2-37. SUBPROGRAM U33 (U33). U33 loads itself (load tape) into core from tape A1. It then reads in and checks the tape search program and, if the transfer is proper, branches to execute the tape search program.

a. Inputs. No inputs are defined.

b. Outputs. No outputs are defined.

c. Program Logic. FD U33

(1) Steps 1-3. The bootstrap subprogram is read into core from tape A1 and the redundancy indicator is turned ØFF. The subprogram skips over the next record or loader subprogram and continues at step 4.

(2) Steps 4-12. The first three words of the record are brought into core. The first word, which contains the last address of the subprogram, is used to initialize the address of the first word in the tape search subprogram. The remainder of the subprogram is read with the checksum as the last word of the record. The logical checksum of the record is computed and compared with the actual checksum. If the redundancy indicator is ØN or if the checksums do not agree, the record is backspaced. If the subprogram is to proceed it continues at step 4. Otherwise the subprogram halts for manual intervention. If the redundancy indicator is ØFF and the checksums agree, the subprogram branches to execute the tape search subprogram.

2-68. SUBPROGRAM U51 (U51). U51 makes octal corrections to compiled subprograms or changes the values of parameters in Missile Simulator (SIM). The FORTRAN II reference statement is CALL U51. The FAP reference instruction is TSX U51, 4.

a. Inputs. The input is an octal coded card stored in the card image area CDIO - CDI13. The format of this card is illustrated in figure 2-1. Paragraph 2-209 illustrates the bit configuration in the card image area. Several corrections may be on one card provided they are located sequentially in core and are separated by an octal coded comma. An octal coded blank indicates that all the corrections have been processed.

b. Outputs. The outputs are the instructions or constants corrected in core.

c. Program Logic. FD U51

(1) Steps 1-3. The contents of index registers 1, 2, and 4 are saved. SENSE light 4 is set OFF, and the error indicator SW(70) is set ØFF. Zeros are stored in GRASE-1. These steps initialize U51.

(2) Steps 4-7. Card columns 2-6 (in CDIO) are examined column by column, and column count is updated. If any of the columns have zoning, control is transferred to step 19. If there is no zoning, each digit of the location for the first octal correction is stored in GRASE-1 as it is developed.

(3) Steps 8-10. Zeros are stored in GRASE-2, and the column count is updated to 11.

(4) Steps 11-12. The column count is updated. Starting with column 13 (in CDI2), each successive column is examined. If the column is zoned, control is transferred to step 14. Otherwise the subprogram is continued at the next step.

(5) Step 13. The digit of octal correction is stored in GRASE-2, and control is transferred to step 11 to examine the next column.

(6) Steps 14-18. If the zoning is a plus sign, control is transferred to step 11. If the zoning is a minus sign, the sign is stored in GRASE-2, and control is transferred to step 11. If the zoning is a blank, control is transferred to step 24. If the zoning is a comma, control is transferred to step 25. If the zoning is not one of the above four, it is an error and the subprogram is continued at the next step.

(7) Steps 19-22. SW(70) is set ØN, and a four is stored in the type of error indicator ITYER. The column count is examined. If the error occurred in columns 1-36, the column in error is indicated by setting to one the corresponding bit in CLEL. An error occurring in columns 37-72 is indicated in the same manner in CLER.

(8) Step 23. The contents of the index registers are restored and the subprogram exits to the user program.

(9) Step 24. SENSE light 4 is set ON to indicate that this is the last column to be checked.

(10) Steps 25-27. GRASE-1 is updated for the location in core of the next octal correction. The previous correction is stored in core at the address developed before GRASE-1 was updated. If SENSE light 4 is ON, control is transferred to FD step 23. Otherwise control is transferred to step 11 to examine the next column.

2-69. SUBPROGRAM (U52). U52 is a relocatable binary loader, and loads all share binary card formats except relocatable corrections. This loader will dump any number of subprograms on tape 4 by using a tape dump control card. Any number of these cards may be used. The card may also be used to change the load origin to the first location of any tape dump. Normal exit is with the entry address of the main program in the accumulator. An error exit is made with SW(70) ØN. This is caused by tape reading failure, invalid checksum, invalid card format, or tape dump errors.

a. Inputs. The source of input is determined by the state of SENSE switch 5. If ON, the cards are read off line (tape 10); if OFF, column binary cards are read on line.

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
ITYER	1	Indicator type of error	integer
IRECR	5	Tape record number	integer
IFILE	5	Tape file number	integer
ITBL	300	Table of Transfer vector locations	integer
STBL	300	Table of Transfer vector names	
SW(120)		Print register	
SW(70)		Error register	

The following printed and/or written statements are also outputs:

a. TAPE REDUNDANCY FAILURE

b. U08 OUTPUT ERROR 9L = ------ (word)
FIRST = ------ (address of first instruction of next subprogram)
COMMON = ------ (lowest address of Common used)

c. (lower dump address) NOT IN DECK

d. (lower dump address) NOT IN DECK

e. UNABLE TO FIND FIRST LOCATION FOR TAPE DUMP

f. TRANSFER VECTORS

g. SUBROUTINE ENTRY LOCATIONS

h. (lower dump address) NOT IN DECK

i. LOADING IS COMPLETE

j. TAPE DUMP ERROR 9L = ------ (word)
FIRST = ------ (address of first instruction of next subprogram)
COMMON = ------ (lowest address of Common used)

k. CHECKSUM FAILURE 9L = ------ (word)
FIRST = ------ (address of first instruction of next subprogram)
COMMON = ------ (lowest address of Common used)

l. END OF FILE TAPE A10 9L = ------ (word)
FIRST = ------ (address of first instruction of next subprogram)
COMMON = ------ (lowest address of Common used)

m. ILLEGAL CARD FORMAT 9L = ------ (word)
FIRST = ------ (address of first instruction of next subprogram)
COMMON = ------ (lowest address of Common used)

c. Program Logic. FD U52

(1) Steps 1-2. The tape file number is set to one and the tape record number is set to zero initially for a tape dump.

(2) Steps 3-6. If SENSE switch 5 is ON, the input is on tape and the subprogram continues at step 7. Otherwise column binary cards are read on line and the elements of the left row and right row are converted from column binary to row binary until all rows have been converted. The subprogram continues at step 16.

(3) Steps 7-13. Binary cards are read from tape unit B10. If the end-of-file indicator is ØN, ITYER is set to nine and the subprogram continues at step 122. If the end-of-file indicator and redundancy indicator are ØFF the subprogram continues at step 16. If the redundancy indicator is ØN, the record is backspaced and the subprogram continues at step 7. If failure to read occurs in seven attempts, U08 writes statement a and the subprogram continues at step 14.

(4) Steps 14-14.3. If SW(70) is ØFF, the subprogram continues at the step following the entry branch to this routine. If SW(70) is ØN, the first two characters of the 9 left word are converted to BCD in octal and U08 prints and writes statement b. The subprogram continues at step 15.

(5) Step 15. SW(70) is set ØN and the subprogram continues at step 117.

(6) Step 16-20. The card origin is picked up and the checksum of the 9 left word is saved. If the card is a

correction or transfer card the subprogram continues at step 45. If the card is an origin card the subprogram continues at step 43. If the prefix of the 9 left word is one or zero, or is greater than three, the subprogram continues at step 21. Otherwise the word count is added to the 7 left word for relocatable form and the subprogram continues at step 21.

(7) Steps 21-22. If the 9 right word checksum is zero, the subprogram continues at step 23. If the checksum of the 9 left word and the 9 right word do not agree the subprogram continues at step 120. Otherwise the subprogram continues at step 23.

(8) Steps 23-25. If the card is a name card the subprogram continues at step 34. Otherwise all SENSE lights are turned OFF. If the origin is absolute the subprogram continues at step 26. Otherwise the subprogram continues at step 28.

(9) Steps 26-27. Words in the card image are moved to their final positions. If the word count is less than zero the subprogram continues at step 3. Otherwise the subprogram continues at step 34.

(10) Steps 28-33. The address of the origin and the words in the card image are relocated. The decrement and address part of each word are examined and relocated if necessary. The subprogram continues at step 26.

(11) Steps 34-35. If SENSE light 1 is ON and the transfer vector is greater than or equal to zero the subprogram continues at step 38. Otherwise the subprogram continues at step 36.

(12) Steps 36-37. SENSE ~~switch~~ light 1 is set ON and the entry addresses of the transfer vector are relocated. The subprogram continues at step 26.

(13) Steps 38-42. The transfer vector is increased by one. If a tape dump has been executed, all subprograms loaded after the dump are counted. Instructions that relocate are modified for FORTRAN II convention. A new location of Common is computed and the lower storage location is updated with the subprogram size. The subprogram continues at step 36.

(14) Steps 43-44. If the origin card has a legal format the lower subprogram location is updated and the subprogram continues at step 3. Otherwise the subprogram continues at step 124.

(15) Steps 45-47. If loading is to cease, the subprogram continues at step 49. If the transfer or correction card does not have a legal format, the subprogram continues at step 124. Otherwise, the location of the correction or transfer card is updated and the subprogram continues at step 48.

(16) Step 48. If a dump card is to be processed the

subprogram continues at step 49. Otherwise the subprogram continues at step 74.

(17) Step 49. The internal switch is set to exit from pass two after hook-up and the subprogram continues at step 80.

(18) Steps 50-52. The internal switch is reset and the lower address of the dump is found. The list of subprogram names is searched for initial dump. If the subprogram is found the subprogram continues at step 59. Otherwise the subprogram continues at step 53.

(19) Steps 53-58. U08 writes statement c. The subprogram continues at step 14. SW(120) is set ØN and U08 prints statement d. The subprogram continues at step 14. SW(120) is set ØFF and the subprogram continues at step 15.

(20) Steps 59-65. Entry addresses for the subprograms are computed and matched with the stored entry address. If the entry addresses do not agree the subprogram continues at step 66. Otherwise the subprogram continues to match addresses until all the subprogram addresses have been processed. SW(120) is set ØN and U08 prints statement e. The subprogram checks for an error in U08 (step 14). ~~The subprogram continues at step 14.~~ If none occurred, SW(120) is set ØFF and the subprogram continues at step 25.

(21) Steps 66-73. If the 6 right word is not zero, the transfer vector is scanned and compared with the 6 right word. If the address does not match with one in the

list the subprogram continues at step 53. Otherwise the last address of the tape dump is set to the entry address to the subprogram. When the 6 right word is zero or when the last address of the tape dump has been established, the addresses in the calling sequence of U03 are set up, and U03 writes the subprogram or tape unit 4. If Sw(70) is ØN, the subprogram continues at step 118. Otherwise the transfer vector address is reset to link only the subprograms after the dump when loading is complete. The tape dump subprogram counter is reset to zero and the subprogram continues at step 3.

(22) Steps 74-79. If the transfer vector linkage is to be established, the subprogram continues at step 80. If the 9 left word is zero the subprogram continues at step 3. If the transfer address is to be relocated, the subprogram proceeds to relocate the transfer address. The transfer address is set for return to the calling sequence. If exit to the calling sequence is to occur, the subprogram continues at step 116. Otherwise the subprogram continues at step 3.

(23) Steps 80-86. If the internal register is ØN the subprogram continues at step 97. Otherwise U08 writes statement f and the subprogram continues at step 14. The transfer vector entry is converted to BCD in octal. If the table of transfer vector entries is complete, the subprogram continues at step 87. Otherwise U08 writes the transfer

vector entry and the subprogram continues at step 14. The subprogram continues to process all the entries.

(24) Steps 87-93. U08 blanks out the remainder of each line in the transfer list and the subprogram continues at step 14. If SENSE light 1 is OFF, the subprogram continues at step 97. Otherwise U08 writes statement g and the subprogram continues at step 14. SENSE light 1 is set ON and the subprogram name is converted to BCD in octal.

(25) Steps 94-96. If the table of subroutine names is complete the subprogram continues at step 87. Otherwise U08 writes the subprogram names and location and the subprogram continues at step 14. The subprogram continues to process each subprogram name and transfer address.

(26) Steps 97-98. A transfer convention is established and, if the last subprogram in the list is being processed, the subprogram continues at step 108. Otherwise the subprogram continues at step 99.

(27) Step 99. If the vector size is zero the subprogram continues at step 107. Otherwise the subprogram continues at step 100.

(28) Steps 100-104. If the subprogram name compares with the name in the transfer vector the subprogram continues at step 105. Otherwise the next entry in the transfer vector is compared until all entries have been processed. U08 writes statement h and the subprogram

continues at step 14. A transfer is set up and the subprogram continues at step 106.

(29) Step 105. A transfer to the subprogram is established and the subprogram continues at step 106.

(30) Step 106. If all vector names have been set up the subprogram continues at step 107. Otherwise the next name in the list is picked up and the subprogram continues at step 100.

(31) Steps 107-115. If links have not been established for all the vectors, the next vector is picked up and the subprogram continues at step 99. If loading is to continue, the subprogram continues at step 50. Otherwise the transfer vector is searched for the entry point to the main program. If the entry is found and normal exit is to occur, SW(120) is set ØN. If normal exit is not to take place at this time the subprogram continues at step 3. If the entry is not found, SW(120) is set ØN. U08 prints statement 1 and the subprogram continues at step 14. All indicators are reset to the original contents.

(32) Step 116. The entry address to the main program is set to the accumulator.

(33) Step 117. All index registers are restored to their original contents and control is returned to the user subprogram.

(34) Steps 118-119. Words are converted to BCD in octal and U08 writes and prints statement j. The subprogram continues at step 15.

(35) Steps 120-121. Words are converted to BCD in octal and U08 writes and prints statement k. The subprogram continues at step 15.

(36) Steps 122-123. Words are converted to BCD in octal and U08 writes and prints statement l. The subprogram continues at step 15.

(37) Steps 124-125. Words are converted to BCD in octal and U08 prints and writes statement m. The subprogram continues at step 15.

2-70. HARDCORE SUBPROGRAMS.

2-71. The subprograms described in this area initialize core, bring in the loader program, and perform tape duplication and/or comparison, depending on the setting of SENSE switches. The subprograms are as follows:

a.	TPSRCH	U02	Tape Search
b.	U03	U03	Binary Tape Write
c.	U04	U04	Binary Tape Read
d.	U08	U08	Print Card Image
e.	U34	U34	Generate Tape

"MOD F1"

2-72. SUBPROGRAM U02 (TPSRCH). TPSRCH is brought into core from tape 1 by the bootstrap program to initialize core with either the utility area or the TTP depending on the setting of SENSE switch 1. The FORTRAN II reference statement is CALL TPSRCH.

SENSE Switch	Setting	Operation
1	OFF	Areas A, B, B2 and Common of TTP are read
	ON	Utility programs are read as determined by the settings of SENSE switches 2, 3, 4, and 6
2	ON	The Loader is read in and executed
3	ON	The Tape Duplication program is read in and executed
4	ON	The Tape Compare program is read in and executed
6	ON	The RSDØRE dump tape is read into core

a. Inputs. No inputs are defined.

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
ØN	1	Used to set simulated switches	Hollerith
ØFF	1	Used to set simulated switches	Hollerith

The following printed and/or written statements are also outputs:

a. TAPE READ IN ERROR

b. ENTERED UTILITY CONTROL

c. BIN LOADER PROG SELECTED

d. TAPE DUPLIC PROG SELECTED

e. NO PROGRAM SELECTED

f. TAPE CMPARE PROG SELECTED

g. RSDORE DUMP PROG SELECTED

h. PROGRAM FINISHED

c. Program Logic. FD U02

(1) Steps 1-6. Four locations are used to simulate saving the index registers and the subprogram name. ØN and ØFF are initialized for simulation of SENSE switches. The floating point trap address is set to be used with FPT when underflow or overflow is detected. If SENSE switch 1 is OFF, U04 reads Area B2 into core from tape 1. If SENSE switch 1 is ON, the subprogram continues at step 15.

(2) Steps 6.01-6.04. If SW(70) is ØFF the subprogram returns to the step following the branch to this routine. If SW(70) is ØN, SW(120) is set ØN and the accumulator is set to ITYER. U08 prints statement a and the subprogram continues at step 34.

(3) Steps 7-14. U04 reads Area B1 from tape 1 and the subprogram continues at step 6.01. U04 reads Area A from tape 1 and the subprogram continues at step 6.01. U04 reads Common from tape 1 and the subprogram continues at step 6.01.

The B Common indicator SW(17) and the B subprogram indicator SW(33) are set ØN. The subprogram exits to DØCNT.

(4) Steps 15-16. SW(120) is set ØN and U08 prints statement b.

(5) Steps 17-18. If SENSE switch 2 is OFF, the subprogram continues at step 23. Otherwise SW(120) is set ØN and U08 prints statement c. The subprogram continues at step 19.

(6) Steps 19-22. U04 reads the Binary Loader Program from tape 1 and the subprogram continues at step 6.01. Tape 1 is rewound and the subprogram exits to Loader.

(7) Steps 23-28. If SENSE switch 3 is OFF, the subprogram continues at step 29. Otherwise SW(120) is set ØN and U08 prints statement d. U04 reads the Tape Duplication program into core from tape 1 and the subprogram continues at step 6.01. Tape 1 is rewound, DUPTP duplicates the tape, and the subprogram continues at step 35.

(8) Steps 29-34. If SENSE switch 4 is OFF, subprogram continues at step 39. If SENSE switch 4 is ON, SW(120) is set ØN and U08 prints statement f. U04 reads the Tape Compare subprogram from tape 1 and the subprogram continues at step 6.01. Tape 1 is rewound, CØMPTP compares the tapes, and the subprogram continues at step 35.

(9) Steps 35-36. SW(120) is set ØN and U08 prints

statement h.

(10) Steps 37-38. If the subprogram is to continue, it continues at step 17. Otherwise the program halts for manual intervention.

(11) Steps 39-45. If SENSE switch 6 if OFF, U08 prints statement e and control is transferred to step 37. Otherwise SW(120) is set ØN, U08 prints statement g, U04 reads the RSDØRE dump tape from tape 1, and the subprogram continues at step 6.01. Tape 1 is rewound, PRTCØN controls the decimal dump of the binary RSD tape, and control is transferred to step 35.

2-73. SUBPROGRAM U03, U04 (U03, U04). U03 writes a binary record on magnetic tape; U04 reads a binary record from magnetic tape. The subprogram descriptions are combined because several routines are common to both. However, the subprograms are used independently of each other. The FORTRAN II reference statement is CALL U03 (K, L, M, N) or CALL U04 (I, J, N).

a. Inputs.

(1) Inputs to U03 are a record in binary format, the address N of the register containing the address of the tape upon which the record is to be written, the first address K and last address L of the information to be written, and the location M of the instruction whose decrement contains the starting address of the subprogram. IRECR and IFILE contain the record number and file number respectively.

(2) Inputs to U04 are the addresses I and J of the registers containing the record number and file number of the binary record that is to be read and the address N of the register containing the address of the tape from which the record is to be read.

b. Outputs.

(1) The output of U03 is a binary record written on magnetic tape. The first word of the record contains the file number, record number, and the first address of information plus the number of words in the record. The second

word contains the transfer instruction, the number of words, and the starting address of the subprogram. The last word of the record is a checksum. If the end-of-tape is reached, the record consists of all zeros, and a five is stored in ITYER. If the record is written correctly, IRECR is stepped by one. If the record cannot be written correctly, a hash record (all sevens) is written and the error indicator SW(70) is set ØN.

(2) The output of U04 is a binary record read into the work area. If the record cannot be found or a checksum error occurs, a code of two or one, respectively, is stored in ITYER. If a tape redundancy occurs, the redundancy tape indicator SW(123) is set ØN. For all errors, SW(70) is set ØN.

c. Program Logic. FD U03, U04

(1) Steps 1-3. Enters at this point when U03 is operated. The exit address for U03 to return to the user subprogram is set up and the contents of index registers 1 and 2 are saved. The input-output operations are initialized for the tape unit that is to be used.

(2) Steps 4-6. Work area 1, work area-1, work area-2, work area-3, and ITYER are initialized to zero, and SW(70) and SW(123) are set ØFF. The second word of the record to be written is set up to consist of a transfer instruction, the

number of words, and the starting address of the subprogram and is stored in work area-1. The first word of the record to be written is set up to consist of the file number, record number, and the first location of information plus the number of words to be written and is stored in work area-2.

(3) Steps 7-13. The first and second words of the record are written on binary tape, and a checksum is accumulated. The end-of-tape indicator is interrogated and, if it is ØN, control is transferred to step 30. If this is not the end-of-tape, each remaining word of information is written and a checksum accumulated. The checksum is stored in work area-5 and is written as the last word of the record. If no tape redundancy occurred, control is transferred to step 16. If tape redundancy had not occurred before, the tape is backspaced, and control is transferred to step 7. If the redundancy had occurred before, the subprogram continues at the next step.

(4) Steps 14-15. The tape is backspaced and a hash record (all sevens) is written. Control is transferred to step 4.

(5) Steps 16-18. The tape is backspaced. The record is read, and a checksum is accumulated. If no tape redundancy occurred, control is transferred to step 24; otherwise the subprogram continues at the next step.

(6) Steps 19-20. Work area 1 is used as a counter and is interrogated to determine if ten attempts have been made to read the record. Until ten attempts have been made, control is transferred to step 16. After the tenth attempt, if the subprogram is performing a write (UO3) operation, control is tranferred to step 14; otherwise the subprogram is continued at the next step.

(7) Step 21. The tape redundancy indicator SW(123) is set ØN.

(8) Step 22. The error indicator SW(70) is set ØN.

(9) Step 23. The contents of index registers 1 and 2 are restored and the subprogram exits to the user subprogram.

(10) Step 24. If the checksum read in on tape agrees with the computed checksum, control is transferred to step 28; otherwise the subprogram is continued at the next step.

(11) Steps 25-26. Work area 1 is used as a counter to determine if ten attempts have been made to read the record. Until ten attempts have been made, control is transferred to step 16. After the tenth attempt, if the subprogram is performing a write (UO3) operation, control is transferred to step 14; otherwise the subprogram is continued at the next step.

(12) Step 27. A one is stored in ITYER, and control is transferred to step 22.

(13) Steps 28-29. If the computed checksum does not agree with the checksum stored in work area-5, control is transferred to step 25. If the checksums agree, IRECR is stepped by one and control is transferred to step 23.

(14) Step 30. The tape is backspaced.

(15) Steps 31-32. This is the physical end of the tape reel. Zeros are stored in work area 1. An end-of-tape record of zeros is written.

(16) Step 33. The tape is backspaced.

(17) Steps 34-44. The first and second words of the record are read. If an end-of-file is indicated, control is transferred to step 45. If this is an end-of-tape record, a 10 is stored in ITYER, the end-of-tape indicator is reset, and control is transferred to step 22. If this should have been an end-of-tape record, the tape is backspaced, a hash record (all sevens) is written, and control is transferred to step 31. If this is a hash record, control is transferred to step 34. If the word stored in work area-2 is zero, control is transferred to step 48. If this is not a hash record or end-of-tape record, and work area-2 is not zero, the record and file numbers requested are compared with the record and file numbers on the tape. If the numbers are equal, control is transferred to step 48. If the tape number is less than the requested number, control is transferred to step 34 to read a new record; otherwise the sub-

program continues at the next step.

(18) Steps 45-47. If this is the first time a wrong positioning of the tape or an end-of-file has occurred, the tape is rewound and control is transferred to step 34. If this is not the first time either error has occurred, a two is stored in ITYER and control is transferred to step 22.

(19) Steps 48-53. The remainder of the record is read. If a tape redundancy occurred, work area 1 is used as a counter to determine if ten attempts have been made to read the record. Until ten attempts have been made, control is transferred to step 33; after the tenth attempt, control is transferred to step 21. If a tape redundancy did not occur, a checksum is accumulated. If the checksum on the tape does agree with the accumulated checksum, control is transferred to step 23. If the checksums do not agree, work area 1, which is used as a counter, is interrogated to determine if ten attempts have been made to read the record. Until ten attempts have been made, control is transferred tc step 33; after the tenth attempt, control is transferred to step 27.

(20) Steps 54-58. Enter at this point when U04 is operated. The contents of index registers 1 and 2 are saved. The input-output operations are initialized for the tape that is to be used. Work area 1, work area-1, work area-2, work area-6, and ITYER are initialized to zero.

SW(70) and SW(123) are set ØFF. The exit address for U04 to return to the user subprogram is set up. The file number and record number are stored in work area-2, and control is transferred to step 34 to begin reading the record.

2-74. SUBPROGRAM U08 (U08). U08 prints information on-line or writes it on tape A7 for later use. The FORTRAN II reference statement is CALL U08. The FAP reference instruction is TSX U08, 4.

a. Inputs. The inputs are: a statement in BCD format and the first and last addresses of this statement. SW(120) is set ØN if the statement is to be printed and is set ØFF if the statement is to be written.

b. Outputs. If the statement is to be written, the output has the same format as the input and is written on tape A7. If the statement is to be printed the output is the statement in card image format (Hollerith coding).

c. Program Logic. FD U08

(1) Steps 1-5. The contents of index registers 1, 2, and 4 are saved. Tape redundancy error indicator SW(123), tape comparison error indicator SW(122), and error indicator SW(70) are set ØFF. The number of words in the statement (the last address of the statement plus one minus the first address of the statement) is computed and, if this is greater than zero, control is transferred to step 8. Otherwise a three is stored in ITYER.

(2) Step 6. Switch SW(70) is set ØN.

(3) Step 7. The contents of the index registers are restored and the subprogram exits to the user subprogram.

(4) Steps 8-9. SW(120) is interrogated by INTRØG. If SW(120) is ØN, the statement is to be printed and control is transferred to step 18. If SW(120) is ØFF, the statement is to be written and the subprogram continues at the next step.

(5) Steps 10-13. The information is written. If the physical end of the tape reel has been reached, a 10 is stored in ITYER and control is transferred to step 6. If no tape redundancy occurs, control is transferred to step 7. Otherwise the subprogram continues at the next step.

(6) Steps 14-17. Index register 2 is used to modify the transfer for a tape redundancy error. THE PRECEDING RECORD IS VOID is written the first time the redundancy occurs, and control is transferred to step 10. If the redundancy test has failed twice, SW(123) is set ØN, an eight is stored in ITYER, and control is transferred to step 6.

(7) Step 18. The first character of the statement is examined. If it is a carriage control character, the printer is set appropriately and the conversion begins with the second character; otherwise the conversion begins with the first character of the statement.

(8) Steps 19-20. Each character is converted from 6-bit BCD to Hollerith coding in a card image format. A maximum of 72 characters are converted and printed.

(9) Steps 21-23. The printed characters are echo

checked and, if they do not agree with the original characters, a seven is stored in ITYER and control is transferred to step 6. If the entire statement has been printed, control is transferred to step 7; otherwise control is transferred to step 19.

2-75. SUBPROGRAM U34 (U34). U34 writes the bootstrap and loader with its auxiliary subprograms on tape A4. The FORTRAN II reference statement is CALL U34.

a. Inputs. No inputs are defined.

b. Outputs. The outputs are the bootstrap and loader ensamble written on tape A4.

c. Program Logic. The end-of-file and tape redundancy indicators are turned ØFF. The bootstrap is written on tape A4. U03 writes the loader (record 63, file 0) on tape A4. If an error occurred in U03 (SW(70) = ØN), ITYER is displayed and the subprogram is halted for manual intervention. If SW(70) is ØFF, the subprogram returns to loader control.