ST

TEXAS INSTRUMENTS

Improving Man's Effectiveness Through Electronics

Model 990 Computer

Prototyping Monitor (PXRATE) System Operation Guide

(Supplement to the Model 990 Computer Prototyping System Operation Guide)

MANUAL NO. 946243-9701
ORIGINAL ISSUE 1 JANUARY 1977

Digital Systems Division

LIST OF EFFECTIVE PAGES

INSERT LATEST CHANGED PAGES DESTROY SUPERSEDED PAGES

Note: The portion of the text affected by the changes is indicated by a vertical bar in the outer margins of the page.

Model 990 Computer Prototyping Monitor (PXRATE) System Operation Guide (946243-9701)

Original Issue . . . . . 1 January 1977

Total number of pages in this publication is 56 consisting of the following:

PAGE NO.	CHANGE NO.	PAGE NO.	CHANGE NO.	PAGE NO.	CHANGE NO.
Cover	0				
Effective Pages	0				
iii - vi	0				
1-1 - 1-6	0				
2-1 - 2-6	0				
3-1 - 3-6	0				
4-1 - 4-18	0				
Appendix A Divider	0				
A-1 - A-4	0				
Alphabetical Index Div	0				
Index-1 - Index-2	0				
User's Response	0				
Business Reply	0				
Cover Blank	0				
Cover	0				

PREFACE

This supplement to the *Model 990 Computer Prototyping System Operation Guide* describes an enhancement of the TMS 9900 Prototyping System, defined as the Prototyping Monitor (PX) with Resident Assembler, Tracer, Linking Loader, and Editor (PXRATE). This supplement documents changes to the TMS 9900 software modules which now comprise the PXRATE system. It also describes the hardware required to support the PXRATE system.

This manual is intended for users of the TMS 9900 Prototyping System software and for users who require the capability to generate, edit, assemble, load, and debug user application programs and generate firmware for the 990/4 Computer, the 990/10 Computer, and the TMS 9900 Microprocessor. It is also intended for users who desire to perform input/output operations on the 733 ASR Data Terminal to/from peripheral devices.

This manual is divided into four sections and one appendix as follows:

I General Description – Describes the PXRATE software and the hardware needed to support it.

II System Installation and Operation – Gives the sources of information on unpacking, installing, and operating the supporting hardware. Provides step-by-step procedures for PXRATE system software cassette generation.

III Teletypewriter Input/Output Supervisor Calls – Describes the extension of I/O supervisor calls to include interface to the teletypewriter paper tape reader and punch.

IV Demonstration of the PXRATE Software Package – Three sample programs are presented (numbers 3, 4, and 5). Sample program 3 illustrates the use of the PXRATE system in the development of a short user program. Sample program 4 demonstrates the use of supervisor I/O calls on the Model 33 ASR Data Terminal paper tape reader and paper tape punch. Sample program 5 demonstrates the use of the BNPF utility overlay with paper tape instead of magnetic tape cassette.

A Paper Tape Reader/Punch File and Data Formats

The following publications contain additional information needed to use the 990 Prototyping System.

Title	Part Number
Model 990/4 Computer System Hardware Reference Manual	945251-9701
Model 990 Computer TMS 9900 Microprocessor Assembly Language Programmer's Guide	943441-9701
Model 990 Computer Model 733 ASR/KSR Data Terminal Installation and Operation	945259-9701

Title	Part Number
Model 990 Computer PROM Programming Module Installation and Operation	945258-9701
Model 990 Computer Programming Card	943440-9701
Model 990 Computer TTY/EIA Interface Module Installation and Operation Instruction Manual	946240-9701

TABLE OF CONTENTS

APPENDIX

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I

GENERAL DESCRIPTION

1.1 INTRODUCTION

This supplement to the *Model 990 Computer Prototyping System Operation Guide* documents an enhancement of the TMS 9900 Prototyping System, defined as the Prototyping Monitor (PX) with Resident Assembler, Tracer, Linking Loader, and Editor (PXRATE). This section presents an overview of the enhanced software and the hardware required to support it. The first portion of the section describes the purpose and capabilities of the enhanced software and identifies the hardware. The remainder of this section briefly describes changes to the modules that comprise the enhanced system software. These modules include the debug monitor and its overlays, the text editor, and the one-pass assembler.

1.2 PURPOSE AND CAPABILITIES OF THE ENHANCED SYSTEM (PXRATE)

The enhanced TMS 9900 Prototyping System software augments the capabilities of the TMS 9900 Prototyping System software. The original system is documented in the *Model 990 Computer Prototyping System Operation Guide*, Manual Number 945255-9701. User familiarity with the original system documentation and operation is prerequisite to this supplement. The PXRATE system requires a minimum memory size of 12K words. In this 12K, all software development/debug capabilities are coresident (text editor, assembler, linking loader, trace, and program debug). Coresidency of these capabilities negates the time-consuming process of reloading (from cassette) the appropriate utility between program development steps. This coresident package of software also supports the use of the Model 33 ASR Data Terminal paper tape reader/punch.

1.2.1 DESCRIPTION OF SYSTEM ENHANCEMENTS. The purpose of the PXRATE prototyping system software is to provide the capability to generate, edit, assemble, load, and debug user application programs and to generate firmware. The PXRATE package also includes a Device Service Routine (DSR) which supports the use of a TTY paper tape reader/punch. The former instruction trace overlay is no longer an overlay, but is now a command in the resident monitor. In addition to the debug functions, the debug monitor provides supervisor calls to perform input/output (I/O) operations on the 733 ASR Data Terminal and utility routines, such as decimal ASCII to binary, hexadecimal ASCII to binary, binary to decimal ASCII, and binary to hexadecimal ASCII conversion. Overlays to the debug monitor provide the capability to dump/reload a program from/to memory to/from tape in a compressed absolute format. In addition, monitor overlays are provided to support the PROM Programmer Package and BNPF (or HIGH/LOW) formatted dumps to cassette tape or to paper tape. The BNPF overlay also provides the capability to load tape in BNPF format back into memory.

The PXRATE system software is available in object format on a read-only magnetic tape cassette and in source format on punched cards; however, the system source must be assembled and linked using a 990/10 Program Development System. The PXRATE system software provides the capability to generate source and object which is upward compatible with other 990 systems. See Appendix A of *Model 990 Computer Prototyping System Operation Guide* for restrictions.

1.2.2 HARDWARE CONFIGURATION REQUIRED FOR PXRATE SYSTEM SOFTWARE. The hardware required for the PXRATE prototyping system is the same as that described in the Prototyping System Manual except that a minimum of 12K words of memory is required in the enhanced system. See figures 1-1 through 1-4 for detailed memory requirements of systems with 12, 16, 20, and 24K words of memory. Part of the user area of the memory is shared by the text editor and the one-pass assembler for tables and buffer space.

The Model 33 ASR teletypewriter may be configured as an optional peripheral (TTY/EIA Kit, Part Number 974704-0002). It may not be used as the system console. The interface card must be inserted in the left half of computer chassis slot 6, corresponding to CRU base address 20_{16}.

1.3 SYSTEM PART NUMBERS

Refer to Manual Number 945255-9701, paragraph 1.3, page 1-5.

1.4 SOFTWARE MODULES

In the PXRATE prototyping system, the debug monitor, trace capability, text editor and the one-pass assembler have been combined into one module. In addition, when PXRATE is initially loaded, the linking loader overlay is automatically placed in the overlay area. This software package now becomes coresident in memory. The instruction trace overlay now becomes a part of the monitor and is no longer an overlay. The PXRATE package is supplied on cassette tape. Two new commands now reside in the coresident package. These commands are PA to activate the one-pass assembler, and TE to activate the text editor. No parameters are necessary for either command. Following the invocation of either of these commands, the operation of the assembler or text editor is the same as described in Section IV of the Prototyping System Manual. Activation and operation of the trace and linking loader has not been altered, except that no overlay process is required prior to their use. Since the linking loader is located in the monitor overlay area, it will be overwritten if another overlay is placed in memory, and must be reloaded to be used again.

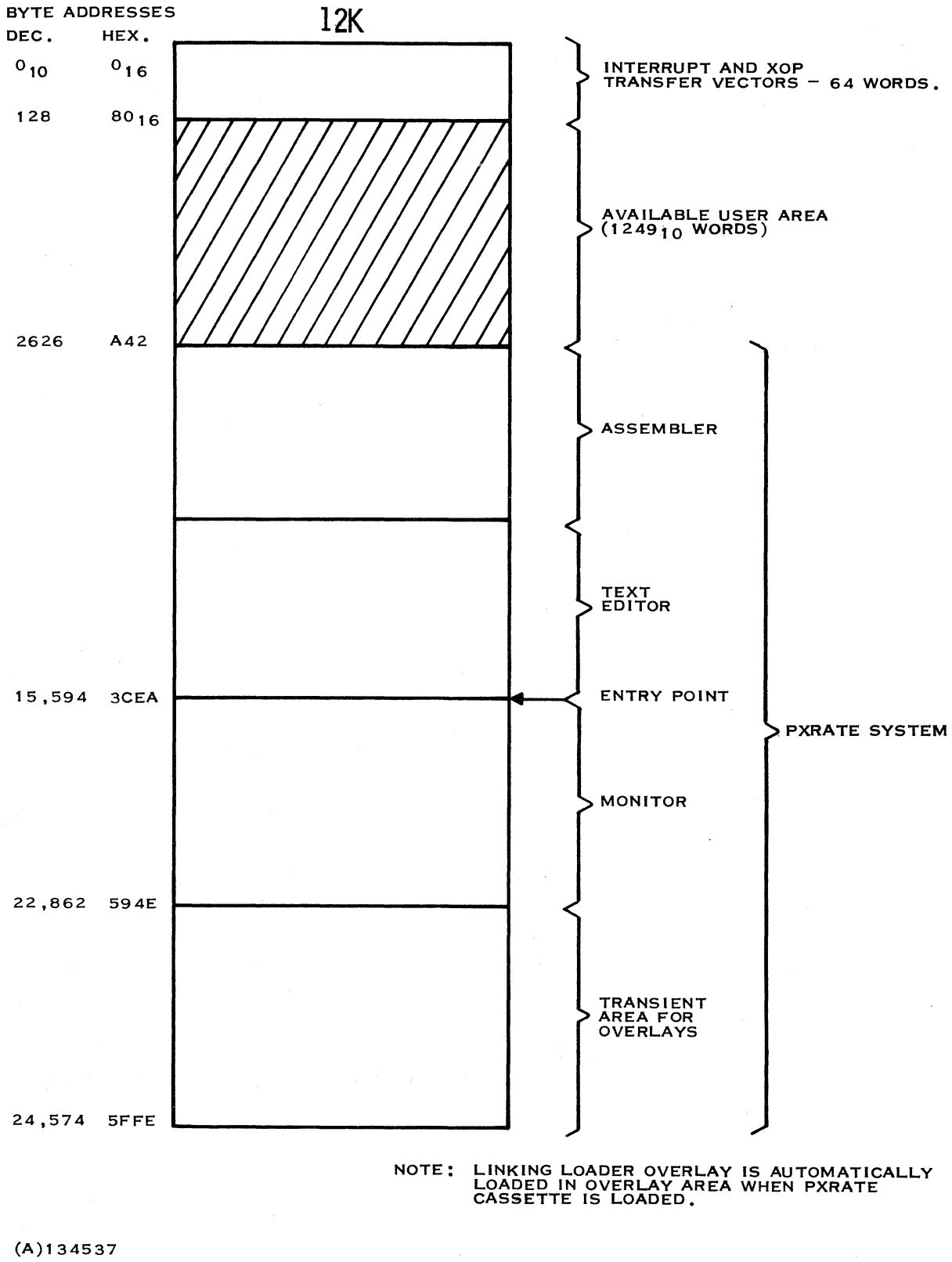


Figure 1-1. 12K Hardware Memory Configuration

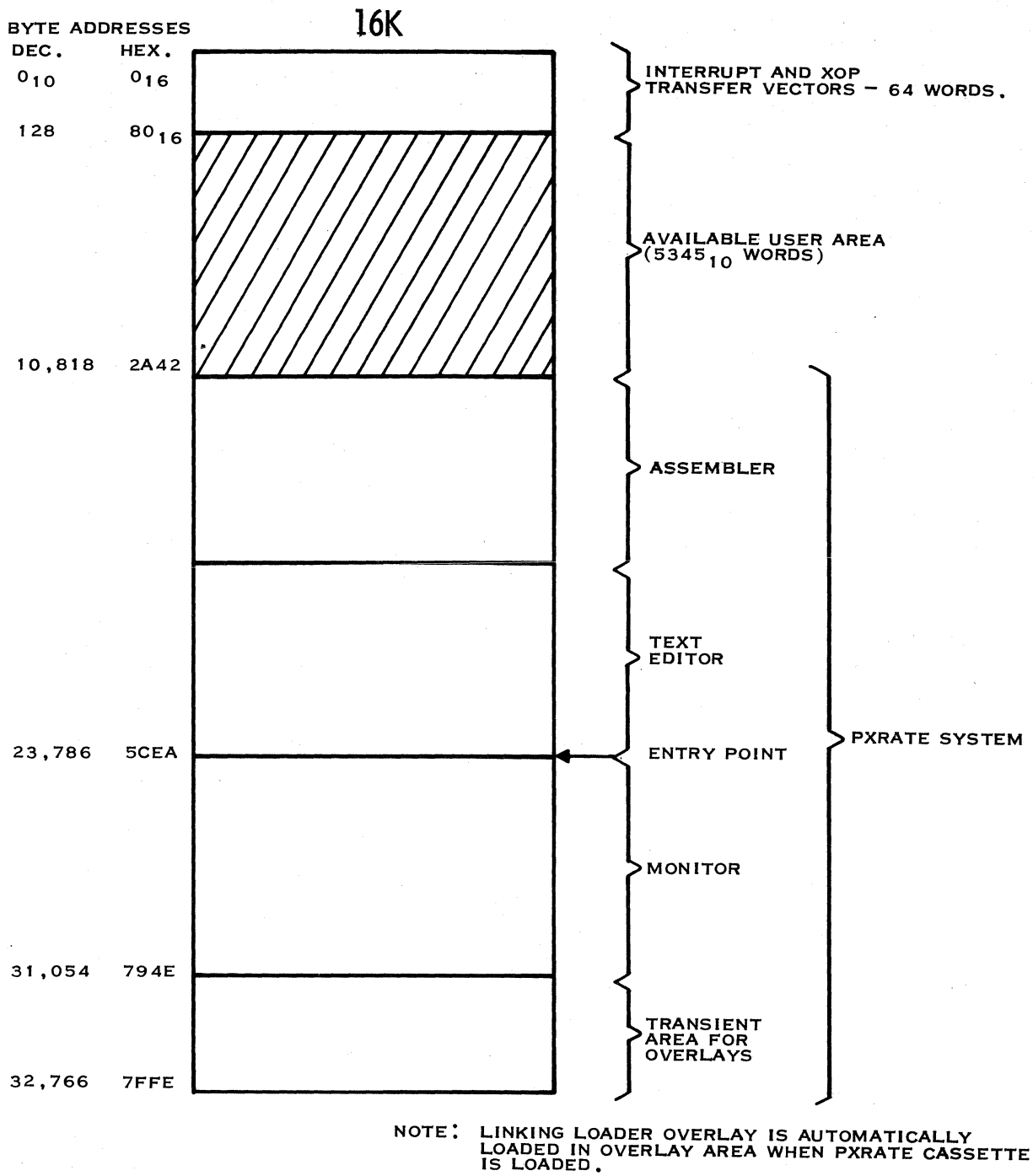


Figure 1-2. 16K Hardware Memory Configuration

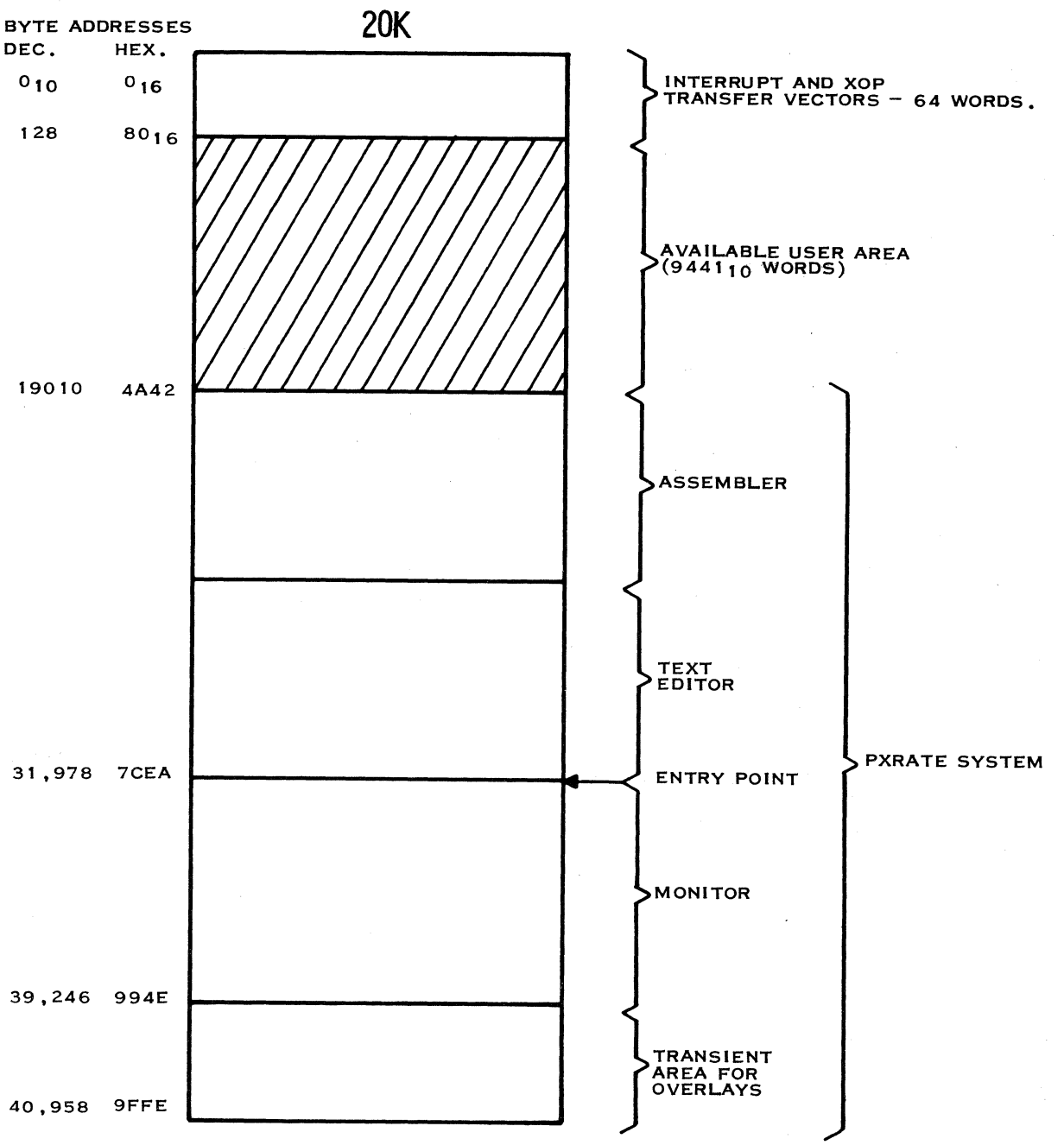


Figure 1-3. 20K Hardware Memory Configuration

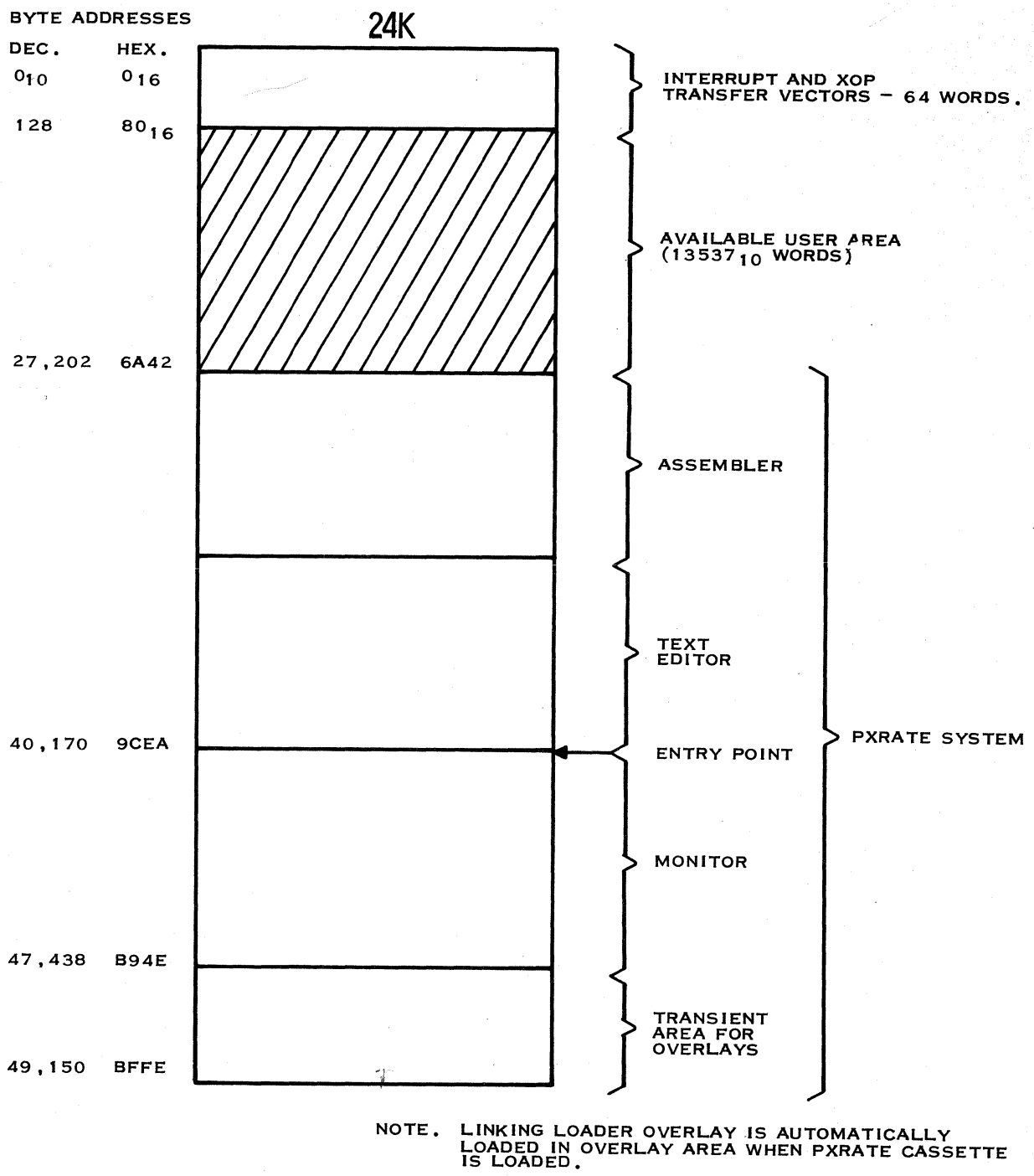


Figure 1-4. 24K Hardware Memory Configuration

SECTION II

SYSTEM INSTALLATION AND OPERATION

2.1 INTRODUCTION

Refer to Manual Number 945255-9701, Section II, paragraph 2.1, page 2-1.

2.2 UNPACKING AND INSTALLATION OF HARDWARE

Refer to Manual Number 945255-9701, Section II, paragraph 2.2, page 2-1.

2.3 HARDWARE OPERATION

Refer to Manual Number 945255-9701, Section II, paragraph 2.3, page 2-3.

2.4 PXRATE SYSTEM SOFTWARE CASSETTE GENERATION

Absolute versions of the PXRATE package are supplied for each memory configuration. The PXRATE prototyping system kit includes two cassettes containing 13 files total. The files are arranged as shown in table 2-1.

Note that absolute code versions of PXRATE are supplied for all of the available memory configurations, saving the user from having to create them during installation. There is one set of overlays in relocatable object format, which can be used with any of the versions of PXRATE.

Paragraphs 2.4.1 and 2.4.2 detail the procedures used to create a bootstrap tape cassette of the PXRATE monitor and individual cassettes for each overlay.

Table 2-1. PXRATE System Software Cassettes

PXRATE Cassette 1 of 2

Side A	Side B
1. Upfront Loader	1. Upfront Loader
2. Absolute Code Version of PXRATE for 12K Machine	2. Absolute Code Version of PXRATE for 20K Machine
3. Absolute Code for 16K Machine	3. Absolute Code for 24K Machine

PXRATE Cassette 2 of 2

Side A	Side B
1. Linking Loader Overlay	1. Relocatable Code Version of PXRATE
2. Absolute Dump/Load Overlay	
3. PROM Programmers, Part 1	
4. PROM Programmers, Part 2	
5. BNPF Dump Overlay	
6. HIGH/LOW DUMP Overlay	

2.4.1 PROCEDURE FOR MAKING CASSETTE FOR MONITOR FROM PXRATE MASTER.

For the convenience of the user, the text of the following procedure is keyed by parenthetical numbers to the 733 ASR panel switches illustrated in figure 2-1.

1. Turn on power to the 733 ASR Data Terminal.

2. On the ASR panel:

   a. Set KEYBOARD (1) to OFF.

   b. Set PLAYBACK (2) to LOCAL.

   c. Set RECORD (3) to LOCAL.

   d. Set PRINTER (4) to OFF.

   e. Set CASSETTE 1 to PLAYBACK (10); CASSETTE 2 will be in RECORD.

3. Before inserting a tape cassette into a transport, check the status of the write tabs on the bottom of the tape. If the tape is to be written on, the tab for the side of the tape to be written on should cover the hole in the cassette case; if it is a read-only tape, the tab should not cover the hole. Holding a tape cassette in front of you with the tape side up, the write tab for that side of the tape will be on the bottom right of the tape cassette. See figure 2-2.

4. Select the side of PXRATE cassette 1 that is specified in table 2-1 for computer memory size. Insert into cassette 1 transport.

   a. Open the cassette 1 transport door.

   b. Insert the tape cassette with the tape side up as shown in figure 2-3.

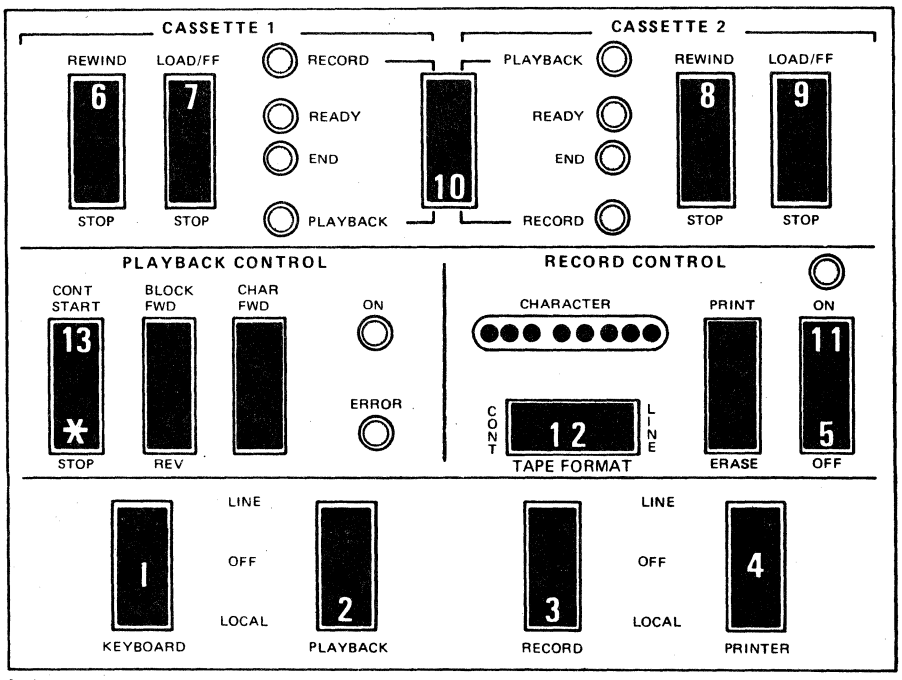


Figure 2-1. 733 ASR Panel

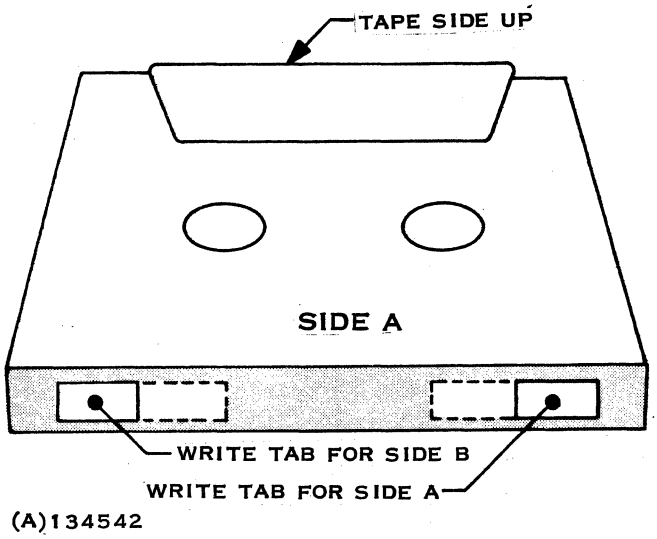


Figure 2-2. Tape Cassette Write Tabs

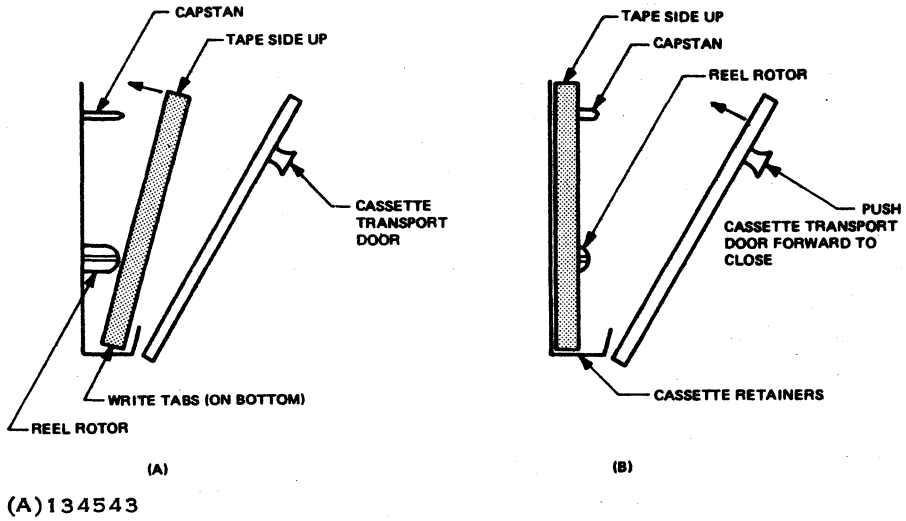


Figure 2-3. Tape Cassette Installation

c. Press the tape cassette down and in, being sure that the capstan and reel rotors fit into the proper holes.

d. Close the cassette transport door.

5. Insert the copy tape cassette side A into the cassette 2 transport, and:

a. Set RECORD CONTROL to OFF (5).

b. Set CASSETTE 1 to REWIND (6).

c. When the END lamp lights, press LOAD/FF (7). The READY lamp should light after a few seconds.

d. Set CASSETTE 2 to REWIND (8).

e. When the END lamp lights, press LOAD/FF (9). The READY lamp should light after a few seconds.

f. Set RECORD CONTROL to ON (11). The ON lamp should light.

g. Set TAPE FORMAT (12) to LINE.

h. Press CONT START (13) to begin high-speed tape duplication. The upfront loader file will be recorded from cassette 1 to cassette 2 (original tape to copy tape) when the CHARACTER lamps stop flashing.

NOTE

To stop the duplication process at any point, press (CONT) STOP.

i. Set RECORD CONTROL to OFF (5). This clears the buffer and records the last block of data on the tape.

6. If the desired version of PXRATE cassette 1 is the second file (see table 2-1), skip step 6.a and continue with step 6.b.

   a. Press CONT START (13) to read through the second file. This brings the third file into position for recording.

   b. Set RECORD CONTROL to ON (11).

   c. Press CONT START (13) to record the monitor to the copy cassette. When the CHARACTER lamps stop flashing, recording is complete.

   d. Set RECORD CONTROL to OFF (5).

   e. Set CASSETTES 1 and 2 to REWIND (6) (8).

7. Remove PXRATE cassette 1 and store in a safe place.

8. Remove copy cassette and label it "PXRATE BOOTSTRAP TAPE". Adjust write tab on bottom of tape to uncover hole for side A of tape (see step 3 and figure 2-2). This protects tape side A from being accidentally overwritten. The tape is now ready for use.

9. For loading the PXRATE tape, refer to paragraph 2.5.1 of the Prototyping System Manual.

2.4.2 PROCEDURE FOR MAKING CASSETTES FOR OVERLAYS. The user should copy each of the overlay object files to a separate cassette for convenience in using the system. The following procedure should be used. The text of this procedure is keyed by parenthetical numbers to the 733 ASR panel switches illustrated in figure 2-1.

1. Select side A of PXRATE cassette 2, and insert into cassette drive 1. Insert a blank cassette, side A, into cassette drive 2. Rewind and ready both tapes. File 1 is now ready for recording (see table 2-1).

2. Set RECORD CONTROL to ON (11).

3. Press CONT START (13). This causes the selected file to be recorded on the copy cassette.

4. After copying the PROM Programmer part 1, repeat step 4 to record the next file (PROM Programmer part 2) onto the same cassette. Do NOT turn RECORD CONTROL to OFF between these two files.

5. Set RECORD CONTROL to OFF (5) to clear the buffer and record the last block of data on the tape.

6. Rewind and remove the copy cassette and label with the appropriate file name. Protect tape side A from overwrite by setting write tab to uncover hole on bottom of tape for side A (see figure 2-2).

7. If there are any remaining overlay object files on PXRATE cassette 2, insert a blank cassette into cassette drive 2. Rewind and ready blank cassette and repeat steps 2 through 6 (note that step 4 applies only to the PROM Programmer overlay). Continue this process until each overlay is recorded on a separate cassette.

8. Rewind and remove PXRATE cassette 2 and store it in a safe place.

The relocatable code version of PXRATE (table 2-1, cassette 2 side B) is supplied in the event that the user would like to "custom place" the monitor in memory. The user must affix the appropriate D-tag to the beginning of the relocatable version (see the example on page 2-5 of the Prototyping System Manual for details).

Relevant Data:

Length of monitor (including editor and assembler, but not including overlay area)	=	$4FOC_{16}$ bytes
Length of longest overlay	=	$6BO_{16}$ bytes
Monitor entry point	=	D-tag value + $32A8_{16}$ bytes

CAUTION

It is possible for the user to overwrite the assembler, text editor, and monitor by loading a user program larger than the available user space (see figures 1-1 through 1-4). If this should happen, reload the entire PXRATE package from the cassette tape.

SECTION III

TELETYPEWRITER INPUT/OUTPUT (I/O) SUPERVISOR CALLS

3.1 INTRODUCTION

This section describes the extension of prototyping system I/O supervisor calls to include program I/O interface to the teletypewriter paper tape reader and punch (part of the Model 33 ASR Data Terminal). The user is referred to paragraph 3.5 of the Prototyping System Manual for a general description of supervisor I/O calls. Read and write supervisor call codes are consistent with those of the rest of the prototyping system.

3.1.1 HARDWARE (MODEL 33 ASR DATA TERMINAL). The Model 33 ASR Data Terminal is an automatic send/receive teletypewriter. It has a paper tape reader and punch for automatic input and output. The reader is a means of automatically entering programs that are on punched paper tape. The punch is a means of recording object code output of the computer on paper tape. File data of other types may be read or punched in a similar manner. The reader may be put online for input, and the punch may be put online for output.

NOTE

The Model 33 ASR Data Terminal is supported as an optional peripheral only. It may not be configured as the system console in place of the 733 ASR terminal.

Data is punched on paper tape serially by character. The individual bits of each character are aligned across the width of the tape and are punched in parallel. The hole in the tape represents a one bit and the absence of a hole represents a zero bit. Data is transmitted and received simultaneously (full duplex) over current loops in 7-bit ASCII for textual character or 8-bit code for direct mode. The Model 33 ASR requires a Teletypewriter Peripheral Kit (ASR 3320/5JE Interface), TI Part Number 974704-0002, which is supported by the computer CRU DSR and I/O supervisor calls.

3.1.2 REFERENCES. For further information, refer to *TTY/EIA Interface Module Installation and Operation Instruction Manual*, Section II, TI Part Number 946240-9701. A complete description of the Model 33 ASR may be found in *Technical Manual, 33 Teletypewriter Sets*, Volumes 1 and 2 (Bulletin 310B) and *Model 33 Page Printer Parts* (Bulletin 1184B) supplied with the teletypewriter. If your teletypewriter was not supplied by Texas Instruments, modifications are necessary as outlined in the above referenced *TTY/EIA Interface Module Installation and Operation Instruction Manual.*

3.2 ACCESS TO MODEL 33 ASR DATA TERMINAL

The PXRATE system accesses the TTY paper tape reader/punch by the invocation of supervisor calls. A supervisor call is made with an extended operation (XOP) assembly language machine instruction using an extended operation code of 15. The XOP instruction specifies an address pointing to a Physical Record Block (PRB) containing the supervisor call and any necessary arguments. The format of the PRB is illustrated in figure 3-1.

A detailed discussion of the PRB parameters is presented in paragraph 3.5.2 of the Prototyping System Manual.

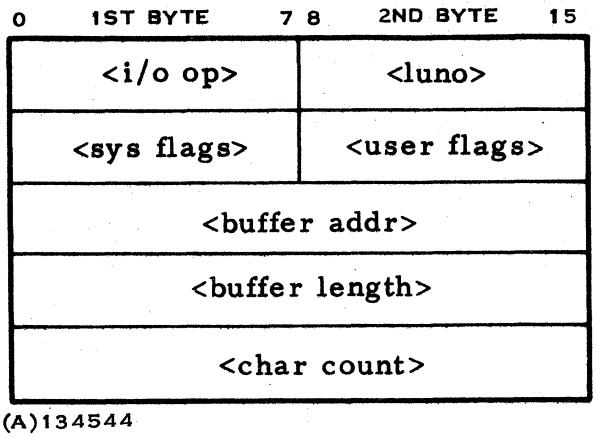


Figure 3-1. Format of Physical Record Block

3.3 LOGICAL UNIT NUMBER (LUNO) ASSIGNMENT

Before execution of any supervisor call I/O operation, a LUNO must be assigned to the particular peripheral device with which I/O communication is desired (refer to paragraph 3.4.2 of the Prototyping System Manual).

With the Model 33 ASR Data Terminal, the device symbols and names of interest are:

PTR	Paper Tape Reader
PTP	Paper Tape Punch

An example of LUNO assignment is:

```
.AL,3,PTR
```

This assigns Logical Unit Number 3 to the paper tape reader, so that whenever LUNO 3 appears in a PRB, the paper tape reader will be accessed through the execution of a supervisor I/O operation. It should be noted that default LUNOs have not been provided for Model 33 ASR Data Terminal support (refer to paragraph 3.4.2 of the Prototyping System Manual).

3.4 OPERATION CODE (OPCODE) INTERPRETATION

The monitor's interpretation of the various opcodes in the PRB depend upon the particular teletypewriter device being accessed. The following two paragraphs describe those interpretations.

3.4.1 PAPER TAPE READER. The functions of the possible opcodes for the tape reader are described in table 3-1.

3.4.2 PAPER TAPE PUNCH. The functions of the possible opcodes for the tape punch are described in table 3-2.

3.5 ERROR RETURNS

Error detection is performed by the Device Service Routine (DSR) with meanings as defined in table 3-3.

When a supervisor call causes one of these errors to occur, the code is returned in the user flags byte of the PRB (see figure 3-1).

Table 3-1. Functions of Opcode for Tape Reader

Hexadecimal I/O Opcode	Name	Function
00	Open	Sets reader flag so data can be read from device, error 3* if already opened.
01	Close	Resets reader flag.
02	Close EOF	Error 2*
03	Open Rewind	Same as Open
04	Close Unload	Same as Close
05	Read Status	Ignored
06	Forward Space	Ignored
07	Backspace	Ignored
08	FMP	Ignored
09	Read ASCII	If reader flag is set, data is read from the paper tape. Termination is caused either by a carriage return or user's buffer full. Only seven bits of data are transferred to the user's buffer with the eighth bit (parity) ignored. At termination the tape continues until a reader-off character is sensed, during which no data is stored. The number of characters read is returned to the user's character count.
A	Read Direct	Checks if reader flag is set. If set, data is read until the first non-null frame is encountered. Termination is caused by user's buffer full, and tape continues to pass until a reader-off is sensed during which no data is stored. The number of characters read is returned to the user's character count.
B	Write ASCII	Error 2*
C	Write Direct	Error 2*
D	Write EOF	Error 2*

*See table 3-3 for error meanings.

3.6 PAPER TAPE FILE AND DATA FORMATS

Two types of paper tape I/O are supported: ASCII and Direct. The type of data to be punched or the interpretation of the data to be read depends on the particular type of opcode used (see Opcode Interpretation, paragraph 3.4).

3.6.1 ASCII. This type of data represents characters in ASCII 7-bit code. ASCII mode is used for storing and retrieving character information. The ASCII paper tape format and an actual paper tape sample are illustrated in figure 3-2.

Table 3-2. Functions of Opcode for Tape Punch

Hexadecimal I/O Opcode	Name	Function
00	Open	Sets write flag so data can be punched to device, error 3* if already opened.
01	Close	Resets write flag.
02	Close EOF	Resets write flag and punches ASCII EOF, and 80 null frames.
03	Open Rewind	Same as Open and punches 80 null frames.
04	Close Unload	Same as Close operation
05	Read Status	Ignored
06	Forward Space	Ignored
07	Backspace	Ignored
08	FMP	Ignored
09	Read ASCII	Error 2*
A	Read Direct	Error 2*
B	Write ASCII	Checks if write flag is set. If set, data is punched to the paper tape specified by the user's PRB, where word 6 tells the number of characters and word 4 tells the address of characters to be punched. Only seven bits of data are punched where the eighth bit (parity) is always zero. Termination is caused by end of character count, and then an ASCII EOR is punched.
C	Write Direct	Checks if write flag is set. If set, punches the specified number of characters and Direct EOR.
D	Write EOF	If write flag is set, punches ASCII end of file.
E	Rewind	Ignored
F	Unload	Ignored

*See table 3-3 for error meanings.

3.6.2 **DIRECT.** This is an 8-bit code in which straight 8-bit binary values are represented on tape. A hole represents a one bit and lack of a hole represents a zero bit. It is used for storing and retrieving binary data on paper tape. The Direct paper tape format and an actual paper tape sample are illustrated in figure 3-3.

For additional information regarding the ASR 33 Data Terminal paper tape reader and paper tape punch file and data formats, see Appendix A of this supplement.

Table 3-3. Error Code Functions

Error Code	Function
2	Illegal opcode
3	Device already open
5	Device not open

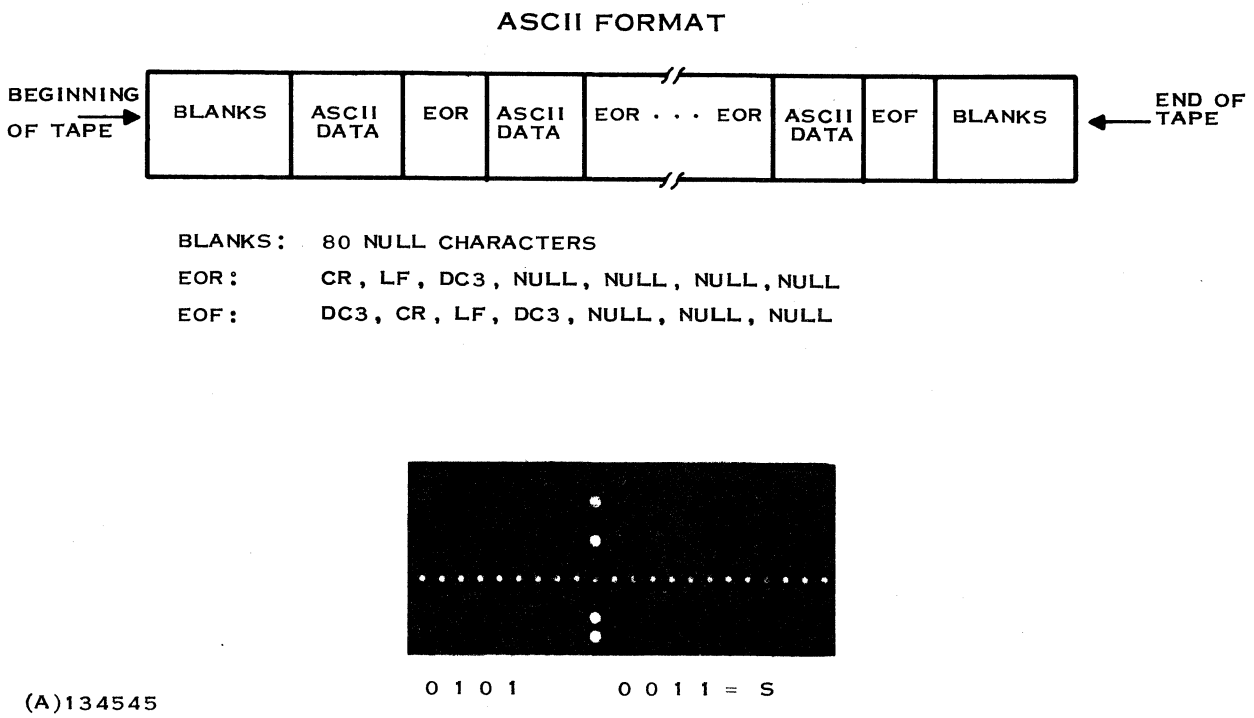


Figure 3-2. ASCII Paper Tape Format with Sample Paper Tape

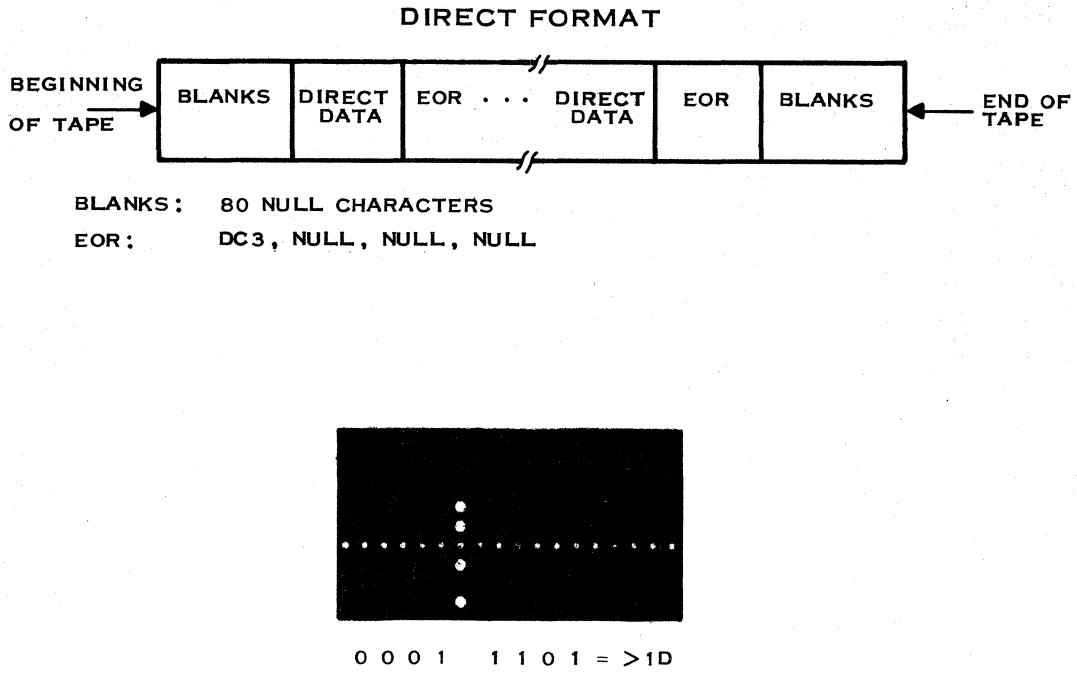


Figure 3-3. Direct Paper Tape Format with Sample Paper Tape

SECTION IV

DEMONSTRATION OF THE PXRATE SOFTWARE PACKAGE

4.1 SAMPLE PROGRAMS

Sample programs 1 and 2 exist in the Prototyping System Manual. These programs may be used with the PXRATE system. A minimum of 4K-word user memory space is required with sample program 2. Therefore, at least a 16K-word memory prototyping system is required for sample program 2.

Three new sample programs (numbers 3, 4, and 5) are presented in this section. Sample program 3 illustrates the use of the PXRATE system in the development of a short user program. Sample program 4 demonstrates the use of supervisor I/O calls on the Model 33 ASR Data Terminal paper tape reader and paper tape punch. Sample program 5 demonstrates the use of the BNPF utility overlay with paper tape instead of magnetic tape cassette.

These sample programs are presented with the assumption that the user has read the Prototyping System Manual and this supplement to the manual, and has been introduced to the functions used in these documents.

The source for sample programs 3 and 4 are provided on tape cassette; sample program 5 requires only the BNPF overlay.

Included in each of the following sample programs is a brief explanation of the procedure and a listing of most, if not all, of the actual procedures followed. In some places, comments have been added within computer printout reproductions. The following conventions are used in each of the sample program listings:

- Comments that explain to the user what to do or what is happening are delineated by parentheses.
- Keys that must be depressed on the log keyboard are indicated in the listings by an underscore.
- Angled brackets enclose a single key to be depressed; e.g., (<CR>) directs the user to depress the carriage return.

4.1.1 SAMPLE PROGRAM 3, USER PROGRAM DEVELOPMENT. This program demonstrates the use of the PXRATE system in the development of a user program. The source for this program is supplied to the user on a cassette. The program takes a string of ASCII digits and, given the radix, computes the binary value of the number. An intentional error has been introduced into the program to demonstrate use of the PXRATE system to detect, locate, and correct programming errors.

4.1.2 TEXT EDITOR. The user should ready the sample program cassette in the cassette 1 transport of the 733 ASR Data Terminal and ready a blank cassette in the cassette 2 transport. Now some comments are added to the source by means of the text editor. See figure 4-1.

```
(EDIT THE SOURCE PROGRAM)

.IE      (<CR>)

PXREDT  948927 ** 12JUL76

POSITION TAPES, ENTER CR


(<CR>)
 ?D50
END OF FILE
 ?T  (<CR>)
 ?P50  (<CR>)
 0001          IDT  'BASCON'
 0002 *
 0003 *        THIS PROGRAM TAKES AN ASCII NUMBER AND
 0004 *        CONVERTS IT TO A SPECIFIC BASE IN 2'S
 0005 *        COMPLEMENT NOTATION.  THE POSITIVE
 0006 *        NUMBER MUST BE IN 7-BIT ASCII FORMAT
 0007 *        STARTING AT LOCATION (INDEPENDENT OF PROGRAM
 0008 *        BASE ADDRESS) >104, ONE DIGIT PER BYTE.
 0009 *        THE BASE VALUE SHOULD BE A DATA WORD AT
 0010 *        LOCATION >100; THE LENGTH OF THE NUMBER
 0011 *        TO BE CONVERTED, IN DIGITS, SHOULD BE AT
 0012 *        DATA LOCATION >102.
 0013 *        THE RESULT IS RETURNED AT LOCATION >120.
 0014 *        NO ASCII VALIDITY CHECKS OR OVERFLOW CHECKS
 0015 *        ARE MADE.
 0016 *
 0017 ENTRY    LWPI WRKS
 0018          CLR  R2                  R2=LOOP COUNTER
 0019          CLR  R3                  R3=INTERMEDIATE ACCUMULATOR
 0020 LOOP     MPY  @BASE,R3            MULT. SUBRESULT BY BASE
 0021          MOV  R4,R3               LOAD LEAST SIG. BITS OF RESULT
 0022          MOVB @NUMBER(R2),R4
 0023          SRL  R4,8
 0024          A    R4,R3               ADD NEXT DIGIT
 0025          AI   R3,-30
 0026          INC  R2
 0027          C    R2,@LENGTH
 0028          JNE  LOOP                JUMP BACK IF NOT DONE
 0029 MOV      MOV  R3,@RESULT
 0030          XOP  @OUT,15
 0031          AORG >100
 0032 BASE     DATA >10
 0033 LENGTH   DATA >6
 0034 NUMBER   BSS  >10
 0035          AORG >120
 0036 RESULT   DATA 0
 0037 WRKS     BSS  >20
 0038 OUT      DATA >0400
 0039          END  ENTRY
LAST LINE
 ?
(A)134547 (1/2)
```

Figure 4-1. Edit the Source Program (Sheet 1 of 2)

```
(ADD SOME COMMENT LINES TO THE SOURCE)
 ?D30  (<CR>)
 ?P    (<CR>)
 0031          AORG >100
 ?U      "
 ?P      "
 0030          XOP  @OUT,15
 ?C      "
 XOP @OUT,15 RETURN TO MONITOR   (CONTROL-I'S USED FOR TABS)

(<CR>,<CR>)

 ?T   (<CR>)
 ?D31   "
 ?P     "
 0031          AORG >100
 ?I     "
*
* DATA STORAGE AREA
*   (<CR>,<CR>)

 ?T    (<CR>)
 ?D29   "
 ?P6    "
               XOP  @OUT,15          RETURN TO MONITOR
 0031          AORG >100
       *
       * DATA STORAGE AREA
       *
 0032 BASE     DATA >10
 ?Q  (<CR>)
END EDIT
TERMINATE/CONTINUE?T (<CR>)

.     (HIT <RECORD CONTROL OFF> SWITCH TO PUT AN EOF ON TAPE  )


(A)134547 (2/2)
```

Figure 4-1. Edit the Source Program (Sheet 2 of 2)

The edited source program is recorded on the blank cassette by the PXRATE text editor. Now, remove the sample program cassette from the cassette 1 transport.

4.1.3 ONE-PASS ASSEMBLER. In order to assemble the program, the edited source cassette (which has just been recorded) should be readied in the cassette 1 transport with a blank cassette in the cassette 2 transport to accept the object code. Follow the procedure as demonstrated in figure 4-2, recording the object code on the blank cassette in the cassette 2 transport.

```
.(ASSEMBLE THE PROGRAM)

.PA  (<CR>)

PXRASM  948925 ** 12JUL76
PREDEFINED REGISTERS? Y (<CR>)

ASM/TERM? A (<CR>)



                                                        PAGE 0001

  0001                          IDT  'BASCON'
  0002                  *
  0003                  *       THIS PROGRAM TAKES AN ASCII NUMBER AND
  0004                  *       CONVERTS IT TO A SPECIFIC BASE IN 2'S
  0005                  *       COMPLEMENT NOTATION.  THE POSITIVE
  0006                  *       NUMBER MUST BE IN 7-BIT ASCII FORMAT
  0007                  *       STARTING AT LOCATION (INDEPENDENT OF PROGRAM
  0008                  *       BASE ADDRESS) >104, ONE DIGIT PER BYTE.
  0009                  *       THE BASE VALUE SHOULD BE A DATA WORD AT
  0010                  *       LOCATION >100; THE LENGTH OF THE NUMBER
  0011                  *       TO BE CONVERTED, IN DIGITS, SHOULD BE AT
  0012                  *       DATA LOCATION >102.
  0013                  *       THE RESULT IS RETURNED AT LOCATION >120.
  0014                  *       NO ASCII VALIDITY CHECKS OR OVERFLOW CHECKS
  0015                  *       ARE MADE.
  0016                  *
  0017  0000  02E0  ENTRY  LWPI WRKS
        0002  ----
  0018  0004  04C2         CLR  R2                 R2=LOOP COUNTER
  0019  0006  04C3         CLR  R3                 R3=INTERMEDIATE ACCUMULATOR
  0020  0008  38E0  LOOP   MPY  @BASE,R3           MULT. SUBRESULT BY BASE
        000A  ----
  0021  000C  C0C4         MOV  R4,R3              LOAD LEAST SIG. BITS OF RESULT
  0022  000E  D122         MOVB @NUMBER(R2),R4
        0010  ----
  0023  0012  0984         SRL  R4,8
  0024  0014  A0C4         A    R4,R3              ADD NEXT DIGIT
  0025  0016  0223         AI   R3,-30
        0018  FFE2
  0026  001A  0582         INC  R2
  0027  001C  8802         C    R2,@LENGTH
        001E  ----
  0028  0020  16F3         JNE  LOOP               JUMP BACK IF NOT DONE
  0029  0022  C803  MOV    MOV  R3,@RESULT
        0024  ----
  0030  0026  2FE0         XOP  @OUT,15            RETURN TO MONITOR
        0028  ----
  0031  0100               AORG >100
  0032                  *
  0033                  * DATA STORAGE AREA
  0034                  *
  0035  0100  0010  BASE   DATA >10
        000A**0100
  0036  0102  0006  LENGTH DATA >6
        001E**0102
  0037  0104        NUMBER BSS  >10
        0010**0104
  0038  0120               AORG >120
  0039  0120  0000  RESULT DATA 0
        0024**0120
  0040  0122        WRKS   BSS  >20
        0002**0122
  0041  0142  0400  OUT    DATA >0400
        0028**0142
  0042                      END  ENTRY

 0000 ERRORS

 ASM/TERM? T (<CR>)

 .(HIT <RECORD CONTROL OFF> SWITCH)
(A)134548
```

Figure 4-2. Assemble the Program

4.1.4 LOAD UPDATED PROGRAM AND EXECUTE. Move the object cassette from cassette 2 transport to the cassette 1 transport in order to load the object code (figure 4-3). (Alternately, the cassette could be kept in the cassette 2 transport, the RECORD/PLAYBACK switch set to PLAYBACK from it, and the LP command below modified to LP, 8. See paragraph 3.4.5 of the Prototyping System Manual.)

After a trial execution, as demonstrated in figure 4-3, the result shown in address 120_{16} is incorrect.

4.1.5 PROGRAM TRACE. Using the trace capability of the PXRATE system, the user can single step through program execution, using the space bar to advance to the next instruction (figure 4-4).

At this point, the program trace has been stopped by means of the escape key, since an error is indicated by the last line of trace corresponding to the instruction found at line 25 at the program (AI R3, -30). It was intended to subtract 30 from R3; however, it must be hexadecimal 30, not decimal 30. The easiest method of correction for this single error is to patch memory location 18_{16} value $FFE2_{16}$, which actually resides in memory at $B8_{16}$ because of the $A0_{16}$ offset from the load command. The new value for 18_{16} should be $FFDO_{16} = -30_{16}$.

4.1.6 PATCH AND EXECUTE PROGRAM, VERIFY RESULTS. The patch is made, the program executed, and the results verified (figure 4-5).

This is the correct result since E_{16} is 14_{10} or 32_4. If the user wishes, a permanent fix may be made by changing "AI R3, -30" to AI R3, ->30" in the source using the text editor.

4.2 SAMPLE PROGRAM 4, PAPER TAPE UNIT DEMONSTRATION USING I/O SUPERVISOR CALLS

The general flow of the program to demonstrate paper tape outputs an initial prompt that expects one of three possible inputs; all other inputs cause the initial prompt to be output again. The three inputs are: E, to demonstrate error codes; O, to demonstrate reads and writes; and S, to return to the monitor. If the E or O options are taken, another prompt is output which expects an opcode followed by a mode type (R for reader and P for punch). The opcode is

```
(LOAD UPDATED PROGRAM)

.LP  (<CR>)
.
(SET UP SOME TRIAL INPUT, TO CONVERT '32' IN BASE 4 TO BINARY)
.MM,100 (<CR>)
0100=0010  4    (HIT <SPACE BAR> TO MODIFY NEXT LOCATION)
0102=0006  2          "
0104=3322  3332  (HIT <CR>)
.EX (<CR>)
.

(THE PROGRAM SHOULD HAVE EXECUTED; WE NOW EXAMINE THE RESULTS)

.IM,120  (<CR>)
0120=0068
.
     (WRONG!!)


(A)134549
```

Figure 4-3. Load Updated Program

```
(EXECUTE PROGRAM UNDER TRACE)

.ST,0,PSEBADEBA  (<CR>)
.SR,0,A0,FF,0,S  (<CR>)
.MR  (<CR>)

PC=00A0      (<SPACE BAR>)
WP=0122  0       "
ST=C000  0       "
.RU   (<CR>)
00A0  SE=0000  SB=0072  SA=0072  (HIT <SPACE BAR> FOR EACH NEW LINE)
00A4  SE=0126  SB=0002  SA=0000
00A6  SE=0128  SB=0068  SA=0000
00A8  SE=0100  SB=0004  SA=0004  DE=0128  DB=0000  DA=0000
      012A=0000
00AC  SE=012A  SB=0000  SA=0000  DE=0128  DB=0000  DA=0000
00AE  SE=0104  SB=3332  SA=3332  DE=012A  DB=0000  DA=3300
00B2  SE=012A  SB=3300  SA=0033
00B4  SE=012A  SB=0033  SA=0033  DE=0128  DB=0000  DA=0033
00B6  SE=0128  SB=0033  SA=0015
                                 (AHAAA!!--HIT <ESC>)
.
```

(A)134550

Figure 4-4. Execute Program Under Trace

```
(PATCH AND EXECUTE PROGRAM, THEN VERIFY THE RESULTS)

.IM,100,104  (<CR>)
0100=0004  0002  3332
.


 (TO PATCH ASSEMBLY LISTING LOCATION >18, WE MUST ADD THE PROGRAM LOAD BIAS)

.HA,18,A0  (<CR>)
SUM=00B8   00184  DIFF=FF78  -00136
.MM,B8   (<CR>)
00B8=FFE2  FFD0
.MR   (<CR>)

PC=00BA  A0     (<CR>)
.EX   (<CR>)
.IM,120  (<CR>)
0120=000E
.
      (THE CORRECT ANSWER!!)
```

(A)134551

Figure 4-5. Patch and Execute Program, Then Verify Results

processed, and the system and user's flags are output unless an open reader call is done. After the opcode is processed, the program returns to the initial prompt. If the user detects an error (in the user's flag word), it can be cleared either by clearing that word in memory using the MM command in the monitor or by taking the E route from the initial prompt. When the user takes the E route, only the error flags associated with the punch are cleared; therefore, demonstration of error codes must be with respect to the punch. When the punch is opened, the teletypewriter echoes back two bells and expects the user to turn-on the punch and press the RUB OUT key. Because RUB OUT is not echoed back, it will not appear on the paper tape.

4.2.1 **DEMONSTRATION PROCEDURE.** To perform the assembly of OPTEST, ready the cassette by placing the OPTEST source cassette into the cassette 1 transport, rewind the cassette, and press the LOAD/FF switch after the cassette has stopped. Also, place a cassette into the cassette 2 transport to record the assembled object, readying it like the source. The ASR output is shown in figure 4-6. Figure 4-7 shows the assembler output.

4.2.2 **LOADING AND OPERATING OPTEST.** The general outline for exercising the paper tape unit using OPTEST will be (1) to demonstrate the error codes; (2) to examine the output buffer for write ASCII and Direct, and to modify the input buffer to compare changes to that buffer after the paper tape is read back into memory; (3) to punch paper tape both for ASCII and Direct; (4) to read back into memory the paper tape that was produced from (3); and (5) to examine the buffer changes that were produced by reading the ASCII paper tape and the Direct paper tape.

To load OPTEST, take the object cassette produced by the assembler, put it into the cassette 1 transport, and ready the tape as before. Before executing the program, LUNO 9 must be assigned to the paper tape reader (PTR) and LUNO A to the paper tape punch (PTP). The ASR output is shown in figure 4-8.

4.2.2.1 **Error Codes.** Error code 5 indicates the device is not open, and error code 2 indicates an illegal opcode. When the system and user's flags are output, the rightmost value will be the error code and the leftmost value will indicate an error to the monitor. In this sample case, the error should be a 4, which is an unrecoverable I/O error to the monitor. For error code 5, try to punch tape (BP) before the punch is open, and for error 2, try to read from the punch (9P). Note that the teletypewriter interface must be at CRU base 20_{16}, which is the left half of computer chassis slot 6 at the bottom of the chassis. If a response is not obtained, try turning the teletypewriter on and off; or feeding some paper tape through the reader on local; or turn the computer off, reseat the interface card, reboot the monitor, and start over from paragraph 4.2.2. The ASR output is shown in figure 4-9.

4.2.2.2 **Examine and Modify Buffers.** Now look at the monitor output buffer and modify the monitor input buffers for read ASCII and Direct by making these two buffers all zeros. The output buffer is from 282 to 294; the ASCII input buffer is from 296 to 2A8; and the Direct input buffer is from 2AA to 2BE. The IM and MM commands are used to perform these operations. The ASR output is shown in figure 4-10.

```
(ASSEMBLE THE DEMONSTRATION PROGRAM)

.PA    (<CR>)
PXRASM  948925 ** 12JUL76
PREDEFINED REGISTERS? N    (<CR>)

ASM/TERM? A    (<CR>)


(A)134552
```

Figure 4-6. Assemble Demonstration Program

```
                                                        PAGE 0001

0001                        IDT  'OPTEST'
0002                    *
0003                    * REGISTER EQUATES
0004                    *
0005          0000      R0    EQU  0
0006          0001      R1    EQU  1
0007          0002      R2    EQU  2
0008          0003      R3    EQU  3
0009          0004      R4    EQU  4
0010          000B      R11   EQU  11
0011                    *
0012  0000              WSP   BSS  32                WORK SPACE REGISTERS
0013  0020  02E0        START LWPI WSP               GET WORK SPACE
      0022  0000'
0014  0024  2FE0        AGAIN XOP  @HEAD,15          INPUT S,E,O FOE STOP, ERR PRO
      0026  ----
0015  0028  2FE0              XOP  @REPLY,15         OR OP CODE PRO
      002A  ----
0016                    *
0017                    * DECISION BRANCH R2=LUNO; R3= OP CODE
0018                    *
0019  002C  C060              MOV  @INP,R1           GET INPUT
      002E  ----
0020  0030  0241              ANDI R1,>7F00          MASK OFF GARBAGE
      0032  7F00
0021  0034  0221              AI   R1,->4500         IS IT AN E?
      0036  BB00
0022  0038  13--              JEQ  ERRPRO            YES, ERROR PROCEDURE
0023  003A  0221              AI   R1,->4F00         IS IT AN O?
      003C  F600
0024  003E  13--              JEQ  OPCPRO            YES, OP CODE PROCEDURE
0025  0040  0221              AI   R1,->5300         IS IT AN S?
      0042  FC00
0026  0044  16EF              JNE  AGAIN             NO, PROMPTAGAIN
0027                    *
0028                    * RETURN TO MONITOR
0029                    *
0030  0046  2FE0              XOP  @RET,15           RETURN
      0048  ----
0031                    *
0032                    *  ERROR PROCEDURE
0033                    *
0034                    ERRPRO EVEN
      0038**1308
0035  004A  2FE0              XOP  @ASKUP,15         OUTPUT PROMPT
      004C  ----
0036  004E  2FE0              XOP  @INPUT,15         INPUT REPLY
      0050  ----
0037  0052  D820              MOVB @INBUF,@VHL       SET UP FOR CONVERSION
      0054  ----
      0056  ----
0038  0058  2FE0              XOP  @CONBI,15         CONVERT
      005A  ----
0039  005C  06C0              SWPB R0                RESULTS IN LEFT BYTE
0040  005E  D800              MOVB R0,@IOUP          DEPOSIT FOR PROCESS


(A)134553  (1/5)
```

Figure 4-7. IDT OPTEST (Sheet 1 of 5)

```
                                                                PAGE 0002

      0060  ----
0041  0062  06A0          BL    @TESTUT             PROCESS CALL
      0064  ----
0042  0066  04E0          CLR   @SYSFL              CLEAR ALL ERR FLAGS
      0068  ----
0043  006A  10DC  OVER    JMP   AGAIN               DO IT AGAIN
0044                    *
0045                    * OP CODE PROCEDURE
0046                    *
0047                    OPCPRO EVEN
      003E**1316
0048  006C  2FE0          XOP   @ASKOP,15           OUTPUT PROMPT
      006E  ----
0049  0070  2FE0          XOP   @INPUT,15           GET REPLY
      0072  ----
0050  0074  D820          MOVB  @INBUF,@VHL         SET UP FOR CONVERSION
      0076  ----
      0078  ----
0051  007A  2FE0          XOP   @CUMBI,15           CONVERT
      007C  ----
0052  007E  06C0          SWPB  R0                  PUT IN LEFT BYTE
0053  0080  C0C0          MOV   R0,R3               GET REPLY VALUE
0054  0082  0243          ANDI  R3,>7F00            FIRST REPLY ONLY
      0084  7F00
0055  0086  C0A0          MOV   @INBUF,R2
      0088  ----
0056  008A  0242          ANDI  R2,>7F              SECOND REPLY ONLY
      008C  007F
0057  008E  0282          CI    R2,>50              IS IT PUNCH?
      0090  0050
0058  0092  16--          JNE   REDTES              NO, CHECK ASCII OR DIRECT
0059  0094  D800          MOVB  R0,@IOUP            DEPOSIT RESULTS
      0096  ----
0060  0098  06A0          BL    @TESTUT             PROCESS CALL
      009A  ----
0061  009C  10C3          JMP   AGAIN               DO IT AGAIN
0062                    *
0063                    *  CHECK READ ASCII OR DIRECT
0064                    *
0065                    REDTES EVEN
      0092**1605
0066  009E  0282          CI    R2,>52              IS IT READ?
      00A0  0052
0067  00A2  16C0          JNE   AGAIN               NO, BAD START
0068  00A4  0283          CI    R3,>8000            IS IT READ DIRECT?
      00A6  8000
0069  00A8  16--          JNE   WRTASC              CHECK READ ASCII
0070  00AA  2FE0          XOP   @REDDIR,15          PROCESS IT
      00AC  ----
0071  00AE  10DD          JMP   OVER                DO IT AGAIN
0072  00B0  0283  WRTASC  CI    R3,>9000            IS IT READ ASCII?
      00B2  9000
      00A8**1603
0073  00B4  16--          JNE   REDOPN              IS IT OPEN READER?
0074  00B6  2FE0          XOP   @REDASC,15          PROCESS IT
```

(A)134553 (2/5)

Figure 4-7. IDT OPTEST (Sheet 2 of 5)

```
                                                                PAGE 0003

      00B8  ----
0075  00BA  10D7          JMP   OVER                DO IT AGAIN
0076                    *
0077  00BC  0283  REDOPN CI    R3,>0                 IS OP CODE OPEN VALUE?
      00BE  0000
      00B4**1603
0078  00C0  16D4          JNE   OVER                NO, BAD START
0079  00C2  2FE0          XOP   @OPNRED,15          YES, OPEN READER
      00C4  ----
0080  00C6  10D1          JMP   OVER                PROMPT AGAIN
0081                    *
0082                    * ALL PURPOSE XOP
0083                    *
0084                    TESTOT EVEN
      0064**00C8'
      009A**00C8'
0085  00C8  2FE0          XOP   @SUPCL,15           PROCESS CALL
      00CA  ----
0086  00CC  C820          MOV   @CHRLIN,@OUTBUF     GET CR, LF
      00CE  ----
      00D0  ----
0087  00D2  04E0          CLR   @OUT2
      00D4  ----
0088  00D6  2FE0          XOP   @OUTRES,15          OUTPUT CR, LF
      00D8  ----
0089  00DA  C020          MOV   @SYSFL,R0           FLAGS IN R0
      00DC  ----
0090  00DE  2FE0          XOP   @CONHS,15           CONVERT TO HSCII
      00E0  ----
0091  00E2  2FE0          XOP   @OUTRES,15          OUTPUT RESULTS
      00E4  ----
0092  00E6  045B          B     *R11                RETURN
0093                    *
0094                    * PRB DATA TABLES
0095                    *
0096                    HEAD  EVEN
      0026**00E8'
0097  00E8  0000          DATA  0                   OUTPUT MESS TO LOG
0098  00EA  0B00          DATA  >B00
0099  00EC  0000          DATA  0
0100  00EE  ----          DATA  HEADER
0101  00F0  0000          DATA  0
0102  00F2  004C          DATA  76
0103                    *
0104  00F4  0000  ASKOP   DATA  0                   OUTPUT PROMPT MESS
      004C**00F4'
      006E**00F4'
0105  00F6  0B00          DATA  >B00
0106  00F8  0000          DATA  0
0107  00FA  ----          DATA  OUTMES
0108  00FC  0000          DATA  0
0109  00FE  0034          DATA  52
0110                    *
0111  0100  0000  INPCD   DATA  0                   REPLY
      0050**0100'


(A)134553 (3/5)
```

Figure 4-7. IDT OPTEST (Sheet 3 of 5)

```
                                                                PAGE 0004

      0072**0100'
0112  0102  0900          DATA >900
0113  0104  0000          DATA 0
0114  0106  ----          DATA INBUF
0115  0108  0002          DATA 2
0116  010A  0000          DATA 0
0117                      *
0118  010C  0D00  COMBI    DATA >0D00              CONVERT REPLY TO FORM
      005A**010C'
      007C**010C'
0119  010E    30          TEXT '000'
0120  0111    30  VAL     BYTE >30
      0056**0111'
      0078**0111'
0121                      *
0122                      CONAS   EVEN
      00E0**0112'
0123  0112  0C00          DATA >C00
0124  0114  0000  OUTBUF  DATA 0
      00D0**0114'
0125  0116  0000  OUT2    DATA 0
      00D4**0116'
0126                      *
0127                      OUTRES  EVEN
      00D8**0118'
      00E4**0118'
0128  0118  0000          DATA 0
0129  011A  0B00          DATA >B00
0130  011C  0000          DATA 0
0131  011E  0114'         DATA OUTBUF
0132  0120  0000          DATA 0
0133  0122  0004          DATA 4
0134                      *
0135  0124  0000  SUPCL    DATA 0                  MAIN CALL TO R/P
      00CA**0124'
0136  0126    00  LOOP    BYTE 0
0137  0127    0A  LUNO    BYTE >A
      0060**0126'
      0096**0126'
0138  0128    00  SYSFL   BYTE 0
0139  0129    00  USERFL  BYTE 0
      0068**0128'
      00DC**0128'
0140  012A  ----  BUFAD   DATA BUFFR1
0141  012C  0000  BUFLIN  DATA 0
0142  012E  0014  CHRCT   DATA 20
0143                      *
0144                      READSC  EVEN            READ ASCII
      00B8**0130'
0145  0130  0000          DATA 0
0146  0132  0909  IOOPA   DATA >909
0147  0134  0000          DATA 0
0148  0136  ----          DATA BUFFR2
0149  0138  0014          DATA 20
0150  013A  0000          DATA 0

(A)134553 (4/5)
```

Figure 4-7. IDT OPTEST (Sheet 4 of 5)

```
0151                    *                                       PAGE 0005
0152  013C  0000  OPNRED DATA 0
      00C4**013C'
0153  013E  0009        DATA >009                 OPEN READER
0154  0140  0000        DATA 0,0,0
      0142  0000
      0144  0000
0155                    *
0156  0146  0000  REDDIR DATA 0                   READ DIRECT
      00AC**0146'
0157  0148  0A09  IOOPD  DATA >A09
0158  014A  0000        DATA 0
0159  014C  ----        DATA BUFFR3
0160  014E  0014        DATA 20
0161  0150  0000        DATA 0
0162                    *
0163                    RET     EVEN
      0048**0152'
0164  0152  0400        DATA >400
0165                    *
0166  0154  0000  REPLY  DATA 0                   BRANCH INPUT
      002A**0154'
0167  0156  0900        DATA >900
0168  0158  0000        DATA 0
0169  015A  ----        DATA IMP
0170  015C  0001        DATA 1
0171  015E  0000        DATA 0
0172                    *
0173                    * PAB STORAGE OUTPUT AREA
0174                    *
0175  0160  0A0D  HEADER DATA >A0D
      00EE**0160'
0176  0162    49        TEXT 'INPUT S FOR STOP OR E FOR ERROR PROCEDURE'
0177  018C  0A0D        DATA >A0D
0178  018E    4F        TEXT 'OR U FOR UP CODE PROCEDURE'
0179  01A8  0ACD  CARLIN DATA >A0D
      00CE**01A8'
0180                    *
0181  01AA  0A0D  OUTMES DATA >A0D
      00FA**01AA'
0182  01AC    45        TEXT 'ENTER DESIRED OP-CODE AND MODE (R=READ,'
0183  01D3    20        TEXT ' P=PUNCH'
0184  01DC  0A0D        DATA >A0D
0185                    *
0186  01DE  0000  IMP     DATA 0
      002E**01DE'
      015A**01DE'
0187  01E0  0000  INBUF   DATA 0
      0054**01E0'
      0076**01E0'
      0088**01E0'
      0106**01E0'
0188  01E2    20  BUFFR1 TEXT ' !0123HBCD'
      012A**01E2'
0189  01EC  7E7F        DATA >7E7F,>0809,>070A,>0102,>2020
      01EE  0809
      01F0  070A
      01F2  0102
      01F4  2020
0190  01F6        BUFFR2 BSS  20
      0136**01F6'
0191  020A        BUFFR3 BSS  20
      014C**020A'
0192                    END STAR1

0000 ERRORS

ASM/TERM? T   (<CR>)
(A)134553 (5/5)
```

Figure 4-7. IDT OPTEST (Sheet 5 of 5)

```
(LOAD PROGRAM AND EXHIBIT ERROR CODES; 5= DEVICE NOT OPEN
 2= ILLEGAL OP CODE)

.LP    (<CR>)
.AL,9,PTR    (<CR>)
.AL,A,PTP    (<CR>)
```

(A)134554

Figure 4-8. Load Program

```
.EX    (<CR>)
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
E
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
BP
4005
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
E
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
9P
4002
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
S
.
```

(A)134555

Figure 4-9. Error Codes

```
.IM,282,294    (<CR>)
0282=2021  3031  3233  4142  >4344  7E7F  0809  070A
0292=0102  2020
.MM,296    (<CR>)
0296=0000    (<SPACE BAR>) (ALL LOCATIONS SHOULD BE ZERO)
0298=0000    (<SPACE BAR>)
029A=0000    (<SPACE BAR>)
029C=0000    (<SPACE BAR>)
029E=0000    (<SPACE BAR>)
02A0=0000    (<SPACE BAR>)
02A2=0000    (<SPACE BAR>)
02A4=0000    (<SPACE BAR>)
02A6=0000    (<SPACE BAR>)
02A8=0000    (<SPACE BAR>)
02AA=0000    (<SPACE BAR>)
02AC=0000    (<SPACE BAR>)
02AE=0000    (<SPACE BAR>)
02B0=0000    (<SPACE BAR>)
02B2=0000    (<SPACE BAR>)
02B4=0000    (<SPACE BAR>)
02B6=0000    (<SPACE BAR>)
02B8=0000    (<SPACE BAR>)
02BA=0000    (<SPACE BAR>)
02BC=0000    (<SPACE BAR>)
02BE=0000    (<CR>)
```

(A)134556

Figure 4-10. Examine and Modify Buffers

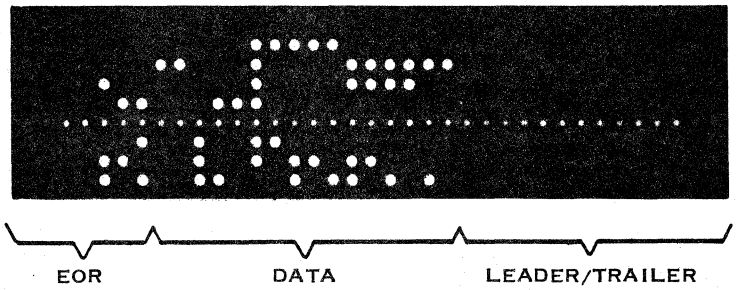
4.2.2.3 Punch Paper Tape. The program will be executed taking "O" option to the initial prompt. Before the user inputs any response to the second prompt, the teletypewriter should be placed in local mode, the HERE-IS key for leader/trailer should be pressed twice, and the teletypewriter should be placed back in the online mode. The first response to the second prompt will be to open the punch (OP). Note that the bell will ring twice alerting the user to turn on the punch and press the RUB OUT key. No indication of the RUB OUT will be punched on the tape because it is not echoed. The next two responses will be to punch ASCII (BP) and to punch Direct (CP). The tape should be similar to figures 4-11 and 4-12. The ASR output is shown in figure 4-13.

Before removing the paper tape just produced, place the TTY in local mode and press the HERE-IS key twice to generate several inches of trailer. Opcodes are available which generate leader/trailer but are not used in this example.

4.2.2.4 Read Paper Tape. The paper tape produced from paragraph 4.2.2.3 is to be used in this example. Again, the "O" option should be taken, an open reader (OR) done, a read ASCII (9R) done, a read Direct (AR) done, and then the "S" option taken to return to the monitor. The ASR output is shown in figure 4-14.

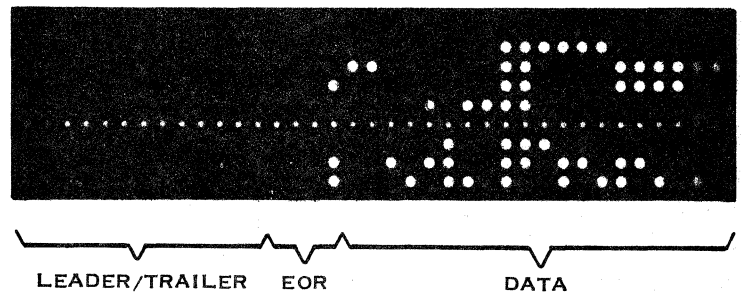
4.2.2.5 Examine Buffer Change. An examination of what was read off the paper tape and into memory, from paragraph 4.2.2.4, will be performed by doing an IM command of the ASCII input buffer and the Direct input buffer. The ASR output is shown in figure 4-15.

The previous exercise read an ASCII tape into a buffer to demonstrate how read ASCII (BP) and read Direct works. The user can, for further experimentation, read the Direct tape, using a read ASCII (BP). This will demonstrate that certain characters on the Direct tape are not legal ASCII values and are ignored in accordance with the file and data formats of Appendix A.



(A)134560

Figure 4-11. ASCII Tape



(A)134561

Figure 4-12. Direct Tape

```
.EX    (<CR>)
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
OP
0000
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
BP
0000
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
CP
0000
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
```

(A)134557

Figure 4-13. Punch Paper Tape

```
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
OR
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
9R
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
O
ENTER DESIRED OP-CODE AND MODE (R=READ, P=PUNCH)
AR
INPUT S FOR STOP OR E FOR ERROR PROCEDURE
 OR O FOR OP CODE PROCEDURE
S
```

(A)134558

Figure 4-14. Read Paper Tape

```
(EXHIBIT READ ASCII AND DIRECT BUFFERS, VERIFY RESULTS)

.IM,296,2A8    (<CR>)
0296=2021  3031  3233  4142  >4344  /E08  0907  2020
02A6=0000  0000
.IM,2AA,2BC    (<CR>)
02AA=2021  3031  3233  4142  >4344  /E/F  0809  070A
02BA=0102  2020
```

(A)134559

Figure 4-15. Exhibit Read ASCII and Direct Buffers, Verify Results

4.3 SAMPLE PROGRAM 5, BNPF USE WITH PAPER TAPE

This procedure illustrates use of the BNPF overlay to dump and load data to and from paper tape in BNPF format. Since the BNPF utility always dumps to LUNO 7 (normally cassette drive 1), reassign LUNO 7 to the paper tape punch for dumps to paper tape or to paper tape reader for loads (or compares) from paper tape. See figure 4-16. A reproduction of the resultant tape can be seen in figure 4-17. A teletypewriter listing can be seen in figure 4-18.

```
(HERE WE LOAD IN THE BNPF OVERLAY)

.OV  (<CR>)
DB
.

(NOW SET UP THE TELETYPE; BE SURE THE INTERFACE CARD IS IN CHASSIS SLOT 20
  IN LOCAL MODE, PUNCH OUT SOME LEADER ON THE PAPER TAPE BY USING THE
  TELETYPE'S <HERE IS> KEY.  THEN PUT THE TELETYPE ON-LINE AND PRESS THE
  PUNCH ON SWITCH.)

.AL,7,PIP  (<CR>)
.(THIS IS NECESSARY SINCE THE BNPF OVERLAY NORMALLY WORKS OFF
   THE LEFT HAND CASSETTE DRIVE)

.MM,500 (<CR>)
0500=0000  3031    (<SPACE BAR>)
0502=0000  3233      "
0504=0000  3435      "
0506=0000  3637      "
0508=0000  3839      "
050A=0000  4243      "
050C=0000  4445      "
050E=0000  4647    (<CR>)
.

(WE NOW DUMP THIS AREA OF MEMORY TO PAPER TAPE IN BNPF FORMAT)

.DB,D,500,50F  (<CR>)

(THE TTY BELL SHOULD RING; WHEN IT DOES, PRESS THE PUNCH-ON AND THE
  TTY <RUBOUT> KEY TO INITIATE PUNCHOUT)

(WHEN PUNCHING IS COMPLETE, PUT THE TTY MOMENTARILY IN LOCAL MODE TO
  PUNCH A TRAILER OF BLANK FRAMES- USE THE <HERE IS> KEY)

(NOW WE CHANGE THE MEMORY AREA TO SEE IF THE TAPE'S DATA CAN THEN BE
  LOADED BACK IN)

.MM,500 (<CR>)
0500=3031  FFFF    (<SPACE BAR>)
0502=3233  FFFF      "
0504=3435  FFFF      "
0506=3637  FFFF      "
0508=3839  FFFF      "
050A=4243  FFFF      "
050C=4445  FFFF      "
050E=4647  FFFF    (<CR>)
.

(NOW THE TAPE IS LOADED BACK IN; LUNO 7 MUST BE ASSIGNED TO THE
 THE READER, AND THE PAPER TAPE SHOULD BE POSITIONED IN THE READER
  ON THE LEADING BLANK FRAMES.)

.AL,7,PTR   (<CR>)
.DB,L       (<CR>)
BEG ADDR=0500
END ADDR=050F
.

(NOW WE CHECK IF THE DATA HAS ACTUALLY LOADED IN)

.IM,500,50F  (<CR>)
0500=3031  3233  3435  3637  >3839  4243  4445  4647
.
 (SUCCESS!!)

(IF THE USER WISHES, HE CAN LIST THE PAPER TAPE IN LOCAL MODE
 ON THE TELETYPE)

(A)134562
```

Figure 4-16. Use of BNPF with Paper Tape

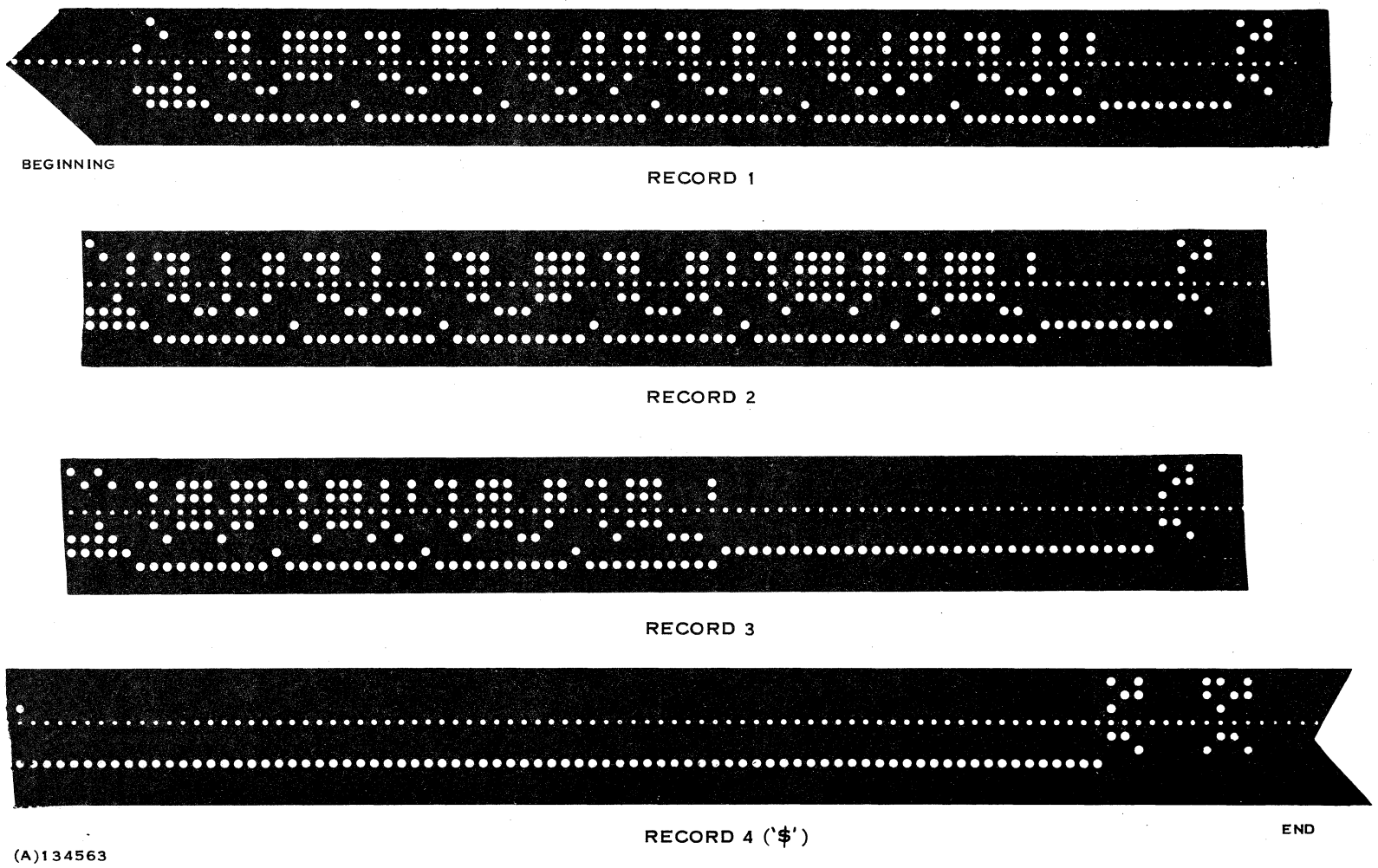


Figure 4-17. Paper Tape Records, BNPF Format

```
1280 BNNPPNNNNF BNNPPNNNPF BNNPPNNPNF BNNPPNNPPF BNNPPNPNNF BNNPPNPNPF
1286 BNNPPNPPNF BNNPPNPPPF BNNPPPNNNF BNNPPPNNPF BNPNNNNPNF BNPNNNNPPF
1292 BNPNNNPNNF BNPNNNPNPF BNPNNNPPNF BNPNNNPPPF
$

(A)134564
```

Figure 4-18. Teletypewriter Listing

APPENDIX A

PAPER TAPE READER/PUNCH FILE AND DATA FORMATS

APPENDIX A

PAPER TAPE READER/PUNCH FILE AND DATA FORMATS

ASR 33

PERIPHERAL: ASR 33 TTY Paper Tape Reader (ASCII, Direct)

PHYSICAL ORGANIZATION: Record, File

END OF RECORD: CR for ASCII, DC3 for Direct

END OF FILE: DC3 for ASCII, does not apply for Direct

CHARACTER SET: As shown in figure A-1

1. HT, FF, BEL, BS, and characters in the range >20 to >7F are stored in the user's buffer.

2. ETB is translated to CR and stored in the user's buffer.

3. CR indicates end of record and is not stored in the user's buffer.

4. DC3 received as the first valid character of a record indicates end of file and is not placed in the user's buffer.

5. The sequence LF, or DEL or LF,DEL or any number of nulls at the beginning of a record are ignored if present. The first character following such a sequence is considered the first valid character in the record.

6. In direct mode, the contents of a physical block on tape is transferred to the buffer without conversion.

b7 → / b6 → / b5 → / BITS					0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL → / ROW ↓	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	|
1	1	0	1	13	CR	GS	−	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	—	o	DEL

(A)134565

Figure A-1. Paper Tape Reader Character Set

ASR 33

PERIPHERAL: ASR 33 TTY Paper Tape Punch (ASCII, Direct)

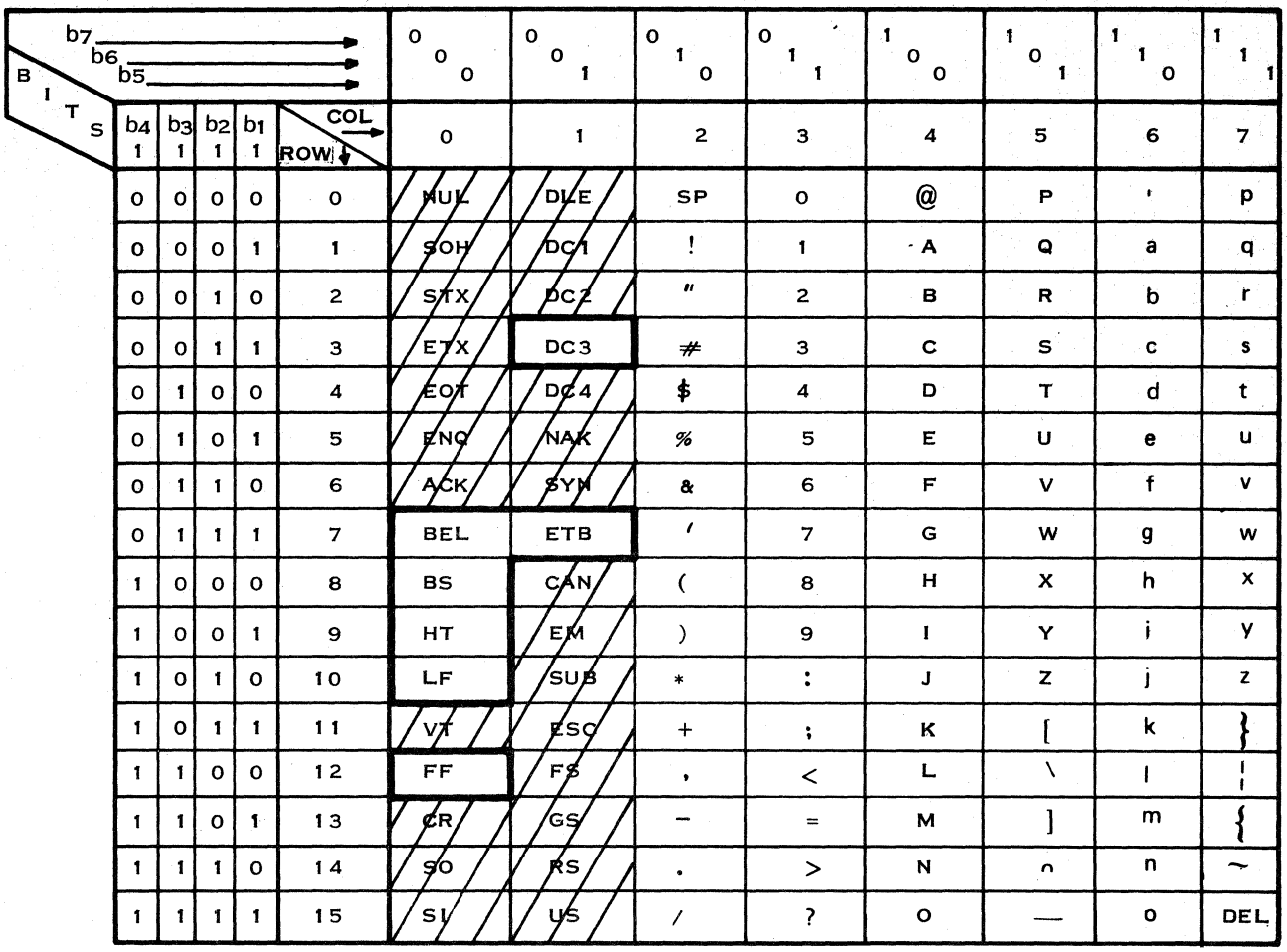
PHYSICAL ORGANIZATION: Character, Record, File

END OF RECORD: Depletion of Character Count

END OF FILE: DC3 for ASCII, does not apply for Direct

CHARACTER SET: As shown in figure A-2

1. HT, FF, BEL, BS, and characters in the range 20_{16} to $7F_{16}$ are output as is.
2. CR in the user's buffer is translated to ETB and output.
3. The end of record character sequence is CR, LF, DC3, NULL, NULL, NULL, NULL. These characters are automatically output to control the paper tape and are not user data characters.
4. The end of file character sequence is DC3, CR, LF, DC3, NULL, NULL, NULL.
5. DC3 is allowed within a record for compatibility with standalone software. It may not be written, however, as the first data character in the record.
6. In direct mode:
   - Data is output unconverted; entire ASCII set is valid.
   - The end of record character sequence is DC3, NULL, NULL, NULL.
   - If DC4 is encountered in the user's buffer, it is output and record is turned on again by sending DC2.
   - The user is responsible for ensuring that last block is dumped to cage.
   - The end of record character sequence is output at the end of each write operation by the DSR.



BITS b7 b6 b5 →					0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4	b3	b2	b1	ROW ↓ COL →	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	|
1	1	0	1	13	CR	GS	-	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

(A)134566

Figure A-2. Paper Tape Punch Character Set

ALPHABETICAL INDEX

ALPHABETICAL INDEX

INTRODUCTION

The following index lists key words and concepts from the subject material of the manual together with the area(s) in the manual that supply major coverage of the listed concept. The numbers along the right side of the listing reference the following manual areas:

- Sections - References to Sections of the manual appear as "Section x" with the symbol x representing any numeric quantity.
- Appendixes - References to Appendixes of the manual appear as "Appendix y" with the symbol y representing any capital letter.
- Paragraphs - References to paragraphs of the manual appear as a series of alphanumeric or numeric characters punctuated with decimal points. Only the first character of the string may be a letter; all subsequent characters are numbers. The first character refers to the section or appendix of the manual in which the paragraph is found.
- Tables - References to tables in the manual are represented by the capital letter T followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the table). The second character is followed by a dash (-) and a number:

  Tx-yy

- Figures - References to figures in the manual are represented by the capital letter F followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the figure). The second character is followed by a dash (-) and a number:

  Fx-yy

- Other entries in the Index - References to other entries in the index are preceded by the word "See" followed by the referenced entry.

USER'S RESPONSE SHEET

Manual Title: Model 990 Computer Prototyping Monitor (PXRATE)
System Operation Guide (946243-9701)

Manual Date: 1 January 1977 **Date of This Letter:** _____

User's Name: _____ **Telephone:** _____

Company: _____ **Office/Department:** _____

Street Address: _____

City/State/Zip Code: _____

Please list any discrepancy found in this manual by page, paragraph, figure, or table number in the following space. If there are any other suggestions that you wish to make, feel free to include them. Thank you.

Location in Manual	Comment/Suggestion

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), TAPE AND MAIL

FOLD

FIRST CLASS
PERMIT NO. 7284
DALLAS, TEXAS

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

TEXAS INSTRUMENTS INCORPORATED
DIGITAL SYSTEMS DIVISION

P.O. BOX 2909 · AUSTIN, TEXAS 78769

ATTN: TECHNICAL PUBLICATIONS
MS 2146

FOLD

TEXAS INSTRUMENTS
INCORPORATED
DIGITAL SYSTEMS DIVISION
POST OFFICE BOX 2909 AUSTIN, TEXAS 78769