```
220A: 01 99 00 02 F0 03 C8 D0 4B
2212: F5 8C 36 25 4C 39 1D 68 77
221A: 85 FB 68 85 FC 68 85 07 21
2222: 68 85 A5 68 85 A6 68 85 54
222A: A7 68 85 A8 68 85 A9 68 AC
2232: 85 AA 45 A2 85 AB A5 07 E3
223A: 0A A8 A5 FC 48 A5 FB 48 4B
2242: B9 A3 22 48 B9 A2 22 48 F9
224A: A5 9D 60 AE 3E 25 9A A9 25
2252: 07 8D 36 25 A0 00 B9 1B 2A
225A: 23 99 00 02 C8 C0 07 D0 DE
2262: F5 A9 00 99 00 02 60 00 6E
226A: 00 04 04 05 05 06 06 07 D3
2272: 07 04 04 05 05 06 06 07 5F
227A: 07 04 04 05 05 06 06 07 67
2282: 07 00 80 00 80 00 80 00 5F
228A: 80 28 A8 28 A8 28 A8 28 10
2292: A8 50 D0 50 D0 50 D0 50 18
229A: D0 00 01 02 02 03 03 04 AD
22A2: 01 22 01 22 C0 E7 A9 E7 13
22AA: 81 E9 F2 21 96 EE 4E 54 FB
22B2: 46 00 2C 40 40 41 41 41 6E
22BA: 42 41 43 41 44 41 45 41 DF
22C2: 46 41 47 41 48 41 49 41 92
22CA: 4A 41 4B 41 4C 41 4D 41 45
22D2: 4E 41 4F 41 50 41 51 41 F7
22DA: 52 41 53 41 54 41 55 41 AA
22E2: 56 41 57 41 58 41 59 41 5D
22EA: 5A 42 41 42 42 42 43 42 1D
22F2: 44 42 45 42 46 42 47 42 C2
22FA: 48 42 49 42 4A 42 4B 42 75
2302: 4C 42 4D 42 4E 42 4F 42 29
230A: 50 42 51 42 52 42 53 42 DB
2312: 54 42 55 42 56 42 57 42 8E
231A: 58 2A 45 52 52 4F 52 2A 83
2322: C5 D8 C9 D4 BA A0 C1 D2 B6
232A: C5 A0 D9 CF D5 A0 D3 D5 62
2332: D2 C5 A0 A8 D9 AF CE A9 C6
233A: BF 00 D3 D0 C5 C5 C4 C3 7A
2342: C1 CC C3 00 D3 D0 C5 C5 48
234A: C4 C3 C1 CC C3 A0 C2 D9 E8
2352: A0 CB C5 D6 C9 CE A0 CD 9A
235A: C1 D2 D4 C9 CE 00 CE C5 47
2362: D7 BA A0 C1 D2 C5 A0 D9 3C
236A: CF D5 A0 D3 D5 D2 C5 A0 85
2372: A8 D9 AF CE A9 BF 00 D7 8A
237A: C9 C4 D4 C8 BA 00 C7 CF 33
2382: D4 CF BA 00 D2 C5 C3 C1 75
238A: CC C3 D5 CC C1 D4 C9 CF 74
2392: CE A0 C9 D3 A0 CF 00 D3 F6
239A: C1 D6 C5 BA 00 CC CF C1 70
23A2: C4 BA 00 C6 CF D2 CD C1 8D
23AA: D4 BA A0 A0 CC C5 C6 D4 08
23B2: AC A0 C3 C5 CE D4 C5 D2 74
23BA: AC A0 CF D2 A0 D2 C9 C7 52
23C2: C8 D4 A0 CA D5 D3 D4 C9 D4
23CA: C6 D9 BF 00 C6 CF D2 CD CB
23D2: C1 D4 BA A0 A0 A3 A0 CF 35
23DA: C6 A0 C4 C5 C3 C9 CD C1 44
23E2: CC A0 D0 CC C1 C3 C5 D3 1B
23EA: BA 00 8D D0 D2 C5 D3 D3 76
23F2: A0 D2 C5 D4 D5 D2 CE 00 DB
23FA: D0 D2 CF C3 C5 D3 D3 C9 83
2402: CE C7 A0 C4 C1 D4 C1 A0 89
240A: D4 D2 C1 CE D3 C6 C5 D2 AE
2412: 00 CE CF D4 A0 C5 CE CF DE
241A: D5 C7 C8 A0 D2 CF CF CD A5
2422: A0 D4 CF A0 C5 CE D4 C5 CC
242A: D2 A0 C4 C1 D4 C1 00 CD 34
2432: CF D6 C5 A0 C3 D5 D2 D3 C9
243A: CF D2 A0 D4 CF A0 D4 CF FA
2442: D0 A0 CC C5 C6 D4 A0 CF AB
244A: C6 A0 CE C5 D7 A0 D0 CF 07
2452: D3 C9 D4 C9 CF CE 00 CD B5
245A: CF D6 C5 A0 C3 D5 D2 D3 F1
2462: CF D2 A0 D4 CF A0 C2 CF FE
246A: D4 D4 CF CD A0 D2 C9 C7 D4
2472: C8 D4 A0 CF C6 A0 C2 CC 70
247A: CF C3 CB 00 D0 D2 C9 CE 49
2482: D4 C9 CE C7 AE AE AE 00 8B
248A: D3 CC CF D4 A0 A3 00 D0 9B
2492: D2 C9 CE D4 A0 D4 CF BA 90
249A: A0 A0 D3 C3 D2 C5 C5 CE 1A
24A2: AC A0 C4 C9 D3 CB A0 CF 7D
24AA: D2 A0 D0 D2 C9 CE D4 C5 C4
24B2: D2 BF 00 C6 C9 CC C5 CE 9C
24BA: C1 CD C5 BA 00 CE CF A0 37
24C2: C5 D2 D2 CF D2 D3 00 D2 B2
24CA: C5 C3 C1 CC C3 D5 CC C1 BC
24D2: D4 C9 CE C7 AE AE AE 00 DB
24DA: CE CF D4 A0 C1 A0 D3 D0 2C
24E2: C5 C5 C4 C3 C1 CC C3 A0 BD
24EA: C6 C9 CC C5 00 00 00 00 FE
24F2: 00 00 00 00 00 00 00 00 3B
```

Program 2: Apple *Speed-Calc* for ProDOS

Please refer to the "MLX" article in this issue before entering the following listing.

```
START ADDRESS:  2000
END ADDRESS:    3D67

2000: 4C A7 3A 00 0A 08 0A 00 1C
2008: A5 AB 33 30 00 14 08 14 E3
2010: 00 8C 32 30 38 33 00 1E 69
2018: 08 1E 00 8C 32 30 38 30 9F
2020: 00 00 00 4C 88 22 20 58 8A
2028: FC AD 61 C0 8D C9 25 A9 12
2030: 00 8D F2 03 A9 09 8D F3 E2
2038: 03 49 A5 8D F4 03 A9 FD DE
2040: 85 39 85 37 A9 1B 85 38 B2
2048: A9 F0 85 36 A9 25 18 69 29
2050: 01 8D 60 25 18 69 4F 85 5D
2058: 6C A9 00 8D 5F 25 8D 61 1E
2060: 25 85 6B 8D D1 22 85 FF FC
2068: 8D C8 25 A9 B9 8D 62 25 CE
2070: A9 09 20 61 09 20 D9 0A 68
2078: A9 23 A0 AE 20 3E 09 20 81
2080: 88 0D 20 25 09 48 20 7C C4
2088: 09 68 AE AC 08 DD AC 08 21
2090: F0 0A CA D0 F8 C9 20 90 F1
2098: E6 4C 37 0C CA 8A 0A AA 46
20A0: A9 08 48 A9 7B 48 BD D3 A7
20A8: 08 48 BD D2 08 48 60 17 1D
20B0: 0E 00 17 06 07 10 03 13 CC
20B8: 0C 18 0A 0B 15 08 02 05 C8
20C0: 21 01 12 04 0D 1B 23 0D 7C
20C8: 31 32 33 34 35 36 37 38 01
20D0: 39 30 2B 2D 2E C4 0A DB 66
20D8: 11 13 10 A8 0C 4E 11 32 E0
20E0: 14 E6 15 9B 19 31 1A 10 13
20E8: 1F DD 10 F6 10 0D 11 37 AF
20F0: 11 CF 1C 16 1D 43 1C 3E FE
20F8: 1E E2 1C 8B 1B C1 15 08 4F
2100: 09 ED 0C 20 58 FC 20 22 DE
2108: 0B 4C 75 08 AD C8 25 49 36
2110: FF 8D C8 25 60 2C 00 C0 95
2118: 10 0B AD 00 C0 8D 10 C0 F7
2120: 29 7F C9 FF 60 A9 00 60 1A
2128: A5 FF F0 07 48 A9 00 85 3A
2130: FF 68 60 20 12 09 F0 FB 2D
2138: 60 20 F2 EB A5 A0 A4 A1 6A
2140: 60 85 FC 84 FB 20 6F 09 44
2148: 20 80 FE A9 00 85 28 85 21
2150: 24 85 25 A9 04 85 29 A0 6E
2158: 00 B1 FB F0 06 20 ED FD 20
2160: C8 D0 F6 60 A2 32 9D 66 9F
2168: 25 CA D0 FA A9 28 8D 99 5C
2170: 25 60 A0 00 A9 20 99 00 72
2178: 04 C8 C0 28 D0 F6 60 AD 5A
2180: 01 04 C9 10 D0 0A AD 0A 92
2188: 04 C9 02 F0 03 4C 94 09 0A
2190: A9 23 A0 A4 20 3E 09 38 13
2198: 20 02 20 90 03 4C 32 0F 35
21A0: 4C 40 0F 09 80 8D 80 02 C8
21A8: A9 3C 8D 81 02 A2 76 A9 C9
21B0: A0 9D 81 02 CA D0 F8 A0 27
21B8: 01 D0 02 A0 00 B9 80 02 E3
21C0: 8D AC 25 A9 DF 99 80 02 9C
21C8: 20 AB 0A 20 12 09 D0 16 B5
21D0: EE AB 25 10 08 A9 DF 99 5B
21D8: 80 02 4C C5 09 AD AC 25 3F
21E0: 99 80 02 4C C5 09 09 80 F9
21E8: 8D AB 25 AD AC 25 99 80 0A
21F0: 02 AD AB 25 AE 95 0A DD 25
21F8: 95 0A F0 2C CA D0 F8 C9 BE
2200: A0 90 BA 8C AC 25 CE AC 1D
2208: 25 A2 77 BD 80 02 C9 3C 2E
2210: F0 AB CA BD 80 02 9D 81 B5
2218: 02 CA EC AC 25 D0 F4 AD 7C
2220: AB 25 99 80 02 C8 D0 95 29
2228: CA 8A 0A AA BD 9E 0A 48 25
2230: BD 9D 0A 48 60 A0 00 B9 BF
2238: 80 02 C9 3C F0 08 29 7F B3
2240: 99 00 03 C8 D0 F1 A9 00 DF
2248: 99 00 03 8C 9C 25 60 AD 6A
2250: D2 22 F0 20 C0 00 F0 01 8F
2258: 88 4C BA 09 AD D2 22 F0 C9
2260: 13 B9 80 02 C9 3C F0 F1 DF
2268: C8 4C BA 09 AD D2 22 F0 F9
2270: 03 4C BA 09 AD AB 25 29 C0
2278: 7F 85 FF 4C 32 0A C0 00 DD
2280: F0 D7 88 98 AA BD 81 02 1F
2288: 9D 80 02 E8 C9 3C D0 F5 61
2290: A9 A0 9D 80 02 4C BA 09 4D
2298: 07 8D 9B 8A 8B 88 95 FF 89
22A0: 31 0A 7A 0A 68 0A 68 0A 36
22A8: 4B 0A 58 0A 7A 0A A2 00 02
22B0: BD 80 02 9D 80 04 BD A8 46
22B8: 02 9D 00 05 BD D0 02 9D 88
22C0: 80 05 E8 E0 28 D0 E9 60 6A
22C8: A9 23 A0 C8 20 3E 09 20 77
22D0: 25 09 C9 59 D0 03 20 D9 65
22D8: 0A 4C 7C 09 20 FA 0A A9 FF
22E0: 09 20 61 09 20 22 0B 20 2E
22E8: 88 0D A9 2C 8D 1C 23 A9 79
22F0: 00 8D 1B 23 A5 6B 85 08 1C
22F8: A5 6C 85 09 60 AD 5F 25 0A
2300: 85 FB AD 60 25 85 FC A0 9D
2308: 00 98 91 FB C8 D0 FB E6 CE
2310: FC A6 FC EC 62 25 D0 F2 29
2318: A9 01 8D 64 25 8D 65 25 BA
2320: 85 1D 85 1E 60 20 28 0B E1
2328: 4C B0 0B A0 05 8C AB 25 03
2330: B9 EB 22 85 28 B9 D3 22 DC
2338: 85 29 A0 00 AE 65 25 A9 9E
2340: 00 8D 99 25 8D 9A 25 F8 89
2348: AD 99 25 18 69 01 8D 99 F5
2350: 25 AD 9A 25 69 00 8D 9A 3B
2358: 25 CA D0 EC D8 A2 00 20 3E
2360: 8D 0B F8 AD 99 25 18 69 25
2368: 01 8D 99 25 AD 9A 25 69 A3
2370: 00 8D 9A 25 D8 EE AB 25 BE
2378: AC AB 25 B9 EB 22 85 28 5B
2380: B9 D3 22 85 29 A0 00 E8 E9
2388: E0 12 D0 D3 20 8D 0B 60 C8
2390: AD 9A 25 18 69 30 91 28 D1
2398: C8 AD 99 25 29 F0 4A 4A 20
23A0: 4A 4A 18 69 30 91 28 C8 19
23A8: AD 99 25 29 0F 18 69 30 3F
23B0: 91 28 60 A0 04 B9 EB 22 E0
23B8: 85 28 B9 D3 22 85 29 A0 5A
23C0: 00 A9 20 91 28 C8 91 28 3E
23C8: C8 91 28 C8 AE 64 25 A9 64
23D0: 00 8D 63 25 8D 66 25 8E 99
23D8: 99 25 4A 69 00 AA CA A9 FE
23E0: 20 91 28 C8 CA D0 FA AD 6A
23E8: 99 25 0A AA BD 1D 23 29 03
23F0: 3F 91 28 C8 BD 1E 23 29 A2
23F8: 3F 91 28 C8 AE 99 25 BD B8
2400: 66 25 4A AA CA CA A9 20 AD
2408: 91 28 C8 CA 10 FA AE 99 4C
2410: 25 BD 66 25 18 6D 63 25 DB
2418: 8D 63 25 E8 BD 66 25 18 1D
2420: 6D 63 25 C9 25 90 AD CA CA
2428: 8E A2 25 A9 20 C0 28 D0 C4
2430: 01 60 91 28 C8 C0 28 D0 30
2438: F9 60 20 A0 09 AD 00 03 A5
2440: F0 3F C9 3D F0 26 AE C4 20
2448: 08 DD C4 08 F0 07 CA D0 2F
2450: F8 A9 01 D0 19 AD 9C 25 8A
2458: C9 25 B0 25 A0 00 A9 03 92
2460: 20 81 0C 20 B7 00 D0 E9 E5
2468: A9 00 F0 02 A9 02 8D 9B CF
2470: 25 AD 1C 23 8D 9D 25 18 B1
2478: 20 02 20 20 62 20 20 3E 69
2480: 1C 4C 7C 08 85 B9 84 B8 CE
2488: 20 B7 00 4C 4A EC A2 32 11
2490: A9 00 8D A8 25 BD 66 25 FB
2498: 18 6D A8 25 8D A8 25 C9 D2
24A0: 25 B0 03 CA D0 EF E8 E8 B5
24A8: 8E AC 25 60 A9 00 2C C9 7D
24B0: 25 30 03 AD 61 C0 0D C8 C3
24B8: 25 8D C7 25 A0 0D A9 24 F2
24C0: 20 3E 09 20 25 09 C9 4C F8
24C8: F0 0F C9 43 F0 0F C9 52 64
24D0: F0 03 4C 85 0D A2 0C D0 10
24D8: 06 A2 08 D0 02 A2 04 AD 2B
```

```
24E0: 1C 23 29 F0 8D AB 25 8A 24
24E8: 0D AB 25 8D AB 25 4C 2F D9
24F0: 0D A9 00 2C C9 25 30 03 33
24F8: AD 61 C0 0D C8 25 8D C7 17
2500: 25 A0 36 A9 24 20 3E 09 8D
2508: 20 76 10 F0 7B A0 00 A9 19
2510: 02 20 B1 0C 20 36 09 C9 0A
2518: 00 D0 6D C0 10 B0 69 AD 14
2520: 1C 23 29 0C 8D AB 25 98 25
2528: 0A 0A 0A 0A 0D AB 25 8D CA
2530: AB 25 AD C7 25 10 41 AD 65
2538: AB 25 8D 1C 23 AD 5F 25 C8
2540: 85 1B AD 60 25 85 1C A0 E7
2548: 01 B1 1B F0 11 85 1A 88 4D
2550: B1 1B 85 19 B1 19 29 03 C3
2558: 0D 1C 23 91 19 C8 A5 1B FF
2560: 18 69 02 85 1B A5 1C 69 BA
2568: 00 85 1C A5 1C C5 6C D0 93
2570: D8 38 20 02 20 4C 85 0D A3
2578: 38 20 02 20 90 0A A0 00 17
2580: AD AB 25 0D 9B 25 91 19 AF
2588: 4C 7C 09 A5 1D 8D A0 25 19
2590: A5 1E 8D A1 25 A9 03 8D 64
2598: 63 25 AE 64 25 86 1D AC 24
25A0: 65 25 84 1E 98 18 69 13 64
25A8: 8D 9E 25 BD 66 25 8D A8 6D
25B0: 25 A9 FF EC A0 25 D0 07 09
25B8: CC A1 25 D0 02 A9 3F 8D 46
25C0: A3 25 98 18 69 05 38 ED 78
25C8: 65 25 A8 B9 D3 22 85 29 1B
25D0: B9 EB 22 85 28 38 20 02 F3
25D8: 20 B0 05 A9 A0 4C 67 0E AD
25E0: AD 9B 25 F0 70 C9 02 F0 3C
25E8: 6C AD A8 25 38 ED 9C 25 14
25F0: AA E8 30 32 E8 AD 9D 25 52
25F8: 29 0C C9 08 F0 28 B0 05 23
2600: 8A 4A F0 22 AA 8E A4 25 62
2608: A9 A0 2D A3 25 AC 63 25 F8
2610: 91 28 C8 CA D0 FA 8C A5 26
2618: 25 AD A8 25 38 ED A4 25 B1
2620: AA A0 02 4C 2E 0E AE A8 9E
2628: 25 AD 63 25 8D A5 25 A0 1F
2630: 02 B1 19 8C A4 25 AC A5 8E
2638: 25 09 80 2D A3 25 91 28 39
2640: AC A4 25 EE A5 25 CA F0 E7
2648: 09 C8 CC 9C 25 D0 E2 20 01
2650: A9 0E 4C 76 0E 20 4E 0F 82
2658: AE 9C 25 CA CA CA EC AB 78
2660: 25 B0 03 4C E6 0D A9 2A 79
2668: 09 80 2D A3 25 AC 63 25 01
2670: AE A8 25 91 28 C8 CA D0 C6
2678: FA A4 1E A6 1D C8 CC 9E DD
2680: 25 F0 05 84 1E 4C A8 0D 05
2688: AC 65 25 84 1E AD A8 25 8F
2690: 18 6D 63 25 8D 63 25 E8 30
2698: 86 1D E0 33 F0 27 BD 66 C4
26A0: 25 18 6D 63 25 C9 28 B0 BA
26A8: 1C 4C A8 0D E0 00 F0 14 F8
26B0: AD 63 25 18 6D A8 25 A8 D3
26B8: 88 A9 A0 2D A3 25 91 28 97
26C0: 88 CA D0 FA 60 A9 28 38 FF
26C8: ED 63 25 8D A8 25 A0 05 82
26D0: 84 1E B9 D3 22 85 29 B9 8E
26D8: EB 22 85 28 AC 63 25 AE C2
26E0: A8 25 A9 A0 91 28 C8 CA 93
26E8: D0 FA E6 1E A4 1E C0 18 52
26F0: D0 E0 AD A0 25 85 1D AD C4
26F8: A1 25 85 1E A0 00 A9 A0 EA
2700: 99 80 02 C8 C0 78 D0 F8 8A
2708: 38 20 02 20 90 35 A0 02 59
2710: A2 00 AD 9B 25 C9 02 D0 44
2718: 09 AC 9C 25 B1 19 8D 9C A5
2720: 25 C8 B1 19 09 80 9D 80 01
2728: 02 E8 C8 CC 9C 25 D0 F2 A5
2730: A9 3C 9D 80 02 AE 9B 25 45
2738: BD 18 23 29 3F 8D 27 04 E4
2740: 4C AB 0A A9 20 8D 27 04 05
2748: A9 3C 8D 80 02 20 AB 0A 26
2750: 60 A9 20 8D 00 02 A0 02 61
2758: B1 19 C9 2A F0 F2 AD 9D ED
2760: 25 4A 4A 4A 4A 8D A6 25 BC
2768: A2 FF C9 0F F0 E2 B1 19 C1
2770: C9 2E D0 09 AE A6 25 F0 25
2778: 10 E8 8E 00 02 99 FF 01 52
2780: C8 CC 9C 25 F0 03 CA D0 46
2788: E5 AD A6 25 F0 1E E0 00 1E
2790: F0 1A AD 00 02 C9 20 D0 DB
2798: 0A A9 2E 99 FF 01 C8 AE F9
27A0: A6 25 E8 A9 30 99 FF 01 2C
27A8: C8 CA D0 F9 A9 20 8D 00 B0
27B0: 02 CC 9C 25 F0 0C B0 3F 71
27B8: B1 19 C9 2E F0 08 C9 35 B2
27C0: B0 0C C8 4C F4 0F C8 B1 6F
27C8: 19 C9 35 90 2A 88 98 C8 33
27D0: AA CA CA BD 00 02 C9 2E 26
27D8: F0 0B 90 0C C9 39 D0 14 1E
27E0: A9 30 9D 00 02 CA 10 EB 0B
27E8: CA 9D 00 02 E8 A9 31 9D 12
27F0: 00 02 D0 03 FE 00 02 88 8E
27F8: 8C 9C 25 AD 00 02 C9 20 EF
2800: D0 09 A9 01 85 1A A9 FF 28
2808: 85 19 60 A9 01 85 1A A9 04
2810: FE 85 19 EE 9C 25 60 A9 37
2818: 00 2C C9 25 30 03 AD 61 49
2820: C0 0D C8 25 8D C7 25 A9 FE
2828: 23 A0 E1 20 3E 09 20 76 3D
2830: 10 A0 00 A9 02 20 81 0C EA
2838: 20 36 09 C9 00 D0 33 C0 4E
2840: 04 90 2F C0 25 B0 2B A5 90
2848: 1D 8D 64 25 AD C7 25 10 50
2850: 07 98 20 61 09 4C 5B 10 A4
2858: 98 A6 1D 9D 66 25 20 8B AF
2860: 0C A5 1D CD AC 25 90 07 C2
2868: AC AC 25 88 8C 64 25 20 C7
2870: B0 0B 4C 7C 09 A9 01 D0 EE
2878: 02 A9 00 8D A7 25 A0 00 20
2880: A9 1F 20 ED FD A9 88 20 18
2888: ED FD 20 25 09 C9 0D F0 20
2890: 3F C9 08 F0 26 C9 7F F0 4B
2898: 22 C9 20 90 ED AE A7 25 18
28A0: D0 0B C9 30 90 E4 C9 3A 7D
28A8: B0 E0 A6 24 E0 26 F0 DA FC
28B0: 99 00 02 09 80 20 ED FD FC
28B8: C8 D0 C5 C0 00 F0 CB A9 6B
28C0: A0 20 ED FD A9 88 20 ED A4
28C8: FD 20 ED FD 88 4C 7D 10 3E
28D0: A9 A0 20 ED FD A9 00 99 31
28D8: 00 02 8C A6 25 AD 00 02 87
28E0: 60 A5 1E C9 C8 F0 12 E6 40
28E8: 1E AD 65 25 18 69 12 C5 03
28F0: 1E B0 06 EE 65 25 20 28 54
28F8: 0B 60 A5 1E C9 01 F0 10 C1
2900: C6 1E AC 65 25 88 C4 1E 1C
2908: 90 06 CE 65 25 20 28 0B 59
2910: 60 A5 1D C9 32 F0 23 E6 BE
2918: 1D AC A2 25 C4 1D B0 1A E0
2920: EE 64 25 AE 64 25 A9 00 9D
2928: 18 7D 66 25 E8 C9 25 90 4E
2930: F7 CA CA E4 1D 90 E9 20 F7
2938: B0 0B 60 A5 1D C9 01 F0 0F
2940: 10 C6 1D AC 64 25 88 C4 48
2948: 1D 90 06 CE 64 25 20 B0 A3
2950: 0B 60 A9 23 A0 E8 20 3E CE
2958: 09 20 72 10 A9 01 85 B9 9C
2960: A9 FF 85 B8 20 B1 00 90 1C
2968: 4E 38 E9 41 30 49 F0 06 CF
2970: C9 02 B0 43 A9 1A 8D AB EE
2978: 25 20 B1 00 90 39 38 E9 5F
2980: 40 30 34 F0 32 C9 1B B0 34
2988: 2E 18 6D AB 25 C9 33 B0 C7
2990: 26 8D AB 25 20 B1 00 B0 99
2998: 1E 20 4A EC 20 36 09 C9 CF
29A0: 00 D0 14 C0 00 F0 10 C0 5A
29A8: C9 B0 0C C0 B7 90 0B A9 59
29B0: B6 8D 65 25 4C BA 11 4C 7C
29B8: 7C 09 8C 65 25 84 1E 20 0B
29C0: 8B 0C AD AB 25 CD AC 25 2B
29C8: 90 0A AC AC 25 88 8C 64 0F
29D0: 25 4C D4 11 8D 64 25 85 42
29D8: 1D 20 22 0B 4C 7C 09 AD CA
29E0: 64 25 C5 1D D0 17 AD 65 DC
29E8: 25 C5 1E D0 10 A9 01 8D C6
29F0: 64 25 85 1D 8D 65 25 85 13
29F8: 1E 20 22 0B 60 AD 64 25 FE
2A00: 85 1D AD 65 25 85 1E 60 46
2A08: 20 B1 00 8D BF 25 20 B1 36
2A10: 00 8D C0 25 20 B1 00 8D 87
2A18: C1 25 20 B1 00 C9 28 F0 1E
2A20: 03 4C 88 22 AE 6A 12 AD 2D
2A28: BF 25 DD 6A 12 F0 06 CA 33
2A30: D0 F5 4C 88 22 AD C0 25 EA
2A38: DD 76 12 F0 02 D0 F0 AD 4D
2A40: C1 25 DD 82 12 D0 E8 8E D6
2A48: 99 25 E0 0B B0 0C 8A 48 92
2A50: A9 00 48 4C 5D 21 68 8D 15
2A58: 99 25 20 B1 00 AE 99 25 F4
2A60: CA 8A 0A AA BD 90 12 48 45
2A68: BD 8F 12 48 60 0C 41 41 3D
2A70: 43 45 49 4C 53 53 53 54 88
2A78: 53 41 42 54 4F 58 4E 4F 1C
2A80: 47 49 51 41 55 56 53 4E 02
2A88: 53 50 54 47 4E 4E 52 4E 38
2A90: 4D 45 AE EB 9D F0 E9 EF E5
2A98: 08 EF 22 EC 40 E9 8F EB B4
2AA0: F0 EF 8C EE 39 F0 A9 13 DD
2AA8: 11 14 20 64 13 8E C2 25 52
2AB0: 8C C4 25 20 B7 00 C9 3A AE
2AB8: D0 3F 20 B1 00 20 64 13 C0
2AC0: 8E C3 25 8C C5 25 20 B7 75
2AC8: 00 C9 29 D0 2C 20 B1 00 07
2AD0: AE C2 25 CA EC C3 25 90 CF
2AD8: 03 4C 88 22 AC C4 25 88 40
2AE0: CC C5 25 90 03 4C 88 22 37
2AE8: E8 C8 A5 1D 8D A9 25 A5 6D
2AF0: 1E 8D AA 25 86 1D 84 1E 2F
2AF8: 60 4C 88 22 18 20 02 20 29
2B00: 90 42 A0 00 B1 19 29 03 8A
2B08: C9 01 F0 38 C8 B1 19 8D F1
2B10: AC 25 A2 00 C8 B1 19 9D 37
2B18: 00 02 E8 C8 CC AC 25 D0 CC
2B20: F4 A5 B8 48 A5 B9 48 A9 44
2B28: 00 9D 00 02 A9 02 A0 00 9C
2B30: 20 B1 0C 68 85 B9 68 85 68
2B38: B8 A5 1D CD C3 25 F0 15 7E
2B40: E6 1D 18 60 AD A9 25 85 3E
2B48: 1D AD AA 25 85 1E 18 20 35
2B50: 02 20 4C 88 22 AD C2 25 34
2B58: 85 1D A5 1E CD C5 25 F0 10
2B60: 04 E6 1E 18 60 38 60 A2 FE
2B68: 00 20 B7 00 C9 41 F0 06 F8
2B70: C9 42 D0 D0 A2 1A 8E AB A9
2B78: 25 20 B1 00 C9 41 90 C4 D8
2B80: C9 5B B0 C0 38 E9 40 18 B6
2B88: 6D AB 25 C9 33 B0 B5 8D 17
2B90: AB 25 20 B1 00 B0 AD 20 63
2B98: 4A EC 20 36 09 C9 00 D0 F6
2BA0: A3 C0 00 F0 9F C0 C9 B0 4C
2BA8: 9B AE AB 25 60 A9 01 8D 79
2BB0: 99 25 A9 00 8D 9A 25 20 93
2BB8: A7 12 20 F9 12 B0 47 20 0D
2BC0: 72 EB A5 A2 48 A5 A1 48 8E
2BC8: A5 A0 48 A5 9F 48 A5 9E 85
2BD0: 48 A5 9D 48 EE 99 25 D0 E5
2BD8: 03 EE 9A 25 20 F9 12 08 27
2BE0: 68 8D AB 25 68 85 A5 68 A3
2BE8: 85 A6 68 85 A7 68 85 A8 A3
2BF0: 68 85 A9 68 85 AA 45 A2 9C
2BF8: 85 AB A5 9D 20 C1 E7 AD 11
2C00: AB 25 48 28 90 B9 AD A9 73
2C08: 25 85 1D AD AA 25 85 1E E5
2C10: 18 20 02 20 60 20 AA 13 AA
2C18: A2 06 B5 9C 95 A4 CA D0 69
2C20: F9 AD 9A 25 AC 99 25 20 BC
2C28: F2 E2 A5 AA 45 A2 85 AB 7D
2C30: A5 9D 20 2E 22 60 20 58 D4
2C38: FC A9 01 8D C6 25 A9 00 90
2C40: 2C C9 25 30 03 AD 61 C0 1B
2C48: 0D C8 25 30 03 4C A2 14 A3
2C50: A9 24 A0 F9 20 3E 09 20 66
2C58: 25 09 C9 53 F0 0B C9 44 7F
2C60: F0 0E C9 50 F0 21 4C AB 43
2C68: 15 A9 03 8D C6 25 D0 35 90
2C70: A9 00 8D C6 25 A0 1D A9 4B
2C78: 25 20 3E 09 20 72 10 20 CE
2C80: E6 1A 90 21 4C 9E 15 A9 A7
2C88: 24 A0 F2 20 3E 09 20 25 F6
2C90: 09 38 E9 30 C9 00 B0 03 6E
2C98: 4C AB 15 C9 08 90 03 4C 16
2CA0: AB 15 8D C6 25 A9 24 A0 EA
2CA8: E6 20 3E 09 20 84 FE AD 93
2CB0: C6 25 F0 14 C9 03 D0 0D 1E
2CB8: AD 05 C3 18 6D 07 C3 C9 FB
2CC0: 50 D0 05 A9 03 20 95 FE 73
2CC8: A5 1D 8D C3 25 8D A0 25 EE
2CD0: A5 1E 8D C5 25 8D A1 25 59
2CD8: A9 01 85 1D 85 1E A9 8D 4E
2CE0: 20 B4 15 A6 1D BD 66 25 55
2CE8: 8D A8 25 AA A9 00 9D 00 0A
2CF0: 03 CA A9 20 9D 00 03 CA 72
2CF8: 10 FA 38 20 02 20 90 58 2B
2D00: AD 9B 25 C9 01 D0 23 AD 98
2D08: A8 25 38 ED 9C 25 AA E8 9D
2D10: 30 14 E8 AD 9D 25 29 0C 5F
```

```
2D18: C9 08 F0 0A B0 27 8A 4A 99
2D20: F0 04 AA 4C 42 15 A2 00 B9
2D28: F0 1B 20 4E 0F AE 9C 25 3C
2D30: CA CA CA EC A8 25 90 CF 95
2D38: AE A8 25 A9 2A 9D FF 02 1D
2D40: CA D0 FA F0 13 A0 02 B1 73
2D48: 19 9D 00 03 E8 C8 EC A8 B3
2D50: 25 F0 05 CC 9C 25 D0 EF F1
2D58: A2 00 BD 00 03 F0 08 09 B0
2D60: 80 20 B4 15 E8 D0 F3 A5 03
2D68: 1D CD C3 25 F0 05 E6 1D 16
2D70: 4C E0 14 A5 1E CD C5 25 DE
2D78: F0 0E E6 1E A9 01 85 1D 07
2D80: A9 8D 20 B4 15 4C E0 14 12
2D88: A9 8D 20 B4 15 AD C6 25 7C
2D90: C9 03 D0 03 20 25 09 A9 2C
2D98: 00 20 95 FE AD C6 25 D0 41
2DA0: 03 20 17 1B AD A0 25 85 D8
2DA8: 1D AD A1 25 85 1E 20 58 C0
2DB0: FC 20 22 0B 4C 7C 09 48 35
2DB8: AD C6 25 F0 04 68 4C ED 97
2DC0: FD 68 4C 2B 1B A9 00 2C 1C
2DC8: C9 25 30 03 AD 61 C0 0D 09
2DD0: C8 25 8D AF 25 A9 00 8D E2
2DD8: B0 25 A5 1D 8D B1 25 A5 7E
2DE0: 1E 8D B2 25 4C 06 16 4C 49
2DE8: 7C 09 A9 00 2C C9 25 30 FB
2DF0: 03 AD 61 C0 0D C8 25 8D D3
2DF8: AF 25 A9 01 8D B0 25 A5 D8
2E00: 1D 8D B1 25 A5 1E 8D B2 4A
2E08: 25 20 41 16 AD A0 25 8D 50
2E10: B5 25 AD A1 25 8D B6 25 52
2E18: 20 4B 16 AE B1 25 CA EC A9
2E20: B5 25 B0 13 AE B2 25 CA 3D
2E28: EC B6 25 B0 0A A9 24 A0 38
2E30: 62 20 3E 09 20 3D 18 AD F1
2E38: B3 25 85 1D AD B4 25 85 4A
2E40: 1E 4C 7C 09 A9 24 A0 C1 BF
2E48: 20 3E 09 4C 52 16 A9 24 8C
2E50: A0 99 20 3E 09 20 88 0D 32
2E58: 20 25 09 AE 86 16 DD 86 E8
2E60: 16 F0 06 CA D0 F8 4C 52 C6
2E68: 16 CA 8A 0A AA A9 16 48 E4
2E70: A9 51 48 BD 8E 16 48 BD F5
2E78: 8D 16 48 60 68 68 A5 1D 7D
2E80: 8D B3 25 A5 1E 8D B4 25 45
2E88: 60 06 00 0B 0A 08 15 0D EE
2E90: DB 11 F6 10 DD 10 37 11 AD
2E98: 0D 11 78 16 AD B9 25 C9 98
2EA0: 33 B0 5B AD BA 25 C9 C9 D0
2EA8: B0 54 AD B7 25 85 1D AD CA
2EB0: B8 25 85 1E 38 20 02 20 AB
2EB8: 90 45 A0 02 AD 9B 25 C9 D2
2EC0: 02 D0 09 AC 9C 25 B1 19 34
2EC8: 8D 9C 25 C8 A2 00 B1 19 D5
2ED0: 9D 00 03 E8 C8 CC 9C 25 C2
2ED8: D0 F4 A9 00 9D 00 03 8E 91
2EE0: 9C 25 20 1E 17 AD B0 25 B0
2EE8: D0 03 20 0C 17 AD B9 25 3B
2EF0: 85 1D AD BA 25 85 1E 18 4C
2EF8: 20 02 20 20 62 20 60 AD ED
2F00: B9 25 85 1D AD BA 25 85 2F
2F08: 1E 18 20 02 20 90 EF 20 E2
2F10: 6E 1E 18 20 02 20 A9 00 16
2F18: A8 91 1B C8 91 1B 4C FB AC
2F20: 16 AD AF 25 30 01 60 AD 31
2F28: 9B 25 C9 02 F0 01 60 AD F0
2F30: B9 25 38 ED B7 25 8D BD C5
2F38: 25 AD BA 25 38 ED B8 25 4E
2F40: 8D BE 25 A2 00 8E 9A 25 78
2F48: BD 00 03 9D 80 02 E8 EC 8A
2F50: 9C 25 D0 F4 A9 00 9D 80 B8
2F58: 02 A9 80 85 B8 A9 02 85 80
2F60: B9 A9 00 85 FB A9 03 85 70
2F68: FC 20 B7 00 20 30 18 20 56
2F70: B1 00 C9 00 D0 03 4C 25 31
2F78: 18 C9 40 D0 03 4C 10 18 EB
2F80: 90 EA C9 43 B0 E6 A2 00 B5
2F88: C9 42 D0 02 A2 1A 8E 99 CA
2F90: 25 20 B1 00 C9 41 90 66 9A
2F98: C9 5B B0 62 38 E9 40 18 F0
2FA0: 6D 99 25 C9 33 B0 57 18 80
2FA8: 6D BD 25 A2 41 C9 1B 90 F3
2FB0: 05 A2 42 38 E9 1A 18 69 57
2FB8: 40 8D 99 25 8A 20 30 18 6D
2FC0: AD 99 25 20 30 18 20 B1 D6
2FC8: 00 B0 33 20 4A EC 20 36 38
2FD0: 09 C9 00 D0 29 C0 00 F0 70
2FD8: 25 C0 C9 B0 21 98 18 6D 47
2FE0: BE 25 A8 A9 00 20 F2 E2 E0
2FE8: 20 34 ED A2 00 BD 00 01 44
2FF0: F0 06 20 30 18 E8 D0 F5 4C
2FF8: 20 B7 00 4C 6F 17 A2 00 37
3000: BD B0 02 F0 06 9D 00 03 58
3008: E8 D0 F5 A9 00 9D 00 03 E3
3010: 4C 2F 18 20 30 18 20 B1 3B
3018: 00 20 30 18 20 B1 00 20 EF
3020: 30 18 20 B1 00 4C 69 17 D8
3028: AC 9A 25 8C 9C 25 A9 00 BF
3030: 91 FB 60 AC 9A 25 C0 78 92
3038: F0 05 91 FB EE 9A 25 60 D0
3040: AD B5 25 38 ED B1 25 18 A5
3048: 6D A0 25 8D BB 25 AD B6 89
3050: 25 38 ED B2 25 18 6D A1 40
3058: 25 8D BC 25 AD B2 25 CD E8
3060: A1 25 B0 03 4C 00 19 AD 63
3068: B1 25 CD A0 25 90 4A AD 5C
3070: B1 25 8D B7 25 AD B2 25 8A
3078: 8D B8 25 AD A0 25 8D B9 BB
3080: 25 AD A1 25 8D BA 25 20 27
3088: 99 16 AD B7 25 CD B5 25 5D
3090: F0 08 EE B7 25 EE B9 25 42
3098: D0 ED AD B8 25 CD B6 25 11
30A0: F0 14 EE B8 25 EE BA 25 67
30A8: AD B1 25 8D B7 25 AD A0 18
30B0: 25 8D B9 25 D0 D1 4C 99 90
30B8: 19 AD B5 25 8D B7 25 AD 5D
30C0: BB 25 8D B9 25 AD B2 25 FF
30C8: 8D B8 25 AD A1 25 8D BA 15
30D0: 25 20 99 16 AD B7 25 CD C4
30D8: B1 25 F0 08 CE B7 25 CE 68
30E0: B9 25 D0 ED AD B8 25 CD C8
30E8: B6 25 F0 CA EE B8 25 EE 4C
30F0: BA 25 AD B5 25 8D B7 25 FC
30F8: AD BB 25 8D B9 25 D0 D1 72
3100: 4C 99 19 AD B1 25 CD A0 4B
3108: 25 90 4A AD B1 25 8D B7 3A
3110: 25 AD B6 25 8D B8 25 AD E0
3118: A0 25 8D B9 25 AD BC 25 DF
3120: 8D BA 25 20 99 16 AD B7 D6
3128: 25 CD B5 25 F0 08 EE B7 D6
3130: 25 EE B9 25 D0 ED AD B8 BC
3138: 25 CD B2 25 F0 14 CE B8 77
3140: 25 CE BA 25 AD B1 25 8D 9E
3148: B7 25 AD A0 25 8D B9 25 87
3150: D0 D1 4C 99 19 AD B5 25 C2
3158: 8D B7 25 AD BB 25 8D B9 36
3160: 25 AD B6 25 8D B8 25 AD 31
3168: BC 25 8D BA 25 20 99 16 C2
3170: AD B7 25 CD B1 25 F0 08 25
3178: CE B7 25 CE B9 25 D0 ED B3
3180: AD B8 25 CD B2 25 F0 14 89
3188: CE B8 25 CE BA 25 AD B5 8D
3190: 25 8D B7 25 AD BB 25 8D 66
3198: B9 25 D0 D1 4C 44 1C A9 AD
31A0: 24 A0 01 20 3E 09 20 72 28
31A8: 10 D0 03 4C 7C 09 20 E6 9B
31B0: 1A 90 03 4C 40 1B A9 FF 2B
31B8: 20 2B 1B A9 FF 20 2B 1B E5
31C0: A5 08 20 2B 1B A5 09 20 50
31C8: 2B 1B A0 32 B9 66 25 20 90
31D0: 2B 1B 88 D0 F7 AD 5F 25 08
31D8: 85 1B AD 60 25 85 1C A0 98
31E0: 01 B1 1B F0 16 A5 1B 20 40
31E8: 2B 1B A5 1C 20 2B 1B 88 8A
31F0: B1 1B 20 2B 1B C8 B1 1B 24
31F8: 20 2B 1B A5 1B 18 69 02 02
3200: 85 1B A5 1C 69 00 85 1C D6
3208: A5 1C C5 6C D0 D1 A9 FF E6
3210: 20 2B 1B A5 6B 85 1B A5 5A
3218: 6C 85 1C A0 00 B1 1B 20 BE
3220: 2B 1B C8 D0 F8 E6 1C A5 48
3228: 1C C5 09 90 F0 F0 EE 20 7F
3230: 17 1B 4C BB 1A A9 24 A0 8C
3238: 07 20 3E 09 20 72 10 D0 3C
3240: 03 4C 7C 09 20 FA 1A 90 0B
3248: 03 4C 40 1B 20 1E 1B C9 74
3250: FF D0 60 20 1E 1B C9 FF E7
3258: D0 59 20 FA 0A 20 1E 1B 57
3260: 85 08 20 1E 1B 85 09 A0 11
3268: 32 20 1E 1B 99 66 25 88 9C
3270: D0 F7 20 1E 1B C9 FF F0 12
3278: 18 85 1B 20 1E 1B 85 1C 34
3280: 20 1E 1B A0 00 91 1B 20 86
3288: 1E 1B A0 01 91 1B 4C 6F E7
3290: 1A A5 6B 85 1B A5 6C 85 FE
3298: 1C A0 00 20 1E 1B 91 1B D0
32A0: C8 D0 F8 E6 1C A5 1C C5 A0
32A8: 09 90 F0 F0 EE 20 17 1B 24
32B0: 4C BB 1A 20 17 1B A9 25 0D
32B8: A0 4A 4C 3E 09 60 B0 08 A0
32C0: A9 25 A0 27 20 3E 09 60 36
32C8: 8D 00 02 20 6F 09 A9 00 29
32D0: 85 24 85 28 A9 04 85 29 C5
32D8: 20 80 FE A9 25 A0 31 20 16
32E0: 3E 09 AD 00 02 20 DA FD A0
32E8: 60 20 04 1B 20 00 BF C1 F9
32F0: 62 1B 20 00 BF C0 65 1B 38
32F8: B0 0C 4C FD 1A 20 04 1B 96
3300: 20 00 BF C8 71 1B 60 AC 60
3308: A6 25 B9 00 02 99 01 02 BC
3310: 88 10 F7 AD A6 25 8D 00 7D
3318: 02 60 20 00 BF CC 7F 1B E6
3320: 60 98 48 8A 48 20 00 BF 11
3328: CA 81 1B 4C 38 1B 8D 00 C5
3330: 02 98 48 8A 48 20 00 BF F1
3338: CB 77 1B 90 0F AA 68 68 2B
3340: 68 68 8A 8D 00 02 20 17 7E
3348: 1B 4C C8 1A 68 AA 68 A8 71
3350: AD 00 02 60 02 60 00 BE 24
3358: 01 00 02 04 01 00 B9 00 3B
3360: 02 00 00 01 01 01 00 02 E5
3368: 07 00 02 C3 04 00 00 01 EF
3370: 00 00 00 00 03 00 02 00 F2
3378: BB 00 04 01 00 02 01 00 57
3380: 00 00 01 01 04 01 00 02 3D
3388: 01 00 00 00 4C C5 1A 20 3D
3390: 58 FC 20 84 FE 20 00 BF E6
3398: C5 51 1B B0 EF A0 00 AD 54
33A0: 00 BE 29 0F 8D 00 02 B9 F6
33A8: 01 BE 99 02 02 C8 CC 00 5F
33B0: 02 D0 F4 C8 8C 00 02 A9 89
33B8: 2F 8D 01 02 20 00 BF C6 A1
33C0: 55 1B B0 C8 20 00 BF C8 84
33C8: 71 1B B0 C0 A9 AF 20 ED 0B
33D0: FD 20 00 BF CA 58 1B B0 D8
33D8: B3 A9 B9 85 FC A9 04 85 2F
33E0: FB A0 00 B1 FB D0 08 C8 84
33E8: B1 FB F0 34 4C 09 1C 8D D4
33F0: 99 25 29 0F AA E8 8E A6 40
33F8: 25 C8 B1 FB 09 80 20 ED 92
3400: FD C8 CC A6 25 D0 F3 A9 9B
3408: 8D 20 ED FD A9 27 18 65 5C
3410: FB 85 FB A5 FC 69 00 85 C4
3418: FC C9 BB F0 B4 4C DE 1B A7
3420: 20 00 BF CC 60 1B A9 24 44
3428: 85 FC A9 54 85 FB 20 54 BD
3430: 09 20 12 09 C9 0D D0 F9 16
3438: 20 58 FC 20 22 0B 4C 7C BA
3440: 09 AD 1B 23 D0 01 60 A9 23
3448: 25 A0 39 20 3E 09 A5 1D 13
3450: 8D A0 25 A5 1E 8D A1 25 36
3458: A9 01 85 1D 85 1E AD 5F B7
3460: 25 85 1B AD 60 25 85 1C B9
3468: A0 01 B1 1B F0 35 85 1A CA
3470: 88 B1 1B 85 19 B1 19 29 30
3478: 03 C9 02 D0 26 38 20 02 76
3480: 20 A2 00 AC 9C 25 B1 19 62
3488: 8D 9C 25 C8 B1 19 9D 00 3D
3490: 03 E8 C8 CC 9C 25 D0 F4 AA
3498: A9 00 9D 00 03 8E 9C 25 3A
34A0: 20 62 20 A5 1B 18 69 02 1E
34A8: 85 1B 90 02 E6 1C E6 1E 60
34B0: A5 1E C9 C9 D0 B2 A9 01 EE
34B8: 85 1E E6 1D A5 1D C9 33 82
34C0: D0 A6 AD A0 25 85 1D AD 22
34C8: A1 25 85 1E 38 20 02 20 44
34D0: 4C 7C 09 20 6E 1E 18 20 DD
34D8: 02 20 A9 00 A8 91 1B C8 0A
34E0: 91 1B 20 3E 1C 60 A9 23 99
34E8: A0 EE 20 3E 09 A9 00 2C 60
34F0: C9 25 30 03 AD 61 C0 0D 3F
34F8: C8 25 30 08 AD 1B 23 49 FE
3500: FF 8D 1B 23 AD 1B 23 C9 4D
3508: 00 F0 06 A9 CE 20 ED FD DA
3510: 60 A9 C6 20 ED FD 20 ED 85
3518: FD 60 EE D2 22 20 B8 09 B0
3520: CE D2 22 AD 00 03 F0 4E 02
3528: C9 3D F0 27 AE C4 08 DD CD
3530: C4 08 F0 08 CA D0 F8 A9 D2
3538: 01 4C 52 1D AD 9C 25 C9 46
```

```
3540: 25 B0 33 A0 00 A9 03 20 A6
3548: 81 0C 20 B7 00 D0 E8 A9 B4
3550: 00 F0 02 A9 02 8D 9B 25 74
3558: 18 20 02 20 B0 09 AD 1C 3A
3560: 23 8D 9D 25 4C 6D 1D A0 B8
3568: 00 B1 19 29 FC 8D 9D 25 73
3570: 20 62 20 20 3E 1C 60 AE 5B
3578: A6 25 CA CA CA CA B0 00 82
3580: 02 C9 45 D0 78 E8 BD 00 F6
3588: 02 8D AB 25 E8 BD 00 02 5F
3590: 38 E9 30 8D 9A 25 E8 BD 69
3598: 00 02 38 E9 30 AE 9A 25 BF
35A0: F0 06 18 69 0A CA D0 FA B6
35A8: 8D 99 25 AD AB 25 C9 2D 72
35B0: F0 4C A2 00 A0 00 BD 00 7B
35B8: 02 C9 45 F0 08 E8 C9 2E F3
35C0: F0 F4 C8 D0 F1 88 8C AB 7D
35C8: 25 AD 99 25 38 ED AB 25 AC
35D0: 8D 99 25 A2 01 A0 01 BD 81
35D8: 00 02 E8 C9 2E F0 F8 C9 6E
35E0: 45 F0 06 99 00 02 C8 D0 EE
35E8: EE A9 30 AE 99 25 99 00 BA
35F0: 02 C8 CA D0 F9 A9 00 99 05
35F8: 00 02 8C A6 25 60 CE 99 C1
3600: 25 A2 00 A0 00 BD 00 02 AA
3608: E8 C9 2E F0 F8 C9 45 F0 9A
3610: 06 99 80 02 C8 D0 EE A9 27
3618: 00 99 80 02 A9 2E 8D 00 3C
3620: 02 AE 99 25 A9 30 9D 00 08
3628: 02 CA D0 FA A2 00 AC 99 1A
3630: 25 C8 BD 80 02 99 00 02 99
3638: F0 04 E8 C8 D0 F4 8C A6 E1
3640: 25 60 20 6F 09 A9 04 85 CE
3648: 29 A9 00 85 28 85 24 AD 59
3650: 61 25 38 E5 08 AB AD 62 BC
3658: 25 E5 09 20 5C 1E 60 20 30
3660: F2 E2 20 34 ED A9 01 85 E3
3668: FC A9 00 85 FB 20 54 09 28
3670: 60 A0 01 B1 1B F0 E7 A9 86
3678: 00 91 1B 88 91 1B B1 19 AA
3680: 29 03 C9 02 D0 09 C8 B1 89
3688: 19 A8 B1 19 4C 8F 1E C8 19
3690: B1 19 85 FB 18 65 19 8D A2
3698: B1 1E A5 19 8D B4 1E A5 CC
36A0: 1A 8D B5 1E 69 00 8D B2 2F
36A8: 1E A5 09 38 ED B2 1E AA 53
36B0: E8 A0 00 B9 FF FF 99 FF 88
36B8: FF C8 D0 F7 EE B2 1E EE 5E
36C0: B5 1E CA D0 EE A5 08 38 4C
36C8: E5 FB 85 08 A5 09 E9 00 7D
36D0: 85 09 AD 5F 25 85 FD AD D6
36D8: 60 25 85 FE A0 01 B1 FD C9
36E0: F0 22 38 88 B1 FD E5 19 48
36E8: 8D 99 25 C8 B1 FD E5 1A 1F
36F0: 0D 99 25 90 0F 88 B1 FD F3
36F8: 38 E5 FB 91 FD C8 B1 FD 08
3700: E9 00 91 FD C8 F0 03 C8 4E
3708: D0 D4 E6 FE C8 A5 FE C5 81
3710: 6C D0 CB 60 A9 23 A0 8A 0E
3718: 20 3E 09 20 25 09 C9 59 83
3720: D0 03 4C 00 C6 4C 7C 09 AA
3728: AD A9 25 85 1D AD AA 25 EE
3730: 85 1E 18 20 02 20 AD AB 85
3738: 25 8D 9B 25 AD AD 25 8D 5E
3740: 9D 25 AD AC 25 8D 9C 25 05
3748: 4C 88 22 48 A5 1D 8D A9 2E
3750: 25 A5 1E 8D AA 25 AD 9B 38
3758: 25 8D AB 25 AD 9D 25 8D 40
3760: AD 25 AD 9C 25 8D AC 25 4C
3768: 68 E9 41 30 BB F0 06 C9 28
3770: 02 B0 B5 A9 1A 85 1D 20 9E
3778: B1 00 E9 40 30 AA F0 A8 B7
3780: C9 1B B0 A4 18 65 1D C9 55
3788: 33 B0 9D 85 1D 20 B1 00 95
3790: B0 96 20 4A EC 20 36 09 03
3798: C9 00 D0 8C C0 00 F0 88 3F
37A0: C0 C9 B0 84 84 1E 38 20 6D
37A8: 02 20 90 07 AD 9B 25 C9 92
37B0: 01 D0 03 4C 25 1F A0 02 E1
37B8: A2 00 B1 19 C9 2A F0 F3 0D
37C0: B1 19 9D 00 02 C8 E8 CC D3
37C8: 9C 25 D0 F4 A9 00 9D 00 C0
37D0: 02 A5 B8 48 A5 B9 48 A0 8A
37D8: 00 A9 02 20 81 0C 68 85 86
37E0: B9 68 85 B8 AD A9 25 85 66
37E8: 1D AD AA 25 85 1E 18 20 ED
37F0: 02 20 AD AB 25 8D 9B 25 94
37F8: AD AD 25 8D 9D 25 AD AC B0

3800: 25 8D 9C 25 60 08 A6 1D D9
3808: CA 86 1B A9 C8 85 1C 18 2A
3810: A9 00 A2 08 6A 66 1B 90 DD
3818: 03 18 65 1C CA 10 F5 85 86
3820: 1C A6 1E CA 8A 18 65 1B 53
3828: 85 1B A5 1C 69 00 85 1C 0B
3830: 06 1B 26 1C A5 1C 6D 60 C9
3838: 25 85 1C A0 01 B1 1B D0 FF
3840: 03 28 18 60 AA 88 B1 1B 3B
3848: 85 19 86 1A 28 90 14 B1 91
3850: 19 29 03 8D 9B 25 B1 19 BE
3858: 29 FC 8D 9D 25 C8 B1 19 F0
3860: 8D 9C 25 38 60 20 6E 1E 65
3868: AD 9B 25 C9 02 F0 32 EE FE
3870: 9C 25 EE 9C 25 A0 00 A5 71
3878: 08 91 1B C8 A5 09 91 1B D0
3880: 88 AD 9B 25 0D 9D 25 91 21
3888: 08 C8 AD 9C 25 91 08 C8 F6
3890: A2 00 BD 00 03 91 08 C8 41
3898: E8 CC 9C 25 D0 F4 4C F1 7B
38A0: 20 20 2D 21 EE A6 25 EE 2C
38A8: A6 25 38 AD A6 25 6D 9C D8
38B0: 25 8D 9C 25 AC A6 25 AD F4
38B8: 9C 25 91 08 A2 00 C8 BD D7
38C0: 00 03 91 08 C8 E8 CC 9C C4
38C8: 25 D0 F4 A0 00 A5 08 91 E0
38D0: 1B C8 A5 09 91 1B 88 AD FD
38D8: 9B 25 0D 9D 25 91 08 C8 24
38E0: AD A6 25 91 08 C8 A2 02 3A
38E8: BD FE 01 91 08 C8 E8 EC 53
38F0: A6 25 D0 F4 A5 08 18 6D 52
38F8: 9C 25 90 06 A5 09 C9 B8 11
3900: F0 0F A5 08 18 6D 9C 25 B8
3908: 85 08 A5 09 69 00 85 09 E3
3910: 60 A9 00 A8 91 1B C8 91 C3
3918: 1B A9 24 A0 7B 20 3E 09 F2
3920: A5 1D 8D 64 25 A5 1E 8D 2E
3928: 65 25 A2 FD 9A 4C 7C 08 D1
3930: BA 8E AE 25 A2 00 A0 00 22
3938: BD 00 03 C9 28 D0 01 C8 D5
3940: C9 29 D0 01 88 9D 00 03 C9
3948: E8 EC 9C 25 D0 EA C0 00 04
3950: F0 03 4C 88 22 A9 00 48 0E
3958: A9 00 85 B8 A9 03 85 B9 F9
3960: 20 B1 00 90 51 C9 2D F0 55
3968: 4D C9 2B F0 49 C9 2E F0 27
3970: 45 C9 50 F0 25 C9 28 F0 A2
3978: 15 C9 41 F0 0B C9 42 F0 14
3980: 07 C9 40 F0 0F 4C 88 22 DC
3988: 20 48 1F 4C B6 21 A9 01 54
3990: 48 4C 5D 21 20 05 12 4C 7D
3998: B6 21 20 B1 00 C9 49 F0 78
39A0: 03 4C 88 22 A9 AE A0 21 45
39A8: 20 F9 EA 20 B1 00 4C B6 E5
39B0: 21 82 49 0F DA A1 20 4A 56
39B8: EC 20 B7 00 F0 78 A2 02 51
39C0: C9 2B F0 35 E8 C9 2D F0 0E
39C8: 30 E8 C9 2A F0 2B E8 C9 39
39D0: 2F F0 26 E8 C9 5E F0 21 35
39D8: C9 29 F0 03 4C 88 22 68 F9
39E0: F0 14 C9 01 F0 07 48 20 6E
39E8: 54 22 4C DC 21 E6 B8 D0 4C
39F0: 02 E6 B9 4C B6 21 4C 53 40
39F8: 12 86 06 68 48 A8 B9 03 B8
3A00: 23 DD 03 23 90 10 20 54 69
3A08: 22 A6 06 68 48 A8 B9 03 D9
3A10: 23 DD 03 23 B0 F0 20 72 1C
3A18: EB A5 A2 48 A5 A1 48 A5 AE
3A20: A0 48 A5 9F 48 A5 9E 48 04
3A28: A5 9D 48 A5 06 48 4C 5D 81
3A30: 21 F0 58 4C 69 EA 68 48 51
3A38: F0 06 20 54 22 4C 33 22 BA
3A40: 68 20 34 ED A0 00 B9 00 CE
3A48: 01 99 00 02 F0 03 C8 D0 B9
3A50: F5 8C A6 25 4C 74 1D 68 E0
3A58: 85 FB 68 85 FC 68 85 07 8F
3A60: 68 85 A5 68 85 A6 68 85 C2
3A68: A7 68 85 A8 68 85 A9 68 1B
3A70: 85 AA 45 A2 85 AB A5 07 52
3A78: 0A A8 A5 FC 48 A5 FB 48 B9
3A80: B9 0B 23 48 B9 0A 23 48 02
3A88: A5 9D 60 AE AE 25 9A A9 17
3A90: 07 8D A6 25 A0 00 B9 83 0F
3A98: 23 99 00 02 C8 C0 07 D0 4D
3AA0: F5 A9 00 99 00 02 60 A9 86
3AA8: 00 85 FB A9 03 85 FD A9 6C
3AB0: 08 85 FC A9 20 85 FE A0 7A

3AB8: 00 B1 FD 91 FB C8 D0 F9 11
3AC0: E6 FC E6 FE A5 FC C9 26 8F
3AC8: D0 EF A9 00 85 F2 20 65 74
3AD0: D6 4C D2 D7 00 00 04 04 A7
3AD8: 05 05 06 06 07 07 04 04 92
3AE0: 05 05 06 06 07 07 04 04 9A
3AE8: 05 05 06 06 07 07 00 80 17
3AF0: 00 80 00 80 00 80 28 A8 88
3AF8: 28 A8 28 A8 28 A8 50 D0 90
3B00: 50 D0 50 D0 50 D0 00 01 B0
3B08: 02 02 03 03 04 3C 22 3C 22
3B10: 22 C0 E7 A9 E7 81 E9 2D A5
3B18: 22 96 EE 4E 54 46 00 2C EF
3B20: 40 40 41 41 41 42 41 43 DB
3B28: 41 44 41 45 41 46 41 47 B9
3B30: 41 48 41 49 41 4A 41 4B 17
3B38: 41 4C 41 4D 41 4E 41 4F 74
3B40: 41 50 41 51 41 52 41 53 D1
3B48: 41 54 41 55 41 56 41 57 2F
3B50: 41 58 41 59 41 5A 42 41 74
3B58: 42 42 42 43 42 44 42 45 E9
3B60: 42 46 42 47 42 48 42 49 47
3B68: 42 4A 42 4B 42 4C 42 4D A4
3B70: 42 4E 42 4F 42 50 42 51 02
3B78: 42 52 42 53 42 54 42 55 5F
3B80: 42 56 42 57 42 58 2A 45 78
3B88: 52 52 4F 52 2A C5 D8 C9 AF
3B90: D4 BA A0 C1 D2 C5 A0 D9 19
3B98: CF D5 A0 D3 D5 D2 C5 A0 E3
3BA0: A8 D9 AF CE A9 BF 00 D3 E4
3BA8: D0 C5 C5 C4 C3 C1 CC C3 80
3BB0: 00 D3 D0 C5 C5 C4 C3 C1 1D
3BB8: CC C3 A0 C2 D9 A0 CB C5 75
3BC0: D6 C9 CE A0 CD C1 D2 D4 E8
3BC8: C9 CE 00 CE C5 D7 BA A0 68
3BD0: C1 D2 C5 A0 D9 CF D5 A0 F9
3BD8: D3 D5 D2 C5 A0 A8 D9 AF 70
3BE0: CE A9 BF 00 D7 C9 C4 D4 65
3BE8: C8 BA 00 C7 CF D4 CF BA 1B
3BF0: 00 D2 C5 C3 C1 CC C3 D5 AF
3BF8: CC C1 D4 C9 CF CE A0 C9 42
3C00: D3 A0 CF 00 D3 C1 D6 C5 9D
3C08: BA 00 CC CF C1 C4 BA 00 0B
3C10: C6 CF D2 CD C1 D4 BA A0 8E
3C18: A0 CC C5 C6 D4 AC A0 C3 97
3C20: C5 CE D4 C5 D2 AC A0 CF 80
3C28: D2 A0 D2 C9 C7 CB D4 A0 D4
3C30: CA D5 D3 D4 C9 C6 D9 BF 28
3C38: 00 C6 CF D2 CD C1 D4 BA 63
3C40: A0 A0 A3 A0 CF C6 A0 C4 4F
3C48: C5 C3 C9 CD C1 CC A0 D0 FD
3C50: CC C1 C3 C5 D3 BA 00 8D 8B
3C58: D0 D2 C5 D3 D3 A0 D2 C5 70
3C60: D4 D5 D2 CE 00 D0 D2 CF B8
3C68: C3 C5 D3 D3 C9 CE C7 A0 A5
3C70: C4 C1 D4 C1 A0 D4 D2 C1 32
3C78: CE D3 C6 C5 D2 00 CE CF 86
3C80: D4 A0 C5 CE CF D5 C7 C8 5F
3C88: A0 D2 CF CF CD A0 D4 CF 67
3C90: A0 C5 CE D4 C5 D2 A0 C4 71
3C98: C1 D4 C1 00 CD CF D6 C5 80
3CA0: A0 C3 D5 D2 D3 CF D2 A0 66
3CA8: D4 CF A0 D4 CF D0 A0 CC B0
3CB0: C5 C6 D4 A0 CF C6 A0 CE 0C
3CB8: C5 D7 A0 D0 CF D3 C9 D4 61
3CC0: C9 CF CE 00 CD CF D6 C5 0D
3CC8: A0 C3 D5 D2 D3 CF D2 A0 8E
3CD0: D4 CF A0 C2 CF D4 D4 CF 33
3CD8: CD A0 D2 C9 C7 C8 D4 A0 03
3CE0: CF C6 A0 C2 CC CF C3 CB 2C
3CE8: 00 D0 D2 C9 CE D4 C9 CE B8
3CF0: C7 AE AE AE 00 D3 CC CF 72
3CF8: D4 A0 A3 00 D0 D2 C9 CE AC
3D00: D4 A0 D4 CF BA A0 A0 D3 12
3D08: C3 D2 C5 C5 CE AC A0 C4 5D
3D10: C9 D3 CB A0 CF D2 A0 D0 C3
3D18: D2 C9 CE D4 C5 D2 BF 00 8E
3D20: C6 C9 CC C5 CE C1 CD C5 45
3D28: BA 00 CE CF A0 C5 D2 D2 6B
3D30: CF D2 D3 00 C5 D2 D2 CF B0
3D38: D2 A0 A3 00 D2 C5 C3 C1 AF
3D40: CC C3 D5 CC C1 D4 C9 CE 5D
3D48: C7 AE AE AE 00 CE CF D4 C2
3D50: A0 C1 A0 D3 D0 C5 C5 C4 CA
3D58: C3 C1 CC C3 A0 C6 C9 CC 7B
3D60: C5 00 FE FE FE FE BA FE F5
```

©

Telecomputing Today

Arlan R. Levitan

Gadgets For Better Telecomputing

I've got a confession to make. I'm a hopeless gadget freak. Every time I see a new piece of equipment that I suspect will make my telecomputing time more productive, I go for it.

Friends who drop in for the first time invariably comment on the number of phones in our computer room. So did the phone company technician who installed them. I still remember the puzzled look on her face. "Four phone lines? " she asked. "I don't mean to be nosy, but what are you going to do with them?"

"One for me and three for the computers," I kidded. "They get kinda lonely during the day and like to call their friends. You saw *WarGames*, didn't you?"

"Uh...sure," she replied, probably wondering if I was a bookie, a psychopathic telephone solicitor, or just a plain nut.

All kidding aside, a dedicated phone line for your computer can be a real plus, especially if you want to receive ordinary phone calls while you're online. It can also help segregate billing for your computer-related calls from your regular phone use.

If you do take voice calls during your online sessions, jamming the phone handset between your shoulder and tilted head while hunched over a keyboard for an hour may leave you looking like a computerized Quasimodo. The solution? A gadget, of course. A hands-free phone device, such as a speakerphone or lightweight NASA-style headset, allows comfortable conversation while you pound away at your keyboard.

Surges And Spikes

Practically everyone knows about surge protectors and the potential dangers of power-line spikes. Yet, although many hobbyists have taken steps to protect their equipment against surges from AC power outlets, the danger of surges traveling over telephone lines into computer equipment is usually ignored. Telephone line surges are relatively rare, but my buddy Fred discovered that all of his AC surge protection was for naught when a nearby lightning strike sent some particularly nasty spikes into his modem, which was connected to his Atari system. Every piece of equipment in the loop was damaged.

At $12.95, Radio Shack's telephone line surge protector (Part #43-102) is reasonably priced insurance. It installs between your modular wall plug and modem. For those who wish to add another level of surge isolation, Data Spec (20120 Plummer Street, Chatsworth, CA 91311), a manufacturer of telecomputing-related goodies, also sells an RS-232 surge protector (Part #RS232SP-300) that installs between your modem and computer using a standard 25-pin RS-232 connector.

Many terminal programs provide a printer on/off feature for those who wish to keep a paper record of their telecomputing sessions. This feature is of limited value if you use transmission speeds faster than 300 baud. Not many printers can keep up with sustained data rates of 120 characters a second or more. When the printer gets behind, the terminal program usually sends an XOFF (CTRL-S) character to the remote system, halting the flow of incoming data until the printer catches up. Then it sends an XON (CTRL-Q) character to resume data transmission. The XON/XOFF cycle goes on ad nauseum, putting a damper on effective transmission speed.

A printer buffer sitting between your system and printer will happily gobble up all the data intended for posterity and control the printer. Printer buffers are available with varying amounts of memory ranging from 8K to 2 megabytes. The most cost-effective approach, for those handy with a screwdriver, is to buy an 8K buffer that is user-expandable to at least 128K. The chips to upgrade from 8K to 128K can be bought for less than $15. Even if you prefer to save the incoming data to disk first and print it out later, a printer buffer can cut the amount of time that your computer is tied up by 90 percent or more.

Hi, BOBs

People who own several computers often use RS-232 switch boxes to toggle modems between machines and transport data between systems with incompatible disk formats. A carefully thought-out switching system can eliminate the drudgery of manually swapping multiple RS-232 cables, allowing changes in cabling with a flick of the wrist. There are dozens of different switch boxes of varying complexity and function. The catalogs of Black Box Corporation (Box 12800, Pittsburgh, PA 15241), MFJ Enterprises (921 Louisville Road, Starkville, MS 39759), and Data Spec will give those who'd rather switch than fight a good idea of what's available.

If you like to make your own cables, these companies also sell some handy diagnostic tools called Break Out Boxes (BOBs). BOBs are typically installed in an RS-232 cable link that is having problems. The best BOBs have Light Emitting Diodes (LEDs) to indicate the electrical status of each line in the link, plus jumpers for testing the effect of wiring changes before whipping out the soldering iron. ©

The Beginners Page

Tom R. Halfhill, Editor

The Hidden Numbers Behind Strings

We dropped a tidbit in last month's column that we promised to explain later—that the alphabetic characters on a monitor screen are merely an outward illusion displayed by computers for our convenience. Internally, computers deal with numbers and *only* with numbers. This has some important implications when you work with character strings in BASIC.

Consider a short routine that asks a user to answer either "yes" or "no" to a question, and which then branches to another part of the program depending on the response. Here's how it might look:

```
10 DIM A$(1):REM This line for Atari
   only
20 PRINT "DO YOU WISH TO
   CONTINUE (Y/N)";
30 INPUT A$
40 IF A$="Y" THEN GOTO 60
50 IF A$="N" THEN END
60 PRINT "Program continues here..."
```

There are a couple of problems with this routine that aren't immediately apparent. At first glance, it seems solid enough: Line 20 asks the question; line 30 fetches and stores the keypress in the string variable A$; line 40 branches to line 60 if the keypress was the letter Y; and line 50 ends the program if the keypress was the letter N.

One problem is a design flaw that doesn't have anything to do with character strings per se: The routine doesn't check for any keypresses besides Y or N. If the user types another key by mistake—or on purpose, just to be mischievous—both IF-THEN tests fail and the program drops through to line 60 as if Y were pressed. There are various approaches to this problem, but one quick solution is to insert line 55 GOTO 20 so the question repeats after each invalid response.

The Computer Is Blind

The main problem we're concerned about, however, has to do with the way computers interpret alphabetic characters. Lines 40 and 50 check for Y or N. But what happens if the user presses a *lowercase* y or n? This can easily happen if the CAPS LOCK key or its equivalent isn't pressed when the program runs. Since this routine doesn't check for y or n, both IF-THEN tests fail and the program drops through to line 60 as if Y were pressed—which may not have been the user's intention at all. Or, if you inserted line 55, the routine keeps pestering the user for a response even though he's frantically pressing what seems to be the right key.

Now, practically anybody who has satisfactorily completed first grade can tell a big Y from a small y or a big N from a small n. But since a computer can't actually see these characters, it can't tell them apart by sight. Instead, it tells characters apart by assigning each one a unique number. Therefore, to a computer, the characters Y and y are as different as A and Z.

To see this for yourself, type PRINT ASC("Y") and press RETURN. The computer should print the number 89 on the screen. This is the *ASCII value* for the uppercase Y character. ASCII stands for American Standard Code for Information Interchange. It's a code developed in the days of teletype terminals which assigns a unique number to each character; the uppercase alphabet from A–Z is numbered 65–90. The ASC() function in BASIC lets you determine any character's ASCII value.

Now type PRINT ASC("y") and press RETURN. Since the lowercase ASCII alphabet is numbered 96–122, the ASCII value of y is 121 on nearly all computers. Exceptions are the Apple II+ and most Commodore computers (save for the Amiga). You can't type this statement on the Apple II+ because it lacks lowercase characters. And on the Commodore computers, you can't type lowercase characters without switching to the alternate character set (press SHIFT-Commodore key). In the standard character set, the ASCII value of uppercase Y is 89, as usual; but when you switch to the alternate set, the ASCII value of the *lowercase* y is 89, and the ASCII value of the *uppercase* Y becomes 217.

Despite these exceptions, you can see the point: Computers handle everything in terms of numbers, so you have to take this into account when writing programs. One way to fix the branching routine above is to substitute these lines:

```
40 IF A$="Y" OR A$="y" THEN
   GOTO 60
50 IF A$="N" OR A$="n" THEN END
```

Censored Characters?

There's another function in BASIC which is the opposite of ASC()—it takes a number and tells you the corresponding ASCII character. Try entering the statement PRINT CHR$(89). The result is the uppercase Y.

Interestingly, some ASCII values represent characters which we can't print here—not because they're obscene and COMPUTE! is a family magazine, but because these "characters" perform a function rather than displaying a letter, number, or symbol. For instance, PRINT CHR$(125) clears the screen on an Atari 400, 800, XL, or XE. PRINT CHR$(147) does the same thing on a Commodore 64, 128, VIC, or PET/CBM. PRINT CHR$(7) rings the internal bell on a Commodore 128 or PET/CBM, Apple, IBM, or Atari ST.

To discover other things you can do by printing these unprintable characters, look for a table of ASCII values in the back of your computer manual or almost any book on BASIC programming. ©

Computers and Society

David D. Thornburg, Associate Editor

The Human Side Of Telecommuting

Several years ago I wrote in this column about *The Network Nation*, a book on human communication via computer written by Starr Hiltz and Murray Turoff (Addison-Wesley, 1978). The authors made several predictions in the book, including the speculation that computerized conferencing would be a prominent form of communication in most organizations by the mid-1980s; would make it possible for a large percentage of the labor force to work at home during at least half of the normal work week; and would indirectly conserve sizable amounts of energy by substituting communication for travel.

Of 14 predictions made by these authors, I want to focus on just these three—not because they haven't yet happened, but because they were very reasonable predictions in 1978.

If these predictions were reasonable then, what has kept them from coming true? Based on the price of gasoline and the high quality of our computer and communications technology, telecommuting seems ripe for development. Some companies have expressed great interest in this style of working, especially since it allows workers to function as independent contractors, thus reducing the employer's overhead.

One company which has conducted an experiment in this field is Avco Lycoming, one of the world's leading manufacturers of gas turbine engines. Given the highly technical nature of this company's business, many of their employees (software designers, for example) are information workers who would be suitable candidates for telecommuting.

In September 1984, one of these employees, Lee Jacko, had asked to take part in a six-month telecommuting experiment. The company worked out the details and arranged for it to be monitored and evaluated by Drs. Herb Spirer and Al Katz from the University of Connecticut.

Water Cooler Conversation

Jacko's reason for trying this experiment was that she planned to be a mother some day, and she wanted to see if she could work effectively in her home. The fact that commuting to work took one hour each way probably contributed to her interest as well. As a software designer and programmer, Jacko is comfortable with computers, and the company set up an IBM PC-XT in her home.

Early in the experiment it was found that she needed to show up at the office one day a week just to stay in touch with her colleagues. In retrospect, this is easy to understand. We don't often think about it, but much of our information-gathering is informal. We join a conversation at the water cooler that leads to a better way to solve a problem, or we hear of a new job opening in another division, and so on. An amazing amount of valuable information is exchanged informally. Many years ago when I worked for a Fortune 500 company, I found that one of the best ways to spread information was to "accidentally" leave it in the office copier!

Jacko also quickly realized that she was missing the benefits of regularly scheduled group meetings. As soon as this problem was identified, a speakerphone was set up in the conference room so she could participate from home.

Jacko is not a loner. She likes being where the action is, and was afraid that this experiment might hurt her career. By being out of sight, she was afraid of being out of mind as well. But in fact, her colleagues were quite supportive and she found that telecommuting didn't hurt her career at all.

She cautions that telecommuting isn't for everyone, however. It takes discipline to work without supervision. Even though she had clearly set goals, it was her own work habits that insured her diligence on the job. To help maintain this discipline, she rose at the same time as her husband each morning, and got dressed just as though she were leaving the house for work. She worked from 8 a.m. to 6 p.m., and her only concession to being at home was an occasional two-hour lunch to compensate for her longer work day. Both Jacko and her supervisor were very happy with the quality of her work.

Social Animals

At the end of the six-month experiment, Jacko was ready to come back to the office. The experience of working at home was good, but she missed being with her colleagues. Now she believes she'd be happy spending four days a week at home for six months, followed by a two-month stint in the office.

The researchers who studied her during this experiment expected to see morale problems, but none appeared. In fact, Jacko maintains that people who work well in isolation would really blossom as telecommuters.

The benefits of telecommuting seem to be great, yet it still is not popular. The reasons probably have more to do with human nature than with technology. We are social animals and seek the company of our peers. Whether it is a collection of aborigines gathering around a water hole, or a gathering of executives around the water cooler, we need face to face contact with other humans on a regular basis. Perhaps one day a week is enough time to socialize in the office. More research needs to be done. We understand the technology; it is human nature that we need to focus on now. ©

Fred D'Ignazio, Associate Editor

Arjan Singh Khalsa: A Prophet Of Bionic Man

Bionic man.

What do these words bring to mind? They make me think of science fiction, a TV show called *The Six Million Dollar Man*, and Lee Majors. Majors starred as the bionic man we are most familiar with—more machine, really, than human. Humans as machines.

But a bionic man can also be a blind person using a talking word processor, or a victim of cerebral palsy blowing into a puff switch to activate a computerized wheelchair or robotic arm. Here, technology doesn't make a person more machinelike. Instead, it enables him or her to be more fully human.

One person with this view of bionic man is Arjan Singh Khalsa, of Berkeley, California. From the tip of his toes to the top of his white turban, Khalsa is a man with a mission: To shape technology in a human image so it can become a prosthetic extension of the human mind and body. He is a proponent of a new man/machine symbiosis—a prophet of bionic man.

The Elegance Of Technology

On the one hand, Khalsa is an evangelist for technology and for its potential to help people. On the other hand, he is an arch-critic of technology who condemns its disruptive effects on people's lives. He is also the founder and president of Educational Software Review, a "technology watchdog" company that tests new educational software from large corporations. And he is producing his own products which embody his goals to make technology more elegant.

"Elegant" is a word he uses a lot. According to Khalsa, technology is elegant when it is a simple, natural extension of a person's mind or body; when it is immediately useful; and when it is being stretched to its limit—in the service of human beings. Khalsa doesn't believe a product is truly elegant unless it can be used by both "enabled" and disabled people.

For example, Educational Software Review is marketing a program called *The Magic Music Teacher* (a $69.95 two-sided disk for the Apple, and soon, for the Commodore 64). Two key features of *The Magic Music Teacher* are that it can be operated by pressing only two keys—or two switches, for a disabled person; and when equipped with an Echo/Cricket speech synthesizer, it talks—so it can be used by a blind person. These features have made the program immensely popular with everyone from the California School for the Blind to the Boston Retarded Children's Choir.

The Magic Music Teacher teaches the children in the choir by using the Suzuki method of hearing a melody, then learning to repeat it. The children quickly master the two switches, and they begin "playing" a musical instrument. According to Khalsa, "The kids laugh and rejoice when they use the program. They are learning that they can succeed at something. Technology and music are increasing the joy in their lives."

It's no surprise that *The Magic Music Teacher* is also a hit with enabled children and adults. "Nobody who has begun using the program has ever used it for less than a half hour," says Khalsa. "It is too easy, and too much fun."

Restoring The Sound

Educational Software Review's other product is the flip side of this same philosophy. After observing dozens of children using computers in classrooms, he noticed that many good educational programs which use sound are muted so other children won't be disturbed. "It's a shame," says Khalsa. "The computer is one of our most powerful learning tools, partly because it reinforces learning with sound as well as images. Then we turn off the sound."

Khalsa thinks this is an example of not properly fitting technology to human beings. With the flick of a switch, technology is disabling hearing children and rendering them deaf. His solution is a computer headset, the LittleJack ($24.95, with a volume control and a connector that allows up to ten children to listen together if they plug their own headsets into an adapter).

Khalsa is looking for licenses to convert more existing products into products appropriate for the 35 million disabled and handicapped people in the U.S. In addition, he's trying out new inventions, like a talking word processor. Khalsa says his word processor is "like a huge Speak 'N' Spell, only it can interface with a computer and is completely programmable. For example, Vietnamese kids can crayon pictures in squares on regular paper, then slip the paper on the word processor's large, flat pad. When they press the pictures, the word processor will print out the words in English describing the pictures; and it will say the words aloud—in English and in Vietnamese."

For Khalsa, a disability can be physical, mental, emotional, cultural—or technological. Machines should never be allowed to disable a person. Instead, they should enable people and help them lead richer, more human lives.

(To contact Arjan Singh Khalsa, write Educational Software Review, 1400 Shattuck Avenue, Suite 774, Berkeley, CA 94709.) ©

INSIGHT: Atari

Bill Wilkinson

Avoiding Memory Confusion In Atari BASIC

After a couple of months of standing on my soap box, I've decided to step off and get back to business again. Before I do, though, here's one more little rant and rave: I can now express my opinion of Atari's new BASIC for the 520ST. In a word: disappointing. Neither ST Logo nor ST BASIC are viable production languages, which means you can't write commercial applications with them. Since even the C compiler included in Atari's $300 software developer's package doesn't support double-precision arithmetic, limiting you to six decimal digits of precision, you'd better be ready to purchase some language from an outside vendor if you're serious about doing any programming on the ST machines.

Several months ago, I asked all you loyal readers to send me a postcard or letter giving ratings to the best or worst Atari-oriented books. Although I was a little underwhelmed by the response, I did get enough ballots to at least select the three favorites. Among these three, however, there was no clear-cut winner. And I happen to feel that is appropriate, since each addresses a different part of the knowledge an Atari programmer needs. Anyway, according to my readers, the best books are (drum roll...the envelope please): *The ABC's of Atari Computers,* by Dave Mentley, published by Datamost; *Your Atari Computer,* by Lon Poole et al, published by Osborne/McGraw Hill; and *Mapping the Atari,* by Ian Chadwick, published by COMPUTE! Books. (Incidentally, you may have noticed that COMPUTE! Books has been shipping the new, revised version of *Mapping the Atari,* which has several appendices and notes devoted to the XL and XE machines.)

The rest of this column responds to a number of reader requests. Although the topic has been covered in COMPUTE! before (at least in part), there are many newcomers out there. And even if you aren't a newcomer, maybe I can provide more insight into the concepts involved.

Finding Free Memory

Q: Where in memory can a programmer put machine language routines, character sets, player/missile graphics, and the like?

A: There is no simple answer, because it depends on which language you're using, which DOS, etc. A couple of years ago, I did an entire series on relocatable machine language which was related to this problem. So this time, let's tackle a simpler and more specific question: Where can I put a custom character set? The following techniques will also work for many other uses, including player/missile graphics.

When allocating memory, Atari BASIC—as well as BASIC XL and BASIC XE—looks at and believes the contents of two memory locations, LOMEM and HIMEM (located at $2E7, decimal 743, and $2E5, decimal 741, respectively). BASIC always starts your program where LOMEM tells it to and lets it grow as high as the value in HIMEM. Remember that this "growing" includes not just your BASIC code, but also the strings and arrays dimensioned by your program. Let's consider LOMEM first.

The fact that a program always starts at LOMEM implies that if we increase the value of LOMEM and then load a program, the memory between the old value and the new one is available for whatever purposes we have in mind. On the other hand, once a BASIC program is loaded into memory, it ignores changes to LOMEM. This means we can have one program change the contents of LOMEM and then chain to another program. The first program is unaffected by the change, but the second will be loaded at the new LOMEM. Programs 1 and 2 demonstrate this technique.

Examine Program 1, which ensures that the memory we wish to reserve starts on a particular boundary. Remember that full character sets (128 characters) must start on 1K memory boundaries, and half sets must start on 512-byte boundaries. There are similar rules for player/missile graphics (see "Atari Animation With P/M Graphics," a three-part series starting in the September 1985 issue of COMPUTE!). If you actually type in and run the programs below, you'll be in for a little surprise. But *do not* omit the REMark statements from Program 2, or you'll miss half the fun. Feel free to omit them from Program 1. For the programs to function properly, you must save Program 2 with the filename PROGRAM2.BAS (see line 900 of Program 1). If you're using cassette instead of disk, change line 900 in Program 1 to RUN "C:" and make sure the tape is cued to Program 2 before you run Program 1.

A minor caution: The reason we base the changes to LOMEM on the contents of locations 128 and 129 (BASIC's internal MEMLO pointer) instead of the actual LOMEM contents is complex. I have discussed it in this column before, but the heart of the problem is that some Atari device drivers (including the 850 Interface Module's R: handler) do not correctly restore LOMEM when the SYSTEM RESET button is pressed. After a reset, BASIC's pointer is more reliable. For the same reason, and for safety's sake, programs bumping LOMEM should always bump it higher than the top of the BASIC program currently in memory. And one last piece of advice: If you run Program

1 over and over again, it keeps raising LOMEM higher and higher. Eventually you'll run out of memory. You probably need some sort of flag elsewhere in memory (Page 6?) which tells the program not to raise LOMEM again.

Modifying HIMEM

Enough about LOMEM; what about HIMEM? Truthfully, if you know how big your program is and what it's going to use in the system, you can put anything you want (character sets, machine language, player/missile shape data, etc.) in the memory between the top of your program and the bottom of screen memory. The only time the contents of HIMEM are used is when BASIC checks to ensure that APPMHI (location 14, $0E) hasn't collided with it. APPMHI is essentially BASIC's high water mark. It keeps track of the top of the run-time stack, which is always above the string and array space, which in turn is always above your program. So, if you *know* that your program, its data, and its stack will never grow too large, you could ignore HIMEM altogether. It's much cleaner, though, to tell the system what you're using by modifying HIMEM.

How and why does HIMEM change if you don't do this? The most usual cause is a change in the graphics mode. For example, while ordinary text screen graphics (GRAPHICS 0) occupy less than 1K of memory, several graphics modes (such as modes 8, 9, 10, 11, and 15) require 8K of screen memory. To demonstrate this, type in and run the following line, preferably after hitting the SYSTEM RESET button:

```
G0=FRE(0):GR.8:PRINT G0,FRE(0),G0-FRE(0)
```

This displays three numbers: memory available for your program(s) in text mode, usable memory in mode 8, and the extra amount used by mode 8 graphics.

Generally, the best method is to always put your own goodies below the area occupied by the most memory-intensive graphics mode you plan to use. So either look in a memory map book to find out how much room a certain graphics mode will take, or simply change modes before using the memory.

For an example, try Program 3. It's essentially the same as Program 2. The difference is simply where we move the character set. The REMarks explain where you should insert your own graphics mode declaration. ©

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: MEMLO Bumper

```
HF 100 REM
DG 110 REM THIS PROGRAM IS U
       SED TO
BH 120 REM RESERVE SIZE"PAGE
       S" OF
IG 130 REM MEMORY FOR PROGRA
       M2.BAS
HJ 140 REM
AE 150 REM (A "PAGE" IS 256
       BYTES
HL 160 REM
CA 170 REM THIS PROGRAM ALSO
        ENSURES
FL 180 REM THAT THE RESERVED
        SPACE
LK 190 REM STARTS ON THE GIV
       EN BOUNDARY
JD 200 REM (TO INSURE,FOR EX
       AMPLE, THAT
OK 210 REM CHARACTER SETS ST
       ART ON 1K
MB 220 REM BYTE BOUNDARIES)
HJ 230 REM
KB 500 SIZE=4:REM MUST BE AT
        LEAST 4 PAGES (1024
       BYTES)!
DI 510 BOUNDARY=4:REM ALSO G
       IVEN IN PAGES
DC 520 IF PEEK(128)<>0 THEN
       POKE 128,0:POKE 743,0
       :SIZE=SIZE+1
MH 530 MEMLO=PEEK(129)+SIZE
GF 540 MEMLO=INT((MEMLO+BOUN
       DARY-1)/BOUNDARY)*BOU
       NDARY
AO 550 POKE 744,MEMLO
AM 560 POKE 129,MEMLO
JO 900 RUN "D:PROGRAM2.BAS"
```

Program 2: Character Set Mover, Version 1

```
DG 150 REM JUST AS A DEMO, T
       HIS PROGRAM
KM 160 REM CHANGES THE CHAR
       SET POINTER,
JJ 170 REM COPIES THE CHARAC
       TER SET
GG 180 REM TO THE RESERVED M
       EMORY,
BF 190 REM AND THEN RADOMLY
       DESTROYS
FI 200 REM THE CHARACTERS!
HH 210 REM
HJ 220 REM HIT RESET TO QUIT
        AND GET
LA 230 REM NORMAL CHARACTERS
        AGAIN.
HK 240 REM
BI 250 GRAPHICS 0
JM 260 SIZE=4:REM SHOULD BE
       THE SAME AS PROGRAM 1
BF 270 POKE 756,PEEK(129)-SI
       ZE:REM CHBAS IS CHANG
       ED
HA 280 BUFFER=PEEK(756)*256
FI 290 POKE 752,1:PRINT :REM
        NO MORE CURSOR
NK 300 FOR ADDR=BUFFER TO BU
       FFER+1023
NA 310 POKE ADDR,0:REM FIRST
        CHANGE ALL CHARS
DO 320 NEXT ADDR:REM TO SAME
        REPEATED PATTERN
CA 330 LIST 150,240:REM JUST
        SOMETHING TO SHOW
IL 340 REM READY TO MOVE THE
        CHARACTERS
HA 350 FOR ADDR=0 TO 1023
LH 360 POKE BUFFER+ADDR,PEEK
       (57344+ADDR)
PE 370 NEXT ADDR
CB 380 REM MOVED...SLOWLY DE
       STROYED
NG 390 POKE INT(RND(0)*1024)
       +BUFFER,INT(RND(0)*25
       6)
```

Program 3: Character Set Mover, Version 2

```
DG 150 REM JUST AS A DEMO, T
       HIS PROGRAM
IA 160 REM CHANGES THE CHAR
       SET POINTER
JJ 170 REM COPIES THE CHARAC
       TER SET
GG 180 REM TO THE RESERVED M
       EMORY,
GD 190 REM AND THEN RANDOMLY
        DESTROYS
FI 200 REM THE CHARACTERS!
HH 210 REM
HJ 220 REM HIT RESET TO QUIT
        AND GET
LA 230 REM NORMAL CHARACTERS
        AGAIN.
HK 240 REM
BG 250 GRAPHICS 7:REM JUST T
       O CLEAR ABOUT 4K OF M
       EMORY!
HP 260 GRAPHICS 0:REM OR OTH
       ER MODE
EF 270 SIZE=4
OM 280 REM ALWAYS DO FOLLOWI
       NG AFTER THE GRAPHICS
        STATEMENT
HJ 290 POKE 741,255:REM ENSU
       RE END-OF-PAGE BOUND
MG 300 MEMHI=INT(PEEK(742)/S
       IZE)*SIZE-SIZE
HB 310 POKE 742,MEMHI-1:REM
       LOWER HIMEM
AH 320 POKE 756,MEMHI:REM CH
       BAS IS CHANGED
GM 330 BUFFER=PEEK(756)*256
FE 340 POKE 752,1:PRINT :REM
        NO MORE CURSOR
CC 350 LIST 150,240:REM JUST
        SOMETHING TO SHOW
IN 360 REM READY TO MOVE THE
        CHARACTERS
HC 370 FOR ADDR=0 TO 1023
LJ 380 POKE BUFFER+ADDR,PEEK
       (57344+ADDR)
PG 390 NEXT ADDR
BK 400 REM MOVED...SLOWLY DE
       STROYED
MP 410 POKE INT(RND(0)*1024)
       +BUFFER,INT(RND(0)*25
       6)
GE 420 GOTO 410
```

©

Programming the TI

C. Regena

Computerized Messages

With the abundance of home computers, people are having fun with computerized messages and electronic communication. For instance, you can program your TI to play "Happy Birthday" to a friend. My December columns for the last few years have contained programs for the TI that can be used for Christmas greetings.

The recent birth of our baby was another occasion for computerized messages. My spouse put a system message on the mainframe computer at work so fellow employees would know our news. Electronic mail carried the message to other colleagues. Some of our relatives and friends have TI computers, so I wrote a birth-announcement program and sent them copies. We mailed printed announcements, complete with graphics, to other friends who don't have computers. We're such proud parents that I decided to include the program here. You can use this general idea to create your own computerized messages.

The music for this program is Brahm's "Lullaby." Line 140 defines a tempo in the variable T. The value of T represents an eighth note, and all the CALL SOUND statements express duration in terms of T. Lines 120 and 130 define sound frequencies for the melody notes. Notice that the DATA statement has eight numbers which correspond to the eight variable names in the READ statements. By the way, these frequencies actually represent the flats for each named note except F.

Line 150 changes the screen color. I had planned to use color 8 (cyan) or 5 (dark blue) for a baby boy, or color 7 (dark red) for a baby girl.

Lines 160–600 combine CALL SOUND statements with CALL CHAR statements to define graphic characters while playing music. Lines 610–650 define the colors for the graphics. Line 620 defines a light-blue color for the stork's hat and part of the baby (try color 10 for a baby girl). Lines 630–650 define the colors for the stork. If you prefer white lettering instead of black, you could change line 630 to FOR N=2 TO 11.

Lines 660–1000 play music while printing the announcement. It displays the graphics on the screen with PRINT instead of CALL HCHAR or CALL VCHAR because the PRINT method is quicker. The CHR$ statement specifies a certain character number to be printed. Most of the stork is composed of characters that are redefined lowercase letters. Release the ALPHA LOCK key to type these letters in the statements.

Lines 1010–1420 continue playing the music. Lines 1430–1450 keep the announcement on the screen until a key is pressed. A keypress clears the screen and ends the program.

If you prefer to save typing, you can obtain a copy of "Announcement" by sending a blank cassette or disk, a stamped, self-addressed mailer, and $3 to:

C. Regena
P.O. Box 1502
Cedar City, UT 84720

```
100 REM ANNOUNCEMENT
110 CALL CLEAR
120 READ BG,BA,BB,C,D,E,F
    ,G
130 DATA 185,208,233,247,
    277,311,349,370
140 T=350
150 CALL SCREEN(8)
160 CALL SOUND(T,BB,5)
170 CALL CHAR(123,"000000
    00003C7CFE")
180 CALL CHAR(97,"0007ØCØ
    80810101")
190 CALL SOUND(T,BB,6)
200 CALL CHAR(98,"FCØ201"
    )
210 CALL CHAR(99,"0000008
    Ø8040404")
220 CALL SOUND(2*T,D,4)
230 CALL CHAR(100,"000E11
    1070888484")
240 CALL CHAR(101,"003057
    8989898909")
250 CALL CHAR(102,"080808
    0808040404")
260 CALL CHAR(103,"040E0E
    00312E2222")
270 CALL SOUND(2*T,D,4,13
    9,8)
280 CALL CHAR(104,"404080
    808040404")
290 CALL CHAR(105,"828140
    7C8380403F")
300 CALL CHAR(106,"5152D4
    A89063FC38")
310 CALL CHAR(107,"E0202Ø
    4080000F3")
320 CALL SOUND(T,BB,5,139
    ,8)
330 CALL CHAR(108,"040404
    020202FC04")
340 CALL CHAR(109,"111110
    0808040404")
350 CALL SOUND(T,BB,4,139
    ,8)
360 CALL CHAR(110,"2020A0
    9050502828")
370 CALL CHAR(111,"372834
    2B2824231")
380 CALL SOUND(2*T,D,4)
390 CALL CHAR(112,"C00000
    807F00008")
400 CALL CHAR(113,"040404
    0CF40808C")
410 CALL CHAR(114,"040404
    0404040404")
420 CALL CHAR(115,"140C0C
    12122141C1")
430 CALL SOUND(2*T,139,8,
    185,8)
440 CALL CHAR(116,"101008
    040201")
450 CALL CHAR(117,"7F0000
    000000C03F")
460 CALL CHAR(118,"C00000
    0000000FF")
470 CALL CHAR(119,"080911
    1222C20201")
480 CALL SOUND(T,BB,5)
490 CALL CHAR(120,"800000
    00000000FF")
500 CALL CHAR(121,"844448
    30202020C")
510 CALL SOUND(T,D,4)
520 CALL CHAR(122,"000000
    0000003844")
530 CALL CHAR(128,"010204
    08103F")
540 CALL SOUND(2*T,G,3,D,
    7,BB,9)
550 CALL CHAR(129,"808080
    87F982808")
560 CALL CHAR(130,"000000
    C0804")
```

```
570 CALL CHAR(131,"808080
    8080808080")
580 CALL SOUND(3*T,F,2,D,
    8,BB,8)
590 CALL CHAR(132,"000003
    ")
600 CALL CHAR(133,"808060
    808")
610 CALL COLOR(13,11,1)
620 CALL COLOR(12,6,1)
630 FOR N=9 TO 11
640 CALL COLOR(N,16,1)
650 NEXT N
660 CALL SOUND(T,E,2,BB,7
    ,BG,9)
670 PRINT TAB(5);CHR$(123
    )
680 CALL SOUND(2*T,E,3,BA
    ,7,175,9)
690 PRINT TAB(4);"abcCHAN
    DLER AND"
700 PRINT "de fghCHERYL R
    EGENA WHITELAW"
710 PRINT "ijklmn"
720 CALL SOUND(2*T,D,4,BA
    ,7,175,9)
730 PRINT " opqrszANNOUNC
    E THE BIRTH OF"
740 PRINT " tuvwxy"
750 PRINT TAB(3);CHR$(128
    );CHR$(129);CHR$(130)
760 CALL SOUND(T,BA,4)
770 PRINT TAB(4);CHR$(131
    );"{4 SPACES}BRETT LY
    NN WHITELAW"
780 CALL SOUND(T,BB,4)
790 PRINT TAB(3);CHR$(132
    );CHR$(133)
800 CALL SOUND(T,C,3)
810 PRINT : :
820 CALL SOUND(T,C,3,BG,8
    )
830 CALL SOUND(2*T,BA,3,1
    39,8)
840 PRINT "BORN:  OCTOBER
     19, 1985"
850 PRINT :"TIME:  2:48 A
    .M."
860 CALL SOUND(T,BA,2)
870 PRINT :"WEIGHT:  8 PO
    UNDS 10 OUNCES"
880 CALL SOUND(T,BB,2)
890 PRINT :"LENGTH:  22 I
    NCHES"
900 CALL SOUND(T,C,2)
910 CALL SOUND(T,C,2,BG,8
    )
920 CALL SOUND(T,139,8)
930 CALL SOUND(T,175,8)
940 CALL SOUND(T,BA,3)
950 CALL SOUND(T,C,2)
960 CALL SOUND(T,F,1)
970 CALL SOUND(T,E,1,BG,6
    )
980 CALL SOUND(2*T,D,2,17
    5,7)
990 PRINT : :"ALSO WELCOM
    ED BY CHERY,"
1000 PRINT "RICHARD, CIND
    Y, BOB, RANDY"
1010 CALL SOUND(2*T,F,2,C
    ,6,BA,8)
1020 CALL SOUND(T,G,2,BB,
    5)
1030 CALL SOUND(T,G,2,BB,
    5,BG,8)
1040 CALL SOUND(T,G,2,BB,
    5,139,7)
1050 CALL SOUND(T,G,2,BB,
    5)
1060 CALL SOUND(T,BG,4)
1070 CALL SOUND(T,BG,3)
1080 CALL SOUND(2*T,G,2,E
    ,5)
1090 CALL SOUND(2*T,G,2,E
    ,5,BG,8)
1100 CALL SOUND(T,E,3,BG,
    8)
1110 CALL SOUND(T,C,4,BG,
    8)
1120 CALL SOUND(4*T,D,3,B
    B,6,BG,8)
1130 CALL SOUND(T,BB,4,13
    9,8)
1140 CALL SOUND(T,BG,4,13
    9,8)
1150 CALL SOUND(T,C,3,BA,
    6)
1160 CALL SOUND(T,C,3,BA,
    6,139,9)
1170 CALL SOUND(T,D,2,BB,
    5)
1180 CALL SOUND(T,D,2,BB,
    5,139,9)
1190 CALL SOUND(T,E,1,C,4
    )
1200 CALL SOUND(T,E,1,C,4
    ,139,9)
1210 CALL SOUND(T,BB,1)
1220 CALL SOUND(T,D,2)
1230 CALL SOUND(T,D,2,BG,
    8)
1240 CALL SOUND(T,D,2,139
    ,8)
1250 CALL SOUND(T,BG,4)
1260 CALL SOUND(T,BG,3)
1270 CALL SOUND(2*T,G,1,E
    ,4)
1280 CALL SOUND(2*T,G,1,E
    ,4,BG,8)
1290 CALL SOUND(T,E,2,BG,
    6)
1300 CALL SOUND(T,C,3,BG,
    6)
1310 CALL SOUND(4*T,D,4,B
    B,8,BG,9)
1320 CALL SOUND(T,BB,4,13
    9,8)
1330 CALL SOUND(T,BG,3,13
    9,8)
1340 CALL SOUND(T,C,3,BA,
    7)
1350 CALL SOUND(T,C,3,139
    ,8)
1360 CALL SOUND(50,D,4)
1370 CALL SOUND(50,C,4)
1380 CALL SOUND(T,BB,3)
1390 CALL SOUND(T,E,4)
1400 CALL SOUND(T,BA,5)
1410 CALL SOUND(T,F,5,C,9
    )
1420 CALL SOUND(4*T,G,5,B
    B,9,BG,12)
1430 CALL KEY(0,K,S)
1440 IF S<1 THEN 1430
1450 CALL CLEAR
1460 END
```

©

IBM Personal Computing

Donald B. Trivette

Compiling BASIC

This month's issue has a couple of articles about the Motorola 68000, the super-fast microprocessor chip that powers the Apple Macintosh, Atari ST, and Commodore Amiga. IBM users aren't left out of this contest. Intel Corporation has its own super-fast microprocessor, the 80286, which is found in the IBM AT, the AT&T PC 6300+, and several AT compatibles. But if you don't want to buy a new computer just now, there's another way to make some of your programs run faster.

Consider the following three-statement BASIC program we'll call PROG1.BAS:

```
10 FOR I=1 TO 2000
20 J=I*I
30 NEXT I
```

It finds the squares of the numbers from 1 to 2,000. It takes eight seconds on a PC or PCjr, three seconds on an IBM AT with its faster microprocessor, and two seconds on AT&T's 6300+. Now, let's see if we can get the speed of the AT&T computer out of a PC or PCjr.

IBM BASIC is an *interpretive* language. This means the computer must translate each statement into machine language instructions before execution. Because PROG1.BAS consists of a loop, BASIC must translate and execute a total of 6,000 statements (three statements 2,000 times). Although the overhead to interpret a single instruction is very small, the overall time adds up. Think how much faster the program could run if each BASIC instruction could be translated into machine language just once, rather than each time it is executed. Basically, that's what a *compiler* does.

Compiling a program is usually a two-step process. First, the source program—that's your BASIC program—is processed by the compiler. The output from the compiler is then processed by a *link* program. The output from the linker is the compiled BASIC program in the form of an .EXE file.

New & Improved Compiler

Last summer, IBM released Version 2.0 of its BASIC Compiler. It incorporates all the new features added to interpreter BASIC since the first version of the BASIC Compiler was released in 1982. These include VIEW, WINDOW, PAINT, SHELL, hard disk commands, and all of the advanced features of the PCjr, such as multivoice music and user-defined palettes. In addition, IBM has added some features to Compiled BASIC that are not available in the interpreter. These include named subprograms, user-defined multiline functions, and separately compiled subprograms. Also, the Compiler manual has been enlarged to two volumes: *BASIC Compiler Fundamentals* and *BASIC Compiler Language Reference*.

There's a price to pay for all these goodies. The old version sold for $300; the new version carries a retail price of $495. And there's another factor to consider: Version 2.0 generates larger .EXE files than Version 1.0.

Unless you need some of the compiler's advanced features, it's easy to use; in fact, it's easier to run than most word processing programs. First, you save the BASIC program on disk with the ASCII option (SAVE "PROG1.BAS",A). Next, you run the compiler by typing its name: BASCOM. It asks for the name of the input file (PROG1.BAS) and any other options you might want to select.

If the input file is PROG1.BAS, the compiler's output goes to a file called PROG1.OBJ. This is known as the *object module* or *object file*. At this point, the program is compiled but not executable. There are still some things the program must know before it can run. To resolve these unknowns (technically known as external references), the object file must be processed by the link program on the compiler disk. Output from the link program is the final program ready to execute—in this case, PROG1.EXE.

PROG1.EXE is known as an *executable module* or a *run module*. To run it, simply type the filename as if it were a DOS command: PROG1. As the table indicates, a compiled program runs three to four times faster than an interpreted one. (The run module produced by the new version of the compiler is no faster than that produced by the old version.) The price to pay for speed is size. The interpreter version uses only 56 bytes of disk space, while the compiled version takes more than 23,000 bytes. ©

Size in Bytes	Compiler 1.0		Compiler 2.0
PROG1.BAS	56		56
PROG1.BAS (ASCII)	74		74
PROG1.OBJ	875		980
PROG1.EXE	18,304		23,334
Compiling Time			
PROG1.BAS	:02		:02
Linking Time			
PROG1.OBJ	1:35		:59
Execution Time	**IBM PC**	**IBM AT**	**AT&T 6300+**
Interpreted PROG1.BAS	:08	:03	:02
Compiled PROG1.BAS	:02	:01	:00.5

Memo Diary

You may have noticed that the year value behaves strangely in this program from the December 1985 issue (p. 65). To solve this, add the following two lines, which were accidentally omitted from Program 1 (Atari and TI owners should add line 1030 *only*):

```
1030 IF D8$ <= D9$ THEN 1050
1040 Y$="/" + RIGHT$ (STR$ (100 +
  Y8), 2)
```

The article failed to mention that you should enter only two digits for the year when you first run the program (for example, 86 for 1986). Entering all four digits results in incorrect days of the week for the dates you select.

The Atari and TI versions (Programs 3 and 6) each have additional corrections. In both versions, the month can only be entered as a number, not as a word. Also, in the TI version, incorrect menu choices crash the program. Make the following changes, suggested by reader David Wentzel:

Atari version:

```
1695 IF LEN(MM$) > 2 THEN 1710
1770 IF MM$<>M$((J-1) * 3 + 1, J * 3)
  THEN 1790
```

TI version:

```
815 IF (A<1) + (A>5) THEN 730
1695 IF LEN(MM$) > 2 THEN 1710
```

Balloon Crazy For TI And IBM

The IBM version (Program 4, p. 59) of this game from the December 1985 issue has a minor bug. When a new screen is drawn after clearing all balloons from a previous screen, the display always shows three clowns remaining regardless of how many are actually left. To correct this, reader Matthew Pomeroy suggests the following change to line 190:

```
190 FOR I=158 TO 158 + (LIVES - 2) *
  8 STEP 8: PUT(I,0), TINY: NEXT:
  GOSUB 350
```

Part of line 390 is missing in the TI version of this game (Program 5, p. 60). The line should read as follows:

```
390 CALL SPRITE(#3,124,14
    ,118,MCOL):: GOSUB 56
    0 :: CALL DELSPRITE(#
    3):: CALL SPRITE(#1,1
    36,14,150,MCOL)
```

Apple ProDOS Disk Menu

This utility program from the December 1985 issue (p. 108) gives a BAD SUBSCRIPT ERROR in line 20 when run because its first line is missing. Add the following:

```
5 DIM A$(24), L$(52)
```

Also, David Mariotti suggests the following improvements which cause the selector bar to skip blank lines when there are fewer than 16 items in the directory display:

```
4115 IF CR > LIM + 2 THEN CR = 3
4210 IF CR = 4 THEN CR = LIM + 4
```

Atari Reset Controller

Errors were accidentally introduced in Program 2 for this article from the January 1986 issue (p. 110) when REM statements were deleted. The GOTO 340 in line 300 should be changed to GOTO 360, and the GOTO 180 in line 320 should be changed to GOTO 200. A good programming rule to help avoid such problems is never GOTO a REM statement.

Apple ML Addresses

In the December 1985 "Reader's Feedback" column, there is an error in line 20 of the ProDOS routine for finding the starting address of machine language programs (p. 18). The statement GOTO 15 should be GOTO 20.

Atari Lightning Renumber

The author of this program from the October 1985 issue (p. 103) has provided a fix for a bug that causes the program to sometimes miss internal line number references in program lines. Line 810 should be changed to read as follows:

```
810 DATA 200,177,203,201,22,240,10,
  201,155,240
```

Skyscape

In addition to the small correction published in last month's "Capute!" column, there are a number of corrections required for the Atari version, and additional changes to the Commodore 64, Apple, and TI versions. In the Atari version, the following lines need to be corrected as shown:

```
FM 520 FOR ZZ=1 TO 40:PRINT
     CHR$(RF+32);:NEXT ZZ:
     GOTO 540
EI 1000 IF ABS(LL)>90 THEN P
     RINT 00$:GOTO 980
HG 1730 IF P(X,6)<K1 AND P(X
     ,6)>MS THEN 1760
IF 2590 IF ABS(LL)>90 THEN P
     RINT 00$:GOTO 2580
CE 2600 GOSUB 2260:IF Z$="N"
      THEN 2560
NG 2610 GOSUB 2510:Q$="S":GO
     TO 1950
```

In the Commodore 64 version, the reinput option of the latitude change feature does not work correctly. Change the THEN 2480 at the end of line 2570 to THEN 2530.

In the Apple version, the day of the week is incorrect after the date is first entered. To correct this, add GOSUB 1670 between the HTAB 5 and the GOSUB 1295 in line 800.

In the TI-99/4A version, the reinput option of the change latitude feature does not work correctly. Change the THEN 2410 at the end of line 2490 to THEN 2460. Also, the DOWN-S in the string in line 500 should read DOWN-N. The TI version states that Extended BASIC is required, but does not mention that expansion memory is also required. TI readers who are interested in modifications necessary to use the progrm without memory expansion should write to COMPUTE! for details. ©

COMPUTE's Author Guide

Most of the following suggestions serve to improve the speed and accuracy of publication. COMPUTE! is primarily interested in new and timely articles on the Commodore 64/128, Atari, Apple, IBM PC/PCjr, Amiga, and Atari ST. We are much more concerned with the content of an article than with its style, but articles should be clear and well-explained.

The guidelines below will permit your good ideas and programs to be more easily edited and published:

1. The upper left corner of the first page should contain your name, address, telephone number, and the date of submission.

2. The following information should appear in the upper right corner of the first page. If your article is specifically directed to one make of computer, please state the brand name and, if applicable, the BASIC or ROM or DOS version(s) involved. In addition, *please indicate the memory requirements of programs.*

3. The underlined title of the article should start about 2/3 of the way down the first page.

4. Following pages should be typed normally, except that in the upper right corner there should be an abbreviation of the title, your last name, and the page number. For example: Memory Map/Smith/2.

5. All lines within the text of the article must be double- or triple-spaced. A one-inch margin should be left at the right, left, top, and bottom of each page. No words should be divided at the ends of lines. And please do not justify. Leave the lines ragged.

6. Standard typing paper should be used (no erasable, onionskin, or other thin paper) and typing should be on one side of the paper only (upper- and lowercase).

7. Sheets should be attached together with a paper clip. Staples should not be used.

8. If you are submitting more than one article, send each one in a separate mailer with its own tape or disk.

9. Short programs (under 20 lines) can easily be included within the text. Longer programs should be separate listings. *It is essential that we have a copy of the program, recorded twice, on a tape or disk.* If your article was written with a word processor, we also appreciate a copy of the text file on the tape or disk. Please use high-quality 10 or 30 minute tapes with the program recorded on both sides. The tape or disk should be labeled with the author's name, the title of the article, and, if applicable, the BASIC/ROM/DOS version(s). Atari tapes should specify whether they are to be LOADed or ENTERed. We prefer to receive Apple programs on disk rather than tape. Tapes are fairly sturdy, but disks need to be enclosed within plastic or cardboard mailers (available at photography, stationery, or computer supply stores).

10. A good general rule is to spell out the numbers zero through ten in your article and write higher numbers as numerals (1024). The exceptions to this are: Figure 5, Table 3, TAB(4), etc. Within ordinary text, however, the zero through ten should appear as words, not numbers. Also, symbols and abbreviations should not be used within text: use "and" (not &), "reference" (not ref.), "through" (not thru).

11. For greater clarity, use all capitals when referring to keys (RETURN, TAB, ESC, SHIFT), BASIC words (LIST, RND, GOTO), and three languages (BASIC, APL, PILOT). Headlines and subheads should, however, be initial caps only, and emphasized words are not capitalized. If you wish to emphasize, underline the word and it will be italicized during typesetting.

12. Articles can be of any length—from a single-line routine to a multi-issue series. The average article is about four to eight double-spaced, typed pages.

13. If you want to include photographs, they should be either 5×7 black and white glossies or color slides.

14. We do not consider articles which are submitted simultaneously to other publishers. If you wish to send an article to another magazine for consideration, please do not submit it to us.

15. COMPUTE! pays between $70 and $800 for published articles. In general, the rate reflects the length and quality of the article. Payment is made upon acceptance. Following submission (Editorial Department, COMPUTE! Magazine, P.O. Box 5406, Greensboro, NC 27403) it will take from four to eight weeks for us to reply. If your work is accepted, you will be notified by a letter which will include a contract for you to sign and return. *Rejected manuscripts are returned to authors who enclose a self-addressed, stamped envelope.*

16. If your article is accepted and you have since made improvements to the program, please submit an entirely new tape or disk and a new copy of the article reflecting the update. We cannot easily make revisions to programs and articles. It is necessary that you send the revised version as if it were a new submission entirely, but be sure to indicate that your submission is a revised version by writing, "Revision" on the envelope and the article.

17. COMPUTE! does not accept unsolicited product reviews. If you are interested in serving on our panel of reviewers, contact the Review Coordinator for details.

COMPUTE!'s Guide To Typing In Programs

Before typing in any program, you should familiarize yourself with your computer. Learn how to use the keyboard to type in and correct BASIC programs. Read your manuals to understand how to save and load BASIC programs to and from your disk drive or cassette unit. Computers are precise—take special care to type the program *exactly* as listed, including any necessary punctuation and symbols, except for special characters as noted below. To help you with this task, we have implemented a special listing convention as well as a program to help check your typing—the "Automatic Proofreader." Please read the following notes before typing in any programs from COMPUTE!. They can save you a lot of time and trouble.

Commodore, Apple, and Atari programs can contain some hard-to-read (and hard-to-type) special characters, so we have developed a listing system that indicates the function of these control characters. (There are no special control characters in our IBM or TI-99/4A listings.) You will find Commodore and Atari special characters within curly braces; *do not type the braces.* For example, {CLEAR} or {CLR} instructs you to insert the symbol which clears the screen on the Atari or Commodore machines. For Commodore, Apple, and Atari, a symbol by itself within curly braces is usually a control key or graphics key. If you see {A}, hold down the CTRL key and press A. This will produce a reverse video character on the Commodore (in quote mode), a graphics character on the Atari, and an invisible control character on the Apple. Commodore computers also have a special control key labeled with the Commodore logo. Graphics characters entered with the Commodore logo key are enclosed in a special bracket that looks like this: [<A>]. In this case, you would hold down the Commodore logo key as you type A. Our Commodore listings are in uppercase, so shifted symbols are underlined. A graphics heart symbol (SHIFT-S) would be listed as S. One exception is {SHIFT-SPACE}. When you see this, hold down SHIFT and press the space bar. If a number precedes a symbol, such as {5 RIGHT}, {6 S}, or [<8 Q>], you would enter five cursor rights, six shifted S's, or eight Commodore-Q's. On the Atari, inverse characters (printed in white on black) should be entered after pressing the inverse video key.

Since spacing is sometimes important, any more than two spaces will be listed. For example, {6 SPACES} means to press the space bar six times. Our listings never leave a space at the end of a line, instead moving it to the next printed line as {SPACE}. For your convenience, we have prepared this quick-reference chart for the Commodore and Atari special characters:

Atari 400/800/XL/XE

When you see	Type	See	
{CLEAR}	ESC SHIFT <		Clear Screen
{UP}	ESC CTRL -		Cursor Up
{DOWN}	ESC CTRL =		Cursor Down
{LEFT}	ESC CTRL +		Cursor Left
{RIGHT}	ESC CTRL *		Cursor Right
{BACK S}	ESC DELETE		Backspace
{DELETE}	ESC CTRL DELETE		Delete character
{INSERT}	ESC CTRL INSERT		Insert character
{DEL LINE}	ESC SHIFT DELETE		Delete line
{INS LINE}	ESC SHIFT INSERT		Insert line
{TAB}	ESC TAB		TAB key
{CLR TAB}	ESC CTRL TAB		Clear tab
{SET TAB}	ESC SHIFT TAB		Set tab stop
{BELL}	ESC CTRL 2		Ring buzzer
{ESC}	ESC ESC		ESCape key

Commodore PET/CBM/VIC/64/128/16/+4

When You Read:	Press:	See:
{CLR}	SHIFT CLR/HOME	
{HOME}	CLR/HOME	
{UP}	SHIFT ↑ CRSR ↓	
{DOWN}	↑ CRSR ↓	
{LEFT}	SHIFT ← CRSR →	
{RIGHT}	← CRSR →	
{RVS}	CTRL 9	
{OFF}	CTRL 0	
{BLK}	CTRL 1	
{WHT}	CTRL 2	
{RED}	CTRL 3	
{CYN}	CTRL 4	
{PUR}	CTRL 5	
{GRN}	CTRL 6	
{BLU}	CTRL 7	
{YEL}	CTRL 8	

When You Read:	Press:	See:
{ 1 }	COMMODORE 1	
{ 2 }	COMMODORE 2	
{ 3 }	COMMODORE 3	
{ 4 }	COMMODORE 4	
{ 5 }	COMMODORE 5	
{ 6 }	COMMODORE 6	
{ 7 }	COMMODORE 7	
{ 8 }	COMMODORE 8	
{ F1 }	f1	
{ F2 }	SHIFT f1	
{ F3 }	f3	
{ F4 }	SHIFT f3	
{ F5 }	f5	
{ F6 }	SHIFT f5	
{ F7 }	f7	
{ F8 }	SHIFT f7	
←	←	

The Automatic Proofreader

We have developed a series of simple, yet effective programs that can help check your typing. Type in the appropriate Proofreader program listed below, then save it for future use. On the VIC, 64, or Atari, run the Proofreader to activate it, then enter NEW to erase the BASIC loader (the Proofreader remains active, hidden in memory, as a machine language program). Pressing RUN/STOP-RESTORE or SYSTEM RESET deactivates the Proofreader. You can use SYS 886 to reactivate the VIC/64 Proofreader, or PRINT USR(1536) to reenable the Atari Proofreader. On the Apple, the Proofreader automatically erases the BASIC portion of itself after you activate it by typing RUN, leaving only the machine language portion in memory. It works with either DOS 3.3 or ProDOS. Disable the Apple Proofreader by pressing CTRL-RESET before running another BASIC program. The IBM Proofreader is a BASIC program that simulates the IBM BASIC line editor, letting you enter, edit, list, save, and load programs that you type. Type RUN to activate.

Once the Proofreader is active, try typing in a line. As soon as you press RETURN, either a decimal number (on the Commodore), a hexadecimal number (on the Apple), or a pair of letters (on the Atari or IBM) appears. The number or pair of letters is called a *checksum*. Try making a change in the line, and notice how the checksum changes.

All you need to do is compare the value provided by the Proofreader with the checksum printed in the program listing in the magazine. In Commodore listings, the checksum is a number from 0 to 255. It is set off from the rest of the line with *rem*. This prevents a syntax error if the checksum is typed in, but the REM statements and checksums need *not* be typed in. It is just there for your information.

In Atari, Apple, and IBM listings, the checksum is given to the left of each line number. Just type in the program one line at a time (without the printed checksum) and compare the checksum generated by the Proofreader to the checksum in the listing. If they match, go on to the next line. If not, check your typing: You've made a mistake. On the Commodore, Atari, and Apple Proofreaders, spaces are not counted as part of the checksum, so be sure you type the right number of spaces between quote marks. The Commodore and Atari Proofreaders do not check to see that you've typed the characters in the right order, so if characters are transposed, the checksum still matches the listing. Because of the checksum method used, do not type abbreviations, such as ? for PRINT. The IBM Proofreader is the pickiest of all; it *will* detect errors in spacing and transposition. Be sure to leave Caps Lock on, except when typing lowercase characters.

IBM Proofreader Commands

Since the IBM Proofreader replaces the computer's normal BASIC line editor, it has to include many of the direct-mode IBM BASIC commands. The syntax is identical to IBM BASIC. Commands simulated are LIST, LLIST, NEW, FILES, SAVE, and LOAD. When listing your program, press any key (except Ctrl-Break) to stop the listing. If you type NEW, the Proofreader prompts you to press Y to be sure you mean yes.

Two new commands are BASIC and CHECK. BASIC exits the Proofreader back to IBM BASIC, leaving the Proofreader in memory. CHECK works just like LIST, but shows the checksums along with the listing. After you have typed in a program, save it to disk. Then exit the Proofreader with the BASIC command, and load the program in BASIC as usual (this replaces the Proofreader in memory). You can now run the program, but you may want to resave it to disk. The version of your program that you resave from BASIC will take up less space on disk and will load faster, but it can no longer be edited with the Proofreader. If you want to convert a program to Proofreader format, save it to disk with SAVE *"filename"*,A.

Special Proofreader Notes For Commodore Cassette Users

The Proofreader resides in a section of memory called the cassette buffer, which is used during tape LOADs and SAVEs. Therefore, be sure to press RUN/STOP-RESTORE to get the Proofreader out of the way before saving or loading a program. If you want to use the Proofreader with tape, run the Proofreader, then enter these two lines *exactly* as shown, pressing RETURN after each one:

```
A$="PROOFREADER.T":B$="{10
 SPACES}":FOR X=1 TO 4:A$=A$
 +B$:NEXT
FOR X=886 TO 1018:A$=A$+CHR$
 (PEEK(X)):NEXT:OPEN 1,1,1,A$:
 CLOSE1
```

Then insert a blank tape and press RECORD and PLAY to save a special version of the Proofreader. Anytime you need to reload the Proofreader after it has been erased—for example, after you reload a paritally completed program—just rewind the tape, type OPEN1:CLOSE1, then press PLAY. You'll see the message FOUND PROOFREADER.T, but not the familiar LOADING message. Don't worry; the Proofreader is in memory. When READY comes back, enter SYS 886.

Program 1: VIC/64 Proofreader

By Charles Brannon, Program Editor

```
10 PRINT"{CLR}PLEASE WAIT...":
   FORI=886TO1018:READA:CK=CK+
   A:POKEI,A:NEXT
20 IF CK<>17539 THEN PRINT"
   {DOWN}YOU MADE AN ERROR":PR
   INT"IN DATA STATEMENTS.":EN
   D
30 SYS886:PRINT"{CLR}{2 DOWN}P
   ROOFREADER ACTIVATED.":NEW
40 DATA 173,036,003,201,150,20
   8,001,096,141,151,003,173
50 DATA 037,003,141,152,003,16
   9,150,141,036,003,169,003
60 DATA 141,037,003,169,000,13
   3,254,096,032,087,241,133
70 DATA 251,134,252,132,253,00
   8,201,013,240,017,201,032
80 DATA 240,005,024,101,254,13
   3,254,165,251,166,252,164
90 DATA 253,040,096,169,013,03
   2,210,255,165,214,141,251
100 DATA 003,206,251,003,169,0
    00,133,216,169,019,032,210
110 DATA 255,169,018,032,210,2
    55,169,58,032,210,255,166
120 DATA 254,169,000,133,254,1
    72,151,003,192,087,208,006
130 DATA 032,205,189,076,235,0
    03,032,205,221,169,032,032
140 DATA 210,255,032,210,255,1
    73,251,003,133,214,076,173
150 DATA 003
```

Program 2: Atari Proofreader

By Charles Brannon, Program Editor

```
100 GRAPHICS 0
110 FOR I=1536 TO 1700:RE
    AD A:POKE I,A:CK=CK+A
    :NEXT I
120 IF CK<>19072 THEN ? "
    Error in DATA Stateme
    nts.  Check Typing.":
    END
130 A=USR(1536)
140 ? :? "Automatic Proof
    reader Now Activated.
    "
150 END
160 DATA 104,160,0,185,26
    ,3,201,69,240,7
170 DATA 200,200,192,34,2
    08,243,96,200,169,74
180 DATA 153,26,3,200,169
    ,6,153,26,3,162
190 DATA 0,189,0,228,157,
    74,6,232,224,16
200 DATA 208,245,169,93,1
    41,78,6,169,6,141
210 DATA 79,6,24,173,4,22
    8,105,1,141,95
```

```
220 DATA 6,173,5,228,105,
    0,141,96,6,169
230 DATA 0,133,203,96,247
    ,238,125,241,93,6
240 DATA 244,241,115,241,
    124,241,76,205,238
250 DATA 0,0,0,0,0,32,62,
    246,8,201
260 DATA 155,240,13,201,3
    2,240,7,72,24,101
270 DATA 203,133,203,104,
    40,96,72,152,72,138
280 DATA 72,160,0,169,128
    ,145,88,200,192,40
290 DATA 208,249,165,203,
    74,74,74,74,24,105
300 DATA 161,160,3,145,88
    ,165,203,41,15,24
310 DATA 105,161,200,145,
    88,169,0,133,203,104
320 DATA 170,104,168,104,
    40,96
```

Program 3: IBM Proofreader

By Charles Brannon, Program Editor

```
MC 10 'Automatic Proofreader Ver
      sion 3.0 (Lines 205,206 ad
      ded/190 deleted/470,490 ch
      anged from V2.0)
LD 100 DIM L$(500),LNUM(500):COL
       OR 0,7,7:KEY OFF:CLS:MAX=
       0:LNUM(0)=65536!
PK 110 ON ERROR GOTO 120:KEY 15,
       CHR$(4)+CHR$(70):ON KEY(1
       5) GOSUB 640:KEY (15) ON:
       GOTO 130
BE 120 RESUME 130
BJ 130 DEF SEG=&H40:W=PEEK(&H4A)
IH 140 ON ERROR GOTO 650:PRINT:P
       RINT"Proofreader Ready."
KB 150 LINE INPUT L$:Y=CSRLIN-IN
       T(LEN(L$)/W)-1:LOCATE Y,1
CA 160 DEF SEG=0:POKE 1050,30:PO
       KE 1052,34:POKE 1054,0:PO
       KE 1055,79:POKE 1056,13:P
       OKE 1057,28:LINE INPUT L$
       :DEF SEG:IF L$="" THEN 15
       0
BC 170 IF LEFT$(L$,1)=" " THEN L
       $=MID$(L$,2):GOTO 170
NN 180 IF VAL(LEFT$(L$,2))=0 AND
        MID$(L$,3,1)=" " THEN L$
       =MID$(L$,4)
ND 200 IF ASC(L$)>57 THEN 260 'n
       o line number, therefore
       command
JB 205 BL=INSTR(L$," "):IF BL=0
       THEN BL$=L$:GOTO 206 ELSE
        BL$=LEFT$(L$,BL-1)
GH 206 LNUM=VAL(BL$):TEXT$=MID$(
       L$,LEN(STR$(LNUM))+1)
OG 210 IF TEXT$="" THEN GOSUB 54
       0:IF LNUM=LNUM(P) THEN GO
       SUB 560:GOTO 150 ELSE 150
MB 220 CKSUM=0:FOR I=1 TO LEN(L$
       ):CKSUM=(CKSUM+ASC(MID$(L
       $,I))*I) AND 255:NEXT:LOC
       ATE Y,1:PRINT CHR$(65+CKS
       UM/16)+CHR$(65+(CKSUM AND
        15))+" "+L$
JE 230 GOSUB 540:IF LNUM(P)=LNUM
        THEN L$(P)=TEXT$:GOTO 15
       0 'replace line
CL 240 GOSUB 580:GOTO 150 'inser
       t the line
AD 260 TEXT$="":FOR I=1 TO LEN(L
       $):A=ASC(MID$(L$,I)):TEXT
       $=TEXT$+CHR$(A+32*(A>96 A
       ND A<123)):NEXT
LP 270 DELIMITER=INSTR(TEXT$," "
       ):COMMAND$=TEXT$:ARG$="":
       IF DELIMITER THEN COMMAND
       $=LEFT$(TEXT$,DELIMITER-1
       ):ARG$=MID$(TEXT$,DELIMIT
       ER+1) ELSE DELIMITER=INST
       R(TEXT$,CHR$(34)):IF DELI
       MITER THEN COMMAND$=LEFT$
       (TEXT$,DELIMITER-1):ARG$=
       MID$(TEXT$,DELIMITER)
FC 280 IF COMMAND$<>"LIST" THEN
       410
ID 290 OPEN "scrn:" FOR OUTPUT A
       S #1
LH 300 IF ARG$="" THEN FIRST=0:P
       =MAX-1:GOTO 340
IJ 310 DELIMITER=INSTR(ARG$,"-")
       :IF DELIMITER=0 THEN LNUM
       =VAL(ARG$):GOSUB 540:FIRS
       T=P:GOTO 340
BP 320 FIRST=VAL(LEFT$(ARG$,DELI
       MITER)):LAST=VAL(MID$(ARG
       $,DELIMITER+1))
EC 330 LNUM=FIRST:GOSUB 540:FIRS
       T=P:LNUM=LAST:GOSUB 540:I
       F P=0 THEN P=MAX-1
GD 340 FOR X=FIRST TO P:N$=MID$(
       STR$(LNUM(X)),2)+" "
KA 350 IF CKFLAG=0 THEN A$="":GO
       TO 370
PF 360 CKSUM=0:A$=N$+L$(X):FOR I
       =1 TO LEN(A$):CKSUM=(CKSU
       M+ASC(MID$(A$,I))*I) AND
       255:NEXT:A$=CHR$(65+CKSUM
       /16)+CHR$(65+(CKSUM AND 1
       5))+" "
DO 370 PRINT #1,A$+N$+L$(X)
JJ 380 IF INKEY$<>"" THEN X=P
OF 390 NEXT :CLOSE #1:CKFLAG=0
CA 400 GOTO 130
PD 410 IF COMMAND$="LLIST" THEN
       OPEN "lpt1:" FOR OUTPUT A
       S #1:GOTO 300
BM 420 IF COMMAND$="CHECK" THEN
       CKFLAG=1:GOTO 290
KA 430 IF COMMAND$<>"SAVE" THEN
       450
CL 440 GOSUB 600:OPEN ARG$ FOR O
       UTPUT AS #1:ARG$="":GOTO
       300
OE 450 IF COMMAND$<>"LOAD" THEN
       490
PG 460 GOSUB 600:OPEN ARG$ FOR I
       NPUT AS #1:MAX=0:P=0
KA 470 WHILE NOT EOF(1):LINE INP
       UT #1,L$:BL=INSTR(L$," ")
       :BL$=LEFT$(L$,BL-1):LNUM(
       P)=VAL(BL$):L$(P)=MID$(L$
       ,LEN(STR$(VAL(BL$)))+1):P
       =P+1:WEND
KK 480 MAX=P:CLOSE #1:GOTO 130
GJ 490 IF COMMAND$="NEW" THEN IN
       PUT "Erase program - Are
       you sure";L$:IF LEFT$(L$,
       1)="y" OR LEFT$(L$,1)="Y"
        THEN MAX=0:LNUM(0)=65536
       !:GOTO 130:ELSE 130
CL 500 IF COMMAND$="BASIC" THEN
       COLOR 7,0,0:ON ERROR GOTO
        0:CLS:END
NC 510 IF COMMAND$<>"FILES" THEN
        520
IH 515 IF ARG$="" THEN ARG$="A:"
        ELSE SEL=1:GOSUB 600
IO 517 FILES ARG$:GOTO 130
DD 520 PRINT"Syntax error":GOTO
       130
BO 540 P=0:WHILE LNUM>LNUM(P) AN
       D P<MAX:P=P+1:WEND:RETURN
KM 560 MAX=MAX-1:FOR X=P TO MAX:
       LNUM(X)=LNUM(X+1):L$(X)=L
       $(X+1):NEXT:RETURN
BK 580 MAX=MAX+1:FOR X=MAX TO P+
       1 STEP -1:LNUM(X)=LNUM(X-
       1):L$(X)=L$(X-1):NEXT:L$(
       P)=TEXT$:LNUM(P)=LNUM:RET
       URN
BA 600 IF LEFT$(ARG$,1)<>CHR$(34
       ) THEN 620 ELSE ARG$=MID$
       (ARG$,2)
EE 610 IF RIGHT$(ARG$,1)=CHR$(34
       ) THEN ARG$=LEFT$(ARG$,LE
       N(ARG$)-1)
LA 620 IF SEL=0 AND INSTR(ARG$,"
       .")=0 THEN ARG$=ARG$+".BA
       S"
DD 630 SEL=0:RETURN
HM 640 CLOSE #1:CKFLAG=0:PRINT"S
       topped.":RETURN 150
II 650 PRINT "Error #";ERR:RESUM
       E 150
```

Program 4: Apple Proofreader

By Tim Victor, Editorial Programmer

```
10 C = 0: FOR I = 768 TO 768 +
   68: READ A:C = C + A: POKE I
   ,A: NEXT
20 IF C < > 7258 THEN PRINT "ER
   ROR IN PROOFREADER DATA STAT
   EMENTS": END
30 IF PEEK (190 * 256) < > 76 T
   HEN POKE 56,0: POKE 57,3: CA
   LL 1002: GOTO 50
40 PRINT CHR$ (4);"IN#A$300"
50 POKE 34,0: HOME : POKE 34,1:
    VTAB 2: PRINT "PROOFREADER
   INSTALLED"
60 NEW
100 DATA 216,32,27,253,201,141
110 DATA 208,60,138,72,169,0
120 DATA 72,189,255,1,201,160
130 DATA 240,8,104,10,125,255
140 DATA 1,105,0,72,202,208
150 DATA 238,104,170,41,15,9
160 DATA 48,201,58,144,2,233
170 DATA 57,141,1,4,138,74
180 DATA 74,74,74,41,15,9
190 DATA 48,201,58,144,2,233
200 DATA 57,141,0,4,104,170
210 DATA 169,141,96
```

MLX Machine Language Entry Program For Commodore 64 and Apple

Ottis Cowper, Technical Editor and Tim Victor, Editorial Programmer

"MLX" is a labor-saving utility that allows almost fail-safe entry of machine language programs. The Apple version runs on the II, II+, IIe, and IIc, with either DOS 3.3 or ProDOS.

"MLX" is a new way to enter long machine language (ML) programs without a lot of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter invalid characters or let you continue if there's a mistake in a line. It won't even let you enter a line or digit out of sequence. For the Commodore 64, this new version of MLX was first introduced in the December 1985 issue. No version of 64 MLX published before that date can be used to enter the MLX-format listings in this issue.

Using MLX

Type in and save some copies of whichever version of MLX is appropriate for your computer (you'll want to use it to enter future ML programs from COMPUTE!). Program 1 is for the Commodore 64, and Program 2 is for the Apple. For Apple MLX, it doesn't matter whether you save the program on a disk formatted for DOS 3.3 or ProDOS. Programs entered with Apple MLX, however, must be saved to a disk formatted with the same operating system as MLX itself. If you have an Apple IIe or IIc, make sure that the key marked *Caps Lock* is in the down position.

When you're ready to enter an ML program, load and run MLX. It asks you for a starting address and an ending address. These addresses appear in the article accompanying the MLX-format program listing you're typing. If you're unfamiliar with machine language, the addresses (and all other values you enter in MLX) may appear strange. Instead of the usual decimal numbers you're accustomed to, these numbers are in *hexadecimal*—a base 16 numbering system commonly used by ML programmers. Hexadecimal—hex for short—includes the numerals 0–9 and the letters A–F. But don't worry—even if you know nothing about ML or hex, you should have no trouble using MLX.

After you enter the starting and ending addresses, the 64 version will offer you the option of clearing the workspace. Choose this option if you're starting to enter a new listing. If you're continuing a listing that's partially typed from a previous session, don't choose this option.

A functions menu will appear. The first option in the menu is ENTER DATA. If you're just starting to type in a program, pick this. Press the E key, and type the first number in the first line of the program listing. If you've already typed in part of a program, type the line number where you left off typing at the end of the previous session. In any case, make sure the address you enter corresponds to the address of a line in the listing you are entering. Otherwise, you'll be unable to enter the data correctly. In the 64 version, if you pressed E by mistake, you can return to the command menu by pressing RETURN alone when asked for the address. (You can get back to the menu from most options by pressing RETURN with no other input.)

Once you're in Enter mode, MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight data bytes and a checksum. Although an MLX-format listing appears similar to the "hex dump" machine language listings you may be accustomed to, the extra checksum number on the end allows MLX to check your typing. (Apple users *can* enter the data from an MLX listing using the built-in monitor if the rightmost column of data is omitted, but we recommend against it. It's much easier to let MLX do the proofreading and error checking for you.)

When you enter a line, MLX recalculates the checksum from the eight bytes and the address and compares this value to the number from the ninth column. If the values match, the data is added to the workspace area, and the prompt for the next line of data appears (the 64 version gives a pleasant beep to indicate that the line was entered correctly). But if MLX detects a typing error, you'll be notified of the mistake. The 64 version will sound a low buzz and display an error message, then redisplay the line for editing. Apple MLX sounds a beep to alert you of the error and then erases the incorrect line and prompts you to reenter it correctly.

After you have entered the last number on the last line of the listing, the Apple version will return to the command menu. At this point you should immediately choose the option S to save your data. The 64 version automatically moves to the Save option after the last number is entered.

Invalid Characters Banned

In 64 MLX, only a few keys are active while you're entering data, so you may have to unlearn some habits. You *do not* type spaces between the columns; the new MLX automatically inserts these for you. You *do not* press RETURN after typing the last number in a line; the new MLX automatically enters and checks the line after you type the last digit.

Apple MLX is fairly flexible about how you type in the numbers. You can put extra spaces between numbers or leave the spaces out entirely, compressing a line into 18 keypresses. But be careful not to put a space between two digits in the middle of a number. MLX will read two single-digit numbers instead of one two-digit number (F 6 means F *and* 6, not F6). You must press RETURN to enter the line.

Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), nothing happens (the 64 version gives a warning buzz to indicate an invalid keypress). Even better, MLX checks for transposed characters. If you're supposed to type in *A0* and instead enter *0A*, MLX will catch your mistake.

Editing Features

To correct typing mistakes before finishing a line in the 64 version, use the INST/DEL key to delete the character to the left of the cursor. (The cursor-left key also deletes.) If you mess up a line really badly, press CLR/HOME to start the line over. The RETURN key is also active, but only before any data is typed on a line. Pressing RETURN at this point returns you to the command menu. After you type a character of data, MLX disables RETURN until the cursor returns to the start of a line. Remember, you can press CLR/HOME to quickly get to a line number prompt.

More editing features are available when correcting lines in which 64 MLX has detected an error. To make corrections in a line that MLX has redisplayed for editing, compare the line on the

screen with the one printed in the listing, then move the cursor to the mistake and type the correct key. The cursor left and right keys provide the normal cursor controls. (The INST/DEL key now works as an alternative cursor-left key.) You cannot move left beyond the first character in the line. If you try to move beyond the rightmost character, you'll reenter the line. During editing, RETURN is active; pressing it tells MLX to recheck the line. You can press the CLR/HOME key to clear the entire line if you want to start from scratch, or if you want to get to a line number prompt to use RETURN to get back to the menu.

Apple MLX also includes some editing features. The left- and right-arrow keys allow you to back up and go forward on the line you're entering so that you can retype data. Pressing the CONTROL (CTRL) and D keys at the same time (*delete*) removes the character under the cursor, shortening the line by one character. Pressing CONTROL-I (*insert*) puts a space under the cursor and shifts the rest of the line to the right, making the line one character longer. If the cursor is at the right end of the line, neither CONTROL-D nor CONTROL-I has any effect. To leave Enter mode, press the RETURN key when MLX prompts you with a new line address.

Display Data

The second menu choice, DISPLAY DATA, examines memory and shows the contents in the same format as the program listing (including the checksum). When you press D, MLX asks you for a starting address. Be sure that the starting address you give corresponds to a line number in the listing. Otherwise, the checksum display will be meaningless. MLX displays program lines until it reaches the end of the program, at which point the menu is redisplayed. With Apple MLX, you can stop the display and return to the menu by pressing any key. The 64 version allows you to stop the display and get back to the menu by pressing RETURN, or to pause the display by pressing the space bar (press space again to restart the display).

Other Menu Options

Two more menu selections let you save programs and load them back into the computer. These are SAVE FILE (SAVE DATA in the 64 version) and LOAD FILE; their operation is quite straightforward. When you press S or L, MLX asks you for the filename. The 64 version will follow this by asking you to press either D or T to select disk or tape.

Those using the 64 version will notice the disk drive starting and stopping several times during a load or save. Don't panic; this is normal behavior. MLX opens and reads from or writes to the file instead of using the usual LOAD and SAVE commands. Disk users should also note that the drive prefix 0: is automatically added to the filename (line 750), so this should *not* be included when entering the name. (This also precludes the use of @ for Save-with-Replace, so remember to give each version you save a different name.)

Remember that MLX saves the entire workspace area from the starting address to the ending address, so the save or load may take longer than you might expect if you've entered only a small amount of data from a long listing. When saving a partially completed listing, make sure to note the address where you stopped typing so you'll know where to resume entry when you reload.

MLX reports any errors detected during the save or load. For the 64 version, the standard disk or tape error messages will be displayed. (Tape users should bear in mind that the Commodore 64 is never able to detect errors when saving to tape.) The 64 version also has three special load error messages: INCORRECT STARTING ADDRESS, which means the file you're trying to load does not have the starting address you specified when you ran MLX; LOAD ENDED AT *address*, which means the file you're trying to load ends before the ending address you specified when you started MLX; and TRUNCATED AT ENDING ADDRESS, which means the file you're trying to load extends beyond the ending address you specified when you started MLX. If you see one of these messages and feel certain that you've loaded the right file, exit and rerun MLX, being careful to enter the correct starting and ending addresses.

The Apple version simply displays the message DISK ERROR if a problem is detected during a Save or Load. If you're not sure why a disk error has occurred, check the drive. Make sure there's a formatted disk in the drive and that it was formatted by the same operating system you're using for MLX (ProDOS or DOS 3.3). If you're trying to save a file and see an error message, the disk might be full. Either save the file on another disk or quit MLX (by pressing the Q key), delete an old file or two, then run MLX again. Your typing should still be safe in memory. If the error message appears during a Load, you may have specified a filename that doesn't exist on the disk.

The Quit menu option has the obvious effect—it stops MLX and enters BASIC. In the 64 version the RUN/STOP key is disabled, so the Q option lets you exit the program without turning off the computer. (Of course, RUN/STOP–RESTORE for the 64 or CONTROL-RESET for the Apple also gets you out.) The 64 version will ask for verification; press Y to exit to BASIC, or any other key to return to the menu. After quitting, you can type RUN again and reenter MLX without losing your data, as long as you don't use the clear workspace option in 64 MLX.

The Finished Product

When you've finished typing all the data for an ML program and saved your work, you're ready to see the results. The instructions for loading and using the finished product vary from program to program. Some Commodore 64 ML programs are designed to be loaded and run like BASIC programs, so all you need to type is LOAD *"filename"*,8 for disk or LOAD *"filename"* for tape, and then RUN. (Such programs usually have 0801 as their MLX starting address.) Others must be reloaded to specific addresses with a command such as LOAD *"filename"*,8,1 for disk or LOAD *"filename"*,1,1 for tape, then started with a SYS to a particular memory address. (On the Commodore 64, the most common starting address for such programs is 49152, which corresponds to MLX address C000.) In either case, you should always refer to the article which accompanies the ML listing for information on loading and running the program. For the Apple, you need to BRUN the program, or you may BLOAD and start the program with a CALL. Again, refer to the article accompanying the machine language program for instructions.

An Ounce Of Prevention

By the time you finish typing in the data for a long ML program, you'll have several hours invested in the project. Don't take chances—use our "Automatic Proofreader" to type the new MLX, and then test your copy *thoroughly* before first using it to enter any significant amount of data. Make sure all the menu options work as they should. Enter fragments of the program starting at several different addresses, then use the Display option to verify that the data has been entered correctly. And be sure to test the Save and Load options several times to ensure that you can recall your work from disk or tape. Don't let a simple typing error in the new MLX cost you several nights of hard work.

In the Apple version, line 100 traps all errors to line 610. If MLX is typed in correctly, then only disk errors should normally be encountered. A disk error

message when you're not trying to access the drive—for example, when you first start entering data—indicates a typing error in the MLX program itself. If this occurs, hit CONTROL-RESET to break out of MLX and carefully compare your entry against the printed listing.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: MLX For Commodore 64

Version by Ottis Cowper, Technical Editor

```
100 POKE 56,50:CLR:DIM IN$,I,J
    ,A,B,A$,B$,A(7),N$ :rem 34
110 C4=48:C6=16:C7=7:Z2=2:Z4=2
    54:Z5=255:Z6=256:Z7=127
                       :rem 238
120 FA=PEEK(45)+Z6*PEEK(46):BS
    =PEEK(55)+Z6*PEEK(56):H$="
    0123456789ABCDEF" :rem 118
130 R$=CHR$(13):L$="{LEFT}":S$
    =" ":D$=CHR$(20):Z$=CHR$(0
    ):T$="{13 RIGHT}" :rem 173
140 SD=54272:FOR I=SD TO SD+23
    :POKE I,0:NEXT:POKE SD+24,
    15:POKE 788,52     :rem 194
150 PRINT"{CLR}"CHR$(142)CHR$(
    8):POKE 53280,15:POKE 5328
    1,15               :rem 104
160 PRINT T$" {RED}{RVS}
    {2 SPACES}{8 @}{2 SPACES}"
    SPC(28)"{2 SPACES}{OFF}
    {BLU} MLX II {RED}{RVS}
    {2 SPACES}"SPC(28)"
    {12 SPACES}{BLU}" :rem 121
170 PRINT"{3 DOWN}{3 SPACES}CO
    MPUTE!'S MACHINE LANGUAGE
    {SPACE}EDITOR{3 DOWN}"
                       :rem 135
180 PRINT"{BLK}STARTING ADDRES
    S{4}";:GOSUB300:SA=AD:GOSU
    B1040:IF F THEN180:rem 113
190 PRINT"{BLK}{2 SPACES}ENDIN
    G ADDRESS{4}";:GOSUB300:EA
    =AD:GOSUB1030:IF F THEN190
                       :rem 173
200 INPUT"{3 DOWN}{BLK}CLEAR W
    ORKSPACE [Y/N]{4}";A$:IF L
    EFT$(A$,1)<>"Y"THEN220
                       :rem 9
210 PRINT"{2 DOWN}{BLU}WORKING
    ...";:FORI=BS TO BS+EA-SA+
    7:POKE I,0:NEXT:PRINT"DONE
    "                  :rem 139
220 PRINTTAB(10)"{2 DOWN}{BLK}
    {RVS} MLX COMMAND MENU
    {DOWN}{4}":PRINT T$"{RVS}E
    {OFF}NTER DATA"    :rem 62
230 PRINT T$"{RVS}D{OFF}ISPLAY
     DATA":PRINT T$"{RVS}L
    {OFF}OAD DATA"     :rem 19
240 PRINT T$"{RVS}S{OFF}AVE FI
    LE":PRINT T$"{RVS}Q{OFF}UI
    T{2 DOWN}{BLK}"    :rem 238
250 GET A$:IF A$=N$ THEN250
                       :rem 127
260 A=0:FOR I=1 TO 5:IF A$=MID
    $("EDLSQ",I,1)THEN A=I:I=5
                       :rem 42
270 NEXT:ON A GOTO420,610,690,
    700,280:GOSUB1060:GOTO250
                       :rem 97
280 PRINT"{RVS} QUIT ":INPUT"
    {DOWN}{4}ARE YOU SURE [Y/N
    ]";A$:IF LEFT$(A$,1)<>"Y"T
    HEN220             :rem 189
290 POKE SD+24,0:END   :rem 95
300 IN$=N$:AD=0:INPUTIN$:IFLEN
    (IN$)<>4THENRETURN :rem 31
310 B$=IN$:GOSUB320:AD=A:B$=MI
    D$(IN$,3):GOSUB320:AD=AD*2
    56+A:RETURN        :rem 225
320 A=0:FOR J=1 TO 2:A$=MID$(B
    $,J,1):B=ASC(A$)-C4+(A$>"@
    ")*C7:A=A*C6+B     :rem 143
330 IF B<0 OR B>15 THEN AD=0:A
    =-1:J=2            :rem 132
340 NEXT:RETURN        :rem 240
350 B=INT(A/C6):PRINT MID$(H$,
    B+1,1);:B=A-B*C6:PRINT MID
    $(H$,B+1,1);:RETURN:rem 42
360 A=INT(AD/Z6):GOSUB350:A=AD
    -A*Z6:GOSUB350:PRINT":";
                       :rem 32
370 CK=INT(AD/Z6):CK=AD-Z4*CK+
    Z5*(CK>Z7):GOTO390:rem 131
380 CK=CK*Z2+Z5*(CK>Z7)+A
                       :rem 168
390 CK=CK+Z5*(CK>Z5):RETURN
                       :rem 159
400 PRINT"{DOWN}STARTING AT{4}
    ";:GOSUB300:IF IN$<>N$ THE
    N GOSUB1030:IF F THEN400
                       :rem 75
410 RETURN             :rem 117
420 PRINT"{RVS} ENTER DATA ":G
    OSUB400:IF IN$=N$ THEN220
                       :rem 85
430 OPEN3,3:PRINT      :rem 34
440 POKE198,0:GOSUB360:IF F TH
    EN PRINT IN$:PRINT"{UP}
    {5 RIGHT}";        :rem 6
450 FOR I=0 TO 24 STEP 3:B$=S$
    :FOR J=1 TO 2:IF F THEN B$
    =MID$(IN$,I+J,1)   :rem 226
460 PRINT"{RVS}"B$L$;:IF I<24T
    HEN PRINT"{OFF}";  :rem 15
470 GET A$:IF A$=N$ THEN470
                       :rem 135
480 IF(A$>"/"ANDA$<":")OR(A$>"
    @"ANDA$<"G")THEN540
                       :rem 100
490 IF A$=R$ AND((I=0)AND(J=1)
    OR F)THEN PRINT B$;:J=2:NE
    XT:I=24:GOTO550    :rem 46
500 IF A$="{HOME}" THEN PRINT
    {SPACE}B$:J=2:NEXT:I=24:NE
    XT:F=0:GOTO440     :rem 66
510 IF(A$="{RIGHT}")ANDF THENP
    RINT B$L$;:GOTO540:rem 107
520 IF A$<>L$ AND A$<>D$ OR((I
    =0)AND(J=1))THEN GOSUB1060
    :GOTO470           :rem 232
530 A$=L$+S$+L$:PRINT B$L$;:J=
    2-J:IF J THEN PRINT L$;:I=
    I-3                :rem 12
540 PRINT A$;:NEXT J:PRINT S$;
                       :rem 2
550 NEXT I:PRINT:PRINT"{UP}
    {5 RIGHT}";:INPUT#3,IN$:IF
     IN$=N$ THEN CLOSE3:GOTO22
    0                  :rem 106
560 FOR I=1 TO 25 STEP3:B$=MID
    $(IN$,I):GOSUB320:IF I<25
    {SPACE}THEN GOSUB380:A(I/3
    )=A                :rem 81
570 NEXT:IF A<>CK THEN GOSUB10
    60:PRINT"{BLK}{RVS} ERROR:
     REENTER LINE {4}":F=1:GOT
    O440               :rem 161
580 GOSUB1080:B=BS+AD-SA:FOR I
    =0 TO 7:POKE B+I,A(I):NEXT
                       :rem 245
590 AD=AD+8:IF AD>EA THEN CLOS
    E3:PRINT"{DOWN}{BLU}** END
     OF ENTRY **{BLK}{2 DOWN}"
    :GOTO700           :rem 207
600 F=0:GOTO440        :rem 84
610 PRINT"{CLR}{DOWN}{RVS} DIS
    PLAY DATA ":GOSUB400:IF IN
    $=N$ THEN220       :rem 146
620 PRINT"{DOWN}{BLU}PRESS:
    {RVS}SPACE{OFF} TO PAUSE,
    {SPACE}{RVS}RETURN{OFF} TO
     BREAK{4}{DOWN}"   :rem 241
630 GOSUB360:B=BS+AD-SA:FORI=B
    TO B+7:A=PEEK(I):GOSUB350:
    GOSUB380:PRINT S$; :rem 56
640 NEXT:PRINT"{RVS}";:A=CK:GO
    SUB350:PRINT       :rem 144
650 F=1:AD=AD+8:IF AD>EA THENP
    RINT"{DOWN}{BLU}** END OF
    {SPACE}DATA **":GOTO220
                       :rem 170
660 GET A$:IF A$=R$ THEN GOSUB
    1080:GOTO220       :rem 65
670 IF A$=S$ THEN F=F+1:GOSUB1
    080                :rem 28
680 ONFGOTO630,660,630:rem 224
690 PRINT"{DOWN}{RVS} LOAD DAT
    A ":OP=1:GOTO710   :rem 31
700 PRINT"{DOWN}{RVS} SAVE FIL
    E ":OP=0           :rem 32
710 IN$=N$:INPUT"{DOWN}FILENAM
    E{4}";IN$:IF IN$=N$ THEN22
    0                  :rem 229
720 F=0:PRINT"{DOWN}{BLK}{RVS}
    T{OFF}APE OR {RVS}D{OFF}IS
    K: {4}";           :rem 66
730 GET A$:IF A$="T"THEN PRINT
    "T{DOWN}":GOTO880  :rem 90
740 IF A$<>"D"THEN730  :rem 90
750 PRINT"D{DOWN}":OPEN15,8,15
    ,"I0:":B=EA-SA:IN$="0:"+IN
    $:IF OP THEN810    :rem 163
760 OPEN 1,8,8,IN$+",P,W":GOSU
    B860:IF A THEN220  :rem 66
770 AH=INT(SA/256):AL=SA-(AH*2
    56):PRINT#1,CHR$(AL);CHR$(
    AH);               :rem 221
780 FOR I=0 TO B:PRINT#1,CHR$(
    PEEK(BS+I));:IF ST THEN800
                       :rem 171
790 NEXT:CLOSE1:CLOSE15:GOTO94
    0                  :rem 230
800 GOSUB1060:PRINT"{DOWN}
    {BLK}ERROR DURING SAVE:{4}
    ":GOSUB860:GOTO220 :rem 61
810 OPEN 1,8,8,IN$+",P,R":GOSU
    B860:IF A THEN220  :rem 57
820 GET#1,A$,B$:AD=ASC(A$+Z$)+
    256*ASC(B$+Z$):IF AD<>SA T
    HEN F=1:GOTO850    :rem 155
830 FOR I=0 TO B:GET#1,A$:POKE
     BS+I,ASC(A$+Z$):IF ST AND
    (I<>B)THEN F=2:AD=I:I=B
                       :rem 180
840 NEXT:IF ST<>64 THEN F=3
                       :rem 20
850 CLOSE1:CLOSE15:ON ABS(F>0)
    +1 GOTO960,970     :rem 12
860 INPUT#15,A,A$:IF A THEN CL
    OSE1:CLOSE15:GOSUB1060:PRI
    NT"{RVS}ERROR: "A$:rem 114
870 RETURN             :rem 127
880 POKE183,PEEK(FA+2):POKE187
    ,PEEK(FA+3):POKE188,PEEK(F
    A+4):IFOP=0THEN920:rem 178
890 SYS 63466:IF(PEEK(783)AND1
    )THEN GOSUB1060:PRINT"
    {DOWN}{RVS} FILE NOT FOUND
```

```
":GOTO690          :rem 34
900 AD=PEEK(829)+256*PEEK(830)
:IF AD<>SA THEN F=1:GOTO97
0                  :rem 201
910 A=PEEK(831)+256*PEEK(832)-
1:F=F-2*(A<EA)-3*(A>EA):AD
=A-AD:GOTO930      :rem 75
920 A=SA:B=EA+1:GOSUB1010:POKE
780,3:SYS 63338    :rem 107
930 A=BS:B=BS+(EA-SA)+1:GOSUB1
010:ON OP GOTO950:SYS 6359
1                  :rem 38
940 GOSUB1080:PRINT"{BLU}** SA
VE COMPLETED **":GOTO220
                   :rem 139
950 POKE147,0:SYS 63562:IF ST<
>64 THEN970        :rem 39
960 GOSUB1080:PRINT"{BLU}** LO
AD COMPLETED **":GOTO220
                   :rem 126
970 GOSUB1060:PRINT"{BLK}{RVS}
ERROR DURING LOAD:{DOWN}
{4}":ON F GOSUB980,990,100
0:GOTO220          :rem 233
980 PRINT"INCORRECT STARTING A
DDRESS (";:GOSUB360:PRINT"
)":RETURN          :rem 145
990 PRINT"LOAD ENDED AT ";:AD=
SA+AD:GOSUB360:PRINT D$:RE
TURN               :rem 159
1000 PRINT"TRUNCATED AT ENDING
 ADDRESS":RETURN :rem 166
1010 AH=INT(A/256):AL=A-(AH*25
6):POKE193,AL:POKE194,AH
                   :rem 95
1020 AH=INT(B/256):AL=B-(AH*25
6):POKE174,AL:POKE175,AH:
RETURN             :rem 122
1030 IF AD<SA OR AD>EA THEN105
0                  :rem 135
1040 IF(AD>511 AND AD<40960)OR
(AD>49151 AND AD<53248)TH
EN GOSUB1080:F=0:RETURN
                   :rem 104
1050 GOSUB1060:PRINT"{RVS} INV
ALID ADDRESS {DOWN}{BLK}"
:F=1:RETURN        :rem 224
1060 POKE SD+5,31:POKE SD+6,20
8:POKE SD,240:POKE SD+1,4
:POKE SD+4,33      :rem 19
1070 FOR S=1 TO 100:NEXT:GOTO1
090                :rem 90
1080 POKE SD+5,8:POKE SD+6,240
:POKE SD,0:POKE SD+1,90:P
OKE SD+4,17        :rem 182
1090 FOR S=1 TO 100:NEXT:POKE
{SPACE}SD+4,0:POKE SD,0:P
OKE SD+1,0:RETURN  :rem 8
```

Program 2: MLX For Apple

Version by Tim Victor, Editorial Programmer

```
100 N = 9: HOME : NORMAL : PRIN
    T "APPLE MLX": POKE 34,2: O
    NERR  GOTO 610
110 VTAB 1: HTAB 20: PRINT "STA
    RT ADDRESS";: GOSUB 530: IF
     A = 0 THEN  PRINT  CHR$ (7
    ): GOTO 110
120 S = A
130 VTAB 2: HTAB 20: PRINT "END
     ADDRESS  ";: GOSUB 530: IF
     S >  = A OR A = 0 THEN  PR
    INT  CHR$ (7): GOTO 130
140 E = A
150 PRINT : PRINT "CHOOSE: (E)NT
    ER DATA";: HTAB 22: PRINT "
    (D)ISPLAY DATA": HTAB 8: PR
    INT "(L)OAD FILE  (S)AVE FI
    LE  (Q)UIT": PRINT
160 GET A$: FOR I = 1 TO 5: IF
    A$ < > MID$ ("EDLSQ",I,1) T
    HEN  NEXT : GOTO 160
170 ON I GOTO 270,220,180,200:
    POKE 34,0: END
180 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BLOAD";A$;",A";S
190 GOTO 150
200 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BSAVE";A$;",A";S;",L"
    ;E - S
210 GOTO 150
220 GOSUB 590: IF B = 0 THEN 15
    0
230 FOR B = B TO E STEP 8:L = 4
    :A = B: GOSUB 580: PRINT A$
    ;": ";:L = 2
240 FOR F = 0 TO 7:V(F + 1) = P
    EEK (B + F): NEXT : GOSUB 5
    60:V(9) = C
250 FOR F = 1 TO N:A = V(F): GO
    SUB 580: PRINT A$" ";: NEXT
     : PRINT : IF  PEEK (49152)
     < 128 THEN  NEXT
260 POKE 49168,0: GOTO 150
270 GOSUB 590: IF B = 0 THEN 15
    0
280 FOR B = B TO E STEP 8
290 HTAB 1:A = B:L = 4: GOSUB 5
    80: PRINT A$;": ";: CALL 64
    668:A$ = "":P = 0: GOSUB 33
    0: IF L = 0 THEN 150
300 GOSUB 470: IF F < > N THEN
    PRINT CHR$ (7);: GOTO 290
310 IF N = 9 THEN GOSUB 560: IF
     C < > V(9) THEN PRINT CHR$
    (7);: GOTO 290
320 FOR F = 1 TO 8: POKE B + F
    - 1,V(F): NEXT : PRINT : NE
    XT : GOTO 150
330 IF LEN (A$) = 33 THEN A$ =
    O$:P = O: PRINT CHR$ (7);
340 L = LEN (A$):O$ = A$:O = P:
    L$ = "": IF P > 0 THEN L$ =
     LEFT$ (A$,P)
350 R$ = "": IF P < L - 1 THEN
    R$ =  RIGHT$ (A$,L - P - 1)
360 HTAB 7: PRINT L$;: FLASH :
    IF P < L THEN PRINT MID$ (A
    $,P + 1,1);: NORMAL : PRINT
    R$;
370 PRINT " ";: NORMAL
380 K = PEEK (49152): IF K < 12
    8 THEN 380
390 POKE 49168,0:K = K - 128
400 IF K = 13 THEN HTAB 7: PRIN
    T A$;" ";: RETURN
410 IF K = 32 OR K > 47 AND K <
     58 OR K > 64 AND K < 71 TH
    EN A$ = L$ + CHR$ (K) + R$:
    P = P + 1
420 IF K = 4 THEN A$ = L$ + R$
430 IF K = 9 THEN A$ = L$ + " "
     +  MID$ (A$,P + 1,1) + R$
440 IF K = 8 THEN P = P - (P >
    0)
450 IF K = 21 THEN P = P + (P <
     L)
460 GOTO 330
470 F = 1:D = 0: FOR P = 1 TO L
    EN (A$):C$ = MID$ (A$,P,1):
     IF F > N AND C$ < > " " TH
    EN  RETURN
480 IF C$ < > " " THEN  GOSUB 5
    20:V(F) = J + 16 * (D = 1)
    * V(F):D = D + 1
490 IF D > 0 AND C$ = " " OR D
    = 2 THEN D = 0:F = F + 1
500 NEXT : IF D = 0 THEN F = F
    - 1
510 RETURN
520 J = ASC (C$):J = J - 48 - 7
     * (J > 64): RETURN
530 A = 0: INPUT A$:A$ = LEFT$
    (A$,4): IF LEN (A$) = 0 THE
    N RETURN
540 FOR P = 1 TO LEN (A$):C$ =
    MID$ (A$,P,1): IF C$ < "0"
    OR C$ > "9" AND C$ < "A" OR
     C$ > "Z" THEN A = 0: RETUR
    N
550 GOSUB 520:A = A * 16 + J: N
    EXT : RETURN
560 C = INT (B / 256):C = B - 2
    54 * C - 255 * (C > 127):C
    = C - 255 * (C > 255)
570 FOR F = 1 TO 8:C = C * 2 -
    255 * (C > 127) + V(F):C =
    C - 255 * (C > 255): NEXT :
     RETURN
580 I = FRE (0):A$ = "": FOR I
    = 1 TO L:T =  INT (A / 16):
    A$ =  MID$ ("0123456789ABCD
    EF",A - 16 * T + 1,1) + A$:
    A = T: NEXT : RETURN
590 PRINT "FROM ADDRESS ";: GOS
    UB 530: IF S > A OR E < A O
    R A = 0 THEN B = 0: RETURN
600 B = S + 8 * INT ((A - S) /
    8): RETURN
610 PRINT "DISK ERROR": GOTO 15
    0
```


Attention Programmers

COMPUTE! magazine is currently looking for quality articles on Commodore, Atari, Apple, and IBM computers (including the Commodore Amiga and Atari ST). If you have an interesting home application, educational program, programming utility, or game, submit it to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Or write for a copy of our "Writer's Guidelines."

MLX Machine Language Entry Program For Atari

Charles Brannon, Program Editor

MLX is a labor-saving utility that allows almost fail-safe entry of machine language programs published in COMPUTE!. *You need to know nothing about machine language to use MLX—it was designed for everyone.*

"MLX" is a new way to enter long machine language (ML) programs with a minimum of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter illegal characters when you should be typing numbers. It won't let you enter numbers greater than 255 (forbidden in ML). It won't let you enter the wrong numbers on the wrong line. In addition, MLX creates a ready-to-use tape or disk file.

Using MLX

Type in and save MLX (you'll want to use it in the future). When you're ready to type in an ML program, run MLX. MLX asks you for three numbers: the starting address, the ending address, and the run/init address. These numbers are given in the article accompanying the ML program presented in MLX format. You must also choose one of three options for saving the file: as a boot tape, as disk binary file, or as boot disk. The article with the ML program should specify which formats may be used.

When you run MLX, you'll see a prompt corresponding to the starting address. The prompt is the current line you are entering from the listing. It increases by six each time you enter a line. That's because each line has seven numbers—six actual data numbers plus a checksum number. The checksum verifies that you typed the previous six numbers correctly. If you enter any of the six numbers wrong, or enter the checksum wrong, the computer rings a buzzer and prompts you to reenter the line. If you enter it correctly, a bell tone sounds and you continue to the next line.

MLX accepts only numbers as input. If you make a typing error, press the DEL/BACK SPACE; the entire number is deleted. You can press it as many times as necessary back to the start of the line. If you enter three-digit numbers as listed, the computer automatically prints the comma and goes on to accept the next number. If you enter fewer than three digits, you can press the comma key, the space bar, or the RETURN key to advance to the next number. The checksum automatically appears in inverse video for emphasis.

MLX Commands

When you finish typing an ML listing (assuming you type it all in one session), you can then save the completed program on tape or disk. Follow the screen instructions. If you get any errors while saving, you probably have a bad disk, or the disk is full, or you've made a typo when entering the MLX program itself.

You don't have to enter the whole ML program in one sitting. MLX lets you enter as much as you want, save it, and then reload the file from tape or disk later. MLX recognizes these commands:

CTRL-S	**Save**
CTRL-L	**Load**
CTRL-N	**New Address**
CTRL-D	**Display**

To issue a command, hold down the CTRL key (CONTROL on the XL models) and press the indicated key. When you enter a command, MLX jumps out of the line you've been typing, so we recommend you do it at a new prompt. Use the Save command (CTRL-S) to save what you've been working on. It will save on tape or disk, as if you've finished, but the tape or disk won't work, of course, until you finish the typing. Remember to make a note of what address you stop at. The next time you run MLX, answer all the prompts as you did before—regardless of where you stopped typing—then insert the disk or tape. When you get to the line number prompt, press CTRL-L to reload the partly completed file into memory. Then use the New Address command to resume typing.

To use the New Address command, press CTRL-N and enter the address where you previously stopped. The prompt will change, and you can then continue typing. Always enter a New Address that matches up with one of the line numbers in the MLX-format listing, or else the checksum won't work. The Display command lets you display a section of your typing. After you press CTRL-D, enter two addresses within the line number range of the listing. You can break out of the listing display and return to the prompt by pressing any key.

Atari MLX: Machine Language Entry

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
DA 100 GRAPHICS 0:DL=PEEK(56
       0)+256*PEEK(561)+4:PO
       KE DL-1,71:POKE DL+2,
       6
NJ 110 POSITION 8,0:? "MLX":
       POSITION 23,0:? "fast
       safe entry":POKE 710,
       0:?
JK 120 ? "Starting Address";
       :INPUT BEG:? "  Endin
       g Address";:INPUT FIN
       :? "Run/Init Address"
       ;:INPUT STARTADR
DD 130 DIM A(6),BUFFER$(FIN-
       BEG+127),T$(20),F$(20
       ),CIO$(7),SECTOR$(128
       ),DSKINV$(6)
JJ 140 OPEN #1,4,0,"K:":? :?
       ,"Tape or Disk:";
BM 150 BUFFER$=CHR$(0):BUFFE
       R$(FIN-BEG+30)=BUFFER
       $:BUFFER$(2)=BUFFER$:
       SECTOR$=BUFFER$
GC 160 ADDR=BEG:CIO$="hhh":C
       IO$(4)=CHR$(170):CIO$
       (5)="LV":CIO$(7)=CHR$
       (228)
EJ 170 GET #1,MEDIA:IF MEDIA
       <>84 AND MEDIA<>68 TH
       EN 170
PO 180 ? CHR$(MEDIA):? :IF M
       EDIA<>ASC("T") THEN B
       UFFER$="":GOTO 250
PL 190 BEG=BEG-24:BUFFER$=CH
       R$(0):BUFFER$(2)=CHR$
       (INT((FIN-BEG+127)/12
       8))
KF 200 H=INT(BEG/256):L=BEG-
       H*256:BUFFER$(3)=CHR$
       (L):BUFFER$(4)=CHR$(H
       )
EC 210 PINIT=BEG+8:H=INT(PIN
       IT/256):L=PINIT-H*256
       :BUFFER$(5)=CHR$(L):B
       UFFER$(6)=CHR$(H)
PB 220 FOR I=7 TO 24:READ A:
       BUFFER$(I)=CHR$(A):NE
       XT I:DATA 24,96,169,6
       0,141,2,211,169,0,133
       ,10,169,0,133,11,76,0
       ,0
DP 230 H=INT(STARTADR/256):L
       =STARTADR-H*256:BUFFE
       R$(15)=CHR$(L):BUFFER
       $(19)=CHR$(H)
KL 240 BUFFER$(23)=CHR$(L):B
       UFFER$(24)=CHR$(H)
HI 250 IF MEDIA<>ASC("D") TH
       EN 360
DO 260 ? :? "Boot Disk or Bi
       nary File:";
LI 270 GET #1,DTYPE:IF DTYPE
```

```
      <>68 AND DTYPE<>70 TH
      EN 270
GM 280 ? CHR$(DTYPE):IF DTYP
      E=70 THEN 360
PJ 290 BEG=BEG-30:BUFFER$=CH
      R$(0):BUFFER$(2)=CHR$
      (INT((FIN-BEG+127)/12
      8))
KG 300 H=INT(BEG/256):L=BEG-
      H*256:BUFFER$(3)=CHR$
      (L):BUFFER$(4)=CHR$(H
      )
HH 310 PINIT=STARTADR:H=INT(
      PINIT/256):L=PINIT-H*
      256:BUFFER$(5)=CHR$(L
      ):BUFFER$(6)=CHR$(H)
AQ 320 RESTORE 330:FOR I=7 T
      O 30:READ A:BUFFER$(I
      )=CHR$(A):NEXT I
GA 330 DATA 169,0,141,231,2,
      133,14,169,0,141,232,
      2,133,15,169,0,133,10
      ,169,0,133,11,24,96
OB 340 H=INT(BEG/256):L=BEG-
      H*256:BUFFER$(8)=CHR$
      (L):BUFFER$(15)=CHR$(
      H)
DQ 350 H=INT(STARTADR/256):L
      =STARTADR-H*256:BUFFE
      R$(22)=CHR$(L):BUFFER
      $(26)=CHR$(H)
JP 360 GRAPHICS 0:POKE 712,1
      0:POKE 710,10:POKE 70
      9,2
JK 370 ? ADDR;":";:FOR J=1 T
      O 6
NF 380 GOSUB 570:IF N=-1 THE
      N J=J-1:GOTO 380
BF 390 IF N=-19 THEN 720
OI 400 IF N=-12 THEN LET REA
      D=1:GOTO 720
AI 410 TRAP 410:IF N=-14 THE
      N ? :? "New Address";
      :INPUT ADDR:? :GOTO 3
      70
JD 420 TRAP 32767:IF N<>-4 T
      HEN 480
AJ 430 TRAP 430:? :? "Displa
      y:From";:INPUT F:? ,"
      To";:INPUT T:TRAP 327
      67
ML 440 IF F<BEG OR F>FIN OR
      T<BEG OR T>FIN OR T<F
       THEN ? CHR$(253);"At
       least ";BEG;", Not M
      ore Than ";FIN:GOTO 4
      30
MH 450 FOR I=F TO T STEP 6:?
       :? I;":";:FOR K=0 TO
       5:N=PEEK(ADR(BUFFER$
      )+I+K-BEG):T$="000":T
      $(4-LEN(STR$(N)))=STR
      $(N)
MA 460 IF PEEK(764)<255 THEN
       GET #1,A:POP :POP :?
       :GOTO 370
FM 470 ? T$;",";:NEXT K:? CH
      R$(126);:NEXT I:? :?
      :GOTO 370
GA 480 IF N<0 THEN ? :GOTO 3
      70
MH 490 A(J)=N:NEXT J
JM 500 CKSUM=ADDR-INT(ADDR/2
      56)*256:FOR I=1 TO 6:
      CKSUM=CKSUM+A(I):CKSU
      M=CKSUM-256*(CKSUM>25
      5):NEXT I
KK 510 RF=128:SOUND 0,200,12
      ,8:GOSUB 570:SOUND 0,
      0,0,0:RF=0:? CHR$(126
      )
CN 520 IF N<>CKSUM THEN ? :?
       "Incorrect";CHR$(253
      );:? :GOTO 370
EK 530 FOR W=15 TO 0 STEP -1
      :SOUND 0,50,10,W:NEXT
       W
FL 540 FOR I=1 TO 6:POKE ADR
      (BUFFER$)+ADDR-BEG+I-
      1,A(I):NEXT I
HB 550 ADDR=ADDR+6:IF ADDR<=
      FIN THEN 370
GM 560 GOTO 710
FI 570 N=0:Z=0
PH 580 GET #1,A:IF A=155 OR
      A=44 OR A=32 THEN 670
FB 590 IF A<32 THEN N=-A:RET
      URN
EB 600 IF A<>126 THEN 630
ML 610 GOSUB 690:IF I=1 AND
      T=44 THEN N=-1:? CHR$
      (126);:GOTO 690
GN 620 GOTO 570
GJ 630 IF A<48 OR A>57 THEN
      580
AN 640 ? CHR$(A+RF);:N=N*10+
      A-48
EB 650 IF N>255 THEN ? CHR$(
      253);:A=126:GOTO 600
EH 660 Z=Z+1:IF Z<3 THEN 580
JH 670 IF Z=0 THEN ? CHR$(25
      3);:GOTO 570
KC 680 ? ",";:RETURN
ND 690 POKE 752,1:FOR I=1 TO
       3:? CHR$(30);:GET #6
      ,T:IF T<>44 AND T<>58
       THEN ? CHR$(A);:NEXT
       I
PI 700 POKE 752,0:? " ";CHR$
      (126);:RETURN
KM 710 GRAPHICS 0:POKE 710,2
      6:POKE 712,26:POKE 70
      9,2
FF 720 IF MEDIA=ASC("T") THE
      N 890
OJ 730 REM DISK
OK 740 IF READ THEN ? :? "Lo
      ad File":?
IG 750 IF DTYPE<>70 THEN 104
      0
AE 760 ? :? "Enter AUTORUN.S
      YS for automatic use"
      :? :? "Enter filename
      ":INPUT T$
GF 770 F$=T$:IF LEN(T$)>2 TH
      EN IF T$(1,2)<>"D:" T
      HEN F$="D:":F$(3)=T$
NJ 780 TRAP 870:CLOSE #2:OPE
      N #2,8-4*READ,0,F$:?
      :? "Working..."
JM 790 IF READ THEN FOR I=1
      TO 6:GET #2,A:NEXT I:
      GOTO 820
PO 800 PUT #2,255:PUT #2,255
DJ 810 H=INT(BEG/256):L=BEG-
      H*256:PUT #2,L:PUT #2
      ,H:H=INT(FIN/256):L=F
      IN-H*256:PUT #2,L:PUT
       #2,H
NF 820 GOSUB 970:IF PEEK(195
      )>1 THEN 870
IF 830 IF STARTADR=0 OR READ
       THEN 850
FD 840 PUT #2,224:PUT #2,2:P
      UT #2,225:PUT #2,2:H=
      INT(STARTADR/256):L=S
      TARTADR-H*256:PUT #2,
      L:PUT #2,H
HH 850 TRAP 32767:CLOSE #2:?
       "Finished.":IF READ
       THEN ? :? :LET READ=0
      :GOTO 360
HF 860 END
FO 870 ? "Error ";PEEK(195);
      " trying to access":?
       F$:CLOSE #2:? :GOTO
      760
MC 880 REM BOOT TAPE
HN 890 IF READ THEN ? :? "Re
      ad Tape"
HI 900 ? :? :? "Insert, Rewi
      nd Tape.":? "Press PL
      AY ";:IF  NOT READ TH
      EN ? "& RECORD"
LP 910 ? :? "Press RETURN wh
      en ready:";
JH 920 TRAP 960:CLOSE #2:OPE
      N #2,8-4*READ,128,"C:
      ":? :? "Working..."
NH 930 GOSUB 970:IF PEEK(195
      )>1 THEN 960
HH 940 CLOSE #2:TRAP 32767:?
       "Finished.":? :? :IF
       READ THEN LET READ=0
      :GOTO 360
HF 950 END
CD 960 ? :? "Error ";PEEK(19
      5);" when reading/wri
      ting boot tape":? :CL
      OSE #2:GOTO 890
MB 970 REM CIO Load/Save Fil
      e#2 opened  READ=8 fo
      r write, READ=1 for r
      ead
EA 980 X=32:REM File#2,$20
EF 990 ICCOM=834:ICBADR=836:
      ICBLEN=840:ICSTAT=835
MD 1000 H=INT(ADR(BUFFER$)/2
      56):L=ADR(BUFFER$)-H
      *256:POKE ICBADR+X,L
      :POKE ICBADR+X+1,H
FH 1010 L=FIN-BEG+1:H=INT(L/
      256):L=L-H*256:POKE
      ICBLEN+X,L:POKE ICBL
      EN+X+1,H
MD 1020 POKE ICCOM+X,11-4*RE
      AD:A=USR(ADR(CIO$),X
      )
BG 1030 POKE 195,PEEK(ICSTAT
      ):RETURN
KA 1040 REM SECTOR I/O
GC 1050 IF READ THEN 1100
HE 1060 ? :? "Format Disk In
       Drive 1? (Y/N):";
FC 1070 GET #1,A:IF A<>78 AN
      D A<>89 THEN 1070
EC 1080 ? CHR$(A):IF A=78 TH
      EN 1100
CP 1090 ? :? "Formatting..."
      :XIO 254,#2,0,0,"D:"
      :? "Format Complete"
      :?
AC 1100 NR=INT((FIN-BEG+127)
      /128):BUFFER$(FIN-BE
      G+2)=CHR$(0):IF READ
       THEN ? "Reading..."
      :GOTO 1120
LE 1110 ? "Writing..."
LI 1120 FOR I=1 TO NR:S=I
IO 1130 IF READ THEN GOSUB 1
      220:BUFFER$(I*128-12
      7)=SECTOR$:GOTO 1160
PL 1140 SECTOR$=BUFFER$(I*12
      8-127)
AM 1150 GOSUB 1220
DN 1160 IF PEEK(DSTATS)<>1 T
      HEN 1200
FB 1170 NEXT I
GM 1180 IF  NOT READ THEN EN
      D
DH 1190 ? :? :LET READ=0:GOT
      O 360
JJ 1200 ? "Error on disk acc
      ess.":? "May need fo
      rmatting.":GOTO 1040
KI 1210 REM
```

Free Publications

Use the handy order form in the tax package to obtain free IRS publications on over 90 different tax topics.

A PUBLIC SERVICE MESSAGE FROM THE INTERNAL REVENUE SERVICE

```
BL 1220 REM SECTOR ACCESS S
        UBROUTINE
IG 1230 REM Drive ONE
IH 1240 REM Pass buffer in S
        ECTOR$
MP 1250 REM sector # in vari
        able S
EG 1260 REM READ=1 for read,
KJ 1270 REM READ=0 for write
BN 1280 BASE=3*256
GL 1290 DUNIT=BASE+1:DCOMND=
        BASE+2:DSTATS=BASE+3
NL 1300 DBUFLO=BASE+4:DBUFHI
        =BASE+5
AI 1310 DBYTLO=BASE+8:DBYTHI
        =BASE+9
JA 1320 DAUX1=BASE+10:DAUX2=
        BASE+11
PN 1330 REM DIM DSKINV$(4)
CA 1340 DSKINV$="hLS":DSKINV
        $(4)=CHR$(228)
PF 1350 POKE DUNIT,1:A=ADR(S
        ECTOR$):H=INT(A/256)
        :L=A-256*H
BP 1360 POKE DBUFHI,H
CO 1370 POKE DBUFLO,L
PD 1380 POKE DCOMND,87-5*REA
        D
AA 1390 POKE DAUX2,INT(S/256
        ):POKE DAUX1,S-PEEK(
        DAUX2)*256
KJ 1400 A=USR(ADR(DSKINV$))
KG 1410 RETURN
```

©

Here are 152 reasons to buy at Elek-Tek, not to mention the fastest delivery anywhere.

Save 30% to 43%
off Manufacturer Suggested Retail prices on
EPSON • Okidata • Star • TOSHIBA • DIABLO •

IOMEGA
BERNOULLI BOX
(Modular)

1. 10 meg ½ height Drive for IBM-PC/XT/AT & compatibles 1675
2. 20 meg ½ height Drive for IBM-PC/XT/AT & compatibles 2335
3. Non-Bootable Interface Card . . . 104
4. Bootable Interface Card . . . 234
5. 10 meg cartridges for above (3 pak special) 125
6. Head Cleaning Kit 59

EPSON®
EPSON FX 85

16. LX 80 . . . . . $ 215
17. LX 90 230
18. RX 100+ 310
19. FX 85 340
20. FX 185 475
21. LQ 800 parallel . . . . . CALL
22. DX10 Daisy Wheel 10CPS . . 230
23. DX20 Daisy Wheel 20CPS . 320
24. DX-35 Daisy Wheel 35CPS . 640

TOSHIBA

47. Toshiba P341 . . . . . CALL
48. Toshiba P351 990
49. Toshiba P1340 . . . . . CALL
50. Toshiba P321 . . . . . CALL

OKIDATA
FULL LINE AVAILABLE . . . . . CALL

star
FULL LINE AVAILABLE . . . . . CALL

SPECIAL PURCHASE!
Diablo D-36 Daisy Wheel 35CPS
Mfr. Sugg. Ret. $1495
Elek-Tek Price $450

HEWLETT-PACKARD

71. HP-41CV . . . . . $ 168
72. HP-41CX 245
73. Optical Wand 95
74. Card Reader 145
75. Printer 283
76. Ext. Memory Module 60
77. Ext. Function Module 60
78. Time Module 60
79. HP-IL Loop Module 95
80. Digital Cassette Drive 400
81. Printer/Plotter (HP-IL) 335
82. **HP-9114A** Disk Drive 600

Hewlett-Packard
QUAD MEMORY MODULE
$25 Mfr. Sugg. Ret. $75
Upgrades the HP41C to the power of the HP41CV

SLIMLINE Shirtpocket Styled Power Packed Programmable LCD PROBLEM SOLVERS

83. **HP-11C** Scientific 58
84. **HP-15C** Scientific 90
85. **HP-12C** Financial 90
86. **HP-16C** Programmer 90

PERSONAL COMPUTATION DEALER

HP-ThinkJet

HP-2225B
$350

87. **HP-71B Computer** . . . . $ 399
88. **HP-82400A** Series 70 card reader 125
89. **HP-82401A** HP-IL interface 95
90. **HP-82402A** 4K memory module 60
91. **HP-82700A** 8K memory module 150
92. **HP-82441A** Assembly/Forth 120
93. **HP-82480A** HP-71 math pac 75
94. **HP-82482A** HP-71 finance pac 60
95. **HP-82483A** Surveying Pac 120
96. **HP-82484A** Curve Fitting 75
97. **HP-82485A** Text Editor 60
98. **HP-82488A** Data Com. Pac 120

HARD DRIVES

Dysan CORPORATION

Internal Hard Disk Subsystems

7. **10 MEGABYTE** 500
8. **20 MEGABYTE** 600

Each system includes:

- Micro Science Drive
- Adaptek Controller
- 2 cables
- Documentation
- Dysan 1yr limited warranty
- For use in IBM PC, XT or most compatibles

PRINTER BUFFERS

9. **DDP-16** 64K Par./Par. 70

QUADRAM

10. **QRMP8** 8K Parallel/Parallel 125
11. **QRMPS8** 8K Parallel/Serial 139
12. **QRMSP8** 8K Serial/Parallel 139
13. **QRMSS8** 8K Serial/Serial 139

POWER PROTECTORS

14. **PowerMite PG600-S9** 6 outlet Surge Suppressor . . . 30
15. **ISOBAR 8** 8 outlet Surge Suppressor . . . 50

PRODUCTS FOR IBM-PC

25. **Amdek 310A** Amber Monitor . . . . . $ 150
26. **Generic Multi** Multifunction Board, 64K 129
27. **Generic Multi 384K** Multifunction Board, 384K 163
28. **AST Six Pak +** Multifunction Board, 64K 225
29. **AST Six Pak + (loaded)** Multifunction Board, 384K 279
30. **AST Megaplus II** Multifunction Board, 64K 270
31. **Quadram Quadboard** Multifunction Board, 0K 195
32. **Quadram Quadboard** Multi. Board, 64K/384K . . 210/267
33. **Orchid Tech.** PC Turbo 186 570
34. **Paradise** Modular Graphics Card 290
35. **Hercules** Monochrome Card 329
36. **Hercules Color** Color Graphic Card 170
37. **Novation 490605-1** 2400BPS inc. Mite Software . . 620
38. **Novation 490592-1** 1200B Internal Modem No software . 150
39. **Novation 490603** ½ Card Modem 2400BPS No Software 425
40. **Hayes 1200B** Internal modem w/software . . . 359
41. **AT&T 4000** 300/1200 Ext. Modem 335
42. **Hayes 1200** External modem 380
43. **Hayes 2400** External modem 599
44. **US Robotics Courier 2400** Ext. 2400B Smart Modem . . . 460
45. **US Robotics Telpac** Telecomm. Software 75
46. **Toshiba ND 04D** ½ ht. DSDD Disk Drive 90

IBM is a trademark of International Business Machines Corp.

PRODUCTS FOR APPLE

51. **Toshiba HND 04DT** ½ Height DSDD Disk Drive . $ 70
52. **U—Print IIc-16K** Par Interface external 85
53. **Grappler PLUS** Graphics Interface 80
54. **Buffered Grappler** 16K Buffer (Expand to 64K) . . 150
55. **Kensington System Saver** Fan and Surge Suppr. 69
56. **Super Cooling Fan** Fan and Surge Suppr. 29
57. **Gamma** Disk Controller Card 48
58. **Gamma II** SS Disk Drive for II, IIe, IIplus . . . 125
59. **Gamma IIc** SS Disk Drive for IIc 125
60. **Rana ELITE 2** DSSD Disk Drive 335
61. **Rana ELITE 3** DSDD Disk Drive 420
62. **ERAM 80** 80 col; 64K RAM for IIe . . . 115
63. **Novation Apple Cat II** 300 Baud Modem 200
64. **Novation 212** 300/1200 Baud Modem . . 390
65. **Nov 49059112** Mod. & Soft. for MacIntosh . CALL
66. **Hayes Micromodem IIe** 300 Baud Modem 145
67. **Hayes Smartmodem IIc** 300 Baud Modem for IIc . . 240
68. **Wico 501030** Analog Joystick 36
69. **Amdek Monitors** . . . . . CALL
70. **Zenith Monitors** . . . . . CALL

HUGE SAVINGS ON DISKETTES

	Dysan	maxell	3M	SONY	wabash	MEMOREX	BASF
3½" SSDD	23.00	22.00	23.00	23.00	20.00	—	—
DSDD	29.00	29.00	30.00	35.00	24.00	—	—
5¼" SSDD	17.00	14.00	14.00	13.00	11.50	11.50	11.00
DSDD	21.00	18.00	18.00	16.00	12.50	14.00	12.00
SSDD96TPI	24.00	24.00	24.00	—	—	—	—
DSDD96TPI	36.00	33.00	35.00	—	—	—	—
5¼" DSDDHD (For IBM AT)	33.00	29.00	29.00	—	24.00	—	24.00
8" SSDD**	22.00	29.00	25.00	—	19.00	—	—
8" DSDD**	26.00	32.00	29.00	—	20.00	—	—

**Unformatted

Call for Quantity pricing for 10 boxes or more.

3M DATA CARTRIDGES

147. **DC100** . . . . . $ 14.00
148. **DC300A** 18.00
149. **DC300XL** 21.00
150. **DC300XL/P** 22.00
151. **DC600A** 23.50
152. **DC1000** 15.00

Call for Quantity pricing for 10 cartridges or more

CALL TOLL FREE 800-621-1269 EXCEPT Illinois, Alaska **CANADIAN TOLL FREE 800-458-9133**

ELEK-TEK, inc.
6557 N. Lincoln Ave., Chicago, IL 60645
(312) 631-7800 (312) 677-7660

Corp. Accts. Invited. Min. Ord. $15.00. Visa or MasterCard by Mail or Phone. Mail Cashier's Check, Mon. Ord., Personal Check (2 wks. to clear) Add $4.00 1st item. (AK, HI, P.R., Canada add $10.00 first item) $1.00 ea. add'l shpg. & handl. Shipments to IL address add 7% tax. Prices subj. to change. WRITE for free catalog. RETURN POLICY: Defectives Only: Most products replaced within 30 days of purchase with identical merchandise only. Computer and large peripherals replaced only when defective on arrival (within 3 work days of delivery). Other problems covered by mfr. warranty. ALL ELEK-TEK MERCHANDISE IS BRAND NEW, FIRST QUALITY AND COMPLETE. Delivery subject to availability. **DUNS #09-718-0517**

ATARI 130XE

ATARI 130XE Super Computer Package
130XE Computer
1050 Disk Drive
1027 Printer
Atariwriter
CALL

ATARI PRINTER INTERFACES

Uprint A 54.95
Uprint AW/16K ... 79.95
Uprint AW/64K ... 99.95
MPP 1150 ..... 59.95

INDUS GT DISK DRIVE ... Call

ATARI 130XE SUPER PRINTER PKGS.

SG-10 Printer and U-Print A ... 279
Panasonic 1091 and U-Print A ... 309
Super Printer Packages have no extra shipping charges or credit card surcharges when shipped in Continental USA

ATARI 130XE SOFTWARE

BRODERBUND
Print Shop 28.95
Karateka 20.95
Print Shop Graph. I, II, or III ... 19.95
Print Shop Comp. ... 27.95

INFOCOM
See Commodore 64 section for items and prices

ELECTRONIC ARTS
Archon II 24.95
Archon 19.95
Seven Cit. of Gold ... 24.95
Skyfox 24.95
Pinball Const. 24.95
One on One 24.95

MICROPROSE
Silent Service 23.95
Gunship 23.95
Accrojet 23.95
F-15 Strike Eagle 23.95
Kennedy Approach .. 23.95

OSS
Basic XE-Cart 52.95
MAC 65 XL-Cart 49.95
Action-Cart 49.95
Basic XL-Cart 39.95
All Tool Kits 20.95

BATTERIES INCLUDED
Home Pak 34.95
Paper Clip 39.95
B-Graph 34.95

SYNAPSE
Syncalc 32.95
Synfile 32.95
Syntrend 25.95
Syncalc Templates... 16.95
Loderunner Rescue .. 20.95
Mindwheel 27.95
Essex 27.95
Brimstone 27.95

SSI
See Commodore 64 section for items and prices

MISCELLANEOUS 130XE
Hacker 19.95
Amer. Cross Ctry. ... 19.95
Flight Simulator II ... 34.95
Ultima II 37.95
Ultima III 37.95
Universe 69.95
Letter Perfect 39.95
Data Perfect 39.95
Halley Project 27.95
Ultima I 23.95
Ultima IV 41.95
MMG Basic Comp. ... 69.95

ATARI 520ST*

Atari 520ST-RGB System...Call
Atari 520ST-Monochrome Sys. ..Call
SF314DS/DD 1 Megabyte Disk Drive .....Call
We warranty all 520ST computers purchased from ComputAbility for ninety days.
***Please call for stock availability on Atari ST products before ordering by mail.**

ATARI 520ST SOFTWARE

HABA SYSTEMS
Hippo C 54.95
Haba Write 54.95
Haba Calc 54.95
Haba Graph 54.95
Haba File 54.95

MISCELLANEOUS ST
VIP Professional 129.95
Ultima II 39.95
Perry Mason 34.95
Degas 27.95
Farenheit 451 ..... 34.95
Amazon 34.95
Hacker 29.95
The Final Word 94.95
Deja Vu 39.95
PC/Intercom 89.95
Hex 27.95
Crimson Crown 27.95
Mudpies 23.95
King's Quest II ..... Call
Gato ..... Call
Borrowed Time 34.95
Personal Prolog 79.95
Personal Pascal 64.95
Zoomracks ..... Call
Mi-Term 54.95
Regent Word 34.95

INFOCOM ST
Deadline 34.95
Starcross 34.95
Zork I, II, or III 29.95
Witness 27.95
Suspended 34.95
Planetfall 27.95
Sorcerer 29.95
Seastalker 27.95
Cutthroats 27.95
Hitchiker 27.95
Suspect 29.95
Wishbringer 27.95
Infidel 29.95
Enchanter 27.95
Spellbreaker 34.95
Mind Forever Voy. ... 34.95

AMIGA

AMIGA PERSONAL computer*
Call for Hardware and add-on peripherals prices

AMIGA SOFTWARE

Archon 27.95
Artic Fox 27.95
Deluxe Paint 59.95
Fin. Cookbook 34.95
One on One 27.95
Seven Cities 27.95
Skyfox 27.95
Hacker 29.95
Mindshadow 29.95
***Please call for availability of Amiga products before ordering by mail.**

EST. 1982

ComputAbility™

We stock hundreds of programs for the Apple, Atari, C-64 and IBM. If you don't see it listed here, don't hesitate to call.

No Surcharge for MasterCard/Visa.

VISA MasterCard

APPLE PRINTER INTERFACES AND BOARDS

Apricorn Parallel w/Graphics 69.95
Apricorn 16K Expansion Board 82.95
Apricorn 80 Column Brd. ... 64.95
Apricorn RS232 Interface 69.95
U-Print-Apple IIC w/64K 109.95
U-Print-Apple IIC w/16K 89.95

LASER 3000 PERSONAL COMPUTER

Apple 2+ Compatible
Built-in Microsoft Basic in ROM
Built-in RGB and Composite Video
Built-in 80 Column Display
Built-in Centronics Printer Interface
Built-in Numeric Keypad
192K RAM
Bundled Productivity Software
Disk Drive
All this and more for the amazing **LOW** price
$399

APPLE SOFTWARE

BRODERBUND
Print Shop 33.95
Print Shop Graphics I, II, or III 17.95
Print Shop Comp. ... 27.95
Karateka 21.95
Carmen Sandiego 25.95
Science Tool Kit 39.95
Bank Street Writer ... 44.95
Fantavision ..... Call

ELECTRONIC ARTS
Adventure Const. 34.95
Archon II 27.95
Bard's Tale 29.95
Imagic Football 24.95
Auto-Duel 34.95
Skyfox 27.95
Lords of Conquest ... 27.95
One on One 27.95
Ultima III 39.95
Ultima IV 39.95

EPYX
Ballblazer 24.95
Winter Games 24.95
Summer Games II 24.95
World's Great/ Football 24.95

INFOCOM
See Atari 520ST section for items and prices.

MICROPROSE
See Atari 130XE section for items and prices.

MINDSCAPE
Color Me 20.95
Crossword Magic 34.95
Halley Project 29.95
A View To Kill 27.95
Racter 29.95
The Mist 27.95
Perfect Score 49.95
Voodoo Island 27.95

DAVIDSON
Math Blaster 34.95
Alge-Blaster 34.95
Spell It 34.95
Word Attack 34.95

SIMON & SCHUSTER
Typing Tutor III 34.95
Kobiashi Adv 29.95
Webster Spell Chk. ... 34.95
Webster Thesaurus ... 84.95
Lovejoy SAT 49.95

SIR-TECH
Wizardry/Diam. 23.95
Wizardry/Legacy 27.95
Wizardry/Proving ... 33.95
Wizardry/Wernda ... 29.95
Wiziprint 19.95

SSI
See Commodore 64 section for items & prices.

APPLE MISCELLANEOUS
Beach-Head 23.95
Beach-Head II 23.95
Gamemaker 27.95
Hacker 27.95
Hardball 24.95
Sundog 27.95
Paper Clip 64.95
DLM Software ..... Call
The Works 34.95
Star League Base. ... 23.95
Educalc 34.95
Friendly Filer 27.95
Note Card Maker 27.95
Hayden Software ..... Call
Microleague Base. ... 29.95
Random House ..... Call
PFS Software ..... Call
Newsroom 39.95
Gato 27.95
Weekly Reader ..... Call

COMMODORE 128

C-128 Computer .. Call
1571 Disk Drive .. Call
1902 Monitor .... Call
1670 Modem .... Call
Westridge 6470 300/ 1200 Modem .. 179.95

IBM PC

IBM PC SYSTEMS

Configured to your specific needs
Call for lowest price on IBM-PC, IBM-XT or IBM-AT
Corona PC-400 Compatible .. Call
Corona Portable PC Compatible .. Call
PC Multifunction Boards
We carry the complete line of **AST, Hercules, Paradise, STB, and Quadram Call for current prices**

IBM PC SOFTWARE

Print Shop 39.95
Print Shop Graph. I .. 27.95
Bank Street Writer ... 49.95
Ancient Art of War .. 29.95

BORLAND
Sidekick 37.95
Turbo Pascal 49.95

BLUE CHIP
Baron 34.95
Squire 34.95
Millionaire 34.95
Tycoon 34.95

DIGITAL RESEARCH
Call for items and prices.

INFOCOM
See Atari 520ST for items and prices

LEADING EDGE
Nutshell 69.95
LE/WP Basic 67.95
LE/Word Proc +Speller 169.95

MICROPROSE
F-15 Strike Eagle 23.95
Kennedy Approach .. 27.95
Acrojet 27.95
Silent Service 27.95

MICROSOFT
Flight Simulator 38.95
Word 249.00
Multiplan 134.95

MINDSCAPE
See Apple Section for items and prices.

SIERRA
King's Quest 34.95
King's Quest II 34.95
Ultima II 39.95

THOUGHTWARE
Call for items and prices.

IBM MISCELLANEOUS
PFS ..... Call
Gato 27.95
Wizardry 39.95
Strip Poker 27.95
Electric Desk 204.95
D-Base III ..... Call
Sideways 39.95
Home Pak 34.95
Sargon III 34.95
Peachtree ..... Call
Jet 34.95
BPI Business ..... Call
Newsroom 39.95

COMMODORE 128 SOFTWARE

Multiplan 64/128. 44.95
Consultant 52.95
Paper Clip +Spell .. 54.95
Swiftcalc 49.95
Wordwriter 49.95
Data Manager .. 49.95
Fleet System II .. 44.95
Superbase 128 .. 69.95
Mach V/128 .... 34.95

C-64 SUPER COMPUTER PACKAGE

C-64 Computer
1541 Disk Drive
803 Printer
ONLY $399

C-64 SUPER PRINTER PKGS.

SG-10 & G-Wiz ... 279
Panasonic 1091 & G-Wiz 309
Legend 1080 & G-Wiz 269
If Xetec Supergraphic is desired add $10 to package.
Super Printer packages have no added shipping or charge card surcharges added when shipped in Continental USA
Westridge AA/AD Modem .. 49.95
Cardco G-Wiz ... 54.95

GENERAL HARDWARE

STAR MICRONICS INC. THE POWER BEHIND THE PRINTED WORD

SG-10 215
SG-15 369
SD-10 339
SD-15 449
SR-10 489
SR-15 ..... Call

PRINTERS
Panasonic 1091 245
Legend 808 169
Legend 1080 209
Powertype 309
Juki 5510 389
Epson ..... Call

PRINTER BUFFERS
Microfazer ..... From 169
U-Buff 16K 79.95
U-Buff 64K 99.95

MODEMS
US Robotics 2400 469
Volksmodem 1200 189
Prometheus 1200 319
Password 1200 209
Novation ..... Call
PC Modems ..... Call

MONITORS
Technika MJ-22RGB 269
Sakata SC-100 169
Samsung 12" Green .. 79.95
Samsung 12" Amb. .. 79.95
Taxan ..... Call
Amdek ..... Call
Commodore 1802 169

COMMODORE 64 SOFTWARE

ACCESS
Beach-Head 21.95
Beach-Head II 24.95
Raid/Moscow 24.95
Mach V-Cart 21.95

INFOCOM
Zork I 24.95
Zork II, or III 27.95
Deadline 29.95
Starcross 29.95
Witness 29.95
Planetfall 24.95
Hitchiker 24.95
Enchanter 24.95
Cutthroats 24.95
Sorcerer 29.95
Spellbreaker 29.95

SSI
Battallion Commander 24.95
Battle of Antietnam 32.95
Fighter Command (No Atari) 37.95
Norway 85 (No Atari) 21.95
Panzer Grenidier 24.95
USAAF 37.95
Breakthrough/ Ardennes 37.95
Kampgruppe 37.95
Phantasie (No Atari) 24.95
Broadsides 24.95
Carrier Force 37.95
Comp. Ambush 37.95
Mech Brigade (No Atari) 37.95
Field of Fire (No Apple) 24.95
Op. Mkt. Garden 32.95
Pro Tour Golf (No Atari) 24.95
Gemstone Warrior 21.95
Imp. Galactum 24.95
Computer Baseball ... 24.95
Comp. Quarterback ... 24.95

ELECTRONIC ARTS
Adv. Construction ... 29.95
Mail Order Monster ... 24.95
Racing Destruction ... 24.95
Ultima III 37.95
Ultima IV 41.95
Bard's Tale 27.95
See Atari 130XE section for rest of items & prices.

EPYX
Eidolon 24.95
Koronis Rift 24.95
Winter Games 24.95
Apshai Trilogy 24.95
Fast Load-Cart 24.95
Program/Tool Kit 29.95

MICROPROSE
See Atari 130XE section for items and prices.

MISCELLANEOUS COMMODORE 64
Print Shop 28.95
Cal-Kit 34.95
Superbase 64 ..... 47.95
Karateka 20.95
Hacker 20.95
Gamemaker 27.95
Ultima II 37.95
Karate Champ 25.95
Essex 27.95
Kung Fu/ Kung Fu Master 25.95
9 Princes of Amber ... 22.95
Spy vs. Spy Vol. II ... 23.95
Maxi Golf 24.95
Internat'l Hockey 19.95
The Fourth Proto. 23.95
Blazing Paddles 24.95
Mirage Word 34.95
Mirage Database 34.95
Welcome Aboard 19.95
Super Huey 14.95
Spell It 34.95
Math Blaster 34.95
Word Attack 34.95
Odesta Chess 49.95
Brimstone 27.95

EST. 1982

ComputAbility™

P.O. Box 17882, Milwaukee, WI 53217
ORDER LINES OPEN
Mon-Fri 11 a.m. - 7 p.m. CST • Sat. 12 p.m. - 5 p.m. CST

To Order Call Toll Free
800-558-0003

For Technical Info., Order Inquiries, or for Wisc. Orders
414-351-2007

ORDERING INFORMATION: Please **specify system.** For fast delivery send cashier's check or money order. Personal and company checks allow 14 business days to clear. School P.O.'s welcome. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 for software orders, 4% shipping for hardware, minimum $4.00. Master Card and Visa orders please include card #, expiration date and signature. WI residents please include 5% sales tax. HI, AK, FPO, APO, Puerto Rico and Canadian orders, please add 5% shipping, **minimum $5.00.** All other **foreign orders add 15% shipping, minimum $10.00.** All orders shipped outside the Continental U.S.A. are shipped first class insured U.S. mail. If foreign shipping charges exceed the minimum amount, you will be charged the additional amount to get your package to you quickly and safely. All goods are new and include factory warranty. Due to our low prices all sales are final. **All defective returns must have a return authorization number.** Please call (414) 351-2007 to obtain an R.A.# or your return will not be accepted. Prices and availability subject to change without notice.

Classified

SOFTWARE

TI-99/4A NEW STATES AND CAPITALS GAME
Hi-Res map of USA. Send $12 for cass.
Or $1 for more info. to: TRINITY SYSTEMS
1022 Grandview, Pittsburgh, PA 15237

TI-99/4A QUALITY SOFTWARE for Business, Home and Entertainment ** BONUS Software Offer! ** Send for FREE Catalog to MICRO-BIZ HAWAII, Box 1108 Pearl City, HI 96782

TI-99/4A Software/Hardware bargains.
Hard-to-find items. Huge selection.
Fast service. Free catalog. D.E.C.,
Box 690, Hicksville, NY 11801

FANTASTIC DAILY NUMBER FORECASTER!
Guaranteed Winners or Money Back!
Picks up to 3 STRAIGHT WINNERS most every week, playing 1 to 3 a day!
Apple, IBM, C64, Atari, 1 drive. Many reports of hitting for THOUAND. Send SASE for info. $99.95 on disk only to:
Z-Way, P.O. Box 9017, Canton, OH 44711

PROGRAMS FOR THE TANDY 1000
Send $1 for list of educat. & entertainment programs. Refundable with first purchase.
SODA POP SW, POB 653, Kenosha, WI 53141

FREE APPLE SOFTWARE
Over 1,000 Public Domain Programs on 50 diskettes, $5 each, plus $1 shipping per order. Send $1 for catalog, refundable.
C & H ENTERPRISES
Box 29243, Memphis, TN 38127

TI-99/4A DISK OWNERS: Great new software for home and business - SCHEDULE MANAGER ($19.95), DISK DATA BASE ($15.00) TUNNELS OF DOOM GAME EDITOR ($20.00), etc. Send orders to ASGARD SOFTWARE, P.O.B. 10306, ROCKVILLE, MD 20850. Catalog on request.

LOTTO PICKER. Improve your chances for those Million Dollar Jackpots! Picks LOTTO, WIN-4, & Daily Numbers. All USA & Can. games incl. Expandable! IBM/C64/TI99 $29.95. Order Now! 1-800-341-1950 Ext. 77. Mail Orders: Ridge, 170 Broadway, #201-C, NYC, NY 10038. Catalog.

SAVE MONEY! EASY TAX SIMPLIFIES THE 15 most common IRS tax forms. Faster, line by line preparation on screen/printer. Commodore 64, disk. Send $39.95 plus $2.00 s.h. to Hybrid Software, 1739 Schilder Lane, Waverly, OH 45690

PROJECT PLANNING/MANAGEMENT using the C64, SX, or C128. Data sheet for SASE - Prgm for $106.95 (CA res. add 6% sls tx). LAWCO, Dept. C, Box 2009, Manteca, CA 95336

Genealogy Program for the C64. **"FAMILY TREE"** will produce Pedigree Charts, Family Group Records, Individual Files, Indexes, Searches of Ancestors. LDS version available. "The Best" genealogy program for the 64. $49.95, GENEALOGY SOFTWARE, POB 1151, PORT HURON, MI 48061, (519) 344-3990.

Animal Records maintained with **"PETIGREE"** for the C64. Produces Litter, Awards, Breeding Show, Individual Records, Pedigree Charts. $69.95. GENEALOGY SOFTWARE, POB 1151, PORT HURON, MI 48061, (519) 344-3990.

FREE SOFTWARE CATALOG!
Call Toll-Free 1-800-554-1162, Tevex, Inc.
Save ⅓ off retail prices. We carry SSI, Elect. Arts, Infocom, and many more!

CHEMISTRY TUTORIAL - Science software programs idea for teaching and review of the basics of chemistry. The chemical and physical properties of 106 known elements. This program has a complete and unique presentation of the **PERIODIC TABLE.** Supplied on dual-sided disk (C64 and Apple IIe versions) . . . $29.95. **SPRINGHILL LABORATORY COMPUTER SOFTWARE**, P.O. Box 155, Clarksville, Ohio 45113

COMMODORE: TRY BEFORE YOU BUY. Top 25 best-selling games, utilities, new releases. Visa, MasterCard. Free brochure. Rent-A-Disk, 908 9th Ave., Huntington, WV 25701 (304) 522-1665

ATARIWARE: The BEST PD software from Atari Enthusiasts across the U.S.! 80 disks to choose from $5 each. Catalog with SASE. KD-ACE, 1187 Dunbar Ct., Orange Pk, FL 32073

INTEREST CALCULATIONS.
MAI-2.10 lets your computer help analyze investment decisions. Calc: future value, present value, annuities, sinking funds, loan pymt sched., + more! Menu driven/Simple. IBM PC/XT/jr or compat. Only $49.95 + $7 s/h/t. ck/mo. Munier Associates, Inc., Dept. A5, P.O. Box 79314, Houston, Texas 77279 (713) 784-4348.

HARDWARE

Hey! Monitor Cables $5.95. Joystick ext. 6'- $3.95, 12'-$4.95. p/h incl. Comdr Disk + Print Cables Custom $3 + .75/ft. + p/h. JCRL, 5043 E. Mitchell, Phoenix, AZ 85018 (602) 990-4643

BUY/SELL USED MICRO EQUIPMENT
Quickly, easily. National Listing Service.
No Commission. TECTRAN, 1-800-832-8726 (orders) 1-617-491-4888 (info.)

HARDWARE & SOFTWARE 30% BELOW RETAIL. Apple, Atari, C64, IBM-PC, TI-99. Over 1000 titles. Hard to find items. Atari 520 ST Computer/Color-$829.00. Send $1.00 for catalog. Specify computer. Multi-Video, P.O. Box 246, East Amherst, NY 14051.

MISCELLANEOUS

RIBBONS for ANY PRINTER at LOW PRICES!!
DELTA MICRONICS
BOX 10933, ERIE, PA 16514
(814) 455-5667

HELP IS ON THE WAY!
Just call 1-800-334-0868 to get your free copy of the latest COMPUTE! Books Catalog! If you need help in getting information on all of the latest COMPUTE! book titles available plus all COMPUTE! backlist titles, call us today!

C64 USERS - FREE BROCHURE! Game and instructional programs, each include detailed analysis, beg. or int. level. SASE to:
C16 H.O.S, 19730 Ave 18, Madera, CA 93637

FREE CATALOG. Specify TI99, Commodore, IBM. Hardware, Software, Accessories. Competition Computer, 2629 W. National, Milwaukee, WI 53204 (800) 662-9253

Maxell MD1, $1.29-MD2, $1.99. Dysan 104/1D, $1.79-104/2D, $2.39. Shipping $3.75. Also Verbatim, IBM, 3M, BASF. TAPE WORLD, 220 Spring St., Butler, PA 16001, 1-800-245-6000: Visa, MC.

DISK SALE! - SS/DD 35-trk for Apple w/sleeve & label-10/$5.80, bulk-100/$45. Standard SS/DD w/sleeve & label-10/$7.50, bulk-100/$59. DS/DD w/sleeve & label-10/$8.50, bulk-100/$67. 3½" SS for Mac-10/$19.99. PREMIUM QUALITY, LIFETIME WARRANTY! Money-back satisfaction guarantee! Min. order $20. Send check or pay by MC/VISA/AE $3 shipping, + $2 if C.O.D. - UNITECH, 20 Hurley St., Cambridge, MA 02141. (800) 343-0472, in Mass. (617) "UNI-TECH".

FREE CATALOG . . . PRINTER PAPER
Discount Prices . . . Quick Delivery
Universal Services, Inc., P.O. Box 484, Grand Haven, Michigan 49417

EARN MONEY, PART OR FULL TIME, AT HOME with your computer-manual & forms-$9.95. Write Computer Programs for Profit! How-to guide with forms, letters, tips-$7.95. Also-Computer Consultant Handbook. How to earn $ consulting with business-$7.95. JV Tech, P.O. Box 563, Ludington, MI 49431

SERVICE MANUAL WITH SCHEMATICS FOR ATARI 800XL-$19.50; Atari 1050-$19.50. (805) 927-4667. Visa/MC. Free catalog. Electronic Dimension, Box 1846, San Luis Obispo, CA 93406

IBM PCJR REPORT: THE NATIONAL NEWSLETTER. PCjr-specific articles, reviews, Public Domain from across the nation. $18./yr. PCJR CLUB, POB 95067, Schaumburg, IL 60195

Don't take chance of LOSING DATA caused by HUMIDITY. Disk manufacturers spec. 80% max. Get 15 reusable drypacks for only $2 to protect all your disks permanently.
CHINGHAI, Box 2687, Costa Mesa, CA 92628

COMPUTE! Classified is a low-cost way to tell over 350,000 microcomputer owners about your product or service.

Rates: $25 per line, minimum of four lines. Any or all of the first line set in capital letters at no charge. Add $15 per line for boldface words, or $50 for the entire ad set in boldface (any number of lines.)
Terms: Prepayment is required. Check, money order, American Express, Visa, or MasterCard is accepted. Make checks payable to COMPUTE! Publications.
Form: Ads are subject to publisher's approval and must be either typed or legibly printed. One line equals 40 letters and spaces between words. Please underline words to be set in boldface.
General Information: Advertisers using post office box numbers in their ads must supply permanent address and telephone numbers. Orders will not be acknowledged. Ad will appear in next available issue after receipt.
Closing: 10th of the third month preceding cover date (e.g., June issue closes March 10th). Send order and remittance to: Harry Blair, Classified Manager, COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. To place an ad by phone, call Harry Blair at (919) 275-9809.
Notice: COMPUTE! Publications cannot be responsible for offers or claims of advertisers, but will attempt to screen out misleading or questionable copy.

MUST LIQUIDATE At Far Below Dealer Cost!
TOTAL Personal Computer System

Factory Reconditioned with Factory Warranty!

Carries easily as a suitcase! Plugs into 115V outlet!

GREAT GIFT IDEA!

Sorry, we're not permitted to PRINT the famous brand-name. BUT, we CAN "tell all" if you call us TOLL FREE: 1-800-328-0609!

THE COMPUTER

Snap-on computer keyboard! 64K RAM, 20K ROM. Full-size typewriter keyboard. Upper and lower case letters, numerals, symbols, reverse characters. 2 cursor control keys, 4 function keys, programmable to 8. Music synthesizer with 3 independent voices, each with 9 octave range. Input/output ports accommodate . . . user, serial, ROM cartridge, joysticks, external monitor, phone modem.

Built-in disk drive! Intelligent high speed unit with 5¼" floppy disk recorder. 170K formatted data storage; 35 tracks. 16K ROM. Uses single sided, single density disk. Serial interface. Second serial port to chain second drive or printer.

Built-in color monitor! Displays 40 columns x 25 lines of text on 5" screen. High resolution. 320 x 200 pixels. 16 background, character colors.

Built-in ROM cartridge port! Insert ROM program cartridge. Multitude of subjects available in stores across the nation!

Original List Price $995.00

Liquidation Priced At Only **$388**

Item H-927-63631-00 Ship, handling: $20.00

THE PRINTER

Print method: Bi-directional impact dot matrix.
Character matrix: 6 x 7 dot matrix.
Characters: Upper and lower case letters, numerals and symbols. All PET graphic characters.
Graphics: 7 vertical dots — maximum 480 columns. Dot addressable.
Character codes: CBM ASCII code.
Print speed: 60 characters per second.
Maximum columns: 80 columns.
Character spacing: 10 characters per inch.
Line feed spacing: 6 lines per inch in character mode or 8 lines per inch selectable. 9 lines per inch in graphics mode.
Line feed speed: 5 lines per second in character mode. 7.5 lines per second in graphics mode.
Paper feed: Friction feed.
Paper width: 4.5" to 8.5" width.
Multiple copies: Original plus maximum of two copies.
Dimensions: 13"W x 8"D x 3¼"H. Wt.: 6½ lbs. **Power:** 120V AC, 60 Hz.

Original List Price: $200.00

Liquidation Priced At **$119**

Item H-927-63831-00 Ship, handling: $7.00

THE SOFTWARE

"Easy Script" One of the most powerful word processors at any price! Cut re-typing, create documents from standard paragraphs, do personalized letters, see and change a document before it is printed. Instruction manual has extensive training section that simplifies use . . . even for someone who has never used a computer or word processor before!

"The Manager" A sophisticated database manager for business or home use. Business uses: accounts payable/receivable, inventory, appointments, task manager. Home uses: mailing lists, home inventory, recipes, collection organizer, investment tracking, checkbook balancing. School uses: research article index, gradebook.

Mfr. Sug. Retail: $73.98

Liquidation Price **$24**

Item H-927-64011-03 Ship, handling: $3.00

BUY INDIVIDUAL UNITS OR GET THIS ULTRA-FAMOUS SYSTEM AT ONE LOW PACKAGE PRICE!

TOTAL Personal Computer System available at FAR BELOW dealer cost!

Original List Price $1,293.00

TOTAL PACKAGE PRICE **$488**

Item H-927-64011-02 Ship, handling: $24.00

Compatible with above Computer System (Not included in package price.)

JOYSTICKS (Set of 2)
Mfr. List: $59.90 pr.
Liquidation Price . . . . . **$19** pr.
Item H-927-63622-01 S/H: $4.00 pr.

64K MODEM Factory New/First Quality! (90-Day Lmt. Factory Warranty.)
Mfr. List: $124.95
Liquidation Price **$19**
Item H-927-63646-00 S/H: $4.00

VISA MasterCard

Credit card members can order by phone, 24 hours a day, 7 days a week.

Toll-Free: 1-800-328-0609

TeleCheck Your check is welcome! No delays when you pay by check!

C.O.M.B. Direct Marketing Corp.
Authorized Liquidator
14605 28th Ave. N. ● Mpls., MN 55441-3397

SEND ME THE ITEMS I HAVE LISTED BELOW

Sales outside continental U.S. are subject to special conditions. Please call or write to inquire.

Item No.	#	Item	Price	S/H
		TOTAL		

C.O.M.B. Direct Marketing Corp. Item H-927
14605 28th Ave. N./Minneapolis, MN 55441-3397

Send the items indicated at left. (Minnesota residents add 6% sales tax. Please allow 3-4 weeks delivery. Sorry, no C.O.D.)

☐ My check or money order is enclosed. (No delays in processing orders paid by check, thanks to TeleCheck.)

Charge: ☐ MasterCard ☐ VISA

Acct. No. _____ Exp. _____

PLEASE PRINT CLEARLY

Name _____

Address _____

City _____

State _____ ZIP _____

Phone ( ) _____

Sign Here _____

Advertisers Index

COMPUTE!'s FREE Reader Information Service

Use these cards to request FREE information about the products advertised in this issue. Clearly print or type your full name and address. Only one card should be used per person. Circle the numbers that correspond to the key number appearing in the advertisers index.

Send in the card and the advertisers will receive your inquiry. Although every effort is made to insure that only advertisers wishing to provide product information have reader service numbers, COMPUTE! cannot be responsible if advertisers do not provide literature to readers.

Please use these cards *only* for subscribing or for requesting product information. Editorial and customer service inquiries should be addressed to: COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Check the expiration date on the card to insure proper handling.

Use these cards and this address only for COMPUTE!'s Reader Information Service. Do not send with payment in any form.

COMPUTE!

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

Circle 999 for a one year new U.S. subscription to COMPUTE!. You will be billed for $24.

Please let us know. Do you own: plan to buy:

- ☐ 270 Apple _____ ☐ 271
- ☐ 272 Atari _____ ☐ 273
- ☐ 274 Commodore _____ ☐ 275
- ☐ 276 IBM _____ ☐ 277
- ☐ 278 TI-99/4A _____ ☐ 279
- ☐ 280 Other _____ (specify model) ☐ 281

Please print or type name and address.
Limit one card per person.

Name _____

Address _____

City _____

State/Province _____ Zip _____

Country _____

Please Include ZIP Code Expiration 3/31/86 CO286

SUBSCRIBE TO COMPUTE!

For Fastest Service, Call Our **Toll-Free** US Order Line
800-334-0868
In NC call 919-275-9809

My Computer Is:

01 ☐ Apple 02 ☐ Atari 03 ☐ Commodore 64
04 ☐ VIC-20 05 ☐ IBM 06 ☐ TI-99/4A
99 ☐ Other _____ ☐ Don't yet have one.

☐ $24.00 One Year US Subscription
☐ $45.00 Two Year US Subscription

(Readers outside of the US, please see our foreign readers subscription card or inquire for rates).

Name _____

Address _____

City _____ State _____ Zip _____

☐ Payment Enclosed ☐ Bill me
Charge my: ☐ VISA ☐ MasterCard ☐ American Express
Account No. _____ Expires ____/____

Your subscription will begin with the next available issue. Please allow 4-6 weeks for delivery of first issue. Subscription prices subject to change at any time.
The COMPUTE! subscriber list is made available to carefully screened companies that offer products and services that may be of interest to our readers. If you prefer not to receive such mailings, please check this box ☐.

J1280

Place
Stamp
Here

COMPUTE! Reader Service
P.O. Box 2141
Radnor, PA 19089

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 7478 DES MOINES, IOWA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!
PO BOX 10954
DES MOINES, IOWA 50347

RUSH POSTAGE-PAID CARD FOR YOUR FREE CATALOG

- **COMPUTER ELECTRONICS** training prepares you to service and repair all computers as you build your own 16-bit IBM PC compatible computer. Total system program includes disk drive, test equipment, bundled software, and NRI Discovery Lab.
- **TV/VIDEO/AUDIO SERVICING** includes training with 27" color TV with built-in stereo decoder, VCR, and actual instruction on videotape.
- **SATELLITE COMMUNICATIONS** training gives you the skills to service both commercial and consumer satellite earth station equipment as you assemble your own home satellite TV system.
- **ROBOTICS** training features remote-controlled mobile programmable robot you build, experiment with, and keep along with other test equipment.
- **INDUSTRIAL ELECTRONICS** training prepares you to service and maintain computer-controlled industrial systems. Training includes color computer, disk drive, exclusive NRI training software, and NRI Discovery Lab.

CHECK ☑ ONE

- ☐ Computer Electronics
- ☐ TV/Video/Audio Servicing
- ☐ Satellite Communications
- ☐ Robotics
- ☐ Data Communications
- ☐ Industrial Electronics
- ☐ Communications Electronics
- ☐ Electronic Design Technology
- ☐ Telephone Servicing
- ☐ Digital Electronics
- ☐ Basic Electronics
- ☐ Bookkeeping & Accounting
- ☐ Appliance Servicing
- ☐ Small Engine Servicing
- ☐ Air Conditioning, Heating, Refrigeration, & Solar Technology
- ☐ Locksmithing & Electronic Security
- ☐ Building Construction & Remodeling
- ☐ Automotive Servicing

Name _____ (Please Print) _____ Age _____

Street _____

City _____ State _____ Zip _____

Accredited by the National Home Study Council

For career courses approved for Veterans Training, check for details.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 10008 WASHINGTON, D.C.

POSTAGE WILL BE PAID BY ADDRESSEE

NRI Schools

McGraw Hill Continuing
Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016-9973

Flight Simulator II Scenery Disks

The Challenge of Accomplished Flight

With a realism comparable to (and in some ways even surpassing) $100,000 aircraft flight simulators, Flight Simulator II includes full flight instrumentation and avionics, and provides a full-color out-the-window view. Instruments are arranged in the format standard to modern aircraft. All the radios needed for IFR flight are included. Front, rear, left, right, and diagonal views let you look in any direction. Program features are clearly documented in a 96-page Pilot's Operating Handbook.

For training in proper flight techniques, Flight Simulator II includes another 96-page instruction manual, compiled by two professional flight instructors with over 8,000 hours flight time and 12,000 hours of aviation teaching experience. You'll learn correct FAA-recommended flight procedures, from basic aircraft control through instrument approaches. To reward your accomplishments, the manual even includes a section on aerobatic maneuvers.

The Realism and Beauty of Flight

Go sight-seeing over detailed, realistic United States scenery. High-speed graphic drivers provide an animated out-the-window view in either day, dusk, or night flying modes.

Flight Simulator II features over 80 airports in four different scenery areas: New York, Chicago, Seattle, and Los Angeles. Six additional Scenery Disks covering the entire Western half of the United States are now available in IBM and C64/128 disk formats.

Apple and Atari versions will be released soon. Each disk covers a geographical region of the country in detail, and is very reasonably priced.

The Pure Fun of "World War I Ace"

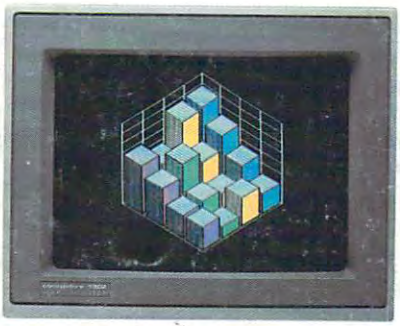
When you think you're ready, you can test your flying skills with the "World War I Ace" aerial battle game. This game sends you on a bombing run over heavily-defended enemy territory. Six enemy fighters will attempt to engage you in combat as soon as war is declared. Your aircraft can carry five bombs, and your machine guns are loaded with 100 rounds of ammunition.

See Your Dealer. Flight Simulator II is available on disk for the Apple II, Atari XL/XE, and Commodore 64/128 computers for $49.95. Scenery Disks for the C64 and IBM PC (Jet or Microsoft Flight Simulator) are $19.95 each. A complete Western U.S. Scenery six-disk set is also available for $99.95. For additional product or ordering information, call (800) 637-4983.

subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

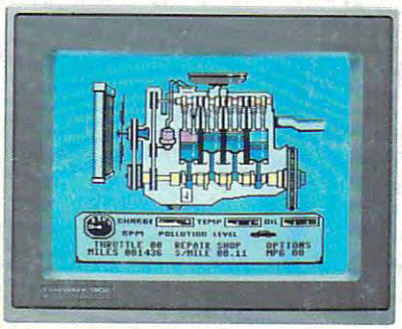
Order Line: (800) 637-4983
(except in Illinois, Alaska, and Hawaii)

All you need to do this



graph a spreadsheet

write a novel



fix an engine

compose a song

paint a picture

r banking

learn to fly

organize a data base

tell a story

forecast sales

When it comes to personal computers, you want the smartest you can own. At a price that makes sense.

The new Commodore 128™ system has a powerful 128K memory, expandable to 512K. An 80-column display and 64, 128 and CP/M® modes for easy access to thousands of educational, business and home programs. And a keyboard, with built-in numeric keypad, that operates with little effort.

Discover the personal computer that does more for you. At the price you've been waiting for. From the company that sells more personal computers than IBM® or Apple®.

COMMODORE 128 PERSONAL COMPUTER

A Higher Intelligence

www.commodore.ca