RPC 4000

GENERAL PRECISION, INC. / COMMERCIAL COMPUTER DIVISION

PROXY

PHOTO-READER ORIENTED HEX SYSTEM

J4-284-4

Disclaimer:

"The authors of this program material, the POOL organization and General Precision believe this program to be correct; however, they bear no responsibility, financial or otherwise, for errors resulting from its use. This program is distributed only to individual and installation members of POOL. Further distribution of this manual and accompanying tapes for use by non-members is prohibited."

SC 4125

PRINTED IN U.S.A.

Title: PROXY (Photo-Reader Oriented heX sYstem)

Author: H. J. Bowlden

Installation: Parma Research Laboratory
Union Carbide Corporation
Parma 30, Ohio

Date: November 5, 1962

Classification: J4-284-4

ABSTRACT

PROXY is a hexadecimal system for the RPC-4000 including a dumper (output routine), verifier, loader and monitor-search feature. It is designed to aid in efficient utilization of the photo-electric reader. Program loading times with the photo-reader vary (depending on the material being loaded) from 1/16 second to a maximum of less than 1.5 seconds per track (or 0.5 to less than 10 seconds on the standard 60 character-per-second reader).

TABLE OF CONTENTS

PROXY

INTRODUCTION

Description of the System

The PROXY system is designed specifically to improve the utilization of the photo-electric reader on the RPC-4000 system. The needs of users without a photo-reader have, however, also been considered, and such users will find the advantages of the system not outweighed by any disadvantages.

The heart of the system is a one-track program loader occupying track 126 which reads hex words of up to 16 characters in a special interlace pattern designed to insure minimum memory access time. This loader requires, for each track loaded, a code word containing the track at a q of 11 and repeated at a q of 29. These two copies are compared before the track is loaded to prevent erroneous loading due to incorrect reading of the code. A checksum is read and compared with memory for each track loaded. A hex transfer may be placed after any track, and may be executed with or without a stop as desired. If all sectors of a track contain the same value, they are loaded by a special format consisting of one 15-character word containing the loading code (as above) and the value to be loaded, followed by a word of 8-characters or less giving the checksum. This loader is loaded by a checksumming bootstrap, and will stop if it is not loaded correctly.

One of the two available formats, referred to below as type B or strip format, consists of the above loader followed by tracks of data for loading, followed by a transfer word. Each track is preceded by a one-inch gap and an end-of-message code. This format is recommended for loading on the brush reader, or using strip mode on the photo-reader.

Tapes prepared in the format referred to below as type A or reel format contain, following the loader, a search-monitor routine to be stored by the loader in track 125. This is followed by a nonstop transfer to the beginning of the search routine at 12500. Next on the tape is a special code word containing the "key" (a character labeling the program for use by the search routine). This code word is designed so that it may be read equally well in either direction.

The tracks of data in reel format are not separated by end-of-message codes; although errors in loading codes or checksums are detected at each track, the tape is rewound by the monitor to the last key code to reload. These keys may be inserted at the user's option; it is necessary to compromise between the added searching time introduced by each unnecessary end-of-message code (due mainly to the 300 millisecond delay built into the photo-reader) and the loss of time in rewinding in case of an error.

The search routine should only be allowed to take full control for tapes mounted on reels, because of the inherent dangers of jams in bi-directional tape motion in strip mode. The operation of the search routine may be suppressed, for loading in strip mode or on the brush reader, by depressing sense switch 1.

The search-monitor routine adds the following features. A reel-mounted tape containing a number of separate programs, each with a distinctive "key" character, may be started at any load point (that is, before any program). Once the search routine has asked for, and been given, a "key," it searches the tape at high speed and automatically loads the requested program. If no such key is included on the reel, an indication is given. Errors in reading a loading code or checksum cause the tape to be automatically rewound to the most recently detected key code and reloaded. Four attempts are made on any given block, with "NO" being typed each time. Since each block (including sub-blocks of a single program file, separated by intermediate key codes) has its own distinctive key code word, the monitor cannot "get lost" if a stop-code is missed and it goes into the following block. By ending each file with a nonstop transfer back to the monitor, a user may, if it is desirable, provide for the automatic loading of several program files from a tape reel under control of a short tape placed in the 4500 reader. In this case, the desired hex transfer (either stop or nonstop) to start program execution may be placed on this control tape.

Memory Requirements

Punch/verify routine	-	9 tracks, plus track 125.
The relocatable version has its bootstrap on track 126.		
Loader	-	track 126
Monitor	-	track 125

Operating Times

Punching:	30 char/sec. punch	- 1-10 sec. per track
	high-speed punch	- 0.6-3.75 sec. per track
Loading:	60 char/sec. reader	- 0.5-5 sec. per track
	photo-electric reader	- 0.2-1.5 sec. per track
Searching:	500 char/sec., plus about 0.3 sec. at each end-of-message code.	
Verification:	60 char/sec. reader	- 1.2-6 sec. per track
	photo-electric reader	- 0.8-2.6 sec. per track

Sense Switches

1: punch and verify control data read from 4500 reader if on; from typewriter if off (loading reel format tapes: ignore search mode if on).

2: punch/verify control requests printed on typewriter if off; suppressed if on. If SS1 is on, control data copied from reader to typewriter only if SS2 off. (loading reel format tapes: read key code from 4500 reader).

4: off = punch, on = verify.

8, 16, 32: not used.

NOTE: 1 and 2 not used when loading strip format tapes; 4 not used when loading any tapes.

Operating Procedures: Punching and Verifying

At punch/verify time, communications consist of answers to typed requests. The possible answers and results are indicated in Table I.

Since the monitor is stored on track 125, it must be reloaded after any program tape using track 125 for its own bootstrap.

The sequence for punching and verifying could then be as follows:

a) Load program tape (s) (preferably after clearing appropriate portion of memory to zero).

b) Load PROXY. The relocatable version requires any nine adjacent tracks, plus tracks 125 and 126.

c) Sense Switches off. Start Compute, and answer questions as in Table I, ending with PAUSE* or FINAL*.

d) Tear off tape, roll up and place in reader. Sense switch 4 on.

e) Start Compute, and answer all questions exactly as in (c), except that the first question will be INPUT SEL. instead of PUNCH SEL.

TABLE I

PUNCH/VERIFY

STEP	REQUEST	ANSWER	RESULT	THEN GO TO	COMMENTS
1	PUNCH SEL. (if S.SW.4 off)	45*	Select Output on 4500 Punch	2	
		46*	Select Output on 4600 Punch	2	
		HSP*	Select Output on high speed punch	2	
		(Any other legal selection code, except 101)		2	
2	FILE KEY	*	Punch 30-inch leader and loader, and set up for strip mode tape	3	
		C*	-punch 30-inch leader, loader, monitor, and key code word	3	C stands for any acceptable key character.
		FIRST*	Punch 128-inch leader and special "first" file	2	
		FINAL*	Punch special "final" file and 128-inch trailer, then deselect and stop	1 or 6	1 if s.sw.4 off
		PAUSE*	deselect and stop	1 or 6	6 if s.sw.4 on
3	FIRST TRACK (track)*		--	4	
4	FINAL TRACK (track)*		--	5	
5	TRANS ADDR.	*	Punch block without transfer	3	
		(dec.addr.)*	Punch block and stop-and-transfer to this addr.	2	
		T (dec.addr.)*	Punch block and nonstop transfer to this addr.	2	
		SEARCH*	Punch block and nonstop transfer to monitor (12500)	2	Reel format only
		REPEAT*	Punch block and nonstop transfer to "repeat" routine, check gap and key code	3	Reel format only
6	INPUT SEL.	45*	Select verify input on 4500 rdr.	2	
		46*	Select verify input on 4600 rdr.	2	
		PR*	Select verify input on photoreader	2	
		(or any legitimate selection code except 101) (When verifying, sections 2-5 are to be interpreted as saying the corresponding material is read and checked.)			

Error Tests: Punching

If the final track is smaller than the initial track (note that only the last three digits are used, and they are interpreted modulo 128), control returns to step 3 of Table I from step 4. Also, if the final track is greater than 122 (in reel format) or 123 (in strip format) control returns to step 3 from step 4.

In step 5, strip format, a SEARCH or REPEAT will be rejected, causing step 5 to be repeated.

An invalid key in step 2 causes step 2 to be repeated. Note that only the last character is considered, and this must be a digit (1-9) or a letter (A-Z).

Error Display: Verifying

In the tape does not check correctly, a printout is given of the actual contents of the word on tape (up to 16 characters; the printout adds leading zeros and a space to divide it into two 8-character segments for legibility) followed by the correct form of the same information. This line might have the form (for example) TAPE= A0B785F9 DDF79504, MEM=A0BF85F9 DDF79504. A stop occurs, at which the tape could be marked, if desired, for off-line correction. Pressing Start Compute will continue checking.

Operating Procedure: Loading a Strip Format Tape

Place tape in reader and execute standard bootstrapping sequence. An error in reading a loading code or in a checksum will cause the routine to print NO and stop. Move the tape back to a checkgap (1-inch gap) and press Start Compute to try again. If tape is just beyond a gap, it would be advisable to move it back one extra. Each gap has an end-of-message code, and the backing up may be done on the photo-reader using off-line search. Since the loader is checksummed, an error may occur at the first block. In this case, start over.

Operating Procedure: Loading a Reel Format Tape

a) In 4500 reader or strip mode: depress switch 1 and proceed as for strip format tapes. Do not attempt to load FIRST or FINAL files, or files with a SEARCH transfer. If you forget to depress sw. 1, answer the "K" request with a slash (/) and depress sw. 1. If an error is detected "NO" is printed. To try again, move the tape back (manually; there is no e.o.m. code) to the last one-inch gap and press "start compute."

b) To load a file from a reel, place tape in reader at any load point (beginning of tape, or any 30-inch file gap). Sense switches 1 and 2 off. Execute normal bootstrap sequence. When computer types "K", type in key code of desired file, and stop code. The code "0" (zero) will cause the tape to rewind to the beginning (for unloading). The code "/" will cause loading of "this" file. If the requested file is not on the tape, the computer will type "NK" and stop with the light on. Feel free to try another key. At the end of any loading, the tape is always positioned at a load point.

c) Loading from a reel under tape control: Place a tape with the sequence of file keys to be loaded in the 4500 reader. Depress sense switch 2, and otherwise follow (b) above. If the files all end in a "SEARCH" transfer, they will be loaded automatically. In this case, the loading must be stopped by a blank key (stop code only) on the tape, followed by a word containing an instruction in hex. This instruction is read and executed, and will usually be a stop-and-transfer.

Punching Track 123 (125)

If it is desired to punch a program stored (partly) on track 123 (125), this must be loaded after PROXY, and the program must be punched in strip mode.

PROXY may be used to punch itself. In reel mode, the monitor (track 125) is automatically included; in strip mode, track 123 must be punched by specific control instructions.

To use a punch control tape in the 4500 reader, punch the answers to the control questions for punching in the proper sequence, without carriage returns, then follow with a duplicate copy (except for the first, which must give the input selection for verifying). Then at step (c) above, depress Sense Switch 1.

Preparing a Searchable Reel

A complete reel, suitable for searching, consists of the following tape files:

a) "first" file.

b) Program files, in order of keys (digits 1-9 before letters A-Z) (not all the keys need be present in any one reel).

c) "final" file.

APPENDIX

Details of tape formats.

a) The loader begins with a special bootstrap on track 127 which loads the loader itself onto track 126. The latter is punched in the interlace sequence (0) (32)* (1) (33)*.....(31) (63)*(32) xx*, where xx represents an instruction which is loaded into 12700 and causes the bootstrap to be interrupted. The next word contains the checksum for track 126, and after this is read control is transferred to the sumchecking portion of the loader.

b) The interlace pattern

Each nonconstant track to be loaded is punched in the following form (including the monitor (track 125) on reel-mode tapes).

(code)(0)*(16)(32)*(48)(60)*(12)(28)*(44)(56)*(8)(24)*(40)(52)*(4)(20)*(36)(1)*

(17)(33)*(49)(61)* ..... (37)(2)*

(18)(34)* ..... (38)(3)*

(19)(35)* ..... (39)(sum)*

c) Transfer Codes

A stop-and-transfer to (L) is punched as the hex equivalent of HLT*0*(L)*; a nonstop transfer is punched as the hex equivalent of SNS*6400*(L)*.

d) Key Codes

The key code word for a tape file of code A (for example) is punched in the form GA (e.o.m.) AG*. Here (e.o.m.) stands for the end-of-message code, and G is a constant character which, since it is not hex, is readily detected by the monitor. When a file is segmented by "REPEAT", the inserted keys have this G changed to H, I, J, etc. (in sequence).

PROXY ASSEMBLY 10/22/62

COM PROXY -- PROGRAM 4049.1 HUB 10/15/62

PHOTO-READER ORIENTED HEX SYSTEM *

COM MONITOR AND LOADER IN ABSOLUTE -- LOADER
IN REGION 6 TO OPERATE FROM 126, MONITOR IN 125*

```
*           RST*    *          *                                  *
*           EQV*    [    ]*    0*                                 *
*           REG*    600101*    200*                               *
*           RES*    100*       100*                               *
*           SET*    [    ]*    *
```

```
12500*      RAL*    12502*     12504*     12500     03   12502  12504   MONITOR ENTRY*
12504*      CLL*    12407*     12509*     12504     27   12407  12509   ERROR RETURN*
12509*      SNS*    198*       12513*     12509     00   00111  12513   *
12513*      TBC*    12515*     12517*     12513     23   12515  12517   *
12517*      SNS*    298*       12521*     12517     00   00219  12521   SW1OFF=SEARCH*
12521*      PRD*    9599*      12525*     12521     16   09521  12525   *
12525*      TBC*    12527*     12529*     12525     23   12527  12529   *
12529*      PRD*    7099*      12533*     12529     16   07029  12533   SW2OFF=TYP CONTROL*
12533*      PRC*    K*         12537*     12533     16   03633  12537   *
12527*      PRD*    6699*      12537*     12527     16   06627  12537   SW2ON=RDR CONTROL*
12537*      MPT*    6798*      12541*     12537     15   06739  12541   *
12541*      INP*    6400*      12542*     12541     08   06400  12542   TARGET KEY*
12542*      EXC*    298*       12546*     12542     09   00244  12546   *
12546       CME*    12548*     12550*     12546     20   12548  12550   *
12550*      TBC*    12552*     12554*     12550     23   12552  12554   *
12554*      RAL*    12556*     12558*     12554     03   12556  12558   3F*
12558*      CME*    12560*     12562*     12558     20   12560  12562   ALF/*
12562*      TBC*    12503*     12507*     12562     23   12503  12507   "/" LOADS THIS FILE*
12503*      HLT*    21*        12515*     12503     00   00021  12515   *
12507*      SRL*    106*       12520*     12507     12   00106  12520   *
12520*      STU*    RECRC7*    12526*     12520     24   12723  12526   *
12526*      RAL*    12538*     12530*     12526     03   12528  12530   3AT2*
```

```
12530*   CLL*    RECRC4*   12539*    12530    27   12736   12539    INITIALIZE ERROR COUNTER*
12539*   LDX*    7200*     12543*    12539    07   07200   12543    SEARCH FORWARD*
12543*   PRD*    7499*     12547*    12543    16   07443   12547    *
12547*   MPT*    6798*     12551*    12547    15   06749   12551    *
12551*   INP*    6400*     12553*    12551    08   06400   12553    READ WORD*
12553*   EXC*    298*      12557*    12553    09   00255   12557    *
12557*   CME*    12559*    12561*    12557    20   12559   12561    KEY MASK*
12561*   TBC*    12563*    12506*    12561    23   12563   12506    IS IT A KEY?*
12563*   XPRD*   99*       12508*    12563   x16   00063   12508    NO*
12508*   TBC*    12510*    12547*    12508    23   12510   12547    *
12506*   RAL*    12511*    12514*    12506    03   12511   12514    YES*
12514*   CME*    RECRC7*   12531*    12514    20   12723   12531    MASK = FFF*
12531*   TBC*    12534*    12535*    12531    23   12534   12535    THE RIGHT KEY?*
12535*   CMG*    RECRC7*   12544*    12535    21   12739   12544    NO*
12544*   TBC*    12549*    12555*    12544    23   12549   12555    *
12549*   CXE*    7300*     12563*    12549    01   07300   12563    *
12555*   LDX*    7300*     12563*    12555    07   07300   12563    SEARCH REVERSE*
12510*   PRC*    N*        12521*    12510    16   03910   12521    SEARCH FAILS*
12534*   CXE*    7300*     12538*    12534    01   07300   12538    FOUND*
12538*   TBC*    12539*    12603*    12538    23   12539   12603    *
12515*   INP*    6400*     12516*    12515    08   06400   12516    SW1ON=NO SEARCH*
12516*   CLL*    RECRC7*   12603*    12516    27   12723   12603    12603=LOADER*
12512*   RAU*    RECRC7*   12518*    12512    02   12715   12518    "RPEAT" ENTRY*
12518*   ADU*    12522*    12520*    12518    28   12522   12520    1AT31*
12502*   SNS*    198*      12519*    12502    00   00104   12519    ERROR EXIT*
12519*   TBC*    12501*    12523*    12519    23   12501   12523    *
12501*   HLT*    21*       12600*    12501    00   00021   12600    SW1ON=MANUAL RESTART*
12523*   RAU*    RECRC4*   12532*    12523    02   12728   12532    SW1OFF=AUTOMATIC*
12532*   SBU*    12536*    12540*    12532    30   12536   12540    1AT2*
12540*   STU*    RECRC4*   12524*    12540    24   12744   12524    *
12524*   TMI*    12545*    12555*    12524    22   12545   12555    *
12545*   HLT*    0*        12555*    12545    00   00000   12555    *
12552*   INP*    0*        12654*    12552    08   00000   12654    KEY BLANK, READ HEX TRANSFER*
12556*   HEX*    8000*     3F*       12556    8000003F              *
12560*   ALF*    /*        *         12560    0000003E              *
12548*   HEX*    FFFF*     FFC0*     12548    FFFFFFC0              *
12505*   HEX*    8000*     0*        12505    80000000              *
12528*   HEX*    6000*     0*        12528    60000000              *
```

```
12559*    HEX*    FFFF*    FFDF*    12559    EFFFFFDF              *
12511*    HEX*    0*       FFF*     12511    00000FFF              *
12522*    HEX*    0*       1*       12522    00000001              *
12536*    HEX*    2000*    0*       12536    20000000              *

600000*   MPT*    6798*    12608*   00100    15  06702  12608      LOADER ENTRY*
600003*   MPT*    6798*    12608*   00103    15  06705  12608      *
600008*   RAU*    12612*   12615*   00108    02  12612  12615      FFFFFFFF*
600015*   INP*    0*       12640*   00115    08  00000  12640      CODE WORD*
600040*   TMI*    12654*   12645*   00140    22  12654  12645      TRANSFER?*
600045*   SAU*    12447*   12650*   00145    04  12447  12650      [ADDR=12607]*
600050*   LDC*    12453*   12658*   00150    06  12453  12658      [ADDR=12613]*
600058*   MPT*    198*     12604*   00158    15  00160  12604      *
600004*   EXC*    498*     12607*   00104    09  00406  12607      *
600007*   CXE*    [  ]*    12611*   00107    01  00000  12611      CHECK TRACK CODE*
600011*   ADU*    12613*   12616*   00111    28  12613  12616      *
600016*   SAU*    12618*   12620*   00116    04  12618  12620      *
600020*   TBC*    12624*   12619*   00120    23  12624  12619      *
600024*   TMI*    12626*   12634*   00124    22  12626  12634      *
600026*   LDC*    12429*   12632*   00126    06  12429  12632      [ADDR=12653]*
600032*   XSTL*   98*      12636*   00132   x25  00034  12636      CONSTANT TRACK*
600036*   MPT*    6798*    12644*   00136    15  06738  12644      *
600044*   MPT*    398*     12660*   00144    15  00346  12660      *
600060*   INP*    0*       12643*   00160    08  00000  12643      READ SUM*

600062*   XCLL*   0*       12605*   00162   x27  00000  12605      READ AND STORE*
600005*   INP*    0*       12614*   00105    08  00000  12614      *
600014*   XCLU*   16*      12628*   00114   x26  00016  12628      *
600028*   XCLL*   32*      12637*   00128   x27  00032  12637      *
600037*   INP*    0*       12646*   00137    08  00000  12646      *
600046*   XCLU*   48*      12655*   00146   x26  00048  12655      *
600055*   XCLL*   60*      12601*   00155   x27  00060  12601      *
600001*   INP*    0*       12610*   00101    08  00000  12610      *
600010*   XCLU*   12*      12623*   00110   x26  00012  12623      *
600023*   XCLL*   28*      12633*   00123   x27  00028  12633      *
600033*   INP*    0*       12642*   00133    08  00000  12642      *
600042*   XCLU*   44*      12652*   00142   x26  00044  12652      *
600052*   XCLL*   56*      12661*   00152   x27  00056  12661      *
```

```
600061*  INP*   0*          12606*  00161    08  00000  12606  *
600006*  XCLU*  8*          12618*  00106   x26  00008  12618  *
600018   CXE*   [     ]*    12622*  00118    01  00000  12622  *
600022*  XCLL*  24*         12629*  00122   x27  00024  12629  *
600029*  INP*   0*          12638*  00129    08  00000  12638  *
600038*  XCLU*  40*         12648*  00138   x26  00040  12648  *
600048*  XCLL*  52*         12657*  00148   x27  00052  12657  *
600057*  INP*   0*          12602*  00157    08  00000  12602  *
600002*  XCLU*  4*          12609*  00102   x26  00004  12609  *
600009*  XLDX*  1*          12617*  00109   x07  00001  12617  *
600017*  XCLL*  19*         12625*  00117   x27  00019  12625  *
600025*  INP*   0*          12634*  00125    08  00000  12634  *
600034*  XCLU*  35*         12641*  00134   x26  00035  12641  *
600041*  TBC*   12643*      12662*  00141    23  12643  12662  *
600043*  CLL*   RECRC5*     12647*  00143    27  12745  12647  CHECK SUM*
600047*  RAL*   12649*      12651*  00147    03  12649  12651  *
600051*  LDC*   12653*      12656*  00151    06  12653  12656  *
600056*  XADU*  54*         12659*  00156   x28  00054  12659  *
600059*  CME*   RECRC5*     12663*  00159    20  12761  12663  *
600063*  TBC*   12603*      12619*  00163    23  12603  12619  *
600019*  PRD*   9899*       12627*  00119    16  09819  12627  HERE ON ERROR*
600027*  PRC*   CR*         12629*  00127    16  00127  12639  *
600039*  PRC*   N*          12621*  00139    16  03939  12621  *
600021*  PRC*   O*          12631*  00121    16  04021  12631  *
600031*  HLT*   21*         12600*  00131    00  00021  12600  OR SNS 104 12519*
600064*  CLU*   12718*      RECRC5* 00200    26  12718  12721  ESCAPE FROM BOOTSTRAP*
RECRC5*  LDC*   RECRC7*    RECRC1* 12721    06  12723  12725  COUNT=63*
RECRC1*  ADU*   12627*      12630*  12725    28  12627  12630  GET SUM*
600030*  PRD*   9899*       12635*  00130    16  09830  12635  *
600035*  CME*   RECRC2*     12663*  00135    20  12742  12663  *
600054*  CLL*   RECRC0*     RECRC0* 00154    27  12756  12700  TRANSFER CODE*
600053*  SNS*   0*          6300*   00153    00  00000  06300  *
600012*  HEX*   FFFF*       FFFF*   00112    FFFFFFFF          *
600013*  SNS*   3*          500*    00113    00  00003  00500  *
600049*  HEX*   FFFF*       FFFF*   00149    FFFFFFFF          *
```

```
*COM* HEX PUNCH/VERIFY SUBROUTINES *
*           RST*     *          *                                                    *
*           SET*1FS*PCON*PCH*PCHSC*PPP*FEED*VER3*PW2*PV3**
*           EQR*     RPEAT*     313*                                                THESE FOR TRACK CODES*
*           EQR*     SRCH*      413*                                                *
*           EQR*     NO-OP*     513*                                                *
*           EQR*     HTR*       613*                                                *
*           EQR*     VCODE*     713*                                                *
*           SET*RPEAT*SRCH*NO-OP*HTR*VCODE**
*           REG*     P00316*    318*                                                I/O SELECTION TABLE*
*           RES*     321*       323*                                                *
*           RES*     325*       327*                                                *
*           EQV*     PCON*      400*                                                *
*           RES*     0*         63*                                                 *
*           RES*     700*       12263*                                              *
*           REG*     B00300*    308*                                                BOOTSTRAP*
*           RES*     332*       340*                                                *
*           REG*     A00341*    352*                                                ALPH HEADINGS*

RECRC0*     SNS*     0*         57*        12700     00  00000  00057  *
P8*         PRU*     1699*      PP7*       00257     17  01657  00261  *
PP7*        MPT*     198*       *          00261     15  00163  00201  *
*           MPT*     298*       *          00201     15  00203  00205  *
*           PRU*     1699*      SKIP01*    00205     17  01605  00210  *
*           MPT*     198*       *          00210     15  00112  00214  *
*           MPT*     298*       P6*        00214     15  00216  00218  *
P6*         PRU*     1699*      *          00218     17  01618  00222  *
*           MPT*     198*       *          00222     15  00124  00226  *
*           MPT*     298*       P5*        00226     15  00228  00230  *
P5*         PRU*     1699*      SKIP01*    00230     17  01630  00235  *
*           MPT*     198*       *          00235     15  00137  00239  *
*           MPT*     298*       P4*        00239     15  00241  00243  *
P4*         PRU*     1699*      *          00243     17  01643  00247  *
*           MPT*     198*       *          00247     15  00149  00251  *
*           MPT*     298*       P3*        00251     15  00253  00255  *
P3*         PRU*     1699*      SKIP01*    00255     17  01655  00260  *
*           MPT*     198*       *          00260     15  00162  00400  *
*           MPT*     298*       P2*        00400     15  00202  00204  *
```

```
P2*         PRU*    1699*   *       00204    17  01604  00208   *
*           MPT*    198*    *       00208    15  00110  00212   *
*           MPT*    298*    P1*     00212    15  00214  00216   *
P1*         PRU*    1699*   SKIP01* 00216    17  01616  00221   *
*           XCXE*   0*      RECRC1* 00221   x01  00000  12725   *

RECRC0*     SNS*    0*      47*     12732    00  00000  00047   *
PCH8*       CLL*    RECRC1* P8*     00447    27  12749  00257   *
RECRC0*     SNS*    0*      11*     12700    00  00000  00011   *
PCHWD*      TBC*    PCH8*   *       00211    23  00447  00215   *
*           CLL*    RECRC1* *       00215    27  12717  00219   *
*           EXC*    198*    *       00219    09  00121  00223   *
*           CME*    ZERO*   *       00223    20  00225  00227   *
*           TBC*    RECRC1* *       00227    23  12733  00231   *
*           CME*    4FS*    *       00231    20  00233  00435   *
*           TBC*    1-4*    *       00435    23  00237  00439   *
*           CME*    6FS*    *       00439    20  00241  00443   HERE 5-8*
*           TBC*    5-6*    *       00443    23  00245  00547   *
*           CME*    7FS*    *       00547    20  00249  00451   HERE 7-8*
*           TBC*    PP7*    P8*     00451    23  00261  00257   *

RECRC0*     SNS*    0*      52*     12700    00  00000  00052   *
PCH*        XCXE*   1*      PCH2*   00252   x01  00001  00256   *
PCH2*       CLL*    RECRC2* PCH4*   00256    27  12758  00360   *
PCH4*       CLU*    RECRC6* 12400*  00360    26  12762  12400   *
RECRC0*     RAU*    RECRC3* PW2*    12700    02  12703  00405   IN 12400, PUNCH ONLY*
PW2*        RAL*    *       PCHWD*  00405    03  00207  00211   *
*           RAU*    12738*  LPC*    00207    02  12738  00240   *
LPC*        RAL*    LLL*    *       00240    03  00242  00244   *
*           TBC*    PCH8*   PCHWD*  00244    23  00447  00211   *
LLL*        PRD*    825*    12734*  00242    16  00825  12734   *

1-4*        CME*    2FS*    *       00237    20  00529  00441   HERE 1-4*
*           TBC*    1-2*    *       00441    23  00543  00445   *
*           CME*    3FS*    *       00445    20  00647  00449   HERE 3-4*
*           TBC*    THREE*  FOUR*   00449    23  00551  00253   *
5-6*        CME*    5FS*    *       00245    20  00248  00250   *
*           TBC*    FIVE*   SIX*    00250    23  00452  00254   *
1-2*        CME*    1FS*    *       00543    20  00545  00448   *
```

```
*         TBC*    ONE*      TWO*     00448    23   00450  00552   *
SIX*      SRL*    108*      P6*      00254    12   00108  00218   *
FIVE*     SRL*    112*      P5*      00452    12   00112  00230   *
FOUR*     SRL*    116*      P4*      00253    12   00116  00243   *
THREE*    SRL*    120*      P3*      00551    12   00120  00255   *
TWO*      LDC*    7C*       *        00552    06   00354  00356   *
*         MPT*    198*      P2*      00356    15   00158  00204   *
ONE*      SRL*    4*        *        00450    12   00004  00361   *
*         EXC*    298*      P1*      00361    09   00263  00216   *

RECRC0*   SNS*    0*        22*      12716    00   00000  00022   *
PCHSC*    CLL*    RECRC2*   *        00422    27   12726  00228   *
*         SNS*    498*      *        00228    00   00430  00232   *
*         TBC*    VVV*      *        00232    23   00234  00236   *
*         XCXE*   1*        LPC*     00236    x01  00001  00240   *
VVV*      STL*    RECPC3*   PCH4*     00234    25   12743  00360   *

RECRC0*   SNS*    0*        54*      12700    00   00000  00054   *
VER3*     INP*    0*        VER3*    00454    08   00000  00454   IN 12400, VERIFY ONLY*
VER3*     STU*    RECRC1*   *        00454    24   12757  00259   *
*         CLL*    DB3TTT*   *        00259    27   12461  00263   *
*         RAL*    F---F*    *        00263    03   00401  00203   *
*         CME*    RECRC3*   *        00203    20   12711  00213   *
*         TBC*    *         ERR3*    00213    23   00315  00217   *
*         RAU*    DB3TTT*   *        00315    02   12621  00423   *
*         CME*    RECRC6*   *        00423    20   12730  00432   *
*         TBC*    12734*    ERR3*    00432    23   12734  00217   *
ERR3*     PRD*    9899*     *        00217    16   09817  00421   ERROR*
*         PRC*    CR*       *        00421    16   00121  00425   *
*         RAU*    ATAPE*    *        00425    02   00427  00229   *
*         RAL*    *         AAA*     00229    03   00331  00433   *
*         RAU*    RECRC1*   *        00331    02   12733  00535   *
*         RAL*    *         PCH8*    00535    03   00437  00447   *
*         PRD*    6191*     *        00437    16   06141  00541   *
*         RAU*    DB3TTT*   *        00541    02   12461  00363   *
*         RAL*    *         PCH8*    00363    03   00501  00447   *
*         RAU*    AMEM=*    *        00501    02   00403  00505   *
*         RAL*    *         AAA*     00505    03   00407  00433   *
*         RAU*    RECRC3*   *        00407    02   12711  00314   *
```

```
*           RAL*    *          PCH8*     00314     03   00416  00447    *
*           PRD*    6191*      *         00416     16   06125  00220    *
*           RAU*    RECRC6*    *         00220     02   12722  00224    *
*           RAL*    *          PCH8*     00224     03   00426  00447    *
*           HLT*    15*        RECRC2*   00426     00   00015  12734    *

ZERO*       HEX*    0*         0*        00225     00000000             *
ATAPE*      ALF*    TAPE=*     *         00427     2D6A97B5             *
AMEM=*      ALF*    ,MEM=*     *         00403     3499E9B5             *
F---F*      HEX*    FFFF*      FFFF*     00401     FFFFFFFF             *
1FS*        HEX*    0*         F*        00545     0000000F             *
2FS*        HEX*    0*         FF*       00539     000000FF             *
3FS*        HEX*    0*         FFF*      00647     00000FFF             *
4FS*        HEX*    0*         FFFF*     00233     0000FFFF             *
5FS*        HEX*    F*         FFFF*     00248     000FFFFF             *
6FS*        HEX*    FF*        FFFF*     00241     00FFFFFF             *
7FS*        HEX*    FFF*       FFFF*     00249     0FFFFFFF             *
7C*         SNS*    0*         700*      00354     00   00000  00700    *

*           AVL*    700*       712*
*           AVL*    714*       763*
RECRC0*     SNS*    0*         60*       12700     00   00000  00060    *
PPP11*      CXE*    600033*    *         00460     01   00133  00500    *
*           TBC*    PPP5*      PPP10*    00500     23   00202  00404    *
PPP10*      XRAU*   0*         SKIP24*   00404     x02  00000  00532    9-32*
*           CLU*    RECRC3*    *         00532     26   12735  00537    *
*           XRAU*   32*        SKIP07*   00537     x02  00032  00548    41-64*
*           RAL*    *          PCH*      00548     03   00550  00252    *
*           XLDX*   1*         PPP11*    00550     x07  00001  00460    *
RECRC0*     SNS*    0*         40*       12700     00   00000  00040    *
PPP12*      CXE*    [    ]*    SKIP14*   00440     01   00000  00258    *
*           TBC*    PPP13*     PPP14*    00258     23   00560  00262    *
PPP14*      XRAU*   0*         SKIP10*   00262     x02  00000  00312    0-8*
*           CLU*    RECRC3*    *         00312     26   12719  00521    *
*           XRAU*   32*        SKIP17*   00521     x02  00032  00442    32-40*
*           RAL*    *          PCH2*     00442     03   00444  00256    *
*           XLDX*   1*         PPP12*    00444     x07  00001  00440    *
PPP13*      CXE*    B00010*    *         00560     01   00309  00600    *
*           TBC*    PLD*       PPP10*    00600     23   00402  00404    *
PPP5*       MPT*    398*       *         00202     15   00304  00206    *
```

```
*           LDC*    63C*        *           00206    06  00408  00310   *
*           ADU*    600015*     *           00310    28  00115  00417   *
*           CLU*    RECRC3*     *           00417    26  12719  00621   *
*           RAU*    ALLFS*      *           00621    02  00523  00525   *
*           RAL*    RECRC5*     PCH*        00525    03  12729  00252   *

RECRC0*     SNS*    0*          18*         12756    00  00000  00018   *
PPP*        LDX*    B00001*     *           00418    07  00300  00522   *
*           CLL*    RECRC5*     DOP*        00522    27  12729  00431   *
PLD*        LDX*    600000*     DOP*        00402    07  00100  00431   *
DOP*        EXC*    898*        *           00431    09  00833  00635   *
*           ADU*    9D*         *           00635    28  00637  00639   *
*           SAU*    PPP12*      PPP14*      00639    04  00440  00262   *
*           SET*    AAA*        *

63C*        SNS*    0*          6300*       00408    00  00000  06300   *
ALLFS*      HEX*    FFFF*       FFFF*       00523    FFFFFFFF          *
9D*         SNS*    9*          0*          00637    00  00009  00000   *

FEED*       CLL*    RECRC2*     *           00700    27  12702  00504   FEED X/2" OF TAPE*
*           SNS*    498*        FEED2*      00504    00  00406  00508   *
FEED2*      TBC*    RECRC2*     *           00508    23  12710  00412   *
*           PRC*    TF*         SKIP09*     00412    16  00012  00625   *
*           PRC*    TF*         SKIP08*     00625    16  00025  00737   *
*           PRC*    TF*         SKIP05*     00737    16  00037  00246   *
*           XLDX*   12763*      *           00246   x07  12763  00650   *
*           PRC*    TF*         SKIP04*     00650    16  00050  00358   *
*           CXE*    0*          *           00358    01  00000  00362   *
*           PRC*    TF*         FEED2*      00362    16  00062  00508   *

B00001*     CLL*    12719*      12727*      00300    27  12719  12727   BOOTSTRAP
B00002*     INP*    0*          12716*      00301    08  00000  12716   *
B00003*     XCLL*   32*         12702*      00302   x27  00032  12702   *
B00004*     XLDX*   1*          12706*      00303   x07  00001  12706   *
B00005*     SNS*    12600*      0*          00304    00  12600  00000   *
B00006*     LDC*    12723*      12725*      00305    06  12723  12725   *
B00007*     ADU*    12627*      12630*      00306    28  12627  12630   *
B00008*     SNS*    0*          6300*       00307    00  00000  06300   *
B00009*     XCLU*   0*          12736*      00308   x26  00000  12736   *
```

```
B00033*   CLU*   12732*   RECRC0*   00332   26   12732   12740   *
B00034*   CLU*   12730*   RECRC6*   00333   26   12730   12738   *
B00035*   CLU*   12736*   RECRC6*   00334   26   12736   12738   *
B00036*   CLU*   12734*   RECRC6*   00335   26   12734   12738   *
B00037*   EXC*   629*     RECRC6*   00336   09   00629   12738   *
B00038*   CLU*   12729*   RECRC6*   00337   26   12729   12738   *
B00039*   CLU*   12733*   RECRC6*   00338   26   12733   12738   *
B00040*   CLU*   12731*   RECRC6*   00339   26   12731   12738   *
B00041*   CLU*   12732*   RECRC6*   00340   26   12732   12738   *

*COM*   OPERATOR COMMUNICATION *

*         RST*   *         *                                      *
*         SET*RDBIN*ALPH*RDTRK*AAA**

BIN*      CLU*   RECRC3*   *         00238   26   12743   00645   *
*         XCXE*  0*        *         00645   x01  00000   00549   *
*         SRL*   116*      *         00549   12   00116   00608   *
*         EXT*   1FS*      *         00608   18   00545   00747   *
*         MPT*   98*       L1*       00747   15   00049   00651   *
L1*       CLU*   RECRC2*   *         00651   26   12758   00660   *
*         SRL*   104*      *         00660   12   00104   00507   *
*         ADU*   RECRC2*   *         00507   28   12710   00512   *
*         MPT*   98*       *         00512   15   00014   00516   *
*         CLU*   RECRC2*   *         00516   26   12718   00320   *
*         SRL*   104*      *         00320   12   00104   00531   *
*         ADU*   RECRC3*   *         00531   28   12735   00438   *
*         ADU*   RECRC2*   *         00438   28   12742   00544   *
*         TBC*   *         OUT*      00544   23   00446   00648   *
*         MPT*   198*      *         00446   15   00148   00750   *
*         MPT*   198*      *         00750   15   00152   00554   *
*         CLU*   RECRC3*   L1*       00554   26   12759   00651   *
OUT*      MPT*   298*      *         00648   15   00250   00652   *
*         EXT*   *         RECRC0*   00652   18   00654   12756   *
*         SNS*   6400*     12763*    00654   00   06400   12763   *
```

```
RDBIN*    CLL*    RECRC0*   *        00463    27   12708  00410   *
*         RAU*    MZ*       *        00410    02   00612  00414   *
*         INP*    0*        *        00414    08   00000  00415   *
*         TMI*    RECRC0*   *        00415    22   12724  00319   *
*         EXC*    298*      *        00319    09   00221  00623   *
*         EXT*    FAT11*    *        00623    18   00725  00527   *
*         SBU*    RAT11*    *        00527    30   00329  00631   *
*         TMI*    HTR*      *        00631    22   00613  00735   *
*         EXC*    498*      *        00735    09   00437  00739   *
*         XCXE*   0*        *        00739    x01  00000  00643   *
*         XTBC*   RPEAT*    0*       00643    x23  00313  00000   *
HTR*      RAU*    MZ*       ST7*     00613    02   00612  00514   *
NO-OP*    RAU*    1AT12*    ST7*     00513    02   00515  00514   *
VCODE*    RAU*    MZ*       ST7*     00713    02   00612  00514   *
ST7*      STU*    RECRC7*   BIN*     00514    24   12723  00238   *

MZ*       HEX*    8000*     0*       00612    80000000            *
1AT12*    HEX*    8008*     0*       00515    80080000            *
FAT11*    HEX*    F0*       0*       00725    00F00000            *
RAT11*    HEX*    R0*       0*       00329    00B00000            *

SRCH*     RAU*    RECRC4*   *        00413    02   12720  00622   *
*         TMI*    AT2*      *        00622    22   00324  00526   *
*         RAL*    *         NO-OP*   00526    03   00328  00513   *
*         HEX*    1*        2500*    00328    00012500            *

RPEAT*    RAU*    RECRC7*   *        00313    02   12715  00517   *
*         ADU*    DKEY*     *        00517    28   00419  00721   *
*         TMI*    AT2*      *        00721    22   00324  00626   *
*         SBU*    1AT7*     *        00626    30   00428  00330   *
*         CLU*    RECRC7*   *        00330    26   12739  00641   *
*         RAU*    *         RECRC0*  00641    02   00743  12748   *
*         SNS*    6400*     12512*   00743    00   06400  12512   *
1AT7*     HEX*    100*      0*       00428    01000000            *
```

```
ALPH*     CLL*    RECRC6*   AT2*      00712     27   12714  00324   *
AT2*      PRD*    9599*     *         00324     16   09524  00528   *
*         PRD*    12699*    *         00528     16   12628  00632   *
*         SNS*    198*      *         00632     00   00134  00436   *
*         TBC*    RDR*      TYP*      00436     23   00538  00540   *
RDR*      PRD*    6699*     *         00538     16   06638  00542   SW1ON = RDR IN*
*         SNS*    298*      *         00542     00   00244  00546   *
*         TBC*    2ON*      *         00546     23   00748  00751   *
*         PRD*    12599*    T2*       00751     16   12551  00355   SW2OFF = PRINT*
TYP*      PRD*    7099*     T2*       00540     16   07040  00355   SW1OFF = TYP IN*
T2*       PRC*    CR*       *         00355     16   00155  00359   *
*         XRAU*   0*        *         00359    x02   00000  00563   *
*         RAL*    *         AAA*      00563     03   00601  00433   *
*         XRAU*   1*        *         00601    x02   00001  00605   *
*         PRC*    SP*       *         00605     16   06105  00209   *
*         RAL*    *         AAA*      00209     03   00311  00433   *
*         PRD*    292*      2ON*      00311     16   00218  00748   *
2ON*      SRL*    163*      RECRC6*   00748     12   00163  12754   *

AAA*      CLL*    RECRC0*   *         00433     27   12740  00642   *
*         EXC*    198*      *         00642     09   00144  00646   *
*         SRL*    102*      AL1*      00646     12   00102  00455   *
AL1*      CME*    ZERO*     *         00455     20   00357  00459   *
*         TBC*    RECRC0*   *         00459     23   12700  00663   *
*         PRU*    6499*     *         00663     17   06463  00503   *
*         SRL*    106*      AL1*      00503     12   00106  00455   *

RDTRK*    CLL*    RECRC1*   *         00555     27   12757  00559   *
*         RAL*    *         ALPH*     00559     03   00461  00712   *
*         INP*    0*        *         00461     08   00000  00462   *
*         SRL*    108*      *         00462     12   00108  00614   *
*         RAU*    XX*       *         00614     02   00616  00518   *
*         CLU*    RECRC0*   BIN*      00518     26   12724  00238   *
XX*       SRL*    113*      RECRC1*   00616     12   00113  12741   *
ZERO*     HEX*    0*        0*        00357     00000000           *
DKEY*     HEX*    200*      1*        00419     02000001           *
```

COM MAIN CONTROL ROUTINE*

```
*            RST*     *          *                                          *
*            SET*     DUMP*      *
*            EQR*     DUMP*      0*                                         *
*            AVL*     1*         63*                                        *
*            AVL*     800*       863*                                       *

RECRC0*      SNS*     0*         57*        12748     00   00000  00057     *
LL10*        CLU*     RECRC3*    LL20*      00057     26   12759  00061     *
LL20*        XRAU*    0*         SKIP04*    00061    x02   00000  00005     *
*            RAL*     *          PCH*       00005     03   00007  00252     *
*            XRAU*    16*        SKIP10*    00007    x02   00016  00021     *
*            CLU*     RECRC3*    *          00021     26   12727  00029     *
*            XRAU*    32*        SKIP04*    00029    x02   00032  00037     *
*            RAL*     *          PCH*       00037     03   00039  00252     *
*            XRAL*    48*        SKIP10*    00039    x03   00048  00053     *
*            XRAU*    60*        SKIP08*    00053    x02   00060  00001     *
*            CLL*     RECRC3*    *          00001     27   12703  00705     *
*            RAL*     *          PCH*       00705     03   00607  00252     *
*            XRAU*    12*        SKIP06*    00607    x02   00012  00017     *
*            CLU*     RECRC3*    SKIP02*    00017     26   12719  00023     *
*            XRAU*    28*        SKIP07*    00023    x02   00028  00034     *
*            RAL*     *          PCH*       00034     03   00036  00252     *
*            XRAU*    44*        SKIP09*    00036    x02   00044  00049     *
*            CLU*     RECRC3*    *          00049     26   12751  00353     *
*            XRAU*    56*        SKIP04*    00353    x02   00056  00561     *
*            RAL*     *          PCH*       00561     03   00063  00252     *
*            XRAU*    8*         SKIP10*    00063    x02   00008  00013     *
*            CLU*     RECRC3*    *          00013     26   12719  00821     *
*            XRAU*    24*        SKIP05*    00821    x02   00024  00030     *
*            RAL*     *          PCH*       00030     03   00032  00252     *
*            XRAL*    40*        SKIP09*    00032    x03   00040  00045     *
*            XRAU*    52*        SKIP08*    00045    x02   00052  00457     *
*            CLL*     RECRC3*    SKIP01*    00457     27   12759  00062     *
*            RAL*     *          PCH*       00062     03   00800  00252     *
*            XRAU*    4*         SKIP05*    00800    x02   00004  00009     *
*            CLU*     RECRC3*    SKIP01*    00009     26   12711  00014     *
*            XRAU*    20*        SKIP07*    00014    x02   00020  00025     *
*            XLDX*    1*         SKIP01*    00025    x07   00001  00430     *
```

*	RAL*	*	PCH*	00430	03	00732	00252	*
*	XRAU*	35*	SKIP05*	00732	x02	00035	00041	*
*	CLU*	RECRC3*	LLL*	00041	26	12743	00745	*
LLL*	CXE*	[ ]*	*	00745	01	00000	00649	*
*	TBC*	*	LL20*	00649	23	00051	00061	*
*	LDC*	63C1*	*	00051	06	00453	00055	*
*	XADU*	54*	*	00055	x28	00054	00059	*
*	RAL*	LOOK*	PCH*	00059	03	00661	00252	*
LOOK*	RAU*	RECRC4*	*	00661	02	12704	00006	*
*	EXC*	498*	CLAST*	00006	09	00408	00010	*
CLAST*	CXE*	[ ]*	*	00010	01	00000	00714	*
*	ADU*	1AND1*	*	00714	28	00016	00018	*
*	TBC*	WAS*	NXTRK*	00018	23	00020	00022	*
RECRC0*	SNS*	0*	12*	12724	00	00000	00012	*
*	NXT*	0*	10*					*
RECRC4*	TMI*	L4L*	L3L*	12712	22	00814	00026	*
NXTRK*	LDX*	2*	*	00022	07	00002	00726	*
*	RAL*	*	FEED*	00726	03	00028	00700	*
*	TMI*	L4L*	L3L*	00028	22	00814	00026	*
L4L*	SNS*	498*	*	00814	00	00416	00618	*
*	TBC*	L3L*	*	00618	23	00026	00722	*
*	PRD*	1399*	L3L*	00722	16	01322	00026	*
L3L*	STU*	RECRC4*	*	00026	24	12728	00530	*
*	ADU*	4AT17*	*	00530	28	00832	00434	*
*	SAU*	LLL*	*	00434	04	00745	00047	*
*	RAL*	MZ*	*	00047	03	00749	00851	*
*	EXC*	698*	*	00851	09	00653	00655	*
*	XLDX*	12760*	*	00655	x07	12760	00659	*
*	LDC*	63C6*	*	00659	06	00761	00763	*
*	XCME*	98*	ML4*	00763	x20	00001	00003	*
ML4*	TBC*	0*	ML5*	00003	23	00000	00707	*
ML5*	MPT*	6798*	SOME-*	00707	15	06709	00011	*
ML4*	TBC*	SOME-*	ML5*	00003	23	00011	00707	*
SOME-*	ADL*	C2*	*	00011	29	00813	00015	*
*	RAU*	ALLFS*	*	00015	02	00617	00019	*
*	LDC*	63C2*	*	00019	06	00822	00024	*
*	XEXT*	98*	ML6*	00024	x18	00026	00628	*
ML6*	TMI*	0*	ML7*	00628	22	00000	00033	*
ML7*	EXC*	398*	*	00033	09	00335	00837	*
*	TMI*	DIFF*	*	00837	22	00839	00741	*
*	EXC*	398*	ALL*	00741	09	00343	00845	*

ML6*	TMI*	ALL*	ML7*	00628	22	00845	00033	*
ALL*	ADU*	1AT31*	ML8*	00845	28	00847	00849	*
ML8*	TBC*	0*	ML9*	00849	23	00000	00553	*
ML9*	LDC*	63C3*	*	00553	06	00755	00557	*
*	XCMG*	98*	MLM*	00557	x21	00059	00861	*
ML8*	TBC*	MLM*	ML9*	00849	23	00861	00553	*
DIFF*	XCXE*	0*	MLM*	00839	x01	00000	00861	*
MLM*	RAU*	RECRC4*	*	00861	02	12704	00406	*
*	EXT*	MMM*	*	00406	18	00008	00510	*
*	TBC*	*	SAME*	00510	23	00012	00615	*
*	EXC*	498*	LL10*	00012	09	00414	00057	*
SAME*	ADU*	1AT18*	*	00615	28	00717	00519	*
*	CLU*	RECRC3*	*	00519	26	12727	00429	*
*	XRAU*	98*	*	00429	x02	00031	00533	*
*	RAL*	*	PCH*	00533	03	00035	00252	*
*	MPT*	336*	*	00035	15	00336	00038	*
*	LDC*	63C5*	*	00038	06	00040	00042	*
*	XADU*	98*	*	00042	x28	00044	00046	*
*	RAL*	LOOK*	PCHSC*	00046	03	00661	00422	*
RECRC0*	SNS*	0*	0*	12724	00	00000	00000	*
RECRC0*	SNS*	0*	L6L*	12700	00	00000	00004	*
WAS*	RAU*	RECRC5*	*	00020	02	12729	00031	NO MORE*
*	TMI*	L2*	*	00031	22	00633	00835	TRANSFER?*
*	RAL*	*	PCHS C*	00835	03	00638	00422	YES*
*	RAU*	RECRC7*	*	00638	02	12747	00050	*
*	TMI*	L10*	*	00050	22	00052	00054	KEY CODE?*
*	SNS*	498*	*	00054	00	00456	00058	YES*
*	TBC*	*	L5L*	00058	23	00060	00562	*
*	INP*	0*	L6L*	00060	08	00000	00004	*
L5L*	RAL*	*	AAA*	00562	03	00701	00433	*
*	PRD*	890*	L6L*	00701	16	00808	00004	STOP CODE*
L6L*	RAU*	RECRC7*	*	00004	02	12707	00309	*
*	RAL*	ALLFS*	*	00309	03	00617	00619	*
*	CME*	KEY0*	*	00619	20	00723	00825	*
*	TBC*	L10*	*	00825	23	00052	00529	FIRST?*
*	CME*	KEY/*	*	00529	20	00731	00733	NO*
*	TBC*	*	L2*	00733	23	00536	00633	FINAL?*

```
*         RAU*    FTR*     *        00536    02   00738  00640   YES*
*         RAL*    *        PCHSC*   00640    03   00742  00422   *
*         LDX*    400*     *        00742    07   00400  00746   *
*         RAL*    LPAUS*   FEED*    00746    03   00048  00700   PUNCH TRAILER*
FTR*      SNS*    6400*    12555*   00738    00   06400  12555   *

DUMP*     SNS*    498*     *        00000    00   00402  00604   *
*         TBC*    VV1*     PP1*     00604    23   00506  00708   PUNCH OR VERIFY?*
PP1       LDX*    A00001*  *        00708    07   00341  00812   *
*         RAU*    PP2*     *        00812    02   00715  00817   *
*         CLU*    12624*   *        00817    26   12624  00826   [12400]*
*         RAL*    *        RDTRK*   00826    03   00728  00555   *
*         RAL*    SKIP05*  LL1L*    00728    03   00636  00838   *
*         SNS*    12700*   4*       00636    00   12700  00004   *
PP2*      RAU*    12703*   PW2*     00715    02   12703  00405   *
VV1       LDX*    A00011*  *        00506    07   00351  00610   *
*         RAU*    VV2*     *        00610    02   00815  00718   *
*         CLU*    12624*   *        00718    26   12624  00027   *
*         RAL*    *        RDTRK*   00027    03   00629  00555   *
*         RAL*    *        LL1L*    00629    03   00831  00838   *
*         SNS*    12700*   8*       00831    00   12700  00008   *
VV2*      INP*    0*       VER3*    00815    08   00000  00454   *
LL1L*     LDC*    2C*      *        00838    06   00740  00842   *
*         CME*    P00001*  SKIP07*  00842    20   00316  00627   *
*         TBC*    *        SETP*    00627    23   00729  00833   SPECIAL?*
*         EXC*    1098*    *        00729    09   01031  00534   YES*
*         SRL*    113*     *        00534    12   00113  00754   *
*         EXC*    498*     *        00754    09   00456  00458   *
*         XRAU*   PCON*    SETP*    00458   x02   00400  00833   *
P00001*   SNS*    4500*    0*       00316    00   04500  00000   45*
P00002*   SNS*    4600*    0*       00317    00   04600  00000   46*
P00003*   SNS*    10100*   0*       00318    00   10100  00000   PR, HSP*
P00006*   SNS     9700*    0*       00321    00   09700  00000   PUNCH TABLE*
P00007*   SNS*    11300*   0*       00322    00   11300  00000   *
P00008*   SNS*    10600*   0*       00323    00   10600  00000   *
P00010*   SNS*    6400*    0*       00325    00   06400  00000   READ TABLE*
P00011*   SNS*    8000*    0*       00326    00   08000  00000   *
P00012*   SNS*    7400*    0*       00327    00   07400  00000   *
```

```
SETP*    ADU*    SEL*     *       00833    28   00736  00840   *
*        STU*    DB3SEL*  L10*    00840    24   12442  00052   *
L10*     LDX*    A00009*  *       00052    07   00349  00056   *
*        RAL*    *        ALPH*   00056    03   00558  00712   *
*        EXC*    498*     *       00558    09   00460  00662   0 TO X*
*        RAU*    MZ*      *       00662    02   00749  00752   *
*        INP*    6400*    *       00752    08   06400  00653   FILE KEY*
*        TMI*    L31*     *       00653    22   00855  00657   BLANK = NO MONITOR*
*        EXC*    298*     *       00657    09   00259  00762   *
*        RAL*    A5DEL*   *       00762    03   00801  00603   *
*        CME*    FIRST*   *       00603    20   00805  00807   *
*        TBC*    LFIRS*   *       00807    23   00409  00411   *
*        CME*    FINAL*   *       00411    20   00716  00818   *
*        TBC*    LFINA*   *       00818    23   00420  00823   *
*        CME*    PAUSE*   *       00823    20   00727  00829   *
*        TBC*    LPAUS*   *       00829    23   00048  00634   *
*        RAL*    3F*      *       00634    03   00836  00841   *
*        CMG*    ALF0*    *       00841    21   00043  00846   *
*        TBC*    L10*     *       00846    23   00052  00850   *
*        CMG*    ALFZ*    *       00850    21   00852  00854   *
*        TBC*    *        L10*    00854    23   00456  00052   *
*        EXT*    3F*      *       00456    18   00836  00843   SET UP KEY CODE*
*        EXC*    598*     *       00843    09   00545  00848   *
*        SLT*    112*     *       00848    12   00112  00703   *
*        EXC*    1098*    *       00703    09   01005  00808   *
*        SLT*    106*     *       00808    12   00106  00424   *
*        ADU*    BKEY*    L31*    00424    28   00827  00855   *
L31*     XCXE*   1*       *       00855    x01  00001  00759   *
*        LDX*    60*      L25*    00759    07   00060  00863   *
L25*     STU*    RECRC7*  *       00863    24   12707  00509   KEY CODE*
*        STU*    RECRC4*  GO*     00509    24   12712  00816   MODE FLAG*
GO*      PRD*    12699*   *       00816    16   12616  00520   *
*        PRD*    9599*    DB3SEL* 00520    16   09520  12442   *
RECRC0*  PRD*    0*       3SEL*   12748    16   00000  00753   *
SEL*     PRD*    99*      3SEL*   00736    16   00036  00753   *
3SEL*    TBC*    NOBOO*   *       00753    23   00556  00757   *
*        SNS*    498*     *       00757    00   00459  00862   *
*        TBC*    L8L*     *       00862    23   00002  00502   *
*        RAL*    *        FEED*   00502    03   00704  00700   *
*        PRD*    1392*    L8L*    00704    16   01310  00002   *
```

```
L8L*    RAL*    *          PPP*     00002   03   00804   00418   PUNCH BOOTSTRAP*
*       RAU*    RECRC4*    *        00804   02   12712   00719   *
*       TMI*    L2*        *        00719   22   00633   00524   *
*       RAU*    125T*      *        00524   02   00828   00630   *
*       CLU*    RECRC5*    *        00630   26   12737   00044   *
*       RAU*    125C*      *        00044   02   00853   00656   *
*       SAU*    CLAST*     NXTRK*   00656   04   00010   00022   MONITOR [TRACK 125] *
L2*     LDX*    A00003*    *        00633   07   00343   00644   *
*       RAL*    *          RDTRK*   00644   03   00756   00555   FIRST TRACK*
*       EXC*    598*       *        00756   09   00558   00760   *
*       SRL*    18*        *        00760   12   00018   00624   *
*       EXC*    1098*      *        00624   09   01026   00730   *
*       ADU*    1AT4*      *        00730   28   00734   00744   *
*       RAL*    C2*        *        00744   03   00813   00819   *
*       EXC*    398*       *        00819   09   00321   00724   *
*       MST*    RECRC4*    *        00724   05   12728   00830   *
*       LDX*    A00005*    *        00830   07   00345   00834   *
*       RAL*    *          RDTRK*   00834   03   00844   00555   FINAL TRACK*
*       SAU*    CLAST*     *        00844   04   00010   00620   *
*       RAL*    DM*        *        00620   03   00824   00856   *
*       CMG*    DT122*     *        00856   21   00658   00860   *
*       TBC*    L223*      L222*    00860   23   00602   00702   *
L223*   ADU*    1D*        *        00602   28   00606   00609   *
*       CMG*    RECRC4*    *        00609   21   12712   00720   *
*       TBC*    L2*        L100*    00720   23   00633   00857   *
L100*   LDX*    A00007*    *        00857   07   00347   00802   *
*       RAL*    *          ALPH*    00802   03   00706   00712   *
*       RAL*    *          RDBIN*   00706   03   00709   00463   TRANS ADDR.*
*       CLU*    RECRC5*    *        00709   26   12713   00820   *
*       LDX*    60*        L30*     00820   07   00060   00758   *
L30*    XCXE*   0*         GO*      00758   x01  00000   00816   *
NOBOO*  RAU*    RECRC4*    NXTRK*   00556   02   12760   00022   *
L222*   EXC*    498*       *        00702   09   00404   00806   *
*       CXE*    12300*     *        00806   01   12300   00710   *
*       TBC*    *          L2*      00710   23   00858   00633   *
*       RAU*    RECRC4*    *        00858   02   12760   00803   *
*       TMI*    *          L2*      00803   22   00809   00633   *
*       EXC*    898*       L223*    00809   09   00811   00602   *
```

```
FIRST*    ALF*    FIRST*    *        00805    1F8ABB2D          *
LFIRS*    LDX*    400*      *        00409    07  00400  00859  *
*         RAU*    KEY0*     L25*     00859    02  00723  00863  *
KEY0*     HEX*    6000*     D420*    00723    6000D420          *
FINAL*    ALF*    FINAL*    *        00716    1F8A76A5          *
LFINA*    RAU*    KEY/*     L31*     00420    02  00731  00855  *
KEY/*     HEX*    61D0*     DFA0*    00731    61D0DFA0          *
PAUSE*    ALF*    PAUSE*    *        00727    296AEB1E          *
LPAUS*    PRD*    9599*     *        00048    16  09548  00810  *
*         PRD*    12699*    *        00810    16  12610  00511  *
*         HLT*    1*        DUMP*    00511    00  00001  00000  *

125C*     01*     12500*    358*     00853    01  12500  00358  *
125T*     SNS*    6400*     12500*   00828    00  06400  12500  *
2C*       SNS*    0*        200*     00740    00  00000  00200  *
BKEY*     HEX*    6000*     D020*    00827    6000D020          *
MMM*      HEX*    7FFF*     FFFF*    00008    7FFFFFFF          *
63C1*     SNS*    0*        6300*    00453    00  00000  06300  *
1AT18*    DEC*    18*       1*       00717    00002000          *
A5DEL*    HEX*    3FFF*     FFFF*    00801    3FFFFFFF          *
ALF/*     ALF*    /*        *        00611    0000003E          *
1AND1*    SNS*    100*      2*       00016    00  00100  00002  *
4AT17*    SNS*    4*        0*       00832    00  00004  00000  *
MZ*       HEX*    8000*     0*       00749    80000000          *
63C6*     SNS*    0*        6300*    00761    00  00000  06300  *
C2*       HEX*    7FFF*     FFFF*    00813    7FFFFFFF          *
ALLFS*    HEX*    FFFF*     FFFF*    00617    FFFFFFFF          *
63C2*     SNS*    0*        6300*    00822    00  00000  06300  *
1AT31*    HEX*    0*        1*       00847    00000001          *
63C3*     SNS*    0*        6300*    00755    00  00000  06300  *
63C5*     SNS*    0*        6300*    00040    00  00000  06300  *
3F*       HEX*    0*        3F*      00836    0000003F          *
ALF0*     ALF*    0*        *        00043    00000010          *
ALFZ*     ALF*    Z*        *        00852    00000033          *
1AT4*     HEX*    800*      0*       00734    08000000          *
DM*       SNS*    12763*    0*       00824    00  12763  00000  *
DT122     SNS*    12200*    0*       00658    00  12200  00000  *
1D*       SNS*    1*        0*       00606    00  00001  00000  *
```

```
A00001*  ALF*  PUNCH*  *          00341  29BA7721  *
A00002*  ALF*  SEL.  *  *          00342  2C7A5F3D  *
A00003*  ALF*  FIRST *  *          00343  1F8ABB2D  *
A00004*  ALF*  TRACK*  *          00344  2DADA724  *
A00005*  ALF*  FINAL *  *          00345  1F8A76A5  *
A00006*  ALF*  TRACK*  *          00346  2DADA724  *
A00007*  ALF*  TRANS*  *          00347  2DADA9EC  *
A00008*  ALF*  ADDR.*  *          00348  1A75DAFC  *
A00009*  ALF*  FILE  *  *          00349  1F8A57BD  *
A00010*  ALF*  KEY   *  *          00350  247B2F7D  *
A00011*  ALF*  INPUT *  *          00351  229E9BAD  *
A00012*  ALF*  SEL.  *  *          00352  2C7A5F3D  *
*        END*  *       DUMP*                       *  00716
```

AVAILABLE: 611,711,811

PROXY PUNCHING ITSELF AT 00000

```
PUNCH SEL      45*
FILE KEY       *
FIRST TRACK    123*
FINAL TRACK    123*
TRANS ADDR.    *
FIRST TRACK    0*
FINAL TRACK    8*
TRANS ADDR.    0*
FILE KEY       PAUSE*
```