

29-MAR-84 08:16 PAGE 98

```

07 ROMSOFTF
811 REATEST INTE
812 ;QUICK GREATEST INTEGER FUNCTION.
813 ;LEAVES INT(FAC) IN FAC.
814 ;ASSUMES FAC IS SIGNED.
815 ;INT = 8388608
816 ;IF ZERO, GOT IT.
817 ;GET NUMBER OF PLACES TO SHIFT.
818 ;PUT 377 IN WHEN SHFIR SHIFTS BYTES.
819 ;TRULY NEGATE QUANTITY IN FAC.
820 ;IF NUMBER OF PLACES .GE. 7
821 ;START SHIFTN BYTS, THEN BITS
822 ;ZERO BITS; ADDER WANTS 0
823 ;CNT IN CTR
824 ;GET SIGN BIT.
825 ;SAVE FIR SHIFTED BYTE.
826 ;SHIFT THE REST.
827 ;ZERO BITS.
828 ;FORGET IT.
829 ;CLR OVERFLOW BYTE.
830 ;MAKE FAC POSITIVE
831 ;GET COMPLEMENT OF SIGN IN CARRY.
832 ;MAKE IT REALLY ZERO.
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

```

08 ROMSOFTG
LOADING POINT INPUT ROUTINE.
29-MAR-84 08:16 PAGE 99
37 ;NUMBER INPUT IS LEFT IN FAC.
38 ;AT ENTRY TPTPTR POINTS TO THE 1ST CHARACTER IN A TEXT BUF
39 ;THE FIRST CHARACTER IS ALSO IN ACCA. FIN PACKS THE DIGITS
40 ;INTO THE FAC AS AN INTEGER AND KEEPS TRACK OF WHERE THE
41 ;DECIMAL POINT IS. THE NUMBER OF DIGITS AFTER THE DP.
42 ;SEEN. DECNT IS THE NUMBER OF DIGITS AFTER THE DP.
43 ;AT THE END DECNT AND THE EXPONENT ARE USED TO
44 ;DETERMINE HOW MANY TIMES TO MULTIPLY OR DIVIDE BY TEN
45 ;TO GET THE CORRECT NUMBER.
46 FIN:
47 LDX #0
48 LDX #011+1
49 STX DECNT,X
50 FINZLP
51 FINZLP
52 FINZLP
53 FINZLP
54 FINZLP
55 FINZLP
56 FINZLP
57 FINZLP
58 FINZLP
59 FINZLP
60 FINZLP
61 FINZLP
62 FINZLP
63 FINZLP
64 FINZLP
65 FINZLP
66 FINZLP
67 FINZLP
68 FINZLP
69 FINZLP
70 FINZLP
71 FINZLP
72 FINZLP
73 FINZLP
74 FINZLP
75 FINZLP
76 FINZLP
77 FINZLP
78 FINZLP
79 FINZLP
80 FINZLP
81 FINZLP
82 FINZLP
83 FINZLP
84 FINZLP
85 FINZLP
86 FINZLP
87 FINZLP
88 FINZLP
89 FINZLP
90 FINZLP
91 FINZLP
92 FINZLP
93 FINZLP
94 FINZLP
95 FINZLP
96 FINZLP
97 FINZLP
98 FINZLP
99 FINZLP
100 FINZLP
101 FINZLP
102 FINZLP
103 FINZLP
104 FINZLP
105 FINZLP
106 FINZLP
107 FINZLP
108 FINZLP
109 FINZLP
110 FINZLP
111 FINZLP
112 FINZLP
113 FINZLP
114 FINZLP
115 FINZLP
116 FINZLP
117 FINZLP
118 FINZLP
119 FINZLP
120 FINZLP
121 FINZLP
122 FINZLP
123 FINZLP
124 FINZLP
125 FINZLP
126 FINZLP
127 FINZLP
128 FINZLP
129 FINZLP
130 FINZLP
131 FINZLP
132 FINZLP
133 FINZLP
134 FINZLP
135 FINZLP
136 FINZLP
137 FINZLP
138 FINZLP
139 FINZLP
140 FINZLP
141 FINZLP
142 FINZLP
143 FINZLP
144 FINZLP
145 FINZLP
146 FINZLP
147 FINZLP
148 FINZLP
149 FINZLP
150 FINZLP
151 FINZLP
152 FINZLP
153 FINZLP
154 FINZLP
155 FINZLP
156 FINZLP
157 FINZLP
158 FINZLP
159 FINZLP
160 FINZLP
161 FINZLP
162 FINZLP
163 FINZLP
164 FINZLP
165 FINZLP
166 FINZLP
167 FINZLP
168 FINZLP
169 FINZLP
170 FINZLP
171 FINZLP
172 FINZLP
173 FINZLP
174 FINZLP
175 FINZLP
176 FINZLP
177 FINZLP
178 FINZLP
179 FINZLP
180 FINZLP
181 FINZLP
182 FINZLP
183 FINZLP
184 FINZLP
185 FINZLP
186 FINZLP
187 FINZLP
188 FINZLP
189 FINZLP
190 FINZLP
191 FINZLP
192 FINZLP
193 FINZLP
194 FINZLP
195 FINZLP
196 FINZLP
197 FINZLP
198 FINZLP
199 FINZLP
200 FINZLP
201 FINZLP
202 FINZLP
203 FINZLP
204 FINZLP
205 FINZLP
206 FINZLP
207 FINZLP
208 FINZLP
209 FINZLP
210 FINZLP
211 FINZLP
212 FINZLP
213 FINZLP
214 FINZLP
215 FINZLP
216 FINZLP
217 FINZLP
218 FINZLP
219 FINZLP
220 FINZLP
221 FINZLP
222 FINZLP
223 FINZLP
224 FINZLP
225 FINZLP
226 FINZLP
227 FINZLP
228 FINZLP
229 FINZLP
230 FINZLP
231 FINZLP
232 FINZLP
233 FINZLP
234 FINZLP
235 FINZLP
236 FINZLP
237 FINZLP
238 FINZLP
239 FINZLP
240 FINZLP
241 FINZLP
242 FINZLP
243 FINZLP
244 FINZLP
245 FINZLP
246 FINZLP
247 FINZLP
248 FINZLP
249 FINZLP
250 FINZLP
251 FINZLP
252 FINZLP
253 FINZLP
254 FINZLP
255 FINZLP
256 FINZLP
257 FINZLP
258 FINZLP
259 FINZLP
260 FINZLP
261 FINZLP
262 FINZLP
263 FINZLP
264 FINZLP
265 FINZLP
266 FINZLP
267 FINZLP
268 FINZLP
269 FINZLP
270 FINZLP
271 FINZLP
272 FINZLP
273 FINZLP
274 FINZLP
275 FINZLP
276 FINZLP
277 FINZLP
278 FINZLP
279 FINZLP
280 FINZLP
281 FINZLP
282 FINZLP
283 FINZLP
284 FINZLP
285 FINZLP
286 FINZLP
287 FINZLP
288 FINZLP
289 FINZLP
290 FINZLP
291 FINZLP
292 FINZLP
293 FINZLP
294 FINZLP
295 FINZLP
296 FINZLP
297 FINZLP
298 FINZLP
299 FINZLP
300 FINZLP
301 FINZLP
302 FINZLP
303 FINZLP
304 FINZLP
305 FINZLP
306 FINZLP
307 FINZLP
308 FINZLP
309 FINZLP
310 FINZLP
311 FINZLP
312 FINZLP
313 FINZLP
314 FINZLP
315 FINZLP
316 FINZLP
317 FINZLP
318 FINZLP
319 FINZLP
320 FINZLP
321 FINZLP
322 FINZLP
323 FINZLP
324 FINZLP
325 FINZLP
326 FINZLP
327 FINZLP
328 FINZLP
329 FINZLP
330 FINZLP
331 FINZLP
332 FINZLP
333 FINZLP
334 FINZLP
335 FINZLP
336 FINZLP
337 FINZLP
338 FINZLP
339 FINZLP
340 FINZLP
341 FINZLP
342 FINZLP
343 FINZLP
344 FINZLP
345 FINZLP
346 FINZLP
347 FINZLP
348 FINZLP
349 FINZLP
350 FINZLP
351 FINZLP
352 FINZLP
353 FINZLP
354 FINZLP
355 FINZLP
356 FINZLP
357 FINZLP
358 FINZLP
359 FINZLP
360 FINZLP
361 FINZLP
362 FINZLP
363 FINZLP
364 FINZLP
365 FINZLP
366 FINZLP
367 FINZLP
368 FINZLP
369 FINZLP
370 FINZLP
371 FINZLP
372 FINZLP
373 FINZLP
374 FINZLP
375 FINZLP
376 FINZLP
377 FINZLP
378 FINZLP
379 FINZLP
380 FINZLP
381 FINZLP
382 FINZLP
383 FINZLP
384 FINZLP
385 FINZLP
386 FINZLP
387 FINZLP
388 FINZLP
389 FINZLP
390 FINZLP
391 FINZLP
392 FINZLP
393 FINZLP
394 FINZLP
395 FINZLP
396 FINZLP
397 FINZLP
398 FINZLP
399 FINZLP
400 FINZLP
401 FINZLP
402 FINZLP
403 FINZLP
404 FINZLP
405 FINZLP
406 FINZLP
407 FINZLP
408 FINZLP
409 FINZLP
410 FINZLP
411 FINZLP
412 FINZLP
413 FINZLP
414 FINZLP
415 FINZLP
416 FINZLP
417 FINZLP
418 FINZLP
419 FINZLP
420 FINZLP
421 FINZLP
422 FINZLP
423 FINZLP
424 FINZLP
425 FINZLP
426 FINZLP
427 FINZLP
428 FINZLP
429 FINZLP
430 FINZLP
431 FINZLP
432 FINZLP
433 FINZLP
434 FINZLP
435 FINZLP
436 FINZLP
437 FINZLP
438 FINZLP
439 FINZLP
440 FINZLP
441 FINZLP
442 FINZLP
443 FINZLP
444 FINZLP
445 FINZLP
446 FINZLP
447 FINZLP
448 FINZLP
449 FINZLP
450 FINZLP
451 FINZLP
452 FINZLP
453 FINZLP
454 FINZLP
455 FINZLP
456 FINZLP
457 FINZLP
458 FINZLP
459 FINZLP
460 FINZLP
461 FINZLP
462 FINZLP
463 FINZLP
464 FINZLP
465 FINZLP
466 FINZLP
467 FINZLP
468 FINZLP
469 FINZLP
470 FINZLP
471 FINZLP
472 FINZLP
473 FINZLP
474 FINZLP
475 FINZLP
476 FINZLP
477 FINZLP
478 FINZLP
479 FINZLP
480 FINZLP
481 FINZLP
482 FINZLP
483 FINZLP
484 FINZLP
485 FINZLP
486 FINZLP
487 FINZLP
488 FINZLP
489 FINZLP
490 FINZLP
491 FINZLP
492 FINZLP
493 FINZLP
494 FINZLP
495 FINZLP
496 FINZLP
497 FINZLP
498 FINZLP
499 FINZLP
500 FINZLP
501 FINZLP
502 FINZLP
503 FINZLP
504 FINZLP
505 FINZLP
506 FINZLP
507 FINZLP
508 FINZLP
509 FINZLP
510 FINZLP
511 FINZLP
512 FINZLP
513 FINZLP
514 FINZLP
515 FINZLP
516 FINZLP
517 FINZLP
518 FINZLP
519 FINZLP
520 FINZLP
521 FINZLP
522 FINZLP
523 FINZLP
524 FINZLP
525 FINZLP
526 FINZLP
527 FINZLP
528 FINZLP
529 FINZLP
530 FINZLP
531 FINZLP
532 FINZLP
533 FINZLP
534 FINZLP
535 FINZLP
536 FINZLP
537 FINZLP
538 FINZLP
539 FINZLP
540 FINZLP
541 FINZLP
542 FINZLP
543 FINZLP
544 FINZLP
545 FINZLP
546 FINZLP
547 FINZLP
548 FINZLP
549 FINZLP
550 FINZLP
551 FINZLP
552 FINZLP
553 FINZLP
554 FINZLP
555 FINZLP
556 FINZLP
557 FINZLP
558 FINZLP
559 FINZLP
560 FINZLP
561 FINZLP
562 FINZLP
563 FINZLP
564 FINZLP
565 FINZLP
566 FINZLP
567 FINZLP
568 FINZLP
569 FINZLP
570 FINZLP
571 FINZLP
572 FINZLP
573 FINZLP
574 FINZLP
575 FINZLP
576 FINZLP
577 FINZLP
578 FINZLP
579 FINZLP
580 FINZLP
581 FINZLP
582 FINZLP
583 FINZLP
584 FINZLP
585 FINZLP
586 FINZLP
587 FINZLP
588 FINZLP
589 FINZLP
590 FINZLP
591 FINZLP
592 FINZLP
593 FINZLP
594 FINZLP
595 FINZLP
596 FINZLP
597 FINZLP
598 FINZLP
599 FINZLP
600 FINZLP
601 FINZLP
602 FINZLP
603 FINZLP
604 FINZLP
605 FINZLP
606 FINZLP
607 FINZLP
608 FINZLP
609 FINZLP
610 FINZLP
611 FINZLP
612 FINZLP
613 FINZLP
614 FINZLP
615 FINZLP
616 FINZLP
617 FINZLP
618 FINZLP
619 FINZLP
620 FINZLP
621 FINZLP
622 FINZLP
623 FINZLP
624 FINZLP
625 FINZLP
626 FINZLP
627 FINZLP
628 FINZLP
629 FINZLP
630 FINZLP
631 FINZLP
632 FINZLP
633 FINZLP
634 FINZLP
635 FINZLP
636 FINZLP
637 FINZLP
638 FINZLP
639 FINZLP
640 FINZLP
641 FINZLP
642 FINZLP
643 FINZLP
644 FINZLP
645 FINZLP
646 FINZLP
647 FINZLP
648 FINZLP
649 FINZLP
650 FINZLP
651 FINZLP
652 FINZLP
653 FINZLP
654 FINZLP
655 FINZLP
656 FINZLP
657 FINZLP
658 FINZLP
659 FINZLP
660 FINZLP
661 FINZLP
662 FINZLP
663 FINZLP
664 FINZLP
665 FINZLP
666 FINZLP
667 FINZLP
668 FINZLP
669 FINZLP
670 FINZLP
671 FINZLP
672 FINZLP
673 FINZLP
674 FINZLP
675 FINZLP
676 FINZLP
677 FINZLP
678 FINZLP
679 FINZLP
680 FINZLP
681 FINZLP
682 FINZLP
683 FINZLP
684 FINZLP
685 FINZLP
686 FINZLP
687 FINZLP
688 FINZLP
689 FINZLP
690 FINZLP
691 FINZLP
692 FINZLP
693 FINZLP
694 FINZLP
695 FINZLP
696 FINZLP
697 FINZLP
698 FINZLP
699 FINZLP
700 FINZLP
701 FINZLP
702 FINZLP
703 FINZLP
704 FINZLP
705 FINZLP
706 FINZLP
707 FINZLP
708 FINZLP
709 FINZLP
710 FINZLP
711 FINZLP
712 FINZLP
713 FINZLP
714 FINZLP
715 FINZLP
716 FINZLP
717 FINZLP
718 FINZLP
719 FINZLP
720 FINZLP
721 FINZLP
722 FINZLP
723 FINZLP
724 FINZLP
725 FINZLP
726 FINZLP
727 FINZLP
728 FINZLP
729 FINZLP
730 FINZLP
731 FINZLP
732 FINZLP
733 FINZLP
734 FINZLP
735 FINZLP
736 FINZLP
737 FINZLP
738 FINZLP
739 FINZLP
740 FINZLP
741 FINZLP
742 FINZLP
743 FINZLP
744 FINZLP
745 FINZLP
746 FINZLP
747 FINZLP
748 FINZLP
749 FINZLP
750 FINZLP
751 FINZLP
752 FINZLP
753 FINZLP
754 FINZLP
755 FINZLP
756 FINZLP
757 FINZLP
758 FINZLP
759 FINZLP
760 FINZLP
761 FINZLP
762 FINZLP
763 FINZLP
764 FINZLP
765 FINZLP
766 FINZLP
767 FINZLP
768 FINZLP
769 FINZLP
770 FINZLP
771 FINZLP
772 FINZLP
773 FINZLP
774 FINZLP
775 FINZLP
776 FINZLP
777 FINZLP
778 FINZLP
779 FINZLP
780 FINZLP
781 FINZLP
782 FINZLP
783 FINZLP
784 FINZLP
785 FINZLP
786 FINZLP
787 FINZLP
788 FINZLP
789 FINZLP
790 FINZLP
791 FINZLP
792 FINZLP
793 FINZLP
794 FINZLP
795 FINZLP
796 FINZLP
797 FINZLP
798 FINZLP
799 FINZLP
800 FINZLP
801 FINZLP
802 FINZLP
803 FINZLP
804 FINZLP
805 FINZLP
806 FINZLP
807 FINZLP
808 FINZLP
809 FINZLP
810 FINZLP
811 FINZLP
812 FINZLP
813 FINZLP
814 FINZLP
815 FINZLP
816 FINZLP
817 FINZLP
818 FINZLP
819 FINZLP
820 FINZLP
821 FINZLP
822 FINZLP
823 FINZLP
824 FINZLP
825 FINZLP
826 FINZLP
827 FINZLP
828 FINZLP
829 FINZLP
830 FINZLP
831 FINZLP
832 FINZLP
833 FINZLP
834 FINZLP
835 FINZLP
836 FINZLP
837 FINZLP
838 FINZLP
839 FINZLP
840 FINZLP
841 FINZLP
842 FINZLP
843 FINZLP
844 FINZLP
845 FINZLP
846 FINZLP
847 FINZLP
848 FINZLP
849 FINZLP
850 FINZLP
851 FINZLP
852 FINZLP
853 FINZLP
854 FINZLP
855 FINZLP
856 FINZLP
857 FINZLP
858 FINZLP
859 FINZLP
860 FINZLP
861 FINZLP
862 FINZLP
863 FINZLP
864 FINZLP
865 FINZLP
866 FINZLP
867 FINZLP
868 FINZLP
869 FINZLP
870 FINZLP
871 FINZLP
872 FINZLP
873 FINZLP
874 FINZLP
875 FINZLP
876 FINZLP
877 FINZLP
878 FINZLP
879 FINZLP
880 FINZLP
881 FINZLP
882 FINZLP
883 FINZLP
884 FINZLP
885 FINZLP
886 FINZLP
887 FINZLP
888 FINZLP
889 FINZLP
890 FINZLP
891 FINZLP
892 FINZLP
893 FINZLP
894 FINZLP
895 FINZLP
896 FINZLP
897 FINZLP
898 FINZLP
899 FINZLP
900 FINZLP
901 FINZLP
902 FINZLP
903 FINZLP
904 FINZLP
905 FINZLP
906 FINZLP
907 FINZLP
908 FINZLP
909 FINZLP
910 FINZLP
911 FINZLP
912 FINZLP
913 FINZLP
914 FINZLP
915 FINZLP
916 FINZLP
917 FINZLP
918 FINZLP
919 FINZLP
920 FINZLP
921 FINZLP
922 FINZLP
923 FINZLP
924 FINZLP
925 FINZLP
926 FINZLP
927 FINZLP
928 FINZLP
929 FINZLP
930 FINZLP
931 FINZLP
932 FINZLP
933 FINZLP
934 FINZLP
935 FINZLP
936 FINZLP
937 FINZLP
938 FINZLP
939 FINZLP
940 FINZLP
941 FINZLP
942 FINZLP
943 FINZLP
944 FINZLP
945 FINZLP
946 FINZLP
947 FINZLP
948 FINZLP
949 FINZLP
950 FINZLP
951 FINZLP
952 FINZLP
953 FINZLP
954 FINZLP
955 FINZLP
956 FINZLP
957 FINZLP
958 FINZLP
959 FINZLP
960 FINZLP
961 FINZLP
962 FINZLP
963 FINZLP
964 FINZLP
965 FINZLP
966 FINZLP
967 FINZLP
968 FINZLP
969 FINZLP
970 FINZLP
971 FINZLP
972 FINZLP
973 FINZLP
974 FINZLP
975 FINZLP
976 FINZLP
977 FINZLP
978 FINZLP
979 FINZLP
980 FINZLP
981 FINZLP
982 FINZLP
983 FINZLP
984 FINZLP
985 FINZLP
986 FINZLP
987 FINZLP
988 FINZLP
989 FINZLP
990 FINZLP
991 FINZLP
992 FINZLP
993 FINZLP
994 FINZLP
995 FINZLP
996 FINZLP
997 FINZLP
998 FINZLP
999 FINZLP
1000 FINZLP

```

Source: David T Craig

"DTCA2DOC-007-101.PICT" 131 KB 2001-03-26 dpi: 250h x 250v pix: 1998h x 2209v

"DTCA2DOC-007-102.PICT" 150 KB 2001-03-26 dpi: 250h x 250v pix: 1993h x 2345v

[illegible]

```

29-MAR-84 08:16 PAGE 105

XPON 03 ROMSQF16
383 JSR SQR(X)=X.5
384 JSR SQR(X)=X.5
385 JSR SQR(X)=X.5
386 JSR SQR(X)=X.5
387 JSR SQR(X)=X.5
388 JSR SQR(X)=X.5
389 JSR SQR(X)=X.5
390 JSR SQR(X)=X.5
391 JSR SQR(X)=X.5
392 JSR SQR(X)=X.5
393 JSR SQR(X)=X.5
394 JSR SQR(X)=X.5
395 JSR SQR(X)=X.5
396 JSR SQR(X)=X.5
397 JSR SQR(X)=X.5
398 JSR SQR(X)=X.5
399 JSR SQR(X)=X.5
400 JSR SQR(X)=X.5
401 JSR SQR(X)=X.5
402 JSR SQR(X)=X.5
403 JSR SQR(X)=X.5
404 JSR SQR(X)=X.5
405 JSR SQR(X)=X.5
406 JSR SQR(X)=X.5
407 JSR SQR(X)=X.5
408 JSR SQR(X)=X.5
409 JSR SQR(X)=X.5
410 JSR SQR(X)=X.5
411 JSR SQR(X)=X.5
412 JSR SQR(X)=X.5
413 JSR SQR(X)=X.5
414 JSR SQR(X)=X.5
415 JSR SQR(X)=X.5
416 JSR SQR(X)=X.5
417 JSR SQR(X)=X.5
418 JSR SQR(X)=X.5
419 JSR SQR(X)=X.5
420 JSR SQR(X)=X.5
421 JSR SQR(X)=X.5
422 JSR SQR(X)=X.5
423 JSR SQR(X)=X.5
424 JSR SQR(X)=X.5
425 JSR SQR(X)=X.5
426 JSR SQR(X)=X.5
427 JSR SQR(X)=X.5
428 JSR SQR(X)=X.5
429 JSR SQR(X)=X.5
430 JSR SQR(X)=X.5
431 JSR SQR(X)=X.5
432 JSR SQR(X)=X.5
433 JSR SQR(X)=X.5
434 JSR SQR(X)=X.5
435 JSR SQR(X)=X.5

;SQUARE ROOT FUNCTION --- SQR(A)
;USE SQR(X)=X.5
;SQR:
;MOVE FAC INTO ARG.
;PUT MEMORY INTO FAC.
;INTO ACCX,
;LAST THING FETCHED IS FACEXP. INTO FPM.
;EXPONENTIATION -- XY.
;N.B. DO=1
;FIRST CHECK IF Y=0. IF SO, THE RESULT IS 1.
;THEN CHECK IF X=0. IF SO, THE RESULT IS 0.
;IF X.GT.0. IF NOT CHECK THAT Y IS AN INTEGER.
;IF SO, NEGATIVE X, SO THAT LOG DOESN'T GIVE FCERR.
;IF X IS NEGATIVE AND Y IS ODD, NEGATE THE RESULT
;RETURNED BY EXP.
;TO COMPUTE THE RESULT USE XY=EXP((Y*LOG(X))).
;IF FAC=0, JUST EXPONENTIATE THAT.
;IS X=0?
;ZERO FAC.
;SAVE FOR LATER IN A TEMP.
;GOOD IN CASE NO ONE CALLS INT.
;NO PROBLEMS IF X.GT.0.
;INTEGERIZE THE FAC.
;GET ADDR OF COMPERAND.
;EQUAL?
;LEAVE X NEG. LOG WILL BLOW HIM OUT.
;NEGATE X. MAKE POSITIVE.
;GET EVENNESS.
;ALTERNATE ENTRY POINT.
;SAVE EVENNESS FOR TER.
;FIND LOG.
;MULTIPLY FAC TIMES LOG(X).
;EXPONENTIATE THE FAC.
;IS IT EVEN?
;YES. OR X.GT.0.

;NEGATE THE NUMBER IN FAC.
;NEGOP:
;LDA #0
;BEQ FACEXP
;LDA #0
;BEQ FACGRTS
;LDA #0
;BEQ FACCSGN
;STA #0
;RTS

;NEGRTS:

```


[illegible]

```

;;TO SAVE IN ARG WITHOUT ROUND.
;;IF ABS(FAC) -GE. 128, TOO BIG.
;;OVERFLOW OR OVERFLOW.
;;GET LOW PART.

;;OVERFLOW OR OVERFLOW !!
;;SUBTRACT 1.
;;SAVE A WHILE.
;;PREP TO SWAP FAC AND ARG.

;;NEGATE FAC.

;;MULTIPLY BY POSITIVE 1.0.
;;GET SCALE FACTOR.
;;MODIFY FACEXP AND CHECK FOR OVERFLOW.
;;HAS TO DO JSR DUE TO PULAS IN

```

```

03 RMSOFTS
EF5C:
EF5D:
EF5E:
EF5F:
EF60:
EF61:
EF62:
EF63:
EF64:
EF65:
EF66:
EF67:
EF68:
EF69:
EF70:
EF71:
EF72:
EF73:
EF74:
EF75:
EF76:
EF77:
EF78:
EF79:
EF80:
EF81:
EF82:
EF83:
EF84:
EF85:
EF86:
EF87:
EF88:
EF89:
EF90:
EF91:
EF92:
EF93:
EF94:
EF95:
EF96:
EF97:
EF98:
EF99:
EFA0:
EFA1:
EFA2:
EFA3:
EFA4:
EFA5:
EFA6:
EFA7:
EFA8:
EFA9:
EFAA:
EFAB:
EFAC:
EFAD:
EFAE:
EFAF:
EFB0:
EFB1:
EFB2:
EFB3:
EFB4:
EFB5:
EFB6:
EFB7:
EFB8:
EFB9:
EFBA:
EFBB:
EFBC:
EFBD:
EFBE:
EFBF:
EFC0:
EFC1:
EFC2:
EFC3:
EFC4:
EFC5:
EFC6:
EFC7:
EFC8:
EFC9:
EFCA:
EFCB:
EFCC:
EFCD:
EFCE:
EFCF:
EFD0:
EFD1:
EFD2:
EFD3:
EFD4:
EFD5:
EFD6:
EFD7:
EFD8:
EFD9:
EFDA:
EFDB:
EFDC:
EFDD:
EFDE:
EFDF:
EFE0:
EFE1:
EFE2:
EFE3:
EFE4:
EFE5:
EFE6:
EFE7:
EFE8:
EFE9:
EFEA:
EFEB:
EFEC:
EFED:
EFEE:
EFEF:
EFF0:
EFF1:
EFF2:
EFF3:
EFF4:
EFF5:
EFF6:
EFF7:
EFF8:
EFF9:
EFGA:
EFGB:
EFGC:
EFGD:
EFG E

```

POLYNIAL EVALUATOR AND THE RANDOM NU 29-MAR-84 08:16 PAGE 108
;EVALUATE P(X2)*X
;POINTER TO DEGREE IS IN Y,A.
;THE CONSTANT FOLLOW THE DEGREE.
;FOR X=FAC, COMPUTE:
;C0*X+C1*X2+C2*X3+C3*X4+...+C(N)*X(2*N+1)
POLYX:
STA POLYPT
STY POLYPT+1
JSR MOV1F
LDA #TEMPF1
JSR #MULT
JSR POLY1
LDA #TEMPF1
LDY #TEMPF1
JMP EVALUATOR.
;POLYNOMIAL EVALUATOR.
;POINTER TO DEGREE IS IN Y,A.
;COMPUTE:
;C0+C1*X+C2*X2+C3*X3+C4*X4+...+C(N-1)*X(N-1)+C(N)*XN.
POLY:
STA POLYPT
STY POLYPT+1
JSR MOV2F
LDA (POLYPT),Y
LDY DEGREE
POLY3
POLYPT+1
POLYPT+1
POLYPT+1
FMULT
POLYPT
POLYPT+1
#4+1
POLY4
POLYPT
POLYPT+1
FADD
#TEMPF2
#TEMPF2
DEGREE
POLY2
RANDRT: -RANDOM NUMBER GENERATOR.
;RAN THE LAST RANDOM NUMBER GENERATED IS RETURNED.
;IF ARG=0, A NEW SEQUENCE OF RANDOM NUMBERS IS
;STARTED USING THE ARGUMENT.
;MULTIPLY THE PREVIOUS RANDOM NUMBER IN THE SEQUENCE,
;AND ADD IN ANOTHER RANDOM CONSTANT. THE THEN HO
;AND LO BYTES ARE SHIFED IN BY RT, & THE EXPONENT IS PUT WHERE
;IT WILL BE SET TO 200 SO THE RESULT WILL BE LESS THAN 1. THIS
;IS THEN NORMALIZED AND SAVED FOR THE NEXT TIME.
;THE HO AND LOW OF GETTING A NUMBER LESS THAN OR GREATER

"DTCA2DOC-007-109.PICT" 104 KB 2001-03-26 dpi: 250h x 250v pix: 1957h x 2527v

```

29-MAR-84 08:16 PAGE 110

;PNTR TO PI/2.
;ADD IT IN.

;GET PNTR TO DIVISOR.
;GET SIGN OF RESULT.
;GET RESULT INTO ARG.
;INTEGERIZE FAC.
;ALWAYS HAVE THE SAME SIGN.
;KEEP ONLY THE FRACTIONAL PART.

;GET PNTR TO 1/4.
;COMPUTE 1/4-FAC.
;SAVE SIGN FOR LATER.

;FIRST QUADRANT.
;ADD 1/2 TO FAC.
;SIGN IS NEGATIVE?

;QUADRANTS II AND III COME HERE.
;IF POSITIVE, NEGATE IT.

;POINTER TO 1/4.
;ADD IT IN.
;GET ORIGINAL QUADRANT.

;IF NEGATIVE, NEGATE RESULT.

;DO APPROXIMATION POLYNOMIAL.
;MOVE FAC INTO TEMPORARY.
;REMEMBER WHETHER TO NEGATE.
;COMPUTE THE SIN.

;PUT SIGN INTO OTHER TEMP.

;PUT THIS MEMORY LOC INTO FAC.

;COSINE FUNCTION.
;USE COS(X)=SIN(X+PI/2)
;PI2
;PI2
;FADD
;PNTR TO PI/2.
;ADD IT IN.

;FALL INTO SIN.
;SINE IDENTITIES TO GET FAC IN QUADRANTS I OR IV.
;THE FAC IS DIVIDED BY 2*PI & THE INTEGER PART IS IGNORED
;BECAUSE SIN(X+PI)=SIN(X). THEN ARGUMENT CAN BE COMPARED
;WITH PI/2 & (2*PI)=1/4.
;IDENTITIES ARE THEN USED TO GET THE RESULT IN QUADRANTS
;I OR IV. AN APPROXIMATION POLYNOMIAL IS THEN USED TO
;COMPUTE SIN(X).

;SIN:
;JSR
;LDA #TWOPI
;LDY #<TWOPI
;LDX #ARGSGN
;JSR FDIVF
;JSR MOVAF
;LDA #0
;STA ARISGN
;JSR FSUBT
;LDY #FR4
;LDX #<FR4
;JSR FASGN
;SIN1
;FADDH
;FACSGN
;SIN2
;TANSGN
;TANSGN
;NEGOP
;FR4
;FADD
;SIN3
;NEGOPCON
;SINCON
;POLYX
;FUNCTION
;MOV1F
;LDA #0
;STA TANSGN
;LDY #TEMPFF3
;LDX #TEMPFF3
;JSR GMOVMPF3
;LDY #TEMPFF1
;LDX #TEMPFF1
;JSR MOVEM

;SIN1:
;SIN2:
;SIN3:
;POLYX:
;TAN:

```

Page 0114 of 0150

29-MAR-84 08:16 PAGE 112

310
320
330
340
350
360
370
380

DFB
DFB
DFB
DFB
DFB

INE, CO

749
750
751
752
753
754
755

03 ROMSOFTG

EO97:C8
EO98:D4
EO99:C8
EO9A:D5
EO9B:C4
EO9C:CE
EO9D:CA

```

29-MAR-84 08:16 PAGE 113

;USE IDENTITIES TO GET ARG BETWEEN 0 AND 1 AND THEN USE AN
;APPROXIMATION POLYNOMIAL TO COMPUTE ARCTAN(X).
ATN:
LDA PHA
BPL ATN1
JSR NEGOP
;FACSGN
;WHAT IS SIGN?
;(MEANWHILE SAVE FOR LATER.)
;IF NEGATIVE, NEGATE FAC.
;SAVE THIS TOO FOR LATER.
;SEE IF FAC >= 1.0.
;IT IS LESS THAN 1.
;GET PNTR TO 1.0.
;COMPUTE RECROCAL.
;PNTR TO ARCTAN CONSTANTS.
;WAS ORIGINAL ARGUMENT >LT. 1 ?
;YES.
;SUBTRACT ARCTAGN FROM PI/2.
;WAS ORIGINAL ARGUMENT POSITIVE?
;YES.
;IF NEGATIVE, NEGATE RESULT.
;ALL DONE.
;DEGREE-1.
;-.0006847939119

; -.004850942156

; -.01611170184

; -.03420963805

; -.05427913276

; .07245719654

RCT 758
759 ;USE IDENTITIES TO GET ARG BETWEEN 0 AND 1 AND THEN USE AN
760 ;APPROXIMATION POLYNOMIAL TO COMPUTE ARCTAN(X).
761 ATN:
762 LDA PHA
763 BPL ATN1
764 JSR NEGOP
765 ;FACSGN
766 ;WHAT IS SIGN?
767 ;(MEANWHILE SAVE FOR LATER.)
768 ;IF NEGATIVE, NEGATE FAC.
769 ;SAVE THIS TOO FOR LATER.
770 ;SEE IF FAC >= 1.0.
771 ;IT IS LESS THAN 1.
772 ;GET PNTR TO 1.0.
773 ;COMPUTE RECROCAL.
774 ;PNTR TO ARCTAN CONSTANTS.
775 ;WAS ORIGINAL ARGUMENT >LT. 1 ?
776 ;YES.
777 ;SUBTRACT ARCTAGN FROM PI/2.
778 ;WAS ORIGINAL ARGUMENT POSITIVE?
779 ;YES.
780 ;IF NEGATIVE, NEGATE RESULT.
781 ;ALL DONE.
782 ;DEGREE-1.
783 ;-.0006847939119
784
785 ; -.004850942156
786
787 ; -.01611170184
788
789 ; -.03420963805
790
791 ; -.05427913276
792
793 ; .07245719654
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```


"DTCA2DOC-007-115.PICT" 140 KB 2001-03-26 dpi: 250h x 250v pix: 1978h x 2340v

TYPE A CR.

SET PRE-BUF BYTES NON-ZERO FOR CHEAD

SET UP STRING TEMPORARIES.

```

;GET SIZE OF MEMORY INPUT.
;HIGHEST ADDRESS.

```

THIS IS THE SIZE OF MEMORY.
TOP OF STRINGS TOO.

SET UP TEXT TABLE.

PREPARE TO USE "REASON".
;SET UP EVERYTHING ELSE.

JMP (INDIRECT)

"DTCA2DOC-007-116.PICT" 126 KB 2001-03-26 dpi: 250h x 250v pix: 1998h x 2421v

29-MAR-84 08:16 PAGE 117

(START+1)

JMP
FIN

Y 119
120

09 ROMSOFTI

\$
F1D5:

"DTCA2DOC-007-118.PICT" 122 KB 2001-03-26 dpi: 250h x 250v pix: 2003h x 2229v

```

29-MAR-84 08:16 PAGE 119

PPLESOFT EXTENSIONS
181 09 ROMSOFTI
182 F22E:4C 9F F3
183 F231:
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

;Y-REG=X-COORD
;OUT OF RANGE

;THIRD ARG IN ACC

ARG IN ACC

ARG IN Y-REG

;COLOR TO A-REG.

;PREPARE FOR TABV SUBR.
;OFF OF SCREEN?

PLOTDO
$F819
$F800
;ACC=Y-COORD
FIRST
#40
TOOBIG
PLOTDO
LOLLY
ATFNS
H2
XBOUND
#48
TOOBIG
ATFNS
;THIRD
H2
#40
TOOBIG
FIRST
HLINE
LOLLY
ATFNS
XBOUND
FIRST
VLINXX
$F828
ATFNS
;THIRD
#40
TOOBIG
FIRST
VLIN
GETBYT
SETCOL
PLIST
OFF
$F364
GETBYT
#18
TOOBIG
TABV

```

```

29-MAR-84 08:16 PAGE 120

;FOR OUTPUT SPEED
;ADJUST TRFLAG FOR TRACE.
;BACK TO CALLER.

;ALWAYS TAKEN
;GET ARG FOR HIMEM:
;PROGRAM OVERFLOW?

PPLESOFT EXTENSIONS
239 TABV EQU $F858
240 SETSPD: JSR GETBYT
241 TAX # $FF
242 INX
243 STX
244 RTS
245 SEC
246 BCC
247 ORG C
248 CLC
249 ROR
250 RTS
251
252 * SETNORM: LDA # $FF
253 INVERSE: BNE SETTYPE
254 SETTYPE: LDA # $3F
255 SETMODE: LDA # $00
256 FLASH: STX INVF LG
257 RTS OR MASK
258
259 * HIMEMSET: LDA # $7F
260 BNE SETMODE
261
262 JSR FRMNUM
263 LDA GETADR
264 LDBA POKER
265 LDBA STREND+1
266 LDBA STREND+1
267 BCS DOSET
268 JMP OMERR
269 LDA POKER
270 STX MEMSIZ
271 LDA FRETOP
272 STX POKER+1
273 LDA MEMSIZ+1
274 STX FRETOP+1
275 RTS
276
277 LOMEMSET: FRMNUM
278 JSR GETADR
279 LDA POKER
280 LDBA MEMSIZ+1
281 LDBA MEMFULL
282 BCS POKER
283 LDA VARTAB
284 LDA POKER+1
285 LDBA VARTAB+1
286 BCC MEMFULL
287 LDA POKER
288 STX VARTAB
289 LDA POKER+1
290 STX VARTAB+1
291 RTS
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

Source: David T Craig

"DTCA2DOC-007-122.PICT" 135 KB 2001-03-26 dpi: 250h x 250v pix: 1998h x 2517v

```

09 ROMSOFTI
F3AD:50 B8 F7 F3BD
F3AE:20 0A
F3B1:90 0A
F3B3:0A 30
F3B4:0A 30
F3B7:0A 30
F3B8:0A 30
F3B9:0A 30
F3BA:0A 30
F3BD:20 0E F8 F3BD
F3C0:90 07 F8 F3C9
F3C2:0A 54 C0
F3C5:0A 54 C0
F3C7:0A 54 C0
F3C9:0A 54 C0
F3CA:0A 54 C0
F3CE:0A 54 C0
F3D0:0A 54 C0
F3D3:0A 54 C0
F3D7:0A 54 C0
F3D8:0A 54 C0
F3D9:0A 54 C0
F3DA:0A 54 C0
F3DB:0A 54 C0
F3DC:0A 54 C0
F3DD:0A 54 C0
F3DE:0A 54 C0
F3DF:0A 54 C0
F3E0:0A 54 C0
F3E1:0A 54 C0
F3E2:0A 54 C0
F3E3:0A 54 C0
F3E4:0A 54 C0
F3E5:0A 54 C0
F3E6:0A 54 C0
F3E7:0A 54 C0
F3E8:0A 54 C0
F3E9:0A 54 C0
F3EA:0A 54 C0
F3EB:0A 54 C0
F3EC:0A 54 C0
F3ED:0A 54 C0
F3EE:0A 54 C0
F3EF:0A 54 C0
F3F0:0A 54 C0
F3F1:0A 54 C0
F3F2:0A 54 C0
F3F3:0A 54 C0
F3F4:0A 54 C0
F3F5:0A 54 C0
F3F6:0A 54 C0
F3F7:0A 54 C0
F3F8:0A 54 C0
F3F9:0A 54 C0
F3FA:0A 54 C0
F3FB:0A 54 C0
F3FC:0A 54 C0
F3FD:0A 54 C0
F3FE:0A 54 C0
F3FF:0A 54 C0

PPLESOFT EXTENSIONS
413 PLOT1
414 JSR RHY
415 BCC PHX
416 LDA PHX
417 TAX
418 LSR
419 TXAR
420 RORC
421 STAU
422 JSR EQU
423 JSR EQU
424 JSR EQU
425 JSR EQU
426 JSR EQU
427 JSR EQU
428 JSR EQU
429 JSR EQU
430 JSR EQU
431 JSR EQU
432 JSR EQU
433 JSR EQU
434 JSR EQU
435 JSR EQU
436 JSR EQU
437 JSR EQU
438 JSR EQU
439 JSR EQU
440 JSR EQU
441 JSR EQU
442 JSR EQU
443 JSR EQU
444 JSR EQU
445 JSR EQU
446 JSR EQU
447 JSR EQU
448 JSR EQU
449 JSR EQU
450 JSR EQU
451 JSR EQU
452 JSR EQU
453 JSR EQU
454 JSR EQU
455 JSR EQU
456 JSR EQU
457 JSR EQU
458 JSR EQU
459 JSR EQU
460 JSR EQU
461 JSR EQU
462 JSR EQU
463 JSR EQU
464 JSR EQU
465 JSR EQU
466 JSR EQU
467 JSR EQU
468 JSR EQU
469 JSR EQU
470 JSR EQU

PLOT3
423 JSR EQU
424 JSR EQU
425 JSR EQU
426 JSR EQU
427 JSR EQU
428 JSR EQU
429 JSR EQU
430 JSR EQU
431 JSR EQU
432 JSR EQU
433 JSR EQU
434 JSR EQU
435 JSR EQU
436 JSR EQU
437 JSR EQU
438 JSR EQU
439 JSR EQU
440 JSR EQU
441 JSR EQU
442 JSR EQU
443 JSR EQU
444 JSR EQU
445 JSR EQU
446 JSR EQU
447 JSR EQU
448 JSR EQU
449 JSR EQU
450 JSR EQU
451 JSR EQU
452 JSR EQU
453 JSR EQU
454 JSR EQU
455 JSR EQU
456 JSR EQU
457 JSR EQU
458 JSR EQU
459 JSR EQU
460 JSR EQU
461 JSR EQU
462 JSR EQU
463 JSR EQU
464 JSR EQU
465 JSR EQU
466 JSR EQU
467 JSR EQU
468 JSR EQU
469 JSR EQU
470 JSR EQU

PLOT4
433 JSR EQU
434 JSR EQU
435 JSR EQU
436 JSR EQU
437 JSR EQU
438 JSR EQU
439 JSR EQU
440 JSR EQU
441 JSR EQU
442 JSR EQU
443 JSR EQU
444 JSR EQU
445 JSR EQU
446 JSR EQU
447 JSR EQU
448 JSR EQU
449 JSR EQU
450 JSR EQU
451 JSR EQU
452 JSR EQU
453 JSR EQU
454 JSR EQU
455 JSR EQU
456 JSR EQU
457 JSR EQU
458 JSR EQU
459 JSR EQU
460 JSR EQU
461 JSR EQU
462 JSR EQU
463 JSR EQU
464 JSR EQU
465 JSR EQU
466 JSR EQU
467 JSR EQU
468 JSR EQU
469 JSR EQU
470 JSR EQU

VIDSTATE
433 JSR EQU
434 JSR EQU
435 JSR EQU
436 JSR EQU
437 JSR EQU
438 JSR EQU
439 JSR EQU
440 JSR EQU
441 JSR EQU
442 JSR EQU
443 JSR EQU
444 JSR EQU
445 JSR EQU
446 JSR EQU
447 JSR EQU
448 JSR EQU
449 JSR EQU
450 JSR EQU
451 JSR EQU
452 JSR EQU
453 JSR EQU
454 JSR EQU
455 JSR EQU
456 JSR EQU
457 JSR EQU
458 JSR EQU
459 JSR EQU
460 JSR EQU
461 JSR EQU
462 JSR EQU
463 JSR EQU
464 JSR EQU
465 JSR EQU
466 JSR EQU
467 JSR EQU
468 JSR EQU
469 JSR EQU
470 JSR EQU

*
433 JSR EQU
434 JSR EQU
435 JSR EQU
436 JSR EQU
437 JSR EQU
438 JSR EQU
439 JSR EQU
440 JSR EQU
441 JSR EQU
442 JSR EQU
443 JSR EQU
444 JSR EQU
445 JSR EQU
446 JSR EQU
447 JSR EQU
448 JSR EQU
449 JSR EQU
450 JSR EQU
451 JSR EQU
452 JSR EQU
453 JSR EQU
454 JSR EQU
455 JSR EQU
456 JSR EQU
457 JSR EQU
458 JSR EQU
459 JSR EQU
460 JSR EQU
461 JSR EQU
462 JSR EQU
463 JSR EQU
464 JSR EQU
465 JSR EQU
466 JSR EQU
467 JSR EQU
468 JSR EQU
469 JSR EQU
470 JSR EQU

;THIS IS THE
;START=NAME POS.+#DIMS*2+5
;END
;RECALL IS SIMILAR
;STORE:
JSR EQU
LDY EQU
TAX EQU
DEX EQU
SBC EQU
BCS EQU
DEX EQU
STX EQU
JSR EQU
JSR EQU
JMP EQU
NOW EQU
JSR EQU
JSR EQU
LDY EQU
CMP EQU

;OTHER ROUTINES ENTER HERE
;TEST FOR DOUBLE GR, SET Y
;BRANCH IF REG GR
;SAVE X ON TOP OF STACK
;MUST MUCK WITH COLORS IN ALT BANK
;SAVE IN X FOR LATER COLOR RESTORE
;ROTATE 8 BITS RIGHT
;RESET CARRY FOR ALT PAGE INDICATOR
;GO PLOT THE DOT
;BRANCH IF NOT SPECIAL HANDLED
;RESTORE UNTRANSLATED COLOR
;RESTORE ORIGINAL X
;PLOT ALWAYS RETURNS CARRY CLEAR.
;RESTORE ORIGINAL Y
;GET STATE OF DOUBLE GR
;80-STORE MUST ALSO BE ENABLED
;RETURN IN CARRY
;MAKE EM BREAKS
"STORE" ROUTINE.
POS.+LENGTH -1
(SHARES ROUTINES)
;SET UP.
;GET LENGTH
;X=HIGH
;SHOULD IT BE A 3 ??
;WRITE LEN TO CASSETTE
RECALL....
;GET LEN OF FILE
;TOO BIG?

```

```

29-MAR-84 08:16 PAGE 124

PPLESOFT EXTENSIONS
471 0000
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

;COMPARE
; (LOWTR),Y
POKER+1
*+3
OMERR
DOPTRS
C.CSIN
PLIST
OFF

;TITLE "APPLE II HI-RES GRAPHICS ROUTINES"
*****
* APPLE-II HI-RESOLUTION *
* GRAPHICS SUBROUTINES *
* BY WOZ 9/13/77 *
* ALL RIGHTS RESERVED *
*****

```

POINTER TO
SHAPE LIST
RUNNING COLOR MASK
BASE ADR FOR CURRENT
HI-RES PLOT LINE. A

MONITOR A1.

MONITOR A2.

BASIC 'START OF VARS'.

DELTA-X FOR HLIN, SHAPE.

SHAPE TEMP. HLIN, SHAPE-
DELTA-Y FOR HLIN, SHAPE-
ROT QUADRANT (SHAPE).
ERROR FOR HLIN.

PRIOR X-COORD SAVE
AFTER HLIN OR HPLOT, SAVE.
HLIN, HPLOT Y-COORD, HPOSN.
COLOR, FOR HPLOT, SAVE.
HORIZ, OFFSET, SAVE.
HI-RES PAGE (\$20 NORMAL)
SCALE FOR SHAPE, MOVE.

TABLE. COUNT. FACTOR >>> UNDOCUMENTED ZPG BYTE <<<
COLLISION ROTATION SHAPE TABLE.
HOLD OF SHAPE RES VIDEO
SWITCH TO HI-GRAPHS
SELECT TEXT/GRAPHS
SELECT GRAPE MODE.
--BYTE TAPE READ SETUP.
WO-EDGE TAPE SENSE.
READ WITHOUT HEADER.
FOR PRIMARY PAGE.
FOR SECONDARY PAGE.
WHOLE SCREEN GRAPHICS

I-RES EQUATES

[illegible]

10 ROMSOFT J

[illegible]

```

29-MAR-84 08:16 PAGE 126

;DISPLAY SECOND PAGE.
;ALWAYS TAKEN.
;$20 INIT FOR $2000-3FFF
HI-RES SCREEN MEMORY.
SET HIRES DISPLAY MODE
SET GRAPHICS DISPLAY
SET FOR BLACK BKGND.
INIT HI-RES SCREEN MEM
FOR CURRENT PAGE, NORMALLY
$2000-3FFF OR $4000-5FFF
(SHAPE,H) WILL SPECIFY
32 SEPARATE PAGES.
THROUGHOUT THE INIT.
TEST FOR DONE.

** MICROSOFT VERSION
57 * * * * *
58 SETHRH: BIT HIPAGE
59 LDA BNE BIT FULSET
60 BNE LDA #40
61 SET SET
62 LOPAGE
63 MIXSET
64 HPAG
65 HIRES
66 TXTCLE
67 #50
68 HCOLOR1
69 HPAG
70 SHAPEH
71 #50
72 SHAPEL
73 HCOLOR1
74 (SHAPEL),Y
75 CSHFT2
76 BKGND1
77 SHAPEH
78 SHAPEL
79 #51F
80 BKGND1
81
82
83
84
85

```

ENTER WITH Y IN A-REG,
XL IN X-REG,
AND XH IN Y-REG.

```
FOR Y-COORD = 00ABCOEF.  
CALCULATES BASE ADDRESS.  
IN HBASL, HBASH FOR  
ACCESSING SCREEN MEM  
VIA (HBASL), Y ADDRESSING MODE
```

CALCULATES
HBASH = PPPFGHCD,
HBASL = EABA8000

WHERE PPP=001 FOR \$2000.3FFF
SCREEN MEM RANGE AND
PPP=010 FOR \$4000-7FFF
(GIVEN Y-COORD=ABCDEFGH)

DIVIDE XD BY 7 FOR
INDEX FROM BASE ADDR
(QUOTIENT) AND BIT 8
(WHICH SCREEN MEM
MASK SPEC'D BY REMAINDER)

SUBTRACT OUT SEVENS.

WORKS FOR X0 FROM
0 TO 279, LOW-ORDER.
BYTE IN X-REG,
HIGH IN Y-REG, ON ENTRY

IF ON ODD BYTE (CARRY
THEN ROTATE COLOR
BIT FOR 180 DEGREE TO
PRIOR TO COPYING

CALC BIT POSN IN HBAS
HNDX, AND HMASK FROM
Y-COORD IN A-Y-REGS.
X-COORD IN X-BITS OF H
SUBSTITUTE CORRESPONDING

I-RES GRAP

HOX
FOX
DOX

HBASL
A
HBASL
A
HBASL
HBASL

MEMPHIS
444

AHBASH
AHBASL

#50 SN2
#HP0523
#524

#\$7
HPD SN1
HNDX

MSK TBL-\$F9,X
HMASK

ACHCOLOR1
ACHCOLOR2

```
HPPOSN
HCOLOR1
(HBASL),Y
HMASK
(HBASL),Y
(HBASL),Y
```

* BIT OF HCOLOR1.

10 RMSOFTJ

[illegible]

Source: David T Craig

```

29-MAR-84 08:16 PAGE 129
NO 90 DEG ROT (X-OR).
IF B2=0 THEN NO PLOT.
FOR EX-OR INTO SCREEN
SCREEN BIT SET?

ALWAYS TAKEN.
NO 90 DEG ROT.
IF B2=0 THEN NO PLOT.

SET HI-RES SCREEN BIT
TO CORRESPONDING HC
IF BIT OF SCREEN CHAN
THEN INCR COLLN DE

ADD QDRNT TO
SPECIFIED VECTOR
AND MOVE LFT, RT,
UP, OR DWN BASED
ON SIGN AND CARRY.

SIGN FOR UP/DWN SELECT

CALC BASE ADDRESS
(ADR OF LEFTMOST BYTE)
FOR NEXT LINE UP
IN (HBASL, HBASH)
WITH 192-LINE WRAPA

**** BIT MAP ****
FOR ROW = ABCDEFGH,

HBASL = EABAB000
HBASH = PPPFGHCD

WHERE PPP=001 FOR PRIMARY
HI-RES PAGE ($2000-

```


CALC BASE ADR FOR NEX
DOWN TO CHBASL/HBAS

WITH 192-LINE WRAPA

[illegible]

```

29-MAR-84 08:16 PAGE 131

SET (XOL,XOH) AND
YO TO ZERO FOR
REL LINE DRAW
(DX,DY).

ON ENTRY
XL: A-REG
XH: X-REG
Y: Y-REG

CALC ABS(X-XO)
IN (DXL,DXH)

X DIR TO SIGN BIT
OF QDRNT (DX POS)
D=RIGHT (DX NEG)
1=LEFT (DX NEG)

INIT (EL,EH) TO
ABS(X-XO)

CALC -ABS(Y-YO)-1
IN DY.

ROTATE Y DIR INTO
QDRNT SIGN BIT
(0=UP, 1=DOWN)

INIT (COUNT, COUNT)-
TO -(DELTX+DELTY+1)

HORIZ INDEX
ALWAYS TAKEN.
MOVE IN X-DIR. USE
QDRNT B6 FOR LFT/RT

ASSUME CARRY SET.
(EL,EH)-DELTY TO (EL,
NOTE: DY IS (-DELTY)-1
CARRY CLR IF (EL,EH)
GOES NEG.

SCREEN BYTE.
PLOT DOT OF HCOLOR1.
CURRENT BIT MASK.

#0
XOL
XOH
YO

XOL

XOH
QDRNT
HLIN2

$FF
$1

$0
QDRNT
DXH
EH

DXL
EL

XOL
XOH

YO
HLIN3
$FF
$FE
DY
YO
QDRNT

DXL

$FF
DXH
COUNT
HNDX
MOVEX2
A
LFTRT

EL
DY
EH
$0
EH
(HBASL),Y
HCOLOR1
HMASK
(HBASL),Y

PHA
LDA
STA
STA
PLA
PHAC
SBC
PHA
TXAC
SBC
STA
BCS
PLA
EOR
ADC
PHA
SBC
STA
PLA
STA
PLA
STA
TXAC
GLBC
BCC
EOR
ADC
STA
RDR
SBC
TAX
LBC
STA
LBC
BCS
ASL
JSC
LDA
ADC
STA
LBC
STA
LDA
EOR
EOR

I-RES GRAP
254 HLINRL
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311

HILIN

HILIN2

HILIN3

MOVEX

MOVEX2

HCOUNT

F550

F568

F581

F4

```

"DTCA2DOC-007-132.PICT" 75 KB 2001-03-26 dpi: 250h x 250v pix: 1973h x 2123v

29-MAR-84 08:16 PAGE 133

```

10 ROMSOFTJ
20
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

"DTCA2DOC-007-134.PICT" 130 KB 2001-03-26 dpi: 250h x 250v pix: 1962h x 2129v

```

29-MAR-84 08:16 PAGE 135

10 ROMSOFTJ
F650: 56 1A
F651: 84 1B
F652: AA
F653: 4A
F654: 4A
F655: 4A
F656: 4A
F657: 4A
F658: 4A
F659: 4A
F660: 4A
F661: 4A
F662: 4A
F663: 4A
F664: 4A
F665: 4A
F666: 4A
F667: 4A
F668: 4A
F669: 4A
F670: 4A
F671: 4A
F672: 4A
F673: 4A
F674: 4A
F675: 4A
F676: 4A
F677: 4A
F678: 4A
F679: 4A
F680: 4A
F681: 4A
F682: 4A
F683: 4A
F684: 4A
F685: 4A
F686: 4A
F687: 4A
F688: 4A
F689: 4A
F690: 4A
F691: 4A
F692: 4A
F693: 4A
F694: 4A
F695: 4A
F696: 4A
F697: 4A
F698: 4A
F699: 4A
F700: 4A
F701: 4A
F702: 4A
F703: 4A
F704: 4A
F705: 4A
F706: 4A
F707: 4A
F708: 4A
F709: 4A
F710: 4A
F711: 4A
F712: 4A
F713: 4A
F714: 4A
F715: 4A
F716: 4A
F717: 4A
F718: 4A
F719: 4A
F720: 4A
F721: 4A
F722: 4A
F723: 4A
F724: 4A
F725: 4A
F726: 4A
F727: 4A
F728: 4A
F729: 4A
F730: 4A
F731: 4A
F732: 4A
F733: 4A
F734: 4A
F735: 4A
F736: 4A
F737: 4A
F738: 4A
F739: 4A
F740: 4A
F741: 4A
F742: 4A
F743: 4A
F744: 4A
F745: 4A
F746: 4A
F747: 4A
F748: 4A
F749: 4A
F750: 4A
F751: 4A
F752: 4A
F753: 4A
F754: 4A
F755: 4A
F756: 4A
F757: 4A
F758: 4A
F759: 4A
F760: 4A
F761: 4A
F762: 4A
F763: 4A
F764: 4A
F765: 4A
F766: 4A
F767: 4A
F768: 4A
F769: 4A
F770: 4A
F771: 4A
F772: 4A
F773: 4A
F774: 4A
F775: 4A
F776: 4A
F777: 4A
F778: 4A
F779: 4A
F780: 4A
F781: 4A
F782: 4A
F783: 4A
F784: 4A
F785: 4A
F786: 4A
F787: 4A
F788: 4A
F789: 4A
F790: 4A
F791: 4A
F792: 4A
F793: 4A
F794: 4A
F795: 4A
F796: 4A
F797: 4A
F798: 4A
F799: 4A
F800: 4A
F801: 4A
F802: 4A
F803: 4A
F804: 4A
F805: 4A
F806: 4A
F807: 4A
F808: 4A
F809: 4A
F810: 4A
F811: 4A
F812: 4A
F813: 4A
F814: 4A
F815: 4A
F816: 4A
F817: 4A
F818: 4A
F819: 4A
F820: 4A
F821: 4A
F822: 4A
F823: 4A
F824: 4A
F825: 4A
F826: 4A
F827: 4A
F828: 4A
F829: 4A
F830: 4A
F831: 4A
F832: 4A
F833: 4A
F834: 4A
F835: 4A
F836: 4A
F837: 4A
F838: 4A
F839: 4A
F840: 4A
F841: 4A
F842: 4A
F843: 4A
F844: 4A
F845: 4A
F846: 4A
F847: 4A
F848: 4A
F849: 4A
F850: 4A
F851: 4A
F852: 4A
F853: 4A
F854: 4A
F855: 4A
F856: 4A
F857: 4A
F858: 4A
F859: 4A
F860: 4A
F861: 4A
F862: 4A
F863: 4A
F864: 4A
F865: 4A
F866: 4A
F867: 4A
F868: 4A
F869: 4A
F870: 4A
F871: 4A
F872: 4A
F873: 4A
F874: 4A
F875: 4A
F876: 4A
F877: 4A
F878: 4A
F879: 4A
F880: 4A
F881: 4A
F882: 4A
F883: 4A
F884: 4A
F885: 4A
F886: 4A
F887: 4A
F888: 4A
F889: 4A
F890: 4A
F891: 4A
F892: 4A
F893: 4A
F894: 4A
F895: 4A
F896: 4A
F897: 4A
F898: 4A
F899: 4A
F900: 4A
F901: 4A
F902: 4A
F903: 4A
F904: 4A
F905: 4A
F906: 4A
F907: 4A
F908: 4A
F909: 4A
F910: 4A
F911: 4A
F912: 4A
F913: 4A
F914: 4A
F915: 4A
F916: 4A
F917: 4A
F918: 4A
F919: 4A
F920: 4A
F921: 4A
F922: 4A
F923: 4A
F924: 4A
F925: 4A
F926: 4A
F927: 4A
F928: 4A
F929: 4A
F930: 4A
F931: 4A
F932: 4A
F933: 4A
F934: 4A
F935: 4A
F936: 4A
F937: 4A
F938: 4A
F939: 4A
F940: 4A
F941: 4A
F942: 4A
F943: 4A
F944: 4A
F945: 4A
F946: 4A
F947: 4A
F948: 4A
F949: 4A
F950: 4A
F951: 4A
F952: 4A
F953: 4A
F954: 4A
F955: 4A
F956: 4A
F957: 4A
F958: 4A
F959: 4A
F960: 4A
F961: 4A
F962: 4A
F963: 4A
F964: 4A
F965: 4A
F966: 4A
F967: 4A
F968: 4A
F969: 4A
F970: 4A
F971: 4A
F972: 4A
F973: 4A
F974: 4A
F975: 4A
F976: 4A
F977: 4A
F978: 4A
F979: 4A
F980: 4A
F981: 4A
F982: 4A
F983: 4A
F984: 4A
F985: 4A
F986: 4A
F987: 4A
F988: 4A
F989: 4A
F990: 4A
F991: 4A
F992: 4A
F993: 4A
F994: 4A
F995: 4A
F996: 4A
F997: 4A
F998: 4A
F999: 4A

I-RES GRAP
435 * * EX-OR SHAPE INTO SCREEN.
436 * *
437 * * ROT = 0 TO 3 (QUADRANT ONLY)
438 * * SCALE IS USED
439 * *
440 * *
441 * * XDRAW
442 * * XDRAW1
443 * *
444 * *
445 * *
446 * *
447 * *
448 * *
449 * *
450 * *
451 * *
452 * *
453 * *
454 * *
455 * *
456 * *
457 * *
458 * *
459 * *
460 * *
461 * *
462 * *
463 * *
464 * *
465 * *
466 * *
467 * *
468 * *
469 * *
470 * *
471 * *
472 * *
473 * *
474 * *
475 * *
476 * *
477 * *
478 * *
479 * *
480 * *
481 * *
482 * *
483 * *
484 * *
485 * *
486 * *
487 * *
488 * *
489 * *
490 * *
491 * *
492 * *

SHAPE DEFINITION
POINTER.
ROT ($0-$3F)
QDRNT 0=UP, 1=RT,
2=DN, 3=LFT.
SAVE COS AND SIN
VALS IN DXL AND DY,
INDEX FROM HI-RES
BASE ADR.
CLEAR COLLISION DETECT
1ST SHAPE DEF BYTE.
EL,EH FOR FRACTIONAL
L,R,U,D, VECTORS.
SCALE FACTOR.
IF FRAC COS OVFL
THEN MOVE IN
SPECIFIED VECTOR
DIRECTION
IF FRAC SIN OVFL
THEN MOVE IN
SPECIFIED VECTOR
DIRECTION +90 DEG.
LOOP ON SCALE
FACTOR.
NEXT 3-BIT VECTOR
OF SHAPE DEF.
NEXT BYTE OF
SHAPE DEF.
DONE IF ZERO.

```

"DTCA2DOC-007-136.PICT" 132 KB 2001-03-26 dpi: 250h x 250v pix: 1998h x 2300v

```

29-MAR-84 08:16 PAGE 137

;ACC=SHAPE NUMBER
;ZERO INDEX

SHAPEX
SHAPEH
#0
(SHAPEL,X)
DRAWOK
LOLLY
*

HTOOBIG
A
PREDRAW2
SHAPEH

(SHAPEL),Y
SHAPEL

(SHAPEL),Y
SHAPEX
SHAPEH
CHRGOT
#ATTKN
SHFINI
SYNCHR
HFNS
HPOSN
ROTBYT

PREDRAW
DRAW1
PREDRAW
PREDRAW1
LOLLY
VIDSTATE
XBND2
#40
XBND3
#80
XBND3
40 THEN SURELY LESS THAN 80
* IF LESS
XBND2
VLINE
V2
#48
XBND3
PLOT00
V2
A
VLINE

;SHAPE # *2
;IF OVEFLOW

;INDX INTO TABLE
;RELATIVE TABLE.
;PRESERVE
;ADD 2-BYTE INDEX TO SHAPE
;START ADDRESS

;OPTIONALLY FOLLOWED BY X,Y ARGS.
;IS IT THERE?
;IF NOT, FINISH UP.

;POSITION TO X,Y

;SET UP,
;THEN DRAW IT.

;DETERMINE IF DOUBLE GR

```



```

10 ROMSOFTJ
F797: A4 F0
F798: 20 F3
F799: C4 F3
F79A: 80 F795
F79B: 80 F795
F79C: 80 F795
F79D: 80 F795
F79E: 80 F795
F79F: 80 F795
F7A0: 80 F795
F7A1: 80 F795
F7A2: 80 F795
F7A3: 80 F795
F7A4: 80 F795
F7A5: 80 F795
F7A6: 80 F795
F7A7: 80 F795
F7A8: 80 F795
F7A9: 80 F795
F7AA: 80 F795
F7AB: 80 F795
F7AC: 80 F795
F7AD: 80 F795
F7AE: 80 F795
F7AF: 80 F795
F7B0: 80 F795
F7B1: 80 F795
F7B2: 80 F795
F7B3: 80 F795
F7B4: 80 F795
F7B5: 80 F795
F7B6: 80 F795
F7B7: 80 F795
F7B8: 80 F795
F7B9: 80 F795
F7BA: 80 F795
F7BB: 80 F795
F7BC: 80 F795
F7BD: 80 F795
F7BE: 80 F795
F7BF: 80 F795
F7C0: 80 F795
F7C1: 80 F795
F7C2: 80 F795
F7C3: 80 F795
F7C4: 80 F795
F7C5: 80 F795
F7C6: 80 F795
F7C7: 80 F795
F7C8: 80 F795
F7C9: 80 F795
F7CA: 80 F795
F7CB: 80 F795
F7CC: 80 F795
F7CD: 80 F795
F7CE: 80 F795
F7CF: 80 F795
F7D0: 80 F795
F7D1: 80 F795
F7D2: 80 F795
F7D3: 80 F795
F7D4: 80 F795
F7D5: 80 F795
F7D6: 80 F795
F7D7: 80 F795
F7D8: 80 F795
F7D9: 80 F795
F7DA: 80 F795
F7DB: 80 F795
F7DC: 80 F795
F7DD: 80 F795
F7DE: 80 F795
F7DF: 80 F795
F7E0: 80 F795
F7E1: 80 F795
F7E2: 80 F795
F7E3: 80 F795
F7E4: 80 F795
F7E5: 80 F795
F7E6: 80 F795
F7E7: 80 F795
F7E8: 80 F795
F7E9: 80 F795
F7EA: 80 F795
F7EB: 80 F795
F7EC: 80 F795
F7ED: 80 F795
F7EE: 80 F795
F7EF: 80 F795
F7F0: 80 F795
F7F1: 80 F795
F7F2: 80 F795
F7F3: 80 F795
F7F4: 80 F795
F7F5: 80 F795
F7F6: 80 F795
F7F7: 80 F795
F7F8: 80 F795
F7F9: 80 F795
F7FA: 80 F795
F7FB: 80 F795
F7FC: 80 F795
F7FD: 80 F795
F7FE: 80 F795
F7FF: 80 F795

I-RES GRAP
609
610
611 HLINE1
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666

RDSCRN41
*VIDSETY
XQXRTS
*TABER2
HTAB3
HTDONE
GETARYPT
*TABER3
*HTAB:
HTAB1:
F7F3
F7
HTAB2:

LDY
JSR
COPY
BCC
BNE
JSR
BRA
EQU
PHA
JSR
PLA
PHP
JSR
PLP
BCC
STP
CMP
ASL
AND
EQU
RTS
JSR
BCC
TYA
EOR
LSR
TAC
BCA
LDA
RTS
TXA
BIT
BMI
DSE
SSEC
TXAC
SBC
RTS
LDA
STA
JSR
STZ
RTS
SBC
RTS
JSR
DEX
LDA
CMP
BCS
LDA
JSR
STX

```

29-MAR-84 08:16 PAGE 138

;BRANCH IF DONE WITH PLOT

;BRANCH ALWAYS TAKEN

;ADJUST Y AS NECESSARY, CARRY SET=PG2GR

;ROM RDSCRN

;EFFECTIVE 4 BIT ROTATE LEFT
; FOR ALT MEM COLORS

;TEST 40/80 GRAFIX

;CALC SPACES TO NEXT COMMA
; TAB POSITION IN EITHER 40 OR 80 COLUMNS
;SKIP THE NEXT 2 BYTES

;GET ARG OF STATEMENT

"DTCA2DOC-007-139.PICT" 97 KB 2001-03-26 dpi: 250h x 250v pix: 2013h x 2224v

```

10 ROMSOFTJ
I-RES GRAP
725 STA A1H
726 RTS
727 THIS ROUTINE
728 ARRAY BY
729 #540
730 SUBFLG
731 PTRGET
732 #500
733 SUBFLG
734 VARTIO
735 GETBYT
736
737 HTAB:
738
739 HTAB1:
740 #40
741 HTDONE
742 #40
743 CRDO
744
745 HTAB1
746 $24
747 "KRW"
748
749
750
F300:

```

29-MAR-84 08:16 PAGE 140

FAKES OUT PTRGET INTO FINDING ME AN
ONLY IT'S NAME.
;MAKE SURE BIT 6 SET.
;GO FIND ARRAY.
;MAKE NORMAL FOR NEXT TIME
;TO ALLOW SUBSCRIPTING!
;GET ARG OF STATEMENT

;CURSOR HORIZONTAL.
;MATCHES ROM JUNK

[illegible]

"DTCA2DOC-007-142.PICT" 171 KB 2001-03-26 dpi: 250h x 250v pix: 1993h x 2103v

[illegible][illegible]

29-MAR-84 08:16 PAGE 144

10 SYMBOL	SORTED BY SYMBOL	BY SYMBOL
3A0A	NEGFC	NEGFC
3A0B	NEGRTS	NEGRTS
3A0C	NEWNY1	NEWNY1
3A0D	NEWNY2	NEWNY2
3A0E	NEWNY3	NEWNY3
3A0F	NEWNY4	NEWNY4
3A10	NEWNY5	NEWNY5
3A11	NEWNY6	NEWNY6
3A12	NEWNY7	NEWNY7
3A13	NEWNY8	NEWNY8
3A14	NEWNY9	NEWNY9
3A15	NEWNY10	NEWNY10
3A16	NEWNY11	NEWNY11
3A17	NEWNY12	NEWNY12
3A18	NEWNY13	NEWNY13
3A19	NEWNY14	NEWNY14
3A1A	NEWNY15	NEWNY15
3A1B	NEWNY16	NEWNY16
3A1C	NEWNY17	NEWNY17
3A1D	NEWNY18	NEWNY18
3A1E	NEWNY19	NEWNY19
3A1F	NEWNY20	NEWNY20
3A20	NEWNY21	NEWNY21
3A21	NEWNY22	NEWNY22
3A22	NEWNY23	NEWNY23
3A23	NEWNY24	NEWNY24
3A24	NEWNY25	NEWNY25
3A25	NEWNY26	NEWNY26
3A26	NEWNY27	NEWNY27
3A27	NEWNY28	NEWNY28
3A28	NEWNY29	NEWNY29
3A29	NEWNY30	NEWNY30
3A2A	NEWNY31	NEWNY31
3A2B	NEWNY32	NEWNY32
3A2C	NEWNY33	NEWNY33
3A2D	NEWNY34	NEWNY34
3A2E	NEWNY35	NEWNY35
3A2F	NEWNY36	NEWNY36
3A30	NEWNY37	NEWNY37
3A31	NEWNY38	NEWNY38
3A32	NEWNY39	NEWNY39
3A33	NEWNY40	NEWNY40
3A34	NEWNY41	NEWNY41
3A35	NEWNY42	NEWNY42
3A36	NEWNY43	NEWNY43
3A37	NEWNY44	NEWNY44
3A38	NEWNY45	NEWNY45
3A39	NEWNY46	NEWNY46
3A3A	NEWNY47	NEWNY47
3A3B	NEWNY48	NEWNY48
3A3C	NEWNY49	NEWNY49
3A3D	NEWNY50	NEWNY50
3A3E	NEWNY51	NEWNY51
3A3F	NEWNY52	NEWNY52
3A40	NEWNY53	NEWNY53
3A41	NEWNY54	NEWNY54
3A42	NEWNY55	NEWNY55
3A43	NEWNY56	NEWNY56
3A44	NEWNY57	NEWNY57
3A45	NEWNY58	NEWNY58
3A46	NEWNY59	NEWNY59
3A47	NEWNY60	NEWNY60
3A48	NEWNY61	NEWNY61
3A49	NEWNY62	NEWNY62
3A4A	NEWNY63	NEWNY63
3A4B	NEWNY64	NEWNY64
3A4C	NEWNY65	NEWNY65
3A4D	NEWNY66	NEWNY66
3A4E	NEWNY67	NEWNY67
3A4F	NEWNY68	NEWNY68
3A50	NEWNY69	NEWNY69
3A51	NEWNY70	NEWNY70
3A52	NEWNY71	NEWNY71
3A53	NEWNY72	NEWNY72
3A54	NEWNY73	NEWNY73
3A55	NEWNY74	NEWNY74
3A56	NEWNY75	NEWNY75
3A57	NEWNY76	NEWNY76
3A58	NEWNY77	NEWNY77
3A59	NEWNY78	NEWNY78
3A5A	NEWNY79	NEWNY79
3A5B	NEWNY80	NEWNY80
3A5C	NEWNY81	NEWNY81
3A5D	NEWNY82	NEWNY82
3A5E	NEWNY83	NEWNY83
3A5F	NEWNY84	NEWNY84
3A60	NEWNY85	NEWNY85
3A61	NEWNY86	NEWNY86
3A62	NEWNY87	NEWNY87
3A63	NEWNY88	NEWNY88
3A64	NEWNY89	NEWNY89
3A65	NEWNY90	NEWNY90
3A66	NEWNY91	NEWNY91
3A67	NEWNY92	NEWNY92
3A68	NEWNY93	NEWNY93
3A69	NEWNY94	NEWNY94
3A6A	NEWNY95	NEWNY95
3A6B	NEWNY96	NEWNY96
3A6C	NEWNY97	NEWNY97
3A6D	NEWNY98	NEWNY98
3A6E	NEWNY99	NEWNY99
3A6F	NEWNY100	NEWNY100

*** SUCCESSFUL ASSEMBLY : = NO ERRORS
*** CREATED ON 00-SEP-83 15.03

** TOTAL LINES ASSEMBLED 6989
** FREE SPACE PAGE COUNT 37

Apple][Computer

AppleSoft BASIC Interpreter Source Code Listing

FIRES



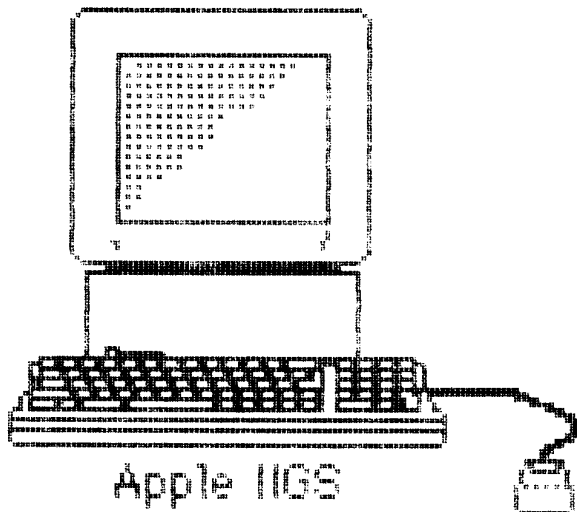
Apple][
Apple][+



Apple //e



Apple //c



Apple IIGS