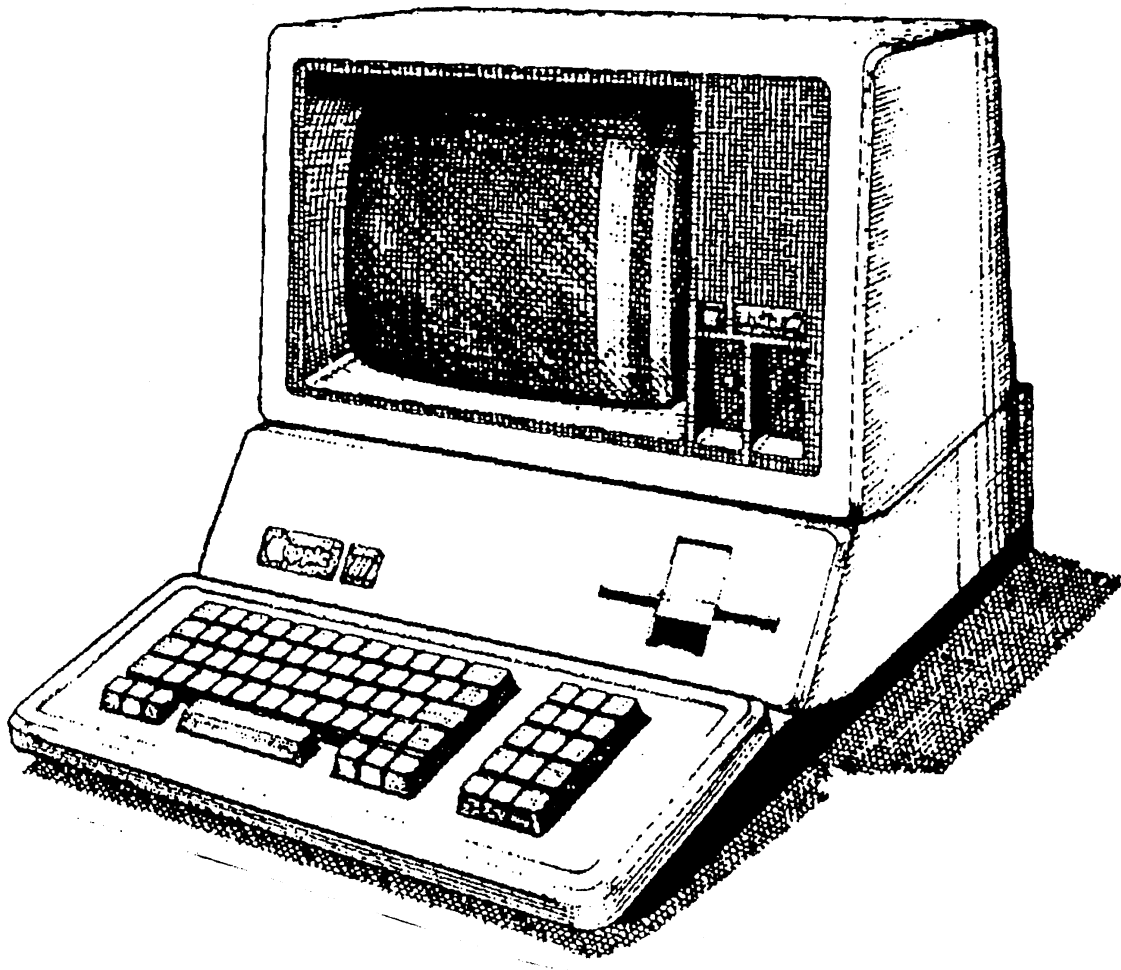




# Apple /// Computer Information



DOCUMENT NAME	#
EXERSOS SOURCE CODE LISTING 1.0	162
1982	

**Ex Libris David T. Craig**

DAVID T. CRAIG

```

1 Apple /// EXERSOS source code
2 Written by Apple Computer in Business BASIC (1981/1982)
3
4
5 1000 REM -- ExerSOS x1.0 - Jeff/Don/Bob.
6 1010 DIM n$(41),ref(40),path$(40),lev(40),buffer$(256),qw$(41)
7 1020 n$(1)="Create":n$(2)="Destroy":n$(3)="Rename":n$(4)="Set_File_Info"
8 1030 n$(5)="Get_File_Info":n$(6)="Volume":n$(7)="Set Prefix"
9 1040 n$(8)="Get Prefix":n$(9)="Open":n$(10)="New Line":n$(11)="Read"
10 1050 n$(12)="Write":n$(13)="Close":n$(14)="Flush":n$(15)="Set Mark"
11 1060 n$(16)="Get Mark":n$(17)="Set_EOF":n$(18)="Get_EOF":n$(19)="Set_Level"
12 1070 n$(20)="Get_Level":n$(23)="D_Status"
13 1080 n$(24)="D_Control":n$(25)="Get_Dev_Num":n$(26)="D_Info"
14 1090 n$(27)="Request_Seg":n$(28)="Find_Seg":n$(29)="Change_Seg"
15 1100 n$(30)="Get_Seg_Info":n$(31)="Get_Seg_Num":n$(32)="Release_Seg"
16 1110 n$(33)="Set_Fence":n$(34)="Get_Fence":n$(35)="Set Time"
17 1120 n$(36)="Get Time":n$(37)="Get_Analog":n$(38)="Display_Directory"
18 1130 n$(39)="Display_Open_File_Table":n$(40)="Display_Active_Memory_Segments"
19 1140 n$(41)="Display/Edit_Contents_of_Data_Buffer"
20 1150 qw$(1)="C0":qw$(2)="C1":qw$(3)="C2":qw$(4)="C3":qw$(5)="C4"
21 1160 qw$(6)="C5":qw$(7)="C6":qw$(8)="C7":qw$(9)="C8":qw$(10)="C9"
22 1170 qw$(11)="CA":qw$(12)="CB":qw$(13)="CC":qw$(14)="CD":qw$(15)="CE"
23 1180 qw$(16)="CF":qw$(17)="D0":qw$(18)="D1":qw$(19)="D2":qw$(20)="D3"
24 1190 qw$(23)="82":qw$(24)="83":qw$(25)="84":qw$(26)="85":qw$(27)="40"
25 1200 qw$(28)="41":qw$(29)="42":qw$(30)="43":qw$(31)="44":qw$(32)="45"
26 1210 qw$(33)="60":qw$(34)="61":qw$(35)="62":qw$(36)="63":qw$(37)="64"
27 1220 qw$(38)="1":qw$(39)="2":qw$(40)="3":qw$(41)="4"
28 1230 videoff$=CHR$(14):videon$=CHR$(15):clrlines$=CHR$(30):bell$=CHR$(7)
29 1240 uparrow=11:downarrow=10:leftarrow=8:rightarrow=21:escape=27:retrn=13
30 1250 arrows$=CHR$(uparrow)+CHR$(downarrow)+CHR$(leftarrow)+CHR$(rightarrow)
31 1260 TEXT:HOME:PRINT videoff$
32 1270 WINDOW 1,1 TO 80,5:INVERSE:HOME:NORMAL
33 1280 WINDOW 3,2 TO 78,4:HOME:VPOS=2:HPOS=30
34 1290 PRINT"** ExerSOS 1.0 **";videon$
35 1300 ON ERR GOTO 1350
36 1310 INVOKE"tool.code","screen.code"
37 1320 OFF ERR
38 1330 PERFORM findbuffer
39 1340 PERFORM geterror(@error%):IF error%=0 THEN 1370
40 1350 INVERSE:HOME:PRINT:HPOS=20:PRINT"** CANNOT ALLOCATE I/O BUFFER **"
41 1360 TEXT:VPOS=20:END
42 1370 REM -- paint the main menu --
43 1380 PRINT videoff$
44 1390 WINDOW 1,1 TO 80,24:INVERSE:HOME
45 1400 HPOS=4:PRINT"File Calls":HPOS=25:PRINT"Device Calls";
46 1410 HPOS=44:PRINT"Memory Calls":HPOS=64:PRINT"Utility Calls"
47 1420 HPOS=43:VPOS=10:PRINT"ExerSOS Utilities"
48 1430 VPOS=24:HPOS=35:PRINT"(c) 1982 by Apple Computer";
49 1440 NORMAL:WINDOW 3,2 TO 21,23:HOME
50 1450 FOR A=1 TO 20
51 1460 HPOS=2:VPOS=a+1:PRINT qw$(A);" ";N$(A);:NEXT
52 1470 WINDOW 24,2 TO 40,9:HOME
53 1480 FOR A=23 TO 26
54 1490 VPOS=a-21:HPOS=2:PRINT qw$(A);" ";N$(A):NEXT
55 1500 WINDOW 43,2 TO 60,9:HOME
56 1510 FOR A=27 TO 32
57 1520 VPOS=a-23:HPOS=2:PRINT qw$(A);" ";N$(A):NEXT
58 1530 WINDOW 63,2 TO 78,9:HOME
59 1540 FOR A=33 TO 37
60 1550 VPOS=a-31:HPOS=2:PRINT qw$(A);" ";N$(A):NEXT
61 1560 WINDOW 24,11 TO 78,19:HOME:PRINT
62 1570 FOR A=38 TO 41
63 1580 HPOS=10:PRINT qw$(A);" ";N$(A):NEXT
64 1590 WINDOW 24,21 TO 78,23:HOME
65 1600 WINDOW 1,1 TO 80,24:PERFORM savescreen(%0)
66 1610 REM -- paint the secondary menu --
67 1620 HOME
68 1630 INVERSE:WINDOW 1,1 TO 40,21:HOME:PRINT:HPOS=12:PRINT"Input Parameters"
69 1640 NORMAL:WINDOW 3,3 TO 38,20:HOME
70 1650 INVERSE:WINDOW 39,1 TO 80,21:HOME:PRINT:HPOS=12:PRINT"Output Parameters"
71 1660 NORMAL:WINDOW 41,3 TO 78,20:HOME
72 1670 WINDOW 1,1 TO 80,24
73 1680 PERFORM savescreen(%1)
74 1690 REM -- paint the utilities screen --
75 1700 NORMAL:HOME
76 1710 WINDOW 1,1 TO 80,21:INVERSE:HOME
77 1720 NORMAL:WINDOW 3,2 TO 78,20:HOME
    
```

*see sos manual preface for full names*

*6502 external modules*

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 2

```

78 1730 WINDOW 1,1 TO 80,24
79 1740 PERFORM savescreen(%2):NORMAL:HOME:PRINT videon$
80 1750 REM -- get selection --
81 1760 TEXT:HOME:PERFORM restorescreen(%0)
82 1770 WINDOW 35,21 TO 78,23:HOME
83 1780 y=2:desc$="Enter selection (0 to EXIT) $"
84 1790 ON ERR GOTO 1770
85 1800 GOSUB 6920
86 1810 OFF ERR
87 1820 selection=-1:IF num=0 THEN 1870
88 1830 FOR i=1 TO 41:IF TEN(qw$(i))=num THEN selection=i
89 1840 NEXT
90 1850 IF selection<0 OR selection>41 THEN 1770
91 1860 IF selection>0 THEN 1900
92 1870 TEXT:HOME:VPOS=12:HPOS=25:INPUT"Okay to Exit ExerSOS? ";a$
93 1880 IF a$<>"Y" AND a$<>"y" THEN 1750
94 1890 PERFORM relbuffer:HOME:NEW:END
95 1900 REM -- dispatch the calls --
96 1910 TEXT:HOME
97 1920 IF selection<38 THEN PERFORM restorescreen(%1):ELSE PERFORM restorescreen(%2)
98 1930 INVERSE:HPOS=(80-LEN(n$(selection)))/2:PRINT n$(selection);:NORMAL
99 1940 WINDOW 4,4 TO 37,20:VPOS=1
100 1950 ON selection GOTO 1980,2160,2210,2270,2540,2810,2890,2940,3010,
101 3150,3250,3340,3420,3530,3590,3690,3760,3860,
102 3930,3980,1770,1770,4040,4240,4400,4460,4690,
103 4840,5040,5150,5270
104 1960 ON selection-31 GOTO 5360,5420,5480,5540,5730,5860,5960,5960,
105 5960,5960
106 1970 GOTO 1750
107 1980 REM -- CREATE call --
108 1990 y=1:desc$="pathname":GOSUB 6920:pathname$=string$
109 2000 y=2:min&=0:max&=8:desc$="length"
110 2010 GOSUB 6920:length%=num
111 2020 IF length%=0 THEN 2130
112 2030 y=3:desc$="file_type$":min&=0:max&=255
113 2040 GOSUB 6920:fileid%=num
114 2050 IF length%<3 THEN 2130
115 2060 y=4:desc$="aux_type":min&=0:max&=65535
116 2070 GOSUB 6920:auxid%=num
117 2080 IF length%<4 THEN 2130
118 2090 y=5:desc$="storage_type [1 or $D]":GOSUB 6920:type%=num
119 2100 IF type%<>1 AND type%<>13 THEN 2090
120 2110 IF length%<8 THEN 2130
121 2120 y=6:desc$="EOF [0 ... $0FFFFFFF]":GOSUB 6920:eofile%=num&
122 2130 PERFORM soscreate(@pathname$,length%,fileid%,
123 %auxid%,type%,eofile%)
124 2140 GOSUB 7250
125 2150 GOTO 7320
126 2160 REM -- DESTROY call --
127 2170 y=1:desc$="pathname":GOSUB 6920:pathname$=string$
128 2180 PERFORM destroy(@pathname$)
129 2190 GOSUB 7250
130 2200 GOTO 7320
131 2210 REM -- RENAME call --
132 2220 y=1:desc$="pathname":GOSUB 6920:pathname$=string$
133 2230 y=2:desc$="new_pathname":GOSUB 6920:newpath$=string$
134 2240 PERFORM sosrename(@pathname$,@newpath$)
135 2250 GOSUB 7250
136 2260 GOTO 7320
137 2270 REM -- Set FILE_INFO call --
138 2280 y=1:desc$="pathname":GOSUB 6920:pathname$=string$
139 2290 y=2:min&=0:max&=15:desc$="length"
140 2300 GOSUB 6920:length%=num
141 2310 IF length%=0 THEN 2510
142 2320 y=3:min&=0:max&=255:desc$="access"
143 2330 GOSUB 6920:attrib%=num
144 2340 IF length%<2 THEN 2510
145 2350 y=4:min&=0:max&=255:desc$="file_type"
146 2360 GOSUB 6920:fileid%=num
147 2370 IF length%<4 THEN 2510
148 2380 y=5:min&=0:max&=65535:desc$="aux_type"
149 2390 GOSUB 6920:auxid%=num
150 2400 IF length%<15 THEN 2510
151 2410 y=6:min&=0:max&=99:desc$="year"
152 2420 GOSUB 6920:year%=num
153 2430 y=7:min&=0:max&=12:desc$="month"
154 2440 GOSUB 6920:month%=num

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 3

```

155 2450 y=8:min&=0:max&=31:desc$="date"
156 2460 GOSUB 6920:date%=num
157 2470 y=9:min&=0:max&=24:desc$="hour"
158 2480 GOSUB 6920:hour%=num
159 2490 y=10:min&=0:max&=60:desc$="minute"
160 2500 GOSUB 6920:minute%=num
161 2510 PERFORM setfileinfo (@pathname$, %length%, %attrib%, %fileid%,
162 %auxid%, %year%, %month%, %date%, %hour%,
163 %minute%)
164 2520 GOSUB 7250
165 2530 GOTO 7320
166 2540 REM -- Get_FILE_INFO call --
167 2550 y=1:desc$="pathname":GOSUB 6920:pathname%=string$
168 2560 y=2:min&=0:max&=15:desc$="length"
169 2570 GOSUB 6920:length%=num
170 2580 PERFORM getfileinfo (@pathname$, %length%, @attrib%, @fileid%,
171 @auxid%, @type%, @eofile%, @blocks%, @year%,
172 @month%, @date%, @hour%, @minute%)
173 2590 GOSUB 7250
174 2600 IF length%=0 THEN 2800
175 2610 desc$="access":num=attrib%:GOSUB 7160
176 2620 IF length%<2 THEN 2800
177 2630 desc$="file_type":num=fileid%:GOSUB 7160
178 2640 IF length%<4 THEN 2800
179 2650 desc$="aux_type":IF type%=15 THEN desc$="num_blocks"
180 2660 num=auxid%:GOSUB 7160
181 2670 IF length%<5 THEN 2800
182 2680 desc$="storage_type":num=type%:GOSUB 7160
183 2690 IF length%<9 THEN 2800
184 2700 desc$="EOF":num=&eofile%&:num=CONV(eofile%):GOSUB 7160
185 2710 IF length%<11 THEN 2800
186 2720 desc$="blocks used":num=blocks%:GOSUB 7160
187 2730 IF length%<15 THEN 2800
188 2740 PRINT"last_mod"
189 2750 desc$=" year":num=year%:GOSUB 7160
190 2760 desc$=" month":num=month%:GOSUB 7160
191 2770 desc$=" date":num=date%:GOSUB 7160
192 2780 desc$=" hour":num=hour%:GOSUB 7160
193 2790 desc$=" minute":num=minute%:GOSUB 7160
194 2800 GOTO 7320
195 2810 REM -- VOLUME call --
196 2820 y=1:desc$="dev_name":GOSUB 6920:devname%=string$
197 2830 PERFORM volume (@devname$, @volname$, @totblks%, @freeblks%)
198 2840 GOSUB 7250
199 2850 PRINT"vol_name = ";volname$
200 2860 desc$="total_blocks":num=totblks%:GOSUB 7160
201 2870 desc$="free_blocks":num=freeblks%:GOSUB 7160
202 2880 GOTO 7320
203 2890 REM -- Set_PREFIX call --
204 2900 y=1:desc$="pathname":GOSUB 6920:pathname%=string$
205 2910 PERFORM setprefix (@pathname%)
206 2920 GOSUB 7250
207 2930 GOTO 7320
208 2940 REM -- Get_PREFIX call --
209 2950 y=1:min&=0:max&=255:desc$="length"
210 2960 GOSUB 6920:length%=num
211 2970 PERFORM getprefix (@pathname$, %length%)
212 2980 GOSUB 7250
213 2990 PRINT"pathname = ";pathname$
214 3000 GOTO 7320
215 3010 REM -- OPEN call --
216 3020 y=1:desc$="pathname":GOSUB 6920:pathname%=string$
217 3030 y=2:min&=0:max&=1:desc$="length"
218 3040 GOSUB 6920:length%=num
219 3050 IF length%=0 THEN 3080
220 3060 y=3:min&=0:max&=3:desc$="req_access"
221 3070 GOSUB 6920:access%=num
222 3080 PERFORM sosopen (@pathname$, %length%, %access%, @refnum%)
223 3090 GOSUB 7250
224 3100 desc$="ref_num":num=refnum%:GOSUB 7160
225 3110 index=refnum%:IF index>16 THEN index=index-111
226 3120 PERFORM getlevel (@level%)
227 3130 ref(index)=refnum%:path$(index)=pathname%:lev(index)=level%
228 3140 GOTO 7320
229 3150 REM -- NEW_LINE call --
230 3160 y=1:min&=0:max&=255:desc$="ref_num"
231 3170 GOSUB 6920:refnum%=num

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 4

```

232 3180 y=2:min&=0:max&=255:desc$="is_newline"
233 3190 GOSUB 6920:isnew%=num
234 3200 y=3:min&=0:max&=255:desc$="newline_char"
235 3210 GOSUB 6920:newchar%=num
236 3220 PERFORM newline(%refnum%,%isnew%,%newchar%)
237 3230 GOSUB 7250
238 3240 GOTO 7320
239 3250 REM -- READ call --
240 3260 y=1:min&=0:max&=255:desc$="ref_num"
241 3270 GOSUB 6920:refnum%=num
242 3280 y=2:min&=0:max&=16384:desc$="request_count"
243 3290 GOSUB 6920:bytes%=num
244 3300 PERFORM sosread(%refnum%,%bytes%,@bytesread%)
245 3310 GOSUB 7250
246 3320 desc$="transfer_count":num=bytesread%:GOSUB 7160
247 3330 GOTO 7320
248 3340 REM -- WRITE call --
249 3350 y=1:min&=0:max&=255:desc$="ref_num"
250 3360 GOSUB 6920:refnum%=num
251 3370 y=2:min&=0:max&=16384:desc$="request_count"
252 3380 GOSUB 6920:bytes%=num
253 3390 PERFORM soswrite(%refnum%,%bytes%)
254 3400 GOSUB 7250
255 3410 GOTO 7320
256 3420 REM -- CLOSE call --
257 3430 y=1:min&=0:max&=255:desc$="ref_num"
258 3440 GOSUB 6920:refnum%=num
259 3450 PERFORM socclose(%refnum%)
260 3460 GOSUB 7250
261 3470 IF refnum%>16 THEN refnum%=refnum%-111
262 3480 IF refnum%<>0 THEN ref(refnum%)=0:GOTO 3520
263 3490 PERFORM getlevel(@level%)
264 3500 FOR a=1 TO 40:IF lev(a)>=level% THEN ref(a)=0
265 3510 NEXT
266 3520 GOTO 7320
267 3530 REM -- FLUSH call --
268 3540 y=1:min&=0:max&=255:desc$="ref_num"
269 3550 GOSUB 6920:refnum%=num
270 3560 PERFORM flush(%refnum%)
271 3570 GOSUB 7250
272 3580 GOTO 7320
273 3590 REM -- Set_MARK call --
274 3600 y=1:min&=0:max&=255:desc$="ref_num"
275 3610 GOSUB 6920:refnum%=num
276 3620 y=2:min&=0:max&=3:desc$="base"
277 3630 GOSUB 6920:base%=num
278 3640 y=3:desc$="displacement"
279 3650 GOSUB 6920:displacement%=num&
280 3660 PERFORM setmark(%refnum%,%base%,&displacement%)
281 3670 GOSUB 7250
282 3680 GOTO 7320
283 3690 REM -- Get_MARK call --
284 3700 y=1:min&=0:max&=255:desc$="ref_num"
285 3710 GOSUB 6920:refnum%=num
286 3720 PERFORM getmark(%refnum%,@mark%)
287 3730 GOSUB 7250
288 3740 desc$="mark":num=&mark%:num=CONV(mark%):GOSUB 7160
289 3750 GOTO 7320
290 3760 REM -- Set_EOF call --
291 3770 y=1:min&=0:max&=255:desc$="ref_num"
292 3780 GOSUB 6920:refnum%=num
293 3790 y=2:min&=0:max&=3:desc$="base"
294 3800 GOSUB 6920:base%=num
295 3810 y=3:desc$="displacement"
296 3820 GOSUB 6920:displacement%=num&
297 3830 PERFORM seteof(%refnum%,%base%,&displacement%)
298 3840 GOSUB 7250
299 3850 GOTO 7320
300 3860 REM -- Get_EOF call --
301 3870 y=1:min&=0:max&=255:desc$="ref_num"
302 3880 GOSUB 6920:refnum%=num
303 3890 PERFORM geteof(%refnum%,@eofile%)
304 3900 GOSUB 7250
305 3910 desc$="EOF":num=&eofile%$8num?COLT(eofile%):EOSU@ 7160
306 3920 GOTO 7320
307 3930 REM -- Set_LEVEL call --
308 3940 y=1:desc$="level":GOSUB 6920:level%?nuo

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 5

```

309 3970 PERDORO"setlevgl(%level%)
310 3960 GOSUB 7250
311 3970 GOTO 7320
312 3980 REM -- Get_LEVEL call --
313 3990 PRINT"[None]"
314 4000 PERFORM getlevel(@level%)
315 4010 GOSUB 7250
316 4020 desc$="level":num=level%:GOSUB 7160
317 4030 GOTO 7320
318 4040 REM -- D STATUS call --
319 4050 y=1:min&=0:max&=255:desc$="dev_num"
320 4060 GOSUB 6920:refnum%=num
321 4070 y=2:min&=0:max&=255:desc$="dstatus_call": GOSUB 4280:
322 GOSUB 6920:code%=num
323 4090 first%=255
324 4100 min&=0:max&=0:y=3:desc$="Initialize Buffer [Y or N]"
325 6310 GOSUB"4922:a$=PIGHT$(string$,1)
326 4120 IF a$="Y" OR a$="y" THEN init%=1:ELSE init%=0
327 4130 PERFORM dstatus(%devnum%,%code%,%first%,%init%,@devlist%)
328 4140 GOSUB 7250
329 4150 y=3:desc$="Amount of keyboard output":GOSUB 6920:length%=num
330 4160 IF length%>255 THEN 4150
331 4170 IF length%=0 THEN 4230
332 4180 WINDOW 42,8 TO 78,20:HOME
333 4190 FOR a=1 TO length%
334 4200 CONV%(ascii):GOSUB 7160
335 4220 NEXT
336 4230 GOTO 7320
337 4240 REO"-- F_COLTRML ccnl --
338 6050 y=1:min&=0:max&=255:desc$="dev_num"
339 4260 GOSUB 6920:refnum%=num
340 4270 y=2:min&=0:max&=255:desc$="control_code"
341 4280 GOSUB 6920:code%=num
342 4290 y=3:min&=0:max&=255:desc$="length of input"
343 4300 GOSUB 6920:length%=num
344 4310 IF length%=0 THEN 4370
345 4320 devlist$=""
346 4330 FOR a=1 TO length%:y=a+4:IF y>24 THEN y=24
347 4340 min&=0:maz$=257:dggc$="byve #"+STP$(a+
348 4350 GOSUB 6920:devlist$=devlist$+CHR$(num)
349 4360 NEXT
350 4370 PERFORM dcontrol(%devnum%,%aodg%,@dgvlitq$. 'le7250
351 4390 GOTO 7320
352 4400 REM -- Get_DEV_NUM call --
353 4410 y=1:desc$="dev_name":GOSUB 6920:devname$=string$
354 4420 PERFORM getdevnum(@devname$,@devnum%)
355 4430 GOSUB 7250
356 4440 desc$="dev_num":num=devnum%:GOSUB 7160
357 4450 GOTO 7320
358 4460 REM -- D_INFO call --
359 4470 y=1:min&=0:max&=24:desc$="dev_num"
360 4480 GOSUB 6920:devnum%=num
361 4490 y=2:min&=0:max&=11:desc$="length"
362 4500 GOSUB 6920:length%=num
363 4510 PERFORM dinfo(%devnum%,@devname%,%length%,@slotnum%,
364 @unitnum%,@devtype%,@devsub%,@blocks%,
365 @manuf%,@vernum%)
366 4520 GOSUB 7250
367 4530 PRINT"dev name = ";devname$
368 4540 IF length%<1 THEN 4680
369 4550 desc$="slot num":num=slotnum%:GOSUB 7160
370 4560 IF length%<2 THEN 4680
371 4570 desc$="unit num":num=unitnum%:GOSUB 7160
372 4580 IF length%<4 THEN 4680
373 4590 desc$="dev type":num=devtype%:GOSUB 7160
374 4600 IF length%<6 THEN 4680
375 4610 desc$="sub type":num=devsub%:GOSUB 7160
376 4620 IF length%<8 THEN 4680
377 4630 desc$="total blocks":num=blocks%:GOSUB 7160
378 4640 IF length%<10 THEN 4680
379 4650 desc$="manuf_id":num=manuf%:GOSUB 7160
380 4660 IF length%<11 THEN 4680
381 4670 desc$="version_num":num=vernum%:GOSUB 7160
382 4680 GOTO 7320
383 4690 REM -- REQUEST_SEG call --
384 4700 y=1:desc$="base bank"
385 4710 GOSUB 6920:basebk%=num

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 6

```

386 4720 y=2:desc$="base page"
387 4730 GOSUB 6920:basepg%=num
388 4740 y=3:desc$="limit bank"
389 4750 GOSUB 6920:limitbk%=num
390 4760 y=4:desc$="limit page"
391 4770 GOSUB 6920:limitpg%=num
392 4780 y=5:min&=0:max&=255:desc$="seg_id"
393 4790 GOSUB 6920:segid%=num
394 4800 PERFORM reqseg(%basebk%,%basepg%,%limitbk%,
395 %limitpg%,%segid%,@segnum%)
396 4810 GOSUB 7250
397 4820 desc$="seg_num":num=segnum%:GOSUB 7160
398 4830 GOTO 7320
399 4840 REM -- FIND_SEG call --
400 4850 y=1:min&=0:max&=255:desc$="search_mode"
401 4860 GOSUB 6920:mode%=num
402 4870 y=2:min&=0:max&=255:desc$="seg_id"
403 4880 GOSUB 6920:segid%=num
404 4890 y=3:min&=0:max&=65535:desc$="pages"
405 4900 GOSUB 6920:page%=num
406 4910 PERFORM findseg(%mode%,%segid%,@page%,@basebk%,@basepg%,
407 @limitbk%,@limitpg%,@segnum%)
408 4920 PERFORM geterror(@error%)
409 4930 IF error%<>225 THEN GOSUB 7250:GOTO 4970:
410 REM spcl case for SEGRQDN
411 4940 WINDOW 42,4 TO 78,20:HOME
412 4950 PRINT"SOS Error Code = ";:HPOS=23
413 4960 PRINT RIGHT$(HEX$(ERROR%),2):PRINT
414 4970 desc$="pages":num=page%:GOSUB 7160
415 4980 desc$="base bank":num=basebk%:GOSUB 7160
416 4990 desc$="base page":num=basepg%:GOSUB 7160
417 5000 desc$="limit bank":num=limitbk%:GOSUB 7160
418 5010 desc$="limit page":num=limitpg%:GOSUB 7160
419 5020 desc$="seg_num":num=segnum%:GOSUB 7160
420 5030 GOTO 7320
421 5040 REM -- CHANGE_SEG call --
422 5050 y=1:min&=0:max&=255:desc$="seg_num"
423 5060 GOSUB 6920:segnum%=num
424 5070 y=2:min&=0:max&=255:desc$="change_mode"
425 5080 GOSUB 6920:mode%=num
426 5090 y=3:min&=0:max&=65535:desc$="pages"
427 5100 GOSUB 6920:page%=num
428 5110 PERFORM changeseg(%segnum%,%mode%,@page%)
429 5120 GOSUB 7250
430 5130 desc$="pages":num=page%:GOSUB 7160
431 5140 GOTO 7320
432 5150 REM -- Get_SEG_INFO call --
433 5160 y=1:min&=0:max&=255:desc$="seg_num"
434 5170 GOSUB 6920:segnum%=num
435 5180 PERFORM getseginfo(%segnum%,@basebk%,@basepg%,
436 @limitbk%,@limitpg%,@page%,@segid%)
437 5190 GOSUB 7250
438 5200 desc$="base bank":num=basebk%:GOSUB 7160
439 5210 desc$="base page":num=basepg%:GOSUB 7160
440 5220 desc$="limit bank":num=limitbk%:GOSUB 7160
441 5230 desc$="limit page":num=limitpg%:GOSUB 7160
442 5240 desc$="pages":num=page%:GOSUB 7160
443 5250 desc$="seg_id":num=segid%:GOSUB 7160
444 5260 GOTO 7320
445 5270 REM -- Get_SEG_NUM call --
446 5280 y=1:desc$="seg bank"
447 5290 GOSUB 6920:segbk%=num
448 5300 y=2:desc$="seg page"
449 5310 GOSUB 6920:seggp%=num
450 5320 PERFORM getsegnum(%segbk%,%seggp%,@segnum%)
451 5330 GOSUB 7250
452 5340 desc$="seg_num":num=segnum%:GOSUB 7160
453 5350 GOTO 7320
454 5360 REM -- RELEASE_SEG call --
455 5370 y=1:min&=0:max&=255:desc$="seg_num"
456 5380 GOSUB 6920:segnum%=num
457 5390 PERFORM relseg(%segnum%)
458 5400 GOSUB 7250
459 5410 GOTO 7320
460 5420 REM -- Set_FENCE call --
461 5430 y=1:min&=0:max&=255:desc$="fence"
462 5440 GOSUB 6920:priority%=num

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 7

```

463 5450 PERFORM setfence(%priority%)
464 5460 GOSUB 7250
465 5470 GOTO 7320
466 5480 REM -- Get_FENCE call --
467 5490 PRINT"[None]"
468 5500 PERFORM getfence(@priority%)
469 5510 GOSUB 7250
470 5520 desc$="fence":num=priority%:GOSUB 7160
471 5530 GOTO 7320
472 5540 REM -- Set TIME call --
473 5550 y=1:min&=1980:max&=9999:desc$="year":GOSUB 6920
474 5560 tyme$=RIGHT$("0000"+STR$(num),4)
475 5570 y=2:min&=0:max&=12:desc$="month":GOSUB 6920
476 5580 tyme$=tyme$+RIGHT$("0000"+STR$(num),2)
477 5590 y=3:min&=0:max&=31:desc$="date":GOSUB 6920
478 5600 tyme$=tyme$+RIGHT$("0000"+STR$(num),2)
479 5610 tyme$=tyme$+STR$(000)
480 5620 y=5:min&=0:max&=23:desc$="hour":GOSUB 6920
481 5630 tyme$=tyme$+RIGHT$("0000"+STR$(num),2)
482 5640 y=6:min&=0:max&=59:desc$="minute":GOSUB 6920
483 5650 tyme$=tyme$+RIGHT$("0000"+STR$(num),2)
484 5660 y=7:min&=0:max&=59:desc$="second":GOSUB 6920
485 5670 tyme$=tyme$+RIGHT$("0000"+STR$(num),2)
486 5680 y=8:min&=0:max&=999:desc$="microsecond":GOSUB 6920
487 5690 tyme$=tyme$+RIGHT$("0000"+STR$(num),3)
488 5700 PERFORM settime(@tyme$)
489 5710 GOSUB 7250
490 5720 GOTO 7320
491 5730 REM -- Get TIME call --
492 5740 PRINT"[None]"
493 5750 PERFORM gettime(@tyme$)
494 5760 GOSUB 7250
495 5770 desc$="year":num=VAL(MID$(tyme$,1,4)):GOSUB 7160
496 5780 desc$="month":num=VAL(MID$(tyme$,5,2)):GOSUB 7160
497 5790 desc$="date":num=VAL(MID$(tyme$,7,2)):GOSUB 7160
498 5800 desc$="day of week":num=VAL(MID$(tyme$,9,1)):GOSUB 7160
499 5810 desc$="hour":num=VAL(MID$(tyme$,10,2)):GOSUB 7160
500 5820 desc$="minute":num=VAL(MID$(tyme$,12,2)):GOSUB 7160
501 5830 desc$="second":num=VAL(MID$(tyme$,14,2)):GOSUB 7160
502 5840 desc$="microsecond":num=VAL(MID$(tyme$,16)):GOSUB 7160
503 5850 GOTO 7320
504 5860 REM -- Get ANALOG call --
505 5870 y=1:min&=0:max&=255:desc$="joy_mode"
506 5880 GOSUB 6920:jmode%=num
507 5890 PERFORM joystick(%jmode%,@jval1%,@jval2%,@jval3%,@jval4%)
508 5900 GOSUB 7250
509 5910 desc$="JSn-B":num=jval1%:GOSUB 7160
510 5920 desc$="JSn-Sw":num=jval2%:GOSUB 7160
511 5930 desc$="JSn-X":num=jval3%:GOSUB 7160
512 5940 desc$="JSn-Y":num=jval4%:GOSUB 7160
513 5950 GOTO 7320
514 5960 REM -- ExerSOS utility dispatcher --
515 5970 WINDOW 5,3 TO 76,19:HOME
516 5980 ON selection-37 GOTO 5990,6060,6170,6320
517 5990 REM -- CATALOG utility --
518 6000 HPOS=25:y=1:desc$="Directory Pathname"
519 6010 GOSUB 6920:pathname$=string$:PRINT
520 6020 ON ERR OFF ERR:HPOS=25:PRINT"Device/Pathname Not Found":GOTO 7320
521 6032 KD LGL (pcthncoe$+?0 VHEN AATALOG:ELSE CATALOG pathname$
522 6040 OFF ERR
523 6050 PRINT:GOTO 7320
524 6060 REM -- Display File Vable"utklit{ --
525 6270 ""HPOQ=15:PRINT"Refnum";:HPOS=30:PRINT"Pathname";
526 6080 HPOS=55:PRINT"Level":PRINT
527 6090 a=0
528 6100 FOR refnum=1 TO 40
529 6110 IF ref(refnum)=0 THEN 6140
530 6120 HPOS=17:PRINT ref(refnum);:HPOS=30:PRINT path$(refnum);
531 6130 HPOS=57:PRINT lev(refnum):a=1
532 6140 NEXT
533 6150 IF a=0 THEN HPOS=31:PRINT"[None]"
534 6160 PRINT:GOTO 7320
535 6170 REM /- Dksplcy Acvivy Memory"Qegoents --
536 6180 HPOS=5:PRINT"Seg Num Base Address Limit Address";
537 6190 PRINT" Pages ""See ID"8PRKNT
538 6202 DOR segnum%=1 TO 30
539 6210 PERFORM getseginfo(%segnum%,@basebk%,@basepg%,@limitbk%,

```



4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 8

```

540                                     @limitpg%,@page%,@segid%)
541 6220   PERFORM geterror(@error%):IF error%<>0 THEN 6300
542 6230   HPOS=7:PRINT"$";RIGHT$(HEX$(segnum%),2);
543 6240   HPOS=19:PRINT"$";RIGHT$(HEX$(basebk%),2);
544 6250   HPOS=24:PRINT"$";RIGHT$(HEX$(basepg%),2);
545 6260   HPOS=37:PRINT"$";RIGHT$(HEX$(nimkvbk'),2)9
546 4270   HPOS=43:PRINT"$";RIGHT$(HEX$(limitpg%),2);
547 6280   HPOS=57:PRINT"$";RIGHT$(HEX$(page%),2);
548 6290   HPOS=67:RPINV"$";RIGHT$(HEX$(segid%),2)
549 6300   NEXT
550 6310   GOTO 7320
551 6320   REM -- Display/Edit Contents of IO Buffer --
552 6330   HPOS=25:y=1:desc$="Page Number":min&=0:max&=63
553 6340   GOSUB 6920:pagenum%=num:HOME
554 6350   PERFORM displaybuffer(@part1$,@part2$,%pagenum%)
555 6360   bufptr=0:n$=part1$:GOSUB 6720
556 6370   n$=part2$:GOSUB 6720
557 6380   WINDOW 1,22 TO 80,24:VPOS=0:HRMS=6
558 6190   PRKLT"<ESC>"to Exit, <arrow-keys> to move cursor,
559         <value-RETURN> to alter data"
560 6400   WINDOW 5,3 TO 76,19
561 6410   x=18y=38laqvxx=3:laqvxx?3:lasty=1:was$=""
562 6420   REM -- Editor --
563 6430   VPOS=lasty:HPOS=lastxx:PRINT" ";HPOS=lastxx+3:PRINT" ";
564 6440   HPOS=52+lastx:PRINT was$;
565 6450   xx=(x-1)*3+1:z=x+(y-1)*16
566 6460   VPOS=y:HPOS=xx:PRINT["";HPOS=xx+3:PRINT"]";
567 6470   is$=buffer$(z):IF ASC(is$)<32 THEN is$=""
568 6480   HPOS=52+x:INVERSE:PRINT is$;NORMAL
569 6490   lastx=x:lastxx=xx:lasty=y:was$=is$
570 6500   REM - get a byte -
571 6510   GET a$:a=ASC(a$)
572 6520   IF a=escape THEN HPOS=xx:PRINT" ";HPOS=xx+3:PRINT" ";:
573         HPOS=52+lastx:PRINT was$;GOTO 7320
574 6530   IF INSTR("0123456789ABCDEFabcdef",a$)<>0 THEN 6600
575 6540   IF INSTR(arrows$,a$)=0 THEN PRINT bell$;GOTO 6500
576 6550   IF a=uparrow THEN y=y-1:IF y<1 THEN y=16
577 6560   IF a=downarrow THEN y=y+1:IF y>16 THEN y=1
578 6570   IF a=leftarrow THEN x=x-1:IF x<1 THEN x=16:y=y-1:IF y<1 THEN y=16
579 6580   IF a=rightarrow THEN x=x+1:IF x>16 THEN x=1:y=y+1:IF y>16 THEN y=1
580 6590   GOTO 6420
581 6600   REM - prep for user input -
582 6610   lo$a$:xx=xx+1:old$=buffer$(z)
583 6620   old$=RIGHT$(HEX$(ASC(old$)),2)
584 6630   hi$=LEFT$(old$,1)
585 6640   REM - value input -
586 6650   HPOS=xx:PRINT hi$;lo$;
587 6660   GET a$:a=ASC(a$)
588 6670   IF a=escape THEN HPOS=xx:PRINT old$;GOTO 6420
589 6680   IF a=return THEN GOSUB 6820:GOTO 6420
590 6690   IF INSTR("0123456789ABCDEFabcdef",a$)=0 THEN PRINT bell$;GOTO 6640
591 6700   hi$=lo$:lo$a$
592 6710   GOTO 6640
593 6720   REM - display half of buffer -
594 6730   ptr=0
595 6740   FOR a=1 TO 8:x$="":HPOS=2
596 6750     FOR b=1 TO 16:ptr=ptr+1
597 6760       n=ASC(MID$(n$,ptr,1))
598 6770       bufptr=bufptr+1:buffer$(bufptr)=CHR$(n)
599 6780       PRINT RIGHT$(HEX$(n),2);" ";
600 6790       IF n>31 THEN x$=x$+CHR$(n):ELSE x$=x$+"."
601 6800       NEXT:PRINT" ";x$
602 6810     NEXT:RETURN
603 6820   REM - calc value & update -
604 6830   a=TEN(hi$+lo$):a$=CHR$(a):buffer$(z)=a$
605 6840   IF z>128 THEN z=z-128
606 6850   IF y<9 THEN n$=part1$:ELSE n$=part2$
607 6860   IF z>1 THEN l$=LEFT$(n$,z-1):ELSE l$=""
608 6870   IF z<128 THEN r$=RIGHT$(n$,128-z):ELSE r$=""
609 6880   IF y<9 THEN part1$=l$+a$+r$:ELSE part2$=l$+a$+r$
610 6890   PERFORM replacebuffer(@part1$,@part2$,%pagenum%)
611 6900   IF a<32 THEN was$="" :ELSE was$=a$
612 6910   RETURN
613 6920   REM -- input a numeric or string value --
614 6930   VPOS=y:hx=0:IF RIGHT$(desc$,1)="$" THEN hx=1:
615         desc$=LEFT$(desc$,LEN(desc$)-1)
616 6940   PRINT desc$;

```

4/11/90 9:01 AM

Source Disk:a3.EXERSOS.TEXT

Page 9

```

617 6950 IF hx=0 AND CONV(min<<>0 OR max<<>0) THEN
618 PRINT [";min; " ... ";max;"];
619 6960 IF hx=1 AND CONV(min<<>0 OR max<<>0) THEN
620 PRINT [";HEX$(CONV(min&));" ... ";HEX$(CONV(max&));"];
621 6970 IF hx THEN INPUT " - $";string$:ELSE INPUT " - ";string$
622 6980 IF string$="" THEN POP:min&=0:max&=0:GOTO 1750
623 6990 lstr=LEN(string$):IF lstr=0 THEN 6930
624 7000 IF RIGHT$(string$,1)<>" " THEN 7030
625 7010 IF lstr=1 THEN 6930
626 7020 string$=LEFT$(string$,lstr-1):GOTO 6990
627 7030 num&=CONV&(VAL(string$))
628 7040 IF NOT((LEFT$(string$,1)="$") OR (hx=1)) THEN 7100
629 7050 IF LEFT$(string$,1)="$" THEN string$=RIGHT$(string$,lstr-1)
630 7060 num&=CONV&(TEN(string$))
631 7070 IF num&<0 THEN num&=65536+num&
632 7080 IF lstr<=4 THEN 7100
633 7090 num&=num&+(CONV&(TEN(LEFT$(string$,LEN(string$)-4))*65536))
634 7100 IF num&<65536 THEN num=CONV(num&):ELSE num=0
635 7110 IF num>32767 THEN num=(num-65536)
636 7120 IF min&=0 AND max&=0 THEN RETURN
637 7130 IF num&<min& OR num&>max& THEN VPOS=y:PRINT clrline$:GOTO 6920
638 7140 min&=0:max&=0
639 7150 RETURN
640 7160 REM -- output a value with its description & hex value --
641 7170 IF num<1000000 THEN num&=CONV&(num):REM [ else num& has been loaded ]
642 7180 PRINT desc$," = ";:HPOS=17
643 7190 IF num&<0 THEN num&=num&+65536
644 7200 PRINT USING"8#";num&,:HPOS=28:PRINT"$";
645 7210 num1&=num& MOD 65536
646 7220 IF num&>65535 THEN PRINT HEX$(CONV((num&-num1&)/65536));
647 7230 PRINT HEX$(CONV(num1&))
648 7240 RETURN
649 7250 REM -- error handler --
650 7260 WINDOW 42,4 TO 78,20:HOME
651 7270 ERROR%=0
652 7280 PERFORM geterror(@error%)
653 7290 PRINT"SOS Error Code = ";:HPOS=23
654 7300 PRINT RIGHT$(HEX$(error%),2):PRINT
655 7310 IF error%=0 THEN RETURN:ELSE POP
656 7320 REM -- see if he's done. if so, back to main menu --
657 7330 WINDOW 1,22 TO 80,24:HOME
658 7340 HPOS=12:VPOS=2
659 7350 PRINT"Enter <ESC> to return to Main Menu, <Any Key> to continue"
660 7360 GET a$:HOME
661 7370 IF a$=CHR$(escape) THEN 1750
662 7380 WINDOW 42,4 TO 78,20:HOME
663 7390 TEXT:GOTO 1940
664
665 FINIS
666

```