A/S REGNECENTRALEN
SCANDINAVIAN INFORMATION PROCESSING SYSTEMS

SYSTEM LIBRARY

RCSL NO: 55-D52
AUTHOR : Tom Sandvang
EDITION: Sep. 1969

RC 4000 PERIPHERAL DEVICES

RC 150 PAPER TAPE PUNCH
REFERENCE MANUAL

ABSTRACT

This report describes the logical structure of the RC 150 paper tape punch when used in connection with the RC 4000 computer.

CONTENTS

INFORMATION DEPARTMENT
DK-2500 VALBY · BJERREGAARDSVEJ 5 · PHONE: (01) 46 08 88 · TELEX: 6464 reinf dk · CABLES: INFOCENTRALEN

MAIN CHARACTERISTICS

The paper tape punch is connected to the low-speed data channel by means of a buffer register of 24 bits.

It punches paper tape with a maximum speed of 150 characters per second.

The paper tape punch is supplied with a local/remote switch which can be set manually. In the remote state the device is program controlled. In the local state the operator may insert or adjust a roll of paper tape.

PAPER TAPE SPECIFICATION

The paper tape punch accepts paper tape of the following types:

8 tracks (one inch tape)
7 tracks (seven-eights inch tape)
5 tracks (eleven-sixteenths inch tape)

In the internal representation of a character the tracks are identified by the track identifiers b1 through b8 each track having the weight as shown:

track identifier	b8	b7	b6	b5	b4	b3	b2	b1
weight	128	64	32	16	8	4	2	1

A bit value of one corresponds to a punched hole in the corresponding track.

COMMANDS

The paper tape punch accepts sense and write commands, the latter with three modifications.

In the input/output instruction specifying the sense command the value of the modifier field i.e. bit 18 - 21 in the effective address is irrelevant.

The use of read commands, control commands and other modifications of the write command than specified have no effect at all i.e. the device is then considered disconnected.

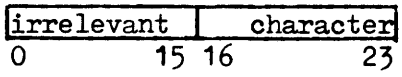
WRITE COMMANDS

The following write commands are available:

3	write odd
7	write even
11	write general

The integers denote the values of bits 18 - 23 in the effective address of the input/output instruction.

A write command initiates the output of one character from a working register. The contents of the working register is interpreted as follows:



After the initiation of a write command the punch is busy until the operation is completed successfully or terminated by an error condition. ~~The punch delivers an interrupt signal when it becomes available.~~

The write odd and write even commands transfer a 7-bit character:

0 b7 b6 b5 b4 b3 b2 b1

and inserts a bit b8 in order to make the parity of bits b8 - b1 odd and even respectively.

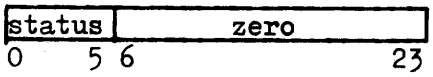
The write general command transfers an 8-bit character:

b8 b7 b6 b5 b4 b3 b2 b1

which is punched out directly.

SENSE COMMAND

When the paper tape punch is available a status word can be transferred from the buffer register to a working register by means of a sense command. The status word consists of 6 status bits:



The status bits have the following meaning:

0 intervention
1 (not used = 0)
2 timer
3 (not used = 0)
4 (not used = 0)
5 end of tape

Intervention: The intervention status indicates that the operator has interfered with the device by means of the local/remote switch.

Timer: The timer status indicates the termination of a write operation that lasts longer than approximately 1 second. The cause may be either a mechanical fault in the tape feeding and punching mechanism or that the tape reel is empty or blocked.

End of tape: The end of tape status indicates that the supply of paper tape is running low. The punch is adjusted so it is possible to punch at least 4000 characters more. Subsequent write operations are accepted and completed if possible, but will always result in the status indication end-of-tape. If more than 4000 characters are output after the first end-of-tape indication the timer status will sooner or later be set together with the end-of-tape status when the tape reel actually becomes empty. The end-of-tape status bit is cleared either by a write operation or by pressing the tape feed lever when the paper tape has been changed.

LOCAL/REMOTE SWITCH

If the operator switches to local during a write operation the transition to the local state is delayed until the operation has been completed.

If a write command is initiated in the local state the punch accepts the command and becomes busy. The actual operation, however, is delayed until the operator switches to remote control.

When the punch becomes local the intervention status bit is set. It can only be cleared by a write operation which is initiated in the remote state (and not when the write operation is initiated in the local state).

The local and the remote state is set by the switch on the operator control panel.

Blank tape (0 holes) can be produced in the local state by means of the tape feed lever and therefore the local state will be set automatically as long as the tape feed lever is activated if not already set from the control panel.

A/S REGNECENTRALEN
SCANDINAVIAN INFORMATION PROCESSING SYSTEMS
SYSTEM LIBRARY
RCSL NO: 55-D51
AUTHOR : Tom Sandvang
EDITION: Sep. 1969

RC 4000 PERIPHERAL DEVICES

RC 2000 PAPER TAPE READER
REFERENCE MANUAL

ABSTRACT

This report describes the logical structure of the RC 2000 paper tape reader when used in connection with the RC 4000 computer.

CONTENTS

INFORMATION DEPARTMENT
DK-2500 VALBY · BJERREGAARDSVEJ 5 · PHONE: (01) 46 08 88 · TELEX: 6464 rcnl dk · CABLES: INFOCENTRALEN

MAIN CHARACTERISTICS

The paper tape reader is connected to the low-speed data channel by means of a buffer register of 24 bits.

The reader is supplied with an internal buffer with a capacity of 256 characters of 8 bits each. The internal buffer is automatically filled by means of a serve-controlled motor with a maximum speed of 2000 characters/second. The reader produces an interrupt signal when one of the two following conditions are fulfilled:

1: The number of characters in the internal buffer exceeds 64.

2: The end of the paper tape has been reached and there is less than 65 characters in the internal buffer.

Characters may now be transferred from the internal buffer through the buffer register to a working register by means of read and sense commands.

When the internal buffer is less then half-full the motor is started again.

PAPER TAPE SPECIFICATIONS

The paper tape reader accepts paper tape of the following types:

8 tracks (one inch tape)
7 tracks (seven-eights inch tape)
6 tracks (square holes)
5 tracks (eleven-sixteenths inch tape)

In the internal representation of a character the tracks are identified by the track identifiers b8 through b1 each having the weight as shown:

track identifier	b8	b7	b6	b5	b4	b3	b2	b1
weight	128	64	32	16	8	4	2	1

A bit value of one corresponds to a punched hole in the corresponding track.

COMMANDS

The paper tape reader accepts sense and read commands, the latter with three modifications.

In the input/output instruction specifying the sense command the value of the modifier field i.e. bits 18 - 21 in the effective address is irrelevant.

The use of write commands, control commands and other modifications of the read command than specified have no effect at all i.e. the device is then considered disconnected.

READ COMMANDS

The following read commands are available:

2	read odd
6	read even
10	read general

The integers denote the values of bits 18 - 23 in the effective address of the input/output instruction.

A read command initiates the transfer of one character from the internal buffer to the buffer register. During this, the reader is busy for approximately 20 microseconds.

The read odd and read even commands transfer a 7-bit character:

0 b7 b6 b5 b4 b3 b2 b1

and sets a status bit if the parity of bits b8 - b1 is even and odd, respectively.

The read general command transfers an 8-bit character:

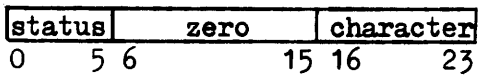
b8 b7 b6 b5 b4 b3 b2 b1

and sets a status bit if the parity of bits b8 - b1 is even.

Blank tape (0 holes) is skipped automatically in read operations.

SENSE COMMAND

When the reader is available a status word can be transferred from the buffer register to a working register by means of a sense command. The rightmost 8 bits of the status word contain the last character received in the buffer register, whereas the leftmost 6 bits are status bits:



The status bits have the following meaning:

0 end of buffer
1 parity
2 (not used = 0)
3 (not used = 0)
4 (not used = 0)
5 end of tape

End of buffer: If the internal buffer is empty and paper tape is still present the reader always generates a character equal to zero and the status bit end-of-buffer.

Parity: The meaning of the parity indication has already been explained under READ COMMANDS.

End of tape: If the internal buffer is empty and no paper tape is present in the reader a character equal to zero and the status bit end-of-tape are always generated.

OPERATOR PUSH-BUTTONS

The paper tape reader is supplied with four push-buttons called reset, read, skip and up.

Reset: The reset button immediately causes the internal buffer to be cleared. If paper tape is present the buffer will now be refilled with 100 - 200 characters.

Read: The read button does not change the previous contents of the internal buffer. It merely causes the motor to start if both paper tape is present and the buffer is less than half-full.

Skip: The skip button causes the motor to skip characters on the paper tape without changing the contents of the internal buffer.

Up: The up button opens the pressure lid. The motor now remains stopped until the pressure lid is closed and the reset or the read button is pressed.

RC 2000 PAPER TAPE READER

```
B.

S.

W.      <:RC 2000 PAPER TAPE READER<0>:>

E.
```

```
S. B1000, E15                    ;  TEST RC 2000 1.2  JL  1/588 E

W.        <: 1.2 READ TEST TAPE<0>                          :>

          RS.   W2          B845. ;  (W2) = RETURN ADDRESS
          SO    W1            1
          JL.               B380.
          AL.   W0          B760. ;  RUN NO. 0 (INITIATE)
          AM                 (16) ;  TABLE 1 (4) =
          RS    W0            +6  ;  SUB - INTERRUPT
B200:     AL.   W0          B940. ;  MOUNT TEST TAPE
          AM                 (18)
          JL    W2           (+0)
          AM                 (18)
          JL    W2           (+6)
          SE    W2            10
          JL.               B200. ;  WAIT FOR NL
          AL    W0             0
          RS.   W0          B847. ;  FLAG = 0
          AM                 (22)
          RL    W1            +0
          RS.   W1          B841. ;  STORE DEV. NO. < 6
          RL.   W1          B838.
          AM                 (22)
          LS    W1           (+2) ;  SHIFT  - CHANNEL NO.
          MS                  0
          LO    W1            0
          RS.   W1          B837. ;  STORE INT. MASK
          JL.               B755.

B380:     MS.               B846. ;  STORE INT. MASK
          ML.               B837. ;  LOAD  INT. MASK
          AL    W1            0
          RS.   W1          B827. ;  FOR L := 0, 1, 2 DO
B390:     LS    W1             2  ;  BEGIN L
          AL    W1    X1       2
          WA.   W1          B841.
          RS.   W1          B840.
          AL    W0             1
          RS.   W0          B828. ;    FOR I := 1, -1 DO
B400:     AL    W0            -9  ;    BEGIN I
          RS.   W0          B829. ;      FOR J := -9 STEP 1 UNTIL 9 DO
B410:     RL.   W3          B828. ;      BEGIN J
          WM.   W3          B829.
          AL    W3    X3       9  ;          CHARSET :=
          LS    W3             1
          RL.   W3    X3    B960. ;             TABLE(10 + I*J)
          AM.               (B828.)
          SE    W1    X1      -1  ;          IF I = -1 THEN
          AC    W3    X3       1  ;          CHARSET := COMPL(CHARSET)
```

```
B430:  AL    W0          1  ;          FOR K := 1 STEP 1 UNTIL 8 DO
       RS.   W0       B830. ;          BEGIN K
B440:  AL    W1          1
       AM.             (B827.)
       SN    W2   X2    -1  ;            CNT :=
       AL    W1          0  ;            IF PAR. = 1 THEN 0
       AL    W0          1  ;                        ELSE 1
       AL    W2   X3     0  ;            CHAR :=
       LA.   W2       B870. ;            CHARSET & 8.377
B460:  SZ    W2         (0)
       AL    W1   X1     1  ;            CNT := CNT + BITNO.
       LS    W0          1
       SE    W0        256
       JL.            B460.
B470:  LS    W1         22  ;            PAR. BIT := CNT < 22
       WA    W2          2  ;            CHAR := CHAR + PAR. BIT
       RL.   W0       B827.
       LS    W0         -1
       LS    W0          7  ;            CHAR(16) :=
       WA.   W0       B876. ;            IF PAR <> 2 THEN 0
       LA    W2          0
       RS.   W2       B835.
B500:  IO.           (B840.);            READ
       JL.   W2         E0. ;            SENSE
       SX               3
       JL.            B500.
       SE.   W2       (B838.)
       JL.            B590.
       AL    W0          0  ;  END OF BUFFER
B560:  AM.           (B847.);                                  4.0 USEC
       SN    W1   X1    -1  ;  IF FLAG THEN                    3.5 USEC
       JL.            B570. ;     GOTO B570 ELSE
       SL.   W0       (B836.);  IF CNT > LIM THEN             4.5 USEC
       JL.            B565. ;     GOTO B565
       BA.   W0          1  ;  CNT := CNT + 1                 4.0 USEC
       JL.            B560. ;  GOTO B560                      3.0 USEC
B565:  AL.   W0       B950. ;  NO INTERRUPT FOR 1 SEC
       AM              (18)
       JL    W2        (+0)
B570:  AL    W1          0  ;            FLAG := FALSE
       RS.   W1       B847.
       JL.            B500.
B590:  SN.   W2       (B875.)
       JL.            B730. ;            GO TO END OF TAPE
       SE.   W2       (B835.)
       JL.            B710.
B650:  LS    W3         -1  ;            CHARSET := CHARSET > 1
       RL.   W1       B830.
       SL    W1          8
       JL.            B660.
       AL    W1   X1     1
       RS.   W1       B830.
       JL.            B440. ;          END K
```

```
B660: RL.   W1        B829.
      SL    W1          9
      JL.             B670.
      AL    W1    X1    1
      RS.   W1        B829.
      JL.             B410. ;       END J
B670: RL.   W1        B828.
      SE    W1          1
      JL.             B680.
      AL    W1         -1
      RS.   W1        B828.
      JL.             B400. ;     END I
B680: RL.   W1        B827.
      SN    W1          2
      JL.             B750.
      AL    W1    X1    1
      RS.   W1        B827.
      JL.             B390. ;   END L

B710: RS.   W2        B834. ;   READ ERROR
      RL.   W1        B827.
      LS    W1          3
      AL.   W0    X1  B930.
      AM                (18)
      JL    W2          (+0)
      AL.   W0        B955. ;   READ CHAR.
      AM                (18)
      JL    W2          (+0)
      AL.   W0        B834.
      AL    W1          24
      AM                (18)
      JL    W2          (+4)
      AL.   W0        B956. ;   EXPECTED CHAR.
      AM                (18)
      JL    W2          (+0)
      AL.   W0        B835.
      AL    W1          24
      AM                (18)
      JL    W2          (+4)
      JL.             B650.

B730: AL.   W0        B920. ;   END OF TAPE
      AM                (18)
      JL    W2          (+0)
      AM                (18)
      JL    W2          (+6)
      SE    W2          10
      JL.             B730. ;   WAIT FOR NL

B750: ML.             B846. ;   RELOAD INT. MASK
B755: JL.            (B845.);   JUMP TO DIRECTORY
```

```
B760: AL    W1          1  ;  SUB - INTERRUPT
      RS.   W1       B847. ;  FLAG =TRUE
      JL         X2     0  ;  RETURN

;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

  E0: RS.   W0          E8.
      RS.   W1          E9.
      RS.   W2         E10.
      AL    W0          0  ;  COUNTER = 0
      AL    W1          0
  E1: IO.   W2       (B841.);  SENSE                      6.0 USEC.
      RS.   W2         E11. ;  (E11) = STATUS             4.5 USEC.
      XS                3   ;                             4.0 USEC.
      LA.   W1         -2   ;                             3.5 USEC.
      LS    W1          1   ;  (W1) = 2*(EX.- REG.)       3.5 USEC.
      JL.         X1    2   ;                             4.0 USEC.
      JL.              E6.  ;  EX = 00
      JL.              E4.  ;                             3.0 USEC.
      JL.              E3.
  E2: AL.   W0         E13. ;  EX = 11
      JL.              E5.
  E3: AL.   W0         E12. ;  EX = 10
      JL.              E5.
  E4: BA.   W0          1   ;  EX = 01                    4.0 USEC.
      SH.   W0       (E7.);  CNT = CNT + 1               4.5 USEC.
      JL.              E1.  ;  SENSE AGAIN               3.0 USEC.
      AL.   W0         E15.
  E5: AM               (18)
      JL    W2         (+0)
  E6: RL.   W0          E8.
      LS    W1          -1
      XL                3
      RL.   W1          E9.
      RL.   W2         E11. ;  (W2) = STATUS
      JL.             (E10.);  RETURN

  E7:                 2500  ;  LIMIT (25000 COUNTS PER SEC.)
  E8:                    0  ;  W0
  E9:                    0  ;  W1
 E10:                    0  ;  W2  (RETURN - ADDRESS)
 E11:                    0  ;  STATUS

 E12: <:<10>DEVICE DISCONNECTED<0>:>
 E13: <:<10>EXCEPTION-REG.=811<0>:>
 E15: <:<10>DEVICE BUSY FOR 0.1 SEC.<0>:>

B827:                    0  ;  L  PARITY = 0, 1, 2
B828:                    0  ;  I  1 (NON-COMPL.) OR -1 (COMPL.)
B829:                    0  ;  J  PATTERN NO. (-9, -8, ... ,9)
B830:                    0  ;  K  SHIFTCOUNTER
```

```
B834:                    0  ;  READ CHAR.
B835:                    0  ;  EXP. CHAR.
B836:                52600  ;  1 SEC LIMIT
B837:                    0  ;  INT. MASK OF TESTPROGRAM
B838:          8.40000000   ;  BIT 0 = 1
B839:          8.04000000   ;  BIT 3 = 1
B840:                    0  ;  DEV<6 + PAR<2 + 2
B841:                    0  ;  DEV<6 + 0
B845:                    0  ;  RETURN ADDRESS
B846:                    0  ;  INT. MASK OF DIRECTORY
B847:                    0  ;  INTERRUPT-FLAG
B870:          8.00000377
B875:          8.01000000   ;  BIT 5 = 1
B876:          8.37777577   ;  BIT 0 AND 16 = 0

B920: <:<10>END OF TAPE<0>:>
B930: <:<10>ODD PAR.<0>  <10>EVEN PAR.<0> <10>GENERAL<0>:>
B940: <:<10>MOUNT TEST TAPE<0>:>
B950: <:<10>NO INTERRUPT FOR 1 SEC<0>:>
B955: <:<10>READ    <0>:>
B956: <:<10>EXPECTED<0>:>

B960: 2.10000000100000010000000
      2.11000000110000011000000
      2.10100000101000010100000
      2.10010000100100010010000
      2.10001000100010010001000
      2.11100000111000011100000
      2.10110000101100010110000
      2.11010000110100011010000
      2.10011000100110010011000
      2.11001000110010011001000
      2.10101000101010010101000
      2.10010100100101010010100
      2.11110000111100011110000
      2.10111000101110010111000
      2.11011000110110011011000
      2.10101100101011010101100
      2.11001100110011011001100
      2.10110100101101010110100
      2.10101010101010101010101

B.    B0
W.    B0:  I.                  0  ;  TYPE OUT LENGTH OF PROGRAM
E.

E.
```

```
;                                       CHAPTER          2.1
;                                       PROGRAM          2.1
;                                       PAGE               1


S. B1000, E15                       ;   TEST RC 2000 2.1  JL  1/588 E

W.        <: 2.1 READ ARB. TAPE<0>                              :>

          RS.   W2          B845. ;   (W2) = RETURN ADDRESS
          SO    W1            1
          JL.               B380.
    B200: AL.   W0          B890. ;   RUN NO. 0 (INITIATE)
          AM                 (18) ;   PARITY =
          JL    W2           (+0)
          AM                 (18)
          JL    W2           (+6) ;   (W1) = 0 OR E OR G
          AL    W1             0  ;   (W2) = PARITY
          SN    W2           111
          JL.               B230. ;   PARITY = 0
          SN    W2           101
          JL.               B220. ;   PARITY = 1
          SE    W2           103
          JL.               B200. ;   ERROR IN INPUT
          AL    W1    X1       4  ;   GENERAL
    B220: AL    W1    X1       4  ;   EVEN
    B230: RS.   W1          B846. ;   PARITY IS STORED
          AL.   W0    X1    B930. ;   OUTTEXT
          AM                 (18)
          JL    W2           (+0)
          AM                 (22)
          RL    W1            +0
          RS.   W1          B841. ;   STORE DEV. NO. < 6
          WA.   W1          B846.
          AL    W1    X1       2
          RS.   W1          B840.
          JL.               B755.

    B380: IO.               (B840.)
          JL.   W2            E0. ;   SENSE
          JL.               B380.

    B755: JL.               (B845.);  JUMP TO DIRECTORY
```

```
;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

    E0: RS.   W0          E8.
        RS.   W1          E9.
        RS.   W2         E10.
        AL    W0           0  ;  COUNTER = 0
        AL    W1           0
    E1: IO.   W2       (B841.);  SENSE                    6.0 USEC.
        RS.   W2         E11. ;  (E11) = STATUS           4.5 USEC.
        XS                3   ;                           4.0 USEC.
        LA.   W1         -2   ;                           3.5 USEC.
        LS    W1          1   ;  (W1) = 2*(EX.- REG.)     3.5 USEC.
        JL.       X1      2   ;                           4.0 USEC.
        JL.              E6.
        JL.              E4. ;                           3.0 USEC.
        JL.              E3.
    E2: AL.   W0         E13. ;  EX = 11
        JL.              E5.
    E3: AL.   W0         E12. ;  EX = 10
        JL.              E5.
    E4: BA.   W0          1   ;  EX = 01                 4.0 USEC.
        SH.   W0        (E7.);  CNT = CNT + 1            4.5 USEC.
        JL.              E1. ;  SENSE AGAIN             3.0 USEC.
        AL.   W0         E15.
    E5: AM               (18)
        JL    W2         (+0)
    E6: RL.   W0          E8.
        LS    W1          -1
        XL                3
        RL.   W1          E9.
        RL.   W2         E11. ;  (W2) = STATUS
        JL.             (E10.);  RETURN

    E7:                 2500  ;  LIMIT (25000 COUNTS PER SEC.)
    E8:                    0  ;  W0
    E9:                    0  ;  W1
   E10:                    0  ;  W2  (RETURN - ADDRESS)
   E11:                    0  ;  STATUS

   E12: <:<10>DEVICE DISCONNECTED<0>:>
   E13: <:<10>EXCEPTION-REG.=B11<0>:>
   E15: <:<10>DEVICE BUSY FOR 0.1 SEC.<0>:>
```

```
B840:                    0  ;  DEV<6 + PAR<2 + 2
B841:                    0  ;  DEV<6 + 0
B845:                    0  ;  RETURN ADDRESS
B846:                    0  ;  (PARITY = 0,1,2) < 2

B890: <:<10>PARITY = <0>:>
B930: <:DD<0>   VEN<0>  ENERAL<0>:>

B.    B0
W.    B0:  I.            0  ;  TYPE LENGTH OF PROGRAM
E.

E.
E.
```

RC 150 PAPER TAPE PUNCH

```
B.

S.

W.      <:RC 150 PAPER TAPE PUNCH<0>:>

E.
```

```
S. B1000, E15                    ;  TEST RC 150 1.2  JL  1/588 E

W.        <: 1.2 PUNCH AND READ<0>                          :>

          RS.   W2          B845. ;  (W2) = RETURN ADDRESS
          RS.   W1          B844.
          SO    W1            1
          JL.               B380.
          AL    W0              0  ;  INITIATE TEST
          RS.   W0          B831. ;  RUN NO. = 0 (READ )
          RS.   W0          B842. ;  RUN NO. = 0 (PUNCH)
          RS.   W0          B847. ;  FLAG = 0
          AL.   W0          B760.
          AM                 (16) ;  TABLE 1 (4) =
          RS    W0            +6  ;  SUB - INTERRUPT
          AL    W0             1
          RS.   W0          B834. ;  PUNCH = TRUE
          AM                 (22)
          RL    W1            +0
          RS.   W1          B841. ;  STORE DEV. NO. < 6
          RL.   W1          B838.
          AM                 (22)
          LS    W1            (+2) ;  SHIFT  - CHANNEL NO.
          MS                   0
          LO    W1             0
          RS.   W1          B837. ;  STORE INT. MASK
          JL.               B755.

B380:     MS.               B846. ;  STORE INT. MASK
          ML.               B837. ;  LOAD  INT. MASK
          RL.   W1          B842.
          AL    W1    X1        1
          RS.   W1          B842. ;  RUN NO. = RUN NO. + 1
B385:     AL    W0              1
          RS.   W0          B827. ;  FOR L := 1,  2 DO
B390:     AL    W0              1  ;  BEGIN L
          RS.   W0          B828. ;    FOR I := 1, -1 DO
B400:     AL    W0             -9  ;    BEGIN I
          RS.   W0          B829. ;      FOR J := -9 STEP 1 UNTIL 9 DO
B410:     RL.   W3          B828. ;      BEGIN J
          WM.   W3          B829.
          AL    W3    X3        9  ;          CHARSET :=
          LS    W3              1
          RL.   W3    X3    B960. ;             TABLE(10 + I*J)
          AM.               (B828.)
          SE    W1    X1       -1  ;          IF I = -1 THEN
          AC    W3    X3        1  ;          CHARSET := COMPL(CHARSET)
```

```
        RL.   W0          B827.
        SE    W0             1   ;  PARITY :=
        JL.                B411.
        AL    W1             2   ;          IF L = 1 THEN 2
        RS.   W1           B835.
        JL.                B420.
B411:   AL    W1             0
        AL    W0             1
B412:   SZ    W3           (0)
        AL    W1    X1       1   ;          ELSE IF BITNO = ODD
        LS    W0             1   ;            THEN 0
        SH    W0           128   ;          ELSE IF BITNO = EVEN
        JL.                B412. ;            THEN 1
B414:   AL    W1    X1       1
        LA.   W1           B833.
        RS.   W1           B835.
B420:   LS    W1             2
        WA.   W1           B841.
        AL    W1    X1       3
        RS.   W1           B840. ;          FOR K := 1 STEP 1 UNTIL 8 DO
B430:   AL    W0             1
        RS.   W0           B830. ;          BEGIN K
B440:   RL.   W0           B834. ;            CHAR := CHARSET
        SE    W0             0   ;            IF PUNCH = TRUE THEN
        JL.                B500. ;              PUNCH CHAR ELSE
        JL.                B600. ;              READ CHAR
B450:   LS    W3            -1   ;            CHARSET := CHARSET>1
        RL.   W1           B830.
        SL    W1             8
        JL.                B460.
        AL    W1    X1       1
        RS.   W1           B830.
        JL.                B440. ;          END K
B460:   RL.   W1           B829.
        SL    W1             9
        JL.                B470.
        AL    W1    X1       1
        RS.   W1           B829.
        JL.                B410. ;        END J
B470:   RL.   W1           B828.
        SE    W1             1
        JL.                B480.
        AL    W1            -1
        RS.   W1           B828.
        JL.                B400. ;      END I
```

```
B480: RL.   W1        B827.
      SE    W1          1
      JL.             B750.
      AL    W1          2
      RS.   W1        B827.
      JL.             B390. ;  END L

B500: IO.   W3       (B840.);  PUNCH BEGIN
B510: JL.   W2          E0. ;  SENSE
      SX                3
      JL.             B515.
      SN    W2          0
B515: JL.             B450. ;  STATUS .EQ. 0
      SE.   W2       (B836.)
      JL.             B530.
B520: AL.   W0          B920. ;  END OF TAPE
      AM                (18) ;  LOAD RC 150
      JL    W2          (+0)
      AM                (18)
      JL    W2          (+6)
      SE    W2          10
      JL.             B520. ;  WAIT FOR NL
      JL.             B515. ;  SENSE AGAIN
B530: RS.   W2          B848. ;  STATUS - ERROR
      AL.   W0          B926. ;  STATUS =
      AM                (18)
      JL    W2          (+0)
      AL.   W0          B848.
      AL    W1          24
      AM                (18)
      JL    W2          (+4)
      JL.             B515. ;  PUNCH END

B600: IO                0<6 + 10  ;  READ GENERAL BEGIN
      IO    W1          0<6 +  0  ;  (W1) = NEXT CHAR.
      SX                    3     ;  (W3) = EXPECTED CHAR.
      JL.                  -4
      SN.   W1          (B838.)
      JL.             B600. ;  END OF BUFFER
      WS    W1            6   ;  W1 - W3
      LA.   W1          B832.
      SN    W1            0
      JL.             B620.
      WA    W1            6   ;  ERROR
      LS    W1           16
      RS.   W1          B825. ;  STORE READ CHAR.
      AL    W1    X3      0
      LS    W1           16
      RS.   W1          B843. ;  STORE EXP. CHAR
      AL.   W0          B925. ;  RUN NO.
      AM                (18)
      JL    W2          (+0)
      AL.   W0          B831.
      AM                (18)
      JL    W2          (+2)
```

```
        AL    W1          6
        WM.   W1        B835.
        AL.   W0    X1  B950. ;  PUNCH MODE
        AM              (18)
        JL    W2        (+0)
        AL.   W0        B930. ;  READ
        AM              (18)
        JL    W2        (+0)
        AL.   W0        B825.
        AL    W1          8
        AM              (18)
        JL    W2        (+4)
        AL.   W0        B940. ;  EXPECTED
        AM              (18)
        JL    W2        (+0)
        AL.   W0        B843.
        AL    W1          8
        AM              (18)
        JL    W2        (+4)
B620:   JL.             B450. ;  READ GENERAL END

B750:   RL.   W1        B844.
        SO    W1          2
        JL.             B754. ;  GO TO NEXT RUN
        RL.   W1        B831. ;  READ PAPER TAPE
        SN.   W1       (B842.)
        JL.             B754.
        AL    W1    X1    1
        RS.   W1        B831.
        SE    W1          1
        JL.             B753.
        AL    W0          0
        RS.   W0        B834. ;  PUNCH = FALSE
B752:   AL.   W0        B915. ;  LOAD RC 2000
        AM              (18)
        JL    W2        (+0)
        AM              (18)
        JL    W2        (+6)
        SE    W2         10
        JL.             B752. ;  WAIT FOR NL
B753:   JL.             B385. ;  GO TO READ

B754:   ML.             B846. ;  RELOAD INT. MASK
B755:   JL.            (B845.); JUMP TO DIRECTORY

B760:   AL    W0          1   ;  INTERRUPT - SERVICE
        RS.   W0        B847. ;  FLAG = 1
        JL        X2      0   ;  RETURN
```

```
;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

    E0: RS.   W0          E8.
        RS.   W1          E9.
        RS.   W2         E10.
        AL    W0           0  ;  COUNTER = 0
        AL    W1           0
    E1: IO.   W2        (B841.); SENSE                    6.0 USEC.
        RS.   W2         E11. ;  (E11) = STATUS           4.5 USEC.
        XS                3   ;                           4.0 USEC.
        LA.   W1         -2   ;                           3.5 USEC.
        LS    W1          1   ;  (W1) = 2*(EX.- REG.)     3.5 USEC.
        JL.       X1      2   ;                           4.0 USEC.
        JL.              +8
        JL.              E4. ;                           3.0 USEC.
        JL.              E3.
        JL.              E2.
        RL.   W2        B847. ;  EX = 00
        AL.   W0         E14.
        SN    W2          0
        JL.              E5.
        AL    W2          0
        RS.   W2        B847.
        JL.              E6.
    E2: AL.   W0         E13. ;  EX = 11
        JL.              E5.
    E3: AL.   W0         E12. ;  EX = 10
        JL.              E5.
    E4: BA.   W0          1   ;  EX = 01                  4.0 USEC.
        SH.   W0         (E7.); CNT = CNT + 1             4.5 USEC.
        JL.              E1. ;  SENSE AGAIN               3.0 USEC.
        AL.   W0         E15.
    E5: AM               (18)
        JL    W2         (+0)
    E6: RL.   W0         E8.
        LS    W1         -1
        XL               3
        RL.   W1         E9.
        RL.   W2         E11. ;  (W2) = STATUS
        JL.             (E10.); RETURN

    E7:                 50000  ;  LIMIT (25000 COUNTS PER SEC.)
    E8:                     0  ;  W0
    E9:                     0  ;  W1
   E10:                     0  ;  W2  (RETURN - ADDRESS)
   E11:                     0  ;  STATUS

   E12: <:<10>DEVICE DISCONNECTED<0>:>
   E13: <:<10>EXCEPTION-REG.=B11<0>:>
   E14: <:<10>NO INTERRUPT FROM DEVICE<0>:>
   E15: <:<10>DEVICE BUSY FOR 2 SEC.<0>:>
```

```
B825:                    0  ;  READ CHAR.
B827:                    0  ;  L  SEQUENCE NO. = 1, 2
B828:                    0  ;  I  1 (NON-COMPL.) OR -1 (COMPL.)
B829:                    0  ;  J  PATTERN NO. (-9, -8, ... ,9)
B830:                    0  ;  K  SHIFTCOUNTER
B831:                    0  ;  RUN NO. (READ)
B832:                8.377
B833:                    1
B834:                    0  ;  1 (PUNCH), 0 (READ GEN.)
B835:                    0  ;  PARITY (0,1,2)
B836:           8.01000000  ;  BIT 5 = 1
B837:                    0  ;  INT. MASK OF TESTPROGRAM
B838:           8.40000000  ;  BIT 0 = 1
B839:           8.04000000  ;  BIT 3 = 1
B840:                    0  ;  DEV. NO. < 6 + PAR. < 2 + 3
B841:                    0  ;  DEV. NO. < 6 + 0
B842:                    0  ;  RUN NO. (PUNCH)
B843:                    0  ;  EXPECTED CHAR.
B844:                    0  ;  FIRST - LAST CALL
B845:                    0  ;  RETURN ADDRESS
B846:                    0  ;  INT. MASK OF DIRECTORY
B847:                    0  ;  INTERRUPT-FLAG
B848:                    0  ;  STATUS

B915: <:<10>LOAD RC 2000<0>:>
B920: <:<10>LOAD RC 150<0>:>
B925: <:<10>RUN NO.<0>:>
B926: <:<10>STATUS = <0>:>
B930: <: PARITY<10>READ    <0>:>
B940: <:<10>EXPECTED<0>:>
B950: <:<10>ODD<0>    <10>EVEN<0>   <10>GENERAL<0>:>
```

```
B960: 2.10000000100000010000000
      2.11000000110000011000000
      2.10100000101000010100000
      2.10010000100100010010000
      2.10001000010001000010001000
      2.11100000111000001110000000
      2.10110000101100001011000
      2.11010000110100001101000
      2.10011000100110001001100
      2.11001000110010001100100
      2.10101000101010001010100
      2.10010100100101001001010
      2.11110000111100001111000
      2.10111000101110001011100
      2.11011000110110001101100
      2.10101100101011001010110
      2.11001100110011001100110
      2.10110100101101001011010
      2.10101010101010101010101010

B.    B0
W.    B0:  I.                  0  ;  TYPE OUT LENGTH OF PROGRAM
E.

E.
```

```
S. B1000, E15                    ; TEST RC 150 1.3  JL  1/588 E
W.        <: 1.3 TIMER<0>                                  :>
          RS.   W2          B845. ; (W2) = RETURN ADDRESS
          RS.   W1          B844.
          SO    W1             1
          JL.               B380.
          AL    W0              0 ; RUN NO. 0 (INITIATE)
          RS.   W0          B847. ; FLAG = 0
          AL.   W0          B760.
          AM                 (16) ; TABLE1(4) =
          RS    W0            +6 ; SUB - INTERRUPT
          AM                 (22)
          RL    W1            +0
          RS.   W1          B841. ; STORE DEV. NO. < 6
          AL    W1    X1        3
          RS.   W1          B840.
          RL.   W1          B838.
          AM                 (22)
          LS    W1           (+2) ; SHIFT  - CHANNEL NO.
          MS                    0
          LO    W1              0
          RS.   W1          B837. ; STORE INT. MASK
B360:     AL.   W0          B930. ; DISCON. MOTOR POWER
          AM                 (18)
          JL    W2           (+0)
          AM                 (18)
          JL    W2           (+6)
          SE    W2             10
          JL.               B360. ; WAIT FOR NL
          JL.               B755.

B380:     MS.               B846. ; STORE INT. MASK
          ML.               B837. ; LOAD  INT. MASK
          AL    W0              0
          IO.   W0         (B840.); PUNCH IS INITIALIZED
          JL.   W2             E0. ; SENSE
          SX                    3
          JL.                  +4
          SN.   W2         (B831.)
          JL.                B500.
          RS.   W2           B832.
          AL.   W0           B940. ; STATUS - ERROR
          AM                 (18)
          JL    W2           (+0)
          AL.   W0           B832. ; <STATUS>
          AL    W1             24
          AM                 (18)
          JL    W2           (+4)
```

```
B500: RL.   W0          B834. ;  (W0) = NUMBER OF CYCLES
      AL    W3            0
      WD.   W0          B835. ;  (W0) = TIME IN MSEC
      RS.   W0          B833.
      AL.   W0          B920. ;  TIMER
      AM                (18)
      JL    W2          (+0)
      AL.   W0          B833. ;  <TIME>
      AM                (18)
      JL    W2          (+2)
      AL.   W0          B960. ;  MSEC
      AM                (18)
      JL    W2          (+0)

B750: RL.   W1          B844.
      SO    W1            2
      JL.               B754.
B751: AL.   W0          B970. ;  CON. MOTOR POWER
      AM                (18)
      JL    W2          (+0)
      AM                (18)
      JL    W2          (+6)
      SE    W2            10
      JL.               B751.

B754: ML.               B846. ;  RELOAD INT. MASK
B755: JL.              (B845.);  JUMP TO DIRECTORY

B760: AL    W0              1 ;  SUB - INTERRUPT
      RS.   W0          B847. ;  FLAG = 1
      JL          X2        0 ;  RETURN
```

```
;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

    E0: RS.   W0          E8.
        RS.   W1          E9.
        RS.   W2          E10.
        AL    W0           0   ;  COUNTER = 0
        AL    W1           0
    E1: IO.   W2         (B841.); SENSE                      6.0 USEC.
        RS.   W2          E11. ;  (E11) = STATUS             4.5 USEC.
        XS                 3   ;                             4.0 USEC.
        LA.   W1          -2   ;                             3.5 USEC.
        LS    W1           1   ;  (W1) = 2*(EX.- REG.)       3.5 USEC.
        JL.       X1       2   ;                             4.0 USEC.
        JL.                +8
        JL.                E4. ;                             3.0 USEC.
        JL.                E3.
        JL.                E2.
        RS.   W0          B834. ;  EX = 00
        RL.   W2          B847.
        AL.   W0          E14.
        SN    W2           0
        JL.               E5.
        AL    W2           0
        RS.   W2          B847.
        JL.               E6.
    E2: AL.   W0          E13. ;  EX = 11
        JL.               E5.
    E3: AL.   W0          E12. ;  EX = 10
        JL.               E5.
    E4: BA.   W0           1   ;  EX = 01                    4.0 USEC.
        SH.   W0          (E7.); CNT = CNT + 1               4.5 USEC.
        JL.               E1. ;  SENSE AGAIN                3.0 USEC.
        AL.   W0          E15.
    E5: AM                (18)
        JL    W2          (+0)
    E6: RL.   W0          E8.
        LS    W1          -1
        XL                 3
        RL.   W1          E9.
        RL.   W2          E11. ;  (W2) = STATUS
        JL.              (E10.); RETURN

    E7:                  50000  ;  LIMIT (25000 COUNTS PER SEC.)
    E8:                      0  ;  W0
    E9:                      0  ;  W1
   E10:                      0  ;  W2  (RETURN - ADDRESS)
   E11:                      0  ;  STATUS
```

```
E12: <:<10>DEVICE DISCONNECTED<0>:>
E13: <:<10>EXCEPTION-REG.=B11<0>:>
E14: <:<10>NO INTERRUPT FROM DEVICE<0>:>
E15: <:<10>DEVICE BUSY FOR 2 SEC.<0>:>

B831:              8.10000000  ;  BIT 2 = 1
B832:                       0  ;  STATUS
B833:                       0  ;  TIME, MEASURED
B834:                       0  ;  COUNTER
B835:                      25  ;  25 CYCLES PER MSEC
B837:                       0  ;  INT. MASK OF TESTPROGRAM
B838:              8.40000000  ;  BIT 0 = 1
B839:              8.04000000  ;  BIT 3 = 1
B840:                       0  ;  DEV. NO. < 6 + 3
B841:                       0  ;  DEV. NO. < 6 + 0
B844:                       0  ;  FIRST - LAST CALL
B845:                       0  ;  RETURN ADDRESS
B846:                       0  ;  INT. MASK OF DIRECTORY
B847:                       0  ;  INTERRUPT - FLAG

B920: <:<10>TIMER:<0>:>
B930: <:<10>DISCONNECT RC 150 MOTOR POWER<0>:>
B940: <:<10>STATUS = <0>:>
B960: <: MSEC<0>:>
B970: <:<10>CONNECT RC 150 MOTOR POWER<0>:>

B.    B0
W.    B0:  I.                   0  ;  TYPE OUT LENGTH OF PROGRAM
E.

E.
```

```
S. B1000 , E15                    ; TEST RC 150 1.4  JL  1/588 E

W.        <: 1.4 START-UP TIME<0>                              :>

          RS.   W2          B845. ; (W2) = RETURN ADDRESS
          SO    W1            1
          JL.               B380.
          AL.   W0          B760. ; RUN NO. 0 (INITIATE)
          AM                (16) ; TABLE 1(4) =
          RS    W0           +6  ; SUB - INTERRUPT
          AL    W0            0
          RS.   W0          B847. ; FLAG DOWN
          AM                (22)
          RL    W1           +0
          RS.   W1          B841. ; STORE DEV. NO. < 6
          AL    W1    X1      3
          RS.   W1          B840.
          RL.   W1          B838.
          AM                (22)
          LS    W1          (+2) ; SHIFT  - CHANNEL NO.
          MS                  0
          LO    W1            0
          RS.   W1          B837. ; STORE INT. MASK
          JL.               B755.

B380:     MS.               B846. ; STORE INT. MASK
          ML.               B837. ; LOAD  INT. MASK
          RL.   W1          B833. ; 4 SEC. WAITING-LOOP
B396:     AL    W1    X1      -1  ; 2.0 USEC
          SE    W1            0  ; 3.0 USEC
          JL.               B396. ; 3.0 USEC
          IO.   W1         (B840.); PUNCH IS INITIALIZED
          JL.   W2           E0. ; SENSE
          SX                  3
          JL.                +4
B500:     SN    W2            0
          JL.               B550. ; STATUS .EQ. 0
          SN.   W2         (B831.)
          JL.               B585. ; END OF TAPE
          RS.   W2          B832.
          AL.   W0          B940. ; STATUS - ERROR
          AM                (18)
          JL    W2          (+0)
          AL.   W0          B832. ; <STATUS>
          AL    W1            24
          AM                (18)
          JL    W2          (+4)
```

```
B550:  RL.   W0          B834. ;  (W0) = NUMBER OF CYCLES
       AL    W3            0
       WD.   W0          B835. ;  (W0) = TIME IN MSEC
       RS.   W0          B842.
       AL.   W0          B920. ;  START-UP TIME =
       AM                (18)
       JL    W2          (+0)
       AL.   W0          B842. ;  <TIME>
       AM                (18)
       JL    W2          (+2)
       AL.   W0          B960. ;  MSEC
       AM                (18)
       JL    W2          (+0)
       JL.               B750. ;  GO TO NEXT RUN
B585:  AL.   W0          B930. ;  LOAD RC 150
       AM                (18)
       JL    W2          (+0)
       AM                (18)
       JL    W2          (+6)
       SE    W2            10
       JL.               B585. ;  WAIT FOR NL

B750:  ML.               B846. ;  RELOAD INT. MASK
B755:  JL.              (B845.);  JUMP TO DIRECTORY

B760:  AL    W1            1   ;  SUB - INTERRUPT
       RS.   W1          B847. ;  FLAG = 1
       JL          X2      0   ;  RETURN
```

```
;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

    E0: RS.   W0          E8.
        RS.   W1          E9.
        RS.   W2         E10.
        AL    W0           0  ;  COUNTER = 0
        AL    W1           0
    E1: IO.   W2        (B841.); SENSE                    6.0 USEC.
        RS.   W2          E11. ;  (E11) = STATUS          4.5 USEC.
        XS                 3   ;                          4.0 USEC.
        LA.   W1          -2   ;                          3.5 USEC.
        LS    W1           1   ;  (W1) = 2*(EX.- REG.)    3.5 USEC.
        JL.       X1       2   ;                          4.0 USEC.
        JL.               +8
        JL.                E4. ;                         3.0 USEC.
        JL.                E3.
        JL.                E2.
        RS.   W0          B834. ;  EX = 00
        RL.   W2          B847.
        AL.   W0           E14.
        SN    W2           0
        JL.                E5.
        AL    W2           0
        RS.   W2          B847.
        JL.                E6.
    E2: AL.   W0           E13. ;  EX = 11
        JL.                E5.
    E3: AL.   W0           E12. ;  EX = 10
        JL.                E5.
    E4: BA.   W0            1   ;  EX = 01                4.0 USEC.
        SH.   W0          (E7.); CNT = CNT + 1            4.5 USEC.
        JL.                E1. ;  SENSE AGAIN             3.0 USEC.
        AL.   W0           E15.
    E5: AM                (18)
        JL    W2          (+0)
    E6: RL.   W0           E8.
        LS    W1           -1
        XL                  3
        RL.   W1           E9.
        RL.   W2           E11. ;  (W2) = STATUS
        JL.               (E10.); RETURN

    E7:                  25000  ;  LIMIT (25000 COUNTS PER SEC.)
    E8:                      0  ;  W0
    E9:                      0  ;  W1
   E10:                      0  ;  W2  (RETURN - ADDRESS)
   E11:                      0  ;  STATUS
```

```
 E12: <:<10>DEVICE DISCONNECTED<0>:>
 E13: <:<10>EXCEPTION-REG.=B11<0>:>
 E14: <:<10>NO INTERRUPT FROM DEVICE<0>:>
 E15: <:<10>DEVICE BUSY FOR 1 SEC.<0>:>

B831:              8.01000000  ;  BIT 5 = 1
B832:                       0  ;  STATUS
B833:                  500000  ;  NUMB. OF CYCLES FOR 4 SEC
B834:                       0  ;  COUNTER
B835:                      25  ;  25 CYCLES PER MSEC
B837:                       0  ;  INT. MASK OF TESTPROGRAM
B838:              8.40000000  ;  BIT 0 = 1
B839:              8.04000000  ;  BIT 3 = 1
B840:                       0  ;  DEV. NO. < 6 + 3
B841:                       0  ;  DEV. NO. < 6 + 0
B842:                       0  ;  MEASURED TIME
B845:                       0  ;  RETURN ADDRESS
B846:                       0  ;  INT. MASK OF DIRECTORY
B847:                       0  ;  INTERRUPT-FLAG

B920: <:<10>START-UP TIME = <0>:>
B930: <:<10>LOAD RC 150<0>:>
B940: <:<10>STATUS = <0>:>
B960: <: MSEC<0>:>


B.  B0
W.  B0:  I.                 0 ;  TYPE OUT LENGTH OF PROGRAM
E.


E.
```

```
S. B1000, E15                    ;  TEST RC 150 2.1  JL  1/588 E

W.        <: 2.1 PUNCH SEQUENCE<0>                          :>

          RS.   W2          B845. ;  (W2) = RETURN ADDRESS
          SO    W1            1
          JL.               B380.
          AM                 (22) ;  RUN NO. 0 (INITIATE)
          RL    W1            +0
          RS.   W1          B841. ;  STORE DEV. NO. < 6
B260:     AL.   W0          B920. ;  SEQUENCE =
          AM                 (18)
          JL    W2           (+0)
          AL    W1             0  ;  NO := 0
B265:     AM                 (18)
          JL    W2           (+8) ;  READ(DEC, TERM)
          HS.   W0    X1    B848. ;  TABLE(NO) := DEC
          SH    W0           -1  ;  IF DEC < 0
          JL.               B260. ;     THEN GOTO B260
          SN    W2            10  ;  IF TERM = NL
          JL.               B280. ;     THEN GOTO B280 ELSE
          SE    W2            44  ;  IF TERM = , THEN GOTO B270
          JL.               B260. ;                 ELSE GOTO B260
B270:     SL    W1            49  ;  IF NO = 49
          JL.               B280. ;     THEN GOTO B280
          AL    W1    X1       1  ;  NO := NO + 1
          JL.               B265. ;  GOTO B265
B280:     RS.   W1          B834. ;  NUM := NO
B285:     AL.   W0          B940. ;  PARITY =
          AM                 (18)
          JL    W2           (+0)
          AM                 (18)
          JL    W2           (+6) ;  (W1) = 0 OR E OR G
          AL    W1             0  ;  (W2) = PARITY
          SN    W2            111
          JL.               B293. ;  PARITY = 0
          SN    W2            101
          JL.               B292. ;  PARITY = 1
          SE    W2            103
          JL.               B285.
          AL    W1    X1       1  ;  GENERAL
B292:     AL    W1    X1       1  ;  EVEN
B293:     LS    W1             2  ;  ODD
          AL    W0    X1       0  ;  (W0) = PARITY<2 +
          WA.   W0          B841. ;               DEV<6 +
          WA.   W0          B832. ;                   3
          RS.   W0          B840.
```

```
        AL.   W0    X1    B950. ; OUTTEXT
        AM                (18)
        JL    W2          (+0)
        JL.               B755.

B380:   AL    W1              0  ; (W1) = INDEX
B390:   HL.   W0    X1    B848. ; (W0) = NEXT CHAR.
        IO.   W0          (B840.); PUNCH
        JL.   W2            E0. ; SENSE
        SO.   W2          (B831.)
        JL.               B450. ; TAPE OK
B400:   AL.   W0            B930. ; LOAD RC 150
        AM                (18)
        JL    W2          (+0)
        AM                (18)
        JL    W2          (+6)
        SE    W2            10
        JL.               B400. ; WAIT FOR NL
B450:   SL.   W1          (B834.)
        JL.               B755. ; GO TO NEXT RUN
        AL    W1    X1        1
        JL.               B390. ; GO TO NEXT CHAR.

B755:   JL.               (B845.); JUMP TO DIRECTORY
```

```
;  THE FOLLOWING SENSE-PROCEDURE, WHICH CONTAINS A STANDARD
;  EXAMINATION OF THE EXCEPTION-REGISTER, IS CALLED IN THIS WAY:
;      INPUT:  (W2) = RETURN - ADDRESS
;      OUTPUT: (W2) = STATUS

    E0: RS.   W0          E8.
        RS.   W1          E9.
        RS.   W2         E10.
        AL    W0           0  ;  COUNTER = 0
        AL    W1           0
    E1: IO.   W2       (B841.);  SENSE                    6.0 USEC.
        RS.   W2         E11. ;  (E11) = STATUS           4.5 USEC.
        XS                3   ;                           4.0 USEC.
        LA.   W1         -2   ;                           3.5 USEC.
        LS    W1          1   ;  (W1) = 2*(EX.- REG.)     3.5 USEC.
        JL.       X1      2   ;                           4.0 USEC.
        JL.               E6.
        JL.               E4. ;                           3.0 USEC.
        JL.               E3.
    E2: AL.   W0         E13. ;  EX = 11
        JL.               E5.
    E3: AL.   W0         E12. ;  EX = 10
        JL.               E5.
    E4: BA.   W0          1   ;  EX = 01                  4.0 USEC.
        SH.   W0        (E7.);  CNT = CNT + 1             4.5 USEC.
        JL.               E1. ;  SENSE AGAIN              3.0 USEC.
        AL.   W0         E15.
    E5: AM               (18)
        JL    W2         (+0)
    E6: RL.   W0          E8.
        LS    W1          -1
        XL                3
        RL.   W1          E9.
        RL.   W2         E11. ;  (W2) = STATUS
        JL.             (E10.);  RETURN

    E7:                 25000  ;  LIMIT (25000 COUNTS PER SEC.)
    E8:                     0  ;  W0
    E9:                     0  ;  W1
   E10:                     0  ;  W2  (RETURN - ADDRESS)
   E11:                     0  ;  STATUS

   E12: <:<10>DEVICE DISCONNECTED<0>:>
   E13: <:<10>EXCEPTION-REG.=811<0>:>
   E15: <:<10>DEVICE BUSY FOR 1 SEC.<0>:>
```

```
B831:              8.01000000  ;  BIT 5 = 1
B832:                       3
B834:                       0  ;  LENGTH OF TABLE
B840:                       0  ;  DEV. < 6 + PAR. < 2 + 3
B841:                       0  ;  DEV. < 6 + 0
B845:                       0  ;  RETURN ADDRESS
B848:                 0, R. 25  ;  TABLE (MAX. 50 CHARACTERS)

B920: <:SEQUENCE = <0>:>
B930: <:<10>LOAD RC 150<0>:>
B940: <:PARITY = <0>:>
B950: <:DD<0>   VEN<0>  ENERAL<0>:>

B.    B0
W.    B0:  I.                   0  ;  TYPE OUT LENGTH OF PROGRAM
E.

E.


E.
```

```
b.

s. a2
w.
a1: a2
<:RC 150 paper tape punch<0>:>
a2=k-a1
e.

s. a20, b20, c20
w.
a1: a2
<:punch one character 100 times<0>                    :>

; The following is the first instruction in the program.  Note that when
; the program is entered, w1 and w2 are as described in VB431, p. 24.
;
; In run no. zero, ask for the decimal value of the character.

      rs.  w2  b1.        ; save return address;
      so   w1  1          ; if bit 23 of w1 = 1 then
      jl.      a4.        ;  begin

a3:   al.  w0  c1.        ;   write(<:decimal value...:>);
      am       (18)       ;
      jl   w2  (+0)       ;

      am       (18)       ;   b2:= read integer;
      jl   w2  (+8)       ;   (w0:=terminator)
      rs.  w0  b2.        ;

      se   w2  10         ;   if terminator <> newline then
      jl.      a3.        ;     go to a3;

      sl   w0  0          ;   if b2<0 or b2 > 255 then
      sl   w0  256        ;
      jl.      a3.        ;     go to a3;

      jl.      (b1.)      ;   return;
                          ;  end;

; If this is not run no. zero, punch the character 100 times and return.

a4:   rl.  w0  b2.        ; w0:=b2;
      rl   w1  (22)       ; w1:=device number shift 6;
      al   w3  1          ; w3:=1;

a5:   io   w0  x1+11      ; write char(w0); (in general mode)
a6:   io   w2  x1+0       ; sense; (ignore result in w2)
      sx       1          ;  if busy then
      jl.      a6.        ;    go to a6;

      al   w3  x3+1       ; w3:=w3+1;
      sh   w3  100        ; if w3 <= 100 then
      jl.      a5.        ;    go to a5;

      jl.      (b1.)      ; return;

b1: 0                     ; return address
b2: 0                     ; char to be punched

c1: <:<10>decimal value of characyer to be punched:<0>:>

a2=k-a1

e.
e.
```

Udskrift af tabel 3

```
core - store contents
first word addr. = 52
last  word addr. = 82
mode =  decimal

      52        2550
      54        2572
      56        3338
      58        4074
      60           0
      62           0
      64           0
      66           0
      68           0
      70           0
      72           0
      74           0
      76           0
      78           0
      80           0
      82           0
select
```