

CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL



```
1 0001 0 MODULE termdefs ( IDENT = 'V04-000'  
2 0002 0 ADDRESSING_MODE (EXTERNAL = GENERAL)) =  
3 0003 1 BEGIN  
4 0004 1  
5 0005 1  
6 0006 1 *****  
7 0007 1 *  
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *  
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *  
10 0010 1 * ALL RIGHTS RESERVED. *  
11 0011 1 *  
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *  
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *  
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *  
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *  
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *  
17 0017 1 * TRANSFERRED. *  
18 0018 1 *  
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *  
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *  
21 0021 1 * CORPORATION. *  
22 0022 1 *  
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *  
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *  
25 0025 1 *  
26 0026 1 *  
27 0027 1 *****  
28 0028 1  
29 0029 1 **  
30 0030 1 FACILITY: SET and SHOW  
31 0031 1  
32 0032 1 ABSTRACT:  
33 0033 1  
34 0034 1 This module defines initialized storage used by both SET TERMINAL  
35 0035 1 and SHOW TERMINAL.  
36 0036 1  
37 0037 1 ENVIRONMENT:  
38 0038 1  
39 0039 1 VAX/VMS operating system, user mode  
40 0040 1  
41 0041 1 AUTHOR: Gerry Smith 21-Mar-1983  
42 0042 1  
43 0043 1 Modified by:  
44 0044 1  
45 0045 1 V03-010 EMB113 Ellen M. Batbouta 13-Aug-1984  
46 0046 1 Fix the characteristic, SET_SPEED, to be set correctly.  
47 0047 1  
48 0048 1 V03-009 PLL000 Pamela L. Levesque 6-Jul-1984  
49 0049 1 Change placeholder terminal names from UNKNOWN1 - 8  
50 0050 1 to DUMMY1 - 8 to avoid ambiguity with type UNKNOWN.  
51 0051 1  
52 0052 1 V03-008 EMD0067 Ellen M. Dusseault 23-Mar-1984  
53 0053 1 Clear decrct2 for all terminals except VT200_Series  
54 0054 1 terminals.  
55 0055 1  
56 0056 1 V03-007 EMD0060 Ellen M. Dusseault 08-Mar-1984  
57 0057 1 Remove the terminal characteristic, line editing,
```

```

58 0058 1 |
59 0059 1 |
60 0060 1 |
61 0061 1 |
62 0062 1 |
63 0063 1 |
64 0064 1 |
65 0065 1 |
66 0066 1 |
67 0067 1 |
68 0068 1 |
69 0069 1 |
70 0070 1 |
71 0071 1 |
72 0072 1 |
73 0073 1 |
74 0074 1 |
75 0075 1 |
76 0076 1 |
77 0077 1 |
78 0078 1 |
79 0079 1 |
80 0080 1 |
81 0081 1 |
82 0082 1 |
83 0083 1 |
84 0084 1 |
85 0085 1 |
86 0086 1 |
87 0087 1 |
88 0088 1 |
89 0089 1 |
90 0090 1 |
91 0091 1 |
92 0092 1 |

```

from the table of characteristics to set with each device. The decision of whether to turn line editing on will be left up to the system manager. But line editing will be cleared if dec\_crt is cleared.

V03-006 EMD0055 Ellen M. Dusseault 28-Feb-1984  
Add new device name, PRO-Series, to table of devices along with its permanent characteristics. New terminal characteristic, DEC\_CRT2, is available.

V03-005 MMD0235 Meg Dumont, 4-Feb-1984 14:45  
Add descriptors for SET TERM/PROTOCOL qualifiers. Needed for support of switching terminal ports to and from asynch ddcmp lines.

V03-004 MIR0300 MICHAEL I. ROSENBLUM 02-FEB-1984  
PUT IN TERMINAL TYPES LOST IN THE TRANSLATION TO BLISS, MAKE NOLINE EDITING THE DEFAULT ON NON-ANSICRT TERMINALS. CHANGE MACRO TO ALLOW A NAME DIFFERENT FROM THE TT DEVICE CODE.

V03-003 MIR0083 Michael I. Rosenblum 23-Aug-1983  
Add syspwd qualifier. Add in CT inquire sequence for Version 1.5 of POS.

V03-002 MIR0071 Michael I. Rosenblum 22-Jul-1983  
Change VT200\_series terminals to be 8-bit terminals  
Make holdscreen a user preference feature in all but the block mode terminals.

V03-001 MIR0035 Michael I. Rosenblum 27-Apr-1983  
Add new devdepend bits, parity=none, and negated second device dependant definitions.

```

: 94 0093 1 :
: 95 0094 1 : This module creates a number of global data structures. The structures
: 96 0095 1 : are used by both SET TERMINAL and SHOW TERMINAL. The resulting literals
: 97 0096 1 : and structures are described as follows:
: 98 0097 1 :
: 99 0098 1 :     TERMS_NUM - a GLOBAL LITERAL, telling how many terminal types
100 0099 1 :     are in the tables.
101 0100 1 :     TERMS_TABLE - a BLOCKVECTOR of TERMS_NUM blocks. Each block
102 0101 1 :     contains the terminal type, the default page width,
103 0102 1 :     page length, and bits to set/clear in the two terminal
104 0103 1 :     characteristics longwords. In addition, if the terminal
105 0104 1 :     is capable of identifying itself, the number of possible
106 0105 1 :     response strings is included, as well as the address of
107 0106 1 :     a table of descriptors of those response strings.
108 0107 1 :     TERMS_NAME - a VECTOR of TERMS_NUM descriptor addresses. For each
109 0108 1 :     terminal type described in TERMS_TABLE, there is a
110 0109 1 :     string containing the terminal's name.
111 0110 1 :     TERMS_TT_NUM - a GLOBAL LITERAL, the number of keywords/bits which
112 0111 1 :     are described in the first characteristics longword
113 0112 1 :     TERMS_TT_KEY - a VECTOR of descriptor addresses. These are the
114 0113 1 :     names for the keywords which SET/SHOW terminal
115 0114 1 :     understand, which go in the first char longword
116 0115 1 :     TERMS_TT_BIT - a VECTOR of bitmasks which correspond to the keywords
117 0116 1 :     described in TERMS_TT_KEY
118 0117 1 :     TERMS_TT2_NUM - a GLOBAL LITERAL, the number of keywords/bits which
119 0118 1 :     are described in the second characteristics longword
120 0119 1 :     TERMS_TT2_KEY - a VECTOR of descriptor addresses. These are the
121 0120 1 :     names for the keywords which SET/SHOW terminal
122 0121 1 :     understand, which go in the second char longword
123 0122 1 :     TERMS_TT2_BIT - a VECTOR of bitmasks which correspond to the keywords
124 0123 1 :     described in TERMS_TT_KEY
125 0124 1 :     TERMS_xxxx - where "xxxx" is the terminal type/name. numerous
126 0125 1 :     GLOBAL LITERALS whose value corresponds to their
127 0126 1 :     position (index?) into the terminal tables
: 128 0127 1 :

```

```

130 0128 1 LIBRARY 'SYSS$LIBRARY:LIB';
131 0129 1
132 0130 1
133 0131 1 COMPILETIME
134 0132 1     index = 0,
135 0133 1     count = 0;
136 0134 1
137 0135 1 MACRO
138 0136 1     tt$_dummy1 = 0%,
139 0137 1     tt$_dummy2 = 0%,
140 0138 1     tt$_dummy3 = 0%,
141 0139 1     tt$_dummy4 = 0%,
142 0140 1     tt$_dummy5 = 0%,
143 0141 1     tt$_dummy6 = 0%,
144 0142 1     tt$_dummy7 = 0%,
145 0143 1     tt$_dummy8 = 0%;
146 0144 1
147 0145 1
148 0146 1     Auxiliary macros invoked by the keyword macros
149 0147 1
150 0148 1 MACRO
151 0149 1
152 0150 1     Create a bitmask
153 0151 1
154 0152 1     make_mask (prefix) [item] = %NAME(prefix, item)%,
155 0153 1
156 0154 1
157 0155 1     Create a table pointing to terminal names
158 0156 1
159 0157 1     resp_desc [string] = %ASCII %STRING(string)%,
160 0158 1
161 0159 1
162 0160 1     Figure out how many strings in a macro
163 0161 1
164 0162 1     count_strings [string] = %ASSIGN(count, count+1)%,
165 0163 1
166 0164 1
167 0165 1     Create a specific bitmask, given the prefix and name
168 0166 1
169 0167 1     bitmask (prefix) [key,item] = %NAME(prefix, '$m_', item)%,
170 0168 1
171 0169 1
172 0170 1     Another descriptor macro
173 0171 1
174 0172 1     make_keyword [key, item] = %ASCII %STRING(key)%,
175 0173 1
176 0174 1
177 0175 1     Make a table of CLI keywords, and their corresponding TT(2)$M_name
178 0176 1     bitmask. This macro will produce:
179 0177 1     TERMS_TT[CLR/SET]_NUM - count of 1st longword bits to clear or set
180 0178 1     TERMS_TT[CLR/SET]_NUM - count of 2nd longword bits to clear or set
181 0179 1     TERMS_TT(2)[CLR/SET]_KEY - table of ASCII keywords
182 0180 1     TERMS_TT(2)[CLR/SET]_BIT - corresponding bits to set or clear
183 0181 1
184 M 0182 1     bit_table (prefix, type) =
185 M 0183 1     GLOBAL [LITERAL %NAME('term$', prefix, .type, '_num') = (%LENGTH - 2)/2;
186 M 0184 1     GLOBAL %NAME('term$', prefix, .type, '_key') : VECTOR[(%LENGTH-2)/2]

```

! VAX/VMS system definitions

TERMDEFS  
V04-000

```
.. 187      M 0185 1      INITIAL(make_keyword(%REMAINING));  
.. 188      M 0186 1      GLOBAL %NAME('term$_', prefix, 'type', 'bit') : VECTOR[(%LENGTH-2)/2]  
.. 189      0187 1      INITIAL(bitmask(%EXPAND prefix, %REMAINING));%  
.. 190      0188 1  
.. 191      0189 1  
.. 192      0190 1      ;  
.. 193      0191 1      Make a table of terminal request strings. These are strings which  
.. 194      0192 1      terminals respond to with identifying strings.  
.. 195      M 0193 1      request table (request) =  
.. 196      M 0194 1      GLOBAL LITERAL term$_regnum = %LENGTH;  
.. 197      M 0195 1      GLOBAL term$_reqblk : VECTOR[term$_regnum]  
.. 198      0196 1      INITIAL(resp_desc(%EXPAND request, %REMAINING));%  
.. 199      0197 1  
.. 200      0198 1  
.. 201      0199 1      ;  
.. 202      0200 1  
.. 203      0201 1
```

```

205      0202 1 |
206      0203 1 | The following keyword macros create various structures which are accessible
207      0204 1 | to both SET TERMINAL and SHOW TERMINAL.
208      0205 1 |
209      0206 1 |
210      0207 1 |
211      0208 1 | TERM_INDEX defines the literals which are used as an index to a
212      0209 1 | bitvector, to define each terminal type.
213      0210 1 |
214      0211 1 | KEYWORDMACRO
215      M 0212 1 |   term_index (type, ascnam, length, width, set1, clear1, set2, clear2, resp) =
216      M 0213 1 |   GLOBAL LITERAL %NAME('term$_', type) = %NUMBER(index);
217      M 0214 1 |   %ASSIGN(index, index+1)%;
218      0215 1 |
219      0216 1 |
220      0217 1 | NAME_BLOCK creates a descriptor for the name of a terminal
221      0218 1 |
222      0219 1 | KEYWORDMACRO
223      M 0220 1 |   name_block (type,
224      M 0221 1 |   ascnam,
225      M 0222 1 |   length=0,
226      M 0223 1 |   width=0,
227      M 0224 1 |   set1,
228      M 0225 1 |   clear1,
229      M 0226 1 |   set2,
230      M 0227 1 |   clear2,
231      M 0228 1 |   resp) =
232      M 0229 1 |   %if %NULL(ascnam)
233      M 0230 1 |   %THEN %ASCID %STRING(type)
234      M 0231 1 |   %ELSE %ASCID %STRING(ascnam)
235      M 0232 1 |   %FI
236      M 0233 1 |   %;
237      0234 1 |
238      0235 1 |
239      0236 1 | RESP_BLOCK creates a vector of terminal response code descriptors,
240      0237 1 | as well as a literal that tells how many different codes there are,
241      0238 1 | for a specified terminal.
242      0239 1 |
243      0240 1 | KEYWORDMACRO
244      M 0241 1 |   resp_block (type,
245      M 0242 1 |   ascnam,
246      M 0243 1 |   length=0,
247      M 0244 1 |   width=0,
248      M 0245 1 |   set1,
249      M 0246 1 |   clear1,
250      M 0247 1 |   set2,
251      M 0248 1 |   clear2,
252      M 0249 1 |   resp) =
253      M 0250 1 |
254      M 0251 1 |   %IF NOT %NULL(resp)
255      M 0252 1 |   %THEN
256      M 0253 1 |   %ASSIGN(count, 0)
257      M 0254 1 |   count strings(%REMOVE(resp))
258      M 0255 1 |   LITERAL %NAME(type, '_num') = count;
259      M 0256 1 |   OWN %NAME(type, '_block') : VECTOR[count]
260      M 0257 1 |   INITIAL(resp_desc(%REMOVE(resp)));
261      M 0258 1 |   %FI

```



```

: 262 0259 1
: 263 0260 1
: 264 0261 1
: 265 0262 1
: 266 0263 1
: 267 0264 1
: 268 0265 1
: 269 0266 1
: 270 0267 1
: 271 0268 1
: 272 0269 1
: 273 0270 1
: 274 0271 1
: 275 0272 1
: 276 0273 1
: 277 0274 1
: 278 0275 1
: 279 0276 1
: 280 0277 1
: 281 0278 1
: 282 0279 1
: 283 0280 1
: 284 0281 1
: 285 0282 1
: 286 0283 1
: 287 0284 1
: 288 0285 1
: 289 0286 1
: 290 0287 1
: 291 0288 1
: 292 0289 1
: 293 M 0290 1
: 294 M M 0291 1
: 295 M M M 0292 1
: 296 M M M 0293 1
: 297 M M M 0294 1
: 298 M M M 0295 1
: 299 M M M 0296 1
: 300 M M M 0297 1
: 301 M M M 0298 1
: 302 M M M 0299 1
: 303 M M M 0300 1
: 304 M M M 0301 1
: 305 M M M 0302 1
: 306 M M M 0303 1
: 307 M M M 0304 1
: 308 M M M 0305 1
: 309 M M M 0306 1
: 310 M M M 0307 1
: 311 M M M 0308 1
: 312 M M M 0309 1
: 313 M M M 0310 1
: 314 M M M 0311 1
: 315 M M M 0312 1
: 316 M M M 0313 1
: 317 M M M 0314 1
: 318 M M M 0315 1

```

```

%:
TERM_BLOCK creates a blockvector table. Each entry in the table consists
of a block that looks like this :

```

```

+-----+-----+
| page width | type |
+-----+-----+
| length | bits to set in 1st l.w. |
+-----+-----+
|          | bits to set in 2nd l.w. |
+-----+-----+
|          | bits to clear in 1st l.w. |
+-----+-----+
|          | bits to clear in 2nd l.w. |
+-----+-----+
|          | number of response strings |
+-----+-----+
|          | address of response string |
+-----+-----+

```

KEYWORDMACRO

```

term_block (type,
            ascnam,
            length=0,
            width=0,
            set1,
            clear1,
            set2,
            clear2,
            resp) =
(width^16) OR %NAME('tc$', type)^8,
%IF %NULL(set1)
%THEN length^24 OR tt$m_wrap OR tt$m_ttsync
%ELSE length^24 OR tt$m_wrap OR tt$m_ttsync
OR make_mask(tt$m_, %REMOVE(set1))
%FI,
%IF %NULL(set2)
%THEN 0
%ELSE 0 OR make_mask(tt2$m_, %REMOVE(set2))
%FI,
%IF %NULL(clear1)
%THEN tt$m_lfill OR tt$m_scope OR tt$m_holdscreen OR
tt$m_readsync OR
tt$m_notypeahd OR tt$m_mechtab OR tt$m_ttsync OR
tt$m_wrap OR tt$m_mechform
%ELSE tt$m_lfill OR tt$m_scope OR tt$m_holdscreen OR
tt$m_readsync OR
tt$m_notypeahd OR tt$m_mechtab OR tt$m_ttsync OR
tt$m_wrap OR tt$m_mechform OR
make_mask(tt$m_, %REMOVE(clear1))
%FI,

```

```

319      M 0316 1
320      M 0317 1      %IF %NULL(clear2)
321      M 0318 1      %THEN tt2$m_sixel OR tt2$m_printer OR tt2$m_drct
322      M 0319 1      %ELSE tt2$m_sixel OR tt2$m_printer OR tt2$m_drct OR
323      M 0320 1      make_mask(tt2$m_, %REMOVE(clear2))
324      M 0321 1      %FI,
325      M 0322 1
326      M 0323 1      %IF %NULL(resp)
327      M 0324 1      %THEN 0,0
328      M 0325 1      %ELSE %NAME(type, '_num'), %NAME(type, '_block')
329      M 0326 1      %FI
330      M 0327 1      %:
331      M 0328 1
332      M 0329 1      MACRO
333      M 0330 1      :
334      M 0331 1      Process one terminal, making one NAME_BLOCK entry
335      M 0332 1      :
336      M 0333 1      make_name_block [one terminal] =
337      M 0334 1      name_block(%REMOVE(one_terminal))%,
338      M 0335 1      :
339      M 0336 1      Process one terminal, making one TERM_BLOCK block
340      M 0337 1      :
341      M 0338 1      make_term_block [one terminal] =
342      M 0339 1      term_block(%REMOVE(one_terminal))%,
343      M 0340 1      :
344      M 0341 1      Process one terminal, making one TERM_INDEX literal
345      M 0342 1      :
346      M 0343 1      make_term_index [one terminal] =
347      M 0344 1      term_index(%REMOVE(one_terminal))%,
348      M 0345 1      :
349      M 0346 1      Process one terminal, making a response block
350      M 0347 1      :
351      M 0348 1      make_resp_block [one terminal] =
352      M 0349 1      resp_block(%REMOVE(one_terminal))%,
353      M 0350 1      :
354      M 0351 1      Here's the master macro, that calls all the other ones.
355      M 0352 1      :
356      M 0353 1      define types (list) =
357      M 0354 1      %ASSIGN(index, 0)
358      M 0355 1      make_term_index(%EXPAND list, %REMAINING)
359      M 0356 1      make_resp_block(%EXPAND list, %REMAINING)
360      M 0357 1      GLOBAL LITERAL term$num = %LENGTH;
361      M 0358 1      GLOBAL
362      M 0359 1      term$name : VECTOR[term$num]
363      M 0360 1      INITIAL(make_name_block(%EXPAND list, %REMAINING)),
364      M 0361 1      term$table : BLOCKVECTOR[term$num, 28, BYTE]
365      M 0362 1      INITIAL(make_term_block(%EXPAND list, %REMAINING)); % :
366      M 0363 1
367      M 0364 1
368      M 0365 1
369      M 0366 1
370      M 0367 1

```

```

372      0368 1 |
373      0369 1 | Define the terminal types. NOTE THAT THE FIRST 15 TERMINAL TYPES ARE IN
374      0370 1 | ORDER, AND SHOULD NOT BE CHANGED.
375      0371 1 |
376      P 0372 1 | define_types(
377      P P 0373 1 |
378      P P 0374 1 |     (type = la36,
379      P P 0375 1 |       width = 132,
380      P P 0376 1 |       leng.h = 66,
381      P P 0377 1 |       set1 = lower,
382      P P 0378 1 |       clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
383      P P 0379 1 |
384      P P 0380 1 |     (type = la120,
385      P P 0381 1 |       width = 132,
386      P P 0382 1 |       length = 66,
387      P P 0383 1 |       set1 = (mechform, lower, mechtabs),
388      P P 0384 1 |       clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
389      P P 0385 1 |       resp = '[?2;')',
390      P P 0386 1 |
391      P P 0387 1 |     (type = vt05,
392      P P 0388 1 |       width = 72,
393      P P 0389 1 |       length = 20,
394      P P 0390 1 |       set1 = (scope, lffill, mechtabs),
395      P P 0391 1 |       clear1 = (holdscreen, lower),
396      P P 0392 1 |       clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
397      P P 0393 1 |
398      P P 0394 1 |     (type = vt52,
399      P P 0395 1 |       width = 80,
400      P P 0396 1 |       length = 24,
401      P P 0397 1 |       set1 = (lower, scope, mechtabs),
402      P P 0398 1 |       clear1 = (holdscreen, mechform),
403      P P 0399 1 |       clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
404      P P 0400 1 |       resp = ('/K', '/L', '/Z', '/A', '/H', '/J', '/B', '/J', '/M', '/Q', '/R', '/S', '/T')),
405      P P 0401 1 |
406      P P 0402 1 |     (type = vt55,
407      P P 0403 1 |       width = 80,
408      P P 0404 1 |       length = 24,
409      P P 0405 1 |       set1 = (mechtabs, scope, lower),
410      P P 0406 1 |       clear1 = (holdscreen, mechform),
411      P P 0407 1 |       clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
412      P P 0408 1 |
413      P P 0409 1 |     (type = vt100,
414      P P 0410 1 |       width = 80,
415      P P 0411 1 |       length = 24,
416      P P 0412 1 |       set1 = (mechtabs, scope, lower),
417      P P 0413 1 |       clear1 = (mechform),
418      P P 0414 1 |       set2 = (deccrt, ansicrt, avo),
419      P P 0415 1 |       clear2 = (deccrt2, regis, block, edit),
420      P P 0416 1 |       resp = '[?1;')',
421      P P 0417 1 |
422      P P 0418 1 |     (type = ft1),
423      P P 0419 1 |     (type = ft2),
424      P P 0420 1 |     (type = ft3),
425      P P 0421 1 |     (type = ft4),
426      P P 0422 1 |     (type = ft5),
427      P P 0423 1 |     (type = ft6),
428      P P 0424 1 |     (type = ft7).

```

```

429 P 0425 1 (type = ft8),
430 P 0426 1
431 P 0427 1 (type = unknown,
432 P 0428 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
433 P 0429 1
434 P 0430 1 (type = la34,
435 P 0431 1 width = 132,
436 P 0432 1 length = 66,
437 P 0433 1 set1 = (mechform, lower),
438 P 0434 1 clear1 = mechtap,
439 P 0435 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
440 P 0436 1 resp = '[?3;')',
441 P 0437 1
442 P 0438 1 (type = la38,
443 P 0439 1 width = 132,
444 P 0440 1 length = 66,
445 P 0441 1 set1 = (mechform, lower),
446 P 0442 1 clear1 = mechtap,
447 P 0443 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
448 P 0444 1
449 P 0445 1 (type = la100,
450 P 0446 1 width = 80,
451 P 0447 1 length = 66,
452 P 0448 1 set1 = (mechform, lower, mechtap),
453 P 0449 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
454 P 0450 1 resp = '[?10;')',
455 P 0451 1
456 P 0452 1 (type = la12,
457 P 0453 1 width = 80,
458 P 0454 1 length = 66,
459 P 0455 1 set1 = (mechform, lower, mechtap),
460 P 0456 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
461 P 0457 1 resp = '[?15;')',
462 P 0458 1
463 P 0459 1 (type = la24,
464 P 0460 1 width = 132,
465 P 0461 1 length = 66,
466 P 0462 1 set1 = (mechform, lower, mechtap),
467 P 0463 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
468 P 0464 1
469 P 0465 1 (type = TQ BTS,
470 P 0466 1 ascnam = BTS,
471 P 0467 1 width = 80,
472 P 0468 1 length = 23,
473 P 0469 1 set1 = (mechform, lower),
474 P 0470 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing)),
475 P 0471 1
476 P 0472 1 (type = lgp02,
477 P 0473 1 width = 132,
478 P 0474 1 length = 66,
479 P 0475 1 set1 = (mechform, lower, mechtap),
480 P 0476 1 clear2 = (deccrt2, decprt, ansicrt, regis, block, avo, edit, editing),
481 P 0477 1 resp = '[?13;')',
482 P 0478 1
483 P 0479 1 (type = vk100,
484 P 0480 1 width = 84,
485 P 0481 1 length = 24,

```

```

: 486      P 0482 1      set1 = (mehtab, scope, lower),
: 487      P 0483 1      clear1 = (mechform),
: 488      P 0484 1      set2 = ansicrt,
: 489      P 0485 1      clear2 = (deccrt2, block, deccrt, avo, edit, regis, editing),
: 490      P 0486 1      resp = '[?5;]',
: 491      P 0487 1
: 492      P 0488 1      (type = vt101,
: 493      P 0489 1      width = 80,
: 494      P 0490 1      length = 24,
: 495      P 0491 1      set1 = (mehtab, scope, lower),
: 496      P 0492 1      clear1 = (mechform),
: 497      P 0493 1      set2 = (deccrt, ansicrt),
: 498      P 0494 1      clear2 = (deccrt2, regis, block, avo, edit),
: 499      P 0495 1      resp = '[?1;]',
: 500      P 0496 1
: 501      P 0497 1      (type = vt102,
: 502      P 0498 1      width = 80,
: 503      P 0499 1      length = 24,
: 504      P 0500 1      set1 = (mehtab, scope, lower),
: 505      P 0501 1      clear1 = (mechform),
: 506      P 0502 1      set2 = (deccrt, ansicrt, avo, edit, printer),
: 507      P 0503 1      clear2 = (deccrt2, regis, block),
: 508      P 0504 1      resp = ('[?6;', '[?61;', '?61;)'),
: 509      P 0505 1
: 510      P 0506 1      (type = vt105,
: 511      P 0507 1      width = 80,
: 512      P 0508 1      length = 24,
: 513      P 0509 1      set1 = (mehtab, scope, lower),
: 514      P 0510 1      clear1 = (mechform),
: 515      P 0511 1      set2 = (deccrt, ansicrt),
: 516      P 0512 1      clear2 = (deccrt2, regis, block),
: 517      P 0513 1      resp = '[?1;]',
: 518      P 0514 1
: 519      P 0515 1      (type = vt125,
: 520      P 0516 1      width = 80,
: 521      P 0517 1      length = 24,
: 522      P 0518 1      set1 = (mehtab, scope, lower),
: 523      P 0519 1      clear1 = (hostsync, mechform),
: 524      P 0520 1      set2 = (deccrt, ansicrt, regis, avo, printer, sixel),
: 525      P 0521 1      clear2 = (block, deccrt2, edit),
: 526      P 0522 1      resp = '[?12;]',
: 527      P 0523 1
: 528      P 0524 1      (type = vt131,
: 529      P 0525 1      width = 80,
: 530      P 0526 1      length = 24,
: 531      P 0527 1      set1 = (hostsync, mehtab, scope, lower),
: 532      P 0528 1      clear1 = mechform,
: 533      P 0529 1      set2 = (deccrt, ansicrt, block, edit, avo),
: 534      P 0530 1      clear2 = (deccrt2, regis),
: 535      P 0531 1      resp = '[?7;]',
: 536      P 0532 1
: 537      P 0533 1      (type = vt132,
: 538      P 0534 1      width = 80,
: 539      P 0535 1      length = 24,
: 540      P 0536 1      set1 = (hostsync, mehtab, scope, lower),
: 541      P 0537 1      clear1 = mechform,
: 542      P 0538 1      set2 = (deccrt, ansicrt, block, avo, edit),

```

```

543 P 0539 1 clear2 = ( decprt2, regis),
544 P 0540 1 resp = '['?4;'),
545 P 0541 1
546 P 0542 1 (type = vt173,
547 P 0543 1 width = 80,
548 P 0544 1 length = 16,
549 P 0545 1 set1 = (hostsync, mehtab, scope, lower),
550 P 0546 1 clear1 = (mechform',
551 P 0547 1 set2 = block,
552 P 0548 1 clear2 = (decprt2, decprt, ansicrt, regis, editing),
553 P 0549 1 resp = ';'),
554 P 0550 1
555 P 0551 1 (type = VT200_Series,
556 P 0552 1 width = 80,
557 P 0553 1 length = 24,
558 P 0554 1 set1 = (mehtab, scope, lower, eightbit),
559 P 0555 1 clear1 = (mechform),
560 P 0556 1 set2 = (decprt2, decprt, ansicrt, avo, edit),
561 P 0557 1 clear2 = (fallback, regis, block),
562 P 0558 1 resp = ('[?62;', '?62;')),
563 P 0559 1
564 P 0560 1 (type = PRO_Series,
565 P 0561 1 width = 80,
566 P 0562 1 length = 24,
567 P 0563 1 set1 = (mehtab, scope, lower),
568 P 0564 1 clear1 = (mechform),
569 P 0565 1 set2 = (decprt, ansicrt, avo, edit, printer),
570 P 0566 1 clear2 = (decprt2, fallback, block),
571 P 0567 1 resp = ('?23;', '?23;', '['?23;', '['?22;', '?22;')),
572 P 0568 1
573 P 0569 1
574 P 0570 1
575 P 0571 1
576 P 0572 1
577 P 0573 1
578 P 0574 1
579 P 0575 1
580 P 0576 1
581 P 0577 1
582 P 0578 1
583 P 0579 1
584 P 0580 1
585 P 0581 1
586 P 0582 1
587 P 0583 1

```

Add some more dummy terminal types, in the event that other terminals must be supported between releases.

```

(type = dummy1),
(type = dummy2),
(type = dummy3),
(type = dummy4),
(type = dummy5),
(type = dummy6),
(type = dummy7),
(type = dummy8)
);

```

```

589      0584 1 |
590      0585 1 | Define the keywords and the corresponding bits in the
591      0586 1 | two device-dependent longwords.
592      0587 1 |
593      P 0588 1 | bit_table(tt, set,
594      P 0589 1 |     eight_bit,      eightbit,
595      P 0590 1 |     escape,         escape,
596      P 0591 1 |     form,           mechform,
597      P 0592 1 |     halfdup,        halfdup,
598      P 0593 1 |     hold_screen,    holdscreen,
599      P 0594 1 |     hostsync,       hostsync,
600      P 0595 1 |     lowercase,      lower,
601      P 0596 1 |     modem,          modem,
602      P 0597 1 |     passall,        passall,
603      P 0598 1 |     readsync,       readsync,
604      P 0599 1 |     scope,          scope,
605      P 0600 1 |     scribe,         script,
606      P 0601 1 |     tab,            mehtab,
607      P 0602 1 |     ttsync,         ttsync,
608      0603 1 |     wrap,           wrap);
609      0604 1 |
610      P 0605 1 | bit_table(tt, clr,
611      P 0606 1 |     uppercase,      lower,
612      P 0607 1 |     hardcopy,       scope,
613      P 0608 1 |     interactive,   passall,
614      P 0609 1 |     fulldup,        halfdup,
615      P 0610 1 |     broadcast,      nobrdcst,
616      P 0611 1 |     echo,           noecho,
617      P 0612 1 |     typeahead,     notypeahd
618      0613 1 | );
619      0614 1 |
620      P 0615 1 | bit_table(tt2, set,
621      P 0616 1 |     advanced_video, avo,
622      P 0617 1 |     altypeahd,     altypeahd,
623      P 0618 1 |     ansi_crt,      ansicrt,
624      P 0619 1 |     application_keypad, app_keypad,
625      P 0620 1 |     autobaud,      autobaud,
626      P 0621 1 |     block_mode,    block,
627      P 0622 1 |     brdcstmbx,     brdcstmbx,
628      P 0623 1 |     disconnect,    disconnect,
629      P 0624 1 |     dialup,        dialup,
630      P 0625 1 |     edit,          edit,
631      P 0626 1 |     fallback,      fallback,
632      P 0627 1 |     hangup,        hangup,
633      P 0628 1 |     line_edit,     editing,
634      P 0629 1 |     insert_editing, insert,
635      P 0630 1 |     local_echo,    localecho,
636      P 0631 1 |     printer_port,  printer,
637      P 0632 1 |     pasthru,       pasthru,
638      P 0633 1 |     regis,         regis,
639      P 0634 1 |     secure_server, secure,
640      P 0635 1 |     sixel,         sixel,
641      P 0636 1 |     soft_characters, drcs,
642      P 0637 1 |     syspassword,   syspwd,
643      P 0638 1 |     xon,           xon,
644      P 0639 1 |     dma,           dma);
645      P 0640 1 | bit_table(tt2, clr,

```

```

! qualifiers that have
! duplicate sets should be
! in the beginning of this list
! if you add 1 you must change the count
! in LOG_RESULTS

```

! SEE TT bit table for important information

TERMDEFS  
V04-000

D 15  
16-Sep-1984 01:16:30  
14-Sep-1984 12:09:59

VAX-11 Bliss-32 V4.0-742  
[CLIUTL.SRC]TERMDEFS.B32;1

Page 14  
(6)

```
: 646      P 0641 1      numeric_keypad, app_keypad,  
: 647      P 0642 1      overstrike,      insert,  
: 648      0643 1      set_speed,       set_speed);
```



```

: 650      0644 1  |
: 651      0645 1  | List all the request strings that terminals might respond to
: 652      0646 1  | first is for vt52 style
: 653      0647 1  | second is the VT100 style
: 654      0648 1  | third is the vt200 style
: 655      0649 1  |
: 656      PP 0650 1  | request_table ('z'
: 657      0651 1  |             '[c'
: 658      0652 1  |             '[0c');
: 659      0653 1  |
: 660      0654 1  |
: 661      0655 1  | Create a table of all terminal speeds
: 662      0656 1  |
: 663      0657 1  | GLOBAL LITERAL term$_spdnum = 17;
: 664      0658 1  | GLOBAL term$_spdbl : VECTOR[17]
: 665      0659 1  |     INITIAL(%ASCID '0', %ASCID '50', %ASCID '75'
: 666      0660 1  |             %ASCID '110', %ASCID '134', %ASCID '150',
: 667      0661 1  |             %ASCID '300', %ASCID '600', %ASCID '1200',
: 668      0662 1  |             %ASCID '1800', %ASCID '2000', %ASCID '2400',
: 669      0663 1  |             %ASCID '3600', %ASCID '4800', %ASCID '7200',
: 670      0664 1  |             %ASCID '9600', %ASCID '19200'),
: 671      0665 1  |
: 672      0666 1  | Declare the even/odd Descriptors
: 673      0667 1  |
: 674      0668 1  |     term$_even : VECTOR[2]
: 675      0669 1  |             INITIAL(%CHARCOUNT ('EVEN'),
: 676      0670 1  |             UPLIT BYTE ('EVEN')),
: 677      0671 1  |     term$_odd  : VECTOR[2]
: 678      0672 1  |             INITIAL(%CHARCOUNT ('ODD'),
: 679      0673 1  |             UPLIT BYTE ('ODD')),
: 680      0674 1  |     term$_none : VECTOR[2]
: 681      0675 1  |             INITIAL(%CHARCOUNT ('NONE'),
: 682      0676 1  |             UPLIT BYTE ('NONE')),
: 683      0677 1  |
: 684      0678 1  | ! Declare the protocol descriptors
: 685      0679 1  |
: 686      0680 1  |     protocol$_none : VECTOR[2]
: 687      0681 1  |             INITIAL(%CHARCOUNT ('NONE'),
: 688      0682 1  |             UPLIT BYTE ('NONE')),
: 689      0683 1  |     protocol$_ddcmp : VECTOR[2]
: 690      0684 1  |             INITIAL(%CHARCOUNT ('DDCMP'),
: 691      0685 1  |             UPLIT BYTE ('DDCMP'));

```

: 693  
: 694  
0686 1 END  
0687 0 ELUDOM

```

.TITLE TERMDEFS
.IDENT \V04-000\

.PSECT $PLITS,NOWRT,NOEXE,2

3B 32 3F 5B 00000 P.AAB: .ASCII \[?2;\
      010E0004 00004 P.AAA: .LONG 17694724
      00000000' 00008 P.AAA: .ADDRESS P.AAB
00 00 4B 2F 0000C P.AAD: .ASCII \K\<0><0>
      010E0002 00010 P.AAC: .LONG 17694722
      00000000' 00014 P.AAD: .ADDRESS P.AAD
00 00 4C 2F 00018 P.AAF: .ASCII \L\<0><0>
      010E0002 0001C P.AAE: .LONG 17694722
      00000000' 00020 P.AAF: .ADDRESS P.AAF
00 00 5A 2F 00024 P.AAH: .ASCII \Z\<0><0>
      010E0002 00028 P.AAG: .LONG 17694722
      00000000' 0002C P.AAH: .ADDRESS P.AAH
00 00 41 2F 00030 P.AAJ: .ASCII \A\<0><0>
      010E0002 00034 P.AAI: .LONG 17694722
      00000000' 00038 P.AAJ: .ADDRESS P.AAJ
00 00 48 2F 0003C P.AAL: .ASCII \H\<0><0>
      010E0002 00040 P.AAK: .LONG 17694722
      00000000' 00044 P.AAL: .ADDRESS P.AAL
00 00 4A 2F 00048 P.AAN: .ASCII \J\<0><0>
      010E0002 0004C P.AAM: .LONG 17694722
      00000000' 00050 P.AAN: .ADDRESS P.AAN
00 00 42 2F 00054 P.AAP: .ASCII \B\<0><0>
      010E0002 00058 P.AAO: .LONG 17694722
      00000000' 0005C P.AAP: .ADDRESS P.AAP
00 00 4A 2F 00060 P.AAR: .ASCII \J\<0><0>
      010E0002 00064 P.AAQ: .LONG 17694722
      00000000' 00068 P.AAR: .ADDRESS P.AAR
00 00 4D 2F 0006C P.AAT: .ASCII \M\<0><0>
      010E0002 00070 P.AAS: .LONG 17694722
      00000000' 00074 P.AAT: .ADDRESS P.AAT
00 00 51 2F 00078 P.AAV: .ASCII \Q\<0><0>
      010E0002 0007C P.AAU: .LONG 17694722
      00000000' 00080 P.AAV: .ADDRESS P.AAV
00 00 52 2F 00084 P.AAX: .ASCII \R\<0><0>
      010E0002 00088 P.AAW: .LONG 17694722
      00000000' 0008C P.AAX: .ADDRESS P.AAX
00 00 53 2F 00090 P.AAZ: .ASCII \S\<0><0>
      010E0002 00094 P.AAY: .LONG 17694722
      00000000' 00098 P.AAZ: .ADDRESS P.AAZ
00 00 54 2F 0009C P.ABB: .ASCII \T\<0><0>
      010E0002 000A0 P.ABA: .LONG 17694722
      00000000' 000A4 P.ABB: .ADDRESS P.ABB
3B 31 3F 5B 000A8 P.ABD: .ASCII \[?1;\
      010E0004 000AC P.ABC: .LONG 17694724
      00000000' 000B0 P.ABD: .ADDRESS P.ABD
3B 33 3F 5B 000B4 P.ABF: .ASCII \[?3;\
      010E0004 000B8 P.ABE: .LONG 17694724
      00000000' 000BC P.ABE: .ADDRESS P.ABF

```

.....

00	00	00	3B	30	31	3F 5B	000C0	P.ABH:	.ASCII	\[?10:\<0><0><0>	
						010E0005	000C8	P.ABG:	.LONG	17694725	
						00000000	000CC		.ADDRESS	P.ABH	
00	00	00	3B	35	31	3F 5B	000D0	P.ABJ:	.ASCII	\[?15:\<0><0><0>	
						010E0005	000D8	P.ABI:	.LONG	17694725	
						00000000	000DC		.ADDRESS	P.ABJ	
00	00	00	3B	33	31	3F 5B	000E0	P.ABL:	.ASCII	\[?13:\<0><0><0>	
						010E0005	000E8	P.ABK:	.LONG	17694725	
						00000000	000EC		.ADDRESS	P.ABL	
					38	35	3F 5B	000F0	P.ABN:	.ASCII	\[?5:\
						010E0004	000F4	P.ABM:	.LONG	17694724	
						00000000	000F8		.ADDRESS	P.ABN	
					38	31	3F 5B	000FC	P.ABP:	.ASCII	\[?1:\
						010E0004	00100	P.ABO:	.LONG	17694724	
						00000000	00104		.ADDRESS	P.ABP	
					38	36	3F 5B	00108	P.ABR:	.ASCII	\[?6:\
						010E0004	0010C	P.ABQ:	.LONG	17694724	
						00000000	00110		.ADDRESS	P.ABR	
00	00	00	3B	31	36	3F 5B	00114	P.ABT:	.ASCII	\[?61:\<0><0><0>	
						010E0005	0011C	P.ABS:	.LONG	17694725	
						00000000	00120		.ADDRESS	P.ABT	
					38	31	36 3F	00124	P.ABV:	.ASCII	\?61:\
						010E0004	00128	P.ABU:	.LONG	17694724	
						00000000	0012C		.ADDRESS	P.ABV	
					38	31	3F 5B	00130	P.ABX:	.ASCII	\[?1:\
						010E0004	00134	P.ABW:	.LONG	17694724	
						00000000	00138		.ADDRESS	P.ABX	
00	00	00	3B	32	31	3F 5B	0013C	P.ABZ:	.ASCII	\[?12:\<0><0><0>	
						010E0005	00144	P.ABY:	.LONG	17694725	
						00000000	00148		.ADDRESS	P.ABZ	
					38	37	3F 5B	0014C	P.ACB:	.ASCII	\[?7:\
						010E0004	00150	P.ACA:	.LONG	17694724	
						00000000	00154		.ADDRESS	P.ACB	
					38	34	3F 5B	00158	P.ACD:	.ASCII	\[?4:\
						010E0004	0015C	P.ACC:	.LONG	17694724	
						00000000	00160		.ADDRESS	P.ACD	
					00	00	00 3B	00164	P.ACF:	.ASCII	\:\<0><0><0>
						010E0001	00168	P.ACE:	.LONG	17694721	
						00000000	0016C		.ADDRESS	P.ACF	
00	00	00	3B	32	36	3F 5B	00170	P.ACH:	.ASCII	\[?62:\<0><0><0>	
						010E0005	00178	P.ACG:	.LONG	17694725	
						00000000	0017C		.ADDRESS	P.ACH	
					38	32	36 3F	00180	P.ACJ:	.ASCII	\?62:\
						010E0004	00184	P.ACI:	.LONG	17694724	
						00000000	00188		.ADDRESS	P.ACJ	
					38	31	33 32	0018C	P.ACL:	.ASCII	\231:\
						010E0004	00190	P.ACK:	.LONG	17694724	
						00000000	00194		.ADDRESS	P.ACL	
					38	33	32 3F	00198	P.ACN:	.ASCII	\?23:\
						010E0004	0019C	P.ACM:	.LONG	17694724	
						00000000	001A0		.ADDRESS	P.ACN	
00	00	00	3B	33	32	3F 5B	001A4	P.ACP:	.ASCII	\[?23:\<0><0><0>	
						010E0005	001AC	P.ACO:	.LONG	17694725	
						00000000	001B0		.ADDRESS	P.ACP	
00	00	00	3B	32	32	3F 5B	001B4	P.ACR:	.ASCII	\[?22:\<0><0><0>	
						010E0005	001BC	P.ACQ:	.LONG	17694725	
						00000000	001C0		.ADDRESS	P.ACR	

.....

	3B	32	32	3F	001C4	P.ACT:	.ASCII	\?22:\			
					010E0004	001C8	P.ACS:	.LONG	17694724		
					00000000	001CC		.ADDRESS	P.ACT		
	36	33	41	4C	001D0	P.ACIV:	.ASCII	\LA36\			
					010E0004	001D4	P.ACU:	.LONG	17694724		
					00000000	001D8		.ADDRESS	P.ACIV		
00	00	00	30	32	31	41	4C	001DC	P.ACX:	.ASCII	\LA120\<0><0><0>
					010E0005	001E4	P.ACW:	.LONG	17694725		
					00000000	001E8		.ADDRESS	P.ACX		
	35	30	54	56	001EC	P.ACZ:	.ASCII	\VT05\			
					010E0004	001F0	P.ACY:	.LONG	17694724		
					00000000	001F4		.ADDRESS	P.ACZ		
	32	35	54	56	001F8	P.ADB:	.ASCII	\VT52\			
					010E0004	001FC	P.ADA:	.LONG	17694724		
					00000000	00200		.ADDRESS	P.ADB		
	35	35	54	56	00204	P.ADD:	.ASCII	\VT55\			
					010E0004	00208	P.ADC:	.LONG	17694724		
					00000000	0020C		.ADDRESS	P.ADD		
00	00	00	30	30	31	54	56	00210	P.ADF:	.ASCII	\VT100\<0><0><0>
					010E0005	00218	P.ADE:	.LONG	17694725		
					00000000	0021C		.ADDRESS	P.ADF		
	00	31	54	46	00220	P.ADH:	.ASCII	\FT1\<0>			
					010E0003	00224	P.ADG:	.LONG	17694723		
					00000000	00228		.ADDRESS	P.ADH		
	00	32	54	46	0022C	P.ADJ:	.ASCII	\FT2\<0>			
					010E0003	00230	P.ADI:	.LONG	17694723		
					00000000	00234		.ADDRESS	P.ADJ		
	00	33	54	46	00238	P.ADL:	.ASCII	\FT3\<0>			
					010E0003	0023C	P.ADK:	.LONG	17694723		
					00000000	00240		.ADDRESS	P.ADL		
	00	34	54	46	00244	P.ADN:	.ASCII	\FT4\<0>			
					010E0003	00248	P.ADM:	.LONG	17694723		
					00000000	0024C		.ADDRESS	P.ADN		
	00	35	54	46	00250	P.ADP:	.ASCII	\FT5\<0>			
					010E0003	00254	P.ADO:	.LONG	17694723		
					00000000	00258		.ADDRESS	P.ADP		
	00	36	54	46	0025C	P.ADR:	.ASCII	\FT6\<0>			
					010E0003	00260	P.ADQ:	.LONG	17694723		
					00000000	00264		.ADDRESS	P.ADR		
	00	37	54	46	00268	P.ADT:	.ASCII	\FT7\<0>			
					010E0003	0026C	P.ADS:	.LONG	17694723		
					00000000	00270		.ADDRESS	P.ADT		
	00	38	54	46	00274	P.ADV:	.ASCII	\FT8\<0>			
					010E0003	00278	P.ADU:	.LONG	17694723		
					00000000	0027C		.ADDRESS	P.ADV		
00	4E	57	4F	4E	4B	4E	55	00280	P.ADX:	.ASCII	\UNKNOWN\<0>
					010E0007	00288	P.ADW:	.LONG	17694727		
					00000000	0028C		.ADDRESS	P.ADX		
	34	33	41	4C	00290	P.ADZ:	.ASCII	\LA34\			
					010E0004	00294	P.ADY:	.LONG	17694724		
					00000000	00298		.ADDRESS	P.ADZ		
	38	33	41	4C	0029C	P.AEB:	.ASCII	\LA38\			
					010E0004	002A0	P.AEA:	.LONG	17694724		
					00000000	002A4		.ADDRESS	P.AEB		
00	00	00	30	30	31	41	4C	002A8	P.AED:	.ASCII	\LA100\<0><0><0>
					010E0005	002B0	P.AEC:	.LONG	17694725		
					00000000	002B4		.ADDRESS	P.AED		

.....

										32	31	41	4C	002B8	P.AEF:	.ASCII	\LA12\ 010E0004	002BC	P.AEE:	.LONG	17694724		
														00000000			002C0			.ADDRESS	P.AEF		
										34	32	41	4C	002C4	P.AEH:	.ASCII	\LA24\ 010E0004	002C8	P.AEG:	.LONG	17694724		
														00000000			002CC			.ADDRESS	P.AEH		
										00	53	54	42	002D0	P.AEJ:	.ASCII	\BTS\<<0>	002D4	P.AEI:	.LONG	17694723		
														00000000			002D8			.ADDRESS	P.AEJ		
00	00	00								32	30	50	51	4C	002DC	P.AEL:	.ASCII	\LQP02\<<0><0><0>	002E4	P.AEK:	.LONG	17694725	
														00000000			002E8			.ADDRESS	P.AEL		
00	00	00								30	30	31	4B	56	002EC	P.AEN:	.ASCII	\VK100\<<0><0><0>	002F4	P.AEM:	.LONG	17694725	
														00000000			002F8			.ADDRESS	P.AEN		
00	00	00								31	30	31	54	56	002FC	P.AEP:	.ASCII	\VT101\<<0><0><0>	00304	P.AEO:	.LONG	17694725	
														00000000			00308			.ADDRESS	P.AEP		
00	00	00								32	30	31	54	56	0030C	P.AER:	.ASCII	\VT102\<<0><0><0>	00314	P.AEQ:	.LONG	17694725	
														00000000			00318			.ADDRESS	P.AER		
00	00	00								35	30	31	54	56	0031C	P.AET:	.ASCII	\VT105\<<0><0><0>	00324	P.AES:	.LONG	17694725	
														00000000			00328			.ADDRESS	P.AET		
00	00	00								35	32	31	54	56	0032C	P.AEV:	.ASCII	\VT125\<<0><0><0>	00334	P.AEU:	.LONG	17694725	
														00000000			00338			.ADDRESS	P.AEV		
00	00	00								31	33	31	54	56	0033C	P.AEX:	.ASCII	\VT131\<<0><0><0>	00344	P.AEW:	.LONG	17694725	
														00000000			00348			.ADDRESS	P.AEX		
00	00	00								32	33	31	54	56	0034C	P.AEZ:	.ASCII	\VT132\<<0><0><0>	00354	P.AEY:	.LONG	17694725	
														00000000			00358			.ADDRESS	P.AEZ		
00	00	00								33	37	31	54	56	0035C	P.AFB:	.ASCII	\VT173\<<0><0><0>	00364	P.AFA:	.LONG	17694725	
														00000000			00368			.ADDRESS	P.AFB		
53	45	49	52	45	53	5F				30	30	32	54	56	0036C	P.AFD:	.ASCII	\VT200 SERIES\ 010E000C	00378	P.AFC:	.LONG	17694732	
														00000000			0037C			.ADDRESS	P.AFD		
00	00	53	45	49	52	45	53	5F		4F	4F	52	50	00380	P.AFF:	.ASCII	\PRO SERIES\<<0><0>	0038C	P.AFE:	.LONG	17694730		
														00000000			00390			.ADDRESS	P.AFF		
00	00									31	59	4D	4D	55	44	00394	P.AFH:	.ASCII	\DUMMY1\<<0><0>	0039C	P.AFG:	.LONG	17694726
														00000000			003A0			.ADDRESS	P.AFH		
00	00									32	59	4D	4D	55	44	003A4	P.AFJ:	.ASCII	\DUMMY2\<<0><0>	003AC	P.AFI:	.LONG	17694726
														00000000			003B0			.ADDRESS	P.AFJ		
00	00									33	59	4D	4D	55	44	003B4	P.AFL:	.ASCII	\DUMMY3\<<0><0>	003BC	P.AFK:	.LONG	17694726
														00000000			003C0			.ADDRESS	P.AFL		
00	00									34	59	4D	4D	55	44	003C4	P.AFN:	.ASCII	\DUMMY4\<<0><0>	003CC	P.AFM:	.LONG	17694726
														00000000			003D0			.ADDRESS	P.AFN		
00	00									35	59	4D	4D	55	44	003D4	P.AFP:	.ASCII	\DUMMY5\<<0><0>	003DC	P.AFO:	.LONG	17694726
														00000000			003E0			.ADDRESS	P.AFP		

.....

00	00	36	59	4D	4D	55	44	003E4	P.AFR:	.ASCII	\DUMMY6\<0><0>				
						010E0006		003EC	P.AFQ:	.LONG	17694726				
						00000000		003F0		.ADDRESS	P.AFR				
00	00	37	59	4D	4D	55	44	003F4	P.AFT:	.ASCII	\DUMMY7\<0><0>				
						010E0006		003FC	P.AFS:	.LONG	17694726				
						00000000		00400		.ADDRESS	P.AFT				
00	00	38	59	4D	4D	55	44	00404	P.AFV:	.ASCII	\DUMMY8\<0><0>				
						010E0006		0040C	P.AFU:	.LONG	17694726				
						00000000		00410		.ADDRESS	P.AFV				
00	00	00	54	49	42	5F	54	48	47	49	45	00414	P.AFX:	.ASCII	\EIGHT BIT\<0><0><0>
						010E0009		00420	P.AFW:	.LONG	17694729				
						00000000		00424		.ADDRESS	P.AFX				
00	00	45	50	41	43	53	45	00428	P.AFZ:	.ASCII	\ESCAPE\<0><0>				
						010E0006		00430	P.AFY:	.LONG	17694726				
						00000000		00434		.ADDRESS	P.AFZ				
					4D	52	4F	46	00438	P.AGB:	.ASCII	\FORM\<0><0>			
						010E0004		0043C	P.AGA:	.LONG	17694724				
						00000000		00440		.ADDRESS	P.AGB				
00	50	55	44	46	4C	41	48	00444	P.AGD:	.ASCII	\HALFDUP\<0>				
						010E0007		0044C	P.AGC:	.LONG	17694727				
						00000000		00450		.ADDRESS	P.AGD				
00	4E	45	45	52	43	53	5F	44	4C	4F	48	00454	P.AGF:	.ASCII	\HOLD SCREEN\<0>
						010E000B		00460	P.AGE:	.LONG	17694731				
						00000000		00464		.ADDRESS	P.AGF				
43	4E	59	53	54	53	4F	48	00468	P.AGH:	.ASCII	\HOSTSYNC\<0><0>				
						010E0008		00470	P.AGG:	.LONG	17694728				
						00000000		00474		.ADDRESS	P.AGH				
00	00	00	45	53	41	43	52	45	57	4F	4C	00478	P.AGJ:	.ASCII	\LOWERCASE\<0><0><0>
						010E0009		00484	P.AGI:	.LONG	17694729				
						00000000		00488		.ADDRESS	P.AGJ				
00	00	00	4D	45	44	4F	4D	0048C	P.AGL:	.ASCII	\MODEM\<0><0><0>				
						010E0005		00494	P.AGK:	.LONG	17694725				
						00000000		00498		.ADDRESS	P.AGL				
00	4C	4C	41	53	53	41	50	0049C	P.AGN:	.ASCII	\PASSALL\<0>				
						010E0007		004A4	P.AGM:	.LONG	17694727				
						00000000		004A8		.ADDRESS	P.AGN				
43	4E	59	53	44	41	45	52	004AC	P.AGP:	.ASCII	\READSYNC\<0><0>				
						010E0008		004B4	P.AGO:	.LONG	17694728				
						00000000		004B8		.ADDRESS	P.AGP				
00	00	00	45	50	4F	43	53	004BC	P.AGR:	.ASCII	\SCOPE\<0><0><0>				
						010E0005		004C4	P.AGQ:	.LONG	17694725				
						00000000		004C8		.ADDRESS	P.AGR				
00	00	45	42	49	52	43	53	004CC	P.AGT:	.ASCII	\SCRIBE\<0><0>				
						010E0006		004D4	P.AGS:	.LONG	17694726				
						00000000		004D8		.ADDRESS	P.AGT				
					00	42	41	54	004DC	P.AGV:	.ASCII	\TAB\<0>			
						010E0003		004E0	P.AGU:	.LONG	17694723				
						00000000		004E4		.ADDRESS	P.AGV				
00	00	43	4E	59	53	54	54	004E8	P.AGX:	.ASCII	\TTSYNC\<0><0>				
						010E0006		004F0	P.AGW:	.LONG	17694726				
						00000000		004F4		.ADDRESS	P.AGX				
					50	41	52	57	004F8	P.AGZ:	.ASCII	\WRAP\<0><0>			
						010E0004		004FC	P.AGY:	.LONG	17694724				
						00000000		00500		.ADDRESS	P.AGZ				
00	00	00	45	53	41	43	52	45	50	50	55	00504	P.AHB:	.ASCII	\UPPERCASE\<0><0><0>
						010E0009		00510	P.AHA:	.LONG	17694729				
						00000000		00514		.ADDRESS	P.AHB				

.....

59	50	4F	43	44	52	41	48	00518	P.AHD:	.ASCII	\HARDCOPY\ 010E0008							
								00520	P.AHC:	.LONG	17694728 00000000							
00	45	56	49	54	43	41	52	45	54	4E	49	00524		.ADDRESS	P.AHD			
												00528	P.AHF:	.ASCII	\INTERACTIVE\<>			
												00534	P.AHE:	.LONG	17694731 00000000			
00	50	55	44	4C	4C	55	46	00538		.ADDRESS	P.AHF							
												0053C	P.AHH:	.ASCII	\FULLDUP\<>			
												00544	P.AHG:	.LONG	17694727 00000000			
00	00	00	54	53	41	43	44	41	4F	52	42	00548		.ADDRESS	P.AHH			
												0054C	P.AHJ:	.ASCII	\BROADCAST\<><><>			
												00558	P.AHI:	.LONG	17694729 00000000			
									4F	48	43	45	0055C		.ADDRESS	P.AHJ		
												00560	P.AHL:	.ASCII	\ECHO\ 010E0004			
												00564	P.AHK:	.LONG	17694724 00000000			
00	00	00	44	41	45	48	41	45	50	59	54	00568		.ADDRESS	P.AHL			
												0056C	P.AHN:	.ASCII	\TYPEAHEAD\<><><>			
												00578	P.AHM:	.LONG	17694729 00000000			
00	4F	45	44	49	56	5F	44	45	43	4E	41	56	44	41	0057C		.ADDRESS	P.AHM
												00580	P.AHP:	.ASCII	\ADVANCED_VIDEO\<><>			
												0058F						
												00590	P.AHO:	.LONG	17694734 00000000			
00	00	00	44	48	41	45	50	59	54	4C	41	00594		.ADDRESS	P.AHP			
												00598	P.AHR:	.ASCII	\ALTYPEAHD\<><><>			
												005A4	P.AHQ:	.LONG	17694729 00000000			
												005A8		.ADDRESS	P.AHR			
			54	52	43	5F	49	53	4E	41	005AC	P.AHT:	.ASCII	\ANSI CRT\ 010E0008				
												005B4	P.AHS:	.LONG	17694728 00000000			
59	45	4B	5F	4E	4F	49	54	41	43	49	4C	50	50	41	005B8		.ADDRESS	P.AHT
										00	00	44	41	50	005BC	P.AHV:	.ASCII	\APPLICATION_KEYPAD\<><>
												005CB						
												005D0	P.AHU:	.LONG	17694738 00000000			
												005D4		.ADDRESS	P.AHV			
												005D8	P.AHX:	.ASCII	\AUTOBAUD\ 010E0008			
												005E0	P.AHW:	.LONG	17694728 00000000			
00	00	45	44	4F	4D	5F	4B	43	4F	4C	42	005E4		.ADDRESS	P.AHX			
												005E8	P.AHZ:	.ASCII	\BLOCK_MODE\<><>			
												005F4	P.AHY:	.LONG	17694730 00000000			
00	00	00	58	42	4D	54	53	43	44	52	42	005F8		.ADDRESS	P.AHZ			
												005FC	P.AIB:	.ASCII	\BRDCSTMBX\<><><>			
												00608	P.AIA:	.LONG	17694729 00000000			
00	00	54	43	45	4E	4E	4F	43	53	49	44	0060C		.ADDRESS	P.AIB			
												00610	P.AID:	.ASCII	\DISCONNECT\<><>			
												0061C	P.AIC:	.LONG	17694730 00000000			
												00620		.ADDRESS	P.AID			
												00624	P.AIF:	.ASCII	\DIALUP\<><>			
												0062C	P.AIE:	.LONG	17694726 00000000			
												00630		.ADDRESS	P.AIF			
												00634	P.AIH:	.ASCII	\EDIT\ 010E0004			
												00638	P.AIG:	.LONG	17694724 00000000			
												0063C		.ADDRESS	P.AIH			
												00640	P.AIJ:	.ASCII	\FALLBACK\ 010E0008			
												00648	P.AII:	.LONG	17694728 00000000			
												0064C		.ADDRESS	P.AIJ			
												00650	P.AIL:	.ASCII	\HANGUP\<><>			
												00658	P.AIK:	.LONG	17694726 00000000			
00	00	00	54	49	44	45	5F	45	4E	49	4C	0065C		.ADDRESS	P.AIL			
												00660	P.AIN:	.ASCII	\LINE_EDIT\<><><>			

00	47	4E	49	54	49	44	45	5F	54	52	45	53	4E	49	0066C	P.AIM:	.LONG	17694729	
															00670		.ADDRESS	P.AIN	
															00674	P.AIP:	.ASCII	\INSERT_EDITING\<0><0>	
															00683				
															010E000E	00684	P.AIO:	.LONG	17694734
															00000000	00688		.ADDRESS	P.AIP
			00	00	4F	48	43	45	5F	4C	41	43	4F	4C	0068C	P.AIR:	.ASCII	\LOCAL_ECHO\<0><0>	
															010E000A	00698	P.AIQ:	.LONG	17694730
															00000000	0069C		.ADDRESS	P.AIR
			54	52	4F	50	5F	52	45	54	4E	49	52	50	006A0	P.AIT:	.ASCII	\PRINTER_PORT\<0>	
															010E000C	006AC	P.AIS:	.LONG	17694732
															00000000	006B0		.ADDRESS	P.AIT
							00	55	52	48	54	53	41	50	006B4	P.AIV:	.ASCII	\PASTHRU\<0>	
															010E0007	006BC	P.AIU:	.LONG	17694727
															00000000	006C0		.ADDRESS	P.AIV
							00	00	00	53	49	47	45	52	006C4	P.AIX:	.ASCII	\REGIS\<0><0><0>	
															010E0005	006CC	P.AIW:	.LONG	17694725
															00000000	006D0		.ADDRESS	P.AIX
00	00	52	45	56	52	45	53	5F	45	52	55	43	45	53	006D4	P.AIZ:	.ASCII	\SECURE_SERVER\<0><0><0>	
															00	006E3			
															010E000D	006E4	P.AIY:	.LONG	17694733
															00000000	006E8		.ADDRESS	P.AIZ
							00	00	00	4C	45	58	49	53	006EC	P.AJB:	.ASCII	\SIXEL\<0><0><0>	
															010E0005	006F4	P.AJA:	.LONG	17694725
															00000000	006F8		.ADDRESS	P.AJB
53	52	45	54	43	41	52	41	48	43	5F	54	46	4F	53	006FC	P.AJD:	.ASCII	\SOFT_CHARACTERS\<0>	
															00	0070B			
															010E000F	0070C	P.AJC:	.LONG	17694735
															00000000	00710		.ADDRESS	P.AJD
			00	44	52	4F	57	53	53	41	50	53	59	53	00714	P.AJF:	.ASCII	\SYSPASSWORD\<0>	
															010E000B	00720	P.AJE:	.LONG	17694731
															00000000	00724		.ADDRESS	P.AJF
												00	4E	4F	58	00728	P.AJH:	.ASCII	\XON\<0>
															010E0003	0072C	P.AJG:	.LONG	17694723
															00000000	00730		.ADDRESS	P.AJH
												00	41	4D	44	00734	P.AJJ:	.ASCII	\DMA\<0>
															010E0003	00738	P.AJI:	.LONG	17694723
															00000000	0073C		.ADDRESS	P.AJJ
00	44	41	50	59	45	4B	5F	43	49	52	45	4D	55	4E	00740	P.AJL:	.ASCII	\NUMERIC_KEYPAD\<0><0>	
															00	0074F			
															010E000E	00750	P.AJK:	.LONG	17694734
															00000000	00754		.ADDRESS	P.AJL
			00	00	45	4B	49	52	54	53	52	45	56	4F	00758	P.AJN:	.ASCII	\OVERSTRIKE\<0><0>	
															010E000A	00764	P.AJM:	.LONG	17694730
															00000000	00768		.ADDRESS	P.AJN
			00	00	00	44	45	45	50	53	5F	54	45	53	0076C	P.AJP:	.ASCII	\SET_SPEED\<0><0><0>	
															010E0009	00778	P.AJO:	.LONG	17694729
															00000000	0077C		.ADDRESS	P.AJP
												00	00	5A	1B	00780	P.AJR:	.ASCII	<27>\Z\<0><0>
															010E0002	00784	P.AJQ:	.LONG	17694722
															00000000	00788		.ADDRESS	P.AJR
												00	63	5B	1B	0078C	P.AJT:	.ASCII	<27>\[c\<0>
															010E0003	00790	P.AJS:	.LONG	17694723
															00000000	00794		.ADDRESS	P.AJT
												63	30	5B	1B	00798	P.AJV:	.ASCII	<27>\[0c\<0>
															010E0004	0079C	P.AJU:	.LONG	17694724
															00000000	007A0		.ADDRESS	P.AJV

.....



```

00 00 00 30 007A4 P.AJX: .ASCII \0\<0><0><0>
      010E0001 007A8 P.AJW: .LONG 17694721
      00000000 007AC .ADDRESS P.AJX
00 00 30 35 007B0 P.AJZ: .ASCII \50\<0><0>
      010E0002 007B4 P.AJY: .LONG 17694722
      00000000 007B8 .ADDRESS P.AJZ
00 00 35 37 007BC P.AKB: .ASCII \75\<0><0>
      010E0002 007C0 P.AKA: .LONG 17694722
      00000000 007C4 .ADDRESS P.AKB
00 30 31 31 007C8 P.AKD: .ASCII \110\<0>
      010E0003 007CC P.AKE: .LONG 17694723
      00000000 007D0 .ADDRESS P.AKD
00 34 33 31 007D4 P.AKF: .ASCII \134\<0>
      010E0003 007D8 P.AKH: .LONG 17694723
      00000000 007DC .ADDRESS P.AKF
00 30 35 31 007E0 P.AKH: .ASCII \150\<0>
      010E0003 007E4 P.AKG: .LONG 17694723
      00000000 007E8 .ADDRESS P.AKH
00 30 30 33 007EC P.AKJ: .ASCII \300\<0>
      010E0003 007F0 P.AKI: .LONG 17694723
      00000000 007F4 .ADDRESS P.AKJ
00 30 30 36 007F8 P.AKL: .ASCII \600\<0>
      010E0003 007FC P.AKK: .LONG 17694723
      00000000 00800 .ADDRESS P.AKL
30 30 32 31 00804 P.AKN: .ASCII \1200\
      010E0004 00808 P.AKM: .LONG 17694724
      00000000 0080C .ADDRESS P.AKN
30 30 38 31 00810 P.AKP: .ASCII \1800\
      010E0004 00814 P.AKO: .LONG 17694724
      00000000 00818 .ADDRESS P.AKP
30 30 30 32 0081C P.AKR: .ASCII \2000\
      010E0004 00820 P.AKQ: .LONG 17694724
      00000000 00824 .ADDRESS P.AKR
30 30 34 32 00828 P.AKT: .ASCII \2400\
      010E0004 0082C P.AKS: .LONG 17694724
      00000000 00830 .ADDRESS P.AKT
30 30 36 33 00834 P.AKV: .ASCII \3600\
      010E0004 00838 P.AKU: .LONG 17694724
      00000000 0083C .ADDRESS P.AKV
30 30 38 34 00840 P.AKX: .ASCII \4800\
      010E0004 00844 P.AKW: .LONG 17694724
      00000000 00848 .ADDRESS P.AKX
30 30 32 37 0084C P.AKZ: .ASCII \7200\
      010E0004 00850 P.AKY: .LONG 17694724
      00000000 00854 .ADDRESS P.AKZ
30 30 36 39 00858 P.ALB: .ASCII \9600\
      010E0004 0085C P.ALA: .LONG 17694724
      00000000 00860 .ADDRESS P.ALB
00 00 00 30 30 32 39 31 00864 P.ALD: .ASCII \19200\<0><0><0>
      010E0005 0086C P.ALC: .LONG 17694725
      00000000 00870 .ADDRESS P.ALD
4E 45 56 45 00874 P.ALE: .ASCII \EVEN\
      44 44 4F 00878 P.ALF: .ASCII \ODD\
45 4E 4F 4E 0087B P.ALG: .ASCII \NONE\
45 4E 4F 4E 0087F P.ALH: .ASCII \NONE\
50 4D 43 44 44 00883 P.ALI: .ASCII \DDCMP\

```

.....

.PSECT \$OWNS,NOEXE,2

00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00004	LA120_BLOCK: VT52_BLOCK: ADDRESS P.AAA	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	0001C	ADDRESS P.AAC, P.AAE, P.AAG, P.AAI, P.AAK, -	:
					00000000'	00034	P.AAM, P.AAO, P.AAQ, P.AAS, P.AAU, P.AAW, -	:
					00000000'	00038	P.AAY, P.ABA	:
							VT100_BLOCK: ADDRESS P.ABC	:
					00000000'	0003C	LA34_BLOCK: ADDRESS P.ABE	:
					00000000'	00040	LA100_BLOCK: ADDRESS P.ABG	:
					00000000'	00044	LA12_BLOCK: ADDRESS P.ABI	:
					00000000'	00048	LQP02_BLOCK: ADDRESS P.ABK	:
					00000000'	0004C	VK100_BLOCK: ADDRESS P.ABM	:
					00000000'	00050	VT101_BLOCK: ADDRESS P.ABO	:
	00000000'	00000000'	00000000'	00054	VT102_BLOCK: ADDRESS P.ABQ, P.ABS, P.ABU			:
					00000000'	00060	VT105_BLOCK: ADDRESS P.ABW	:
					00000000'	00064	VT125_BLOCK: ADDRESS P.ABY	:
					00000000'	00068	VT131_BLOCK: ADDRESS P.ACA	:
					00000000'	0006C	VT132_BLOCK: ADDRESS P.ACC	:
					00000000'	00070	VT173_BLOCK: ADDRESS P.ACE	:
			00000000'	00000000'	00074	VT200_SERIES_BLOCK: ADDRESS P.ACG, P.ACI		:
	00000000'	00000000'	00000000'	00000000'	0007C	PRO_SERIES_BLOCK: ADDRESS P.ACK, P.ACM, P.ACO, P.ACQ, P.ACS		:

.PSECT \$GLOBAL\$,NOEXE,2

00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00000	TERMS_NAME:: ADDRESS P.ACU, P.ACW, P.ACY, P.ADA, P.ADC, -	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00018	P.ADE, P.ADG, P.ADI, P.ADK, P.ADM, P.ADO, -	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00030	P.ADQ, P.ADS, P.ADU, P.ADW, P.ADY, P.AEA, -	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00048	P.AEC, P.AEE, P.AEG, P.AEI, P.AEK, P.AEM, -	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00060	P.AEO, P.AEQ, P.AES, P.AEU, P.AEW, P.AEY, -	:
00000000'	00000000'	00000000'	00000000'	00000000'	00000000'	00078	P.AFA, P.AFC, P.AFE, P.AFG, P.AFI, P.AFK, -	:
					00000000'	00090	P.AFM, P.AFO, P.AFQ, P.AFS, P.AFU	:
00000000	7F701000	000C5B24	00000000	420002A0	00842000	000A0	TERMS_TABLE:: .LONG 8658944, 1107296928, 0, 809764, -	:
7F701000	000C5B24	00000000	420803A0	00842100	00000000	000B8	2138050560, 0, 0, 8659200, 1107821472, 0, -	:
					00000001	000D0	809764, 2138050560, 1	:
					00000000'	000D4	ADDRESS LA120_BLOCK	:
00000000	7F701000	000C5BA4	00000000	14001B20	00480100	000D8	.LONG 4718848, 335551264, 0, 809892, -	:
7F701000	000C5B24	00000000	180013A0	00504000	00000000	000F0	2138050560, 0, 0, 5259264, 402658208, 0, -	:
					00000000	00108	809764, 2138050560, 13	:

00000000	7F701000	000C5B24	00000000	180013A0	00000000'	0010C
56700000	000C5B24	29000000	180013A0	00506000	00504100	00110
					00000000	00128
					00000001	00140
					00000000'	00144
00000000	00700000	000C5B24	00000000	00000220	00001000	00148
00700000	000C5B24	00000000	00000220	00001100	00000000	00160
000C5B24	00000000	00000220	00001200	00000000	00000000	00178
00000000	00000220	00001300	00000000	00000000	00700000	00190
00000220	00001400	00000000	00000000	00700000	000C5B24	001A8
00001500	00000000	00000000	00700000	000C5B24	00000000	001C0
00000000	00000000	00700000	000C5B24	00000000	00000220	001D8
00000000	00700000	000C5B24	00000000	00000220	00001600	001F0
00700000	000C5B24	00000000	00000220	00001700	00000000	00208
000C5B24	00000000	00000220	00000000	00000000	00000000	00220
00000000	420802A0	00842200	00000000	00000000	7F701000	00238
			00000001	7F701000	000C5B24	00250
					00000000'	0025C
00000000	7F701000	000C5B24	00000000	420802A0	00842300	00260
7F701000	000C5B24	00000000	420803A0	00502500	00000000	00278
					00000001	00290
					00000000'	00294
00000001	7F701000	000C5B24	00000000	420803A0	00502400	00298
					00000000'	002B0
00000000	7F701000	000C5B24	00000000	420803A0	00842500	002B4
7F701000	000C5B24	00000000	170802A0	00500400	00000000	002CC
000C5B24	00000000	420803A0	00842600	00000000	00000000	002E4
				00000001	7F701000	002FC
					00000000'	00304
00000001	7E701000	000C5B24	01000000	180013A0	00540200	00308
					00000000'	00320
00000001	5E7000J0	000C5B24	21000000	180013A0	00506100	00324
					00000000'	0033C
00000003	46700000	000C5B24	39400000	180013A0	00506200	00340
					00000000'	00358
00000001	46700000	000C5B24	21000000	180013A0	00506300	0035C
					00000000'	00374
00000001	54700000	000C5B34	28500000	180013A0	00506400	00378
					00000000'	00390
00000001	42700000	000C5B24	3D000000	180013B0	00506500	00394
					00000000'	003AC
00000001	42700000	000C5B24	3D000000	180013B0	00506600	003B0
					00000000'	003C8
00000001	63701000	000C5B24	04000000	100013B0	00500300	003CC
					00000000'	003E4
00000002	06704000	000C5B24	79000000	180093A0	00506E00	003E8
					00000000'	00400

```

.ADDRESS VT52 BLOCK
.LONG 5259520, 402658208, 0, 809764, -
      2138050560, 0, 0, 5267456, 402658208, -
      687865856, 809764, 1450180608, 1
.ADDRESS VT100 BLOCK
.LONG 4096, 544, 0, 809764, 7340032, 0, 0, -
      4352, 544, 0, 809764, 7340032, 0, 0, -
      4608, 544, 0, 809764, 7340032, 0, 0, -
      4864, 544, 0, 809764, 7340032, 0, 0, -
      5120, 544, 0, 809764, 7340032, 0, 0, -
      5376, 544, 0, 809764, 7340032, 0, 0, -
      5632, 544, 0, 809764, 7340032, 0, 0, -
      5888, 544, 0, 809764, 7340032, 0, 0, 0, -
      544, 0, 809764, 2138050560, 0, 0, -
      8659456, 1107821216, 0, 809764, -
      2138050560, 1
.ADDRESS LA34 BLOCK
.LONG 8659772, 1107821216, 0, 809764, -
      2138050560, 0, 0, 5252352, 1107821472, 0, -
      809764, 2138050560, 1
.ADDRESS LA100 BLOCK
.LONG 5252096, 1107821472, 0, 809764, -
      2138050560, 1
.ADDRESS LA12 BLOCK
.LONG 8660224, 1107821472, 0, 809764, -
      2138050560, 0, 0, 5243904, 386400928, 0, -
      809764, 2138050560, 0, 0, 8660480, -
      1107821472, 0, 809764, 2138050560, 1
.ADDRESS LQP02 BLOCK
.LONG 5505536, 402658208, 16777216, 809764, -
      2121273344, 1
.ADDRESS VK100 BLOCK
.LONG 5267712, 402658208, 553648128, 809764, -
      1584398336, 1
.ADDRESS VT101 BLOCK
.LONG 5267968, 402658208, 960495616, 809764, -
      1181745152, 3
.ADDRESS VT102 BLOCK
.LONG 5268224, 402658208, 553648128, 809764, -
      1181745152, 1
.ADDRESS VT105 BLOCK
.LONG 5268480, 402658208, 726663168, 809780, -
      1416626176, 1
.ADDRESS VT125 BLOCK
.LONG 5268736, 402658224, 1023410176, 809764, -
      1114636288, 1
.ADDRESS VT131 BLOCK
.LONG 5268992, 402658224, 1023410176, 809764, -
      1114636288, 1
.ADDRESS VT132 BLOCK
.LONG 5243648, 268440496, 67108864, 809764, -
      1668288512, 1
.ADDRESS VT173 BLOCK
.LONG 5271040, 402690976, 2030043136, 809764, -
      108019712, 2
.ADDRESS VT200_SERIES_BLOCK

```

00000005	44704000	000C5B24	39400000	180013A0	00506F00	00404	.LONG	5271296, 402658208, 960495616, 809764, -	:
							.ADDRESS	PRO SERIES BLOCK	:
00000000	00700000	000C5B24	00000000	00000220	00000000	0041C	.LONG	0, 544, 0, 809764, 7340032, 0, 0, 0, 544, -	:
00700000	000C5B24	00000000	00000220	00000000	00000000	00438		0, 809764, 7340032, 0, 0, 0, 544, 0, -	:
000C5B24	00000000	00000220	00000000	00000000	00000000	00450		809764, 7340032, 0, 0, 0, 544, 0, 809764, -	:
00000000	00000220	00000000	00000000	00000000	00700000	00468		7340032, 0, 0, 0, 544, 0, 809764, -	:
00000220	00000000	00000000	00000000	00700000	000C5B24	00480		7340032, 0, 0, 0, 544, 0, 809764, -	:
00000000	00000000	00000000	00700000	000C5B24	00000000	00498		7340032, 0, 0, 0, 544, 0, 809764, -	:
00000000	00000000	00700000	000C5B24	00000000	00000220	00480		7340032, 0, 0, 0, 544, 0, 809764, -	:
00000000	00700000	000C5B24	00000000	00000220	00000000	004C8		7340032, 0, 0, 0, 544, 0, 809764, -	:
00700000	000C5B24	00000000	00000220	00000000	00000000	004E0		7340032, 0, 0	:
						004F8			:
00000000	00000000	00000000	00000000	00000000	00000000	00500	TERMS_TTSET KEY::		:
							.ADDRESS	P.AFW, P.AFY, P.AGA, P.AGC, P.AGE, -	:
00000000	00000000	00000000	00000000	00000000	00000000	00518		P.AGG, P.AGI, P.AGK, P.AGM, P.AGO, P.AGQ, -	:
						00530		P.AGS, P.AGU, P.AGW, P.AGY	:
00000010	00004000	00100000	00080000	00000008	00008000	0053C	TERMS_TTSET BIT::		:
							.LONG	32768, 8, 524288, 1048576, 16384, 16, -	:
00000040	00001000	00040000	00000001	00200000	00000080	00554		128, 2097152, 1, 262144, 4096, 64, 256, -	:
						0056C		32, 512	:
00000000	00000000	00000000	00000000	00000000	00000000	00578	TERMS_TTCLR KEY::		:
							.ADDRESS	P.AHA, P.AHC, P.AHE, P.AHG, P.AHI, -	:
						00590		P.AHK, P.AHM	:
00000002	00020000	00100000	00000001	00001000	00000080	00594	TERMS_TTCLR BIT::		:
							.LONG	128, 4096, 1, 1048576, 131072, 2, 4	:
						005AC			:
00000000	00000000	00000000	00000000	00000000	00000000	005B0	TERMS_TT2SET KEY::		:
							.ADDRESS	P.AHO, P.AHQ, P.AHS, P.AHU, P.AHW, -	:
00000000	00000000	00000000	00000000	00000000	00000000	005C8		P.AHY, P.AIA, P.AIC, P.AIE, P.AIG, P.AII, -	:
00000000	00000000	00000000	00000000	00000000	00000000	005E0		P.AIK, P.AIM, P.AIO, P.AIQ, P.AIS, P.AIU, -	:
00000000	00000000	00000000	00000000	00000000	00000000	005F8		P.AIW, P.AIY, P.AJA, P.AJC, P.AJE, P.AJG, -	:
								P.AJI	:
04000000	00000002	00800000	01000000	00000080	08000000	00610	TERMS_TT2SET BIT::		:
							.LONG	134217728, 128, 16777216, 8388608, 2, -	:
00000004	00004000	10000000	00008000	00020000	00000010	00628		67108864, 16, 131072, 32768, 268435456, -	:
02000000	00040000	00400000	00000001	00002000	00001000	00640		16384, 4, 4096, 8192, 1, 4194304, 262144, -	:
00000040	00000020	00080000	00200000	00100000	00010000	00658		33554432, 65536, 1048576, 2097152, -	:
								524288, 32, 64	:
						00670	TERMS_TT2CLR KEY::		:
							.ADDRESS	P.AJK, P.AJM, P.AJO	:
						0067C	TERMS_TT2CLR BIT::		:
							.LONG	8388608, 8192, 256	:
						00688	TERMS_REQBLK::		:
							.ADDRESS	P.AJQ, P.AJS, P.AJU	:
00000000	00000000	00000000	00000000	00000000	00000000	00694	TERMS_SPDBLK::		:
							.ADDRESS	P.AJW, P.AJY, P.AKA, P.AKC, P.AKE, -	:
00000000	00000000	00000000	00000000	00000000	00000000	006AC		P.AKG, P.AKI, P.AKK, P.AKM, P.AKO, P.AKQ, -	:
						006C4		P.AKS, P.AKU, P.AKW, P.AKY, P.ALA, P.ALC	:
						006D8	TERMS_EVEN::		:
							.LONG	4	:
						006DC	.ADDRESS	P.ALE	:
						006E0	TERMS_ODD::		:
							.LONG	3	:
						006E4	.ADDRESS	P.ALF	:
						006E8	TERMS_NONE::		:
							.LONG	4	:

```
00000000' 006EC .ADDRESS P.ALG  
00000004 006F0 PROTOCOL$ NONE::  
          .CONG 4  
00000000' 006F4 .ADDRESS P.ALH  
00000005 006F8 PROTOCOL$ DDCMP::  
          .CONG 5  
00000000' 006FC .ADDRESS P.ALI
```

;  
:  
:  
:  
:

```
TERMS_LA36== 0  
TERMS_LA120== 1  
TERMS_VT05== 2  
TERMS_VT52== 3  
TERMS_VT55== 4  
TERMS_VT100== 5  
TERMS_FT1== 6  
TERMS_FT2== 7  
TERMS_FT3== 8  
TERMS_FT4== 9  
TERMS_FT5== 10  
TERMS_FT6== 11  
TERMS_FT7== 12  
TERMS_FT8== 13  
TERMS_UNKNOWN== 14  
TERMS_LA34== 15  
TERMS_LA38== 16  
TERMS_LA100== 17  
TERMS_LA12== 18  
TERMS_LA24== 19  
TERMS_TQ_BTS== 20  
TERMS_LQP02== 21  
TERMS_VK100== 22  
TERMS_VT101== 23  
TERMS_VT102== 24  
TERMS_VT105== 25  
TERMS_VT125== 26  
TERMS_VT131== 27  
TERMS_VT132== 28  
TERMS_VT173== 29  
TERMS_VT200_SERIES== 30  
TERMS_PRO_SERIES== 31  
TERMS_DUMMY1== 32  
TERMS_DUMMY2== 33  
TERMS_DUMMY3== 34  
TERMS_DUMMY4== 35  
TERMS_DUMMY5== 36  
TERMS_DUMMY6== 37  
TERMS_DUMMY7== 38  
TERMS_DUMMY8== 39  
TERMS_NUM== 40  
TERMS_TTSET_NUM== 15  
TERMS_TTCLR_NUM== 7  
TERMS_IT2SET_NUM== 24  
TERMS_IT2CLR_NUM== 3  
TERMS_REQNUM== 3  
TERMS_SPDNUM== 17
```

PSECT SUMMARY

Name	Bytes	Attributes
SPLITS	2184	NOVEC,NOWRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$OWNS	144	NOVEC, WRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$GLOBALS	1792	NOVEC, WRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
. ABS .	0	NOVEC,NOWRT,NORD ,NOEXE,NOSHR, LCL, ABS, CON,NOPIC,ALIGN(0)

Library Statistics

File	----- Symbols -----		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	78 0	1000	00:01.8

COMMAND QUALIFIERS

:  
: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:TERMDEFS/OBJ=OBJ\$:TERMDEFS MSRC\$:TERMDEFS/UPDATE=(ENHS:TERMDEFS)  
: Size: 0 code + 4120 data bytes  
: Run Time: 00:46.0  
: Elapsed Time: 02:13.2  
: Lines/CPU Min: 897  
: Lexemes/CPU-Min: 61351  
: Memory Used: 212 pages  
: Compilation Complete

0059 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

This image displays a grid of 144 small terminal window screenshots, arranged in 12 rows and 12 columns. Each window shows a different set of system commands and their outputs, typical of a VAX/VMS environment. The text is rendered in a monospaced font, characteristic of early computer terminals. Some of the more prominent and legible text within the windows includes:

- TERMDEFS LIS**: A list of terminal definitions.
- SUBMITMSG LIS**: A list of submitted messages.
- TRANQUELE LIS**: A list of processes or jobs.
- SUBMIT LIS**: A list of submitted jobs.
- SWITCHRM LIS**: A list of switch routines.

The screenshots show various stages of system operation, including command prompts, error messages, and lists of system resources. The overall appearance is that of a comprehensive test or demonstration of the VAX/VMS operating system's capabilities.