THE LOST "TRICKS" [PART 2]
BY MIKE WHITE

LAST TIME I ASKED IF YOU HAD LEARNED ENOUGH FROM PAST TUTORIALS THAT I WROTE (THAT WERE POSTED BY ADAM TRIONFO) TO DO AN "AB 4x2 MULTICART LOADER"! WELL, I FOUND MYSELF AT THIS POINT WHEN I UNDERTOOK THE TASK A FEW YEARS AGO TO "DOUBLE UP" THE # OF AB PROGRAMS ON WARD SHRAKE'S "MULTICART"! THIS USES THE "ON BOARD MENU" LIKE THE "BRB 2x4 BASICART LOADER" DOES, (THAT WAS LISTED LAST TIME), AND DOESN'T REQUIRE ANY MODIFICATIONS TO AB EITHER! IT GOES:

```
!2000 55 [MENU FLAG] <USER>
!2001 BB [ADDRESS OF] <1ST LINK>
!2002 3F [NEXT SEGMENT]
!2003 E2 [ADDRESS OF]
!2004 3F [#1 TITLE STRING]
!2005 B8 [ADDRESS OF]
!2006 2F [#1 PROGRAM LOADER]
!2007 ↓↓ [BASIC PROGRAM #1]
  ↓
!2FA7 DB IN A,(14) <SWAP LOOP> (TO !4FA0)
!2FA8 14 [INPUT = KEY]         (TO !4FA1)
!2FA9 B7 OR A                  (TO !4FA2)
!2FAA 28 JR Z,FB               (TO !4FA3)
!2FAB FB [WAIT FOR KEY]        (TO !4FA4)
!2FAC FB EI                    (TO !4FA5)
!2FAD CD CALL,2D06             (TO !4FA6)
!2FAE 06 [ADDRESS OF]          (TO !4FA7)
!2FAF 2D [? IN AB]             (TO !4FA8)
!2FB0 11 LD DE,4EBD            (TO !4FA9)
!2FB1 BD [SET BASIC]           (TO !4FAA)
!2FB2 4E [POINTER]             (TO !4FAB)
!2FB3 D5 PUSH DE               (TO !4FAC)
!2FB4 1B DEC DE                (TO !4FAD)
!2FB5 C3 JP,2563               (TO !4FAE)
!2FB6 63 [JUMP INTO AB]        (TO !4FAF)
!2FB7 25 [WARM START]          (TO !4FB0)
!2FB8 F3 DI <#1 PROGRAM LOADER>
!2FB9 FF RST 38
!2FBA 00 [INTP C]
!2FBB 17 [SET OUT+1]
!2FBC B0 [TO &(10)]
!2FBD 2C [TO &(9)]
!2FBE 18 [TO &(14)]
!2FBF 19 [COL SET+1]
!2FC0 FF [ADDRESS OF]
!2FC1 2F [COLOR TABLE #1]
!2FC2 5F [MOVE+1]
!2FC3 00 [DESTINATION]
!2FC4 40 [ADDRESS]
!2FC5 A0 [# OF BYTES]
!2FC6 0F [TO MOVE]
!2FC7 07 [SCORCE]
!2FC8 20 [ADDRESS]
!2FC9 02 [X INT C] <INTERPRETED JUMP>
!2FCA FB EI
!2FCB FF RST 38
!2FCC 51 [PAWS+1]
!2FCD 7F [DELAY TIME]
!2FCE F3 DI
!2FCF 31 LD SP,4FEA
!2FD0 EA [SET STACK]
!2FD1 4F [POINTER]
```

```
!2FD2 21 LD,HL 4000
!2FD3 00 [STARTING ADDRESS]
!2FD4 40 [TO CLEAR SCREEN]
!2FD5 3E LD A,55 <CLEAR LOOP>
!2FD6 55 [CLEAR MASK]
!2FD7 A6 AND (HL)
!2FD8 77 LD (HL),A
!2FD9 23 INC HL
!2FDA 3E LD A,4E
!2FDB 4E [END]
!2FDC BC CP,H
!2FDD 20 JR NZ,F6
!2FDE F6 [TO CLEAR LOOP]
!2FDF FF RST 38
!2FE0 35 [STR DIS+1]
!2FE1 14 [HOR. POS.]
!2FE2 20 [VER. POS.]
!2FE3 28 [MAGIC BYTE]
!2FE4 C9 [ADDRESS OF]
!2FE5 3F [INSTRUCTION STRING]
!2FE6 ED IM2
!2FE7 5E [SET INTERRUPT MODE]
!2FE8 3E LD A,20
!2FE9 20 [SET VECTOR]
!2FEA ED LD I,A
!2FEB 47 [SET HIGH ORDER BYTE]
!2FEC 3E LD A,22
!2FED 22 [SET LOW ORDER BYTE]
!2FEE D3 OUT (0D),A
!2FEF 0D [TO &(13)]
!2FF0 3E LD A,C8
!2FF1 C8 [INTERRUPT TRIGGER]
!2FF2 D3 OUT (0F),A
!2FF3 0F [TO &(15)]
!2FF4 FF RST 38
!2FF5 5F [MOVE+1]
!2FF6 A0 [DESTINATION]
!2FF7 4F [ADDRESS]
!2FF8 11 [# OF BYTES]
!2FF9 00 [TO MOVE]
!2FFA A7 [SCORCE]
!2FFB 2F [ADDRESS]
!2FFC C3 JP,4FA0
!2FFD A0 [JUMP INTO]
!2FFE 4F [SWAP LOOP]
!2FFF 00 [TO &(7)] <COLOR TABLE #1>
!3000 00 [TO &(6)]
!3001 07 [TO &(5)]
!3002 07 [TO &(4)]
!3003 5A [TO &(3)]
!3004 84 [TO &(2)]
!3005 06 [TO &(1)]
```

```
!3006 00 [TO &(0)]
!3007 ↓↓ [BASIC PROGRAM #2]
  ↓
!3FA7 F3 DI <#2 PROGRAM LOADER>
!3FA8 FF RST 38
!3FA9 00 [INTP C]
!3FAA 17 [SET OUT+1]
!3FAB B0 [TO &(10)]
!3FAC 2C [TO &(9)]
!3FAD 18 [TO &(14)]
!3FAE 19 [COL SET+1]
!3FAF C1 [ADDRESS OF]
!3FB0 3F [COLOR TABLE #2]
!3FB1 5F [MOVE+1]
!3FB2 00 [DESTINATION]
!3FB3 40 [ADDRESS]
!3FB4 A0 [# OF BYTES]
!3FB5 0F [TO MOVE]
!3FB6 07 [SCORCE]
!3FB7 30 [ADDRESS]
!3FB8 0B [M JUMP+1]
!3FB9 C9 [ADDRESS OF]
!3FBA 2F [INTERPRETED JUMP]
!3FBB 18 [ADDRESS OF] <2ND LINK>
!3FBC 02 [NEXT SEGMENT]
!3FBD ?? [ADDRESS OF]
!3FBE 3F [#2 TITLE STRING]
!3FBF A7 [ADDRESS OF]
!3FC0 3F [#2 PROGRAM LOADER]
!3FC1 00 [TO &(7)] <COLOR TABLE #2>
!3FC2 00 [TO &(6)]
!3FC3 07 [TO &(5)]
!3FC4 07 [TO &(4)]
!3FC5 5A [TO &(3)]
```

```
!3FC6 84 [TO &(2)]
!3FC7 06 [TO &(1)]
!3FC8 00 [TO &(0)]
!3FC9 53 S <INSTRUCTION STRING>
!3FCA 57 W
!3FCB 49 I
!3FCC 54 T
!3FCD 43 C
!3FCE 48 H
!3FCF 20 [SPACE]
!3FD0 54 T
!3FD1 30 0
!3FD2 20 [SPACE]
!3FD3 22 "
!3FD4 41 A
!3FD5 42 B
!3FD6 22 "
!3FD7 6B (CHANGE E,D)
!3FD8 2D (HOR. POS.)
!3FD9 30 (VER. POS.)
!3FDA 50 P
!3FDB 52 R
!3FDC 45 E
!3FDD 53 S
!3FDE 53 S
!3FDF 20 [SPACE]
!3FE0 3D =
!3FE1 00 (END)
!3FE2 ↓↓ <#1 TITLE STRING>
  ↓
!3F?? 00 (END)
!3F?? ↓↓ <#2 TITLE STRING>
  ↓
!3FFF 00 (END)
```

NOTICE THE SIMILARITIES BETWEEN THE BRB LOADER (LAST TIME) AND THE ABOVE LISTING. THE "ON BOARD MENU" IS USED JUST LIKE BEFORE, EXCEPT THAT THE 2ND SEGMENT IS NOW AT !3FBB. ALSO, IF YOU'RE CURIOUS, THE FOLLOWING SEGMENTS ARE:

```
!01FA C4 [ADDRESS OF] <CALCULATOR LINK>
!01FB 0D [NEXT SEGMENT]
!01FC DD [ADDRESS OF]
!01FC 0D [CALCULATOR STRING]
!01FD 20 [ADDRESS OF]
!01FE 10 [CALCULATOR PROGRAM]
  ↓
!0218 BE [ADDRESS OF] <GUNFIGHT LINK>
!0219 0D [NEXT SEGMENT]
!021A CA [ADDRESS OF]
!021B 0D [GUNFIGHT STRING]
!021C DE [ADDRESS OF]
!021D 17 [GUNFIGHT PROGRAM]
  ↓
!0DBE FA [ADDRESS OF] <CHECKMATE LINK>
!0DBF 01 [NEXT SEGMENT]
!0DC0 D3 [ADDRESS OF]
!0DC1 0D [CHECKMATE STRING]
!0DC2 28 [ADDRESS OF]
!0DC3 13 [CHECKMATE PROGRAM]
!0DC4 00 [END OF MENU] <SCRIBBLING LINK>
!0DC5 00 [MARKER]
!0DC6 E8 [ADDRESS OF]
```

```
!0DC7 OD [SCRIBBLING STRING]
!0DC8 19 [ADDRESS OF]
!0DC9 0E [SCRIBBLING PROGRAM]
!0DCA 47 G <GUNFIGHT STRING>
!0DCB 55 U
!0DCC 4E N
!0DCD 46 F
!0DCE 49 I
!0DCF 47 G
!0DD0 48 H
!0DD1 54 T
!0DD2 00 (END)
!0DD3 43 C <CHECKMATE STRING>
!0DD4 48 H
!0DD5 45 E
!0DD6 43 C
!0DD7 4B K
!0DD8 4D M
!0DD9 41 A
!0DDA 54 T
!0DDB 45 E
!0DDC 00 (END)
```

```
!0DDD 43 C <CALCULATOR STRING>
!0DDE 41 A
!0DDF 4C L
!0DE0 43 C
!0DE1 55 U
!0DE2 4C L
!0DE3 41 A
!0DE4 54 T
!0DE5 4F O
!0DE6 52 R
!0DE7 00 (END)
```

```
!0DE8 53 S <SCRIBBLING STRING>
!0DE9 43 C
!0DEA 52 R
!0DEB 49 I
!0DEC 42 B
!0DED 42 B
!0DEE 4C L
!0DEF 49 I
!0DF0 4E N
!0DF1 47 G
!0DF2 00 (END)
```

HERE YOU'LL FIND THE "ON BOARD" SEGMENTS OF THE "LINKED LIST" USED BY "MENU", BUT NOT LISTED LAST TIME WITH "COLD START"! OTHER THAN THE SEGMENT FOR THE "CALCULATOR LINK" BEING OUT OF SEQUENCE IN MEMORY, THERE ARE NO REAL SURPRISES HERE. THE BALLY PROGRAMMERS LEFT THE OPENING FOR CARTRIDGE TITLES AT THE TOP OF THE MENU, AND ENDED WITH A "0000 LINK", WHICH FLAGS "MENU" TO STOP. THE STRINGS ARE IN ASCII CODE AND "STR DIS" PRINTS THEM ON THE SCREEN. YOU SEE, "MENU" IS A MASTER SUBROUTINE THAT CALLS OTHER RST 38 SUBS! NEXT TIME PERHAPS, I'LL TALK ABOUT SOME OF THEM.

"MENU" EXITS BY JUMPING TO WHERE THE 3RD ADDRESS (IN THE SEGMENT THAT PRINTED THE TITLE THAT THE GAME PLAYER SELECTED) POINTS TO! IN OUR "AB MULTICART LOADER" (SHOWN ABOVE) A PROGRAM #1 SELECTION WOULD JUMP TO THE ADDRESS AT !2005+!2006 WHICH IS !2FB8! HERE THE INTERRUPTS ARE DISABLED SO AS TO CHANGE &(14) (SEE LAST TIME)! THEN THE INTERPRETER GANGS "SET OUT+1" "COL SET+1" AND "MOVE+1". HOWEVER, IF PROGRAM #2 WAS SELECTED THE 2ND SEGMENT OF THE "LINKED LIST" (AT !3FBB) GETS USED, AND THE EXITING ADDRESS APPEARS AT !3FBF+!3FC0 AS !3FA7! NOTICE THAT !3FA7 THROUGH !3FB7 IS ALMOST IDENTICAL TO !2FB8 THROUGH !2FC8! IN FACT, EXCEPT FOR THE COLOR TABLE ADDRESS, THE SCORCE ADDRESS IN "MOVE", AND ANY OPTIONAL CHANGES YOU MAKE TO &(9) AND &(10), BOTH PROGRAM LOADERS COULD SHARE A COMMON PATH! RST 38 SUB 10+1 TITLED "M JUMP+1" (AT !3FB8) JOINS LOADER #2 WITH LOADER #1 AT !2FC9 (SEE ARCADIAN VOL.5 PG.133 WHERE ANDY GUEVARA WROTE A BIT ON THE INTERPRETER AND ITS COMPLIMENT). THIS MEANS THAT FROM HERE ON, BOTH LOADERS DO SHARE A COMMON PATH! NOTE: THIS LISTING SHOWS DEFAULT VALUES IN THE COLOR TABLES AND FOR &(9) AND &(10). FEEL FREE TO CHANGE ANY OF THIS AS DESIRED.

LAST TIME WE USED "M CALL" AND "M RET" TO "NEST" PART OF AN INTERPRETED STRING SO BOTH LOADERS COULD "TIME" SHARE A COMMON PATH. IF THERE'S ANY QUESTIONS ABOUT ANY OF THAT, PLEASE ASK! THIS TIME WE DON'T NEED TO "NEST" ANYTHING, BUT WE STILL "TIME" SHARE AS MUCH AS POSSIBLE!

WITH ALL THE DIFFERENCES NOW BEHIND US, BOTH LOADERS EXIT THE INTERPRETER VIA "X INT C" AT !2FC9! THEN THE SCREEN INTERRUPT IS RE-ENABLED SO "PAWS+1" WON'T TIE UP (SEE ARCADIAN VOL.5 PG.100). YOU SEE, THIS ROUTINE HALTS THE Z80 AND COUNTS INTERRUPTS UNTIL THE "DELAY TIME" (AT !2FCD) RUNS OUT! THEREFORE, A !FF VALUE GIVES ABOUT 4.3 SEC. AND 00 NO DELAY! !7F IS SHOWN (ABOUT 2 SEC.) BUT THIS BYTE CAN CHANGE HOWEVER YOU WISH! THIS ALLOWS US TO VIEW OUR "TITLE SCREEN" (THE SCREEN THAT THE "AB" PROGRAM WAS TAPED WITH)! "MOVE+1" PUT IT ON THE SCREEN AND "PAWS+1" LETS US SEE IT BEFORE WE CLEAR IT OFF TO PRINT OUR "INSTRUCTION STRING"! MORE ON THAT LATER.

AGAIN WE "DISABLE INTERRUPTS", THIS TIME TO CHANGE FROM THE "ON BOARD" ROUTINE TO AB'S OWN! WE NOW MOVE THE STACK POINTER TO WHERE AB KEEPS IT (SEE PAST TUTORIALS), AND LOAD "HL" WITH !4000 SO THAT WE CAN CLEAR THE SCREEN! NOTE: WE CAN'T USE "FILL" THIS TIME BECAUSE OUR BASIC PROGRAM RESIDES ON THE SCREEN! SO, WE BORROW A LESSON FROM **"REPACK FAST"** AND ENTER A "HOMEMADE" LOOP WHERE "A" IS LOADED WITH !55, THEN WE USE LOGIC "AND" WITH THE BYTE IN MEMORY POINTED TO BY (HL)! THE RESULT "ALWAYS" APPEARS IN "A", SO WE LOAD [THE SAME BYTE IN MEMORY POINTED TO BY (HL)] WITH "A", INCREMENT "HL", RELOAD "A" WITH !4E AND COMPARE "A" TO "H"! THE COMPARE INSTRUCTION SUBTRACTS "H" FROM "A" AND DISCARDS THE RESULTS, KEEPING ONLY THE FLAGS! NEXT COMES THE "JUMP RELATIVE ON NON ZERO CONDITION" INSTRUCTION, WITH AN OFFSET OF !F6 THAT SENDS US BACK TO !2FD5 UNTIL "HL" REACHES A VALUE OF !4E00 (OR 19968 DEC.). IF ANY QUESTIONS EXHIST, PLEASE ASK!

"STR DIS+1" IS NEXT CALLED TO PRINT THE INSTRUCTION STRING! THIS REALLY IS 2 STRINGS WITH A SPECIAL CHANGE INSTRUCTION BETWEEN THAT PRINTS OUT WITH ONLY 1 RST 38 SUB CALL! NO TUTORIAL ANYWHERE IN THE 1980'S MENTIONED THIS, BUT IN THE 300 PAGE "NUTTING MANUEL" (ON PG.38) IS A CHART THAT RESEMBLES:

!64 C	!68 (NONE)	!6C C	!70 IX	!74 IX,C	!78 IX	!7C IX,C
!65 E,C	!69 E	!6D E,C	!71 IX,E	!75 IX,E,C	!79 IX,E	!7D IX,E,C
!66 D,C	!6A D	!6E D,C	!72 IX,D	!76 IX,D,C	!7A IX,D	!7E IX,D,C
!67 E,D,C	!6B E,D	!6F E,D,C	!73 IX,E,D	!77 IX,E,D,C	!7B IX,E,D	!7F IX,E,D,C

THIS IS THE CHART OF "CONTROL BYTES" FOR "STR DIS"! ANY BYTE IN A STRING WITH A VALUE BETWEEN 1 AND 99 (!01 THROUGH !63) PRINTS OUT AS A CHARACTER (OF SOME TYPE), WHILE BYTE VALUES FROM 128 TO 255 (OR !80 AND UP) USE THE "IX" REGISTER TO POINT TO AN ALTERNATE CHARACTER FONT FOR "HOMEMADE" CHARACTERS (SEE ARCADIAN VOL.5 PG.14,15,37,72 ON "CHR DIS" BECAUSE "STR DIS" CALLS "CHR DIS" FOR EACH CHARACTER IT DISPLAYS)! ONLY A BYTE FROM 100 THROUGH 127 (!64 THROUGH !7F) IS SEEN AS A "CONTROL BYTE" AND THE ABOVE CHART SHOWS HOW MANY OF THE FOLLOWING BYTES GET "SUCKED" INTO THE Z80'S REGISTERS, WHERE THEY GO, AND IN WHAT ORDER, FOR EACH "CONTROL BYTE"! WITH THIS CHART, WE ARE NOW ABLE TO CHANGE LOCATION (SUCH AS A "CARRAGE RETURN" [SEE ABOVE]), OR COLOR (AS SEEN IN THE "ON BOARD" MESSAGE STRING SHOWN LAST TIME IN THE LISTING FOR "COLD START", OR SIZE, OR SCREEN WRITE, OR CHARACTER FONTS, OR ANY NUMBER OF THESE CHARACTERISTICS AT ONCE! THIS MEANS THAT IN ONE STRING YOU CAN HAVE MULTIPLE SIZES, COLORS, FONTS, EVEN OVERLAYS! **EVEN IN THE "ON BOARD MENU"!!** "EMPTY SPACE" (SHOWN LAST TIME) COULD BE USED FOR THIS! HOWEVER, 2 RULES MUST BE OBSERVED! 1: IN THE BRB BASICART LOADER (LAST TIME) YOU'LL HAVE TO USE ANOTHER STRING UNLESS YOU WANT YOUR "CHANGES" TO APPEAR IN THE INSTRUCTIONS AS WELL AS "MENU", AND 2: IF YOU CHANGE ANYTHING IN "MENU" IT MUST BE RESTORED!! THE DEFAULTS FOR "MENU" ARE: IX=(ADDRESS OF NEXT LINK), E=(DOESN'T MATTER), D=(PROGRAM NUMBER)x8+4, C=09! FINALLY, IF "MENU" IS CALLED DIRECTLY AS SHOWN LAST TIME, YOU CAN ENHANSE THE "TOP OF SCREEN MESSAGE" THIS WAY AS WELL! STILL MORE ON THIS NEXT TIME!

AFTER THE INSTRUCTION STRING PRINTS, THE SCREEN INTERRUPT IS RESET FOR "AB" (AT !2FE6 TO !2FF3). FIRST, THE Z80 IS SET TO IM2 (SEE LAST TIME). THEN THE VECTOR'S MSB IS GOTTEN TO THE "I" REGISTER BY THE ONLY WAY THAT THE Z80 INSTRUCTION SET ALLOWS! THE LSB IS SENT TO &(13) IN A SIMILAR FASHION (BY BORROWING "A"), AND DITTO FOR THE VALUE OF !C8 (200 DEC.) TO &(15)! NOTE: A VALUE OF 200 INSURES THAT THE SCREEN INTERRUPT SERVICE ROUTINE OCCURS DURING VERTICAL RETRACE. MORE ABOUT THIS NEXT TIME.

THIS PART WAS TOTALLY DIFFERENT THAN LAST TIME AS WE DID IT BY HAND, RATHER THAN "CALL !2164". HOWEVER, A LOOK AT !2164 IN BRB REVEALS:

```
!2164 F3 DI
!2165 ED IM2
!2166 5E [SET INTERRUPT MODE]
!2167 3E LD A,20
!2168 20 [SET VECTOR]
!2169 ED LD I,A
!216A 47 [SET HIGH ORDER BYTE]
!216B 3E LD A,08
!216C 08 [SET ADDRESS CHIP MODE]
!216D D3 OUT (0E),A
!216E 0E [TO &(14)]
!216F 3E LD A,C8
!2170 C8 [INTERRUPT TRIGGER]
!2171 D3 OUT (0F),A
!2172 0F [TO &(15)]
!2173 3E LD A,12
!2174 12 [SET LOW ORDER BYTE]
!2175 D3 OUT (0D),A
!2176 0D [TO &(13)]
!2177 FB EI
!2178 C9 RET
```

OTHER THAN THE ADDRESS CHANGE TO &(13) AND THE INCLUSION OF &(14), THE SAME THING IS ACCOMPLISHED!

IN AB, THE INTERRUPT VECTOR POINTS US TO !2022 WHERE RESIDES !4E95 (20117 DEC.) (SEE AB HANDBOOK PG.103)! THERE IT READS:

```
!4E95 (20117) C3 JP,21FD
!4E96 (20118) FD [TO INTERRUPT]
!4E97 (20119) 21 [SERVICE ROUTINE]
```

THIS WAS PUT IN SO THAT ANYONE CAN "SPLICE" ANYTHING THEY WANT INTO THE INTERRUPT SERVICE ROUTINE! TO "SPLICE", USE %(20118)=X, AND %(20118)=8701 TO "UNSPLICE"! NOTE: "X" IS THE BEGINNING ADDRESS OF A "HOMEMADE" ROUTINE THAT RUNS 60 TIMES PER SECOND AND ENDS WITH A "JP,21FD" INSTRUCTION!

IN BRB IT'S EVEN EASIER TO GET INTO THE INTERRUPT SERVICE ROUTINE! WITH EVERY SCREEN INTERRUPT, THE 14TH BIT OF THE VARIABLE "NT" IS TESTED! IF A "1" IS FOUND, THE INTERRUPT SERVICE ROUTINE CALLS !6DCF AS A SUBROUTINE! THEREFORE, START YOUR "HOMEMADE" ROUTINE (ENDING WITH "!C9 RET"), (OR USE A JUMP INSTRUCTION), AT !6DCF (28111 DEC.), AND USE "NT=!20XX TO ACTIVATE AND "NT=!00XX" (OR "DEFAULT ") TO STOP! NOTE: ANY NOTE TIME (THE LOWER BYTE) WILL STILL WORK, UNLESS YOU DO SOMETHING FANCY! NEXT TIME!!

THERE WAS NO EASY WAY TO GET INTO THE INTERRUPT SERVICE ROUTINE IN OLD BB, OR VIPERSOFT! THE ONLY WAY TO DO IT WAS TO RE-INITIALIZE THE "VECTOR"! HOWEVER, ":RETURN " RESET THE VECTOR! (SEE THE Z80 MINICOURSE, OR NEXT TIME TO LEARN MORE!)

THIS BRINGS US TO !2FF4, WHERE WE CALL "MOVE+1" ONE LAST TIME TO PUT OUR "SWAP LOOP" INTO !4FA0 THROUGH !4FB0 FROM !2FA7 THROUGH !2FB7. THE REASON IS THAT WHEN WE CHANGE THE (MULTICART'S) "DIP SWITCHES", ANYTHING IN THE !2XXX AND !3XXX ADDRESSES WILL DISAPPEAR! NOTICE ALSO THAT THE "EI" INSTRUCTION (ENABLE INTERRUPTS) HAS BEEN WITHHELD. THIS IS BECAUSE THE INTERRUPT SERVICE ROUTINE.IN "AB" IS IN THE BASIC CARTRIDGE AND UNTIL THE DIP SWITCHES ARE SET FOR "AB", THE BASIC CARTRIDGE CANNOT BE "SEEN" BY THE COMPUTER!

KNOWING ALL THAT, WE NOW JUMP FROM !2FFD TO !4FA0 SO THAT THE COMPUTER DOESN'T GET LOST WHEN THE DIP SWITCHES ARE CHANGED! HERE, WE ENTER A 5 BYTE LOOP THAT LOADS THE "A" REGISTER FROM INPUT &(20), THEN "OR'S" "A" WITH ITSELF, WHICH ONLY SETS THE FLAGS (SEE ABOVE)! NEXT IS THE "JUMP RELATIVE ON ZERO CONDITION" INSTRUCTION WITH AN OFFSET OF !FB. THIS CAUSES THE COMPUTER TO JUMP BACK TO !4FA0 AS LONG AS THE ZERO FLAG STAYS SET (BY THE "OR" INSTRUCTION)! THIS KEEPS THE COMPUTER IN THIS TINY LOOP UNTIL A RIGHT-MOST KEYPAD [&(20)] KEY IS PRESSED!

WITH THE COMPUTER "PAUSED", WE **NOW** THROW THE DIP SWITCHES TO "AB". NOTE: IF YOU NOW PULL THE MULTICART OUT, AND PLUG A REAL "ASTRO BASIC" INTO THE CARTRIDGE SLOT, YOU'LL BE ABLE TO USE THE TAPING INTERFACE!

WHEN "AB" IS FIRMLY SET IN (MULTICART OR ASTRO BASIC), PRESS ANY KEY IN THE RIGHT-MOST COLUMN (EXCEPT "LIST " WHICH IS THE "TRACE" KEY IN AB)! THIS LOADS "A" WITH A NON-ZERO VALUE AND THE "ZERO FLAG" RESETS VIA THE "OR" INSTRUCTION, CAUSING THE LOOP TO END! THEN THE "EI" INSTRUCTION STARTS THE SCREEN INTERRUPT IN "AB" NOW THAT "AB" IS PRESENT (IN THE !2XXX AREA)! THIS IS FOLLOWED BY THE CALL TO !2D06! I STILL DON'T KNOW WHY, BUT WE'RE PLAYING IT SAFE (SEE THE "BEYOND REPACK" TUTORIAL WHERE WE SAW THIS SEQUENCE BEFORE). THEN WE "LOAD DE WITH !4EBD" AND PUSH THAT ONTO THE STACK SO THAT THE TAPING INTERFACE WORKS (SEE ABOVE). "DE" IS THEN DECREMENTED TO !4EBC (20156 DEC.) WHICH IS THE CORRECT ADDRESS FOR "AUTO RUN" IN "AB". THEN WE JUMP INTO "AB" AT AB'S OWN "WARM START" ADDRESS (!2563) [JUST LIKE BEFORE]!

NOW THAT YOU'VE GOT A COMPLETE "WALK THROUGH" OF THE "AB 4x2 MULTICART LOADER", DO YOU NOW UNDERSTAND THE "BRB 2x4 BASICART LOADER" OF LAST TIME? IF NOT, WRITE ME! MY ADDRESS (AND ETC.) WAS IN THE TUTORIAL "FILE SEARCH II"! FINALLY, HERE'S A BASIC PROGRAM FOR YOU TO KEY IN AND TRY [AB ONLY!]:

AMERICAN FLAG

BY BARRY ELLERSON

(ARCADIAN VOL.6 PG.76)

SIMPLIFIED

BY MIKE WHITE

```
>10 CLEAR
>20 T=20000
>30 L=%(T)
>40 S=20258
>50 %(S)=15093
>60 %(S+2)=20055
>70 %(S+4)=25342
>80 %(S+6)=-14274
>90 %(S+8)=552
>100 %(S+10)=25150
>110 %(S+12)=22322
>120 %(S+14)=-11442
>130 %(S+16)=15887
>140 %(S+18)=10328
```

```
>150 %(S+20)=15874
>160 %(S+22)=12808
>170 %(S+24)=20054
>180 %(S+26)=-15375
>190 %(S+28)=8701
>200 &(0)=88
>210 &(1)=88
>220 &(2)=7
>230 &(3)=7
>240 %(T)=0
>250 &(10)=182
>260 Z=3
>270 &(9)=128
>280 FOR Y=33TO 5STEP -14
>290 BOX 34,Y,92,7,1
>300 NEXT Y
>310 FOR Y=YTO -40STEP -14
>320 BOX 0,Y,160,7,1
>330 NEXT Y
>340 FC=7
>350 %(20118)=S
>360 &(9)=145
>370 FOR Y=39TO -5STEP -5
>380 Z=-Z
>390 FOR X=Z-74TO -16STEP 12
>400 BOX X,Y,5,4,1
>410 BOX X,Y,5,2,3
>420 BOX X,Y,3,4,3
>430 BOX X,Y-1,1,1,3
>440 BOX X,Y+2,1,2,3
>450 NEXT X
>460 NEXT Y
>470 IF KP%(T)=L
```

```
DISSASEMBLY OF MACHINE CODE:
          ↓↓↓↓

20258 F5 PUSH AF <"BC" DIVIDER>
20259 3A LD A,(4E57)
20260 57 [FROM UPPER]
20261 4E [BYTE OF "BC"]
20262 FE CP 62
20263 62 [LOCATE SCAN]
20264 3E LD A,C8
20265 C8 [SET FOR BOTTOM]
20266 28 JR Z,02
20267 02 [TO SET TRIGGER]
20268 3E LD A,62
20269 62 [SET FOR CENTER]
20270 32 LD (4E57),A <SET TRIGGER>
20271 57 [TO UPPER]
20272 4E [BYTE OF "BC"]
20273 D3 OUT (0F)
20274 0F [TO &(15)]
20275 3E LD A,58
20276 58 [SET FOR BLUE]
20277 28 JR Z,02
20278 02 [TO SET "BC"]
20279 3E LD A,08
20280 08 [SET FOR RED]
20281 32 LD (4E56),A <SET "BC">
20282 56 [TO LOWER]
20283 4E [BYTE OF "BC"]
20284 F1 POP AF
20285 C3 JP 21FD
20286 FD [CONTINUE INTERRUPT]
20287 21 [SERVICE ROUTINE]
```

THIS IS THE SIMPLE WAY TO DO BARRY ELLERSON'S FLAG! THE STARS (BEING COPYS OF THE ONES BARRY DID) AND THE WHITE STRIPS ARE BOTH DONE WITH "BOX " COMMANDS. &(2), &(3), AND "FC" ARE ALL SET TO WHITE. &(0) AND &(1) ARE SET TO RED, AND &(9) IS SET TO AN "FC" BOARDER SO THAT THE EDGES OF THE FLAG CAN BE DISTINGUISHED! THEN IT CHANGES TO DIVIDE THE SCREEN HORIZONTALLY FOR THE BLUE FIELD. THE STRIPES ARE 7 PIXELS HIGH (AND 13x7=91 NOT 88 OR LESS), MEANING THAT &(10) HAD TO BE LOWERED TO 182 AND THAT EXPOSES PART OF THE SCRATCHPAD! VARIABLES "A" THROUGH "K" ARE VISIBLE AND MUST REMAIN AT ZERO. THE "TXTUNF" IS ALSO VISIBLE, AND GETS COPIED TO "L" AND THEN ZEROED OUT SO OUR FLAG LOOKS RIGHT (SEE LINES 30+240, AND 470 WHERE IT GETS RESTORED). NOTE: THERE IS NO "CALL" COMMAND! THE MACHINE CODE RESIDES AT 20258 (END OF STACK AREA) AND "SPLICES" INTO THE SCREEN INTERRUPT SERVICE ROUTINE AT LINE 350. THIS CHANGES "BC" DYNAMICALLY, TO DIVIDE THE LEFT SIDE OF THE SCREEN VERTICALLY FROM BLUE TO RED! NEXT TIME, WE'LL "WALK THROUGH" THIS 30 BYTE ROUTINE. UNTILL THEN, KEEP BUGGIN'! (NEXT TIME WILL END THIS SERIES, UNLESS QUESTIONS ARISE!)