```
0001  .MAIN
01
02                ;                                       RCSL:     44-RT 1612
03                ;                                       AUTHOR:   H.O.HANSEN
04                ;                                       EDITED:   78.03.03
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23                ;                       PROGRAM
24                ;                       *******
25
26
27                ;                   PRINTER TESTPROGRAM
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48                ;       KEYWORDS:   RC3600 LINE/SERIAL PRINTERS,STAND ALONE TEST-
49                ;                   PROGRAM.
50
51                ;       ABSTRACT:   THE TESTPROG. IS A DIAGNOSTIC- AND EXERCISER
52                ;                   PROGRAM, WHICH MAY BE USED TO CHECK AND DIAG-
53                ;                   NOSE THE CONTROLLERS AND PRINTERS: LPC 301,
54                ;                   302,303,304,305,306,307 AND RC3632,3633,3634,
55                ;                   3635,3636,3637,3638,3639,3641,3643.
56
57
58
59
60
61
62                ;       ASCII TAPE:     RCSL  44-RT 1613
63                ;       BINARY TAPE:    RCSL  44-RT 1546
64                ;       BINARY CARDS:   RCSL  44-RT 1614
```

```
| 0002  .MAIN
01                ;
02                ; THIS USER-GUIDANCE (PAGE 0-16) CONTENTS:
03                ;
04                ;               1.  ABSTRACT
05                ;               2.  MACHINE REQUIREMENTS
06                ;               3.  SWITCH-SETTINGS
07                ;               4.  OPERATING PROCEDURE
08                ;               5.  DESCRIPTION OF TESTLOOPS
09                ;               6.  ERROR-HANDLING
10                ;               7.  COMMUNICATION WITH THE PROGRAM (MESSAGES FROM
11                ;                   THE TEST)
12                ;               8.  PRINTERTABLE
13                ;               9.  PRINTEREQUIPMENT
14                ;
15                ;
16                ;
17                ; 1.ABSTRACT
18                ;
19                ;       THE DIAGNOSTIC-PART OF THE PROGRAM WILL TEST THE PRINTER-
20                ;       CONTROLLER AND DETECT HARDWAREFAILURES. IT IS FURTHERMORE
21                ;       POSSIBLE, IF WANTED, TO TEST THE SPECIAL PAPERCOMMAND-
22                ;       AND STATUSLOGIC OF THE PRINTER ITSELF.
23                ;       THE EXERCISER-PART OF THE PROGRAM IS USED TO PRINT PAGES
24                ;       WITH ANY INPUT'ED CHARACTER OR TO PRINT PAGES WITH ALL
25                ;       PRINTABLE CHARACTERS ROTATED TO ALL PRINTPOSITIONS.
26                ;
27                ;
28                ;
29                ; 2.MACHINE REQUIREMENTS
30                ;
31                ;       RC3600 FAMILY PROCESSOR
32                ;       8 K READ/WRITE CORE-MEMORY
33                ;       TELETYPE OR OPERATORS CONTROL PANEL
34                ;       RC 3600 LINE-OR SERIALPRINTER  -  SEE SEC. 9.
35                ;
36                ;
37                ;
38                ; 3.SWITCH-SETTINGS
39                ;
40                ;       3.1 STARTING ADDRESSES
41                ;
42                ;               400   TEST LOOP BUILDER  - TO RUN PART OF OR
43                ;                     COMPLETE TESTPROG.(SEC. 4.1)
44                ;               401   TEST LOOP BUILDER - TO RUN PART OF OR COM-
45                ;                     PLETE TESTPROG. - WITH INITIALIZED EQUIP-
46                ;                     MENT AND PRINTERCONSTANTS (SEC. 4.2)
47                ;               402   CONTROLLERDIAGNOSTICTEST WITH INITIALIZED
48                ;                     EQUIPMENT AND PRINTERCONSTANTS (SEC. 4.3)
49                ;               403   CHARACTERMOTION- AND CHARACTERROTATIONPRO-
50                ;                     GRAM WITH INITIALIZED EQUIPMENT AND PRIN-
51                ;                     TERCONSTANS (SEC. 4.4)
52                ;               404   RESTART OF THE RECENT DEFINED TESTLOOP
53                ;                     (SEC. 4.5)
54                ;               405   HAMMER ADJUST PROGRAM RC3632/33/34/35/36
55                ;               406   HAMMER ADJUST PROGRAM RC3641
56                ;
57                ;       3.2 SWITCHCONTROL
58                ;
59                ;               SW0=LOOP IN TEST IN CASE OF ERROR
60                ;               SW10=INHIBIT PRINTOUT
61                ;               SW11=PRINT FAILURE-RATE IN TESTLOOPPROGRAM
62                ;               SW12=NO HALT IN CASE OF ERROR IN TESTLOOP-PROGR.
63                ;               SW13=WAIT AFTER DIS MESSAGE.  DIS IS THE 16 CHAR
64                ;               SW14=HALT AFTER DIS MESSAGE.  DISPLAY AT OPERAT.
65                ;               SW15=CLEAR DIS AFTER MESSAGE. CONTROL PANEL.
```

```
I 0003  .MAIN
01                ;
02                ;       3.3 CPUNO
03                ;
04                ;               FOR TIMING PURPOSE THE PROGRAM DETERMINES IN
05                ;               WHICH CPU IT RESIDUES. IF IT FAILS, THE PROGRAM
06                ;               WILL TRY TO CONTINUE AFTER THE MESSAGE "MISE-
07                ;               RABLE TIMING".IF IMPOSSIBLE, THE PROGRAM WILL
08                ;               ASK YOU TO IDENTIFY THE CPU WITH A NUMBER BE-
09                ;               TWEEN 0 AND 6.INSERT THE NO. THIS WAY:
10                ;                       RESET
11                ;                       DEPOSIT # INTO AC2
12                ;                       CONTINUE
13                ;
14                ;               USE:            2 FOR NOVA 1200, RC3603-BREAK
15                ;                               4 FOR RC3603 WITHOUT BREAK
16                ;                               5 FOR NOVA 2 - 16K
17                ;                               6 FOR NOVA 2 - 8 K
18                ;               (THE MEMORY TYPE FOR THE FIRST 8K IS RELEVANT)
19                ;
20                ;
21                ; 4.OPERATING PROCEDURE
22                ;
23                ;
24                ;       LOAD PROGRAM USING THE BINARY LOADER, OR, WHEN IT RE-
25                ;       SIDES ON DISK, BY CALLING "LPTTEST" THROUGH HIPBOOT.
26                ;       AFTER LOADING, SOME MESSAGES WILL APPEAR (SEC.7.0), AND
27                ;       THE BINARY-LOADER-TS WILL BE CORRECTLY PLACED IN CORE.
28                ;
29                ;       THE PROGRAM CONTAINS FOUR PRINTERTABLES, USED BY THE
30                ;       EXERCISER-ROUT. BEFORE EXECUTION OF THE TESTLOOP, YOU
31                ;       CAN CHOOSE ONE OF THESE FOUR. WHEN YOUR PRINTER HAS AN-
32                ;       OTHER CHARACTERSET, AND THE EQUIPMENT CONTAINES A PTR,
33                ;       YOU CAN MAKE YOUR OWN TABLE (SEC. 8.) AND READ IT INTO
34                ;       CORE.
35                ;
36                ;  ***  IMPORTANT:
37                ;
38                ;       TO RUN THE TEST, SET DRUMSPEED-SWITCH TO "HIGH"-MODE (IF
39                ;       LPT IS A TWO-SPEED-PRINTER) FOR CORRECT EXECUTION OF
40                ;       THE TIMINGTEST-GROUP. SET LPT ON-LINE,AND SET ALL OTHER
41                ;       DEVICES, WITH CONNECTION TO THE CPU, OFF-LINE (EXCEPT
42                ;       TTY),TO INHIBIT UNWANTED INTERFERENCE. IF THE CONTROLLER
43                ;       IS LPC 306 YOU HAVE TO OPEN SWITCH S20,S5,S7,S11,S13 AND
44                ;       CLOSE S6,S8,S12 AND S14.TO START THE TEST YOU HAVE TO SET
45                ;        CPU-PANEL-SWITCHES TO A STARTLOCATION ACCORDING TO THE
46                ;       WANTED TESTPROG. PRESS EXAMINE  -  SET CONTROLSWITCHES
47                ;       -  AND PRESS CONTINUE.
48                ;
49                ;       THE EXPIRATION AFTER A START IS AS FOLLOWS:
50                ;
51                ;       THE QUESTIONMANEGER (FIRST PART OF PROGRAM) WILL ASK FOR
52                ;       EQUIPMENT-TYPE (IF WANTED) AND FOR WHICH PROGRAMPARTS
53                ;       IT HAVE TO INCLUDE IN THE PASS. WHEN THAT'S DEFINED
54                ;       THROUGH THE ANSWERS,THE TESTMANEGER (2. PART OF PROGRAM)
55                ;       WILL START THE TESTEXECUTION.
56                ;
57                ;       IN COMMUNICATION WITH THE PROGRAM ALL INPUTS TO THE
58                ;       QUESTIONMANEGER SHOULD BE TERMINATED WITH A CR,LF OR
59                ;       NL (TTI). WHEN YOU ARE USING DIS/NUK/FUB, THE TERMINATOR
60                ;       MAY BE EITHER NL OR CONT. CONT. HAVE TO BE USED WHEN THE
61                ;       CONTINUE-INDICATOR LIGTHS UP ON THE FUB-PANEL.
62                ;       IN CASE OF SITUATIONS, WHERE THE OPERATOR HAVE TO
63                ;       EXAMINE THE PRINTOUT OR TO CONSULT THE PROGRAMLISTING,
64                ;       THE CONDITION TO CONTINUE THE EXECUTION IS TO INPUT
65                ;       ONE OF THE MENTIONED TERMINATORS.
```

```
I 0004   .MAIN
01                ;
02                ;       4.0 STARTLOCATIONS (GENERAL)
03                ;
04                ;             PROGRAMSTART IN LOC.400 WILL GIVE YOU POSIBILITY
05                ;             TO DEFINE BOTH ALL THE EQUIPMENT-CONSTANTS
06                ;             (LIKE CONTROLLERNO. AND CHAR/LINE) AND THE
07                ;             CONTENT OF A TEST-PASS.
08                ;             PROGRAMSTART IN LOC.401 WILL GIVE YOU THE SAME
09                ;             POSIBILITY ACCORDING TO THE TEST-PASS, BUT
10                ;             NOW THE EQUIPMENT-CONSTANTS ARE INITIALIZED
11                ;             TO BE:
12                ;
13                ;
14                ;
15                ;          INITIALIZED EQUIPMENT-CONSTANTS:
16                ;
17                ;                   CONTROLLER = FIRST (I.E. DEV.NO.17 OR 37)
18                ;                   NO. OF CHANNELS IN VFU = 12 (8 WHEN
19                ;                   RC3637/38/39)
20                ;                   NO. OF CHAR. ON DRUM = 64
21                ;                   CHAR/LINE = 80
22                ;                   INPUTMODE TO EXERCISER = OCTAL
23                ;                   EXERCISER WILL NOT USE THE PRINTERTABLE
24                ;                   (EXCEPT FOR THE UPPER- AND LOWER HEADS)
25                ;
26                ;
27                ;             AFTER PROGRAMSTART IN LOC.402 YOU CAN EXECUTE
28                ;             A CONTROLLERTEST. THE EQUIPMENT-CONSTANTS ARE
29                ;             INITIALIZED AS MENTIONED ABOVE.
30                ;
31                ;
32                ;
33                ;
34                ;       4.1 TEST LOOP BUILDER (SA 400)
35                ;
36                ;             WITH THE TEST LOOP BUILDER YOU CAN BUILD-UP
37                ;             YOUR OWN TESTFIGURE, I.E. YOU HAVE THE POSSI-
38                ;             BILITY TO INITIALIZE THE PROGRAM WITH INPUTS
39                ;             AS: PRINTERTYPE,CONTROLLERTYPE,EQUIPMENTCON-
40                ;             STANTS AND TO DECIDE IF THE TEST HAVE TO RUN ALL,
41                ;             OR A PART OF THE TESTROUTINEGROUPS (FOR FURTHER
42                ;             INFORMATION, SEE SEC.7 - COMMUNICATION WITH THE
43                ;             PROG.).
44                ;             ALSO THE EXERCISERPROGRAMS (SEC. 4.4) MAY BE
45                ;             BUILD-IN AS A PART OF THE PROGRAM.
46                ;             AFTER HAVING SET THE SWITCHES ACCORDING TO
47                ;             SEC.3.2, THE TEST WILL START FOLLOWING THE SE-
48                ;             LECTED TESTFIGURE. IN CASE OF CONTROLLER- OR
49                ;             PRINTERFAILURES A LOCATION AND, IF WANTED, A
50                ;             PERCENT-VALUE WILL BE PRINTED. THE LOCATIONNO.
51                ;             CORRELATES A FAILURE-DESCRIPTION IN THE LIST-
52                ;             ING OF THE PROGRAM.
53                ;             HAVING TERMINATED THE RUN, THE PROGRAM WILL
54                ;             WRITE THE PASSNUMBER AND REEXECUTE THE TEST.
55                ;
56                ;
57                ;       4.2 TEST LOOP BUILDER WITH INITIAL. CONSTANTS (SA 401)
58                ;
59                ;             THE TEST-PROCEDURE IS IN THIS CASE JUST ALIKE
60                ;             THE ONE MENTIONED IN SEC. 4.1, BUT NOW YOU
61                ;             HAVE ONLY POSSIBILITY TO DEFINE THE TESTLOOP
62                ;             AND THE PRINTERTYPE,NOT THE CONTROLLERCONSTANTS
63                ;             NOR THE OTHER PRINTERCONSTANTS, WHICH ARE INI-
64                ;             TIALIZED (SEE SEC. 4.0)
```

```
| 0005  .MAIN
01                ;
02                ;       4.3 CONTROLLER-DIAGNOSTICTEST (SA 402)
03                ;
04                ;               THE TESTFIGURE IS NOW SET, SO THAT A PASS WILL
05                ;               CONSIST OF THE CONTROLLER-DIAGNOSTICTEST ONLY.
06                ;               THE PRINTER- AND CONTROLLERCONSTANTS ARE
07                ;               INITIALIZED - SEE SEC. 4.0. THESE TEST WILL
08                ;               RUN FULLY WITHOUT OPERATOR-INVOLVEMENT AND PA-
09                ;               PERCONSUMPTION, AND WILL BECAUSE OF THAT ONLY
10                ;               TEST A PART OF THE CONTROLLER- AND PRINTER-
11                ;               HARDWARE.
12                ;
13                ;
14                ;       4.4 PRINTEREXERCISER (SA 403)
15                ;
16                ;               THESE PROGRAM CONSISTS OF TWO PARTS, A CHARAC-
17                ;               TERPRINTINGPROGRAM, AND A CHARACTERROTATION-
18                ;               PROGRAM. IN BOTH CASES THE PRINTER- AND CON-
19                ;               TROLLERCONSTANTS ARE INITIALIZED (SEC. 4.0).
20                ;               WITH THE FIRST PROGRAM YOU CAN PRINT TWO PAGES
21                ;               (ONE IF RC3668/39/37) ON THE LPT, FILLED-UP
22                ;               WITH ANY CHARACTER, INPUT'ED FROM TTI/NUK.
23                ;               THE SECOND PROGRAM WILL PRINT TWO PAGES (ONE IF
24                ;               RC3638/39/37) WITH ALL PRINTABLE CHARACTERS RO-
25                ;               TATED TO ALL PRINTPOSSITIONS.
26                ;               AFTER A TERMINATION OF ONE OF THE TWO PARTS IT
27                ;               IS POSSIBLE TO REPEAT THE PREVIOUS EXECUTED PART,
28                ;               OR TO CONTINUE TO THE NEXT.
29                ;
30                ;
31                ;
32                ;       4.5 RESTART OF TESTLOOP (SA 404)
33                ;
34                ;               AT ANY TIME DURING THE EXECUTION OF A TESTLOOP,
35                ;               YOU CAN INTERRUPT THE PROGRAM BY USE OF CPU-
36                ;               HALT-SWITCH. AFTER A RESTART IN LOC. 404, THE
37                ;               ONCE BUILDED TESTLOOP THEN WILL BE EXECUTED
38                ;               FROM THE BEGINNING.
39                ;
40                ;
41                ;       4.6 HAMMER ADJUST PROGRAM (SA 405-406)
42                ;
43                ;               FOR USE DURING HAMMERBANK ADJUSTMENT, THESE ROU-
44                ;               TINES HAVE BEEN INCLUDED IN THE TEST. AFTER HAV-
45                ;               ING ANSWERED THE INITIALIZING QUESTIONS, THE
46                ;               PROGRAM STARTS MOVING THE HAMMERS. ONLY TWO HAM-
47                ;               MERS ARE MOVED.THE REF. HAMMER AND THE CURRENT
48                ;               HAMMER (= STARTHAMMER, FIRST TIME). EACH TIME THE
49                ;               ON/OFF-LINE SWITCH IS PRESSED, THE CURRENT HAMMER
50                ;               IS INCREMENTED WITH ONE.
51                ;
52                ;
53                ;
54                ; 5.DESCRIPTION OF TESTLOOPS
55                ;
56                ;
57                ;       THERE ARE TWO DIFFERENT WAYS TO USE THE ROUTINES FOR
58                ;       TESTLOOP:  SINGLE OR MULTIPLE ERRORHALT:
59                ;
60                ;       SINGLE:   SETP1          MULTIPLE:   SETP1
61                ;                 ERRORHALT                  ERRORHALT
62                ;                 LOOP                       ERRORHALT
63                ;                                            ERRORHALT
64                ;                                            LOOP
```

```
| 0006  .MAIN
01                ;
02                ;       IN CASE OF A CONSTANT ERROR THE RATE WILL BE PRINTED
03                ;       THIS WAY:
04                ;
05                ;       PC XXXXXX 100%                  PC XXXXXX 300%
06                ;       WHERE THE LAST IS A MULTIPLE OF 100%.
07                ;
08                ;       THERE ARE THREE POSSIBILITIES FOR THE NUMBER OF LOOPS IN
09                ;       A CYCLE,I.E. HOW MANY TIMES THE PROGRAM RUNS THROUGH THE
10                ;       INSTRUCTIONS BETWEEN SETP1 AND LOOP. IF SWITCH 0 IS 0
11                ;       THE FIRST TIME AN ERROR IS DETECTED, THE PROGRAM PRO-
12                ;       CEEDS TO NEXT INSTRUCTION AFTER LOOP.
13                ;
14                ;       NUMBER OF LOOPS:        SETP0:  1
15                ;                               SETP1:  10
16                ;                               SETP2:  100
17                ;
18                ;       IF THE ERROR IS NOT CONSTANTLY, IT IS POSSIBLE TO SEE
19                ;       THESE FAILURE RATES IF SINGLE OPERATION:
20                ;
21                ;       SETP0:    100% ERROR IN THE LOOP
22                ;       SETP1:    100,50,33,25,20,16,14,12,11,10 % FOR ERROR IN
23                ;                   1, 2, 3, 4, 5, 6, 7, 8, 9,10 -TH LOOP.
24                ;       SETP2:    LIKE SETP1, ONLY ADD 9...1 % FOR ERROR IN
25                ;                 11...100 -TH LOOP.
26                ;
27                ;       IF MULTIPLE OPERATION THE RATES DEPEND ON HOW MANY OF
28                ;       THE ERRORS ARE FOUND IN THE FIRST LOOP WITH ERROR:
29                ;                 100 % COULD BE 1 ERROR IN FIRST LOOP
30                ;                       OR       2 ERRORS IN SECOND LOOP
31                ;                       OR      3 ERRORS IN THIRD LOOP ETC.
32
33                ;       IF SWITCH 0 IS 1 THE PROGRAM WILL REMAIN IN THE LOOP WITH
34                ;       ERROR. FIRST TIME AN ERROR IS SEEN THE PROGRAM WILL
35                ;       HALT (IF NOT SWITCH 3). THEN THE PROGRAM CONTINUES UN-
36                ;       TILL ALL NUMBER OF LOOPS ARE PERFORMED. THEN A NEW CYCLE
37                ;       IS ENTERED CALLED ERRORCYCLE. AFTER EACH ERRORCYCLE A NEW
38                ;       IS STARTED UNTILL SWITCH 0 IS SET TO 0.
39
40                ;       IN ALL CYCLES (FIRST OR ERROR) THE PC (PROGRAMCOUNTER
41                ;       OF ERROR) IS WRITTEN FIRST TIME AN ERROR IS SEEN AND IN
42                ;       MULTIPLE OPERATION MORE THAN ONE ERRORHALT COULD WRITE
43                ;       THE PC. BUT ONLY WITHIN THAT FIRST LOOP WITH ERROR,
44                ;       FAILURERATE IS PRINTED WHEN THE CYCLE IS FINISHED. THE
45                ;       PROGRAM ONLY HALTS IN THE FIRST CYCLE (DEPENDING ON SW3)
46                ;       THE ERRORCYCLES HAVE ANOTHER AMOUNT OF LOOPS THAN FIRST
47                ;       CYCLE:
48                ;
49                ;       NUMBER OF LOOPS:        FIRST CYCLE     ERROR CYCLE
50                ;                       SETP0:      1               10
51                ;                       SETP1:      10              100
52                ;                       SETP2:      100             100
53                ;
54                ;       IN SINGLE OPERATION FAILURE RATE IS TELLING HOW OFTEN THE
55                ;       ERROR APPEARES. IN MULTIPLE OPERATION FAILURE RATE
56                ;       SHOULD BE USED VERY CAREFULLY.
```

```
| 0007  .MAIN
01                ;
02                ;
03                ; 6.ERROR-HANDLING
04                ;
05                ;
06                ;       IF THE TESTPROGRAM RECOGNIZES AN ERROR, IT WRITES THE
07                ;       CONTENTS OF 3 AC'S (FOR STATUS INFORMATION) AND "PC:
08                ;       XXXXXX" ON TTO OR DIS. THE NUMBER XXXXXX IS THE LOCA-
09                ;       TION IN THE PROGRAMLISTNING (PAGE 15-43), WHERE YOU CAN
10                ;       FIND THE ROUTINE WHICH DETECTED THE ERROR AND A SHORT
11                ;       DESCRIPTION.
12                ;       AFTER HAVING ANNOUNCED AN ERROR,THE TEST GOES ON IF SW3
13                ;       IS SET, AND IF IT IS NOT SET, THE PROGRAM HALTS.
14                ;       FOR PROCEEDING YOU HAVE TO ACTIVATE THE CPU-CONTINUE-
15                ;       SWITCH.THE PROGRAM NOW EITHER WILL LOOP 10 OR 100 TIMES
16                ;       IN THE "ERROR-ROUTINE", WRITE FAILURE-RATE OR CONTI-
17                ;       NUE TO NEXT TESTROUTINE, DEPENDING ON THE CONTROL-
18                ;       SWITCHSETTING.
19                ;
20                ;       IN CASE OF STATUSERROR YOU CAN COMPARE THE EXPECTED AND
21                ;       THE RECIEVED STATUS BY LOOKING AT THE CONTENT OF AC0(AC-
22                ;       TUAL STATUS) AND AC1 (EXPECTED STATUS),THE TWO STATUSWORDS
23                ;       NEEDS NOT TO BE EXACTLY ALIKE, BUT IN "STATA"-TESTS THE
24                ;       EXPECTED STATUSBIT(S) SHOULD BE A PART OF THE ACTUAL STA-
25                ;       TUSWORD, AND IN "STATN"-TESTS THE "EXPECTED"(CORRECTLY:
26                ;       NON-EXPECTED) BIT(S) SHOULD NOT BE INCLUDED IN THE
27                ;       WORD.
28                ;
29                ;       NOTICE: GENERALLY THE FIRST ERRORDETECTION IN A PASS
30                ;               GIVES THE MOST INFORMATION, BECAUSE OF THE FACT
31                ;               THAT THE FOLLOWING ERRORS CAN BE DUE TO THE
32                ;               FIRST ONE.
33                ;               THE PROGRAM MAY AFTER DETECTING AN ERROR AND
34                ;               RECEIVING A PROCEED-CONDITION GO TO AN INFI-
35                ;               NITE LOOP - WAITING FOR LPT-BUSY OR DONE. IF
36                ;               THIS HAPPENS, YOU HAVE TO STOP THE CPU AND RE-
37                ;               PAIR THE ERROR DETECTED FIRST, BEFORE RESTART
38                ;               OF THE TESTLOOP.
39                ;
40                ;
41                ;
42                ; 7.COMMUNICATION WITH THE PROGRAM (MESSAGES FROM THE TEST)
43                ;
44                ;
45                ; 7.0   MESSAGES FROM THE TEST
46                ;
47                ;
48                ;       DURING LOAD'ING,AFTER A START AND AFTER CHANGING THE CON-
49                ;       TROL-SWITCHES AN AMOUNT OF MESSAGES APPEARS,OF WICH SOME
50                ;       OR ALL MAY BE DISPLAYED. THESE MESSAGES ARE SHOWN IN THIS
51                ;       CHAPTER:
52                ;
53                ;
54                ;       1)      LOADING UNIT OFF
55                ;
56                ;               SET LOADING UNIT (DISC,MT) OFF-LINE TO INHIBIT
57                ;               UNWANTED INTERFERENCE DURING THE TESTEXECUTION.
58                ;
59                ;       2)      SWITCHES: XXXXXX
60                ;
61                ;               WHEN THE CONTROL-SWITCH-SETTING IS CHANGED AT
62                ;               ANY TIME DURING THE TEST, THIS MESSAGE WILL AP-
63                ;               PEAR, AND ANNOUNCE THE NEW SETTING.
```

```
| 0008  .MAIN
01                ;
02                ;       3)      MISSERABLE TIMING
03                ;
04                ;               RTC IS UNSTABLE
05                ;
06                ;               SET CPUNO > 2
07                ;
08                ;               WHEN SOMETHING GOES WRONG DURING THE TRY TO FIND
09                ;               THE CPU-TYPE, MORE OR ONE OF THE MESSAGES ABOVE
10                ;               IS TYPED. TO GET FURTHER INFORMATION ABOUT DE-
11                ;               FINING CPU-TYPE (SET CPUNO)  - SEE SEC. 3.3.
12                ;
13                ;       4)      LAST LOC XXXXXX
14                ;
15                ;               THE PROG. FINDS THE TOP OF MEMORY, AND ANNOUNCES
16                ;               THE RESULT WITH THIS MESSAGE.
17                ;
18                ;       5)      BINARY LOADER OK
19                ;
20                ;               AFTER LOAD'ING OF PROGRAM, A BINARY-PAPERTAPE-
21                ;               LOADER IS PLACED IN THE TOP OF CORE, WITH SA =
22                ;               LAST LOC.
23                ;
24                ;       6)      XXXXXX STARTADDR
25                ;
26                ;               WHEN STARTING THE TESTPROGRAM IN LOC 400 - LOC
27                ;               404, THE USED STARTADDRESS WILL BE ANNOUNCED.
28                ;
29                ;       7)      RC3600 LPT-TEST
30                ;
31                ;               THIS PROGRAM-NAME WILL APPEAR IN A LOT OF DIF-
32                ;               FERENT SITUATIONS, AND DOESN'T REQUIRE ACTION.
33                ;
34                ;
35                ; 7.1   QUESTIONMANEGER. THE QUESTIONMANEGER IS THE PART
36                ;       OF THE PROGRAM, WHICH AFTER A CORRECT START WILL
37                ;       INITIALIZE OR, IF WANTED, ASK FOR THE EQUIPMENTCON-
38                ;       STANTS, ETC. AFTER HAVING RECEIVED THIS DATA, THE
39                ;       PROGRAM IS INITIALIZED ACCORDING TO THE ANSWERS.
40                ;       BY USE OF THE QUESTIONMANEGER YOU CAN BUILD-UP
41                ;       YOUR OWN TESTLOOP, I.E. YOU CAN MASK-OUT UNWAN-
42                ;       TED TESTGROUPS, GO STRAIGHT TO THE EXERCISERROU-
43                ;       TINES ETC.
44                ;       WHEN ALL PRINTER/CONTROLLER-CONSTANTS ARE DEFINED,
45                ;       AND A TESTLOOP IS BUILDED-UP (BY THE OPERATOR,
46                ;       SA 400,SA 401, OR BY THE PROGRAM ITSELF, SA 402, SA
47                ;       403) THE CONTROL OF THE EXECUTION IS TAKEN OVER BY
48                ;       THE TESTMANEGER (SEC. 7.2).
49                ;       IN THE FOLLOWING SECTION ALL OUTPUTS FROM THE QUES-
50                ;       TIONMANEGER ARE LISTED, WITH THE POSSIBLE OPERATOR-AN-
51                ;       SWERS AND A DESCRIPTION. BOTH THE DISPLAY-MESSAGE (D)
52                ;       AND THE TTC-MESSAGE (T) ARE SHOWN.
53                ;
54                ;
55                ;       MESSAGES FROM QUESTIONMANEGER:
56                ;
57                ;
58                ;       1)      D: PRINTERTYPE ?
59                ;
60                ;               T: PRINTERTYPE: RC3632,3633(0),RC3634,3635(1),
61                ;                  RC3636(2),RC3641(3),RC3638(4),RC3639(5),
62                ;                  RC3637(6) OR RC3643(7) ?
63                ;
64                ;               DEFINE THE PRINTERTYPE BY INPUTING A NUMBER AS
65                ;               FOLLOWS:
```

```
| 0009  .MAIN
01                ;
02                ;                       0 - PRINTER IS RC3632 OR 3633 (LPT 704)
03                ;                       1 - PRINTER IS RC3634 OR 3635 (LPT 702)
04                ;                       2 - PRINTER IS RC3636 (LPT 701)
05                ;                       3 - PRINTER IS RC3641 (LPT 705/709)
06                ;                       4 - PRINTER IS RC3638 (LPT 706/708)
07                ;                       5 - PRINTER IS RC3639 (LPT 707)
08                ;                       6 - PRINTER IS RC3637 (LPT 719)
09                ;                       7 - PRINTER IS RC3643 (LPT 722)
10                ;
11                ;
12                ;       1A)     D: LPC306/7 OR NOT?
13                ;
14                ;               T: LPC306/307(1) OR NOT(0) ?
15                ;
16                ;               INFORM THE PROGRAM ABOUT CONTROLLER-TYPE.
17                ;
18                ;
19                ;       2)      D: CONTROLLERTYPE ?
20                ;
21                ;               T: FIRST(1) OR SECOND(2) CONTROLLER ?
22                ;
23                ;               AFTER ANSWERING THE QUESTION, THE PROGRAM WILL
24                ;               DEFINE THE DEVICENUMBER OF THE CONTROLLER AC-
25                ;               CORDING TO THE PREVIOUS DEFINED PRINTERTYPE.
26                ;
27                ;                       1 - FIRST CONTROLLER I.E. DEV.17 OR 37
28                ;                       2 - SECOND CONTROLLER I.E. DEV.57 OR 67
29                ;
30                ;
31                ;       3)      D: DIAGN.TST.OR NOT
32                ;
33                ;               T: DIAGNOSTICTEST(1) OR NOT(0) ?
34                ;
35                ;               IF YOU WANT TO RUN THE DIAGNOSTICPART OF THE
36                ;               PRCGRAM, INPUT A 1 -  IF NOT, INPUT A 0.
37                ;
38                ;
39                ;       4)      D: OPERT.TST OR NOT
40                ;
41                ;               T: TESTS WITH OPERATOR-INVOLVEMENT(1) OR NOT(0)
42                ;
43                ;               SOME OF THE TESTGROUPS NEEDS INVOLVEMENT FROM
44                ;               THE OPERATOR. IF YOU DON'T WANT TO INCLUDE THESE
45                ;               GROUPS IN THE TEST, THEN ANSWER  0, ELSE ANSWER
46                ;               1.
47                ;
48                ;
49                ;       5)      D: PAPERTEST OR NOT
50                ;
51                ;               T: TESTS WITH PAPERCONSUMPTION(1) OR NOT(0) ?
52                ;
53                ;               THE TESTGROUPS WHICH CHECKS THE PAPERFEED-,SKIP-
54                ;               LOGIK ETC. ARE ALL CONSUMING PAPER. IN LONGTIME-
55                ;               TESTS THIS MAY BE UNWANTED,THEREFORE YOU HAVE THE
56                ;               POSSIBILITY TO MASK-OUT THESE TESTS BY ANSWERING
57                ;               0 TO QUESTION 5.
58                ;
59                ;
60                ;       6)      D: NO.OF CHN.IN VFT
61                ;
62                ;               T: NO. OF CHANNELS IN VERTICAL FORMAT TAPE ?
63                ;
64                ;               DEFINE THE DECIMAL-NUMBER OF CHANNELS IN VER-
65                ;               TICAL-FORMAT-TAPE. I.E. 8 OR 12
```

```
| 0010  .MAIN
01                ;
02                ;        7)      D: CHA.PRINT OR NOT
03                ;
04                ;                T: CHARACTERPRINT-TEST(1) OR NOT(0) ?
05                ;
06                ;                THE EXERCISERPROGRAM, WITH WHICH YOU CAN PRINT
07                ;                PAGES WITH ANY INPUT'ED CHARACTER, MAY BE SE-
08
09                ;                LECTED BY ANSWERING 1 . IF THIS PROGRAM HAS NOT
10                ;                TO BE INCLUDE'D IN THE TEST-LOOP, ANSWER 0.
11                ;
12                ;
13                ;        8)      D: NO.OF CHAR/LINE?
14                ;
15                ;                T: NO. OF CHARACTERS PER LINE ?
16                ;
17                ;                THE CHARACTERPRINTING-ROUTINES NEEDS INFORMATION
18                ;                ABOUT THE CONTENTS OF CHAR. PER LINE. INPUT THE
19                ;                NUMBER IN DECIMAL.
20                ;
21                ;
22                ;        9)      D: USE P.TAB OR NOT
23                ;
24                ;                T: USE PRINTERTABLE(1) OR PASS-BY(0) ?
25                ;
26                ;                IN THE EXERCISERPROGRAMS ALL PRINTOUT WILL BE
27                ;                EXECUTED THROUGH THE LOADED (BY PROGRAM OR MANUAL
28                ;                PRINTERTABLE IF YOU ANSWER 1. IF YOU WANT TO
29                ;                PRINT SPECIAL CHARACTERS, WHICH MAY-BE WOULD BE
30                ;                CHANGED BY THE TABLE, YOU CAN PASS IT BY, BY
31                ;                ANSWERING 0. THE PRINTED CHARACTER(S) WILL THEN
32                ;                HAVE THE ABSOLUTE VALUE OF EITHER THE INPUT'ED ONE
33                ;                (CHARACTERPRINTROUTINE) OR OF THE STATE OF THE
34                ;                CHARACTERCOUNTER (CHARACTERROTATION-ROUTINE).
35                ;
36                ;
37                ;        10)     D: SELECT PR.TABLE
38                ;
39                ;                T: PRINTERTABLE: ASCII(0),STANDARD ST.(1),STAN-
40                ;                   DARD ST0(2),PL1,TYPE 70(3),READ FROM PTR(4) ?
41                ;
42                ;                EVEN IF YOU WANT TO PASS-BY THE PRINTERTABLE,
43                ;                YOU HAVE TO INPUT THE CORRECT NUMBER FOR FAULT-
44                ;                LESS PRINTING OF THE PAGE-HEADS IN EXECISER-PROG.
45                ;
46                ;                SELECT THE PRINTERTABLE (SEE SEC. 7.), ACCORDING
47                ;                TO THE CHARACTERSET ON DRUM, BY ANSWERING:
48                ;
49                ;                        0 - ASCII
50                ;                        1 - RC STANDARD, STARTING .
51                ;                        2 - RC STANDARD, STARTING 0
52                ;                        3 - PL1, TYPE 70
53                ;                        4 - TABLE HAS TO BE READ IN FROM PTR
54                ;
55                ;                IN CASE 0-3 THE QUESTIONMANEGER INSERTS THE ACTU-
56                ;                AL TABLE, AND CONTINUES THEN TO NEXT QUESTION.
57                ;                IN CASE 4, THE PROGRAM WRITES "LOAD PTR, STRIKE
58                ;                RETURN-KEY" OR "LDA PR-TYPE CONT". NOW ONE OF
59                ;                THE TERMINATIONS WILL START LOADING OF TAPE,AND
60                ;                TABLE-SELFSTART WILL TURN OVER CONTROL TO THE
61                ;                QUESTIONMANEGER AGAIN. ERRONEOUS LOADING (SUMERRO
62                ;                OR OVERWRITING) WILL CAUSE THE ERRORMESSAGE 11).
```

```
| 0011  ,MAIN
01                ;
02                ;       11)     D&T: CHECKSUM-ERROR
03                ;
04                ;               DURING THE LOADING OF A PRINTERTABLE FROM PTR,
05                ;               THE LOADER HAS DETECTED AN CHECKSUMERROR. TRY
06                ;               TO READ THE TAPE AGAIN FROM THE BEGINNING, BY
07                ;               RELOADING THE TAPE AND PRESSING CONT.,CR,NL OR
08                ;               LF. IF STILL UNSUCCESSFULL, YOU HAVE TO ASSEMBLE
09                ;               A NEW TAPE.
10                ;
11                ;       12)     D: ASC.OR OCT.INPUT
12                ;
13                ;               T: INPUTMODE = ASCII-ALPHABET(1) OR OCTAL(0) ?
14                ;
15                ;               BEFORE INPUT'ING A CHAR. TO THE EXERCISER YOU
16                ;               HAVE TO DECIDE THE INPUT-MODE.WHEN YOU'RE OPERA-
17                ;               TING FROM DIS/NUK YOU HAVE TO CHOOSE THE OCTAL-
18                ;               INPUT (0). WHEN YOU'RE USING TTY, YOU CAN CHOOSE
19                ;               ONE OF THE MODES, BUT INPUT'ING A CHARACTER IN
20                ;               ASCII (1) WILL BE THE EASIEST.
21                ;
22                ;
23                ;       13)     D: CHA.MOTIO. OR NOT
24                ;
25                ;               T. CHARACTERMOTION-TEST(1) OR NOT(0) ?
26                ;
27                ;               TO INCLUDE THE CHARACTER-ROTATION-PROGRAM IN THE
28                ;               PASS, YOU HAVE TO ANSWER 1. IF THIS PROGRAM IS
29                ;               NOT WANTED, ANSWER 0.
30                ;
31                ;
32                ;       14)     D: NO.OF CH.ON DRUM
33                ;
34                ;               T: NO. OF CHARACTERS ON DRUM ?
35                ;
36                ;               THE CHARACTER-ROTATION-PROGRAM NEEDS INFORMATION
37                ;               ABOUT HOW MANY DIFFERENT CHARACTERS IT HAS TO
38                ;               PRINT. THEREFORE YOU HAVE TO INPUT THE NUMBER OF
39                ;               CHARACTERS ON DRUM, I.E. 64 OR 96.
40                ;
41                ;       15)     D&T:  # OF HAMMERS=
42                ;
43                ;               INPUT THE TOTAL NUMBER OF HAMMERS ON THE PRINTER
44                ;               IN DECIMAL.
45                ;
46                ;       16)     D&T:  # OF ZONES=
47                ;
48                ;               INPUT THE NUMBER OF ZONES IN DECIMAL.
49                ;
50                ;       17)     D&T:  STARTHAMMER=
51                ;
52                ;               INPUT THE DECIMAL NUMBER OF THE HAMMER WITH WHICH
53                ;               YOU WANT TO START THE ADJUSTMENT.
54                ;
55                ;
56                ; 7.2   TESTMANEGER. IT'S THE TESTMANEGER WHICH IS RESPONSIBLE
57                ;       FOR THE EXEC. OF THE TESTLOOP,SET BY THE QUESTIONMANEGER.
58                ;       IT WILL RUN THE WANTED TESTGROUPS, WRITE ERRORLOCATIONS,
59                ;       COMMUNICATE WITH THE OPERATOR IN THE ROUTINEGROUPS WHERE
60                ;       OPERATOR-INVOLVEMENT IS NECESSARY, AND COUNT AND ANNOUN-
61                ;       CE THE PASS-NUMBER.
62                ;       IN THE FOLLOWING SECTION YOU CAN FIND THE MESSAGES FROM
63                ;       THE TESTMANEGER AND THE MEANING OF THEM.
64                ;
65                ;
66                ;        MESSAGES FROM TESTMANEGER:
```

```
| 0012  .MAIN
01                ;
02                ;       1)      D&T:  TESTEXECU. START
03                ;
04                ;               INDICATES, THAT THE INITIALIZATION IS FINISED
05                ;               AND THE EXECUTION OF DIAGNOSTICTEST WILL START
06                ;
07                ;       2)      D: EXAM. PRINTOUT     COMP. PC: XXXXXX
08                ;
09                ;               T: EXAMINE PRINTOUT, AND COMPARE WITH PC: XXXXXX
10                ;
11                ;               WHEN THIS OUTPUT APPEARS,YOU HAVE TO TAKE-OUT
12                ;               THE PRINTED FORMS FROM THE LPT AND COMPARE IT
13                ;               WITH THE DESCRIPTION IN THE TESTLISTNING AT PC:
14                ;               XXXXXX. FOR PROCEEDING TYPE CR,LF,NL OR CONT.
15                ;
16                ;
17                ;       3)      D&T: OPEN DRUMGATE
18                ;
19                ;               OPEN THE DRUMGATE OF THE PRINTER,TYPE CR,LF,NL
20                ;               OR CONT., AND WAIT FOR NEXT MESSAGE.
21                ;
22                ;       4)      D: CL.DRG-LOCAL LPT
23                ;
24                ;               T: CLOSE DRUMGATE - SET LPT IN LOCAL STATE
25                ;
26                ;               THE TTO-MESSAGE OUGHT TO BE EXPLANETION ENOUGH,
27                ;               FOR PROCEED'ING TYPE CR,LF,NL OR CONT.
28                ;
29                ;
30                ;       5)      D: LINE-LEAV,1 FORM
31                ;
32                ;               T: SET LPT ON LINE - REMOVE PAPER EXCEPT ONE
33                ;               FORM
34                ;
35                ;               ALSO HERE THE TTO-MESSAGE EXPLAINS ALMOST EVERY-
36                ;               THING. THE LOWER PAPERSWITCH HAVE TO BE FREE,
37                ;               FOR PROCEEDING - TYPE CR,LF,NL OR CONT.
38                ;
39                ;
40                ;       6)      D&T: REMOVE PAPER
41                ;
42                ;               TAKE-OUT ALL THE PAPER FROM THE RESERVOIR,IF IT
43                ;               IS NOT ALREADY RUN THROUGH. TYPE,LF,NL OR CONT.
44                ;
45                ;
46                ;       7)      D&T: MAKE LPT USEABLE
47                ;
48                ;               INSTALL THE PAPER CORRECTLY, SET PRINTER ON-LINE
49                ;               AND TYPE CR,LF,NL OR CONT. TO EXECUTE THE PRIN-
50                ;               TING-ROUTINES OR TO REPEAT THE PASS.
51                ;
52                ;
53                ;       8)      D&T: DID BELL RING ?
54                ;
55                ;               A BELL-COMMAND (RC3638/39) SHOULD MAKE THE PRIN-
56                ;               TERBELL RING FOR 2 SEC. IF IT DIDN'T RING - A
57                ;               HARDWAREFAILURE IN THE BELL-CIRCUIT IS DETECTED.
58                ;
59                ;
60                ;       9)      D&T: DESELECT PRINTER
61                ;
62                ;               RC3637/38/39 ONLY. SET PRINTER OFF-LINE BY PRESS-
63                ;               ING THE "DESELECT"-BOTTON.
```

```
| 0013  .MAIN
01
02                ;        10)     D: SEL.PR-REMOV.PAP
03                ;
04                ;                T: SELECT LPT - REMOVE PAPER
05                ;
06                ;                THE TTO-OUTPUT EXPLAINS EVERYTHING. ALSO THIS
07                ;                STRING IS OUT-PUT'ED IN CASE OF TESTING RC3637/
08                ;                38/39 ONLY.
09                ;
10                ;
11                ;        11)     D&T: REPLACE PAPER
12                ;
13                ;                INSTALL THE PAPER CORRECTLY, AND SET PRINTER
14                ;                ON-LINE.
15                ;
16                ;
17                ;        12)     D: INPUT CHAR.
18                ;
19                ;                T: INPUT CHARACTER
20                ;
21                ;                INPUT THE CHARACTER TO THE PRINT-CHARACTER-
22                ;                PROGRAM IN THE DECIDED INPUT-MODE. TERMINATE
23                ;                WITH NL.
24                ;
25                ;        13)     D&T: LPT ISN'T AVAIL.
26                ;
27                ;                IN THE EXERCISERROUTINES THE LPT-STATUS IS CHECK-
28                ;                ED AND IF NOT OK, THIS MESSAGE IS PRINTED. AFTER
29                ;                TYPING THIS STRING THE PROGRAM WAITS UNTIL THE
30                ;                OPERATOR ACTIVATES CR,LF,NL (TTI) OR CONT(NUK).
31                ;                EXAMINE THE PRINTERCONTROL-PANEL (IS IT ON-LINE?),
32                ;                THE PAPERFEED ETC. WHEN THE OPERATOR MAKES THE
33                ;                PROGRAM CONTINUE, THE STATUS IS CHECKED AGAIN.
34                ;                IF OK, THE PROG. GOES ON, IF STILL NOT,"HARDWARE-
35                ;                FAILURE" IS TYPED.
36                ;
37                ;
38                ;        14)     D&T: HARDWARE-FAILURE
39                ;
40                ;                THE EXERCISERPROGRAMS CAN'T BE EXECUTED BECAUSE
41                ;                OF A CONTROLLERFAILURE. THE PROGRAM HALTS, AND
42                ;                YOU MAY NOW START THE DIAGNOSTICTEST, TO DETECT
43                ;                THE FAILURE.
44                ;
45                ;
46                ;        15)     D&T:  OFF-LINE/ON-LINE
47                ;
48                ;                IN CASE OF BUFFER-CLEAR-FAILURE THIS MESSAGE AP-
49                ;                PEARS. THE OPERATOR HAVE TO CLEAR THE BUFFER BY
50                ;                SELECTING THE PRINTER OFF-LINE AND THEN ON-LINE.
51                ;                TYPE NL OR CONT. TO CONTINUE.
52                ;
53                ;
54                ;        16)     D: REPEAT OR CONT.?
55                ;
56                ;                T: REPEAT PRINTROUTINE(1) OR CONTINUE(0) ?
57                ;
58                ;                AFTER HAVING EXECUTED ONE OF THE TWO EXERCISER-
59                ;                ROUTINES THIS QUESTION IS TYPED. BY ANSWERING 1
60                ;                YOU WILL REEXECUTE THE JUST FINISHED EXERCISER-
61                ;                ROUTINE AND BY ANSWERING 0, YOU WILL PROCEED IN
62                ;                THE TESTLOOP.
```

```
01
02                ;        17)      D&T: END OF PASS: XXXXXX
03                ;
04                ;                 WHEN THE TESTMANEGER HAS FINISHED A PASS THIS
05                ;                 TEXT WILL APPEAR - AND A RESTART OF THE BUIL-
06                ;                 DED TESTLOOP WILL BE EXECUTED.
07                ;
08                ;
09                ;        18)      D&T: SET LPT ON-LINE
10                ;
11                ;                 IF LPT-STATUS AFTER A PASS IS = OFF-LINE, THIS
12                ;                 MESSAGE IS TYPED. THE OPERATOR HAS TO SET PRINTER
13                ;                 ON-LINE AND THEN INPUT A NL(TTI) OR CONT.(NUK).
14                ;
15                ;
16                ;        19)      D&T:  REF.HAMM.= XXXXXX
17                ;
18                ;                 AFTER HAVING INITIALIZED THE HAMMER ADJUST PROG.
19                ;                 AND DURING THE RUN, THE PROGRAM WILL ANNOUNCE
20                ;                 WHICH HAMMER HAS TO BE USED AS REFERENCE HAMMER
21                ;                 FOR SCOPING. THE REF. HAMMER WILL DIFFER ACCOR-
22                ;                 DINGLY TO THE CURRENT HAMMER IN MULTI-ZONE
23                ;                 PRINTERS.
24                ;
25                ;
26                ;
27                ; 8.PRINTERTABLE
28                ;
29                ;UNDER CONTROL OF THE QUESTIONMANEGER IT IS POSSIBLE TO MAKE THE
30                ;PRT.TABLE MATCH THE ALPHABET USED ON THE LINEPRINTER (SEC.7.1-
31                ;QUESTION 10). IF YOU CAN'T USE ONE OF THE FOUR MENTIONED BELOW,
32                ;YOU HAVE TO PRODUCE A BINARY PAPERTAPE, CONTAINING A TABLE AS
33                ;DESCRIBED IN THIS SECTION.
34                ;
35                ;EXISTING:       44-RT 535        ASCII
36                ;                44-RT 529        RC STANDARDTYPE 71/78 STARTING .
37                ;                44-RT 532        RC STANDARDTYPE 71/78 STARTING 0
38                ;                                          (SKEWED 4 POSITIONS)
39                ;                44-RT 1213       PL 1, TYPE 70
40                ;
41                ;
42                ;HOW TO PRODUCE A NEW TABLE:
43                ;
44                ;THE TABLE HAS 200 OKTAL (128 DECIMAL) BOXES. INPUT KEY
45                ;IS THE ASCII VALUE OF THE CHARACTER TO PRINT ADDED TO 2000.
46                ;THE RESULT IS ADDRESS OF A BOX. EACH BOX OCCUPIES A CORE WORD.
47                ;IT IS BUILT UP OF TWO FIGURES, THE FIRST IS THE CLASS OF THE
48                ;CHARACTER TO BE PRINTED: 0 FOR PRINT, 6 FOR BLIND. THE SECOND
49                ;IS THE CHARACTER VALUE AT THE PRESENT PRINT DRUM, BELOW
50                ;VALUE 40 OCTAL FOLLOWING CHARACTERS MAY BE USED: 11 TAB,
51                ;12 LF, 14 FF AND 15 CR. ALL OTHERS BELOW 40 WILL GIVE SPACE.
52                ;
53                ;
54                ;THE FIRST BOX SHOULD CONTAIN THE VALUE FOR THE NULL CHAR
55                ;AND THE LAST THE VALUE FOR THE DEL CHAR, WHICH BOTH NORMALLY
56                ;ARE BLIND. IF YOU COUNT 0,1,2, ,7,10, , THE BOX 101 SHALL
57                ;CONTAIN THE PRINT DRUM VALUE FOR AN A. IF THE DRUM DO NOT
58                ;HAVE SMALL LETTERS, FILL IN THE VALUE FOR BIG ONES. NOW PUNCH
59                ;AN ASCII TAPE LIKE THIS:
```

```
| 0015  .MAIN
01
02                ;    .LOC 2000
03                ;    .RDX 8          ;WHICH RDX YOU WANT
04                ;    .TXTE?
05                ;    <6><0>          ;(2000) FIRST BOX, BOX 0
06                ;    <6><0>
07                ;    .
08                ;    .
09                ;    .
10                ;    <0><101>        ;(2101) BOX 101 FOR A, FOR ASCII DRUM
11                ;    .                ;101 IS USED, FOR TYPE 71 137 IS USED.
12                ;    .
13                ;    .
14                ;    <6><0>?         ;(2177) BOX 177, LAST
15                ;    .RDX 8
16                ;    .END 101
17                ;
18                ;
19                ;PRODUCE A BINARY TAPE, AND LOAD THIS TO MEMORY, WHEN QUE-
20                ;STIONMANEGER TYPES: "LOAD PTR, CONT.".
21                ;
22                ;
23                ;
24                ;
25                ;
26                ; 9.PRINTEREQUIPMENT
27                ;
28                ;
29                ;        THE PRINTER CAN BE ANYONE OF THE RC3600 SERIAL-,LINEPRIN-
30                ;        TERSERIES,WITH EXCEPTION OF THE CHARABAND-PRINTER RC3645
31                ;
32                ;        THE PRINTERCONTROLLER CAN BE EITHER LPC 301,302,303,304
33                ;        305 OR THE UNIVERSAL ONES, LPC 306, LPC307
34                ;
35                ;        TO RUN THE DIAGNOSTICROUTINES WITH OPERATOR-INVOLVEMENT
36                ;        AND PAPERCONSUMPTION ON A RC3600 LINEPRINTER, A SPECIAL
37                ;        TAPE-LOOP (VFT) TO THE TAPELOOP-READER (VFU) MUST BE PRE-
38                ;        PARED AS SHOWN:
39                ;
40                ;        RC 3632/33/34/35/36/41/43:
41                ;
42                ;
43                ;                CH       CH              CH
44                ;                12        8               1
45                ;
46                ;                 . . . . . . . . . . . . 0
47                ;                 . . . . . . . . . . . 0 .
48                ;                 . . . . . . . . . . 0 . .
49                ;                 . . . . . . . . . 0 . . .
50                ;                 . . . . . . . . 0 . . . .
51                ;                 . . . . . . . 0 . . . . .
52                ;                 . . . . . . 0 . . . . . .
53                ;                 . . . . . 0 . . . . . . .
54                ;                 . . . . 0 . . . . . . . .
55                ;                 . . . 0 . . . . . . . . .
56                ;                 . . 0 . . . . . . . . . .
57                ;                 . 0 . . . . . . . . . . .
                  ;                 0 . . . . . . . . . . . .
```

```
| 0016  .MAIN
01
02                ;       THE FIGURE IS MADE FOR A 12-CHANNELREADER, BUT THE
03                ;       LOOKING FOR AN 8-CHANNEL-TAPE IS JUST ALIKE, WITH EX-
04                ;       CEPTION OF THE NUMBER OF CHANNELS.
05                ;
06                ;       RC 3638/39:
07                ;
08                ;
09                ;                 CH       CH     CH
10                ;                  1        5      8
11                ;
12                ;                  0 0 0 0 0 0 0 0          ; FF
13                ;                  . . . . 0 . . .
14                ;                  . . . . 0 . . .
15                ;                  . . . . 0 . . .          ; VT
16                ;                  . . . . 0 . . .
17                ;                  . . . . 0 . . .
18                ;                  . . . . 0 . . .
19                ;                  . . . . 0 . . .
20                ;                  . . . . 0 . . .
21                ;                  . . . . 0 . . .
22                ;                  . . . . 0 . . .
23                ;                  . . . . 0 . . .
24                ;
25                ;       RC 3637:
26                ;
27                ;                 CHCH     CH     CH
28                ;                  1 2      5      8
29                ;
30                ;                  0 0 0 0 0 . . 0          ; FF
31                ;                  . 0 0 . 0 . . 0
32                ;                  . 0 0 . 0 . . 0
33                ;                  . 0 0 . 0 . . 0
34                ;                  . 0 0 . 0 . . 0          ; VT
35                ;                  . 0 0 . 0 . . 0
36                ;                  . 0 0 . 0 . . 0
37                ;                  . 0 0 . 0 . . 0
38                ;                  . 0 0 . 0 . . 0
39                ;                  . 0 0 . 0 . . 0
40                ;                  . 0 0 . 0 . . 0
41                ;                  . 0 0 . 0 . . 0
42
43
44
45                ;
46                ;END OF PROGRAM-DESCRIPTION
```

```
| 0017  .MAIN
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40                ;TAPE 2          PAGE ZERO FOR TAPE 3,4,5
41
42        000000 .LOC 0
43
44 00000 003160           2*MELOC          ;MESS AFTER RDOS LOAD AND STORE PC ON INT!
45 00001 000000           0                ;ADDR OF INTR. SERVICE ROUTINE
46 00002 001400           REBIN            ;SELFSTART ADDR FOR RDOS ETC.
47 00003 000000           0                ;0=HALT, 1=SELFSTART PROG AFTER REBIN
48 00004 000000           0                ;@ADDR FOR SELFSTART PROG AFTER REBIN
49 00005 000000           0                ;FOR LOAD RDOS, USED BY POW. INTR, FITYP
50
51        000020 .LOC 20
52
53 00020 000000 IDX0:     0                ;AUTO INCREMENT LOCATION
54 00021 000000 IDX1:     0                ;AUTO INCREMENT LOCATION
55 00022 000000 IDX2:     0                ;AUTO INCREMENT LOCATION
56 00023 000000 IDX3:     0                ;AUTO INCREMENT LOCATION
```

```
| 0018  .MAIN
01
02          000040 .LOC 40                  ;INDIRECT ADDRESSES
03                                          ;NOT IN AUTO INC,DEC LOC.
04 00040 000412 IMESS:  XMESS
05 00041 000664 ICHAR:  XCHAR
06 00042 000724 ITYPE:  XTYPE
07 00043 001043 ICRLF:  XCRLF
08 00044 001104 IDISP:  XDISP
09 00045 001134 IDOUT:  XDOUT
10 00046 001165 IDICL:  XDICL
11 00047 001176 IDATT:  DISATT
12 00050 001221 IHAAT:  HAATT
13 00051 001014 ITBIN:  XTBIN
14 00052 000560 ITOCT:  XTOCT
15 00053 000470 ITDEC:  XTDEC
16 00054 000550 ITZOC:  XTZOC
17 00055 001020 IDBIN:  XDBIN
18 00056 000564 IDOCT:  XDOCT
19 00057 000464 IDDEC:  XDDEC
20 00060 000554 IDZOC:  XDZOC
21 00061 002427 IWAIT:  XWAIT
22 00062 001244 IWAOP:  XWTOP
23 00063 002477 ITISK:  RTIME
24 00064 002553 ITIMS:  MSTIM
25 00065 002702 ITIRO:  XTIMS
26 00066 002653 IMULT:  XMULT
27 00067 002665 IDIVS:  XDIVS
28 00070 002666 IDIVD:  XDIVD
29 00071 005211 IQUES:  XQUES
30 00072 001736 ISAMS:  XSAMS
31 00073 001312 IRESW:  XRESW
32 00074 000000 HMEND:  0               ;TOP OF MEMORY, LOADER PROTECT
33 00075 000000 DIGIN:  0               ;INPUT BUFFER FOR INPUT ROUTINES.
34
35              ;PRINTER TABLE HANDLING AND (POWER) RESTART:
36
37          000076 .LOC 76
38
39 00076 002100 POWZE:  JMP     @POWRE  ;INSTRUCTION TO BE STORED IN CELL ZERO
40 00077 004740 IRESA:  SWISA           ;PROGRAM RESTART ADDR.
41 00100 004662 POWRE:  POWCN           ;POWER RESTART ADDR
42 00101 063077 PRINT:  HALT            ;IMPORTANT TO KEEP THIS AND NEXT IN
43 00102 002077 STOP:   JMP     @IRESA  ;101,102 BECAUSE PRINTER TABLE SELFSTART
44
45 00103 006053 IGTBI:  GETBI
46 00104 005732 IGTOK:  GETOK
47 00105 005427 IGTDC:  GETDC
48 00106 005644 IGTSC:  GETSC
49 00107 006162 IGTTX:  GETTX
50 00110 006417 ISTP0:  ENTP0
51 00111 006424 ISTP1:  ENTP1
52 00112 006431 ISTP2:  ENTP2
53 00113 006450 ILOOP:  CYCLE
54 00114 006620 IHALT:  ERROR
55 00115 006721 ISTAA:  XSTAA
56 00116 006735 ISTAN:  XSTAN
57 00117 006745 ISTAW:  XSTAW
58 00120 006777 ISTAS:  XSTAS
59 00121 006762 ISTAP:  XSTAP
60 00122 007007 ILORE:  XLORE
61 00123 007025 IPASS:  XPASS
```

```
I 0019  .MAIN
01
02                ;  DEFINITIONS
03
04                ;TTI=10
05                ;TTO=11
06                ;RTC=14
07                ;LPT=17
08        000032 FUN=32
09        000033 FUB=33
10        000034 NUK=34
11        000035 DIS=35
12        000017 XLPT=LPT
13        000010 XTTI=TTI
14        000011 XTTO=TTO
15        000014 XRTC=RTC
16
06        006040 BLDRR<JRR        @HLDRR   :D@BH DDFHNHTHNN BDLNV BNRRDRPNNDR
```

```
18      006041 CCHAR=JSR        @ICHAR  ;WITH A CALL OF A ROUTINE.
19      006042 CTYPE=JSR        @ITYPE
20      006043 CCRLF=JSR        @ICRLF
21      006044 CDISP=JSR        @IDISP
22      006045 CDOUT=JSR        @IDOUT
23      006046 CDICL=JSR        @IDICL
24      006047 CDATT=JSR        @IDATT
25      006050 CHAAT=JSR        @IHAAT
26      006051 CTBIN=JSR        @ITBIN
27      006052 CTOCT=JSR        @ITOCT
28      006053 CTDEC=JSR        @ITDEC
29      006054 CTZOC=JSR        @ITZOC
30      006055 CDBIN=JSR        @IDBIN
31      006056 CDOCT=JSR        @IDOCT
32      006057 CDDEC=JSR        @IDDEC
33      006060 CDZOC=JSR        @IDZOC
34      006061 CWAIT=JSR        @IWAIT
35      006062 WATOP=JSR        @IWAOP
36      006063 TIMSK=JSR        @ITISK
37      006064 TIMMS=JSR        @ITIMS
38      006065 TIMRO=JSR        @ITIRO
39      006066 MULTI=JSR        @IMULT
40      006067 DIVIS=JSR        @IDIVS
41      006070 DIVID=JSR        @IDIVD
42      006071 CQUES=JSR        @IGUES
43      006072 CSAMS=JSR        @ISAMS
44      006073 CRESW=JSR        @IRESW
45      006103 CGTBI=JSR        @IGTBI
46      006104 CGTOK=JSR        @IGTOK
47      006105 CGTDC=JSR        @IGTDC
48      006106 CGTSC=JSR        @IGTSC
49      006107 CGTTX=JSR        @IGTTX
50      006110 SETP0=JSR        @ISTP0
51      006111 SETP1=JSR        @ISTP1
52      006112 SETP2=JSR        @ISTP2
53      006113 LOOP=JSR         @ILOOP
54      006114 EHALT=JSR        @IHALT
55      006115 STATA=JSR        @ISTAA
56      006116 STATN=JSR        @ISTAN
57      006117 STATW=JSR        @ISTAW
58      006120 STATS=JSR        @ISTAS
59      006121 STATP=JSR        @ISTAP
60      006122 CLORE=JSR        @ILORE
61      006123 CPASS=JSR        @IPASS
```

```
I 0020  .MAIN
01
02                ;LPTZERO
03
04                ; LPT- DEFINITIONS AND PAGE-ZERO-CONSTANTS
05
06
07        000017 DEV= XLPT
08
09
10  00124 000000 GRORET: 0
11  00125 010750 IRDER:  RDERR
12  00126 007307 ILRST:  XLRST
13  00127 012561 IBUSS:  SBUSY
14  00130 012575 IDONS:  SDONE
15  00131 012644 ICLBU:  CLEBU
16  00132 000000 MRK47:  0                ;<> 0 IF RC3643
17  00133 060017 LASTD:  060017
18  00134 060017 DEV17:  060017
19  00135 060037 DEV37:  060037
20  00136 060057 DEV57:  060057
21  00137 060067 DEV67:  060067
22  00140 000001 TBUC:   1                ;TIME-CONSTANTS FOR ACTUAL PRINTER
23  00141 000000 TDOS:   0                ;WILL BE PLACED HERE
24  00142 000000 TDOSC:  0
25  00143 000000 TS15L:  0
26  00144 002015 PRTCR:  002015           ;PRINTCOMMAFD
27  00145 000000 PRTFF:  0                ;FORM-FEEDCOMMAND
28  00146 000000 SPA1L:  0                ;SPACE-ONE-LINECOMMAND
29  00147 000000 CHALI:  0                ;CHAR/LINE
30  00150 000000 LIPAG:  0                ;LINES/PAGE
31  00151 000037 LDSPA:  000037           ;LOAD A SPACE
32  00152 000177 DELET:  177
33
34
35
36        006130 SETDO=JSR        @IDONS
37        006127 SETBU=JSR        @IBUSS
38        006126 LPRST=JSR        @ILRST
39        006131 CLBUF=JSR        @ICLBU
40
```

```
| 0021  .MAIN
01
02
03                ;INSTRUCTIONS, CHANGED IN STANDARDROUTINES FOR USE IN LPT-TEST:
04
05
06
07
08                ;       LOC. 440        THE INSTR. TIMSK IS CHANGED TO JUMP.+3,
09                ;                       BECAUSE THAT LPT IS NOT USED BY OUT-
10                ;                       PUTROUTINES.
11                ;
12                ;       LOC. 741        THE JUMP XFORM INSTR. IS CHANGED TO JMP
13                ;                       TYPE2 FOR DIRECTLY PRINT ON TTO, BY-
14                ;                       PASSING LPT-PRINT.
15                ;
16                ;       LOC. 1055       THE PRINTERTABLE IS ONLY USED BY THE EX-
17                ;                       ERCISERPROGRAMS. THE ACTION-ENTRY-TABLE
18                ;                       THEREFORE POINTS ON A ROUTINE IN THIS
19                ;                       PROGRAMPART, NAMELY PRTCH INSTEAD OF
20                ;                       ACT0.
21                ;
22                ;       LOC. 1063       SAME AS PREVIOUS EXPLANATION.
23                ;
24                ;       LOC. 6446       IORST IS CHANGED TO LPRST (JSR XLRST) TO
25                ;                       LET XLRST-ROUT. CHECK THE LPT-STATE BE-
26                ;                       FORE EXECUTION OF I/O-RESET.
27                ;
28                ;       LOC. 6511       SAME AS LOC. 6446
29                ;
30
31
32
33
34
35
36
37
38 00153 007136 BEG01:  STAR1
39 00154 007142 BEG02:  STAR2
40 00155 007147 BEG03:  STAR3
41 00156 007164 BEG04:  STAR4
42 00157 007175 BEG05:  RESTR
43 00160 015727 BEG06:  HMJUS
44 00161 015715 BEG07:  JUS22
```

```
| 0022  .MAIN
01
02                ; STARTLOCATIONS
03
04
05
06          000400          .LOC 400
07
08 00400 006153           JSR      @BEG01   ;ASK ALL QUESTIONS AND SET EQUIPMENT-
09                                          ;CONSTANTS AND TESTFIGURE ACCORDING
10                                          ;TO THE ANSWERS.
11 00401 006154           JSR      @BEG02   ;INITIALIZE LPT-TYPE,CONTROLERTYPE AND
12                                          ;OTHER EQUIPMENT-CONSTANTS, BUT ASK
13                                          ;QUESTIONS ABOUT TESTEXECUTION.
14 00402 006155           JSR      @BEG03   ;EXECUTE THE DIAGNOSTICTEST (WITHOUT
15                                          ;TESTS INCLUDING OPERATOR-INVOLVEMENT).
16                                          ;ALL EQUIPMENTCONSTANTS ARE INI-
17                                          ;TIALIZED.
18 00403 006156           JSR      @BEG04   ;EXECUTE THE CHARACTERMOTIONTESTS,
19                                          ;WITH INITIALIZED EQUIPMENTCONSTANTS.
20                                          ;ASK FOR INPUTMODE IN THE CHARACTER-
21                                          ;PRINT-ROUTINE.
22 00404 006157           JSR      @BEG05   ;RESTART OF RECENT DEFINED TESTLOOP
23 00405 006160           JSR      @BEG06   ;HAMMER ADJUST PROGRAM - DP2420/40/70
24 00406 006161           JSR      @BEG07   ;HAMMER ADJUST PROGRAM - DP2230
25
26
27
28
29                ; LPTZERO
30
```

```
I 0023  .MAIN
02
03              ;TAPE 3         OUTPUT ROUTINES AND OTHERS.
04
05      000412  .LOC 412        ;DO NOT MOVE UP.
06
07                              ;FOR THE USE SEE EACH ROUTINE.
08
09              ;MESS   TYPE A TEXT MESSAGE ON TTO AND LPT
10              ;CHAR   TYPE A CHAR ON TTO AND LPT, CALCULATE P-BIT
11              ;TYPE   TYPE A CHAR ON TTO AND LPT
12              ;CRLF   TYPE A CR AND A LF ON TTO AND LPT
13              ;DISP   DISPLAY A TEXT MESSAGE ON DIS
14              ;DOUT   DISPLAY A CHAR ON DIS
15              ;DCL    CLEAR DIS
16              ;DISATT DISPLAY ATTENTION: BEEP AND WAIT 3 SECONDS
17              ;HAATT  HALT ATTENTION: BEEP
18              ;TBIN   TYPE BINARY NUMBER ON TTO AND LPT
19              ;TOCT   TYPE OCTAL NUMBER ON TTO AND LPT
20              ;TDEC   TYPE DECIMAL NUMBER ON TTO AND LPT
21              ;TZOC   TYPE OCTAL NUMBER WITHOUT LEADING ZEROES.
22              ;DBIN   DISPLAY BINARY NUMBER ON DIS
23              ;DOCT   DISPLAY OCTAL NUMBER ON DIS
24              ;DDEC   DISPLAY DECIMAL NUMBER ON DIS
25              ;DZOC   DISPLAY OCTAL NUMBER WITHOUT LEADING ZEROES.
26              ;WAIT   WAIT SOME MILLISECONDS
27              ;WATOP  WAIT FOR OPERATOR ACTION
28              ;TIMSK  TIMER FOR SKIP INSTRUCTION
29              ;TIMRO  TIMER FOR ROUTINE
30              ;TIMMS  TIME MESSURE ROUTINE
31              ;MULTI  MULTIPLY
32              ;DIVIS  DIVIDE SINGLE
33              ;DIVID  DIVIDE DOUBLE
34              ;QUES   OUTPUT QUESTIONS
35              ;SAMS   START ADDR MESSAGE
36              ;RESW   READ SWITCHES, REPORT CHANGES
37              ;SAPTB  MOVE PRINTER TABLE
38              ;SAMEX  EXTEND TO 64K WORDS MEM MODE
39              ;SAMNM  RESET TO 32K WORDS MEM MODE
40              ;REBIN  INITIALIZE, RESTORE BIN LOADER
41
42
43              ;AN ADD ON PRINTERTABLE SHOULD HAVE THIS FORMAT:
44              ;   .LOC 2000
45              ;   .RDX Y
46              ;   .TXTE?
47              ;   <6><0>      ;TOTALLY 200 BOXES WITH
48              ;               ;<TYPE OF ACTION><CHAR TO PRINT>
49              ;   <6><0>?
50              ;   .END 101
51              ;               ;TYPE OF ACTION:        0=PRINT
52              ;                                       6=BLIND
```

```
I 0024  .MAIN
01
02              ;LPT, TTC AND DIS NON INTERRUPT PACKAGE
03              ;IF THE DEVICE(S) ARE NOT PRESENT THE ROUTINES CONTINUES.
04              ;IF A DEVICE IS CONNECTED (SET TO ONLINE) THE PROGRAM
05              ;MUST BE RESTARTED IN THE WANTED STARTADDRESS.
06
07              ;LPT AND TTC ROUTINES:
08              ;>MESS< PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLER.
09              ;>CHAR< PRINTS ASCII CHARACTOR, AC0-R.
10              ;AC0-L MUST BE 0. CORRECTS THE PARITY, 11 SIMULATE TAB.
11              ;>TYPE< PRINTS AC0-R. MUST HAVE PROPER PARITY. RETURN IS
12              ;TO CALL+1.REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIRED.
13              ;>CRLF< PRINTS A CARRIAGE RETURN FOLLOWED BY A LINE FEED.
14              ;>TOCT< PRINTS AC1 IN OCTAL, 6 DIGITS AND 6 PRINTPOSITIONS.
15              ;>TDEC< PRINTS AC1 IN DECIMAL, LEADING 0'S SUPPRESSED, WITH SIGN
16              ;RANGE -32768 TO -1 AND +0 TO +32767, 6 PRINTPOS, UP TO 5 DIGITS.
17              ;>TOCT< PRINTS AC1 IN OCTAL, LEADING 0'S SUPPRESSED,
18              ;6 PRINTPOSITIONS AND UP TO 6 DIGITS.
19              ;THE ROUTINES TOCT, TZOC AND TDEC ARE FOLLOWED BY A TAB
20              ;BUT ONLY IF NOT ALLREADY AT A TABPOINT.
21              ;>TBIN< PRINTS AC1 IN BINARY, AC1-R 8 DIGITS ONLY, 8 PRINTPOS.
22
23              ;DIS ROUTINES:
24              ;>DISP< PRINTS ACSII MESSAGES AS SPECIFIED BY ASSEMBLER.
25              ;>DOUT< PRINTS ASSII CHARACTOR, AC0-R. AC0-L MUST BE 0. RETURN
26              ;TO CALL+1. REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIRED.
27              ;>DICL< CLEARS THE DISPLAY.
28              ;>DISATT< ATTENTION DISPLAY, SEE ROUTINE
29              ;>HAATT<  ATTENTION HALT, SEE ROUTINE
30              ;>DOCT< DISPLAYS AC1 IN OCTAL, 6 DIGITS AND 6 PRINTPOSITIONS.
31              ;>DDEC< DISPLAYS AC1 IN DECIMAL LEADING 0'S SUPPRESSED, WITH SIGN
32              ;RANGE -32768 TO -1 AND +0 TO +32767, 6 PRINTPOS, UP TO 5 DIGITS.
33              ;>DZOC< DISPLAYS AC1 IN OCTAL, LEADING 0'S SUPPRESSED,
34              ;6 PRINTPOSITIONS AND UP TO 6 DIGITS.
35              ;>DBIN< DISPLAYS AC1 IN BINARY, AC1-R 8 DIGITS ONLY, 8 PRINTPOS.
36
37              ;       LDA     1,NUMBER ;ALL NUMBER ROUTINES RESTORE AC1
38              ;CALL   CTDEC
39              ;       CTOCT
40              ;       CTZOC
41              ;       CDDEC
42              ;       CDOCT
43              ;       CDZOC
44              ;       CTBIN
45              ;       CDBIN
46              ;       MOVS    1,1     ;FOR THE OTHER 8 BITS
47              ;       CTBIN
48              ;       CDBIN
49              ;       LDA     0,CHAR  ;ALL CHARACTER ROUTINES RESTORE AC0
50              ;       CTYPE
51              ;       CCHAR
52              ;       CDOUT
53              ;       CDICL
54              ;       CDISP
55              ;       TEXTLABEL
56              ;       CDATT
57              ;       CCRLF
58              ;       CMESS
59              ;       LABELTEXT
60              ;       CHAAT
61              ;       HALT
```

```
| 0025  .MAIN
01
02 00412 054450 XMESS:  STA     3,RPOUT ;PRINT A TEXT MESSAGE
03 00413 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
04 00414 010446         ISZ     RPOUT
05 00415 004556         JSR     PINHI
06 00416 002444         JMP     @RPOUT  ;NO PRINT
07 00417 034443         LDA     3,RPOUT ;AC3 POINTS TO MESSAGEPOINTER+1
08 00420 031777         LDA     2,-1,3  ;AC2 POINTS TO MESSAGE
09 00421 024440         LDA     1,CHMAS ;A 8 BIT MASK
10 00422 021000 MESSA:  LDA     0,0,2   ;AC0=DATA WORD
11 00423 125112         MOVL#   1,1,SZC
12 00424 123701         ANDS    1,0,SKP
13 00425 123401         AND     1,0,SKP ;AC0=DATA CHARACTOR RIGHT
14 00426 151400         INC     2,2     ;INC TO NEXT WORD
15 00427 124000         COM     1,1     ;FLIP MASK
16 00430 004404         JSR     MESCH   ;PRINT
17 00431 000771         JMP     MESSA   ;ANOTHER
18 00432 004405         JSR     BZOUT   ;TERMINATE MESS
19 00433 002427         JMP     @RPOUT  ;EXIT
20
21 00434 101015 MESCH:  MOV#    0,0,SNR ;TEST LAST CHAR
22 00435 001401         JMP     1,3     ;RETURN +2 IF NULL
23 00436 002420         JMP     @ZCHAR  ;TYPE CHAR
24
25 00437 054412 BZOUT:  STA     3,RBZOT
26 00440 000403         JMP     .+3     ;WAIT IF LPT/TTO BUSY
27 00441 001750 XLPTT:  1000.           ;MAX 1 SEC LPT
28 00442 063517         SKPBZ   XLPT
29 00443 044776         STA     1,XLPTT ;REMOVE WAITING, LPT NOT CONNECTED
30 00444 006063         TIMSK
31 00445 000454 XTTOT:  300.            ;MAX 300 MSEC TTO
32 00446 063511         SKPBZ   XTTO
33 00447 044776         STA     1,XTTOT ;REMOVE WAITING, TTO NOT CONNECTED
34 00450 002401         JMP     @RBZOT
35
36 00451 000000 RBZOT:  0
37 00452 000000 RINHI:  0
38 00453 000000 SADIG:  0
39 00454 000000 RXDEC:  0
40 00455 000040 CHINH:  40
41 00456 000672 ZCHAR:  YCHAR
42 00457 001142 ZDOUT:  YDOUT
43 00460 000712 ZTAB3:  YTAB3
44 00461 000377 CHMAS:  377
45 00462 000000 RPOUT:  0
46 00463 000000 PDECR:  0
47
48 00464 054776 XDDEC:  STA     3,RPOUT ;DISPLAY DECIMAL NUMBER
49 00465 004415         JSR     YPDEC
50 00466 000502         JMP     NXDIS   ;DISPLAY NEXT DIGIT
51 00467 000411         JMP     EXDIS   ;EXIT
52
53 00470 054772 XTDEC:  STA     3,RPOUT ;TYPE DECIMAL NUMBER
54 00471 004411         JSR     YPDEC
55 00472 000402         JMP     TYPNX   ;TYPE NEXT DIGIT
56 00473 000404         JMP     EXTYP   ;EXIT
57
58 00474 054760 TYPNX:  STA     3,RXDEC
59 00475 004575         JSR     YCHAR   ;TYPE DIGIT
60 00476 002756         JMP     @RXDEC  ;NEXT DIGIT, SIGPR OR DECPR
61
62 00477 006761 EXTYP:  JSR     @ZTAB3  ;YPDEC/ZOCT/POCT FINISH RETURN, TYPE TAB
63 00500 024753 EXDIS:  LDA     1,SADIG ;DISPLAY FINISHED, RESTORE PARAM
64 00501 002761         JMP     @RPOUT  ;EXIT
```

```
I 0026  .MAIN
01
02 00502 044751 YPDEC:  STA     1,SADIG ;SAVE PARAM FOR REPEAT
03 00503 054760          STA     3,PDECR
04 00504 030756          LDA     2,RPOUT ;SPEC RETURN IF PRINT SW
05 00505 004466          JSR     PINHI
06 00506 002754          JMP     @RPOUT  ;NO PRINT
07 00507 020436          LDA     0,CHPLU ;SETUP PLUS
08 00510 040437          STA     0,CHFLG ;SIGN FLAG
09 00511 020552          LDA     0,CHSPA ;OR PLUS: CHPLU
10 00512 040432          STA     0,CHSIG ;PRINT SPACE OR PLUS
11 00513 102620          SUBZR   0,0     ;AC0:=100000
12 00514 106415          SUB#    0,1,SNR ;TEST FOR -32768
13 00515 000404          JMP     PDEC2
14 00516 107415          AND#    0,1,SNR ;TEST FOR NEGATIVE
15 00517 000404          JMP     PDEC3   ;POS
16 00520 124400          NEG     1,1     ;NEG
17 00521 020425 PDEC2:   LDA     0,CHMIN ;SETUP MINUS
18 00522 040422          STA     0,CHSIG
19 00523 020537 PDEC3:   LDA     0,CHSP  ;SUPPRESS LEADING ZEROES
20 00524 030516          LDA     2,DECTB ;PRINT AC1 IN DECIMAL
21 00525 000466          JMP     PDEC1
22
23 00526 054415 SIGN:    STA     3,SIGNR
24 00527 034420          LDA     3,CHFLG
25 00530 030415          LDA     2,CHPLU
26 00531 172414          SUB#    3,2,SZR ;TEST PRINT SIGN
27 00532 002411          JMP     @SIGNR  ;IF NO SIGN, RETURN
28 00533 034527          LDA     3,CHSP
29 00534 116415          SUB#    0,3,SNR ;TEST FIRST DIGIT
30 00535 002406          JMP     @SIGNR  ;IF LEADING SPACE, RETURN
31 00536 040411          STA     0,CHFLG ;STORE ASCII AND DESTROY
32 00537 020405          LDA     0,CHSIG ;PRINT SIGN FLAG
33 00540 006723          JSR     @PDECR  ;OUTPUT SIGN
34 00541 020406          LDA     0,CHFLG ;RESTORE ASCII DIGIT
35 00542 002401          JMP     @SIGNR  ;PRINT FIRST DIGIT
36
37 00543 000000 SIGNR:   0
38 00544 000000 CHSIG:   0
39 00545 000053 CHPLU:   53
40 00546 000055 CHMIN:   55
41 00547 000000 CHFLG:   0
42
43 00550 054712 XTZOC:   STA     3,RPOUT ;TYPE ZERO OCTAL NUMBER
44 00551 004430          JSR     YZOCT
45 00552 000722          JMP     TYPNX   ;TYPE NEXT DIGIT
46 00553 000724          JMP     EXTYP   ;EXIT
47
48 00554 054706 XDZOC:   STA     3,RPOUT ;DISPLAY ZERO OCTAL NUMBER
49 00555 004424          JSR     YZOCT
50 00556 000412          JMP     NXDIS   ;DISPLAY NEXT DIGIT
51 00557 000721          JMP     EXDIS   ;EXIT
52
53 00560 054702 XTOCT:   STA     3,RPOUT ;TYPE OCTAL NUMBER
54 00561 004422          JSR     YPOCT
55 00562 000712          JMP     TYPNX   ;TYPE NEXT DIGIT
56 00563 000714          JMP     EXTYP   ;EXIT
57
58 00564 054676 XDOCT:   STA     3,RPOUT ;DISPLAY OCTAL NUMBER
59 00565 004416          JSR     YPOCT
60 00566 000402          JMP     NXDIS   ;DISPLAY NEXT DIGIT
61 00567 000711          JMP     EXDIS   ;EXIT
62
63 00570 054664 NXDIS:   STA     3,RXDEC
64 00571 006666          JSR     @ZDOUT  ;DISPLAY DIGIT
65 00572 002662          JMP     @RXDEC  ;NEXT DIGIT
```

```
I 0027  .MAIN
01
02 00573 054657 PINHI:  STA     3,RINHI ;AFTER ACTIVATING SETPX IN TESTLOOP
03 00574 000403 SETAC:  JMP     NINHI   ;THIS IS CHANGED TO CRESW (READS 2 ROU)
04 00575 034660         LDA     3,CHINH ;SW 10 FOR NO PRINTING
05 00576 157405         AND     2,3,SNR ;RETURN+1 IF INHIBIT
06 00577 010653 NINHI:  ISZ     RINHI   ;RETURN+2 IF PRINTING
07 00600 002652         JMP     @RINHI  ;EXIT
08
09 00601 020461 YZOCT:  LDA     0,CHSP  ;ZERO OCTAL ROUTINE
10 00602 101001         MOV     0,0,SKP
11
12 00603 020565 YPOCT:  LDA     0,CHAR0 ;OCTAL ROUTINE
13 00604 044647         STA     1,SADIG ;SAVE PARAM FOR REPEAT
14 00605 054656         STA     3,PDECR
15 00606 030654         LDA     2,RPOUT ;SPEC RETURN IF PRINT SW
16 00607 004764         JSR     PINHI
17 00610 002652         JMP     @RPOUT  ;NO PRINT
18 00611 030440         LDA     2,OCTAB ;PRINT AC1 IN OCTAL
19 00612 040735         STA     0,CHFLG ;NO-SIGN FLAG
20 00613 040556 PDEC1:  STA     0,ZSUPP ;BOTH ENTRYS PRINT NUMBER
21 00614 050401         STA     2,.+1   ;THEN TAB TO NEXT POSITION
22 00615 000000 DECOCT: 0               ;A LDA 2,TABLE INSTRUCTION
23 00616 010777         ISZ     .-1
24 00617 151005         MOV     2,2,SNR ;IF TABLE ENTRY=0 THEN
25 00620 000420         JMP     DECEX   ;EXIT WITH TAB SPECIAL IF TYPE ROU.
26 00621 034550         LDA     3,ZSUPP ;ZEROS SUPPRESS STUF
27 00622 102400         SUB     0,0
28 00623 146512 DECOT:  SUBL#   2,1,SZC
29 00624 000405         JMP     DECP
30 00625 146400         SUB     2,1     ;FORM THE DIGIT
31 00626 034542         LDA     3,CHAR0 ;DO NOT SUPPRESS
32 00627 101400         INC     0,0     ;FOLLOWING ZEROES
33 00630 000773         JMP     DECOT
34 00631 151235 DECP:   MOVZR#  2,2,SNR ;IF LAST DIGIT THEN
35 00632 034536         LDA     3,CHAR0 ;AC3=ZERO, NOT SUPPR CHAR
36 00633 054536         STA     3,ZSUPP ;AC0=DIGIT
37 00634 163000         ADD     3,0     ;MAKE ASCII
38 00635 004671         JSR     SIGN    ;TEST SIGN
39 00636 006625         JSR     @PDECR  ;OUTPUT DIGIT
40 00637 000756         JMP     DECOCT  ;GET NEXT DIGIT
41 00640 010623 DECEX:  ISZ     PDECR   ;RETURN ADDR FOR JSR Y-ROUTINE
42 00641 002622         JMP     @PDECR  ;EXIT WITH TAB IF TYPING ROUTINE
43
44 00642 030426 DECTB:  LDA     2,.+1+.-DECOCT
45       000012 .RDX 10
46 00643 023420         10000
47 00644 001750         1000
48 00645 000144         100
49 00646 000012         10
50 00647 000001         1
51 00650 000000         0
52       000010 .RDX 8
53
54 00651 030435 OCTAB:  LDA     2,.+1+.-DECOCT
55 00652 100000         100000
56 00653 010000         10000
57 00654 001000         1000
58 00655 000100         100
59 00656 000010         10
60 00657 000001         1
61 00660 000000         0
62
63 00661 000000 CHRET:  0
64 00662 000240 CHSP:   240             ;LEADING ZERO SUPPRESS CHAR + TAB SIML
65 00663 000240 CHSPA:  240             ;A SPACE FOR + IN PDEC
```

```
I 0028  .MAIN
01
02 00664 040507 XCHAR:  STA     0,SACHA ;SAVE PARAM FOR REPEAT
03 00665 054774         STA     3,CHRET
04 00666 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
05 00667 004704         JSR     PINHI
06 00670 002771         JMP     @CHRET  ;NO TYPE
07 00671 000402         JMP     QCHAR
08
09 00672 054767 YCHAR:  STA     3,CHRET ;PRINT AC0 RIGHT
10 00673 101320 QCHAR:  MOVZS   0,0
11 00674 040500         STA     0,CHSAV
12 00675 176000 CHAR2:  ADC     3,3     ;COMPUTE THE PARITY
13 00676 117000         ADD     0,3     ;FOR EVEN
14 00677 163404         AND     3,0,SZR
15 00700 000775         JMP     CHAR2
16 00701 176660         SUBCR   3,3     ;COMBIND PARITY WITH CHAR
17 00702 020472         LDA     0,CHSAV
18 00703 163300         ADDS    3,0
19 00704 034463 CHAR1:  LDA     3,CHTAB ;IS THIS A TAB
20 00705 116405         SUB     0,3,SNR
21 00706 000413         JMP     CHAR4   ;YES
22 00707 004423         JSR     YTYPE   ;NO PRINT IT
23 00710 020463         LDA     0,SACHA ;RESTORE PARAM
24 00711 002750         JMP     @CHRET  ;EXIT
25
26 00712 054747 YTAB3:  STA     3,CHRET ;NUMBER ROUTINE TAB RETURN
27 00713 020462 CHAR3:  LDA     0,CHORZ ;SIMULATE A TAB
28 00714 034462         LDA     3,CHAR7 ;VIA 1 TO 8 SPACES
29 00715 117404         AND     0,3,SZR
30 00716 000403         JMP     CHAR4
31 00717 020454         LDA     0,SACHA ;RESTORE PARAM
32 00720 002741         JMP     @CHRET  ;EXIT AFTER TAB
33 00721 020741 CHAR4:  LDA     0,CHSP
34 00722 004410         JSR     YTYPE
35 00723 000770         JMP     CHAR3
36
37 00724 040554 XTYPE:  STA     0,REG0  ;SAVE PARAM FOR REPEAT
38 00725 054556         STA     3,REG3  ;SAVE RETURN
39 00726 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
40 00727 004644         JSR     PINHI
41 00730 002553         JMP     @REG3   ;NO TYPE
42 00731 000403         JMP     QTYPE
43
44 00732 054551 YTYPE:  STA     3,REG3  ;SAVE RETURN
45 00733 040545         STA     0,REG0  ;SAVE PARAM FOR TYPING
46 00734 010441 QTYPE:  ISZ     CHORZ   ;INC HORIZIONAL POSITION
47 00735 044544         STA     1,REG1  ;SAVE AC1 AND AC2 FOR NUMBER AND
48 00736 050544         STA     2,REG2  ;MESS ROUTINES
49 00737 006440         JSR     @TBZOT
50 00740 020540         LDA     0,REG0
51 00741 000403         JMP     TYPE2
52 00742 061017 TYPE1:  DOA     0,XLPT  ;SEND CHAR
53 00743 060117         NIOS    XLPT    ;START LPT
54 00744 020534 TYPE2:  LDA     0,REG0
55 00745 024532         LDA     1,CHLF
56 00746 106415         SUB#    0,1,SNR ;LF ?
57 00747 000404         JMP     TYPE3   ;YES, LF
58 00750 024526         LDA     1,CHCR
59 00751 106414         SUB#    0,1,SZR ;CR ?
60 00752 000404         JMP     TYPE4   ;NO CR
61 00753 006061 TYPE3:  CWAIT           ;IF CR, LF WAIT FOR
62 00754 001237         SECM2           ;DATAPOINT 20 MSEC
63 00755 020523         LDA     0,REG0  ;RESTORE AC0
64 00756 061011 TYPE4:  DOA     0,XTTO  ;SEND CHAR
65 00757 060111         NIOS    XTTO    ;START TTO
```

```
I 0029  .MAIN
01
02 00760 152400 TYPE5:  SUB     2,2
03 00761 024516         LDA     1,CHLF
04 00762 106415         SUB#    0,1,SNR ;IF LF
05 00763 050412         STA     2,CHORZ ;CLEAR HORZ POS
06 00764 030516         LDA     2,REG2
07 00765 024514         LDA     1,REG1  ;AC0 = REG0 FOR REPEAT
08 00766 002515         JMP     @REG3   ;EXIT
09
10 00767 000011 CHTAB:  11
11 00770 000060 CHAR0:  60
12 00771 000000 ZSUPP:  0
13 00772 177770 NN10:   -10
14 00773 000000 SACHA:  0
15 00774 000000 CHSAV:  0
16 00775 000000 CHORZ:  0
17 00776 000007 CHAR7:  7
18 00777 000437 TBZOT:  BZOUT
19
20 01000 030561 XFORM:  LDA     2,RMSK  ;TRANSFORM CHAR TO
21 01001 143400         AND     2,0     ;PRINTER ALPHABET
22 01002 030556         LDA     2,PTAB
23 01003 113000         ADD     0,2     ;AC2:=CHAR+TABLE
24 01004 025000         LDA     1,0,2
25 01005 030554         LDA     2,RMSK
26 01006 133400         AND     1,2     ;AC2:=CLASS
27 01007 020554         LDA     0,LMSK
28 01010 123700         ANDS    1,0     ;AC0:=CHAR
29 01011 024443         LDA     1,ACTN
30 01012 133000         ADD     1,2     ;AC2:=ACTN+1+CLASS
31 01013 003000         JMP     @0,2    ;GO TO ACTION
32 01014 054461 XTBIN:  STA     3,CRBIR
33 01015 004407         JSR     YPBIN
34 01016 004654         JSR     YCHAR   ;TYPE DIGIT
35 01017 000420         JMP     PBINN   ;NEXT DIGIT
36 01020 054455 XDBIN:  STA     3,CRBIR
37 01021 004403         JSR     YPBIN
38 01022 004520         JSR     YDOUT   ;DISPLAY DIGIT
39 01023 000414         JMP     PBINN   ;NEXT DIGIT
40 01024 044447 YPBIN:  STA     1,SABIN ;SAVE PARAM FOR REPEAT
41 01025 054447         STA     3,PBINR
42 01026 030447         LDA     2,CRBIR ;SPEC RETURN IF PRINT SW
43 01027 006535         JSR     @TINHI
44 01030 002445         JMP     @CRBIR  ;NO PRINT
45 01031 030741         LDA     2,NN10  ;8 TIMES
46 01032 125300         MOVS    1,1
47 01033 020735 PBINC:  LDA     0,CHAR0
48 01034 125102         MOVL    1,1,SZC
49 01035 101400         INC     0,0     ;AC0:="CHAR1"
50 01036 002436         JMP     @PBINR  ;OUTPUT DIGIT
51 01037 151404 PBINN:  INC     2,2,SZR
52 01040 000773         JMP     PBINC
53 01041 024432         LDA     1,SABIN ;RESTORE PARAM
54 01042 002433         JMP     @CRBIR
55
56 01043 054432 XCRLF:  STA     3,CRBIR ;SAVE RETURN
57 01044 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
58 01045 006517         JSR     @TINHI
59 01046 002427         JMP     @CRBIR  ;NO TYPE
60 01047 020427         LDA     0,CHCR
61 01050 004622         JSR     YCHAR
62 01051 020426         LDA     0,CHLF
63 01052 004620         JSR     YCHAR   ;PRINT CR,LF
64 01053 002422         JMP     @CRBIR  ;EXIT
```

```
1 0030  .MAIN
01
02 01054 001055 ACTN:   .+1             ;ACTION ENTRY TABLE
03 01055 013743         PRTCH           ;NORMAL ACTION
04 01056 001067         ILL
05 01057 001067         ILL
06 01060 001067         ILL
07 01061 001067         ILL
08 01062 001067         ILL
09 01063 013743         PRTCH           ;BLIND
10 01064 001067         ILL
11 01065 001067         ILL
12 01066 001067         ILL
13
14 01067 063077 ILL:    HALT
15 01070 000777         JMP     .-1     ;ILLEGAL
16 01071 000651 ACT0:   JMP     TYPE1
17 01072 000652 ACT6:   JMP     TYPE2
18
19 01073 000000 SABIN:  0
20 01074 000000 PBINR:  0
21 01075 000000 CRBIR:  0
22 01076 000215 CHCR:   215
23 01077 000012 CHLF:   12
24 01100 000000 REG0:   0
25 01101 000000 REG1:   0
26 01102 000000 REG2:   0
27 01103 000000 REG3:   0
28
29 01104 054771 XDISP:  STA     3,CRBIR ;DISPLAY MESSAGE
30 01105 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
31 01106 010767         ISZ     CRBIR
32 01107 006455         JSR     @TINHI
33 01110 002765         JMP     @CRBIR  ;NO PRINT
34 01111 034764         LDA     3,CRBIR ;AC3=POINTS TO MESSAGEPOINTER+1
35 01112 031777         LDA     2,-1,3  ;AC2 POINTS TO MESSAGE
36 01113 024447         LDA     1,CMSK
37 01114 021000 DISP1:  LDA     0,0,2   ;AC0=DATAWORD
38 01115 125112         MOVL#   1,1,SZC
39 01116 123701         ANDS    1,0,SKP
40 01117 123401         AND     1,0,SKP ;AC0=CHAR, RIGHT
41 01120 151400         INC     2,2     ;INC TO NEXT WORD
42 01121 124000         COM     1,1     ;FLIP MASK
43 01122 004407         JSR     DDICH   ;GO DISPLAY
44 01123 000771         JMP     DISP1   ;ANOTHER
45 01124 006063         TIMSK           ;TERMINATE DISP
46 01125 000050 FDIST:  40.             ;MAX 40 MSEC
47 01126 063535         SKPBZ   DIS
48 01127 044776         STA     1,FDIST ;REMOVE WAITING, DIS NOT CONNECTED
49 01130 002745         JMP     @CRBIR  ;RETURN
50
51 01131 101015 DDICH:  MOV#    0,0,SNR ;TEST LAST CHAR
52 01132 001401         JMP     1,3     ;RETURN +2 IF NULL
53 01133 000407         JMP     YDOUT   ;DISPLAY
```

```
I 0031  .MAIN
01
02 01134 040744 XDOUT:  STA     0,REG0  ;SAVE PARAM FOR REPEAT
03 01135 054746         STA     3,REG3  ;SAVE RETURN
04 01136 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
05 01137 006425         JSR     @TINHI
06 01140 002743         JMP     @REG3   ;NO DISPLAY
07 01141 000403         JMP     QDOUT
08
09 01142 054741 YDOUT:  STA     3,REG3  ;SAVE RETURN
10 01143 040735         STA     0,REG0  ;SAVE PARAM FOR DISPLAYING
11 01144 044735 QDOUT:  STA     1,REG1  ;SAVE AC1 AND AC2 FOR NUMBER AND
12 01145 050735         STA     2,REG2  ;MESS ROUTINES
13 01146 006063         TIMSK           ;WAIT IF DIS BUSY
14 01147 000050 SDIST:  40.             ;MAX 40 MSEC
15 01150 063535         SKPBZ   DIS
16 01151 044776         STA     1,SDIST ;REMOVE WAITING, DIS NOT CONNECTED
17 01152 020726         LDA     0,REG0
18 01153 061035         DOA     0,DIS   ;SEND CHAR
19 01154 060135         NIOS    DIS     ;START DIS
20 01155 024724         LDA     1,REG1
21 01156 030724         LDA     2,REG2  ;AC0 = REG0 FOR REPEAT
22 01157 002724         JMP     @REG3   ;EXIT
23
24 01160 002000 PTAB:   TABLE
25 01161 000177 RMSK:   177
26 01162 000377 CMSK:   377
27 01163 077400 LMSK:   077400
28 01164 000573 TINHI:  PINHI
29
30 01165 054710 XDICL:  STA     3,CRBIR ;SAVE RETURN
31 01166 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
32 01167 006775         JSR     @TINHI
33 01170 002705         JMP     @CRBIR  ;NO DISPLAY
34 01171 000402         JMP     QDICL
35
36 01172 054703 YDICL:  STA     3,CRBIR
37 01173 020704 QDICL:  LDA     0,CHLF
38 01174 004746         JSR     YDOUT   ;DISPLAY CLEAR
39 01175 002700         JMP     @CRBIR  ;RETURN
```

```
I 0032  .MAIN
01
02                          ;RC 3600, ATTENTION DISPLAY OUTPUT
03                          ;BY MEANS OF ACCUSTIC ALARM FOR 50 MS
04                          ;AND WAIT FOR 3 SECONDS TO ALLOW
05                          ;THE OPERATOR TO READ THE MESSAGE.
06                          ;
07                          ;CALL   CDATT
08                          ;
09                          ;RC 3600, ATTENTION HALT BY MEANS OF
10                          ;ACCUSTIC ALARM FOR 50 MS BEFORE HALT.
11                          ;
12                          ;CALL   CHAAT
13                          ;
14 01176 054676 DISATT: STA     3,PBINR
15 01177 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
16 01200 006764         JSR     @TINHI
17 01201 002673         JMP     @PBINR  ;NO PRINT
18 01202 004424         JSR     YHAAT
19 01203 006073         CRESW           ;READS 2 ROUTINE
20 01204 024436         LDA     1,SECS4 ;SW13 WAITING DIS
21 01205 133415         AND#    1,2,SNR
22 01206 000403         JMP     DISSW
23 01207 006061         CWAIT
24 01210 001236         SEC3
25
26 01211 006073 DISSW:  CRESW           ;READS 2 ROUTINE
27 01212 024427         LDA     1,SECS2
28 01213 133414         AND#    1,2,SZR
29 01214 063077         HALT            ;HALT IF SWITCH 14
30 01215 126520         SUBZL   1,1     ;AC1:=1
31 01216 133414         AND#    1,2,SZR
32 01217 004753         JSR     YDICL   ;CLEAR DIS IF SWITCH 15
33 01220 002654         JMP     @PBINR
34
35 01221 054654 HAATT:  STA     3,CRBIR
36 01222 171000         MOV     3,2     ;SPEC RETURN IF PRINT SW
37 01223 006741         JSR     @TINHI
38 01224 002651         JMP     @CRBIR  ;NO "PRINT"
39 01225 000402         JMP     QHAAT
40
41 01226 054647 YHAAT:  STA     3,CRBIR
42 01227 020414 QHAAT:  LDA     0,SEC4K
43 01230 061032         DOA     0,FUN
44 01231 006061         CWAIT
45 01232 001240         SECM5
46 01233 102400         SUB     0,0
47 01234 061032         DOA     0,FUN
48 01235 002640         JMP     @CRBIR
49
50       000012         .RDX 10         ;CHANGING SEC3 AFFECTS CRESW AND POWON.
51 01236 005670 SEC3:   3000            ;CHANGE TO 1000(0) FOR 1(0) SEC WAITING.
52 01237 000024 SECM2:  20              ;20 MSEC FOR TYPE ROUTINE DATAPOINT DELAY
53 01240 000062 SECM5:  50
54       000010         .RDX 8
55 01241 000002 SECS2:  2
56 01242 000004 SECS4:  4
57 01243 004000 SEC4K:  4000
```

```
| 0033  .MAIN
01
02                ;PROCEDURE WAIT OPERATOR
03                ;TURNS ON THE FUNCTION INDICATOR FROM ARG,
04                ;WAITS FOR THE OPERATOR TO PRESS BUTTON FROM ARG,
05                ;OR TO PRESS KEY AT TTY
06                ;AND TURNS OFF THE FUNCTION INDICATOR FROM ARG.
07                ;AND WRITES PROG NAME AT DIS (SEE LABEL: PROG),
08                ;OR TYPE $ AT TTY
09                ;UNCHANGED:     AC0
10                ;DESTROYED:     AC1,AC2,AC3,CARRY
11                ;CALL:  WATOP
12                ;       ARG
13
14 01244 025400 XWTOP:  LDA     1,0,3   ; FUNCTION MASK FROM ARGUMENT
15 01245 175400         INC     3,3     ; FOR INDICATORS AND BUTTONS
16 01246 054437         STA     3,WTORE ; SAVE RETURN
17 01247 034440         LDA     3,WHIGH
18 01250 040441         STA     0,WACSAV
19 01251 020435         LDA     0,WLOWL
20 01252 065032         DOA     1,FUN   ; TURN ON INDICATOR ARG
21 01253 063610 WTNOK:  SKPDN   XTTI
22 01254 000411         JMP     WTFUB
23 01255 070410         DIA     2,XTTI
24 01256 151300         MOVS    2,2     ; REMOVE P-BIT BY
25 01257 151120         MOVZL   2,2     ; MULTIPLYING WITH 512 DEC.
26 01260 060210         NIOC    XTTI
27 01261 172033         ADCZ#   3,2,SNC ; IS KEY BETWEEN OR
28 01262 142032         ADCZ#   2,0,SZC ; EQUAL TO LIMITS ?
29 01263 000402         JMP     WTFUB   ; NO
30 01264 000404         JMP     WTOK    ; YES
31 01265 070433 WTFUB:  DIA     2,FUB   ; SENSE BUTTON ARG
32 01266 147415         AND#    2,1,SNR ; IS ARG PRESSED?
33 01267 000764         JMP     WTNOK   ; NO - SENSE AGAIN
34 01270 152400 WTOK:   SUB     2,2
35 01271 071032         DOA     2,FUN   ; YES - TURN OFF INDICATOR ARG
36 01272 070433         DIA     2,FUB   ; WAIT TILL THE OPERATOR
37 01273 147414         AND#    2,1,SZR ; GETS HIS DIRTY FINGER
38 01274 000776         JMP     .-2     ; OFF THE BUTTON
39 01275 020413         LDA     0,WCH44
40 01276 006041         CCHAR
41 01277 006046         CDICL
42 01300 006044         CDISP
43 01301 015460         PROG            ; "ACTUAL PROGRAM NAME"
44 01302 006050         CHAAT
45 01303 020406         LDA     0,WACSAV
46 01304 002401         JMP     @WTORE  ; RETURN
47 01305 000000 WTORE:  0
48 01306 011000 WLOWL:  11*1000         ; LOW LIMIT CHAR * 219
49 01307 040000 WHIGH:  40*1000         ; HIGH LIMIT CHAR * 219
50 01310 000044 WCH44:  44
51 01311 000000 WACSAV: 0
52
53                ;INSERT ACTUAL PROGRAM NAME AFTER A LABEL "PROG:"
```

```
| 0034  .MAIN
01
02                ;ROUTINE READ SWITCHES.
03                ;SWITCHES ARE READ TO AC2. AC0 AND AC1 ARE SAVED.
04                ;THERE WILL BE A PAUSE FOR 3 SEC TO SEE IF MORE SWITCHES
05                ;ARE CHANGED. LAST SW REG CONTENT IN LSTSW.
06                ;SWITCH SETTINGS ARE REPORTED AT TTY/LPT.
07                ;CALL   CRESW
08                ;       RETURN
09
10 01312 040451 XRESW:  STA     0,SSWR0 ;SAVE AC'S
11 01313 044451         STA     1,SSWR1
12 01314 050451         STA     2,SSWR2 ;FOR PRINT INHIBIT RETURN
13 01315 054445         STA     3,RRESW ;SAVE RETURN
14 01316 024450         LDA     1,LSTSW ;LAST SW REG
15 01317 070477         READS   2       ;NEW SW REG
16 01320 132415         SUB#    1,2,SNR ;CHANGED ?
17 01321 000431         JMP     NRESW   ;NO, RETURN
18 01322 050444 ARESW:  STA     2,LSTSW ;YES, SEE IF MORE CHANGE
19 01323 006061         CWAIT           ;AFTER 3 SECONDS
20 01324 001236         SEC3
21 01325 024441         LDA     1,LSTSW
22 01326 070477         READS   2
23 01327 132414         SUB#    1,2,SZR ;CHANGED ?
24 01330 000772         JMP     ARESW   ;YES, WAIT MORE CHANGING
25 01331 050435         STA     2,LSTSW ;NO, CHANGE FINISHED
26 01332 026436         LDA     1,@XSTAC
27 01333 044436         STA     1,SSTAC ;SAVE PRINT INHIBIT SITU
28 01334 030436         LDA     2,KSTAC ;GET PRINT INHI INITIAL FOR PRINT
29 01335 052433         STA     2,@XSTAC ;SW REG WITHOUT FURTHER CALL CRESW
30 01336 006043         CCRLF
31 01337 006040         CMESS
32 01340 001522         MSWRG           ;SWITCHES: XXXXXX
33 01341 024425         LDA     1,LSTSW
34 01342 006052         CTOCT
35 01343 006424         JSR     @WBZOT  ;WAIT LPT,TTO
36 01344 030425         LDA     2,SSTAC
37 01345 052423         STA     2,@XSTAC ;RESTORE PRINT INHIBIT SITU
38 01346 034414         LDA     3,RRESW ;IF RRESW IS ALMOST EQUAL
39 01347 030421         LDA     2,XSTAC ;SETAC IT WAS PRINT INHIBIT
40 01350 156645         SUBOR   2,3,SNR ;WHO CALLED. DON'T RETURN
41 01351 000405         JMP     BRESW   ;BUT REPEAT THE OUTPUT ROUTINE
42 01352 020411 NRESW:  LDA     0,SSWR0
43 01353 024411         LDA     1,SSWR1 ;RESTORE AC'S
44 01354 030412         LDA     2,LSTSW ;NEW SW POSITION
45 01355 002405         JMP     @RRESW  ;RETURN
46
47 01356 014407 BRESW:  DSZ     SSWR2   ;REPEAT CALL OF ROUTINE
48 01357 020404         LDA     0,SSWR0 ;WHICH WAS INTERRUPTED
49 01360 024404         LDA     1,SSWR1 ;IN PRINT INHIBIT BY CRESW.
50 01361 002404         JMP     @SSWR2  ;RETURN TO MAIN PROG CALL.
51
52 01362 000000 RRESW:  0
53 01363 000000 SSWR0:  0
54 01364 000000 SSWR1:  0
55 01365 000000 SSWR2:  0
56 01366 000000 LSTSW:  0
57 01367 000437 WBZOT:  BZOUT
58 01370 000574 XSTAC:  SETAC
59 01371 000000 SSTAC:  0
60 01372 000403 KSTAC:  403             ;JMP NINHI, ASM VALUE FROM SETAC
61 01373 000000 LALOC:  0               ;LAST LOC IN MEM
62 01374 077777 LAPRG:  077777          ;LAST LOC FOR PROGRAM
63 01375 077635 BINFI:  77635           ;FIRST ADDR OF BINARY LOADER IF 32K.
64 01376 000177 BINLA:  177             ;CONSTANT FOR LAST ADDR IN MEM
65 01377 002756 XFITY:  FITYP
```

```
| 0035  .MAIN
01
02                ;SUBROUTINE TO RELOAD BOOTSTRAP LOADER AND BINARY LOADER.
03                ;DO USE IN HEAD OF ANY PROGRAM AS MANY THINGS ARE INITIALIZED.
04                ;FOR LOADERS INFO SEE LOADER BELOW.
05                ;
06                ;CALL   START IN REBIN AFTER LOADING PROGRAM SELFSTARTING.
07
08 01400 020076 REBIN:  LDA     0,POWZE ;SET RESTART ADDR TO CELL ZERO
09 01401 040000         STA     0,0     ;FOR POWER RESTART.
10 01402 006465         JSR     @XMEXT  ;TRY TO SET MEM EXT FLAG (RC3603 ONLY)
11 01403 006463         JSR     @XMEND  ;STORE MEM END LOC IN HMEND=AC2=XX7600
12 01404 024772         LDA     1,BINLA ;AC1:=177
13 01405 147000         ADD     2,1     ;AC1:=LAST LOC IN MEM=XX7777
14 01406 044765         STA     1,LALOC ;STORE IT FOR PRINT
15 01407 125112         MOVL#   1,1,SZC ;MORE THAN 32K ?
16 01410 024764         LDA     1,LAPRG ;YES, AC1:=077777, LAST LOC FOR PROG
17 01411 034764         LDA     3,BINFI ;NO, LAST LOC IN FIRST 32K
18 01412 137400         AND     1,3     ;AC3:=FIRST ADDR LOADER
19 01413 030534         LDA     2,BINAD ;AC2:=ADDR OF BIN DATA
20 01414 021000 MODAT:  LDA     0,0,2
21 01415 041400         STA     0,0,3   ;MOVE DATA
22 01416 166415         SUB#    3,1,SNR ;LAST DATA ?
23 01417 000404         JMP     GETYP   ;YES
24 01420 151400         INC     2,2
25 01421 175400         INC     3,3     ;NEXT DATA ADDR
26 01422 000772         JMP     MODAT
27 01423 126400 GETYP:  SUB     1,1
28 01424 044742         STA     1,LSTSW ;SET LAST SW REG TO ALL ZERO
29 01425 006752         JSR     @XFITY  ;FIND CPU TYPE
30 01426 006073         CRESW           ;OUTPUT OF SWITCH SETTINGS
31 01427 006436         JSR     @XPCPT  ;PRINT CPU TYPE
32 01430 006044         CDISP
33 01431 001501         MLLCC
34 01432 006040         CMESS
35 01433 001501         MLLOC           ;<15><12>LAST LOC. XX7777
36 01434 024737         LDA     1,LALOC
37 01435 006052         CTOCT
38 01436 006056         CDOCT
39 01437 006061         CWAIT           ;WAIT 3 SEC TO READ MESS. DON'T
40 01440 001236         SEC3            ;USE CDATT AS SW AREN'T SET.
41 01441 006044         CDISP
42 01442 001510         MBILO
43 01443 006040         CMESS
44 01444 001510         MBILO           ;<15><12>BINARY LOADER OK
45 01445 006061         CWAIT
46 01446 001236         SEC3
47 01447 006046         CDICL
48 01450 006044         CDISP
49 01451 015460         PROG
50 01452 006043         CCRLF
51 01453 006040         CMESS
52 01454 015460         PROG            ;ACTUAL PROGRAM NAME
53 01455 006061         CWAIT
54 01456 001236         SEC3
55 01457 062701         CICP    0,1     ;TRY TO SET MEM EXT FLAG (RC3603 ONLY)
56 01460 020003         LDA     0,3     ;CELL 3 = 0 FOR HALT
57 01461 101004         MOV     0,0,SZR ;CELL 3 = 1 FOR
58 01462 002004         JMP     @4      ;SELFSTART ADDR 4
59 01463 002401         JMP     @.+1    ;CHOOSE YOUR OWN START ADDR,
60 01464 004740         SWISA           ;NORMAL HALT, BUT IF NO SWITCH PANEL ?
61
62 01465 003016 XPCPT:  PRTYP
63 01466 001716 XMEND:  GMEND
64 01467 002340 XMEXT:  TMEND
```

```
| 0036  .MAIN
01
02        000001          .TXTM 1          ;RDOS TEXT PACKING MODE
03
04                MELOC:  .TXT !LOADING UNIT OFF! ;"LOADING UNIT OFF"
   01470 046117
   01471 040504
   01472 044516
   01473 043440
   01474 052516
   01475 044524
   01476 020117
   01477 043106
   01500 000000
05
06        000000          .TXTM 0          ;NORMAL TEXT PACKING MODE
07
08                MLLOC:  .TXT !<15><12>LAST LOC. !       ;"<15><12>LAST LOC. "
   01501 005015
   01502 040514
   01503 052123
   01504 046040
   01505 041517
   01506 020056
   01507 000000
09
10                MBILO:  .TXT !<15><12>BINARY LOADER OK!
   01510 005015
   01511 044502
   01512 040516
   01513 054522
   01514 046040
   01515 040517
   01516 042504
   01517 020122
   01520 045517
   01521 000000
11                        ;"<15><12>BINARY LOADER OK"
12
13                MSWRG:  .TXT !SWITCHES: !       ;"SWITCHES: "
   01522 053523
   01523 052111
   01524 044103
   01525 051505
   01526 020072
   01527 000000
14
15                MSAMS:  .TXT ! STARTADDR!       ;" STARTADDR"
   01530 051440
   01531 040524
   01532 052122
   01533 042101
   01534 051104
   01535 000000
16
17                UNTIM:  .TXT !MISERABLE TIMING! ;"MISERABLE TIMING"
   01536 044515
   01537 042523
   01540 040522
   01541 046102
   01542 020105
   01543 044524
   01544 044515
   01545 043516
   01546 000000
```

```
| 0037  .MAIN
01
02                ;BINARY LOADER TS
03                ;ERRORHALT XX7752 FOR OVERWRITE LOADER OR CHECKSUM ERROR.
04                ;READYHALT XX7676 IF LOADED PROG ISN'T SELFSTARTING.
05                ;ERRORBLOCK=IGNORE BLOCK
06                ;REPEAT BLOCK=MULTIPLE DATA BLOCK
07                ;COUNT=WORD COUNT IN BLOCK
08
09 01547 001550 BINAD:  .+1               ;ADDR OF BIN LOADER DATA
10                ;SUBROUTINE TO BUILD A WORD IN AC2
11 01550 054425 BUILD:  STA     3,TEMP1   ;SAVE RETURN
12 01551 004406         JSR     GTCHR     ;GET FIRST BYTE
13 01552 171300         MOVS    3,2       ;PUT INTO LH OF AC2
14 01553 004404         JSR     GTCHR     ;GET NEXT BYTE
15 01554 173300         ADDS    3,2       ;FORM WORD IN AC2
16 01555 143000         ADD     2,0       ;ADD INTO CHECKSUM
17 01556 002417         JMP     @TEMP1    ;AND RETURN
18
19                ;READ A BYTE INTO AC3
20                ;IF SWITCH0=0 USE TELETYPE ELSE USE PTR
21 01557 054417 GTCHR:  STA     3,TEMP2   ;SAVE RETURN
22 01560 034417         LDA     3,SAVE    ;TEST WHICH DEVICE
23 01561 175103         MOVL    3,3,SNC
24 01562 000406         JMP     GTTTI     ;TTI
25 01563 063612         SKPDN   PTR       ;PTR
26 01564 000777         JMP     .-1
27 01565 074412         DIA     3,PTR     ;READ AND START
28 01566 060112         NIOS    PTR
29 01567 002407         JMP     @TEMP2    ;AND RETURN
30
31 01570 063610 GTTTI:  SKPDN   TTI
32 01571 000777         JMP     .-1
33 01572 074410         DIA     3,TTI     ;READ AND START
34 01573 060110         NIOS    TTI
35 01574 002402         JMP     @TEMP2    ;AND RETURN
36 01575 000000 TEMP1:  0
37 01576 000000 TEMP2:  0
38 01577 000000 SAVE:   0
39
40                ;TEST BLOCK TYPE
41 01600 125224 BTEST:  MOVZR   1,1,SZR   ;1=START BLOCK (,END XX)
42 01601 000411         JMP     IGNOR     ;NO, IGNORE BLOCK
43 01602 101004         MOV     0,0,SZR   ;TEST THE CHECKSUM
44 01603 000461         JMP     CHKER     ;ERROR
45 01604 030505         LDA     2,ADDRS   ;GET ADDR
46 01605 062677         IORST             ;DO A RESET
47 01606 151113         MOVL#   2,2,SNC   ;TEST BIT 0
48 01607 001000         JMP     0,2       ;0=START PROGRAM
49 01610 063077         HALT              ;1=HALT
50 01611 000777         JMP     .-1       ;DON'T PROCEED
51
52                ;IGNORE BLOCK
53 01612 004745 IGNOR:  JSR     GTCHR     ;READ UNTIL AN ALL
54 01613 020404         LDA     0,BC377   ;ONES BYTE IS SEEN
55 01614 116404         SUB     0,3,SZR   ;IGNORING ERROR MESS
56 01615 000775         JMP     IGNOR
57 01616 000407         JMP     BLOCK     ;OK, GO INTO BLOCK MODE
58 01617 000377 BC377:  377
59
60                ;START OF PROGRAM BINARY LOADER TS
61 01620 062677 START:  IORST             ;RESET
62 01621 060477         READS   0         ;READ THE SWITCH REGISTER
63 01622 040755         STA     0,SAVE    ;AND SAVE IT FOR GTCHR
64 01623 060110         NIOS TTI          ;START BOTH READERS
65 01624 060112         NIOS PTR
```

```
| 0038  .MAIN
01
02                    ;READ IN A BLOCK
03 01625 004732 BLOCK:  JSR     GTCHR   ;GET A BYTE
04 01626 171305         MOVS    3,2,SNR ;AND TEST FOR NUL
05 01627 000776         JMP     BLOCK   ;YES, KEEP READING
06 01630 004727         JSR     GTCHR   ;OK. GET NEXT BYTE
07 01631 173300         ADDS    3,2     ;AND FORM COUNT. AC2:=# OF WORDS
08 01632 141000         MOV     2,0     ;SET CHECKSUM. AC0:=CHECKSUM TILL NOW
09 01633 145000         MOV     2,1     ;SET COUNTER
10 01634 004714         JSR     BUILD   ;GET ADDRESS
11 01635 050454         STA     2,ADDRS
12 01636 004712         JSR     BUILD   ;ADD IN THE CHECKSUM FROM TAPE
13 01637 125113         MOVL#   1,1,SNC ;TEST BLOCK TYPE
14 01640 000740         JMP     BTEST   ;NOT A DATABLOCK
15 01641 044427         STA     1,COUNT ;STORE WORD COUNT
16
17                    ;READ IN THE DATA BLOCK
18 01642 030734 DATA:   LDA     2,TEMP2 ;LAST STA IN TEMP2 WAS JSR RETURN
19 01643 034423         LDA     3,DIFF  ;ADDR 4 CELLS AFTER PROGRAM START:
20 01644 172400         SUB     3,2     ;AC2:=FIRST ADDR IN LOADER
21 01645 034444         LDA     3,ADDRS ;ADDR IN WHICH TO STORE
22 01646 136400         SUB     1,3     ;ADD NEG WC TC CHECK SPACE
23 01647 172023         ADCZ    3,2,SNC ;FOR WHOLE BLOCK
24 01650 000414         JMP     CHKER   ;NO, HALT THE LOADER
25 01651 030416         LDA     2,BC20  ;IF WC > 20 (OCTAL, NEG)
26 01652 147033         ADDZ#   2,1,SNC ;IT IS A REPEAT BLOCK
27 01653 010415         ISZ     COUNT   ;WHERE WC IS ONE LESS THAN COUNT
28 01654 147022         ADDZ    2,1,SZC ;IF REPEAT BLOCK SKIP NEXT TO READ DATA
29 01655 125113 STORE:  MOVL#   1,1,SNC ;DON'T READ IN NEW DATA IF REPEAT BLOCK
30 01656 004672         JSR     BUILD
31 01657 052432         STA     2,@ADDRS
32 01660 010431         ISZ     ADDRS   ;NEXT ADDR
33 01661 010407         ISZ     COUNT   ;TEST COUNT
34 01662 000773         JMP     STORE   ;MORE DATA
35 01663 101004         MOV     0,0,SZR ;TEST CHECKSUM
36 01664 063077 CHKER:  HALT            ;ERROR IN CHECKSUM, AC0=VALUE
37 01665 000740         JMP     BLOCK   ;OK,GET NEXT BLOCK
38 01666 000004 DIFF:   4
39 01667 000020 BC20:   20              ;REPEAT BLOCKS HAVE WD>20
40 01670 000000 COUNT:  0
41
42                    ;BOOTSTRAP LOADER TS FOR PTR:   ENTER AT BSTRP
43 01671 126440 GET:    SUBO    1,1
44 01672 063612         SKPDN   12      ;10 FOR TTI
45 01673 000777         JMP     .-1
46 01674 060412         DIA     0,12    ;10 FOR TTI
47 01675 060112         NIOS    12      ;10 FOR TTI
48 01676 127100         ADDL    1,1
49 01677 127100         ADDL    1,1
50 01700 107003         ADD     0,1,SNC
51 01701 000771         JMP     GET+1
52 01702 001400         JMP     0,3
53 01703 060112 BSTRP:  NIOS    12      ;10 FOR TTI
54 01704 004765         JSR     GET
55 01705 044402         STA     1,.+2
56 01706 004763         JSR     GET
57 01707 000000         0
58 01710 000000         0               ;FOR BOOTSTRAP
59 01711 000000 ADDRS:  0
60 01712 000706 BEND:   JMP     START   ;START OF BINARY LOADER
```

```
| 0039  .MAIN
01
02                          ;FIND THE TOP OF MEMORY (4K SEGMENTS)
03                          ;STORE HIGHEST USEABLE ADDRESS IN
04                          ;HMEND, PROTECTING THE BIN LOADER.
05                          ;THE CONTENTS IN MEMORY ARE LEFT UNCHANGED.
06                          ;CALL   JSR     GMEND
07                          ;       RETURN
08 01713 010000 DMEND:      10000           ;4K CONSTANT
09 01714 014000 EMEND:      14000           ;6K+1 ADDR
10 01715 004200 FMEND:      004200          ;2K+LOADER CONSTANT
11 01716 054451 GMEND:      STA     3,RSAMS
12 01717 030775             LDA     2,EMEND ;PASS 1 AND 2K SEGMENTS
13 01720 020773             LDA     0,DMEND
14 01721 035000 AMEND:      LDA     3,0,2   ;SAVE LOC CONTENT
15 01722 051000             STA     2,0,2   ;STORE INTO UPPER
16 01723 025000             LDA     1,0,2   ;GET IT BACK
17 01724 146404             SUB     2,1,SZR
18 01725 000405             JMP     BMEND
19 01726 055000             STA     3,0,2   ;RESTORE LOC CONTENT
20 01727 113020             ADDZ    0,2
21 01730 151013 JMEND:      MOV#    2,2,SNC ;CHECK LAST 64K/32K WITH L-SHIFT
22 01731 000770             JMP     AMEND
23 01732 024763 BMEND:      LDA     1,FMEND ;WENT TOO FAR
24 01733 132400             SUB     1,2     ;PROTECT LOADERS
25 01734 050074             STA     2,HMEND ;SET END MEM
26 01735 002432             JMP     @RSAMS
27
28                          ;ROUTINE TO PRINT WHICH STARTADDR SA OF MORE
29                          ;IS USED. INPUT:  SA IN AC1.
30                          ;CALL   CSAMS
31                          ;       RETURN
32 01736 054431 XSAMS:      STA     3,RSAMS
33 01737 044431             STA     1,SSAMS
34 01740 030431             LDA     2,CXLPT ;SET OUTPUT DEVICE ROUTINES
35 01741 052433             STA     2,@XXLPT ;TO WAIT FOR ONLINE DEVICES.
36 01742 030430             LDA     2,CXTTO
37 01743 052432             STA     2,@XXTTO
38 01744 030427             LDA     2,CDIS
39 01745 052431             STA     2,@XFDIS
40 01746 052431             STA     2,@XSDIS
41 01747 006046             CDICL
42 01750 006043             CCRLF
43 01751 024417             LDA     1,SSAMS
44 01752 152520             SUBZL   2,2     ;AC2:=1
45 01753 146400             SUB     2,1     ;SUBTRACT 1 FROM JSR ADDR
46 01754 006056             CDOCT
47 01755 006052             CTOCT           ;PRINT SA
48 01756 006044             CDISP
49 01757 001530             MSAMS
50 01760 006040             CMESS
51 01761 001530             MSAMS           ;XXXXXX STARTADDR
52 01762 006047             CDATT           ;OUTPUT OF SWITCH SETTINGS OFTEN HERE
53 01763 006040             CMESS           ;PRINT CR,LF WITH MESS TO WAIT
54 01764 004743             MCRLF           ;FOR DEVICE READY BEFORE IORST
55 01765 062677             IORST
56 01766 002401             JMP     @RSAMS
57 01767 000000 RSAMS:      0
58 01770 000000 SSAMS:      0
59 01771 001750 CXLPT:      1000.           ;TIMEOUT CONSTANTS FOR
60 01772 000454 CXTTO:      300.            ;OUTPUT DEVICES
61 01773 000050 CDIS:       40.
62 01774 000441 XXLPT:      XLPTT
63 01775 000445 XXTTO:      XTTOT
64 01776 001125 XFDIS:      FDIST
65 01777 001147 XSDIS:      SDIST
```

```
| 0040  .MAIN
01                          ;TABLE TO BE CHANGED TO CORRECT PRINTER ALPHABET.
02
03        002000 .LOC     2000
04        000010 .RDX     8
05
06               TABLE:
07
08               .TXTE?
09 02000 000006 <6><0>
10 02001 000006 <6><0>
11 02002 000006 <6><0>
12 02003 000006 <6><0>
13 02004 000006 <6><0>
14 02005 000006 <6><0>
15 02006 000006 <6><0>
16 02007 000006 <6><0>
17 02010 000006 <6><0>
18 02011 004400 <0><11>
19 02012 005000 <0><12>
20 02013 000006 <6><0>
21 02014 006000 <0><14>
22 02015 106400 <0><15>
23 02016 000006 <6><0>
24 02017 000006 <6><0>
25 02020 000006 <6><0>
26 02021 000006 <6><0>
27 02022 000006 <6><0>
28 02023 000006 <6><0>
29 02024 000006 <6><0>
30 02025 000006 <6><0>
31 02026 000006 <6><0>
32 02027 000006 <6><0>
33 02030 000006 <6><0>
34 02031 000006 <6><0>
35 02032 000006 <6><0>
36 02033 000006 <6><0>
37 02034 000006 <6><0>
38 02035 000006 <6><0>
39 02036 000006 <6><0>
40 02037 000006 <6><0>
41 02040 117400 <0><37>
42 02041 020400 <0><41>
43 02042 021000 <0><42>
44 02043 121400 <0><43>
45 02044 022000 <0><44>
46 02045 122400 <0><45>
47 02046 123000 <0><46>
48 02047 023400 <0><47>
49 02050 024000 <0><50>
50 02051 124400 <0><51>
51 02052 125000 <0><52>
52 02053 025400 <0><53>
53 02054 126000 <0><54>
54 02055 026400 <0><55>
55 02056 027000 <0><56>
56 02057 127400 <0><57>
57 02060 030000 <0><60>
58 02061 130400 <0><61>
59 02062 131000 <0><62>
60 02063 031400 <0><63>
61 02064 132000 <0><64>
62 02065 032400 <0><65>
63 02066 033000 <0><66>
64 02067 133400 <0><67>
65 02070 134000 <0><70>
```

```
| 0041   .MAIN
02 02071 034400 <0><71>
03 02072 035000 <0><72>
04 02073 135400 <0><73>
05 02074 036000 <0><74>
06 02075 136400 <0><75>
07 02076 137000 <0><76>
08 02077 037400 <0><77>
09 02100 140000 <0><100>
10 02101 040400 <0><101>
11 02102 041000 <0><102>
12 02103 141400 <0><103>
13 02104 042000 <0><104>
14 02105 142400 <0><105>
15 02106 143000 <0><106>
16 02107 043400 <0><107>
17 02110 044000 <0><110>
18 02111 144400 <0><111>
19 02112 145000 <0><112>
20 02113 045400 <0><113>
21 02114 146000 <0><114>
22 02115 046400 <0><115>
23 02116 047000 <0><116>
24 02117 147400 <0><117>
25 02120 050000 <0><120>
26 02121 150400 <0><121>
27 02122 151000 <0><122>
28 02123 051400 <0><123>
29 02124 152000 <0><124>
30 02125 052400 <0><125>
31 02126 053000 <0><126>
32 02127 153400 <0><127>
33 02130 154000 <0><130>
34 02131 054400 <0><131>
35 02132 055000 <0><132>
36 02133 155400 <0><133>
37 02134 056000 <0><134>
38 02135 156400 <0><135>
39 02136 157000 <0><136>
40 02137 057400 <0><137>
41 02140 006000 <0><14>
42 02141 040400 <0><101>
43 02142 041000 <0><102>
44 02143 141400 <0><103>
45 02144 042000 <0><104>
46 02145 142400 <0><105>
47 02146 143000 <0><106>
48 02147 043400 <0><107>
49 02150 044000 <0><110>
50 02151 144400 <0><111>
51 02152 145000 <0><112>
52 02153 045400 <0><113>
53 02154 146000 <0><114>
54 02155 046400 <0><115>
55 02156 047000 <0><116>
56 02157 147400 <0><117>
57 02160 050000 <0><120>
58 02161 150400 <0><121>
59 02162 151000 <0><122>
60 02163 051400 <0><123>
61 02164 152000 <0><124>
62 02165 052400 <0><125>
63 02166 053000 <0><126>
64 02167 153400 <0><127>
65 02170 154000 <0><130>
```

```
| 0042  .MAIN
01 02171 054400 <0><131>
02 02172 055000 <0><132>
03 02173 140000 <0><100>
04 02174 121400 <0><43>
05 02175 156400 <0><135>
06 02176 004400 <0><11>
07 02177 000006 <6><0>?
   02200 000000
08       000010 .RDX 8
09                              ;START ADDR FOR "HELP" PROGRAMS
10       002201 .LOC 2201
11
12 02201 002403          SAPTB
13 02202 006777          JSR     @.-1    ;START MOVE PRINTER TABLE
14 02203 002260          SAMEX
15 02204 006777          JSR     @.-1    ;START SET TO 64K MODE, MEM SIZE ?
16 02205 002265          SAMNM
17 02206 006777          JSR     @.-1    ;START SET TO 32K MODE, MEM SIZE ?
18 02207 005011          EXMEM
19 02210 006777          JSR     @.-1    ;START EXAMINE MEMORY
20 02211 005127          DPMEM
21 02212 006777          JSR     @.-1    ;START DEPOSIT MEMORY
22 02213 007076          TROHA
23 02214 006777          JSR     @.-1    ;START TROUBLE BREAKPOINT HALT
24 02215 007110          TROLO
25 02216 006777          JSR     @.-1    ;START TROUBLE BREAKPOINT LOOP REPORT
26 02217 007122          TRORE
27 02220 006777          JSR     @.-1    ;START TROUBLE BREAKPOINT RESET
28 02221 002314          LOADB
29 02222 006777          JSR     @.-1    ;START BINARY LOADER, READ FROM PTR/TTI
30
31              MLPTT:   .TXT !LPT TABLE!        ;"LPT TABLE"
   02223 050114
   02224 020124
   02225 040524
   02226 046102
   02227 000105
32                       ;ROUTINE TO MOVE PRINTER TABLE.
33                       ;INPUT: TABLE # IN AC2
34                       ;CALL:  JSR     MOPTB
35
36 02230 054415 MOPTB:   STA     3,MOPTR ;SAVE RETIRE
37 02231 034417          LDA     3,MOGTT ;ADDR OF POINTER
38 02232 157000          ADD     2,3     ;AC3:=TABLE ENTRY
39 02233 031400          LDA     2,0,3   ;AC2:=ADDR OF TABLE, START
40 02234 024413          LDA     1,MOLAD ;LAST STORE ADDR
41 02235 034411          LDA     3,MOFID ;FIRST STORE ADDR
42 02236 021000 MOREP:   LDA     0,0,2
43 02237 041400          STA     0,0,3   ;MOVE CHAR
44 02240 166415          SUB#    3,1,SNR ;LAST CHAR ?
45 02241 002404          JMP     @MOPTR  ;YES, RETIRE
46 02242 151400          INC     2,2
47 02243 175400          INC     3,3     ;NEXT CHAR ADDR
48 02244 000772          JMP     MOREP   ;REPEAT MOVE CHAR
49 02245 000000 MOPTR:   0               ;RETURN ADDR
50 02246 002000 MOFID:   2000            ;FIRST STORE ADDR
51 02247 002200 MOLAD:   2200            ;LAST STORE ADDR
52 02250 002250 MOGTT:   .               ;GET TABLE ADDR
53 02251 003656          PTAB1           ;ASCII TABLE ADDR
54 02252 004057          PTAB2           ;RC STANDARD 71/78 START .
55 02253 004260          PTAB3           ;RC STANDARD 71/78 START 0
56 02254 004461          PTAB4           ;PL1 TYPE 70 TABLE ADDR
57 02255 000004 ALPTT:   4               ;SUGGESTED ANSWER TABLE #
58 02256 000004 ULPTT:   4               ;UPPER LIMIT
59 02257 000001 LLPTT:   1               ;LOWER LIMIT
```

```
I 0043  .MAIN
01
02                      ;PROGRAMS TO SET MAX MEMORY LOCATION. THEY WILL SET
03                      ;MAX MEM LOC IN HMEND BOTH FOR 32K AND 64K MAX MODE.
04                      ;BUT 64K MAX (MEMORY EXTENSION) MODE WILL BE RESET
05                      ;AFTER EACH IORST. THEREFORE IF THE PROGRAMS SHOULD
06                      ;USE THIS MODE IT SHOULD BE MADE TO CHECK HMEND TO SEE
07                      ;IF IT IS BIGGER THAN 32K. IF SO YOU SHOULD FIRE THE
08                      ;DICP 0,1 COMMAND BEFORE USING MEMORY LOC BEYOND 32 K.
09
10                      ;PROGRAM TO SET MAX MEM LOC TO 64K WORDS MODE.
11
12 02260 165000 SAMEX:  MOV     3,1
13 02261 006072         CSAMS           ;START ADDR MESSAGE
14 02262 004456         JSR     TMEND   ;TRY SET MEM EXT FLAG (RC3603 ONLY)
15 02263 000411         JMP     SAMMS
16 02264 151013 MCMEX:  MOV#    2,2,SNC
17
18                      ;PROGRAM TO SET MAX MEM LOC TO 32K WORDS MODE.
19 02265 165000 SAMNM:  MOV     3,1
20 02266 006072         CSAMS           ;START ADDR MESSAGE
21 02267 024404         LDA     1,MCMNM
22 02270 046423         STA     1,@MIMEX
23 02271 062677         IORST           ;RESET MEM EXT FLAG (RC3603 ONLY)
24 02272 000402         JMP     SAMMS
25 02273 151113 MCMNM:  MOVL#   2,2,SNC
26 02274 006416 SAMMS:  JSR     @IMEND  ;AC2=HMEND=XX7600
27 02275 006044         CDISP
28 02276 001501         MLLOC
29 02277 006040         CMESS
30 02300 001501         MLLOC           ;<15><12>LAST LOC. XX7777
31 02301 020074         LDA     0,HMEND
32 02302 026407         LDA     1,@SAMCO ;AC1:=177
33 02303 107000         ADD     0,1
34 02304 006056         CDOCT
35 02305 006052         CTOCT
36 02306 006047         CDATI
37 02307 002401         JMP     @,+1    ;PROGRAM FINISHED
38 02310 004740         SWISA           ;RESTART MAIN PROGRAM.
39 02311 001376 SAMCO:  BINLA
40 02312 001716 IMEND:  GMEND
41 02313 001730 MIMEX:  JMEND
42
43                      ;ROUTINE RESTORE BINARY LOADER AND
44                      ;START LOADING PTR/TTI DEPENDING ON SWITCH 0.
45
46 02314 165000 LOADB:  MOV     3,1
47 02315 006072         CSAMS           ;START ADDR MESSAGE
48 02316 024755         LDA     1,MCMNM
49 02317 046774         STA     1,@MIMEX ;MAX 32K MODE
50 02320 062677         IORST           ;RESET MEM EXT FLAG
51 02321 006771         JSR     @IMEND  ;AC2=HMEND=0X7600
52 02322 026767         LDA     1,@SAMCO ;AC1:=177
53 02323 147000         ADD     2,1     ;LAST LOC = 0X7777
54 02324 036412         LDA     3,@LOADF ;FIRST ADDR LOADER
55 02325 137400         AND     1,3     ;IN THIS MEM
56 02326 032411         LDA     2,@LOADA ;ADDR OF BINARY DATA
57 02327 021000 LOADR:  LDA     0,0,2
58 02330 041400         STA     0,0,3   ;MOVE DATA
59 02331 166415         SUB#    3,1,SNR ;LAST DATA ?
60 02332 001400         JMP     0,3     ;YES, START BINARY LOADER
61 02333 151400         INC     2,2
62 02334 175400         INC     3,3     ;NEXT DATA ADDR
63 02335 000772         JMP     LOADR
64 02336 001375 LOADF:  BINFI           ;FIRST ADDR BIN LOADER (32K)
65 02337 001547 LOADA:  BINAD           ;ADDR POINTER TO BIN DATA
```

```
I 0044  .MAIN
01
02                          ;ROUTINE TO CHECK FOR MORE THAN 32K WORDS MEMORY.
03                          ;(MEMORY EXTEND OPTION).
04
05 02340 054435 TMEND:  STA     3,RMEND
06 02341 062677         IORST               ;RESET MEM EXT FLAG
07 02342 062701         DICP    0,1         ;TRY TO SET MEM EXT FLAG (RC3603 ONLY).
08 02343 030433         LDA     2,OMEND     ;DON'T USE SKPDN 1 TO CHECK FLAG
09 02344 034433         LDA     3,PMEND     ;AS NOT IMPLEMENTED IN ALL CPU'S
10 02345 021000         LDA     0,0,2       ;SAVE LOWER LOC CONTENT
11 02346 040432         STA     0,QMEND
12 02347 051000         STA     2,0,2       ;STORE LOWER ADDR
13 02350 025400         LDA     1,0,3       ;SAVE UPPER LOC CONTENT
14 02351 044430         STA     1,SMEND
15 02352 055400         STA     3,0,3       ;STORE UPPER ADDR
16 02353 021400         LDA     0,0,3       ;GET IT BACK
17 02354 116414         SUB#    0,3,SZR
18 02355 000411         JMP     UMEND       ;NOT MORE THAN 32K BUT MEM EXT OPTION
19 02356 025000         LDA     1,0,2       ;MORE THAN 32K OR NOT MEM EXT OPTION
20 02357 132415         SUB#    1,2,SNR     ;UPPER STORED IN LOWER (15 BIT ADDR) ?
21 02360 000406         JMP     UMEND       ;NO, MEM EXT OPTION
22 02361 020417         LDA     0,QMEND     ;YES, NOT MEM EXT OPTION, MAX 32K
23 02362 041000         STA     0,0,2       ;RESTORE LOWER LOC CONTENT
24 02363 024710         LDA     1,MCMNM     ;SET 32K MAX
25 02364 046727         STA     1,@MIMEX
26 02365 002410         JMP     @RMEND      ;EXIT
27 02366 020412 UMEND:  LDA     0,QMEND
28 02367 041000         STA     0,0,2       ;RESTORE LOWER LOC CONTENT
29 02370 024411         LDA     1,SMEND
30 02371 045400         STA     1,0,3       ;RESTORE UPPER LOC CONTENT
31 02372 030672         LDA     2,MCMEX     ;SET 64K MAX
32 02373 052720         STA     2,@MIMEX
33 02374 002401         JMP     @RMEND      ;EXIT
34 02375 000000 RMEND:  0                   ;RETURN ADDR
35 02376 002402 OMEND:  VMEND               ;LOWER ADDR
36 02377 102402 PMEND:  @VMEND              ;UPPER ADDR 16 BIT, NOT INDIRECT
37 02400 000000 QMEND:  0                   ;SAVE LOWER CONTENT
38 02401 000000 SMEND:  0                   ;SAVE UPPER CONTENT
39 02402 000000 VMEND:  0                   ;WORK CELL LOWER ADDR
40
41                          ;PROGRAM TO GET NEW PRINTER TABLE
42
43 02403 165000 SAPTB:  MOV     3,1
44 02404 006072         CSAMS               ;START ADDR MESSAGE
45 02405 006071 RLPTT:  CGUES
46 02406 002223         MLPTT               ;LPT TABLE
47 02407 002223         MLPTT
48 02410 002255         ALPTT               ;SUGGESTED ANSWER
49 02411 006053         CTDEC
50 02412 006057         CDDEC
51 02413 006105         CGTDC               ;READ ANSWER
52 02414 000402         JMP     .+2         ;SUGGESTED ACCEPTED
53 02415 000770         JMP     RLPTT       ;ERROR RETURN
54 02416 030075         LDA     2,DIGIN     ;ANSWER INPUT'ED
55 02417 024637         LDA     1,ULPTT     ;UPPER LIMIT
56 02420 020637         LDA     0,LLPTT     ;LOWER LIMIT
57 02421 132033         ADCZ#   1,2,SNC
58 02422 142032         ADCZ#   2,0,SZC     ;AC0=<AC2=<AC1
59 02423 000762         JMP     RLPTT       ;OUTSIDE LIMITS
60 02424 004604         JSR     MOPTB       ;INPUT ACCEPTED, MOVE TABLE
61 02425 002401         JMP     @.+1        ;PROGRAM FINISHED
62 02426 004740         SWISA               ;RESTART MAIN PROGRAM
```

```
| 0045  .MAIN
02
03                ;TAPE 3B
04
05                        ;DELAY SUBROUTINE
06                        ;ARGUMENT FOLLOWING CALL IS ADDRESS
07                        ;OF DELAY CONSTANT.
08                        ;DELAY IN INCREMENTS OF 1 MSEC
09                        ;AC2 & 3 ARE USED, DEFINE WAIT, ARG
10                        ;
11                        ;CALL    CWAIT
12                        ;        ARG
13                        ;
14 02427 054416 XWAIT:    STA      3,WARET ;SAVE RETURN
15 02430 010415           ISZ      WARET   ;PASS ARG
16 02431 033400           LDA      2,@0,3  ;FETCH ARG
17 02432 050414           STA      2,WIRET ;STORE ARG
18 02433 030414           LDA      2,KINDI ;FETCH KNOVA ADDRESS
19 02434 036537           LDA      3,@XCPN ;FETCH CPUNO
20 02435 157000           ADD      2,3     ;COMPUTE KADDRESS
21 02436 031400           LDA      2,0,3   ;FETCH CPU CONSTANT
22 02437 050437 NWAIT:    STA      2,KINC  ;STORE IT FOR INC OF 1 MSEC
23 02440 014436           DSZ      KINC    ;X USEC LOOP
24 02441 000777           JMP      .-1     ;X USEC
25 02442 014404           DSZ      WIRET   ;NO. OF MS STILL TO WAIT
26 02443 000774           JMP      NWAIT   ;COUNT ARG NOT ENDED
27 02444 002401           JMP      @WARET  ;COUNT ARG ENDED, RETURN
28 02445 000000 WARET:    0
29 02446 000000 WIRET:    0
30
31 02447 002450 KINDI:    .+1               ;ADDRESS OF KNOVA
32        000012          .RDX 10
33 02450 000175 KCP0:     125               ;NOVA
34 02451 000001 KCP1:     1                 ;
35 02452 000341 KCP2:     225               ;1200
36 02453 000567 KCP3:     375               ;SUPER
37 02454 000505 KCP4:     325               ;SUPER SC/830
38 02455 000536 KCP5:     350               ;800/NOVA 2-16K
39 02456 000620 KCP6:     400               ;NOVA 2-8K
40 02457 000001 KCP7:     1                 ;CONSTANTS ABOVE ARE FOR GUESSED CPU TYPE
41 02460 000001 KCP10:    1                 ;CONSTANTS BELOW ARE EXACTLY.
42 02461 000200 KCP11:    128               ;NOVA    7,8 USEC
43 02462 000336 KCP12:    222               ;1200    4,5 USEC
44 02463 000601 KCP13:    385               ;800     2,6 USEC
45 02464 000601 KCP14:    385               ;SUPER   2,6 USEC
46 02465 000764 KCP15:    500               ;SUPER SC  2,0 USEC
47 02466 000620 KCP16:    400               ;NOVA2/8K  2,5 USEC
48 02467 000502 KCP17:    322               ;NOVA2/16K 3,1 USEC
49 02470 000531 KCP20:    345               ;RC3603/RC3609           2,90 USEC
50 02471 000334 KCP21:    220               ;RC3603/RC3609/BREAK ON  4,55 USEC
51 02472 000463 KCP22:    307               ;RC3603/RC3608           3,26 USEC
52 02473 000314 KCP23:    204               ;RC3603/RC3608/BREAK ON  4,90 USEC
53 02474 000515 KCP24:    333               ;NOVA2/DATARAM 900 NSEC  3,00 USEC
54 02475 000001 KCP25:    1                 ;FOR TROUBLE, PUT HERE RELEVANT VALUE.
55        000010          .RDX 8
56 02476 000000 KINC:     0
```

```
| 0046  .MAIN
01
02
03                          ;PROCEDURE TIMER ON SKP
04                          ;CALL:  TIMSK
05                          ;       MSEC (>0) TO WAIT MAX FOR
06                          ;       SKP INSTR.
07                          ;       TIME OUT RETURN
08                          ;       NORMAL RETURN
09
10 02477 025400 RTIME:  LDA     1,0,3   ;AC1:=TIME
11 02500 125015         MOV#    1,1,SNR ;IF TIME = ZERO
12 02501 125400         INC     1,1     ;GIVE IT A CHANGE
13 02502 021401         LDA     0,1,3   ;AC0:=INSTR
14 02503 040410         STA     0,STSKP ;STORE INSTR
15 02504 124400         NEG     1,1     ;AC1:=-TIME
16 02505 020417         LDA     0,CPU00 ;FETCH CPU0-ADDR.
17 02506 032465         LDA     2,@XCPN ;FETCH CPUNO
18 02507 113000         ADD     0,2     ;COMPUTE CPU-ADDR.
19 02510 021000         LDA     0,0,2   ;FETCH CPU-CONSTANT
20 02511 040412         STA     0,CPUINC;STORE IT FOR INC OF 1MS
21 02512 020411 STINC:  LDA     0,CPUINC;# OF LOOPS FOR 1 MSEC
22 02513 000000 STSKP:  0               ; X USEC (SKP INSTR.)
23 02514 000402         JMP     .+2     ; X USEC
24 02515 001403         JMP     3,3     ;NORMAL RETURN
25 02516 101404         INC     0,0,SZR ; X USEC
26 02517 000774         JMP     STSKP   ; X USEC
27 02520 125404         INC     1,1,SZR ;COUNT # OF MSEC
28 02521 000771         JMP     STINC   ;MORE MSEC
29 02522 001402         JMP     2,3     ;TIME OUT RETURN, AC0=AC1=0
30
31 02523 000000 CPUINC: 0
32 02524 002525 CPU00:  .+1             ;ADDR. OF CPU0
33       000012         .RDX 10
34 02525 177677 CPU0:   -65             ;NOVA
35 02526 177777 CPU1:   -1
36 02527 177552 CPU2:   -150            ;1200
37 02530 177470 CPU3:   -200            ;SUPER
38 02531 177437 CPU4:   -225            ;SUPER SC/830
39 02532 177406 CPU5:   -250            ;800/NOVA 2-16K
40 02533 177324 CPU6:   -300            ;NOVA 2-8K
41 02534 177777 CPU7:   -1              ;
42 02535 177777 CPU10:  -1              ;
43 02536 177676 CPU11:  -66             ;NOVA    15,2 USEC
44 02537 177550 CPU12:  -152            ;1200     6,6 USEC
45 02540 177371 CPU13:  -263            ;800      3,8 USEC
46 02541 177504 CPU14:  -188            ;SUPER    5,3 USEC
47 02542 177427 CPU15:  -233            ;SUPER SC  4,3 USEC
48 02543 177343 CPU16:  -285            ;NOVA2/8K  3,5 USEC
49 02544 177422 CPU17:  -238            ;NOVA2/16K 4,2 USEC
50 02545 177371 CPU20:  -263            ;RC3603/RC3609           3,80 USEC
51 02546 177565 CPU21:  -139            ;RC3603/RC3609/BREAK ON 7,20 USEC
52 02547 177416 CPU22:  -242            ;RC3603/RC3608           4,14 USEC
53 02550 177571 CPU23:  -135            ;RC3603/RC3608/BREAK ON 7,40 USEC
54 02551 177371 CPU24:  -263            ;NOVA2/DATARAM 900 NSEC 3,80 USEC
55 02552 177777 CPU25:  -1              ;FOR TROUBLE, PUT HERE RELEVANT VALUE.
56       000010         .RDX 8
```

```
| 0047  .MAIN
01
02                          ;ROUTINE TO MESSURE TIME.
03                          ;CALL    TIMMS
04                          ;        SKP INSTR. TO BE MESSURED
05                          ;        TIMEOUTRETURN    TIMEM:=MAX TIME
06                          ;        NORMAL RETURN    TIMEM:=MESSURED TIME
07                          ;THE TIME IS MESSURED IN TENS OF USEC.
08                          ;LOOPS IN THE INSTRUCTIONS MARKED X USEC UNTIL
09                          ;BIT 0 IS SET (TIMEOUT, MAX MESSUREABLE TIME)
10                          ;COUNTING FROM -1 IN THE INC INSTRUCTION OR UNTIL
11                          ;SKP INSTRUCTION TO BE MESSURED IS EFFECTIVE.
12                          ;COULD BE EXTENDED TO TWICE THE TIME POSSIBLE HERE
13                          ;IF CARRY IS USED INSTEAD OF BIT 0, BUT NO PRINT ROUT.
14
15 02553 054477 MSTIM:   STA      3,MSTIR ;SAVE RETURN
16 02554 021400          LDA      0,0,3   ;GET SKP INSTR.
17 02555 040405          STA      0,MSTIB ;TO BE MESSURED.
18 02556 102040          ADCO     0,0     ;AC0:=177777, C:=1
19 02557 101400 MSTIA:   INC      0,0     ;TO AVOID OVERFLOW       X USEC
20 02560 101112          MOVL#    0,0,SZC ;INCREASE LOOP TIME      X USEC
21 02561 000443          JMP      MSTIO   ;TIME OUT, BIT 0=1
22 02562 000000 MSTIB:   0                ;SKP INSTR. TO           X USEC
23 02563 000774          JMP      MSTIA   ;BE TIMED.               X USEC
24 02564 101400          INC      0,0     ;COMPENSATE INITIALIZATION 4 INSTR.
25 02565 101400          INC      0,0     ;COMPENSATE STARTING AT -1
26 02566 040402          STA      0,TIMEX ;TIMECOUNT MESSURED
27 02567 000441          JMP      MSTID   ;CALCULATE, BIT 0=0
28
29 02570 000000 TIMEX:   0                ;TIMECOUNT USED ABOVE
30 02571 000000 TIMCT:   0                ;TIMECONSTANT FROM TABLE BELOW
31 02572 000000 TIMEM:   0                ;TIME MESSURED
32 02573 003015 XCPN:    CPUNO
33
34 02574 002575 TCP00:   .+1              ;100 MSEC CONSTANTS:
35       000012          .RDX 10
36 02575 012574 TCP0:    5500             ;NOVA
37 02576 077777 TCP1:    32767            ;       WILL GIVE TIMEOUT = 100 MSEC
38 02577 030324 TCP2:    12500            ;1200
39 02600 040164 TCP3:    16500            ;SUPER
40 02601 051774 TCP4:    21500            ;SUPER SC/830
41 02602 055714 TCP5:    23500            ;800/NOVA2-16K
42 02603 063604 TCP6:    26500            ;NOVA2-8K
43 02604 077777 TCP7:    32767            ;
44 02605 077777 TCP10:   32767            ;                X USEC:
45 02606 012566 TCP11:   5494             ;NOVA            18,2 USEC
46 02607 030442 TCP12:   12578            ;1200             7,95 USEC
47 02610 060650 TCP13:   25000            ;800              4,0 USEC
48 02611 040011 TCP14:   16393            ;SUPER            6,1 USEC
49 02612 055327 TCP15:   23255            ;SUPER SC         4,6 USEC
50 02613 063314 TCP16:   26316            ;NOVA 2 - 8K      3,8 USEC
51 02614 054307 TCP17:   22727            ;NOVA 2 - 16K     4,4 SEC
52 02615 052001 TCP20:   21505            ;RC3603/RC3609           4,65 USEC
53 02616 030206 TCP21:   12422            ;RC3603/RC3609/BREAK ON 8,05 USEC
54 02617 047545 TCP22:   20325            ;RC3603/RC3608           4,92 USEC
55 02620 027531 TCP23:   12121            ;RC3603/RC3608/BREAK ON 8,25 USEC
56 02621 060650 TCP24:   25000            ;NOVA2/DATARAM 900 NSEC 4,00 USEC
57 02622 077777 TCP25:   32767            ;FOR TROUBLE, PUT HERE RELEVANT VALUE.
58
59 02623 023420 MST10:   10000
60       000010          .RDX 8
61
62 02624 102220 MSTIO:   ADCZR    0,0     ;AC0:=077777, CARRY:=1
63 02625 040743          STA      0,TIMEX ;TIMEOUT TIMECOUNT = 32767
64 02626 000402          JMP      MSTID   ;CALCULATE
```

```
| 0048   ,MAIN
01
02 02627 000000 MSTIC:  0                ;CARRY FLAG
03 02630 101200 MSTID:  MOVR     0,0     ;
04 02631 040776         STA      0,MSTIC ;SAVE CARRY
05 02632 026741         LDA      1,@XCPN ;
06 02633 030741         LDA      2,TCP00 ;
07 02634 133000         ADD      1,2     ;
08 02635 025000         LDA      1,0,2   ;FETCH CPU CONSTANT
09 02636 044733         STA      1,TIMCT ;
10 02637 024731         LDA      1,TIMEX ;
11 02640 030763         LDA      2,MST10 ;
12 02641 006066         MULTI            ;TIMECOUNT X 10000
13 02642 030727         LDA      2,TIMCT ;
14 02643 006070         DIVID            ;DIVIDED BY CPU CONSTANT
15 02644 044726         STA      1,TIMEM ;TIME IN TENS OF USEC
16 02645 020762         LDA      0,MSTIC ;GET CARRY FLAG
17 02646 010404         ISZ      MSTIR   ;PASS SKP ARG.
18 02647 101103         MOVL     0,0,SNC ;IF CARRY THEN TIMEOUT
19 02650 010402         ISZ      MSTIR   ;PASS TIMEOUT RETURN
20 02651 002401         JMP      @MSTIR  ;RETURN
21 02652 000000 MSTIR:  0                ;RETURN ADDR.
22
23                      ;ROUTINE TO MULTIPLY
24                      ;CALL    MULTI
25                      ;        RETURN
26                      ;
27                      ;        AC0, AC1:=AC1*AC2
28                      ;
29 02653 102460 XMULT:  SUBC     0,0     ;MULTIPLY C(1)*(2)
30 02654 054425         STA      3,MSAV  ;RESULT TO C(0),(1)
31 02655 034423         LDA      3,MDCTR ;AC2 UNCHANGED
32 02656 125203 MLOOP:  MOVR     1,1,SNC ;CARRY UNCHANGED
33 02657 101201         MOVR     0,0,SKP
34 02660 143220         ADDZR    2,0
35 02661 175404         INC      3,3,SZR
36 02662 000774         JMP      MLOOP
37 02663 125260         MOVCR    1,1
38 02664 002415         JMP      @MSAV
39
40                      ;ROUTINES TO DIVIDE
41                      ;CALL    DIVIS
42                      ;        RETURN
43                      ;
44                      ;        AC0:=REMAINDER
45                      ;        AC1:=QUOTIENT FOR AC1/AC2
46                      ;
47                      ;CALL    DIVID
48                      ;        RETURN
49                      ;
50                      ;        AC0:=REMAINDER
51                      ;        AC1:=QUOTIENT FOR AC0, AC1/AC2
52                      ;
53 02665 102400 XDIVS:  SUB      0,0     ;DIVIDE      C(1)/C(2)
54 02666 054413 XDIVD:  STA      3,MSAV  ;DIVIDE C(0),C(1)/C(2)
55 02667 034411         LDA      3,MDCTR ;C(0)=REMAINDER
56 02670 125120         MOVZL    1,1     ;C(1)=QUOTIENT
57 02671 101100 DLOOP:  MOVL     0,0     ;AC2 UNCHANGED
58 02672 142412         SUB#     2,0,SZC ;CARRY...?
59 02673 142400         SUB      2,0
60 02674 125100         MOVL     1,1
61 02675 175404         INC      3,3,SZR
62 02676 000773         JMP      DLOOP
63 02677 002402         JMP      @MSAV
64 02700 177760 MDCTR:  -20
65 02701 000000 MSAV:   0
```

```
| 0049  .MAIN
01
02                          ;SUBROUTINE TIMER ON ROUTINE
03                          ;
04                          ;CALL    TIMRO
05                          ;        ADDR. OF TIME TO WAIT MAX (MSEC)
06                          ;        JSR @PAGE ZERO (DEFINITION)      OR JMP .+1
07                          ;        ARGUMENT FOR JSR INSTR.          OR SKP INSTR.
08                          ;        TIMEOUT RETURN
09                          ;        EVENT RETURN
10
11 02702 054430 XTIMS:   STA      3,XTIMR ;
12 02703 033400          LDA      2,@0,3  ;GET TIME, # OF MSEC
13 02704 050425          STA      2,XTIMC ;TO COUNT
14 02705 010425          ISZ      XTIMR
15 02706 032424          LDA      2,@XTIMR ;GET CALL DEFINITION
16 02707 050406          STA      2,XTIMD ;TO USE FOR TIMER
17 02710 010422          ISZ      XTIMR
18 02711 032421          LDA      2,@XTIMR ;GET ARGUMENT
19 02712 050404          STA      2,XTIMA ;FOR CALLED ROUTINE
20 02713 010417          ISZ      XTIMR    ;COUNT RETURN ADDR TO PASS ARG.
21 02714 010415          ISZ      XTIMC    ;COUNT TIME FOR CORRECT USE
22 02715 000401 XTIMD:   JMP      .+1      ;JSR    XX
23 02716 000401 XTIMA:   JMP      .+1      ;ARGUMENT YY
24 02717 000403          JMP      XTIMT    ;NOT YET RETURN, TEST TIMER
25 02720 010412          ISZ      XTIMR    ;EVENT RETURN, COUNT RETURN ADDR.
26 02721 002411          JMP      @XTIMR   ;TO PASS ERROR RETURN.
27 02722 014407 XTIMT:   DSZ      XTIMC    ;TIMECOUNT STILL NOT FINISHED ?
28 02723 000402          JMP      XTIMW    ;STALL 1 MSEC
29 02724 002406          JMP      @XTIMR   ;TIMEOUT RETURN
30 02725 006061 XTIMW:   CWAIT
31 02726 002730          XTIM1
32 02727 000766          JMP      XTIMD    ;TEST EVENT
33 02730 000001 XTIM1:   1                 ;1 MSEC CONSTANT
34 02731 000000 XTIMC:   0                 ;X MSEC COUNTER
35 02732 000000 XTIMR:   0                 ;RETURN ADDR.
36
37                       UNTST:  .TXT !, RUN RC 3600 INSTRUCTION TIMER TEST!
   02733 020054
   02734 052522
   02735 020116
   02736 041522
   02737 031440
   02740 030066
   02741 020060
   02742 047111
   02743 052123
   02744 052522
   02745 052103
   02746 047511
   02747 020116
   02750 044524
   02751 042515
   02752 020122
   02753 042524
   02754 052123
   02755 000000
38                          ;", RUN RC 3600 INSTRUCTION TIMER TEST"
```

```
I 0050  .MAIN
01
02                          ;ROUTINE TO FIND TYPE OF CPU.
03                          ;CALLED BY REBIN
04                          ;THIS ROUTINE IS SETTING A NUMBER INDICATING
05                          ;TYPE AND SPEED OF CPU INTO CPUNO. SEE
06                          ;ROUTINE TO MESSURE TIME FOR DEFINITION OF #.
07
08 02756 054432 FITYP:  STA     3,RETYP
09 02757 062677         IORST
10 02760 020434         LDA     0,NUTYP ;NUMBER OF TRYING AGAIN
11 02761 040432         STA     0,AGTYP
12 02762 004447 TRTYP:  JSR     TYMER
13 02763 125014         MOV#    1,1,SZR ;AC1=0, SEARCH MORE
14 02764 000422         JMP     STTYP
15 02765 006424         JSR     @XOMER
16 02766 125014         MOV#    1,1,SZR ;AC1=0, SEARCH MORE
17 02767 000417         JMP     STTYP
18 02770 014423         DSZ     AGTYP   ;COUNT DOWN # OF
19 02771 000771         JMP     TRTYP   ;RETRY TO IDENTIFY CPU
20 02772 006420         JSR     @XWTYP
21 02773 044422         STA     1,CPUNO
22 02774 006044         CDISP
23 02775 001536         UNTIM
24 02776 006040         CMESS
25 02777 001536         UNTIM           ;MISERABLE TIMING
26 03000 006040         CMESS
27 03001 002733         UNTST           ;RUN INSTR TIMER TEST
28 03002 006043         CCRLF
29 03003 006061         CWAIT           ;WAIT 3 SEC TO READ MESS. DON'T
30 03004 001236         SEC3            ;USE CDATT AS SW AREN'T SET.
31 03005 002403         JMP     @RETYP  ;EXIT
32 03006 044407 STTYP:  STA     1,CPUNO
33 03007 002401         JMP     @RETYP  ;EXIT
34 03010 000000 RETYP:  0
35 03011 003275 XOMER:  TOMER
36 03012 003475 XWTYP:  NWTYP
37 03013 000000 AGTYP:  0
38 03014 000007 NUTYP:  7               ;# OF CALLS OF TYMER, TOMER.
39 03015 000006 CPUNO:  6               ;TYPE OF CPU, NOVA 2-8K SHOWN
40
41 03016 054772 PRTYP:  STA     3,RETYP ;PRINT CPU TYPE NUMBER
42 03017 006044         CDISP
43 03020 005002         MCPUT
44 03021 006040         CMESS
45 03022 005002         MCPUT           ;<15><12>CPU TYPE:
46 03023 024772         LDA     1,CPUNO
47 03024 006052         CTOCT
48 03025 006056         CDOCT
49 03026 006061         CWAIT           ;WAIT 3 SEC TO READ MESS. DON'T
50 03027 001236         SEC3            ;USE CDATT AS SW AREN'T SET.
51 03030 002760         JMP     @RETYP  ;EXIT
```

```
| 0051  .MAIN
01
02 03031 054572 TYMER:  STA     3,SVTYME
03 03032 020576         LDA     0,XTYME ;GET ADDRESS OF TYME LIST.
04 03033 040021         STA     0,IDX1  ;STORE IN AUTO POINTER.
05 03034 020575         LDA     0,XINST ;GET ADDR OF INSTRUCTION LIST
06 03035 040020         STA     0,IDX0  ;STORE IN AUTO POINTER.
07 03036 030567         LDA     2,INXW5
08 03037 050023         STA     2,IDX3  ;INITIALIZE IDX3 FOR LDA INST.
09 03040 152440         SUBO    2,2     ;AC2:=0, NULL CHAR
10 03041 071011         DOA     2,XTTO  ;SEND CHAR
11 03042 060111         NIOS    XTTO    ;START TTO AND
12 03043 063511         SKPBZ   XTTO    ;SYNCHRONIZE
13 03044 000777         JMP     .-1     ;WITH TTO CLOCK
14 03045 071011         DOA     2,XTTO  ;SEND A CHAR
15 03046 060111         NIOS    XTTO    ;AND MESSURE TIME, NOT ACCURATE
16 03047 006064         TIMMS           ;CPU TYPE NOT KNOWN BUT TO GIVE AN IDEA
17 03050 063511         SKPBZ   XTTO    ;OF TTO SPEED
18 03051 000401         JMP     .+1     ;AC0 IS MESSURED TIME
19 03052 022544         LDA     0,@TYMEM
20 03053 024544         LDA     1,TYLIM ;LIMIT TO DESTINGV. SPEED
21 03054 034544         LDA     3,TYTTS ;SLOW TTY CONSTANT
22 03055 106432         SUBZ#   0,1,SZC ;IS TTY FAST ?
23 03056 034543         LDA     3,TYTTF ;YES, FAST TTY CONSTANT
24 03057 054535         STA     3,TYMEN ;STORE # OF COUNTS
25 03060 152440         SUBO    2,2     ;AC2:=0, NULL CHAR
26 03061 071011         DOA     2,XTTO  ;SEND CHAR
27 03062 060111         NIOS    XTTO    ;START TTO AND
28 03063 063511         SKPBZ   XTTO    ;SYNCHRONIZE PROGRAM
29 03064 000777         JMP     .-1     ;WITH TTY CLOCK.
30 03065 152440 TYMA:   SUBO    2,2     ;CLEAR AC2
31 03066 141000         MOV     2,0     ;CLEAR AC0 ALSO.
32 03067 026020         LDA     1,@IDX0 ;GET INST. FROM LIST
33 03070 125015         MOV#    1,1,SNR ;ZERO MARKS END OF INSTR. LIST
34 03071 000455         JMP     SCORE   ;
35 03072 071011         DOA     2,XTTO  ;SEND NULL CHAR, STARTING LATER
36 03073 034451         LDA     3,CSKP  ;GET ALC-SKP MASK AND
37 03074 137400         AND     1,3     ;AND WITH INSTR.
38 03075 175123         MOVZL   3,3,SNC ;CKN BIT 0, THE ALC BIT
39 03076 000403         JMP     TYMD    ;BIT 0 = 0 MEANS NO ALC CODE.
40 03077 175004         MOV     3,3,SZR ;CKN 3 LSB'S FOR SKP CODE.
41 03100 000411         JMP     TYMB    ;FOUND ALC-SKP CODE.
42 03101 044403 TYMD:   STA     1,TYMJ  ;STORE INSTR. IN TIME LOOP.
43 03102 060111         NIOS    XTTO    ;START TTO, FOR NON ALC-SKKP INSTR.
44 03103 151400 TYMF:   INC     2,2     ;THESE **********************
45 03104 000000 TYMJ:   0               ;   INSTRUCTIONS ************
46 03105 063511         SKPBZ   XTTO    ;      FORM THE *************
47 03106 000775         JMP     TYMF    ;         TIMING LOOP *******
48 03107 052021 TYMC:   STA     2,@IDX1 ;STORE COUNT INTO TYME LIST.
49 03110 000755         JMP     TYMA    ;LOOP.
50 03111 044403 TYMB:   STA     1,TYMH  ;STORE ALC-SKP IN TIME LOOP.
51 03112 060111         NIOS    XTTO    ;START TTO
52 03113 151400 TYMG:   INC     2,2     ;THESE **********************
53 03114 000000 TYMH:   0               ;   INSTRUCTIONS ************
54 03115 000401         JMP     .+1     ;      FORM THE *************
55 03116 063511         SKPBZ   XTTO    ;         TIMING ************
56 03117 000774         JMP     TYMG    ;            LOOP ***********
57 03120 000767         JMP     TYMC    ;FOR ALC-SKP INSTRUCTIONS.
58
59 03121 000000 INSW1:  0               ;HERE WORKS ISZ INSAD FROM TYMJ
60 03122 100005 INDW1:  #5              ;HERE WORKS LDA @INDAD FROM TYMJ
```

```
| 0052  ,MAIN
01
02 03123 101000 INSTR:  MOV     0,0       ;THIS IS THE 16 INSTRUCTION
03 03124 103000         ADD     0,0       ;LIST, SELECTED TO DEVELOPE
04 03125 103401         AND     0,0,SKP   ;AN IDENTITY PROFILE OF THE
05 03126 020005         LDA     0,5       ;PROCESSOR IN THE COURSE OF
06 03127 040005         STA     0,5       ;BEING EXECUTED. THE INSTR. ARE
07 03130 010415         ISZ     INSAD     ;LATER STORED IN TYMJ,TYMH
08 03131 000401         JMP     INSTA
09 03132 004401 INSTA:  JSR     INSTB
10 03133 022005 INSTB:  LDA     0,@5
11 03134 022023         LDA     0,@IDX3
12 03135 022416         LDA     0,@INDAD  ;LABEL INDAD PLACED .+16
13 03136 060400         DIA     0,0
14 03137 061000         DOA     0,0
15 03140 061477         INTA    0
16 03141 063400         SKPBN   0
17 03142 060100         NIOS    0
18 03143 000000         0                 ;END OF INSTR. LIST MARKER
19 03144 100007 CSKP:   100007            ;ONLY FOR ASSEMBLING ISZ INSAD
20 03145 000000 INSAD:  0                 ;TO WORK IN INSW1 AND INSW2.
21
22 03146 102520 SCORE:  SUBZL   0,0       ;SET THE
23 03147 040455         STA     0,ORDINAL ;ORDINAL COUNTER TO +1.
24 03150 020460 SCORA:  LDA     0,XTYME   ;GET TYME LIST INITIAL
25 03151 040021         STA     0,IDX1    ;ADDRESS FOR AUTO INC.
26 03152 020455         LDA     0,XC20    ;SET UP THE X16 COUNTER.
27 03153 040453 INDAD:  STA     0,XX16    ;WITH A COUNT OF 16 DEC.
28 03154 026021         LDA     1,@IDX1   ;GET TYME ENTRY INTO AC1 AS FIRST
29                                        ;BIG TYME. THEN SEARCH FOR BIGGER TYME.
30 03155 022021 SCORB:  LDA     0,@IDX1   ;CK MAGNITYDE OF NX TYME ENTRY.
31 03156 106033         ADCZ#   0,1,SNC   ;SKPS IF AC0 < AC1
32 03157 105000         MOV     0,1       ;AC0 > OR = AC1, AC1 = BIGGEST TYME.
33 03160 014446         DSZ     XX16      ;COUNT DOWN # OF TYMES.
34 03161 000774         JMP     SCORB     ;STILL MORE TYMES, SO LOOP.
35
36                      ;REVIEW CLEARS ALL BIG TYMES. AC1 = THE BIGGEST TYME NOW.
37                      ;LOC'S = AC1 OR (AC1-1) OR (AC1-2) ARE CLEARED AND THERE
38                      ;POS IN RANK LIST ARE SET TO THE (C) OF THE ORDINAL COUNT
39 03162 125015 REVU:   MOV#    1,1,SNR   ;IF A SCORE PASS IS COMPL WITH
40 03163 000566         JMP     KEYS      ;AC1 = 0, TYME IS CLEAR, SO GO FORM KEY.
41 03164 020444         LDA     0,XTYME   ;REINITIALIZE RVTMP WITH
42 03165 040435         STA     0,RVTMP   ;TYME - 1.
43 03166 020444         LDA     0,XRANK   ;INITIALIZE IDX2 WITH
44 03167 040022         STA     0,IDX2    ;RANK - 1.
45 03170 020437         LDA     0,XC20    ;RESET X16 COUNTER
46 03171 040435         STA     0,XX16    ;BACK TO 16 DECIMAL.
47 03172 010430 REVUA:  ISZ     RVTMP     ;INC TYMES LIST POINTER.
48 03173 030421         LDA     2,TYMEN   ;GET TOLERANCE COUNT
49 03174 022426         LDA     0,@RVTMP  ;GET TYME ENTRY AND
50 03175 106415 REVUD:  SUB#    0,1,SNR   ;COMP WITH LARGEST TYME ENTRY.
51 03176 000411         JMP     REVUB     ;IF BIGGEST TYME, STORE ORDINAL.
52 03177 101400         INC     0,0       ;ADD +1 AND COMP AGAIN WITH BIG TYME.
53 03200 151404         INC     2,2,SZR   ;TEST FOR TOLERANCE BIG TYME 0,-1,-2,...
54 03201 000774         JMP     REVUD     ;IF BIGGEST TYME -1,-2,-3...-N, STORE ORD
55 03202 022022         LDA     0,@IDX2   ;ENTRY OUT OF RANGE, INC RANK POINTER.
56 03203 014423 REVUC:  DSZ     XX16      ;DECREMENT THE X16 POINTER.
57 03204 000766         JMP     REVUA     ;STILL MORE TYMES, SO LOOP.
58 03205 010417         ISZ     ORDINAL   ;INC ORDINAL COUNT.
59 03206 000742         JMP     SCORA     ;GO TO SCORA TO FIND NX BIG TYME.
60 03207 030415 REVUB:  LDA     2,ORDINAL ;GET ORDINAL COUNT
61 03210 052022         STA     2,@IDX2   ;AND STORE IT IN PRESENT RANK LOC.
62 03211 176440         SUBO    3,3       ;CLEARING AC3.
63 03212 056410         STA     3,@RVTMP  ;CLEARS LOC. IN TYME LIST.
64 03213 000770         JMP     REVUC     ;
```

```
I 0053  ,MAIN
01
02 03214 177775 TYMEN:  -3                      ;-# OF COUNTS, TOLERENCE FOR TYMES
03 03215 177767 TYRTC:  -9.                     ;-3 WILL GIVE BIG, BIG-1, BIG-2 IS EQUAL.
04 03216 002572 TYMEM:  TIMEM                   ;MESSURED TIME FOR ONE TTO CHAR
05 03217 001212 TYLIM:  650.                    ;LIMIT TO TELL SPEED > 1600 BAUD
06 03220 177772 TYTTS:  -6                      ;COUNT TOLERANCE SLOW TTY
07 03221 177775 TYTTF:  -3                      ;COUNT TOLERANCE FAST TTY
08 03222 000000 RVTMP:  0
09 03223 000000 SVTYME: 0
10 03224 000000 ORDINAL: 0
11 03225 000005 INXW5:  5                       ;START LDA 0,@IDX3 IN CELL 5
12 03226 000000 XX16:   0
13 03227 000020 XC20:   20
14 03230 003232 XTYME:  TYME-1
15 03231 003122 XINST:  INSTR-1
16 03232 003253 XRANK:  RANK-1
17       000020 TYME:   .BLK 20
18 03253 000000 ENTYM:  0                       ;END OF TYME LIST MARKER.
19       000020 RANK:   .BLK 20
20 03274 000000 ENRNK:  0                       ;END OF RANK LIST MARKER.
21
22 03275 054726 TOMER:  STA     3,SVTYME
23 03276 020717         LDA     0,TYRTC ;GET RTC TOLERANCE
24 03277 040715         STA     0,TYMEN ;STORE # OF COUNTS
25 03300 020730         LDA     0,XTYME ;GET ADDRESS OF TYME LIST.
26 03301 040021         STA     0,IDX1  ;STORE IN AUTO POINTER.
27 03302 020727         LDA     0,XINST ;GET ADDR OF INSTRUCTION LIST
28 03303 040020         STA     0,IDX0  ;STORE IN AUTO POINTER.
29 03304 030721         LDA     2,INXW5
30 03305 050023         STA     2,IDX3  ;INITIALIZE IDX3 FOR LDA INST.
31 03306 102520         SUBZL   0,0
32 03307 101120         MOVZL   0,0     ;AC0:=2
33 03310 061014         DOA     0,XRTC  ;SET RTC FREQUENCY TO 100 HZ (10 MSEC).
34 03311 060114         NIOS    XRTC
35 03312 063514         SKPBZ   XRTC    ;SYNCHRONIZE PROGRAM
36 03313 000777         JMP     .-1     ;WITH RTC.
37 03314 152440 TOMA:   SUBO    2,2     ;CLEAR AC2
38 03315 141000         MOV     2,0     ;CLEAR AC0 ALSO.
39 03316 026020         LDA     1,@IDX0 ;GET INST. FROM LIST
40 03317 125015         MOV#    1,1,SNR ;ZERO MARKS END OF INSTR. LIST
41 03320 000626         JMP     SCORE   ;
42 03321 034623         LDA     3,CSKP  ;GET ALC-SKP MASK AND
43 03322 137400         AND     1,3     ;AND WITH INSTR.
44 03323 175123         MOVZL   3,3,SNC ;CKN BIT 0, THE ALC BIT
45 03324 000403         JMP     TOMD    ;BIT 0 = 0 MEANS NO ALC CODE.
46 03325 175004         MOV     3,3,SZR ;CKN 3 LSB'S FOR SKP CODE.
47 03326 000411         JMP     TOMB    ;FOUND ALC-SKP CODE.
48 03327 044403 TOMD:   STA     1,TOMJ  ;STORE INSTR. IN TIME LOOP.
49 03330 060114         NIOS    XRTC    ;START RTC
50 03331 151400 TOMF:   INC     2,2     ;THESE **********************
51 03332 000000 TOMJ:   0               ;   INSTRUCTIONS ************
52 03333 063514         SKPBZ   XRTC    ;      FORM THE *************
53 03334 000775         JMP     TOMF    ;         TIMING LOOP *******
54                                      ;FOR NON ALC-SKP INSTR.
55 03335 052021 TOMC:   STA     2,@IDX1 ;STORE COUNT INTO TYME LIST.
56 03336 000756         JMP     TOMA    ;LOOP.
57 03337 044403 TOMB:   STA     1,TOMH  ;STORE ALC-SKP IN TIME LOOP.
58 03340 060114         NIOS    XRTC    ;START RTC
59 03341 151400 TOMG:   INC     2,2     ;THESE **********************
60 03342 000000 TOMH:   0               ;   INSTRUCTIONS ************
61 03343 000401         JMP     .+1     ;      FORM THE *************
62 03344 063514         SKPBZ   XRTC    ;         TIMING ************
63 03345 000774         JMP     TOMG    ;            LOOP ***********
64 03346 000767         JMP     TOMC    ;FOR ALC-SKP INSTRUCTIONS.
65 03347 000000 INSW2:  0               ;HERE WORKS ISZ INSAD FROM TOMJ.
66 03350 100000 INDW2:  @0              ;HERE WORKS LDA @INDAD FROM TOMJ.
```

```
I 0054  .MAIN
01
02 03351 020661 KEYS:   LDA     0,XRANK ;REINITIALIZE RANK LIST
03 03352 040022         STA     0,IDX2  ;AUTO INC POINTER.
04 03353 102520         SUBZL   0,0     ;BIT 15:=1 AS END OF KEY MARKER.
05
06 03354 026022 KEYA:   LDA     1,@IDX2 ;GET ORDINAL COUNT FROM RANK LIST.
07 03355 125223         MOVZR   1,1,SNC ;CK IF # IS ODD OR EVEN.
08 03356 000404         JMP     KEYB    ;EVEN
09 03357 101122         MOVZL   0,0,SZC ;ROTATE 0 INTO BIT 15 FOR ODD ORDINAL.
10 03360 000405         JMP     SESAME  ;CARRY BIT SET MEANS END OF KEY.
11 03361 000773         JMP     KEYA    ;MORE-ON-KEY.
12 03362 101142 KEYB:   MOVOL   0,0,SZC ;ROTATE 1 INTO BIT 15 FOR EVEN ORDINAL.
13 03363 000402         JMP     SESAME  ;CARRY BIT SET MEANS END OF KEY.
14 03364 000770         JMP     KEYA    ;MORE-ON-KEY.
15
16 03365 000401 SESAME: JMP     .+1     ;FOR TROUBLE HALT
17 03366 024415         LDA     1,SES11 ;GET FIRST CPUNO
18 03367 034415         LDA     3,LOCKA ;ADDR OF LOCK TABLE
19 03370 031400 SESA1:  LDA     2,0,3   ;GET KEY FROM TABLE
20 03371 151015         MOV#    2,2,SNR ;ZERO ?
21 03372 000407         JMP     SESOUT  ;YES, END MARK: KEY NOT KNOWN
22 03373 150015         COM#    2,2,SNR ;ALL ONES ?
23 03374 125400         INC     1,1     ;YES, NEXT CPU MARK, INC CPUNO
24 03375 112435         SUBZ#   0,2,SNR ;COMPARE KEY WITH LOCK ENTRY
25 03376 000404         JMP     SESEX   ;A HIT ! RETURN WITH CPUNO IN AC1
26 03377 175400         INC     3,3     ;INC TO NEXT LOCK ENTRY
27 03400 000770         JMP     SESA1   ;AND LOOP
28 03401 126440 SESOUT: SUBO    1,1     ;CLEAR AC1 AS KEY INDICATOR FOR NO
29 03402 002621 SESEX:  JMP     @SVTYME ;CPU FOUND, RETURN.
30 03403 000011 SES11:  11
31
32 03404 003405 LOCKA:  .+1
33 03405 064450         064450          ;NOVA KEY.                CPU #   11
34 03406 177777         177777
35 03407 016137         016137          ;NOVA 1200 KEY.                   12
36 03410 177777         177777
37 03411 034174         034174          ;NOVA 800 KEY.                    13
38 03412 177777         177777
39 03413 002544         002544          ;SUPERNOVA KEY.                   14
40 03414 177777         177777
41 03415 007126         007126          ;SUPERNOVA SC KEY.                15
42 03416 177777         177777
43 03417 034652         034652          ;NOVA 2 - 8K KEY.                 16
44 03420 177777         177777
45 03421 022512         022512          ;NOVA 2 - 16K KEY.                17
46 03422 177777         177777
47 03423 170225         170225          ;RC3603/RC3609, 16K KEY           20
48 03424 170265         170265
49 03425 170365         170365
50 03426 172225         172225
51 03427 172265         172265
52 03430 177777         177777
53 03431 144567         144567          ;RC3603/RC3609/BREAK ON KEY       21
54 03432 146427         146427
55 03433 166610         166610
56 03434 177777         177777
```

```
I 0055  .MAIN
01
02 03435 166065          166065          ;RC3603/RC3608, 32K KEY          22
03 03436 166165          166165
04 03437 166225          166225
05 03440 166325          166325
06 03441 172065          172065
07 03442 172325          172325
08 03443 174050          174050
09 03444 174150          174150
10 03445 174210          174210
11 03446 174250          174250
12 03447 174310          174310
13 03450 176010          176010
14 03451 177777          177777
15 03452 144352          144352          ;RC3603/RC3608/BREAK ON KEY      23
16 03453 154352          154352
17 03454 160135          160135
18 03455 162025          162025
19 03456 162202          162202
20 03457 162225          162225
21 03460 166025          166025
22 03461 170312          170312
23 03462 170352          170352
24 03463 172012          172012
25 03464 172052          172052
26 03465 172202          172202
27 03466 176025          176025
28 03467 176050          176050
29 03470 177777          177777
30 03471 020653          020653          ;NOVA 2 - DATARAM 900 NSEC KEY   24
31 03472 177777          177777
32 03473 000000          0               ;SPACE IF TROUBLES               25
33                                       ;CORRECT JMP .+1 IN SESAME TO A HALT
34                                       ;RESTART IN 1400 (REBIN). PUT HERE
35                                       ;KEY FROM AC0 EXAMINED WHEN STOPPED,
36                                       ;AND CORRECT CPUNO 25'S CONSTANTS IN THE
37                                       ;3 ROUTINES CWAIT, TIMSK, TIMMS
38 03474 000000          0               ;END OF TABLE, NO CPU FOUND, #   0
```

```
I 0056  .MAIN
01
02                          ;FIND TYPE OF CPU USING RTC
03                          ;PART OF CPUTYP
04 03475 054444 NWTYP:  STA     3,NIRET ;SAVE RETURN
05 03476 102520         SUBZL   0,0     ;AC0:=1
06 03477 101120         MOVZL   0,0     ;AC0:=2
07 03500 126400         SUB     1,1
08 03501 061014         DOA     0,XRTC  ;SET RTC FREQUENCY
09 03502 060114         NIOS    XRTC    ;TO 100 HZ (10 MSEC)
10 03503 063514         SKPBZ   XRTC
11 03504 000777         JMP     .-1     ;SYNCHRONIZE RTC
12 03505 060114         NIOS    XRTC    ;START RTC
13 03506 125400         INC     1,1     ;COUNTS          ;X USEC
14 03507 063514         SKPBZ   XRTC    ;                ;X USEC
15 03510 000776         JMP     .-2     ;LOOP FOR 10 MS  ;X USEC
16 03511 004465         JSR     NTTYP   ;TEST RTC STABILITY
17 03512 030520         LDA     2,NN500 ;STEP -500
18 03513 034520         LDA     3,NN9   ;8 TIMES
19 03514 141000         MOV     2,0     ;STARTING AT -500=SPEED
20 03515 175405 NXTYP:  INC     3,3,SNR ;NEXT STEP OR
21 03516 000413         JMP     NFTYP   ;TYPE OF NOVA NOT FOUND
22 03517 143020         ADDZ    2,0     ;SPEED:=SPEED-500
23 03520 107013         ADD#    0,1,SNC ;NO. OF COUNTS>=(-SPEED)
24 03521 000774         JMP     NXTYP   ;THEN GO TO NXTYP ELSE
25 03522 030512         LDA     2,NC8   ;CONVERT TIMES TO CPUNO
26 03523 173000         ADD     3,2     ;AS MENTIONED BELOW
27 03524 034427         LDA     3,NEXIS ;CHECK IN
28 03525 157000         ADD     2,3     ;THE TABEL THAT
29 03526 025400         LDA     1,0,3   ;THE FOUND CPUNO
30 03527 125004         MOV     1,1,SZR ;EXIST
31 03530 000403         JMP     NRTYP   ;YES, RETURN
32 03531 006043 NFTYP:  CCRLF           ;NO
33 03532 004432 NITYP:  JSR     NOTYP   ;GET OPERATORS CPUNO
34 03533 050405 NRTYP:  STA     2,NCTYP ;SAVE CPUNO
35 03534 006046         CDICL
36 03535 006043         CCRLF
37 03536 024402         LDA     1,NCTYP ;RESTORE CPUNO
38 03537 002402         JMP     @NIRET  ;EXIT
39 03540 000000 NCTYP:  0
40 03541 000000 NIRET:  0
41              NOTYM:  .TXT ! SET CPUNO > AC2! ;" SET CPUNO > AC2"
   03542 051440
   03543 052105
   03544 041440
   03545 052520
   03546 047516
   03547 037040
   03550 040440
   03551 031103
   03552 000000
42                          ;RANGE          LOOPS OF X USEC CPUNO   TYPE
43 03553 003554 NEXIS:  .+1     ;   0- 500
44 03554 000001         1       ; 500-1000       794    12,6    0       NOVA
45 03555 000000         0       ;1000-1500                      1
46 03556 000001         1       ;1500-2000      1724     5,80   2       3608BREAK
47                              ;               1770     5,65   2       3609BREAK
48                              ;               1905     5,25   2       12XX
49 03557 000001         1       ;2000-2500      2222     4,5    3       SUPER
50 03560 000001         1       ;2500-3000      2702     3,7    4       SUPER SC
51                              ;               2778     3,6    4       830
52 03561 000001         1       ;3000-3500      3012     3,32   5       3603/3608
53                              ;               3125     3,2    5       NOVA2-16K
54                              ;               3226     3,10   5       3603/3609
55                              ;               3333     3,0    5    800,820,840
56                              ;               3448     2,9    5    N-2-DATARAM
57 03562 000001         1       ;3500-4000      3703     2,7    6       NOVA2-8K
58 03563 000000         0       ;4000-4500                      7
```

```
| 0057  .MAIN
01
02 03564 054411 NOTYP:  STA     3,NOTRE
03 03565 006046         CDICL           ;CLEAR DIS
04 03566 006044         CDISP           ;DIS - MESSAGE
05 03567 003542         NOTYM
06 03570 006040         CMESS
07 03571 003542         NOTYM           ; SET CPUNO > AC2
08 03572 006043         CCRLF           ;NO DISATT, WAIT ROUT, NO CONSTANT
09 03573 063077         HALT            ;PUT CPUNO INTO AC2
10 03574 002401         JMP     @NOTRE  ;CONTINUE
11 03575 000000 NOTRE:  0
12
13 03576 020437 NTTYP:  LDA     0,NC125 ;TEST STABILITY
14 03577 040431         STA     0,NTDEC ;125 TIMES
15 03600 044431         STA     1,NTRES ;STORE FIRST RESULT
16 03601 054426         STA     3,NTRET
17 03602 126440 NTREP:  SUBO    1,1     ;AC1:=0
18 03603 060114         NIOS    XRTC    ;START RTC
19 03604 125400         INC     1,1     ;COUNTS
20 03605 063514         SKPBZ   XRTC    ;LOOP FOR 10 MSEC
21 03606 000776         JMP     .-2
22 03607 034422         LDA     3,NTRES ;GET FIRST COUNT
23 03610 161220         MOVZR   3,0     ;50 %
24 03611 101220         MOVZR   0,0     ;25 %
25 03612 101220         MOVZR   0,0     ;12,5 %
26 03613 101220         MOVZR   0,0     ;6,25 %
27 03614 101220         MOVZR   0,0     ;AC0:=3,12 %
28 03615 117000         ADD     0,3     ;AC3:=103,12 %
29 03616 030413         LDA     2,NTRES
30 03617 112400         SUB     0,2     ;AC2:=96,88 %
31 03620 166433         SUBZ#   3,1,SNC
32 03621 132432         SUBZ#   1,2,SZC
33 03622 000414         JMP     NYTYP   ;OUTSIDE TOLERANCE
34 03623 014405         DSZ     NTDEC   ;97 %<COUNT<103 %
35 03624 000756         JMP     NTREP   ;TRY 125 TIMES
36 03625 024404         LDA     1,NTRES ;GET FIRST RESULT
37 03626 002401         JMP     @NTRET  ;EXIT
38 03627 000000 NTRET:  0
39 03630 000000 NTDEC:  0
40 03631 000000 NTRES:  0
41       000012         .RDX 10
42 03632 177014 NN500:  -500
43 03633 177767 NN9:    -9
44 03634 000010 NC8:    8
45 03635 000175 NC125:  125
46       000010         .RDX 8
47 03636 006046 NYTYP:  CDICL
48 03637 006044         CDISP
49 03640 003645         NZTYP           ;RTC IS UNSTABLE,
50 03641 006043         CCRLF
51 03642 006040         CMESS
52 03643 003645         NZTYP           ;NO DISATT, WAIT ROUT, NO CONSTANT
53 03644 000666         JMP     NITYP
54
55              NZTYP:  .TXT !RTC IS UNSTABLE,! ;"RTC IS UNSTABLE,"
   03645 052122
   03646 020103
   03647 051511
   03650 052440
   03651 051516
   03652 040524
   03653 046102
   03654 026105
   03655 000000
```

```
| 0058   .MAIN
01
02          000010 .RDX     8
03                 PTAB1:
04                 .TXTE?
05  03656  000006 <6><0>
06  03657  000006 <6><0>
07  03660  000006 <6><0>
08  03661  000006 <6><0>
09  03662  000006 <6><0>
10  03663  000006 <6><0>
11  03664  000006 <6><0>
12  03665  000006 <6><0>
13  03666  000006 <6><0>
14  03667  004400 <0><11>
15  03670  005000 <0><12>
16  03671  000006 <6><0>
17  03672  006000 <0><14>
18  03673  106400 <0><15>
19  03674  000006 <6><0>
20  03675  000006 <6><0>
21  03676  000006 <6><0>
22  03677  000006 <6><0>
23  03700  000006 <6><0>
24  03701  000006 <6><0>
25  03702  000006 <6><0>
26  03703  000006 <6><0>
27  03704  000006 <6><0>
28  03705  000006 <6><0>
29  03706  000006 <6><0>
30  03707  000006 <6><0>
31  03710  000006 <6><0>
32  03711  000006 <6><0>
33  03712  000006 <6><0>
34  03713  000006 <6><0>
35  03714  000006 <6><0>
36  03715  000006 <6><0>
37  03716  117400 <0><37>
38  03717  020400 <0><41>
39  03720  021000 <0><42>
40  03721  121400 <0><43>
41  03722  022000 <0><44>
42  03723  122400 <0><45>
43  03724  123000 <0><46>
44  03725  023400 <0><47>
45  03726  024000 <0><50>
46  03727  124400 <0><51>
47  03730  125000 <0><52>
48  03731  025400 <0><53>
49  03732  126000 <0><54>
50  03733  026400 <0><55>
51  03734  027000 <0><56>
52  03735  127400 <0><57>
53  03736  030000 <0><60>
54  03737  130400 <0><61>
55  03740  131000 <0><62>
56  03741  031400 <0><63>
57  03742  132000 <0><64>
58  03743  032400 <0><65>
59  03744  033000 <0><66>
60  03745  133400 <0><67>
61  03746  134000 <0><70>
62  03747  034400 <0><71>
63  03750  035000 <0><72>
64  03751  135400 <0><73>
65  03752  036000 <0><74>
66  03753  136400 <0><75>
67  03754  137000 <0><76>
```

```
| 0059   .MAIN
01 03755 037400 <0><77>
02 03756 140000 <0><100>
03 03757 040400 <0><101>
04 03760 041000 <0><102>
05 03761 141400 <0><103>
06 03762 042000 <0><104>
07 03763 142400 <0><105>
08 03764 143000 <0><106>
09 03765 043400 <0><107>
10 03766 044000 <0><110>
11 03767 144400 <0><111>
12 03770 145000 <0><112>
13 03771 045400 <0><113>
14 03772 146000 <0><114>
15 03773 046400 <0><115>
16 03774 047000 <0><116>
17 03775 147400 <0><117>
18 03776 050000 <0><120>
19 03777 150400 <0><121>
20 04000 151000 <0><122>
21 04001 051400 <0><123>
22 04002 152000 <0><124>
23 04003 052400 <0><125>
24 04004 053000 <0><126>
25 04005 153400 <0><127>
26 04006 154000 <0><130>
27 04007 054400 <0><131>
28 04010 055000 <0><132>
29 04011 155400 <0><133>
30 04012 056000 <0><134>
31 04013 156400 <0><135>
32 04014 157000 <0><136>
33 04015 057400 <0><137>
34 04016 006000 <0><14>
35 04017 040400 <0><101>
36 04020 041000 <0><102>
37 04021 141400 <0><103>
38 04022 042000 <0><104>
39 04023 142400 <0><105>
40 04024 143000 <0><106>
41 04025 043400 <0><107>
42 04026 044000 <0><110>
43 04027 144400 <0><111>
44 04030 145000 <0><112>
45 04031 045400 <0><113>
46 04032 146000 <0><114>
47 04033 046400 <0><115>
48 04034 047000 <0><116>
49 04035 147400 <0><117>
50 04036 050000 <0><120>
51 04037 150400 <0><121>
52 04040 151000 <0><122>
53 04041 051400 <0><123>
54 04042 152000 <0><124>
55 04043 052400 <0><125>
56 04044 053000 <0><126>
57 04045 153400 <0><127>
58 04046 154000 <0><130>
59 04047 054400 <0><131>
60 04050 055000 <0><132>
61 04051 140000 <0><100>
62 04052 121400 <0><43>
63 04053 156400 <0><135>
64 04054 004400 <0><11>
65 04055 000006 <6><0>?
   04056 000000
```

```
| 0060   .MAIN
01
02          000012 .RDX     10
03                 PTAB2:
04                 .TXTE?
05  04057  000006 <6><0>
06  04060  000006 <6><0>
07  04061  000006 <6><0>
08  04062  000006 <6><0>
09  04063  000006 <6><0>
10  04064  000006 <6><0>
11  04065  000006 <6><0>
12  04066  000006 <6><0>
13  04067  000006 <6><0>
14  04070  004400 <0><9>
15  04071  005000 <0><10>
16  04072  000006 <6><0>
17  04073  006000 <0><12>
18  04074  106400 <0><13>
19  04075  000006 <6><0>
20  04076  000006 <6><0>
21  04077  000006 <6><0>
22  04100  000006 <6><0>
23  04101  000006 <6><0>
24  04102  000006 <6><0>
25  04103  000006 <6><0>
26  04104  000006 <6><0>
27  04105  000006 <6><0>
28  04106  000006 <6><0>
29  04107  000006 <6><0>
30  04110  000006 <6><0>
31  04111  000006 <6><0>
32  04112  000006 <6><0>
33  04113  000006 <6><0>
34  04114  000006 <6><0>
35  04115  000006 <6><0>
36  04116  000006 <6><0>
37  04117  117400 <0><31>
38  04120  036000 <0><60>
39  04121  137000 <0><62>
40  04122  040400 <0><65>
41  04123  131000 <0><50>
42  04124  034400 <0><57>
43  04125  035000 <0><58>
44  04126  136400 <0><61>
45  04127  030000 <0><48>
46  04130  130400 <0><49>
47  04131  135400 <0><59>
48  04132  027000 <0><46>
49  04133  020400 <0><33>
50  04134  127400 <0><47>
51  04135  120000 <0><32>
52  04136  133400 <0><55>
53  04137  022000 <0><36>
54  04140  122400 <0><37>
55  04141  123000 <0><38>
56  04142  023400 <0><39>
57  04143  024000 <0><40>
58  04144  124400 <0><41>
59  04145  125000 <0><42>
60  04146  025400 <0><43>
61  04147  126000 <0><44>
62  04150  026400 <0><45>
63  04151  021000 <0><34>
64  04152  121400 <0><35>
65  04153  032400 <0><53>
66  04154  132000 <0><52>
67  04155  033000 <0><54>
```

```
| 0061   .MAIN
01  04156  037400  <0><63>
02  04157  041000  <0><66>
03  04160  057400  <0><95>
04  04161  157000  <0><94>
05  04162  156400  <0><93>
06  04163  056000  <0><92>
07  04164  155400  <0><91>
08  04165  055000  <0><90>
09  04166  054400  <0><89>
10  04167  154000  <0><88>
11  04170  153400  <0><87>
12  04171  053000  <0><86>
13  04172  052400  <0><85>
14  04173  152000  <0><84>
15  04174  051400  <0><83>
16  04175  151000  <0><82>
17  04176  150400  <0><81>
18  04177  050000  <0><80>
19  04200  147400  <0><79>
20  04201  047000  <0><78>
21  04202  046400  <0><77>
22  04203  146000  <0><76>
23  04204  045400  <0><75>
24  04205  145000  <0><74>
25  04206  144400  <0><73>
26  04207  142400  <0><69>
27  04210  042000  <0><68>
28  04211  141400  <0><67>
29  04212  044000  <0><72>
30  04213  043400  <0><71>
31  04214  143000  <0><70>
32  04215  031400  <0><51>
33  04216  134000  <0><56>
34  04217  006000  <0><12>
35  04220  057400  <0><95>
36  04221  157000  <0><94>
37  04222  156400  <0><93>
38  04223  056000  <0><92>
39  04224  155400  <0><91>
40  04225  055000  <0><90>
41  04226  054400  <0><89>
42  04227  154000  <0><88>
43  04230  153400  <0><87>
44  04231  053000  <0><86>
45  04232  052400  <0><85>
46  04233  152000  <0><84>
47  04234  051400  <0><83>
48  04235  151000  <0><82>
49  04236  150400  <0><81>
50  04237  050000  <0><80>
51  04240  147400  <0><79>
52  04241  047000  <0><78>
53  04242  046400  <0><77>
54  04243  146000  <0><76>
55  04244  045400  <0><75>
56  04245  145000  <0><74>
57  04246  144400  <0><73>
58  04247  142400  <0><69>
59  04250  042000  <0><68>
60  04251  141400  <0><67>
61  04252  041000  <0><66>
62  04253  040400  <0><65>
63  04254  143000  <0><70>
64  04255  004400  <0><9>
65  04256  000006  <6><0>?
    04257  000000
```

```
I 0062  .MAIN
01
02          000012 .RDX     10
03                 PTAB3:
04                 .TXTE?
05  04260  000006 <6><0>
06  04261  000006 <6><0>
07  04262  000006 <6><0>
08  04263  000006 <6><0>
09  04264  000006 <6><0>
10  04265  000006 <6><0>
11  04266  000006 <6><0>
12  04267  000006 <6><0>
13  04270  000006 <6><0>
14  04271  004400 <0><9>
15  04272  005000 <0><10>
16  04273  000006 <6><0>
17  04274  006000 <0><12>
18  04275  106400 <0><13>
19  04276  000006 <6><0>
20  04277  000006 <6><0>
21  04300  000006 <6><0>
22  04301  000006 <6><0>
23  04302  000006 <6><0>
24  04303  000006 <6><0>
25  04304  000006 <6><0>
26  04305  000006 <6><0>
27  04306  000006 <6><0>
28  04307  000006 <6><0>
29  04310  000006 <6><0>
30  04311  000006 <6><0>
31  04312  000006 <6><0>
32  04313  000006 <6><0>
33  04314  000006 <6><0>
34  04315  000006 <6><0>
35  04316  000006 <6><0>
36  04317  000006 <6><0>
37  04320  117400 <0><31>
38  04321  134000 <0><56>
39  04322  035000 <0><58>
40  04323  136400 <0><61>
41  04324  027000 <0><46>
42  04325  032400 <0><53>
43  04326  033000 <0><54>
44  04327  034400 <0><57>
45  04330  126000 <0><44>
46  04331  026400 <0><45>
47  04332  133400 <0><55>
48  04333  125000 <0><42>
49  04334  156400 <0><93>
50  04335  025400 <0><43>
51  04336  056000 <0><92>
52  04337  031400 <0><51>
53  04340  120000 <0><32>
54  04341  020400 <0><33>
55  04342  021000 <0><34>
56  04343  121400 <0><35>
57  04344  022000 <0><36>
58  04345  122400 <0><37>
59  04346  123000 <0><38>
60  04347  023400 <0><39>
61  04350  024000 <0><40>
62  04351  124400 <0><41>
63  04352  157000 <0><94>
64  04353  057400 <0><95>
65  04354  130400 <0><49>
66  04355  030000 <0><48>
67  04356  131000 <0><50>
```

```
| 0063  .MAIN
01 04357 135400 <0><59>
02 04360 137000 <0><62>
03 04361 155400 <0><91>
04 04362 055000 <0><90>
05 04363 054400 <0><89>
06 04364 154000 <0><88>
07 04365 153400 <0><87>
08 04366 053000 <0><86>
09 04367 052400 <0><85>
10 04370 152000 <0><84>
11 04371 051400 <0><83>
12 04372 151000 <0><82>
13 04373 150400 <0><81>
14 04374 050000 <0><80>
15 04375 147400 <0><79>
16 04376 047000 <0><78>
17 04377 046400 <0><77>
18 04400 146000 <0><76>
19 04401 045400 <0><75>
20 04402 145000 <0><74>
21 04403 144400 <0><73>
22 04404 044000 <0><72>
23 04405 043400 <0><71>
24 04406 143000 <0><70>
25 04407 142400 <0><69>
26 04410 040400 <0><65>
27 04411 140000 <0><64>
28 04412 037400 <0><63>
29 04413 042000 <0><68>
30 04414 141400 <0><67>
31 04415 041000 <0><66>
32 04416 127400 <0><47>
33 04417 132000 <0><52>
34 04420 006000 <0><12>
35 04421 155400 <0><91>
36 04422 055000 <0><90>
37 04423 054400 <0><89>
38 04424 154000 <0><88>
39 04425 153400 <0><87>
40 04426 053000 <0><86>
41 04427 052400 <0><85>
42 04430 152000 <0><84>
43 04431 051400 <0><83>
44 04432 151000 <0><82>
45 04433 150400 <0><81>
46 04434 050000 <0><80>
47 04435 147400 <0><79>
48 04436 047000 <0><78>
49 04437 046400 <0><77>
50 04440 146000 <0><76>
51 04441 045400 <0><75>
52 04442 145000 <0><74>
53 04443 144400 <0><73>
54 04444 044000 <0><72>
55 04445 043400 <0><71>
56 04446 143000 <0><70>
57 04447 142400 <0><69>
58 04450 040400 <0><65>
59 04451 140000 <0><64>
60 04452 037400 <0><63>
61 04453 137000 <0><62>
62 04454 136400 <0><61>
63 04455 041000 <0><66>
64 04456 004400 <0><9>
65 04457 000006 <6><0>?
   04460 000000
```

```
I 0064  .MAIN
01
02          000010 .RDX     8
03                 PTAB4:
04                 .TXTE?
05  04461  000006 <6><0>
06  04462  000006 <6><0>
07  04463  000006 <6><0>
08  04464  000006 <6><0>
09  04465  000006 <6><0>
10  04466  000006 <6><0>
11  04467  000006 <6><0>
12  04470  000006 <6><0>
13  04471  000006 <6><0>
14  04472  004400 <0><11>
15  04473  005000 <0><12>
16  04474  000006 <6><0>
17  04475  006000 <0><14>
18  04476  106400 <0><15>
19  04477  000006 <6><0>
20  04500  000006 <6><0>
21  04501  000006 <6><0>
22  04502  000006 <6><0>
23  04503  000006 <6><0>
24  04504  000006 <6><0>
25  04505  000006 <6><0>
26  04506  000006 <6><0>
27  04507  000006 <6><0>
28  04510  000006 <6><0>
29  04511  000006 <6><0>
30  04512  000006 <6><0>
31  04513  000006 <6><0>
32  04514  000006 <6><0>
33  04515  000006 <6><0>
34  04516  000006 <6><0>
35  04517  000006 <6><0>
36  04520  000006 <6><0>
37  04521  117400 <0><37>
38  04522  022000 <0><44>
39  04523  025400 <0><53>
40  04524  157000 <0><136>
41  04525  047000 <0><116>
42  04526  156400 <0><135>
43  04527  136400 <0><75>
44  04530  127400 <0><57>
45  04531  026400 <0><55>
46  04532  027000 <0><56>
47  04533  045400 <0><113>
48  04534  137000 <0><76>
49  04535  146000 <0><114>
50  04536  145000 <0><112>
51  04537  044000 <0><110>
52  04540  046400 <0><115>
53  04541  037400 <0><77>
54  04542  140000 <0><100>
55  04543  040400 <0><101>
56  04544  041000 <0><102>
57  04545  141400 <0><103>
58  04546  042000 <0><104>
59  04547  142400 <0><105>
60  04550  143000 <0><106>
61  04551  043400 <0><107>
62  04552  144400 <0><111>
63  04553  126000 <0><54>
64  04554  122400 <0><45>
65  04555  021000 <0><42>
66  04556  030000 <0><60>
67  04557  124400 <0><51>
```

```
| 0065   .MAIN
01 04560 125000 <0><52>
02 04561 057400 <0><137>
03 04562 147400 <0><117>
04 04563 050000 <0><120>
05 04564 150400 <0><121>
06 04565 151000 <0><122>
07 04566 051400 <0><123>
08 04567 152000 <0><124>
09 04570 052400 <0><125>
10 04571 053000 <0><126>
11 04572 153400 <0><127>
12 04573 154000 <0><130>
13 04574 054400 <0><131>
14 04575 055000 <0><132>
15 04576 155400 <0><133>
16 04577 056000 <0><134>
17 04600 130400 <0><61>
18 04601 131000 <0><62>
19 04602 031400 <0><63>
20 04603 132000 <0><64>
21 04604 032400 <0><65>
22 04605 033000 <0><66>
23 04606 133400 <0><67>
24 04607 134000 <0><70>
25 04610 034400 <0><71>
26 04611 035000 <0><72>
27 04612 135400 <0><73>
28 04613 036000 <0><74>
29 04614 020400 <0><41>
30 04615 123000 <0><46>
31 04616 023400 <0><47>
32 04617 024000 <0><50>
33 04620 121400 <0><43>
34 04621 006000 <0><14>
35 04622 147400 <0><117>
36 04623 050000 <0><120>
37 04624 150400 <0><121>
38 04625 151000 <0><122>
39 04626 051400 <0><123>
40 04627 152000 <0><124>
41 04630 052400 <0><125>
42 04631 053000 <0><126>
43 04632 153400 <0><127>
44 04633 154000 <0><130>
45 04634 054400 <0><131>
46 04635 055000 <0><132>
47 04636 155400 <0><133>
48 04637 056000 <0><134>
49 04640 130400 <0><61>
50 04641 131000 <0><62>
51 04642 031400 <0><63>
52 04643 132000 <0><64>
53 04644 032400 <0><65>
54 04645 033000 <0><66>
55 04646 133400 <0><67>
56 04647 134000 <0><70>
57 04650 034400 <0><71>
58 04651 035000 <0><72>
59 04652 135400 <0><73>
60 04653 036000 <0><74>
61 04654 057400 <0><137>
62 04655 157000 <0><136>
63 04656 023400 <0><47>
64 04657 004400 <0><11>
65 04660 000006 <6><0>?
   04661 000000
```

```
| 0066  .MAIN
01
02                          ;POWER RESTART ROUTINE
03
04 04662 026454 POWON:  LDA     1,@PSTAC ;GET PRINT INHI INITIAL FOR PRINT
05 04663 046454          STA     1,@PSETP
06 04664 062677          IORST
07 04665 006061          CWAIT           ;WAIT 3 SECONDS TO TERMINAL READY
08 04666 001236          SEC3
09 04667 020442          LDA     0,PCOTT ;PRINT 5 CR, LF FOR TTY, SILENT
10 04670 040440          STA     0,PCOUN
11 04671 006043          CCRLF
12 04672 014436          DSZ     PCOUN
13 04673 000776          JMP     .-2
14 04674 020436          LDA     0,PCH14 ;FF FOR LPT AND CLEARING SOME CRT'S
15 04675 006041          CCHAR
16 04676 006061          CWAIT           ;20 MSEC FOR CRT
17 04677 001237          SECM2
18 04700 020433          LDA     0,PCH35 ;HOME UP FOR CRT
19 04701 006041          CCHAR
20 04702 006061          CWAIT           ;20 MSEC FOR CRT
21 04703 001237          SECM2
22 04704 020430          LDA     0,PCH37 ;ERASE EOF FOR CRT
23 04705 006041          CCHAR
24 04706 006061          CWAIT           ;20 MSEC FOR CRT
25 04707 001237          SECM2
26 04710 006040          CMESS
27 04711 004745          MPOWO           ;POWER
28 04712 006043          CCRLF
29 04713 006040          CMESS
30 04714 015460          PROG            ;ACTUAL PROG NAME
31 04715 006071 RPSAQ:   CQUES
32 04716 004755          MSAQU           ;SET SWITCHES, START ADDR
33 04717 004750          DSAQU
34 04720 004735          PSAAN           ;SUGGESTED ANSWER
35 04721 006054          CTZOC
36 04722 006060          CDZOC
37 04723 006104          CGTOK           ;READ ANSWER
38 04724 000402          JMP     .+2     ;SUGGESTED ACCEPTED BY OPERATOR
39 04725 000770          JMP     RPSAQ   ;ERROR RETURN
40 04726 030075          LDA     2,DIGIN ;ANSWER INPUT'ED
41 04727 001000          JMP     0,2     ;START PROG
42
43 04730 000000 PCOUN:  0               ;COUNTER
44 04731 000005 PCOTT:  5               ;5 CR,LF
45 04732 000014 PCH14:  14              ;FF
46 04733 000035 PCH35:  35              ;HOME UP
47 04734 000037 PCH37:  37              ;ERASE EOF
48 04735 000400 PSAAN:  400             ;SUGGESTED START ADDR
49 04736 001372 PSTAC:  KSTAC           ;ASM VALUE OF JMP NINHI
50 04737 000574 PSETP:  SETAC           ;PRINT INHIBIT ROUTINE
51
52                          ;INITIAL START ADDRESS ROUTINE
53                          ;USED TO HELP START WITHOUT SWITCHES
54
55 04740 026776 SWISA:  LDA     1,@PSTAC ;GET PRINT INHI INITIAL FOR PRINT
56 04741 046776          STA     1,@PSETP
57 04742 000753          JMP     RPSAQ   ;USE POWER RESTART ROUTINE
```

```
| 0067  .MAIN
01
02                MCRLF:  .TXT !<15><12>! ;"<15><12>"
   04743 005015
   04744 000000
03
04                MPOWO:  .TXT !POWER!    ;"POWER"
   04745 047520
   04746 042527
   04747 000122
05
06                DSAQU:  .TXT !SEE3.2,SA!        ;"SEE3.2,SA"
   04750 042523
   04751 031505
   04752 031056
   04753 051454
   04754 000101
07
08                MSAQU:  .TXT !SET SWITCHES TO CONTROL, (3.2), STARTADDR!
   04755 042523
   04756 020124
   04757 053523
   04760 052111
   04761 044103
   04762 051505
   04763 052040
   04764 020117
   04765 047503
   04766 052116
   04767 047522
   04770 026114
   04771 024040
   04772 027063
   04773 024462
   04774 020054
   04775 052123
   04776 051101
   04777 040524
   05000 042104
   05001 000122
09                      ;"SET SWITCHES TO CONTROL, (3.2), STARTADDR"
10
11                MCPUT:  .TXT !<15><12>CPU TYPE: !       ;"<15><12>CPU TYPE: "
   05002 005015
   05003 050103
   05004 020125
   05005 054524
   05006 042520
   05007 020072
   05010 000000
```

```
| 0068  .MAIN
01
02                          ;ROUTINE TO EXAMINE MEMORY.
03
04 05011 165000 EXMEM:  MOV     3,1       ;
05 05012 006072         CSAMS             ;START ADDR MESSAGE
06 05013 006071 EXMMF:  CQUES
07 05014 005070         MXMMF             ;EXAMINE MEM FROM
08 05015 005101         DXMMF             ;SEE NEXT QUESTION, TO (INCL)
09 05016 005064         XFROM             ;SUGGESTED ANSWER
10 05017 006052         CTOCT
11 05020 006056         CDOCT
12 05021 006104         CGTOK             ;READ ANSWER
13 05022 000402         JMP     .+2       ;SUGGESTED ACCEPTED
14 05023 000770         JMP     EXMMF     ;ERROR RETURN
15 05024 020075         LDA     0,DIGIN   ;ANSWER INPUT'ED
16                                        ;MOVR# 0,0,SZC ;EVEN ?
17                                        ;JMP   EXMMF   ;NO, ERROR
18                                        ;LDA 1,UPPERLIMIT
19                                        ;LDA 2,LOWERLIMIT
20                                        ;ADCZ# 1,0,SNC
21                                        ;ADCZ# 0,2,SZC ;AC2=<AC0=<AC1 ?
22                                        ;JMP   EXMMF   ;OUTSIDE LIMITS
23 05025 040440         STA     0,FMADR   ;INPUT ACCEPTED
24 05026 006071 EXMMT:  CQUES
25 05027 005106         MXMMT             ;TO INCL.
26 05030 005106         MXMMT             ;USE THE SAME MESS AT DIS
27 05031 005066         XTOIN             ;SUGGESTED ANSWER
28 05032 006052         CTOCT
29 05033 006056         CDOCT
30 05034 006104         CGTOK             ;READ ANSWER
31 05035 000402         JMP     .+2       ;SUGGESTED ACCEPTED
32 05036 000770         JMP     EXMMT     ;ERROR RETURN
33 05037 020075         LDA     0,DIGIN   ;ANSWER INPUT'ED
34 05040 040427         STA     0,LMADR   ;INPUT ACCEPTED
35 05041 006043 EXPRT:  CCRLF
36 05042 006046         CDICL
37 05043 024422         LDA     1,FMADR
38 05044 006052         CTOCT
39 05045 006056         CDOCT
40 05046 030417         LDA     2,FMADR
41 05047 025000         LDA     1,0,2
42 05050 006052         CTOCT
43 05051 006056         CDOCT
44 05052 006047         CDATT
45 05053 024414         LDA     1,LMADR
46 05054 030411         LDA     2,FMADR
47 05055 010410         ISZ     FMADR     ;TO NEXT LOC
48 05056 000401         JMP     .+1
49 05057 132414         SUB#    1,2,SZR   ;LAST LOC ?
50 05060 000761         JMP     EXPRT     ;NO, NEXT
51 05061 006043         CCRLF             ;YES, PROGRAM FINISHED
52 05062 002401         JMP     @.+1
53 05063 004740         SWISA             ;RESTART MAIN PROGRAM
54
55 05064 000034 XFROM:  34                ;FIRST MEM LOC QUES
56 05065 000000 FMADR:  0                 ;ANSWER
57 05066 000037 XTOIN:  37                ;LAST MEM LOC QUES, INCL
58 05067 000000 LMADR:  0                 ;ANSWER
```

```
I 0069  .MAIN
01
02                      MXMMF:  .TXT !EXAMINE MEM FROM !        ;"EXAMINE MEM FROM "
   05070 054105
   05071 046501
   05072 047111
   05073 020105
   05074 042515
   05075 020115
   05076 051106
   05077 046517
   05100 000040
03
04                      DXMMF:  .TXT !X M FROM !        ;"X M FROM "
   05101 020130
   05102 020115
   05103 051106
   05104 046517
   05105 000040
05
06                      MXMMT:  .TXT !TO INCL. !        ;"TO INCL. "
   05106 047524
   05107 044440
   05110 041516
   05111 027114
   05112 000040
07
08                      MDMMC:  .TXT !DEPOSIT: !        ;"DEPOSIT: "
   05113 042504
   05114 047520
   05115 044523
   05116 035124
   05117 000040
09
10                      MDMMF:  .TXT !FROM LOC !        ;"FROM LOC "
   05120 051106
   05121 046517
   05122 046040
   05123 041517
   05124 000040
11
12 05125 177777 DPCON:  177777          ;DEPOSIT CONTENT QUES
13 05126 000000 CMADR:  0               ;ANSWER
```

```
| 0070  .MAIN
01
02                          ;ROUTINE TO DEPOSIT IN MEMORY.
03
04 05127 165000 DPMEM:  MOV     3,1
05 05130 006072         CSAMS           ;START ADDR MESSAGE
06 05131 006071 DPMMC:  CQUES
07 05132 005113         MDMMC           ;DEPOSIT:
08 05133 005113         MDMMC
09 05134 005125         DPCON           ;SUGGESTED ANSWER
10 05135 006052         CTOCT
11 05136 006056         CDOCT
12 05137 006104         CGTOK           ;READ ANSWER
13 05140 000402         JMP     .+2     ;SUGGESTED ACCEPTED
14 05141 000770         JMP     DPMMC   ;ERROR RETURN
15 05142 020075         LDA     0,DIGIN ;ANSWER INPUT'ED
16 05143 040763         STA     0,CMADR ;INPUT ACCEPTED
17 05144 006071 DPMMF:  CQUES
18 05145 005120         MDMMF           ;FROM LOC
19 05146 005120         MDMMF
20 05147 005064         XFROM           ;SUGGESTED ANSWER
21 05150 006052         CTOCT
22 05151 006056         CDOCT
23 05152 006104         CGTOK           ;READ ANSWER
24 05153 000402         JMP     .+2     ;SUGGESTED ACCEPTED
25 05154 000770         JMP     DPMMF   ;ERROR RETURN
26 05155 020075         LDA     0,DIGIN ;ANSWER INPUT'ED
27 05156 040707         STA     0,FMADR ;INPUT ACCEPTED
28 05157 006071 DPMMT:  CQUES
29 05160 005106         MXMMT           ;TO INCL.
30 05161 005106         MXMMT
31 05162 005066         XTOIN           ;SUGGESTED ANSWER
32 05163 006052         CTOCT
33 05164 006056         CDOCT
34 05165 006104         CGTOK           ;READ ANSWER
35 05166 000402         JMP     .+2     ;SUGGESTED ACCEPTED
36 05167 000770         JMP     DPMMT   ;ERROR RETURN
37 05170 020075         LDA     0,DIGIN ;ANSWER INPUT'ED
38 05171 040676         STA     0,LMADR ;INPUT ACCEPTED
39 05172 024734         LDA     1,CMADR
40 05173 030672 DPPRT:  LDA     2,FMADR
41 05174 045000         STA     1,0,2
42 05175 034672         LDA     3,LMADR
43 05176 010667         ISZ     FMADR   ;TO NEXT LOC
44 05177 000401         JMP     .+1
45 05200 172414         SUB#    3,2,SZR ;LAST LOC ?
46 05201 000772         JMP     DPPRT   ;NO, NEXT
47 05202 002401         JMP     @.+1    ;YES, PROGRAM FINISHED
48 05203 004740         SWISA           ;RESTART MAIN PROGRAM
```

```
| 0071  .MAIN
01
02 05204 000000 RQUES:  0                ;RETURN ADDR QUES ROUTINE
03 05205 000000 QUESA:  0                ;SUGG. ANSWER
04
05              MXQUE:  .TXT !?!         ;"?"
   05206 000077
06
07              MX2SP:  .TXT !  !        ;"2 SPACE"
   05207 020040
   05210 000000
08
09                      ;ROUTINE TO OUTPUT QUESTIONS.
10                      ;HOW TO USE, SEE EXMEM.
11                      ;CALL   CQUES
12                      ;       MQUES   ;LABEL TEXT TTO/LPT 1,9,17,25 LETTERS
13                      ;       DQUES   ;LABEL TEXT DIS ALLWAYS 9 LETTERS
14                      ;       AQUES   ;LABEL SUGGESTED ANSWER
15                      ;       CTOCT   ;PRINT ROUTINE FOR SUGG. ANSWER
16                      ;       CDOCT   ;DISPLAY ROUTINE FOR SUGG. ANSWER
17                      ;       RETURN  ;TO RELEVANT CALL OF INPUT ROUTINE.
18
19 05211 054773 XQUES:  STA     3,RQUES
20 05212 025400         LDA     1,0,3   ;1. PARAM
21 05213 044417         STA     1,QUESM
22 05214 025401         LDA     1,1,3   ;2. PARAM
23 05215 044413         STA     1,QUESD
24 05216 027402         LDA     1,#2,3  ;3. PARAM
25 05217 044075         STA     1,DIGIN
26 05220 044765         STA     1,QUESA
27 05221 025403         LDA     1,3,3   ;4. PARAM
28 05222 044412         STA     1,QUEST
29 05223 025404         LDA     1,4,3   ;5. PARAM
30 05224 044411         STA     1,QUESS
31 05225 006046         CDICL
32 05226 006043         CCRLF
33 05227 006044         CDISP
34 05230 000000 QUESD:  0                ;DISPLAY MESSAGE
35 05231 006040         CMESS
36 05232 000000 QUESM:  0                ;PRINT MESSAGE
37 05233 024752         LDA     1,QUESA
38 05234 000000 QUEST:  0                ;NUMBER PRINT ROUTINE TTO/LPT
39 05235 000000 QUESS:  0                ;NUMBER PRINT ROUTINE DIS
40 05236 006044         CDISP
41 05237 005206         MXQUE
42 05240 006040         CMESS
43 05241 005206         MXQUE            ;?
44 05242 006040         CMESS
45 05243 005207         MX2SP            ;2 SPACE
46 05244 034740         LDA     3,RQUES
47 05245 001405         JMP     5,3      ;BYPASS PARAM., RETURN
48
49
50              ;TAPE 3
51
52              .EOT
```

```
0072  .MAIN
01
02              ;TAPE 4
03
04              ; GENERAL INPUT-ROUTINES.          VERSION  760119 BY HH
05
06              ; GETDC          GET DECIMAL NUMBER
07              ; GETOK          GET OKTAL NUMBER
08              ; GETBI          GET BINARY NUMBER
09              ; GETSC          GET SINGLE CHARACTER
10              ; GETTX          GET TEXT STRING
11
12
13              ; ABSTRACTS:
14
15              ;    AFTER CALLING DIFFERENT ROUTINES THE OPERATER CAN KEY-IN A
16              ; DEC.-, OCT.- OR A BINARY NUMBER, A CHARACTER OR A TEXT-STRING.
17              ;    THE NUMBER OR THE CHARACTER WILL AFTER A TERMINATION-INPUT
18              ; BE STORED IN "DIGIN". THE TEXT-STRING IS PACKED IN A BUFFER
19              ; CALLED "TEXIN". THE ADDRESS OF TEXIN IS STORED IN "DIGIN"
20              ; FOR INDIRECT USE.
21
22
23
24              ; DESCRIPTION:
25
26              ; "NUMBER"-INPUTROUTINES:
27
28              ; >GETDC< FOR A DECIMALNUMBER DX, WHERE -32768 <= DX <=32767.
29              ; ACCEPTED INPUTS ARE +,-,0,1 ....9.
30
31              ; >GETOK< FOR AN OCTALNUMBER OX, WHERE 0 <= OX <= 177777.
32              ; ACCEPTED INPUTS ARE 0,1, .... 7.
33
34              ; >GETBI< FOR A BINARY NUMBER BX, WHERE 0 <= BX <= 11111111.
35              ; ACCEPTED INPUTS ARE 0,1.
36
37              ; THE SPACE CHAR IS BLIND.
38
39
40              ; A TERMINATION-INPUT WILL STORE THE ENTIRE NUMBER IN THE PAGE-
41              ; ZERO-ADRESS "DIGIN".
42
43
44
45              ; "CHARACTER"-INPUTROUTINE:
46
47              ; >GETSC< FOR A SINGLE CHARACTER. ACCEPTED INPUT IS
48              ; CHARACTERS WITH THE OCTAL ASCII-CODE 11, 33, 40 - 176.
49
50              ; A TERMINATION-INPUT WILL STORE THE CHARACTER-ASCII-CODE
51              ; (WITHOUT PARITY) IN THE RIGTH HALF OF "DIGIN".
52
53
54
55              ; "TEXT"-INPUTROUTINE:
56
57              ; >GETTX< FOR A CHARACTERSTRING OF MAX. 80 CHARACTERS.
58              ; ACCEPTED INPUT IS CHARACTERS WITH THE OCTAL CODE 11, 40 - 176.
59
60              ; AFTER A TERMINATION THE STRING IS PACKED R-L AND THE 3 LAST
61              ; BYTES IN THE STRING WILL ALWAYS CONTENT CR,LF,0. THEN IT IS
62              ; STORED IN "TEXIN", FIRST UP TO 80 BYTES OF INPUT, FOLLOWED
63              ; BY THE 3 TERMINATION BYTES.
```

```
I 0073  .MAIN
01
02
03
04              ; OPERATING PROCEDURE:
05
06              ;    THE ROUTINES ARE CALLED BY THE INSTRUCTION: C>NAME<,
07              ; WHICH EQUALS JSR "ROUTINE", AFTER RECEIVING A LEGAL NUMBER,
08              ; CHARACTER OR TEXT-STRING FOLLOWED BY A TERMINATIOR, THE
09              ; PROGRAM RETURNS TO CALL+3, WHEN A FORMAT- OR OVERFLOWERROR
10              ; OCCURS, THE RETURN IS CALL+2, WHEN ONLY A TERMINATOR IS
11              ; INPUT, THE RETURN IS TO CALL+1.
12
13              ; CALL  CGTDC           ;OR CGTOK, CGTBI, CGTSC, CGTTX
14              ;       TERM RETURN
15              ;       ERROR RETURN
16              ;       NORMAL RETURN
17
18              ; TERMINATOR:
19              ;   IN ALL ROUTINES THE INPUTS: NL, LF, CR OR FF  WILL
20              ; TERMINATE THE MESSAGE.
21
22              ;   EVERY ERRORRETURN WILL INITIALISE THE BUFFERS, COUNTERS
23              ; ETC., AND THE RE-INPUT'ING THEN HAVE TO START AT THE LAST
24              ; TERMINATION POINT, AN * IS PRINTED.
25
26              ; CANCEL:
27              ;   IT'S POSSIBLE TO CANCEL THE LAST-KEYED DIGIT OR CHARACTER
28              ; BY INPUT'ING A RUBOUT, DEL OR CAN CHARACTER ,THE PROGRAM
29              ; ECHOES THEN A "$", AND WAITS FOR A NEW INPUT, IT IS NOT
30              ; POSSIBLE TO CANCEL MORE THAN ONE CHAR, AN ATTEMT WILL
31              ; GIVE ERROR RETURN.
32
33              ; THE NULL CHAR IS BLIND.
34
35              ; IF A CHARACTER IS NOT TERMINATOR, BLIND, CANCEL OR
36              ; ACCEPTED, IT IS ILLEGAL AND CAUSES ERROR RETURN.
```

```
| 0074  .MAIN
01
02                ; SUBROUTINES USED BY ALL INPUTROUTINES
03
04                ; GETCH GETS A CHARACTER FROM TTI OR NUK TO AC0
05
06 05246 054456 GETCH:  STA     3,GETRE
07 05247 004443         JSR     BZNUK    ; CHECK BUSY NUK
08 05250 004431         JSR     INTTI    ; INPUTDEVICE = TTI
09 05251 004446         JSR     INNUK    ; INPUTDEVICE = NUK
10 05252 024461         LDA     1,RAZER ; READ ASCII FROM NUK
11 05253 034454         LDA     3,RALIF ; AC0=KEY
12 05254 054452         STA     3,RADYN
13 05255 101122 RANDI:  MOVZL   0,0,SZC ;
14 05256 000421         JMP     RAEND    ; DIGIT FOUND
15 05257 125400         INC     1,1      ; AC1 = DIGIT
16 05260 014446         DSZ     RADYN    ; 0-9
17 05261 000774         JMP     RANDI
18 05262 126400         SUB     1,1
19 05263 101122         MOVZL   0,0,SZC
20 05264 024445         LDA     1,RAPLU ; AC1 = +
21 05265 101122         MOVZL   0,0,SZC
22 05266 024444         LDA     1,RAMIN ; AC1 = -
23 05267 101122         MOVZL   0,0,SZC
24 05270 024437         LDA     1,RALIF ; AC1 = LF
25 05271 101122         MOVZL   0,0,SZC
26 05272 024436         LDA     1,RASPA ; AC1 = SP
27 05273 101122         MOVZL   0,0,SZC
28 05274 024440         LDA     1,RACAN ; AC1 = CAN
29 05275 127405         AND     1,1,SNR
30 05276 063077         HALT             ; NO KEYS
31 05277 121000 RAEND:  MOV     1,0      ; AC1 TO AC0
32 05300 002424         JMP     @GETRE
33
34 05301 063610 INTTI:  SKPDN   XTTI     ;IS TTI KEY PRESSED ?
35 05302 001400         JMP     0,3      ; NO, CHECK OTHER INPUT DEVICE
36 05303 024422         LDA     1,HC177 ; YES, GET CHAR
37 05304 060410         DIA     0,XTTI
38 05305 123400         AND     1,0
39 05306 060110         NIOS    XTTI
40 05307 101015         MOV#    0,0,SNR ; NULL CHAR ?
41 05310 000771         JMP     INTTI
42 05311 002413         JMP     @GETRE
43
44 05312 126400 BZNUK:  SUB     1,1      ; AC1:=0
45 05313 060434 AANUK:  DIA     0,NUK    ; TEST BUZY NUK
46 05314 122414         SUB#    1,0,SZR ; IS AC0=0  ?
47 05315 000776         JMP     AANUK    ; NO - KEYBOARD NOT READY
48 05316 001400         JMP     0,3      ; YES, RETURN.
49
50 05317 126400 INNUK:  SUB     1,1      ; AC1:= 0
51 05320 060434 BBNUK:  DIA     0,NUK    ; GET KEY BITS
52 05321 122415         SUB#    1,0,SNR ; ARE THEY ZERO ?
53 05322 001776         JMP     -2,3     ; YES, OTHER INPUT, TEST TTI
54 05323 001400         JMP     0,3      ; NO, KEY DEPRESSED
55
56 05324 000000 GETRE:  0
57 05325 000177 HC177:  177
58 05326 000000 RADYN:  0
59 05327 000012 RALIF:  12
60 05330 000040 RASPA:  40
61 05331 000053 RAPLU:  53
62 05332 000055 RAMIN:  55
63 05333 000060 RAZER:  60
64 05334 000030 RACAN:  30
```

```
| 0075  .MAIN
01
02 05335 030551 DELTE:  LDA     2,HC30  ; THE DELTE ROUTINE LOOKS FOR RUBOUT
03 05336 112415         SUB#    0,2,SNR ; IS AC0= 30 ?
04 05337 000405         JMP     DELOV   ; YES - RUBOUT
05 05340 030765         LDA     2,HC177 ; NO
06 05341 112415         SUB#    0,2,SNR ; IS AC0= 177 ?
07 05342 000402         JMP     DELOV   ; YES - RUBOUT
08 05343 001401         JMP     1,3     ; NO
09
10 05344 054462 DELOV:  STA     3,DELRE ; CHECK FOR TOO MANY RUBOUTS
11 05345 030460         LDA     2,LASTN ; OR THE FIRST CHAR IS RUBOUT
12 05346 151102         MOVL    2,2,SZC ; IS LASTN = 177777 ?
13 05347 000417         JMP     ILLGR   ; YES - FIRST IS RUBOUT
14 05350 151102         MOVL    2,2,SZC ; IS LASTN = 077777 ?
15 05351 000415         JMP     ILLGR   ; YES - TOO MANY RUBOUTS
16 05352 152220         ADCZR   2,2     ; NO - PLACE 077777 IN LASTN
17 05353 050452         STA     2,LASTN ; TO INDICATE RUBOUT
18 05354 020534         LDA     0,HC44  ; TYPE $
19 05355 004402         JSR     TYPIN
20 05356 002450         JMP     @DELRE  ; RETURN + 1
21
22 05357 054406 TYPIN:  STA     3,TYPRE ; TYPE THE CHAR IN AC0 ON TTY/DIS
23 05360 004430         JSR     TERMT   ; TERMINATOR ?
24 05361 002404         JMP     @TYPRE  ; YES, DO NOT COPY
25 05362 006045         CDOUT           ; TRY TO TYPE CHAR ON DISPLAY
26 05363 006041         CCHAR           ; TRY TO TYPE CHAR ON TTO
27 05364 002401         JMP     @TYPRE
28 05365 000000 TYPRE:  0
29
30 05366 020522 ILLGR:  LDA     0,HC44  ; ILLEGAL ROUTINE. TYPE $
31 05367 004770 ILLEG:  JSR     TYPIN   ; COPY LAST CHAR
32 05370 020521         LDA     0,HC52
33 05371 004766         JSR     TYPIN   ; TYPE *
34 05372 126400         SUB     1,1     ; AC1:= 0
35 05373 044075         STA     1,DIGIN ; DIGIN:= 0
36 05374 046427         STA     1,@XTXCO ; TEXIN:= CR,LF,0
37 05375 006427         JSR     @XTXND  ; FOR EMPTY TEXTSTRING
38 05376 010424         ISZ     INRET   ; ILLEGAL OR OVERFLOW RETURN
39 05377 000403         JMP     ONTER   ; IS TO CALL + 2
40
41 05400 010422 INTER:  ISZ     INRET   ; TERMINATION RETURN TO CALL+3
42 05401 010421         ISZ     INRET   ; FOR ACCEPTED INPUT
43 05402 006040 ONTER:  CMESS           ; TYPE CR,LF WITH MESS TO WAIT
44 05403 004743         MCRLF           ; FOR DEVICE READY.
45 05404 006047         CDATT           ; RETURN IS TO CALL + 1
46 05405 006044         CDISP           ; FOR NO INPUT, ONLY TERMINATION
47 05406 004743         MCRLF           ; (FOR ANSWER QUE. WITH AN OKAY).
48 05407 002413         JMP     @INRET  ; RETURN TO MAIN PROGRAM
49
50 05410 030474 TERMT:  LDA     2,HC13  ; THE TERMT ROUTINE LOOKS FOR TERM.
51 05411 112415         SUB#    0,2,SNR ; IS AC0=13 ?
52 05412 001401         JMP     1,3     ; YES - IT IS NOT A TERM
53 05413 030472         LDA     2,HC15  ; NO
54 05414 112433         SUBZ#   0,2,SNC ; IS AC0=<15 ?
55 05415 001401         JMP     1,3     ; NO, IT'S NOT A TERMINATOR
56 05416 030465         LDA     2,HC11  ; YES
57 05417 112432         SUBZ#   0,2,SZC ; IS AC0=< 11 ?
58 05420 001401         JMP     1,3     ; YES - IT IS NOT A TERMINATOR
59 05421 001400         JMP     0,3     ; NO - IT IS A TERMINATOR, RETURN
60
61 05422 000000 INRET:  0               ; RETURN ADDRESS TO MAIN PROGRAM
62 05423 006331 XTXCO:  TXCOU           ; ADDRESS OF TEXTBUFFER COUNTER
63 05424 006402 XTXND:  TXEND           ; ADDRESS OF TEXT TERM ROUTINE
64 05425 000000 LASTN:  0               ; X77777 FOR FIRST/RUBOUT, ELSE = CHAR
65 05426 000000 DELRE:  0               ; RETURN ADDRESS FOR DELTE
```

```
| 0076  .MAIN
01
02                ; DECIMAL-NUMBER-INPUTROUTINE.
03
04 05427 054773 GETDC:  STA     3,INRET ; INITIALIZE
05 05430 126000         ADC     1,1     ; AC1:=177777
06 05431 046540         STA     1,@LAST4 ; SET LAST4
07 05432 044534         STA     1,DSIGN ; SET DSIGN TO +
08 05433 126400         SUB     1,1     ; AC1:= 0
09 05434 044534         STA     1,NUMB4 ; CLEAR NUMBER
10 05435 004611 GETD1:  JSR     GETCH   ; GET CHARACTER TO AC0
11 05436 004752         JSR     TERMT   ; TEST FOR TERMINATOR
12 05437 000456         JMP     TERMDC  ; IT IS A TERMINATOR
13 05440 004675         JSR     DELTE   ; IT'S NOT A TERMINATOR- IS IT A DEL ?
14 05441 000774         JMP     GETD1   ; IT IS A DELETE-CHAR
15 05442 004412         JSR     LETE4   ; IT'S NOT A DELETE-CHAR
16 05443 004510         JSR     CHRAN   ; PUT CHAR IN RANGE 0-9
17 05444 004527         JSR     OFTDC   ; TEST FOR OVERFLOW
18 05445 000403         JMP     DELDC   ; FIRST
19 05446 000402         JMP     DELDC   ; LAST IS RUBOUT
20 05447 004462         JSR     PLADC   ; ADD LAST4 TO NUMB4 * 10-DEC.
21 05450 024517 DELDC:  LDA     1,DCDIG ; STORE NEW DIGIT IN LAST4
22 05451 046520         STA     1,@LAST4
23 05452 004705 ECHO4:  JSR     TYPIN   ; TYPE NEW CHARACTER
24 05453 000762         JMP     GETD1   ; GET NEXT CHARACTER
25
26                ; SUBROUTINES USED BY GETDC.
27
28 05454 054426 LETE4:  STA     3,LERE4 ; LEGAL TEST
29 05455 034432         LDA     3,HC40
30 05456 111000         MOV     0,2     ; AC2:= CHAR
31 05457 024513         LDA     1,HC60
32 05460 116415         SUB#    0,3,SNR ; IS AC0 = 40 ?
33 05461 000771         JMP     ECHO4   ; YES - THE CHAR IS A SPACE
34 05462 034427         LDA     3,HC52  ; NO
35 05463 116432         SUBZ#   0,3,SZC ; IS AC0 > 52 ?
36 05464 004703         JSR     ILLEG   ; NO - THE CHAR IS NOT LEGAL
37 05465 034427         LDA     3,HC72  ; YES
38 05466 116033         ADCZ#   0,3,SNC ; IS AC0 < 72 ?
39 05467 004700         JSR     ILLEG   ; NO - NOT LEGAL
40 05470 034422         LDA     3,HC53  ; YES
41 05471 116415         SUB#    0,3,SNR ; IS AC0 >< 53 ?
42 05472 133000         ADD     1,2     ; NO - IT'S A PLUSSIGN, ADD 60
43 05473 034420         LDA     3,HC55  ; YES
44 05474 116415         SUB#    0,3,SNR ; IS AC0 >< 55 ?
45 05475 133000         ADD     1,2     ; NO - IT'S A MINUSSIGN ADD 60
46 05476 034474         LDA     3,HC60  ; YES
47 05477 156032         ADCZ#   2,3,SZC ; IS AC0 >= 60 ?
48 05500 004667         JSR     ILLEG   ; NO NOT LEGAL
49 05501 002401         JMP     @LERE4  ; YES - IT'S A VALID DIGIT
50 05502 000000 LERE4:  0
51
52 05503 000011 HC11:   11
53 05504 000013 HC13:   13
54 05505 000015 HC15:   15
55 05506 000030 HC30:   30
56 05507 000040 HC40:   40
57 05510 000044 HC44:   44
58 05511 000052 HC52:   52
59 05512 000053 HC53:   53
60 05513 000055 HC55:   55
61 05514 000072 HC72:   72
```

```
| 0077  .MAIN
01
02 05515 004456 TERMDC: JSR     OFTDC   ; TERMINATION, TEST OVERFLOW
03 05516 000664         JMP     ONTER   ; TERM IS FIRST
04 05517 000402         JMP     TERDC   ; LAST IS RUBOUT
05 05520 004411         JSR     PLADC   ; ADD LAST DIGIT TO NUMB4
06 05521 024447 TERDC:  LDA     1,NUMB4
07 05522 030444         LDA     2,DSIGN
08 05523 125015         MOV#    1,1,SNR ; IS AC1 = 0 ?
09 05524 000403         JMP     TERM1   ; YES - ZERO ALLWAYS POSITIVE
10 05525 151014         MOV#    2,2,SZR ; IS DSIGN = 0 ?
11 05526 124400         NEG     1,1     ; NO - IT IS A NEGATIVE NUMBER
12 05527 044075 TERM1:  STA     1,DIGIN ; YES - IT'S A POSITIVE NUMBER
13 05530 000650         JMP     INTER   ; OUTPUT TERM CHAR
14
15 05531 054421 PLADC:  STA     3,REPL4 ; PLACE LAST DIGIT/SIGN
16 05532 026437         LDA     1,@LAST4
17 05533 152400         SUB     2,2     ; AC2:= 0
18 05534 034756         LDA     3,HC53  ; LAST = + ?
19 05535 136415         SUB#    1,3,SNR ; NO
20 05536 000412         JMP     PLASI   ; YES, SET DSIGN
21 05537 152520         SUBZL   2,2     ; AC2:= 1
22 05540 034753         LDA     3,HC55  ; LAST = - ?
23 05541 136415         SUB#    1,3,SNR ; NO
24 05542 000406         JMP     PLASI   ; YES, SET DSIGN
25 05543 030425         LDA     2,NUMB4
26 05544 004413         JSR     MULTE   ; MULTIPLY NUMB4 WITH 10-DEC.
27 05545 133000         ADD     1,2     ; ADD LAST4 TO NUMB4*10
28 05546 050422         STA     2,NUMB4 ; PLACE NEW NUMBER
29 05547 002403         JMP     @REPL4
30 05550 050416 PLASI:  STA     2,DSIGN ; PLACE NEW SIGN
31 05551 002401         JMP     @REPL4  ; RETURN
32 05552 000000 REPL4:  0
33
34 05553 024417 CHRAN:  LDA     1,HC60  ; PUT CHAR IN RANGE 0 - 9 OR
35 05554 132400         SUB     1,2     ; PUT CHAR "+" TO 53 OR
36 05555 050412         STA     2,DCDIG ; PUT CHAR "-" TO 55
37 05556 001400         JMP     0,3     ; + AND - WAS ADDED 60 IN LEGAL TEST
38
39 05557 044406 MULTE:  STA     1,MULSA ; NUMBER TO BE MULT. BY 10 IN AC2
40 05560 145120         MOVZL   2,1     ; AC2=NUM, AC1=2*NUM
41 05561 125120         MOVZL   1,1     ; AC2=NUM, AC1=4*NUM
42 05562 133120         ADDZL   1,2     ; AC2=10*NUM, AC1=4*NUM
43 05563 024402         LDA     1,MULSA
44 05564 001400         JMP     0,3     ; RESULT IN AC2
45
46 05565 000000 MULSA:  0
47 05566 000000 DSIGN:  0               ; SIGNFLAG, 0=+, 1=-, 177777=NO SIGN=+
48 05567 000000 DCDIG:  0
49 05570 000000 NUMB4:  0
50 05571 005425 LAST4:  LASTN
51 05572 000060 HC60:   60
```

```
I 0078  .MAIN
01
02 05573 054450 OFTDC:  STA     3,RETOF ; OVERFLOW TEST FOR SIGN AND
03 05574 026775         LDA     1,@LAST4 ; (PREV*10)+LAST < LIMIT DX
04 05575 125102         MOVL    1,1,SZC ; IS THE LAST RUBBED OUT ?
05 05576 002445         JMP     @RETOF  ; OR IS IT THE FIRST ? YES, FIRST
06 05577 010444         ISZ     RETOF
07 05600 125102         MOVL    1,1,SZC
08 05601 002442         JMP     @RETOF  ; YES, RUBOUT
09 05602 034710         LDA     3,HC53  ; NO, IS LAST A + ?
10 05603 136415         SUB#    1,3,SNR ; NO
11 05604 000432         JMP     OFTSI   ; YES
12 05605 034706         LDA     3,HC55  ; IS LAST A - ?
13 05606 136415         SUB#    1,3,SNR ; NO
14 05607 000427         JMP     OFTSI   ; YES
15 05610 176400         SUB     3,3     ; AC3:= 0
16 05611 030755         LDA     2,DSIGN ; IS DSIGN = 177777
17 05612 151112         MOVL#   2,2,SZC ; NO
18 05613 054753         STA     3,DSIGN ; YES, SET FIRST +
19 05614 030754         LDA     2,NUM84 ; AC2:=PREVIOUS (PREV)
20 05615 155120         MOVZL   2,3     ; PREV*2
21 05616 175120         MOVZL   3,3     ; PREV*4
22 05617 175112         MOVL#   3,3,SZC ; PREV*8>=65536 IF
23 05620 006556         JSR     @XILLG  ; PREV >= 8192
24 05621 173122         ADDZL   3,2,SZC ; PREV*10>=65540 IF
25 05622 006554         JSR     @XILLG  ; PREV >= 6554
26 05623 151112         MOVL#   2,2,SZC ; PREV*10>=32770 IF
27 05624 006552         JSR     @XILLG  ; PREV >= 3277
28 05625 133000         ADD     1,2     ; (PREV*10)+LAST>32767
29 05626 151113         MOVL#   2,2,SNC ; (<=32760)+0...9>32767 ?
30 05627 000412         JMP     OFTRE   ; NO, RETURN
31 05630 034736         LDA     3,DSIGN ; YES, TEST +- 32768,32769
32 05631 175005         MOV     3,3,SNR ; TEST SIGN
33 05632 006544         JSR     @XILLG  ; +32768, +32769
34 05633 151134         MOVZL#  2,2,SZR ; -32768, OK RETURN
35 05634 006542         JSR     @XILLG  ; -32769
36 05635 000404         JMP     OFTRE   ; RETURN
37
38 05636 030730 OFTSI:  LDA     2,DSIGN ; OVERFLOW TEST SIGN
39 05637 151113         MOVL#   2,2,SNC ; IS IT FIRST SIGN ? YES
40 05640 006536         JSR     @XILLG  ; NO
41 05641 010402 OFTRE:  ISZ     RETOF   ; PASS RUBOUT/FIRST RETURN
42 05642 002401         JMP     @RETOF  ; RETURN
43
44 05643 000000 RETOF:  0               ; RETURN ADDRESS
```

```
| 0079  .MAIN
01
02                ; CHARACTER-INPUT-ROUTINE
03
04 05644 056537 GETSC:  STA     3,@XINRT
05 05645 126000         ADC     1,1      ; AC1:=177777
06 05646 046440         STA     1,@LAST1 ; SET LAST1
07 05647 126400         SUB     1,1      ; AC1:= 0
08 05650 044435         STA     1,NUMSC  ; CLEAR NUMSC
09 05651 006530 GETS1:  JSR     @XGTCH   ; GET CHAR TO AC0
10 05652 006526         JSR     @XTRMT   ; IS IT A TERMINATOR ?
11 05653 000450         JMP     TERMSC   ; YES
12 05654 006523         JSR     @XDLTE   ; NO - IS IT A DEL CHAR ?
13 05655 000774         JMP     GETS1    ; YES
14 05656 004410         JSR     LETE1    ; NO - IS IT LEGAL ?
15 05657 004430         JSR     OFTSC    ; IT'S A LEGAL CHAR, TEST OVERFLOW
16 05660 000403         JMP     DELSC    ; FIRST
17 05661 000402         JMP     DELSC    ; LAST IS RUBOUT
18 05662 004436         JSR     PLASC    ; PLACE LAST IN NUMSC
19 05663 042423 DELSC:  STA     0,@LAST1 ; STORE NEW CHAR IN LAST1
20 05664 006516         JSR     @XTYPN   ; TYPE NEW CHARACTER
21 05665 000764         JMP     GETS1    ; GET NEXT INPUT
22
23                ; SUBROUTINES USED BY GETSC
24
25 05666 054413 LETE1:  STA     3,LERE1 ; LEGAL TEST
26 05667 030413         LDA     2,H1C40
27 05670 034414         LDA     3,H1C11
28 05671 024412         LDA     1,H1C33
29 05672 106415         SUB#    0,1,SNR ; IS AC0 = 33 ?
30 05673 002406         JMP     @LERE1  ; YES
31 05674 116415         SUB#    0,3,SNR ; NO - IS AC0 = 11 ?
32 05675 002404         JMP     @LERE1  ; YES
33 05676 112032         ADCZ#   0,2,SZC ; NO - IS AC0 >= 40 ?
34 05677 006477         JSR     @XILLG  ; NO
35 05700 002401         JMP     @LERE1  ; YES - IT IS A LEGAL CHAR
36 05701 000000 LERE1:  0
37
38 05702 000040 H1C40:  40
39 05703 000033 H1C33:  33
40 05704 000011 H1C11:  11
41 05705 000000 NUMSC:  0
42 05706 005425 LAST1:  LASTN
43
44 05707 026777 OFTSC:  LDA     1,@LAST1 ; OVERFLOW TEST
45 05710 125102         MOVL    1,1,SZC ; IS THE LAST CHAR RUBBED OUT ?
46 05711 001400         JMP     0,3     ; OR IS IT THE FIRST ? YES, FIRST
47 05712 125102         MOVL    1,1,SZC
48 05713 001401         JMP     1,3     ; YES, RUBOUT
49 05714 030771         LDA     2,NUMSC ; NUMSC = 0 FOR FIRST
50 05715 151004         MOV     2,2,SZR ; IS IT THE FIRST CHAR ?
51 05716 006460         JSR     @XILLG  ; NO, OVERFLOW
52 05717 001402         JMP     2,3     ; YES, RETURN
53
54 05720 032766 PLASC:  LDA     2,@LAST1 ; PLACE LAST CHAR
55 05721 050764         STA     2,NUMSC
56 05722 001400         JMP     0,3
57
58 05723 004764 TERMSC: JSR     OFTSC   ; TERMINATION, TEST OVERFLOW
59 05724 002461         JMP     @XCNTR  ; TERM IS FIRST
60 05725 000402         JMP     TERSC   ; LAST IS RUBOUT
61 05726 004772         JSR     PLASC   ; STORE LAST CHAR IN NUMSC
62 05727 024756 TERSC:  LDA     1,NUMSC ; MOVE CHAR TO DIGIN
63 05730 044075         STA     1,DIGIN
64 05731 002453         JMP     @XINTR  ; OUTPUT TERM CHAR
```

```
| 0080  .MAIN
01
02                ; OCTAL-NUMBER-INPUTROUTINE
03
04 05732 056451 GETOK:  STA     3,@XINRT
05 05733 126000         ADC     1,1      ; AC1:=177777
06 05734 046506         STA     1,@LAST2
07 05735 126400         SUB     1,1      ; AC1:= 0
08 05736 044505         STA     1,NUMB2
09 05737 006442 GETO1:  JSR     @XGTCH   ; GET CHAR TO AC0
10 05740 006440         JSR     @XTRMT   ; IS IT A TERMINATOR ?
11 05741 000503         JMP     TERMOK   ; YES
12 05742 006435         JSR     @XDLTE   ; NO - IS IT A DEL CHAR ?
13 05743 000774         JMP     GETO1    ; YES
14 05744 004412         JSR     LETE2    ; NO - IS IT LEGAL ?
15 05745 004441         JSR     CHRA2    ; IT IS A LEGAL DIGIT
16 05746 004445         JSR     OFTE2    ; TEST FOR OVERFLOW
17 05747 000403         JMP     DELOK    ; FIRST
18 05750 000402         JMP     DELOK    ; LAST IS RUBOUT
19 05751 004460         JSR     PLAOK    ; ADD LAST2 TO NUMB2 * 8-DEC
20 05752 024423 DELOK:  LDA     1,OKDIG  ; STORE NEW DIGIT IN LAST2
21 05753 046467         STA     1,@LAST2
22 05754 006426 ECHO2:  JSR     @XTYPN   ; TYPE NEW CHARACTER
23 05755 000762         JMP     GETO1    ; GET NEXT CHARACTER
24
25                ; SUBROUTINES USED BY GETOK
26
27 05756 054413 LETE2:  STA     3,LERE2  ; LEGAL TEST
28 05757 034413         LDA     3,H2C40
29 05760 116415         SUB#    0,3,SNR  ; IS AC0 = 40 ?
30 05761 000773         JMP     ECHO2    ; YES - THE CHAR IS A SPACE
31 05762 034411         LDA     3,H2C60  ; NO
32 05763 116032         ADCZ#   0,3,SZC  ; IS AC0 >= 60 ?
33 05764 006412         JSR     @XILLG   ; NO - NOT LEGAL
34 05765 034407         LDA     3,H2C70  ; YES
35 05766 116033         ADCZ#   0,3,SNC  ; IS AC0 < 70 ?
36 05767 006407         JSR     @XILLG   ; NO - NOT LEGAL
37 05770 002401         JMP     @LERE2   ; YES - IT IS A VALID DIGIT
38 05771 000000 LERE2:  0
39
40 05772 000040 H2C40:  40
41 05773 000060 H2C60:  60
42 05774 000070 H2C70:  70
43 05775 000000 OKDIG:  0
44
45 05776 005367 XILLG:  ILLEG
46 05777 005335 XDLTE:  DELTE
47 06000 005410 XTRMT:  TERMT
48 06001 005246 XGTCH:  GETCH
49 06002 005357 XTYPN:  TYPIN
50 06003 005422 XINRT:  INRET
51 06004 005400 XINTR:  INTER
52 06005 005402 XONTR:  ONTER
53
54 06006 024765 CHRA2:  LDA     1,H2C60
55 06007 111000         MOV     0,2
56 06010 132400         SUB     1,2      ; PUT CHAR IN RANGE 0-7
57 06011 050764         STA     2,OKDIG
58 06012 001400         JMP     0,3
```

```
I 0081  .MAIN
01
02 06013 054415 OFTE2:  STA     3,REOF2 ; OVERFLOW TEST
03 06014 030427         LDA     2,NUMB2
04 06015 026425         LDA     1,@LAST2 ; IS THE LAST DIGIT RUBBED OUT ?
05 06016 125102         MOVL    1,1,SZC ; OR IS IT THE FIRST ?
06 06017 002411         JMP     @REOF2  ; YES, FIRST
07 06020 010410         ISZ     REOF2
08 06021 125102         MOVL    1,1,SZC
09 06022 002406         JMP     @REOF2  ; YES, RUBOUT
10 06023 034416         LDA     3,OF2CO ; NO
11 06024 156433         SUBZ#   2,3,SNC ; IS NUMB2 <= 17777 ?
12 06025 006751         JSR     @XILLG  ; NO - OVERFLOW
13 06026 010402         ISZ     REOF2   ; YES
14 06027 002401         JMP     @REOF2  ; RETURN
15 06030 000000 REOF2:  0
16
17 06031 026411 PLAOK:  LDA     1,@LAST2 ; PLACE LAST DIGIT
18 06032 030411         LDA     2,NUMB2
19 06033 151120         MOVZL   2,2     ; MULTIPLY NUMB2 WITH 8-DEC
20 06034 151120         MOVZL   2,2
21 06035 151120         MOVZL   2,2
22 06036 133000         ADD     1,2     ; ADD LAST DIGIT
23 06037 050404         STA     2,NUMB2
24 06040 001400         JMP     0,3
25
26 06041 017777 OF2CO:  17777
27 06042 005425 LAST2:  LASTN
28 06043 000000 NUMB2:  0
29
30 06044 004747 TERMOK: JSR     OFTE2   ; TERMINATION, TEST OVERFLOW
31 06045 002740         JMP     @XONTR  ;TERM IS FIRST
32 06046 000402         JMP     TEROK   ; LAST IS RUBOUT
33 06047 004762         JSR     PLAOK   ; ADD LAST DIGIT TO NUMB2
34 06050 024773 TEROK:  LDA     1,NUMB2
35 06051 044075         STA     1,DIGIN
36 06052 002732         JMP     @XINTR  ; OUTPUT TERM CHAR
```

```
| 0082  .MAIN
01
02                ; BINARY-NUMBER-INPUTROUTINE
03
04 06053 056730 GETBI:  STA     3,@XINRT
05 06054 126000         ADC     1,1      ; AC1:=177777
06 06055 046474         STA     1,@LAST3
07 06056 126400         SUB     1,1      ; AC1:= 0
08 06057 044473         STA     1,NUMB3
09 06060 006721 GETB1:  JSR     @XGTCH   ; GET CHAR TO AC0
10 06061 006717         JSR     @XTRMT   ; IS IT A TERMINATOR ?
11 06062 000471         JMP     TERMBI   ; YES
12 06063 006714         JSR     @XDLTE   ; NO - IS IT A DEL CHAR ?
13 06064 000774         JMP     GETB1    ; YES
14 06065 004412         JSR     LETE3    ; NO - IS IT LEGAL ?
15 06066 004431         JSR     CHRA3    ; IT IS A LEGAL DIGIT
16 06067 004435         JSR     OFTE3    ; TEST FOR OVERFLOW
17 06070 000403         JMP     DELBI    ; FIRST
18 06071 000402         JMP     DELBI    ; LAST IS RUBOUT
19 06072 004450         JSR     PLABI    ; ADD LAST3 TO NUMB3 * 2-DEC
20 06073 024423 DELBI:  LDA     1,BIDIG  ; STORE NEW DIGIT IN LAST3
21 06074 046455         STA     1,@LAST3
22 06075 006705 ECHO3:  JSR     @XTYPN   ; TYPE NEW CHARACTER
23 06076 000762         JMP     GETB1    ; GET NEXT CHARACTER
24
25                ; SUBROUTINES USED BY GETBI
26
27 06077 054413 LETE3:  STA     3,LERE3  ; LEGAL TEST
28 06100 034413         LDA     3,H3C40
29 06101 116415         SUB#    0,3,SNR  ; IS AC0 = 40 ?
30 06102 000773         JMP     ECHO3    ; YES - THE CHAR IS A SPACE
31 06103 034411         LDA     3,H3C60  ; NO
32 06104 116032         ADCZ#   0,3,SZC  ; IS AC0 >= 60 ?
33 06105 006671         JSR     @XILLG   ; NO - NOT LEGAL
34 06106 034407         LDA     3,H3C62  ; YES
35 06107 116033         ADCZ#   0,3,SNC; IS AC0 < 62 ?
36 06110 006666         JSR     @XILLG   ; NO - NOT LEGAL
37 06111 002401         JMP     @LERE3   ; YES
38 06112 000000 LERE3:  0
39
40 06113 000040 H3C40:  40
41 06114 000060 H3C60:  60
42 06115 000062 H3C62:  62
43 06116 000000 BIDIG:  0
44
45
46 06117 024775 CHRA3:  LDA     1,H3C60  ; PUT CHAR IN RANGE 0-1
47 06120 111000         MOV     0,2
48 06121 132400         SUB     1,2
49 06122 050774         STA     2,BIDIG
50 06123 001400         JMP     0,3
```

```
| 0083  .MAIN
01
02 06124 054415 OFTE3:  STA     3,REOF3 ; OVERFLOW TEST
03 06125 030425         LDA     2,NUMB3
04 06126 026423         LDA     1,@LAST3 ; IS THE LAST DIGIT RUBBED OUT ?
05 06127 125102         MOVL    1,1,SZC ; OR IS IT THE FIRST ?
06 06130 002411         JMP     @REOF3  ; YES, FIRST
07 06131 010410         ISZ     REOF3
08 06132 125102         MOVL    1,1,SZC
09 06133 002406         JMP     @REOF3  ; YES, RUBOUT
10 06134 034414         LDA     3,OF3CO ; NO
11 06135 156433         SUBZ#   2,3,SNC ; IS NUMB3 <= 177 ?
12 06136 006640         JSR     @XILLG  ; NO - OVERFLOW
13 06137 010402         ISZ     REOF3   ; YES
14 06140 002401         JMP     @REOF3  ; RETURN
15 06141 000000 REOF3:  0
16
17 06142 026407 PLABI:  LDA     1,@LAST3 ; PLACE LAST DIGIT
18 06143 030407         LDA     2,NUMB3
19 06144 151120         MOVZL   2,2     ; MULTIPLY WITH 2-DEC
20 06145 133000         ADD     1,2     ; ADD LAST DIGIT
21 06146 050404         STA     2,NUMB3
22 06147 001400         JMP     0,3
23
24 06150 000177 OF3CO:  177
25 06151 005425 LAST3:  LASTN
26 06152 000000 NUMB3:  0
27
28 06153 004751 TERMBI: JSR     OFTE3   ; TERMINATION, TEST OVERFLOW
29 06154 002631         JMP     @XONTR  ; TERM IS FIRST
30 06155 000402         JMP     TERBI   ; LAST IS RUBOUT
31 06156 004764         JSR     PLABI   ; ADD LAST DIGIT TO NUMB3
32 06157 024773 TERBI:  LDA     1,NUMB3 ; MOVE THE BINARY NUMBER TO DIGIN
33 06160 044075         STA     1,DIGIN
34 06161 002623         JMP     @XINTR  ; OUTPUT TERM CHAR
```

```
I 0084  .MAIN
01
02                ; TEXT-INPUT-ROUTINE
03
04 06162 056555 GETTX:  STA     3,@YINRT
05 06163 126000         ADC     1,1      ; AC1:=177777
06 06164 046543         STA     1,@LAST5
07 06165 126400         SUB     1,1      ; AC1:= 0
08 06166 044543         STA     1,TXCOU
09 06167 006543 GETT1:  JSR     @YGTCH   ; GET CHAR TO AC0
10 06170 006543         JSR     @YTRMT   ; IS IT A TERMINATOR ?
11 06171 000551         JMP     TERMXT   ; YES
12 06172 006542         JSR     @YDLTE   ; NO - IS IT A DEL CHAR ?
13 06173 000774         JMP     GETT1    ; YES
14 06174 004410         JSR     LETE5    ; NO - IS IT LEGAL ?
15 06175 004423         JSR     OFTTX    ; TEST FOR OVERFLOW
16 06176 000403         JMP     DELTX    ; FIRST
17 06177 000402         JMP     DELTX    ; LAST IS RUBOUT
18 06200 004505         JSR     PLATX    ; STORE LAST5 IN TEXTBUFFER
19 06201 042526 DELTX:  STA     0,@LAST5 ; STORE NEW CHAR IN LAST5
20 06202 006533         JSR     @YTYPN   ; TYPE NEW CHAR
21 06203 000764         JMP     GETT1    ; GET NEXT CHARACTER
22
23                ; SUBROUTINES USED BY GETTX
24
25 06204 054410 LETE5:  STA     3,LERE5 ; LEGAL TEST
26 06205 034410         LDA     3,H5C11
27 06206 030410         LDA     2,H5C40
28 06207 116415         SUB#    0,3,SNR ; IS AC0 = 11 ?
29 06210 002404         JMP     @LERE5  ; YES
30 06211 112032         ADCZ#   0,2,SZC ; NO - IS AC0 >= 40 ?
31 06212 006524         JSR     @YILLG  ; NO - THE CHAR IS NOT LEGAL
32 06213 002401         JMP     @LERE5  ; YES
33 06214 000000 LERE5:  0
34
35 06215 000011 H5C11:  11
36 06216 000040 H5C40:  40
37 06217 000117 H5C79:  117              ; TEXTBUFFER LENGTH-1, OKTAL
38
39 06220 026507 OFTTX:  LDA     1,@LAST5 ; OVERFLOW TEST
40 06221 125102         MOVL    1,1,SZC ; IS THE LAST CHAR RUBBED OUT ?
41 06222 001400         JMP     0,3      ; OR IS IT THE FIRST ? YES, FIRST
42 06223 125102         MOVL    1,1,SZC
43 06224 001401         JMP     1,3      ; YES, RUBOUT
44 06225 024504         LDA     1,TXCOU ; NO
45 06226 030771         LDA     2,H5C79
46 06227 132433         SUBZ#   1,2,SNC ; IS TXCOU <= 79-DEC
47 06230 006506         JSR     @YILLG  ; NO - OVERFLOW
48 06231 001402         JMP     2,3      ; YES, RETURN
49
50 06232 006233 TEXIA:  .+1              ; ADDRESS OF TEXTBUFFER
51       000051 TEXIN:  .BLK 51          ; TEXTBUFFER 80 BYTES + CR,LF
52 06304 000000 TEXEN:  0                ; END OF TEXTBUFFER: NUL CHAR
```

```
I 0085  .MAIN
01
02                                          ; PLACE LAST CHAR
03 06305 054421 PLATX:  STA     3,REPL5 ; LAST5 TO CHAHA OR
04 06306 026421         LDA     1,@LAST5 ; LAST5 + CHAHA TO BUFFER
05 06307 010422         ISZ     TXCOU   ; INCREMENT CHARACTER-COUNTER
06 06310 030421         LDA     2,TXCOU
07 06311 151213         MOVR#   2,2,SNC ; IS TXCOU EVEN ?
08 06312 000403         JMP     STOTX   ; YES
09 06313 044415         STA     1,CHAHA ; NO - MOVE LAST5 TO CHAHA
10 06314 002412         JMP     @REPL5
11 06315 030413 STOTX:  LDA     2,CHAHA ; CHAHA IS CHAR HALF BUFFER
12 06316 125320         MOVZS   1,1
13 06317 147000         ADD     2,1     ; AC1 = "LAST5,CHAHA"
14 06320 030411         LDA     2,TXCOU ; CALCULATE ADRESS OF BUFFEREND+1
15 06321 151220         MOVZR   2,2
16 06322 034710         LDA     3,TEXIA
17 06323 173000         ADD     3,2     ; AC2:= TEXIN+TXCOU/2
18 06324 045377         STA     1,-1,2  ; STORE LAST TWO CHAR INTO BUFFEREND+1
19 06325 002401         JMP     @REPL5
20 06326 000000 REPL5:  0
21
22 06327 005425 LAST5:  LASTN
23 06330 000000 CHAHA:  0               ; THE CHAR BEFORE LAST5
24 06331 000000 TXCOU:  0               ; TEXT COUNTER BYTE ADDRESS
25
26 06332 005246 YGTCH:  GETCH
27 06333 005410 YTRMT:  TERMT
28 06334 005335 YDLTE:  DELTE
29 06335 005357 YTYPN:  TYPIN
30 06336 005367 YILLG:  ILLEG
31 06337 005422 YINRT:  INRET
32 06340 005400 YINTR:  INTER
33 06341 005402 YONTR:  ONTER
34
35 06342 004656 TERMXT: JSR     OFTTX   ; TERMINATION, TEST OVERFLOW
36 06343 002776         JMP     @YONTR  ; TERM IS FIRST
37 06344 000402         JMP     TERTX   ; LAST IS RUBOUT
38 06345 004740         JSR     PLATX   ; PLACE THE CHAR BEFORE TERM CHAR
39 06346 024763 TERTX:  LDA     1,TXCOU ; IN LAST5
40 06347 125213         MOVR#   1,1,SNC ; IS TXCOU EVEN ?
41 06350 000417         JMP     NBEVEN  ; YES
42 06351 020443         LDA     0,H5C15 ; NO
43 06352 101320         MOVZS   0,0
44 06353 030755         LDA     2,CHAHA
45 06354 113000         ADD     0,2     ; AC2 = "CR,CHAR"
46 06355 010754         ISZ     TXCOU   ; INCREMENT CHAR COUNT FOR CR
47 06356 004413         JSR     COUDI
48 06357 004415         JSR     STABU   ; STORE AC2 INTO BUFFER
49 06360 030435         LDA     2,H5C12 ; AC2 = " 0,LF"
50 06361 004410         JSR     COUDI
51 06362 125400         INC     1,1     ; INCREMENT BUFF ADDR FOR 0,LF
52 06363 004411         JSR     STABU   ; STORE AC2 INTO BUFFER
53 06364 030646 OUT5:   LDA     2,TEXIA
54 06365 050075         STA     2,DIGIN ; ADDRESS OF TEXTBUFFER
55 06366 002752         JMP     @YINTR  ; OUTPUT TERM CHAR
56
57 06367 004413 NBEVEN: JSR     TXEND   ; STORE LF,CR,0,0
58 06370 000774         JMP     OUT5    ; TERMINATE
```

```
I 0086  .MAIN
01
02 06371 024740 COUDI:  LDA     1,TXCOU
03 06372 125220         MOVZR   1,1       ; DIVIDE TXCOU WITH 2
04 06373 001400         JMP     0,3
05
06 06374 054405 STABU:  STA     3,RET5
07 06375 034635         LDA     3,TEXIA ; CALCULATE ADRESS OF BUFFEREND
08 06376 137000         ADD     1,3
09 06377 051777         STA     2,-1,3  ; STORE AC2 INTO BUFFER
10 06400 002401         JMP     @RET5
11 06401 000000 RET5:   0
12
13 06402 054411 TXEND:  STA     3,TXNDR
14 06403 030413         LDA     2,HLFCR ; AC2 = "LF,CR"
15 06404 004765         JSR     COUDI
16 06405 125400         INC     1,1
17 06406 004766         JSR     STABU   ; STORE AC2 INTO BUFFER
18 06407 125400         INC     1,1
19 06410 152400         SUB     2,2     ; AC2:= 0
20 06411 004763         JSR     STABU   ; STORE "0,0" INTO BUFFER
21 06412 002401         JMP     @TXNDR
22 06413 000000 TXNDR:  0
23
24 06414 000015 H5C15:  15
25 06415 000012 H5C12:  12
26 06416 005015 HLFCR:  5015
```

```
I 0087  .MAIN
01
02                      ;TESTLOOP ROUTINE
03                      ;CALL   SETPX    ;SETP0,SETP1,SETP2
04                      ;                ;PROGRAM LOOP
05                      ;       EHALT    ;ERROR HALT ROUTINE
06                      ;       LOOP     ;CYCLE LOOP ROUTINE
07                      ;
08                      ;                ;NEXT TEST CYCLE
09
10              ;SETPX: ;IORST AND SET # OF LOOPS IN
11                      ;FIRST CYCLUS    ERROR CYCLUS IF SWITCH 0
12                      ;SETP0  10:0     10:1
13                      ;SETP1  10:1     10:2
14                      ;SETP2  10:2     10:2
15              ;EHALT: ;IF NOT FIRST LOOP WITH ERROR IN A CYCLE:
16                      ;       DO NOTHING, LOOP
17                      ;IF FIRST LOOP WITH ERROR IN A CYCLE:
18                      ;       PRINT AC0, AC1, AC2 (NOT ON DIS) AND
19                      ;       PRINT PC XXXXXX IF NOT SWITCH 10 (INHIBIT PRINT)
20                      ;IF FIRST LOOP WITH ERROR AT ALL:
21                      ;       HALT IN EACH ERROR WITH AC3=PC
22                      ;       OTHER ACS RELEVANT INFO.
23                      ;ELSE DO NOTHING, LOOP
24              ;LOOP:  ;IF CYCLE NOT FINISHED (# OF LOOP NOT FINISHED):
25                      ;       IF NO ERRORS AT ALL UNTIL NOW:
26                      ;               IORST, LOOP
27                      ;       IF ERROR AND SWITCH 0 = 0:
28                      ;               PRINT FAILURE RATE OF LAST CYCLE IF
29                      ;               SWITCH 11 = 1 AND SWITCH 10 = 0
30                      ;               PROCEED TO NEXT TEST CYCLE
31                      ;       ELSE:   IORST, LOOP
32                      ;IF CYCLE FINISHED:
33                      ;       IF NO ERRORS AT ALL:
34                      ;               PROCEED TO NEXT TEST CYCLE
35                      ;       ELSE:   PRINT FAILURE RATE OF LAST CYCLE IF
36                      ;               SWITCH 11 = 1 AND SWITCH 10 = 0
37                      ;       IF SWITCH 0 = 0:
38                      ;               PROCEED TO NEXT TEST CYCLE
39                      ;       ELSE:   IORST, LOOP.
40 06417 054522 ENTP0:  STA     3,LOOPR ;INITIALIZE EACH TEST
41 06420 034504         LDA     3,ITRP1
42 06421 054505         STA     3,ITRAG
43 06422 176520         SUBZL   3,3     ;AC3:= 1
44 06423 000412         JMP     ENTCO
45 06424 054515 ENTP1:  STA     3,LOOPR
46 06425 034500         LDA     3,ITRP2
47 06426 054500         STA     3,ITRAG
48 06427 034475         LDA     3,ITRP1
49 06430 000405         JMP     ENTCO
50 06431 054510 ENTP2:  STA     3,LOOPR
51 06432 034473         LDA     3,ITRP2
52 06433 054473         STA     3,ITRAG
53 06434 034471         LDA     3,ITRP2
54 06435 054472 ENTCO:  STA     3,ITR
55 06436 054472         STA     3,ITRCT
56 06437 176400         SUB     3,3     ;AC3:= 0
57 06440 054471         STA     3,EFLAG ;SET FIRST ERROR FLAG = 0
58 06441 054471         STA     3,ERRCT ;SET ERROR COUNT = 0
59 06442 054471         STA     3,ERBCT ;SET ERROR BUFFER COUNT = 0
60 06443 054471         STA     3,EBFLG ;SET FIRST ERROR FLAG BUFFER = 0
61 06444 034501         LDA     3,SETSW ;SET PRINT INHIBIT ON SW 10
62 06445 056477         STA     3,@ISTAC ;IN FUNCTION
63 06446 006126         LPRST           ;I/O RESET
64 06447 002472         JMP     @LOOPR  ;LOOP ITERATE RETURN
```

```
| 0088  .MAIN
01
02 06450 054465 CYCLE:  STA     3,RETUR ;END OF TEST ITERATION ROUTINE
03 06451 050465         STA     2,CSAV2
04 06452 044465         STA     1,CSAV1 ;SAVE THE ACS'
05 06453 040465         STA     0,CSAV0
06 06454 020457         LDA     0,ERBCT ;ADD BUFFER COUNT
07 06455 024455         LDA     1,ERRCT ;TO ERROR COUNT
08 06456 107000         ADD     0,1
09 06457 044453         STA     1,ERRCT
10 06460 020454         LDA     0,EBFLG ;MOVE FIRST FLAG
11 06461 040450         STA     0,EFLAG ;BUFFER TO FLAG
12 06462 014446         DSZ     ITRCT
13 06463 000416         JMP     CYCTS   ;NOT N TIMES ITERATED, LOOP
14 06464 030445         LDA     2,EFLAG ;CYCLUS FINISHED
15 06465 151005         MOV     2,2,SNR ;ERRORS AT ALL ?
16 06466 000432         JMP     NOEX    ;NO ERRORS
17 06467 004457         JSR     FRATE   ;YES, PRINT FAILURE RATE
18 06470 102400         SUB     0,0
19 06471 040441         STA     0,ERRCT ;RESET ERROR COUNT
20 06472 034434         LDA     3,ITRAG
21 06473 054434         STA     3,ITR
22 06474 054434         STA     3,ITRCT
23 06475 006073         CRESW           ;READS 2 ROUTINE
24 06476 151112         MOVL#   2,2,SZC ;SWITCH 0 ?
25 06477 000412         JMP     CYMOR   ;(1) = LOOP IN ERROR
26 06500 000420         JMP     NOEX    ;(0) = PROCEED TO NEXT TEST
27
28 06501 034430 CYCTS:  LDA     3,EFLAG ;LOOP FINISHED
29 06502 175005         MOV     3,3,SNR ;ERRORS UNTIL NOW ?
30 06503 000406         JMP     CYMOR   ;NO, LOOP
31 06504 006073         CRESW           ;YES, READS 2 ROUTINE
32 06505 151112         MOVL#   2,2,SZC ;SWITCH 0 ?
33 06506 000403         JMP     CYMOR   ;(1) = LOOP IN ERROR
34 06507 004437         JSR     FRATE   ;PRINT FAILURE RATE
35 06510 000410         JMP     NOEX    ;(0)=PROCEED TO NEXT TEST
36
37 06511 006126 CYMOR:  LPRST           ;I/O RESET
38 06512 176400         SUB     3,3     ;AC3:=0
39 06513 054420         STA     3,ERBCT ;RESET ERROR BUFFER COUNTER
40 06514 020424         LDA     0,CSAV0
41 06515 024422         LDA     1,CSAV1
42 06516 030420         LDA     2,CSAV2 ;RESTORE AC'S
43 06517 002422         JMP     @LOOPR  ;LOOP MORE
44
45 06520 020420 NOEX:   LDA     0,CSAV0 ;EXIT TO NEXT TEST
46 06521 024416         LDA     1,CSAV1
47 06522 030414         LDA     2,CSAV2 ;RESTORE AC'S
48 06523 002412         JMP     @RETUR  ;PROCEED TO NEXT TEST
49
50 06524 000012 ITRP1:  12              ;10!1 LOOP CONSTANT
51 06525 000144 ITRP2:  144             ;10!2 LOOP CONSTANT
52 06526 000000 ITRAG:  0               ;# OF LOOP IF ERROR
53 06527 000000 ITR:    0               ;# OF LOOP IN CYCLUS ACTUAL
54 06530 000000 ITRCT:  0               ;LOOP COUNTER
55 06531 000000 EFLAG:  0               ;FIRST ERROR FLAG FOR ALL CYCLES
56 06532 000000 ERRCT:  0               ;ERROR COUNTER
57 06533 000000 ERBCT:  0               ;ERROR BUFFER COUNTER
58 06534 000000 EBFLG:  0               ;FIRST ERROR FLAG BUFFER
59 06535 000000 RETUR:  0
60 06536 000000 CSAV2:  0
61 06537 000000 CSAV1:  0
62 06540 000000 CSAV0:  0
63 06541 000000 LOOPR:  0
64 06542 000010 ERHSW:  10              ;SWITCH 12
```

```
| 0089  .MAIN
01
02 06543 000437 IBZOT:  BZOUT
03 06544 000574 ISTAC:  SETAC
04 06545 006073 SETSW:  CRESW           ;FOR PRINT INHIBIT ROUTINE
05
06 06546 054432 FRATE:  STA     3,FRATR ;PRINT FAILURE RATE
07 06547 006073         CRESW           ;IF SWITCH 10 = 0
08 06550 024433         LDA     1,INHSW ;AND SWITCH 11 = 1
09 06551 133414         AND#    1,2,SZR
10 06552 002426         JMP     @FRATR  ;INHIBIT PRINTOUT
11 06553 024431         LDA     1,FRASW
12 06554 133415         AND#    1,2,SNR
13 06555 000420         JMP     NORAT   ;PRINT CRLF ONLY
14 06556 020423         LDA     0,CH40
15 06557 006041         CCHAR
16 06560 024752         LDA     1,ERRCT
17 06561 030744         LDA     2,ITRP2
18 06562 006066         MULTI           ;AC1*AC2
19 06563 030744         LDA     2,ITR
20 06564 034744         LDA     3,ITRCT
21 06565 172400         SUB     3,2     ;AC2:=# OF LOOPS TILL NOW
22 06566 006070         DIVID           ;AC0,AC1/AC2
23 06567 006053         CTDEC
24 06570 006057         CDDEC           ;PRINT VALUE
25 06571 020411         LDA     0,PCENT ;EXAMPLE:       89%
26 06572 006045         CDOUT           ;VALUE=ERRCT*100/(ITR-ITRCT)
27 06573 006041         CCHAR
28 06574 006047         CDATT
29 06575 006040 NORAT:  CMESS           ;PRINT CRLF WITH MESS TO WAIT
30 06576 004743         MCRLF           ;FOR DEVICE READY BEFORE NEXT IORST.
31 06577 002401         JMP     @FRATR
32 06600 000000 FRATR:  0
33 06601 000040 CH40:   40
34 06602 000245 PCENT:  245
35 06603 000040 INHSW:  40              ;SWITCH 10
36 06604 000020 FRASW:  20              ;SWITCH 11
37
38              MHEAD:  .TXT !AC0<11>AC1<11>AC2<15><12>PC<40>!  ;"AC0<11>AC1<11>A(
   06605 041501
   06606 004460
   06607 041501
   06610 004461
   06611 041501
   06612 006462
   06613 050012
   06614 020103
   06615 000000
39
40              DHEAD:  .TXT !PC<40>!   ;"PC<40>"
   06616 041520
   06617 000040
41
42 06620 054715 ERROR:  STA     3,RETUR ;ERROR SUBROUTINE
43 06621 034711         LDA     3,ERRCT
44 06622 175005         MOV     3,3,SNR ;FIRST ERROR LOOP ?
45 06623 000404         JMP     ERR01
46 06624 010707 ERRET:  ISZ     ERBCT   ;COUNT BUFFER ERROR COUNTER
47 06625 002710         JMP     @RETUR  ;EXIT
48 06626 002707         JMP     @RETUR  ;EXIT IF SKIP
```

```
| 0090  .MAIN
01
02 06627 040711 ERR01:  STA     0,CSAV0 ;FIRST ERROR LOOP !
03 06630 044707         STA     1,CSAV1 ;(IN FIRST CYCLUS
04 06631 050705         STA     2,CSAV2 ;OR ERROR CYCLUS)
05 06632 006073         CRESW           ;READS 2 ROUTINE
06 06633 024750         LDA     1,INHSW ;SWITCH 10
07 06634 133414         AND#    1,2,SZR
08 06635 000455         JMP     ERR03   ;INHIBIT PRINTOUT
09 06636 006043         CCRLF
10 06637 024701         LDA     1,CSAV0
11 06640 006052         CTOCT
12 06641 024676         LDA     1,CSAV1
13 06642 006052         CTOCT
14 06643 024673         LDA     1,CSAV2
15 06644 006052         CTOCT           ;PRINT AC'S ONLY AT TTY, LPT
16 06645 006043         CCRLF           ;PRINT CARRIAGE
17 06646 006040         CMESS           ;PRINT HEADER
18 06647 006605         MHEAD
19 06650 006046         CDICL
20 06651 006044         CDISP
21 06652 006616         DHEAD
22 06653 020662         LDA     0,RETUR
23 06654 126000         ADC     1,1
24 06655 107000         ADD     0,1
25 06656 006052         CTOCT
26 06657 006056         CDOCT           ;PRINT PC OF ERROR
27 06660 006663         JSR     @IBZOT  ;WAIT FOR LPT/TTY BEFORE NEXT IORST
28 06661 020650         LDA     0,EFLAG
29 06662 101005         MOV     0,0,SNR
30 06663 000406         JMP     ERR02   ;FIRST ERROR LOOP AT ALL
31 06664 006047         CDATT
32 06665 020653         LDA     0,CSAV0
33 06666 024651         LDA     1,CSAV1
34 06667 030647         LDA     2,CSAV2 ;RESTORE ACS
35 06670 000734         JMP     ERRET
36
37 06671 006050 ERR02:  CHAAT           ;FIRST ERROR LOOP AT ALL: HALT
38 06672 126000         ADC     1,1     ;AC1:=177777
39 06673 044641         STA     1,EBFLG ;SET FIRST ERROR BUFFER FLAG
40 06674 006073         CRESW           ;READS 2 ROUTINE
41 06675 034645         LDA     3,ERHSW
42 06676 157404         AND     2,3,SZR ;SW 12 ?
43 06677 000407         JMP     ERRNH   ;NO HALT
44 06700 034635         LDA     3,RETUR
45 06701 137000         ADD     1,3     ;ERROR. AC3=PC OF ERROR
46 06702 030634         LDA     2,CSAV2
47 06703 024634         LDA     1,CSAV1
48 06704 020634         LDA     0,CSAV0
49 06705 063077         HALT            ;OPERATOR-SET SWITCHES!
50 06706 020632 ERRNH:  LDA     0,CSAV0
51 06707 024630         LDA     1,CSAV1
52 06710 030626         LDA     2,CSAV2
53 06711 000713         JMP     ERRET
54
55 06712 020617 ERR03:  LDA     0,EFLAG ;NO PRINTING
56 06713 101005         MOV     0,0,SNR
57 06714 000755         JMP     ERR02   ;FIRST ERROR LOOP AT ALL
58 06715 020623         LDA     0,CSAV0
59 06716 024621         LDA     1,CSAV1
60 06717 030617         LDA     2,CSAV2
61 06720 000704         JMP     ERRET
```

```
I 0091  .MAIN
01
02                      ;GET ARGUMENT STATUS
03                      ;AND CHECK THAT ALL OF
04                      ;THE BITS ARE PRESENT IN THE ACTUAL STATUS
05                      ;RETURN+2 IF BITS ARE PRESENT, OTHERWISE +1
06                      ;EXIT WITH AC1=EXPECTED, AC0=ACTUAL STATUS
07                      ;
08                      ;CALL   STATA
09                      ;       ARG
10
11 06721 054412 XSTAA:  STA     3,RXAST
12 06722 010411         ISZ     RXAST
13 06723 025400         LDA     1,0,3   ;AC1=EXPECTED BITS
14 06724 060417 SDEV1:  DIA     0,DEV
15 06725 040407         STA     0,SXAST ;SAVE AC0
16 06726 123400         AND     1,0
17 06727 122415         SUB#    1,0,SNR
18 06730 010403         ISZ     RXAST   ;OK
19 06731 020403         LDA     0,SXAST ;RESTORE STATUS
20 06732 002401         JMP     @RXAST
21 06733 000000 RXAST:  0
22 06734 000000 SXAST:  0
23
24                      ;GET ARGUMENT STATUS
25                      ;AND CHECK THAT NONE OF
26                      ;THE BITS ARE PRESENT IN ACTUAL STATUS
27                      ;RETURN +1 IF BITS PRESENT, OTHERWISE +2
28                      ;EXIT WITH AC1=NON-EXPECTED, AC0=ACTUAL STATUS
29                      ;
30                      ;CALL   STATN
31                      ;       ARG
32
33
34 06735 054407 XSTAN:  STA     3,RXNST
35 06736 010406         ISZ     RXNST
36 06737 025400         LDA     1,0,3   ;AC1=NONEXPECTED STATUS
37 06740 060417 SDEV2:  DIA     0,DEV   ;AC0=ACTUAL STATUS
38 06741 107415         AND#    0,1,SNR
39 06742 010402         ISZ     RXNST   ;OK
40 06743 002401         JMP     @RXNST
41 06744 000000 RXNST:  0
42
43                      ;COMBINE ARGUMENT STATUS WITH (SWITCH STATUS
44                      ;MASK) AND CHECK WHOLE AGAINST ACTUAL.
45                      ;RETURN+2 IF STATUS MATCH, +1 OTHERWISE
46                      ;EXIT WITH AC1=EXPECTED STATUS, AC0=ACTUAL
47                      ;
48                      ;CALL   STATW
49                      ;       ARG
50                      ;
51 06745 054413 XSTAW:  STA     3,RXWST
52 06746 010412         ISZ     RXWST
53 06747 025400         LDA     1,0,3
54 06750 006073         CRESW           ;READS 2 ROUTINE
55 06751 034410         LDA     3,SMASK ;MASK FOR SWITCH REGISTER
56 06752 157400         AND     2,3
57 06753 167000         ADD     3,1     ;AC1=EXP STATUS
58 06754 060417 SDEV3:  DIA     0,DEV   ;AC0=ACTUAL STATUS
59 06755 106415         SUB#    0,1,SNR
60 06756 010402         ISZ     RXWST   ;OK
61 06757 002401         JMP     @RXWST
62 06760 000000 RXWST:  0
63 06761 000000 SMASK:  000000          ;CHANGE MASK TO ONES FOR
64                                      ;THOSE BITS TO USE FROM SW.
```

```
I 0092  .MAIN
01
02                          ;GET ARGUMENT STATUS AND ARGUMENT MASK
03                          ;AND CHECK FOR EQUALITY AGAINST ACTUAL,
04                          ;EXCEPT THOSE BIT, WHERE MASK BIT ARE NULL.
05                          ;RETURN +3 IF STATUS MATCH, OTHERWISE +2.
06                          ;EXIT WITH AC0 = ACTUAL, AC1 = EXPECTED, AC2 = MASK.
07                          ;
08                          ;CALL   STATP
09                          ;       ARG
10                          ;       MASK
11
12 06762 054414 XSTAP:  STA     3,RXPST
13 06763 010413         ISZ     RXPST
14 06764 010412         ISZ     RXPST
15 06765 025400         LDA     1,0,3   ;AC1:=EXPECTED PART
16 06766 031401         LDA     2,1,3   ;AC2:=MASK
17 06767 147400         AND     2,1
18 06770 060417 SDEV4:  DIA     0,DEV   ;AC0:=ACTUAL STATUS
19 06771 115000         MOV     0,3     ;AC3:=ACTUAL STATUS FOR MASK
20 06772 157400         AND     2,3
21 06773 136415         SUB#    1,3,SNR
22 06774 010402         ISZ     RXPST   ;OK
23 06775 002401         JMP     @RXPST
24 06776 000000 RXPST:  0
25
26                          ;GET ARGUMENT STATUS
27                          ;AND CHECK THAT SOME OF
28                          ;THE BITS ARE PRESENT IN THE ACTUAL STATUS
29                          ;RETURN+2 IF BITS ARE PRESENT, OTHERWISE +1
30                          ;EXIT WITH AC1=EXPECTED, AC0=ACTUAL STATUS
31                          ;
32                          ;CALL   STATS
33                          ;       ARG
34
35 06777 054407 XSTAS:  STA     3,RXSST
36 07000 010406         ISZ     RXSST
37 07001 025400         LDA     1,0,3   ;AC1:=EXPECTED STATUS
38 07002 060417 SDEV5:  DIA     0,DEV   ;AC0:=ACTUAL STATUS
39 07003 107414         AND#    0,1,SZR
40 07004 010402         ISZ     RXSST   ;OK
41 07005 002401         JMP     @RXSST
42 07006 000000 RXSST:  0
43
44                          ;ROUTINE LOOP REPORT
45
46 07007 054454 XLORE:  STA     3,RPASS
47 07010 006046         CDICL
48 07011 006043         CCRLF
49 07012 024451         LDA     1,RPASS
50 07013 152520         SUBZL   2,2     ;AC2:=1
51 07014 146400         SUB     2,1     ;SUBTRACT 1 FROM JSR ADDR
52 07015 006056         CDOCT
53 07016 006052         CTOCT           ;PRINT ADDR
54 07017 006044         CDISP
55 07020 007070         MLORE
56 07021 006040         CMESS
57 07022 007070         MLORE           ;XXXXXX LOOP-ADDR
58 07023 006047         CDATT
59 07024 006437         JSR     @RPASS  ;RETURN TO START LOOP
```

```
| 0093  .MAIN
01
02                        ;ROUTINE TO HANDLE PASS #
03                        ;CORRECT PASSC TO YOUR CHOICE OF # OF RUNS BETWEEN
04                        ;EACH PASS MESSAGE (2 MINUTES INTERVAL IS CONVIENIENT).
05                        ;CORRECT TEXT MPASS, DPASS ACCORDINGLY.
06                        ;INITIALIZE WHEN PROGRAM STARTED/RESTARTED:
07                        ;       PASSN TO 0
08                        ;       PASSB TO # OF RUNS BETWEEN MESS = PASSC
09                        ;CALL:  CPASS
10                        ;       RETURN
11
12 07025 054436 XPASS:    STA     3,RPASS
13 07026 014432           DSZ     PASSB   ;RUN COUNT DOWN
14 07027 002434           JMP     @RPASS  ;NO MESSAGE
15 07030 020431           LDA     0,PASSC
16 07031 040427           STA     0,PASSB ;INITIALIZE RUN COUNT
17 07032 006046           CDICL             ;WRITE PASS MESSAGE
18 07033 006043           CCRLF
19 07034 006040           CMESS
20 07035 005207           MX2SP
21 07036 010424           ISZ     PASSN   ;COUNT PASS #
22 07037 024423           LDA     1,PASSN
23 07040 006053           CTDEC
24 07041 006057           CDDEC
25 07042 006044           CDISP
26 07043 007054           DPASS
27 07044 006040           CMESS
28 07045 007050           MPASS
29 07046 006047           CDATT
30 07047 002414           JMP     @RPASS  ;RETURN
31
32                MPASS:  .TXT    !. PASS!
   07050 020056
   07051 040520
   07052 051523
   07053 000000
33
34                DPASS:  .TXT    !. PASS!
   07054 020056
   07055 040520
   07056 051523
   07057 000000
35
36
37                ;PASS-ADMINISTRATOR CONSTANTS AND COUNTERS (HAS TO BE SET BE-
38                ;FORE START OF TESTEXECUTION).
39
40 07060 000000 PASSB:  0                 ;COUNTER FOR # OF RUNS BETWEEN PASS MESS
41 07061 000012 PASSC:  10.               ;# OF RUNS BETWEEN PASS MESS
42 07062 000000 PASSN:  0                 ;# OF PASS COUNTER
43
44 07063 000000 RPASS:  0                 ;RETURN ADDR
45 07064 007202 TROTA:  TROTB             ;POINTER TO TABLE FOR BREAKPOINT
46 07065 063077 TROHC:  HALT              ;TROUBLE HALT COMMAND
47 07066 006122 TROLC:  CLORE             ;TROUBLE LOOP REPORT COMMAND
48 07067 000401 TRORC:  JMP     .+1       ;TROUBLE RESET COMMAND
49
50                MLORE:  .TXT ! LOOP-ADDR!       ;" LOOP-ADDR"
   07070 046040
   07071 047517
   07072 026520
   07073 042101
   07074 051104
   07075 000000
```

```
| 0094  .MAIN
01
02                ;ROUTINES TO HELP LOCATING THE LOOP WHICH MAY DESTROY THE
03                ;RUNNING PROGRAM.
04                ;ALL LOOPS SHOULD START THIS WAY:
05                ;A#:    JMP     .+1       ;FOR PROGRAM TROUBLESHOOTING.
06                ;       MOV     0,1       ;INITIALIZE LOOP CODE
07                ;       SETP1             ;SETUP 1,10,100 LOOPS
08                ;THIS GIVES 3 POSSIBILITIES FOR CHANGING THE JMP .+1 TO MORE
09                ;SUITABLE CONTENT IN CASE OF TROUBLES:
10                ;A:  MANUAL INSERT JMP @RTEST FROM LAST LOOP AXX. THE PROGRAM
11                ;     THEN RUNS THE FIRST LOOPS UNTILL THIS COMMAND IS SEEN,
12                ;     FOLLOWED BY PASS MESSAGE AFTER N RUNS. THE BREAKPOINT
13                ;     IS UP TO YOU.
14                ;B:  START PROGRAM IN SA 2214 WHICH CHANGES THE JMP .+1 TO HALT
15                ;     BUT ONLY IN THE BREAKPOINTS CHOOSEN BY THE PROGRAMMER
16                ;     IN THE FOLLOWING TABLE. THE PROGRAM THEN RUNS FROM
17                ;     BREAKPOINT TO BREAKPOINT IF CONTINUE IS DEPRESSED.
18                ;C:  START PROGRAM IN SA 2216 WHICH CHANGES THE JMP .+1 TO CLORE,
19                ;     A ROUTINE, CALL LOOP REPORT, WHICH PRINTS THE PC EACH TIME
20                ;     THE BREAKPOINTS CHOOSEN BY THE PROGRAMMER IN
21                ;     FOLLOWING TABLE IS PASSED.
22                ;D:  STARTING PROGRAM IN SA 2220 WILL RESET ALL BREAKPOINTS
23                ;     MENTIONED IN THE TABLE TO JMP .+1.
24
25                          ;TROUBLE HALT
26 07076 165000 TROHA:  MOV     3,1
27 07077 006072         CSAMS             ;START ADDR MESSAGE
28 07100 036764         LDA     3,@TROTA
29 07101 054023         STA     3,IDX3    ;INIT TABLEPOINTER
30 07102 024763         LDA     1,TROHC   ;HALT COMMAND
31 07103 032023 TROHL:  LDA     2,@IDX3
32 07104 151015         MOV#    2,2,SNR   ;END OF TABLE ?
33 07105 000427         JMP     TROEN     ;YES, RETURN
34 07106 045000         STA     1,0,2     ;STORE HALT IN LOOP START
35 07107 000774         JMP     TROHL     ;AGAIN
36
37                          ;TROUBLE LOOP REPORT
38 07110 165000 TROLO:  MOV     3,1
39 07111 006072         CSAMS             ;START ADDR MESSAGE
40 07112 036752         LDA     3,@TROTA
41 07113 054023         STA     3,IDX3    ;INIT TABLEPOINTER
42 07114 024752         LDA     1,TROLC   ;LOOP REPORT COMMAND
43 07115 032023 TROLL:  LDA     2,@IDX3
44 07116 151015         MOV#    2,2,SNR   ;END OF TABLE ?
45 07117 000415         JMP     TROEN     ;YES, RETURN
46 07120 045000         STA     1,0,2     ;STORE CLORE IN LOOP START
47 07121 000774         JMP     TROLL     ;AGAIN
48
49                          ;TROUBLE RESET
50 07122 165000 TRORE:  MOV     3,1
51 07123 006072         CSAMS             ;START ADDR MESSAGE
52 07124 036740         LDA     3,@TROTA
53 07125 054023         STA     3,IDX3    ;INIT TABLEPOINTER
54 07126 024741         LDA     1,TRORC   ;JMP .+1 COMMAND
55 07127 032023 TRORL:  LDA     2,@IDX3
56 07130 151015         MOV#    2,2,SNR   ;END OF TABLE ?
57 07131 000403         JMP     TROEN     ;YES, RETURN
58 07132 045000         STA     1,0,2     ;STORE JMP .+1 IN LOOP START
59 07133 000774         JMP     TRORL     ;AGAIN
60 07134 002401 TROEN:  JMP     @.+1
61 07135 004740         SWISA             ;RESTART MAIN PROGRAM
```

```
0095  .MAIN
02
03                ; DIAGNOSTIC- AND EXERCISERPROGRAM
04
05                ; STARTROUTINES
06
07 07136 165000 STAR1:  MOV     3,1       ;MOVE STARTADDRESS FROM AC3 TO AC1
08 07137 006072         CSAMS             ;TYPE START-ADDRESS
09 07140 004444         JSR     INIA      ;INITIALIZE PROGRAM
10 07141 000516         JMP     ASK0      ;GO TO QUESTIONMANEGER
11
12 07142 165000 STAR2:  MOV     3,1
13 07143 006072         CSAMS             ;TYPE START-ADDRESS
14 07144 004443         JSR     INIB      ;INITIALIZE PROGRAM
15 07145 006472         JSR     @TYINI    ;INITIALIZE CONSTANTS
16 07146 000513         JMP     ASK3
17
18 07147 165000 STAR3:  MOV     3,1
19 07150 006072         CSAMS             ;TYPE START-ADDRESS
20 07151 004436         JSR     INIB      ;INITIALIZE PROGRAM
21 07152 006465         JSR     @TYINI    ;INITIALIZE CONSTANTS
22 07153 006477         JSR     @IYDIA    ;RETURN .+2
23 07154 000000         0
24 07155 006463         JSR     @NOOPT    ;MASK-OUT TESTS EXCEPT DIAGNOSTICTESTS
25                                        ;WITHOUT OPERATOR-INVOLVEMENT
26 07156 006463         JSR     @NOPIN    ;MASK-OUT TESTS WITH PAPERCONSUMPTION
27 07157 006463         JSR     @NOPRP    ;RETURN .+2
28 07160 000000         0
29 07161 006462         JSR     @NOROP    ;RETURN .+2
30 07162 000000         0
31 07163 000512         JMP     QUEFI+1
32
33 07164 165000 STAR4:  MOV     3,1
34 07165 006072         CSAMS             ;TYPE START-ADDRESS
35 07166 004421         JSR     INIB      ;INITIALIZE PROGRAM
36 07167 006450         JSR     @TYINI    ;INITIALIZE CONSTANTS
37 07170 006465         JSR     @TBASK    ;SELECT PRINTER-TABLE
38 07171 020455         LDA     0,REMRK
39 07172 042455         STA     0,@NODRE;SET RETURNADR. FROM NODIA-ROUT.
40 07173 006451         JSR     @NOTDI    ;MASK-OUT TESTS EXCEPT CHAR.MOTION-TESTS
41 07174 000500 STA4A:  JMP     QUEFI
42
43 07175 165000 RESTR:  MOV     3,1
44 07176 006072         CSAMS             ;TYPE START-ADDRESS
45 07177 102440         SUBO    0,0
46 07200 040540         STA     0,PASCO ;CLEAR PASS-COUNTER
47 07201 000473         JMP     QUEFI   ;RESTART TEST
48
49
50 07202 007202 TROTB:  .                 ;TABLE WITH LOOP ADDRESSES
51 07203 000000         0                 ;(NOT USED IN THIS PROGRAM)
```

```
| 0096  .MAIN
01
02                ;ROUTINE TO INITIALIZE TESTPROGRAM
03
04 07204 102440 INIA:   SUBO    0,0      ;AC0:=0
05 07205 042440         STA     0,@MOMRI;MASK-IN QUE06 AND QUE07
06 07206 000404         JMP     RETST
07 07207 102000 INIB:   ADC     0,0      ;AC0:=177777
08 07210 042435         STA     0,@MOMRI;MASK-OUT QUE06 AND QUE07
09 07211 042440         STA     0,@IPASM;MAKE PRT.ROUT NOT USE PRT.TABLE
10 07212 054420 RETST:  STA     3,REINI
11 07213 062677         IORST
12 07214 020417         LDA     0,INTAD
13 07215 040001         STA     0,1      ;SET INT.ADR. = JUMP BACK
14 07216 102440         SUBO    0,0      ;AC0:=0
15 07217 040132         STA     0,MRK47;CLEAR 3643 MARK
16 07220 042433         STA     0,@ICLMK;CLEAR CHAR/LINE-DEF-MARK
17 07221 042415         STA     0,@M1CEN;CLEAR CEN.101-MARK
18 07222 042413         STA     0,@IEXIN;CLEAR INIEX
19 07223 040515         STA     0,PASCO ;INITIALIZE PASSCOUNTER
20 07224 042410         STA     0,@MRKLP;CLEAR LINES/PAGE-DEF.-MARK
21 07225 042427         STA     0,@MRKTB;CLEAR "SELECT-PRT-TABLE"-MARK
22 07226 042430         STA     0,@IFCLE
23 07227 020421         LDA     0,AASK5
24 07230 042417         STA     0,@NODRE;SET RETURN FROM "NODIA"-ROUTINE
25 07231 002401         JMP     @REINI   ;RETURN TO CALL OF INIA- OR INIB-ROUTINE
26
27 07232 000000 REINI:  0
28 07233 007321 INTAD:  INTSV
29 07234 013520 MRKLP:  LPMRK
30 07235 010400 IEXIN:  INIEX
31 07236 007774 M1CEN:  CEN1M
32 07237 007453 TYINI:  INITY
33 07240 010225 NOOPT:  NOMAN
34 07241 010254 NOPIN:  INNOP
35 07242 010325 NOPRP:  NOMO1
36 07243 010441 NOROP:  NOROT
37 07244 010165 NOTDI:  NODIA
38 07245 010112 MOMRI:  MOMRK
39 07246 007174 REMRK:  STA4A
40 07247 010203 NODRE:  RENOD
41 07250 007266 AASK5:  ASK5
42 07251 014170 IPASM:  PASMK
43 07252 010162 IYDIA:  YEDIA
44 07253 010402 ICLMK:  CHLMK
45 07254 010600 MRKTB:  TDEFM
46 07255 010514 TBASK:  ASKTB
47 07256 012635 IFCLE:  FCLER
```

```
I 0097  .MAIN
01
02                ; QUESTION-MANEGER
03
04
05 07257 006452 ASK0:   JSR     @QUE0    ;ASK FOR PRINTERTYPE - SET TESTFIGURE
06                                        ;AND TIMING-CONSTANTS
07
08 07260 006452 ASK1:   JSR     @QUE1    ;RETURN TO HERE
09                                        ;THEN ASK FOR CONTROLERTYPE - EXAMINE
10                                        ;PREVIOUS DEV. NO. AND CHANGE I/O-INSTR.
11                                        ;IF NECESSARY
12
13 07261 006452 ASK3:   JSR     @QUE3    ;ASK FOR DIAGNOSTICTEST
14 07262 006762          JSR     @NOTDI   ;DIAGNOSTIC NOT WANTED - MASK OUT TESTS
15                                        ;AND RETURN TO ASK5
16
17 07263 006451 ASK4:   JSR     @QUE4    ;DIAGNOSTIC WANTED
18                                        ;ASK FOR TESTS WITH OPERATOR-INVOLVEMENT
19                                        ;RETURN TO .+1 OR .+3
20 07264 006754          JSR     @NOOPT   ;OPERATORTESTS NOT WANTED - MASK OUT
21                                        ;THE ROUTINES - RETURN TO .+1
22
23 07265 006450 ASK4A:  JSR     @QUE4I   ;ASK FOR TESTS WITH PAPERCONSUMPTION
24                                        ;MASK-OUT IF NOT WANTED.  IF WANTED ASK
25                                        ;FOR CHANNELS IN VFU. RETURN .+1
26
27 07266 006450 ASK5:   JSR     @QUE5    ;ASK FOR CHARACTERPRINTINGPROGRAM
28                                        ; -RETURN TO .+1 OR .+2
29 07267 006753          JSR     @NOPRP   ;NOT WANTED - RETURN TO .+2
30 07270 006433          JSR     @YEPRP   ;WANTED - TEST INIEX, IF ZERO THEN
31                                        ;ASK FOR CHAR/LINE,INPUTMODE AND SET
32                                        ;MOMRK
33 07271 006446 ASK8:   JSR     @QUE8    ;ASK FOR CHAR.-ROTATIONPROGRAM - RETURN
34                                        ;TO .+1 OR .+2
35 07272 006751          JSR     @NOROP   ;NOT WANTED - RETURN TO .+2
36 07273 006431          JSR     @YEROP   ;WANTED - TEST INIEX, IF ZERO THEN TEST
37                                        ;MOMRK, ASK FOR CHAR/LINE AND CHAR/DRUM
38                                        ;IF NECESSARY
39
40 07274 000000 QUEFI:  0                 ;A JUMP .+1 OR .+10 - INSTR.
41 07275 006043          CCRLF
42 07276 006046          CDICL
43 07277 006040          CMESS
44 07300 014274          TSTST            ;"TESTEXECU. START"
45 07301 006044          CDISP
46 07302 014274          TSTST
47 07303 006047          CDATT
48 07304 006043          CCRLF
49 07305 006043          CCRLF
50 07306 000433          JMP     TSTMAN   ;START TESTEXECUTION
```

```
I 0098  .MAIN
01
02                ;XLRST IS USED BY TESTLOOP, AND IS EXECUTED BEFORE EACH
03                ;OUT-JUMP FROM THE LOOPROUTINE INSTEAD OF THE ORIG. IORST
04
05 07307 054413 XLRST:  STA     3,EXRET
06 07310 063517         SKPBZ   XLPT
07 07311 000403         JMP     TRYWA
08 07312 062677 CONT3:  IORST
09 07313 002407         JMP     @EXRET
10 07314 006063 TRYWA:  TIMSK           ;WAIT FOR BUSY=0 (MAX. 1 SEC)
11 07315 001750         1000.           ;IS CHANGED TO 1 IN CT.. & BT..-ROUTINES
12 07316 063517         SKPBZ   XLPT
13 07317 000401         JMP     .+1
14 07320 000772         JMP     CONT3
15
16
17                ;INTERRUPTROUTINE
18
19 07321 002000 INTSV:  JMP     @0      ;IN CASE OF INTERRUPT JUMP BACK TO
20                                      ;INTERRUPTED ROUTINE
21
22
23 07322 000000 EXRET:  0
24 07323 010335 YEPRP:  YEMO1
25 07324 010450 YEROP:  YEROT
26 07325 007437 IACTSF: ACTSF
27 07326 007373 IPOINT: POINT
28 07327 000020 CHO16:  20
29 07330 000000 COU16:  0
30 07331 007640 QUE0:   QUE00
31 07332 010023 QUE1:   QUE01
32 07333 010144 QUE3:   QUE03
33 07334 010204 QUE4:   QUE04
34 07335 010233 QUE4I:  QUE4A
35 07336 010306 QUE5:   QUE05
36 07337 010422 QUE8:   QUE08
```

```
| 0099  .MAIN
01
02                ; TESTMANEGER
03
04 07340 000000 PASCO:  0
05
06 07341 020474 TSTMAN: LDA     0,TESTF ;MOVE TESTFIGURE FROM TESTF TO ACTSF
07 07342 040475         STA     0,ACTSF
08 07343 020473         LDA     0,TESTF+1
09 07344 040474         STA     0,ACTSF+1
10 07345 020760         LDA     0,IACTSF;INITIALIZE POINTERS
11 07346 040466         STA     0,TFPIL
12 07347 020760         LDA     0,CH016
13 07350 040760         STA     0,COU16
14 07351 020755         LDA     0,IPOINT
15 07352 101400         INC     0,0
16 07353 040420         STA     0,POINT ;MAKE POINT:=POINT + 1
17 07354 022460 SUBMAN: LDA     @0,TFPIL;EXAMINE THE ACTUEL TESTFIGURE
18 07355 101102         MOVL    0,0,SZC
19 07356 000411         JMP     JATST   ;THE ROUTINE ON WHICH POINT IS POINTING
20                                      ;HAS TO BE USED
21 07357 042455         STA     @0,TFPIL;THE ROUTINE HAS NOT TO BE USED
22 07360 010413 ONTST:  ISZ     POINT
23 07361 014747         DSZ     COU16   ;ARE ALL 16 BIT IN TESTF EXAMINED ?
24 07362 000772         JMP     SUBMAN  ;NO - EXAMINE NEXT BIT
25 07363 010451         ISZ     TFPIL   ;YES, ALL BITS ARE EXAMINED - LET TFPIL
26                                      ;POINT TO THE NEXT PART OF THE ACTUEL
27                                      ;TESTFIGURE
28 07364 020743         LDA     0,CH016
29 07365 040743         STA     0,COU16 ;RESET BITCOUNTER
30 07366 000766         JMP     SUBMAN
31
32 07367 042445 JATST:  STA     @0,TFPIL
33 07370 032403         LDA     @2,POINT
34 07371 005000         JSR     0,2     ;EXECUTE THE ACTUEL TESTROUTINE-GROUP
35 07372 000766         JMP     ONTST   ;RETURN TO HERE
36
37
38 07373 000000 POINT:  0
39 07374 010771         CT00            ;TABLE WITH ADDRESS' OF ALL TESTROUTINE-
40 07375 011074         BT00            ;GROUPS
41 07376 011344         BT15
42 07377 011406         IT00
43 07400 011564         CT00
44 07401 011600         CT02
45 07402 011651         CT06
46 07403 000000         0
47 07404 011700         ST00
48 07405 011723         ST02
49 07406 012033         TT00
50 07407 012067         TT02
51 07410 012103         TT03
52 07411 000000         0
53 07412 000000         0
54 07413 000000         0
55 07414 012416         CT04
56 07415 012127         DT00
57 07416 012226         DT01
58 07417 012320         DT02
59 07420 000000         0
60 07421 012736         CT07
61 07422 012770         CT08
```

```
| 0100  .MAIN
01
02 07423 013024          WT00
03 07424 013156          WT09
04 07425 013245          WT13
05 07426 013313          WT15
06 07427 013461          WT24
07 07430 000000          0
08 07431 013572          LIPRT
09 07432 013757          ROPRT
10 07433 007506          TSTEND
11
12 07434 007437 TFPIL:   ACTSF
13 07435 000000 TESTF:   0                ;TESTFIGURE ACCORDING TO PRINTERTYPE
14 07436 000000          0
15 07437 000000 ACTSF:   0                ;SAME AS TESTF, WILL BE SHIFTED LEFT
16 07440 000000          0                ;DURING THE TEST
17
18 07441 176340 TY24S:   176340           ;TESTFIGURES,WHICH SELECTS THE DIFF.
19 07442 140707          140707           ;TESTROUTINEGROUPS IN THE "POINT-TABLE".
20
21 07443 176320 TY22S:   176320
22 07444 140707          140707
23
24 07445 175210 TYCT1:   175210
25 07446 063067          063067
26
27 07447 175210 TYCT2:   175210
28 07450 053067          053067
29
30 07451 175200 TYLOG:   175200
31 07452 042047          042047
32
33               ;ROUTINE TO INITIALIZE TYPE AND CONSTANTS OF PRINTER
34
35 07453 054425 INITY:   STA      3,INIRE
36 07454 006475          JSR      @ZEASK  ;ASK FOR PRINTERTYPE
37 07455 020430          LDA      0,ACONT ;RETURN TO HERE
38 07456 042424          STA      @0,IRET0;SET RETURNADDRESS FROM FCENEX
39 07457 002422          JMP      @IFCEN  ;SET DEV.CODE:= 17 OR 57
40 07460 020415 CONTF:   LDA      0,VCINI
41 07461 042422          STA      @0,IVECH;SET VCHNB:=11
42 07462 020412          LDA      0,CHDST
43 07463 042414          STA      @0,INCHS;SET CHAR/DRUM:= 64
44 07464 020420          LDA      0,INICO
45 07465 040147          STA      0,CHALI ;SET CHAR/LINE
46 07466 102000          ADC      0,0
47 07467 042467          STA      0,@ICHLM;SET CHAR/LINE-DEF MARK
48 07470 042462          STA      0,@IMODE;SET INMODE=OCTAL
49 07471 042462          STA      0,@PAMRK;PASS BY PRINTERTABLE
50 07472 042404          STA      0,@EXINI;MASK OUT LATER QUESTIONS ABOUT
51                                         ;CHAR/LINE, CHAR/DRUM , INPUTMODE AND
52                                         ;CHANNELS IN VFT
53 07473 002405          JMP      @INIRE
54
55 07474 000137 CHDST:   95.              ;MAX. PRINTABLE CHAR (64 +31 - DEC)
56 07475 000013 VCINI:   11.
57 07476 010400 EXINI:   INIEX
58 07477 013704 INCHS:   CHSET
59 07500 000000 INIRE:   0
60 07501 010044 IFCEN:   FCENEX
61 07502 007775 IRET0:   RET00
62 07503 012014 IVECH:   VCHNB
63 07504 000120 INICO:   80.
64 07505 007460 ACONT:   CONTF
```

```
| 0101  .MAIN
01
02 07506 010632 TSTEND: ISZ     PASCO   ;INCREMENT PASS-COUNTER
03 07507 006043         CCRLF
04 07510 006046         CDICL
05 07511 006040         CMESS
06 07512 014724         ENDPA           ;"END OF PASS: "
07 07513 006044         CDISP
08 07514 015241         DEOPA
09 07515 024623         LDA     1,PASCO
10 07516 006053         CTDEC           ;PRINT PASS-NUMBER
11 07517 006057         CDDEC
12 07520 006043         CCRLF
13 07521 006047         CDATT
14 07522 006046         CDICL
15 07523 062677         ICRST
16 07524 006127         SETBU           ;SET LPT-BUSY, WAIT UNTIL IT RETURNS
17 07525 000404         JMP     SETON   ;TO ZERO.
18 07526 006115         STATA           ;TEST STATUS
19 07527 000200         000200
20 07530 000412         JMP     TSTPA   ;NOT OFF-LINE
21 07531 006043 SETON:  CCRLF           ;OFF-LINE
22 07532 006046         CDICL
23 07533 006040         CMESS
24 07534 015477         SETOL           ;"SET LPT ON-LINE"
25 07535 006044         CDISP
26 07536 015477         SETOL
27 07537 006050         CHAAT
28 07540 006062         WATOP           ;   PROCEED
29 07541 020000         020000
30 07542 102220 TSTPA:  ADCZR   0,0     ;AC0:= 077777
31 07543 026414         LDA     1,@IPASC
32 07544 122032         ADCZ#   1,0,SZC ;IS PASCO >= 077777 ?
33 07545 002413         JMP     @ITSMAN ; NO
34 07546 102440         SUBC    0,0     ;YES
35 07547 042410         STA     0,@IPASC        ;RESET PASSCOUNTER
36 07550 002410         JMP     @ITSMAN ;REPEAT COMPLETE TEST
37
38
39 07551 007640 ZEASK:  QUE00
40 07552 010351 IMODE:  INMRK
41 07553 014170 PAMRK:  PASMK
42 07554 007435 TFIG0:  TESTF
43 07555 007436 TFIG1:  TESTF+1
44 07556 010402 ICHLM:  CHLMK
45 07557 007340 IPASC:  PASCO
46 07560 007341 ITSMAN: TSTMAN
```

```
I 0102  .MAIN
01
02                ;ROUTINES TO SET TESTFIGURE ACCORDING TO PRINTERTYPE
03
04
05 07561 054454 DP24:    STA     3,SETRE
06 07562 020657          LDA     0,TY24S ;SET TESTFIGURE
07 07563 040652          STA     0,TESTF
08 07564 020656          LDA     0,TY24S+1
09 07565 040651          STA     0,TESTF+1
10 07566 006444          JSR     @ISET1  ;SET MARKS AND CHARCODES
11 07567 002446          JMP     @SETRE
12
13 07570 054445 DP22:    STA     3,SETRE
14 07571 020652          LDA     0,TY22S ;SET TESTFIGURE
15 07572 040643          STA     0,TESTF
16 07573 020651          LDA     0,TY22S+1
17 07574 040642          STA     0,TESTF+1
18 07575 006435          JSR     @ISET1  ;SET MARKS AND CHARCODES
19 07576 002437          JMP     @SETRE
20
21 07577 054436 CENT1:   STA     3,SETRE
22 07600 020645          LDA     0,TYCT1 ;SET TESTFIGURE
23 07601 042753          STA     0,@TFIG0
24 07602 020644          LDA     0,TYCT1+1
25 07603 042752          STA     0,@TFIG1
26 07604 102000          ADC     0,0
27 07605 040567          STA     0,CEN1M
28 07606 006425          JSR     @ISET2  ;SET MARKS AND CHARCODES
29 07607 002426          JMP     @SETRE
30
31 07610 054425 CENT2:   STA     3,SETRE
32 07611 020636          LDA     0,TYCT2 ;SET TESTFIGURE
33 07612 042742          STA     0,@TFIG0
34 07613 020635          LDA     0,TYCT2+1
35 07614 042741          STA     0,@TFIG1
36 07615 006416          JSR     @ISET2  ;SET MARKS AND CHARCODES
37 07616 002417          JMP     @SETRE
38
39 07617 054416 LOGBX:   STA     3,SETRE
40 07620 022416          LDA     0,@TYLOI;SET TESTFIGURE
41 07621 042733          STA     0,@TFIG0
42 07622 010414          ISZ     TYLOI
43 07623 022413          LDA     0,@TYLOI
44 07624 042731          STA     0,@TFIG1
45 07625 014411          DSZ     TYLOI
46 07626 102000          ADC     0,0
47 07627 040545          STA     0,CEN1M
48 07630 006404          JSR     @ISET3  ;SET MARKS AND CHARCODES
49 07631 002404          JMP     @SETRE
50
51
52 07632 010524 ISET1:   SETDP
53 07633 010540 ISET2:   SETCE
54 07634 010555 ISET3:   SELOG
55 07635 000000 SETRE:   0
56 07636 007451 TYLOI:   TYLOG
57 07637 177677 MSK47:   177677
```

```
| 0103  .MAIN
01
02 07640 054535 QUE00:  STA     3,RET00 ;ASK FOR PRINTERTYPE
03 07641 006043 QUR00:  CCRLF
04 07642 006046         CDICL
05 07643 006040         CMESS
06 07644 014337         MTYAS           ;"PRINTERTYPE: RC3632,3633(0),RC3634,
07 07645 006044         CDISP           ;3635(1),RC3636(2),RC3641(3),RC3638(4),
08 07646 014305         DTYAS           ;RC3639(5), RC3637(6) OR RC3643(7) ? "
09 07647 006050         CHAAT
10 07650 006104         CGTOK
11 07651 000770         JMP     QUR00
12 07652 000767         JMP     QUR00   ;ILLEGAL INPUT
13 07653 020075         LDA     0,DIGIN
14 07654 101015         MOV#    0,0,SNR
15 07655 000431         JMP     TIC70   ;INPUT WAS 0
16 07656 014075         DSZ     DIGIN
17 07657 000402         JMP     .+2
18 07660 000431         JMP     TIC40   ;INPUT WAS 1
19 07661 014075         DSZ     DIGIN
20 07662 000402         JMP     .+2
21 07663 000431         JMP     TIC20   ;INPUT WAS 2
22 07664 014075         DSZ     DIGIN
23 07665 000402         JMP     .+2
24 07666 000431         JMP     TIC30   ;INPUT WAS 3
25 07667 014075         DSZ     DIGIN
26 07670 000402         JMP     .+2
27 07671 000431         JMP     TI101   ;INPUT WAS 4
28 07672 014075         DSZ     DIGIN
29 07673 000402         JMP     .+2
30 07674 000431         JMP     TI102   ;INPUT WAS 5
31 07675 014075         DSZ     DIGIN
32 07676 000402         JMP     .+2
33 07677 000431         JMP     LOGAB   ;INPUT WAS 6
34 07700 014075         DSZ     DIGIN
35 07701 000740         JMP     QUR00   ;ILLEGAL INPUT
36 07702 102000         ADC     0,0     ;INPUT WAS 7
37 07703 040132         STA     0,MRK47 ;SET 3643 MARK
38 07704 000413         JMP     TIC30
39
40
41 07705 054470 TICIN:  STA     3,RET00
42 07706 004653 TIC70:  JSR     DP24    ;SET TESTFIGURE
43 07707 020454         LDA     0,ADP70
44 07710 000422         JMP     SETTI
45 07711 004650 TIC40:  JSR     DP24    ;SET TESTFIGURE
46 07712 020452         LDA     0,ADP40
47 07713 000417         JMP     SETTI
48 07714 004645 TIC20:  JSR     DP24    ;SET TESTFIGURE
49 07715 020450         LDA     0,ADP20
50 07716 000414         JMP     SETTI
51 07717 004651 TIC30:  JSR     DP22    ;SET TESTFIGURE
52 07720 020446         LDA     0,ADP30
53 07721 000411         JMP     SETTI
54 07722 004655 TI101:  JSR     CENT1   ;SET TESTFIGURE
55 07723 020444         LDA     0,AC101
56 07724 000406         JMP     SETTI
57 07725 004663 TI102:  JSR     CENT2   ;SET TESTFIGURE
58 07726 020442         LDA     0,AC102
59 07727 000403         JMP     SETTI
60 07730 004667 LOGAB:  JSR     LOGBX   ;SET TESTFIGURE
61 07731 020440         LDA     0,ALOGB
```

```
I 0104  .MAIN
01
02 07732 040020 SETTI:  STA     0,IDX0
03 07733 022020         LDA     0,@IDX0 ;SET LPT-TIMING-CONSTANTS ACCORDING
04 07734 040141         STA     0,TDOS  ;TO PRINTERTYPE
05 07735 022020         LDA     0,@IDX0
06 07736 040142         STA     0,TDOSC
07 07737 022020         LDA     0,@IDX0
08 07740 040143         STA     0,TS15L
09 07741 006043 QURXX:  CCRLF
10 07742 006046         CDICL
11 07743 006040         CMESS
12 07744 015507         MA307
13 07745 006044         CDISP
14 07746 015652         DA307
15 07747 006050         CHAAT
16 07750 006104         CGTOK
17 07751 000770         JMP     QURXX
18 07752 000767         JMP     QURXX
19 07753 020075         LDA     0,DIGIN
20 07754 101015         MOV#    0,0,SNR
21 07755 000404         JMP     .+4     ;INPUT WAS 0
22 07756 014075         DSZ     DIGIN
23 07757 000762         JMP     QURXX   ;INPUT WAS <> 0 OR <> 1
24 07760 102000         ADC     0,0     ;INPUT WAS 1
25 07761 040132         STA     0,MRK47
26 07762 002413         JMP     @RET00
27
28 07763 007775 ADP70:  DP70C-1         ;POINTERS TO DIFF. TIMING-CONSTANTS
29 07764 010000 ADP40:  DP40C-1
30 07765 010003 ADP20:  DP20C-1
31 07766 010006 ADP30:  DP30C-1
32 07767 010011 AC101:  C101C-1
33 07770 010014 AC102:  C102C-1
34 07771 010017 ALOGB:  LOGBC-1
35 07772 000000 MARKL:  0               ;=177777 IF LPT = LOGABAX ELSE 0
36 07773 000000 MARKC:  0               ;=177777 IF LPT = CENT. OR LOG. ELSE 0
37 07774 000000 CEN1M:  0               ;=177777 IF LPT = CENT,101AL ELSE 0
38 07775 000000 RET00:  0
39
40              ;TABLE WITH PRINTER-TIMINGCONSTANTS. UNITS IN MS
41 07776 000170 DP70C:  120.            ;TIME FOR DONE TO SET AFTER A PRINTCOM.
42 07777 000106         70.             ;SAME AS PREVIOUS - BUT CRITICAL
43 10000 000132         90.             ;TIME TO SPACE 15 LINES
44
45 10001 000372 DP40C:  250.
46 10002 000226         150.
47 10003 000132         90.
48
49 10004 000764 DP20C:  500.
50 10005 000454         300.
51 10006 000240         160.
52
53 10007 000620 DP30C:  400.
54 10010 000372         250.
55 10011 000264         180.
56
57 10012 000144 C101C:  100.            ;NO MEANING-BUT > TIME FOR PRINT 1 CHAR
58 10013 000144         100.
59 10014 000132         90.             ;TIME TO SPACE ONE LINE
60
61 10015 000144 C102C:  100.            ;NO MEANING-BUT > TIME FOR PRINT 1 CHAR
62 10016 000144         100.
63 10017 000132         90.             ;TIME TO SPACE ONE LINE
64
65 10020 000310 LOGBC:  200.            ;NO MEANING-BUT > TIME FOR PRINT 1 CHAR
66 10021 000310         200.
67 10022 000372         250.            ;TIME TO SPACE ONE LINE
```

```
I 0105  .MAIN
01
02 10023 054752 QUE01:  STA     3,RET00 ;ASK FOR CONTROLLERTYPE
03 10024 006043 QUR01:  CCRLF
04 10025 006046         CDICL
05 10026 006040         CMESS
06 10027 014315         MCOAS           ;"FIRST(1) OR SECOND(2) CONTROLLER ? "
07 10030 006044         CDISP
08 10031 015247         DCOAS
09 10032 006050         CHAAT
10 10033 006104         CGTOK
11 10034 000770         JMP     QUR01
12 10035 000767         JMP     QUR01   ;ILLEGAL INPUT
13 10036 014075         DSZ     DIGIN
14 10037 000402         JMP     .+2
15 10040 000404         JMP     FCENEX  ;INPUT WAS 1
16 10041 014075         DSZ     DIGIN
17 10042 000762         JMP     QUR01   ;INPUT WAS NOT 1 OR 2
18 10043 000405         JMP     SCENEX  ;INPUT WAS 2
19
20
21 10044 020727 FCENEX: LDA     0,MARKC
22 10045 101015         MOV#    0,0,SNR ;IS MARKC = 177777 ?
23 10046 000414         JMP     FIDEV   ;NO, LPT = DP-FIRST
24 10047 000405         JMP     CRFIR   ;YES, LPT = CENTR. OR LOGABAX-FIRST
25
26 10050 020723 SCENEX: LDA     0,MARKC
27 10051 101015         MOV#    0,0,SNR ;IS MARKC = 177777 ?
28 10052 000406         JMP     SEDEV   ;NO, LPT = DP-SECOND
29 10053 000403         JMP     CRSEC   ;YES, LPT = CENTR. OR LOGABAX-SECOND
30
31              ;ROUTINE TO CHANGE ALL I/O-INSTR. TO LPT, SO THAT THE DEVICE-
32              ;NO = THE SELECTED PRINTER AND CONTROLLER
33
34 10054 020135 CRFIR:  LDA     0,DEV37
35 10055 000406         JMP     DVTST
36 10056 020137 CRSEC:  LDA     0,DEV67
37 10057 000404         JMP     DVTST
38 10060 020136 SEDEV:  LDA     0,DEV57
39 10061 000402         JMP     DVTST
40 10062 020134 FIDEV:  LDA     0,DEV17
41
42 10063 024133 DVTST:  LDA     1,LASTD
43 10064 106415         SUB#    0,1,SNR ;IS LASTD = THE WANTED DEVICENO.?
44 10065 002710         JMP     @RET00  ;YES- CHANGE OF DEV.CODE ISN'T NECESSARY
45 10066 040133 CHADV:  STA     0,LASTD ;NO - CHANGE ALL LPT-I/O-INSTUCTIONS
46 10067 030424         LDA     2,BEGAD
47 10070 050427         STA     2,ISTPOI
48 10071 034423         LDA     3,ENDAD
49 10072 156420         SUBZ    2,3
50 10073 054425         STA     3,INSCOU
51 10074 032423 NXINS:  LDA     @2,ISTPOI;AC2:=INSTRUCTION TO BE EXAMINED
52 10075 010422         ISZ     ISTPOI
53 10076 034417         LDA     3,IOMSK ;AC3:=160077
54 10077 014421         DSZ     INSCOU
55 10100 157401         AND     2,3,SKP
56 10101 002674         JMP     @RET00  ;ALL INSTRUCTIONS ARE EXAMINED
57 10102 166414         SUB#    3,1,SZR ;AC1:=I/O-INST. TO LATEST USED DEV.NO.
58                                      ;WITH OUTMASKED FUNCTION-BITS
59 10103 000771         JMP     NXINS   ;EXAMINE NEXT INSTRUCTION
60 10104 034412         LDA     3,FUMSK ;THE INST. WAS AN I/O-INST. TO LPT
61                                      ;AC3:= 017700
62 10105 173400         AND     3,2     ;AC2:= THE ACTUEL FUNCTION-BITS
63 10106 113000         ADD     0,2
64 10107 034410         LDA     3,ISTPOI
65 10110 051777         STA     2,-1,3
66 10111 000763         JMP     NXINS   ;GET NEXT INSTRUCTION
```

```
| 0106  .MAIN
01
02 10112 000000 MOMRK:  0
03 10113 000412 BEGAD:  XMESS
04 10114 016242 ENDAD:  EOFTS
05 10115 160077 IOMSK:  160077
06 10116 017700 FUMSK:  017700
07 10117 000000 ISTPOI: 0
08 10120 000000 INSCOU: 0
09 10121 012014 IVCHNB: VCHNB
10
11
12 10122 054421 QUE02:  STA     3,RET01 ;ASK FOR NO. OF CHANNELS IN VFU.
13 10123 006043 QUR02:  CCRLF
14 10124 006046         CDICL
15 10125 006040         CMESS
16 10126 014441         MVCHN           ;"NO. OF CHANNELS IN VERTICAL FORMAT
17 10127 006044         CDISP           ;UNIT ? "
18 10130 014467         DVCHN
19 10131 006050         CHAAT
20 10132 006105         CGTDC
21 10133 000770         JMP     QUR02
22 10134 000767         JMP     QUR02   ;ILLEGAL INPUT
23 10135 014075         DSZ     DIGIN
24 10136 000402         JMP     .+2
25 10137 000764         JMP     QUR02
26 10140 020075         LDA     0,DIGIN
27 10141 042760         STA     @0,IVCHNB;STORE NO.-1 IN VCHNB
28 10142 002401         JMP     @RET01
29
30 10143 000000 RET01:  0
```

```
| 0107  .MAIN
01
02 10144 054777 QUE03:  STA     3,RET01 ;ASK FOR DIAGNOSTICTESTS
03 10145 006043 QUR03:  CCRLF
04 10146 006046         CDICL
05 10147 006040         CMESS
06 10150 014500         MDIAT           ;"DIAGNOSTICTEST(1) OR NOT(0) ? "
07 10151 006044         CDISP
08 10152 014430         DDIAT
09 10153 006050         CHAAT
10 10154 006103         CGTBI
11 10155 000770         JMP     QUR03
12 10156 000767         JMP     QUR03   ;ILLEGAL INPUT
13 10157 034764         LDA     3,RET01
14 10160 014075         DSZ     DIGIN
15 10161 001400         JMP     0,3     ;INPUT WAS 0
16 10162 020415 YEDIA:  LDA     0,JMP01 ;INPUT WAS 1
17 10163 042416         STA     0,@FIQUE
18 10164 001401         JMP     1,3
19
20
21
22 10165 102440 NODIA:  SUBO    0,0     ;MASK OUT DIAGNOSTICTESTS
23 10166 030545         LDA     2,TFIGU
24 10167 041000         STA     0,0,2
25 10170 020412         LDA     0,NDCON
26 10171 025001         LDA     1,1,2
27 10172 123400         AND     1,0
28 10173 041001         STA     0,1,2
29 10174 020404         LDA     0,JMP10 ;SET JMP-INSTR TO PASS-BY "TESTEXECU.
30 10175 042404         STA     0,@FIQUE;START"-MESSAGE
31 10176 002405         JMP     @RENOD  ;RETURN TO ASK5
32
33 10177 000401 JMP01:  JMP     .+1
34 10200 000410 JMP10:  JMP     .+10
35 10201 007274 FIQUE:  GUEFI
36 10202 000007 NDCON:  7
37 10203 007266 RENOD:  ASK5
38
39
40
41 10204 054737 QUE04:  STA     3,RET01 ;ASK FOR OPERATOR-TESTS
42 10205 006043 QUR04:  CCRLF
43 10206 006046         CDICL
44 10207 006040         CMESS
45 10210 014520         MMANT           ;"TESTS WITH OPERATOR-INVOLVEMENT(1)
46 10211 006044         CDISP           ;OR NOT(0) ? "
47 10212 014550         DMANT
48 10213 006050         CHAAT
49 10214 006103         CGTBI
50 10215 000770         JMP     QUR04
51 10216 000767         JMP     QUR04   ;ILLEGAL INPUT
52 10217 034724         LDA     3,RET01
53 10220 014075         DSZ     DIGIN
54 10221 001400         JMP     0,3     ;INPUT WAS 0
55 10222 175400         INC     3,3     ;PASS-BY QUEST. ABOUT PAPER-COMP.-TESTS
56 10223 175400         INC     3,3     ;BUT ASK FOR CHAN/VFT
57 10224 000440         JMP     INYEP
58
59
60 10225 030506 NOMAN:  LDA     2,TFIGU ;MASK OUT OPERATOR-TESTS
61 10226 021001         LDA     0,1,2
62 10227 024753         LDA     1,NDCON
63 10230 123400         AND     1,0
64 10231 041001         STA     0,1,2
65 10232 001400         JMP     0,3
```

```
| 0108  ,MAIN
01
02 10233 054543 QUE4A:  STA     3,RET02
03 10234 006043 QUR4A:  CCRLF
04 10235 006046         CDICL
05 10236 006040         CMESS
06 10237 015365         MPACO           ;"TESTS WITH PAPERCONSUMPTION(1)
07 10240 006044         CDISP           ;OR NOT(0) ? "
08 10241 015354         CPACO
09 10242 006050         CHAAT
10 10243 006103         CGTBI
11 10244 000770         JMP     QUR4A
12 10245 000767         JMP     QUR4A   ;ILLEGAL INPUT
13 10246 014075         DSZ     DIGIN
14 10247 000406         JMP     NOPAP   ;INPUT WAS 0
15 10250 020530         LDA     0,INIEX ;INPUT WAS 1
16 10251 101014         MOV#    0,0,SZR
17 10252 002524         JMP     @RET02
18 10253 000424         JMP     YEPAP
19
20
21 10254 054522 INNOP:  STA     3,RET02
22 10255 030456 NOPAP:  LDA     2,TFIGU
23 10256 021000         LDA     0,0,2
24 10257 024404         LDA     1,NPCON
25 10260 123400         AND     1,0
26 10261 041000         STA     0,0,2
27 10262 002514         JMP     @RET02
28
29 10263 154677 NPCON:  154677
30
31
32
33
34
35 10264 054512 INYEP:  STA     3,RET02
36 10265 020513         LDA     0,INIEX
37 10266 101014         MOV#    0,0,SZR ;IS INITIALIZED EXE. WANTED ?
38 10267 002507         JMP     @RET02  ;YES
39 10270 020504         LDA     0,CHINS ;NO
40 10271 040414         STA     0,SAVIS
41 10272 020705         LDA     0,JMP01
42 10273 040501         STA     0,CHINS
43 10274 004456         JSR     QUE06   ;ASK FOR CHAR/LINE
44 10275 020410         LDA     0,SAVIS
45 10276 040476         STA     0,CHINS
46 10277 022405 YEPAP:  LDA     0,@CMARK
47 10300 101014         MOV#    0,0,SZR
48 10301 002475         JMP     @RET02  ;PRINTER IS CENTRONICS OR LOGABAX
49 10302 004620         JSR     QUE02   ;ASK FOR NUMBER OF CHAN. IN VFT.
50 10303 002473         JMP     @RET02
51
52
53 10304 007773 CMARK:  MARKC
54 10305 000000 SAVIS:  0
```

```
| 0109  .MAIN
01
02 10306 054635 QUE05:  STA     3,RET01 ;ASK FOR CHAR.PRINTINGPROGRAM
03 10307 006043 QUR05:  CCRLF
04 10310 006046         CDICL
05 10311 006040         CMESS
06 10312 014561         MMOTS           ;"CHARACTERPRINTTEST(1) OR NOT(0) ? "
07 10313 006044         CDISP
08 10314 014603         DMOTS
09 10315 006050         CHAAT
10 10316 006103         CGTBI
11 10317 000770         JMP     QUR05
12 10320 000767         JMP     QUR05   ;ILLEGAL INPUT
13 10321 034622         LDA     3,RET01
14 10322 014075         DSZ     DIGIN
15 10323 001400         JMP     0,3     ;INPUT WAS 0
16 10324 001401         JMP     1,3     ;INPUT WAS 1
17
18
19 10325 030406 NOMO1:  LDA     2,TFIGU ;MASK OUT CHAR.PRINTERPROGRAM
20 10326 021001         LDA     0,1,2
21 10327 024405         LDA     1,CNOM1
22 10330 123400         AND     1,0
23 10331 041001         STA     0,1,2
24 10332 001401         JMP     1,3
25
26 10333 007435 TFIGU:  TESTF
27 10334 177773 CNOM1:  177773
28
29 10335 054441 YEMO1:  STA     3,RET02
30 10336 020442         LDA     0,INIEX
31 10337 101014         MOV#    0,0,SZR ;DID THE OPERATOR WANT INITIALIZED EXE.?
32 10340 000554         JMP     ASKTB   ;YES - SELECT PRT-TABLE
33 10341 102000         ADC     0,0     ;NO - AC0:=177777
34 10342 042546         STA     0,@IMCMK
35 10343 004407         JSR     QUE06   ;ASK FOR CHAR/LINE- STORE INPUT IN CHALI
36 10344 004437         JSR     QUE07   ;ASK FOR INPUT-MODE
37 10345 102001         ADC     0,0,SKP ;OCTAL-MODE
38 10346 102440         SUBO    0,0     ;ASCII-MODE
39 10347 040402         STA     0,INMRK
40 10350 002426         JMP     @RET02
41
42
43
44 10351 000000 INMRK:  0
45
46 10352 054425 QUE06:  STA     3,RET03 ;ASK FOR NO. OF CHAR. PER LINE
47 10353 020427         LDA     0,CHLMK
48 10354 101014         MOV#    0,0,SZR ;IS CHAR/LINE DEFINED BEFORE?
49 10355 000417         JMP     CHINS   ;YES
50 10356 006043 QUR06:  CCRLF
51 10357 006046         CDICL
52 10360 006040         CMESS
53 10361 015045         MCHPL           ;"NO. OF CHARACTERS PER LINE ? "
54 10362 006044         CDISP
55 10363 015064         DCHPL
56 10364 006050         CHAAT
57 10365 006105         CGTDC
58 10366 000770         JMP     QUR06
59 10367 000767         JMP     QUR06   ;ILLEGAL INPUT
60 10370 102000         ADC     0,0
61 10371 040411         STA     0,CHLMK
62 10372 020075         LDA     0,DIGIN
63 10373 040147         STA     0,CHALI
64 10374 006405 CHINS:  JSR     @IGU6A  ;ASK FOR USE OF PRT.TABLE
65 10375 002402         JMP     @RET03
```

```
| 0110  .MAIN
01
02 10376 000000 RET02:  0
03 10377 000000 RET03:  0
04 10400 000000 INIEX:  0
05 10401 010633 IQU6A:  GUE6A
06 10402 000000 CHLMK:  0               ;=177777 WHEN CHAR/LINE IS DEFINED
07
08
09 10403 054774 QUE07:  STA     3,RET03 ;ASK FOR INPUTMODE
10 10404 006043 QUR07:  CCRLF
11 10405 006046         CDICL
12 10406 006040         CMESS
13 10407 015201         MINMO           ;"INPUTMODE = ASCII-ALPHABET(1) OR
14 10410 006044         CDISP           ;OCTAL(0) ? "
15 10411 015230         DINMO
16 10412 006050         CHAAT
17 10413 006103         CGTBI
18 10414 000770         JMP     QUR07
19 10415 000767         JMP     QUR07   ;ILLEGAL INPUT
20 10416 034761         LDA     3,RET03
21 10417 014075         DSZ     DIGIN
22 10420 001400         JMP     0,3     ;INMODE = OCTAL
23 10421 001401         JMP     1,3     ;INMODE = ASCII
24
25
26
27 10422 054754 QUE08:  STA     3,RET02 ;ASK FOR CHAR.MOTIONPROGRAM
28 10423 006043 QUR08:  CCRLF
29 10424 006046         CDICL
30 10425 006040         CMESS
31 10426 015106         MROTT           ;"CHARACTERMOTION-TEST(1) OR NOT(0) ? "
32 10427 006044         CDISP
33 10430 014614         DROTT
34 10431 006050         CHAAT
35 10432 006103         CGTBI
36 10433 000770         JMP     QUR08
37 10434 000767         JMP     QUR08   ;ILLEGAL INPUT
38 10435 034741         LDA     3,RET02
39 10436 014075         DSZ     DIGIN
40 10437 001400         JMP     0,3     ;CHAR.MOT.TEST NOT WANTED
41 10440 001401         JMP     1,3     ;CHAR.MOT.TEST WANTED
42
43
44 10441 030672 NOROT:  LDA     2,TFIGU ;MASK OUT CHAR.ROTATIONPROGRAM
45 10442 021001         LDA     0,1,2
46 10443 024404         LDA     1,NRCON
47 10444 123400         AND     1,0
48 10445 041001         STA     0,1,2
49 10446 001401         JMP     1,3
50
51 10447 177775 NRCON:  177775
52
53 10450 054437 YEROT:  STA     3,RET04
54 10451 020727         LDA     0,INIEX
55 10452 101014         MOV#    0,0,SZR ;IS INITIALIZED EXE. WANTED ?
56 10453 000441         JMP     ASKTB   ;YES - SELECT PRT-TABLE
57 10454 022434         LDA     0,@IMOMK;NO
58 10455 101015         MOV#    0,0,SNR ;IS CHAR/LINE DEFINED BEFORE ?
59 10456 004674         JSR     QUE06   ;NO - ASK FOR CHAR/LINE
60 10457 004402         JSR     QUE09   ;YES - ASK FOR CHAR/DRUM
61 10460 002427         JMP     @RET04
```

```
| 0111  .MAIN
01
02 10461 054430 QUE09:  STA     3,RET05
03 10462 022502          LDA     0,@ICMRK
04 10463 101015          MOV#    0,0,SNR
05 10464 000403          JMP     QUR09
06 10465 020426          LDA     0,CH064 ;PRINTER IS CENTRONICS OR LOGABAX
07 10466 000414          JMP     QU9EX
08 10467 006043 QUR09:   CCRLF           ;ASK FOR NO. OF CHAR. ON DRUM
09 10470 006046          CDICL
10 10471 006040          CMESS
11 10472 014625          MCHPD           ;"NO. OF CHARACTERS ON DRUM ? "
12 10473 006044          CDISP
13 10474 014644          DCHPD
14 10475 006050          CHAAT
15 10476 006105          CGTDC
16 10477 000770          JMP     QUR09
17 10500 000767          JMP     QUR09   ;ILLEGAL INPUT
18 10501 020075          LDA     0,DIGIN
19 10502 024410 QU9EX:   LDA     1,HOH31
20 10503 123000          ADD     1,0     ;LAST PRINTABLE CHAR. IS NO.+31
21 10504 042402          STA     @0,ICHSE;STA NO. IN CHSET
22 10505 002404          JMP     @RET05
23
24 10506 013704 ICHSE:   CHSET
25 10507 000000 RET04:   0
26 10510 010112 IMOMK:   MOMRK
27 10511 000000 RET05:   0
28 10512 000037 HOH31:   31.
29 10513 000100 CH064:   64.
30
31
32 10514 054463 ASKTB:   STA     3,TBRET
33 10515 020463          LDA     0,TDEFM
34 10516 101014          MOV#    0,0,SZR ;IS PRT-TABLE DEFINED BEFORE
35 10517 001400          JMP     0,3     ;YES
36 10520 102000          ADC     0,0     ;NO
37 10521 040457          STA     0,TDEFM
38 10522 004464          JSR     QTABL   ;SELECT PRINTER-TABLE
39 10523 002454          JMP     @TBRET
```

```
| 0112  .MAIN
01
02                ;THE ROUTINES SETDP,SETCE AND SELOG WILL PLACE THE ACTUAL PAPER-
03                ;COMMANDS IN THE LOCATIONS WHERE THEY WILL BE USED DURING THE
04                ;TEST. THE COMMANDCODE IS SET ACCORDING TO THE PRINTERTYPE
05
06 10524 102440 SETDP:  SUBO    0,0
07 10525 042437         STA     0,@ICMRK;MARKC=0
08 10526 042444         STA     0,@ILMRK;MARKL:=0
09 10527 020444         LDA     0,DOSE1
10 10530 042445         STA     0,@IDOSE;PLACE A COMMAND TO SET DONE-FF
11 10531 020434         LDA     0,DPLIF ;A LINE-FEEDCOMMAND
12 10532 040146         STA     0,SPA1L
13 10533 020433         LDA     0,DPFF  ;A FORMFEED-COMMAND
14 10534 040145         STA     0,PRTFF
15 10535 020444         LDA     0,SPDPR
16 10536 040151         STA     0,LDSPA
17 10537 001400         JMP     0,3
18
19 10540 102000 SETCE:  ADC     0,0
20 10541 042423         STA     0,@ICMRK;MARKC=177777
21 10542 102440         SUBO    0,0
22 10543 042427         STA     0,@ILMRK;MARKL:=0
23 10544 020427         LDA     0,DOSE1
24 10545 042430         STA     0,@IDOSE;PLACE A COMMAND TO SET DONE-FF
25 10546 020421         LDA     0,CELIF ;A LINE-FEEDCOMMAND
26 10547 040146         STA     0,SPA1L
27 10550 020420         LDA     0,CEFF  ;A FORM-FEEDCOMMAND
28 10551 040145         STA     0,PRTFF
29 10552 020430         LDA     0,SPCEN
30 10553 040151         STA     0,LDSPA
31 10554 001400         JMP     0,3
32
33 10555 054421 SELOG:  STA     3,LOGRE
34 10556 004762         JSR     SETCE   ;SET CENTRONICS-PAPERCOMMANDS
35 10557 020415         LDA     0,DOSE2
36 10560 042415         STA     0,@IDOSE;PLACE COMMAND TO SET DONE-FF
37 10561 102000         ADC     0,0
38 10562 042410         STA     0,@ILMRK;MARKL:=177777
39 10563 002413         JMP     @LOGRE
40
41 10564 007773 ICMRK:  MARKC
42 10565 002501 DPLIF:  2501
43 10566 002400 DPFF:   2400
44 10567 002412 CELIF:  2412
45 10570 002414 CEFF:   2414
46 10571 014170 MKPAS:  PASMK
47 10572 007772 ILMRK:  MARKL
48 10573 002000 DOSE1:  2000
49 10574 002015 DOSE2:  2015
50 10575 012525 IDOSE:  DOSET
51 10576 000000 LOGRE:  0
52 10577 000000 TBRET:  0
53 10600 000000 TDEFM:  0               ;=177777 IF PRT-TABLE IS DEFINED ELSE 0
```

```
| 0113  .MAIN
01
02 10601 000037 SPDPR:  37
03 10602 000040 SPCEN:  40
04 10603 002040 SPTAB:  2040
05 10604 117400 DPTSP:  117400          ;"SPACE"-CHAR, DP-PRINTER ("37")
06 10605 120000 CETSP:  120000          ;"SPACE"-CHAR, CENTRONICS OR LOGABAX
07                                      ;("40")
08
09 10606 054703 QTABL:  STA     3,RET05
10 10607 022755         LDA     0,@ICMRK
11 10610 101015         MOV#    0,0,SNR
12 10611 000403         JMP     QUR6A
13 10612 102000         ADC     0,0     ;PRINTER IS CENTRONICS OR LOGABAX
14 10613 000416         JMP     QUE6F
15 10614 006043 QUR6A:  CCRLF
16 10615 006046         CDICL
17 10616 006040         CMESS           ;"USE PRINTERTABLE(1) OR PASS-BY(0) ?"
18 10617 015413         MUTAB
19 10620 006044         CDISP
20 10621 015436         DUTAB
21 10622 006050         CHAAT
22 10623 006103         CGTBI
23 10624 000770         JMP     QUR6A
24 10625 000767         JMP     QUR6A   ;ILLEGAL INPUT
25 10626 014075         DSZ     DIGIN
26 10627 102001         ADC     0,0,SKP ;INPUT WAS 0
27 10630 102440         SUBO    0,0     ;INPUT WAS 1
28 10631 042740 QUE6F:  STA     0,@MKPAS
29 10632 002657         JMP     @RET05
30
31
32 10633 054437 QUE6A:  STA     3,RET06
33 10634 022730         LDA     0,@ICMRK
34 10635 101015         MOV#    0,0,SNR
35 10636 000403         JMP     QTABR   ;PRINTER IS DP
36 10637 102440         SUBO    0,0     ;PRINTER IS CENTRONICS OR LOGABAX
37 10640 000446         JMP     TABL0
38 10641 006043 QTABR:  CCRLF
39 10642 006046         CDICL
40 10643 006040         CMESS
41 10644 015535         TABME           ;"SELECT PR.TABLE"
42 10645 006044         CDISP
43 10646 015524         TABDE
44 10647 006050         CHAAT
45 10650 006104         CGTOK
46 10651 000770         JMP     QTABR
47 10652 000767         JMP     QTABR   ;ILLEGAL INPUT
48 10653 020075         LDA     0,DIGIN
49 10654 101015         MOV#    0,0,SNR
50 10655 000431         JMP     TABL0   ;INPUT WAS 0
51 10656 014075         DSZ     DIGIN
52 10657 000402         JMP     .+2
53 10660 000426         JMP     TABL0   ;INPUT WAS 1
54 10661 014075         DSZ     DIGIN
55 10662 000402         JMP     .+2
56 10663 000423         JMP     TABL0   ;INPUT WAS 2
57 10664 014075         DSZ     DIGIN
58 10665 000402         JMP     .+2
59 10666 000420         JMP     TABL0   ;INPUT WAS 3
60 10667 014075         DSZ     DIGIN
61 10670 000751         JMP     QTABR   ;ILLEGAL INPUT
62 10671 000420         JMP     TABL4   ;INPUT WAS 4
```

```
| 0114  .MAIN
01
02 10672 000000 RET06:  0
03 10673 000401 JMPON:  JMP     .+1
04 10674 000403 JMP03:  JMP     .+3
05 10675 001577 ASAVE:  SAVE
06 10676 001620 LDTAB:  START
07 10677 010740 ALDRT:  RDRET
08 10700 004662 POADR:  POWON
09 10701 000000 REDSV:  0
10 10702 002230 PTBMO:  MOPTB
11 10703 063077 HALTC:  HALT
12 10704 002125 INSRE:  JMP     @IRDER
13 10705 001664 ICHKE:  CHKER
14
15 10706 111400 TABL0:  INC     0,2     ;AC2:= PRINTERTABLENO.
16 10707 006773         JSR     @PTBMO  ;SET CORRECT PRINTERTABLE,RETURN .+1
17 10710 000450         JMP     SETSP
18
19
20              ;ROUTINE TO LOAD A NEW PRINTERTABLE FROM PTR BY USE OF
21              ;THE BINARY-LOADER-TS
22
23 10711 020766 TABL4:  LDA     0,ALDRT ;SET PRT.TABLE SELF-START ADDRESS
24 10712 040100         STA     0,POWRE
25 10713 020760         LDA     0,JMPON
26 10714 040101         STA     0,PRINT
27 10715 030761         LDA     2,LDTAB ;CHANGE READS-INSTR. IN BINARY-LOADER
28 10716 021001         LDA     0,1,2   ;SAVE THE READS-INSTR
29 10717 040762         STA     0,REDSV
30 10720 102000         ADC     0,0     ;AC0:= 177777
31 10721 042754         STA     0,@ASAVE;LET SWITCH-REGISTER SELECT PTR
32 10722 020752         LDA     0,JMP03
33 10723 041001         STA     0,1,2
34 10724 020760         LDA     0,INSRE
35 10725 042760         STA     0,@ICHKE;CHANGE CHECKERROR-HALT TO JMP RDERR
36 10726 006043 TAB4R:  CCRLF
37 10727 006046         CDICL
38 10730 006040         CMESS
39 10731 015623         MLPTR           ;"LOAD PTR, STRIKE RETURN-KEY"
40 10732 006044         CDISP
41 10733 015641         DLPTR           ;"LDA PR-TYPE CONT"
42 10734 006050         CHAAT
43 10735 006062         WATOP
44 10736 020000         020000
45 10737 002737         JMP     @LDTAB  ;GO TO BINARY-LOADER-TS
46
47              ;RETURN TO HERE FROM BINARY-LOADER, THROUGH PRINTERTABLE-
48              ;SELFSTART
49
50 10740 030736 RDRET:  LDA     2,LDTAB ;RESTORE INSTR. IN BINARY-LOADER-TS
51 10741 020740         LDA     0,REDSV
52 10742 041001         STA     0,1,2
53 10743 020735         LDA     0,POADR
54 10744 040100         STA     0,POWRE
55 10745 020736         LDA     0,HALTC
56 10746 042737         STA     0,@ICHKE
57 10747 000411         JMP     SETSP
```

```
| 0115  .MAIN
01
02                ;RETURN TO HERE WHEN A CHECKSUM-ERROR IS DETECTED BY THE
03                ;BINARY-LOADER
04
05 10750 006043 RDERR:  CCRLF
06 10751 006046         CDICL
07 10752 006040         CMESS
08 10753 015613         CHKSE
09 10754 006044         CDISP
10 10755 015613         CHKSE           ;"CHECKSUM-ERROR"
11 10756 006047         CDATT
12 10757 000747         JMP     TAB4R   ;RELOAD TAPE
13
14 10760 022604 SETSP:  LDA     0,@ICMRK;INSERT CORRECT SPACE-CHAR AFTER
15 10761 101015         MOV#    0,0,SNR ;HAVING MOVED PRINTERTABLE
16 10762 000405         JMP     IDPSP   ;PRINTER IS DP
17 10763 020622         LDA     0,CETSP ;PRINTER IS CENTRONICS OR LOGABAX
18 10764 042617 SPOUT:  STA     0,@SPTAB;INSERT SPACE IN PRT.TABLE (LOC 2040)
19 10765 004621         JSR     QTABL   ;ASK FOR USE OF PR.-TABLE
20 10766 002704         JMP     @RET06
21
22 10767 020615 IDPSP:  LDA     0,DPTSP
23 10770 000774         JMP     SPOUT
```

```
| 0116  .MAIN
01                ;
02                ; LINE/SERIAL - PRINTERTEST. START OF TESTROUTINEGROUPS
03
04
05                ;CPU- AND BUSTESTS (AND OTHERS)
06
07
08 10771 054124 CT00:   STA     3,GRORET
09 10772 102520         SUBZL   0,0       ;AC0:= 1
10 10773 042477         STA     0,@ITRW1;MODIFY WAIT-CONST. IN XLRST-ROUT.
11 10774 006112         SETP2
12 10775 063500         SKPBZ   0
13 10776 006114         EHALT             ;THE SELB-LINE IS GROUNDED
14 10777 006113         LOOP
15
16 11000 006112 CT01:   SETP2
17 11001 063700         SKPDZ   0
18 11002 006114         EHALT             ;THE SELD-LINE IS GROUNDED
19 11003 006113         LOOP
20
21 11004 006112 CT02:   SETP2
22 11005 102000         ADC     0,0
23 11006 060400         DIA     0,0
24 11007 100015         COM#    0,0,SNR
25 11010 006114         EHALT             ;DIA DID NOT READ ANYTHING (CPU-ERROR)
26 11011 006113         LOOP
27
28 11012 006112 CT03:   SETP2
29 11013 102440         SUBO    0,0
30 11014 060400         DIA     0,0
31 11015 101004         MOV     0,0,SZR
32 11016 006114         EHALT             ;DATALINES ARE GROUNDED
33 11017 006113         LOOP
34
35 11020 006112 CT04:   SETP2
36 11021 062677         IORST
37 11022 063517         SKPBZ   XLPT
38 11023 006114         EHALT             ;I/O-RESET FAILED TO CLEAR
39 11024 006113         LOOP              ;LPT-BUSY
40
41 11025 006112 CT05:   SETP2
42 11026 060217         NIOC    XLPT
43 11027 063517         SKPBZ   XLPT
44 11030 006114         EHALT             ;CLEAR-PULSE CAN'T CLEAR LPT-BUSY
45 11031 006113         LOOP
46
47 11032 006112 CT06:   SETP2
48 11033 062677         IORST
49 11034 063717         SKPDZ   XLPT
50 11035 006114         EHALT             ;I/O-RESET FAILED TO CLEAR
51 11036 006113         LOOP              ;LPT-DONE
52
53 11037 006112 CT07:   SETP2
54 11040 060217         NIOC    XLPT
55 11041 063717         SKPDZ   XLPT
56 11042 006114         EHALT             ;CLEAR-PULSE CAN'T CLEAR LPT-DONE
57 11043 006113         LOOP
58
59 11044 006112 CT08:   SETP2
60 11045 063417         SKPBN   XLPT
61 11046 000402         JMP     .+2
62 11047 006114         EHALT             ;CPU-SKIP-ERROR
63 11050 006113         LOOP
```

```
! 0117  .MAIN
01
02
03 11051 006112 CT09:   SETP2
04 11052 063617         SKPDN   XLPT
05 11053 000402         JMP     .+2
06 11054 006114         EHALT           ;CPU-SKIP-ERRCR
07 11055 006113         LOOP
08
09 11056 006112 CT10:   SETP2
10 11057 060177         INTEN
11 11060 000401         JMP     .+1
12 11061 063477         SKPBN   CPU
13 11062 006114         EHALT           ;INTERRUPT-ON-FLAGERROR (CPU-ERROR)
14 11063 006113         LOOP
15
16 11064 006112 CT11:   SETP2
17 11065 060277         INTDS
18 11066 063577         SKPBZ   CPU
19 11067 006114         EHALT           ;INTERRUPT-ON-FLAGERROR (CPU-ERROR)
20 11070 006113         LOOP
21
22 11071 002124         JMP     @GRORET
23
24 11072 007315 ITRW1:  TRYWA+1
25 11073 007772 MKLOG:  MARKL
```

```
I 0118  .MAIN
01
02                ;BUSY-, DONE- AND DEVICE SELECT- LOGICTESTS.
03
04 11074 054124 BT00:   STA     3,GRORET
05 11075 006112         SETP2
06 11076 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
07 11077 061017         DOA     0,XLPT
08 11100 060117         NIOS    XLPT
09 11101 063417         SKPBN   XLPT
10 11102 006114         EHALT           ;STARTPULSE CAN NOT SET THE LPT-BUSY-
11 11103 006113         LOOP            ;FLOP
12
13 11104 006111 BT01:   SETP1
14 11105 020144         LDA     0,PRTCR ;LOAD A CARRIAGE-RETURN TO PRINTER
15 11106 061017         DOA     0,XLPT
16 11107 060117         NIOS    XLPT
17 11110 063417         SKPBN   XLPT
18 11111 006114         EHALT           ;STARTPULSE CAN'T SET LPT-BUSY-FF
19 11112 006061         CWAIT           ;WAIT TDOS-MS
20 11113 000141         TDOS
21 11114 006113         LOOP
22
23 11115 022756 BT02:   LDA     0,@MKLOG
24 11116 101014         MOV#    0,0,SZR
25 11117 000416         JMP     BT03    ;PRINTER IS LOGABAX
26 11120 006111         SETP1
27 11121 020144         LDA     0,PRTCR ;LOAD A CARRIAGE-RETURN TO PRINTER
28 11122 061017         DOA     0,XLPT
29 11123 060117         NIOS    XLPT
30 11124 060317         NIOP    XLPT    ;WAIT APP. 6-8 US
31 11125 060317         NIOP    XLPT
32 11126 060317         NIOP    XLPT
33 11127 060217         NIOC    XLPT
34 11130 063517         SKPBZ   XLPT
35 11131 006114         EHALT           ;THE LPT-BUSY-FLOP CAN NOT BE CLEARED
36 11132 006061         CWAIT           ;BY CLEAR-PULSE
37 11133 000141         TDOS            ;WAIT TDOS-MS
38 11134 006113         LOOP
39
40 11135 006112 BT03:   SETP2
41 11136 020144         LDA     0,PRTCR ;LOAD A CARRIAGE RETURN TO PRINTER
42 11137 061017         DOA     0,XLPT
43 11140 060117         NIOS    XLPT
44 11141 006061         CWAIT           ;WAIT TDOS MS
45 11142 000141         TDOS
46 11143 063517         SKPBZ   XLPT
47 11144 006114         EHALT           ;BUSY-FLOP ISN'T CLEARED
48 11145 006113         LOOP            ;AFTER PRINTING
49
50 11146 006131 BT04:   CLBUF           ;CLEAR LINE-BUFFER
51 11147 006112         SETP2
52 11150 062677         IORST
53 11151 060417         DIA     0,XLPT
54 11152 101014         MOV#    0,0,SZR
55 11153 006114         EHALT           ;I/O-RESET DID NOT CLEAR LPT-A-REG.
56 11154 006113         LOOP
57
58 11155 006112 BT05:   SETP2
59 11156 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
60 11157 061017         DOA     0,XLPT
61 11160 060017         NIO     XLPT
62 11161 063517         SKPBZ   XLPT
63 11162 006114         EHALT           ;SELECTING THE LPT WITHOUT A STARTPULSE
64 11163 006113         LOOP            ;SHOULDN'T SET THE LPT-BUSY-FLOP.
```

```
| 0119  .MAIN
01
02 11164 006131 BT06:   CLBUF           ;CLEAR LINE BUFFER
03 11165 006112         SETP2
04 11166 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
05 11167 061017         DOA     0,XLPT
06 11170 060100         NIOS    0
07 11171 063517         SKPBZ   XLPT
08 11172 006114         EHALT           ;A STARTPULSE TO DEV. 0 SHOULDN'T SET
09 11173 006113         LOOP            ;THE LPT-BUSY-FLOP
10
11 11174 006112 BT07:   SETP2           ;START DIFF. DEVICES (EXCEPT RECENT LPT)
12 11175 020151         LDA     0,LDSPA
13 11176 061017         DOA     0,XLPT
14 11177 024777         LDA     1,.-1   ;AC1:= I/O-COMMAND TO RECENT LPT
15 11200 030537         LDA     2,DEVMS
16 11201 133400         AND     1,2     ;MASK OUT RECENT DEVICECODE
17 11202 050534         STA     2,LPTCO
18 11203 024535         LDA     1,CON16 ;AC1:= 16-DEC
19 11204 044535         STA     1,COUBT
20 11205 034536         LDA     3,DEVCO ;AC3:= ADDRESS OF TABLE WITH DEVICE-CODES
21 11206 175400 SET06:  INC     3,3
22 11207 025400         LDA     1,0,3   ;AC1:= A DEVICE-CODE
23 11210 132415         SUB#    1,2,SNR ;SKIP IF NOT = RECENT DEVICE-CODE
24 11211 000407         JMP     DECR
25 11212 020530         LDA     0,NIOIN ;NIOS-INSTR TO DEVICE 0
26 11213 123000         ADD     1,0     ;FORM A NIOS-INSTR
27 11214 040401         STA     0,.+1   ;STORE IT .+1
28 11215 060100         NIOS    0
29 11216 063517         SKPBZ   XLPT
30 11217 000404         JMP     ERRBT
31 11220 014521 DECR:   DSZ     COUBT   ;SKIP IF IT WAS LAST DEVICE-CODE
32 11221 000765         JMP     SET06   ;NOT LAST DEVICE-CODE, LOAD NEXT CODE
33 11222 000402         JMP     .+2     ;LAST DEVICE-CODE, EXIT
34 11223 006114 ERRBT:  EHALT           ;THE LPT WAS SELECTED BY AN ADDRESS
35 11224 062677         IORST           ;DIFFERENT FROM THE LPT-DEVICE-CODE
36 11225 006113         LOOP            ;ERRONEOUS DEVICE-CODE IN AC1 - CORRECT
37                                      ;LPT-DEVICE-CODE IN AC2
38
39
40 11226 006131 BT08:   CLBUF           ;CLEAR LINE-BUFFER
41 11227 006112         SETP2
42 11230 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
43 11231 061017         DOA     0,XLPT
44 11232 060117         NIOS    XLPT
45 11233 006061         CWAIT           ;WAIT TBUC MS (= 1 MS)
46 11234 000140         TBUC
47 11235 063517         SKPBZ   XLPT
48 11236 006114         EHALT           ;BUSY-FLOP ISN'T CLEARED AFTER LOAD'ING
49 11237 006113         LOOP
50
51
52 11240 006112 BT09:   SETP2
53 11241 020144         LDA     0,PRTCR ;LOAD A CARRIAGE-RETURN TO PRINTER
54 11242 061017         DOA     0,XLPT
55 11243 060117         NIOS    XLPT
56 11244 006063         TIMSK           ;WAIT FOR LPT-DONE (MAX 1.5 SEC)
57 11245 002734         1500.
58 11246 063617         SKPDN   XLPT
59 11247 000402         JMP     .+2     ;TIME OUT
60 11250 000402         JMP     .+2     ;OK
61 11251 006114         EHALT           ;AFTER EXE. A PRINTCOMMAND THE LPT-DONE-
62                                      ;FLOP WAS NOT SET.
63 11252 006113         LOOP
```

```
I 0120  .MAIN
01
02 11253 006112 BT10:   SETP2
03 11254 006130         SETDO           ;SET LPT-DONE
04 11255 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
05 11256 060217         NIOC    XLPT    ;WITH A PRINT-COMMAND
06 11257 063717         SKPDZ   XLPT
07 11260 006114         EHALT           ;THE CLEARPULSE FAILED TO CLEAR THE
08 11261 006113         LOOP            ;LPT-DONE-FLOP.
09
10 11262 006112 BT11:   SETP2
11 11263 006130         SETDO           ;SET LPT-DONE
12 11264 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
13 11265 062677         IORST           ;WITH A PRINT-COMMAND
14 11266 063717         SKPDZ   XLPT
15 11267 006114         EHALT           ;I/O-RESET FAILED TO CLEAR THE LPT-
16 11270 006113         LOOP            ;DONE-FLOP
17
18 11271 006112 BT12:   SETP2
19 11272 006130         SETDO           ;SET LPT-DONE
20 11273 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
21 11274 020151         LDA     0,LDSPA ;WITH A PRINT-COMMAND
22 11275 061017         DOA     0,XLPT
23 11276 060117         NIOS    XLPT
24 11277 063717         SKPDZ   XLPT
25 11300 006114         EHALT           ;THE STARTPULSE DID NOT CLEAR THE LPT-
26 11301 006113         LOOP            ;DONE-FLOP.
27
28 11302 006131 BT13:   CLBUF           ;CLEAR LINE-BUFFER
29 11303 006112         SETP2
30 11304 020151         LDA     0,LDSPA ;LOAD A CHAR (SPACE) TO LPT
31 11305 061017         DOA     0,XLPT
32 11306 060117         NIOS    XLPT
33 11307 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 10 MS)
34 11310 000012         10.
35 11311 063517         SKPBZ   XLPT
36 11312 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
37 11313 063717         SKPDZ   XLPT
38 11314 006114         EHALT           ;THE DONE-FLOP WAS SET AFTER EXE. A
39 11315 006113         LOOP            ;LOAD-COMMAND.
40
41 11316 006131 BT14:   CLBUF           ;CLEAR LINE-BUFFER
42 11317 006111         SETP1
43 11320 020144         LDA     0,PRTCR ;LOAD A CARRIAGE-RETURN TO PRINTER
44 11321 061017         DOA     0,XLPT
45 11322 060017         NIO     XLPT
46 11323 060117         NIOS    XLPT
47 11324 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 1.5 SEC)
48 11325 002734         1500.
49 11326 063517         SKPBZ   XLPT
50 11327 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
51 11330 063617         SKPDN   XLPT
52 11331 006114         EHALT           ;THE CONTENT OF THE DATAREG. IS
53 11332 006113         LOOP            ;CHANGED BY SELECTING LPT, OR DONE-FF
54 11333 020451         LDA     0,CD800 ;ISN'T SET AFTER A PRINTCOMMAND
55
56 11334 042451         STA     0,@ITRW2;RESTORE WAIT-CONST. IN XLRST-ROUT.
57 11335 002124         JMP     @GRORET
```

```
| 0121  .MAIN
01
02 11336 000000 LPTCO:  0
03 11337 000077 DEVMS:  77
04 11340 000020 CON16:  16.
05 11341 000000 COUBT:  0
06 11342 060100 NIOIN:  060100
07 11343 011360 DEVCO:  COTAB-1
08
09
10 11344 054124 BT15:   STA     3,GRORET
11 11345 006111         SETP1
12 11346 020146         LDA     0,SPA1L ;SPACE ONE LINE - COMMAND
13 11347 061017         DOA     0,XLPT
14 11350 060117         NIOS    XLPT
15 11351 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 1SEC)
16 11352 001750         1000.
17 11353 063517         SKPBZ   XLPT
18 11354 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
19 11355 063617         SKPDN   XLPT
20 11356 006114         EHALT           ;AFTER EXE. A PAPERCOMMAND THE LPT-DONE
21 11357 006113         LOOP            ;FLOP WASN'T SET
22
23 11360 002124         JMP     @GRORET
24
25
26 11361 000007 COTAB:  7
27 11362 000010         10
28 11363 000013         13
29 11364 000015         15
30 11365 000016         16
31 11366 000017         17
32 11367 000020         20
33 11370 000027         27
34 11371 000037         37
35 11372 000040         40
36 11373 000043         43
37 11374 000045         45
38 11375 000046         46
39 11376 000047         47
40 11377 000057         57
41 11400 000060         60
42 11401 000067         67
43
44 11402 000010 MSK12:  000010
45 11403 000400 NOOPR:  000400
46 11404 001440 CD800:  800.
47 11405 007315 ITRW2:  TRYWA+1
```

```
| 0122  .MAIN
01                ;TEST OF INTERRUPT-LOGIC.
02
03                ;EVERY INTERRUPT FROM A DEVICE WILL CAUSE A JUMP @ LOC 0, I.E.
04                ;THE EXECUTION OF THE INTERRUPTED ROUTINE WILL CONTINUE. THE
05                ;PROGRAM WILL ACT AS IF NOTHING HAS HAPPEND, ONLY THE CPU-"BUSY"-
06                ;FLOP IS AFFECTED.
07
08 11406 054124 IT00:   STA     3,GRORET
09 11407 006112         SETP2
10 11410 102000         ADC     0,0       ;AC0:=177777
11 11411 062077         MSKO    0
12 11412 060177         INTEN
13 11413 000401         JMP     .+1
14 11414 063477         SKPBN   CPU
15 11415 006114         EHALT             ;AN INTERRUPT OCCURED EVEN IF EVERY
16 11416 006113         LOOP              ;DEVICE IS MASKED OUT.
17
18 11417 006112 IT01:   SETP2
19 11420 060177         INTEN
20 11421 006130         SETDO             ;SET LPT-DONE
21 11422 006114         EHALT             ;IT'S IMPOSSIBLE TO SET DONE-FF
22 11423 063577         SKPBZ   CPU       ;WITH A PRINTCOMMAND
23 11424 006114         EHALT             ;SET'ING OF LPT-DONE DID NOT CREATE
24 11425 060277         INTDS             ;AN INTERRUPT.
25 11426 006113         LOOP
26
27 11427 006112 IT02:   SETP2
28 11430 020752         LDA     0,MSK12
29 11431 062077         MSKO    0         ;MASK-OUT LPT
30 11432 060177         INTEN
31 11433 006130         SETDO
32 11434 006114         EHALT             ;IT'S IMPOSSIBLE TO SET DONE-FF
33 11435 063477         SKPBN   CPU       ;WITH A PRINT-COMMAND
34 11436 006114         EHALT             ;THE LPT-INTDS-FLOP WAS NOT SET BY THE
35 11437 060277         INTDS             ;MASK-OUT-INSTRUCTION.
36 11440 006113         LOOP
37
38 11441 006112 IT03:   SETP2
39 11442 020740         LDA     0,MSK12
40 11443 062077         MSKO    0         ;MASK-OUT LPT
41 11444 062677         IORST
42 11445 060177         INTEN
43 11446 006130         SETDO
44 11447 006114         EHALT             ;IT'S IMPOSSIBLE TO SET DONE-FF
45 11450 063577         SKPBZ   CPU       ;WITH A PRINT-COMMAND
46 11451 006114         EHALT             ;I/O-RESET FAILED TO CLEAR THE INTDS-
47 11452 060277         INTDS             ;FLOP.
48 11453 006113         LOOP
49
50 11454 006112 IT04:   SETP2
51 11455 060177         INTEN
52 11456 006130         SETDO
53 11457 006114         EHALT             ;IT'S IMPOSSIBLE TO SET DONE-FF
54 11460 061477         INTA    0         ;WITH A PRINT-COMMAND
55 11461 101005         MOV     0,0,SNR
56 11462 006114         EHALT             ;WITH LPT-DONE SET, NO DEVICE-NO. WAS
57 11463 006113         LOOP              ;READ BACK BY THE INTA-INSTRUCTION.
58
59 11464 006112 IT05:   SETP2
60 11465 060177         INTEN
61 11466 006130         SETDO
62 11467 006114         EHALT             ;IT'S IMPOSSIBLE TO SET DONE-FF
63 11470 061477         INTA    0         ;WITH A PRINT-COMMAND
64 11471 024645         LDA     1,LPTCO
65 11472 106414         SUB#    0,1,SZR
66 11473 006114         EHALT             ;THE INTA-INSTR. DID NOT READ THE LPT-
67 11474 006113         LOOP              ;DEVICE-CODE.
```

```
I 0123  .MAIN
01
02 11475 006112 IT06:   SETP2
03 11476 020704         LDA     0,MSK12
04 11477 062077         MSKO    0       ;MASK-OUT LPT
05 11500 060177         INTEN
06 11501 006130         SETDO
07 11502 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
08 11503 061477         INTA    0       ;WITH A PRINT-COMMAND
09 11504 024632         LDA     1,LPTCO
10 11505 106415         SUB#    0,1,SNR
11 11506 006114         EHALT           ;INTA READ LPT-CODE, EVEN IF THE
12 11507 006113         LOOP            ;LPT SHOULD BE MASKED OUT
13
14 11510 006112 IT07:   SETP2
15 11511 060177         INTEN
16 11512 006130         SETDO
17 11513 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
18 11514 060400         DIA     0,0     ;WITH A PRINT-COMMAND
19 11515 101004         MOV     0,0,SZR
20 11516 006114         EHALT           ;THE LPT-DEVICE-CODE WAS PLACED ON THE
21 11517 006113         LOOP            ;DATABUS WITHOUT AN INTA-INSTRUCTION.
22
23 11520 006112 IT08:   SETP2
24 11521 060277         INTDS
25 11522 006130         SETDO
26 11523 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
27 11524 060177         INTEN           ;WITH A PRINT-COMMAND
28 11525 060217         NIOC    XLPT
29 11526 063477         SKPBN   CPU
30 11527 006114         EHALT           ;THE INTR-FLOP WAS NOT CLEARED BY THE
31 11530 006113         LOOP            ;CLEAR PULSE.
32
33 11531 006112 IT09:   SETP2
34 11532 060277         INTDS
35 11533 006130         SETDO
36 11534 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
37 11535 020151         LDA     0,LDSPA ;WITH A PRINT-COMMAND
38 11536 061017         DOA     0,XLPT
39 11537 060177         INTEN
40 11540 060117         NIOS    XLPT
41 11541 063477         SKPBN   CPU
42 11542 006114         EHALT           ;THE INTR-FLOP WASN'T CLEARED BY THE
43 11543 006113         LOOP            ;STARTPULSE
44
45 11544 006131 IT10:   CLBUF           ;CLEAR LINE-BUFFER
46 11545 006112         SETP2
47 11546 060177         INTEN
48 11547 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
49 11550 061017         DOA     0,XLPT
50 11551 060117         NIOS    XLPT
51 11552 006063         TIMSK           ;WAIT FOR LPT-BUSY = 0 (MAX 10MS)
52 11553 000012         10.
53 11554 063517         SKPBZ   XLPT
54 11555 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO 0
55 11556 063477         SKPBN   CPU
56 11557 006114         EHALT           ;AN INTERRUPT OCCURED EVEN IF IT WASN'T
57 11560 060277         INTDS           ;A PRINT- OR PAPERCOMMAND
58 11561 006113         LOOP
59
60 11562 006131         CLBUF           ;CLEAR LINE-BUFFER
61 11563 002124         JMP     @GRORET
```

```
| 0124  .MAIN
01
02                ;TEST OF COMMAND-LOGIC.
03
04
05
06 11564 054124 OT00:   STA     3,GRORET
07 11565 006112         SETP2
08 11566 020615         LDA     0,NOOPR ;A NO-OPERATION-INSTR.
09 11567 061017         DOA     0,XLPT
10 11570 060117         NIOS    XLPT
11 11571 000401         JMP     .+1
12 11572 063517         SKPBZ   XLPT
13 11573 000402         JMP     .+2
14 11574 063717         SKPDZ   XLPT
15 11575 006114         EHALT           ;THE NO-OP-INSTRUCTION WAS NOT EXE. AS
16 11576 006113         LOOP            ;A NO-OP, ERROR IN BUSY- OR STROBE-
17                                      ;INHIBITGATES
18
19 11577 002124         JMP     @GRORET
20
21 11600 054124 OT02:   STA     3,GRORET;(ONLY RC3632/33/34/35/36/41)
22 11601 020143         LDA     0,TS15L ;SPECIFIED TIME TO SPACE 15 LINES
23 11602 105220         MOVZR   0,1
24 11603 044443         STA     1,TS15X ;SET LOWER TIMELIMIT
25 11604 040441         STA     0,TS15U ;SET UPPER TIMELIMIT (=TS15X+TS15U)
26 11605 006110         SETP0
27 11606 006130         SETDO
28 11607 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
29                                      ;WITH A PRINT-COMMAND
30 11610 020434         LDA     0,SP15L ;A "SPACE 15 LINES"-COMMAND
31 11611 061017         DOA     0,XLPT
32 11612 060117         NIOS    XLPT
33 11613 063417         SKPBN   XLPT
34 11614 000410         JMP     OERR2
35 11615 006061         CWAIT           ;WAIT SPECIFIED TIME - 50%
36 11616 011646         TS15X
37 11617 063717         SKPDZ   XLPT
38 11620 000404         JMP     OERR2
39 11621 006061         CWAIT           ;WAIT SPECIFIED TIME + 50% (FROM START)
40 11622 011645         TS15U
41 11623 063617         SKPDN   XLPT
42 11624 006114 OERR2:  EHALT           ;THE COMMAND: SPACE 15 LINES WASN'T EXE.
43 11625 006113         LOOP            ;AS A PAPER-COMMAND, OR IF IT REALY DID
44                                      ;SPACE, THE PAPERFEED-TIMING ISN'T COR-
45                                      ;RECT.
46                                      ;..... NOT  FATAL  ERROR  ........
```

```
I 0125  .MAIN
01
02 11626 006131 OT03:   CLBUF           ;CLEAR LINE-BUFFER
03 11627 006111         SETP1
04 11630 020144         LDA     0,PRTCR ;LOAD A PRINTCOMMAND, CLEAR IT BY IORST
05 11631 061017         DOA     0,XLPT
06 11632 062677         IORST
07 11633 060117         NIOS    XLPT
08 11634 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 1 SEC)
09 11635 001750         1000.
10 11636 063517         SKPBZ   XLPT
11 11637 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
12 11640 063717         SKPDZ   XLPT
13 11641 006114         EHALT           ;THE I/O-RESET DID NOT CLEAR THE COM-
14 11642 006113         LOOP            ;MANDREG.
15
16
17 11643 002124         JMP     @GRORET
18
19
20
21
22
23
24 11644 002517 SP15L:  002517
25 11645 000000 TS15U:  0               ;=TS15L
26 11646 000000 TS15X:  0               ;= 50% OF TS15L
27 11647 000000 COU02:  0
28 11650 000004 CLPT4:  4
29
30
31
32
33 11651 054124 OT06:   STA     3,GRORET;(ONLY RC3637/38/39)
34 11652 020143         LDA     0,TS15L ;SPECIFIED TIME TO SPACE 1 LINE
35 11653 105220         MOVZR   0,1
36 11654 044772         STA     1,TS15X ;LOWER TIME-LIMIT
37 11655 040770         STA     0,TS15U ;UPPER TIME-LIMIT - TS15X
38 11656 006111         SETP1
39 11657 006130         SETDO
40 11660 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
41 11661 020146         LDA     0,SPA1L ;WITH A PRINT-COMMAND
42 11662 061017         DOA     0,XLPT
43 11663 060117         NIOS    XLPT    ;SPACE ONE LINE
44 11664 063417         SKPBN   XLPT
45 11665 000410         JMP     OERR4
46 11666 006061         CWAIT           ;WAIT SPECIFIED TIME - 50%
47 11667 011646         TS15X
48 11670 063717         SKPDZ   XLPT
49 11671 000404         JMP     OERR4
50 11672 006061         CWAIT           ;WAIT SPECIFIED TIME + 50% (FROM START)
51 11673 011645         TS15U
52 11674 063617         SKPDN   XLPT
53 11675 006114 OERR4:  EHALT           ;THE COMMAND: SPACE 1 LINE WASN'T EXE.
54 11676 006113         LOOP            ;AS A PAPERCOMMAND, OR IF IT REALY DID
55                                      ;SPACE, THE PAPERFEED-TIMING ISN'T COR-
56                                      ;RECT.
57                                      ;....... NOT  FATAL  ERROR  .......
58
59 11677 000727         JMP     OT03
```

```
 | 0126  .MAIN
 01
 02                ; TEST OF STATUSREG. (= TESTS WITH OPERATOR-INFOLVEMENT).
 03
 04
 05 11700 054124 ST00:   STA     3,GRORET
 06 11701 006112         SETP2
 07 11702 062677         IORST
 08 11703 102440         SUBO    0,0
 09 11704 060417         DIA     0,XLPT
 10 11705 101014         MOV#    0,0,SZR
 11 11706 006114         EHALT           ;THE STATUSBUFFER-OUTPUT IS NOT ZERO
 12 11707 006113         LOOP            ;EVEN IF IT IS CLEARED BY I/O-RESET
 13
 14 11710 006131 ST01:   CLBUF           ;CLEAR LINE-BUFFER
 15 11711 006112         SETP2
 16 11712 062677         IORST
 17 11713 004504         JSR     CRPRT   ;LOAD A CARRIAGE-RETURN TO PRINTER
 18 11714 006114         EHALT           ;DONE-FF ISN'T SET AFTER A PRINT-
 19                                      ;COMMAND
 20 11715 126440         SUBO    1,1     ;EXPECTED STATUS IN AC1
 21 11716 060417         DIA     0,XLPT
 22 11717 101014         MOV#    0,0,SZR
 23 11720 006114         EHALT           ;A SATUSBIT IS SET,EVEN IF THERE SHOULD
 24 11721 006113         LOOP            ;BE NONE AFTER A PRINT-OPERATION.
 25
 26 11722 002124         JMP     @GRORET
 27
 28
 29 11723 054124 ST02:   STA     3,GRORET
 30 11724 006110         SETP0
 31 11725 004451         JSR     SKPC7   ;EXE. A SKIP TO CHANNEL 7(11) - COMMAND
 32 11726 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO AFTER
 33 11727 063617         SKPDN   XLPT    ;A PAPER-SKIP-COMMAND
 34 11730 006114         EHALT           ;AFTER EXE. A PAPER-SKIP-COMMAND THE
 35 11731 006113         LOOP            ;LPT-DONE-FLOP IS NOT SET.
 36
 37 11732 006110 ST03:   SETP0
 38 11733 004443         JSR     SKPC7
 39 11734 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO AFTER
 40                                      ;A PAPER-SKIP-COMMAND
 41 11735 006115         STATA           ;TEST STATUS
 42 11736 010000         010000
 43 11737 006114         EHALT           ;STATUSBIT 3 WAS NOT SET AFTER SENSING
 44 11740 006113         LOOP            ;A HOLE IN CHANNEL 7 (11)
 45
 46 11741 006110 ST04:   SETP0
 47 11742 004434         JSR     SKPC7
 48 11743 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO AFTER
 49 11744 102440         SUBO    0,0     ;A PAPER-SKIP-COMMAND
 50 11745 060400         DIA     0,0
 51 11746 024427         LDA     1,BIT3M
 52 11747 107404         AND     0,1,SZR
 53 11750 006114         EHALT           ;THE LPT-STATUSREG. IS CONNECTED TO THE
 54 11751 006113         LOOP            ;DATALINES EVEN IF THE SELECTED DEVICE
 55                                      ;IS DEV. 0.
 56
 57 11752 006110 ST05:   SETP0
 58 11753 004423         JSR     SKPC7
 59 11754 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO AFTER
 60 11755 062677         IORST           ;A PAPER-SKIP-COMMAND
 61 11756 006116         STATN           ;TEST STATUS
 62 11757 010000         010000
 63 11760 006114         EHALT           ;I/O-RESET FAILED TO CLEAR STATUSREG.
 64 11761 006113         LOOP
```

```
| 0127  .MAIN
01
02 11762 006110 ST06:   SETP0
03 11763 004413         JSR     SKPC7
04 11764 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO AFTER
05 11765 062677         ICRST           ;A PAPER-SKIP-COMMAND
06 11766 006127         SETBU           ;SET BUSY,WAIT UNTIL IT RETURNS TO 0
07 11767 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
08 11770 060417         DIA     0,XLPT
09 11771 101014         MOV#    0,0,SZR
10 11772 006114         EHALT           ;I/O-RESET DID NOT CLEAR THE CAR-
11                                      ;TAPE SENSED-FLOP.
12 11773 006113         LOOP
13
14 11774 002124         JMP     @GRORET
15
16
17
18
19
20
21
22
23
24
25
26 11775 010000 BIT3M:  010000
27
28 11776 054417 SKPC7:  STA     3,SKPRE
29 11777 010416         ISZ     SKPRE
30 12000 020414         LDA     0,VCHNB ;LOAD HIGHEST CHANNELNO.
31 12001 024145         LDA     1,PRTFF ;LOAD "SKIP TO CHANNEL 0"-COMMAND
32 12002 123000         ADD     1,0     ;AC0:=SKIP TO CHANNEL 7(11)-COMMAND
33 12003 061017         DOA     0,XLPT
34 12004 060117         NIOS    XLPT
35 12005 006063         TIMSK           ;WAIT FOR BUSY=0 (MAX 2 SEC)
36 12006 003720         2000.
37 12007 063517         SKPBZ   XLPT
38 12010 000402         JMP     .+2     ;TIME OUT - RETURN CALL+1
39 12011 002404         JMP     @SKPRE  ;NORMAL OUT - RETURN CALL+2
40 12012 014403         DSZ     SKPRE
41 12013 002402         JMP     @SKPRE
42
43 12014 000000 VCHNB:  0               ;HIGHEST AVAILABLE CHANNEL IN VFT
44 12015 000000 SKPRE:  0
45 12016 000000 CRPRR:  0
46
47
48
49 12017 054777 CRPRT:  STA     3,CRPRR
50 12020 010776         ISZ     CRPRR
51 12021 020144         LDA     0,PRTCR ;EXE. A PRINT-COMMAND
52 12022 061017         DOA     0,XLPT
53 12023 060117         NIOS    XLPT
54 12024 006063         TIMSK           ;WAIT FOR DONE=1 (MAX 1 SEC)
55 12025 001750         1000.
56 12026 063617         SKPDN   XLPT
57 12027 000402         JMP     .+2     ;TIME OUT - RETURN CALL+1
58 12030 002766         JMP     @CRPRR  ;NORMAL OUT - RETURN CALL+2
59 12031 014765         DSZ     CRPRR
60 12032 002764         JMP     @CRPRR
```

```
I 0128  .MAIN
01                ;PRINTERTIMING-TESTS
02
03 12033 054124 TT00:   STA     3,GRORET
04 12034 006112         SETP2
05 12035 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
06 12036 061017         DOA     0,XLPT
07 12037 060117         NIOS    XLPT
08 12040 060300         NIOP    0       ;WAIT APP. 8 - 12 US
09 12041 060300         NIOP    0
10 12042 060300         NIOP    0
11 12043 060300         NIOP    0
12 12044 063517         SKPBZ   XLPT    ;.....  NOT  FATAL  ERROR  .....
13 12045 006114         EHALT           ;LPT-BUSY-FLOP DID NOT RETURN TO
14 12046 006113         LOOP            ;ZERO  APP. 8 US. AFTER A LOADCOMMAND
15                                      ;IS PRT.TYPE AND DRUMSPEED SET CORRECTLY?
16
17
18 12047 006131 TT01:   CLBUF           ;CLEAR LINE-BUFFER
19 12050 006111         SETP1
20 12051 020142         LDA     0,TDOSC
21 12052 040405         STA     0,CONTI
22 12053 020144         LDA     0,PRTCR ;LOAD A CARRIAGE-RETURN TO PRINTER
23 12054 061017         DOA     0,XLPT
24 12055 060117         NIOS    XLPT
25 12056 006063         TIMSK           ;WAIT FOR DONE = 1 (MAX TDOSC SEC)
26 12057 000000 CONTI:  0
27 12060 063617         SKPDN   XLPT    ;.......  NOT  FATAL  ERROR  .........
28 12061 000402         JMP     .+2     ;DONE-FF IS SET CORRECTLY AFTER A PRINT-
29 12062 000403         JMP     .+3     ;COMMAND (IF NO ERRORS UNTIL NOW), BUT
30 12063 020774         LDA     0,CONTI ;IT HAPPENS TOO LATE. THE MAX-SET-TIME
31 12064 006114         EHALT           ;= CONTENT OF AC0
32 12065 006113         LOOP            ;IS PRT.TYPE AND DRUMSPEED SET CORRECTLY?
33
34 12066 002124         JMP     @GRORET
35
36 12067 054124 TT02:   STA     3,GRORET
37 12070 006112         SETP2
38 12071 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
39 12072 061017         DOA     0,XLPT
40 12073 060117         NIOS    XLPT
41 12074 060300         NIOP    0       ;WAIT APP. 6-8 US
42 12075 060300         NIOP    0
43 12076 060300         NIOP    0
44 12077 063517         SKPBZ   XLPT    ;.......  NOT  FATAL  ERROR  .........
45 12100 006114         EHALT           ;LPT-BUSY-FLOP DIDN'T RETURN TO
46 12101 006113         LOOP            ;ZERO APP. 6 US AFTER A LOADCOMMAND
47 12102 000745         JMP     TT01    ;IS PRT.TYPE AND DRUMSPEED SET CORRECTLY?
```

```
I 0129  .MAIN
01
02 12103 054124 TT03:   STA     3,GRORET
03 12104 006112         SETP2
04 12105 020151         LDA     0,LDSPA ;LOAD A SPACE TO PRINTER
05 12106 061017         DOA     0,XLPT
06 12107 060117         NIOS    XLPT
07 12110 060300         NIOP    0       ;WAIT APP. 20 - 28 US
08 12111 060300         NIOP    0
09 12112 060300         NIOP    0
10 12113 060300         NIOP    0
11 12114 060300         NIOP    0
12 12115 060300         NIOP    0
13 12116 060300         NIOP    0
14 12117 060300         NIOP    0
15 12120 060300         NIOP    0
16 12121 060300         NIOP    0
17 12122 063517         SKPBZ   XLPT    ;.....  NOT  FATAL  ERROR  .........
18 12123 006114         EHALT           ;LPT-BUSY-FLOP DIDN'T RETURN TO ZERO
19 12124 006113         LOOP            ;APP. 20 US AFTER A LOADCOMMAND
20                                      ;IS PRT.TYPE AND DRUMSPEED SET CORRECTLY?
21
22 12125 006131         CLBUF           ;CLEAR LINE-BUFFER
23 12126 002124         JMP     @GRORET
```

```
0130  .MAIN
                ;DATAREGISTERTEST.

12127 054124 DT00:   STA     3,GRORET;PRINT COMPLEMENTED CHARACTERS
12130 006474         JSR     @LPDEF  ;DEFINE LINES/PAGE
12131 006131         CLBUF           ;CLEAR LINE-BUFFER
12132 006110         SETP0
12133 020145         LDA     0,PRTFF ;PRINT FF
12134 006466         JSR     @IPRIN
12135 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12136 020150         LDA     0,LIPAG
12137 040457         STA     0,LICOU ;SET LINECOUNTER
12140 020147 CONT0:  LDA     0,CHALI
12141 040454         STA     0,COUD0 ;SET CHARCOUNTER
12142 020455 CONTD:  LDA     0,PRTST
12143 006457         JSR     @IPRIN  ;PRINT CHARCODE 52
12144 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12145 014450         DSZ     COUD0
12146 000402         JMP     .+2
12147 000426         JMP     OUTD0   ;END OF LINE
12150 022455         LDA     0,@ILOM1
12151 101015         MOV#    0,0,SNR
12152 000403         JMP     .+3     ;PRINTER IS NOT LOGABAX
12153 006061         CWAIT           ;WAIT 7 MS TO TEST FIFO-BUFFER
12154 013017         CBELL           ;USED AS CONST
12155 020443         LDA     0,PRTU
12156 006444         JSR     @IPRIN  ;PRINT CHARCODE 125
12157 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12160 014435         DSZ     COUD0
12161 000402         JMP     .+2
12162 000413         JMP     OUTD0   ;END OF LINE
12163 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 5 MS)
12164 000005         5
12165 063517         SKPBZ   XLPT
12166 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
12167 020432         LDA     0,PRCOL ;SEND CHARCODE 72 TO LPT
12170 061017         DOA     0,XLPT
12171 062677         IORST           ;CLEAR  DATAREG.
12172 060117         NIOS    XLPT
12173 014422         DSZ     COUD0
12174 000746         JMP     CONTD   ;NOT END OF LINE
12175 020144 OUTD0:  LDA     0,PRTCR ;END OF LINE
12176 006424         JSR     @IPRIN
12177 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12200 020146         LDA     0,SPA1L
12201 006421         JSR     @IPRIN
12202 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12203 014413         DSZ     LICOU   ;TEST FOR END OF PAGE
12204 000734         JMP     CONT0   ;NOT END OF PAGE
12205 020145         LDA     0,PRTFF ;END OF PAGE
12206 006414         JSR     @IPRIN
12207 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12210 006413         JSR     @PRIEX
12211 006062         WATOP           ;THE PRINTED HAVE TO BE LINES WITH:
12212 020000         020000          ;
12213 006113         LOOP            ;CHARCODE 52,CHARCODE 125,(SPACE),
                                     ;CHARCODE 52,CHARCODE 125  ETC.
                                     ;
                                     ;THAT MEANS IN ASCII: *,U,(SPACE),
                                     ;AND IN RCSTANDARD: 6,K,(SPACE).
                                     ;THE SPACES WILL ONLY BE PRINTED IF A
                                     ;CHAR WITH CHARCODE 0 WILL BE CONVER-
                                     ;TED TO A SPACE EITHER BY THE CONTROL-
                                     ;LER OR BY THE PRINTER ITSELF.
                                     ;IF THE PRINTOUT IS LIKE THIS, THEN
                                     ;THE DATAREG. IS OK. IF THE SPACES
                                     ;(OR THE "MISSING" SPACES) ARE
```

```
| 0131  .MAIN
01
02                                      ;SUBSTITUDED BY CHAR'S WITH CHARCODE
03                                      ;72 (":" OR "&"), THEN THE I/O-RESET-
04                                      ;PULSE FAILED TO CLEAR THE DATAREG.
05
06                                      ;     PROCEED
07 12214 002124          JMP     @GRORET
08
09 12215 000000 COUD0:   0
10 12216 000000 LICOU:   0
11 12217 000052 PRTST:   52
12 12220 000125 PRTU:    125
13 12221 000072 PRCOL:   72
14 12222 012534 IPRIN:   PRIND
15 12223 012615 PRIEX:   EXPRI
16 12224 013531 LPDEF:   DEFLP
17 12225 007772 ILOM1:   MARKL
18
19 12226 054124 DT01:    STA     3,GRORET;PRINT ELONGATED Y'S (RC3638)
20 12227 006775          JSR     @LPDEF  ;DEFINE LINES/PAGE
21 12230 004421          JSR     SETPA
22 12231 006110          SETP0
23 12232 000407          JMP     ELGNL   ;START WITH NL
24 12233 004427 PRTON:   JSR     SETEL   ;SET PRINTER IN ELONGATE MODE
25 12234 020514          LDA     0,CHARY
26 12235 004435          JSR     LD33C   ;LOAD 33 Y'S - RETURN .+1 OR .+2
27 12236 000402          JMP     .+2     ;END OF LINE
28 12237 004433          JSR     LD33C   ;NOT END OF LINE - LOAD 33 Y'S
29 12240 000401          JMP     .+1
30 12241 020144 ELGNL:   LDA     0,PRTCR ;PRINT AND SPACE ONE LINE
31 12242 006760          JSR     @IPRIN
32 12243 006114          EHALT           ;BUSY-FF=1, PRINT IMPOSS.
33 12244 020146          LDA     0,SPA1L
34 12245 006755          JSR     @IPRIN
35 12246 006114          EHALT           ;BUSY-FF=1, PRINT IMPOSS.
36 12247 004437          JSR     ENDTS   ;TEST FOR END OF PAGE
37 12250 000763          JMP     PRTON   ;NOT END
38
39 12251 054476 SETPA:   STA     3,RESET ;PRINT FF AND SET LINES/PAGE=1/2 LIPAG
40 12252 006131          CLBUF           ;CLEAR LINE-BUFFER
41 12253 020145          LDA     0,PRTFF
42 12254 006746          JSR     @IPRIN
43 12255 006114          EHALT           ;BUSY-FF=1, PRINT IMPOSS.
44 12256 020150          LDA     0,LIPAG
45 12257 101220          MOVZR   0,0
46 12260 040472          STA     0,COULI
47 12261 002466          JMP     @RESET
48
49 12262 054465 SETEL:   STA     3,RESET ;SET PRINTER IN ELONGATE-MODE
50 12263 020466          LDA     0,ELONG
51 12264 006736          JSR     @IPRIN
52 12265 006114          EHALT           ;BUSY-FF=1, PRINT IMPOSS.
53 12266 020147          LDA     0,CHALI ;SET CHAR/LINE
54 12267 101220          MOVZR   0,0
55 12270 040463          STA     0,COUCH
56 12271 002456          JMP     @RESET
```

```
| 0132  .MAIN
01
02 12272 054455 LD33C:  STA     3,RESET ;LOAD AC0 TO PRINTER 33 TIMES
03 12273 024461         LDA     1,H0H33
04 12274 044461         STA     1,C0U33
05 12275 006725 LOADC:  JSR     @IPRIN  ;LOAD AC0 TO PRINTER
06 12276 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
07 12277 014454         DSZ     COUCH   ;TEST FOR END OF LINE
08 12300 000402         JMP     .+2
09 12301 002446         JMP     @RESET  ;END OF LINE
10 12302 014453         DSZ     C0U33
11 12303 000772         JMP     LOADC
12 12304 010443         ISZ     RESET   ;AC0 HAS BEEN LOADED 33 TIMES
13 12305 002442         JMP     @RESET
14
15 12306 014444 ENDTS:  DSZ     COULI   ;TEST FOR END OF PAGE
16 12307 001400         JMP     0,3     ;NOT END OF PAGE
17 12310 020145         LDA     0,PRTFF ;END OF PAGE
18 12311 006711         JSR     @IPRIN
19 12312 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
20 12313 006710         JSR     @PRIEX
21 12314 006062         WATOP           ;THE PRINTED HAVE TO BE HALF
22 12315 020000         020000          ;A PAGE WITH LINES FILLED UP
23 12316 006113         LOOP            ;WITH ELONGATED Y'S
24
25 12317 000437         JMP     DT03
26
27
28 12320 054124 DT02:   STA     3,GRORET;PRINT ELONGATED Y'S (RC3639)
29 12321 006703         JSR     @LPDEF  ;DEFINE LINES/PAGE
30 12322 004727         JSR     SETPA
31 12323 006110         SETP0
32 12324 004736 PRT02:  JSR     SETEL   ;SET PRINTER IN ELONGATE MODE
33 12325 020423         LDA     0,CHARY
34 12326 004744         JSR     LD33C   ;LOAD 33 Y'S - RETURN TO .+1 OR .+2
35 12327 000410         JMP     OUT02   ;END OF LINE
36 12330 020424         LDA     0,H0H33 ;NOT END OF LINE
37 12331 024422         LDA     1,COUCH
38 12332 123000         ADD     1,0
39 12333 040420         STA     0,COUCH
40 12334 020151         LDA     0,LDSPA
41 12335 004735         JSR     LD33C   ;LOAD 33 SPACES - RETURN TO .+1 OR .+2
42 12336 000401         JMP     .+1     ;END OF LINE
43 12337 020144 OUT02:  LDA     0,PRTCR ;PRINT AND SPACE 1 LINE
44 12340 006662         JSR     @IPRIN
45 12341 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
46 12342 020146         LDA     0,SPA1L
47 12343 006657         JSR     @IPRIN
48 12344 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
49 12345 004741         JSR     ENDTS   ;TEST FOR END OF PAGE
50 12346 000756         JMP     PRT02   ;NOT END OF PAGE
51
52 12347 000000 RESET:  0
53 12350 000131 CHARY:  131
54 12351 000016 ELONG:  16
55 12352 000000 COULI:  0
56 12353 000000 COUCH:  0
57 12354 000041 H0H33:  33.
58 12355 000000 C0U33:  0
```

```
! 0133  ,MAIN
01
02
03 12356 020145 DT03:   LDA     0,PRTFF ;LOAD PRINTERBUFFER AND DELETE (RC3638/
04 12357 006643         JSR     @IPRIN  ;39)
05 12360 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
06 12361 006111         SETP1
07 12362 020147         LDA     0,CHALI ;AC0:= NO OF CHAR PER LINE
08 12363 040632         STA     0,COUD0
09 12364 020633 COND2:  LDA     0,PRTST
10 12365 006635         JSR     @IPRIN  ;LOAD A LINE WITH CHARCODE 52
11 12366 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
12 12367 014626         DSZ     COUD0   ;TEST FOR END OF LINE
13 12370 000774         JMP     COND2   ;NOT END
14 12371 020152         LDA     0,DELET ;AC0:= A DELETE COMMAND
15 12372 006630         JSR     @IPRIN  ;DELETE THE ENTIRE BUFFER
16 12373 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
17 12374 063417         SKPBN   XLPT
18 12375 000404         JMP     OERR6
19 12376 006061         CWAIT           ;WAIT 4 MS
20 12377 012415         MS004
21 12400 063517         SKPBZ   XLPT
22 12401 006114 OERR6:  EHALT           ;THE DELETE-COMMAND ISN'T EXECUTED
23 12402 020144         LDA     0,PRTCR ;CORRECTLY - THE TIMING IS WRONG
24 12403 006617         JSR     @IPRIN
25 12404 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
26 12405 020146         LDA     0,SPA1L
27 12406 006614         JSR     @IPRIN
28 12407 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
29 12410 006113         LOOP
30 12411 006612         JSR     @PRIEX
31 12412 006062         WATOP           ;IF LPT PRINTED NOTHING, THE DELETE-
32 12413 020000         020000          ;COMMAND IS EXECUTED CORRECTLY
33
34 12414 002124         JMP     @GRORET
35
36
37
38
39
40 12415 000004 MS004:  4
```

```
| 0134  .MAIN
01
02                ;TEST OF PAPERCOMMANDLOGIC
03
04
05 12416 054124 OT04:   STA     3,GRORET;TEST OF PAPERSKIP-LOGIC (RC3632-41)
06 12417 006110         SETP0
07 12420 006130         SETDO
08 12421 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
09 12422 022445         LDA     0,@NOVCH;WITH A PRINT-COMMAND
10 12423 040476         STA     0,COU04
11 12424 010475         ISZ     COU04
12 12425 020145         LDA     0,PRTFF
13 12426 040475         STA     0,SKPCC
14 12427 004462 CONT1:  JSR     FORMC   ;SKIP TO CHANNEL 0,1,2,3,4 .... 7 (11).
15 12430 006114         EHALT           ;BUSY-FF=1, PRINT(FORMC-ROUT) IMPOSS.
16 12431 006063         TIMSK           ;WAIT FOR DONE-FF = 1 (MAX 2 SEC)
17 12432 003720         2000.
18 12433 063617         SKPDN   XLPT
19 12434 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
20 12435 004435         JSR     PRT1L   ;PRINT A LINE WITH *'S
21 12436 014463         DSZ     COU04
22 12437 000770         JMP     CONT1
23 12440 004555         JSR     EXPRI
24 12441 006062         WATCP           ;LOOK THAT THE PRINTED IS:
25 12442 020000         020000          ;SKIP TO CHANNEL 0
26 12443 006113         LOOP            ;**************..... (OR CHARCODE 52)
27                                      ;SKIP TO CHANNEL 1
28                                      ;**************.......
29                                      ;SKIP TO CHANNEL 2
30                                      ;**************......
31                                      ;...ETC...
32
33                                      ;     PROCEED
34
35
36 12444 006110 OT05:   SETP0
37 12445 006130         SETDO
38 12446 006114         EHALT           ;IT'S IMPOSSIBLE TO SET DONE-FF
39                                      ;WITH A PRINT-COMMAND
40 12447 152520         SUBZL   2,2     ;AC2:=1
41 12450 020145         LDA     0,PRTFF ;SKIP TO CHANNEL 0
42 12451 004463 CONT2:  JSR     PRIND   ;PRINT AC0
43 12452 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
44 12453 004417         JSR     PRT1L   ;PRINT A LINE WITH CHARCODE 52
45 12454 020450         LDA     0,SKPNC
46 12455 143000         ADD     2,0
47 12456 151120         MOVZL   2,2
48 12457 024443         LDA     1,HOC16
49 12460 146432         SUBZ#   2,1,SZC ;IS AC2 > 16-DEC.
50 12461 000770         JMP     CONT2   ;NO
51 12462 004533         JSR     EXPRI   ;YES
52 12463 006062         WATCP           ;THE PRINTED OUTPUT HAS TO BE:
53 12464 020000         020000          ;FF
54 12465 006113         LOOP            ;**************...... (OR CHARCODE 52)
55                                      ;SKIP 1 LINE
56                                      ;**************......
57                                      ;SKIP 2 LINES
58                                      ;**************.........
59                                      ;SKIP 4 LINES
60                                      ;**************.....
61                                      ;SKIP 8 LINES
62                                      ;************......
63
64
65                                      ;     PROCEED
66 12466 002124         JMP     @GRORET
```

```
| 0135  .MAIN
01
02 12467 012014 NOVCH:  VCHNB
03 12470 000000 COU00:  0
04 12471 000052 STPRT:  52
05
06 12472 054426 PRT1L:  STA     3,PRT1R ;ROUTINE WILL PRINT 1 LINE WITH *'S
07 12473 020147         LDA     0,CHALI:OR ANOTHER CHAR WITH CHARCODE 52
08 12474 040774         STA     0,COU00
09 12475 014423         DSZ     PRT1R
10 12476 024422         LDA     1,PRT1R ;AC1:= CALL OF PRT1L-ROUT
11 12477 020772 STLDA:  LDA     0,STPRT ;AC0:= CHARCODE 52
12 12500 004434         JSR     PRIND
13 12501 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
14 12502 014766         DSZ     COU00   ;AC1=CALL-LOC OF PRT1L-ROUT.
15 12503 000774         JMP     STLDA
16 12504 020144         LDA     0,PRTCR ;END OF LINE, LOAD A PRINT-COMMAND (CR)
17 12505 004427         JSR     PRIND
18 12506 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
19 12507 010411         ISZ     PRT1R   ;AC1=CALL-LOC OF PRT1L-ROUT.
20 12510 002410         JMP     @PRT1R  ;RETURN CALL+1
21
22
23 12511 054407 FORMC:  STA     3,PRT1R
24 12512 010406         ISZ     PRT1R
25 12513 020410         LDA     0,SKPCC ;A SKIP TO CHANNEL X - COMMAND
26 12514 004420         JSR     PRIND
27 12515 014403         DSZ     PRT1R   ;PRINT ERROR
28 12516 010405         ISZ     SKPCC   ;INCREMENT CHANNEL NO.
29 12517 002401         JMP     @PRT1R
30
31 12520 000000 PRT1R:  0
32 12521 000000 COU04:  0
33 12522 000020 HOC16:  20
34 12523 000000 SKPCC:  0
35 12524 002500 SKPNC:  002500          ;"SKIP-LINES"-COMMAND
36 12525 002000 DOSET:  002000          ;NO-OP, BUT DONE-FF WILL BE SET
37 12526 000000 REEXP:  0
38 12527 000000 BUDOR:  0
39 12530 000000 PRIRT:  0
40 12531 000000 A0SAV:  0
41 12532 000000 A1SAV:  0
42 12533 000000 A2SAV:  0
43
44              ;ROUTINE WILL WAIT FOR LPT-BUSY,AND THEN LOAD AC0 TO PRINTER
45
46 12534 054774 PRIND:  STA     3,PRIRT ;WAIT FOR BUSY=0 AND PRINT AC0
47 12535 010773         ISZ     PRIRT
48 12536 040773         STA     0,A0SAV
49 12537 044773         STA     1,A1SAV
50 12540 050773         STA     2,A2SAV
51 12541 006063         TIMSK
52 12542 011610         5000.
53 12543 063517         SKPBZ   XLPT
54 12544 000402         JMP     .+2     ;TIME OUT
55 12545 000406         JMP     ACOUT   ;NORMAL OUT
56 12546 020763         LDA     0,A0SAV
57 12547 024763         LDA     1,A1SAV
58 12550 030763         LDA     2,A2SAV
59 12551 014757         DSZ     PRIRT
60 12552 002756         JMP     @PRIRT  ;RETURN CALL+1 (ERROR-RETURN)
61 12553 020756 ACOUT:  LDA     0,A0SAV
62 12554 024756         LDA     1,A1SAV
63 12555 030756         LDA     2,A2SAV
64 12556 061017         DOA     0,XLPT
65 12557 060117         NIOS    XLPT
66 12560 002750         JMP     @PRIRT  ;RETURN CALL+2
```

```
| 0136  .MAIN
01
02 12561 054746 SBUSY:  STA     3,BUDOR
03 12562 010745         ISZ     BUDOR
04 12563 020151         LDA     0,LDSPA ;ROUTINE WILL SET LPT-BUSY, AND
05 12564 061017         DOA     0,XLPT  ;WAIT UNTIL IT RETURNS TO ZERO
06 12565 060117         NIOS    XLPT
07 12566 006063         TIMSK           ;WAIT FOR BUSY-FF=0 (MAX 5 MS)
08 12567 000005         5
09 12570 063517         SKPBZ   XLPT
10 12571 000402         JMP     .+2     ;TIME OUT
11 12572 002735         JMP     @BUDOR  ;OK - RETURN CALL + 2
12 12573 014734         DSZ     BUDOR
13 12574 002733         JMP     @BUDOR  ;RETURN CALL + 1
14
15
16 12575 054732 SDONE:  STA     3,BUDOR ;ROUTINE WILL SET LPT-DONE-FF
17 12576 010731         ISZ     BUDOR
18 12577 006063         TIMSK           ;WAIT FOR BUSY = 0 (MAX 300 MS)
19 12600 000454         300.
20 12601 063517         SKPBZ   XLPT
21 12602 000401         JMP     .+1
22 12603 020722         LDA     0,DOSET
23 12604 061017         DOA     0,XLPT
24 12605 060117         NIOS    XLPT
25 12606 006063         TIMSK           ;WAIT FOR DONE-FF=1 (MAX 2 SEC)
26 12607 003720         2000.
27 12610 063617         SKPDN   XLPT
28 12611 000402         JMP     .+2     ;TIME-OUT
29 12612 002715         JMP     @BUDOR  ;OK - RETURN CALL+2
30 12613 014714         DSZ     BUDOR
31 12614 002713         JMP     @BUDOR  ;RETURN CALL+1
32
33
34 12615 054711 EXPRI:  STA     3,REEXP
35 12616 006043         CCRLF
36 12617 006046         CDICL
37 12620 006040         CMESS
38 12621 015257         EXA00           ;"EXAMINE PRINTOUT, AND COMPARE
39 12622 006044         CDISP           ;WITH PC: "
40 12623 015314         EXA01           ;"EXAM. PRINTOUT"
41 12624 006047         CDATT
42 12625 006046         CDICL
43 12626 006044         CDISP
44 12627 015324         EXA02           ;"COMP. PC: "
45 12630 024676         LDA     1,REEXP
46 12631 006052         CTOCT           ;PRINT THE LOCATION TO COMPARE WITH
47 12632 006056         CDOCT
48 12633 006050         CHAAT
49 12634 002672         JMP     @REEXP
50
51
52 12635 000000 FCLER:  0
53 12636 000000 CLERM:  0
54 12637 000000 CLRET:  0
55 12640 000004 MSEC4:  4
56 12641 012472 I1LPR:  PRT1L
57 12642 007773 ICEMK:  MARKC
58 12643 007772 ILOMK:  MARKL
```

```
I 0137  .MAIN
01
02                ; CLEBU-ROUTINE (CALLED BY CLBUF) WILL CLEAR THE LINE-BUFFER
03                ; IF THE PRINTER IS A CENTRONICS OR LOGABAX. IT IS NESSESCARY
04                ; TO DO THIS BEFORE THE BUFFER BECOMES FULL, (WHICH WOULD HAP-
05                ; PEN IF SOME TESTLOOPS FILLS IT UP WITH E.G. SPACES 2*100 TI-
06                ; MES), BECOURSE IT WOULD RESULT IN AN OVERFLOW-ERROR.
07
08 12644 054773 CLEBU:  STA     3,CLRET
09 12645 040664         STA     0,A0SAV ;SAVE AC'S
10 12646 044664         STA     1,A1SAV
11 12647 050664         STA     2,A2SAV
12 12650 102440         SUBC    0,0
13 12651 040765         STA     0,CLERM
14 12652 014765         DSZ     CLRET   ;DECREMENT CLRET TO IDENTIFY CALL WHEN
15 12653 006110         SETP0           ;ERROR IN CLEBU ROUT.
16 12654 024762         LDA     1,CLERM
17 12655 125014         MOV#    1,1,SZR ;SKIP IF FIRST RUN IN CURRENT LOOP
18 12656 000437         JMP     CLBOU
19 12657 022763         LDA     0,@ICEMK
20 12660 101015         MOV#    0,0,SNR
21 12661 000414         JMP     CLLOG   ;PRINTER IS DP
22 12662 022761         LDA     0,@ILOMK
23 12663 101014         MOV#    0,0,SZR
24 12664 000411         JMP     CLLOG   ;PRINTER IS LOGABAX
25 12665 020152         LDA     0,DELET ;PRINTER IS CENTRONICS
26 12666 024751         LDA     1,CLRET
27 12667 004645         JSR     PRIND   ;PRINT A DELETE-COMMAND
28 12670 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
29 12671 006061         CWAIT           ;AC1=CALL-LOC. OF CLEBU-ROUTINE
30 12672 012640         MSEC4
31 12673 063517         SKPBZ   XLPT
32 12674 000412         JMP     CLERR
33 12675 020144 CLLOG:  LDA     0,PRTCR
34 12676 024741         LDA     1,CLRET
35 12677 004635         JSR     PRIND   ;PRINT CARRIAGE-RETURN TO DELETE BUFFER
36 12700 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
37 12701 006063         TIMSK           ;AC1=CALL-LOC. OF CLEBU-ROUTINE
38 12702 004704         2500.
39 12703 063517         SKPBZ   XLPT
40 12704 000402         JMP     CLERR   ;TIME-OUT
41 12705 000410         JMP     CLBOU   ;NORMAL OUT
42 12706 020727 CLERR:  LDA     0,FCLER
43 12707 101014         MOV#    0,0,SZR
44 12710 000414         JMP     MANCL   ;NOT FIRST CLEAR-ERROR
45 12711 102000         ADC     0,0
46 12712 040723         STA     0,FCLER
47 12713 024724         LDA     1,CLRET
48 12714 006114         EHALT           ;THE DELETE-TRY WASN'T EXECUTED
49 12715 062677 CLBOU:  IORST           ;CORRECTLY (BUSY-FF DOESN'T RETURN
50 12716 006113         LOOP            ;TO ZERO). THE LOCATION OF THE CALL
51 12717 024613         LDA     1,A1SAV ;OF CLEBU-ROUTINE IS PLACED IN AC1.
52 12720 030613         LDA     2,A2SAV ;IF THIS CALL RESIDES IN THE DIAG-
53 12721 020610         LDA     0,A0SAV ;NOSTICTEST, THEN CONTINUE TO GET FUR-
54 12722 010715         ISZ     CLRET   ;THER INFORMATION. IF NOT, THEN START
55 12723 002714         JMP     @CLRET  ;THE DIAGNOSTICTEST FOR DETECTION OF
56                                      ;THE ERROR.
57
58 12724 006043 MANCL:  CCRLF
59 12725 006046         CDICL
60 12726 006040         CMESS
61 12727 015303         MANSE           ;"OFF-LINE/ON-LINE"
62 12730 006044         CDISP
63 12731 015303         MANSE
64 12732 006062         WAITP
65 12733 020000         020000
66 12734 014702         DSZ     CLERM
67 12735 000760         JMP     CLBOU
```

```
| 0138  .MAIN
01
02 12736 054124 OT07:   STA     3,GRORET;TEST OF PAPERSKIP-LOGIC (RC3637/38/39)
03 12737 006110         SETP0
04 12740 020462         LDA     0,CHNNO
05 12741 040462         STA     0,CHNCO
06 12742 020145         LDA     0,PRTFF
07 12743 006456         JSR     @IPRI2
08 12744 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
09 12745 006674 SKPON:  JSR     @I1LPR  ;PRINT A LINE WITH STARS
10 12746 020146         LDA     0,SPA1L
11 12747 006452         JSR     @IPRI2  ;SKIP ONE LINE
12 12750 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
13 12751 006670         JSR     @I1LPR
14 12752 020444         LDA     0,SKPC2
15 12753 006446         JSR     @IPRI2  ;VERTICAL TAB
16 12754 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
17 12755 014446         DSZ     CHNCO
18 12756 000767         JMP     SKPON
19 12757 006662         JSR     @I1LPR
20 12760 020145         LDA     0,PRTFF
21 12761 006440         JSR     @IPRI2
22 12762 006114         EHALT           ;BUSY-FF=1, PRINT IMPOSS.
23 12763 004632         JSR     EXPRI
24 12764 006062         WATOP           ;THE PRINTED OUTPUT HAS TO BE:
25 12765 020000         020000          ;FORMFEED
26 12766 006113         LOOP            ;**************...(OR CHARCODE 52)
27                                      ; LF
28                                      ;**************.....
29                                      ; VT
30                                      ;****************...
31                                      ; LF
32                                      ; ..... ETC ......
33                                      ;
34                                      ;FORMFEED
35 12767 002124         JMP     @GRORET
36                                      ;   PROCEED
37
38 12770 054124 OT08:   STA     3,GRORET
39 12771 006110         SETP0           ;TEST OF BELL-COMMAND (RC3638/39)
40 12772 020425         LDA     0,CBELL
41 12773 006426         JSR     @IPRI2  ;"PRINT" A BELL COMMAND
42 12774 006114         EHALT           ;BUSY-FF= 1, PRINT IMPOSSIBLE
43 12775 063417         SKPBN   XLPT
44 12776 000404         JMP     ERR08
45 12777 006061         CWAIT           ;WAIT 3.0 SEC
46 13000 013020         S3.0
47 13001 063517         SKPBZ   XLPT
48 13002 006114 ERR08:  EHALT           ;BELL DIDN'T RING FOR 2 SEC. OR
49 13003 006043         CCRLF           ;LPT-BUSY ISN'T RESET AFTER
50 13004 006046         CDICL           ;RINGING
51 13005 006040         CMESS
52 13006 014742         BELAC           ;"DID BELL RING ? "
53 13007 006044         CDISP
54 13010 014742         BELAC
55 13011 006050         CHAAT
56 13012 006062         WATOP
57 13013 020000         020000
58 13014 006113         LOOP
59 13015 002124         JMP     @GRORET
60
61 13016 000013 SKPC2:  13              ;VERTICAL TAB COMMAND
62 13017 000007 CBELL:  7               ;BELL-COMMAND
63 13020 005670 S3.0:   3000.
64 13021 012534 IPRI2:  PRIND
65 13022 000013 CHNNO:  11.
66 13023 000000 CHNCO:  0
```

```
I 0139  .MAIN
01
02                ; TEST OF STATUSLOGIC (WITH OPERATOR-INVOLVEMENT)
03
04
05 13024 054124 WT00:   STA     3,GRORET
06 13025 006043         CCRLF
07 13026 006046         CDICL
08 13027 006040         CMESS
09 13030 014733         DRUMG           ;"OPEN DRUMGATE"
10 13031 006044         CDISP
11 13032 014733         DRUMG
12 13033 006050         CHAAT
13 13034 006062         WATOP           ;   PROCEED
14 13035 020000         020000
15
16 13036 006112         SETP2
17 13037 020144         LDA     0,PRTCR ;LOAD A PRINT-COMMAND TO PRINTER
18 13040 061017         DOA     0,XLPT
19 13041 060117         NIOS    XLPT
20 13042 063517         SKPBZ   XLPT
21 13043 000402         JMP     .+2
22 13044 063617         SKPDN   XLPT
23 13045 006114         EHALT           ;THE "NOT READY"-STATUS DID NOT PREVENT
24 13046 006113         LOOP            ;STROBE OR SET'ING OF LPT-BUSY.
25
26 13047 006112 WT01:   SETP2
27 13050 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
28 13051 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
29 13052 006115         STATA           ;TEST STATUS
30 13053 040000         040000
31 13054 006114         EHALT           ;THE "NOT-READY"-STATUSBIT WAS NOT
32 13055 006113         LOOP            ;SET IN THE STATUS BUFFER
33
34 13056 006112 WT02:   SETP2
35 13057 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
36 13060 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
37 13061 062677         IORST
38 13062 006116         STATN           ;TEST STATUS
39 13063 040000         040000
40 13064 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "NOT-READY"-BIT
41 13065 006113         LOOP
42
43 13066 006112 WT03:   SETP2
44 13067 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
45 13070 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
46 13071 006115         STATA           ;TEST STATUS
47 13072 000001         000001
48 13073 006114         EHALT           ;THE NOT AVAIL.-STATUSBIT WAS NOT
49 13074 006113         LOOP            ;SET IN THE STATUSBUFFER.
50
51 13075 006112 WT04:   SETP2
52 13076 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
53 13077 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
54 13100 062677         IORST
55 13101 006116         STATN           ;TEST STATUS
56 13102 000001         000001
57 13103 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "NOT-AVAIL."-BIT
58 13104 006113         LOOP
```

```
| 0140  ,MAIN
01
02 13105 006043 WT05:   CCRLF
03 13106 006046         CDICL
04 13107 006040         CMESS
05 13110 015010         MLOCL           ;"CLOSE DRUMGATE - SET LPT IN LOCAL-
06 13111 006044         CDISP           ; STATE"
07 13112 015034         DLOCL
08 13113 006050         CHAAT
09 13114 006062         WATOP           ;   PROCEED
10 13115 020000         020000
11
12 13116 006111         SETP1
13 13117 020144         LDA     0,PRTCR ;LOAD A PRINT-COMMAND TO PRINTER
14 13120 061017         DOA     0,XLPT
15 13121 060117         NIOS    XLPT
16 13122 063517         SKPBZ   XLPT
17 13123 000402         JMP     .+2
18 13124 063617         SKPDN   XLPT
19 13125 006114         EHALT           ;THE "OFF-LINE"-STATUS DID NOT PREVENT
20 13126 006113         LOOP            ;STROBE OR SET'ING OF LPT-BUSY
21
22 13127 006112 WT06:   SETP2
23 13130 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
24 13131 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
25 13132 006115         STATA           ;TEST STATUS
26 13133 000200         000200
27 13134 006114         EHALT           ;THE"OFF-LINE"-STATUSBIT WAS NOT SET IN
28 13135 006113         LOOP            ;THE STATUSREGISTER.
29
30 13136 006112 WT07:   SETP2
31 13137 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
32 13140 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
33 13141 062677         IORST
34 13142 006116         STATN           ;TEST STATUS
35 13143 000200         000200
36 13144 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "OFF-LINE"-BIT
37 13145 006113         LOOP
38
39 13146 006112 WT08:   SETP2
40 13147 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
41 13150 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
42 13151 006115         STATA           ;TEST STATUS
43 13152 000001         000001
44 13153 006114         EHALT           ;THE "NOT AVAIL."-STATUSBIT WAS NOT
45 13154 006113         LOOP            ;SET IN THE STATUSREGISTER.
46
47 13155 002124         JMP     @GRORET
48
49 13156 054124 WT09:   STA     3,GRORET
50 13157 006043         CCRLF
51 13160 006046         CDICL
52 13161 006040         CMESS
53 13162 015131         MEOFP           ;"SET LPT ON-LINE -  REMOVE PAPER EXCEPT
54 13163 006044         CDISP           ;ONE FORM"
55 13164 015161         DEOFP
56 13165 006050         CHAAT
57 13166 006062         WATOP           ;   PROCEED
58 13167 020000         020000
59
60 13170 006112         SETP2
61 13171 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
62 13172 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
63 13173 006115         STATA           ;TEST STATUS
64 13174 000020         000020
65 13175 006114         EHALT           ;THE "END OF PAPER"-STATUSBIT WAS NOT
66 13176 006113         LOOP            ;SET IN THE STATUSREGISTER
```

```
| 0141   .MAIN
01
02 13177 006112 WT10:    SETP2
03 13200 006127          SETBU            ;SET BUSY,WAIT UNTIL IT RETURNS TO 0
04 13201 006114          EHALT            ;BUSY-FF DOESN'T RETURN TO ZERO
05 13202 062677          IORST
06 13203 006116          STATN            ;TEST STATUS
07 13204 000020          000020
08 13205 006114          EHALT            ;I/O-RESET DIDN'T CLEAR "END OF PAPER"-
09 13206 006113          LOOP             ;BIT
10
11 13207 006110 WT11:    SETP0
12 13210 006127          SETBU            ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
13 13211 006114          EHALT            ;BUSY-FF DOESN'T RETURN TO ZERO
14 13212 020146          LDA      0,SPA1L ;MOVE PAPER ONE LINE
15 13213 006431          JSR      @IPRI1
16 13214 006114          EHALT            ;BUSY-FF=1, PRINT IMPOSS.
17 13215 006063          TIMSK            ;WAIT FOR BUSY-FF=0 (MAX 1 SEC)
18 13216 001750          1000.
19 13217 063517          SKPBZ    XLPT
20 13220 006114          EHALT            ;BUSY-FF DOESN'T RETURN TO ZERO
21 13221 006113          LOOP
22
23 13222 006110 WT12:    SETP0
24 13223 006127          SETBU            ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
25 13224 006114          EHALT            ;BUSY-FF DOESN'T RETURN TO ZERO
26 13225 020145          LDA      0,PRTFF
27 13226 006416          JSR      @IPRI1  ;SKIP TO TOP OF FORM
28 13227 006114          EHALT            ;BUSY-FF=1, PRINT IMPOSS.
29 13230 006063          TIMSK            ;WAIT FOR BUSY-FF=0 (MAX 3 SEC)
30 13231 005670          3000.
31 13232 063517          SKPBZ    XLPT
32 13233 006114          EHALT            ;BUSY-FF DOESN'T RETURN TO ZERO
33 13234 020132          LDA      0,MRK47
34 13235 101014          MOV#     0,0,SZR ;SKIP IF NOT RC3643
35 13236 000404          JMP      .+4
36 13237 006115          STATA            ;TEST STATUS
37 13240 100000          100000
38 13241 006114          EHALT            ;THE "PAPER FAULT"-BIT ISN'T SET EVEN
39 13242 006113          LOOP             ;IF IT SHOULD BE AT NEXT TOP OF
40                                        ;FORM
41 13243 002124          JMP      @GRORET
42
43 13244 012534 IPRI1:   PRIND
44
45 13245 054124 WT13:    STA      3,GRORET
46 13246 020132          LDA      0,MRK47
47 13247 101014          MOV#     0,0,SZR
48 13250 000431          JMP      USELP
49 13251 006043          CCRLF
50 13252 006046          CDICL
51 13253 006040          CMESS
52 13254 015172          PAPFA            ;"REMOVE PAPER"
53 13255 006044          CDISP
54 13256 015172          PAPFA
55 13257 006050          CHAAT
56 13260 006062          WATOP            ;     PROCEED
57 13261 020000          020000
```

```
| 0142  .MAIN
01
02 13262 006112           SETP2
03 13263 006127           SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
04 13264 006114           EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
05 13265 006115           STATA           ;TEST STATUS
06 13266 100000           100000
07 13267 006114           EHALT           ;THE "PAPERFAULT"-STATUSBIT WAS NOT
08 13270 006113           LOOP            ;SET IN THE STATUSREGISTER.
09 13271 006112 WT14:     SETP2
10 13272 006127           SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
11 13273 006114           EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
12 13274 062677           IORST
13 13275 006116           STATN           ;TEST STATUS
14 13276 100000           100000
15 13277 006114           EHALT           ;I/O-RESET DIDN'T CLEAR "PAPERFAULT"-BIT
16 13300 006113           LOOP
17
18 13301 006043 USELP:    CCRLF
19 13302 006046           CDICL
20 13303 006040           CMESS
21 13304 014753           MUSEL           ;"MAKE LPT USEABLE"
22 13305 006044           CDISP
23 13306 014753           MUSEL
24 13307 006050           CHAAT
25 13310 006062           WATOP           ;     PROCEED
26 13311 020000           020000
27 13312 002124           JMP     @GRORET
28
29 13313 054124 WT15:     STA     3,GRORET
30 13314 006043           CCRLF
31 13315 006046           CDICL
32 13316 006040           CMESS
33 13317 015332           SETLO           ;"DESELECT PRINTER"
34 13320 006044           CDISP
35 13321 015332           SETLO
36 13322 006050           CHAAT
37 13323 006062           WATOP           ;       PROCEED
38 13324 020000           020000
39
40 13325 006112           SETP2
41 13326 020144           LDA     0,PRTCR ;LOAD A PRINT-COMMAND TO PRINTER
42 13327 061017           DOA     0,XLPT
43 13330 060117           NIOS    XLPT
44 13331 063517           SKPBZ   XLPT
45 13332 000402           JMP     .+2
46 13333 063617           SKPDN   XLPT
47 13334 006114           EHALT           ;THE "OFF-LINE" AND "NOT READY"-STATUS-
48 13335 006113           LOOP            ;DID NOT PREVENT STROBE OR SET'ING
49                                        ;OF LPT-BUSY
50 13336 006112 WT16:     SETP2
51 13337 006127           SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
52 13340 006114           EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
53 13341 006115           STATA           ;TEST STATUS
54 13342 040000           040000
55 13343 006114           EHALT           ;THE "OFF-LINE"-STATUSBIT WAS NOT
56 13344 006113           LOOP            ;SET IN THE STATUSREG.
```

```
| 0143  .MAIN
01
02 13345 022550 WT17:   LDA     0,@ILOGM
03 13346 101014         MOV#    0,0,SZR
04 13347 000410         JMP     WT18    ;PRINTER IS LOGBX.(HAS NO "NOT-READY"-
05 13350 006112         SETP2           ;STATUS).
06 13351 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
07 13352 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
08 13353 006115         STATA           ;TEST STATUS
09 13354 000100         000100
10 13355 006114         EHALT           ;THE "NOT READY"-STATUSBIT WAS NOT
11 13356 006113         LOOP            ;SET IN THE STATUSREG.
12
13 13357 006112 WT18:   SETP2
14 13360 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
15 13361 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
16 13362 006115         STATA           ;TEST STATUS
17 13363 000001         000001
18 13364 006114         EHALT           ;THE "NOT AVAIL."-STATUSBIT WAS NOT
19 13365 006113         LOOP            ;SET IN THE STATUSREG.
20 13366 006112 WT19:   SETP2
21 13367 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
22 13370 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
23 13371 062677         IORST
24 13372 006116         STATN           ;TEST STATUS
25 13373 040000         040000
26 13374 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "OFF-LINE"-BIT
27 13375 006113         LOOP
28
29 13376 006112 WT20:   SETP2
30 13377 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
31 13400 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
32 13401 062677         IORST
33 13402 006116         STATN           ;TEST STATUS
34 13403 000100         000100
35 13404 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "NOT-READY"-BIT
36 13405 006113         LOOP
37
38 13406 006112 WT21:   SETP2
39 13407 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
40 13410 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
41 13411 062677         IORST
42 13412 006116         STATN           ;TEST STATUS
43 13413 000001         000001
44 13414 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "NOT AVAIL."-BIT
45 13415 006113         LOOP
46
47 13416 006043 WT22:   CCRLF
48 13417 006046         CDICL
49 13420 006040         CMESS
50 13421 014764         MREPA           ;"SELECT LPT - REMOVE PAPER"
51 13422 006044         CDISP
52 13423 015343         DREPA           ;"SEL.PR-REMOV.PAP"
53 13424 006050         CHAAT
54 13425 006062         WATOP           ;   PROCEED
55 13426 020000         020000
56
57 13427 006112         SETP2
58 13430 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
59 13431 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
60 13432 006115         STATA           ;TEST STATUS
61 13433 000020         000020
62 13434 006114         EHALT           ;THE "END OF PAPER"-STATUSBIT WAS NOT
63 13435 006113         LOOP            ;SET IN THE STATUSREG.
```

```
I 0144  .MAIN
01
02 13436 006112 WT23:   SETP2
03 13437 006127         SETBU           ;SET BUSY, WAIT UNTIL IT RETURNS TO 0
04 13440 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
05 13441 062677         IORST
06 13442 006116         STATN           ;TEST STATUS
07 13443 000020         000020
08 13444 006114         EHALT           ;I/O-RESET DIDN'T CLEAR "END OF PAPER"-
09 13445 006113         LOOP            ;BIT
10
11 13446 006043         CCRLF
12 13447 006046         CDICL
13 13450 006040         CMESS
14 13451 015001         INSPA           ;"REPLACE PAPER"
15 13452 006044         CDISP
16 13453 015001         INSPA
17 13454 006050         CHAAT
18 13455 006062         WATOP           ;   PROCEED
19 13456 020000         020000
20 13457 006131         CLBUF           ;CLEAR LINE-BUFFER
21
22 13460 002124         JMP     @GRORET
23
24 13461 054124 WT24:   STA     3,GRORET
25 13462 006110         SETP0
26 13463 020431         LDA     0,SELOF ;SELECT PRINTER OFF
27 13464 061017         DOA     0,XLPT
28 13465 060117         NIOS    XLPT
29 13466 006061         CWAIT           ;WAIT 1 MS
30 13467 000140         TBUC
31 13470 006063         TIMSK           ;WAIT FOR BUSY-FF = 0 (MAX 5MS)
32 13471 000005         5
33 13472 063517         SKPBZ   XLPT
34 13473 006114         EHALT           ;BUSY-FF DOESN'T RETURN TO ZERO
35 13474 006061         CWAIT           ;WAIT 0.3 SEC
36 13475 013516         S0.3
37 13476 006115         STATA           ;TEST STATUS
38 13477 040000         040000
39 13500 006114         EHALT           ;THE "SELECT-OFF"-COMMAND WASN'T
40                                      ;EXECUTED CORRECTLY
41
42 13501 006043         CCRLF
43 13502 006046         CDICL
44 13503 006040         CMESS
45 13504 015477         SETOL           ;"SET LPT ON-LINE"
46 13505 006044         CDISP
47 13506 015477         SETOL
48 13507 006050         CHAAT
49 13510 006062         WATOP
50 13511 020000         020000
51
52 13512 006113         LOOP
53
54
55 13513 002124         JMP     @GRORET
56
57 13514 000023 SELOF:  23              ;A "SELECT OFF"-COMMAND
58 13515 007772 ILOGM:  MARKL
59 13516 000454 S0.3:   300.
60
61
```

```
| 0145  .MAIN
01
02                ; LPTPART3   PRINTER - EXERCISERPROGRAM
03
04                ; CHARACTERPRINTING - AND MOTIONROUTINES
05
06 13517 007773 IMRKC:  MARKC
07 13520 000000 LPMRK:  0
08 13521 000076 INILI:  76
09 13522 000000 DEFLR:  0
10 13523 013746 COMPR:  PRCOM
11 13524 000005 HOH05:  5
12 13525 014052 TESTS:  STTST
13 13526 000000 LCOUN:  0
14 13527 000024 WMS20:  20.
15 13530 000132 MAXLI:  90.
16
17
18                ;ROUTINE TO COUNT OR CALCULATE NO. OF LINES PER PAGE
19
20
21 13531 054771 DEFLP:  STA     3,DEFLR ;CALCULATE OR DEFINE LINES/PAGE
22 13532 020766         LDA     0,LPMRK
23 13533 101014         MOV#    0,0,SZR ;IS LINES/PAGE DEFINED BEFORE ?
24 13534 001400         JMP     0,3     ;YES
25 13535 014763         DSZ     LPMRK   ;NO
26 13536 022761         LDA     0,@IMRKC
27 13537 101014         MOV#    0,0,SZR ;IS PRINTER CENTRONICS OR LOGABAX ?
28 13540 000427         JMP     LIDEF   ;YES - DEFINE NO. OF LINES PER PAGE
29 13541 102440         SUBO    0,0     ;NO
30 13542 040764         STA     0,LCOUN
31 13543 020145         LDA     0,PRTFF ;PRINT FORMFEED
32 13544 006757         JSR     @COMPR
33 13545 006061 SP1LI:  CWAIT
34 13546 013527         WMS20
35 13547 020146         LDA     0,SPA1L ;COUNT LINES UNTIL END OF PAGE
36 13550 006753         JSR     @COMPR
37 13551 024757         LDA     1,MAXLI
38 13552 010754         ISZ     LCOUN
39 13553 020753         LDA     0,LCOUN ;NO. OF LINES UNTIL NOW
40 13554 106433         SUBZ#   0,1,SNC ;SKIP IF LCOUN < MAXLI
41 13555 000412         JMP     LIDEF   ;GIVE UP  -  DEFINE LINES/PAGE
42 13556 006747         JSR     @TESTS  ;WAIT FOR BUSY = 0
43 13557 006115         STATA           ;TEST FOR CARRIAGE OVERFLOW (=E.OF PAGE)
44 13560 010000         010000
45 13561 000764         JMP     SP1LI   ;NOT END OF PAGE
46 13562 020744         LDA     0,LCOUN ;END OF PAGE
47 13563 034741         LDA     3,HOH05
48 13564 162420         SUBZ    3,0     ;MAKE ROOM FOR HEADS
49 13565 040150         STA     0,LIPAG ;SET LINES/PAGE
50 13566 002734         JMP     @DEFLR
51
52
53
54 13567 020732 LIDEF:  LDA     0,INILI ;AC0:= 76
55 13570 040150         STA     0,LIPAG ;DEFINE NO. OF LINES PER PAGE
56 13571 002731         JMP     @DEFLR  ;RETURN TO CALL+1 OF DEFLP-ROUT.
```

```
| 0146  .MAIN
01
02 13572 054430 LIPRT:  STA     3,LIRET ;PRINT 2 PAGES WITH INPUTED CHAR.
03 13573 004736          JSR     DEFLP
04 13574 006043 RELPR:  CCRLF
05 13575 006046          CDICL
06 13576 006040          CMESS
07 13577 015075          INCHA           ; "INPUT CHARACTER "
08 13600 006044          CDISP
09 13601 015470          DINCH
10 13602 006050          CHAAT
11 13603 022416          LDA     0,@INMOD
12 13604 101014          MOV#    0,0,SZR ;IS INPUT ASCII OR OCTAL ?
13 13605 000417          JMP     OCHIN   ;OCTAL
14 13606 006106          CGTSC           ;ASCII
15 13607 000765          JMP     RELPR
16 13610 000764 ASKREP: JMP     RELPR   ;ILLEGAL INPUT
17 13611 020075          LDA     0,DIGIN
18 13612 040411          STA     0,CHAIN
19 13613 004415          JSR     PRTLI   ;PRINT 2 PAGES WITH THE WANTED CHAR.
20 13614 020145          LDA     0,PRTFF ;PRINT FORM-FEED
21 13615 004531          JSR     PRCOM
22 13616 004471          JSR     REPASK  ;IS REPEATING OF CHAR.
23                                       ;PRINTING PGR. WANTED ?
24 13617 000755          JMP     RELPR   ;YES
25 13620 002402          JMP     @LIRET  ;NO - RETURN
26
27 13621 010351 INMOD:  INMRK
28 13622 000000 LIRET:  0
29 13623 000000 CHAIN:  0
30
31 13624 006104 OCHIN:  CGTOK
32 13625 000763          JMP     ASKREP
33 13626 000762          JMP     ASKREP  ;ILLEGAL INPUT
34 13627 000762          JMP     ASKREP+1;NORMAL RETURN
```

```
| 0147  .MAIN
01
02 13630 054453 PRTLI:  STA     3,PRTRE
03 13631 006131         CLBUF           ;CLEAR LINE-BUFFER
04 13632 062677         IORST
05 13633 126440         SUBO    1,1
06 13634 044441         STA     1,PAGEC
07 13635 044441         STA     1,VERTC
08 13636 020145 PRSC0:  LDA     0,PRTFF
09 13637 004507         JSR     PRCOM   ;PRINT FORMFEED
10 13640 006445         JSR     @IHEA1  ;PRINT UPPER HEAD (COLUMN-NUMBERS)
11 13641 126440         SUBO    1,1     ;AC1:= 0
12 13642 044435         STA     1,HORZC
13 13643 000407         JMP     PRSC2
14 13644 126440 PRSC1:  SUBO    1,1
15 13645 044432         STA     1,HORZC ;CLEAR HORIZONTAL CHAR-COUNTER
16 13646 020144         LDA     0,PRTCR
17 13647 004477         JSR     PRCOM   ;PRINT CARRIAGE RETURN
18 13650 020146         LDA     0,SPA1L
19 13651 004475         JSR     PRCOM   ;SPACE ONE LINE
20 13652 020751 PRSC2:  LDA     0,CHAIN ;LOAD CHAR. TO BE PRINTED
21 13653 004463         JSR     PRAC0   ;PRINT IT
22 13654 010423         ISZ     HORZC
23 13655 020422         LDA     0,HORZC
24 13656 024147         LDA     1,CHALI
25 13657 106404         SUB     0,1,SZR ;TEST FOR END OF LINE
26 13660 000772         JMP     PRSC2   ;NOT END
27 13661 010415 PRSC3:  ISZ     VERTC   ;END
28 13662 020150         LDA     0,LIPAG
29 13663 024413         LDA     1,VERTC
30 13664 106404         SUB     0,1,SZR ;TEST FOR END OF PAGE
31 13665 000757         JMP     PRSC1   ;NOT END
32 13666 044410         STA     1,VERTC ;END  -  CLEAR LINECOUNTER
33 13667 006417         JSR     @IHEA2  ;PRINT LOWER HEAD (COLUMN-NUMBERS)
34 13670 010405 PRSC4:  ISZ     PAGEC
35 13671 020404         LDA     0,PAGEC ;AC0:= PAGECOUNTER
36 13672 004436         JSR     TECEN   ;TEST FOR END OF PRINT
37 13673 000743         JMP     PRSC0   ;NOT END
38 13674 002407         JMP     @PRTRE  ;RETURN
```

```
| 0148   .MAIN
01
02 13675 000000 PAGEC:   0
03 13676 000000 VERTC:   0
04 13677 000000 HORZC:   0
05 13700 000040 FSTCH:   40
06 13701 000000 ROTEM:   0
07 13702 000000 LETTER:  0
08 13703 000000 PRTRE:   0
09 13704 000000 CHSET:   0
10 13705 014070 IHEA1:   HEAD1
11 13706 014076 IHEA2:   HEAD2
12
13
14 13707 054417 REPASK:  STA      3,REPAR ;ASK FOR ROUTINE-REPEAT OR CONTINUE
15 13710 006043          CCRLF
16 13711 006046          CDICL
17 13712 006040          CMESS
18 13713 014655          MREPC            ;"REPEAT PRINTROUTINE(1) OR CONTINUE(0)? '
19 13714 006044          CDISP
20 13715 014702          DREPC
21 13716 006047          CDATT
22 13717 006103          CGTBI
23 13720 000771          JMP      REPASK+2
24 13721 000770          JMP      REPASK+2;ILLEGAL INPUT
25 13722 034404          LDA      3,REPAR
26 13723 014075          DSZ      DIGIN
27 13724 001401          JMP      1,3      ;GO ON
28 13725 001400          JMP      0,3      ;REPEAT
29
30 13726 000000 REPAR:   0
31 13727 007773 CENMK:   MARKC
32
33
34               ;ROUTINE TO TEST FOR END OF PRINT (AC0=PAGES PRINTED UNTIL NOW)
35
36
37 13730 026777 TECEN:   LDA      1,@CENMK
38 13731 125014          MOV#     1,1,SZR ;IS PRINTER CENTRONICS OR LOGABAX ?
39 13732 001401          JMP      1,3     ;YES - END OF PRINT (NO.OF PAGES=1)
40 13733 101212          MOVR#    0,0,SZC ;NO - 2 PAGES ?
41 13734 001400          JMP      0,3     ;NO
42 13735 001401          JMP      1,3     ;YES - END OF PRINT
```

```
I 0149  .MAIN
01
02                ;ROUTINE TO PRINT AC0 (IF WANTED THROUGH THE PRINTER-TABLE)
03
04 13736 054417 PRAC0:  STA     3,PRRET
05 13737 004513         JSR     STTST   ;TEST STATUS OF LPT
06 13740 032413         LDA     2,@PASMI
07 13741 151015         MOV#    2,2,SNR ;SKIP IF PR.TABLE NOT USED
08 13742 002412         JMP     @IFORM  ;FORM THE CHAR. AND EXAMINE PRINTERTABLE
09 13743 061017 PRTCH:  DOA     0,XLPT  ;THE ACTION ENTRY TABLE POINTS TO THIS
10 13744 060117         NIOS    XLPT    ;INSTRUCTION
11 13745 002410         JMP     @PRRET
12
13                ;ROUTINE TO PRINT AC0 (USED WHEN THE CHAR IN AC0 IS A PRINT-
14                ;OR PAPER-COMMAND)
15
16 13746 054407 PRCOM:  STA     3,PRRET ;OUTPUT A PRINTERCOMMAND - THE
17 13747 004503         JSR     STTST   ;PRINTERTABLE IS NOT USED
18 13750 061017         DOA     0,XLPT
19 13751 060117         NIOS    XLPT
20 13752 002403         JMP     @PRRET
21
22 13753 014170 PASMI:  PASMK
23 13754 001000 IFORM:  XFORM
24 13755 000000 PRRET:  0
25 13756 013531 IDEFL:  DEFLP
26
27
28
29
30
31
32
33
34                ; CHARACTERROTATION - ROUTINE
35
36
37 13757 054410 ROPRT:  STA     3,RORET ;PRINT 2 PAGES WITH ALL CHAR. IN ALL POS.
38 13760 006776         JSR     @IDEFL  ;DEFINE OR CALCULATE NO.OF LINES/PAGE
39 13761 004407 REROP:  JSR     PRROT   ;GO TO PRINTROUTINE
40 13762 020145         LDA     0,PRTFF ;PRINT FORM-FEED
41 13763 004763         JSR     PRCOM
42 13764 004723         JSR     REPASK  ;IS REPEATING OF CHAR.ROT.PRG. WANTED ?
43 13765 000774         JMP     REROP   ;YES
44 13766 002401         JMP     @RORET
45
46 13767 000000 RORET:  0
```

```
| 0150  .MAIN
01
02 13770 054713 PRROT:  STA     3,PRTRE
03 13771 006131         CLBUF           ;CLEAR LINE-BUFFER
04 13772 062677         IORST
05 13773 102400         SUB     0,0
06 13774 040701         STA     0,PAGEC ;CLEAR PAGECOUNTER
07 13775 040701         STA     0,VERTC ;CLEAR LINECOUNTER
08 13776 020702 ROT0:   LDA     0,FSTCH ;FIRST CHAR. ON PAGE
09 13777 040702         STA     0, ROTEM
10 14000 020145         LDA     0,PRTFF ;PRINT FORMFEED
11 14001 004745         JSR     PRCOM
12 14002 004466         JSR     HEAD1   ;PRINT UPPER HEAD (COLUMN-NUMBERS)
13 14003 102440         SUBO    0,0
14 14004 040673         STA     0,HORZC
15 14005 020674         LDA     0,ROTEM
16 14006 040674         STA     0,LETTER
17 14007 000411         JMP     ROT2
18 14010 102400 ROT1:   SUB     0,0
19 14011 040666         STA     0,HORZC ;CLEAR CHAR.COUNTER
20 14012 020667         LDA     0,ROTEM
21 14013 040667         STA     0,LETTER
22 14014 020144         LDA     0,PRTCR ;PRINT CARRIAGE RETURN
23 14015 004731         JSR     PRCOM
24 14016 020146         LDA     0,SPA1L
25 14017 004727         JSR     PRCOM
26 14020 020662 ROT2:   LDA     0,LETTER
27 14021 004715         JSR     PRAC0   ;PRINT THE CHARACTER IN LETTER
28 14022 010660         ISZ     LETTER  ;INCREMENT TO NEXT CHAR. IN ALPHABET
29 14023 020657         LDA     0,LETTER
30 14024 024660         LDA     1,CHSET
31 14025 030653         LDA     2,FSTCH
32 14026 106415         SUB#    0,1,SNR ;WAS IT THE LAST CHAR. IN THE DRUMSET
33 14027 050653         STA     2,LETTER;YES
34 14030 010647         ISZ     HORZC   ;NO
35 14031 020646         LDA     0,HORZC
36 14032 024147         LDA     1,CHALI
37 14033 106414         SUB#    0,1,SZR ;WAS IT THE LAST CHAR. ON THE LINE ?
38 14034 000764         JMP     ROT2    ;NO
39 14035 010641 ROT3:   ISZ     VERTC   ;YES
40 14036 010643         ISZ     ROTEM
41 14037 020150         LDA     0,LIPAG
42 14040 024636         LDA     1,VERTC
43 14041 106404         SUB     0,1,SZR ;WAS IT THE LAST LINE ON THE PAGE ?
44 14042 000746         JMP     ROT1    ;NO
45 14043 010632 ROT4:   ISZ     PAGEC   ;YES
46 14044 044632         STA     1,VERTC ;CLEAR LINECOUNTER
47 14045 004431         JSR     HEAD2   ;PRINT LOWER HEAD (COLUMN-NUMBERS)
48 14046 020627         LDA     0,PAGEC ;AC0:= PAGECOUNTER
49 14047 004661         JSR     TECEN   ;TEST FOR END OF PRINT
50 14050 000726         JMP     ROT0    ;NOT END
51 14051 002632         JMP     @PRTRE  ;RETURN
52
53
54              ;ROUTINE TO TEST STATUS OF LPT. IF OK IT WILL WAIT FOR
55              ;BUSY=0 (MAX 6 SEC) - IF NOT OK THE ROUT. NOAVA WILL BE EXE.
56
57 14052 054413 STTST:  STA     3,STRET ;TEST STATUS OF LPT - IF NOT
58 14053 064417         DIA     1,XLPT  ;AVAILABLE THEN GO TO NOAVA.
59 14054 125212         MOVR#   1,1,SZC
60 14055 004550         JSR     NOAVA
61 14056 040410         STA     0,AC0SA
62 14057 006063         TIMSK           ;WAIT FOR BUSY=0 (MAX 6 SEC)
63 14060 013560         6000.
64 14061 063517         SKPBZ   XLPT
65 14062 002405         JMP     @HAIFA  ;TIME-OUT  -  HARDWARE-FAILURE
66 14063 020403         LDA     0,AC0SA ;NORMAL OUT
67 14064 002401         JMP     @STRET
```

```
| 0151   .MAIN
01
02 14065 000000 STRET:  0
03 14066 000000 AC0SA:  0
04 14067 014263 HAIFA:  HWFAI
05
06                ;ROUTINES (HEAD1,HEAD2) TO PRINT UPPER AND LOWER HEADS, THAT
07                ;MEANS LINES WITH COLUMN-NUMBERS
08
09 14070 054463 HEAD1:  STA     3,REHEA ;PRINT AN UPPER HEAD WITH
10 14071 004562         JSR     TABUS   ;COLUMNNUMBERS - USE PRINTERTABLE
11 14072 004416         JSR     NOTEN
12 14073 004476         JSR     NOCNE
13 14074 004564         JSR     TABRE   ;RESTORE PRINTER-TABLE-USEMARK
14 14075 002456         JMP     @REHEA
15
16 14076 054455 HEAD2:  STA     3,REHEA ;PRINT A LOWER HEAD WITH
17 14077 020144         LDA     0,PRTCR ;COLUMNNUMBERS - USE PRINTERTABLE
18 14100 004646         JSR     PRCOM   ;PRINT LAST-LOADED LINE AND SPACE 1 LINE
19 14101 020146         LDA     0,SPA1L
20 14102 004644         JSR     PRCOM
21 14103 004550         JSR     TABUS
22 14104 004465         JSR     NOONE
23 14105 004403         JSR     NOTEN
24 14106 004552         JSR     TABRE   ;RESTORE PRINTER-TABLE-USEMARK
25 14107 002444         JMP     @REHEA
26
27 14110 054444 NOTEN:  STA     3,RETNO ;PRINT TEN'S
28 14111 020147         LDA     0,CHALI ;AC0:= NO. OF CHAR/LINE
29 14112 040443         STA     0,HECO1
30 14113 020445         LDA     0,AZERO ;AC0:= 0-CHAR (ASCII-60)
31 14114 040442         STA     0,HECO2
32 14115 020151         LDA     0,LDSPA
33 14116 006446         JSR     @PRAC   ;PRINT A SPACE
34 14117 014436         DSZ     HECO1
35 14120 000402         JMP     .+2
36 14121 000425         JMP     PRINL   ;END OF LINE - PRINT IT
37 14122 000471 SPAC8:  JMP     PR8SP   ;PRINT 8 SPACES
38 14123 010433         ISZ     HECO2   ;INCREMENT CHAR
39 14124 020432         LDA     0,HECO2
40 14125 006437         JSR     @PRAC   ;PRINT A NUMBER (1-9)
41 14126 014427         DSZ     HECO1   ;TEST FOR END OF LINE
42 14127 000402         JMP     .+2
43 14130 000416         JMP     PRINL   ;END OF LINE - PRINT IT
44 14131 020427         LDA     0,AZERO
45 14132 006432         JSR     @PRAC   ;PRINT A ZERO
46 14133 014422         DSZ     HECO1   ;TEST FOR END OF LINE
47 14134 000402         JMP     .+2
48 14135 000411         JMP     PRINL   ;END OF LINE - PRINT IT
49 14136 020420         LDA     0,HECO2
50 14137 024422         LDA     1,HOH71
51 14140 106032         ADCZ#   0,1,SZC ;SKIP IF HECO2 >=71 (DECI 9)
52 14141 000761         JMP     SPAC8
53 14142 020416         LDA     0,AZERO
54 14143 040413         STA     0,HECO2 ;RESET HECO2 TO 60 (DECI 0)
55 14144 014412         DSZ     HECO2
56 14145 000755         JMP     SPAC8   ;LOAD ON NUMBERS
```

```
I 0152  .MAIN
01
02 14146 020144 PRINL:  LDA     0,PRTCR ;PRINT LINE
03 14147 006410         JSR     @IPCOM
04 14150 020146         LDA     0,SPA1L ;SPACE ONE LINE
05 14151 006406         JSR     @IPCOM
06 14152 002402         JMP     @RETNO  ;RETURN TO CALL+1 OF NOTEN OR NOONE
07
08
09 14153 000000 REHEA:  0
10 14154 000000 RETNO:  0
11 14155 000000 HECO1:  0
12 14156 000000 HECO2:  0
13 14157 013746 IPCOM:  PRCOM
14 14160 000060 AZERO:  60
15 14161 000071 HOH71:  71
16 14162 000010 HOH8:   8.
17 14163 000000 HECO3:  0
18 14164 013736 PRAC:   PRAC0
19 14165 000000 MKSAV:  0
20 14166 000000 NOARE:  0
21 14167 000000 NOAS0:  0
22 14170 000000 PASMK:  0
23
24
25 14171 054763 NOONE:  STA     3,RETNO ;PRINT ONE'S
26 14172 020147         LDA     0,CHALI ;AC0:= NO. OF CHAR/LINE
27 14173 040762         STA     0,HECO1
28 14174 020764         LDA     0,AZERO ;AC0:= 60 (DECI 0)
29 14175 040761         STA     0,HECO2
30 14176 010760         ISZ     HECO2
31 14177 020757 PRTNO:  LDA     0,HECO2
32 14200 006764         JSR     @PRAC   ;PRINT DIGIT
33 14201 010755         ISZ     HECO2   ;INCREMENT DIGIT
34 14202 014753         DSZ     HECO1   ;TEST FOR END OF LINE
35 14203 000402         JMP     .+2
36 14204 000742         JMP     PRINL   ;END OF LINE - PRINT IT
37 14205 024754         LDA     1,HOH71
38 14206 106032         ADCZ#   0,1,SZC ;SKIP IF AC0>=71 (DECI 9)
39 14207 000770         JMP     PRTNO   ;PRINT-ON NUMBERS
40 14210 020750         LDA     0,AZERO
41 14211 040745         STA     0,HECO2 ;RESET HECO2 TO 60 (DECI 0)
42 14212 000765         JMP     PRTNO   ;PRINT-ON NUMBERS
43
44
45
46
47
48 14213 020747 PR8SP:  LDA     0,HOH8  ;PRINT 8 SPACES
49 14214 040747         STA     0,HECO3
50 14215 020151 RESPA:  LDA     0,LDSPA
51 14216 006746         JSR     @PRAC
52 14217 014736         DSZ     HECO1   ;TEST FOR END OF LINE
53 14220 000402         JMP     .+2
54 14221 000725         JMP     PRINL   ;END OF LINE - PRINT IT
55 14222 014741         DSZ     HECO3
56 14223 000772         JMP     RESPA
57 14224 000677         JMP     SPAC8+1 ;8 SPACES LOADED - RETURN TO CALL+1
```

```
| 0153   .MAIN
01
02 14225 040742 NOAVA:  STA     0,NOAS0 ;LPT ISN'T AVAILABLE, TYPE MESSAGE AND
03 14226 054740         STA     3,NOARE ;WAIT FOR OPERATORACTION,THEN TEST STATUS
04 14227 006043         CCRLF           ;AGAIN
05 14230 006046         CDICL
06 14231 006040         CMESS
07 14232 014713         NOTAV           ;"LPT ISN'T AVAIL."
08 14233 006044         CDISP
09 14234 014713         NOTAV
10 14235 006047         CDATT
11 14236 006062         WATOP           ;WAIT FOR KEY(TTI) OR CONT(FUB)
12 14237 020000         020000
13 14240 062677         IORST
14 14241 024151         LDA     1,LDSPA
15 14242 065017         DOA     1,XLPT
16 14243 060117         NIOS    XLPT
17 14244 063517         SKPBZ   XLPT
18 14245 000777         JMP     .-1
19 14246 064417         DIA     1,XLPT
20 14247 125212         MOVR#   1,1,SZC
21 14250 000413         JMP     HWFAI   ;STATUS IS STILL NOT OK-HARDWAREFAILURE
22 14251 020716         LDA     0,NOAS0 ;STATUS IS OK NOW
23 14252 002714         JMP     @NCARE
24
25
26 14253 020715 TABUS:  LDA     0,PASMK ;SAVE THE PRT.-TABLE-USE-MRK.
27 14254 040711         STA     0,MKSAV
28 14255 102440         SUBO    0,0
29 14256 040712         STA     0,PASMK
30 14257 001400         JMP     0,3
31
32 14260 020705 TABRE:  LDA     0,MKSAV ;RESTORE THE TABLE-USE-MARK
33 14261 040707         STA     0,PASMK
34 14262 001400         JMP     0,3
35
36
37 14263 006043 HWFAI:  CCRLF
38 14264 006046         CDICL
39 14265 006040         CMESS
40 14266 015447         HFAIL           ;"HARDWARE-FAILURE"
41 14267 006044         CDISP
42 14270 015447         HFAIL
43 14271 006047         CDATT
44 14272 063077         HALT
45 14273 000777         JMP     .-1
46
47
```

```
| 0154  .MAIN
01
02                ; TEXT - STRINGS
03
04                TSTST:  .TXTE !TESTEXECU. START!
   14274 142724
   14275 152123
   14276 154305
   14277 141705
   14300 027125
   14301 051640
   14302 040724
   14303 152322
   14304 000000
05
06
07                DTYAS:  .TXTE !PRINTERTYPE ? !
   14305 151120
   14306 047311
   14307 142724
   14310 152322
   14311 050131
   14312 120305
   14313 120077
   14314 000000
08
09
10                MCOAS:  .TXTE !FIRST(1) OR SECOND(2) CONTROLLER ? !
   14315 144706
   14316 051722
   14317 024324
   14320 124661
   14321 147640
   14322 120322
   14323 142523
   14324 147703
   14325 042116
   14326 131050
   14327 120251
   14330 147703
   14331 152116
   14332 147722
   14333 146314
   14334 151305
   14335 037640
   14336 000240
```

```
! 0155  .MAIN
01
02                MTYAS:  .TXTE !<15><12><12>PRINTERTYPE: RC3632,3633(0),RC3634,
    14337 005215
    14340 050012
    14341 144722
    14342 152116
    14343 151305
    14344 054724
    14345 142520
    14346 120072
    14347 141722
    14350 033063
    14351 131063
    14352 031654
    14353 031466
    14354 024063
    14355 124460
    14356 151254
    14357 031703
    14360 031466
    14361 126264
03  14362 033063 3635(1),RC3636(2),<15><12>RC3641(3),RC3638(4),RC3639(5),RC3637
    14363 032463
    14364 130450
    14365 126251
    14366 141722
    14367 033063
    14370 033063
    14371 131050
    14372 126251
    14373 005215
    14374 141722
    14375 033063
    14376 130664
    14377 031450
    14400 126251
    14401 141722
    14402 033063
    14403 134063
    14404 132050
    14405 126251
    14406 141722
    14407 033063
    14410 034463
    14411 032450
    14412 126251
    14413 141722
    14414 033063
    14415 133463
04  14416 033050 (6) OR RC3643(7) ? !
    14417 120251
    14420 151317
    14421 151240
    14422 031703
    14423 132066
    14424 024063
    14425 124667
    14426 037640
    14427 000240
```

```
| 0156  .MAIN
01
02                DDIAT:   .TXTE !DIAGN.TST.OR NOT!
   14430 144504
   14431 043501
   14432 027116
   14433 051724
   14434 027324
   14435 151317
   14436 047240
   14437 152317
   14440 000000
03
04                MVCHN:   .TXTE !NO. OF CHANNELS IN VERTICAL FORMAT UNIT ? !
   14441 147516
   14442 120056
   14443 143317
   14444 141640
   14445 040510
   14446 047116
   14447 146305
   14450 120123
   14451 047311
   14452 053240
   14453 151305
   14454 144724
   14455 040703
   14456 120314
   14457 147706
   14460 046722
   14461 152101
   14462 052640
   14463 144516
   14464 120324
   14465 120077
   14466 000000
05
06                DVCHN:   .TXTE !NO.OF CHN.IN VFU!
   14467 147516
   14470 147456
   14471 120306
   14472 044303
   14473 027116
   14474 047311
   14475 053240
   14476 052706
   14477 000000
07
08                MDIAT:   .TXTE !DIAGNOSTICTEST(1) OR NOT(0) ? !
   14500 144504
   14501 043501
   14502 147516
   14503 152123
   14504 141711
   14505 142724
   14506 152123
   14507 130450
   14510 120251
   14511 151317
   14512 047240
   14513 152317
   14514 030050
   14515 120251
   14516 120077
   14517 000000
```

```
I 0157  .MAIN
01
02
03                MMANT:  .TXTE !TESTS WITH OPERATOR-INVOLVEMENT(1) OR NOT(0) ? !
   14520 142724
   14521 152123
   14522 120123
   14523 144727
   14524 044324
   14525 147640
   14526 142520
   14527 040722
   14530 147724
   14531 026722
   14532 047311
   14533 147526
   14534 053314
   14535 046705
   14536 047305
   14537 024324
   14540 124661
   14541 147640
   14542 120322
   14543 147516
   14544 024324
   14545 124460
   14546 037640
   14547 000240
04
05
06                DMANT:  .TXTE !OPERT.TST.OR NOT!
   14550 050317
   14551 151305
   14552 027324
   14553 051724
   14554 027324
   14555 151317
   14556 047240
   14557 152317
   14560 000000
07
08
09                MMOTS:  .TXTE !CHARACTERPRINT-TEST(1) OR NOT(0) ? !
   14561 044303
   14562 151101
   14563 141501
   14564 142724
   14565 050322
   14566 144722
   14567 152116
   14570 152055
   14571 051705
   14572 024324
   14573 124661
   14574 147640
   14575 120322
   14576 147516
   14577 024324
   14600 124460
   14601 037640
   14602 000240
```

```
| 0158  .MAIN
01
02                DMOTS:  .TXTE !CHA.PRINT OR NOT!
   14603 044303
   14604 027101
   14605 151120
   14606 047311
   14607 120324
   14610 151317
   14611 047240
   14612 152317
   14613 000000
03
04
05                DROTT:  .TXTE !CHA.MOTIO.OR NOT!
   14614 044303
   14615 027101
   14616 147515
   14617 144724
   14620 027317
   14621 151317
   14622 047240
   14623 152317
   14624 000000
06
07
08                MCHPD:  .TXTE !NO. OF CHARACTERS ON DRUM ? !
   14625 147516
   14626 120056
   14627 143317
   14630 141640
   14631 040510
   14632 040722
   14633 152303
   14634 151305
   14635 120123
   14636 047317
   14637 042240
   14640 052722
   14641 120115
   14642 120077
   14643 000000
09
10
11                DCHPD:  .TXTE !NO.OF CH.ON DRUM!
   14644 147516
   14645 147456
   14646 120306
   14647 044303
   14650 147456
   14651 120116
   14652 151104
   14653 046525
   14654 000000
```

```
| 0159  .MAIN
01
02                MREPC:  .TXTE !REPEAT PRINTROUTINE(1) OR CONTINUE(0) ? !
   14655 142722
   14656 142520
   14657 152101
   14660 050240
   14661 144722
   14662 152116
   14663 147722
   14664 152125
   14665 047311
   14666 024305
   14667 124661
   14670 147640
   14671 120322
   14672 147703
   14673 152116
   14674 047311
   14675 142525
   14676 030050
   14677 120251
   14700 120077
   14701 000000
03
04
05                DREPC:  .TXTE !REPEAT OR CONT.?!
   14702 142722
   14703 142520
   14704 152101
   14705 147640
   14706 120322
   14707 147703
   14710 152116
   14711 037456
   14712 000000
06
07
08                NOTAV:  .TXTE !LPT ISN'T AVAIL.!
   14713 050314
   14714 120324
   14715 051711
   14716 023516
   14717 120324
   14720 053101
   14721 144501
   14722 027314
   14723 000000
09
10
11                ENDPA:  .TXTE !END OF PASS: !
   14724 047305
   14725 120104
   14726 143317
   14727 050240
   14730 051501
   14731 035123
   14732 000240
```

```
| 0160  .MAIN
01
02                DRUMG:  .TXTE !OPEN DRUMGATE!
   14733 050317
   14734 047305
   14735 042240
   14736 052722
   14737 043515
   14740 152101
   14741 000305
03
04
05                BELAC:  .TXTE !DID BELL RING ? !
   14742 144504
   14743 120104
   14744 142502
   14745 146314
   14746 151240
   14747 047311
   14750 120107
   14751 120077
   14752 000000
06
07
08                MUSEL:  .TXTE !MAKE LPT USEABLE!
   14753 040515
   14754 142513
   14755 146240
   14756 152120
   14757 052640
   14760 142523
   14761 041101
   14762 142714
   14763 000000
09
10
11                MREPA:  .TXTE !SELECT LPT - REMOVE PAPER!
   14764 142523
   14765 142714
   14766 152303
   14767 146240
   14770 152120
   14771 026640
   14772 151240
   14773 046705
   14774 053317
   14775 120305
   14776 040520
   14777 142520
   15000 000322
12
13
14                INSPA:  .TXTE !REPLACE PAPER!
   15001 142722
   15002 146120
   15003 141501
   15004 120305
   15005 040520
   15006 142520
   15007 000322
```

```
| 0161  .MAIN
01
02                MLOCL:  .TXTE !CLOSE DRUMGATE - SET LPT IN LOCAL-STATE!
   15010 146303
   15011 051717
   15012 120305
   15013 151104
   15014 046525
   15015 040507
   15016 142724
   15017 026640
   15020 051640
   15021 152305
   15022 146240
   15023 152120
   15024 144640
   15025 120116
   15026 147714
   15027 040703
   15030 026714
   15031 152123
   15032 152101
   15033 000305
03
04
05                DLOCL:  .TXTE !CL.DRG-LOCAL LPT!
   15034 146303
   15035 042056
   15036 043722
   15037 146055
   15040 141717
   15041 146101
   15042 146240
   15043 152120
   15044 000000
06
07
08                MCHPL:  .TXTE !NO. OF CHARACTERS PER LINE ? !
   15045 147516
   15046 120056
   15047 143317
   15050 141640
   15051 040510
   15052 040722
   15053 152303
   15054 151305
   15055 120123
   15056 142520
   15057 120322
   15060 144714
   15061 142516
   15062 037640
   15063 000240
09
10
11                DCHPL:  .TXTE !NO.OF CHAR/LINE?!
   15064 147516
   15065 147456
   15066 120306
   15067 044303
   15070 151101
   15071 146257
   15072 047311
   15073 037705
   15074 000000
```

```
! 0162  .MAIN
01
02                INCHA:   .TXTE !INPUT CHARACTER !
   15075 047311
   15076 052520
   15077 120324
   15100 044303
   15101 151101
   15102 141501
   15103 142724
   15104 120322
   15105 000000
03
04
05                MROTT:   .TXTE !CHARACTERMOTION-TEST(1) OR NOT(0) ? !
   15106 044303
   15107 151101
   15110 141501
   15111 142724
   15112 046722
   15113 152317
   15114 147711
   15115 026516
   15116 142724
   15117 152123
   15120 130450
   15121 120251
   15122 151317
   15123 047240
   15124 152317
   15125 030050
   15126 120251
   15127 120077
   15130 000000
06
07
08                MEOFP:   .TXTE !SET LPT ON LINE - REMOVE PAPER EXCEPT ONE FORM!
   15131 142523
   15132 120324
   15133 050314
   15134 120324
   15135 047317
   15136 146240
   15137 047311
   15140 120305
   15141 120055
   15142 142722
   15143 147515
   15144 142526
   15145 050240
   15146 050101
   15147 151305
   15150 142640
   15151 141730
   15152 050305
   15153 120324
   15154 047317
   15155 120305
   15156 147706
   15157 046722
   15160 000000
```

```
! 0163  .MAIN
01                DEOFP:  .TXTE !LINE-LEAV.1 FORM!
   15161 144714
   15162 142516
   15163 146055
   15164 040705
   15165 027126
   15166 120261
   15167 147706
   15170 046722
   15171 000000
02
03
04                PAPFA:  .TXTE !REMOVE PAPER!
   15172 142722
   15173 147515
   15174 142526
   15175 050240
   15176 050101
   15177 151305
   15200 000000
05
06
07                MINMO:  .TXTE !INPUTMODE = ASCII-ALPHABET(1) OR OCTAL(0) ? !
   15201 047311
   15202 052520
   15203 046724
   15204 042317
   15205 120305
   15206 120275
   15207 051501
   15210 144703
   15211 026711
   15212 146101
   15213 044120
   15214 041101
   15215 152305
   15216 130450
   15217 120251
   15220 151317
   15221 147640
   15222 152303
   15223 146101
   15224 030050
   15225 120251
   15226 120077
   15227 000000
08
09
10                DINMO:  .TXTE !ASC.OR OCT.INPUT!
   15230 051501
   15231 027303
   15232 151317
   15233 147640
   15234 152303
   15235 144456
   15236 050116
   15237 152125
   15240 000000
```

```
| 0164  .MAIN
01                DEOPA:  .TXTE !E.O.PASS: !
   15241 027305
   15242 027317
   15243 040520
   15244 051523
   15245 120072
   15246 000000
02
03
04                DCOAS:  .TXTE !CONTROLLERTYPE?!
   15247 147703
   15250 152116
   15251 147722
   15252 146314
   15253 151305
   15254 054724
   15255 142520
   15256 000077
05
06
07                EXA00:  .TXTE !EXAMINE PRINTOUT, AND COMPARE WITH PC: !
   15257 154305
   15260 046501
   15261 047311
   15262 120305
   15263 151120
   15264 047311
   15265 147724
   15266 152125
   15267 120254
   15270 047101
   15271 120104
   15272 147703
   15273 050115
   15274 151101
   15275 120305
   15276 144727
   15277 044324
   15300 050240
   15301 035303
   15302 000240
08
09
10                MANSE:  .TXTE !OFF-LINE/ON-LINE!
   15303 143317
   15304 026706
   15305 144714
   15306 142516
   15307 147657
   15310 026516
   15311 144714
   15312 142516
   15313 000000
```

```
| 0165  .MAIN
01
02                EXA01:   .TXTE !EXAM. PRINTOUT!
   15314 154305
   15315 046501
   15316 120056
   15317 151120
   15320 047311
   15321 147724
   15322 152125
   15323 000000
03
04
05                EXA02:   .TXTE !COMP. PC: !
   15324 147703
   15325 050115
   15326 120056
   15327 141520
   15330 120072
   15331 000000
06
07
08                SETLO:   .TXTE !DESELECT PRINTER!
   15332 142504
   15333 142523
   15334 142714
   15335 152303
   15336 050240
   15337 144722
   15340 152116
   15341 151305
   15342 000000
09
10
11                DREPA:   .TXTE !SEL.PR-REMOV.PAP!
   15343 142523
   15344 027314
   15345 151120
   15346 151055
   15347 046705
   15350 053317
   15351 050056
   15352 050101
   15353 000000
12
13
14                DPACO:   .TXTE !PAPERTEST OR NOT!
   15354 040520
   15355 142520
   15356 152322
   15357 051705
   15360 120324
   15361 151317
   15362 047240
   15363 152317
   15364 000000
```

```
| 0166  .MAIN
01
02                MPACO:  .TXTE !TESTS WITH PAPERCONSUMPTION(1) OR NOT(0) ? !
   15365 142724
   15366 152123
   15367 120123
   15370 144727
   15371 044324
   15372 050240
   15373 050101
   15374 151305
   15375 147703
   15376 051516
   15377 046525
   15400 152120
   15401 147711
   15402 024116
   15403 124661
   15404 147640
   15405 120322
   15406 147516
   15407 024324
   15410 124460
   15411 037640
   15412 000240
03
04
05                MUTAB:  .TXTE !USE PRINTERTABLE(1) OR PASS-BY(0) ? !
   15413 051525
   15414 120305
   15415 151120
   15416 047311
   15417 142724
   15420 152322
   15421 041101
   15422 142714
   15423 130450
   15424 120251
   15425 151317
   15426 050240
   15427 051501
   15430 026523
   15431 054502
   15432 030050
   15433 120251
   15434 120077
   15435 000000
06
07
08                DUTAB:  .TXTE !USE P.TAB OR NOT!
   15436 051525
   15437 120305
   15440 027120
   15441 040724
   15442 120102
   15443 151317
   15444 047240
   15445 152317
   15446 000000
```

```
| 0167  .MAIN
01
02                HFAIL:   .TXTE !HARDWARE-FAILURE!
   15447 040510
   15450 042322
   15451 040727
   15452 142722
   15453 143055
   15454 144501
   15455 052714
   15456 142722
   15457 000000
03
04
05                PROG:    .TXTE !RC3600 LPT-TEST!
   15460 141722
   15461 033063
   15462 030060
   15463 146240
   15464 152120
   15465 152055
   15466 051705
   15467 000324
06
07
08                DINCH:   .TXTE !INPUT CHAR. !
   15470 047311
   15471 052520
   15472 120324
   15473 044303
   15474 151101
   15475 120056
   15476 000000
09
10
11                SETOL:   .TXTE !SET LPT ON-LINE!
   15477 142523
   15500 120324
   15501 050314
   15502 120324
   15503 047317
   15504 146055
   15505 047311
   15506 000305
12
13
14                MA307:   .TXTE    !LPC306/307(1) OR NOT(0) ?!
   15507 050314
   15510 031703
   15511 033060
   15512 031657
   15513 133460
   15514 130450
   15515 120251
   15516 151317
   15517 047240
   15520 152317
   15521 030050
   15522 120251
   15523 000077
```

```
! 0168  .MAIN
01
02                TABDE:  .TXTE !SELECT PR.TABLE !
   15524 142523
   15525 142714
   15526 152303
   15527 050240
   15530 027322
   15531 040724
   15532 146102
   15533 120305
   15534 000000
03
04
05                TABME:  .TXTE !PRINTERTABLE: ASCII(0),STANDARD ST.(1),STANDARD
   15535 151120
   15536 047311
   15537 142724
   15540 152322
   15541 041101
   15542 142714
   15543 120072
   15544 051501
   15545 144703
   15546 024311
   15547 124460
   15550 051654
   15551 040724
   15552 042116
   15553 151101
   15554 120104
   15555 152123
   15556 024056
   15557 124661
   15560 051654
   15561 040724
   15562 042116
   15563 151101
   15564 120104
06 15565 152123 ST0(2),<15><12>PL1,TYPE 70(3),READ FROM PTR(4) ? !
   15566 024060
   15567 124662
   15570 106654
   15571 050012
   15572 130714
   15573 152254
   15574 050131
   15575 120305
   15576 030267
   15577 031450
   15600 126251
   15601 142722
   15602 042101
   15603 143240
   15604 147722
   15605 120115
   15606 152120
   15607 024322
   15610 124664
   15611 037640
   15612 000240
```

```
! 0169  .MAIN
01
02                CHKSE:   .TXTE !CHECKSUM-ERROR!
   15613 044303
   15614 141705
   15615 051513
   15616 046525
   15617 142455
   15620 151322
   15621 151317
   15622 000000
03
04
05                MLPTR:   .TXTE !LOAD PTR, STRIKE RETURN-KEY!
   15623 147714
   15624 042101
   15625 050240
   15626 151324
   15627 120254
   15630 152123
   15631 144722
   15632 142513
   15633 151240
   15634 152305
   15635 151125
   15636 026516
   15637 142513
   15640 000131
06
07
08                DLPTR:   .TXTE !LDA PR-TYPE CONT!
   15641 042314
   15642 120101
   15643 151120
   15644 152055
   15645 050131
   15646 120305
   15647 147703
   15650 152116
   15651 000000
09
10
11                DA307:   .TXTE   !LPC306/7 OR NOT?!
   15652 050314
   15653 031703
   15654 033060
   15655 133657
   15656 147640
   15657 120322
   15660 147516
   15661 037724
   15662 000000
```

```
| 0170  .MAIN
01
02                REFHAM: .TXTE   !REF.HAMM.=!
   15663 142722
   15664 027306
   15665 040510
   15666 046515
   15667 136456
   15670 000000
03
04                NOOFHA: .TXTE   !# OF HAMMERS=!
   15671 120243
   15672 143317
   15673 044240
   15674 046501
   15675 142515
   15676 051722
   15677 000275
05
06                NOOFZO: .TXTE   !# OF ZONES=!
   15700 120243
   15701 143317
   15702 055240
   15703 047317
   15704 051705
   15705 000275
07
08                STRHAM: .TXTE   !STARTHAMMER=!
   15706 152123
   15707 151101
   15710 044324
   15711 046501
   15712 142515
   15713 136722
   15714 000000
09
10
11
12
13
14
15
16
17
18
19
20
21
22 15715 165000 JUS22:  MOV     3,1         ;MOVE STARTADDRESS TO AC1
23 15716 006072         CSAMS               ;TYPE STARTADDRESS
24 15717 102000         ADC     0,0         ;AC0:= 177777
25 15720 040575         STA     0,M2230
26 15721 020405         LDA     0,HM136 ;AC0:= 136.
27 15722 040574         STA     0,NOHAM ;SET NO. OF HAMMERS
28 15723 102520         SUBZL   0,0         ;AC0:= 1
29 15724 040573         STA     0,NOZON ;SET NO. OF ZONES
30 15725 000436         JMP     RASK3
31
32 15726 000210 HM136:  136.
```

```
| 0171  .MAIN
01
02 15727 165000 HMJUS:  MOV     3,1       ;MOVE STARTADDRESS TO AC1
03 15730 006072         CSAMS             ;TYPE STARTADDRESS
04 15731 102440         SUBO    0,0
05 15732 040563         STA     0,M2230
06 15733 006043         CCRLF
07 15734 006046         CDICL
08 15735 006040         CMESS
09 15736 015671         NOOFHA            ;"# OF HAMMERS="
10 15737 006044         CDISP
11 15740 015671         NOOFHA
12 15741 006050         CHAAT
13 15742 006105         CGTDC             ;GET DECIMAL NUMBER TO "DIGIN"
14 15743 000764         JMP     HMJUS     ;ILLEGAL INPUT
15 15744 000763         JMP     HMJUS     ;ILLEGAL INPUT
16
17 15745 020075         LDA     0,DIGIN
18 15746 040550         STA     0,NOHAM
19
20 15747 006043 RASK2:  CCRLF
21 15750 006046         CDICL
22 15751 006040         CMESS
23 15752 015700         NOOFZO            ;"# OF ZONES="
24 15753 006044         CDISP
25 15754 015700         NOOFZO
26 15755 006050         CHAAT
27 15756 006105         CGTDC
28 15757 000770         JMP     RASK2     ;ILLEGAL INPUT
29 15760 000767         JMP     RASK2     ;ILLEGAL INPUT
30
31 15761 020075         LDA     0,DIGIN
32 15762 040535         STA     0,NOZON
33
34 15763 006043 RASK3:  CCRLF
35 15764 006046         CDICL
36 15765 006040         CMESS
37 15766 015706         STRHAM            ;"STARTHAMMER="
38 15767 006044         CDISP
39 15770 015706         STRHAM
40 15771 006050         CHAAT
41 15772 006105         CGTDC
42 15773 000770         JMP     RASK3     ;ILLEGAL INPUT
43 15774 000767         JMP     RASK3     ;ILLEGAL INPUT
44
45 15775 020520         LDA     0,M2230
46 15776 101015         MOV#    0,0,SNR   ;SKIP IF DP2230
47 15777 000407         JMP     NTCHE
48 16000 020075         LDA     0,DIGIN   ;AC0:= STARTHAMMER
49 16001 101120         MOVZL   0,0       ;AC0:= STARTHAMMER * 2
50 16002 126000         ADC     1,1       ;AC1:= -1
51 16003 123000         ADD     1,0       ;AC0:= PRINTPOSS. TO START WITH
52 16004 040514         STA     0,STHAM
53 16005 000403         JMP     HMINI
54
55 16006 020075 NTCHE:  LDA     0,DIGIN
56 16007 040511         STA     0,STHAM
```

```
I 0172  .MAIN
01
02 16010 126000 HMINI:  ADC     1,1      ;AC1:= -1,TO FORCE MESS ABOUT REF. HAMMER
03 16011 044553         STA     1,RFHAM  ;REF. HAMMER
04
05 16012 024504         LDA     1,NOHAM  ;AC1:= NUMBER OF HAMMERS
06 16013 030504         LDA     2,NOZON  ;AC2:= NUMBER OF ZONES
07 16014 006067         DIVIS            ;DIVIDE AC1 WITH AC2,RESULT IN AC1
08
09 16015 020560         LDA     0,HM120  ;AC0:= 120.
10 16016 122032         ADCZ#   1,0,SZC  ;SKIP IF AC1>=AC0
11 16017 000403         JMP     .+3
12 16020 020476         LDA     0,NOHAM
13 16021 000407         JMP     SETZO    ;SET ZONESIZE = NO.OF HAMM.
14 16022 020555         LDA     0,HM60   ;AC0:= 60.
15 16023 122032         ADCZ#   1,0,SZC  ;SKIP IF AC1>=AC0
16 16024 000403         JMP     .+3
17 16025 020551         LDA     0,HM72   ;AC0:= 72. = ZONESIZE
18 16026 000402         JMP     SETZO
19 16027 020551         LDA     0,HM24   ;AC0:= 24. = ZONESIZE
20
21 16030 040544 SETZO:  STA     0,ZONEW  ;SET ZONE-SIZE
22
23 16031 030467         LDA     2,STHAM  ;HAMMER TO START WITH
24 16032 050533 OFFON:  STA     2,CURHM  ;CURRENT HAMMER IN TEST
25
26 16033 126440         SUBO    1,1
27 16034 020540         LDA     0,ZONEW
28
29 16035 107000         ADD     0,1
30 16036 146433         SUBZ#   2,1,SNC  ;SKIP IF (CURRENT HAMM)<=(ZONESIZE*N)
31 16037 000776         JMP     .-2
32 16040 125400         INC     1,1
33 16041 106400         SUB     0,1      ;AC1:= CURRENT REF. HAMMER
34
35 16042 020522         LDA     0,RFHAM
36 16043 106415         SUB#    0,1,SNR  ;SKIP IF NEW REF. HAMMER
37 16044 000415         JMP     MOSTR
38
39 16045 044517         STA     1,RFHAM
40 16046 006043         CCRLF
41 16047 006046         CDICL
42 16050 006040         CMESS
43 16051 015663         REFHAM           ;"REF.HAMM.="
44 16052 006044         CDISP
45 16053 015663         REFHAM
46 16054 024510         LDA     1,RFHAM
47 16055 006053         CTDEC
48 16056 006057         CDDEC
49 16057 006047         CDATT
50 16060 006043         CCRLF
51
52 16061 020505 MOSTR:  LDA     0,REPCO  ;AC0:= 100
53 16062 040505         STA     0,REPPR  ;NUMBER OF PRINTCOMMANDS BEFORE LF
54
55 16063 152520 LISTR:  SUBZL   2,2      ;AC2:= 1 = CURRENT COUNTER
56
57 16064 024500 RFSTR:  LDA     1,RFHAM
58 16065 132414         SUB#    1,2,SZR  ;SKIP IF (CURRENT COUNTER)=(REF.HAMMER)
59 16066 004433         JSR     PRHSP    ;PRINT A SPACE - RETURN .+2
60 16067 004434         JSR     PRHCH    ;PRINT CHARACTER - RETURN .+2
61
62 16070 000774         JMP     RFSTR
63
64 16071 020474         LDA     0,CURHM  ;CURRENT HAMMER IN TEST
65 16072 106415         SUB#    0,1,SNR  ;SKIP IF (CURRENT HAMM.)<>(REF. HAMM)
66 16073 000406         JMP     LHSTR    ;FILL UP WITh SPACES
```

```
| 0173  .MAIN
01
02 16074 024471 HASTR:  LDA     1,CURHM
03 16075 132414         SUB#    1,2,SZR ;SKIP IF (CURRENT COUNTER)=(CURRENT HAMM)
04 16076 004423         JSR     PRHSP   ;PRINT A SPACE - RETURN .+2
05 16077 004424         JSR     PRHCH   ;PRINT CHARACTER - RETURN .+2
06
07 16100 000774         JMP     HASTR
08
09 16101 024415 LHSTR:  LDA     1,NOHAM ;FILL UP WITH SPACES
10 16102 146432         SUBZ#   2,1,SZC ;SKIP IF (CURRENT COUNTER)>(LAST HAMMER)
11 16103 004416         JSR     PRHSP   ;PRINT A SPACE - RETURN .+2
12 16104 000402         JMP     .+2
13 16105 000774         JMP     LHSTR
14
15 16106 020144         LDA     0,PRTCR ;LINE FILLED UP - PRINT IT
16 16107 004424         JSR     HPRIN   ;PRINT CR - RETURN .+1
17 16110 014457         DSZ     REPPR   ;SKIP IF TIME FOR LINE-FEED
18 16111 000752         JMP     LISTR   ;FILL LINE AGAIN
19 16112 020456         LDA     0,LFCOM
20 16113 004420         JSR     HPRIN   ;PRINT LINE-FEED - RETURN .+1
21 16114 000745         JMP     MOSTR   ;SET PRINTCOUNTER AND FILL LINE AGAIN
22
23 16115 000000 M2230:  0               ;=177777 IF PRINTER = DP2230, ELSE = 0
24 16116 000000 NOHAM:  0               ;OPERATOR DEFINED NO. OF HAMMERS
25 16117 000000 NOZON:  0               ;OPERATOR DEFINED NO. OF ZONES
26 16120 000000 STHAM:  0               ;OPERATOR DEFINED HAMMER TO START WITH
27
28
29 16121 102440 PRHSP:  SUBO    0,0     ;PRINT A SPACE - RETURN + 2
30 16122 000407         JMP     PRHC1
31 16123 020446 PRHCH:  LDA     0,HCHAE ;PRINT A "E" - RETURN + 2
32 16124 054460         STA     3,PRRTU
33 16125 024770         LDA     1,M2230
34 16126 125014         MOV#    1,1,SZR ;SKIP IF NOT DP2230
35 16127 004456         JSR     CHCAL   ;LOOK UP CHARACTER FROM TABLE
36 16130 034454         LDA     3,PRRTU
37 16131 151400 PRHC1:  INC     2,2     ;INCREMENT CURRENT COUNTER
38 16132 175400         INC     3,3     ;INCREMENT RETURN ADDRESS
39 16133 054437 HPRIN:  STA     3,HPRIR ;STORE RETURN
40
41 16134 004423         JSR     HMPRI   ;PRINT AC0
42
43 16135 060417         DIA     0,XLPT  ;CHECK STATUS
44 16136 034435         LDA     3,HMOFF ;AC3:= 200
45 16137 117415         AND#    0,3,SNR
46 16140 002432         JMP     @HPRIR  ;PRINTER IS ON-LINE  -  RETURN
47
48 16141 060517         DIAS    0,XLPT  ;PRINTER IS OFFLINE  - WAIT
49 16142 117414         AND#    0,3,SZR
50 16143 000776         JMP     .-2     ;STILL OFF-LINE
51
52 16144 020424         LDA     0,LFCOM ;ON-LINE AGAIN, PRINT LF
53 16145 004412         JSR     HMPRI
54 16146 030417         LDA     2,CURHM
55 16147 151400         INC     2,2     ;GOTO NEXT HAMMER
56 16150 020745         LDA     0,M2230 ;<> 0 IF DP2230
57 16151 101004         MOV     0,0,SZR ;SKIP IF NOT DP2230
58 16152 151400         INC     2,2     ;IF DP2230 THEN INC. TWO PRINTPOSS.
59 16153 020743         LDA     0,NOHAM
60 16154 142433         SUBZ#   2,0,SNC ;SKIP IF NEXT HAMM. <= NO HAMM.
61 16155 000654         JMP     OFFON-1 ;NEXT HAMM. := START HAMM.
62 16156 000654         JMP     OFFON
```

```
| 0174  .MAIN
01
02 16157 061017 HMPRI:  DOA     0,XLPT
03 16160 060117         NIOS    XLPT
04 16161 063517         SKPBZ   XLPT
05 16162 000777         JMP     .-1
06 16163 001400         JMP     0,3
07
08 16164 000000 RFHAM:  0       ;CURRENT REFERENCE HAMMER
09 16165 000000 CURHM:  0       ;CURRENT HAMMER IN TEST
10 16166 000100 REPCO:  100     ;NUMBER OF PRINTS BEFORE LF
11 16167 000000 REPPR:  0       ;# OF PRINTS - COUNTER
12 16170 002501 LFCOM:  2501    ;LINE-FEED COMMAND
13 16171 000105 HCHAE:  105     ;ASCII "E"
14 16172 000000 HPRIR:  0       ;RETURN ADDRESS FROM PRINT ROUTINE
15 16173 000200 HMOFF:  200     ;OFF-LINE STATUS
16 16174 000000 ZONEW:  0       ;CALCULATED SIZE OF A ZONE
17 16175 000170 HM120:  120.    ;CONSTANTS
18 16176 000110 HM72:   72.
19 16177 000074 HM60:   60.
20 16200 000030 HM24:   24.
21 16201 000021 HM17:   17.
22 16202 000000 HMSA3:  0
23 16203 000000 HMSA2:  0
24 16204 000000 PRRTU:  0               ;RETURN SAVE
25
26
27
28 16205 054775 CHCAL:  STA     3,HMSA3
29 16206 050775         STA     2,HMSA2
30 16207 145220         MOVZR   2,1
31 16210 030771         LDA     2,HM17  ;AC2:= 17.
32 16211 006067         DIVIS
33                                      ;AC0:= REMAINDER OF AC1/AC2
34 16212 034406         LDA     3,HMTAB ;AC3:= ADDRESS OF TABLE
35 16213 117000         ADD     0,3
36 16214 021400         LDA     0,0,3   ;AC0:= CURRENT CHAR FROM TABLE
37 16215 030766         LDA     2,HMSA2
38 16216 034764         LDA     3,HMSA3
39 16217 001400         JMP     0,3
40
41 16220 016221 HMTAB:  .+1
42 16221 000105         105
43 16222 000103         103
44 16223 000101         101
45 16224 000140         140
46 16225 000073         73
47 16226 000061         61
48 16227 000063         63
49 16230 000065         65
50 16231 000067         67
51 16232 000104         104
52 16233 000102         102
53 16234 000056         56
54 16235 000072         72
55 16236 000060         60
56 16237 000062         62
57 16240 000064         64
58 16241 000066         66
59
60
61 16242 000000 EOFTS:  0               ;END OF TEST MARK
```

```
| 0175   .MAIN
01
02
03          001400            .END    REBIN
  0176   .MAIN

A0SAV    012531     135/40   135/48   135/56   135/61   137/09   137/53
A1SAV    012532     135/41   135/49   135/57   135/62   137/10   137/51
A2SAV    012533     135/42   135/50   135/58   135/63   137/11   137/52
AANUK    005313      74/45    74/47
AASK5    007250      96/23    96/41
AC0SA    014066     150/61   151/02   151/03
AC101    007767     103/55   104/32
AC102    007770     103/58   104/33
ACONT    007505     100/37   101/00
ACOUT    012553     135/55   135/61
ACT0     001071      30/16
ACT6     001072      30/17
ACTN     001054      29/29    30/02
ACTSF    007437      98/26    99/07    99/09   100/12   100/15
ADDRS    001711      37/45    38/11    38/21    38/31    38/32    38/59
ADP20    007765     103/49   104/30
ADP30    007766     103/52   104/31
ADP40    007764     103/46   104/29
ADP70    007763     103/43   104/28
AGTYP    003013      50/11    50/18    50/37
ALDRT    010677     114/07   114/23
ALOGB    007771     103/61   104/34
ALPTT    002255      42/57    44/48
AMEND    001721      39/14    39/22
ARESW    001322      34/18    34/24
ASAVE    010675     114/05   114/31
ASK0     007257      95/10    97/05
ASK1     007260      97/08
ASK3     007261      95/16    97/13
ASK4     007263      97/17
ASK4A    007265      97/23
ASK5     007266      96/41    97/27   107/37
ASK8     007271      97/33
ASKRE    013610     146/16   146/32   146/33   146/34
ASKTB    010514      96/46   109/32   110/56   111/32
AZERO    014160     151/30   151/44   151/53   152/14   152/28   152/40
BBNUK    005320      74/51
BC20     001667      38/25    38/39
BC377    001617      37/54    37/58
BEG01    000153      21/38    22/08
BEG02    000154      21/39    22/11
BEG03    000155      21/40    22/14
BEG04    000156      21/41    22/18
BEG05    000157      21/42    22/22
BEG06    000160      21/43    22/23
BEG07    000161      21/44    22/24
BEGAD    010113     105/46   106/03
BELAC    014742     138/52   138/54   160/05
BEND     001712      38/60
BIDIG    006116      82/20    82/43    82/49
BINAD    001547      35/19    37/09    44/01
BINFI    001375      34/63    35/17    44/00
BINLA    001376      35/00    35/12    43/39
BIT3M    011775     126/51   127/26
BLOCK    001625      37/57    38/03    38/05    38/37
BMEND    001732      39/18    39/23
BRESW    001356      34/41    34/47
BSTRP    001703      38/53
BT00     011074      99/40   118/04
```

```
CEFF    010570    112/27  112/45
CELIF   010567    112/25  112/44
CEN1M   007774     96/31  102/27  102/47  104/37
CENMK   013727    148/31  148/37
CENT1   007577    102/21  103/54
CENT2   007610    102/31  103/57
CETSP   010605    113/06  115/17
CGTBI   006103     19/45  107/10  107/49  108/10  109/10  110/17  110/35  113/22
                  148/22
CGTDC   006105     19/47   44/51  106/20  109/57  111/15  171/13  171/27  171/41
CGTOK   006104     19/46   66/37   68/12   68/30   70/12   70/23   70/34  103/10
                  104/16  105/10  113/45  146/31
CGTSC   006106     19/48  146/14
CGTTX   006107     19/49
CH40    006601     89/14   89/33
CHAAT   006050     19/25   33/44   90/37  101/27  103/09  104/15  105/09  106/19
                  107/09  107/48  108/09  109/09  109/56  110/16  110/34  111/14
                  113/21  113/44  114/42  136/48  138/55  139/12  140/08  140/56
                  141/55  142/24  142/36  143/53  144/17  144/48  146/10  171/12
                  171/26  171/40
CHADV   010066    105/45
CHAHA   006330     85/09   85/11   85/23   85/44
CHAIN   013623    146/18  146/29  147/20
CHALI   000147     20/29  100/45  109/63  130/13  131/53  133/07  135/07  147/24
                  150/36  151/28  152/26
CHAR0   000770     27/12   27/31   27/35   29/11   29/47
CHAR1   000704     28/19
CHAR2   000675     28/12   28/15
CHAR3   000713     28/27   28/35
CHAR4   000721     28/21   28/30   28/33
CHAR7   000776     28/28   29/17
CHARY   012350    131/25  132/33  132/53
CHCAL   016205    173/35  174/28
CHCR    001076     28/58   29/60   30/22
CHDST   007474    100/42  100/55
CHFLG   000547     26/08   26/24   26/31   26/34   26/41   27/19
CHINH   000455     25/40   27/04
CHINS   010374    108/39  108/42  108/45  109/49  110/00
CHKER   001664     37/44   38/24   38/36  114/13
CHKSE   015613    115/08  115/10  169/02
CHLF    001077     28/55   29/03   29/62   30/23   31/37
CHLMK   010402     96/44  101/44  109/47  109/61  110/06
CHMAS   000461     25/09   25/44
CHMIN   000546     26/17   26/40
CHNCO   013023    138/05  138/17  139/02
CHNNO   013022    138/04  139/01
CHO16   007327     98/28   99/12   99/28
CHO64   010513    111/06  111/29
CHORZ   000775     28/27   28/46   29/05   29/16
CHPLU   000545     26/07   26/25   26/39
CHRA2   006006     80/15   80/54
CHRA3   006117     82/15   82/46
CHRAN   005553     76/16   77/34
CHRET   000661     27/63   28/03   28/06   28/09   28/24   28/26   28/32
CHSAV   000774     28/11   28/17   29/15
CHSET   013704    100/58  111/24  148/09  150/30
CHSIG   000544     26/10   26/18   26/32   26/38
CHSP    000662     26/19   26/28   27/09   28/00   28/33
CHSPA   000663     26/09   28/01
```

HM120	016175	172/09	174/17						
HM136	015726	170/26	170/32						
HM17	016201	174/21	174/31						
HM24	016200	172/19	174/20						
HM60	016177	172/14	174/19						
HM72	016176	172/17	174/18						
HMEND	000074	18/32	39/25	43/31					
HMINI	016010	171/53	172/02						
HMJUS	015727	21/43	171/02	171/14	171/15				
HMOFF	016173	173/44	174/15						
HMPRI	016157	173/41	173/53	174/02					
HMSA2	016203	174/23	174/29	174/37					
HMSA3	016202	174/22	174/28	174/38					
HMTAB	016220	174/34	174/41						
HOC16	012522	134/48	135/33						
HOH05	013524	145/11	145/47						
HOH31	010512	111/19	111/28						
HOH33	012354	132/03	132/36	132/57					
HOH71	014161	151/50	152/15	152/37					
HOH8	014162	152/16	152/48						
HORZC	013677	147/12	147/15	147/22	147/23	148/04	150/14	150/19	150/34
		150/35							
HPRIN	016133	173/16	173/20	173/39					
HPRIR	016172	173/39	173/46	174/14					
HWFAI	014263	151/04	153/21	153/37					
I1LPR	012641	136/56	138/09	138/13	138/19				
IACTS	007325	98/26	99/10						
IBUSS	000127	20/13	20/37						
IBZOT	006543	89/02	90/27						
ICEMK	012642	136/57	137/19						
ICHAR	000041	18/05	19/18						
ICHKE	010705	114/13	114/35	114/56					
ICHLM	007556	100/47	101/44						
ICHSE	010506	111/21	111/24						
ICLBU	000131	20/15	20/39						
ICLMK	007253	96/16	96/44						
ICMRK	010564	111/03	112/07	112/20	112/41	113/10	113/33	115/14	
ICRLF	000043	18/07	19/20						
IDATT	000047	18/11	19/24						
IDBIN	000055	18/17	19/30						
IDDEC	000057	18/19	19/32						
IDEFL	013756	149/25	149/38						
IDICL	000046	18/10	19/23						
IDISP	000044	18/08	19/21						
IDIVD	000070	18/28	19/41						
IDIVS	000067	18/27	19/40						
IDOCT	000056	18/18	19/31						
IDONS	000130	20/14	20/36						
IDOSE	010575	112/10	112/24	112/36	112/50				
IDOUT	000045	18/09	19/22						
IDPSP	010767	115/16	115/22						
IDX0	000020	17/53	51/06	51/32	53/28	53/39	104/02	104/03	104/05
		104/07							
IDX1	000021	17/54	51/04	51/48	52/25	52/28	52/30	53/26	53/55
IDX2	000022	17/55	52/44	52/55	52/61	54/03	54/06		
IDX3	000023	17/56	51/08	52/11	53/30	94/29	94/31	94/41	94/43
		94/53	94/55						
IDZOC	000060	18/20	19/33						
IEXIN	007235	96/18	96/30						

IFCEN	007501	100/39	100/60						
IFCLE	007256	96/22	96/47						
IFORM	013754	149/08	149/23						
IGNOR	001612	37/42	37/53	37/56					
IGTBI	000103	18/45	19/45						
IGTDC	000105	18/47	19/47						
IGTOK	000104	18/46	19/46						
IGTSC	000106	18/48	19/48						
IGTTX	000107	18/49	19/49						
IHAAT	000050	18/12	19/25						
IHALT	000114	18/54	19/54						
IHEA1	013705	147/10	148/10						
IHEA2	013706	147/33	148/11						
ILL	001067	30/04	30/05	30/06	30/07	30/08	30/10	30/11	30/12
		30/14							
ILLEG	005367	75/31	76/36	76/39	76/48	80/45	85/30		
ILLGR	005366	75/13	75/15	75/30					
ILMRK	010572	112/08	112/22	112/38	112/47				
ILOGM	013515	143/02	144/58						
ILOM1	012225	130/21	131/17						
ILOMK	012643	136/58	137/22						
ILOOP	000113	18/53	19/53						
ILORE	000122	18/60	19/60						
ILRST	000126	20/12	20/38						
IMEND	002312	43/26	43/40	43/51					
IMESS	000040	18/04	19/17						
IMODE	007552	100/48	101/40						
IMOMK	010510	109/34	110/57	111/26					
IMRKC	013517	145/06	145/26						
IMULT	000066	18/26	19/39						
INCHA	015075	146/07	162/02						
INCHS	007477	100/43	100/58						
INDAD	003153	52/12	52/27						
INDW1	003122	51/60							
INDW2	003350	54/02							
INHSW	006603	89/08	89/35	90/06					
INIA	007204	95/09	96/04						
INIB	007207	95/14	95/20	95/35	96/07				
INICO	007504	100/44	100/63						
INIEX	010400	96/30	100/57	108/15	108/36	109/30	110/04	110/54	
INILI	013521	145/08	145/54						
INIRE	007500	100/35	100/53	100/59					
INITY	007453	96/32	100/35						
INMOD	013621	146/11	146/27						
INMRK	010351	101/40	109/39	109/44	146/27				
INNOP	010254	96/34	108/21						
INNUK	005317	74/09	74/50						
INRET	005422	75/38	75/41	75/42	75/48	75/61	76/04	80/50	85/31
INSAD	003145	52/07	52/20						
INSCO	010120	105/50	105/54	106/08					
INSPA	015001	144/14	144/16	160/14					
INSRE	010704	114/12	114/34						
INSTA	003132	52/08	52/09						
INSTB	003133	52/09	52/10						
INSTR	003123	52/02	53/15						
INSW1	003121	51/59							
INSW2	003347	54/01							
INTAD	007233	96/12	96/28						
INTER	005400	75/41	77/13	80/51	85/32				

MCMEX	002264	43/16	44/31				
MCMNM	002273	43/21	43/25	43/48	44/24		
MCOAS	014315	105/06	154/10				
MCPUT	005002	50/43	50/45	67/11			
MCRLF	004743	39/54	67/02	75/44	75/47	89/30	
MDCTR	002700	48/31	48/55	49/00			
MDIAT	014500	107/06	156/08				
MDMMC	005113	69/08	70/07	70/08			
MDMMF	005120	69/10	70/18	70/19			
MELOC	001470	17/44	36/04				
MEOFP	015131	140/53	162/08				
MESCH	000434	25/16	25/21				
MESSA	000422	25/10	25/17				
MHEAD	006605	89/38	90/18				
MIMEX	002313	43/22	43/41	43/49	44/25	44/32	
MINMO	015201	110/13	163/07				
MKLOG	011073	117/25	118/23				
MKPAS	010571	112/46	113/28				
MKSAV	014165	152/19	153/27	153/32			
MLLOC	001501	35/33	35/35	36/08	43/28	43/30	
MLOCL	015010	140/05	161/02				
MLOOP	002656	48/32	48/36				
MLORE	007070	92/55	92/57	93/50			
MLPTR	015623	114/39	169/05				
MLPTT	002223	42/31	44/46	44/47			
MMANT	014520	107/45	157/03				
MMOTS	014561	109/06	157/09				
MODAT	001414	35/20	35/26				
MOFID	002246	42/41	42/50				
MOGTT	002250	42/37	42/52				
MOLAD	002247	42/40	42/51				
MOMRI	007245	96/05	96/08	96/38			
MOMRK	010112	96/38	106/02	111/26			
MOPTB	002230	42/36	44/60	114/10			
MOPTR	002245	42/36	42/45	42/49			
MOREP	002236	42/42	42/48				
MOSTR	016061	172/37	172/52	173/21			
MPACO	015365	108/06	166/02				
MPASS	007050	93/28	93/32				
MPOWO	004745	66/27	67/04				
MREPA	014764	143/50	160/11				
MREPC	014655	148/18	159/02				
MRK47	000132	20/16	96/15	103/37	104/25	141/33	141/46
MRKLP	007234	96/20	96/29				
MRKTB	007254	96/21	96/45				
MROTT	015106	110/31	162/05				
MS004	012415	133/20	133/40				
MSAMS	001530	36/15	39/49	39/51			
MSAQU	004755	66/32	67/08				
MSAV	002701	48/30	48/38	48/54	48/63	49/01	
MSEC4	012640	136/55	137/30				
MSK12	011402	121/44	122/28	122/39	123/03		
MSK47	007637	102/57					
MST10	002623	47/59	48/11				
MSTIA	002557	47/19	47/23				
MSTIB	002562	47/17	47/22				
MSTIC	002627	48/02	48/04	48/16			
MSTID	002630	47/27	48/00	48/03			
MSTIM	002553	18/24	47/15				

MSTIO	002624	47/21	47/62					
MSTIR	002652	47/15	48/17	48/19	48/20	48/21		
MSWRG	001522	34/32	36/13					
MTYAS	014337	103/06	155/02					
MULSA	005565	77/39	77/43	77/46				
MULTE	005557	77/26	77/39					
MULTI	006066	19/39	48/12	89/18				
MUSEL	014753	142/21	142/23	160/08				
MUTAB	015413	113/18	166/05					
MVCHN	014441	106/16	156/04					
MX2SP	005207	71/07	71/45	93/20				
MXMMF	005070	68/07	69/02					
MXMMT	005106	68/25	68/26	69/06	70/29	70/30		
MXQUE	005206	71/05	71/41	71/43				
NBEVE	006367	85/41	85/57					
NC125	003635	57/13	57/45					
NC8	003634	56/25	57/44					
NCTYP	003540	56/34	56/37	56/39				
NDCON	010202	107/25	107/36	107/62				
NEXIS	003553	56/27	56/43					
NFTYP	003531	56/21	56/32					
NINHI	000577	27/03	27/06					
NIOIN	011342	119/25	121/06					
NIRET	003541	56/04	56/38	56/40				
NITYP	003532	56/33	57/53					
NN10	000772	29/13	29/45					
NN500	003632	56/17	57/42					
NN9	003633	56/18	57/43					
NOARE	014166	152/20	153/03	153/23				
NOAS0	014167	152/21	153/02	153/22				
NOAVA	014225	150/60	153/02					
NODIA	010165	96/37	107/22					
NODRE	007247	95/39	96/24	96/40				
NOEX	006520	88/16	88/26	88/35	88/45			
NOHAM	016116	170/27	171/18	172/05	172/12	173/09	173/24	173/59
NOMAN	010225	96/33	107/60					
NOMO1	010325	96/35	109/19					
NOOFH	015671	170/04	171/09	171/11				
NOOFZ	015700	170/06	171/23	171/25				
NOONE	014171	151/12	151/22	152/25				
NOOPR	011403	121/45	124/08					
NOOPT	007240	95/24	96/33	97/20				
NOPAP	010255	108/14	108/22					
NOPIN	007241	95/26	96/34					
NOPRP	007242	95/27	96/35	97/29				
NORAT	006575	89/13	89/29					
NOROP	007243	95/29	96/36	97/35				
NOROT	010441	96/36	110/44					
NOTAV	014713	153/07	153/09	159/08				
NOTDI	007244	95/40	96/37	97/14				
NOTEN	014110	151/11	151/23	151/27				
NOTRE	003575	57/02	57/10	57/11				
NOTYM	003542	56/41	57/05	57/07				
NOTYP	003564	56/33	57/02					
NOVCH	012467	134/09	135/02					
NOZON	016117	170/29	171/32	172/06	173/25			
NPCON	010263	108/24	108/29					
NRCON	010447	110/46	110/51					
NRESW	001352	34/17	34/42					

PRTFF	000145	20/27	112/14	112/28	127/31	130/08	130/50	131/41	132/17
		133/03	134/12	134/41	138/06	138/20	141/26	145/31	146/20
		147/08	149/40	150/10					
PRTLI	013630	146/19	147/02						
PRTNU	014177	152/31	152/39	152/42					
PRTON	012233	131/24	131/37						
PRTRE	013703	147/02	147/38	148/08	150/02	150/51			
PRTST	012217	130/15	131/11	133/09					
PRTU	012220	130/26	131/12						
PRTYP	003016	35/62	50/41						
PSAAN	004735	66/34	66/48						
PSETP	004737	66/05	66/50	66/56					
PSTAC	004736	66/04	66/49	66/55					
PTAB	001160	29/22	31/24						
PTAB1	003656	42/53	58/03						
PTAB2	004057	42/54	60/03						
PTAB3	004260	42/55	62/03						
PTAB4	004461	42/56	64/03						
PTBMO	010702	114/10	114/16						
QCHAR	000673	28/07	28/10						
QDICL	001173	31/34	31/37						
QDOUT	001144	31/07	31/11						
QHAAT	001227	32/39	32/42						
QMEND	002400	44/11	44/22	44/27	44/37				
QTABL	010606	111/38	113/09	115/19					
QTABR	010641	113/35	113/38	113/46	113/47	113/61			
QTYPE	000734	28/42	28/46						
QU9EX	010502	111/07	111/19						
QUE0	007331	97/05	98/30						
QUE00	007640	98/30	101/39	103/02					
QUE01	010023	98/31	105/02						
QUE02	010122	106/12	108/49						
QUE03	010144	98/32	107/02						
QUE04	010204	98/33	107/41						
QUE05	010306	98/35	109/02						
QUE06	010352	108/43	109/35	109/46	110/59				
QUE07	010403	109/36	110/09						
QUE08	010422	98/36	110/27						
QUE09	010461	110/60	111/02						
QUE1	007332	97/08	98/31						
QUE3	007333	97/13	98/32						
QUE4	007334	97/17	98/33						
QUE4A	010233	98/34	108/02						
QUE4I	007335	97/23	98/34						
QUE5	007336	97/27	98/35						
QUE6A	010633	110/05	113/32						
QUE6F	010631	113/14	113/28						
QUE8	007337	97/33	98/36						
QUEFI	007274	95/31	95/41	95/47	97/40	107/35			
QUESA	005205	71/03	71/26	71/37					
QUESD	005230	71/23	71/34						
QUESM	005232	71/21	71/36						
QUESS	005235	71/30	71/39						
QUEST	005234	71/28	71/39						
QUR00	007641	103/03	103/11	103/12	103/35				
QUR01	010024	105/03	105/11	105/12	105/17				
QUR02	010123	106/13	106/21	106/22	106/25				
QUR03	010145	107/03	107/11	107/12					
QUR04	010205	107/42	107/50	107/51					

QUR05	010307	109/03	109/11	109/12					
QUR06	010356	109/50	109/58	109/59					
QUR07	010404	110/10	110/18	110/19					
QUR08	010423	110/28	110/36	110/37					
QUR09	010467	111/05	111/08	111/16	111/17				
QUR4A	010234	108/03	108/11	108/12					
QUR6A	010614	113/12	113/15	113/23	113/24				
QURXX	007741	104/09	104/17	104/18	104/23				
RACAN	005334	74/28	75/00						
RADYN	005326	74/12	74/16	74/58					
RAEND	005277	74/14	74/31						
RALIF	005327	74/11	74/24	74/59					
RAMIN	005332	74/22	74/62						
RANDI	005255	74/13	74/17						
RANK	003254	53/16	53/19						
RAPLU	005331	74/20	74/61						
RASK2	015747	171/20	171/28	171/29					
RASK3	015763	170/30	171/34	171/42	171/43				
RASPA	005330	74/26	74/60						
RAZER	005333	74/10	74/63						
RBZOT	000451	25/25	25/34	25/36					
RDERR	010750	20/11	115/05						
RDRET	010740	114/07	114/50						
REBIN	001400	17/46	35/08	175/03					
REDSV	010701	114/09	114/29	114/51					
REEXP	012526	135/37	136/34	136/45	136/49				
REFHA	015663	170/02	172/43	172/45					
REG0	001100	28/37	28/45	28/50	28/54	28/63	30/24	31/02	31/10
		31/17							
REG1	001101	28/47	29/07	30/25	31/11	31/20			
REG2	001102	28/48	29/06	30/26	31/12	31/21			
REG3	001103	28/38	28/41	28/44	29/08	30/27	31/03	31/06	31/09
		31/22							
REHEA	014153	151/09	151/14	151/16	151/25	152/09			
REINI	007232	96/10	96/25	96/27					
RELPR	013574	146/04	146/15	146/16	146/24				
REMRK	007246	95/38	96/39						
RENOD	010203	96/40	107/31	107/37					
REOF2	006030	81/02	81/06	81/07	81/09	81/13	81/14	81/15	
REOF3	006141	83/02	83/06	83/07	83/09	83/13	83/14	83/15	
REPAR	013726	148/14	148/25	148/30					
REPAS	013707	146/22	148/14	148/23	148/24	149/42			
REPCO	016166	172/52	174/10						
REPL4	005552	77/15	77/29	77/31	77/32				
REPL5	006326	85/03	85/10	85/19	85/20				
REPPR	016167	172/53	173/17	174/11					
REROP	013761	149/39	149/43						
RESET	012347	131/39	131/47	131/49	131/56	132/02	132/09	132/12	132/13
		132/52							
RESPA	014215	152/50	152/56						
RESTR	007175	21/42	95/43						
RET00	007775	100/61	103/02	103/41	104/26	104/38	105/02	105/44	105/56
RET01	010143	106/12	106/28	106/30	107/02	107/13	107/41	107/52	109/02
		109/13							
RET02	010376	108/02	108/17	108/21	108/27	108/35	108/38	108/48	108/50
		109/29	109/40	110/02	110/27	110/38			
RET03	010377	109/46	110/01	110/03	110/09	110/20			
RET04	010507	110/53	110/61	111/25					
RET05	010511	111/02	111/22	111/27	113/09	113/29			

WT07	013136	140/30							
WT08	013146	140/39							
WT09	013156	100/03	140/49						
WT10	013177	141/02							
WT11	013207	141/11							
WT12	013222	141/23							
WT13	013245	100/04	141/45						
WT14	013271	142/09							
WT15	013313	100/05	142/29						
WT16	013336	142/50							
WT17	013345	143/02							
WT18	013357	143/04	143/13						
WT19	013366	143/20							
WT20	013376	143/29							
WT21	013406	143/38							
WT22	013416	143/47							
WT23	013436	144/02							
WT24	013461	100/06	144/24						
WTFUB	001265	33/22	33/29	33/31					
WTNOK	001253	33/21	33/33						
WTOK	001270	33/30	33/34						
WTORE	001305	33/16	33/46	33/47					
XC20	003227	52/26	52/45	53/13					
XCHAR	000664	18/05	28/02						
XCPN	002573	45/19	46/17	47/32	48/05				
XCRLF	001043	18/07	29/56						
XDBIN	001020	18/17	29/36						
XDDEC	000464	18/19	25/48						
XDICL	001165	18/10	31/30						
XDISP	001104	18/08	30/29						
XDIVD	002666	18/28	48/54						
XDIVS	002665	18/27	48/53						
XDLTE	005777	79/12	80/12	80/46	82/12				
XDOCT	000564	18/18	26/58						
XDOUT	001134	18/09	31/02						
XDZOC	000554	18/20	26/48						
XFDIS	001776	39/39	40/00						
XFITY	001377	35/01	35/29						
XFORM	001000	29/20	149/23						
XFROM	005064	68/09	68/55	70/20					
XGTCH	006001	79/09	80/09	80/48	82/09				
XILLG	005776	78/23	78/25	78/27	78/33	78/35	78/40	79/34	79/51
		80/33	80/36	80/45	81/12	82/33	82/36	83/12	
XINRT	006003	79/04	80/04	80/50	82/04				
XINST	003231	51/05	53/15	53/27					
XINTR	006004	80/00	80/51	81/36	83/34				
XLORE	007007	18/60	92/46						
XLPT	000017	19/12	20/07	25/28	28/52	28/53	98/06	98/12	116/37
		116/42	116/43	116/49	116/54	116/55	116/60	117/04	118/07
		118/08	118/09	118/15	118/16	118/17	118/28	118/29	118/30
		118/31	118/32	118/33	118/34	118/42	118/43	118/46	118/53
		118/60	118/61	118/62	119/05	119/07	119/13	119/29	119/43
		119/44	119/47	119/54	119/55	119/58	120/05	120/06	120/14
		120/22	120/23	120/24	120/31	120/32	120/35	120/37	120/44
		120/45	120/46	120/49	120/51	121/13	121/14	121/17	121/19
		123/28	123/38	123/40	123/49	123/50	123/53	124/09	124/10
		124/12	124/14	124/31	124/32	124/33	124/37	124/41	125/05
		125/07	125/10	125/12	125/42	125/43	125/44	125/48	125/52
		126/09	126/21	126/33	127/08	127/33	127/34	127/37	127/52

XWTYP	003012	50/20	50/36				
XX16	003226	52/27	52/33	52/46	52/56	53/12	
XXLPT	001774	39/35	39/62				
XXTTO	001775	39/37	39/63				
YCHAR	000672	25/41	25/59	28/09	29/34	29/61	29/63
YDICL	001172	31/36	32/32				
YDLTE	006334	84/12	85/28				
YDOUT	001142	25/42	29/38	30/53	31/09	31/38	
YEDIA	010162	96/43	107/16				
YEMO1	010335	98/24	109/29				
YEPAP	010277	108/18	108/46				
YEPRP	007323	97/30	98/24				
YEROP	007324	97/36	98/25				
YEROT	010450	98/25	110/53				
YGTCH	006332	84/09	85/26				
YHAAT	001226	32/18	32/41				
YILLG	006336	84/31	84/47	85/30			
YINRT	006337	84/04	85/31				
YINTR	006340	85/32	85/55				
YONTR	006341	85/33	85/36				
YPBIN	001024	29/33	29/37	29/40			
YPDEC	000502	25/49	25/54	26/02			
YPOCT	000603	26/54	26/59	27/12			
YTAB3	000712	25/43	28/26				
YTRMT	006333	84/10	85/27				
YTYPE	000732	28/22	28/34	28/44			
YTYPN	006335	84/20	85/29				
YZOCT	000601	26/44	26/49	27/09			
ZCHAR	000456	25/23	25/41				
ZDOUT	000457	25/42	27/00				
ZEASK	007551	100/36	101/39				
ZONEW	016174	172/21	172/27	174/16			
ZSUPP	000771	27/20	27/26	27/36	29/12		
ZTAB3	000460	25/43	25/62				