

Auto-Duel

by Tim Scott

Requirements:

Apple II Plus or better
Auto-Duel

Auto-Duel, the Car Wars role-playing game come to computer, is a fun, if not difficult, game. You are to get a large amount of dough and make a car filled with all kinds of weapons such as anti-tank guns, machine guns, and even smoke screens. After earning enough prestige, you could head an FBI investigation to throw Mr. Big into the slammer.

The attributes to your Auto-Duel character are all stored in the file CR. Most of the info, which appears to be a common practice with Origin, is stored into binary coded decimal formats, that is, a "15" in the file is read as fifteen and not as twenty one (as a PEEK from a BASIC program would give).

The Auto-Dueler Editor is able to change the following: location of your duelist and his currently used car, his name, the amount of money he has, his driving, marksmanship, and mechanic skills, his health, his body armor, and the current day, month, and year of the game.


Table 1 shows where the bytes of each attribute are stored in the file CR.

Money, Location, and Date

Unlike the other attributes, these three are stored each in a special way, with money being as close to the "normal" format of the three.

Money is stored in three different bytes, \$14, \$15, and \$16. The unusual thing about the money bytes, is the way they're stored in reverse order, meaning that if you have 609010 dollars, byte \$16 has the value of 60, byte \$15 has the value of 90, and byte \$14 has the value 10.

The Location is stored in a different way than the others but it is not too unusual, being stored in hexadecimal form (values can be \$0A, \$0B, etc.) Each number stored in this byte represents being in a certain city. \$00 stands for being in Watertown, while \$0A means that the duelist is located in Harrisburgh.

Most strange of all the bytes found, is the date. It is stored in normal decimal format, but has a value of -1 of what is shown when, during play, a  is pressed. What that means is,

if the date, according to the game, is 2-9-2030, the bytes, \$22, \$23, and \$24, where the values are put, will be 01, 08 and 29 (the 2000 is added automatically with no controlling bytes that I could find.)

Table 1

Byte	Controlled Attribute
\$04-\$13	Name in ASCII (16 chrs in length)
\$14-\$16	Money
\$17	Prestige
\$18	Driving Skill
\$19	Marksmanship Skill
\$1A	Mechanic Skill
\$1E	Health
\$20	Location
\$26	Body Armor
\$22-\$24	Day, Month, and Year respectively

Entering and Using The Auto-Dueler Editor

The Auto-Dueler Editor is one BASIC program. Type it in, and SAVE it before RUNNING it.

SAVE AD EDITOR

Run the program and enter which drive the Auto-Duel Player Disk will be in. Then press RETURN after inserting the disk into the chosen drive.

A selection menu after that should then appear on the screen. Enter your choice and then press RETURN. Now, depending on which one you selected, should be a screen showing the current value of that choice and give you the option to change it. To change it, just type in the appropriate numbers and then hit RETURN. If you just want to view what the current value was and do not wish to change it, just press RETURN and the value will remain the same.

Now go out there and get Mr. Big!

Auto-Dueler Editor

```

10 REM -----
20 REM == THE AUTO-DUEL ==
30 REM == EDITOR ==
40 REM == BY TIM SCOTT ==
50 REM -----
60 ONERR GOTO 1600
70 DIM PL$(15)
80 FOR I = 0 TO 15 : READ PL$(I) : NEXT
90 D$ = CHR$(4) : BA = 35327
100 TEXT : HOME

```

```

110 VTAB 1 : HTAB 12 : PRINT "AUTO-DUELER^
EDITOR"
120 POKE 34,1
130 HTAB 15 : PRINT "BY^ TIM^ SCOTT"
140 VTAB 10 : HTAB 7 : INPUT "DRIVE^ (1)^ OR^
DRIVE^ (2)^ USED?^ " : A$ : D = VAL (A$) :
IF D < 0 OR D > 2 THEN GOTO 100
150 VTAB 13 : HTAB 9 : PRINT "INSERT^ DISK^
INTO^ DRIVE^ " : D
160 VTAB 15 : HTAB 13 : INPUT "PRESS^
RETURN^ " : A$
170 PRINT D$ "BLOOD^ CR, A$8A00, D" D
180 REM MAIN MENU
190 HOME
200 PRINT : PRINT "CHOOSE^ ONE:^ "
210 PRINT "1)^ NAME"
220 PRINT "2)^ MONEY"
230 PRINT "3)^ PRESTIGE"
240 PRINT "4)^ DRIVING^ SKILL"
250 PRINT "5)^ MARKSMANSHIP"
260 PRINT "6)^ MECHANIC^ SKILL"
270 PRINT "7)^ HEALTH"
280 PRINT "8)^ BODY^ ARMOR"
290 PRINT "9)^ DAY, ^ MONTH, ^ & ^ YEAR"
300 PRINT "10)^ LOCATION"
310 PRINT "11)^ QUIT^ & ^ SAVE"
320 PRINT : INPUT "CHOICE?^ " : A$
330 A = VAL (A$)
340 IF A$ = "" THEN GOTO 190
350 IF A < 1 OR A > 11 THEN GOTO 190
360 ON A GOSUB 460, 570, 740, 830, 910, 990
, 1070, 1150, 1230, 390, 1470
370 GOTO 190
380 REM LOCATION ROUTINE
390 HOME : PRINT
400 PRINT "CURRENT^ LOCATION:^ " : PL$(PEEK
(BA + 29))
410 PRINT : FOR I = 0 TO 15 : PRINT I ")^ "
: PL$(I) : NEXT
420 PRINT : INPUT "NEW^ LOCATION:^ " : NL$ : NL
= VAL (NL$) : IF NL$ = "" THEN RETURN
430 IF NL < 0 OR NL > 15 THEN GOTO 390
440 POKE BA + 29, NL
450 RETURN
460 REM NAME ROUTINE
470 HOME
480 PRINT : PRINT "CURRENT^ NAME:^ " :
490 FOR I = 1 TO 16 : PRINT CHR$(PEEK (BA + I
)) : NEXT
500 PRINT : INPUT "NEW^ NAME^ (16^ CHARS):^ "
: N$
510 IF LEN (N$) > 16 THEN GOTO 470
520 IF LEN (N$) = 0 THEN RETURN
530 FOR I = 1 TO 16 : M$ = MID$ (N$, I, 1)
540 IF M$ = "" THEN M$ = "^ "
550 POKE BA + I, ASC (M$) + 128 : NEXT
560 RETURN
570 REM MONEY ROUTINE
580 HOME

```

```

590 X = 17 : GOSUB 1570 : AM = NU
600 X = 18 : GOSUB 1570 : AM = AM + NU * 100
610 X = 19 : GOSUB 1570 : AM = AM + NU * 10000
620 PRINT : PRINT "CURRENT^ AMOUNT: ^ " ; AM
630 PRINT : INPUT "NEW^ AMOUNT^ (0-999999) : ^ " ; NAS
640 IF NAS = "" THEN RETURN
650 NA = VAL (NAS)
660 IF NA > 999999 THEN GOTO 580
670 POKE BA + 17 , 0 : POKE BA + 18 , 0 : POKE BA
+ 19 , 0
680 ON (NA < 10000) GOTO 700 : PN = VAL (MID$
(NAS , 1 , (INT (LEN (NAS) / 3) ) ) ) : X =
19 : GOSUB 1590
690 NAS = RIGHT$ (NAS , 4)
700 ON (NA < 100) GOTO 720 : PN = VAL (MID$
(NAS , 1 , (INT (LEN (NAS) / 2) ) ) ) : X =
18 : GOSUB 1590
710 NAS = RIGHT$ (NAS , 2)
720 ON (NA = 0) GOTO 730 : PN = VAL (MID$ (NAS
, 1 , (LEN (NAS) ) ) ) : X = 17 : GOSUB 1590
730 RETURN
740 REM PRESTIGE ROUTINE
750 HOME
760 X = 20 : GOSUB 1570
770 PRINT : PRINT "CURRENT^ PRESTIGE: ^ " ; NU
780 PRINT : INPUT "NEW^ PRESTIGE^ (0-99) : ^ "
; PN$
790 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
800 IF PN < 0 OR PN > 99 THEN GOTO 740
810 GOSUB 1590 : RETURN
820 RETURN
830 REM DRIVING SKILL ROUT.
840 HOME
850 X = 21 : GOSUB 1570
860 PRINT : PRINT "CURRENT^ DRIVING^ SKILL: ^
" ; NU
870 PRINT : INPUT "NEW^ DRIVING^ SKILL^
(0-99) : ^ " ; PN$
880 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
890 IF PN < 0 OR PN > 99 THEN GOTO 830
900 GOSUB 1590 : RETURN
910 REM MARKSMAN ROUTINE
920 HOME
930 X = 22 : GOSUB 1570
940 PRINT : PRINT "CURRENT^ MARKSMAN: ^ " ; NU
950 PRINT : INPUT "NEW^ MARKSMAN^ (0-99) : ^ "
; PN$
960 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
970 IF PN < 0 OR PN > 99 THEN GOTO 910
980 GOSUB 1590 : RETURN
990 REM MECHANIC SKILL
1000 HOME
1010 X = 23 : GOSUB 1570
1020 PRINT : PRINT "CURRENT^ MECHANIC^
SKILL: ^ " ; NU
1030 PRINT : INPUT "NEW^ MECHANIC^ SKILL^
(0-99) : ^ " ; PN$
1040 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
1050 IF PN < 0 OR PN > 99 THEN GOTO 990
1060 GOSUB 1590 : RETURN
1070 REM HEALTH ROUTINE
1080 HOME
1090 X = 27 : GOSUB 1570
1100 PRINT : PRINT "CURRENT^ HEALTH: ^ " ; NU
1110 PRINT : INPUT "NEW^ HEALTH^ (0-99) : ^ "
; PN$
1120 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
1130 IF PN < 0 OR PN > 99 THEN GOTO 1070
1140 GOSUB 1590 : RETURN
1150 REM BODY ARMOR ROUTINE
1160 HOME
1170 X = 35 : GOSUB 1570

```

```

1180 PRINT : PRINT "CURRENT^ BODY^ ARMOR: ^ "
; NU
1190 PRINT : INPUT "NEW^ BODY^ ARMOR^
(0-99) : ^ " ; PN$
1200 PN = VAL (PN$) : IF PN$ = "" THEN RETURN
1210 IF PN < 0 OR PN > 99 THEN GOTO 1150
1220 GOSUB 1590 : RETURN
1230 REM DAY/MONTH/YEAR ROUT.
1240 HOME
1250 X = 31 : GOSUB 1570 : D = NU
1260 X = 32 : GOSUB 1570 : M = NU
1270 X = 33 : GOSUB 1570 : Y = NU
1280 PRINT : PRINT "CURRENT^ DAY: ^ " ; D + 1
1290 PRINT " ^ ^ ^ ^ ^ ^ ^ ^ MONTH: ^ " ; M + 1
1300 PRINT " ^ ^ ^ ^ ^ ^ ^ ^ YEAR: ^ " ; 2000 + Y + 1
1310 PRINT : INPUT "NEW^ DAY^ (1-30) : ^ " ; AS$
D = VAL (AS$) : IF AS$ = "" THEN RETURN
1320 IF D < 1 OR D > 30 THEN GOTO 1310
1330 INPUT " ^ ^ ^ ^ ^ ^ ^ ^ MONTH: ^ " ; M
1340 IF M < 1 OR M > 12 THEN PRINT : GOTO 1330
1350 INPUT " ^ ^ ^ ^ ^ ^ ^ ^ YEAR: ^ " ; Y
1360 IF Y < 1 OR Y > 99 THEN PRINT : GOTO 1350
1370 X = 31 : D = D - 1 : PN = D : GOSUB 1590
1380 X = 32 : M = M - 1 : PN = M : GOSUB 1590
1390 X = 33 : Y = Y - 1 : PN = Y : GOSUB 1590
1400 RETURN
1410 DATA ^ WATERTOWN , MANCHESTER , BUFFALO
1420 DATA ^ SYRACUSE , ALBANY , BOSTON
1430 DATA ^ SCANTON , NEW^ YORK , PROVIDENCE
1440 DATA ^ PITTSBURGH , HARRISBURGH
, PHILADELPHIA
1450 DATA ^ ATLANTIC^ CITY , BALTIMORE , DOVER
1460 DATA ^ WASHINGTON
1470 REM SAVE ROUTINE
1480 HOME
1490 PRINT : INPUT "ARE^ YOUR^ SURE?^ (Y/N) : ^ "
; AS$
1500 IF AS$ = "N" THEN HOME : PRINT "DONE... " :
END
1510 IF AS$ = "Y" THEN GOTO 1530
1520 GOTO 1480
1530 PRINT : PRINT "SAVING..."
1540 PRINT D$ "BSAVE^ CR , A$8A00 , L$28"
1550 PRINT : PRINT "SAVED..."
1560 POP : END
1570 NU = INT ( PEEK (BA + X) / 16 ) * 10 +
PEEK (BA + X) - INT ( PEEK (BA + X) / 16 )
* 16
1580 RETURN
1590 POKE BA + X , INT (PN / 10) * 16 + (PN -
INT (PN / 10) * 10) : RETURN
1600 REM ERROR ROUTINE
1610 PRINT
1620 A = PEEK (222)
1630 IF A = 4 THEN PRINT "DISK^ IS^
WRITE-PROTECTED"
1640 IF A = 8 THEN PRINT "I/O^ ERROR"
1650 IF A = 9 THEN PRINT "DISK^ FULL"
1660 IF A = 10 THEN PRINT "FILE^ LOCKED"
1670 IF A = 11 THEN PRINT "SYNTAX^ ERROR^
(DOS)"
1680 IF A = 6 THEN PRINT "NOT^ A^ AUTO-DUEL^
DISK"
1690 END

```

checksums

10	- \$BADD	860	- \$39A0
20	- \$9B13	870	- \$0F0F
30	- \$4D3B	880	- \$D85E
40	- \$AD92	890	- \$869C
50	- \$C899	900	- \$E28D
60	- \$197B	910	- \$24F1

70	- \$6D1A	920	- \$E562
80	- \$99A6	930	- \$F11C
90	- \$3B09	940	- \$DB9B
100	- \$54D2	950	- \$2877
110	- \$60DF	960	- \$DEED
120	- \$882D	970	- \$C693
130	- \$D4B7	980	- \$0DE5
140	- \$C806	990	- \$D4FA
150	- \$7A27	1000	- \$829C
160	- \$869E	1010	- \$23A0
170	- \$44ED	1020	- \$70AA
180	- \$1B65	1030	- \$939E
190	- \$46D5	1040	- \$F217
200	- \$32BC	1050	- \$75EA
210	- \$4C2E	1060	- \$210F
220	- \$84AB	1070	- \$3B2D
230	- \$C48E	1080	- \$3CAE
240	- \$F830	1090	- \$C0F5
250	- \$1407	1100	- \$D3BB
260	- \$A2B9	1110	- \$B37C
270	- \$1257	1120	- \$88FF
280	- \$C880	1130	- \$78F6
290	- \$A618	1140	- \$173A
300	- \$C508	1150	- \$2A21
310	- \$3DD1	1160	- \$48DB
320	- \$298F	1170	- \$04E4
330	- \$5DEF	1180	- \$293A
340	- \$19D6	1190	- \$2361
350	- \$D326	1200	- \$C18F
360	- \$8F7A	1210	- \$AC83
370	- \$3560	1220	- \$EAF6
380	- \$083F	1230	- \$FE88
390	- \$43A8	1240	- \$A585
400	- \$A308	1250	- \$FC34
410	- \$4355	1260	- \$A052
420	- \$FBA6	1270	- \$EE46
430	- \$23C4	1280	- \$CF08
440	- \$5252	1290	- \$79B2
450	- \$3102	1300	- \$4E13
460	- \$02D7	1310	- \$101C
470	- \$5629	1320	- \$3E5F
480	- \$1C91	1330	- \$2B2D
490	- \$57B4	1340	- \$16F6
500	- \$AC70	1350	- \$2ADC
510	- \$84A8	1360	- \$13B4
520	- \$1D0D	1370	- \$34E7
530	- \$566E	1380	- \$C3F0
540	- \$D94B	1390	- \$6653
550	- \$89BC	1400	- \$76C9
560	- \$70F7	1410	- \$0EF6
570	- \$F4EC	1420	- \$49D2
580	- \$7E1F	1430	- \$2319
590	- \$19E7	1440	- \$313C
600	- \$DD1B	1450	- \$A002
610	- \$F1C9	1460	- \$9722
620	- \$FAF8	1470	- \$5070
630	- \$22C5	1480	- \$8F4A
640	- \$0B9B	1490	- \$411E
650	- \$4953	1500	- \$6D96
660	- \$7112	1510	- \$1698
670	- \$6781	1520	- \$FBA1
680	- \$3ACB	1530	- \$2B6F
690	- \$946F	1540	- \$B2B0
700	- \$10A1	1550	- \$80CD
710	- \$5F8B	1560	- \$E264
720	- \$8267	1570	- \$02A2
730	- \$37C7	1580	- \$345E
740	- \$4404	1590	- \$7F1A
750	- \$1EBE	1600	- \$37C8
760	- \$0AB3	1610	- \$CD11
770	- \$5687	1620	- \$B7F0
780	- \$6774	1630	- \$5F20
790	- \$2901	1640	- \$8000
800	- \$A465	1650	- \$BA31
810	- \$B168	1660	- \$AAD0
820	- \$A67C	1670	- \$1FE0
830	- \$F17F	1680	- \$31F5
840	- \$BB37	1690	- \$4E20
850	- \$360E		