

0001 .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
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64

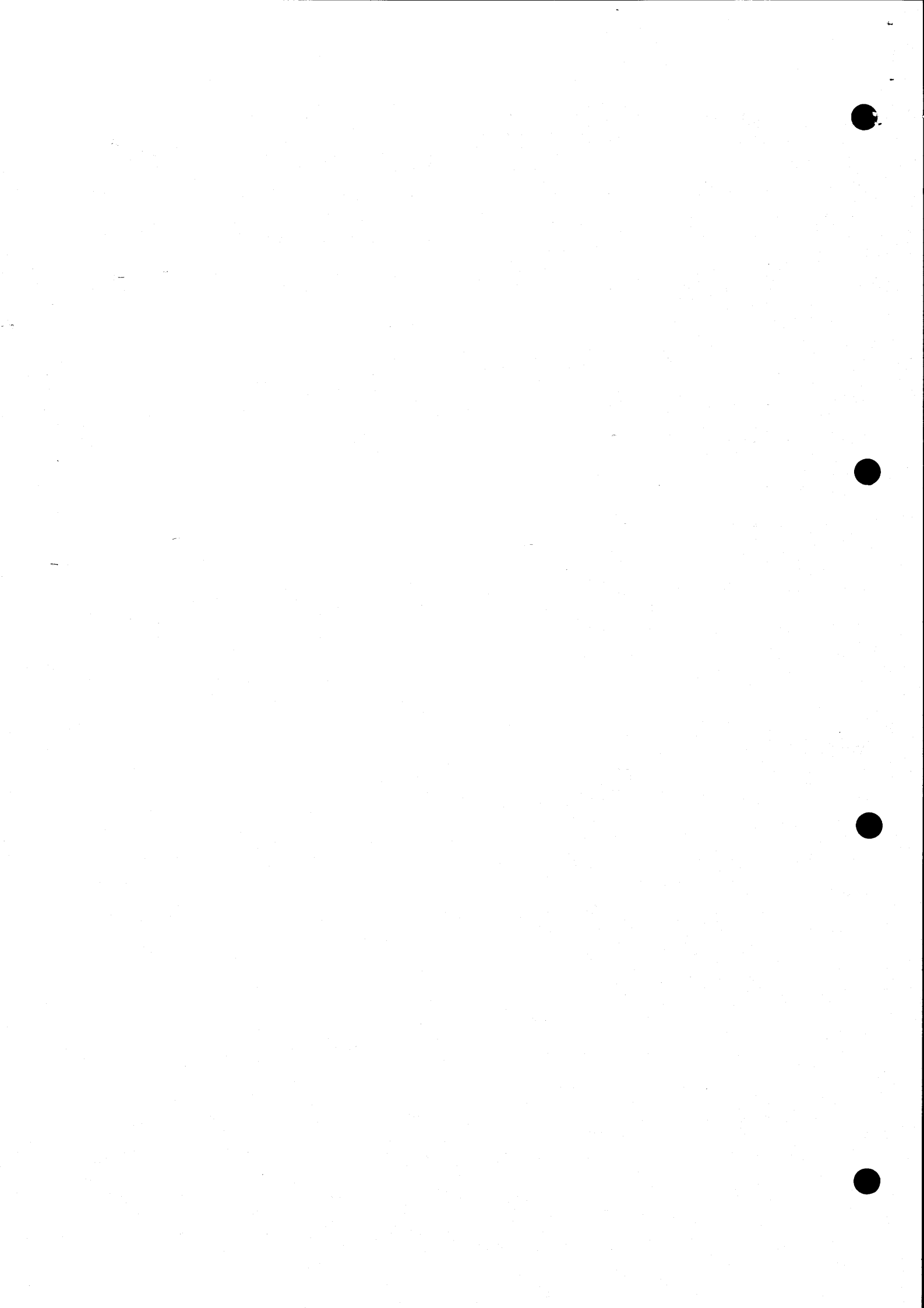
;RCSL: 44-RT 1653

;AUTOR: FLEMMING KAM

;EDITED: 78.01.11

; RC 3600
; READER/PUNCH TEST.
; REVISION 0

;S-BINARY TAPE: RCSL 44-RT 1654 (SELFSTART AUTOLOAD HEA
;ASCII SOURCE: RCSL 44-RT 1655 TAPE 1-5
;BINARY CARD: RCSL 44-RT 1656
;KEYWORDS: PAPER TAPE READER, PAPER TAPE PUNCH,
RC 3665, RC 3675 AND RC 3676.
;ABSTRACT: READER/PUNCH TEST IS A MAINTENANCE PROGRAM
DESIGNED TO DETECT MALFUNCTIONS IN THE HIGH SPEED
PAPER TAPE CONTROLLERS: TRC 701 AND PUC 703.



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

03 ; 0. INDEX

04 ;	1. ABSTRACT	PAGE 1
05 ;	2. MACHINE REQUIREMENTS	PAGE 1
06 ;	3. SWITCH SETTINGS AND ININESS	PAGE 2
07 ;	4. OPERATING PROCEDURE	PAGE 3
08 ;	5. PROGRAM DESCRIPTION	PAGE 5
09 ;	6. SOFTWARE SURVEY	PAGE 8
10 ;	7. MESSAGES FROM THIS TEST	PAGE 9

16 ; 1. ABSTRACT

17 ; READER/PUNCH TEST IS A STAND-ALONE MAINTENANCE
 18 ; PROGRAM DESIGNED TO DETECT MALFUNCTIONS IN THE
 19 ; TYPE RC3665 (75 CPS) PAPER TAPE PUNCH AND THE
 20 ; TYPE RC3675 (2000 CPS) OR RC3676 (500 CPS) PAPER
 21 ; TAPE READERS.

26 ; 2. MACHINE REQUIREMENTS

27 ;	RC 3600 FAMILY PROCESSOR	CPU
28 ;	MINIMUM 8K READ/WRITE MEMORY	MEM
29 ;	TELETYPE OR OPERATORS CONTROL PANEL	TTY/OCP *NOTE
30 ;	PAPER TAPE READER AND/OR PUNCH	PTR/PTP
31 ;	LINEPRINTER, OPTIONALLY	LPT *NOTE

32 ;
 33 ; *NOTE: IF THIS DEVICES IS SET ONLINE AFTER PROGRAM
 34 ; START, THE PROGRAM MUST BE RESTARTED TO GET OUTPUT.

```

01 ;
02 ; 3. SWITCH SETTINGS AND INITIAL MESSAGES:
03 ; 3.1 STARTING ADDRESSES - METHOD 3.3 START 1
04 ; 400 PUNCH DIAGNOSTIC
05 ; 401 READER DIAGNOSTIC
06 ; 402 PUNCH FROM SWITCHES
07 ; 403 PUNCH COUNTER
08 ; 404 READ COUNTER
09 ; 405 ALTERNATE PUNCH
10 ; 406 ALTERNATE READ
11 ; 407 PUNCH FLOATING ZERO
12 ; 410 READ FLOATING ZERO
13 ; 411 RESTART OF TEST, NO QUESTIONS, LAST ANSWERS.
14 ; 1400 RESTART PROGRAM AS WHEN LOADED.
15 ; 2202 GET A NEW PRINTER ALPHABETH (SEE 4.1.1)
16 ; 2204 SET TO 64K WORDS MODE, MEM SIZE ?
17 ; 2206 SET TO 32K WORDS MODE, MEM SIZE ?
18 ; 2210 EXAMINE MEMORY
19 ; 2212 DEPOSIT MEMORY
20 ; 2214 TROUBLE BREAKPOINT HALT
21 ; 2216 TROUBLE BREAKPOINT LOOP REPORT
22 ; 2220 TROUBLE BREAKPOINT RESET
23 ; 2222 START BINARY LOADER, READ FROM PTR/TTI (SW 0)
24 ; 7600 CHANGE PTR/PTP DEVICE CODE, DEFAULT 12/13.
25 ;
26 ; 3.2 CONTROL BY SWITCH SETTING, THE STATE REPORTED
27 ; AT TELETYPE AND LINEPRINTER
28 ; SW0: LOOP IN TEST IN CASE OF ERROR.
29 ; SW10: INHIBIT PRINTOUT IN TESTLOOP-PROGRAM.
30 ; SW11: PRINT FAILURE RATE IN TESTLOOP-PROGRAM.
31 ; SW12: NO HALT IN CASE OF ERROR IN TESTLOOP-PRGCR.
32 ; SW13: WAIT AFTER DIS MESSAGE. DIS IS THE 16 CHAR
33 ; SW14: HALT AFTER DIS MESSAGE. DISPLAY AT OPERAT.
34 ; SW15: CLEAR DIS AFTER MESSAGE. CONTROL PANEL. OCP
35 ;
36 ; 3.3 START
37 ; OF PROGRAM AFTER LOADING:
38 ; SET SWITCHES TO CONTROL. (3.2).
39 ; ANSWER START ADDRESS QUESTION.
40 ;
41 ; 3.3 A RESTART
42 ; OF PROGRAM THE BEST WAY TO INSURE
43 ; CORRECT SWITCH SETTINGS:
44 ; RESET
45 ; SET SWITCHES TO START ADDR. (3.1).
46 ; EXAMINE
47 ; SET SWITCHES TO CONTROL. (3.2).
48 ; CONTINUE.
49 ; THE START ADDR IS LISTED AT TTY, LPT AND DIS.
50 ;
51 ; 3.3 B RESTART
52 ; OF PROGRAM IF RC 3623 CPU:
53 ; SET DATASWITCH 0, 12 - 15 UP (1)
54 ; SET RESET PARITY ERROR DOWN (ON)
55 ; PRESS AUTOCLOAD AND RELEASE
56 ; SET RESET PARITY ERROR UP (OFF)
57 ; SET SWITCHES TO CONTROL (3.2).
58 ; ANSWER THE QUESTION WRITTEN ON TTY/OCP.
59 ;
60 ; 3.3 C POWER RESTART
61 ; OF PROGRAM ONLY IF POWER MONITOR OPTION:
62 ; SET KEY IN LOCK BEFORE REMOVING POWER
63 ; AFTER POWERING UP THE PROGRAM WRITES:
64 ; POWER
65 ; ACTUAL PROGRAM NAME
66 ; SET SWITCHES TO CONTROL. (3.2).
67 ; ANSWER START ADDRESS QUESTION.

```

31 ;
 32 ;
 33 ;
 34 ;
 35 ;
 36 ;
 37 ;
 38 ;
 39 ;
 40 ;
 41 ;
 42 ;
 43 ;
 44 ;
 45 ;
 46 ;
 47 ;
 48 ;
 49 ;
 50 ;
 51 ;
 52 ;
 53 ;
 54 ;
 55 ;
 56 ;
 57 ;
 58 ;
 59 ;
 60 ;
 61 ;
 62 ;
 63 ;
 64 ;
 65 ;
 66 ;
 67 ;

3.4 LOADING MESSAGE:

AFTER LOADING THE PROGRAM WILL WRITE FOLLOWING:
 SWITCHES: 120073 ;THIS IS THE INITIAL STATE OF SW
 CPU TYPE: 000016 ;THIS IDENTIFIES WHICH CPU/MEM
 LAST LOC: 077777 ;THIS INFORMS THE USER; MEM-SIZE
 BINARY LOADER OK ;THIS IS THE TS VERSION PLACED
 AT LAST LOC (MAX 32K) BUT SLIGHTLY MODIFIED AND
 INCL ROOTSTAP. ERRORHALT XX7752
 IF NOT SELFSTARTING PROG: READYHALT XX7676
 ACTUAL PROG NAME ;THIS IS THE LOADED PROGRAM.
 SET SWITCHES TO CONTROL, (3.2), STARTADDR 400 ?

3.5 START WITH BREAK OPTION (RC 3603).

RESET
 SET SWITCHES TO HALT INSTR 063077
 SET REGISTER SELECT TO 6
 DEPOSIT INTO REGISTER
 SET SWITCHES TO ADDRESS 000203
 SET REGISTER SELECT TO 5
 DEPOSIT INTO REGISTER
 SET BREAK SWITCH TO ON
 GO TO 3.3 RESTART WITH SA = 1400

3.6 CPU NO

FOR TIMING PURPOSE THE PROGRAM DETERMINES IN WHICH CPU IT RESIDUES. IF IT FAILS THE PROGRAM WILL TRY TO CONTINUE AFTER THE MESSAGE "MISERABLE TIMING". IF IMPOSSIBLE THE PROGRAM WILL ASK YOU TO IDENTIFY THE CPU WITH A NUMBER BETWEEN 0 AND 6. USE 2 FOR NOVA1200, RC3603-BREAK
 4 FOR RC3603
 5 FOR NOVA 2-16K
 6 FOR NOVA 2-8K WHERE
 THE MEMORY TYPE FOR THE FIRST 8K IS RELEVANT:
 RESET
 DEPOSIT # INTO AC2
 SET SWITCHES TO CONTROL (3.2).
 CONTINUE.
 ...OR CORRECT PROGRAM, READ MESS IN END OF LOCKA.

4. OPERATING PROCEDURE

LOAD THE PROGRAM AND ANSWER THE QUESTIONS GIVEN. IF THE ANSWER IS BEYOND LIMITS THE QUESTION IS REPEATED. IF THE SUGGESTED ANSWER IS OK, ANSWER ONLY NL (RETURN). THE LAST DIGIT ANSWERED CAN BE REMOVED WITH RUBOUT, ELSE IF YOU TYPED WRONG THE QUESTION IS REPEATED BY GIVING DIGITS UNTIL LIMIT IS PASSED.

TESTING THE PAPER TAPE PUNCH.

1. PUNCH DIAGNOSTIC - SA 400
 ALLOW THE DIAGNOSTIC TO RUN FOR AT LEAST ONE PASS. THE PROGRAM WILL TYPE A PASS MESSAGE AT THE COMPLETION OF EACH PASS AT TTY/OCP/LPT.
2. PUNCH COUNTER - SA 403
 ALLOW THE PUNCH TO PUNCH FOR AT LEAST 10 MINUTES.
3. READ COUNTER - SA 404
 REMOVE THE TAPE FROM THE PUNCH AND INSERT THE TAPE IN THE READER. THE TTY READER MAY BE USED IF THE SYSTEMS DOES NOT CONTAIN HIGH SPEED READER. THIS ROUTINE CHECK READ THE COUNTER.
- (4. PUNCH FROM SWITCHES - SA 402, IF ERROR.)
 THE SWITCHES 8-15 IS PUNCHED. IF THE TECHNICAN PANEL NOT IS CONNECTED, THE PUNCH DEPEND OF SW0:
 SW0: 0 SW 12-15 ARE PUNCHED AT CHANNEL 4-1
 SW0: 1 SW 12-15 ARE PUNCHED AT CHANNEL 8-5.

```

01 ;
02 ;
03 ; TESTING THE PAPER TAPE READER.
04 ; 1. READER DIAGNOSTIC - SA 401
05 ; LOAD THE READER WITH A CONTINUS LOOP OF TAPE WITH
06 ; ALTERNATING CHARACTORS OF ALL ONES AND ALL ZEROS.
07 ; (A TAPE MAY BE OBTAINED BY STARTING THE ALTERNATE
08 ; PUNCH ROUTINE - SA 405.)
09 ; ALLOW THE READER DIAGNOSTIC TO RUN FOR AT LEAST ONE
10 ; PASS, THE PROGRAM TYPE A PASS MESSAGE AT THE END OF
11 ; EACH PASS AT TTY/CCP/LPT.
12 ; *NOTE: IF THE PATTERN IN THE PAPER TAPE SPLICE NOT IS
13 ; CORRECT, IS IT POSSIBLE TO GET MISLEADING ERRORS.
14 ; 2. ALTERNATE READ - SA 406
15 ; ALLOW THE READER TO CHECK THE TAPE FOR AT LEAST
16 ; 5 MINUTES.
17 ;
18 ; CHANGE OF DEVICE CODE - SA 7600.
19 ; THE PROGRAM ASK FOR THE NEW DEVICE CODE ASSIGNED
20 ; THE READER (EVEN, DEFAULT 12).THE PUNCH HAS ALWAYS DEVICE
21 ; CODE EQLU TO THE READERS DEV. CODE + 1.
22 ; THE ROUTINE CHANGE THE DEVICE CODES IN ALL I/O-
23 ; INSTRUCTIONS TO READER/PUNCH.
24 ;
25 ;
26 ; IN CASE OF ERROR THE PROGRAM WILL PRINT THE PC AND AC'S.
27 ; THEN THE FURTHER FLOW WILL DEPEND OF THE SWITCHES.
28 ; A PASS MESSAGE IS PRINTED WITH 5 MINUTES INTERVAL.
29 ; IF THE PROGRAM DESTROYES ITSELF OR IN ANOTHER WAY
30 ; DISBEHAVE THERE IS A SUPERVISING METHOD BUILT IN: ALL
31 ; LOOPS ARE STARTED WITH JMP .+1. BY HAND IT IS POSSIBLE TO
32 ; SHORTEN THE PROGR., THEREBY NOT RUN ALL LOOPS BY CHANGING
33 ; TO HALT OR JMP *RTEST (CAUSES PASS MESSAGE F. LOOPS UP TO
34 ; THIS COMMAND). AUTOMATICALLY IT'S POSSIBLE TO INSERT HALT
35 ; TO INSERT A CALL TO A ROUTINE, WHICH PRINTS PC LOOP ADDR,
36 ; TO INSERT BACK THE JMP .+1 IN THE BREAKPOINTS CHOSEN BY
37 ; THE PROGRAMMER (TABLE TROTB) BY CALLING STARTADDR 2214,
38 ; 2216, 2220.
39 ;
40 ; THE PROGRAM CONTAINS AN ASCII PRINTER TABLE. IF THE
41 ; PRINTER HAS ANOTHER DRUM ALPHABETH READ FOLLOWING:
42 ;
43 ; 4.1 PRINTER ALPHABETH CHANGE:
44 ; AT ANY TIME AFTER LOADING THIS PROGRAM IT IS POSSIBLE TO CHANGE
45 ; THE ALPHABETH USED ON THE LINEPRINTER. THERE ARE 2 METHODS:
46 ; 4.1.1 GET ONE OF THE BUILD-IN ALPHABETHS:
47 ; START PROGRAM IN SA 2222
48 ; REMEMBER SWITCHES TO CONTROL. (3.2).
49 ; ANSWER ALPHABETH #, SEE EXISTING BELOW
50 ; THE PROGRAM WILL RESTART AFTER CHANGING THE ALPHABETH.
51 ; ANSWER NEXT START ADDRESS.
52 ; 4.1.2 GET AN -ADD ON- TAPE WITH
53 ; A NEW ALPHABETH AND LOAD THIS UPON THE PROGRAM AND RESTART. THE
54 ; TAPES WITH PRINTER TABLE ALPHABETH ARE DESCRIBED BELOW:
55 ; EXISTING: #1 44-RT 535 ASCII
56 ; #2 44-RT 529 RC STANDARD TYPE 71/78 STARTING .
57 ; #3 44-RT 532 RC STANDARD TYPE 71/78 STARTING 2
58 ; (SKEWED 4 POSITIONS)
59 ; #4 44-RT 1213 PL 1, TYPE 70

```

```

01
02
03 ;HOW TO PRODUCE A NEW TABLE;
04 ;THE TABLE HAS 200 OCTAL (128 DECIMAL) BOXES. INPUT KEY
05 ;IS THE ASCII VALUE OF THE CHARACTER TO PRINT ADDED TO 2000.
06 ;THE RESULT IS ADDRESS OF A BOX. EACH BOX OCCUPIES A CORE WORD.
07 ;IT IS BUILT UP OF TWO FIGURES. THE FIRST IS THE CLASS OF THE
08 ;CHARACTER TO BE PRINTED; 0 FOR PRINT, 6 FOR BLIND. THE SECOND
09 ;IS THE CHARACTER VALUE AT THE PRESENT PRINT DRUM. BELOW
10 ;VALUE 40 OCTAL FOLLOWING CHARACTERS MAY BE USED; 11 TAB,
11 ;12 LF, 14 FF AND 15 CR. ALL OTHERS BELOW 40 WILL GIVE SPACE.
12 ;THE FIRST BOX SHOULD CONTAIN THE VALUE FOR THE NULL CHAR
13 ;AND THE LAST THE VALUE FOR THE DEL CHAR, WHICH BOTH NORMALLY
14 ;ARE BLIND. IF YOU COUNT 0,1,2, ,7,10, , THE BOX 101 SHALL
15 ;CONTAIN THE PRINT DRUM VALUE FOR AN A. IF THE DRUM DO NOT
16 ;HAVE SMALL LETTERS, FILL IN THE VALUE FOR BIG ONES. NOW PUNCH
17 ;AN ASCII TAPE LIKE THIS;
18
19 ; .LOC 2020
20 ; .RDX 8 ;WHICH RDX YOU WANT
21 ; .TXTE?
22 ; <6><2> ; (2000) FIRST BOX, BOX 0
23 ; <6><2>
24 ; .
25 ; .
26 ; <3><121> ; (2101) BOX 101 FOR A. FOR ASCII DRUM
27 ; . ;121 IS USED, FOR TYPE 71 137 IS USED.
28 ; .
29 ; .
30 ; <6><0>? ; (2177) BOX 177, LAST
31 ; .RDX 8
32 ; .END 101
33 ;PRODUCE A BINARY TAPE AND LOAD THIS TO MEMORY WITH
34 ;BINARY LOADER AFTER LOADING OF MAIN PROGRAM.
35
36
37 ;5. PROGRAM DESCRIPTION
38
39 ; 5.1 TESTLOOP FAILURE RATE
40 ; 5.2 STRUCTURE OF PROGRAM, NEXT PAGE.
41
42 ; 5.1 TESTLOOP FAILURE RATE:
43 ;THERE ARE TWO DIFFERENT WAYS TO USE THE ROUTINES FOR TESTLOOP:
44 ;SINGLE OR MULTIPLE ERRORHALT:
45
46 ;SINGLE: SETP1 MULTIPLE: SETP1
47 ; ERRORHALT ERRORHALT
48 ; LOOP ERRORHALT
49 ; ERRORHALT
50 ; LOOP
51
52 ;IN CASE OF A CONSTANT ERROR THE RATE WILL BE PRINTED
53 ;THIS WAY:
54 ;PC XXXXXX 100 % PC XXXXXX 300 %
55 ;WERE THE LAST IS A MULTIPLE OF 100 %.
56
57 ;THERE ARE THREE POSSIBILITIES FOR THE NUMBER OF LOOPS IN A
58 ;CYCLE, I. E. HOW MANY TIMES THE PROGRAM RUNS THROUGH THE
59 ;INSTRUCTIONS BETWEEN SETP1 AND LOOP. IF SWITCH 0 IS 0 FIRST
60 ;TIME AN ERROR IS DETECTED THE PROGRAM PROCEEDS TO NEXT
61 ;INSTRUCTION AFTER LOOP.
62

```

```

01 ;
02 ;
03 ;NUMBER OF LOOPS:      SETP0: 1
04 ;                      SETP1: 10
05 ;                      SETP2: 100
06
07 ;IF THE ERROR IS NOT CONSTANTLY IT IS POSSIBLE TO SEE THESE
08 ;FAILURE RATES IF SINGLE OPERATION:
09
10 ;SETP2: 100 % ERROR IN THE ONE LOOP.
11 ;SETP1: 100,52,33,25,20,16,14,12,11,10 % FOR ERROR IN
12 ;      1, 2, 3, 4, 5, 6, 7, 8, 9, 10 -TH LOOP.
13 ;SETP2: LIKE SETP1, ONLY ADD 9...1 % FOR ERROR IN
14 ;      11...100 -TH LOOP.
15
16 ;IF MULTIPLE OPERATION THE RATES DEPENDS ON HOW MANY OF THE
17 ;ERRORS ARE FOUND IN THE FIRST LOOP WITH ERROR:
18 ;      100 % COULD BE 1 ERROR IN FIRST LOOP
19 ;      OR      2 ERRORS IN SECOND LOOP
20 ;      OR      3 ERRORS IN THIRD LOOP ETC.
21
22
23 ;IF SWITCH 2 IS 1 THE PROGRAM WILL REMAIN IN THE LOOP WITH
24 ;ERROR. FIRST TIME AN ERROR IS SEEN THE PROGRAM WILL HALT
25 ;(IF NOT SWITCH 12). THEN THE PROGRAM CONTINUES UNTILL ALL
26 ;NUMBER OF LOOPS ARE PERFORMED. THEN A NEW CYCLE IS
27 ;ENTERED CALLED ERRORCYCLE. AFTER EACH ERRORCYCLE A NEW IS
28 ;STARTED UNTILL SWITCH 2 IS SET TO 0.
29
30 ;IN ALL CYCLES (FIRST OR ERROR) THE PC (PROGRAM COUNTER OF ERROR)
31 ;IS WRITTEN FIRST TIME AN ERROR IS SEEN AND IN MULTIPLE OPERATION
32 ;MORE THAN ONE ERRORHALT COULD WRITE THE PC. BUT ONLY WITHIN
33 ;THAT FIRST LOOP WITH ERROR. THE FAILURE RATE IS PRINTED WHEN
34 ;THE CYCLE IS FINISHED. THE PROGRAM ONLY HALTS IN THE FIRST CYCLE
35 ;(DEPENDING ON SW 12). THE ERRORCYCLES HAVE ANOTHER AMOUNT OF
36 ;LOOPS THAN FIRST CYCLE:
37
38 ;NUMBER OF LOOPS:      FIRST CYCLE      ERROR CYCLE
39 ;                      SETP0: 1          10
40 ;                      SETP1: 10         100
41 ;                      SETP2: 100        100
42
43 ;IN SINGLE OPERATION FAILURE RATE IS TELLING HOW OFTEN THE ERROR
44 ;APPEARS. IN MULTIPLE OPERATION FAILURE RATE SHOULD BE USED
45 ;VERY CAREFULLY.
46
47 ;
48 ;
49 ;      5.2 STRUCTURE OF PROGRAM:
50 ;      THE PROGRAM IS A NORMAL DIAGNOSTIC PROGRAM, I. E. IN
51 ;      CASE OF ERROR THE PROGRAM WILL HALT WITH AC3 = PC. THE
52 ;      OPERATOR SHOULD THEN LOOK UP IN THE LISTING TO KNOW WHAT
53 ;      WAS CAUSING THE ERRORHALT. THE PROGRAM INCLUDES ONLY FEW
54 ;      SUBROUTINES TO MAKE IT EASIER FOR THE OPERATOR.
55 ;      THE PROGRAM ALSO CONTAINS NORMAL TESTLOOP FACILITIES, SEE
56 ;      5.1. IN CASE OF SEVERAL ERRORHALTS ATTENTION SHOULD ONLY
57 ;      BE GIVEN TO THE FIRST (THE LOWEST PC) BECAUSE THE REST
58 ;      COULD BE DERIVED FROM THE FIRST. TIMING IN A LOOP IS
59 ;      AFFECTED BY PRINTING FIRST ERROR, TOO.

```


21 ;
 22 ;
 23 ;
 24 ;
 25 ;
 26 ;
 27 ;
 28 ;
 29 ;
 30 ;
 31 ;
 32 ;
 33 ;
 34 ;
 35 ;
 36 ;
 37 ;
 38 ;
 39 ;
 40 ;
 41 ;
 42 ;
 43 ;
 44 ;
 45 ;
 46 ;
 47 ;
 48 ;
 49 ;
 50 ;
 51 ;
 52 ;
 53 ;
 54 ;
 55 ;
 56 ;
 57 ;
 58 ;
 59 ;
 60 ;
 61 ;
 62 ;
 63 ;

THE PROGRAM WILL TEST THE PUNCH WITH AID OF FOLLOWING TEST LOOPS:

- A20 CHECK SELB BUS LINE.
- A22 CHECK SELD BUS LINE.
- A23 CHECK DEV. SELECTION.
- A17 CHECK DONE FLOP ZERO.
- A20 CHECK, WITH DONE FLOP ZERO, INTA
- A21 CHECK START BUS LINE.
- A24 CHECK BUSY SETS VIA A START PULSE.
- A26 CHECK CLEAR BUS LINE.
- A28 CHECK DEVICE SELECTION.
- A30 CHECK CLEAR BUSY PUNC.
- A32 " " " " I/O RESET.
- A36 CHECK THAT A START PULSE CLEAR DONE FF.
- A38 " " " CLEAR " " " " " "
- A40 CHECK THAT PUN COMPLEAT CLEAR BUSY.
- A42 " " " " " " " "
- A44 " " " " " SET DONE.
- A46 CHECK INTERRUPT IF DONE=1 AND INT. DIS. =0.
- A48 CHECK THAT INT. DIS =1 PREVENT INTERRUPT.
- A50 CHECK FOR ALL INTERRUPTS.
- A52 CHECK THAT I/O RESET CLEAR PUNCH.
- A54 CHECK INTERRUPT DIS. FLOP.
- A56 CHECK I/O-RESET CLEAR INT. DIS. FF.
- A58 CHECK, WITH INT. DIS=1, INTA.
- A60 CHECK INTERRUPT ACK.
- A62 " " " "
- A64 " " " "
- A66 CHECK THAT DONE NOT SET VIA PUN COMP. IF BUSY IS ZERO.
- A68 CHECK THAT THE PUNCH CAN PUNCH 50 CHAR (NORM. 75 CPS) IN LESS THAN 1 SEC. DO. WITH STOP MOTOR (5 SEC WAIT).
- A82 LOOP TO REPORT ANY CHANGE IN THE STATUS BIT: PAPER TAPE LOW/OK. IN THE FIRST RUN IS THE STATUS ALWAYS REPORTED:
 1: TAPE LOW
 2: TAPE OK.

THE PROGRAM WILL CHECK THE READER WITH AID OF FOLLOWING TEST LOOPS:

- R20 CHECK PTR BUSY FLOP IS ZERO.
- R22 " " DONE " " "
- R24 CHECK A START PULSE TO DEV. 3 NOT SET BUSY FLOP.
- R26 CHECK BUSY NOT SET VIA NIO PTR.
- R28 CHECK FOR NO INTERRUPTS, PTR INT DISABLE.
- R10 " " " " " INT ENABLE.
- R11 CHECK BUSY CLEAR IN LESS THAN 100 MSEC.
- R12 CHECK START BUS LINE.
- R14 CHECK I/O RESET CLEAR BUSY.
- R16 CHECK RD SELECT AND A CLEAR PULSE CLEAR BUSY.
- R18 CHECK DEVICE SELECTION.
- R20 " " " "
- R21 " " " "
- R36 CHECK IORST CLEAR DONE.

```

21 ;
22 ;
23 ; R38 CHECK A CLEAR PULSE AND A RD SEL. CLEAR DONE.
24 ; R40 " " START " " " " " "
25 ; R42 CHECK BUSY SET CAUSE DONE SETS.
26 ; R44 CHECK DONE SET CAUSE A INTERRUPT.
27 ; R46 CHECK INTERRUPT DISABEL FLOP.
28 ; R48 CHECK INT. DIS. FF CLEAR VIA IORST.
29 ; R50 CHECK INT DIS FF SET PREVENT INTERRUPT.
30 ; R52 " " " " " "
31 ; R54 CHECK I/O RESET ABILITY TO CLEAR RD INT REG FLOP.
32 ; R56 CHECK INTERRUPT ACK.
33 ; R58 " " " "
34 ; R60 " " " "
35 ; R62 CHECK DIA TO RD NOT READ ONLY ZEROS.
36 ; R64 CHECK DEV. SELECTION.
37 ; R68 CHECK FOR READ CHANNELS STUCK TO 0/1.
38 ; R70 CHECK DONE NOT SET IF BUSY ZERO.
39 ; R72 CHECK THE READ SPEED.

```

16. SOFTWARE SURVEY.

```

23 ;
24 ; 6.1 DIAGNOSTIC.
25 ; THE READER AND PUNCH DIAGNOSTIC CONSIST OF NUMERIOUS
26 ; SMALL ROUTINES. EACH ROUTINE BEGINS WITH A INITIALIZING
27 ; ROUTINE (SETP1) AND ENDS WITH A ITERATION ROUTINE (LOCP).
28 ; BOTH THE SETUP AND LOOP ROUTINES ISSUE A I/O RESET (ICRST
29 ; PULSE. SYNCING ON I/O RESET WILL PERMIT THE LOGIC UNDER
30 ; TEST BE OBSERVED.
31 ;
32 ; 6.2 READ/PUNCH ROUTINES.
33 ; IN ALL THE READ/PUNCH ROUTINES ARE A RANDOM DELAY.
34 ; IN THE PUNCH ROUTINES IS IT OPTIONALLY. THIS DELAY
35 ; IS USED TO PRODUCE VARIOUS START/STOP CONDITIONS
36 ; TO THE READER AND THE PUNCH.
37 ; WITH THE MASK IN LOC 12256, IS IT POSSIBLE TO
38 ; INHIBIT ERRONEOUSLY BIT POSITIONS.
39 ; 6.2.1 SWITCH PUNCH.
40 ; THE PUNCH FROM SWITCHES ROUTINE IS PROVIDED SUCH
41 ; THAT THE OPERATOR MAY SCOPE THE VARIOUS SIGNALS
42 ; IN THE PUNCH AND PUNCH CONTROLLER. IT IS POSSIBL
43 ; TO PUNCH FROM SWITCHES EVEN IF THE TECHNICIAN
44 ; PANEL NOT IS CONNECTED.
45 ; 6.2.2 PUNCH ALTERNATE ROUTINES.
46 ; THE PUNCH ALTERNATE ROUTINE IS PROVIDED TO
47 ; PRODUCE A TAPE FOR THE READER DIAGNOSTIC.
48 ; 6.2.3 COUNTER ROUTINES
49 ; THE PUNCH COUNT ROUTINE IS THE PRIME TEST OF THE
50 ; PUNCHS ABILITY TO PUNCH DATA. A COUNT PATTERN IS USED
51 ; SO DOUBLE PUNCHING AND FAILURE TO PUNCH MAY
52 ; BE OBSERVED. THE READ COUNT IS PROVIDED TO CHECK
53 ; THE TAPE PUNCHED BY THE PUNCH COUNT ROUTINE.
54 ; 6.2.4 READ ALTERNATE ROUTINE.
55 ; THE READ ALTERNATE ROUTINE IS PROVIDED TO CHECK
56 ; THE READERS ABILITY TO READ CHARACTORS OF ALL
57 ; ONES AND ALL ZEROS UNDER VARIOUS START/STOP
58 ; CONDITIONS.
59 ; 6.2.5 FLOATING ZERO PATTERN ROUTINES.
60 ; THESE PUNCH/READ ROUTINES ARE PROVIDED TO
61 ; CHECK BOTH THE PUNCH AND THE READER UNDER
62 ; VARIOUS CONDITIONS.
63 ;
64 ; 6.3 CHANGE OF DEVICE CODES.
65 ; THE ROUTINES CHANGE THE DEVICE CODES IN THE
66 ; I/O INSTRUCTIONS TO PAPER TAPE READER/PUNCH.
67 ;

```

21 ;
 22 ;
 23 ;
 24 ;
 25 ;
 26 ;
 27 ;
 28 ;
 29 ;
 30 ;
 31 ;
 32 ;
 33 ;
 34 ;
 35 ;
 36 ;
 37 ;
 38 ;
 39 ;
 40 ;
 41 ;
 42 ;
 43 ;
 44 ;
 45 ;
 46 ;
 47 ;
 48 ;
 49 ;
 50 ;
 51 ;
 52 ;
 53 ;
 54 ;
 55 ;
 56 ;
 57 ;
 58 ;

7. MESSAGES FROM THIS TEST
 ALL MESSAGES ARE WRITTEN ON TTY, LPT AND THE 16 CHAR DIS
 ON OPERATORS CONTROL PANEL. IT IS POSSIBLE TO ANSWER
 QUESTIONS AT TTY OR NUK, NUMERIC KEYBOARD ON OPERATORS
 CONTROL PANEL.
 ABOUT THE NUMBERS:
 0-5 DIGITS IS A DECIMAL NUMBER
 RANGE -32768 TO -1 AND 0 TO 32767
 6 DIGITS IS AN OCTAL NUMBER.
 2-6 DIGITS IS AN OCTAL NUMBER WITH
 LEADING ZEROS SUPPRESSED, DON'T USE.
 8 DIGITS IS A BINARY NUMBER.

MESSAGES: LPT/TTY & DIS, IF SPEC DIS MESSAGE: ()
 MISERABLE TIMING, RUN RC 3600 INSTRUCTION TIMER TEST
 (MISERABLE TIMING) TIMING WITH BIG TOLERANCES, CPU
 TYPE IS GUESSED.
 RTC IS UNSTABLE, RTC NOT RUNNING WITH CONSTANT
 SPEED.
 SET CPU NO > AC2 HELP THE PROGRAM TO IDENTIFY CPU,
 SEE 3.5
 NO MESS AT DIS ABOUT SW. POSITION

SWITCHES: 100073
 CPU TYPE: 200016
 LAST LOC. 077777
 BINARY LOADER OK
 REDR/PUNCH TEST IDENTIFIKATION
 SET SWITCHES TO CONTROL, (3.2), STARTADDR 400 ?

SWITCHES: 200000 NO MESS AT DIS ABOUT SW. POSITION
 200000 STARTADDR
 SWITCHES: 170000 NO MESS AT DIS ABOUT SW. POSITION
 PRINT INHIBIT SWITCH SET.

DIAGNOSTIC ROUTINES.
 1. PASS OF 10 RUNS WITH/WITHOUT ERRORS
 (1. PASS 10 R)
 037500 037503 000000 ERROR MESSAGE
 ACC AC1 AC2
 PC 007464 100 %
 (PC 007464 100 %) EXAMINE FOR AC'S AFTER HALT.

PUNCH DIAGNOSTIC FURTHER:
 TAPE LOW
 TAPE OK SEE SECTION 5.2.

READ ROUTINES.
 GOOD= XXX BAD= XXX GOOD IS THE EXPECTED OCTAL VALUE
 BAD THE READ OCTAL VALUE.

1 2311 ,MAIN

21

22

PAGE ZERO FOR TAPE 2,3,4,5

23

24
25

000000 ,LOC 0

26 00000 023160

2*MELOC

;MESS AFTER RDCS LOAD AND STORE PC ON INT

27 00001 010206

INTPT

;ADDR. OF INTER. ROUTINE.

28 00002 001400

REBIN

;SELFSTART ADDR FOR RDCS ETC.

29 00003 000000

2

;0=HALT, 1=SELFSTART PROG AFTER REBIN

10 00004 000000

2

;0ADDR FOR SELFSTART PROG AFTER REBIN

11 00005 000000

2

;FOR LOAD RDCS, USED BY POW, INTR, FITYP

12

13

000020 ,LOC 20

14

15 00020 000000 IX0:

2

;AUTO INCREMENT LOCATION

16 00021 000000 IX1:

2

;AUTO INCREMENT LOCATION

17 00022 000000 IX2:

2

;AUTO INCREMENT LOCATION

18 00023 000000 IX3:

2

;AUTO INCREMENT LOCATION

1 0012 ,MAIN

01
02 000240 ,LOC 40
03
04 00040 000412 IMESS: XMESS
05 00041 000664 ICHAR: XCHAR
06 00042 000724 ITYPE: XTYPE
07 00043 001243 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 001214 ITBIN: XTBIN
14 00052 000563 ITOCT: XTOCT
15 00053 000470 ITDEC: XTDEC
16 00054 000550 ITZCC: XTZCC
17 00055 001020 IOBIN: XOBIN
18 00056 000564 IDOCT: XDOCT
19 00057 000464 IDDEC: XDDEC
20 00060 000554 IDZCC: XDZCC
21 00061 002427 IWAIT: XWAIT
22 00062 001244 INADP: XADP
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 002211 IQUES: XQUES
30 00072 001735 ISAMS: XSAMS
31 00073 001312 IPRESH: XPRESH
32 00074 000200 HXEND: 2
33 00075 000200 DIGIN: 2

;INDIRECT ADDRESSES
;NOT IN AUTO INC,DEC LOC.

;TOP OF MEMORY, LOADER PROTECT
;INPUT BUFFER FOR INPUT ROUTINES.

;PRINTER TABLE HANDLING AND (POWER) RESTART:

34
35
36
37 000276 ,LOC 76

38
39 00076 002100 PCARE: JMP @PCARE ;INSTRUCTION TO BE STORED IN CELL ZERO
40 00077 004740 IPRESA: SAISA ;PROGRAM RESTART ADDR,
41 00100 004662 PCARE: PCACN ;POWER RESTART ADDR
42 00121 003277 PRINT: HALT ;IMPORTANT TO KEEP THIS AND NEXT IN
43 00122 002277 STOP: JMP @IPRESA ;101,102 BECAUSE PRINTER TABLE SELFSTART
44
45 00103 000052 IGTBI: GETBI
46 00104 0005731 IGTCK: GETCK
47 00105 0005426 IGTCC: GETCC
48 00106 0005643 IGTSC: GETSC
49 00107 0006161 IGTTX: GETTX
50 00110 0006416 ISTR0: ENTP0
51 00111 0006423 ISTR1: ENTP1
52 00112 0006433 ISTR2: ENTP2
53 00113 0006447 ILCOP: CYCLE
54 00114 0006617 IHALT: ERROR
55 00115 0006720 ISTAA: XSTAA
56 00116 0006734 ISTAN: XSTAN
57 00117 0006744 ISTAW: XSTAW
58 00120 0006776 ISTAS: XSTAS
59 00121 0006761 ISTAR: XSTAR
60 00122 0007006 ILCRE: XLCRE
61 00123 0007024 IPASS: XPASS

1 2013 .MAIN

31				
32	026240	CMESS=JSR	@IMESS	;EACH DEFINITION BELOW CORRESPONDS
33	026241	CCHAR=JSR	@ICHR	;WITH A CALL OF A ROUTINE.
34	026242	CTYPE=JSR	@ITYPE	
35	026243	CCRLF=JSR	@ICRLF	
36	026244	CDISP=JSR	@IDISP	
37	026245	CDOUT=JSR	@IDOUT	
38	026246	CDICL=JSR	@IDICL	
39	026247	CDATT=JSR	@IDATT	
40	026250	CHAAT=JSR	@IHAAT	
41	026251	CTBIN=JSR	@ITBIN	
42	026252	CTOCT=JSR	@ITOCT	
43	026253	CTDEC=JSR	@ITDEC	
44	026254	CTZOC=JSR	@ITZOC	
45	026255	COBIN=JSR	@IDBIN	
46	026256	COOCT=JSR	@IDOCT	
47	026257	CODEC=JSR	@IDDEC	
48	026260	COZOC=JSR	@IDZOC	
49	026261	CWAIT=JSR	@IWAIT	
50	026262	WAOP=JSR	@IWAOP	
51	026263	TIMSK=JSR	@ITISK	
52	026264	TIMMS=JSR	@ITIMS	
53	026265	TIMRO=JSR	@ITIRO	
54	026266	MULTI=JSR	@IMULT	
55	026267	DIVIS=JSR	@IDIVS	
56	026270	DIVID=JSR	@IDIVD	
57	026271	COUES=JSR	@IGUES	
58	026272	CSAMS=JSR	@ISAMS	
59	026273	CRSW=JSR	@IRESW	
60	026123	CGTBI=JSR	@IGTBI	
61	026124	CGTOK=JSR	@IGTOK	
62	026125	CGTDC=JSR	@IGTDC	
63	026126	CGTSC=JSR	@IGTSC	
64	026127	CGTTX=JSR	@IGTTX	
65	026110	SETP0=JSR	@ISTP0	
66	026111	SETP1=JSR	@ISTP1	
67	026112	SETP2=JSR	@ISTP2	
68	026113	LOOP=JSR	@ILOOP	
69	026114	EHALT=JSR	@IHALT	
70	026115	STATA=JSR	@ISTAA	
71	026116	STATN=JSR	@ISTAN	
72	026117	STATW=JSR	@ISTAW	
73	026120	STATS=JSR	@ISTAS	
74	026121	STATP=JSR	@ISTAP	
75	026122	CLORE=JSR	@ILORE	
76	026123	CPASS=JSR	@IPASS	

1 0014 ,MAIN

01
22 00124 000000 DELAY: 0
23 00125 000000 RC2: 0
24 00126 000000 HSPUN: 0
25 00127 000000 HSREAD: 0
26 00130 000000 TEKPA: 0
27 00131 000001 C1: 1
28 00132 000012 CPTR.: PTR.
29 00133 000013 CPTP.: PTP.
10 00134 000040 C40: 40
11 00135 000000 XORDEV: 0

12
13 ; DEFINITIONS

14
15 ;TTI=10
16 ;TTO=11
17 ;RTC=14
18 ;LPT=17

19 000032 FUN=32
20 000033 FUB=33
21 000034 NUK=34
22 000035 DIS=35
23 000017 XLPT=LPT
24 000010 XTTI=TTI
25 000011 XTTO=TTO
26 000014 XRTC=RTC
27 000017 DEV=XLPT
28 000012 PTR.=12
29 000013 PTP.=13

;DEFINE DEVICE TO TEST FOR STATUS
;DEFAULT READER DEVICE CODE.
;DEFAULT PUNCH DEVICE CODE.

30
31 00136 007257 IPTEST: FUNTE
32 00137 007267 IRTEST: RECTE
33 00140 007300 IPSW: PSWIT
34 00141 007307 IPUNC: PCHCN
35 00142 007316 IREDC: RCOLN
36 00143 007324 IALTP: RALT
37 00144 007333 IALTR: RALT
38 00145 007341 IPUNF: PFLTZ
39 00146 007347 IREDF: RFLTZ
40 00147 007355 IPREST: RESTR
41 00150 000000 RTEST: 0

;RETURN ADDR FROM TESTLOOPS

1 0015 ,MAIN

01

02 000400 ,LOC 400

03 00400 026136 JSR

*IPTEST ;PUNCH DIAGNOSTIC

04 00401 026137 JSR

*IRTEST ;READER DIAGNOSTIC

05 00402 026140 JSR

*IPSW ;PUNCH FROM SWITCHES

06 00403 026141 JSR

*IPUNC ;PUNCH COUNTER

07 00404 026142 JSR

*IREDC ;READ COUNTER

08 00405 026143 JSR

*IALTP ;ALTERNATE PUNCH

09 00406 026144 JSR

*IALTR ;ALTERNATE READ

10 00407 026145 JSR

*IPUNF ;PUNCH FLOATING ZERO

11 00410 026146 JSR

*IREDF ;READ FLOATING ZERO

12 00411 026147 JSR

*IREST ;RESTART, NO QUESTIONS, LAST ANSWERS

13

14

15

16

17

18

;THE STANDARD ROUTINES ARE NOT MODIFIED.

19

;IF MODIFICATIONS, LIST HERE:

20

21

22

;TAPE 1

23

24

,EOT

0016 ,MAIN

```
01
02           ;TAPE 2           OUTPUT ROUTINES AND OTHERS.
03
04           000412 ,LOC 412   ;DC NOT MOVE UP.
05
06           ;FOR THE USE SEE EACH ROUTINE.
07
08 ;MESS      TYPE A TEXT MESSAGE ON TTY AND LPT
09 ;CHAR      TYPE A CHAR ON TTY AND LPT, CALCULATE P-BIT
10 ;TYPE      TYPE A CHAR ON TTY AND LPT
11 ;CR LF     TYPE A CR AND A LF ON TTY AND LPT
12 ;DISP      DISPLAY A TEXT MESSAGE ON DIS
13 ;DOUT      DISPLAY A CHAR ON DIS
14 ;DCL       CLEAR DIS
15 ;DISATT    DISPLAY ATTENTION; BEEP AND WAIT 3 SECONDS
16 ;HAATT     HALT ATTENTION; BEEP
17 ;TBIN      TYPE BINARY NUMBER ON TTY AND LPT
18 ;TOCT      TYPE OCTAL NUMBER ON TTY AND LPT
19 ;TDEC      TYPE DECIMAL NUMBER ON TTY AND LPT
20 ;TZOC      TYPE OCTAL NUMBER WITHOUT LEADING ZEROS.
21 ;DBIN      DISPLAY BINARY NUMBER ON DIS
22 ;DOCT      DISPLAY OCTAL NUMBER ON DIS
23 ;DDEC      DISPLAY DECIMAL NUMBER ON DIS
24 ;DZOC      DISPLAY OCTAL NUMBER WITHOUT LEADING ZEROS.
25 ;WAIT      WAIT SOME MILLISECONDS
26 ;NATORP    WAIT FOR OPERATOR ACTION
27 ;TIMSK     TIMER FOR SKIP INSTRUCTION
28 ;TIMRO     TIMER FOR ROUTINE
29 ;TIMMS     TIME MEASURE ROUTINE
30 ;MULTI     MULTIPLY
31 ;DIVIS     DIVIDE SINGLE
32 ;DIVID     DIVIDE DOUBLE
33 ;QUES      OUTPUT QUESTIONS
34 ;SAMS      START ADDR MESSAGE
35 ;RES*      READ SWITCHES, REPORT CHANGES
36 ;SAPTR     MOVE PRINTER TABLE
37 ;SAMEX     EXTEND TO 64K WORDS MEM MODE
38 ;SAMNM     RESET TO 32K WORDS MEM MODE
39 ;REBIN     INITIALIZE, RESTORE BIN LOADER
40
41
42 ;AN ADD ON PRINTERTABLE SHOULD HAVE THIS FORMAT:
43 ; .LOC 2000
44 ; .RDY Y
45 ; .TXTE?
46 ; <6><2>           ;TOTALLY 200 BOXES WITH
47 ;                   ;<TYPE OF ACTION><CHAR TO PRINT>
48 ; <6><2>?
49 ; .END 101
50 ;
51 ;                   ;TYPE OF ACTION:      2=PRINT
52 ;                                           6=BLIND
```

1 0017 .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 TIO ROUTINES:
08      ;>MESS< PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLER.
09      ;>CHAR< PRINTS ASCII CHARACTER, 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 TAPPOINT.
21      ;>TBIN< PRINTS AC1 IN BINARY, AC1-R 8 DIGITS ONLY, 8 PRINTPOS.
22
23      ;DIS ROUTINES:
24      ;>DISP< PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLER.
25      ;>DOUT< PRINTS ASCII CHARACTER, AC0-R. AC0-L MUST BE 0. RETURN
26      ;TO CALL+1. REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIRED.
27      ;>DIDL< 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      CTDFC
39      ;          CTTOCT
40      ;          CTZOC
41      ;          CDDEC
42      ;          CDOCT
43      ;          CDZOC
44      ;          CTBIN
45      ;          CDBIN
46      ;          MOVSI 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      ;          C-AAT
61      ;          HALT
```

1 0018 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 MESSAGE POINTER+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 0,2,2 ;AC0=DATA WORD
11 00423 125112 MOVL= 1,1,SZC
12 00424 123701 ANDS 1,2,SKP
13 00425 123401 AND 1,2,SKP ;AC0=DATA CHARACTER RIGHT
14 00426 151400 IAC 2,2 ;INC TO NEXT WORD
15 00427 124000 CCM 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,SKR ;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,BZOUT
26 00440 000263 TIMSK ;WAIT IF LPT/TTO BUSY
27 00441 001750 XLPTT: 1200, ;MAX 1 SEC LPT
28 00442 063517 SKPBZ XLPT
29 00443 044776 STA 1,XLPTT ;REMOVE WAITING, LPT NOT CONNECTED
30 00444 000263 TIMSK
31 00445 000454 XTTO: ;MAX 300 MSEC TTO
32 00446 063511 SKPBZ XTTO
33 00447 044776 STA 1,XTTOT ;REMOVE WAITING, TTO NOT CONNECTED
34 00450 002401 JMP @BZOUT
35
36 00451 000000 RBZOT: 0
37 00452 000000 RINHI: 0
38 00453 000000 SADIG: 0
39 00454 000000 RXDEC: 0
40 00455 000240 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 AXDIS ;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 TYPX ;TYPE NEXT DIGIT
56 00473 000404 JMP EXTYP ;EXIT
57
58 00474 054760 TYPX: STA 3,RXDEC
59 00475 004575 JSR YCHAR ;TYPE DIGIT
60 00476 002756 JMP @RXDEC ;NEXT DIGIT, SIGPR OR DECR
61
62 00477 006761 EXTYP: JSR @ZTAB3 ;YPDEC/ZOCT/POCT FINISH RETURN, TYPE TAB
63 00500 004753 EXDIS: LDA 1,SADIG ;DISPLAY FINISHED, RESTORE PARAM
64 00501 002761 JMP @RPOUT ;EXIT
```

1 0019 MAIN

```
01
02 00502 044751 YPDEC: STA 1,SADIG ;SAVE PARAM FOR REPEAT
03 00503 054760 STA 3,PDECR
04 00504 032756 LDA 2,RPOUT ;SPEC RETURN IF PRINT SW
05 00505 024466 JSR PINHI
06 00506 022754 JMP 0,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 2,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 0,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 0,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 0,PDECR ;OUTPUT SIGN
34 00541 020406 LDA 0,CHFLG ;RESTORE ASCII DIGIT
35 00542 002401 JMP 0,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 YZCCT
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 YZCCT
50 00556 000412 JMP NXDIS ;DISPLAY NEXT DIGIT
51 00557 000721 JMP EXDIS ;EXIT
52
53 00560 054702 XT OCT: STA 3,RPOUT ;TYPE OCTAL NUMBER
54 00561 004422 JSR YPCCT
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 YPCCT
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 0,ZCOUT ;DISPLAY DIGIT
65 00572 002662 JMP 0,RXDEC ;NEXT DIGIT
```

```

1 0020 ,MAIN
21
22 00573 054657 PINHI: STA 3,RINH1 ;AFTER ACTIVATING SETPX IN TESTLOOP
23 00574 000403 SETAC: JMP NINH1 ;THIS IS CHANGED TO CRESW (READS 2 ROU)
24 00575 034660 LDA 3,CHINH ;SW 10 FOR NO PRINTING
25 00576 157405 AND 2,3,SAR ;RETURN+1 IF INHIBIT
26 00577 010653 NINH1: ISZ RINH1 ;RETURN+2 IF PRINTING
27 00600 002652 JMP @RINH1 ;EXIT
28
29 00601 027461 YZOCT: LDA 0,CHSP ;ZERO OCTAL ROUTINE
30 00602 101221 MOV 0,0,SKP
31
32 00603 023565 YPOCT: LDA 0,CHAR0 ;OCTAL ROUTINE
33 00604 044647 STA 1,SADIG ;SAVE PARAM FOR REPEAT
34 00605 054656 STA 3,PDECR
35 00606 030654 LDA 2,RPOUT ;SPEC RETURN IF PRINT SW
36 00607 024764 JSR PINHI
37 00610 002652 JMP @RPOUT ;NO PRINT
38 00611 030440 LDA 2,CCTAB ;PRINT AC1 IN OCTAL
39 00612 040735 STA 0,CHFLG ;NO-SIGN FLAG
40 00613 040556 PDEC1: STA 0,ZSUPP ;BOTH ENTRIES PRINT NUMBER
41 00614 050401 STA 2,.*1 ;THEN TAB TO NEXT POSITION
42 00615 003000 DECOCT: 0 ;A LDA 2, TABLE INSTRUCTION
43 00616 010777 ISZ .-1
44 00617 151235 MOV 2,2,SAR ;IF TABLE ENTRY=0 THEN
45 00620 000420 JMP DECEX ;EXIT WITH TAB SPECIAL IF TYPE ROU.
46 00621 034550 LDA 3,ZSUPP ;ZEROS SUPPRESS STUF
47 00622 102400 SUB 2,2
48 00623 146512 DECOT: SUBL# 2,1,SZC
49 00624 000405 JMP DECP
50 00625 146400 SUB 2,1 ;FORM THE DIGIT
51 00626 034542 LDA 3,CHAR0 ;DO NOT SUPPRESS
52 00627 101400 INC 0,2 ;FOLLOWING ZERGES
53 00630 000773 JMP DECOT
54 00631 151235 DECP: MOVZR# 2,2,SAR ;IF LAST DIGIT THEN
55 00632 034536 LDA 3,CHAR0 ;AC3=ZERO, NOT SUPPR CHAR
56 00633 054536 STA 3,ZSUPP ;ACC=0DIGIT
57 00634 163000 ADD 3,0 ;MAKE ASCII
58 00635 004671 JSR SIGN ;TEST SIGN
59 00636 006625 JSR @PDECR ;OUTPUT DIGIT
60 00637 000756 JMP DECOCT ;GET NEXT DIGIT
61 00640 010623 DECEX: ISZ PDECR ;RETURN ADDR FOR JSR Y-ROUTINE
62 00641 002622 JMP @PDECR ;EXIT WITH TAB IF TYPING ROUTINE
63
64 00642 030426 DECT8: LDA 2,.*1+.-DECOCT
65 00643 000012 ,RDX 10
66 00644 023420 10000
67 00645 001750 1000
68 00646 000144 100
69 00647 000012 10
70 00648 000001 1
71 00650 000000 0
72 00651 000010 ,RDX 8
73
74 00651 030435 OCTAB: LDA 2,.*1+.-DECOCT
75 00652 100000 100000
76 00653 010000 10000
77 00654 001000 1000
78 00655 000100 100
79 00656 000010 10
80 00657 000001 1
81 00658 000000 0
82
83 00661 000000 CHRET: 0
84 00662 000240 CHSP: 240 ;LEADING ZERO SUPPRESS CHAR + TAB SIMU
85 00663 000240 CHSPA: 240 ;A SPACE FOR + IN PDEC

```

```

| 0021  MAIN
01
02 00664 040507 XCHAR: STA 0,SACHA ;SAVE PARAM FOR REPEAT
03 00665 054774 STA 3,CHRET
04 00666 171000 MCV 3,2 ;SPEC RETURN IF PRINT SW
05 00667 004704 JSR PIAHI
06 00670 002771 JMP @CHRET ;NO TYPE
07 00671 000402 JMP GCHAR
08
09 00672 054767 YCHAR: STA 3,CHRET ;PRINT ACC RIGHT
10 00673 101320 QCHAR: MCVZS 0,0
11 00674 040500 STA 0,CHSAV
12 00675 176000 CHAR2: ACC 3,3 ;COMPUTE THE PARITY
13 00676 117000 ACC 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 ACDS 3,2
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 000772 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 MCV 3,2 ;SPEC RETURN IF PRINT SW
40 00727 004604 JSR PIAHI
41 00730 002553 JMP @REG3 ;NO TYPE
42 00731 000403 JMP GTYPE
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 CHCRZ ;INC HORIZONTAL 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 000437 JMP XFORM
52 00742 061017 TYPE1: DCA 0,XLPT ;SEND CHAR
53 00743 060117 NJOS 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 SLB# 0,1,SZR ;CR ?
60 00752 000404 JMP TYPE4 ;NO CR
61 00753 006061 TYPE3: C*AIT ;IF CR, LF WAIT FOR
62 00754 001237 SEC#2 ;DATAPOINT 20 MSEC
63 00755 020523 LDA 0,REG0 ;RESTORE ACC
64 00756 061011 TYPE4: DCA 0,XTTO ;SEND CHAR
65 00757 060111 NJOS XTTO ;START TTO

```

1 0022 ,MAIN

```
01
02 00760 152400 TYPES:  SUB      2,2
03 00761 024516      LDA      1,CHLF
04 00762 136415      SUB#    0,1,SAR ;IF LF
05 00763 050412      STA      2,CHORZ ;CLEAR HORZ POS
06 00764 030516      LDA      2,REG2
07 00765 024514      LDA      1,REG1 ;ACC = REG0 FOR REPEAT
08 00766 022515      JMP      @REG3 ;EXIT
09
10 00767 000011 CHTAB:  11
11 00770 000060 CHAR0:  60
12 00771 000200 ZSUPP:  2
13 00772 177770 NK10:  -10
14 00773 000000 SACHA:  2
15 00774 000000 CHSAV:  2
16 00775 000000 CHORZ:  0
17 00776 000007 CHAR7:  7
18 00777 000437 TBZCT:  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 ;AC2:=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
33 01014 054461 XTBIN:  STA      3,CRBIR
34 01015 004407      JSR      YPBIN
35 01016 024654      JSR      YCHAR ;TYPE DIGIT
36 01017 000420      JMP      PBINN ;NEXT DIGIT
37
38 01020 054455 XCBIN:  STA      3,CRBIR
39 01021 004403      JSR      YPBIN
40 01022 004520      JSR      YOCUT ;DISPLAY DIGIT
41 01023 000414      JMP      PBINN ;NEXT DIGIT
42
43 01024 044447 YPBIN:  STA      1,SABIN ;SAVE PARAM FOR REPEAT
44 01025 054447      STA      3,PBINR
45 01026 030447      LDA      2,CRBIR ;SPEC RETURN IF PRINT SW
46 01027 006535      JSR      @TINH1
47 01030 002445      JMP      @CRBIR ;NO PRINT
48 01031 030741      LDA      2,NK10 ;8 TIMES
49 01032 125300      MCVS    1,1
50 01033 020735 PRINC:  LDA      0,CHAR0
51 01034 125102      MCVL    1,1,SZC
52 01035 101400      INC      0,0 ;AC2:="CHAR1"
53 01036 002436      JMP      @PBINR ;OUTPUT DIGIT
54 01037 151404 PBINA:  INC      2,2,SZR
55 01040 000773      JMP      PRINC
56 01041 024432      LDA      1,SABIN ;RESTORE PARAM
57 01042 002433      JMP      @CRBIR
58
59 01043 054432 XCRLF:  STA      3,CRBIR ;SAVE RETURN
60 01044 171000      MCV     3,2 ;SPEC RETURN IF PRINT SW
61 01045 006517      JSR      @TINH1
62 01046 002427      JMP      @CRBIR ;NO TYPE
63 01047 020427      LDA      0,C-OR
64 01050 004622      JSR      YCHAR
65 01051 020426      LDA      0,CHLF
66 01052 004620      JSR      YCHAR ;PRINT CR,LF
67 01053 002422      JMP      @CRBIR ;EXIT
```

1 0723 .MAIN

```
21
22 01054 001055 ACTN: .+1 ;ACTION ENTRY TABLE
23 01055 001071 ACT? ;NORMAL ACTION
24 01056 001067 ILL
25 01057 001067 ILL
26 01060 001067 ILL
27 01061 001067 ILL
28 01062 001067 ILL
29 01063 001072 ACT6 ;BLIND
30 01064 001067 ILL
31 01065 001067 ILL
32 01066 001067 ILL
33
34 01067 003077 ILL: KALT
35 01070 000777 JMP .-1 ;ILLEGAL
36 01071 000651 ACT0: JMP TYPE1
37 01072 000652 ACT6: JMP TYPE2
38
39 01073 000000 SABIN: 0
40 01074 000000 PBINR: 0
41 01075 000000 CRBIR: 0
42 01076 000215 CHCR: 215
43 01077 000012 CHLF: 12
44 01100 000000 REG0: 0
45 01101 000000 REG1: 0
46 01102 000000 REG2: 0
47 01103 000000 REG3: 0
48
49 01104 054771 XDISP: STA 3,CRBIR ;DISPLAY MESSAGE
50 01105 171000 MOV 3,2 ;SPEC RETURN IF PRINT SW
51 01106 010767 ISZ CRBIR
52 01107 006455 JSR @TINHT
53 01110 002765 JMP @CRBIR ;NO PRINT
54 01111 034764 LDA 3,CRBIR ;AC3=POINTS TO MESSAGE POINTER+1
55 01112 031777 LDA 2,-1,3 ;AC2 POINTS TO MESSAGE
56 01113 024447 LDA 1,CMSK
57 01114 021000 DISP1: LDA 0,0,2 ;ACC=DATAWORD
58 01115 125112 MOVL# 1,1,SZC
59 01116 123701 ANDS 1,2,SKP
60 01117 123421 AND 1,0,SKP ;ACC=CHAR. RIGHT
61 01120 151400 INC 2,2 ;INC TO NEXT WORD
62 01121 124000 COM 1,1 ;FLIP MASK
63 01122 004407 JSR DDICH ;GO DISPLAY
64 01123 000771 JMP DISP1 ;ANOTHER
65 01124 006063 TIMSK ;TERMINATE DISP
66 01125 000050 FDIST: 40. ;MAX 40 MSEC
67 01126 063535 SKPBZ DIS
68 01127 044776 STA 1,FDIST ;REMOVE WAITING, DIS NOT CONNECTED
69 01130 002745 JMP @CRBIR ;RETURN
70
71 01131 101015 DDICH: MOV# 0,0,SNR ;TEST LAST CHAR
72 01132 001401 JMP 1,3 ;RETURN +2 IF NULL
73 01133 000407 JMP YDCUT ;DISPLAY
```


1 0224 ,MAIN

```
01
02 01134 054744 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 036425 JSR @TINH1
06 01140 022743 JMP @REG3 ;NO DISPLAY
07 01141 020423 JMP QDOUT
08
09 01142 054741 YDOUT: STA 3,REG3 ;SAVE RETURN
10 01143 040735 STA 2,REG2 ;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 036063 TIMSK ;WAIT IF DIS BUSY
14 01147 002050 SDIST: #2. ;MAX 40 MSEC
15 01150 063535 SKPBZ DIS
16 01151 044776 STA 1,SDIST ;REMOVE WAITING, DIS NOT CONNECTED
17 01152 022726 LDA 0,REG0
18 01153 061035 DCA 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 022724 JMP @REG3 ;EXIT
23
24 01160 002000 PTAB: TABLE
25 01161 000177 RMSK: 177
26 01162 000377 CYSK: 377
27 01163 077400 LMSK: 277400
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 @TINH1
33 01170 002705 JMP @CRBIR ;NO DISPLAY
34 01171 000422 JMP QDICL
35
36 01172 054703 YDICL: STA 3,CRBIR
37 01173 020724 QDICL: LDA 2,CHLF
38 01174 004746 JSR YDOUT ;DISPLAY CLEAR
39 01175 002700 JMP @CRBIR ;RETURN
```

```

1 0225 ,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 3602, ATTENTION HALT BY MEANS OF
10 ;ACCUSTIC ALARM FOR 50 MS BEFORE HALT.
11 ;
12 ;CALL CHAAT
13 ;
14 01176 054676 DISATT: STA 3,PBIR
15 01177 171300 MCV 3,2 ;SPEC RETURN IF PRINT SW
16 01200 006764 JSR @TINH1
17 01201 002673 JMP @PBIR ;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 C*AIT
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 SUSZL 1,1 ;AC1:=1
31 01216 133414 AND# 1,2,SZR
32 01217 004753 JSR YD1CL ;CLEAR DIS IF SWITCH 15
33 01220 002654 JMP @PBIR
34
35 01221 054654 HAATT: STA 3,CRBIR
36 01222 171300 MCV 3,2 ;SPEC RETURN IF PRINT SW
37 01223 006741 JSR @TINH1
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 DCA 0,FUN
44 01231 006061 C*AIT
45 01232 001243 SEC*5
46 01233 102400 SUB 0,0
47 01234 001032 DCA 0,FUN
48 01235 002640 JMP @CRBIR
49
50 ;
51 01236 005673 SEC3: .RDX 10 ;CHANGING SEC3 AFFECTS CRESW AND POWON.
52 01237 000024 SEC4: 22 ;CHANGE TO 1000(0) FOR 1(0) SEC WAITING.
53 01240 000062 SEC5: 52 ;20 MSEC FOR TYPE ROUTINE DATAPoint DELAY
54 ;
55 01241 002002 SEC2: .RDX 8
56 01242 000024 SEC4: 2
57 01243 004000 SEC4K: 4002

```

1 2026 ,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 5 AT TTY
09 ;UNCHANGED: AC2
10 ;DESTROYED: AC1,AC2,AC3,CARRY
11 ;CALL: WATOP
12 ; ARG
13
14 01244 025400 XWTOP: LDA 1,2,3 ; FUNCTION MASK FROM ARGUMENT
15 01245 175430 INC 3,3 ; FOR INDICATORS AND BUTTONS
16 01246 250437 STA 3,WTORE ; SAVE RETURN
17 01247 030440 LDA 3,WHIGH
18 01250 040441 STA 0,WACSAV
19 01251 020435 LDA 0,WLOWL
20 01252 065232 DCA 1,FUN ; TURN ON INDICATOR ARG
21 01253 063610 WTNOK: SKPDN XTTI
22 01254 020411 JMP WTFUB
23 01255 070410 DIA 2,XTTI
24 01256 151300 MOVSV 2,2 ; REMOVE P-BIT BY
25 01257 151120 MCVZL 2,2 ; MULTIPLYING WITH 512 DEC.
26 01260 200210 NIOC XTTI
27 01261 172233 ADCZ# 3,2,SNC ; IS KEY BETWEEN OR
28 01262 142232 ADCZ# 2,2,SZC ; EQUAL TO LIMITS ?
29 01263 020402 JMP WTFUB ; NO
30 01264 020404 JMP WTK ; YES
31 01265 070433 WTFUB: DIA 2,FUB ; SENSE BUTTON ARG
32 01266 147415 AND# 2,1,SMR ; IS ARG PRESSED?
33 01267 020764 JMP WTNOK ; NO - SENSE AGAIN
34 01270 152400 WTK: SUB 2,2
35 01271 071032 DCA 2,FUN ; YES - TURN OFF INDICATOR ARG
36 01272 072433 DIA 2,FUB ; WAIT TILL THE OPERATOR
37 01273 147414 AND# 2,1,SZR ; GETS HIS DIRTY FINGER
38 01274 020776 JMP .-2 ; OFF THE BUTTON
39 01275 020413 LDA 0,WCH44
40 01276 036041 CCHAR
41 01277 036046 CDICL
42 01300 020244 CDISP
43 01301 027245 PROG ; "ACTUAL PROGRAM NAME"
44 01302 036050 CHAAT
45 01303 020406 LDA 0,WACSAV
46 01304 020401 JMP @WTORE ; RETURN
47 01305 020200 WTORE: 0
48 01306 011000 WLOWL: 11*1000 ; LOW LIMIT CHAR * 219
49 01307 040000 WHIGH: 40*1000 ; HIGH LIMIT CHAR * 219
50 01310 020244 WCH44: 44
51 01311 020000 WACSAV: 0
52
53 ;INSERT ACTUAL PROGRAM NAME AFTER A LABEL "PROG:"
```

1 0327 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 040451 STA 1,SSWR1
12 01314 050451 STA 2,SSWR2 ;FOR PRINT INHIBIT RETURN
13 01315 050445 STA 3,RRESW ;SAVE RETURN
14 01316 020450 LDA 1,LSTSW ;LAST SW REG
15 01317 071477 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 020441 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 020430 LDA 1,@XSTAC
27 01333 040430 STA 1,SSTAC ;SAVE PRINT INHIBIT SITU
28 01334 030430 LDA 2,KSTAC ;GET PRINT INHI INITIAL FOR PRINT
29 01335 052433 STA 2,@XSTAC ;SW REG WITHOUT FURTHER CALL CRESW
30 01336 026043 CCRLF
31 01337 006040 CMESS
32 01340 001522 MSWRG ;SWITCHES: XXXXXX
33 01341 020425 LDA 1,LSTSW
34 01342 006052 CTOCT
35 01343 006424 JSR @WBZOT ;WAIT LPT,ITC
36 01344 030425 LDA 2,@XSTAC
37 01345 052423 STA 2,@XSTAC ;RESTORE PRINT INHIBIT SITU
38 01346 030414 LDA 3,RRESW ;IF RRESW IS ALMOST EQUAL
39 01347 030421 LDA 2,XSTAC ;SETAC IT WAS PRINT INHIBIT
40 01350 150645 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 020411 LDA 1,SSWR1 ;RESTORE AC'S
44 01354 030412 LDA 2,LSTSW ;NEW SW POSITION
45 01355 022405 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 020404 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 000423 KSTAC: 403 ;JMP NINHI, ASH VALUE FROM SETAC
61 01373 000000 LALOC: 0 ;LAST LOC IN MEM
62 01374 077777 LAPRG: 277777 ;LAST LOC FOR PROGRAM
63 01375 077635 BINPI: 77635 ;FIRST ADDR OF BINARY LOADER IF 32K.
64 01376 000177 BINLA: 177 ;CONSTANT FOR LAST ADDR IN MEM
65 01377 002756 XFITY: FITYP
```

1 0025 ,MAIN

01
02
03
04
05
06
07

;SUBROUTINE TO RELOAD BOOTSTRAP LOADER AND BINARY LOADER.
;DO USE IN HEAD OF ANY PROGRAM AS MANY THINGS ARE INITIALIZED.
;FOR LOADERS INFO SEE LOADER BELOW.
;
;CALL START IN REBIN AFTER LOADING PROGRAM SELFSTARTING.

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
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64

REBIN: LDA 0,POWZE ;SET RESTART ADDR TO CELL ZERO
STA 0,2 ;FOR POWER RESTART.
JSR 0,XMEXT ;TRY TO SET MEM EXT FLAG (RC3603 ONLY)
JSR 0,XMEND ;STORE MEM END LOC IN HMEND=AC2=XX7600
LDA 1,BINLA ;AC1:=177
ADD 2,1 ;AC1:=LAST LOC IN MEM=XX7777
STA 1,LALOC ;STORE IT FOR PRINT
MOVL# 1,1,SZC ;MORE THAN 32K ?
LDA 1,LAPRG ;YES, AC1:=077777, LAST LOC FOR PROG
LDA 3,BINFI ;NO, LAST LOC IN FIRST 32K
AND 1,3 ;AC3:=FIRST ADDR LOADER
LDA 2,BINAD ;AC2:=ADDR OF BIN DATA
MODAT: LDA 0,2,2
STA 0,2,3 ;MOVE DATA
SUB# 3,1,SAR ;LAST DATA ?
JMP GETYP ;YES
INC 2,2
INC 3,3 ;NEXT DATA ADDR
JMP MODAT
GETYP: SLB 1,1
STA 1,LSTSW ;SET LAST SW REG TO ALL ZERO
JSR 0,XFITY ;FIND CPU TYPE
CRESW ;OUTPUT OF SWITCH SETTINGS
JSR 0,XPCPT ;PRINT CPU TYPE
CDISP
MLLOC
CMESS
MLLOC ;<15><12>LAST LOC, XX7777
LDA 1,LALOC
CTOCT
CDOCT
C*AIT ;WAIT 3 SEC TO READ MESS. DON'T
SEC3 ;USE CDATT AS SW AREN'T SET.
CDISP
M*ILO
CMESS
M*ILO ;<15><12>BINARY LOADER OK
C*AIT
SEC3
CDICL
CDISP
CCRLF
CMESS
PROG ;ACTUAL PROGRAM NAME
C*AIT
SEC3
DICP 0,1 ;TRY TO SET MEM EXT FLAG (RC3603 ONLY)
LDA 0,3 ;CELL 3 = 0 FOR HALT
MOV 0,2,SZR ;CELL 3 = 1 FOR
JMP 0,1 ;SELFSTART ADDR 4
JMP 0,+1 ;CHOOSE YOUR OWN START ADDR,
S*ISA ;NORMAL HALT, BUT IF NO SWITCH PANEL ?
XPCPT: PRYTP
XMEND: GMEND
XMEXT: TMEND

0029 MAIN

01
02 000001 .TXTM 1 ;RDCS TEXT PACKING MODE
03
04 MELOC: .TXT !LOADING UNIT OFF! ;"LOADING UNIT OFF"
01472 046117
01471 043504
01472 040516
01473 043440
01474 052516
01475 040524
01476 020117
01477 043106
01500 000000

05
06 000000 .TXTM 0 ;NORMAL TEXT PACKING MODE
07

08 MLOC: .TXT !<15><12>LAST LOC. 1 ;"<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

;"<15><12>BINARY LOADER OK"

11
12
13 MSWRG: .TXT !SWITCHES: ! ;"SWITCHES: "

01522 053523
01523 052111
01524 044103
01525 051005
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

```

1 0030 ,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 ;ERRCRBLOCK=IGNORE BLOCK
06 ;REPEAT BLOCK=MULTIPLE DATA BLOCK
07 ;COUNT=WORD COUNT IN BLOCK
08
09 01547 001550 BINADR: .+1 ;ADDR OF BIN LOADER DATA
10
11 ;SUBROUTINE TO BUILD A WORD IN AC2
12 01552 054425 BUILD: STA 3,TEMP1 ;SAVE RETURN
13 01551 004406 JSR GTCHR ;GET FIRST BYTE
14 01552 171300 MOV5 3,2 ;PUT INTO LH OF AC2
15 01553 004404 JSR GTCHR ;GET NEXT BYTE
16 01554 173300 ADDS 3,2 ;FORM WORD IN AC2
17 01555 143000 ADD 2,0 ;ADD INTO CHECKSUM
18 01556 002417 JMP 0TEMP1 ;AND RETURN
19
20 ;READ A BYTE INTO AC3
21 ;IF SWITCH0=0 USE TELETYPE ELSE USE PTR
22 01557 054417 GTCHR: STA 3,TEMP2 ;SAVE RETURN
23 01560 034417 LDA 3,SAVE ;TEST WHICH DEVICE
24 01561 175103 MCVL 3,3,SNC
25 01562 000406 JMP GTTTI ;TTI
26 01563 063612 SKPDN PTR ;PTR
27 01564 000777 JMP .-1
28 01565 074412 DIA 3,PTR ;READ AND START
29 01566 060112 NIOS PTR
30 01567 002407 JMP 0TEMP2 ;AND RETURN
31
32 01572 063610 GTTTI: SKPDN TTI
33 01571 000777 JMP .-1
34 01572 074412 DIA 3,TTI ;READ AND START
35 01573 060112 NIOS TTI
36 01574 002402 JMP 0TEMP2 ;AND RETURN
37 01575 000000 TEMP1: 0
38 01576 000000 TEMP2: 0
39 01577 000000 SAVE: 0
40
41 ;TEST BLOCK TYPE
42 01600 125224 BTEST: MCVZR 1,1,SZR ;1=START BLOCK (.END XX)
43 01601 000411 JMP IGNOR ;NO, IGNORE BLOCK
44 01602 101004 MCV 0,0,SZR ;TEST THE CHECKSUM
45 01603 000461 JMP CHKR ;ERPOR
46 01604 030505 LDA 2,ADDRS ;GET ADDR
47 01605 062677 ICRST ;DO A RESET
48 01606 151113 MCVL# 2,2,SNC ;TEST BIT 3
49 01607 001000 JMP 0,2 ;0=START PROGRAM
50 01610 063077 HALT ;1=HALT
51 01611 000777 JMP .-1 ;DON'T PROCEED
52
53 ;IGNORE BLOCK
54 01612 004745 IGNOR: JSR GTCHR ;READ UNTIL AN ALL
55 01613 020404 LDA 0,BC377 ;ONES BYTE IS SEEN
56 01614 116404 SUB 0,3,SZR ;IGNORING ERRCR MESS
57 01615 000775 JMP IGNOR
58 01616 000407 JMP BLOCK ;OK, GO INTO BLOCK MODE
59 01617 000377 BC377: 377

```

1 0031 ,MAIN

```
01
02 ;START OF PROGRAM BINARY LOADER TS
03 01620 062677 START: ICRST ;RESET
04 01621 060477 READS 0 ;READ THE SWITCH REGISTER
05 01622 040755 STA 0,SAVE ;AND SAVE IT FOR GTCR
06 01623 060110 NIOS TTI ;START BOTH READERS
07 01624 060112 NIOS PTR
08
09 ;READ IN A BLOCK
10 01625 004732 BLOCK: JSR GTCR ;GET A BYTE
11 01626 171305 MCVS 3,2,SAR ;AND TEST FOR NUL
12 01627 000776 JMP BLOCK ;YES, KEEP READING
13 01630 004727 JSR GTCR ;OK, GET NEXT BYTE
14 01631 173300 ADDS 3,2 ;AND FORM COUNT, AC2:=# OF WORDS
15 01632 141200 MCV 2,0 ;SET CHECKSUM, AC0:=CHECKSUM TILL NOW
16 01633 145000 MCV 2,1 ;SET COUNTER
17 01634 004714 JSR BUILD ;GET ADDRESS
18 01635 050454 STA 2,ADRS ;ADD IN THE CHECKSUM FROM TAPE
19 01636 004712 JSR BUILD ;ADD IN THE CHECKSUM FROM TAPE
20 01637 125113 MCVL# 1,1,SAC ;TEST BLOCK TYPE
21 01640 000740 JMP BTEST ;NOT A DATABLOCK
22 01641 044427 STA 1,COUNT ;STORE WORD COUNT
23
24 ;READ IN THE DATA BLOCK
25 01642 030734 DATA: LDA 2,TEMP2 ;LAST STA IN TEMP2 WAS JSR RETURN
26 01643 034423 LDA 3,DIFF ;ADDR 4 CELLS AFTER PROGRAM START:
27 01644 172400 SUB 3,2 ;AC2:=FIRST ADDR IN LOADER
28 01645 034414 LDA 3,ADRS ;ADDR IN WHICH TO STORE
29 01646 136400 SUB 1,3 ;ADD NEG WC TO CHECK SPACE
30 01647 172023 ADCZ 3,2,SAC ;FOR WHOLE BLOCK
31 01650 000414 JMP CHKR ;NO, HALT THE LOADER
32 01651 030416 LDA 2,BC22 ;IF WC > 20 (OCTAL, NEG)
33 01652 147033 ADDZ# 2,1,SAC ;IT IS A REPEAT BLOCK
34 01653 010415 ISZ COUNT ;WHERE WC IS ONE LESS THAN COUNT
35 01654 147022 ADDZ 2,1,SZC ;IF REPEAT BLOCK SKIP NEXT TO READ DATA
36 01655 125113 STORE: MCVL# 1,1,SAC ;DON'T READ IN NEW DATA IF REPEAT BLOCK
37 01656 024672 JSR BUILD
38 01657 052432 STA 2,ADRS
39 01660 010431 ISZ ADRS ;NEXT ADDR
40 01661 010407 ISZ COUNT ;TEST COUNT
41 01662 000773 JMP STORE ;MORE DATA
42 01663 101204 MCV 0,2,SZR ;TEST CHECKSUM
43 01664 063077 CHKR: HALT ;ERROR IN CHECKSUM, AC0=VALUE
44 01665 000740 JMP BLOCK ;OK,GET NEXT BLOCK
45 01666 000204 DIFF: 4
46 01667 000220 BC22: 22 ;REPEAT BLOCKS HAVE WC>20
47 01670 000000 COUNT: 0
48
49 ;ROOTSTRAP LOADER TS FOR PTR: ENTER AT BSTRP
50 01671 126440 GET: SUBC 1,1
51 01672 063612 SKPDN 12 ;10 FOR TTI
52 01673 000777 JMP 0,1
53 01674 060412 CIA 0,12 ;10 FOR TTI
54 01675 060112 NIOS 12 ;10 FOR TTI
55 01676 127100 ADDL 1,1
56 01677 127100 ADDL 1,1
57 01700 127203 ADD 0,1,SAC
58 01701 000771 JMP GET+1
59 01702 021400 JMP 0,3
60 01703 060112 BSTRP: NIOS 12 ;10 FOR TTI
61 01704 004765 JSR GET
62 01705 044402 STA 1,+2
63 01706 004763 JSR GET
64 01707 000000 0
65 01710 000000 0 ;FOR ROOTSTRAP
66 01711 000000 ADRS: 2
67 01712 000726 BEND: JMP START ;START OF BINARY LOADER
```


1 0032 MAIN

```

01
02
03 ;FIND THE TOP OF MEMORY (4K SEGMENTS)
04 ;STORE HIGHEST USEABLE ADDRESS IN
05 ;MEMEND, PROTECTING THE BIN LOADER.
06 ;THE CONTENTS IN MEMORY ARE LEFT UNCHANGED.
07 ;
08 ;CALL JSR GMEND
09 ; RETURN
10 ;4K CONSTANT
11 ;6K+1 ADDR
12 ;2K+LOADER CONSTANT
13 ;
14 ;PASS 1 AND 2K SEGMENTS
15 ;
16 ;SAVE LOC CONTENT
17 ;STORE INTO UPPER
18 ;GET IT BACK
19 ;
20 ;RESTORE LOC CONTENT
21 ;
22 ;CHECK LAST 64K/32K WITH L-SHIFT
23 ;
24 ;NEXT TOO FAR
25 ;PROTECT LOADERS
26 ;SET END MEM
27 ;
28
29 ;ROUTINE TO PRINT WHICH STARTADDR SA OF MORE
30 ;IS USED. INPUT: SA IN AC1.
31 ;CALL CSAMS
32 ; RETURN
33 ;XSAMS:
34 ;
35 ;SET OUTPUT DEVICE ROUTINES
36 ;TO WAIT FOR ONLINE DEVICES.
37 ;
38 ;
39 ;
40 ;
41 ;
42 ;
43 ;
44 ;
45 ;AC2:=1
46 ;SUBTRACT 1 FROM JSR ADDR
47 ;
48 ;PRINT SA
49 ;
50 ;
51 ;
52 ;
53 ;XXXXXX STARTADDR
54 ;ONLY FOR PRINT SWITCH SETTINGS IF NEW
55 ;PRINT CR,LF WITH MESS TO WAIT
56 ;FOR DEVICE READY BEFORE IORST
57 ;
58 ;RSAMS:
59 ;
60 ;TIMEOUT CONSTANTS FOR
61 ;OUTPUT DEVICES
62 ;
63 ;
64 ;
65 ;
66 ;

```

1 0033 MAIN

TABLE TO BE CHANGED TO CORRECT PRINTER ALPHABET.

01			
02			
03	002000	.LOC	2000
04	000010	.ROX	8
05			
06		TABLE:	
07			
08		.TXTE?	
09	02000	000006	<6><0>
10	02001	000006	<6><2>
11	02002	000006	<6><0>
12	02003	000006	<6><2>
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>

1 0034 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>

```

1 0035 ,MAIN
01 02171 054400 <0><131>
02 02172 055000 <0><132>
03 02173 140000 <0><130>
04 02174 121400 <0><43>
05 02175 156400 <0><135>
06 02176 004400 <0><11>
07 02177 000000 <6><0>?
02200 000000
08 000010 ,POX 8
09 ;START ADDR FOR "HELP" PROGRAMS
10 02201 ,LOC 2201
11
12 02201 002403 SAPTB
13 02202 006777 JSR 0,-1 ;START MOVE PRINTER TABLE
14 02203 002260 SAMEX
15 02204 006777 JSR 0,-1 ;START SET TO 64K MODE, MEM SIZE ?
16 02205 002265 SAMAM
17 02206 006777 JSR 0,-1 ;START SET TO 32K MODE, MEM SIZE ?
18 02207 005011 EXMEM
19 02210 006777 JSR 0,-1 ;START EXAMINE MEMORY
20 02211 005127 CPMEM
21 02212 006777 JSR 0,-1 ;START DEPOSIT MEMORY
22 02213 007105 TROWA
23 02214 006777 JSR 0,-1 ;START TROUBLE BREAKPOINT HALT
24 02215 007117 TROLO
25 02216 006777 JSR 0,-1 ;START TROUBLE BREAKPOINT LOOP REPORT
26 02217 007131 TRORE
27 02220 006777 JSR 0,-1 ;START TROUBLE BREAKPOINT RESET
28 02221 002314 LOADB
29 02222 006777 JSR 0,-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
33 ;ROUTINE TO MOVE PRINTER TABLE.
34 ;INPUT: TABLE # IN AC2
35 ;CALL: JSR MOPTB
36
37 02230 054415 MOPTB: STA 3,MOPTB ;SAVE RETIRE
38 02231 034417 LDA 3,MOGTT ;ADDR OF POINTER
39 02232 157000 ADD 2,3 ;AC3:=TABLE ENTRY
40 02233 031400 LDA 2,0,3 ;AC2:=ADDR OF TABLE, START
41 02234 024413 LDA 1,MOLAD ;LAST STORE ADDR
42 02235 034411 LDA 3,MOFID ;FIRST STORE ADDR
43 02236 021000 MOREP: LDA 0,0,2
44 02237 041400 STA 0,0,3 ;MOVE CHAR
45 02240 166415 SUB# 3,1,SNR ;LAST CHAR ?
46 02241 002404 JMP 0,MOPTB ;YES, RETIRE
47 02242 151400 INC 2,2
48 02243 175400 INC 3,3 ;NEXT CHAR ADDR
49 02244 000772 JMP MOREP ;REPEAT MOVE CHAR
50 02245 000000 MOPTB: 0 ;RETURN ADDR
51 02246 002000 MOFID: 2000 ;FIRST STORE ADDR
52 02247 002200 MOLAD: 2200 ;LAST STORE ADDR
53 02250 002250 MOGTT: ;GET TABLE ADDR
54 02251 003656 PTAB1 ;ASCII TABLE ADDR
55 02252 004057 PTAB2 ;PC STANDARD 71/78 START
56 02253 004260 PTAB3 ;PC STANDARD 71/78 START 2
57 02254 004461 PTAB4 ;PL1 TYPE 70 TABLE ADDR
58
59 02255 000004 ALPTT: 4 ;SUGGESTED ANSWER TABLE #
60 02256 000004 ULPTT: 4 ;UPPER LIMIT
61 02257 000001 LLPTT: 1 ;LOWER LIMIT

```

1 0036 ,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 ICRST. 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: MCV# 2,2,SNC

17
18 ;PROGRAM TO SET MAX MEM LOC TO 32K WORDS MODE.
19

20 02265 165000 SAMNM: MOV 3,1
21 02266 006072 CSAMS ;START ADDR MESSAGE
22 02267 024404 LDA 1,MCMNM
23 02270 046423 STA 1,MIMEX
24 02271 062677 ICRST ;RESET MEM EXT FLAG (RC3603 ONLY)
25 02272 000402 JMP SAMMS
26 02273 151113 MCMNM: MOVL# 2,2,SNC

27
28 02274 006416 SAMMS: JSR 0,IMEND ;AC2=HMEND=XX7600
29 02275 006044 COISP
30 02276 001501 MLLCC
31 02277 006040 CMSS
32 02300 001501 MLLCC ;<15><12>LAST LOC. XX7777
33 02301 020074 LDA 0,HMEND
34 02302 026407 LDA 1,SAMCC ;AC1:=177
35 02303 107000 ADD 0,1
36 02304 006056 CDOCT
37 02305 006052 CTCT
38 02306 006047 CDATT
39 02307 002401 JMP 0,+1 ;PROGRAM FINISHED
40 02310 004740 SWISA ;RESTART MAIN PROGRAM.
41 02311 001376 SAMCC: BINLA
42 02312 001716 IMEND: GMEND
43 02313 001730 MIMEX: JMEND
44

45 ;ROUTINE RESTORE BINARY LOADER AND
46 ;START LOADING PTR/TTI DEPENDING ON SWITCH 0.
47

48 02314 165000 LOADB: MCV 3,1
49 02315 006072 CSAMS ;START ADDR MESSAGE
50 02316 024755 LDA 1,MCMNM
51 02317 046774 STA 1,MIMEX ;MAX 32K MODE
52 02320 062677 ICRST ;RESET MEM EXT FLAG
53 02321 006771 JSR 0,IMEND ;AC2=HMEND=0X7600
54 02322 026767 LDA 1,SAMCC ;AC1:=177
55 02323 147000 ADD 2,1 ;LAST LOC = 0X7777
56 02324 036412 LDA 3,LOADB ;FIRST ADDR LOADER
57 02325 137400 AND 1,3 ;IN THIS MEM
58 02326 032411 LDA 2,LOADA ;ADDR OF BINARY DATA
59 02327 021000 LOADR: LDA 0,0,2
60 02330 041400 STA 0,0,3 ;MOVE DATA
61 02331 166415 SUB# 3,1,SNR ;LAST DATA ?
62 02332 001400 JMP 0,3 ;YES, START BINARY LOADER
63 02333 151400 INC 2,2
64 02334 175400 INC 3,3 ;NEXT DATA ADDR
65 02335 000772 JMP LOADR
66 02336 001375 LOADF: BINFI ;FIRST ADDR BIN LOADER (32K)
67 02337 001547 LOADA: BINAD ;ADDR POINTER TO BIN DATA

```

1 0037 ,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 ICRST ;RESET MEM EXT FLAG
07 02342 062701 DICP 0,1 ;TRY TO SET MEM EXT FLAG (RC3603 ONLY).
08 02343 032433 LDA 2, OMEND ;DON'T USE SKPDN 1 TO CHECK FLAG
09 02344 034433 LDA 3, PMEND ;AS NOT IMPLEMENTED IN ALL CPUs
10 02345 021000 LDA 0,0,2 ;SAVE LOWER LCC CONTENT
11 02346 040432 STA 0, OMEND
12 02347 051200 STA 2,0,2 ;STORE LOWER ADDR
13 02350 025400 LDA 1,0,3 ;SAVE UPPER LCC 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, OMEND ;YES, NOT MEM EXT OPTION, MAX 32K
23 02362 041000 STA 0,0,2 ;RESTORE LOWER LCC CONTENT
24 02363 024710 LDA 1, MCMEM ;SET 32K MAX
25 02364 046727 STA 1, MIMEX
26 02365 002410 JMP RMEND ;EXIT
27 02366 020412 UMEND: LDA 0, OMEND
28 02367 041000 STA 0,0,2 ;RESTORE LOWER LCC CONTENT
29 02370 024411 LDA 1, SMEND
30 02371 045400 STA 1,0,3 ;RESTORE UPPER LCC 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 15 BIT, NOT INDIRECT
37 02400 000000 OMEND: 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: MCV 3,1
44 02404 006072 CSAMS ;START ADDR MESSAGE
45 02405 006071 RLPTT: CQUES
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 030275 LDA 2, DIGIN ;ANSWER INPUTTED
55 02417 024637 LDA 1, ULPTT ;UPPER LIMIT
56 02420 020637 LDA 0, LLPTT ;LOWER LIMIT
57 02421 132033 ACCZ# 1,2, SNC
58 02422 142032 ACCZ# 2,0, SZC ;AC2=<AC2=<AC1 ?
59 02423 000762 JMP RLPTT ;OUTSIDE LIMITS
60 02424 000604 JSR MOPTB ;INPUT ACCEPTED, MOVE TABLE
61 02425 002401 JMP .+1 ;PROGRAM FINISHED
62 02426 004740 SAISA ;RESTART MAIN PROGRAM
63
64
65 ;TAPE 2
66
67 ,EOT

```

2038 MAIN

```

01
02 ;TAPE 3
03
04 ;DELAY SUBROUTINE
05 ;ARGUMENT FOLLOWING CALL IS ADDRESS
06 ;OF DELAY CONSTANT.
07 ;DELAY IN INCREMENTS OF 1 MSEC
08 ;AC2 & 3 ARE USED, DEFINE WAIT, ARG
09 ;
10 ;CALL CWAIT
11 ; ARG
12 ;
13 02427 054416 XWAIT: STA 3,WARET ;SAVE RETURN
14 02430 010415 ISZ WARET ;PASS ARG
15 02431 033400 LDA 2,0,3 ;FETCH ARG
16 02432 050414 STA 2,WIRET ;STORE ARG
17 02433 030414 LDA 2,KINDI ;FETCH KNOVA ADDRESS
18 02434 036537 LDA 3,0XCPN ;FETCH CPUNO
19 02435 157000 ADD 2,3 ;COMPUTE KADDRESS
20 02436 031400 LDA 2,0,3 ;FETCH CPU CONSTANT
21 02437 050437 NWAIT: STA 2,KINC ;STORE IT FOR INC OF 1 MSEC
22 02440 014436 DSZ KINC ;X USEC LOOP
23 02441 000777 JMP -1 ;X USEC
24 02442 014404 DSZ WIRET ;NO. OF MS STILL TO WAIT
25 02443 000774 JMP NWAIT ;COUNT ARG NOT ENDED
26 02444 002401 JMP WARET ;COUNT ARG ENDED, RETURN
27 02445 000000 WARET: 0
28 02446 000000 WIRET: 0
29
30 02447 002450 KINDI: .+1 ;ADDRESS OF KNOVA
31 000012 .RDX 10
32 02450 000175 KCP0: 125 ;NOVA
33 02451 000001 KCP1: 1 ;
34 02452 000341 KCP2: 225 ;1200
35 02453 000567 KCP3: 375 ;SUPER
36 02454 000505 KCP4: 325 ;SUPER SC/830
37 02455 000536 KCP5: 350 ;800/NOVA 2-16K
38 02456 000620 KCP6: 400 ;NOVA 2-8K
39 02457 000001 KCP7: 1 ;CONSTANTS ABOVE ARE FOR GUESSED CPU TYPE
40 02460 000001 KCP10: 1 ;CONSTANTS BELOW ARE EXACTLY.
41 02461 000200 KCP11: 128 ;NOVA 7,8 USEC
42 02462 000336 KCP12: 222 ;1200 4,5 USEC
43 02463 000601 KCP13: 385 ;800 2,6 USEC
44 02464 000601 KCP14: 385 ;SUPER 2,6 USEC
45 02465 000764 KCP15: 500 ;SUPER SC 2,2 USEC
46 02466 000620 KCP16: 400 ;NOVA2/8K 2,5 USEC
47 02467 000502 KCP17: 322 ;NOVA2/16K 3,1 USEC
48 02470 000531 KCP20: 345 ;RC3603/RC3609 2,90 USEC
49 02471 000334 KCP21: 220 ;RC3603/RC3609/BREAK ON 4,55 USEC
50 02472 000463 KCP22: 327 ;RC3603/RC3608 3,26 USEC
51 02473 000314 KCP23: 204 ;RC3603/RC3608/BREAK ON 4,90 USEC
52 02474 000515 KCP24: 333 ;NOVA2/DATARAM 900 NSEC 3,00 USEC
53 02475 000001 KCP25: 1 ;FOR TROUBLE, PUT HERE RELEVANT VALUE.
54 000010 .RDX 8
55 02476 000000 KINC: 0

```

1 0039 ,MAIN

01
02
03
04
05
06
07
08
09

;PROCEDURE TIMER ON SKP
;CALL: TIMSK
; MSEC (>0) TO WAIT MAX FOR
; SKP INSTR.
; TIME OUT RETURN
; NORMAL RETURN

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 ;AC2:=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 2,0XCPN ;FETCH CPUN0
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 020000 STSKP: 2 ; X USEC (SKP INSTR.)
23 02514 020402 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, AC2=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: -152 ;1200
37 02530 177470 CPU3: -202 ;SUPER
38 02531 177437 CPU4: -225 ;SUPER SC/832
39 02532 177406 CPU5: -252 ;800/NOVA 2-16K
40 02533 177324 CPU6: -302 ;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: -168 ;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

1 2340 .MAIN

```
01
02 ;ROUTINE TO MEASURE TIME.
03 ;CALL TIMMS
04 ; SKP INSTR. TO BE MEASURED
05 ; TIMEOUTRETURN TIMEM:=MAX TIME
06 ; NORMAL RETURN TIMEM:=MEASURED TIME
07 ;THE TIME IS MEASURED IN TENS OF USEC.
08 ;LOOPS IN THE INSTRUCTIONS MARKED X USEC UNTIL
09 ;BIT 0 IS SET (TIMEOUT, MAX MEASUREABLE TIME)
10 ;COUNTING FROM -1 IN THE INC INSTRUCTION OR UNTIL
11 ;SKP INSTRUCTION TO BE MEASURED 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 MEASURED.
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 MEASURED
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 MEASURED
32 02573 003215 XCPN: CPUNO
33
34 02574 002575 TCP00: .+1 ;100 MSEC CONSTANTS:
35 000012 .RDX 10
36 02575 012574 TCP01: 5500 ;NOVA
37 02576 077777 TCP1: 32767 ; WILL GIVE TIMEOUT = 100 MSEC
38 02577 030324 TCP2: 12500 ;1200
39 02600 043164 TCP3: 16500 ;SUPER
40 02601 051774 TCP4: 21500 ;SUPER SC/830
41 02602 055714 TCP5: 23500 ;800/NOVA2-16K
42 02603 063004 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 030226 TCP21: 12422 ;RC3603/RC3609/BREAK ON 8,25 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 MST10: ADCZR 0,0 ;AC0:=077777, CARRY:=1
63 02625 040743 STA 0,TIMEX ;TIMEOUT TIMECOUNT = 32767
64 02626 000402 JMP MSTID ;CALCULATE
```

1 0241 .MAIN

```
01
02 02627 000000 MSTIC: 0 ;CARRY FLAG
03 02630 101200 MSTID: MOVR 0,2 ;
04 02631 040776 STA 0,MSTIC ;SAVE CARRY
05 02632 026741 LDA 1,AXCPN ;
06 02633 030741 LDA 2,TCP00 ;
07 02634 133000 ADD 1,2 ;
08 02635 025000 LDA 1,2,2 ;FETCH CPU CONSTANT
09 02636 044733 STA 1,TIMCT ;
10 02637 024731 LDA 1,TIMEX ;
11 02640 030763 LDA 2,MSTID ;
12 02641 006066 MULTI ;TIMECOUNT X 10000
13 02642 031727 LDA 2,TIMCT ;
14 02643 006070 DIVID ;DIVIDED BY CPU CONSTANT
15 02644 044720 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 101123 MOVL 0,0,SNC ;IF CARRY THEN TIMEOUT
19 02650 010402 ISZ MSTIR ;PASS TIMEOUT RETURN
20 02651 002401 JMP 0MSTIR ;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 2,2
35 02661 175404 INC 3,3,SZR
36 02662 000774 JMP MLOOP
37 02663 125260 MOVR 1,1
38 02664 002415 JMP 0MSAV
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# 2,2,SZC ;CAPRY...?
59 02673 142400 SUB 2,2
60 02674 125100 MOVL 1,1
61 02675 175404 INC 3,3,SZR
62 02676 000773 JMP DLOOP
63 02677 002402 JMP 0MSAV
64 02700 177760 MDCTR: -22
65 02701 000000 MSAV: 0
```

1 0042 .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 000411 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 000406 JMP @XTIMR ;TIMEOUT RETURN
30 02725 000061 XTIMW: CWAIT
31 02726 000730 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: 2 ;RETURN ADDR.
```

```
36
37 UNTST: .TXT 1, 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"
```

1 0043 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 MEASURE TIME FOR DEFINITION OF #.
07
08 02756 054432 FITYP: STA 3,RETY
09 02757 062677 ICRST
10 02760 020434 LDA 0,NUTYP ;NUMBER OF TRYING AGAIN
11 02761 040432 STA 0,AGTYP
12 02762 024447 TRTYP: JSR TYMER
13 02763 125014 MOV# 1,1,SZR ;AC1=0, SEARCH MORE
14 02764 000422 JMP STTYP
15 02765 000424 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 026420 JSR @XWTYP
21 02773 044422 STA 1,CPUNO
22 02774 006044 CDISP
23 02775 001536 UNTIM
24 02776 006040 CMES
25 02777 001536 UNTIM ;MISERABLE TIMING
26 03000 006040 CMES
27 03001 002733 UNTST ;RUN INSTR TIMER TEST
28 03002 026043 CCRLF
29 03003 006061 CAAIT ;WAIT 3 SEC TO READ MESS. DON'T
30 03004 001236 SEC3 ;USE CDATT AS SW AREN'T SET.
31 03005 002403 JMP @RETY ;EXIT
32 03006 044407 STTYP: STA 1,CPUNO
33 03007 002401 JMP @RETY ;EXIT
34 03010 000000 RETYP: 2
35 03011 003275 XOMER: TOMER
36 03012 003475 XWTYP: NATYP
37 03013 000000 AGTYP: 2
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,RETY ;PRINT CPU TYPE NUMBER
42 03017 006044 CDISP
43 03020 025002 MCPLT
44 03021 006040 CMES
45 03022 025202 MCPLT ;<15><12>CPU TYPE:
46 03023 024772 LDA 1,CPUNO
47 03024 006052 CTOCT
48 03025 006056 CDOCT
49 03026 006061 CAAIT ;WAIT 3 SEC TO READ MESS. DON'T
50 03027 001236 SEC3 ;USE CDATT AS SW AREN'T SET.
51 03030 002760 JMP @RETY ;EXIT
```

1 2044 MAIN

```
01
02 03231 054572 TYMER: STA 3,SVTYME
03 03232 020576 LDA 0,XTYME ;GET ADDRESS OF TYME LIST.
04 03233 040021 STA 0,IDX1 ;STORE IN AUTO POINTER.
05 03234 020575 LDA 0,XINST ;GET ADDR OF INSTRUCTION LIST
06 03235 040020 STA 0,IDX2 ;STORE IN AUTO POINTER.
07 03236 030567 LDA 2,INXW5
08 03237 050023 STA 2,IDX3 ;INITIALIZE IDX3 FOR LDA INST.
09 03240 152440 SUBC 2,2 ;AC2:=0, NULL CHAR
10 03241 071011 CCA 2,XTT0 ;SEND CHAR
11 03242 060111 NIOS XTTO ;START TTO AND
12 03243 063511 SKPBZ XTTO ;SYNCHRONIZE
13 03244 000777 JMP -.1 ;WITH TTO CLOCK
14 03245 071011 DCA 2,XTT0 ;SEND A CHAR
15 03246 060111 NIOS XTTO ;AND MEASURE TIME, NOT ACCURATE
16 03247 060064 TIMYS ;CPU TYPE NOT KNOWN BUT TO GIVE AN IDEA
17 03250 063511 SKPBZ XTTO ;OF TTO SPEED
18 03251 000401 JMP .+1 ;AC2 IS MEASURED TIME
19 03252 022544 LDA 0,#TYMEM
20 03253 024544 LDA 1,TYLIM ;LIMIT TO DESTINGV, SPEED
21 03254 034544 LDA 3,TYTTS ;SLOW TTY CONSTANT
22 03255 126432 SUBZ# 0,1,SZC ;IS TTY FAST ?
23 03256 034543 LDA 3,TYTTF ;YES, FAST TTY CONSTANT
24 03257 054535 STA 3,TYMEN ;STORE # OF COUNTS
25 03260 152440 SUBC 2,2 ;AC2:=0, NULL CHAR.
26 03261 071011 CCA 2,XTT0 ;SEND CHAR
27 03262 060111 NIOS XTTO ;START TTO AND
28 03263 063511 SKPBZ XTTO ;SYNCHRONIZE PROGRAM
29 03264 000777 JMP -.1 ;WITH TTY CLOCK.
30 03265 152440 TYMA: SUBC 2,2 ;CLEAR AC2
31 03266 141000 MOV 2,0 ;CLEAR AC2 ALSO.
32 03267 026020 LDA 1,#IDX0 ;GET INST. FROM LIST
33 03270 125015 MOV# 1,1,SNR ;ZERO MARKS END OF INSTR. LIST
34 03271 000455 JMP SCORE ;
35 03272 071011 CCA 2,XTT0 ;SEND NULL CHAR, STARTING LATER
36 03273 034451 LDA 3,CSKP ;GET ALC-SKP MASK AND
37 03274 137400 AND 1,3 ;AND WITH INSTR.
38 03275 175123 MOVZL 3,3,SNR ;CKN BIT 2, THE ALC BIT
39 03276 000403 JMP TYMD ;BIT 0 = 0 MEANS NO ALC CODE.
40 03277 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-SKPP 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 INSD FROM TYMJ
60 03122 100005 INDY1: #5 ;HERE WORKS LDA #INDAD FROM TYMJ
```

1 2045 MAIN

```
01
02 03123 101000 INSTR: MCV 0,2 ;THIS IS THE 16 INSTRUCTION
03 03124 103000 ADD 0,2 ;LIST, SELECTED TO DEVELOPE
04 03125 103400 AND 0,2,SKP ;AN IDENTITY PROFILE OF THE
05 03126 220005 LDA 0,5 ;PROCESSOR IN THE COURSE OF
06 03127 240005 STA 0,5 ;BEING EXECUTED, THE INSTR. ARE
07 03130 210415 ISZ INSD ;LATER STORED IN TMMJ,TMMH
08 03131 207401 JMP INSTA
09 03132 204401 INSTA: JSR INSTB
10 03133 222005 INSTB: LDA 0,05
11 03134 222023 LDA 0,0IDX3
12 03135 222415 LDA 0,0INDAD ;LABEL INDAD PLACED .+16
13 03136 202400 DJA 0,0
14 03137 201000 DCA 0,0
15 03140 201477 INTA 0
16 03141 203400 SKPRN 0
17 03142 203100 NIOS 0
18 03143 200000 0 ;END OF INSTR. LIST MARKER
19 03144 100007 CSKP: 100007 ;ONLY FOR ASSEMBLING ISZ INSD
20 03145 000000 INSD: 0 ;TO WORK IN INSW1 AND INSW2.
21
22 03146 102520 SCORE: SUBZL 0,0 ;SET THE
23 03147 240455 STA 0,ORDINAL ;ORDINAL COUNTER TO +1.
24 03150 220460 SCORA: LDA 0,XYME ;GET TYME LIST INITIAL
25 03151 240021 STA 0,IDX1 ;ADDRESS FOR AUTO INC.
26 03152 220455 LDA 0,XC20 ;SET UP THE X16 COUNTER.
27 03153 240453 INDAD: STA 0,XX16 ;WITH A COUNT OF 16 DEC.
28 03154 226021 LDA 1,0IDX1 ;GET TYME ENTRY INTO AC1 AS FIRST
29 ;BIG TYME. THEN SEARCH FOR BIGGER TYME.
30 03155 222021 SCORB: LDA 0,0IDX1 ;CK MAGNITUDE OF NX TYME ENTRY.
31 03156 100033 ADCZ# 0,1,SNR ;SKPS IF ACZ < AC1
32 03157 100000 MOV 0,1 ;ACZ > OR = AC1, AC1 = BIGGEST TYME.
33 03160 214440 DSZ XX16 ;COUNT DOWN # OF TYMES.
34 03161 200774 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 200560 JMP KEYS ;AC1 = 0, TYME IS CLEAR, SO GO FORM KEY.
41 03164 220444 LDA 0,XYME ;REINITIALIZE RVTMP WITH
42 03165 240435 STA 0,RVTMP ;TYME = 1.
43 03166 220444 LDA 0,XRANK ;INITIALIZE IDX2 WITH
44 03167 240022 STA 0,IDX2 ;RANK = 1.
45 03170 200437 LDA 0,XC20 ;RESET X16 COUNTER
46 03171 240435 STA 0,XX16 ;BACK TO 16 DECIMAL.
47 03172 210432 REVUA: ISZ RVTMP ;INC TYMES LIST POINTER.
48 03173 200421 LDA 2,TYMEN ;GET TOLERANCE COUNT
49 03174 222420 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 0,2 ;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 200774 JMP REVUD ;IF BIGGEST TYME =1,-2,-3...-N, STORE ORD
55 03202 222022 LDA 0,0IDX2 ;ENTRY OUT OF RANGE, INC RANK POINTER.
56 03203 214423 REVUC: DSZ XX16 ;DECREMENT THE X16 POINTER.
57 03204 200760 JMP REVUA ;STILL MORE TYMES, SO LOOP.
58 03205 010417 ISZ ORDINAL ;INC ORDINAL COUNT.
59 03206 200742 JMP SCORA ;GO TO SCORA TO FIND NX BIG TYME.
60 03207 030415 REVUB: LDA 2,ORDINAL ;GET ORDINAL COUNT
61 03210 052022 STA 2,0IDX2 ;AND STORE IT IN PRESENT RANK LOC.
62 03211 170440 SUBO 3,3 ;CLEARING AC3.
63 03212 050410 STA 3,0RVTMP ;CLEARS LOC. IN TYME LIST.
64 03213 200773 JMP REVUC ;
```

1 0046 ,MAIN

```
01
02 03214 177775 TYMEN: -3 ;# OF COUNTS, TOLERANCE FOR TYMES
03 03215 177767 TYRTC: -9. ;-3 WILL GIVE BIG, BIG-1, BIG-2 IS EQUAL.
04 03216 202572 TYMEM: TIMEM ;MEASURED TIME FOR ONE TIO CHAR
05 03217 201212 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 000000 INXW5: 5 ;START LDA 0,#IDX3 IN CELL 5
12 03226 000000 XX16: 0
13 03227 000000 XC20: 20
14 03230 000000 XTYME: TYME-1
15 03231 000000 XINST: INSTR-1
16 03232 000000 XRANK: RANK-1
17 000000 TYME: .BLK 20
18 03253 000000 ENTYM: 0 ;END OF TYME LIST MARKER.
19 000000 RANK: .BLK 20
20 03274 000000 ENRANK: 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,TYMEM ;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 03312 001014 DCA 0,XRTC ;SET RTC FREQUENCY TO 100 HZ (10 MSEC).
34 03311 060114 NIOS XRTC
35 03312 063514 SKPRZ XRTC ;SYNCHRONIZE PROGRAM
36 03313 030777 JMP .-1 ;WITH RTC.
37 03314 152400 TOMA: SUBC 2,2 ;CLEAR AC2
38 03315 141000 MOV 2,0 ;CLEAR AC2 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 03323 030626 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,SNR ;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 040403 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 SKPRZ XRTC ; FORM THE *****
53 03334 030775 JMP TOMF ; TIMING LOOP *****
54 ;FOR NON ALC-SKP INSTR.
55 03335 052021 TOMC: STA 2,#IDX1 ;STORE COUNT INTO TYME LIST.
56 03336 030756 JMP TOMA ;LOOP.
57 03337 040403 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 060401 JMP .+1 ; FORM THE *****
62 03344 063514 SKPRZ XRTC ; TIMING *****
63 03345 030774 JMP TOMG ; LOOP *****
64 03346 030767 JMP TOMC ;FOR ALC-SKP INSTRUCTIONS.
65 03347 000000 INSW2: 0 ;HERE WORKS ISZ INSD FROM TOMJ.
66 03352 100000 INDW2: #0 ;HERE WORKS LDA #INDAD FROM TOMJ.
```

1 0047 ,MAIN

```
01
02 03351 020661 KEYS: LDA 0,XRANK ;REINITIALIZE RANK LIST
03 03352 040222 STA 2,IDX2 ;AUTO INC POINTER.
04 03353 102520 SUBZL 0,0 ;BIT 15:=1 AS END OF KEY MARKER.
05
26 03354 026222 KEYS: LDA 1,0IDX2 ;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 141122 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 KEYS: MOVCL 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 000773 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 CCM# 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 I RETURN WITH CPUNO IN AC1
26 03377 175400 INC 3,3 ;INC TO NEXT LOCK ENTRY
27 03400 000773 JMP SESA1 ;AND LOOP
28 03401 126440 SESOUT: SUBC 1,1 ;CLEAR AC1 AS KEY INDICATOR FOR NO
29 03402 002621 SESEX: JMP #SVTYM ;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
```


1 0048 MAIN

01
02 03435 166265 166265
03 03436 166165 166165
04 03437 166225 166225
05 03440 166325 166325
06 03441 172365 172365
07 03442 172325 172325
08 03443 174250 174250
09 03444 174150 174150
10 03445 174210 174210
11 03446 174250 174250
12 03447 174310 174310
13 03450 176210 176210
14 03451 177777 177777
15 03452 144352 144352
16 03453 154352 154352
17 03454 160135 160135
18 03455 162025 162025
19 03456 162202 162202
20 03457 162225 162225
21 03460 166225 166225
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 176250 176250
29 03470 177777 177777
30 03471 020653 020653
31 03472 177777 177777
32 03473 000000 0
33
34
35
36
37
38 03474 000000 0

;RC3603/RC3608, 32K KEY 22

;RC3603/RC3608/BREAK ON KEY 23

;NOVA 2 - DATARAM 930 NSEC KEY 24

;SPACE IF TROUBLES 25
;CORRECT JMP .+1 IN SESAME TO A HALT
;RESTART IN 1400 (REBIN). PUT HERE
;KEY FROM ACC EXAMINED WHEN STOPPED,
;AND CORRECT CPUNO 25'S CONSTANTS IN THE
;3 ROUTINES CWAIT, TIMSK, TIMMS
;END OF TABLE, NO CPU FOUND, # 0

1 0050 .MAIN

```
01
02 03564 054411 NOTYP: STA 3,NOTRE
03 03565 006046 CCICL ;CLEAR DIS
04 03566 006044 CDISP ;DIS - MESSAGE
05 03567 003542 NCTYM
06 03570 006040 CMSS
07 03571 003542 NCTYM ; SFT CPUNG > AC2
08 03572 006043 CCRLF ;NO DISATT, WAIT ROUT. NO CONSTANT
09 03573 003077 HALT ;PUT CPUNG INTO AC2
10 03574 002401 JMP 0,NTRE ;CONTINUE
11 03575 000000 NOTRE: 0
12
13 03576 020437 NITYP: LDA 0,AC125 ;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: SUBC 1,1 ;AC1:=0
18 03603 000114 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 101220 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 106433 SUBZ# 3,1,SAC
32 03621 132432 SUBZ# 1,2,SZC
33 03622 000414 JMP NYTYP ;OUTSIDE TOLERANCE
34 03623 014405 DSZ NTDEC ;97 %<COUNT<103 %
35 03624 020756 JMP NTREP ;TRY 125 TIMES
36 03625 024404 LDA 1,NTRES ;GET FIRST RESULT
37 03626 002401 JMP 0,NTRET ;EXIT
38 03627 000000 NTRET: 0
39 03630 000000 NTDEC: 0
40 03631 000000 NTRES: 0
41 000012 .RDX 10
42 03632 177214 NN500: -500
43 03633 177767 NN9: -9
44 03634 000012 NC8: 8
45 03635 000175 NC125: 125
46 000010 .RDX 8
47 03636 006046 NYTYP: CCICL
48 03637 006244 CDISP
49 03640 003645 NZTYP ;RTC IS UNSTABLE,
50 03641 006043 CCRLF
51 03642 006040 CMSS
52 03643 003645 NZTYP ;NO DISATT, WAIT ROUT. NO CONSTANT
53 03644 002666 JMP NITYP
54
55 NZTYP: .TXT !RTC IS UNSTABLE,! ;"RTC IS UNSTABLE,"
03645 052122
03646 020133
03647 051511
03650 052440
03651 051516
03652 040524
03653 046132
03654 026135
03655 000000
```

1 0051 ,MAIN

01
02 000010 ,ROX 8
03 PTAB1:
04 ,TXTE?
05 03656 000000 <6><2>
06 03657 000000 <6><2>
07 03660 000000 <6><2>
08 03661 000000 <6><2>
09 03662 000000 <6><2>
10 03663 000000 <6><2>
11 03664 000000 <6><2>
12 03665 000000 <6><2>
13 03666 000000 <6><2>
14 03667 000000 <2><11>
15 03670 005000 <2><12>
16 03671 000000 <6><2>
17 03672 006000 <2><14>
18 03673 106400 <2><15>
19 03674 000000 <6><2>
20 03675 000000 <6><2>
21 03676 000000 <6><2>
22 03677 000000 <6><2>
23 03700 000000 <6><2>
24 03701 000000 <6><2>
25 03702 000000 <6><2>
26 03703 000000 <6><2>
27 03704 000000 <6><2>
28 03705 000000 <6><2>
29 03706 000000 <6><2>
30 03707 000000 <6><2>
31 03710 000000 <6><2>
32 03711 000000 <6><2>
33 03712 000000 <6><2>
34 03713 000000 <6><2>
35 03714 000000 <6><2>
36 03715 000000 <6><2>
37 03716 117400 <2><37>
38 03717 020400 <2><41>
39 03720 021000 <2><42>
40 03721 121400 <2><43>
41 03722 022000 <2><44>
42 03723 122400 <2><45>
43 03724 123000 <2><46>
44 03725 023400 <2><47>
45 03726 024000 <2><50>
46 03727 124400 <2><51>
47 03730 125000 <2><52>
48 03731 025400 <2><53>
49 03732 126000 <2><54>
50 03733 026400 <2><55>
51 03734 027000 <2><56>
52 03735 127400 <2><57>
53 03736 030000 <2><60>
54 03737 130400 <2><61>
55 03740 131000 <2><62>
56 03741 031400 <2><63>
57 03742 132000 <2><64>
58 03743 032400 <2><65>
59 03744 033000 <2><66>
60 03745 133400 <2><67>
61 03746 134000 <2><70>
62 03747 034400 <2><71>
63 03750 035000 <2><72>
64 03751 135400 <2><73>
65 03752 036000 <2><74>
66 03753 136400 <2><75>
67 03754 137000 <2><76>
68 03755 037400 <2><77>

1 0352 MAIN

02 03756 140000 <0><100>
03 03757 140400 <0><101>
04 03760 141000 <0><102>
05 03761 141400 <0><103>
06 03762 042000 <0><104>
07 03763 142400 <0><105>
08 03764 143000 <0><106>
09 03765 143400 <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 000000 <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>
00 04050 055000 <0><132>
01 04051 140000 <0><100>
02 04052 121400 <0><43>
03 04053 156400 <0><135>
04 04054 004400 <0><11>
05 04055 000026 <6><0>?
04056 000000

1 0353 ,MAIN

31
32 000012 ,ROX 12
33 ,PTAB2:
34 ,TXTE?
35 04057 000000 <6><2>
36 04060 000000 <6><2>
37 04061 000000 <6><2>
38 04062 000000 <6><2>
39 04063 000000 <6><2>
40 04064 000000 <6><2>
41 04065 000000 <6><2>
42 04066 000000 <6><2>
43 04067 000000 <6><2>
44 04070 004400 <2><9>
45 04071 005000 <2><10>
46 04072 000000 <6><2>
47 04073 006000 <2><12>
48 04074 106400 <2><13>
49 04075 000000 <6><2>
50 04076 000000 <6><2>
51 04077 000000 <6><2>
52 04100 000000 <6><2>
53 04101 000000 <6><2>
54 04102 000000 <6><2>
55 04103 000000 <6><2>
56 04104 000000 <6><2>
57 04105 000000 <6><2>
58 04106 000000 <6><2>
59 04107 000000 <6><2>
60 04110 000000 <6><2>
61 04111 000000 <6><2>
62 04112 000000 <6><2>
63 04113 000000 <6><2>
64 04114 000000 <6><2>
65 04115 000000 <6><2>
66 04116 000000 <6><2>
67 04117 117400 <2><31>
68 04120 036000 <2><62>
69 04121 137000 <2><62>
70 04122 040400 <2><65>
71 04123 131000 <2><50>
72 04124 034400 <2><57>
73 04125 035000 <2><58>
74 04126 136400 <2><61>
75 04127 030000 <2><48>
76 04130 130400 <2><49>
77 04131 135400 <2><59>
78 04132 027000 <2><46>
79 04133 027400 <2><33>
80 04134 127400 <2><47>
81 04135 120000 <2><32>
82 04136 133400 <2><55>
83 04137 022000 <2><36>
84 04140 122400 <2><37>
85 04141 123000 <2><38>
86 04142 023400 <2><39>
87 04143 024000 <2><40>
88 04144 124400 <2><41>
89 04145 125000 <2><42>
90 04146 025400 <2><43>
91 04147 126000 <2><44>
92 04150 026400 <2><45>
93 04151 021000 <2><34>
94 04152 121400 <2><35>
95 04153 032400 <2><53>
96 04154 132000 <2><52>
97 04155 033000 <2><54>
98 04156 037400 <2><63>

1 0054 .MAIN

02 04157 241000 <0><66>
03 04160 257400 <0><95>
04 04161 157000 <0><94>
05 04162 156400 <0><93>
06 04163 056000 <0><92>
07 04164 155400 <0><91>
08 04165 255000 <0><90>
09 04166 254400 <0><89>
10 04167 154000 <0><88>
11 04170 153400 <0><87>
12 04171 053000 <0><86>
13 04172 252400 <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 040800 <0><72>
30 04213 040400 <0><71>
31 04214 140000 <0><70>
32 04215 031400 <0><51>
33 04216 134000 <0><56>
34 04217 030000 <0><12>
35 04220 257400 <0><95>
36 04221 157000 <0><94>
37 04222 156400 <0><93>
38 04223 056000 <0><92>
39 04224 155400 <0><91>
40 04225 255000 <0><90>
41 04226 254400 <0><89>
42 04227 154000 <0><88>
43 04230 153400 <0><87>
44 04231 053000 <0><86>
45 04232 252400 <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 140000 <0><70>
64 04255 040400 <0><9>
65 04256 040000 <6><0>?
04257 000000

1 2355 MAIN

01
02 000012 ,RDX 12
03 PTA03:
04 ,TXTE?
05 04260 000000 <6><0>
06 04261 000000 <6><0>
07 04262 000000 <6><0>
08 04263 000000 <6><0>
09 04264 000000 <6><0>
10 04265 000000 <6><0>
11 04266 000000 <6><0>
12 04267 000000 <6><0>
13 04270 000000 <6><0>
14 04271 004400 <0><9>
15 04272 005000 <0><10>
16 04273 000000 <6><0>
17 04274 006000 <0><12>
18 04275 126400 <0><13>
19 04276 000000 <6><0>
20 04277 000000 <6><0>
21 04300 000000 <6><0>
22 04301 000000 <6><0>
23 04302 000000 <6><0>
24 04303 000000 <6><0>
25 04304 000000 <6><0>
26 04305 000000 <6><0>
27 04306 000000 <6><0>
28 04307 000000 <6><0>
29 04310 000000 <6><0>
30 04311 000000 <6><0>
31 04312 000000 <6><0>
32 04313 000000 <6><0>
33 04314 000000 <6><0>
34 04315 000000 <6><0>
35 04316 000000 <6><0>
36 04317 000000 <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>
68 04357 135400 <0><59>

1 0056 ,MAIN

32 04360 137000 <3><62>
33 04361 155400 <0><91>
34 04362 055000 <0><90>
35 04363 254400 <0><89>
36 04364 154000 <0><89>
37 04365 153400 <0><87>
38 04366 053000 <0><86>
39 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 036000 <0><12>
35 04421 155400 <0><91>
36 04422 055000 <0><90>
37 04423 254400 <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 000000 <6><0>?
04460 000000

1 0357 MAIN

31
32 000010 ,RDX 8
33 PTAB4;
34 ,TXTE?
35 04461 000000 <6><2>
36 04462 000000 <6><0>
37 04463 000000 <6><2>
38 04464 000000 <6><2>
39 04465 000000 <6><2>
10 04466 000000 <6><0>
11 04467 000000 <6><0>
12 04470 000000 <6><0>
13 04471 000000 <6><0>
14 04472 004400 <0><11>
15 04473 005000 <0><12>
16 04474 000000 <6><0>
17 04475 006000 <0><14>
18 04476 106400 <0><15>
19 04477 000000 <6><0>
20 04500 000000 <6><0>
21 04501 000000 <6><2>
22 04502 000000 <6><0>
23 04503 000000 <6><0>
24 04504 000000 <6><0>
25 04505 000000 <6><0>
26 04506 000000 <6><0>
27 04507 000000 <6><0>
28 04510 000000 <6><0>
29 04511 000000 <6><0>
30 04512 000000 <6><0>
31 04513 000000 <6><0>
32 04514 000000 <6><0>
33 04515 000000 <6><0>
34 04516 000000 <6><0>
35 04517 000000 <6><0>
36 04520 000000 <6><0>
37 04521 117400 <0><37>
38 04522 022000 <2><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 <2><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 <2><42>
66 04556 030000 <2><60>
67 04557 124400 <0><51>
68 04560 125000 <0><52>

1 2058 MAIN

02 04501 057400 <0><137>
03 04502 147400 <0><117>
04 04503 050000 <0><120>
05 04504 150400 <0><121>
06 04505 151000 <0><122>
07 04506 051400 <0><123>
08 04507 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 027400 <0><41>
30 04615 123000 <0><46>
31 04616 023400 <0><47>
32 04617 024000 <0><50>
33 04620 121400 <0><43>
34 04621 036000 <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 024400 <0><11>
65 04600 020000 <6><0>?
04601 020000

1 0059 ,MAIN

```
01
02 ;POWER RESTART ROUTINE
03
04 04662 026454 POWON: LDA 1,0PSTAC ;GET PRINT INHI INITIAL FOR PRINT
05 04663 046454 STA 1,0PSETP
06 04664 062677 ICRST
07 04665 026061 C*AIT ;WAIT 3 SECONDS TO TERMINAL READY
08 04666 021236 SEC3
09 04667 020442 LDA 0,PCOTT ;PRINT 5 CR, LF FOR TTY, SILENT
10 04670 040442 STA 0,PCOUN
11 04671 006043 CCRLF
12 04672 014436 DSZ PCCUN
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 C*AIT ;20 MSEC FOR CRT
17 04677 021237 SEC*2
18 04700 020433 LDA 0,PCH35 ;HOME UP FOR CRT
19 04701 006041 CCHAR
20 04702 026061 C*AIT ;20 MSEC FOR CRT
21 04703 021237 SEC*2
22 04704 020433 LDA 0,PCH37 ;ERASE EOF FOR CRT
23 04705 006041 CCHAR
24 04706 006061 C*AIT ;20 MSEC FOR CRT
25 04707 021237 SEC*2
26 04710 026040 CMESS
27 04711 004745 MPOWC ;POWER
28 04712 026043 CCRLF
29 04713 026040 CMESS
30 04714 037245 PROG ;ACTUAL PROG NAME
31 04715 006071 RPSAG: CGUES
32 04716 004755 MSAGU ;SET SWITCHES, START ADDR
33 04717 004753 DSAGU
34 04720 004735 PSAAN ;SUGGESTED ANSWER
35 04721 006054 CTZCC
36 04722 006064 CDZCC
37 04723 026104 CGTCK ;READ ANSWER
38 04724 000472 JMP .+2 ;SUGGESTED ACCEPTED BY OPERATOR
39 04725 000773 JMP RPSAG ;ERROR RETURN
40 04726 030075 LDA 0,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 S*ISA: LDA 1,0PSTAC ;GET PRINT INHI INITIAL FOR PRINT
56 04741 046776 STA 1,0PSETP
57 04742 000753 JMP RPSAG ;USE POWER RESTART ROUTINE
```

1 0360 ,MAIN

01

02

04743 005015
04744 000000

MCRLF: .TXT !<15><12>! ;"<15><12>"

03

04

04745 047520
04746 042527
04747 020122

MPOWER: .TXT !POWER! ;"POWER"

05

06

04750 042523
04751 031505
04752 031056
04753 051454
04754 000101

DSAQU: .TXT !SEE3.2,SA! ;"SEE3.2,SA"

07

08

04755 042523
04756 020124
04757 053523
04760 052111
04761 044123
04762 051505
04763 052040
04764 020117
04765 047503
04766 052116
04767 047522
04770 026114
04771 024040
04772 027063
04773 024462
04774 020254
04775 052123
04776 051101
04777 040524
05000 042104
05001 000122

MSAQU: .TXT !SET SWITCHES TO CONTROL, (3.2), STARTADDR!

09

10

11

05002 005015
05003 050103
05004 020125
05005 054524
05006 042520
05007 020072
05010 000000

;"SET SWITCHES TO CONTROL, (3.2), STARTADDR"

MCPUT: .TXT !<15><12>CPU TYPE: ! ;"<15><12>CPU TYPE: "

1 0061 ,MAIN

```
31
32 ;ROUTINE TO EXAMINE MEMORY.
33
34 05211 165000 EXMEM: MOV 3,1 ;
35 05212 006072 CSAMS ;START ADDR MESSAGE
36 05213 006071 EXMMF: CGUES
37 05214 005270 MXMMF ;EXAMINE MEM FROM
38 05215 005101 DXMMF ;SEE NEXT QUESTION, TO (INCL)
39 05216 005064 XFROM ;SUGGESTED ANSWER
40 05217 006052 CTOCT
41 05220 006056 CDOCT
42 05221 006104 CGTCK ;READ ANSWER
43 05222 000402 JMP .+2 ;SUGGESTED ACCEPTED
44 05223 000770 JMP EXMMF ;ERROR RETURN
45 05224 000075 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 05225 000440 STA 0,FMADR ;INPUT ACCEPTED
24 05226 006071 EXMNT: CGUES
25 05227 005106 MXMNT ;TO INCL.
26 05230 005106 MXMNT ;USE THE SAME MESS AT DIS
27 05231 005066 XTOIN ;SUGGESTED ANSWER
28 05232 006052 CTOCT
29 05233 006056 CDOCT
30 05234 006104 CGTCK ;READ ANSWER
31 05235 000402 JMP .+2 ;SUGGESTED ACCEPTED
32 05236 000770 JMP EXMNT ;ERROR RETURN
33 05237 000275 LDA 0,DIGIN ;ANSWER INPUT'ED
34 05240 000427 STA 2,LMADR ;INPUT ACCEPTED
35 05241 006043 EXPRT: CCRLF
36 05242 006046 COICL
37 05243 000402 LDA 1,FMADR
38 05244 006052 CTOCT
39 05245 006056 CDOCT
40 05246 000417 LDA 2,FMADR
41 05247 000000 LDA 1,0,2
42 05250 006052 CTOCT
43 05251 006056 CDOCT
44 05252 006047 CDATT
45 05253 000414 LDA 1,LMADR
46 05254 000411 LDA 2,FMADR
47 05255 000410 ISZ FMADR ;TO NEXT LOC
48 05256 000401 JMP .+1
49 05257 000414 SUB# 1,2,SZR ;LAST LOC ?
50 05260 000761 JMP EXPRT ;NO, NEXT
51 05261 006043 CCRLF ;YES, PROGRAM FINISHED
52 05262 000401 JMP 0,+1
53 05263 000474 S*ISA ;RESTART MAIN PROGRAM
54
55 05264 000034 XFROM: 34 ;FIRST MEM LOC QUES
56 05265 000000 FMADR: 0 ;ANSWER
57 05266 000037 XTOIN: 37 ;LAST MEM LOC QUES, INCL
58 05267 000000 LMADR: 0 ;ANSWER
```

1 0062 MAIN

01

02

05070 054105
05071 046521
05072 047111
05073 020105
05074 042515
05075 020115
05076 051106
05077 046517
05100 000040

MXMMF: .TXT IEXAMINE MEM FROM ; "EXAMINE MEM FROM "

03

04

05101 020130
05102 020115
05103 051106
05104 046517
05105 000040

DXMMF: .TXT IX M FROM ; "X M FROM "

05

06

05106 047524
05107 044440
05110 041516
05111 027114
05112 000040

MXMMT: .TXT ITO INCL. ; "TO INCL. "

07

08

05113 042504
05114 047520
05115 044523
05116 035124
05117 000040

MDMMC: .TXT IDEPOSIT: ; "DEPOSIT: "

09

10

05120 051106
05121 046517
05122 046040
05123 041517
05124 000040

MDMMF: .TXT IFROM LOC ; "FROM LOC "

11

12

13

05125 17777 DPCON: 17777
05126 000000 CMADR: 0

;DEPOSIT CONTENT QUES
;ANSWER

```

1 0263 ,MAIN
01
02 ;ROUTINE TO DEPOSIT IN MEMORY.
03
24 05127 005000 DPMEM: MCV 3,1
25 05130 006072 CSAMS ;START ADDR MESSAGE
26 05131 006071 DPMHC: CGUES
27 05132 005113 MDMHC ;DEPOSIT:
28 05133 005113 MDMHC
29 05134 005125 DPCCN ;SUGGESTED ANSWER
10 05135 006052 CTOCT
11 05136 006056 CDOCT
12 05137 006104 CGTCK ;READ ANSWER
13 05140 000402 JMP .+2 ;SUGGESTED ACCEPTED
14 05141 000770 JMP DPMHC ;ERROR RETURN
15 05142 000275 LDA 0,DIGIN ;ANSWER INPUTTED
16 05143 000763 STA 0,CMADR ;INPUT ACCEPTED
17 05144 006071 DPMHF: CGUES
18 05145 005120 MDMHF ;FROM LOC
19 05146 005120 MDMHF
20 05147 005064 XFROM ;SUGGESTED ANSWER
21 05150 006052 CTOCT
22 05151 006056 CDOCT
23 05152 006104 CGTCK ;READ ANSWER
24 05153 000402 JMP .+2 ;SUGGESTED ACCEPTED
25 05154 000770 JMP DPMHF ;ERROR RETURN
26 05155 000275 LDA 0,DIGIN ;ANSWER INPUTTED
27 05156 000707 STA 0,FMADR ;INPUT ACCEPTED
28 05157 006071 DPMHT: CGUES
29 05160 005106 MXMHT ;TO INCL.
30 05161 005106 MXMHT
31 05162 005066 XTOIN ;SUGGESTED ANSWER
32 05163 006052 CTOCT
33 05164 006056 CDOCT
34 05165 006104 CGTCK ;READ ANSWER
35 05166 000402 JMP .+2 ;SUGGESTED ACCEPTED
36 05167 000770 JMP DPMHT ;ERROR RETURN
37 05170 000275 LDA 0,DIGIN ;ANSWER INPUTTED
38 05171 000766 STA 0,LMADR ;INPUT ACCEPTED
39 05172 000734 LDA 1,CMADR
40 05173 000672 DPPRT: LDA 2,FMADR
41 05174 005000 STA 1,0,2
42 05175 000672 LDA 3,LMADR
43 05176 000667 ISZ FMADR ;TO NEXT LOC
44 05177 000401 JMP .+1
45 05200 000414 SUB# 3,2,SZR ;LAST LOC ?
46 05201 000772 JMP DPPRT ;NO, NEXT
47 05202 000401 JMP .+.1 ;YES, PROGRAM FINISHED
48 05203 000400 SAISA ;RESTART MAIN PROGRAM

```



```

1 02764 ,MAIN
01
02 05204 000000 RQUES: 0 ;RETURN ADDR GUES ROUTINE
03 05205 000000 QUESA: 2 ;SUGG. ANSWER
04
05 05206 000077 MXQUE: .TXT 1?1 ;"?"
06
07 05207 020000 MX2SP: .TXT 1 1 ;"2 SPACE"
08 05210 000000
08
09 ;ROUTINE TO OUTPUT QUESTIONS.
10 ;HOW TO USE, SEE EXMEM.
11 ;CALL CGUES
12 ; MQUES ;LABEL TEXT TTO/LPT 1,9,17,25 LETTERS
13 ; DGUES ;LABEL TEXT DIS ALLWAYS 9 LETTERS
14 ; AQUES ;LABEL SUGGESTED ANSWER
15 ; CTCCT ;PRINT ROUTINE FOR SUGG. ANSWER
16 ; CDCCT ;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 1,2,3 ;1. PARAM
21 05213 044417 STA 1,GUESM
22 05214 025401 LDA 1,1,3 ;2. PARAM
23 05215 044413 STA 1,GUESD
24 05216 027402 LDA 1,2,3 ;3. PARAM
25 05217 044075 STA 1,DIGIN
26 05220 044765 STA 1,GUESA
27 05221 025403 LDA 1,3,3 ;4. PARAM
28 05222 044412 STA 1,GUEST
29 05223 025404 LDA 1,4,3 ;5. PARAM
30 05224 044411 STA 1,GUESS
31 05225 006046 CCICL
32 05226 006043 CCRLF
33 05227 006044 CDISP
34 05230 002000 QUESD: 2 ;DISPLAY MESSAGE
35 05231 006040 CMESS
36 05232 002000 QUESM: 2 ;PRINT MESSAGE
37 05233 024752 LDA 1,GUESA
38 05234 002000 QUEST: 2 ;NUMBER PRINT ROUTINE TTO/LPT
39 05235 002000 GUESS: 2 ;NUMBER PRINT ROUTINE DIS
40 05236 006044 CDISP
41 05237 005206 MXGLE
42 05240 006040 CMESS
43 05241 005206 MXGUE ;?
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

```

2365 ,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-INPLT
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 OKTALNUMBER 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.
```

1 0266 ,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

; OPERATING PROCEDURE:

; THE ROUTINES ARE CALLED BY THE INSTRUCTION: C>NAME<,
; WHICH EQUALS JSR "ROUTINE". AFTER RECEIVING A LEGAL NUMBER,
; CHARACTER OR TEXT-STRING FOLLOWED BY A TERMINATOR, THE
; PROGRAM RETURNS TO CALL+3. WHEN A FORMAT- OR OVERFLOWERROR
; OCCURS, THE RETURN IS CALL+2. WHEN ONLY A TERMINATOR IS
; INPUT, THE RETURN IS TO CALL+1.

; CALL CGTDC ;OR CGTOK, CGTBI, CGTSC, CGTTX
; TERM RETURN
; ERROR RETURN
; NORMAL RETURN

; TERMINATOR:

; IN ALL ROUTINES THE INPUTS: NL, LF, CR OR FF WILL
; TERMINATE THE MESSAGE.

; EVERY ERRORRETURN WILL INITIALISE THE BUFFERS, COUNTERS
; ETC., AND THE RE-INPUTTING THEN HAVE TO START AT THE LAST
; TERMINATION POINT. AN * IS PRINTED.

; CANCEL:

; IT'S POSSIBLE TO CANCEL THE LAST-KEYED DIGIT OR CHARACTER
; BY INPUTTING A RUBOUT, DEL OR CAN CHARACTER. THE PROGRAM
; ECHOES THEN A "S", AND WAITS FOR A NEW INPUT. IT IS NOT
; POSSIBLE TO CANCEL MORE THAN ONE CHAR. AN ATTEMPT WILL
; GIVE ERROR RETURN.

; THE NULL CHAR IS BLIND.

; IF A CHARACTER IS NOT TERMINATOR, BLIND, CANCEL OR
; ACCEPTED, IT IS ILLEGAL AND CAUSES ERROR RETURN.

1 0067 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 024443 JSR BZNUK ; CHECK BUSY NUK
08 05250 024431 JSR INTTI ; INPUTDEVICE = TTI
09 05251 024446 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 022421 JMP RAEND ; DIGIT FOUND
15 05257 125420 INC 1,1 ; AC1 = DIGIT
16 05260 014446 DSZ RADYN ; 0-9
17 05261 020774 JMP RANDI
18 05262 126420 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 05307 022424 JMP #GETRE
33
34 05301 063010 INTTI: SKPDN XTII ; IS TTI KEY PRESSED ?
35 05302 021400 JMP 0,3 ; NO, CHECK OTHER INPUT DEVICE
36 05303 024422 LDA 1,HC177 ; YES, GET CHAR
37 05304 020410 DIA 0,XTII
38 05305 123400 AND 1,2
39 05306 060110 NIOS XTII
40 05307 101215 MOV# 0,0,SNR ; NULL CHAR ?
41 05310 020771 JMP INTTI
42 05311 032413 JMP #GETRE
43
44 05312 126400 BZNUK: SUB 1,1 ; AC1:=0
45 05313 063434 AANUK: DIA 0,NUK ; TEST BUZY NUK
46 05314 122414 SUB# 1,2,SZR ; IS AC0=0 ?
47 05315 020776 JMP AANUK ; NO - KEYBOARD NOT READY
48 05316 031400 JMP 0,3 ; YES, RETURN.
49
50 05317 126400 INNUK: SUB 1,1 ; AC1:= 0
51 05320 063434 B3NUK: DIA 0,NUK ; GET KEY BITS
52 05321 122415 SUB# 1,2,SNR ; ARE THEY ZERO ?
53 05322 021776 JMP -2,3 ; YES, OTHER INPUT, TEST TTI
54 05323 021400 JMP 0,3 ; NO, KEY DEPRESSED
55
56 05324 020000 GETRE: 0
57 05325 020177 HC177: 177
58 05326 020000 RADYN: 2
59 05327 020012 RALIF: 12
60 05330 020040 RASPA: 40
61 05331 020253 RAPLU: 53
62 05332 020255 RAMIN: 55
63 05333 020060 RAZER: 60
64 05334 020030 RACAN: 30
```

1 0268 MAIN

```

21
22 05335 030550 DELTE: LDA      2,HC30 ; THE DELTE ROUTINE LOOKS FOR RUBOUT
23 05336 112415      SUB#    0,2,SNR ; IS AC0= 30 ?
24 05337 000405      JMP     DELOV  ; YES - RUBOUT
25 05340 030765      LDA      2,HC177 ; NO
26 05341 112415      SUB#    0,2,SNR ; IS AC0= 177 ?
27 05342 000402      JMP     DELOV  ; YES - RUBOUT
28 05343 001401      JMP     1,3    ; NO
29
30 05344 054461 DELOV: STA      3,DELRE ; CHECK FOR TOO MANY RUBOUTS
31 05345 030457      LDA      2,LASTN ; OR THE FIRST CHAR IS RUBOUT
32 05346 151102      MOVL    2,2,SZC ; IS LASTN = 177777 ?
33 05347 000417      JMP     ILLGR  ; YES - FIRST IS RUBOUT
34 05350 151102      MOVL    2,2,SZC ; IS LASTN = 077777 ?
35 05351 000415      JMP     ILLGR  ; YES - TOO MANY RUBOUTS
36 05352 152220      ADCZR   2,2    ; NO - PLACE 077777 IN LASTN
37 05353 050451      STA      2,LASTN ; TO INDICATE RUBOUT
38 05354 020533      LDA      0,HC44 ; TYPE S
39 05355 000402      JSR     TYPIN
40 05356 002447      JMP     0DELRE ; RETURN + 1
41
42 05357 054406 TYPIN: STA      3,TYPRE ; TYPE THE CHAR IN AC0 ON TTY/DIS
43 05360 000427      JSR     TERMT ; TERMINATOR ?
44 05361 002404      JMP     0TYPRE ; YES, DO NOT COPY
45 05362 0006045     CCOUT   ; TRY TO TYPE CHAR ON DISPLAY
46 05363 0006041     CCHAR  ; TRY TO TYPE CHAR ON TTY
47 05364 002401      JMP     0TYPRE
48 05365 000000 TYPRE: 0
49
50 05366 020521 ILLGR: LDA      0,HC44 ; ILLEGAL ROUTINE, TYPE S
51 05367 0004773     ILLEG: JSR     TYPIN ; COPY LAST CHAR
52 05370 020520      LDA      0,HC52
53 05371 0004766     JSR     TYPIN ; TYPE *
54 05372 126400      SUB     1,1    ; AC1:= 0
55 05373 044075      STA      1,DIGIN ; DIGIN:= 0
56 05374 046426     STA      1,XTXCO ; TEXIN:= CR,LF,0
57 05375 006426     JSR     0XTXND ; FOR EMPTY TEXTSTRING
58 05376 010423     ISZ    INRET  ; ILLEGAL OR OVERFLOW RETURN
59 05377 000403     JMP     ONTER  ; IS TO CALL + 2
60
61 05400 010421 INTER: ISZ    INRET  ; TERMINATION RETURN TO CALL+3
62 05401 010420     ISZ    INRET  ; FOR ACCEPTED INPUT
63 05402 006040     ONTER: CYES   ; TYPE CR,LF WITH MESS TO WAIT FOR
64 05403 0004743     MCRLF  ; DEVICE READY, RETURN IS TO CALL + 1
65 05404 006044     CDISP  ; FOR NO INPUT, ONLY TERMINATION
66 05405 0004743     MCRLF  ; (FOR ANSWER QUE, WITH AN OKAY).
67 05406 002413     JMP     0INRET ; RETURN TO MAIN PROGRAM
68
69 05407 030474 TERMT: LDA      2,HC13 ; THE TERMT ROUTINE LOOKS FOR TERM.
70 05410 112415     SUB#    0,2,SNR ; IS AC0=13 ?
71 05411 001401     JMP     1,3    ; YES - IT IS NOT A TERM
72 05412 030472     LDA      2,HC15 ; NO
73 05413 112433     SUBZ#   0,2,SAC ; IS AC0=<15 ?
74 05414 001401     JMP     1,3    ; NO, IT'S NOT A TERMINATOR
75 05415 030465     LDA      2,HC11 ; YES
76 05416 112432     SUBZ#   0,2,SZC ; IS AC0=< 11 ?
77 05417 001401     JMP     1,3    ; YES - IT IS NOT A TERMINATOR
78 05422 001400     JMP     0,3    ; NO - IT IS A TERMINATOR, RETURN
79
80 05421 000000 INRET: 0 ; RETURN ADDRESS TO MAIN PROGRAM
81 05422 006330 TXCO:  TXCCU ; ADDRESS OF TEXTBUFFER COUNTER
82 05423 026401 XTXND: TXEND ; ADDRESS OF TEXT TERM ROUTINE
83 05424 000000 LASTN: 0 ; X77777 FOR FIRST/RUBOUT, ELSE = CHAR
84 05425 000000 DELRE: 2 ; RETURN ADDRESS FOR DELTE

```

1 0009 MAIN

```
01
02 ; DECIMAL-NUMBER-INPUTROUTINE.
03
04 05426 054773 GETDC: STA 3,INRET ; INITIALIZE
05 05427 126200 ADC 1,1 ; AC1:=177777
06 05432 046540 STA 1,0LAST4 ; SET LAST4
07 05431 044534 STA 1,DSIGN ; SET DSIGN TO +
08 05432 126400 SUB 1,1 ; AC1:= 0
09 05433 044534 STA 1,NUMB4 ; CLEAR NUMBER
10 05434 004612 GETD1: JSR GETCH ; GET CHARACTER TO AC0
11 05435 004752 JSR TERMT ; TEST FOR TERMINATOR
12 05436 000450 JMP TERMDC ; IT IS A TERMINATOR
13 05437 004676 JSR DELTE ; IT'S NOT A TERMINATOR- IS IT A DEL ?
14 05447 007774 JMP GETD1 ; IT IS A DELETE-CHAR
15 05441 004412 JSR LETE4 ; IT'S NOT A DELETE-CHAR
16 05442 004510 JSR CHRAN ; PUT CHAR IN RANGE 0-9
17 05443 004527 JSR OFTDC ; TEST FOR OVERFLOW
18 05444 000403 JMP DELDC ; FIRST
19 05445 000402 JMP DELDC ; LAST IS RUBOUT
20 05446 004462 JSR PLADC ; ADD LAST4 TO NUMB4 * 10-DEC.
21 05447 004517 DELDC: LDA 1,DCDIG ; STORE NEW DIGIT IN LAST4
22 05452 046520 STA 1,0LAST4
23 05451 004706 ECHO4: JSR TYPIN ; TYPE NEW CHARACTER
24 05452 000762 JMP GETD1 ; GET NEXT CHARACTER
25
```

26 ; SUBROUTINES USED BY GETDC.

```
27
28 05453 054426 LETE4: STA 3,LERE4 ; LEGAL TEST
29 05454 034432 LDA 3,HC40
30 05455 111000 MCV 0,2 ; AC2:= CHAR
31 05456 024513 LDA 1,HC60
32 05457 116415 SUB# 0,3,SNR ; IS AC0 = 40 ?
33 05460 007771 JMP ECHO4 ; YES - THE CHAR IS A SPACE
34 05461 034427 LDA 3,HC52 ; NO
35 05462 116432 SUBZ# 0,3,SZC ; IS AC0 > 52 ?
36 05463 004704 JSR ILLEG ; NO - THE CHAR IS NOT LEGAL
37 05464 034427 LDA 3,HC72 ; YES
38 05465 116033 ADDZ# 0,3,SN0 ; IS AC0 < 72 ?
39 05466 004721 JSR ILLEG ; NO - NOT LEGAL
40 05467 034422 LDA 3,HC53 ; YES
41 05470 116415 SUB# 0,3,SNR ; IS AC0 >> 53 ?
42 05471 133000 ADD 1,2 ; NO - IT'S A PLUSSIGN, ADD 60
43 05472 034420 LDA 3,HC55 ; YES
44 05473 116415 SUB# 0,3,SNR ; IS AC0 >> 55 ?
45 05474 133000 ADD 1,2 ; NO - IT'S A MINUSSIGN ADD 60
46 05475 034474 LDA 3,HC60 ; YES
47 05476 156032 ADDZ# 2,3,SZC ; IS AC0 >= 60 ?
48 05477 000670 JSR ILLEG ; NO NOT LEGAL
49 05500 002401 JMP 0,LERE4 ; YES - IT'S A VALID DIGIT
50 05501 000000 LERE4: 0
51
52 05502 000011 HC11: 11
53 05503 000013 HC13: 13
54 05504 000015 HC15: 15
55 05505 000030 HC30: 30
56 05506 000040 HC40: 40
57 05507 000044 HC44: 44
58 05510 000052 HC52: 52
59 05511 000053 HC53: 53
60 05512 000055 HC55: 55
61 05513 000072 HC72: 72
```

1 0070 ,MAIN

```
01
02 05514 004456 TERMDC: JSR      OFTDC   ; TERMINATION, TEST OVERFLOW
03 05515 000665          JMP      ONTER   ; TERM IS FIRST
04 05516 000402          JMP      TERDC   ; LAST IS RUBOUT
05 05517 004411          JSR      PLADC   ; ADD LAST DIGIT TO NUMB4
06 05520 024447 TERDC:  LDA      1,NUMB4
07 05521 037444          LDA      2,DSIGN
08 05522 125015          MOV#    1,1,SAR ; IS AC1 = 0 ?
09 05523 002403          JMP      TERM1  ; YES - ZERO ALWAYS POSITIVE
10 05524 151014          MOV#    2,2,SZR ; IS DSIGN = 0 ?
11 05525 124400          NEG     1,1     ; NO - IT IS A NEGATIVE NUMBER
12 05526 044075 TERM1: STA      1,DIGIN ; YES - IT'S A POSITIVE NUMBER
13 05527 000651          JMP      INTER  ; OUTPUT TERM CHAR
14
15 05530 054421 PLADC:  STA      3,REPL4 ; PLACE LAST DIGIT/SIGN
16 05531 026437          LDA      1,#LAST4
17 05532 152400          SUB     2,2     ; AC2:= 0
18 05533 034756          LDA      3,HC53 ; LAST = + ?
19 05534 136415          SUB#    1,3,SAR ; NO
20 05535 000412          JMP      PLASI  ; YES, SET DSIGN
21 05536 152520          SUBZL  2,2     ; AC2:= 1
22 05537 034753          LDA      3,HC55 ; LAST = - ?
23 05540 136415          SUB#    1,3,SAR ; NO
24 05541 000406          JMP      PLASI  ; YES, SET DSIGN
25 05542 030425          LDA      2,NUMB4
26 05543 004413          JSR      MULTE  ; MULTIPLY NUMB4 WITH 10-DEC.
27 05544 133000          ADD     1,2     ; ADD LAST4 TO NUMB4*10
28 05545 050422          STA      2,NUMB4 ; PLACE NEW NUMBER
29 05546 002403          JMP      #REPL4
30 05547 050416 PLASI:  STA      2,DSIGN ; PLACE NEW SIGN
31 05550 002401          JMP      #REPL4 ; RETURN
32 05551 000000 REPL4:  2
33
34 05552 024417 CHRAN:  LDA      1,HC60 ; PUT CHAR IN RANGE 0 - 9 OR
35 05553 132400          SUB     1,2     ; PUT CHAR "+" TO 53 OR
36 05554 050412          STA      2,DCDIG ; PUT CHAR "-" TO 55
37 05555 001400          JMP      0,3    ; + AND - WAS ADDED 60 IN LEGAL TEST
38
39 05556 044406 MULTE:  STA      1,MULSA ; NUMBER TO BE MULT. BY 10 IN AC2
40 05557 145120          MOVZL  2,1     ; AC2=NUM, AC1=2*NUM
41 05560 125120          MOVZL  1,1     ; AC2=NUM, AC1=4*NUM
42 05561 133120          ADDZL  1,2     ; AC2=10*NUM, AC1=4*NUM
43 05562 024402          LDA      1,MULSA
44 05563 001400          JMP      0,3    ; RESULT IN AC2
45
46 05564 000000 MULSA:  0
47 05565 000000 DSIGN:  0 ; SIGNFLAG, 0=+, 1=-, 17777=NO SIGN=+
48 05566 000000 DCDIG:  0
49 05567 000000 NUMB4:  0
50 05570 045424 LAST4:  LASTN
51 05571 000060 HC60:   60
```

1 0071 MAIN

```
01
02 05572 054450 OFTDC: STA 3,RETDF ; OVERFLOW TEST FOR SIGN AND
03 05573 026775 LDA 1,0LAST4 ; (PREV*10)+LAST < LIMIT DX
04 05574 125102 MCVL 1,1,SZC ; IS THE LAST RUBBED OUT ?
05 05575 022445 JMP 0,RETDF ; OR IS IT THE FIRST ? YES, FIRST
06 05576 010444 ISZ RETDF
07 05577 125102 MCVL 1,1,SZC
08 05602 002442 JMP 0,RETDF ; YES, RUBOUT
09 05601 034710 LDA 3,HC53 ; NO, IS LAST A + ?
10 05602 136415 SUB# 1,3,SNR ; NO
11 05603 020432 JMP OFTSI ; YES
12 05604 034726 LDA 3,HC55 ; IS LAST A - ?
13 05605 136415 SUB# 1,3,SNR ; NO
14 05606 002427 JMP OFTSI ; YES
15 05607 176400 SUB 3,3 ; AC3:= 0
16 05610 032755 LDA 2,DSIGN ; IS DSIGN = 177777
17 05611 151112 MCVL# 2,2,SZC ; NO
18 05612 054753 STA 3,DSIGN ; YES, SET FIRST +
19 05613 030754 LDA 2,NUMB4 ; AC2:=PREVIOUS (PREV)
20 05614 155120 MOVZL 2,3 ; PREV*2
21 05615 175120 MCVZL 3,3 ; PREV*4
22 05616 175112 MCVL# 3,3,SZC ; PREV*8>=65536 IF
23 05617 006556 JSR 0,XILLG ; PREV >= 8192
24 05620 173122 ADDZL 3,2,SZC ; PREV*10>=65540 IF
25 05621 006554 JSR 0,XILLG ; PREV >= 6554
26 05622 151112 MCVL# 2,2,SZC ; PREV*10>=32770 IF
27 05623 006552 JSR 0,XILLG ; PREV >= 3277
28 05624 133200 ADD 1,2 ; (PREV*10)+LAST>32767
29 05625 151113 MCVL# 2,2,SAC ; (<=32760)+0...9>32767 ?
30 05626 000412 JMP OFTRE ; NO, RETURN
31 05627 034736 LDA 3,DSIGN ; YES, TEST +/- 32768,32769
32 05630 175205 MOV 3,3,SNR ; TEST SIGN
33 05631 006544 JSR 0,XILLG ; +32768, +32769
34 05632 151134 MCVZL# 2,2,SZR ; -32768, OK RETURN
35 05633 006542 JSR 0,XILLG ; -32769
36 05634 000404 JMP OFTRE ; RETURN
37
38 05635 032730 OFTSI: LDA 2,DSIGN ; OVERFLOW TEST SIGN
39 05636 151113 MCVL# 2,2,SAC ; IS IT FIRST SIGN ? YES
40 05637 006536 JSR 0,XILLG ; NO
41 05640 010402 OFTRE: ISZ RETDF ; PASS RUBOUT/FIRST RETURN
42 05641 002401 JMP 0,RETDF ; RETURN
43
44 05642 000200 RETOF: 0 ; RETURN ADDRESS
```


1 0372 MAIN

01

02 ; CHARACTER-INPUT-ROUTINE

03

```
04 05643 056537 GETSC: STA 3,0XINRT
05 05644 126000 ADC 1,1 ; AC1:=177777
06 05645 046440 STA 1,0LAST1 ; SET LAST1
07 05646 126400 SUB 1,1 ; AC1:= 0
08 05647 044435 STA 1,NUMSC ; CLEAR NUMSC
09 05650 006530 GETS1: JSR 0XGTCH ; GET CHAR TO AC0
10 05651 006526 JSR 0XTRMT ; IS IT A TERMINATOR ?
11 05652 000453 JMP TERMSC ; YES
12 05653 006523 JSR 0XDLTE ; NO - IS IT A DEL CHAR ?
13 05654 000774 JMP GETS1 ; YES
14 05655 004410 JSR LETE1 ; NO - IS IT LEGAL ?
15 05656 004430 JSR OFTSC ; IT'S A LEGAL CHAR, TEST OVERFLOW
16 05657 000403 JMP DELSC ; FIRST
17 05660 000402 JMP DELSC ; LAST IS RUBCUT
18 05661 004436 JSR PLASC ; PLACE LAST IN NUMSC
19 05662 042423 DELSC: STA 0,0LAST1 ; STORE NEW CHAR IN LAST1
20 05663 006516 JSR 0XTYPN ; TYPE NEW CHARACTER
21 05664 000764 JMP GETS1 ; GET NEXT INPUT
```

22

23 ; SUBROUTINES USED BY GETSC

24

```
25 05665 054413 LETE1: STA 3,LERE1 ; LEGAL TEST
26 05666 030413 LDA 2,0IC40
27 05667 034414 LDA 3,0IC11
28 05670 024412 LDA 1,0IC33
29 05671 106415 SUB# 0,1,0SR ; IS AC0 = 33 ?
30 05672 002406 JMP 0LERE1 ; YES
31 05673 116415 SUB# 0,3,0SR ; NO - IS AC2 = 11 ?
32 05674 002404 JMP 0LERE1 ; YES
33 05675 112032 ADD# 0,2,0SZC ; NO - IS AC2 >= 40 ?
34 05676 006477 JSR 0XILLG ; NO
35 05677 002401 JMP 0LERE1 ; YES - IT IS A LEGAL CHAR
```

36

05700 000000 LERE1: 0

37

```
38 05701 000040 H1C40: 40
39 05702 000033 H1C33: 33
40 05703 000011 H1C11: 11
41 05704 000000 NUMSC: 0
42 05705 005424 LAST1: LASTN
```

43

```
44 05706 026777 OFTSC: LDA 1,0LAST1 ; OVERFLOW TEST
45 05707 125102 MOVL 1,1,0SZC ; IS THE LAST CHAR RUBBED OUT ?
46 05710 001400 JMP 2,3 ; OR IS IT THE FIRST ? YES, FIRST
47 05711 125122 MOVL 1,1,0SZC
48 05712 001401 JMP 1,3 ; YES, RUBCUT
49 05713 030771 LDA 2,NUMSC ; NUMSC = 0 FOR FIRST
50 05714 151004 MOV 2,2,0SZR ; IS IT THE FIRST CHAR ?
51 05715 006460 JSR 0XILLG ; NO, OVERFLOW
52 05716 001402 JMP 2,3 ; YES, RETURN
```

53

05717 032766 PLASC: LDA 2,0LAST1 ; PLACE LAST CHAR

55

05720 050764 STA 2,NUMSC

56

05721 001400 JMP 2,3

57

05722 000764 TERMSC: JSR OFTSC ; TERMINATION, TEST OVERFLOW

59

05723 002461 JMP 0XCNTN ; TERM IS FIRST

60

05724 002402 JMP TERSC ; LAST IS RUBCUT

61

05725 004772 JSR PLASC ; STORE LAST CHAR IN NUMSC

62

05726 024756 TERSC: LDA 1,NUMSC ; MOVE CHAR TO DIGIN

63

05727 044075 STA 1,DIGIN

64

05730 002453 JMP 0XINTR ; OUTPUT TERM CHAR

1 0073 MAIN

01

02

; OCTAL-NUMBER-INPUTROUTINE

03

```
04 05731 056451 GETOK: STA 3,#XINRT
05 05732 126000 ADC 1,1 ; AC1:=177777
06 05733 046506 STA 1,#LAST2
07 05734 126400 SUB 1,1 ; AC1:= 0
08 05735 044505 STA 1,NUMB2
09 05736 006442 GETO1: JSR 0XGTCH ; GET CHAR TO AC2
10 05737 026440 JSR 0XTRMT ; IS IT A TERMINATOR ?
11 05740 000503 JMP TERMOK ; YES
12 05741 026435 JSR 0XDLTE ; NO - IS IT A DEL CHAR ?
13 05742 000774 JMP GETO1 ; YES
14 05743 000412 JSR LETE2 ; NO - IS IT LEGAL ?
15 05744 000441 JSR CHRA2 ; IT IS A LEGAL DIGIT
16 05745 000445 JSR OFTE2 ; TEST FOR OVERFLOW
17 05746 000423 JMP DELOK ; FIRST
18 05747 000432 JMP DELOK ; LAST IS RUBOUT
19 05750 000460 JSR PLAOK ; ADD LAST2 TO NUMB2 * 8-DEC
20 05751 024423 DELOK: LDA 1,OKDIG ; STORE NEW DIGIT IN LAST2
21 05752 046467 STA 1,#LAST2
22 05753 006426 ECHO2: JSR 0XTYPN ; TYPE NEW CHARACTER
23 05754 002762 JMP GETO1 ; GET NEXT CHARACTER
```

24

; SUBROUTINES USED BY GETOK

25

26

```
27 05755 054413 LETE2: STA 3,LERE2 ; LEGAL TEST
28 05756 034413 LDA 3,H2C40
29 05757 116415 SUB# 0,3,SNR ; IS AC2 = 42 ?
30 05760 002773 JMP ECHO2 ; YES - THE CHAR IS A SPACE
31 05761 034411 LDA 3,H2C60
32 05762 116032 ADC# 0,3,SZC ; IS AC2 >= 62 ?
33 05763 006412 JSR 0XILLG ; NO - NOT LEGAL
34 05764 034427 LDA 3,H2C70 ; YES
35 05765 116233 ADC# 0,3,SN0 ; IS AC2 < 72 ?
36 05766 006407 JSR 0XILLG ; NO - NOT LEGAL
37 05767 002401 JMP 0LERE2 ; YES - IT IS A VALID DIGIT
38 05770 000000 LERE2: 0
```

39

```
40 05771 000040 H2C40: 42
41 05772 000060 H2C60: 62
42 05773 000070 H2C70: 72
43 05774 000000 OKDIG: 0
```

44

```
45 05775 005367 XILLG: ILLEG
46 05776 005335 XDLTE: DELTE
47 05777 005427 XTRMT: TRMT
48 06000 005246 XGTCH: GETCH
49 06001 005357 XTYPN: TYPIN
50 06002 005421 XINRT: INRET
51 06003 005400 XINTR: INTER
52 06004 005402 XONTR: ONTR
```

53

```
54 06005 024765 CHRA2: LDA 1,H2C60
55 06006 111200 MOV 0,2
56 06007 132400 SUB 1,2 ; PUT CHAR IN RANGE 0-7
57 06010 050764 STA 2,OKDIG
58 06011 001400 JMP 0,3
```

```

1 0074 ,MAIN
01
02 00012 054415 OFTE2: STA 3,REOF2 ; OVERFLOW TEST
03 00013 030427 LDA 2,NUMB2
04 00014 026425 LDA 1,0LAST2 ; IS THE LAST DIGIT RUBBED OUT ?
05 00015 125102 MOVL 1,1,SZC ; OR IS IT THE FIRST ?
06 00016 002411 JMP #REOF2 ; YES, FIRST
07 00017 010410 ISZ REOF2
08 00020 125102 MOVL 1,1,SZC
09 00021 002406 JMP #REOF2 ; YES, RUBOUT
10 00022 034410 LDA 3,OF200 ; NO
11 00023 156433 SUBZ# 2,3,SNC ; IS NUMB2 <= 17777 ?
12 00024 006751 JSR #XILLG ; NO = OVERFLOW
13 00025 010402 ISZ RECF2 ; YES
14 00026 002421 JMP #REOF2 ; RETURN
15 00027 000000 RECF2: 0
16
17 00030 026411 PLACK: LDA 1,0LAST2 ; PLACE LAST DIGIT
18 00031 030411 LDA 2,NUMB2
19 00032 151120 MOVZL 2,2 ; MULTIPLY NUMB2 WITH 8-DEC
20 00033 151120 MOVZL 2,2
21 00034 151120 MOVZL 2,2
22 00035 133000 ADD 1,2 ; ADD LAST DIGIT
23 00036 050404 STA 2,NUMB2
24 00037 001400 JMP 0,3
25
26 00040 017777 OF200: 17777
27 00041 005424 LAST2: LASTN
28 00042 000000 NUMB2: 2
29
30 00043 000747 TERMCK: JSR OFTE2 ; TERMINATION, TEST OVERFLOW
31 00044 002740 JMP #XCNTN ; TERM IS FIRST
32 00045 000402 JMP TERCK ; LAST IS RUBOUT
33 00046 004762 JSR PLACK ; ADD LAST DIGIT TO NUMB2
34 00047 024773 TERCK: LDA 1,NUMB2
35 00050 044075 STA 1,DIGIN
36 00051 002732 JMP #XINTR ; OUTPUT TERM CHAR

```

1 0075 .MAIN

```
01  
02 ; BINARY-NUMBER-INPUTROUTINE  
03  
04 06052 056730 GETBI: STA 3,0XINRT  
05 06053 126000 ADC 1,1 ; AC1:=177777  
06 06054 046474 STA 1,0LAST3  
07 06055 126400 SUB 1,1 ; AC1:= 0  
08 06056 044473 STA 1,NUMB3  
09 06057 006721 GETBI: JSR 0XGTCH ; GET CHAR TO ACC  
10 06060 006717 JSR 0XTRMT ; IS IT A TERMINATOR ?  
11 06061 000471 JMP TERMBI ; YES  
12 06062 006714 JSR 0XDLTE ; NO - IS IT A DEL CHAR ?  
13 06063 000774 JMP GETBI ; YES  
14 06064 000412 JSR LETE3 ; NO - IS IT LEGAL ?  
15 06065 000431 JSR CHRA3 ; IT IS A LEGAL DIGIT  
16 06066 000435 JSR OFTE3 ; TEST FOR OVERFLOW  
17 06067 000403 JMP DELBI ; FIRST  
18 06070 000422 JMP DELBI ; LAST IS RUBOUT  
19 06071 000450 JSR PLABI ; ADD LAST3 TO NUMB3 * 2-DEC  
20 06072 024423 DELBI: LDA 1,BIDIG ; STORE NEW DIGIT IN LAST3  
21 06073 046455 STA 1,0LAST3  
22 06074 006705 ECHO3: JSR 0XTYPN ; TYPE NEW CHARACTER  
23 06075 000762 JMP GETBI ; GET NEXT CHARACTER  
24  
25 ; SUBROUTINES USED BY GETBI  
26
```

```
27 06076 054413 LETE3: STA 3,LERE3 ; LEGAL TEST  
28 06077 034413 LDA 3,H3C40  
29 06100 116415 SUB# 2,3,SAR ; IS ACC = 40 ?  
30 06101 000773 JMP ECHO3 ; YES - THE CHAR IS A SPACE  
31 06102 034411 LDA 3,H3C60 ; NO  
32 06103 116032 ADCZ# 2,3,SZC ; IS ACC >= 60 ?  
33 06104 006671 JSR 0XILLG ; NO - NOT LEGAL  
34 06105 034407 LDA 3,H3C62 ; YES  
35 06106 116033 ADCZ# 2,3,SAC ; IS ACC < 62 ?  
36 06107 006666 JSR 0XILLG ; NO - NOT LEGAL  
37 06110 002401 JMP 0LERE3 ; YES  
38 06111 000000 LERE3: 0
```

```
39  
40 06112 000040 H3C40: 40  
41 06113 000060 H3C60: 60  
42 06114 000062 H3C62: 62  
43 06115 000000 BIDIG: 0  
44
```

```
45  
46 06116 024775 CHRA3: LDA 1,H3C60 ; PUT CHAR IN RANGE 0-1  
47 06117 111000 MOV 0,2  
48 06120 132400 SUB 1,2  
49 06121 050774 STA 2,BIDIG  
50 06122 001400 JMP 2,3
```

1 2076 ,MAIN

```
01-
02 06123 054415 OFTE3: STA 3,REOF3 ; OVERFLOW TEST
03 06124 030425 LDA 2,NUMB3
04 06125 026423 LDA 1,@LAST3 ; IS THE LAST DIGIT RUBBED OUT ?
05 06126 125102 MOVL 1,1,SZC ; OR IS IT THE FIRST ?
06 06127 002411 JMP @REOF3 ; YES, FIRST
07 06130 010410 ISZ REOF3
08 06131 125102 MOVL 1,1,SZC
09 06132 002406 JMP @REOF3 ; YES, RUBOUT
10 06133 034414 LDA 3,CF300 ; NO
11 06134 156433 SUBZ# 2,3,SAC ; IS NUMB3 <= 177 ?
12 06135 006640 JSR @XILLG ; NO - OVERFLOW
13 06136 012402 ISZ RECF3 ; YES
14 06137 002401 JMP @REOF3 ; RETURN
15 06140 000000 REOF3: 0
16
17 06141 026407 PLABI: LDA 1,@LAST3 ; PLACE LAST DIGIT
18 06142 030407 LDA 2,NUMB3
19 06143 151120 MOVZL 2,2 ; MULTIPLY WITH 2-DEC
20 06144 133000 ADD 1,2 ; ADD LAST DIGIT
21 06145 050404 STA 2,NUMB3
22 06146 001400 JMP 0,3
23
24 06147 000177 OF300: 177
25 06150 005424 LAST3: LASTN
26 06151 000000 NUMB3: 0
27
28 06152 004751 TERMBI: JSR OFTE3 ; TERMINATION, TEST OVERFLOW
29 06153 002631 JMP @XCNTB ; TERM IS FIRST
30 06154 000402 JMP TERBI ; LAST IS RUBOUT
31 06155 004764 JSR PLABI ; ADD LAST DIGIT TO NUMB3
32 06156 024773 TERBI: LDA 1,NUMB3 ; MOVE THE BINARY NUMBER TO DIGIN
33 06157 044075 STA 1,DIGIN
34 06160 002623 JMP @XINTR ; OUTPUT TERM CHAR
```

1 0077 ,MAIN

01
02 ; TEXT-INPUT-ROUTINE
03

```

04 00161 050555 GETTX: STA 3, #YINRT
05 00162 126000 ADC 1,1 ; AC1:=177777
06 00163 046543 STA 1, #LASTS
07 00164 126400 SUB 1,1 ; AC1:= 0
08 00165 044543 STA 1, TXCOU
09 00166 006543 GETT1: JSR #YGTCH ; GET CHAR TO AC0
10 00167 006543 JSR #YTRMT ; IS IT A TERMINATOR ?
11 00170 000551 JMP TERMXT ; YES
12 00171 006542 JSR #YDLTE ; NO - IS IT A DEL CHAR ?
13 00172 000774 JMP GETT1 ; YES
14 00173 004410 JSR LETES ; NO - IS IT LEGAL ?
15 00174 004423 JSR OFTTX ; TEST FOR OVERFLOW
16 00175 000403 JMP DELTX ; FIRST
17 00176 000402 JMP DELTX ; LAST IS RUBOUT
18 00177 004505 JSR PLATX ; STORE LASTS IN TEXTBUFFER
19 00200 042526 DELTX: STA 0, #LASTS ; STORE NEW CHAR IN LASTS
20 00201 006533 JSR #YTPN ; TYPE NEW CHAR
21 00202 000764 JMP GETT1 ; GET NEXT CHARACTER

```

22
23 ; SUBROUTINES USED BY GETTX
24

```

25 00203 054410 LETES: STA 3, LERES ; LEGAL TEST
26 00204 034410 LDA 3, H5C11
27 00205 030410 LDA 2, H5C40
28 00206 116415 SUB# 0,3, SNR ; IS AC0 = 11 ?
29 00207 002404 JMP #LERES ; YES
30 00210 112032 ADC# 0,2, SZC ; NO - IS AC0 >= 40 ?
31 00211 006524 JSR #YILLG ; NO - THE CHAR IS NOT LEGAL
32 00212 002401 JMP #LERES ; YES
33 00213 000000 LERES: 0

```

```

34
35 00214 000011 H5C11: 11
36 00215 000040 H5C40: 40
37 00216 000117 H5C79: 117 ; TEXTBUFFER LENGTH-1, OKTAL

```

```

38
39 00217 026507 OFTTX: LDA 1, #LASTS ; OVERFLOW TEST
40 00220 125102 MOVL 1,1, SZC ; IS THE LAST CHAR RUBBED OUT ?
41 00221 001400 JMP 0,3 ; OR IS IT THE FIRST ? YES, FIRST
42 00222 125102 MOVL 1,1, SZC
43 00223 001401 JMP 1,3 ; YES, RUBOUT
44 00224 024504 LDA 1, TXCOU ; NO
45 00225 030771 LDA 2, H5C79
46 00226 132433 SUBZ# 1,2, SNC ; IS TXCOU <= 79-DEC
47 00227 006506 JSR #YILLG ; NO - OVERFLOW
48 00230 001402 JMP 2,3 ; YES, RETURN

```

```

49
50 00231 006232 TEXTIA: .+1 ; ADDRESS OF TEXTBUFFER
51 000051 TEXIN: .BLK 51 ; TEXTBUFFER 80 BYTES + CR,LF
52 00303 000000 TEXEN: 0 ; END OF TEXTBUFFER: NUL CHAR

```

1 0078 ,MAIN

```

01
02
03 06304 054421 PLATX: STA 3,REPL5 ; PLACE LAST CHAR
04 06305 026421 LDA 1,PLAST5 ; LAST5 TO CHAHA OR
05 06306 010422 ISZ TXCOU ; INCREMENT CHARACTER-COUNTER
06 06307 030421 LDA 2,TXCOU
07 06310 151213 MCVR# 2,2,SNC ; IS TXCOU EVEN ?
08 06311 007403 JMP STCTX ; YES
09 06312 044415 STA 1,CHAHA ; NO - MOVE LAST5 TO CHAHA
10 06313 002412 JMP 0REPL5
11 06314 030413 STOTX: LDA 2,CHAHA ; CHAHA IS CHAR HALF BUFFER
12 06315 125320 MOVZS 1,1
13 06316 147000 ADD 2,1 ; AC1 = "LAST5,CHAHA"
14 06317 030411 LDA 2,TXCOU ; CALCULATE ADDRESS OF BUFFEREND+1
15 06320 151220 MCVZR 2,2
16 06321 034710 LDA 3,TEXIA
17 06322 173000 ADD 3,2 ; AC2:= TEXIN+TXCOU/2
18 06323 045377 STA 1,-1,2 ; STORE LAST TWO CHAR INTO BUFFEREND+1
19 06324 002401 JMP 0REPL5
20 06325 002000 REPL5: 0
21
22 06326 005424 LAST5: LASTN
23 06327 000000 CHAHA: 0 ; THE CHAR BEFORE LAST5
24 06330 000000 TXCOU: 0 ; TEXT COUNTER BYTE ADDRESS
25
26 06331 005246 YGTCH: GETCH
27 06332 005407 YTRMT: TRMT
28 06333 005335 YDLTE: DELTE
29 06334 005357 YTYPN: TYPN
30 06335 005367 YILLG: ILLEG
31 06336 005421 YINRT: INRET
32 06337 005400 YINTR: INTR
33 06342 005402 YONTR: ONTR
34
35 06341 024656 TERMXT: JSR OFTTX ; TERMINATION, TEST OVERFLOW
36 06342 022776 JMP 0YCNTR ; TERM IS FIRST
37 06343 000402 JMP TERTX ; LAST IS RUBOUT
38 06344 004700 JSR PLATX ; PLACE THE CHAR BEFORE TERM CHAR
39 06345 024763 TERTX: LDA 1,TXCOU ; IN LAST5
40 06346 125213 MCVR# 1,1,SNC ; IS TXCOU EVEN ?
41 06347 000417 JMP NBEVEN ; YES
42 06350 020443 LDA 0,H5C15 ; NO
43 06351 121320 MCVZS 0,2
44 06352 032755 LDA 2,CHAHA
45 06353 113200 ADD 0,2 ; AC2 = "CR,CHAR"
46 06354 010754 ISZ TXCOU ; INCREMENT CHAR COUNT FOR CR
47 06355 004413 JSR COUDI
48 06356 004415 JSR STABU ; STORE AC2 INTO BUFFER
49 06357 030435 LDA 2,H5C12 ; AC2 = " 0,LF"
50 06360 004410 JSR COUDI
51 06361 125400 INC 1,1 ; INCREMENT BUFF ADDR FOR 0,LF
52 06362 004411 JSR STABU ; STORE AC2 INTO BUFFER
53 06363 030646 OUT5: LDA 2,TEXIA
54 06364 050275 STA 2,DIGIN ; ADDRESS OF TEXTBUFFER
55 06365 002752 JMP 0YINTR ; OUTPUT TERM CHAR
56
57 06366 004413 NBEVEN: JSR TXEND ; STORE LF,CR,0,0
58 06367 002774 JMP OUT5 ; TERMINATE

```

1 0079 MAIN

```
01
02 06370 024740 COUDI: LDA 1, TXCCU
03 06371 125220 MOVZ 1,1 ; DIVIDE TXCCU WITH 2
04 06372 001400 JMP 0,3
05
06 06373 054405 STABU: STA 3, RET5
07 06374 034635 LDA 3, TEXIA ; CALCULATE ADDRESS OF BUFFEREND
08 06375 137000 ADD 1,3
09 06376 051777 STA 2,-1,3 ; STORE AC2 INTO BUFFER
10 06377 002401 JMP #RET5
11 06400 000000 RET5: 2
12
13 06401 054411 TXEND: STA 3, TXNDR
14 06402 032413 LDA 2, HLFGR ; AC2 = "LF,GR"
15 06403 004765 JSR COUDI
16 06404 125400 INC 1,1
17 06405 004766 JSR STABU ; STORE AC2 INTO BUFFER
18 06406 125400 INC 1,1
19 06407 152400 SUB 2,2 ; AC2:= 0
20 06410 004763 JSR STABU ; STORE "0,0" INTO BUFFER
21 06411 002401 JMP #TXNDR
22 06412 000000 TXNDR: 0
23
24 06413 000015 H5C15: 15
25 06414 000012 H5C12: 12
26 06415 005015 HLFGR: 5215
27
28
29
```


1 0000 ,MAIN

```
01  
02  
03  
04  
05 ;TESTLOOP ROUTINE  
06 ;CALL SETPX ;SETP2,SETP1,SETP2  
07 ; ;PROGRAM LOOP  
08 ;  
09 ; EHALT ;ERROR HALT ROUTINE  
10 ; LOOP ;CYCLE LOOP ROUTINE  
11 ;  
12 ;NEXT TEST CYCLE  
13  
14 ;SETPX: ;ICRST AND SET # OF LOOPS IN  
15 ;FIRST CYCLUS ERROR CYCLUS IF SWITCH 0  
16 ;SETP2 1010 1011  
17 ;SETP1 1011 1012  
18 ;SETP2 1012 1012  
19 ;EHALT: ;IF NOT FIRST LOOP WITH ERROR IN A CYCLE:  
20 ; DO NOTHING, LOOP  
21 ;IF FIRST LOOP WITH ERROR IN A CYCLE:  
22 ; PRINT AC0, AC1, AC2 (NOT ON DIS) AND  
23 ; PRINT PC XXXXXX IF NOT SWITCH 10 (INHIBIT PRINT)  
24 ;IF FIRST LOOP WITH ERROR AT ALL:  
25 ; HALT IN EACH ERROR WITH AC3=PC  
26 ; OTHER ACS RELEVANT INFO.  
27 ;ELSE DO NOTHING, LOOP  
28 ;LOOP: ;IF CYCLE NOT FINISHED (# OF LOOP NOT FINISHED):  
29 ; IF NO ERRORS AT ALL UNTIL NOW:  
30 ; ICRST, LOOP  
31 ; IF ERROR AND SWITCH 0 = 0:  
32 ; PRINT FAILURE RATE OF LAST CYCLE IF  
33 ; SWITCH 11 = 1 AND SWITCH 10 = 0  
34 ; PROCEED TO NEXT TEST CYCLE  
35 ; ELSE: ICRST, LOOP  
36 ;IF CYCLE FINISHED:  
37 ; IF NO ERRORS AT ALL:  
38 ; PROCEED TO NEXT TEST CYCLE  
39 ; ELSE: PRINT FAILURE RATE OF LAST CYCLE IF  
40 ; SWITCH 11 = 1 AND SWITCH 10 = 0  
41 ; IF SWITCH 0 = 0:  
42 ; PROCEED TO NEXT TEST CYCLE  
43 ; ELSE: ICRST, LOOP.  
43 00416 054522 ENTP0: STA 3,LOOPR ;INITIALIZE EACH TEST  
44 00417 034504 LDA 3,ITRP1  
45 00420 054505 STA 3,ITRAG  
46 00421 176520 SUBZL 3,3 ;AC3:= 1  
47 00422 000412 JMP ENTC0  
48 00423 054515 ENTP1: STA 3,LOOPR  
49 00424 034500 LDA 3,ITRP2  
50 00425 054500 STA 3,ITRAG  
51 00426 034475 LDA 3,ITRP1  
52 00427 000405 JMP ENTC0  
53 00430 054510 ENTP2: STA 3,LOOPR  
54 00431 034473 LDA 3,ITRP2  
55 00432 054473 STA 3,ITRAG  
56 00433 034471 LDA 3,ITRP2  
57 00434 054472 ENTC0: STA 3,ITR  
58 00435 054472 STA 3,ITRCT  
59 00436 176400 SUB 3,3 ;AC3:= 0  
60 00437 054471 STA 3,EFLAG ;SET FIRST ERROR FLAG = 0  
61 00440 054471 STA 3,ERRCT ;SET ERROR COUNT = 0  
62 00441 054471 STA 3,EBFCT ;SET ERROR BUFFER COUNT = 0  
63 00442 054471 STA 3,EBFLG ;SET FIRST ERROR FLAG BUFFER = 0  
64 00443 034501 LDA 3,SETSW ;SET PRINT INHIBIT ON SW 10  
65 00444 054477 STA 3,0ISTAC ;IN FUNCTION  
66 00445 062477 DIC 0,CPU ;I/O RESET  
67 00446 002472 JMP 0,LOOPR ;LOOP ITERATE RETURN
```

1 2081 ,MAIN

```
01
02 06447 054465 CYCLE: STA 3,RETUR ;END OF TEST ITERATION ROUTINE
03 06450 054465 STA 2,CSAV2
04 06451 044465 STA 1,CSAV1 ;SAVE THE ACS'
05 06452 040465 STA 0,CSAV0
06 06453 020457 LDA 0,ERBCT ;ADD BUFFER COUNT
07 06454 024455 LDA 1,ERRCT ;TO ERROR COUNT
08 06455 107000 ADD 0,1
09 06456 044453 STA 1,ERRCT
10 06457 020454 LDA 0,EBFLG ;MOVE FIRST FLAG
11 06460 040450 STA 0,EFLAG ;BUFFER TO FLAG
12 06461 014446 DSZ ITRCT
13 06462 000416 JYP CYCTS ;NOT N TIMES ITERATED, LOOP
14 06463 030445 LDA 2,EFLAG ;CYCLUS FINISHED
15 06464 151005 MOV 2,2,SNR ;ERRORS AT ALL ?
16 06465 000432 JYP NOEX ;NO ERRORS
17 06466 000457 JSR FRATE ;YES, PRINT FAILURE RATE
18 06467 102400 SUB 0,2
19 06470 040441 STA 0,ERRCT ;RESET ERROR COUNT
20 06471 034434 LDA 3,ITRAG
21 06472 054434 STA 3,ITR
22 06473 054434 STA 3,ITRCT
23 06474 006073 CRES* ;READS 2 ROUTINE
24 06475 151112 MOVL# 2,2,SZC ;SWITCH 0 ?
25 06476 000412 JYP CYMOR ;(1) = LOOP IN ERROR
26 06477 000420 JYP NOEX ;(0) = PROCEED TO NEXT TEST
27
28 06500 034430 CYCTS: LDA 3,EFLAG ;LOOP FINISHED
29 06501 175005 MOV 3,3,SNR ;ERRORS UNTIL NOW ?
30 06502 000406 JYP CYMOR ;NO, LOOP
31 06503 026073 CRES* ;YES, READS 2 ROUTINE
32 06504 151112 MOVL# 2,2,SZC ;SWITCH 0 ?
33 06505 000403 JYP CYMOR ;(1) = LOOP IN ERROR
34 06506 000437 JSR FRATE ;PRINT FAILURE RATE
35 06507 000410 JYP NOEX ;(0)=PROCEED TO NEXT TEST
36
37 06510 062477 CYMOR: DTC 0,CPU ;I/O RESET
38 06511 174000 SUB 3,3 ;ACS:=0
39 06512 054420 STA 3,ERBCT ;RESET ERROR BUFFER COUNTER
40 06513 027424 LDA 0,CSAV0
41 06514 024422 LDA 1,CSAV1
42 06515 030420 LDA 2,CSAV2 ;RESTORE AC'S
43 06516 002422 JYP #LOOPR ;LOOP MORE
44
45 06517 020420 NOEX: LDA 0,CSAV0 ;EXIT TO NEXT TEST
46 06520 024416 LDA 1,CSAV1
47 06521 030414 LDA 2,CSAV2 ;RESTORE AC'S
48 06522 002412 JYP #RETUR ;PROCEED TO NEXT TEST
49
50 06523 000012 ITRP1: 12 ;1011 LOOP CONSTANT
51 06524 000144 ITRP2: 144 ;1012 LOOP CONSTANT
52 06525 000200 ITRAG: 0 ;# OF LOOP IF ERROR
53 06526 000200 ITR: 0 ;# OF LOOP IN CYCLUS ACTUAL
54 06527 000000 ITRCT: 0 ;LOOP COUNTER
55 06530 000000 EFLAG: 0 ;FIRST ERROR FLAG FOR ALL CYCLES
56 06531 000000 ERBCT: 0 ;ERROR COUNTER
57 06532 000000 ERRCT: 0 ;ERROR BUFFER COUNTER
58 06533 000000 EBFLG: 0 ;FIRST ERROR FLAG BUFFER
59 06534 000000 RETUR: 0
60 06535 000000 CSAV2: 0
61 06536 000000 CSAV1: 0
62 06537 000000 CSAV0: 0
63 06540 000200 LOOPR: 0
64 06541 000010 ERHSN: 10 ;SWITCH 12
65 06542 000437 IZOUT: BZOUT
66 06543 000574 ISTAC: SETAC
67 06544 000073 SETS*: CRES* ;FOR PRINT INHIBIT ROUTINE
```

1 2082 ,MAIN

```
01
02 06545 054432 FRATE: STA 3,FRATR ;PRINT FAILURE RATE
03 06546 006273 CRES* ;IF SWITCH 10 = 0
04 06547 024433 LDA 1,INH SW ;AND SWITCH 11 = 1
05 06550 133414 AND# 1,2, SZR
06 06551 002426 JMP @FRATR ;INHIBIT PRINTOUT
07 06552 024431 LDA 1,FRASW
08 06553 133415 AND# 1,2, SNR
09 06554 000420 JMP NORAT ;PRINT CRLF ONLY
10 06555 020423 LDA 0,CH40
11 06556 026041 CCHAR
12 06557 024752 LDA 1,ERRCT
13 06560 030744 LDA 2,ITRP2
14 06561 006066 MULTI ;AC1*AC2
15 06562 030744 LDA 2,ITR
16 06563 034744 LDA 3,ITRCT
17 06564 172400 SUB 3,2 ;AC2:=# OF LOOPS TILL NOW
18 06565 006070 DIVID ;AC0,AC1/AC2
19 06566 026053 CTDEC
20 06567 006057 CDDEC ;PRINT VALUE
21 06570 020411 LDA 0,PCENT ;EXAMPLE: 89%
22 06571 006045 CDOUT ;VALUE=ERRCT*100/(ITR-ITRCT)
23 06572 006041 CCHAR
24 06573 006047 CDATT
25 06574 006040 NORAT: CMESS ;PRINT CRLF WITH MESS TO WAIT
26 06575 004743 MCRLF ;FOR DEVICE READY BEFORE NEXT IORST.
27 06576 002401 JMP @FRATR
28 06577 000000 FRATR: 0
29 06600 000340 CH40: 40
30 06601 000245 PCENT: 245
31 06602 000040 INH SW: 40 ;SWITCH 10
32 06603 000020 FRASW: 20 ;SWITCH 11
33
34 MHEAD: .TXT !AC0<11>AC1<11>AC2<15><<12>PC<40>!
06604 041501
06605 004460
06606 041501
06607 004461
06610 041501
06611 006462
06612 050012
06613 020103
06614 000000
35 ;"AC0<11>AC1<11>AC2<15><<12>PC<40>"
36
37 MHEAD: .TXT !PC<40>! ;"PC<40>"
06615 041520
06616 000040
38
39 06617 054715 ERROR: STA 3,RETUR ;ERPOR SUBROUTINE
40 06620 034711 LDA 3,ERRCT
41 06621 175005 MCV 3,3, SNR ;FIRST ERROR LOOP ?
42 06622 000404 JMP ERRO1
43 06623 010707 ERRET: ISZ ERBCT ;COUNT BUFFER ERROR COUNTER
44 06624 002710 JMP @RETUR ;EXIT
45 06625 002707 JMP @RETUR ;EXIT IF SKIP
```

1 0063 .MAIN

```
01
02 00626 040711 ERROR1: STA 0,CSAV0 ;FIRST ERROR LOOP 1
03 00627 044707 STA 1,CSAV1 ;(IN FIRST CYCLUS
04 00630 050705 STA 2,CSAV2 ;OR ERROR CYCLUS)
05 00631 006073 CRESW ;READS 2 ROUTINE
06 00632 024750 LDA 1,INHSW ;SWITCH 10
07 00633 133414 AND# 1,2,SZR
08 00634 000455 JMP ERROR3 ;INHIBIT PRINTOUT
09 00635 006043 CCRLF
10 00636 024721 LDA 1,CSAV0
11 00637 006052 CTOCT
12 00640 024076 LDA 1,CSAV1
13 00641 006052 CTOCT
14 00642 024673 LDA 1,CSAV2
15 00643 006052 CTOCT ;PRINT AC'S ONLY AT TTY, LPT
16 00644 006043 CCRLF ;PRINT CARRIAGE
17 00645 006040 CYESS ;PRINT HEADER
18 00646 006604 MHEAD
19 00647 006046 CDICL
20 00650 006044 CDISP
21 00651 006615 DHEAD
22 00652 020662 LDA 0,RETUR
23 00653 126000 ACC 1,1
24 00654 107000 ADD 0,1
25 00655 006052 CTOCT
26 00656 026056 CDOCT ;PRINT PC OF ERROR
27 00657 006663 JSR 0,IEZOT ;WAIT FOR LPT/TTY BEFORE NEXT IORST
28 00660 020650 LDA 0,EFLAG
29 00661 101005 MOV 0,2,SNR
30 00662 000406 JMP ERROR2 ;FIRST ERROR LOOP AT ALL
31 00663 006047 CDATT
32 00664 020653 LDA 0,CSAV0
33 00665 024651 LDA 1,CSAV1
34 00666 030647 LDA 2,CSAV2 ;RESTORE ACS
35 00667 020734 JMP ERRET
36
37 00670 006050 ERROR2: CHAAT ;FIRST ERROR LOOP AT ALL: HALT
38 00671 126000 ACC 1,1 ;AC1:=177777
39 00672 044601 STA 1,EFLAG ;SET FIRST ERROR BUFFER FLAG
40 00673 006073 CRESW ;READS 2 ROUTINE
41 00674 034645 LDA 3,ERHSW
42 00675 157404 AND 2,3,SZR ;SW 12 ?
43 00676 000407 JMP ERRNH ;NO HALT
44 00677 034635 LDA 3,RETUR
45 00700 137000 ADD 1,3 ;ERROR, AC3=PC OF ERROR
46 00701 030634 LDA 2,CSAV2
47 00702 024634 LDA 1,CSAV1
48 00703 022034 LDA 0,CSAV0
49 00704 063077 HALT ;OPERATOR-SET SWITCHES!
50 00705 020632 ERRNH: LDA 0,CSAV0
51 00706 024633 LDA 1,CSAV1
52 00707 030626 LDA 2,CSAV2
53 00710 000713 JMP ERRET
54
55 00711 020617 ERROR3: LDA 0,EFLAG ;NO PRINTING
56 00712 101005 MOV 0,2,SNR
57 00713 000755 JMP ERROR2 ;FIRST ERROR LOOP AT ALL
58 00714 020623 LDA 0,CSAV0
59 00715 024621 LDA 1,CSAV1
60 00716 030617 LDA 2,CSAV2
61 00717 000704 JMP ERRET
```

1 0084 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 06720 050412 XSTAA: STA 3,RXAST
12 06721 010411 ISZ RXAST
13 06722 025400 LDA 1,0,3 ;AC1=EXPECTED BITS
14 06723 060417 SDEV1: DIA 0,DEV
15 06724 040407 STA 0, SXAST ;SAVE AC0
16 06725 123400 AND 1,2
17 06726 122415 SUB# 1,2,SNR
18 06727 010403 ISZ RXAST ;OK
19 06730 020403 LDA 0, SXAST ;RESTORE STATUS
20 06731 002401 JMP @RXAST
21 06732 000000 RXAST: 0
22 06733 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 06734 050407 XSTAN: STA 3,RXNST
35 06735 010406 ISZ RXNST
36 06736 025400 LDA 1,0,3 ;AC1=NONEXPECTED STATUS
37 06737 060417 SDEV2: DIA 0,DEV ;AC0=ACTUAL STATUS
38 06740 107415 AND# 0,1,SNR
39 06741 010402 ISZ RXNST ;OK
40 06742 002401 JMP @RXNST
41 06743 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 STATA
49 ; ARG
50 ;
51 06744 050413 XSTAW: STA 3,RXNST
52 06745 010412 ISZ RXNST
53 06746 025400 LDA 1,0,3
54 06747 006073 CRES# ;READS 2 ROUTINE
55 06750 030410 LDA 3, SMASK ;MASK FOR SWITCH REGISTER
56 06751 157400 AND 2,3
57 06752 107000 ADD 3,1 ;AC1=EXP STATUS
58 06753 060417 SDEV3: DIA 0,DEV ;AC0=ACTUAL STATUS
59 06754 106415 SUB# 0,1,SNR
60 06755 010402 ISZ RXNST ;OK
61 06756 002401 JMP @RXNST
62 06757 000000 RXNST: 0
63 06760 000000 SMASK: 000000 ;CHANGE MASK TO ONES FOR
64 ;THOSE BITS TO USE FROM SW.
```

1 0085 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 06761 054414 XSTAP: STA 3, RXPST
13 06762 010413 ISZ RXPST
14 06763 010412 ISZ RXPST
15 06764 025402 LDA 1,2,3 ;AC1:=EXPECTED PART
16 06765 031421 LDA 2,1,3 ;AC2:=MASK
17 06766 147402 AND 2,1
18 06767 060417 SDEV4: DIA 2,DEV ;AC0:=ACTUAL STATUS
19 06770 115023 MOV 0,3 ;AC3:=ACTUAL STATUS FOR MASK
20 06771 157402 AND 2,3
21 06772 136415 SUB= 1,3,SNR
22 06773 010402 ISZ RXPST ;OK
23 06774 022401 JMP @RXPST
24 06775 020023 RXPST: 2

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 STAS
33 ; ARG
34 ;

35 06776 054407 XSTAS: STA 3, RXSST
36 06777 010426 ISZ RXSST
37 07002 025402 LDA 1,2,3 ;AC1:=EXPECTED STATUS
38 07001 060417 SDEV5: DIA 2,DEV ;AC2:=ACTUAL STATUS
39 07002 127414 AND= 2,1,SNR
40 07003 010402 ISZ RXSST ;OK
41 07004 022401 JMP @RXSST
42 07005 020023 RXSST: 2

43
44 ;ROUTINE LOOP REPORT
45 ;

46 07006 054461 XLORE: STA 3, RPASS
47 07007 006040 CDICL
48 07010 026043 CCRLEF
49 07011 024450 LDA 1, RPASS
50 07012 152523 SUBZL 2,2 ;AC2:=1
51 07013 146402 SUB 2,1 ;SUBTRACT 1 FROM JSR ADDR
52 07014 026056 CDICL
53 07015 026052 CTCCT ;PRINT ADDR
54 07016 026044 CDISP
55 07017 037077 MLORE
56 07020 026042 CYESS
57 07021 037077 MLORE ;XXXXXX LOOP-ADDR
58 07022 026047 CDATT
59 07023 026444 JSR @RPASS ;RETURN TO START LOOP

1 0086 MAIN

01
02
03
04
05
06
07
08
09
10
11

```
;ROUTINE TO HANDLE PASS #
;CORRECT PASSC TO YOUR CHOICE OF # OF RUNS BETWEEN
;EACH PASS MESSAGE (2 MINUTES INTERVAL IS CONVENIENT).
;CORRECT TEXT MPASS, DPASS ACCORDINGLY.
;INITIALIZE WHEN PROGRAM STARTED/RESTARTED:
;      PASSN TO 0
;      PASSB TO # OF RUNS BETWEEN MESS = PASSC
;CALL: CPASS
;      RETURN
```

12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33

```
07020 054443 XPASS: STA 3,RPASS
07025 010443 DSZ PASSB ;RUN COUNT DOWN
07026 002441 JMP @RPASS ;NO MESSAGE
07027 002442 LDA 0,PASSC
07030 040440 STA 0,PASSB ;INITIALIZE RUN COUNT
07031 006046 CDICL ;WRITE PASS MESSAGE
07032 006043 CCRLF
07033 006040 CMESS
07034 005207 MX2SP
07035 010435 ISZ PASSN ;COUNT PASS #
07036 024434 LDA 1,PASSN
07037 006053 CTDEC
07040 006057 CDDEC
07041 006044 CDISP
07042 007061 DPASS
07043 006040 CMESS
07044 007050 MPASS ; . PASS OF 10 RUNS
07045 006043 CCRLF
07046 006047 CDATT
07047 002420 JMP @RPASS ;RETURN
```

34
35

```
07050 020256 MPASS: .TXT !. PASS OF 10 RUNS! ;". PASS OF 10 RUNS"
07051 040520
07052 051523
07053 047440
07054 020126
07055 030061
07056 051040
07057 047125
07060 000123
```

36
35

```
07061 050056 DPASS: .TXT !.PASS 10 R! ;".PASS 10 R"
07062 051501
07063 020123
07064 030061
07065 051040
07066 000200
```

36
37
38
39
40
41
42
43
44
45
46

```
07067 000000 RPASS: 0 ;RETURN ADDR
07070 000000 PASSB: 0 ;COUNTER FOR # OF RUNS BETWEEN PASS MESS
07071 000012 PASSC: 10 ;# OF RUNS BETWEEN PASS MESS
07072 000002 PASSN: 2 ;# OF PASS COUNTER
07073 007145 TROTA: TROTB ;POINTER TO TABLE FOR BREAKPOINT
07074 003077 TROTC: HALT ;TROUBLE HALT COMMAND
07075 006122 TROLC: CLORE ;TROUBLE LOOP REPORT COMMAND
07076 000401 TRORC: JMP .+1 ;TROUBLE RESET COMMAND
```

46

```
MLCRE: .TXT ! LOOP-ADDR! ;" LOOP-ADDR"
07077 046040
07100 047517
07101 026520
07102 042121
07103 051124
07104 000000
```

1 2287 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 ; MCV 0,1 ;INITIALIZE LOOP CODE
07 ; SETPI ;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 RTST 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 27105 165000 TROHA: MCV 3,1
27 27126 036272 CSAMS ;START ADDR MESSAGE
28 27127 036764 LDA 3,@TROTA
29 27110 054023 STA 3,IDX3 ;INIT TABLEPOINTER
30 27111 024763 LDA 1,TROHC ;HALT COMMAND
31 27112 032023 TROHL: LDA 2,@IDX3
32 27113 151215 MOV# 2,2,SNR ;END OF TABLE ?
33 27114 020427 JMP TRCEN ;YES, RETURN
34 27115 045000 STA 1,0,2 ;STORE HALT IN LOOP START
35 27116 020774 JMP TRCHL ;AGAIN
36
37 ;TROUBLE LOOP REPORT
38 27117 165200 TROLO: MCV 3,1
39 27127 036272 CSAMS ;START ADDR MESSAGE
40 27121 036752 LDA 3,@TROTA
41 27122 054023 STA 3,IDX3 ;INIT TABLEPOINTER
42 27123 024752 LDA 1,TROLC ;LOOP REPORT COMMAND
43 27124 032223 TROLL: LDA 2,@IDX3
44 27125 151215 MOV# 2,2,SNR ;END OF TABLE ?
45 27126 020415 JMP TRCEN ;YES, RETURN
46 27127 045000 STA 1,0,2 ;STORE CLORE IN LOOP START
47 27130 020774 JMP TRCLL ;AGAIN
48
49 ;TROUBLE RESET
50 27131 165000 TRORE: MCV 3,1
51 27132 036272 CSAMS ;START ADDR MESSAGE
52 27133 036740 LDA 3,@TROTA
53 27134 054023 STA 3,IDX3 ;INIT TABLEPOINTER
54 27135 024741 LDA 1,TRORC ;JMP .+1 COMMAND
55 27136 032223 TRORL: LDA 2,@IDX3
56 27137 151215 MOV# 2,2,SNR ;END OF TABLE ?
57 27140 020403 JMP TRCEN ;YES, RETURN
58 27141 045000 STA 1,0,2 ;STORE JMP .+1 IN LOOP START
59 27142 020774 JMP TRCRL ;AGAIN
60 27143 020401 TROEN: JMP .+1
61 27144 020740 SWISA ;RESTART MAIN PROGRAM
62
63
64 ;TAPE 4
65
66 .EOT
```


2058 .MAIN

01
02 ;TAPE6

03
04 07145 007145 TROTB: .

;ADDRESS LABEL FOR LOOPS WHIT
;AUTO BREAKPOINTS.

05			
06	07146	010370	A00
07	07147	010375	A02
08	07150	010402	A03
09	07151	010432	A17
10	07152	010445	A20
11	07153	010453	A21
12	07154	010461	A24
13	07155	010473	A26
14	07156	010506	A28
15	07157	010515	A32
16	07160	010531	A32
17	07161	010544	A34
18	07162	010557	A36
19	07163	010576	A38
20	07164	010611	A40
21	07165	010625	A42
22	07166	010637	A44
23	07167	010651	A46
24	07172	010666	A48
25	07171	010723	A50
26	07172	010714	A52
27	07173	010730	A54
28	07174	010751	A56
29	07175	010770	A58
30	07176	011003	A60
31	07177	011017	A62
32	07200	011034	A64
33	07201	011050	A66
34	07202	011062	A68
35	07203	011114	A70
36	07204	011162	A80
37	07205	011252	R00
38	07206	011257	R02
39	07207	011264	R04
40	07210	011272	R06
41	07211	011300	R08
42	07212	011312	R10
43	07213	011322	R11
44	07214	011334	R12
45	07215	011346	R14
46	07216	011357	R16
47	07217	011370	R18
48	07220	011405	R20
49	07221	011421	R21
50	07222	011464	R36
51	07223	011501	R38
52	07224	011516	R40
53	07225	011535	R42
54	07226	011551	R44
55	07227	011571	R46
56	07230	011612	R48
57	07231	011632	R50
58	07232	011651	R52
59	07233	011667	R54
60	07234	011726	R56
61	07235	011715	R58
62	07236	011734	R60
63	07237	011752	R62
64	07240	011770	R64
65	07241	012006	R68
66	07242	012075	R70
67	07243	012111	R72
68	07244	000000	2

;END OF TABLE.

1 2369 MAIN

01

02

PROG: .TXT !<12>REDR/PUNCH TEST! ;"REDR/PUNCH TEST"

07245 051312
07246 042105
07247 027522
07250 052520
07251 041516
07252 020110
07253 042524
07254 052123
07255 000000

03

04 07256 007261 RENDG: PUNTE+2

05

06 07257 165000 PUNTE: MOV 3,1 ;PUNCH DIAGNOSTIC
07 07260 006272 CSAMS
08 07261 006514 JSR @CINIT ;INITIATE PASSCOUNT ETC,
09 07262 004401 JSR .+1 ;NO QUESTIONS
10 07263 054773 STA 3,RENGG ;NO QUESTIONS, RESTART ADDRESS
11 07264 006476 JSR @CAB0 ;PERFORM TESTLOOPS AS ONE ROUTINE
12 07265 006123 CPASS ;CALL PASS ADMINI.
13 07266 000776 JMP .-2 ;LOOP

14

15 07267 165000 REDTE: MOV 3,1 ;READER DIAGNOSTIC

16

16 07270 006272 CSAMS
17 07271 006524 JSR @CINIT ;INITIATE
18 07272 006531 JSR @CREDC ;ANSWER GUES.
19 07273 004401 JSR .+1
20 07274 054762 STA 3,RENGG ;RESTART ADDRESS
21 07275 006460 JSR @CR00 ;PERFORM TESTLOOPS AS ONE ROUTINE
22 07276 006123 CPASS
23 07277 000776 JMP .-2

24

25 07300 165000 PSWIT: MOV 3,1 ;PUNCH FROM SWITCHES

26

26 07301 006272 CSAMS
27 07302 004474 JSR P0LES ;ANSWER GUES
28 07303 006471 JSR @CPANG
29 07304 004401 JSR .+1
30 07305 054751 STA 3,RENGG
31 07306 002450 JMP @CS4PU

32

33 07307 165000 PCHCN: MOV 3,1 ;PUNCH COUNTER

34

34 07310 006272 CSAMS
35 07311 006464 JSR @CINIT
36 07312 004464 JSR P0LES ;ANSWER GUES.
37 07313 004401 JSR .+1
38 07314 054742 STA 3,RENGG
39 07315 002450 JMP @CCTR

40

41 07316 165000 RCOUN: MOV 3,1 ;READ COUNTER

42

42 07317 006272 CSAMS
43 07320 006455 JSR @CINIT
44 07321 004401 JSR .+1
45 07322 054734 STA 3,RENGG
46 07323 002443 JMP @CCTR

47

48 07324 165000 PALT: MOV 3,1 ;ALTERNATE PUNCH

49

49 07325 006272 CSAMS
50 07326 006447 JSR @CINIT
51 07327 004447 JSR P0LES
52 07330 004401 JSR .+1
53 07331 054725 STA 3,RENGG
54 07332 002435 JMP @CALTP

1 2092 MAIN

```
01
02 27333 165200 RALT:  MOV 3,1 ;ALTERNATE READ
03 27334 006272      CSAMS
04 27335 006440      JSR @CINIT
05 27336 004401      JSR .+1
06 27337 054717      STA 3,RENCG
07 27340 002430      JMP @CALTR
08
09
10 27341 165000 PFLTZ: MOV 3,1 ;PUNCH FLOATING ZERO
11 27342 006272      CSAMS ;PRINT START ADDRES
12 27343 006432      JSR @CINIT
13 27344 004401      JSR .+1
14 27345 054711      STA 3,RENCG
15 27346 002423      JMP @CFLTR
16
17 27347 165000 RFLTZ: MOV 3,1 ;READ FLOATING ZERO
18 27350 006272      CSAMS
19 27351 006424      JSR @CINIT
20 27352 004401      JSR .+1
21 27353 054733      STA 3,RENCG
22 27354 002416      JMP @CFLTR
23
24 27355 165000 RESTR: MOV 3,1 ;RESTART, NO QUESTIONS
25 27356 006272      CSAMS
26 27357 006416      JSR @CINIT
27 27360 034676      LDA 3,RENCG
28 27361 001401      JMP 1,3
29
30
31 27362 010367 CA00:  A00-1
32 27363 011251 CR00:  R00-1
33 27364 010106 CSWPU:  S*PUN
34 27365 010127 CCTRP:  CTRPUN
35 27366 010144 CCTRR:  CTRRED
36 27367 007715 CALTP:  ALTPUN
37 27370 010167 CALTR:  ALTRED
38 27371 010275 CFLTP:  FLTPUN
39 27372 010312 CFLTR:  FLTRED
40 27373 007440 CRE0G:  REDCG
41 27374 007506 CPANG:  PANCG
42 27375 007751 CINIT:  INIT
```

```

1 0091 ,MAIN
01
02 07376 254425 PQUES: STA 3,RETGU ;DELAY WHEN PUNCHING ?
03 07377 236271 CGUES
04 07400 227424 MYES ;PUNCH DELAY
05 07401 227432 CMES ;DELAY
06 07402 227437 SANS ;1
07 07403 236254 CTZCC
08 07404 236260 CDZCC
09 07405 236123 CSTRI ;READ ANSWER
10 07406 233436 JMP DEL ;SUGGESTED ACCEPTED
11 07407 222773 JMP PQUES+1 ;ERROR RETURN
12 07410 222975 LDA 2,DIGIN ;ANSWER INPUT'ED
13 07411 126430 SUB 1,1
14 07412 122414 SUB# 1,0,SZR ;DIGIN=0
15 07413 232434 JMP .+4
16 07414 222875 DEL: LDA 2,DIGIN
17 07415 242124 STA 2,DELAY ;ANSWER EITHER 0 OR 1
18 07416 222425 JMP RETGU
19 07417 224420 LDA 1,SANS
20 07420 122415 SUB# 1,0,SAR
21 07421 222773 JMP DEL
22 07422 222755 JMP PQUES+1 ;ERROR
23
24 07423 222220 RETGU: 2
25
26 MYES: .TXT (PUNCH DELAY) ;"PUNCH DELAY "
07424 252523
07425 241516
07426 222112
07427 242524
07430 242514
07431 222131
27
28 CMES: .TXT (PCH DELAY) ;"PCH DELAY"
07432 241520
07433 222112
07434 242524
07435 242514
07436 222131
29
30 07437 222201 SANS: 1
31
32 07442 254763 REDGU: STA 3,RETGU ;HIGH SPEED READER TYPE ?
33 07441 226271 CGUES
34 07442 227466 PQUES ;READER 2002/500
35 07443 227476 DSUES ;R2022/500
36 07444 227533 SANS ;2020
37 07445 226254 CTZCC
38 07446 236260 CDZCC
39 07447 226125 CSTDC ;READ ANSWER
40 07450 222410 JMP RC2300 ;SUGGESTED ACCEPTED
41 07451 222773 JMP REDGU+1 ;ERROR RETURN
42 07452 222275 LDA 2,DIGIN ;ANSWER INPUT'ED
43 07453 224431 LDA 1,AANS ;ALT.ANSWER. 500
44 07454 126434 SUB 0,1,SZR
45 07455 222425 JMP .+5
46 07456 242125 STA 1,RC2
47 07457 222744 JMP RETGU
48 07460 126220 RC2300: ACC 1,1
49 07461 222775 JMP .-3
50 07462 224423 LDA 1,AANS
51 07463 126424 SUB 0,1,SZR
52 07464 222755 JMP REDGU+1 ;ERROR
53 07465 222773 JMP RC2300

```

1 0092 MAIN

01-

02 MQUES: .TXT IREADER 2000/5001

07466 042522
07467 042131
07472 051125
07471 031040
07472 030060
07473 027460
07474 030265
07475 000060

03

04 MQUES: .TXT IR2000/5001

07475 031122
07477 030060
07500 027460
07501 030265
07502 000060

05

06 07503 022200 ANS: 2000
07 07504 000764 AANS: 500.
08 07505 003720 AAANS: 2000.

09

10
11 07506 054715 PANGU: STA 3,RETGU ;TECHNICIAN PANEL ?
12 07507 006071 MQUES
13 07510 007523 ;TECHNICIAN PANEL ON
14 07511 007535 DGUE
15 07512 027541 PANS ;'YES'
16 07513 006054 CTZCC
17 07514 026060 CDZCC
18 07515 026133 CGTBI ;READ ANSW.
19 07516 000402 JMP .+2 ;SUGGESTED ACCEPTED
20 07517 020770 JMP PANGU+1 ;ERROR RETURN
21 07520 020275 LDA 0,DIGTN ;ANSWER INPUTED
22 07521 040130 STA 0,TEKPA
23 07522 002701 JMP 0,RETGU

24

25 MQUES: .TXT ITECHNICIAN PANEL ONI

07523 042524
07524 044123
07525 040516
07526 040523
07527 047131
07530 050240
07531 047131
07532 046135
07533 047440
07534 000116

26

27 MQUES: .TXT ITEC.PANI

07535 042524
07536 027133
07537 040520
07540 000116

28

29 07541 000001 PANS: 1

30

31

1 0093 ,MAIN

```
J1
32          007600          .LCC          7602
33 27600 204421  DEVC1: JSR          .+1          ;ROUTINE TO CHANGE THE DEVICE CODE,
34 27621 165200          MCV          3,1          ;THE NEW DEV CODE HAVE TO BE EVEN -
35 27622 006272          CSAMS          ;THE PTR CODE.
36 27623 206271          CGUES
37 27624 207677          MDEVG          ;"NEW CODE "
38 27625 007710          CDEVG
39 27626 007715          DEVC          ;12 (OCT)
40 27627 006254          CTZCC
41 27612 006260          CDZCC
42 27611 006124          CGTCK
43 27612 002422          JMP          .+2          ;SUGGESTED ACCEPTED.
44 27613 000765          JMP          DEVC          ;ERROR
45 27614 030275          LDA          2,CIGIN          ;GET ANSWER. - THE NEW CODE FOR PTR.
46 27615 151400          INC          2,2
47 27616 024457          LDA          1,I/O,PR          ;GET I/O-INSTR CODE - WITH OLD DEV. CODE
48 27617 125420          INC          1,1
49 27622 044451          STA          1,DEVCD          ;I/O INSTR
50 27621 050133          STA          2,CPTP          ; CODE STORE
51 27622 050450          STA          2,DEVCD          ;NEW CODE
52 27623 004416          JSR          DEVC          ;CHANGE THE DEV CODE IN I/O - PTR (ODD)
53 27624 024451          LDA          1,I/O,PR
54 27625 044444          STA          1,DEVCD          ;I/O - INSTR.
55 27626 030275          LDA          2,CIGIN          ;GET NEW CODE
56 27627 050132          STA          2,CPTR          ;SAVE DEV. CODE
57 27630 050442          STA          2,DEVCD          ;NEW CODE
58 27631 004410          JSR          DEVC          ;CHANGE THE DEV. CODE IN I/O -PTR (EVEN)
59 27632 024435          LDA          1,M100
60 27633 020442          LDA          0,I/O,PR
61 27634 123420          AND          1,2          ;GET ZEROS IN DEV.CODE FIELD -
62 27635 024435          LDA          1,DEVCD          ;GET NEW CODE
63 27636 123200          ADD          1,2          ;GENERATE NEW INSTR. -DEV CODE.
64 27637 040436          STA          2,I/O,PR
65 27640 022277          JMP          #IRESA
66
67 27641 054435  DEVC2: STA          3,DEVRET
68 27642 030431          LDA          2,FIRST
69 27643 021020          LDA          0,2,2
70 27644 024424          LDA          1,C1OT
71 27645 123420          AND          1,2          ;MASK OUT INSTR AND DEV. CODE.
72 27646 024423          LDA          1,DEVCD          ;GET THE SEEKED INSTR AND DEV. CODE.
73 27647 106415          SUB#          0,1,SNR
74 27650 000406          JMP          DEVC2          ;CODE FOUND.
75
76 27651 151400  DEVC1: INC          2,2
77 27652 020422          LDA          0,LAST
78 27653 112414          SUB#          0,2,SZR          ;LAST LOC ?
79 27654 002767          JMP          DEVC+2
80 27655 002421          JMP          #DEVRET
81
82 27656 021200  DEVC2: LDA          0,2,2
83 27657 024410          LDA          1,M100
84 27660 123420          AND          1,2          ;PUT ZEROS IN DEV. CODE FIELD.
85 27661 024411          LDA          1,DEVCD          ;GET NEW CODE
86 27662 123200          ADD          1,2          ;GENERATE NEW I/O - INSTR.
87 27663 041200          STA          0,2,2
88 27664 145200          MCV          2,1
89 27665 131200          MCV          1,2
90 27666 020763          JMP          DEVC1
```

1 2294 ,MAIN

01
 02 27667 177700 M102: -102
 03 27670 160077 CIOT: 160077
 04 27671 200200 DVCD: 0
 05 27672 200000 DVCD: 2
 06 27673 207716 FIRST: ALTPUN
 07 27674 212157 LAST: LASTP
 08 27675 260012 IO.PR: 62012
 09 27676 200000 DEVRET: 0

11 MDEVG: .TXT !<12><15>NEW DEVICE CODE! ;"NEW DEVICE CODE"

27677 226412
 27700 242516
 27701 220127
 27702 242504
 27703 244526
 27704 242523
 27705 241440
 27706 242117
 27707 200105

12 ODEVG: .TXT !NEW CODE! ;"NEW CODE"

27710 242516
 27711 220127
 27712 247503
 27713 242504
 27714 200000

13 27715 200012 DEVC: 12 ;PTR STANDARD CODE

14
 15 27716 200401 ALTPUN: JMP .+1 ;PUNCH ALTERNATE ONES/ZEROS.
 16 27717 122400 SUB 0,0
 17 27720 240505 STA 0,CTR ;INITIALIZE DATA
 18 27721 200413 JSR PUNDEL ;PUNCH DELAY
 19 27722 230503 LDA 2,CTR
 20 27723 150000 COM 2,2
 21 27724 250521 STA 2,CTR
 22 27725 220535 LDA 0,MASK ;
 23 27726 113400 AND 0,2 ;INHIBIT BITS.
 24 27727 206404 JSR IPUT ;PUNCH THE CHAR.
 25 27730 214533 DSZ LENGHT
 26 27731 200771 JYP ALTPUN+4
 27 27732 200767 JYP ALTPUN+3

28
 29 27733 210071 IPUT: PUT

30
 31 27734 254411 PUNDEL: STA 3,PUNRET;PUNCH DELAY LOGIC
 32 27735 220124 LDA 0,DELAY
 33 27736 101004 MCV 0,2,SZR
 34 27737 200404 JMP .+4
 35 27740 176520 SUBZL 3,3
 36 27741 254522 STA 3,LENGHT;SET LENGHT TO 1 CHAR.
 37 27742 202403 JMP #PUNRET
 38 27743 204463 JSR RDWT ;RANDOMNUMBER DELAY
 39 27744 202401 JYP #PUNRET

40
 41
 42 27745 200000 PUNRET: 0
 43 27746 123456 TRAN: 123456
 44 27747 200177 C177: 177
 45 27750 200000 CCTIM: 0
 46

1 2295 ,MAIN

```
01
02 27751 254431 INIT: STA 3,INTR ;INITIALIZETE ROUTINE
03 27752 226425 LDA 1,#INIPC
04 27753 246425 STA 1,#INIPB; INITIATE RUN-COUNTER
05 27754 122440 SUBC 2,2 ;ACC:=0, C:=0
06 27755 242424 STA 2,#INIPX; PASSCOUNTER:=0
07 27756 262677 ICRST ;RESET I/O
08 27757 262112 NIOS PTR.
09 27760 263412 SKPRN PTR.
12 27761 122421 SUB 2,2,SKP ;INITIATE FOR HIGH SPEED READER
11 27762 122200 ADC 2,2 ;AND PUNCH
12 27763 242127 STA 2,HSPREAD
13 27764 122420 SLB 2,2
14 27765 261113 DCAS 2,PTP.
15 27766 263413 SKPRN PTP.
16 27767 122401 SUB 2,2,SKP
17 27772 122200 ADC 2,2
18 27771 242126 STA 2,HSPUN
19 27772 222412 LDA 2,C1222
20 27773 242412 STA 2,#INIRZ ;INITIATE NULL CHAR. COUNTER
21 27774 122420 SUB 2,2
22 27775 242412 STA 2,#XFLAG ;RESET 1. RUN FLAG.
23 27776 222424 JMP #INIR
24
25 27777 227271 INIPC: PASSC
26 12227 227273 INIPB: PASSB
27 12221 227272 INIPX: PASSN
28 12222 222222 INIR: 2
29 12223 212273 INIRZ: RZCTR
30 12224 221222 C1222: 1222
31 12225 211162 XFLAG: FLAG
32
33
34 12226 263777 INTR: SKPDZ CPU ;INTERRUPT ROUTINE
35 12227 222410 JMP POWIN ;POWER INTER
36 12212 242414 STA 2,SAVE2
37 12211 222222 LDA 2,2 ;GET INTR PROG. ADDRE.
38 12212 242411 STA 2,HJVAR
39 12213 222276 LDA 2,POWZE
40 12214 242222 STA 2,2 ;STORE POWER REST.ADDRES
41 12215 222427 LDA 2,SAVE2 ;RESTORE ACC
42 12216 222425 JMP #HJVAR ;RETURN TO INTER. PROG
43
44 12217 222276 POWIN: LDA 1,POWZE ;POWER INTER,
45 12222 242222 STA 1,2
46 12221 263277 HALT
47 12222 222777 JMP .-1
48
49 12223 222222 HJVAR: 2
50 12224 222222 SAVE2: 2
51 12225 222222 CTR: 2
52
```


1 0096 MAIN

```

J1
J2 10226 054436 RDAT: STA 3,RER ;RANDOM NUMBER DELAY
J3 10227 020717 LDA 0,TRAN
J4 10230 004415 JSR RDCTR
J5 10231 040715 STA 0,TRAN
J6 10232 024715 LDA 1,C177
J7 10233 107405 AND 0,1,SNR ;MASK OUT LOW ORDER BITS
J8 10234 125400 INC 1,1
J9 10235 044426 STA 1,LENGHT;SET BLOCK LENGTH
J10 10236 024427 LDA 1,C774X
J11 10237 107725 ANDS 0,1,SNR ;MASK OUT HIGH ORDER BITS
J12 10240 125420 INC 1,1
J13 10241 044727 STA 1,CCTIM ;SET DELAY IN MSEC.
J14 10242 006261 C=AIT
J15 10243 007750 CCTIM ;WAIT (1-127 MSEC)
J16 10244 022420 JMP 0,RER
J17
J18 10245 024421 RDCTR: LDA 1,RDCT1
J19 10246 044421 STA 1,RDCT2
J20 10247 125120 MOVZL 0,1
J21 10250 125120 MOVZL 1,1
J22 10251 014416 CSZ RDCT2
J23 10252 000776 JMP .-2
J24 10253 127200 ADD 0,1
J25 10254 125120 MOVZL 1,1
J26 10255 125120 MOVZL 1,1
J27 10256 123200 ADD 1,2
J28 10257 024411 LDA 1,RDCT3
J29 10260 123200 ADD 1,2
J30 10261 021400 JMP 0,3 ;RETURN
J31
J32 10262 000377 MASK: 377 ;MASK TO INHIBIT BIT-POSITIONS
J33 10263 000000 LENGHT: 0
J34 10264 000000 RER: 0
J35 10265 077400 C774X: 77420
J36 10266 000210 RDCT1: 12
J37 10267 000000 RDCT2: 0
J38 10270 053031 RDCT3: 33031
J39
J40
J41 10271 054414 PUT: STA 3,PUTRE ;PUNCH THE CHARACTER IN AC2
J42 10272 024126 LDA 1,HSPUN
J43 10273 125025 MOV 1,1,SNR
J44 10274 000405 JMP PUT1
J45 10275 003513 SKPBZ PTP. ;HS-PUNCH.
J46 10276 000777 JMP .-1
J47 10277 071113 DCAS 2,PTP. ;PUNCH.
J48 10100 002405 JMP #PUTRE
J49
J50 10101 003511 PUT1: SKPBZ TTC
J51 10102 000777 JMP .-1
J52 10103 071111 DCAS 2,TTC
J53 10104 002401 JMP #PUTRE
J54
J55 10105 000000 PUTRE: 0
J56
```

1 2297 MAIN

```
01
02 10106 220401 SWPUN: JWP      .+1      ;PUNCH FROM SWITCHES.
03 10107 220425      JSR      PUNDEL   ;PUNCH DELAY
04 10110 220405      JSR      READ     ;READ SWITCHES
05 10111 220460      JSR      PUT      ;PUNCH THE CHAR. IN AC2
06 10112 214751      DSZ      LENGHT
07 10113 220775      JWP      SWPUN+2
08 10114 220773      JWP      SWPUN+1
09
10 10115 270477 READ:  READS    2          ;READ THE SWITCHES
11 10116 220130      LDA      0,TEKPA
12 10117 121204      MCV     2,0,SZR
13 10120 221400      JWP      0,3          ;TCP ON.
14 10121 151133      MCVZL= 2,2,SNC ;TEST SW0
15 10122 221400      JWP      0,3
16 10123 151120      MCVZL  2,2
17 10124 151120      MCVZL  2,2
18 10125 153120      ADDZL  2,2          ;AC2:=AC2 SHIFT 4
19 10126 221400      JWP      0,3
20
21
22
23 10127 220401 CTRPUN: JWP      .+1      ;PUNCH A COUNTER
24 10130 122220      ADDZ   0,2          ;AC1:=-1
25 10131 242674      STA     0,CTR      ;INITIATE COUNTER
26 10132 220402      JSR     PUNDEL   ;PUNCH DELAY
27 10133 212672      ISZ    CTR
28 10134 220401      JWP     .+1
29 10135 222670      LDA     0,CTR
30 10136 232724      LDA     2,MASK
31 10137 113400      AND     0,2
32 10140 2204731     JSR     PUT      ;PUNCH AC2
33 10141 214722     DSZ     LENGHT
34 10142 220771     JWP     CTRPUN+4
35 10143 220767     JWP     CTRPUN+3
36
37
38 10144 220401 CTRRED: JWP      .+1      ;READ A COUNTER.
39 10145 220421     JSR     XGET     ;READ A CHAR. TO ACC
40 10146 101215     MCV=   0,2,SNR ;SKIP LEADING TRAILER
41 10147 220776     JWP     .-2
42 10150 242473     STA     0,CTR1   ;SYNC
43 10151 224655 CTRR1: JSR     RDWT     ;DELAY
44 10152 224474     JSR     XGET     ;GET A CHAR TO ACC
45 10153 212470     ISZ    CTR1
46 10154 220421     JWP     .+1
47 10155 220466     LDA     1,CTR1
48 10156 232724     LDA     2,MASK
49 10157 147400     AND     2,1
50 10160 122414     SUB=   1,2,SZR ;CHECK THE COUNT READ
51 10161 220424     JWP     CTRR2   ;ERRPOR
52 10162 216462     DSZ    0XLENG
53 10163 220767     JWP     CTRR1+1
54 10164 220765     JWP     CTRR1
55 10165 220426 CTRR2: JSR     RDERR   ;READER ERROR
56 10166 220756     JWP     CTRRED  ;RESYNC THE READ.
```

```

1 0298 ,MAIN
01
02 10167 000401 ALTRED: JMP      .+1      ;ALTERNATE READ.
03 10170 000456          JSR      XGET     ;GET A CHAR. TO AC 0
04 10171 101015          MCV#   0,2,SNR
05 10172 000776          JMP      .-2      ;SKIP LEADING TRAILER
06 10173 126000          ADC     1,1      ;AC1:=-1
07 10174 004447          STA     1,CTR1   ;SYNC
08 10175 004631 ALTR1: JSR      RDWT     ;DELAY
09 10176 004450          JSR      XGET     ;GET A CHAR
10 10177 024444          LDA     1,CTR1
11 10200 124000          CCM     1,1      ;INVERT AC1
12 10201 044442          STA     1,CTR1
13 10202 032443          LDA     2,0XMASK
14 10203 147400          AND     2,1
15 10204 122414          SUB#   1,0,SZR   ;CHECK READ CHAR.
16 10205 000404          JMP     ALTR2    ;ERROR
17 10206 016436          DSZ    0,XLENG
18 10207 000767          JMP     ALTR1+1
19 10210 000765          JMP     ALTR1
20 10211 004402 ALTR2: JSR      RDERR
21 10212 000755          JMP     ALTRED   ;RESYNC THE READER.
22
23 10213 054427 RDERR: STA     3,RRET   ;ERROR PRINTER.
24 10214 044414          STA     1,GOOD
25 10215 042414          STA     0,BAD
26 10216 006043          CCRLF                      ;NEW LINE
27 10217 006040          CMESH
28 10220 010232          GOODEG
29 10221 024407          LDA     1,GOOD
30 10222 006052          CTCCT                      ;TYPE AC1
31 10223 026040          CMESH
32 10224 010236          BADEG
33 10225 024404          LDA     1,BAD
34 10226 026052          CTCCT                      ;TYPE AC1
35 10227 002413          JMP     0RRET
36
37 10230 000000 GOOD:  0
38 10231 000000 BAD:   0
39          GOODEG: .TXT    ;GOOD = 1
    10232 047507
    10233 042117
    10234 036440
    10235 000040
40          BADEG: .TXT    ;BAD = 1
    10236 041040
    10237 042101
    10240 036440
    10241 000040
41 10242 000000 RRET:  0
42 10243 000000 CTR1:  0
43 10244 010063 XLENG: LENGTH
44 10245 010062 XMASK: MASK

```

1 2299 MAIN

```
01
02 10246 054424 XGET: STA 3,XGETRE;READ A CHARACTER TO AC0
03 10247 024127 LDA 1,HSREA
04 10250 125215 MCV# 1,1,SNR
05 10251 000407 JMP XGET1 ;NO HS READER
06 10252 000061 C=AIT
07 10253 000131 C1 ;WAIT 1 MSEC.
08 10254 063512 SKPRZ PTR.
09 10255 000777 JMP .-1
10 10256 060512 DIAS 0,PTR. ;READ
11 10257 000404 JMP XGET2
12 10260 063510 XGET1: SKPRZ TTI
13 10261 000777 JMP .-1
14 10262 060510 DIAS 0,TTI
15 10263 024411 XGET2: LDA 1,X1000 ;TEST THAT ALL CHAR. ARE NOT ZERO.
16 10264 101014 MCV# 0,0,SRZ
17 10265 044406 STA 1,RZCTR
18 10266 010405 DSZ RZCTR
19 10267 002403 JMP @XGETRE
20 10270 026114 EHALLT ;TO MUCH LEADER ?
21 10271 002401 JMP @XGETRE
22
23 10272 000003 XGETRE: 0
24 10273 000003 RZCTR: 0
25 10274 001000 X1000: 1000
26
27 10275 000401 FLTPUN: JMP .+1 ;PUNCH FLOATING ZERO PATTERN.
28 10276 027467 LDA 0,0377
29 10277 040464 STA 0,FLTCT
30 10300 102400 SUB 2,2 ;AC2:=0
31 10301 026465 JSR @XPUT ;PUNCH LEADER
32 10302 010461 DSZ FLTCT
33 10303 000775 JMP .-3
34
35 10304 010457 PFLT1: ISZ FLTCT ;FLOATING COUNTER
36 10305 101000 MCV 0,0
37 10306 030455 LDA 2,FLTCT
38 10307 000425 JSR FLOT ;GET FLOATING NUMBER.
39 10310 026456 JSR @XPUT ;PUNCH
40 10311 000773 JMP PFLT1
41
42 10312 000401 FLTR0D: JMP .+1 ;READ FLOATING ZERO PATTERN.
43 10313 000733 JSR XGET ;READ A CHAR TO AC 0
44 10314 101015 MCV# 0,0,SNR ;READER MUST START IN LEADER!!
45 10315 000776 JMP .-2
46 10316 102520 SUBZL 2,2 ;AC2:=1
47 10317 050444 STA 2,FLTCT ;SYNC.
48 10320 010443 RFLT1: ISZ FLTCT
49 10321 101000 MCV 0,0
50 10322 000724 JSR XGET ;GET CHAR TO AC 0.
51 10323 105000 MCV 0,1
52 10324 030437 LDA 2,FLTCT
53 10325 024407 JSR FLOT ;GET FLOATING NUMBER.
54 10326 101000 MCV 1,0
55 10327 140000 MCV 2,1
56 10330 106415 SUB# 0,1,SNR
57 10331 000767 JMP RFLT1 ;OK.
58 10332 000661 JSR RDERR ;ERROR.
59 10333 000765 JMP RFLT1
00
01 10334 020430 FLOT: LDA 0,K17 ;TRANSLATE COUNT TO FLOATING.
02 10335 113400 AND 0,2 ;NUMBER.
03 10336 020404 LDA 0,CTABZ
04 10337 113000 ADD 0,2
05 10340 031000 LDA 2,0,2
06 10341 001400 JMP 0,3
```

1 2100 ,MAIN

01			
02	10342	010343	CTAB2: .+1
03	10343	000177	177
04	10344	000277	277
05	10345	000337	337
06	10346	000357	357
07	10347	000367	367
08	10350	000373	373
09	10351	000375	375
10	10352	000376	376
11	10353	000376	376
12	10354	000375	375
13	10355	000373	373
14	10356	000367	367
15	10357	000357	357
16	10360	000337	337
17	10361	000277	277
18	10362	000177	177
19	10363	000000	FLTCT: 0
20	10364	000017	K17 :17
21	10365	000377	D377: 377
22	10366	010071	XPUT: PUT

1 0101 .MAIN

```
31
32 10367 050150 STA 3,RTEST ;THE PUNCH TEST.
33 10370 000401 A02: JMP .+1
34 10371 006111 SETP1 ;THE PUNCH BUSY FLAG
35 10372 063513 SKPBZ PTP. ;SHOULD BE ZERO. CHECK
36 10373 026114 EHALT ;SELB LINE
37 10374 026113 LOOP
38
39 10375 000401 A02: JMP .+1
40 10376 006111 SETP1 ;THE PUNCH DONE FLAG
41 10377 063713 SKPBZ PTP. ;SHOULD BE ZERO. CHECK
42 10000 026114 EHALT ;SELB LINE
43 10401 006113 LOOP
44
45 10402 000401 A03: JMP .+1
46 10403 022134 LDA 0,C40
47 10404 040135 STA 0,XORDEV;
48 10405 006111 A04: SETP1 ;CHECK DEVICE SELECTION
49 10406 060113 NIOS PTP. ;SHOULD NOT AFFECT PTP.
50 10407 006063 TIMSK 120.
51 10410 000104 ;MSEK. TO WAIT MAX FOR
52 10411 063513 SKPBZ PTP ;SKP INSTR.
53 10412 006114 EHALT ;TIMEOUT RETURN
54 10413 060113 NIOS PTP. ;NORMAL RET.
55 10414 020430 LDA 0,CNIOC ;NIOC PTP.
56 10415 024135 LDA 1,XORDEV
57 10416 131020 MOV 1,2
58 10417 113520 ANDZL 0,2
59 10420 107020 ADD 0,1
60 10421 146400 SUB 2,1
61 10422 040401 STA 1,+.1
62 10423 000000 0
63 10424 063413 SKPRN PTP.
64 10425 006114 EHALT
65 10426 006113 LOOP
66 10427 020135 LDA 0,XORDEV
67 10431 101224 MOVZR 0,2,SZR
68 10431 000753 JMP A04-1
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
```

1 0102 MAIN

```

01
02 10445 000401 A20:  JMP      .+1
03 10446 026111      SETP1
04 10447 065477      INTA      1      ;WITH THE DONE FLAG
05 10450 125204      MOV      1,1,SRZ ;ZERO, CHECK THAT
06 10451 026114      EHALT   ;INTA DOES NOT READ
07 10452 026113      LOOP    ;BACK ANY BIT
08
09 10453 000401 A21:  JMP      .+1
10 10454 026111      SETP1
11 10455 060213      NIO     PTP.    ;A TEST TO INSURE THAT
12 10456 063513      SKPBZ  PTP.    ;PUN SELECT WITHOUT A
13 10457 006114      EHALT   ;START PULSE DOES NOT
14 10460 026113      LOOP    ;SET BUSY, PERHAPS THE
15                          ;START LINE IS OPEN.
16 10461 000401 A24:  JMP      .+1
17 10462 026111      SETP1
18 10463 060213      NIOS   PTP.    ;SEND A START PULSE
19 10464 063413      SKPEN  PTP.    ;AND SEE IF BUSY SETS.
20 10465 026114      EHALT   ;CHECK THE NOT
21 10466 026263      TIMSK   ;PUN START LEVEL TO
22 10467 013560      6000.   ;THE SET SIDE OF BUSY.
23 10470 063513      SKPBZ  PTP.    ;MAX 6 SEC
24 10471 006114      EHALT   ;TIMEOUT RETURN
25 10472 026113      LOOP    ;
26
27 10473 000401 A26:  JMP      .+1
28 10474 026111      SETP1
29 10475 060213      NIOS   PTP.    ;START THE PUNCH
30 10476 060213      NIO     PTP.    ;THEN DO A NO
31 10477 063413      SKPEN  PTP.    ;OPERATION. IF BUSY
32 10500 026114      EHALT   ;IS ZERO, PERHAPS THE
33 10501 026263      TIMSK   ;CLEAR LINE IS OPEN.
34 10502 013560      6000.   ;MAX WAIT 6 SEC FOR SKP
35 10503 063513      SKPBZ  PTP.
36 10504 026114      EHALT   ;TIMEOUT RETURN
37 10505 026113      LOOP
38
39 10506 000401 A28:  JMP      .+1
40 10507 026111      SETP1
41 10510 060213      NIOS   PTP.    ;START THE PUNCH
42 10511 060200      NIOC   2      ;THEN PRODUCE A CLEAR
43 10512 063413      SKPEN  PTP.    ;PULSE TO A DEVICE
44 10513 026114      EHALT   ;OTHER THAN PTP, CHECK
45 10514 026113      LOOP    ;AND GATE OF (CLEAR,
46                          ;PUN SELECT).
47 10515 000401 A30:  JMP      .+1
48 10516 026111      SETP1
49 10517 060213      NIOS   PTP.    ;SET BUSY FLAG
50 10520 060213      NIOC   PTP.    ;THE TRY TO CLEAR IT.
51 10521 063513      SKPBZ  PTP.    ;CHECK THE CLEAR SIDE OF
52 10522 026114      EHALT   ;THE BUSY FLOP.
53 10523 026113      NIOS   PTP.
54 10524 026263      TIMSK
55 10525 013560      6000.   ;MAX WAIT 6 SEC FOR SKP
56 10526 063513      SKPBZ  PTP.
57 10527 006114      EHALT   ;TIMEOUT RETURN
58 10530 026113      LOOP
59

```

```

| 2103 .MAIN
01
02 10531 000401 A32:   JMP      .+1
03 10532 006111       SETP1
04 10533 000113       NIOS    PTP.    ;SET THE BUSY FLOP
05 10534 002677       ICRST   ;THEN TRY TO CLEAR IT
06 10535 003513       SKPRZ   PTP.    ;VIA I/O RESET. CHECK
07 10536 006114       EHALT   ;THE I/O RESET PATH TO
08 10537 006063       TIMSK   ;CLEAR PUN BUSY.
09 10540 013560       6000.   ;MAX WAIT 6 SEC FOR SKP
10 10541 003513       SKPRZ   PTP.
11 10542 006114       EHALT   ;TIMEOUT RETURN
12 10543 006113       LOOP
13
14 10544 000401 A34:   JMP      .+1
15 10545 006111       SETP1
16 10546 000113       NIOS    PTP.    ;A CHECK TO INSURE
17 10547 006063       TIMSK   ;I/O RESET WILL CLEAR
18 10550 013560       6000.   ;THE DONE FLAG.
19 10551 003513       SKPRZ   PTP.    ;MAX WAIT 6 SEC FOR SKP
20 10552 006114       EHALT   ;TIMEOUT RETURN
21 10553 002677       ICRST   ;
22 10554 003713       SKPDZ   PTP.
23 10555 006114       EHALT
24 10556 006113       LOOP
25
26 10557 000401 A36:   JMP      .+1
27 10560 006111       SETP1
28 10561 000113       NIOS    PTP.    ;CHECK THAT A START
29 10562 006063       TIMSK   ;PULSE WILL CLEAR
30 10563 013560       6000.   ;THE DONE FLAG.
31 10564 003513       SKPRZ   PTP.    ;MAX WAIT 6 SEC FOR SKP
32 10565 006114       EHALT   ;TIMEOUT RETURN
33 10566 000113       NIOS    PTP.
34 10567 003713       SKPDZ   PTP.
35 10570 006114       EHALT
36 10571 006063       TIMSK
37 10572 013560       6000.   ;MAX WAIT 6 SEC FOR SKP
38 10573 003513       SKPRZ   PTP.
39 10574 006114       EHALT   ;TIMEOUT RETURN
40 10575 006113       LOOP
41
42 10576 000401 A38:   JMP      .+1
43 10577 006111       SETP1
44 10600 000113       NIOS    PTP.    ;A TEST TO INSURE
45 10601 006063       TIMSK   ;THAT THE DONE FLOP
46 10602 013560       6000.   ;IS RESET BY A CLEAR PULSE.
47 10603 003513       SKPRZ   PTP.    ;MAX WAIT 6 SEC FOR SKP
48 10604 006114       EHALT   ;TIMEOUT RETURN
49 10605 000213       NIOS    PTP.
50 10606 003713       SKPDZ   PTP.
51 10607 006114       EHALT
52 10610 006113       LOOP
53
54 10611 000401 A40:   JMP      .+1
55 10612 006111       SETP1
56 10613 000113       NIOS    PTP.    ;THE PUNCH COMPLETE
57 10614 006063       TIMSK   ;LEVEL FAILED TO
58 10615 013560       6000.
59 10616 003513       SKPRZ   PTP.
60 10617 006114       EHALT
61 10620 003613       SKPDN   PTP.    ;SET OR CLEAR BUSY,
62 10621 003513       SKPRZ   PTP.    ;CHECK 4.8 MS DELAY
63 10622 101011       MCV#    0,0,SKP
64 10623 006114       EHALT
65 10624 006113       LOOP
66

```



```

1 0104 .MAIN
01 10625 000401 A42: JWP .+1
02 10626 006111 SETP1 ;THE PUN COMPLETE
03 10627 060113 NIOS PTP. ;LEVEL FAILED TO CLEAR
04 10630 006063 TIMSK ;BUSY,CHECK THE CLOCK INPUT TO
05 10631 013560 6000. ;BUSY,THE FLOP MAY HAVE FAILED.
06 10632 063513 SKPBZ PTP.
07 10633 006114 EHALT ;TIMEOUT RETURN
08 10634 063513 SKPBZ PTP.
09 10635 006114 EHALT
10 10636 006113 LOOP
11
12 10637 000401 A44: JWP .+1
13 10640 006111 SETP1 ;TRY TO SET DONE VIA PUN COMPLETE
14 10641 060113 NIOS PTP. ;CHECK PUN DONE FLOP,CLOCK
15 10642 006063 TIMSK
16 10643 013560 6000.
17 10644 063513 SKPBZ PTP.
18 10645 006114 EHALT
19 10646 063613 SKPCN PTP. ;DATA AND CLEAR INPUTS.
20 10647 006114 EHALT
21 10650 006113 LOOP
22
23 10651 000401 A46: JWP .+1
24 10652 006111 SETP1 ;CHECK THE SET TO PUN INT REG FLOP
25 10653 102620 SUBZR 0,2 ;WHIT DONE (1) AND PUN INT DISABLE
26 10654 0060113 NIOS PTP. ;HOPEFULLY(2). SET INTERRUPT
27 10655 006063 TIMSK ;ENABLE IN CPU, IF ITS CLEARED
28 10656 013560 6000. ;INTERRUPT OCCURED.
29 10657 063513 SKPBZ PTP.
30 10660 006114 EHALT ;TIMEOUT RETURN
31 10661 060177 NIOS CPU
32 10662 101000 MOV 0,2
33 10663 063577 SKPBZ CPU
34 10664 006114 EHALT ;INTERRUPT DID NOT OCCURED
35 10665 006113 LOOP
36
37 10666 000401 A48: JWP .+1
38 10667 006111 SETP1 ;WILL PUN INT DISABLE PREVENT
39 10670 006113 NIOS PTP. ;PUN INT REG FROM SETTING ? .NO.
40 10671 020422 LDA 0,C4 ;
41 10672 062077 MSAC 0 ;CHECK AND OF (DONE(1),PUN INT DISABLE
42 10673 006063 TIMSK
43 10674 013560 6000.
44 10675 063513 SKPBZ PTP.
45 10676 006114 EHALT
46 10677 065477 INTA 1 ;(0)) TO DATA TERMINAL OF PUN INT REG.
47 10700 125014 MOVE 1,1,SZR ;ALSO PUN INT DISABLE FLOP,
48 10701 006114 EHALT ;AND ITS INPUTS
49 10702 006113 LOOP
50
51 10703 000401 A50: JWP .+1
52 10704 006111 SETP1 ;A UNKNOW INTERRUPT
53 10705 060177 NIOS CPU ;OCCURED. CHECK CC GATE
54 10706 101000 MOV 0,2 ;TO INTR.
55 10707 063477 SKPCN CPU
56 10710 006114 EHALT
57 10711 006113 LOOP
58 10712 020422 JWP A52
59
60
61 10713 000004 C4: 4 ;BIT 13

```

1 0125 MAIN

```

01
02 10714 000401 A52:   JMP      .+1
03 10715 006111       SETP1
04 10716 006113       NIOS    PTP.      ;A CHECK TO INSURE THAT
05 10717 006063       TIMSK                    ;THAT I/O RESET WILL
06 10720 013560       6000.
07 10721 063513       SKPRZ   PTP.
08 10722 006114       EHALT
09 10723 060177       NIOS    CPU      ;CLEAR PUN.
10 10724 062477       DIC     0,CPU    ;RESET PULSE
11 10725 063477       SKPRN   CPU
12 10726 006114       EHALT
13 10727 006113       LOOP
14
15 10730 000401 A54:   JMP      .+1
16 10731 006111       SETP1
17 10732 062277       NIOS    CPU      ;A MASK OUT INSTRUCTION
18 10733 102200       ADC     0,0      ;WITHOUT A BIT 13
19 10734 062077       MSKC   0
20 10735 102400       SUB     0,2      ;SHOULD NOT SET
21 10736 062077       MSKC   2        ;PUN INT DISABLE
22 10737 006113       NIOS    PTP.    ;FLOP, CHECK "D" INPUT TO FLOP.
23 10740 006063       TIMSK
24 10741 013560       6000.          ;MAX WAIT TIME 6 SEC FOR SKP
25 10742 063513       SKPRZ   PTP.
26 10743 006114       EHALT
27 10744 060177       NIOS    CPU
28 10745 101000       MOV     0,2
29 10746 063577       SKPRZ   CPU
30 10747 006114       EHALT
31 10750 006113       LOOP
32
33 10751 000401 A56:   JMP      .+1
34 10752 006111       SETP1
35 10753 022740       LDA     0,C4    ;CHECK TO INSURE
36 10754 062277       MSKC   0        ;THAT I/O RESET
37 10755 062077       ICRST                    ;WILL CLEAR THE PUN INT
38 10756 006113       NIOS    PTP.    ;DISABLE FLOP.
39 10757 006063       TIMSK
40 10760 013560       6000.          ;MAX WAIT TIME 6 SEC FOR SKP
41 10761 063513       SKPRZ   PTP.
42 10762 006114       EHALT
43 10763 060177       NIOS    CPU
44 10764 101000       MOV     0,2
45 10765 063577       SKPRZ   CPU
46 10766 006114       EHALT
47 10767 006113       LOOP
48
49
50 10770 000401 A58:   JMP      .+1
51 10771 006111       SETP1
52 10772 060113       NIOS    PTP.    ;WHIT PUN INT REQ(1) SEE IF
53 10773 006063       TIMSK                    ;INTA READS BACK ANY BITS.
54 10774 013560       6000.          ;MAX WAIT 6 SEC. FOR SKP
55 10775 063513       SKPRZ   PTP.
56 10776 006114       EHALT
57 10777 065477       INTA   1        ;TIMEOUT RETURN
58 11000 125015       MOV#   1,1,SAR
59 11201 006114       EHALT
60 11202 006113       LOOP

```

1 2106 ,MAIN

01

```
02 11203 000401 A62: JMP .+1
03 11204 006111 SETP1
04 11205 002113 NIOS PTP. ;CHECK THE DATA 15 LINE ON INTA.
05 11206 006063 TIMSK
06 11207 013560 6000. ;MAX WATI 6 SEC. FOR SKP
07 11210 003513 SKPRZ PTP.
08 11211 006114 EHALT ;TIMEOUT RETURN
09 11212 102520 SUBZL 0,0
10 11213 005477 INTA 1
11 11214 107415 AND# 0,1,SNR
12 11215 006114 EHALT
13 11216 006113 LOOP
```

14

```
15 11217 000401 A62: JMP .+1
16 11220 006111 SETP1 ;CHECK THE DATA 14 LINE ON INTA.
17 11221 002113 NIOS PTP.
18 11222 006063 TIMSK
19 11223 013560 6000. ;MAX WATI 6 SEC. FOR SKP
20 11224 003513 SKPRZ PTP.
21 11225 006114 EHALT ;TIMEOUT RETURN
22 11226 102520 SUBZL 0,0
23 11227 101120 MOVZL 0,2
24 11230 005477 INTA 1
25 11231 107415 AND# 0,1,SNR
26 11232 006114 EHALT
27 11233 006113 LOOP
```

28

```
29 11234 000401 A64: JMP .+1
30 11235 006111 SETP1 ;CHECK FOR INTA
31 11236 002113 NIOS PTP. ;CODE 13 (FOR PTP),
32 11237 006063 TIMSK
33 11240 013560 6000. ;MAX TIME 6 SEC. TO WAIT FOR SKP
34 11241 003513 SKPRZ PTP.
35 11242 006114 EHALT ;TIMEOUT RETURN
36 11243 020133 LDA 0,CPTP.
37 11244 005477 INTA 1
38 11245 105414 SUB# 0,1,SZR
39 11246 006114 EHALT
40 11247 006113 LOOP
```

41

```
42 11250 000401 A66: JMP .+1
43 11251 006111 SETP1 ;THE PUN DONE FLAG WAS SET
44 11252 002113 NIOS PTP. ;VIA PUN COMPLEAT EVEN
45 11253 002213 NIOC PTP. ;WHIT BUSY ZERO.
46 11254 006061 CWAIT ;WAIT 100 MSEC.
47 11255 011152 D100
48 11256 003713 SKPOZ PTP.
49 11257 006114 EHALT
50 11260 006113 LOOP
51 11261 000401 JMP A68
```

52

53

54

1 0107 .MAIN

```
01
02 11062 000401 A68:   JMP      .+1
03 11063 006111       SETP1      ;CHECK THAT 50 CHAR. ARE PUNCHED IN
04 11064 126402       SUB        1,1
05 11065 044471       STA        1,PSUM
06 11066 024466       LDA        1,DSZ
07 11067 044466       STA        1,CCTR
08 11070 060113       NIOS      PTP.      ;1 SEC. OF TIME
09 11071 026063       TIMSK
10 11072 013560       6000.      ;MAX TIME 6 SEC. TO WAIT FOR SKP
11 11073 063513       SKPBZ     PTP.
12 11074 006114       EHALT     ;TIMEOUT RETURN
13 11075 060113 A69:   NIOS      PTP.
14 11076 006064       TIMMS     ;MEASURE SKP, INSTR.
15 11077 063513       SKPBZ     PTP.
16 11100 006114       EHALT     ;TIMEOUT RET.
17 11101 125220       MOVZR     1,1      ;NORM RET. C(1) : TIME IN 10 USEC.
18 11102 125220       MOVZR     1,1      ;AC1:=AC1/4
19 11103 030453       LDA        2,PSUM
20 11104 133000       ADD        1,2
21 11105 050451       STA        2,PSUM
22 11106 010447       DSZ       CCTR
23 11107 000766       JYP       A69
24 11110 020443       LDA        0,CMAX
25 11111 112432       SUBZ#     0,2,SZC
26 11112 006114       EHALT
27 11113 006113       LOOP
28
29 11114 000401 A70:   JMP      .+1
30 11115 006110       SETP2
31 11116 026061       CWAIT     ;WAIT 5 SEC.
32 11117 011151       CSSEC     ;FOR THE PUNCH TO STOP.
33 11120 126400       SUB        1,1
34 11121 044435       STA        1,PSUM
35 11122 024432       LDA        1,DSZ
36 11123 044432       STA        1,CCTR
37 11124 060113       NIOS      PTP.
38 11125 036063       TIMSK
39 11126 013560       6000.      ;MAX TIME 6 SEC. TO WAIT FOR SKP
40 11127 063513       SKPBZ     PTP.
41 11130 006114       EHALT     ;TIMEOUT RETURN
42 11131 060113 A71:   NIOS      PTP.
43 11132 006064       TIMMS     ;MEASURE SKP, INSTR.
44 11133 063513       SKPBZ     PTP.      ;CHECK FOR 50 CHARACTORES
45 11134 006114       EHALT     ;IN 1 SEC OF TIME.
46 11135 125220       MOVZR     1,1      ;NORM RET. C(1): TIME IN 10'S USEC.
47 11136 125220       MOVZR     1,1      ;AC1:=AC1/4
48 11137 030417       LDA        2,PSUM
49 11140 133000       ADD        1,2
50 11141 050415       STA        2,PSUM
51 11142 014413       CSZ       CCTR
52 11143 000766       JYP       A71
53 11144 020407       LDA        0,CMAX
54 11145 112432       SUBZ#     0,2,SZC
55 11146 006114       EHALT     ;PUNCH TO STOP.
56 11147 006113       LOOP
57 11150 000412       JMP      A82
58
59 11151 011610 CSSEC:  5000.
60 11152 000144 DSZ:    100.
61 11153 060650 CMAX:  25000.      ;EQV. TO 1 SEC
62 11154 000062 DSZ:    50.
63 11155 000000 CCTR:    0
64 11156 020200 PSUM:    0
65 11157 000000 STAT:    0
66 11160 000000 FLAG:    0
67 11161 000000 HRET:    0
```

1 0108 .MAIN

```

01
02 11162 000401 A82:  JMP      +1      ;STATUS BIT CHECK.
03 11163 020775      LDA      0,FLAG  ;IN EVERY LOOP CHANGE IN THE STATUS
04 11164 101034      MOV      0,2,SZR ;PAPER TAPE LOW/OK IS REPORTED.
05 11165 000410      JMP      A82
06 11166 102000      ADC      0,2      ;FIRST RUN
07 11167 042771      STA      0,FLAG
08 11170 000413      CIA      0,PTP.
09 11171 043766      STA      0,STAT
10 11172 101232      MOVZR#  0,2,SZC ;TEST BIT 15
11 11173 004413      JSR      M,LOW   ;TAPE LOW
12 11174 004422      JSR      M,CK    ;TAPE OK
13
14 11175 000413 A82:  CIA      0,PTP.  ;GET STATUS WORD.
15 11176 024761      LDA      1,STAT  ;GET OLD STATUS WORD
16 11177 106415      SUB#    0,1,SNR
17 11200 000425      JMP      A84     ;NO CHANGE.
18 11201 043756      STA      0,STAT
19 11202 101232      MOVZR#  0,2,SZC ;TEST STATUS BIT
20 11203 004403      JSR      M,LOW   ;TAPE LOW
21 11204 004412      JSR      M,CK    ;TAPE OK.
22
23 11205 002150 A84:  JMP      #RTEST  ;PERFORM THE TEST AGAIN.
24
25 11206 054753 M,LOW: STA      3,MRET  ;PRINT "TAPE LOW"
26 11207 006040      CMPS    TLOW
27 11210 011225      TLOW
28 11211 006046      CCICL          ;CLEAR DISPLAY
29 11212 006044      CDISP
30 11213 011233      DLOW
31 11214 034745      LDA      3,MRET
32 11215 001401      JMP      1,3     ;RETURN +1
33
34 11216 054743 M,OK:  STA      3,MRET  ;PRINT "TAPE OK"
35 11217 006040      CMPS    TOK
36 11220 011240      TOK
37 11221 006046      CCICL          ;CLEAR DISPLAY
38 11222 006044      CDISP
39 11223 011245      DOK
40 11224 002735      JMP      #MRET   ;RETURN
41
42      TLOW:  .TXT      1<12><15>TAPE LOW!      ;"TAPE LOW"
11225 006412
11226 043524
11227 042520
11230 046040
11231 053517
11232 000000
43      DLOW:  .TXT      1TAPE LOW!
11233 040524
11234 042520
11235 046040
11236 053517
11237 000000
44      TOK:  .TXT      1<12><15>TAPE OK!      ;"TAPE OK"
11240 006412
11241 043524
11242 042520
11243 047440
11244 000113
45      DOK:  .TXT      1TAPE OK!
11245 040524
11246 042520
11247 047440
11250 000113

```

1 2129 MAIN

```
01
02 11251 054153 STA 3,RTEST
03 11252 020401 R00: JMP .+1
04 11253 006111 SETP1 ;PAPER TAPE READER
05 11254 063512 SKPRZ PTR. ;BUSY FLAG SHOULD BE 0
06 11255 006114 EHALT ;CHECK SELB LINE
07 11256 006113 LOOP
08
09 11257 020401 R02: JMP .+1
10 11260 006111 SETP1 ;PTR DONE FLAG
11 11261 063712 SKPRZ PTR. ;SHOULD BE ZERO.
12 11262 006114 EHALT ;CHECK SELD LINE.
13 11263 006113 LOOP
14
15 11264 000401 R04: JMP .+1
16 11265 006111 SETP1 ;A START PULSE TO A
17 11266 060100 NIOS 2 ;DEVICE NOT THE PTR
18 11267 063512 SKPRZ PTR. ;SET PTR BUSY! CHECK
19 11270 006114 EHALT ;AND GATE.
20 11271 006113 LOOP
21
22 11272 000401 R06: JMP .+1
23 11273 006111 SETP1 ;SELECTING THE PTR
24 11274 063212 NIO PTR. ;WITHOUT A START
25 11275 063512 SKPRZ PTR. ;PULSE SET THE BUSY
26 11276 006114 EHALT ;FLOP. PERHAPS THE START
27 11277 006113 LOOP ;LINE IS OPEN.
28
29 11300 000401 R08: JMP .+1
30 11301 006111 SETP1 ;TURN ON THE
31 11302 020561 LDA 0,C28 ;INTERRUPT SYSTEM.
32 11303 062277 MSA0 0 ;CHECK FOR NO INTERRUPTS.
33 11304 122620 SUBZR 0,2 ;BUT!
34 11305 067177 NIOS CPU ;INTERRUPT OCCURED.
35 11306 121230 MOV 0,2 ;CHECK INTR. LINE.
36 11307 063477 SKPRN CPU ;AND CHECK RD INT RG FLOP
37 11310 006114 EHALT
38 11311 006113 LOOP
39
40 11312 000401 R10: JMP .+1
41 11313 006111 SETP1 ;TURN ON THE INTERRUPT.
42 11314 102620 SUBZR 0,0 ;CHECK FOR NO INTERRUPT.
43 11315 067177 NIOS CPU ;INTERRUPT OCCURED.
44 11316 121230 MOV 0,2 ;CHECK THE DONE INPUT
45 11317 063477 SKPRN CPU ;TO AND OF (RD INT DISABLE,
46 11320 006114 EHALT ;RD DONE).
47 11321 006113 LOOP
48
49 11322 000401 R11: JMP .+1
50 11323 006111 SETP1 ;BUSY FAILED TO CLEAR.
51 11324 006261 CAAIT
52 11325 000131 C1 ;WAIT 1 MS (SLOWDOWN READER)
53 11326 062112 NIOS PTR. ;PAPER IN READER ?
54 11327 006263 TIMSK
55 11330 000144 I02. ;MAX WAIT 100MSEC FOR SKP
56 11331 063512 SKPRZ PTR.
57 11332 006114 EHALT ;TIMEOUT RETURN
58 11333 006113 LOOP ;IS LEVER DOWN? ...
59
```

1 0112 MAIN

```
01
02 11334 000401 R12:  JMP      .+1
03 11335 006111      SETP1      ;TRY TO SET THE BUSY FLOP
04 11336 060112      NIOS PTR.    ;VIA START.
05 11337 063412      SKPBN PTR.    ;CHECK AND GATE OF (START,
06 11340 006114      EHALT      ;RD SELECT) AND RD BUSY.
07 11341 006063      TIMSK
08 11342 000144      I00.
09 11343 063512      SKPBZ PTR.
10 11344 006114      EHALT      ;TIMEOUT RETURN
11 11345 006113      LOOP
12
13 11346 000401 R14:  JMP      .+1
14 11347 006111      SETP1      ;SET READER BUSY FLAG
15 11350 006061      C*AIT
16 11351 000131      C1        ;THEN TRY TO CLEAR VIA
17 11352 060112      NIOS PTR.
18 11353 062677      ICRST      ;I/O RESET. CHECK BUSY FLOP,
19 11354 063512      SKPRZ PTR.    ;THE CLEAR INPUT
20 11355 006114      EHALT
21 11356 006113      LOOP
22
23 11357 000401 R16:  JMP      .+1
24 11360 006111      SETP1      ;SET READER BUSY FLAG
25 11361 006061      C*AIT
26 11362 000131      C1        ;THEN TRY TO RESET VIA
27 11363 060112      NIOS PTR.    ;RD SELECT AND A CLEAR PULSE
28 11364 060212      NIOC PTR.
29 11365 063512      SKPBZ PTR.
30 11366 006114      EHALT
31 11367 006113      LOOP
32
33 11370 000401 R18:  JMP      .+1
34 11371 006111      SETP1      ;SET READER BUSY FLOP.
35 11372 006061      C*AIT
36 11373 000131      C1        ;THEN CHECK THAT A
37 11374 060112      NIOS PTR.    ;CLEAR PULSE TO A DEVICE
38 11375 060200      NIOC 0     ;OTHER THAN THE READER
39 11376 063412      SKPBN PTR.    ;WILL NOT CLEAR READER
40 11377 006114      EHALT      ;BUSY.
41 11400 006263      TIMSK
42 11401 000144      I00.
43 11402 063512      SKPBZ PTR.
44 11403 006114      EHALT
45 11404 006113      LOOP
46
47 11405 000401 R20:  JMP      .+1
48 11406 006111      SETP1      ;SET READER BUSY FLOP.
49 11407 006061      C*AIT
50 11410 000131      C1        ;THEN CHECK THAT
51 11411 060112      NIOS PTR.    ;RD SELECT WITHOUT A
52 11412 060212      NIO PTR.    ;
53 11413 063412      SKPBN PTR.    ;CLEAR PULSE WILL NOT
54 11414 006114      EHALT      ;CLEAR READER BUSY.
55 11415 006063      TIMSK
56 11416 000144      I00.
57 11417 063512      SKPBZ PTR.
58 11420 006113      LOOP
59
```

```

1 2111 .MAIN
01
02 11421 007401 R21: JMP .+1
03 11422 020134 LDA 0,C40
04 11423 040135 STA 2,XORDEV
05 11424 006111 R22: SETPR1 ;CHECK DEVICE SELECTION.
06 11425 006061 CAAIT
07 11426 000131 C1 ;WAIT 1MSEC.
08 11427 000112 NIOS PTR.
09 11430 006063 TIMSK
10 11431 000144 100.
11 11432 063512 SKPR2 PTR.
12 11433 006114 EH-ALT ;TIMEOUT RETURN
13 11434 006061 CAAIT
14 11435 000131 C1
15 11436 000112 NIOS PTR.
16 11437 020423 LDA 2,CCNIO
17 11440 020135 LDA 1,XORDEV
18 11441 131000 MOV 1,2
19 11442 113520 ANDZL 0,2
20 11443 107000 ADD 0,1
21 11444 146400 SUB 2,1
22 11445 044401 STA 1,+.1
23 11446 000000 2
24 11447 063412 SKPRN PTR.
25 11450 006114 EH-ALT
26 11451 006063 TIMSK
27 11452 020144 100.
28 11453 063512 SKPR2 PTR.
29 11454 006114 EH-ALT ;TIMEOUT RETURN
30 11455 006113 LOOP
31 11456 020135 LDA 0,XORDEV
32 11457 101224 MOVZR 2,2,SZR
33 11460 000743 JMP R22-1
34 11461 000403 JMP R36
35
36 11462 000212 CCNIO: NIOC PTR.
37 11463 000020 C20: 20 ;BIT 11
38
39 11464 000401 R36: JMP .+1
40 11465 006111 SETPR1 ;I/O RESET FAILED
41 11466 006061 CAAIT
42 11467 000131 C1
43 11470 000112 NIOS PTR. ;TO CLEAR THE
44 11471 006063 TIMSK
45 11472 000144 100.
46 11473 063512 SKPR2 PTR.
47 11474 006114 EH-ALT
48 11475 062677 ICRST ;RD DONE FLAG
49 11476 063712 SKPR2 PTR.
50 11477 006114 EH-ALT
51 11500 006113 LOOP
52
53 11501 000401 R38: JMP .+1
54 11502 006111 SETPR1 ;A CLEAR PULSE
55 11503 006061 CAAIT
56 11504 000131 C1 ;TO THE PAPER TAPE READER
57 11505 000112 NIOS PTR. ;FAILED TO CLEAR RD DONE.
58 11506 006063 TIMSK
59 11507 020144 100.
00 11510 063512 SKPR2 PTR.
01 11511 006114 EH-ALT ;TIMEOUT RETURN
02 11512 000212 NIOC PTR.
03 11513 063712 SKPR2 PTR.
04 11514 006114 EH-ALT
05 11515 006113 LOOP
06

```


1 0112 .MAIN

```
01
02 11516 000401 R40:   JMP      .+1
03 11517 006111       SETP1
04 11520 006061       C*AIT      ;A START PULSE TO THE
05 11521 000131       C1        ;READER FAILED TO
06 11522 000112       NIOS      PTR.    ;CLEAR RD DCNE
07 11523 006063       TIMSK
08 11524 000144       102.
09 11525 003512       SKPRZ    PTR.
10 11526 006114       EHALT    ;TIMEOUT
11 11527 006061       C*AIT
12 11530 000131       C1
13 11531 000112       NIOS      PTR.
14 11532 003712       SKPDZ    PTR.
15 11533 006114       EHALT
16 11534 006113       LOOP
17
18 11535 000401 R42:   JMP      .+1
19 11536 006111       SETP1
20 11537 006061       C*AIT    ;THE READER DCNE FLOP
21 11540 000131       C1
22 11541 000112       NIOS      PTR.    ;FAILED TO SET.
23 11542 006063       TIMSK    ;CHECK THE CLCCK, DATA
24 11543 000144       102.    ;AND RESET INPUTS.
25 11544 003512       SKPRZ    PTR.    ;CHECK ALSO SELD LINE.
26 11545 006114       EHALT    ;TIMEOUT RETURN
27 11546 003612       SKPDN    PTR.
28 11547 006114       EHALT
29 11550 006113       LOOP
30
31 11551 000401 R44:   JMP      .+1
32 11552 006111       SETP1
33 11553 102420       SLBZ    0,0    ;SETTING THE READER
34 11554 062077       MSKO    0      ;DONE FLAG SHOULD
35 11555 062077       ICRST   ;CAUSE A INTERRUPT.
36 11556 000177       NIOS      CPU    ;CHECK THE INTER. LINE,
37 11557 006061       C*AIT    ;RD INT REQ FLOP,
38 11560 000131       C1
39 11561 000112       NIOS      PTR.    ;AND ITS INPUT GATES.
40 11562 006063       TIMSK
41 11563 000144       102.
42 11564 003512       SKPRZ    PTR.
43 11565 006114       EHALT    ;TIMEOUT RETURN
44 11566 003577       SKPRZ    CPU
45 11567 006114       EHALT
46 11570 006113       LOOP
47
48 11571 000401 R46:   JMP      .+1
49 11572 006111       SETP1
50 11573 102000       ACC     0,0    ;THE MSKO INSTRUCTION
51 11574 062077       MSKO    0
52 11575 102420       SLBZ    0,0    ;WHIT BIT 11 A ZERO
53 11576 062077       MSKO    0      ;SHOULD NOT PREVENT
54 11577 000177       NIOS      CPU    ;INTERRUPTS.
55 11600 006061       C*AIT
56 11601 000131       C1
57 11602 000112       NIOS      PTR.
58 11603 006063       TIMSK
59 11604 000144       102.
60 11605 003512       SKPRZ    PTR.
61 11606 006114       EHALT
62 11607 003577       SKPRZ    CPU    ;TIMEOUT
63 11610 006114       EHALT
64 11611 006113       LOOP
65
```

1 0113 MAIN

```
01
02 11612 000401 R48:   JMP      .+1
03 11613 006111       SETP1
04 11614 020647       LDA      0,C20      ;THE RD INT DISABLE
05 11615 002077       MSKO      2          ;FLOP IS SET VIA MSKO.
06 11616 006061       CWAIT
07 11617 000131       C1
08 11620 002677       ICRST
09 11621 000177       NIOS     CPU        ;THE DONE FLAG FAILED.
10 11622 006112       NIOS     PTR.       ;TO INTERRUPT,SUGGESTING
11 11623 006063       TIMSK
12 11624 000144       I20.
13 11625 003512       SKPRZ   PTR.       ;I/O RESET DID NOT CLEAR
14 11626 006114       EHALT
15 11627 003577       SKPRZ   CPU        ;RD INT DISABLE.
16 11630 006114       EHALT
17 11631 006113       LOOP
18
19 11632 000401 R50:   JMP      .+1
20 11633 006111       SETP1
21 11634 006061       CWAIT      ;USING MSKO WITH
22 11635 000131       C1          ;BIT 11(1) SHOULD SET
23 11636 020625       LDA      0,C20
24 11637 002077       MSKO      2          ;RD INT DISABLE AND
25 11640 000177       NIOS     CPU        ;PREVENT THE DONE
26 11641 006112       NIOS     PTR.       ;FLOP FROM INTERRUPTING.
27 11642 006063       TIMSK
28 11643 000144       I20.
29 11644 003512       SKPRZ   PTR.       ;A INTERRUPT OCCURED
30 11645 006114       EHALT
31 11646 003477       SKPRZ   CPU        ;HOWEVER.
32 11647 006114       EHALT
33 11650 006113       LOOP
34
35 11651 000401 R52:   JMP      .+1
36 11652 006111       SETP1
37 11653 020610       LDA      0,C20      ;ANOTHER CHECK OF
38 11654 006061       CWAIT      ;RD INT DISABLE,
39 11655 000131       C1          ;THE CLOCK INPUT.
40 11656 000177       NIOS     CPU
41 11657 006112       NIOS     PTR.
42 11660 006063       TIMSK
43 11661 000144       I20.
44 11662 003512       SKPRZ   PTR.
45 11663 006114       EHALT
46 11664 003577       SKPRZ   CPU        ;TIMEOUT
47 11665 006114       EHALT
48 11666 006113       LOOP
49
50 11667 000401 R54:   JMP      .+1
51 11670 006111       SETP1
52 11671 006061       CWAIT      ;TEST I/O RESET
53 11672 000131       C1
54 11673 000177       NIOS     CPU        ;ABILITY TO CLEAR
55 11674 006112       NIOS     PTR.       ;RD INT REQ FLOP.
56 11675 006063       TIMSK
57 11676 000144       I20.
58 11677 003512       SKPRZ   PTR.
59 11700 006114       EHALT
60 11701 000177       NIOS     CPU        ;TIMEOUT
61 11702 002477       DIC      0,CPU
62 11703 003477       SKPRZ   CPU
63 11704 006114       EHALT
64 11705 006113       LOOP
65
```

1 0114 ,MAIN

```
01
02 11706 000431 R56:   JMP      .+1
03 11707 006111       SETPR1   ;AFTER A I/O RESET
04 11710 062677       ICRST    ;A INTA SHOULD READ
05 11711 061477       INTA     0      ;BACK ALL ZERCS.
06 11712 101004       MOV      0,2,SZR
07 11713 006114       EHALLT
08 11714 006113       LOOP
09
10 11715 000401 R58:   JMP      .+1
11 11716 006111       SETPR1   ;INTA FAILED TO
12 11717 006061       CWAIT
13 11720 000131       C1
14 11721 060112       NIOS    PTR.    ;READ BACK A BIT 14
15 11722 006063       TIMSK   ;WHEN THE DONE FLAG
16 11723 000144       ICR.
17 11724 063512       SKPRZ   PTR.    ;WAS SET.
18 11725 006114       EHALLT   ;TIMEOUT RETURN
19 11726 061477       INTA     0
20 11727 126520       SUBZL   1,1
21 11730 125120       MOVZL   1,1
22 11731 107415       AND#    0,1,SNR
23 11732 006114       EHALLT
24 11733 006113       LOOP
25
26 11734 000401 R60:   JMP      .+1
27 11735 006111       SETPR1   ;CHECK THE CODE
28 11736 006061       CWAIT
29 11737 000131       C1
30 11740 006112       NIOS    PTR.    ;READ BACK ON INTA
31 11741 006063       TIMSK   ;FROM THE READER
32 11742 000144       ICR.
33 11743 063512       SKPRZ   PTR.
34 11744 006114       EHALLT   ;TIMEOUT
35 11745 024132       LDA     1,CPTR.
36 11746 061477       INTA     0
37 11747 106414       SUB#    0,1,SZR
38 11750 006114       EHALLT
39 11751 006113       LOOP
40
41 11752 000431 R62:   JMP      .+1
42 11753 006111       SETPR1   ;CHECK THAT SOME,
43 11754 126620       SUBZR   1,1    ;ANY, BITS ARE READ
44 11755 125225       MOVZR   1,1,SNR ;BACK IN A SET OF
45 11756 006114       EHALLT   ;15 CHARACTORS.
46 11757 006061       CWAIT    ;A ERROR HERE CAN BE CAUSED
47 11760 000131       C1       ;BY A STRIP OF NULL CHAR. IN
48 11761 006112       NIOS    PTR.    ;THE PAPER TAPE !
49 11762 063512       SKPRZ   PTR.
50 11763 000777       JMP      .-1
51 11764 060412       DIA     0,PTR.
52 11765 101005       MOV      0,2,SNR
53 11766 000767       JMP      R62+3
54 11767 006113       LOOP
55
```

1 2115 .MAIN

```
01  
02 11770 000401 R64: JMP .+1  
03 11771 006111 SETP1  
04 11772 006261 R65: CWAIT ;READ UNTIL A CHAR NOT ZERO IS  
05 11773 000131 C1 ;OBTAINED. IF DIA TO DEVICE 0  
06 11774 000112 NIOS PTR. ;READS THE SAME CHAR  
07 11775 003512 SKPRZ PTR. ;THERE IS A RC SELECT  
08 11776 000777 JMP .-1 ;PROBLEM IN THE READER.  
09 11777 000412 DIA 2,PTR.  
10 12000 101005 MOV 2,2,SNR  
11 12001 000771 JMP R65  
12 12002 004400 DIA 1,2  
13 12003 106415 SUB# 2,1,SNR  
14 12004 006114 EHALL  
15 12005 006113 R66: LOOP  
16  
17 12006 000401 R68: JMP .+1  
18 12007 006111 SETP1 ;CHECK THAT THE DATA BUFFER  
19 12010 176020 SUBZR 3,3 ;HOLDS BOTH ZEROS  
20 12011 054422 STA 3,POSC  
21 12012 060112 NIOS PTR. ;AND ONES IN ALL BIT  
22 12013 102000 ADC 2,2 ;POSITIONS.  
23 12014 126000 ADC 1,1  
24 12015 063612 R69: SKPRZ PTR.  
25 12016 000777 JMP .-1  
26 12017 036461 CWAIT ;(CAWAIT DO NOT USE AC3 AND AC1).  
27 12020 000131 C1  
28 12021 070512 DIAS 2,PTR.  
29 12022 147400 AND 2,1 ;LOCK FOR BITS STUCK(1)  
30 12023 150000 COH 2,2  
31 12024 143400 AND 2,2 ;LOCK FOR BITS STUCK(2)  
32 12025 034400 LDA 3,POSC  
33 12026 175220 MOVZR 3,3  
34 12027 054404 STA 3,POSC  
35 12030 175024 MOVZ 3,3,SNR  
36 12031 000764 JMP R69  
37 12032 000402 JMP .+2  
38 12033 000000 POSC: 0
```

1 0116 ,MAIN

```
01
02 12034 101242      MOVCR  0,0,SZC
03 12035 006114      EHALT                ;CH1 STUCK TO 0
04 12036 101242      MOVCR  0,0,SZC
05 12037 006114      EHALT                ;CH2 STUCK TO 0
06 12040 101242      MOVCR  0,0,SZC
07 12041 006114      EHALT                ;CH3 STUCK TO 0
08 12042 101242      MOVCR  0,0,SZC
09 12043 006114      EHALT                ;CH4 STUCK TO 0
10 12044 101242      MOVCR  0,0,SZC
11 12045 006114      EHALT                ;CH5 STUCK TO 0
12 12046 101242      MOVCR  0,0,SZC
13 12047 006114      EHALT                ;CH6 STUCK TO 0
14 12050 101242      MOVCR  0,0,SZC
15 12051 006114      EHALT                ;CH7 STUCK TO 0
16 12052 101242      MOVCR  0,0,SZC
17 12053 006114      EHALT                ;CH8 STUCK TO 0
18 12054 125202      MOVCR  1,1,SZC
19 12055 006114      EHALT                ;CH1 STUCK TO 1
20 12056 125202      MOVCR  1,1,SZC
21 12057 006114      EHALT                ;CH2 STUCK TO 1
22 12060 125202      MOVCR  1,1,SZC
23 12061 006114      EHALT                ;CH3 STUCK TO 1
24 12062 125202      MOVCR  1,1,SZC
25 12063 006114      EHALT                ;CH4 STUCK TO 1
26 12064 125202      MOVCR  1,1,SZC
27 12065 006114      EHALT                ;CH5 STUCK TO 1
28 12066 125202      MOVCR  1,1,SZC
29 12067 006114      EHALT                ;CH6 STUCK TO 1
30 12070 125202      MOVCR  1,1,SZC
31 12071 006114      EHALT                ;CH7 STUCK TO 1
32 12072 125202      MOVCR  1,1,SZC
33 12073 006114      EHALT                ;CH8 STUCK TO 1
34 12074 006113      LOOP                ;ITERATE TEST.
```

35
36
37
38
39
40

;* NOTE !

;THE TEST RA8-69 DEMAND 160 CONSECUTIVE
;CHAR. (40 CM) OF ALTERNATE
;ONES AND ZEROS.

1 0117 MAIN

```

01
02
03 12075 001401 472: JOP .+1
04 12076 006111 SETPI ;THE READER DONE FLAG
05 12077 006261 C=AIT
06 12100 000131 C1 ;WAS SET WHEN THE
07 12101 001112 NIOS PTR. ;BUSY FLAG WAS ZERO.
08 12102 001212 NIOC PTR.
09 12103 006263 TIMSA
10 12104 001760 1220,
11 12105 006302 SKRDN PTR.
12 12106 000402 JMP .+2 ;TIMEOUT RETURN
13 12107 006114 EH=LT
14 12110 006113 LOOP
15
16 12111 001401 472: JOP .+1
17 12112 006111 SETPI ;THE READER IS SLOWER THAN
18 12113 006261 C=AIT ;2 TIMES THE NORM
19 12114 000131 C1 ;READ SPEED.
20 12115 001112 NIOS PTR. ;(2000/500 CPS).
21 12116 006352 SKRPN PTR.
22 12117 000777 JMP .-1
23 12120 000432 LDA 2,C500
24 12121 000432 STA 2,C004
25 12122 000432 SLD 1,1
26 12123 000425 STA 1,SUM
27 12124 001112 473: NIOS PTR. ;READ 500 CHAR.
28 12125 006264 TIMMS
29 12126 006352 SKRPN PTR.
30 12127 006114 EH=LT ;TIMEOUT RETURN
31 12130 000424 LDA 1,ACTIME;NORMAL RET, TIME IN TIME
32 12131 001424 LDA 2,XD120 ;IN 1215 USEC.
33 12132 000267 DIVIS ;DIVIDE AC1/AC2
34 12133 000415 LDA 2,SUM ;AC1:= AC1/AC2
35 12134 000220 ADD 1,2 ;AC2 TIME IN MSEC.
36 12135 000413 STA 2,SUM
37 12136 000413 SZZ COLN
38 12137 000765 JMP R73
39 12140 001410 LDA 1,X500 ;EQUVALENT TO 500 MSEC.
40 12141 000125 LDA 2,RC2
41 12142 001035 XCV 2,2,SAR ;RC2000 READER?
42 12143 001122 ADDZL 1,1 ;NO, AC1:=4*AC1, EQUV. TO 1000 MSEC.
43 12144 000432 SUBZ= 1,2,SZC ;YES
44 12145 006114 EH=LT ;THE READER SLOWER THAN 2 TIMES
45 12146 006113 LOOP ;THE NORM RATE.
46 12147 000152 JMP RTEST ;START THE TEST AGAIN
47
48 12150 000202 SIM: 2
49 12151 000202 COLN: 2
50 12152 000764 S20: 500.
51 12153 000210 C10: 8.
52 12154 002572 CTIME: TIME
53 12155 000104 C0100: 100.
54 12156 000764 S200: 500.
55
56

```

1 0118 ,MAIN

01

02 12157 000000 LASTP: 2

03 001000

.END REBIN

0119 ,MAIN

A00	010370	88/26	90/31	101/03				
A02	010375	88/27	101/09					
A03	010402	88/28	101/15					
A04	010405	101/18	101/38					
A17	010432	88/29	121/41					
A18	010435	121/44						
A20	010445	88/10	121/52	102/02				
A21	010453	88/11	122/09					
A24	010461	88/12	102/16					
A26	010473	88/13	122/27					
A28	010506	88/14	122/39					
A30	010515	88/15	122/47					
A32	010531	88/16	123/02					
A34	010544	88/17	123/14					
A36	010557	88/18	123/26					
A38	010576	88/19	123/42					
A40	010611	88/20	123/54					
A42	010625	88/21	124/01					
A44	010637	88/22	124/12					
A46	010651	88/23	124/23					
A48	010666	88/24	124/37					
A50	010703	88/25	124/51					
A52	010714	88/26	124/58	105/02				
A54	010730	88/27	125/15					
A56	010751	88/28	125/33					
A58	010770	88/29	125/50					
A60	011023	88/30	126/02					
A62	011017	88/31	126/15					
A64	011034	88/32	126/29					
A66	011050	88/33	126/42					
A68	011062	88/34	126/51	107/02				
A69	011075	107/13	127/23					
A70	011114	88/35	127/29					
A71	011131	127/42	127/52					
A80	011102	88/36	127/57	108/02				
A82	011175	128/05	128/14					
A84	011205	128/17	128/23					
AAANS	007505	91/50	92/08					
AANS	007504	91/43	92/07					
AAANK	005313	67/45	67/47					
ACT0	001071	23/03	23/16					
ACT6	001072	23/29	23/17					
ACTH	001054	22/29	23/02					
ADURS	001711	32/46	31/18	31/28	31/38	31/39	32/02	
AGTYP	003013	43/11	43/18	43/37				
ALPTT	002255	35/59	37/48					
ALTPU	007716	92/36	94/06	94/15	94/26	94/27		
ALTR1	010175	98/28	98/18	98/19				
ALTR2	010211	98/16	98/22					
ALTRP	010167	92/37	98/02	98/21				
AMENO	001721	32/15	32/23					
ANS	007503	91/36	92/26					
APESA	001322	27/18	27/24					
BAD	010231	98/25	98/33	98/38				
BAD00	010236	98/32	98/42					
BENUK	005323	67/51						
BC20	001607	31/32	31/46					
BC377	001617	32/55	32/59					
BEND	001712	32/03						

2122 MAIN

BIDIG	201115	75/27	75/43	75/49						
BINAD	201547	28/19	30/29	37/23						
BINFI	201375	27/63	28/17	37/22						
BINLA	201376	28/22	28/12	36/41						
BLOCA	201625	32/58	31/10	31/12	31/44					
BMEAD	201732	32/19	32/24							
BRESA	201356	27/41	27/47							
BSTRP	201723	31/62								
BTEST	201622	32/42	31/21							
BUILD	201552	32/12	31/17	31/19	31/37					
BZNUK	205312	67/27	67/44							
BZOUT	200437	18/18	18/25	22/12	27/57	82/21				
C1	200131	14/27	99/27	139/52	112/16	112/26	112/36	112/52	111/27	
		111/14	111/42	111/56	112/25	112/12	112/21	112/38	112/56	
		113/27	113/22	113/39	113/53	114/13	114/29	114/47	115/25	
		115/27	117/26	117/19						
		117/51								
C12	212153									
C1202	212224		95/32							
C177	207747	94/44	96/26							
C22	211463	129/31	111/37	113/24	113/23	113/37				
C4	210713	124/42	124/61	125/35						
C42	202134	14/12	121/16	111/23						
CSSEC	211151	127/32	127/59							
C776A	210265	96/12	96/35							
CAB2	207362	89/11	92/31							
CALTR	207367	89/54	92/36							
CALTR	207372	92/27	92/37							
CCAR	206241	13/23	26/42	59/15	59/19	59/23	68/26	82/11	82/23	
CCNIO	211462	111/16	111/36							
CCRLF	206243	13/25	27/32	28/52	32/43	43/28	49/32	49/36	52/28	
		52/52	59/11	59/28	61/35	61/51	64/32	83/29	83/16	
		85/43	86/13	86/29	98/26					
		94/45	96/13	96/15						
CCTIM	207752									
CCTP	211155	127/27	127/22	127/36	127/51	127/63				
CCTRP	207365	89/39	92/34							
CCTRP	207366	89/46	92/35							
CDATT	206247	13/29	36/38	21/44	22/24	83/31	85/58	86/32		
CDSIA	206155	13/15								
CODEC	206257	13/17	37/57	82/22	86/24					
COICL	206246	13/28	28/41	28/47	32/42	49/35	52/23	52/47	61/36	
		64/31	83/17	85/47	86/17	128/22	128/37			
COIS	201773	32/39	32/62							
COISP	206244	13/26	26/42	28/32	28/41	28/48	32/49	36/29	43/22	
		43/42	52/24	52/48	64/33	64/42	68/45	83/22	85/54	
		86/25	128/29	128/38						
COOCT	206256	13/16	28/35	32/47	36/36	43/48	61/11	61/29	61/39	
		61/43	63/11	63/22	63/33	83/26	85/52			
COOCT	206245	13/27	62/25	32/22						
COVCD	207671	93/19	93/24	93/42	94/24					
COZCC	206262	13/12	59/36	91/22	91/32	92/17	93/11			
CFLTP	207371	92/15	92/32							
CFLTR	207372	92/22	92/39							
CGTBI	206123	13/32	91/29	92/12						
CGTDC	206125	13/32	37/51	91/39						
CGTOK	206124	13/31	59/37	61/12	61/32	63/12	63/23	63/34	93/12	
CGTSC	206126	13/33								
CGTTA	206127	13/34								
CH42	206622	82/17	82/22							
CHPAT	206252	13/12	26/44	83/37						

0122 .MAIN

CPU23	002550	39/53							
CPU24	002551	39/54							
CPU25	002552	39/55							
CPU3	002530	39/37							
CPU4	002531	39/38							
CPU5	002532	39/39							
CPU6	002533	39/40							
CPU7	002534	39/41							
CPUIN	002523	39/22	39/21	39/31					
CPUNG	003015	40/32	43/21	43/32	43/39	43/46			
CSUES	006071	13/27	37/45	59/31	61/06	61/24	63/36	63/17	63/28
		91/03	91/33	92/12	93/06				
CR00	007303	89/21	90/32						
CRBR	001275	22/33	22/38	22/45	22/47	22/57	22/59	22/02	23/23
		23/21	23/29	23/31	23/33	23/34	23/49	24/32	24/33
		24/36	24/39	25/35	25/38	25/41	25/48		
CREDG	007373	89/18	90/42						
CRES*	006073	13/29	25/19	25/26	28/30	32/53	81/23	81/31	82/23
		82/03	83/05	83/42	84/54				
CSAMS	006072	13/28	36/13	36/21	36/49	37/44	61/05	63/25	67/27
		87/39	87/51	89/27	89/16	89/26	89/34	89/42	89/49
		90/23	92/11	92/18	92/25	93/05			
CSAV0	006537	81/25	81/40	81/45	81/62	83/02	83/10	83/32	83/48
		83/50	83/58						
CSAV1	006536	81/04	81/41	81/46	81/61	83/03	83/12	83/33	83/47
		83/51	83/59						
CSAV2	006535	81/03	81/42	81/47	81/60	83/04	83/14	83/34	83/46
		83/52	83/60						
CSKP	003144	44/36	45/19	46/42					
CSAPL	007364	69/31	90/33						
CTAFZ	010342	99/63	100/22						
CTAIN	006051	13/11							
CTDEC	006053	13/13	37/49	82/19	86/23				
CTIME	012154	117/31	117/52						
CTLOT	006052	13/12	27/34	28/37	32/48	36/37	43/47	61/12	61/28
		61/30	61/42	63/10	63/21	63/32	83/11	83/13	83/15
		83/25	85/53	98/32	98/34				
CTR	010025	94/17	94/19	94/21	95/51	97/25	97/27	97/29	
CTR1	010243	97/42	97/45	97/47	98/07	98/10	98/12	98/42	
CTRP0	010127	90/34	97/23	97/34	97/35				
CTRP1	010151	97/43	97/53	97/54					
CTRP2	010165	97/51	97/55						
CTRPE	010144	90/35	97/38	97/56					
CTYPE	006042	13/04							
CTZ00	006054	13/14	59/35	91/27	91/37	92/16	93/13		
CWAIT	006061	13/19	21/61	25/23	25/44	27/19	28/39	28/45	28/53
		42/30	43/29	43/49	59/07	59/16	59/27	59/24	96/14
		99/26	106/46	127/31	109/51	110/15	112/25	110/35	110/49
		111/26	111/13	111/41	111/55	112/04	112/11	112/22	112/37
		112/55	113/06	113/21	113/38	113/52	114/12	114/28	114/46
		115/24	115/26	117/35	117/18				
CXLPT	001771	32/35	32/60						
CXTTU	001772	32/37	32/61						
CYCLE	006047	12/53	81/02						
CYCTS	006500	81/13	81/28						
CYMRK	006510	81/25	81/37	81/33	81/37				
D102	211152	106/47	127/60						
D377	010365	99/28	120/21						
D52	011154	107/26	127/35	127/62					

0123 MAIN

DE22	012152	117/23	117/52							
DATA	021642	31/25								
DDCIG	025566	69/21	70/36	70/48						
DDEVG	027710	93/29	94/12							
DDICH	021131	23/43	23/51							
DECEX	020640	20/25	20/41							
DECCC	020615	20/22	20/40	20/44	20/54					
DECOT	020623	20/28	20/33							
DECP	020631	20/29	20/34							
DECTB	020642	19/22	20/44							
DEL	027414	91/12	91/16	91/21						
DELAY	020124	14/22	91/17	94/32						
DELBI	020672	75/17	75/18	75/20						
DELOC	025447	69/18	69/19	69/21						
DELOK	025751	73/17	73/18	73/20						
DELOY	025344	68/24	68/27	68/12						
DELRE	025425	68/12	68/23	69/22						
DELSC	025662	72/16	72/17	72/19						
DELTE	025335	68/22	69/13	73/46	78/28					
DELTX	020220	77/16	77/17	77/19						
DEV	020217	14/27	84/14	84/37	84/58	85/18	85/38			
DEVC	027715	93/29	94/13							
DEVC1	027651	93/46	93/62							
DEVC2	027656	93/44	93/52							
DEVC3	027641	93/22	93/28	93/37	93/49					
DEVGM	027600	93/43	93/14							
DEVRE	027676	93/37	93/50	94/29						
DHEAD	020615	82/37	83/21							
DIFF	021666	31/26	31/45							
DIGIN	020075	12/33	37/54	59/42	61/15	61/33	63/15	63/26	63/37	
		64/25	68/35	70/12	72/63	74/35	76/33	78/54	91/12	
		91/16	91/42	92/21	93/15	93/25				
DIS	020035	14/22	23/47	24/15	24/18	24/19				
DISAT	021176	12/11	25/14							
DISP1	021114	23/37	23/44							
DISS4	021211	25/22	25/26							
DIVID	020670	13/26	41/14	82/18						
DIVIS	020667	13/25	117/33							
DLOOP	022671	41/57	41/62							
DLOCA	011233	108/32	108/43							
DYEND	021713	32/29	32/14							
DYES	027432	91/25	91/28							
DCK	011245	108/39	108/45							
DPASS	027261	86/26	86/35							
DPOCN	025125	62/12	63/29							
DPMEM	025127	35/20	63/24							
DPMOC	025131	63/26	63/14							
DPMXF	025144	63/17	63/25							
DPMYT	025157	63/28	63/36							
DPRRT	025173	63/42	63/46							
DCUE	027535	92/14	92/27							
DSUES	027476	91/35	92/24							
DSAGU	024750	59/33	62/26							
DSIGN	025565	69/27	72/27	72/32	72/47	71/16	71/18	71/31	71/38	
DVCO	027672	93/21	93/27	93/32	93/55	94/05				
DYMF	025101	61/28	62/24							
EBFLG	020533	82/63	81/12	81/58	83/39					
ECM02	025753	73/22	73/32							
ECM03	020074	75/22	75/32							

0124 .MAIN

ECHO4	005451	69/23	69/33						
EFLAG	006530	80/60	81/11	81/14	81/28	81/55	83/28	83/55	
EHALT	006114	13/39	99/22	101/06	101/12	101/23	101/34	101/48	102/06
		102/13	102/20	102/24	102/32	102/36	102/44	102/52	102/57
		103/27	103/11	103/20	103/23	103/32	103/35	103/39	103/48
		103/51	103/50	104/00	104/07	104/09	104/18	104/20	104/30
		104/34	104/45	104/48	104/56	105/08	105/12	105/26	105/30
		105/42	105/46	105/56	105/59	106/28	106/12	106/21	106/26
		106/35	106/39	106/49	107/12	107/16	107/26	107/41	107/45
		107/55	107/26	109/12	109/19	109/26	109/37	109/46	109/57
		110/06	110/10	110/20	110/30	110/40	110/44	110/54	111/12
		111/25	111/20	111/47	111/50	111/61	112/00	112/10	112/15
		112/26	112/28	112/43	112/45	112/61	112/63	113/14	113/16
		113/30	113/32	113/45	113/47	113/59	113/63	114/07	114/18
		114/23	114/34	114/38	114/45	115/14	116/03	116/05	116/27
		116/09	116/11	116/13	116/15	116/17	116/19	116/21	116/23
		116/25	116/27	116/29	116/31	116/33	117/13	117/30	117/44
EMEND	001714	32/13	32/13						
ENRDK	003274	46/20							
ENTCO	006434	80/47	82/52	80/57					
ENTP0	006416	12/50	82/43						
ENTP1	006423	12/51	82/48						
ENTP2	006430	12/52	82/53						
ENTYM	003253	46/18							
ERBCT	006532	80/62	81/26	81/39	81/57	82/43			
ERHSA	006541	82/27	83/41						
ERRCT	006531	80/61	81/27	81/09	81/19	81/56	82/12	82/40	
ERRPT	006623	82/43	83/35	83/53	83/61				
ERRNH	006705	83/43	83/50						
ERR01	006626	82/42	83/02						
ERR02	006670	83/30	83/37	83/57					
ERR03	006711	83/28	83/55						
ERROR	006617	12/54	82/30						
EXDJS	002500	18/51	18/63	19/51	19/61				
EXPEM	005011	35/18	61/04						
EXXMF	005013	61/26	61/14						
EXXMT	005026	61/24	61/32						
EXPRT	005041	61/35	61/52						
EXIYP	006477	18/56	18/62	19/46	19/56				
FDIST	001125	23/46	23/48	33/01					
FIRST	007673	93/33	94/26						
FITYP	002756	28/01	43/08						
FLAG	011100	95/31	108/22	108/03	108/07				
FLOT	012334	99/30	99/53	99/61					
FLTCT	012363	99/29	99/32	99/35	99/37	99/47	99/48	99/52	100/19
FLTPU	012275	90/38	99/27						
FLTRE	012312	90/39	99/42						
FMAOK	005405	61/23	61/37	61/40	61/46	61/47	61/56	63/27	63/42
		63/43							
FMEND	001715	32/11	32/24						
FRASH	006603	82/07	82/32						
FRATE	006545	81/17	81/34	82/02					
FRATR	006577	82/22	82/26	82/27	82/28				
FUB	000033	14/20	20/31	26/36					
FUN	000032	14/19	25/43	25/47	26/20	26/35			
GET	001671	31/50	31/50	31/61	31/63				
GETB1	006057	75/29	75/13	75/23					
GETB1	006052	12/45	75/04						
GETCH	005246	67/26	69/10	73/48	78/26				

0125 .MAIN

GETD1	005434	69/10	69/14	69/24			
GETDC	005426	12/47	69/04				
GETQ1	005736	73/29	73/13	73/23			
GETON	005731	12/46	73/04				
GETRE	005324	67/26	67/32	67/42	67/56		
GETS1	005650	72/29	72/13	72/21			
GETSC	005643	12/48	72/04				
GETT1	006166	77/29	77/13	77/21			
GETTA	006161	12/49	77/04				
GETYP	001423	28/23	28/27				
GMEND	001716	28/63	32/12	36/42			
GOOD	010230	98/24	98/29	98/37			
GOODE	010232	98/28	98/39				
GTCAR	001557	30/13	32/15	30/22	30/54	31/10	31/13
GTTT1	001579	30/25	30/32				
H1011	005703	72/27	72/40				
H1033	005702	72/28	72/39				
H1044	005701	72/26	72/38				
H2040	005771	73/28	73/42				
H2060	005772	73/31	73/41	73/54			
H2070	005773	73/34	73/42				
H3040	006112	75/28	75/42				
H3060	006113	75/31	75/41	75/46			
H3062	006114	75/34	75/42				
H5011	006214	77/26	77/35				
H5012	006414	78/49	79/25				
H5015	006413	78/42	79/24				
H5040	006215	77/27	77/36				
H5079	006216	77/37	77/45				
HAATT	001221	12/12	25/35				
HC11	005502	68/55	69/52				
HC13	005503	68/49	69/53				
HC15	005504	68/52	69/54				
HC177	005325	67/36	67/57	68/05			
HC30	005505	68/22	69/55				
HC40	005506	69/29	69/56				
HC44	005507	68/18	68/32	69/57			
HC52	005510	68/32	69/34	69/58			
HC53	005511	69/42	69/50	70/10	71/29		
HC55	005512	69/43	69/60	70/22	71/12		
HC60	005571	69/31	69/46	70/34	70/51		
HC72	005513	69/37	69/61				
HJVAR	010023	95/38	95/42	95/49			
HLECR	006415	79/14	79/26				
HMEND	000074	12/32	32/26	36/33			
HSPUN	000126	14/24	95/18	96/42			
HSREA	000127	14/25	95/12	99/03			
IALT1	000143	14/36	15/08				
IALT2	000144	14/37	15/09				
IBZOT	006542	82/21	83/27				
ICAR	000241	12/25	13/03				
ICRLF	000243	12/27	13/05				
IDATT	000247	12/11	13/09				
IDBIN	000055	12/17	13/15				
IDDEC	000057	12/19	13/17				
IDICL	000046	12/10	13/04				
IDISP	000044	12/08	13/02				
IDIVO	000070	12/23	13/26				
IDIVS	000067	12/27	13/25				

2126 MAIN

IOOCT	000056	12/13	13/16							
IOOCT	000045	12/29	13/37							
IOXC	000020	11/15	44/26	44/32	46/28	46/39				
IOX1	000021	11/16	44/24	44/48	45/25	45/28	45/33	46/26	46/55	
IOX2	000022	11/17	45/44	45/55	45/61	47/23	47/46			
IOX3	000023	11/18	44/28	45/11	46/30	87/29	87/31	87/41	87/43	
		87/53	87/55							
IOZCC	000060	12/22	13/18							
IGNOR	001612	32/43	32/54	32/57						
IGTBI	000123	12/45	13/32							
IGTCC	000125	12/47	13/32							
IGTCK	000124	12/46	13/31							
IGTSC	000126	12/48	13/33							
IGTTX	000127	12/49	13/34							
IHAAT	000050	12/12	13/12							
IHALT	000114	12/54	13/39							
ILL	001267	23/24	23/25	23/26	23/27	23/28	23/10	23/11	23/12	
		23/14								
ILLEG	005367	68/31	69/36	69/39	69/48	73/45	78/31			
ILLGR	005366	68/13	68/15	68/32						
ILOOP	000113	12/53	13/38							
ILORE	000122	12/22	13/45							
IMEND	002312	36/28	36/42	36/53						
IMESS	000040	12/24	13/22							
IMULT	000066	12/26	13/24							
INDAD	003153	45/12	45/27							
INDA1	003122	44/24								
INDA2	003350	47/32								
INISA	006602	82/24	82/31	83/26						
INIPB	010000	95/24	95/26							
INIPC	007777	95/23	95/25							
INIPA	010001	95/26	95/27							
INIPZ	010003	95/28	95/29							
INIT	007751	92/42	95/22							
INITR	010022	95/22	95/23	95/28						
INNOX	005517	67/29	67/52							
INRET	005421	68/38	68/41	68/42	68/47	68/60	69/24	73/52	78/31	
INSE0	003145	45/27	45/22							
INSTA	003132	45/28	45/29							
INSTb	003133	45/29	45/12							
INSTR	003123	45/22	46/15							
INSA1	003121	44/59								
INSA2	003347	47/21								
INTER	005402	68/41	72/13	73/51	78/32					
INTPT	010026	11/27	95/34							
INTTI	005321	67/23	67/34	67/41						
INXAS	003225	44/27	46/11	46/29						
IP.PR	007675	93/17	93/23	93/32	93/34	94/23				
IPASS	003123	12/51	13/46							
IPSA	000140	14/33	15/25							
IPTES	000136	14/31	15/23							
IPUNC	000141	14/34	15/26							
IPUNF	000145	14/32	15/12							
IPUT	007733	94/24	94/29							
ISSBS	000071	12/29	13/27							
IPEDC	000142	14/35	15/27							
IPEDF	000146	14/39	15/11							
IPESA	000077	12/42	12/43	93/35						
IPRES	000147	14/42	15/12							

2127 .MAIN

IRESA	000073	12/31	13/29			
IRTES	000137	14/32	15/04			
ISAMS	000072	12/32	13/28			
ISTAA	000115	12/55	13/42			
ISTAC	000543	81/21	82/02			
ISTAN	000116	12/56	13/41			
ISTAP	000121	12/59	13/44			
ISTAS	000122	12/58	13/43			
ISTAW	000117	12/57	13/42			
ISTPE	000110	12/50	13/35			
ISTP1	000111	12/51	13/36			
ISTP2	000112	12/52	13/37			
ITBIN	000051	12/13	13/11			
ITDEC	000053	12/15	13/13			
ITIMS	000064	12/24	13/22			
ITIRO	000065	12/25	13/23			
ITISK	000063	12/23	13/21			
ITCCT	000052	12/14	13/12			
ITR	000526	80/57	81/21	81/53	82/15	
ITRAG	000525	80/45	80/50	80/55	81/20	81/52
ITRCT	000527	80/58	81/12	81/22	81/54	82/16
ITRP1	000523	80/44	80/51	81/52		
ITRP2	000524	80/49	80/54	80/56	81/51	82/13
ITYPE	000042	12/06	13/04			
ITZOC	000054	12/16	13/14			
IWAIT	000001	12/21	13/19			
IWAOP	000062	12/22	13/20			
JMEND	001732	32/22	36/43			
K17	010364	99/01	100/20			
KCP0	002450	38/32				
KCP1	002451	38/33				
KCP10	002460	38/40				
KCP11	002461	38/41				
KCP12	002462	38/42				
KCP13	002463	38/43				
KCP14	002464	38/44				
KCP15	002465	38/45				
KCP16	002466	38/46				
KCP17	002467	38/47				
KCP2	002452	38/34				
KCP20	002470	38/48				
KCP21	002471	38/49				
KCP22	002472	38/50				
KCP23	002473	38/51				
KCP24	002474	38/52				
KCP25	002475	38/53				
KCP3	002453	38/35				
KCP4	002454	38/36				
KCP5	002455	38/37				
KCP6	002456	38/38				
KCP7	002457	38/39				
KEY4	003354	47/26	47/11	47/14		
KEY5	003362	47/28	47/12			
KEY8	003351	45/40	47/02			
KJNC	002476	38/21	38/22	38/55		
KINDI	002447	38/17	38/30			
KSTAC	001372	27/28	27/63	59/49		
LALOC	001373	27/01	28/14	28/36		
LAPRG	001374	27/02	28/16			

212A MAIN

LAST	007674	93/47	94/27							
LAST1	005725	72/26	72/19	72/42	72/44	72/54				
LAST2	006041	73/26	73/21	74/04	74/17	74/27				
LAST3	006150	75/26	75/21	76/04	76/17	76/25				
LAST4	005570	69/26	69/22	70/16	70/50	71/23				
LAST5	006326	77/26	77/19	77/39	78/04	78/22				
LASTN	005424	68/11	68/17	68/63	70/50	72/42	74/27	76/25	78/22	
LASTP	012157	94/27	118/02							
LENGH	010063	94/25	94/35	96/09	96/33	97/06	97/33	98/43		
LERE1	005700	72/25	72/32	72/32	72/35	72/36				
LERE2	005770	73/27	73/37	73/38						
LERE3	006111	75/27	75/37	75/38						
LERE4	005501	69/28	69/49	69/50						
LERE5	006213	77/25	77/29	77/32	77/33					
LETE1	005665	72/14	72/25							
LETE2	005755	73/14	73/27							
LETE3	006076	75/14	75/27							
LETE4	005453	69/15	69/28							
LETE5	006223	77/14	77/25							
LLPTT	002257	35/61	37/56							
LYADR	005067	61/34	61/45	61/58	63/38	63/42				
LYSK	001163	22/27	24/27							
LOADA	002337	36/58	37/03							
LOADB	002314	35/28	36/48							
LOADF	002336	36/56	37/02							
LOADR	002327	36/59	37/21							
LOCKA	003424	47/18	47/32							
LOOP	006113	13/38	121/27	101/13	101/35	101/49	102/07	102/14	102/25	
		102/37	102/45	102/58	103/12	103/24	103/40	103/52	104/01	
		104/10	104/21	104/35	104/49	104/57	105/13	105/31	105/47	
		105/00	106/13	106/27	106/40	106/50	107/27	107/56	109/07	
		109/13	109/20	109/27	109/38	109/47	109/58	110/11	110/21	
		110/31	110/45	110/58	111/30	111/51	112/01	112/16	112/29	
		112/46	113/00	113/17	113/33	113/48	114/00	114/08	114/24	
		114/39	114/54	115/15	116/34	117/14	117/45			
LOOPR	006540	80/43	80/48	80/53	81/03	81/43	81/63			
LSTSA	001366	27/14	27/18	27/21	27/25	27/33	27/44	27/56	28/28	
M120	007667	93/29	93/53	94/02						
MASK	010062	94/22	90/32	97/32	97/48	98/44				
MBILO	001510	26/42	28/44	29/10						
MOYEX	002264	36/16	37/31							
MOYNM	002273	36/22	36/26	36/50	37/24					
MOPUT	005022	43/43	43/45	43/11						
MORLF	004743	32/55	40/02	68/44	68/46	82/26				
MDOTR	002740	41/31	41/55	42/00						
MDEVG	007677	93/27	94/11							
MDMNC	005113	62/08	63/07	63/08						
MDMNF	005120	62/10	63/18	63/19						
MELOC	001470	11/06	29/04							
MESCH	000434	18/16	18/21							
MESSA	002422	18/10	18/17							
MHEAD	006604	82/34	83/18							
MIMEX	002313	36/23	36/43	36/51	37/25	37/32				
MLLOC	001501	28/33	28/35	29/08	36/32	36/32				
MLOOP	002656	41/32	41/36							
MLORE	007077	85/55	85/57	86/46						
MLPTT	002223	35/31	37/46	37/47						
MYES	007424	91/04	91/26							
MODAT	001414	28/22	28/26							

0129 MAIN

MOFID	002246	35/42	35/51			
MOGTT	002250	35/38	35/53			
MCLAD	002247	35/41	35/52			
MCPFB	002230	35/37	37/62			
MCPTR	002245	35/37	35/46	35/50		
MOREP	002236	35/43	35/49			
MPASS	007050	86/28	86/33			
MPOMO	004745	59/27	60/24			
MALE	007523	92/13	92/25			
MQUES	007486	91/34	92/22			
MRET	011161	108/23	108/25	108/31	108/34	108/40
MSAMS	001530	29/15	32/50	32/52		
MSABU	004755	59/32	60/28			
MSAV	002701	41/30	41/38	41/54	41/63	42/21
MST10	002623	40/59	41/11			
MST1A	002557	40/19	40/23			
MST1B	002562	40/17	40/22			
MST1C	002627	41/02	41/04	41/16		
MST1D	002630	40/27	41/02	41/03		
MST1M	002553	12/24	40/15			
MST1O	002624	40/21	40/62			
MST1R	002652	40/15	41/17	41/19	41/20	41/21
MSARG	001522	27/32	29/13			
MULSA	005564	70/39	70/43	70/46		
MULTE	005556	70/26	70/39			
MULTI	006066	13/24	41/12	82/14		
MX2SP	005207	64/27	64/45	86/20		
MXMNF	005070	61/07	62/02			
MXMYI	005106	61/25	61/26	62/06	63/29	63/33
MXGUE	005206	64/25	64/41	64/43		
M.LON	011206	108/11	108/20	108/25		
M.CK	011216	108/12	108/21	108/34		
MBEVE	006366	78/41	78/57			
MC125	003635	50/13	52/45			
M08	003634	49/25	52/44			
MCTYP	003540	49/34	49/37	49/39		
MEXIS	003553	49/27	49/43			
MFTYP	003531	49/21	49/32			
MINH1	000577	20/03	20/06			
MIRET	003541	49/24	49/38	49/40		
MITYP	003532	49/33	50/53			
MN10	000772	22/13	22/49			
MN500	003632	49/17	50/42			
MN9	003633	49/18	50/43			
MOCX	006517	81/16	81/26	81/35	81/45	
MORAT	006574	82/09	82/25			
MOTRE	003575	50/02	50/10	50/11		
MOTYM	003542	49/41	50/05	50/07		
MCTYP	003564	49/33	50/22			
MRESA	001352	27/17	27/42			
MRTYP	003533	49/31	49/34			
MTDEC	003630	50/14	50/34	50/39		
MTREP	003602	50/17	50/35			
MTRER	003631	50/15	50/22	50/29	50/36	50/40
MTRZ1	003627	50/16	50/37	50/38		
MTTYF	003576	49/16	50/13			
MUK	000034	14/21	67/45	67/51		
NUMB2	006042	73/08	74/23	74/18	74/23	74/28
NUMB3	000151	75/08	76/23	76/18	76/21	76/26
						74/34
						76/32

005567	69/29	70/26	70/25	70/28	70/49	71/19
005724	72/28	72/41	72/49	72/55	72/62	
003714	43/12	43/38				
002437	38/21	38/25				
003475	43/36	49/24				
002572	18/52	19/52	19/62	19/63		
003515	49/22	49/24				
003636	52/33	52/47				
003645	52/49	52/52	53/55			
002651	22/18	22/54				
002212	74/17	74/26				
006147	76/12	76/24				
005572	69/17	70/22	71/22			
006212	73/16	74/22	74/32			
006123	75/16	76/22	76/28			
005642	71/32	71/36	71/41			
005726	72/15	72/44	72/58			
005835	71/11	71/14	71/38			
006217	77/15	77/39	78/35			
005774	73/22	73/43	73/57			
002376	37/22	37/35				
005422	68/39	68/43	70/23	73/52	78/33	
003224	45/23	45/58	45/62	46/12		
006363	78/33	78/52				
007324	14/36	59/48				
007526	92/41	92/11	92/22			
007541	92/15	92/29				
007272	86/13	86/16	86/38	95/26		
007171	86/15	86/37	95/25			
007272	86/21	86/22	86/42	95/27		
001233	22/52	22/55				
001237	22/36	22/41	22/54			
001274	22/44	22/53	23/22	25/14	25/17	25/33
006601	82/21	82/32				
004732	59/14	59/45				
004733	59/13	59/46				
004734	59/22	59/47				
007327	14/34	49/33				
004731	59/29	59/44				
004732	59/12	59/12	59/43			
002613	19/21	22/22				
002521	19/13	19/17				
002523	19/15	19/19				
002463	18/46	19/23	19/33	20/14	20/39	20/41 20/42
003224	99/35	99/42				
007341	14/33	92/12				
004573	18/25	19/25	22/22	22/16	21/25	21/42 24/28
006141	75/19	76/17	76/31			
005533	69/22	72/25	72/15			
006232	73/19	74/17	74/33			
005717	72/18	72/54	72/61			
005547	72/22	72/24	73/32			
006324	77/12	78/23	78/38			
002377	37/39	37/36				
002233	115/27	115/32	115/34	115/38		
002217	95/35	95/44				
004622	12/41	59/24				
002122	12/39	12/41				
002276	12/39	26/28	95/39	95/44		

0131 .MAIN

PGUES	007376	89/27	89/36	89/51	91/02	91/11	91/22			
PRINT	008101	12/42								
PROG	007245	26/43	28/49	28/52	59/30	89/02				
PRTYP	003016	26/62	43/41							
PSAAN	004735	59/34	59/48							
PSETP	004737	59/25	59/52	59/56						
PSTAC	004736	59/04	59/49	59/55						
PSUM	011156	107/05	107/19	107/21	107/34	107/48	107/59	108/00		
PSWIT	007300	14/33	89/25							
PTAB	001160	22/22	24/24							
PTAB1	003656	35/54	51/23							
PTAB2	004057	35/55	53/23							
PTAB3	004260	35/56	55/23							
PTAB4	004461	35/57	57/23							
PTP.	000013	14/09	14/29	95/14	95/15	96/45	96/47	101/05	101/11	
		101/19	101/24	101/33	101/43	101/52	102/11	102/12	102/18	
		102/19	102/23	102/29	102/30	102/31	102/35	102/41	102/43	
		102/49	102/50	102/51	102/53	102/56	103/04	103/06	103/10	
		103/16	103/19	103/22	103/28	103/31	103/33	103/34	103/38	
		103/44	103/47	103/49	103/50	103/56	103/59	103/61	103/62	
		104/03	104/06	104/08	104/14	104/17	104/19	104/26	104/29	
		104/39	104/44	105/04	105/07	105/22	105/25	105/38	105/41	
		105/52	105/55	106/04	106/07	106/17	106/22	106/31	106/34	
		106/44	106/45	106/48	107/08	107/11	107/13	107/15	107/37	
		107/42	107/42	107/44	108/00	108/14				
PTB.	000012	14/08	14/28	95/08	95/09	99/08	99/10	109/05	109/11	
		109/18	109/24	109/25	109/53	109/56	110/04	110/05	110/09	
		110/17	110/19	110/27	110/28	110/29	110/37	110/39	110/43	
		110/51	110/52	110/53	110/57	111/08	111/11	111/15	111/24	
		111/28	111/36	111/43	111/46	111/49	111/57	111/60	111/62	
		111/63	112/06	112/09	112/13	112/14	112/22	112/25	112/27	
		112/39	112/42	112/57	112/60	113/10	113/13	113/26	113/29	
		113/41	113/44	113/55	113/58	114/14	114/17	114/32	114/33	
		114/48	114/49	114/51	115/26	115/07	115/09	115/21	115/24	
		115/28	117/27	117/28	117/11	117/20	117/21	117/27	117/29	
PUNDE	007734	94/18	94/31	97/03	97/26					
PUNRE	007745	94/31	94/37	94/39	94/42					
PUNTE	007257	14/31	89/24	89/06						
PUR	010271	94/29	96/41	97/05	97/32	100/22				
PUR1	010101	96/44	96/50							
PURRE	010125	96/41	96/48	96/53	96/55					
QCHAR	000673	21/27	21/10							
QDIOL	001173	24/34	24/37							
QOCUT	001144	24/27	24/11							
QHAAT	001227	25/39	25/42							
QMEND	002400	37/11	37/22	37/27	37/37					
QTYPE	002734	21/42	21/46							
QUESA	005225	64/23	64/26	64/37						
QUESD	005230	64/23	64/34							
QUESM	005232	64/21	64/36							
QUEST	005235	64/32	64/39							
QUEST	005234	64/25	64/38							
R02	011252	88/37	92/32	109/03						
R02	011257	88/38	109/09							
R04	011264	88/39	109/15							
R06	011272	88/40	109/22							
R08	011300	88/41	109/29							
R10	011312	88/42	109/40							
R11	011322	88/43	109/49							

2132 MAIN

R12	011334	88/44	112/02						
R14	011346	88/45	112/13						
R16	011357	88/46	112/23						
R18	011372	88/47	112/33						
R20	011405	88/48	112/47						
R21	011421	88/49	111/02						
R22	011424	111/25	111/33						
R36	011464	88/50	111/34	111/39					
R38	011501	88/51	111/53						
R40	011516	88/52	112/22						
R42	011535	88/53	112/18						
R44	011551	88/54	112/31						
R46	011571	88/55	112/48						
R48	011612	88/56	113/02						
R50	011632	88/57	113/19						
R52	011651	88/58	113/35						
R54	011667	88/59	113/52						
R56	011726	88/60	114/02						
R58	011715	88/61	114/10						
R60	011734	88/62	114/26						
R62	011752	88/63	114/41	114/53					
R64	011772	89/20	115/22						
R65	011772	115/24	115/11						
R66	012005	115/15							
R68	012006	89/21	115/17						
R69	012015	115/24	115/36						
R70	012075	89/22	117/03						
R72	012111	89/23	117/16						
R73	012124	117/27	117/38						
RACAN	005334	67/28	68/02						
RADYN	005326	67/12	67/16	67/58					
RAEND	005277	67/14	67/31						
RALIF	005327	67/11	67/24	67/50					
RALT	007333	14/37	92/02						
RAMIN	005332	67/22	67/62						
RANDI	005255	67/13	67/17						
RANK	003254	46/16	46/19						
RAPLU	005331	67/20	67/61						
RASPA	005330	67/26	67/60						
RAZER	005333	67/10	67/63						
RBZOT	008451	18/25	18/34	18/36					
RC2	000125	14/23	91/46	117/42					
RC204	007460	91/42	91/48	91/53					
RCOUN	007316	14/35	89/41						
RDOCT1	010066	96/18	96/36						
RDOCT2	010067	96/19	96/22	96/37					
RDOCT3	010072	96/28	96/38						
RDOCTR	010045	96/24	96/18						
RDESK	010213	97/55	98/20	98/23	99/58				
RDAT	010026	94/38	96/02	97/43	98/28				
READ	010115	97/24	97/12						
REBIN	001400	11/28	28/08	118/23					
REDDU	007442	92/42	91/32	91/41	91/52				
REDTE	007267	14/32	89/15						
REG0	001122	21/37	21/45	21/52	21/54	21/63	23/24	24/22	24/10
		24/17							
REG1	001121	21/47	22/27	23/25	24/11	24/22			
REG2	001122	21/48	22/06	23/26	24/12	24/21			
REG3	001123	21/38	21/41	21/44	22/08	23/27	24/23	24/26	24/09

REACQ	007256	24/22	89/17	89/27	89/37	89/38	89/45	89/53	90/06
		89/24	90/14	90/21	90/27				
RECF2	006027	74/22	74/06	74/27	74/29	74/13	74/14	74/15	
RECF3	006140	76/22	76/06	76/37	76/29	76/13	76/14	76/15	
REPL4	005551	72/15	72/29	72/31	72/32				
REPL5	006325	78/23	78/12	78/19	78/22				
REPR	010064	96/22	96/16	96/34					
RESTR	007355	14/42	92/24						
RETE	006400	79/26	79/12	79/11					
RETCP	005602	71/22	71/05	71/26	71/28	71/41	71/42	71/44	
RETCC	007423	91/22	91/18	91/24	91/32	91/47	92/11	92/23	
RETUR	006534	81/22	81/48	81/50	82/39	82/44	82/45	83/22	83/44
RETYP	003010	43/28	43/31	43/33	43/34	43/41	43/51		
REVD	003162	45/39							
REYDA	003172	45/47	45/57						
REYDB	003207	45/51	45/60						
REYLC	003203	45/56	46/22						
REYLD	003175	45/52	45/54						
RFLT1	010320	99/48	99/57	99/50					
RFLT2	007347	14/39	90/17						
RINFI	000452	16/37	22/22	20/26	20/27				
RLPTT	002425	37/45	37/53	37/59					
RKEND	002375	37/25	37/26	37/33	37/34				
RMSK	001161	22/22	22/25	24/25					
RPASS	007067	85/46	85/49	85/50	86/12	86/14	86/31	86/37	
RPOUT	000462	18/22	18/24	18/26	18/27	18/19	18/45	18/48	18/53
		19/20	19/24	19/26	19/43	19/48	19/53	19/58	20/15
		22/17							
RPSAG	004715	59/31	59/39	59/57					
RQCES	005204	64/22	64/19	64/46					
RRESH	001302	27/13	27/38	27/45	27/52				
RRET	010242	98/23	98/35	98/41					
RSANS	001767	32/12	32/27	32/33	32/57	32/58			
RTEST	000152	14/41	171/22	108/23	109/22	117/46			
RTIME	002477	12/23	39/12						
RVTMP	003222	45/42	45/47	45/49	45/63	46/28			
RXAST	006732	84/11	84/12	84/18	84/27	84/21			
RXOEC	000454	16/39	16/58	16/62	19/63	22/21			
RXAST	006743	84/34	84/35	84/39	84/42	84/41			
RXPST	006775	85/12	85/13	85/14	85/22	85/23	85/24		
RXOST	007005	85/35	85/36	85/42	85/41	85/42			
RXAST	006757	84/51	84/52	84/62	84/61	84/62			
RZCTR	010273	95/29	99/17	99/18	99/24				
SABIN	001073	22/43	22/56	23/19					
SACMA	000773	21/22	21/23	21/31	22/14				
SADIG	000453	18/38	18/63	19/22	22/13				
SAMCO	002311	36/34	36/41	36/52					
SAMEX	002260	35/14	36/12						
SAMYS	002274	36/15	36/25	36/28					
SAMNM	002265	35/16	36/22						
SANS	007437	91/26	91/19	91/32					
SAPTB	002403	35/12	37/43						
SAVE	001577	32/23	32/39	31/25					
SAVED	010024	95/36	95/41	95/50					
SCORA	003152	45/24	45/59						
SCOREB	003155	45/32	45/34						
SCOREF	003146	44/34	45/22	46/41					
SCENV	006723	84/14							

2134 .MAIN

SDEV2	006737	84/37								
SDEV3	006753	84/58								
SDEV4	006767	85/18								
SDEV5	007001	85/38								
SOIST	001147	24/14	24/16	33/02						
SECS	001236	25/24	25/51	27/27	28/40	28/46	28/54	43/30	43/52	
		59/08								
SECAK	001243	25/42	25/57							
SECM2	001237	21/62	25/52	59/17	59/21	59/25				
SECM5	001240	25/45	25/53							
SECS2	001241	25/27	25/55							
SECS4	001242	25/20	25/56							
SESA1	003403	47/17	47/30							
SESA1	003370	47/19	47/27							
SESA4	003365	47/10	47/13	47/16						
SESEA	003402	47/25	47/29							
SESC0	003401	47/21	47/28							
SETAC	000574	20/23	27/58	59/50	82/02					
SETP2	006110	13/35	107/32							
SETP1	006111	13/36	101/04	131/12	101/18	101/44	102/03	102/10	102/17	
		102/28	102/40	102/48	103/03	103/15	103/27	103/43	103/55	
		104/02	104/13	104/24	104/38	104/52	105/03	105/16	105/34	
		105/51	106/03	106/16	106/30	106/43	107/03	109/04	109/12	
		109/16	109/23	109/30	109/41	109/50	110/03	110/14	112/24	
		110/34	110/48	111/05	111/40	111/54	112/03	112/19	112/32	
		112/49	113/03	113/20	113/36	113/51	114/03	114/11	114/27	
		114/42	115/03	115/18	117/04	117/17				
SETP2	006112	13/37								
SETSA	006544	81/00	82/03							
SIGN	000526	19/23	20/38							
SIGNR	000543	19/23	19/27	19/37	19/35	19/37				
SMASK	006760	84/55	84/63							
SMEND	002401	37/14	37/29	37/38						
SSA'S	001770	32/34	32/44	32/59						
SSIAC	001371	27/27	27/36	27/59						
SSAR0	001303	27/10	27/42	27/48	27/53					
SSAR1	001304	27/11	27/43	27/49	27/54					
SSAR2	001365	27/12	27/47	27/52	27/55					
STARU	006373	78/48	78/52	79/06	79/17	79/21				
START	001620	31/03	32/03							
STAT	011157	108/01	108/29	108/15	108/18					
STATA	006115	13/40								
STATN	006116	13/41								
STATP	006121	13/44								
STATS	006120	13/43								
STATA	006117	13/42								
STINC	002512	39/21	39/28							
STOP	002102	12/43								
STORE	001655	31/36	31/41							
STCTX	006314	78/08	78/11							
STSKP	002513	39/14	39/22	39/26						
STTYP	003006	43/14	43/17	43/32						
SUM	012150	117/26	117/34	117/35	117/48					
SVTYN	003223	44/22	46/09	46/22	47/29					
SAISA	004740	12/40	28/62	36/40	37/62	59/55	61/53	63/48	87/61	
SAPUN	010106	90/33	97/02	97/27	97/28					
SXAS1	006733	84/15	84/19	84/22						
TABLE	002200	24/24	33/26							
TRZOT	002777	21/49	22/18							

0135 MAIN

TCP0	002575	40/36							
TCP00	002574	40/34	41/36						
TCP1	002576	40/37							
TCP10	002605	40/44							
TCP11	002606	40/45							
TCP12	002607	40/46							
TCP13	002610	40/47							
TCP14	002611	40/48							
TCP15	002612	40/49							
TCP16	002613	40/50							
TCP17	002614	40/51							
TCP2	002577	40/38							
TCP20	002615	40/52							
TCP21	002616	40/53							
TCP22	002617	40/54							
TCP23	002620	40/55							
TCP24	002621	40/56							
TCP25	002622	40/57							
TCP3	002600	40/39							
TCP4	002601	40/40							
TCP5	002602	40/41							
TCP6	002603	40/42							
TCP7	002604	40/43							
TEKPA	000130	14/06	92/22	97/11					
TEMP1	001575	30/12	30/13	30/37					
TEMP2	001576	30/22	30/30	30/36	30/38	31/25			
TERB1	000156	70/30	70/32						
TERDC	005520	70/04	70/00						
TERM1	005526	70/09	70/12						
TERMB	000152	75/11	70/20						
TERMD	005514	69/12	70/22						
TERMO	000043	73/11	74/33						
TERPS	005722	72/11	72/58						
TERPT	005407	68/23	68/49	69/11	73/47	78/27			
TERMX	000341	77/11	78/35						
TEROK	000047	74/32	74/34						
TERSC	005726	72/60	72/62						
TERTX	000345	78/37	78/39						
TEXEN	000303	77/50							
TEXIA	000231	77/50	78/16	78/53	79/07				
TEXIN	000232	77/51							
TIMCT	002571	40/30	41/29	41/13					
TIMEN	002572	40/31	41/15	46/04	117/52				
TIMEX	002570	40/26	40/27	40/03	41/10				
TIMMS	000064	13/22	44/16	107/14	107/43	117/28			
TIMRO	000065	13/23							
TIMSK	000063	13/21	18/20	18/30	23/45	24/13	101/20	102/21	102/33
		102/54	103/08	103/17	103/29	103/36	103/45	103/57	104/04
		104/15	104/27	104/42	105/05	105/23	105/39	105/53	106/05
		106/10	106/32	107/09	107/38	109/54	110/07	110/41	110/55
		111/09	111/26	111/44	111/58	112/07	112/23	112/40	112/58
		113/11	113/27	113/42	113/56	114/15	114/31	117/09	
TINH1	001164	22/46	22/61	23/32	24/05	24/28	24/32	25/16	25/37
TLOA	011225	108/27	108/42						
TMEAD	002340	29/00	36/14	37/05					
TOK	011240	108/36	108/44						
TOYA	003314	46/37	46/56						
TOYB	003337	46/47	46/57						
TOYC	003335	46/55	47/00						

ICMD	003327	46/45	46/48							
ICMEX	0033275	43/35	46/22							
ICMF	003331	46/50	46/53							
ICMG	003341	46/59	46/63							
ICMH	003342	46/57	46/62							
ICMJ	003332	46/48	46/51							
IRAN	007746	94/43	96/03	96/05						
IRCEX	007143	87/33	87/45	87/57	87/60					
IRCHA	007105	35/22	87/26							
IRCHC	007074	86/42	87/32							
IRCHL	007112	87/31	87/35							
IRCLC	007075	86/43	87/42							
IRCLL	007124	87/43	87/47							
IRCLC	007117	35/24	87/38							
IRCFD	007076	86/44	87/54							
IRCFE	007131	35/26	87/52							
IRCRD	007136	87/55	87/59							
IRGTA	007073	86/41	87/28	87/40	87/52					
IRGTE	007145	86/41	88/04							
IRTFP	002702	43/12	43/19							
IXCCU	006330	68/61	77/08	77/44	78/05	78/06	78/14	78/24	78/39	
		78/46	79/22							
IXEND	006401	68/02	78/57	79/13						
IXADR	006412	79/13	79/21	79/22						
IYLIM	003217	44/23	46/05							
IYMA	003065	44/30	44/49							
IYMB	003111	44/41	44/52							
IYMC	003107	44/48	44/57							
IYMD	003101	44/39	44/42							
IYME	003233	46/14	46/17							
IYMEH	003216	44/19	46/04							
IYMEN	003214	44/24	45/48	46/02	46/24					
IYMER	003031	43/12	44/02							
IYMF	003103	44/44	44/47							
IYMG	003113	44/52	44/56							
IYMH	003114	44/50	44/53							
IYMJ	003104	44/42	44/45							
TYPE1	000742	21/52	23/16							
TYPE2	000744	21/54	23/17							
TYPE3	000753	21/57	21/61							
TYPE4	000756	21/60	22/07							
TYPE5	000760	22/42								
TYPIN	005357	68/19	68/22	68/31	68/33	69/23	73/49	78/29		
TYPNX	000474	18/55	18/58	19/45	19/55					
TYPRE	005305	68/22	68/24	68/27	68/28					
TYRTC	003215	46/03	46/23							
TYTTF	003221	44/23	46/07							
TYTTS	003220	44/21	46/06							
ULPTT	002256	35/02	37/55							
UMEND	002366	37/18	37/21	37/27						
UNTIM	001536	29/17	43/23	43/25						
UNTST	002733	42/37	43/27							
UMEND	002402	37/35	37/36	37/39						
AACSA	001311	26/18	26/45	26/51						
AAREF	002445	38/13	38/14	38/26	38/27					
AATOP	006062	13/20								
ARZOT	001367	27/35	27/57							
ACHH4	001310	26/39	26/57							
CHIGH	001307	26/17	26/49							

2138 MAIN

SCIS	021777	32/41	33/02							
STAA	026720	12/55	84/11							
STAC	021373	27/26	27/29	27/37	27/39	27/58				
STAN	026734	12/56	84/34							
STAP	026761	12/59	85/12							
STAS	026776	12/58	85/35							
STAA	026744	12/57	84/51							
TBIN	021014	12/13	22/33							
TDEC	020473	12/15	18/53							
TIMI	022730	42/31	42/33							
TINA	022716	42/19	42/23							
TINC	022731	42/13	42/21	42/27	42/34					
TIND	022715	42/16	42/22	42/32						
TIMR	022732	42/11	42/14	42/15	42/17	42/18	42/20	42/25	42/26	
		42/29	42/35							
TIPS	022702	12/25	42/11							
TIMT	022722	42/24	42/27							
TIMW	022725	42/28	42/30							
TCCO	020560	12/14	19/53							
TCCO	025266	61/27	61/57	63/31						
TRMT	025777	72/10	73/10	73/47	75/10					
TTI	020010	14/24	26/21	26/23	26/26	67/34	67/37	67/39		
TTTO	020011	14/25	18/32	22/00	22/01	44/10	44/11	44/12	44/14	
		44/15	44/17	44/26	44/27	44/28	44/35	44/43	44/46	
		44/51	44/55							
TTPT	023445	18/31	18/33	33/00						
TTXCO	025422	68/36	68/61							
TTXND	025423	68/37	68/62							
TYME	023230	44/03	45/24	45/41	46/14	46/25				
TYPE	024724	12/06	21/37							
TYPN	026001	72/20	73/22	73/49	75/22					
TZOC	020550	12/16	19/43							
VAIT	022427	12/21	38/13							
VATP	021244	12/22	26/14							
VATP	023012	43/20	43/36							
VX10	023226	45/27	45/33	45/46	45/56	46/12				
VXLP	021774	32/36	32/63							
VXTTO	021775	32/38	33/00							
VCHAR	022672	18/41	18/59	21/09	22/35	23/01	23/02			
VDICL	021172	24/36	25/32							
VDLTE	020633	77/12	78/28							
VDOUT	021142	18/42	22/40	23/53	24/09	24/38				
VGTCN	020633	77/09	78/26							
VHAAT	021226	25/18	25/41							
VILLG	020633	77/31	77/47	78/30						
VINRT	020633	77/04	78/31							
VINTR	020633	78/32	78/55							
VONTR	020634	78/33	78/36							
VPBIN	021024	22/34	22/39	22/43						
VPDEC	020502	18/49	18/54	19/02						
VPCCO	020623	19/54	19/59	20/12						
VTAB3	022712	18/43	21/26							
VTRMT	020633	77/10	78/27							
VTYPE	022732	21/22	21/34	21/44						
VTYPN	020633	77/20	78/29							
VZCOT	020601	19/44	19/49	20/09						
VCHAR	020456	18/23	18/41							
VDCOT	020457	18/42	20/00							
VSUPP	020771	20/20	20/26	20/36	22/12					

0139 .MAIN

!TAB3 030460 18/43 18/62

