS.
     4155   TTR 002100003074  TTR 002100003103  TTR 002100003116  TTR 002100003127  HTR 000000000000  TNX 632551446060
     4163   SXA 063400404310  SXA 063400204311  SXA 063400104312  CAL 450000400001  STD 062200004214  PAX 073400100000
     4171   TXI 177777104172  CAL 450000400002  STA 062100004201  STA 062100004203  SUB 040200004315  STA 062100004274
     4177   STA 062100004304  STZ 060000004314  CAL 450000103743  TZE 010000004213  STZ 060000103743  LAS 434000004317
     4205   TRA 002000004207  TRA 002000004240  LAS 434000004316  TRA 002000004212  TRA 002000004216  TSX 007400404155
     4213   TXI 177777104214  TXH 300256104201  TRA 002000004262  XCL 413000000000  CAL 450000004314  STQ 460000004314
     4221   TZE 010000004225  ERA 032200004316  TZE 010000004230  TRA 002000004235  TSX 007400404156  TNX 604324506060
     4227   TRA 002000004213  TSX 007400404156  TNX 602644476060  TSX 007400404156  TNX 606723216060  TRA 002000004213
     4235   TSX 007400404156  TNX 602624476060  TRA 002000004213  XCL 413000000000  CAL 450000004314  STQ 460000004314
     4243   TZE 010000004247  ERA 032200004316  TZE 010000004252  TRA 002000004255  TSX 007400404156  TNX 602343216060
     4251   TRA 002000004213  TSX 007400404156  TNX 602644476060  TRA 002000004213  TSX 007400404156  TNX 602624476060
     4257   TSX 007400404156  TNX 606723216060  TRA 002000004213  CAL 450000004314  TZE 010000004267  ERA 032200004316
     4265   TZE 010000004271  TRA 002000004301  TSX 007400404157  TRA 002000004310  TSX 007400404156  TNX 602644476060
     4273   TSX 007400404160  SLW 060200103742  TSX 007400404155  TSX 007400404156  TNX 606263466060  TRA 002000004310
     4301   TSX 007400404156  TNX 602624476060  TSX 007400404160  SLW 060200103742  TSX 007400404155  TSX 007400404156
     4307   TNX 606263506060  AXT 077400473742  AXT 077400200001  AXT 077400100256  TRA 002000400003  HTR 000000000054
     4315   HTR 000000000001  HTR 000000000054  HTR 000000000061
OCTAL DUMP OF (MAIN) FOLLOWS.
     4320   TTR 002100004331  TSX 007400404320  TXL 777777777777  TXL 777777777777  TXL 777777777777
OCTAL DUMP OF ASSEMBLED PROGRAM FOLLOWS.
    50000   LDQ 056000050016  STQ 460000050021  LXA 053400450020  PSE 076000000001  TRA 002000050013  SLW 060200050022
    50006   CAL 450000050021  ACL 036100050017  SLW 060200050021  CAL 450000050022  LGR 476500000001  TIX 200001450003
    50014   CAL 450000050021  TTR 002100070000  HTR 000000000000  HTR 000001000000  LAS 434000000044  HTR 000000000000
    50022   HTR 000000000000  HTR 000000000010  TRA 002000000000  HTR 000001000000  HTR 000002000000  MZE 400007000000
    50030   HTR 000000000000  HTR 000011000000  HTR 000000000000  CLA 050000050005  FAD 030000050003  STO 060100050013
    50036   LDQ 056000050143  FMP 026000050145  XCA 013100000000  FMP 026000050114  FDP 024100050115  XCA 013100000000
    50044   FDP 024100050146  XCA 013100000000  FDP 024100050132  STQ 460000050154  CLA 050000050154  STO 060100050121
    50052   TRA 002000050036  CLA 050000050150  FSB 030200050142  STO 060100050000  LDQ 056000050121  FMP 026000050143
    50060   STO 060100050154  CLA 050000050145  FAD 030000050154  STO 060100050155  LDQ 056000050121  FMP 026000050155
    50066   STO 060100050156  CLA 050000050114  FAD 030000050156  STO 060100050157  LDQ 056000050121  FMP 026000050157
    50074   STO 060100050160  CLA 050000050115  FAD 030000050160  STO 060100050161  LDQ 056000050121  FMP 026000050161
    50102   STO 060100050162  CLA 050000050146  FAD 030000050162  STO 060100050163  CLA 050000050163  FAD 030000050153
    50110   STO 060100050137  TIX 200001440517  HTR 000000000000  HTR 000000000000  HTR 000000047761  HTR 000000050114
    50116   SLW 060200000001  CLA 050000077777  HTR 000000000000  HTR 000000050154  STO 060100077762  SLW 060200030657
    50124   HTR 000000000014  HTR 000000050115  HTR 000000050126  HTR 000000000000  HTR 000000000000  HTR 000000050132
    50132   STO 060100031043  CLA 050000031043  HTR 000000000000  HTR 000000050121  FAD 030000050212  HTR 000000000021
    50140   FDP 024100050132  STO 060100050113  HTR 000000050030  TRA 002000050142  HTR 000000000126  TRA 002000050144
    50146   HTR 000000000005  HTR 000000000003  HTR 000000000002  HTR 000000050000  HTR 000000000000  CLA 050000000000

                                  .. FOLLOWING   147 CELLS ALL CONTAIN  HTR 000000000000 ..

    50377   HTR 000000000000
END OF POST MORTEM.
END OF RUN.
```

Appendix C

SUGGESTED ADDITIONS TO CAP

This appendix contains a list of suggested modifications to CAP which a student may attempt to make when using CAP as a laboratory exercise. With each modification is given a "point" value which is an indication of the relative difficulty of modification.

The descriptions of many of these additions make reference to similar facilities in FAP (FØRTRAN Assembly Program). Detailed information on the operation of the FAP facilities can be obtained from the FAP reference manual.*

C.1 Symbols

1. Add a test for multiply defined symbols and have CAP indicate with an M every operation involving a multiply defined symbol.	40 points
2. Sort the symbol table after PASS1. Beware, this is a difficult modification. If it fails, nothing else in CAP will work properly.	
a. Interchange sort.	40 points
b. Radix sort or any sort which takes a time comparable to N log N.	75 points
3. Use an exponential table lookup of the sorted symbol table for SYMGET.	75 points
4. Add the pseudo-op EQU which is to operate as in FAP. Check for phase errors, and indicate with a P.	50 points
5. Add a test to flag the eleven illegal characters in the location field. Indicate with an S.	35 points

C.2 Operation Field

1. Add the three-letter prefix codes to CAP as in FAP (that is, PZE, MZE, PØN, etc., and blank field).	25 points
2. Add the pseudo-op ØCT which accepts octal input in the same format as INT. Errors should be indicated with an E.	50 points
3. Add the pseudo-op BSS as in FAP. Check for phase errors and indicate with a P.	40 points
4. Add the pseudo-op HØL which accepts a card in the format of that in Figure C.1. n is a digit from 1 to 9, or if blank or 0 it is assumed to be 10.	50 points

*Reference Manual, FØRTRAN Assembly Program (FAP), IBM Publication C28-6235 (September, 1962).

HØL should then use n words of storage for BCI words as the FAP pseudo-op BCI does.

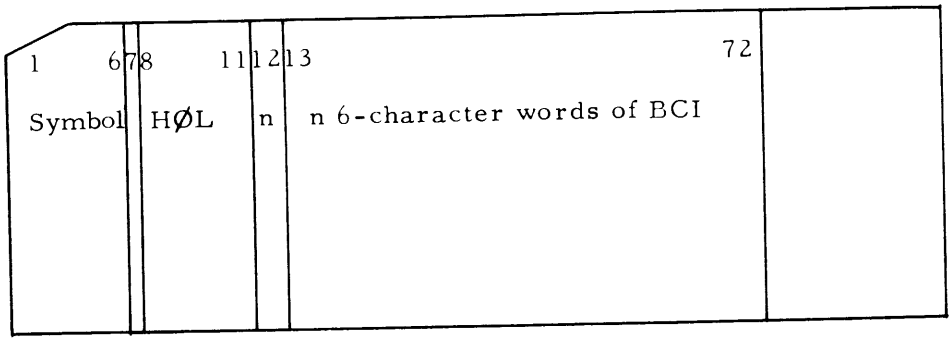


Figure C.1. Format for HØL pseudo-operations.

5. Add the pseudo-op CALL as in FAP except that: 50 points
   a. No transfer vector is formed.
   b. No error words are generated.

6. Allow for indirect addressing of operators with an asterisk. 25 points

7. Use an exponential table lookup for the op-table. (Only 25 points if you did this for the symbol table also.) 75 points

8. Improve the REM pseudo-op so that blanks replace the letters REM in the assembly listing. 25 points

C.3 Variable Field

1. Modify VAREVL to accept a "/" as a break character for division. Be careful of signs. 25 points

2. Modify CAP to consider "$" as a symbol (in the variable field) meaning "this location" as does the "*" in FAP. 40 points

3. Add decimal integer literals. 75 points

4. Modify CAP to accept a tag field and remove the present tags in ØPTBL. 25 points

5. Extend 4 so that CAP will also accept a decrement field and remove the present decrement in ØPTBL. 15 points

C.4 Assembly Listing

1. After the assembly listing, print a listing of symbols defined and their values. 40 points

2. Bonus for literals: After the symbol table, print a listing of literals. 40 points

3. Bonus for multiply defined symbols: Before the assembly listing, print a list of multiply defined symbols and their multiple values. 40 points

4. Bonus for symbol table: Form a table of undefined symbols and print after the assembly listing. 25 points

5. Print ØCT, INT, and CALL in detail mode. 25 points

6. Consider a * in column 1 to indicate a remark as in FAP. 25 points

7. Improve the assembly listing by separating the fields of the octal words, that is, CLA 64 should print as follows: 65 points

0500 00 0 00100

While TXL 1,1,1 should print

-3 00001 1 00001

and INT -32 should print

-000040 000000

while HØL 1 AB should print (if you have added HØL)

602122606060

8. Print the nonerror indications A, T, and D where applicable. For example, the letter A means either "an instruction normally written with an address does not have one" or "an instruction normally written without an address has one." Similarly for T (tag) and D (decrement). 40 points

9. Add the nonerror indications F and Q. F means "a nonindirectly addressable instruction has an indirect address." Q means "the instruction STØ (instead of the probable STQ) follows a divide instruction." 50 points

N.B.: In connection with these last three suggestions (and others) you may note that all operation codes are completely specified by the first four and the last four octal digits. Thus the middle four may be used in ØPTBL for A, T, D, and F information and for controlling printing of instructions. These middle four digits may be masked out of the opcode before inserting in the assembled program.

C.5 Compiler

1. Add diagnostics to CØMP including 75 points
   a. Nonzero reduction level.
   b. Illegal grammar, that is,

multiple "=" signs

A = C(B)

A = C) + (B

A = C + (*C)

2. Let column 7 be used for continuation cards in the same way that column 6 is in FØRTRAN, or column 11 is in MAD. 50 points

3. Add the operator ** to CØMP in such a way that A**B would be compiled as 75 points

```
CLA  A
LDQ  B
TSX  EXP3,4
```

and the result from EXP3 is left in the AC. The operator ** should be given proper precedence.

4. Modify the compiler to accept integer and floating point constants and form these into a table, say, LIT+00 to LIT+99 at the end of the program after TEM. To convert an integer of magnitude less than 2^{27} to floating point, the following sequence of 7090 instructions will work: | 90 points

```
CLA   INT              C(AC) = address integer
ØRA   =Ø233000000000   Put in exponent
FAD   =Ø233000000000   Normalize
STØ   FLT              C(FLT) = floating point equiva-
                       lent of the integer INT
```

5. Improve the efficiency of the compiler by reducing the number of combinations | 50 - 100 points

```
STØ   TEM+n
CLA   TEM+n
```

and replacing the combinations

```
STQ   TEM+n
CLA   TEM+n
```

and

```
STØ   TEM+n
LDQ   TEM+n
```

with

```
XCA
```