

M. David Johnson
<http://www.bds-soft.com>
info@bds-soft.com

Towards

Y

(Yet)

A

(Another)

C

(Chess)

H

(Handler)

by M. David Johnson
2021/09/29

Abstract

Early efforts towards developing YACH (Yet Another Chess Handler) are documented and presented. A plan and procedure for continued progress is outlined. The ultimate goal of encouraging new and potential coders and programmers to enjoy the fun of improving their skills with Assembly Language programming is discussed.

—

This paper and its associated code are available online at:

<http://www.bds-soft.com/cocoPapers.php> .

====

Table of Contents

Abstract	2
Introduction	10
General Methodology	11
ML Foundation	14
DU1616	15
(16-bit by 16-bit Unsigned Integer Divide)	
FUDWRD	18
(Form a Five-Character Decimal Representation of a 16-bit Unsigned Integer)	
PTDWRD	21
(Put a 16-bit Unsigned Integer To the VIDRAM Screen As Up To Five Decimal Digits At a Specific Position)	
YACH Organization	28
FLAGS	32
(System Flags)	
TMPVAR	35
(Temporary Variables)	
MAKEFT	37
(Make the Flags and Temporary Variables FT.BIN Combined File)	
Screen Management	38
CIDGEN	39
(Generates the Chess Board Initial Display Screen)	
B2SACALC	57
(Chess Board to Screen Translation Array Calculation Utility Program)	

B2SARY	60
(Board to Screen Translation Array)	
B2DARY	63
(Board Dark Squares to Screen Translation Array)	
B2LARY	65
(Board Light Squares to Screen Translation Array)	
BSCARY	67
(Board Surface Contents Array)	
MAKESM	69
(Make the Screen Management SM.BIN Combined File)	
General Reporting	70
NOTREC	71
(Command Not Recognized)	
NYIMSG	73
(Not Yet Implemented Postamble)	
MAKEGR	74
(Make the General Reporting GR.BIN Combined File)	
General Commands	75
ZERO2	77
(Castle Kingside Actual Command = 0-0 - Transfer Command)	
ZERO3	78
(Castle Queenside Actual Command = 0-0-0 - Transfer Command)	
ANLYZ	79
(Analyze the Board)	
ASKFR	83
(Ask for Information About the Square at F = File Character R = Rank Character)	

CLRBD	87
(Clears the Board)	
CKMOV	92
(Check the Legalty Of a Proposed Move)	
CKPOS	96
(Check the Legality Of the Displayed Position)	
CSTLK	100
(Castle Kingside - Action Command)	
CSTLQ	104
(Castle Queenside - Action Command)	
DSPBD	108
(Copies the Board Surface Contents Array to the Display Screen)	
FLIP	113
(Flip the Board)	
GSTRT	117
(Restart Accepting Commands After a STOP - Action Command)	
GSTOP	118
(Ignore All Subsequent Commands Except START - Action Command)	
MAKEGC1	119
(Make the General Commands Group One GC1.BIN Combined File)	
LDGAM	120
(Load a Game)	
LDPOS	124
(Load a Position)	
NEW	128
(Sets a New Game Position - Transfer Command)	

NEWPS	129
(Sets a New Game Position - Action Command)	
LTRO2	134
(Castle Kingside Actual Command = O-O - Transfer Command)	
LTRO3	135
(Castle Queenside Actual Command = O-O-O - Transfer Command)	
QUIT	136
(Quit the Game and Return to the Command Prompt)	
SCOFF	137
(Turns Screen Colors Off)	
SCON	141
(Turns Screen Colors On)	
START	149
(Restart Accepting Commands After a STOP - Transfer Command)	
STOP	150
(Ignore All Subsequent Commands Except START - Transfer Command)	
SVGAM	151
(Save the Game)	
SVPOS	155
(Save the Position)	
TKBAK	159
(Take Back the Last Move)	
MAKEGC2	163
(Make the General Commands Group Two GC2.BIN Combined File)	

Movement Commands and Calculations	164
FILCVT	165
(Converts a Chess Board File Character ("A" to "H") to an Array I-Value (0 to 7) i.e. an x-coordinate)	
RNKCVT	170
(Converts a Chess Board Rank Character ("1" to "8") to an Array J-Value (0 to 7) i.e. a y-coordinate)	
FREO	175
(Empty the Square at F = File Character R = Rank Character Actual Command = FR=0 - Transfer Command)	
FRECP	176
(Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character Actual Command = FR=CP - Transfer Command)	
FRFR	177
(Standard Move From F = File Character R = Rank Character To F = File Character R = Rank Character)	
FRFRE	182
(Standard Move Plus Capture En Passant)	
FRFRP	186
(Standard Move Plus Pawn Promotion)	
FRQO	190
(Empty the Square at F = File Character R = Rank Character - Action Command)	

FRQCP	194
(Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character - Action Command)	
MAKEMC	199
(Make the Movement Commands and Calculations MC.BIN Combined File)	
Command Management	200
GETCMD	201
(Get a Command)	
MATCH3	208
(Checks for a Three-Character Command Match)	
MATCH4	213
(Checks for a Four-Character Command Match)	
MATCH5	218
(Checks for a Five-Character Command Match)	
XLTCMD	223
(Translate and Dispatch a Command - The First Third)	
XLTCM2	233
(Translate and Dispatch a Command - The Second Third)	
XLTCM3	243
(Translate and Dispatch a Command - The Final Third)	
MAKECM	255
(Make the Command Management CM.BIN Combined File)	

YACH Supervisory	256
GMLOOP	257
(The Game Loop)	
YACH.ASM	261
(Game Entry Routine - Set Up the Board and Enter the Game Loop)	
MAKEYS	266
(Make the YACH Supervisory YS.BIN Combined File)	
YACH.BAS	267
(The Game Control Program)	

Appendix A: Decimal to Hexadecimal Conversions	269
Appendix B: My 64K CoCo 2 System	271
Appendix C: My 128K CoCo 3 System	273
Appendix D: My 512K CoCo 3 System	275
Appendix E: My CoCo Philosophy	277
Appendix F: New BDS Software License	279

Works Cited	280
-------------------	-----

=====

Introduction

YACH (Yet Another Chess Handler) is intended to eventually help users play Chess against themselves and others, to go over and play through published Chess Games and Chess Problems, and perhaps to develop their own Chess-Like puzzles and games.

Consider this as a rather esoteric example of the latter: two players. One has 32 White Queens. The other has 32 Black Queens. Player One puts a White Queen on the Board. Player Two puts a Black Queen on the Board. They alternate until all 64 Queens are on the Board. Then White makes a move (necessarily of the form QxQ). Then Black makes a move. The Last Queen Standing wins!

This paper documents just the very beginnings of my attempts to accomplish those combined goals.

Much of this paper is just bare Assembly code without much (if any) external explanation. However, the code is well documented and hopefully won't be too difficult to follow. This brevity is partially due to the looming deadline for its submission. (ACK! It's due TOMORROW??).

But, the brevity is not a bug: it's a FEATURE. You see, I want new and potential coders and programmers to learn the joy of searching code on their own. (QUICK - Is anybody actually buying this argument??).

Seriously though, I would indeed appreciate hearing about your experience with this presentation, as part of its purpose really is to help lead those new coders and programmers into the fun of what we do. And, exploring others' code is part of that learning process.

A Note on Numbers: To keep everything simple to understand, and also neatly lined-up, I generally refer to numbers as decimal bytes with three full digits, e.g. 004, 027, 229, etc. See Appendix A for conversions between the decimal and hexadecimal representations of bytes.

In works of this complexity (at least for me) typos and other errors are bound to sneak in. Please let me know about any you discover so I can note and correct them.

M.D.J. 2021/09/29
info@bds-soft.com

=====

General Methodology

First of all, YACH will *never* be a full-blown Chess Server - that's simply not practical in 64K. But I do hope it will someday be helpful and fun as a tool for playing the game. And, I also hope the code provided will be helpful to new coders and programmers as they seek to learn new ways of doing things; including, perhaps, some coding practices which are best to avoid.

I plan to develop YACH in five stages.

This paper is the **FIRST STAGE**. It represents a fairly substantial Proof-Of-Concept. In it, I briefly, but not completely, describe and define the Board, the Pieces, and the Commands. These are then translated into memory definitions, code stubs, Command Management Controls, a Game Loop, and an Entry Routine. The resulting, although rudimentary, system is hopefully sufficient to demonstrate at least the possibility of my ultimate goal. At the moment, only the following six user commands actually work:

CLRBD: Clears the Board
NEW: Sets a New Game Position Transfer Command
NEWPS: Sets a New Game Position Action Command
QUIT: Quit the Game and Return to the Command Prompt
SCOFF: Turns Screen Colors Off
SCON: Turns Screen Colors On

and there's a bug in the synonyms NEW and NEWPS: somehow, they introduce an extraneous asterisk at B3.

But, even in this buggy and abbreviated beginning, I hope you'll notice one of the major features of my approach: modularity. The code is organized into Seven Sections, each of which includes several individual Assembly Language modules. In addition, each Section includes a **MAKEXX.BAS** program to gather those modules into an **XX.BIN** combined Machine Language file. This makes it possible to get all the **LOADM** commends into the very tight **YACH.BAS** Control Program file. (Pleeze don't ask how I learned this would be necessary!!).

TRANSFER COMMANDS AND ACTION COMMANDS

Many of the commands presented here are just what they seem: straightforward commands.

But some commands are identified as "Transfer Commands" while others are designated "Action Commands". The purpose of this is to allow commands to be entered in different forms as may be convenient to the user.

For example, do you prefer to enter the Castling commands as 0-0 and 0-0-0 (using zeroes), or as O-O and O-O-O (using the letter "O")? With YACH, you get to choose whichever is most convenient for you.

The Transfer Commands simply transfer control to their associated Action Commands, often acting as an intermediary. For example, when you type-in the command "0-0", it executes the Transfer Command Routine "**ZERO2**" which then transfers control to the Action Command Routine "**CSTLK**".

You can also enter the Action Commands directly if you wish.

These association sets are:

Transfer Command =====	Action Command =====
0-0 -> ZERO2 ->	CSTLK
O-O -> LTRO2 ->	CSTLK
0-0-0 -> ZERO3 ->	CSTLQ
O-O-O -> LTRO3 ->	CSTLQ
START	GSTRT
STOP	GSTOP
NEW	NEWPS
FR=0 -> FRE0 ->	FRQ0
FR=CP -> FRECP ->	FRQCP

In the **SECOND STAGE** of this project, I plan to complete a subset of the Command codes; a collection sufficient to allow solitaire play, or hot-seat play, somewhat as if using a physical chess set. That is to say that the system at this stage will be just like a physical set: a dumb board and a box of dumb pieces.

Although the YACH System uses typed-in commands instead of a graphical drag-and-drop mechanism, you can get a general idea of what I mean by a dumb board and a dumb set of pieces by taking a look at the similar concept in JavaScript and php on my website at:

<http://www.bds-soft.com/php/games/ChessHelper/chessHelper.php>

In this stage, I will also enable saving and loading of games and positions, using Chess standard fen and pgn notations..

In the **THIRD STAGE**, I plan to complete the remainder of the commands which will provide the ability to check moves and positions for Chess Legality, and which will provide some analytical helps to users.

In the **FOURTH STAGE**, I hope to incorporate play between two different players, playing at two different CoCo's, with the CoCo's communicating with each other via RS-232, or perhaps some other mechanism.

And finally, in the **FIFTH STAGE**, I hope to allow a player seated at a CoCo to play against a Chess Server running on some other machine, either local (perhaps a Raspberry Pi?) or remote.

Meanwhile, I'm having a lot of fun developing this system: I hope you'll get some enjoyment from it too. To try it out, just insert the game disk in Drive 0, do

RUN "YACH.BAS"

and type your commands in the white box (you don't have to pre-position the cursor: just start typing).

("**N.Y.I.**" = Not Yet Implemented).

Also, please note that, at this point, little or no formal testing of the System or it's components has been attempted. That will come later. For now: what works just does. And what doesn't... well...

=====

The ML Foundation - Additions

In preparation for developing the YACH Chess System, three new routines were added to the ML Foundation Collection (MDJ01):

1. DU1616: 16-bit by 16-bit Unsigned Integer Divide
2. FUDWRD: Form a Five-Character Decimal Representation of a 16-bit Unsigned Integer
3. PTDWRD: Put a 16-bit Unsigned Integer To the VIDRAM Screen As Up To Five Decimal Digits At a Specific Position

These routines will become a standard part of the ML Foundation at its next update.

They are included here because they provide a needed service to the YACH System.

=====

DU1616: 16-bit by 16-bit Unsigned Integer Divide

In addition to being able to output a 16-bit unsigned integer in hexadecimal format using **PUTWRD**, it will be helpful to be able to output such data in decimal format. This module (**DU1616**) and the following two modules (**FUDWRD** and **PTDWRD**) accomplish that.

This routine is needed for **PTDWRD** (Put Decimal Word).

```
00100 *****
00110 *
00120 * DU1616.ASM
00130 * MDJ 2021/09/16
00140 *
00150 * 16-BIT BY 16-BIT
00160 * UNSIGNED INTEGER
00170 * DIVIDE
00180 *
00190 * RELOCATABLE
00200 *
00210 * CF. (BARROW 31)
00220 *
00230 * ENTRY CONDITIONS:
00240 *   X = DIVIDEND
00250 *   Y = DIVISOR
00260 *
00270 * EXIT CONDITIONS:
00280 *   X = QUOTIENT
00290 *   Y = REMAINDER
00300 *
00310 * IF DIVIDE BY ZERO:
00320 *   X = $FFFF
00330 *   Y = DIVIDEND
00340 *
00350 *****
00360
00370 * LOW RAM CURSOR ADDRESS
0088 00380 CURPOS EQU    $0088
00390
00400 * SCREEN ADDRESSES
00410 * START OF VIDRAM
0400 00420 VIDRAM EQU    $0400
00430
00440 * ONE BYTE PAST THE
00450 * END OF VIDRAM
```

```

0600      00460 VIDEND EQU      $0600
          00470
          00480 * ML FOUNDATION
          00490 * CORE ADDRESSES
1C00      00500 REGXFR EQU      $1C00
1C0E      00510 VIDCLS EQU      $1C0E
1C1F      00520 PUTCHR EQU      $1C1F
1CD2      00530 GETCHR EQU      $1CD2
1CD5      00540 PUTBYT EQU      $1CD5
1D17      00550 SCROLL EQU      $1D17
1D36      00560 PUTCHA EQU      $1D36
1D57      00570 PUTBYA EQU      $1D57
1D8C      00580 CRLF EQU      $1D8C
1DBD      00590 PK2PRT EQU      $1DBD
1DDA      00600 PRT2PK EQU      $1DDA
1DF7      00610 POLCAT EQU      $1DF7
1E06      00620 PUTWRA EQU      $1E06
1E11      00630 PUTWRD EQU      $1E11
1E1C      00640 BKSPCE EQU      $1E1C
          00650
          00660 * PROGRAM BEING DEVELOPED
          00670 * PROGRAM ORIGIN:
          00680 * (CHANGE AS NEEDED)
1E34      00690          ORG      $1E34
          00700
1E34 34   37   00710 DU1616 PSHS    A,B,X,Y,CC
          00720
          00730 * SET COUNT FOR 16-BIT
          00740 * SHIFTS ON STACK
1E36 C6   10   00750          LDB      #16
1E38 34   04   00760          PSHS    B
1E3A 4F           00770          CLRA
1E3B 5F           00780          CLRB
          00790
          00800 * LOOP 16 TIMES AND
          00810 * FORM THE QUOTIENT
          00820 * ONE BIT AT A TIME
1E3C 68   65   00830 L0001  ASL      5,S
1E3E 69   64   00840          ROL      4,S
1E40 59           00850          ROLB
1E41 49           00860          ROLA
          00870
          00880 * CAN DIVISOR BE SUBTRACTED?
1E42 10A3 66   00890          CMPD    6,S
          00900
          00910 * GO IF NO
1E45 25   04   00920          BLO      L0002

```


			00930
			00940 * DO THE SUBTRACTION
1E47	A3	66	00950 SUBD 6,S
1E49	6C	65	00960 INC 5,S
			00970
			00980 * CONTINUE UNTIL DONE
1E4B	6A	E4	00990 L0002 DEC ,S
1E4D	26	ED	01000 BNE L0001
			01010
			01020 * PUT REMAINDER TO STACKED Y
1E4F	ED	66	01030 STD 6,S
			01040
			01050 * DISCARD THE COUNT BYTE
1E51	32	61	01060 LEAS 1,S
			01070
			01080 * PULL THE RESULTS AND EXIT
1E53	35	37	01090 PULS A,B,X,Y,CC
1E55	39		01100 RTS
		0000	32767 END

00000 TOTAL ERRORS

Testing: DU1616 and FUDWRD will be tested along with PTDWRD in the second following Section (PTDWRD).

=====

FUDWRD: Form a Five-Character Decimal Representation of a 16-bit Unsigned Integer

This routine is also needed for **PTDWRD** (Put Decimal Word).

```
00100 *****
00110 *
00120 * FUDWRD.ASM
00130 * MDJ 2021/09/16
00140 *
00150 * FORM UNSIGNED
00160 * DECIMAL WORD:
00170 *   CONVERT A
00180 *   16-BIT UNSIGNED
00190 *   BINARY INTEGER INTO
00200 *   A FIVE-CHARACTER
00210 *   DECIMAL
00220 *   REPRESENTATION
00230 *
00240 * RELOCATABLE
00250 * (CHECK CALL ADDRESSES)
00260 *
00270 * CALLS DU1616
00280 *
00290 * ENTRY CONDITIONS:
00300 *   X = THE NUMBER
00310 *
00320 * EXIT CONDITIONS:
00330 *   SCRATCHPAD
00340 *   VARIABLES $F3 - $F7
00350 *   = THE DECIMAL
00360 *   CHARACTERS
00370 *   REPRESENTATION
00380 *
00390 * THIS ROUTINE PRESUPPOSES
00400 *   DP = $00
00410 *
00420 *****
00430
00440 * LOW RAM CURSOR ADDRESS
0088 00450 CURPOS EQU $0088
00460
00470 * SCREEN ADDRESSES
```

```

00480 * START OF VIDRAM
0400 00490 VIDRAM EQU $0400
00500
00510 * ONE BYTE PAST THE
00520 * END OF VIDRAM
0600 00530 VIDEND EQU $0600
00540
00550 * ML FOUNDATION
00560 * CORE ADDRESSES
1C00 00570 REGXFR EQU $1C00
1C0E 00580 VIDCLS EQU $1C0E
1C1F 00590 PUTCHR EQU $1C1F
1CD2 00600 GETCHR EQU $1CD2
1CD5 00610 PUTBYT EQU $1CD5
1D17 00620 SCROLL EQU $1D17
1D36 00630 PUTCHA EQU $1D36
1D57 00640 PUTBYA EQU $1D57
1D8C 00650 CRLF EQU $1D8C
1DBD 00660 PK2PRT EQU $1DBD
1DDA 00670 PRT2PK EQU $1DDA
1DF7 00680 POLCAT EQU $1DF7
1E06 00690 PUTWRA EQU $1E06
1E11 00700 PUTWRD EQU $1E11
1E1C 00710 BKSPCE EQU $1E1C
1E34 00720 DU1616 EQU $1E34
00730
00740 * SCRATCHPAD VARIABLES
00F3 00750 DRCHR4 EQU $F3
00F4 00760 DRCHR3 EQU $F4
00F5 00770 DRCHR2 EQU $F5
00F6 00780 DRCHR1 EQU $F6
00F7 00790 DRCHR0 EQU $F7
00800
00810 * PROGRAM BEING DEVELOPED
00820 * PROGRAM ORIGIN:
00830 * (CHANGE AS NEEDED)
1E56 00840 ORG $1E56
00850
1E56 34 64 00860 FUDWRD PSHS B,Y,U
00870
00880 * SET CHARACTER COUNT
1E58 C6 05 00890 LDB #5
00900
00910 * SET CHARACTER POINTER
1E5A CE 00F7 00920 LDU #DRCHR0+1
00930
00940 * SET DIGITAL DIVISOR

```

```

1E5D 108E 000A      00950 L0001  LDY      #10
                   00960
                   00970 * GO DO THE DIVISION
1E61 17    FFD0     00980          LBSR      DU1616
                   00990
                   01000 * ADD POKE MECHANISM
                   01010 * ZERO OFFSET TO DIGIT
1E64 31    A8 70    01020          LEAY      112,Y
                   01030
                   01040 * STORE CHARACTER RESULT
1E67 10AF C2       01050          STY        ,-U
                   01060
                   01070 * ARE WE DONE?
1E6A 5A           01080          DECB
                   01090
                   01100 * GO IF NO
1E6B 26    F0       01110          BNE        L0001
                   01120
                   01130 * EXIT
1E6D 35    64       01140          PULS      B,Y,U
1E6F 39           01150          RTS
                   0000      32767          END

```

00000 TOTAL ERRORS

Testing: DU1616 and FUDWRD will be tested along with PTDWRD in the following Section (PTDWRD).

=====

PTDWRD: Put a 16-bit Unsigned Integer To the VIDRAM Screen As Up To Five Decimal Digits At a Specific Position

PTDWRD accomplishes the aforementioned decimal display of 16-bit unsigned integers, without cursor advance or scrolling.

```
00100 *****
00110 *
00120 * PTDWRD.ASM
00130 * MDJ 2021/09/16
00140 *
00150 * PUT A 16-BIT
00160 * UNSIGNED INTEGER
00170 * TO VIDRAM AS UP TO
00180 * FIVE DECIMAL DIGITS
00190 *
00200 * RELOCATABLE
00210 * (CHECK CALL ADDRESSES)
00220 *
00230 * CALLS FUDWRD
00240 *
00250 * ENTRY CONDITIONS:
00260 *   D = THE 16-BIT NUMBER
00270 *   X = SCREEN LOCATION
00280 *       ($0400 - $05FB)
00290 *       CANNOT BE MORE
00300 *       THAN $05FB
00310 *       BECAUSE NEED
00320 *       ROOM TO PUT
00330 *       5 CHARACTERS
00340 *
00350 * EXIT CONDITIONS:
00360 *   X = NEW SCREEN LOC
00370 *       ($0405 - $0600)
00380 *       $0600 INDICATES
00390 *       END OF VIDRAM
00400 *       HAS BEEN PASSED
00410 *
00420 * THIS ROUTINE PRESUPPOSES
00430 *   DP = $00
00440 *
```

```

00450 *****
00460
00470 * LOW RAM CURSOR ADDRESS
0088 00480 CURPOS EQU $0088
00490
00500 * SCREEN ADDRESSES
00510 * START OF VIDRAM
0400 00520 VIDRAM EQU $0400
00530
00540 * ONE BYTE PAST THE
00550 * END OF VIDRAM
0600 00560 VIDEND EQU $0600
00570
00580 * ML FOUNDATION
00590 * CORE ADDRESSES
1C00 00600 REGXFR EQU $1C00
1C0E 00610 VIDCLS EQU $1C0E
1C1F 00620 PUTCHR EQU $1C1F
1CD2 00630 GETCHR EQU $1CD2
1CD5 00640 PUTBYT EQU $1CD5
1D17 00650 SCROLL EQU $1D17
1D36 00660 PUTCHA EQU $1D36
1D57 00670 PUTBYA EQU $1D57
1D8C 00680 CRLF EQU $1D8C
1DBD 00690 PK2PRT EQU $1DBD
1DDA 00700 PRT2PK EQU $1DDA
1DF7 00710 POLCAT EQU $1DF7
1E06 00720 PUTWRA EQU $1E06
1E11 00730 PUTWRD EQU $1E11
1E1C 00740 BKSPCE EQU $1E1C
1E34 00750 DU1616 EQU $1E34
1E56 00760 FUDWRD EQU $1E56
00770
00780 * SCRATCHPAD VARIABLES
00F3 00790 DRCHR4 EQU $F3
00F4 00800 DRCHR3 EQU $F4
00F5 00810 DRCHR2 EQU $F5
00F6 00820 DRCHR1 EQU $F6
00F7 00830 DRCHR0 EQU $F7
00840
00850 * FORM DECIMAL WORD
00860 * ADDRESS
00870
00880 * PROGRAM BEING DEVELOPED
00890 * PROGRAM ORIGIN:
00900 * (CHANGE AS NEEDED)
1E70 00910 ORG $1E70

```

			00920		
1E70	34	40	00930	PTDWRD	PSHS U
			00940		
			00950	* SAVE THE SCREEN POINTER	
1E72	34	10	00960	PSHS	X
			00970		
			00980	* PREPARE; THEN GO DO	
			00990	* FORM DIGITAL WORD	
1E74	1F	01	01000	TFR	D,X
1E76	17	FFDD	01010	LBSR	FUDWRD
			01020		
			01030	* SET CHARACTER COUNT	
1E79	C6	05	01040	LDB	#5
			01050		
			01060	* SET CHARACTER POINTER	
1E7B	CE	00F3	01070	LDU	#DRCHR4
			01080		
			01090	* RESTORE THE SCREEN POINTER	
1E7E	35	10	01100	PULS	X
			01110		
			01120	* PROCESS A CHARACTER	
1E80	A6	C0	01130	L0001	LDA ,U+
			01140		
			01150	* SKIP LEADING ZEROES	
1E82	81	70	01160	CMPA	#112
1E84	27	02	01170	BEQ	L0002
			01180		
			01190	* DISPLAY THE CHARACTER	
			01200	* AND ADVANCE THE SCREEN	
			01210	* POINTER	
1E86	A7	80	01220	STA	,X+
			01230		
			01240	* ARE WE DONE?	
1E88	5A		01250	L0002	DECB
			01260		
			01270	* GO IF NO	
1E89	26	F5	01280	BNE	L0001
			01290		
			01300	* EXIT	
1E8B	35	40	01310	PULS	U
1E8D	39		01320	RTS	
		0000	32767	END	

00000 TOTAL ERRORS

The Assembly Language Test Routine:

```
00100 *****
00110 *
00120 * MTST0001.ASM
00130 * MDJ 2021/09/17
00140 *
00150 * TEST OF
00160 * PTDWRD.ASM
00170 *
00180 *****
00190
00200 * LOW RAM CURSOR ADDRESS
0088 00210 CURPOS EQU $0088
00220
00230 * SCREEN ADDRESSES
00240 * START OF VIDRAM
0400 00250 VIDRAM EQU $0400
00260
00270 * ONE BYTE PAST THE
00280 * END OF VIDRAM
0600 00290 VIDEND EQU $0600
00300
00310 * ML FOUNDATION
00320 * CORE ADDRESSES
1C00 00330 REGXFR EQU $1C00
1C0E 00340 VIDCLS EQU $1C0E
1C1F 00350 PUTCHR EQU $1C1F
1CD2 00360 GETCHR EQU $1CD2
1CD5 00370 PUTBYT EQU $1CD5
1D17 00380 SCROLL EQU $1D17
1D36 00390 PUTCHA EQU $1D36
1D57 00400 PUTBYA EQU $1D57
1D8C 00410 CRLF EQU $1D8C
1DBD 00420 PK2PRT EQU $1DBD
1DDA 00430 PRT2PK EQU $1DDA
1DF7 00440 POLCAT EQU $1DF7
1E06 00450 PUTWRA EQU $1E06
1E11 00460 PUTWRD EQU $1E11
1E1C 00470 BKSPCE EQU $1E1C
1E34 00480 DU1616 EQU $1E34
1E56 00490 FUDWRD EQU $1E56
1E70 00500 PTDWRD EQU $1E70
00510
00520 * TEST ROUTINE
00530 * PROGRAM ORIGIN:
00540 * (CHANGE AS NEEDED)
```



```

7000          00550          ORG          $7000
              00560
7000 34      16          00570          PSHS          A,B,X
              00580
              00590 * TEST VALUE
7002 CC      4ABC      00600          LDD          #$4ABC
              00610
              00620 * SAVE THE TEST VALUE
7005 34      06          00630          PSHS          A,B
              00640
              00650 * PUT AS HEXADECIMAL
7007 17      AD82      00660          LBSR          CRLF
700A 17      ADF9      00670          LBSR          PUTWRA
700D 17      AD7C      00680          LBSR          CRLF
              00690
              00700 * RESTORE THE TEST VALUE
7010 35      06          00710          PULS          A,B
              00720
              00730 * GET THE CURSOR
7012 9E      88          00740          LDX          $88
              00750
              00760 * PUT AS DECIMAL
7014 17      AE59      00770          LBSR          PTDWRD
7017 17      AD72      00780          LBSR          CRLF
              00790
              00800 * EXIT
701A 35      16          00810          PULS          A,B,X
701C 39          00820          RTS
              0000          32767          END

```

00000 TOTAL ERRORS

The BASIC Language Control Program:

```

1000 '*****
1010 '*
1020 '* MTST0001.BAS
1030 '* MDJ 2021/09/17
1040 '*
1050 '* TEST OF
1060 '* PTDWRD.ASM
1070 '*
1080 '*****
1090 '
1100 'SETUP MEMORY
1110 PCLEAR 1
1120 CLEAR 200, &H1C00

```

```

1130 '
1140 'LOAD ML FOUNDATION
1150 'CORE ROUTINES
1160 LOADM "REGXFR.BIN"
1170 LOADM "VIDCLS.BIN"
1180 LOADM "PUTCHR.BIN"
1190 LOADM "GETCHR.BIN"
1200 LOADM "PUTBYT.BIN"
1210 LOADM "SCROLL.BIN"
1220 LOADM "PUTCHA.BIN"
1230 LOADM "PUTBYA.BIN"
1240 LOADM "CRLF.BIN"
1250 LOADM "PK2PRT.BIN"
1260 LOADM "PRT2PK.BIN"
1270 LOADM "POLCAT.BIN"
1280 LOADM "PUTWRA.BIN"
1290 LOADM "PUTWRD.BIN"
1300 LOADM "BKSPCE.BIN"
1310 LOADM "DU1616.BIN"
1320 LOADM "FUDWRD.BIN"
1330 LOADM "PTDWRD.BIN"
1340 '
1350 'LOAD THE TEST ROUTINE
1360 LOADM "MTST0001.BIN"
1370 '
1380 'ENTER A 16-BIT
1390 'UNSIGNED INTEGER
1400 'TEST VALUE BETWEEN
1410 '0 AND 65535
1420 INPUT TV
1430 '
1440 'INSERT THE VALUE INTO
1450 'THE MTST0001,BIN
1460 'TEST ERoutine
1470 T1 = INT(TV / 256)
1480 T2 = TV - (T1 * 256)
1490 POKE &H7003, T1
1500 POKE &H7004, T2
1510 '
1520 'GO DO THE TEST
1530 EXEC &H7000
1540 '
1550 'GO GET ANOTHER
1560 'TEST VALUE
1570 GOTO 1420
1580 '
32767 END

```

Results:

? 0

0000
00000

? 65535

FFFF
65535

? 19132

4ABC
19132

? 1856

0740
01856

etc.

All as expected.

=====

YACH Organization

For ease of keeping track of what I was doing, the YACH Chess System has been divided into seven Sections, each of which contains several separate routines (aka modules). These Sections and Routines are:

Section One: Flags and Temporary Variables

1. FLAGS: System Flags
2. TMPVAR: Temporary Variables

Section Two: Screen Management

1. CIDGEN: Generates the Chess Board Initial Display Screen
2. B2SACALC: Chess Board to Screen Translation Array Calculation Utility Program
3. B2SARY: Board to Screen Translation Array
4. B2DARY: Board Dark Squares to Screen Translation Array
5. B2LARY: Board Light Squares to Screen Translation Array
6. BSCARY: Board Surface Contents Array

Section Three: General Reporting

1. NOTREC: Command Not Recognized
2. NYIMSG: Not Yet Implemented Postamble

Section Four: General Commands

1. ZERO2: Castle Kingside Actual Command = 0-0 Transfer Command
2. ZERO3: Castle Queenside Actual Command = 0-0-0 Transfer Command
3. ANLYZ: Analyze the Board (N.Y.I. - Not Yet Implemented)
4. ASKFR: Ask for Information About the Square at F = File Character R = Rank Character (N.Y.I. - Not Yet Implemented)
5. CLRBD: Clears the Board

6. CKMOV: Check the Legality Of a Proposed Move (N.Y.I. - Not Yet Implemented)
7. CKPOS: Check the Legality Of the Displayed Position (N.Y.I. - Not Yet Implemented)
8. CSTLK: Castle King-Side Action Command (N.Y.I. - Not Yet Implemented)
9. CSTLQ: Castle Queen-Side Action Command (N.Y.I. - Not Yet Implemented)
10. DSPBD: Copies the Board Surface Contents Array to the Display Screen
11. FLIP: Flip the Board (N.Y.I. - Not Yet Implemented)
12. GSTRT: Restart Accepting Commands After a GSTOP Action Command
13. GSTOP: Ignore All Subsequent Commands Except GSTRT Action Command
14. LDGAM: Load a Game (N.Y.I. - Not Yet Implemented)
15. LDPOS: Load a Position (N.Y.I. - Not Yet Implemented)
16. NEW: Sets a New Game Position Transfer Command
17. NEWPS: Sets a New Game Position Action Command
18. LTRO2: Castle Kingside Actual Command = O-O Transfer Command
19. LTRO3: Castle Queenside Actual Command = O-O-O Transfer Command
20. QUIT: Quit the Game and Return to the Command Prompt
21. SCOFF: Turns Screen Colors Off
22. SCON: Turns Screen Colors On
23. START: Restart Accepting Commands After a STOP Transfer Command
24. STOP: Ignore All Subsequent Commands Except START Transfer Command
25. SVGAM: Save the Game (N.Y.I. - Not Yet Implemented)
26. SVPOS: Save the Position (N.Y.I. - Not Yet Implemented)
27. TKBAK: Take Back the Last Move (N.Y.I. - Not Yet Implemented)

Section Five: Movement Commands And Calculations

1. FILCVT: Converts a Chess Board File Character ("A" to "H") to an Array I-Value (0 to 7) i.e. an x-coordinate
2. RNKCVT: Converts a Chess Board Rank Character ("1" to "8") to an Array J-Value (0 to 7) i.e. a y-coordinate
3. FRE0: Empty the Square at F = File Character R = Rank Character Actual Command = FR=0 Transfer Command
4. FRECP: Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character Actual Command = FR=CP Transfer Command
5. FRFR: Standard Move From F = File Character R = Rank Character To F = File Character R = Rank Character (e.g. E2E4)
6. FRFRE: Standard Move Plus Capture En Passant (N.Y.I. - Not Yet Implemented)
7. FRFRP: Standard Move Plus Pawn Promotion (N.Y.I. - Not Yet Implemented)
8. FRQ0: Empty the Square at F = File Character R = Rank Character Action Command (N.Y.I. - Not Yet Implemented)
9. FRQCP: Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character Action Command (N.Y.I. - Not Yet Implemented)

Section Six: Command Management

1. GETCMD: Get a Command
2. MATCH3: Checks for a Three-Character Command Match
3. MATCH4 Checks for a Four-Character Command Match
4. MATCH5 Checks for a Five-Character Command Match
5. XLTCMD: Translate and Dispatch a Command The First Third
6. XLTCM2: Translate and Dispatch a Command The Second Third
7. XLTCM3: Translate and Dispatch a Command The Final Third

Section Seven: YACH Supervisory

1. GMLOOP: The Game Loop
2. YACH.ASM: Game Entry Routine Set Up the Board and Enter the Game Loop
3. YACH.BAS: The Control Program

This is Section One: Flags and Temporary Variables

1. FLAGS: System Flags
2. TMPVAR: Temporary Variables

I prefer to keep Flags and Temporary Variables all in one location, rather than having them attached to the routines which use them. I find this helps to smooth maintenance efforts.

And besides, when two routines use the same variable, how would you decide to which routine the variable should be attached.

=====

FLAGS: System Flags

There are only three flags at the moment, but we allow space for possible and, perhaps, unforeseen requirements for additional flags.

In the same vein, you'll find copious amounts of unused and unassigned memory space throughout this paper.

In keeping with My CoCo Philosophy (see Appendix E below, and also (MDJ03)) and with my desire to Stuff a Wild CoCo, you might think I would try to avoid such "wasted" space.

And, I will...

when I get down to eventually finalizing this project.

But I find that during development, it's almost always better to leave additional space for the unexpected. Have you ever tried to cram four bytes of code into a three byte space? Well... you could... Yes! That's It! Put a **LBRA** into those three bytes, do the four bytes of work where you just **LBRA**'d to, and then **LBRA** back.

But that gets awfully messy and hard to maintain.

And, Oh yeah, it actually uses a total of ten bytes of space: not very efficient after all.

The Screen Colors Flag is a bit of a misnomer. The actual screen colors don't change from their normal black and green at all. But a Chess Board includes both Light Squares and Dark Squares (which may actually be White and Black, or Beige and Green, or Oak and Mahogany, or whatever).

In this project, the Light Squares are always blank green spaces. But the dark squares may be identified by asterisks. A particular player may find this differentiation helpful, but another player may find it distracting. Yet another may find it helpful for some purposes, but distracting for others.

So I'm providing the capability to have the dark squares be either plain green spaces just like the light squares, or to be identified by the asterisks, as the player chooses. The **SCFLAG** keeps track of which choice is currently in force.

The **STFLAG** will allow you to walk away from the game for a few minutes (or longer) without need for concern about an accidental press of a key messing up the whole game (It's that darned CoCo Cat again !!)

The **BFFLAG** keeps track of whether the Chess Board is in "Normal" position (White side is down) or "Flipped" position (Black side down). [For those who may be excessively politically correct, "No, there was no racial connotation intended by that description." (sigh!)]


```

00100 *****
00110 *
00120 * FLAGS.ASM
00130 * MDJ 2021/09/18
00140 *
00150 * RESERVED FOR
00160 * YACH CHESS
00170 * SYSTEM FLAGS
00180 *
00190 *****
00200
2000 00210          ORG          $2000
00220
00230 * SCREEN COLORS FLAG
00240 *   0 = COLORS OFF
00250 *   1 = COLORS ON
2000 00260 SCFLAG  RMB          1
00270
00280 * GAME STOPPED FLAG
00290 *   0 = STOPPED
00300 *   1 = RUNNING
2001 00310 STFLAG  RMB          1
00320
00330 * BOARD FLIPPED FLAG
00340 *   0 = FLIPPED
00350 *   1 = NOT FLIPPED
2002 00360 BFFLAG  RMB          1
00370
00380 * AS YET UNASSIGNED
2003 00390 FLAG03  RMB          1
2004 00400 FLAG04  RMB          1
2005 00410 FLAG05  RMB          1
2006 00420 FLAG06  RMB          1
2007 00430 FLAG07  RMB          1
2008 00440 FLAG08  RMB          1
2009 00450 FLAG09  RMB          1
200A 00460 FLAG10  RMB          1
200B 00470 FLAG11  RMB          1
200C 00480 FLAG12  RMB          1
200D 00490 FLAG13  RMB          1
200E 00500 FLAG14  RMB          1
200F 00510 FLAG15  RMB          1
2010 00520 FLAG16  RMB          1
2011 00530 FLAG17  RMB          1
2012 00540 FLAG18  RMB          1
2013 00550 FLAG19  RMB          1

```

2014	00560	FLAG20	RMB	1
2015	00570	FLAG21	RMB	1
2016	00580	FLAG22	RMB	1
2017	00590	FLAG23	RMB	1
2018	00600	FLAG24	RMB	1
2019	00610	FLAG25	RMB	1
201A	00620	FLAG26	RMB	1
201B	00630	FLAG27	RMB	1
201C	00640	FLAG28	RMB	1
201D	00650	FLAG29	RMB	1
201E	00660	FLAG30	RMB	1
201F	00670	FLAG31	RMB	1
2020	00680	FLAG32	RMB	1
2021	00690	FLAG33	RMB	1
2022	00700	FLAG34	RMB	1
2023	00710	FLAG35	RMB	1
2024	00720	FLAG36	RMB	1
2025	00730	FLAG37	RMB	1
2026	00740	FLAG38	RMB	1
2027	00750	FLAG39	RMB	1
2028	00760	FLAG40	RMB	1
2029	00770	FLAG41	RMB	1
202A	00780	FLAG42	RMB	1
202B	00790	FLAG43	RMB	1
202C	00800	FLAG44	RMB	1
202D	00810	FLAG45	RMB	1
202E	00820	FLAG46	RMB	1
202F	00830	FLAG47	RMB	1
2030	00840	FLAG48	RMB	1
2031	00850	FLAG49	RMB	1
2032	00860	FLAG50	RMB	1
2033	00870	FLAG51	RMB	1
2034	00880	FLAG52	RMB	1
2035	00890	FLAG53	RMB	1
2036	00900	FLAG54	RMB	1
2037	00910	FLAG55	RMB	1
2038	00920	FLAG56	RMB	1
2039	00930	FLAG57	RMB	1
203A	00940	FLAG58	RMB	1
203B	00950	FLAG59	RMB	1
203C	00960	FLAG60	RMB	1
203D	00970	FLAG61	RMB	1
203E	00980	FLAG62	RMB	1
203F	00990	FLAG63	RMB	1
	01000			
	0000	32767	END	
00000	TOTAL ERRORS			

TMPVAR: Temporary Variables

Temporary Variables are just what they might seem: places to temporarily store stuff.

```
00100 *****
00110 *
00120 * TMPVAR.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * TEMPORARY VARIABLES
00160 * 128 BYTES TOTAL
00170 *
00180 *****
00190
2040 00200          ORG          $2040
00210
00220 * COMMAND VARIABLES
00230 *   SEE GETCMD.ASM
00240
00250 * COMMAND INPUT LETTER
00260 * COUNT
2040 00270 CMDCNT  RMB          1
00280
00290 * COMMAND INPUT
00300 * CHARACTERS
2041 00310 CMDVAR  RMB          5
00320
00330 * COMMAND "FROM"
00340 * FILE CHARACTER
2046 00350 CMDFRF  RMB          1
00360
00370 * COMMAND "FROM"
00380 * RANK CHARACTER
2047 00390 CMDFRR  RMB          1
00400
00410 * COMMAND "TO"
00420 * FILE CHARACTER
2048 00430 CMDTOF  RMB          1
00440
00450 * COMMAND "TO"
00460 * RANK CHARACTER
2049 00470 CMDTOR  RMB          1
00480
00490 * COMMAND "FROM"
00500 * I-VALUE
204A 00510 CMDFRI  RMB          1
```

```

00520
00530 * COMMAND "FROM"
00540 * J-VALUE
204B 00550 CMDFRJ RMB 1
00560
00570 * COMMAND "TO"
00580 * I-VALUE
204C 00590 CMDTOI RMB 1
00600
00610 * COMMAND "TO"
00620 * J-VALUE
204D 00630 CMDTOJ RMB 1
00640
00650 * MATCH VARIABLE
204E 00660 MCHVAR RMB 5
00670
00680 * AS YET UNASSIGNED
2053 00690 TMPUNN RMB 109
0000 32767 END

```

00000 TOTAL ERRORS

=====

MAKEFT: Make the Flags and Temporary Variables FT.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEFT.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "FLAGS.BIN"
1130 LOADM "TMPVAR.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "FT.BIN", &H2000, &H20BF, &H2000
1330 '
32767 END
```

=====

Screen Management

This is Section Two: Screen Management

1. CIDGEN: Generates the Chess Board Initial Display Screen
2. B2SACALC: Chess Board to Screen Translation Array Calculation Utility Program
3. B2SARY: Board to Screen Translation Array
4. B2DARY: Board Dark Squares to Screen Translation Array
5. B2LARY: Board Light Squares to Screen Translation Array
6. BSCARY: Board Surface Contents Array

=====

CIDGEN: Generates the Chess Board Initial Display Screen

```

00100 *****
00110 *
00120 * CIDGEN.ASM
00130 * MDJ 2021/09/03
00140 *
00150 * CHESS SCREEN
00160 * INITIAL DISPLAY
00170 * GENERATOR
00180 *
00190 *****
00200
00210 * SCREEN ADDRESSES
0400 00220 VIDRAM EQU $0400
0600 00230 VIDEND EQU $0600
00240
00250 * EXTERNAL ROUTINE
00260 * ADDRESS
1DDA 00270 PRT2PK EQU $1DDA
00280
20C0 00290 ORG $20C0
00300
20C0 34 32 00310 CIDGEN PSHS A,X,Y
20C2 16 0200 00320 LBRA L22C5
00330
00340 * CHESS SCREEN
00350 * ROW #000
00360 * DISPLAY INITIALIZER
20C5 38 00370 L20C5 FCC /8 /
20 20
20C7 5B 00380 FCC /[ ]/
20 20
20CA 5B 00390 FCC /[*]/
2A 2A
5D 5D
20CD 5B 00400 FCC /[ ]/
20 20
5D 5D
20D0 5B 00410 FCC /[*]/
2A 2A
5D 5D

```

20D3	5B 20 5D	00420	FCC	/[]/
20D6	5B 2A 5D	00430	FCC	/[*]/
20D9	5B 20 5D	00440	FCC	/[]/
20DC	5B 2A 5D	00450	FCC	/[*]/
20DF	20 4D 4F	00460	FCC	/ MO/
20E2	56 45 53	00470	FCC	/VES/
		00480		
		00490	* CHESS SCREEN	
		00500	* ROW #001	
		00510	* DISPLAY INITIALIZER	
20E5	37 20	00520	FCC	/7 /
20E7	5B 2A 5D	00530	FCC	/[*]/
20EA	5B 20 5D	00540	FCC	/[]/
20ED	5B 2A 5D	00550	FCC	/[*]/
20F0	5B 20 5D	00560	FCC	/[]/
20F3	5B 2A 5D	00570	FCC	/[*]/
20F6	5B 20 5D	00580	FCC	/[]/
20F9	5B 2A 5D	00590	FCC	/[*]/
20FC	5B 20	00600	FCC	/[]/

	5D				
20FF	20	00610	FCC	/	/
	20				
	20				
2102	20	00620	FCC	/	/
	20				
	20				
		00630			
		00640	* CHESS SCREEN		
		00650	* ROW #002		
		00660	* DISPLAY INITIALIZER		
2105	36	00670	FCC	/6	/
	20				
2107	5B	00680	FCC	/[]/	
	20				
	5D				
210A	5B	00690	FCC	/[*]/	
	2A				
	5D				
210D	5B	00700	FCC	/[]/	
	20				
	5D				
2110	5B	00710	FCC	/[*]/	
	2A				
	5D				
2113	5B	00720	FCC	/[]/	
	20				
	5D				
2116	5B	00730	FCC	/[*]/	
	2A				
	5D				
2119	5B	00740	FCC	/[]/	
	20				
	5D				
211C	5B	00750	FCC	/[*]/	
	2A				
	5D				
211F	20	00760	FCC	/	/
	20				
	20				
2122	20	00770	FCC	/	/
	20				
	20				
		00780			
		00790	* CHESS SCREEN		
		00800	* ROW #003		
		00810	* DISPLAY INITIALIZER		

2125	35	00820	FCC	/5 /
	20			
2127	5B	00830	FCC	/[*]/
	2A			
	5D			
212A	5B	00840	FCC	/[]/
	20			
	5D			
212D	5B	00850	FCC	/[*]/
	2A			
	5D			
2130	5B	00860	FCC	/[]/
	20			
	5D			
2133	5B	00870	FCC	/[*]/
	2A			
	5D			
2136	5B	00880	FCC	/[]/
	20			
	5D			
2139	5B	00890	FCC	/[*]/
	2A			
	5D			
213C	5B	00900	FCC	/[]/
	20			
	5D			
213F	20	00910	FCC	/ /
	20			
	20			
2142	20	00920	FCC	/ /
	20			
	20			
		00930		
		00940	* CHESS SCREEN	
		00950	* ROW #004	
		00960	* DISPLAY INITIALIZER	
2145	34	00970	FCC	/4 /
	20			
2147	5B	00980	FCC	/[]/
	20			
	5D			
214A	5B	00990	FCC	/[*]/
	2A			
	5D			
214D	5B	01000	FCC	/[]/
	20			
	5D			

2150	5B	01010	FCC	/[*]/
	2A			
	5D			
2153	5B	01020	FCC	/[]/
	20			
	5D			
2156	5B	01030	FCC	/[*]/
	2A			
	5D			
2159	5B	01040	FCC	/[]/
	20			
	5D			
215C	5B	01050	FCC	/[*]/
	2A			
	5D			
215F	20	01060	FCC	/ /
	20			
	20			
2162	20	01070	FCC	/ /
	20			
	20			
		01080	* CHESS SCREEN	
		01090	* ROW #005	
		01100	* DISPLAY INITIALIZER	
2165	33	01110	FCC	/3 /
	20			
2167	5B	01120	FCC	/[*]/
	2A			
	5D			
216A	5B	01130	FCC	/[]/
	20			
	5D			
216D	5B	01140	FCC	/[*]/
	2A			
	5D			
2170	5B	01150	FCC	/[]/
	20			
	5D			
2173	5B	01160	FCC	/[*]/
	2A			
	5D			
2176	5B	01170	FCC	/[]/
	20			
	5D			
2179	5B	01180	FCC	/[*]/
	2A			
	5D			

217C	5B	01190	FCC	/[]/
	20			
	5D			
217F	20	01200	FCC	/ /
	20			
	20			
2182	20	01210	FCC	/ /
	20			
	20			
		01220		
		01230	* CHESS SCREEN	
		01240	* ROW #006	
		01250	* DISPLAY INITIALIZER	
2185	32	01260	FCC	/2 /
	20			
2187	5B	01270	FCC	/[]/
	20			
	5D			
218A	5B	01280	FCC	/[*]/
	2A			
	5D			
218D	5B	01290	FCC	/[]/
	20			
	5D			
2190	5B	01300	FCC	/[*]/
	2A			
	5D			
2193	5B	01310	FCC	/[]/
	20			
	5D			
2196	5B	01320	FCC	/[*]/
	2A			
	5D			
2199	5B	01330	FCC	/[]/
	20			
	5D			
219C	5B	01340	FCC	/[*]/
	2A			
	5D			
219F	20	01350	FCC	/ /
	20			
	20			
21A2	20	01360	FCC	/ /
	20			
	20			
		01370		
		01380	* CHESS SCREEN	

```

01390 * ROW #007
01400 * DISPLAY INITIALIZER
21A5      31      01410      FCC      /1 /
          20
21A7      5B      01420      FCC      /[*]/
          2A
          5D
21AA      5B      01430      FCC      /[ ]/
          20
          5D
21AD      5B      01440      FCC      /[*]/
          2A
          5D
21B0      5B      01450      FCC      /[ ]/
          20
          5D
21B3      5B      01460      FCC      /[*]/
          2A
          5D
21B6      5B      01470      FCC      /[ ]/
          20
          5D
21B9      5B      01480      FCC      /[*]/
          2A
          5D
21BC      5B      01490      FCC      /[ ]/
          20
          5D
21BF      20      01500      FCC      / /
          20
          20
21C2      20      01510      FCC      / /
          20
          20
          01520
          01530 * CHESS SCREEN
          01540 * ROW #008
          01550 * DISPLAY INITIALIZER
21C5      20      01560      FCC      / /
          20
21C7      20      01570      FCC      / A /
          41
          20
21CA      20      01580      FCC      / B /
          42
          20
21CD      20      01590      FCC      / C /

```

	43				
	20				
21D0	20	01600	FCC	/ D /	
	44				
	20				
21D3	20	01610	FCC	/ E /	
	45				
	20				
21D6	20	01620	FCC	/ F /	
	46				
	20				
21D9	20	01630	FCC	/ G /	
	47				
	20				
21DC	20	01640	FCC	/ H /	
	48				
	20				
21DF	20	01650	FCC	/ /	
	20				
	20				
21E2	20	01660	FCC	/ /	
	20				
	20				
		01670			
		01680	* CHESS SCREEN		
		01690	* ROW #009		
		01700	* DISPLAY INITIALIZER		
21E5	20	01710	FCC	/ /	
	20				
21E7	20	01720	FCC	/ /	
	20				
	20				
21EA	20	01730	FCC	/ /	
	20				
	20				
21ED	20	01740	FCC	/ /	
	20				
	20				
21F0	20	01750	FCC	/ /	
	20				
	20				
21F3	20	01760	FCC	/ /	
	20				
	20				
21F6	20	01770	FCC	/ /	
	20				
	20				

21F9	20	01780	FCC	/	/
	20				
	20				
21FC	20	01790	FCC	/	/
	20				
	20				
21FF	20	01800	FCC	/	/
	20				
	20				
2202	20	01810	FCC	/	/
	20				
	20				
		01820			
		01830	* CHESS SCREEN		
		01840	* ROW #010		
		01850	* DISPLAY INITIALIZER		
2205	20	01860	FCC	/	/
	20				
2207	20	01870	FCC	/	/
	20				
	20				
220A	20	01880	FCC	/	/
	20				
	20				
220D	20	01890	FCC	/	/
	20				
	20				
2210	20	01900	FCC	/	/
	20				
	20				
2213	20	01910	FCC	/	/
	20				
	20				
2216	20	01920	FCC	/	/
	20				
	20				
2219	20	01930	FCC	/	/
	20				
	20				
221C	20	01940	FCC	/	/
	20				
	20				
221F	20	01950	FCC	/	/
	20				
	20				
2222	20	01960	FCC	/	/
	20				

	20		01970		
			01980	*	CHESS SCREEN
			01990	*	ROW #011
			02000	*	DISPLAY INITIALIZER
2225	20		02010		FCC / /
	20				
2227	20		02020		FCC / /
	20				
	20				
222A	57		02030		FCC /WHI/
	48				
	49				
222D	54		02040		FCC /TE'/
	45				
	27				
2230	53		02050		FCC /S M/
	20				
	4D				
2233	4F		02060		FCC /OVE/
	56				
	45				
2236	20		02070		FCC / # /
	23				
	20				
2239	20		02080		FCC / /
	20				
	20				
223C	20		02090		FCC / /
	20				
	20				
223F	20		02100		FCC / /
	20				
	20				
2242	20		02110		FCC / /
	20				
	20				
			02120		
			02130	*	CHESS SCREEN
			02140	*	ROW #012
			02150	*	DISPLAY INITIALIZER
2245	20		02160		FCC / /
	20				
2247	20		02170		FCC / /
	20				
	20				
224A	20		02180		FCC / /

	20				
	20				
224D	20	02190	FCC	/	/
	20				
	20				
2250	20	02200	FCC	/	/
	20				
	20				
2253	20	02210	FCC	/	/
	20				
	20				
2256	20	02220	FCC	/	/
	20				
	20				
2259	20	02230	FCC	/	/
	20				
	20				
225C	20	02240	FCC	/	/
	20				
	20				
225F	20	02250	FCC	/	/
	20				
	20				
2262	20	02260	FCC	/	/
	20				
	20				
		02270			
		02280	* CHESS SCREEN		
		02290	* ROW #013		
		02300	* DISPLAY INITIALIZER		
2265	20	02310	FCC	/	/
	20				
2267	20	02320	FCC	/	/
	20				
	20				
226A	43	02330	FCC	/COM/	
	4F				
	4D				
226D	4D	02340	FCC	/MAN/	
	41				
	4E				
2270	44	02350	FCC	/D: /	
	3A				
	20				
2273	20	02360	FCC	/	/
	20				
	20				

2276	20	02370	FCC	/	/
	20				
	20				
2279	20	02380	FCC	/	/
	20				
	20				
227C	20	02390	FCC	/	/
	20				
	20				
227F	20	02400	FCC	/	/
	20				
	20				
2282	20	02410	FCC	/	/
	20				
	20				
		02420			
		02430	* CHESS SCREEN		
		02440	* ROW #014		
		02450	* DISPLAY INITIALIZER		
2285	20	02460	FCC	/	/
	20				
2287	20	02470	FCC	/	/
	20				
	20				
228A	20	02480	FCC	/	/
	20				
	20				
228D	20	02490	FCC	/	/
	20				
	20				
2290	20	02500	FCC	/	/
	20				
	20				
2293	20	02510	FCC	/	/
	20				
	20				
2296	20	02520	FCC	/	/
	20				
	20				
2299	20	02530	FCC	/	/
	20				
	20				
229C	20	02540	FCC	/	/
	20				
	20				
229F	20	02550	FCC	/	/
	20				

	20				
22A2	20	02560	FCC	/	/
	20				
	20				
		02570			
		02580	* CHESS SCREEN		
		02590	* ROW #015		
		02600	* DISPLAY INITIALIZER		
22A5	20	02610	FCC	/	/
	20				
22A7	20	02620	FCC	/	/
	20				
	20				
22AA	4E	02630	FCC	/NOT/	
	4F				
	54				
22AD	45	02640	FCC	/E: /	
	3A				
	20				
22B0	20	02650	FCC	/	/
	20				
	20				
22B3	20	02660	FCC	/	/
	20				
	20				
22B6	20	02670	FCC	/	/
	20				
	20				
22B9	20	02680	FCC	/	/
	20				
	20				
22BC	20	02690	FCC	/	/
	20				
	20				
22BF	20	02700	FCC	/	/
	20				
	20				
22C2	20	02710	FCC	/	/
	20				
	20				
		02720			
		02730	* INITIALIZE THE SCREEN		
		02740			
		02750	* POINT TO BEGINNING OF		
		02760	* SCREEN DATA		
22C5	108E	20C5	02770	L22C5	LDY #L20C5
			02780		

```

02790 * POINT TO BEGINNING OF
02800 * VIDRAM SCREEN
22C9 8E 0400 02810          LDX      #VIDRAM
02820
02830 * GET CHARACTER CODE FROM
02840 * THE FCC DATA LIST
22CC A6 A0 02850 L22CC  LDA      ,Y+
02860
02870 * GO CONVERT IT FROM A
02880 * PRINT CODE TO A
02890 * POKE CODE
22CE 17 FB09 02900          LBSR      PRT2PK
02910
02920 * PUT IT TO VIDRAM
22D1 A7 80 02930          STA      ,X+
02940
02950 * ARE WE DONE?
22D3 8C 0600 02960          CMPX      #VIDEND
02970
02980 * GO IF NO
22D6 25 F4 02990          BLO      L22CC
03000
03010 * EXIT
22D8 35 32 03020          PULS      A,X,Y
22DA 39 0000 03030          RTS
32767          END

```

00000 TOTAL ERRORS

The Assembly Language Test Routine:

```

00100 *****
00110 *
00120 * CTST0002.ASM
00130 * MDJ 2021/09/18
00140 *
00150 * CIDGEN TEST
00160 *
00170 *****
00180
00190 * EXTERNAL ROUTINE
00200 * ADDRESS
1E8A 00210 CIDGEN EQU      $1E8A
00220
7000 00230          ORG      $7000

```

```

00240
00250 * GO DO THE TEST
7000 17 AE87 00260          LBSR    CIDGEN
00270
00280 * HOLD THE SCREEN
7003 20 FE 00290 L0001  BRA    L0001
00300
00310 * EXIT
7005 39 00320          RTS
0000 32767          END

```

00000 TOTAL ERRORS

The BASIC Language Control Program:

```

1000 '*****
1010 '*
1020 '* CTST0002.BAS
1030 '* MDJ 2021/09/18
1040 '*
1050 '* CIDGEN TEST
1060 '*
1070 '*****
1080 '
1090 'SETUP MEMORY
1100 PCLEAR 1
1110 CLEAR 200, &H1C00
1120 '
1130 CLS
1140 PRINT @ 105, "Y. A. C. H."
1150 PRINT @ 162, "(YET ANOTHER CHESS HANDLER)"
1155 PRINT @ 201, "VERSION 0.0.1"
1160 PRINT @ 304, "BY"
1170 PRINT @ 328, "M. DAVID JOHNSON"
1180 PRINT @ 360, "INFO@BDS-SOFT.COM"
1190 PRINT @ 395, "2021/09/29"
1520 '
1530 'LOAD ML ROUTINES
1540 LOADM "REGXFR.BIN"
1550 LOADM "VIDCLS.BIN"
1560 LOADM "PUTCHR.BIN"
1570 LOADM "GETCHR.BIN"
1580 LOADM "PUTBYT.BIN"
1590 LOADM "SCROLL.BIN"
1600 LOADM "PUTCHA.BIN"
1610 LOADM "PUTBYA.BIN"
1620 LOADM "CRLF.BIN"

```

```
1630 LOADM "PK2PRT.BIN"
1640 LOADM "PRT2PK.BIN"
1650 LOADM "POLCAT.BIN"
1660 LOADM "CIDGEN.BIN"
1670 LOADM "CTST0002.BIN"
1680 '
1690 PRINT @ 451, "PRESS ANY KEY  TO CONTINUE"
1710 A$ = INKEY$
1720 IF A$ = "" GOTO 1710
1880 '
1890 'GO DO THE TEST
1900 EXEC &H7000
1910 '
32767 END
```

Results:

Immediately after "RUN", the Splash Screen is displayed:



The Splash Screen continues to display while the machine language routines are being loaded. Upon completion of that loading, the Splash Screen displays an additional line:



And, upon pressing a key, the Chess Board Initial Display Screen is displayed:



As expected.

(Note that the Move Number and Moves List are blanked-out in this early version).

=====

B2SACALC: Chess Board to Screen Translation Array Calculation Utility Program

```
1000 '*****
1010 '*
1020 '* B2SACALC.BAS
1030 '* MDJ 2021/09/04
1040 '*
1050 '* UTILITY PROGRAM TO
1060 '* CALCULATE THE 64
1070 '* BOARD TO SCREEN
1080 '* TRANSLATION VALUES FOR
1090 '* THE CHESS BOARD'S
1100 '* SCREEN TRANSLATION
1110 '* ARRAY
1120 '*
1130 '*****
1140 '
1150 'V = VIDRAM SCREEN ADDRESS
1160 V = &H0400
1170 '
1180 'R = SCREEN ROW START NUM
1190 'J = BOARD ROW NUMBER
1200 J = 8
1210 FOR R = 0 TO 224 STEP 32
1220 J = J - 1
1230 '
1240 'C = SCREEN COLUMN NUMBER
1250 'I = BOARD COLUMN NUMBER
1260 I = -1
1270 FOR C = 3 TO 24 STEP 3
1280 I = I + 1
1290 '
1300 'A = SCREEN ADDRESS
1310 A = V + R + C
1320 '
1330 PRINT I;" ";J;" = &H";
1340 PRINT HEX$(A)
1350 PRINT#-2, I;" ";J;" = &H";
1360 PRINT#-2, HEX$(A)
1370 NEXT C
1380 NEXT R
```

32767 END

Results:

0 , 7 = &H403
1 , 7 = &H406
2 , 7 = &H409
3 , 7 = &H40C
4 , 7 = &H40F
5 , 7 = &H412
6 , 7 = &H415
7 , 7 = &H418
0 , 6 = &H423
1 , 6 = &H426
2 , 6 = &H429
3 , 6 = &H42C
4 , 6 = &H42F
5 , 6 = &H432
6 , 6 = &H435
7 , 6 = &H438
0 , 5 = &H443
1 , 5 = &H446
2 , 5 = &H449
3 , 5 = &H44C
4 , 5 = &H44F
5 , 5 = &H452
6 , 5 = &H455
7 , 5 = &H458
0 , 4 = &H463
1 , 4 = &H466
2 , 4 = &H469
3 , 4 = &H46C
4 , 4 = &H46F
5 , 4 = &H472
6 , 4 = &H475
7 , 4 = &H478
0 , 3 = &H483
1 , 3 = &H486
2 , 3 = &H489
3 , 3 = &H48C
4 , 3 = &H48F
5 , 3 = &H492
6 , 3 = &H495
7 , 3 = &H498
0 , 2 = &H4A3

1 , 2 = &H4A6
2 , 2 = &H4A9
3 , 2 = &H4AC
4 , 2 = &H4AF
5 , 2 = &H4B2
6 , 2 = &H4B5
7 , 2 = &H4B8
0 , 1 = &H4C3
1 , 1 = &H4C6
2 , 1 = &H4C9
3 , 1 = &H4CC
4 , 1 = &H4CF
5 , 1 = &H4D2
6 , 1 = &H4D5
7 , 1 = &H4D8
0 , 0 = &H4E3
1 , 0 = &H4E6
2 , 0 = &H4E9
3 , 0 = &H4EC
4 , 0 = &H4EF
5 , 0 = &H4F2
6 , 0 = &H4F5
7 , 0 = &H4F8

As expected.

=====

B2SARY: Board to Screen Translation Array

Note that this is an 8 x 8, 64-cell array, with I = x-coordinates running horizontally from left-to-right, and J = y-coordinates running vertically from bottom-to-top.

```

00100 *****
00110 *
00120 * B2SARY.ASM
00130 * MDJ 2021/09/04
00140 *
00150 * BOARD TO SCREEN
00160 * TRANSLATION ARRAY
00170 *
00180 * 16-BIT ADDRESSES
00190 * STORED IN
00200 * BOARD ROW ORDER
00210 *
00220 * DIRECTLY ACCESS ENTRIES
00230 * AT ORG ADDRESS +
00240 * ((J * 8) + I) * 2
00250 * WHERE 0 <= I <= 7
00260 * AND 0 <= J <= 7
00270 *
00280 * ALTERNATIVELY, THE ARRAY
00290 * CAN BE TRAVERSED USING
00300 * SOMETHING LIKE A ,U++
00310 * MECHANISM
00320 *
00330 *****
00340
24C0 00350          ORG          $24C0
00360
24C0 04E3 00370 B2SARY  FDB      $04E3   0,0
24C2 04E6 00380          FDB      $04E6   1,0
24C4 04E9 00390          FDB      $04E9   2,0
24C6 04EC 00400          FDB      $04EC   3,0
24C8 04EF 00410          FDB      $04EF   4,0
24CA 04F2 00420          FDB      $04F2   5,0
24CC 04F5 00430          FDB      $04F5   6,0
24CE 04F8 00440          FDB      $04F8   7,0
24D0 04C3 00450          FDB      $04C3   0,1
24D2 04C6 00460          FDB      $04C6   1,1
24D4 04C9 00470          FDB      $04C9   2,1
24D6 04CC 00480          FDB      $04CC   3,1

```

24D8	04CF	00490	FDB	\$04CF	4,1
24DA	04D2	00500	FDB	\$04D2	5,1
24DC	04D5	00510	FDB	\$04D5	6,1
24DE	04D8	00520	FDB	\$04D8	7,1
24E0	04A3	00530	FDB	\$04A3	0,2
24E2	04A6	00540	FDB	\$04A6	1,2
24E4	04A9	00550	FDB	\$04A9	2,2
24E6	04AC	00560	FDB	\$04AC	3,2
24E8	04AF	00570	FDB	\$04AF	4,2
24EA	04B2	00580	FDB	\$04B2	5,2
24EC	04B5	00590	FDB	\$04B5	6,2
24EE	04B8	00600	FDB	\$04B8	7,2
24F0	0483	00610	FDB	\$0483	0,3
24F2	0486	00620	FDB	\$0486	1,3
24F4	0489	00630	FDB	\$0489	2,3
24F6	048C	00640	FDB	\$048C	3,3
24F8	048F	00650	FDB	\$048F	4,3
24FA	0492	00660	FDB	\$0492	5,3
24FC	0495	00670	FDB	\$0495	6,3
24FE	0498	00680	FDB	\$0498	7,3
2500	0463	00690	FDB	\$0463	0,4
2502	0466	00700	FDB	\$0466	1,4
2504	0469	00710	FDB	\$0469	2,4
2506	046C	00720	FDB	\$046C	3,4
2508	046F	00730	FDB	\$046F	4,4
250A	0472	00740	FDB	\$0472	5,4
250C	0475	00750	FDB	\$0475	6,4
250E	0478	00760	FDB	\$0478	7,4
2510	0443	00770	FDB	\$0443	0,5
2512	0446	00780	FDB	\$0446	1,5
2514	0449	00790	FDB	\$0449	2,5
2516	044C	00800	FDB	\$044C	3,5
2518	044F	00810	FDB	\$044F	4,5
251A	0452	00820	FDB	\$0452	5,5
251C	0455	00830	FDB	\$0455	6,5
251E	0458	00840	FDB	\$0458	7,5
2520	0423	00850	FDB	\$0423	0,6
2522	0426	00860	FDB	\$0426	1,6
2524	0429	00870	FDB	\$0429	2,6
2526	042C	00880	FDB	\$042C	3,6
2528	042F	00890	FDB	\$042F	4,6
252A	0432	00900	FDB	\$0432	5,6
252C	0435	00910	FDB	\$0435	6,6
252E	0438	00920	FDB	\$0438	7,6
2530	0403	00930	FDB	\$0403	0,7
2532	0406	00940	FDB	\$0406	1,7
2534	0409	00950	FDB	\$0409	2,7

2536	040C	00960	FDB	\$040C	3,7
2538	040F	00970	FDB	\$040F	4,7
253A	0412	00980	FDB	\$0412	5,7
253C	0415	00990	FDB	\$0415	6,7
253E	0418	01000	FDB	\$0418	7,7
		01010			
	0000	32767	END		

00000 TOTAL ERRORS

=====

B2DARY: Board Dark Squares to Screen Translation Array

```

00100 *****
00110 *
00120 * B2DARY.ASM
00130 * MDJ 2021/09/04
00140 *
00150 * BOARD DARK SQUARES
00160 * TO SCREEN
00170 * TRANSLATION ARRAY
00180 *
00190 * 16-BIT ADDRESSES
00200 * STORED IN
00210 * BOARD ROW ORDER
00220 *
00230 * NOT SUITABLE FOR
00240 * DIRECT ACCESS TO
00250 * ENTRIES
00260 *
00270 * TRAVERSED THE ARRAY
00280 * USING SOMETHING LIKE A
00290 * ,U++ MECHANISM
00300 *
00310 *****
00320
2540          00330          ORG          $2540
00340
2540          04E3          00350 B2DARY  FDB          $04E3          0,0
2542          04E9          00360          FDB          $04E9          2,0
2544          04EF          00370          FDB          $04EF          4,0
2546          04F5          00380          FDB          $04F5          6,0
2548          04C6          00390          FDB          $04C6          1,1
254A          04CC          00400          FDB          $04CC          3,1
254C          04D2          00410          FDB          $04D2          5,1
254E          04D8          00420          FDB          $04D8          7,1
2550          04A3          00430          FDB          $04A3          0,2
2552          04A9          00440          FDB          $04A9          2,2
2554          04AF          00450          FDB          $04AF          4,2
2556          04B5          00460          FDB          $04B5          6,2
2558          0486          00470          FDB          $0486          1,3
255A          048C          00480          FDB          $048C          3,3
255C          0492          00490          FDB          $0492          5,3
255E          0498          00500          FDB          $0498          7,3

```

2560	0463	00510	FDB	\$0463	0,4
2562	0469	00520	FDB	\$0469	2,4
2564	046F	00530	FDB	\$046F	4,4
2566	0475	00540	FDB	\$0475	6,4
2568	0446	00550	FDB	\$0446	1,5
256A	044C	00560	FDB	\$044C	3,5
256C	0452	00570	FDB	\$0452	5,5
256E	0458	00580	FDB	\$0458	7,5
2570	0423	00590	FDB	\$0423	0,6
2572	0429	00600	FDB	\$0429	2,6
2574	042F	00610	FDB	\$042F	4,6
2576	0435	00620	FDB	\$0435	6,6
2578	0406	00630	FDB	\$0406	1,7
257A	040C	00640	FDB	\$040C	3,7
257C	0412	00650	FDB	\$0412	5,7
257E	0418	00660	FDB	\$0418	7,7
		00670			
	0000	32767	END		

00000 TOTAL ERRORS

=====

B2LARY: Board Light Squares to Screen Translation Array

```

00100 *****
00110 *
00120 * B2LARY.ASM
00130 * MDJ 2021/09/04
00140 *
00150 * BOARD LIGHT SQUARES
00160 * TO SCREEN
00170 * TRANSLATION ARRAY
00180 *
00190 * 16-BIT ADDRESSES
00200 * STORED IN
00210 * BOARD ROW ORDER
00220 *
00230 * NOT SUITABLE FOR
00240 * DIRECT ACCESS TO
00250 * ENTRIES
00260 *
00270 * TRAVERSE THE ARRAY
00280 * USING SOMETHING LIKE A
00290 * ,U++ MECHANISM
00300 *
00310 *****
00320
2580          00330          ORG          $2580
00340
2580          04E6          00350 B2LARY  FDB          $04E6          1,0
2582          04EC          00360          FDB          $04EC          3,0
2584          04F2          00370          FDB          $04F2          5,0
2586          04F8          00380          FDB          $04F8          7,0
2588          04C3          00390          FDB          $04C3          0,1
258A          04C9          00400          FDB          $04C9          2,1
258C          04CF          00410          FDB          $04CF          4,1
258E          04D5          00420          FDB          $04D5          6,1
2590          04A6          00430          FDB          $04A6          1,2
2592          04AC          00440          FDB          $04AC          3,2
2594          04B2          00450          FDB          $04B2          5,2
2596          04B8          00460          FDB          $04B8          7,2
2598          0483          00470          FDB          $0483          0,3
259A          0489          00480          FDB          $0489          2,3
259C          048F          00490          FDB          $048F          4,3
259E          0495          00500          FDB          $0495          6,3

```

25A0	0466	00510	FDB	\$0466	1,4
25A2	046C	00520	FDB	\$046C	3,4
25A4	0472	00530	FDB	\$0472	5,4
25A6	0478	00540	FDB	\$0478	7,4
25A8	0443	00550	FDB	\$0443	0,5
25AA	0449	00560	FDB	\$0449	2,5
25AC	044F	00570	FDB	\$044F	4,5
25AE	0455	00580	FDB	\$0455	6,5
25B0	0426	00590	FDB	\$0426	1,6
25B2	042C	00600	FDB	\$042C	3,6
25B4	0432	00610	FDB	\$0432	5,6
25B6	0438	00620	FDB	\$0438	7,6
25B8	0403	00630	FDB	\$0403	0,7
25BA	0409	00640	FDB	\$0409	2,7
25BC	040F	00650	FDB	\$040F	4,7
25BE	0415	00660	FDB	\$0415	6,7
		00670			
	0000	32767	END		

00000 TOTAL ERRORS

=====

BSCARY: Board Surface Contents Array

Note that this is an 8 x 8, 64-cell array, with I = x-coordinates running horizontally from left-to-right, and J = y-coordinates running vertically from bottom-to-top.

```
00100 *****
00110 *
00120 * BSCARY.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * BOARD SURFACE CONTENTS
00160 * ARRAY
00170 *
00180 * I.E. VISUAL
00190 * APPEARANCE
00200 *
00210 * I.E. ACTUAL CHARACTERS
00220 * BY CODE NUMBER
00230 *
00240 * 8-BIT CODES
00250 * STORED IN
00260 * BOARD ROW ORDER
00270 *
00280 * DIRECTLY ACCESS ENTRIES
00290 * AT ORG ADDRESS +
00300 * (J * 8) + I
00310 * WHERE 0 <= I <= 7
00320 *   AND 0 <= J <= 7
00330 *
00340 * ALTERNATIVELY, THE ARRAY
00350 * CAN BE TRAVERSED USING
00360 * SOMETHING LIKE A ,U+
00370 * MECHANISM
00380 *
00390 *****
00400 *
00410 * PIECE CODES
00420 *
00430 *   NONE = SPACE = 096
00440 *
00450 *   WHITE KING   = 075
00460 *   WHITE QUEEN  = 081
00470 *   WHITE ROOK   = 082
```

```

00480 *    WHITE BISHOP = 066
00490 *    WHITE KNIGHT = 078
00500 *    WHITE PAWN   = 080
00510 *
00520 *    BLACK KING   = 011
00530 *    BLACK QUEEN  = 017
00540 *    BLACK ROOK   = 018
00550 *    BLACK BISHOP = 002
00560 *    BLACK KNIGHT = 014
00570 *    BLACK PAWN   = 016
00580 *
00590 *****
00600
25C0      00610          ORG      $25C0
          00620
25C0      00630 BSCARY  RMB      64
          00640
          0000      32767          END

00000 TOTAL ERRORS

```

=====

MAKESM: Make the Screen Management SM.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKESM.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "CIDGEN.BIN"
1130 LOADM "B2SARY.BIN"
1140 LOADM "B2DARY.BIN"
1150 LOADM "B2LARY.BIN"
1160 LOADM "BSCARY.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "SM.BIN", &H20C0, &H25FF, &H20C0
1330 '
32767 END
```

=====

General Reporting

This is Section Three: General Reporting

1. NOTREC: Command Not Recognized
2. NYIMSG: Not Yet Implemented Postamble

=====

NOTREC: Command Not Recognized

```
00100 *****
00110 *
00120 * NOTREC.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND NOT
00160 * RECOGNIZED
00170 *
00180 *****
00190
00200 * SCREEN REPORTING FIELD
05EB 00210 RPTFLD EQU $05EB
00220
2600 00230 ORG $2600
00240
2600 34 36 00250 NOTREC PSHS A,B,X,Y
00260
00270 * POINT TO THE
00280 * REPORTING FIELD
2602 8E 05EB 00290 LDX #RPTFLD
00300
00310 * DISPLAY THE MESSAGE
2605 86 43 00320 LDA #67 C
2607 A7 80 00330 STA ,X+
2609 86 4D 00340 LDA #77 M
260B A7 80 00350 STA ,X+
260D 86 44 00360 LDA #68 D
260F A7 80 00370 STA ,X+
2611 86 60 00380 LDA #96 SP
2613 A7 80 00390 STA ,X+
2615 86 4E 00400 LDA #78 N
2617 A7 80 00410 STA ,X+
2619 86 4F 00420 LDA #79 O
261B A7 80 00430 STA ,X+
261D 86 54 00440 LDA #84 T
261F A7 80 00450 STA ,X+
2621 86 60 00460 LDA #96 SP
2623 A7 80 00470 STA ,X+
2625 86 52 00480 LDA #82 R
2627 A7 80 00490 STA ,X+
2629 86 45 00500 LDA #69 E
262B A7 80 00510 STA ,X+
262D 86 43 00520 LDA #67 C
```

262F	A7	80	00530	STA	,X+	
2631	86	4F	00540	LDA	#79	O
2633	A7	80	00550	STA	,X+	
2635	86	47	00560	LDA	#71	G
2637	A7	80	00570	STA	,X+	
2639	86	4E	00580	LDA	#78	N
263B	A7	80	00590	STA	,X+	
263D	86	49	00600	LDA	#73	I
263F	A7	80	00610	STA	,X+	
2641	86	5A	00620	LDA	#90	Z
2643	A7	80	00630	STA	,X+	
2645	86	45	00640	LDA	#69	E
2647	A7	80	00650	STA	,X+	
2649	86	44	00660	LDA	#68	D
264B	A7	80	00670	STA	,X+	
			00680			
			00690	* EXIT		
264D	35	36	00700	PULS	A,B,X,Y	
264F	39		00710	RTS		
		0000	32767	END		

00000 TOTAL ERRORS

=====

NYIMSG: Not Yet Implemented Postamble

```

00100 *****
00110 *
00120 * NYIMSG.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * "NOT YET
00160 * IMPLEMENTED"
00170 * MESSAGE
00180 *
00190 *****
00200
2680      00210      ORG      $2680
00220
2680 86    60      00230 NYIMSG LDA      #96      SP
2682 A7    80      00240      STA      ,X+
2684 86    6D      00250      LDA      #109     -
2686 A7    80      00260      STA      ,X+
2688 86    60      00270      LDA      #96      SP
268A A7    80      00280      STA      ,X+
268C 86    4E      00290      LDA      #78      N
268E A7    80      00300      STA      ,X+
2690 86    6E      00310      LDA      #110     .
2692 A7    80      00320      STA      ,X+
2694 86    59      00330      LDA      #89      Y
2696 A7    80      00340      STA      ,X+
2698 86    6E      00350      LDA      #110     .
269A A7    80      00360      STA      ,X+
269C 86    49      00370      LDA      #73      I
269E A7    80      00380      STA      ,X+
26A0 86    6E      00390      LDA      #110     .
26A2 A7    80      00400      STA      ,X+
00410
00420 * EXIT
26A4 35    36      00430      PULS     A,B,X,Y
26A6 39      00440      RTS
0000      32767     END

```

00000 TOTAL ERRORS

=====

MAKEGR: Make the General Reporting GR.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEGR.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "NOTREC.BIN"
1130 LOADM "NYIMSG.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "GR.BIN", &H2600, &H26BF, &H2600
1330 '
32767 END
```

=====

General Commands

This is Section Four: General Commands

1. ZERO2: Castle Kingside Actual Command = 0-0 Transfer Command
2. ZERO3: Castle Queenside Actual Command = 0-0-0 Transfer Command
3. ANLYZ: Analyze the Board (N.Y.I. - Not Yet Implemented)
4. ASKFR: Ask for Information About the Square at F = File Character R = Rank Character (N.Y.I. - Not Yet Implemented)
5. CLRBD: Clears the Board
6. CKMOV: Check the Legailty Of a Proposed Move (N.Y.I. - Not Yet Implemented)
7. CKPOS: Check the Legality Of the Displayed Position (N.Y.I. - Not Yet Implemented)
8. CSTLK: Castle King-Side Action Command (N.Y.I. - Not Yet Implemented)
9. CSTLQ: Castle Queen-Side Action Command (N.Y.I. - Not Yet Implemented)
10. DSPBD: Copies the Board Surface Contents Array to the Display Screen
11. FLIP: Flip the Board (N.Y.I. - Not Yet Implemented)
12. GSTRT: Restart Accepting Commands After a GSTOP Action Command
13. GSTOP: Ignore All Subsequent Commands Except GSTRT Action Command
14. LDGAM: Load a Game (N.Y.I. - Not Yet Implemented)
15. LDPOS: Load a Position (N.Y.I. - Not Yet Implemented)
16. NEW: Sets a New Game Position Transfer Command
17. NEWPS: Sets a New Game Position Action Command
18. LTRO2: Castle Kingside Actual Command = O-O Transfer Command
19. LTRO3: Castle Queenside Actual Command = O-O-O Transfer Command

20. QUIT: Quit the Game and Return to the Command Prompt
21. SCOFF: Turns Screen Colors Off
22. SCON: Turns Screen Colors On
23. START: Restart Accepting Commands After a STOP Transfer Command
24. STOP: Ignore All Subsequent Commands Except START Transfer Command
25. SVGAM: Save the Game (N.Y.I. - Not Yet Implemented)
26. SVPOS: Save the Position (N.Y.I. - Not Yet Implemented)
27. TKBAK: Take Back the Last Move (N.Y.I. - Not Yet Implemented)

=====

ZERO2: Castle Kingside

Actual Command = 0-0

Transfer Command

```
00100 *****
00110 *
00120 * ZERO2.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * CASTLE KINGSIDE
00170 * ACTUAL COMMAND = 0-0
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
          2BE0 00220 CSTLK   EQU   $2BE0
26C0          00230
          00240          ORG   $26C0
          00250
26C0 16      051D 00260 * BRANCH TO THE
          00270 * ACTION COMMAND
          00280          LBRA   CSTLK
          00290
          0000 32767          END
```

00000 TOTAL ERRORS

=====

ZERO3: Castle Queenside

Actual Command = 0-0-0

Transfer Command

```
00100 *****
00110 *
00120 * ZERO3.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * CASTLE QUEENSIDE
00170 * ACTUAL COMMAND = 0-0-0
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
2CE0 00220 CSTLQ EQU $2CE0
00230
26D0 00240 ORG $26D0
00250
00260 * BRANCH TO THE
00270 * ACTION COMMAND
26D0 16 060D 00280 LBRA CSTLQ
00290
0000 32767 END
```

00000 TOTAL ERRORS

=====

ANLYZ: Analyze the Board (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * ANLYZ.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * ANALYZE THE
00170 * BOARD
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00
1C0E 00450 VIDCLS EQU $1C0E
1C1F 00460 PUTCHR EQU $1C1F
1CD2 00470 GETCHR EQU $1CD2
1CD5 00480 PUTBYT EQU $1CD5
1D17 00490 SCROLL EQU $1D17
1D36 00500 PUTCHA EQU $1D36

```

1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			
	00920	* GENERAL REPORTING		
2600	00930	NOTREC	EQU	\$2600
2680	00940	NYIMSG	EQU	\$2680
	00950			
	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2	EQU	\$26C0

26D0	00980	ZERO3	EQU	\$26D0
	00990	*ANLYZ	EQU	\$26E0
27E0	01000	ASKFR	EQU	\$27E0
28E0	01010	CLRBD	EQU	\$28E0
29E0	01020	CKMOV	EQU	\$29E0
2AE0	01030	CKPOS	EQU	\$2AE0
2BE0	01040	CSTLK	EQU	\$2BE0
2CE0	01050	CSTLQ	EQU	\$2CE0
2DE0	01060	DSPBD	EQU	\$2DE0
2EE0	01070	FLIP	EQU	\$2EE0
2FE0	01080	GSTRT	EQU	\$2FE0
30E0	01090	GSTOP	EQU	\$30E0
31E0	01100	LDGAM	EQU	\$31E0
32E0	01110	LDPOS	EQU	\$32E0
33E0	01120	NEW	EQU	\$33E0
33F0	01130	NEWPS	EQU	\$33F0
34F0	01140	LTRO2	EQU	\$34F0
3500	01150	LTRO3	EQU	\$3500
3510	01160	QUIT	EQU	\$3510
3520	01170	SCOFF	EQU	\$3520
3620	01180	SCON	EQU	\$3620
3720	01190	START	EQU	\$3720
3730	01200	STOP	EQU	\$3730
3740	01210	SVGAM	EQU	\$3740
3840	01220	SVPOS	EQU	\$3840
3940	01230	TKBAK	EQU	\$3940
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT	EQU	\$3A40
3A60	01280	RNKCVT	EQU	\$3A60
3A80	01290	FRE0	EQU	\$3A80
3A90	01300	FRECP	EQU	\$3A90
3AA0	01310	FRFR	EQU	\$3AA0
3EA0	01320	FRFRE	EQU	\$3EA0
3FA0	01330	FRFRP	EQU	\$3FA0
40A0	01340	FRQ0	EQU	\$40A0
41A0	01350	FRQCP	EQU	\$41A0
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD	EQU	\$42A0
43A0	01390	MATCH3	EQU	\$43A0
4420	01400	MATCH4	EQU	\$4420
44A0	01410	XLTCMD	EQU	\$44A0
48A0	01420	XLTCM2	EQU	\$48A0
	01430			
	01440	* YACH SUPERVISORY		

		4CA0	01450	GMLOOP	EQU	\$4CA0	
		4CE0	01460	YACH	EQU	\$4CE0	
			01470				
26E0			01480		ORG	\$26E0	
			01490				
26E0	34	36	01500	ANLYZ	PSHS	A,B,X,Y	
			01510				
			01520	* POINT TO THE			
			01530	* REPORTING FIELD			
26E2	8E	05EB	01540		LDX	#RPTFLD	
			01550				
			01560	* COMMAND NYI			
			01570	* MESSAGE			
26E5	86	41	01580		LDA	#65	A
26E7	A7	80	01590		STA	,X+	
26E9	86	4E	01600		LDA	#78	N
26EB	A7	80	01610		STA	,X+	
26ED	86	4C	01620		LDA	#76	L
26EF	A7	80	01630		STA	,X+	
26F1	86	59	01640		LDA	#89	Y
26F3	A7	80	01650		STA	,X+	
26F5	86	5A	01660		LDA	#90	Z
26F7	A7	80	01670		STA	,X+	
			01680				
			01690	* BRANCH TO NYI			
			01700	* MESSAGE			
26F9	16	FF84	01710		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

**ASKFR: Ask for Information
About the Square at
F = File Character
R = Rank Character
(N.Y.I. - Not Yet Implemented)**

```
00100 *****
00110 *
00120 * ASKFR.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * ASK FOR INFO
00170 * ABOUT THE SQUARE AT
00180 * F = FILE CHARACTER
00190 * R = RANK CHARACTER
00200 *
00210 * NOT YET
00220 * IMPLEMENTED
00230 *
00240 *****
00250
00260 * LOW RAM CURSOR ADDRESS
0088 00270 CURPOS EQU $0088
00280
00290 * SCREEN ADDRESSES
00300 * START OF VIDRAM
0400 00310 VIDRAM EQU $0400
00320
00330 * COMMAND PROMPT
00340 * SCREEN LOCATION
05AE 00350 CMDPMT EQU $05AE
00360
00370 * SCREEN REPORTING FIELD
05EB 00380 RPTFLD EQU $05EB
00390
00400 * ONE BYTE PAST THE
00410 * END OF VIDRAM
0600 00420 VIDEND EQU $0600
00430
00440 * ML FOUNDATION
```

	00450	* CORE ADDRESSES		
1C00	00460	REGXFR	EQU	\$1C00
1C0E	00470	VIDCLS	EQU	\$1C0E
1C1F	00480	PUTCHR	EQU	\$1C1F
1CD2	00490	GETCHR	EQU	\$1CD2
1CD5	00500	PUTBYT	EQU	\$1CD5
1D17	00510	SCROLL	EQU	\$1D17
1D36	00520	PUTCHA	EQU	\$1D36
1D57	00530	PUTBYA	EQU	\$1D57
1D8C	00540	CRLF	EQU	\$1D8C
1DBD	00550	PK2PRT	EQU	\$1DBD
1DDA	00560	PRT2PK	EQU	\$1DDA
1DF7	00570	POLCAT	EQU	\$1DF7
1E06	00580	PUTWRA	EQU	\$1E06
1E11	00590	PUTWRD	EQU	\$1E11
1E1C	00600	BKSPCE	EQU	\$1E1C
1E34	00610	DU1616	EQU	\$1E34
1E56	00620	FUDWRD	EQU	\$1E56
1E70	00630	PTDWRD	EQU	\$1E70
	00640			
	00650	* YACH ADDRESSES		
	00660	* SYSTEM FLAGS		
2000	00670	FLAGS	EQU	\$2000
2000	00680	SCFLAG	EQU	\$2000
2001	00690	STFLAG	EQU	\$2001
2002	00700	BFFLAG	EQU	\$2002
	00710			
	00720	* SYSTEM TEMPORARY		
	00730	* VARIABLES		
2040	00740	TMPVAR	EQU	\$2040
2040	00750	CMDCNT	EQU	\$2040
2041	00760	CMDVAR	EQU	\$2041
2046	00770	CMDFRF	EQU	\$2046
2047	00780	CMDFRR	EQU	\$2047
2048	00790	CMDTOF	EQU	\$2048
2049	00800	CMDTOR	EQU	\$2049
204A	00810	CMDFRI	EQU	\$204A
204B	00820	CMDFRJ	EQU	\$204B
204C	00830	CMDTOI	EQU	\$204C
204D	00840	CMDTOJ	EQU	\$204D
204E	00850	MCHVAR	EQU	\$204E
	00860			
	00870	* SCREEN MANAGEMENT		
20C0	00880	CIDGEN	EQU	\$20C0
24C0	00890	B2SARY	EQU	\$24C0
2540	00900	B2DARY	EQU	\$2540
2580	00910	B2LARY	EQU	\$2580

25C0	00920	BSCARY	EQU	\$25C0
	00930			
	00940	* GENERAL REPORTING		
2600	00950	NOTREC	EQU	\$2600
2680	00960	NYIMSG	EQU	\$2680
	00970			
	00980	* GENERAL COMMANDS		
26C0	00990	ZERO2	EQU	\$26C0
26D0	01000	ZERO3	EQU	\$26D0
26E0	01010	ANLYZ	EQU	\$26E0
	01020	*ASKFR	EQU	\$27E0
28E0	01030	CLRBD	EQU	\$28E0
29E0	01040	CKMOV	EQU	\$29E0
2AE0	01050	CKPOS	EQU	\$2AE0
2BE0	01060	CSTLK	EQU	\$2BE0
2CE0	01070	CSTLQ	EQU	\$2CE0
2DE0	01080	DSPBD	EQU	\$2DE0
2EE0	01090	FLIP	EQU	\$2EE0
2FE0	01100	GSTRT	EQU	\$2FE0
30E0	01110	GSTOP	EQU	\$30E0
31E0	01120	LDGAM	EQU	\$31E0
32E0	01130	LDPOS	EQU	\$32E0
33E0	01140	NEW	EQU	\$33E0
33F0	01150	NEWPS	EQU	\$33F0
34F0	01160	LTRO2	EQU	\$34F0
3500	01170	LTRO3	EQU	\$3500
3510	01180	QUIT	EQU	\$3510
3520	01190	SCOFF	EQU	\$3520
3620	01200	SCON	EQU	\$3620
3720	01210	START	EQU	\$3720
3730	01220	STOP	EQU	\$3730
3740	01230	SVGAM	EQU	\$3740
3840	01240	SVPOS	EQU	\$3840
3940	01250	TKBAK	EQU	\$3940
	01260			
	01270	* MOVEMENT COMMANDS		
	01280	* AND CALCULATIONS		
3A40	01290	FILCVT	EQU	\$3A40
3A60	01300	RNKCVT	EQU	\$3A60
3A80	01310	FRE0	EQU	\$3A80
3A90	01320	FRECP	EQU	\$3A90
3AA0	01330	FRFR	EQU	\$3AA0
3EA0	01340	FRFRE	EQU	\$3EA0
3FA0	01350	FRFRP	EQU	\$3FA0
40A0	01360	FRQ0	EQU	\$40A0
41A0	01370	FRQCP	EQU	\$41A0
	01380			

```

01390 * COMMAND MANAGEMENT
42A0 01400 GETCMD EQU $42A0
43A0 01410 MATCH3 EQU $43A0
4420 01420 MATCH4 EQU $4420
44A0 01430 XLTCMD EQU $44A0
48A0 01440 XLTCM2 EQU $48A0
01450
01460 * YACH SUPERVISORY
4CA0 01470 GMLOOP EQU $4CA0
4CE0 01480 YACH EQU $4CE0
01490
27E0 01500 ORG $27E0
01510
27E0 34 36 01520 ASKFR PSHS A,B,X,Y
01530
01540 * POINT TO THE
01550 * REPORTING FIELD
27E2 8E 05EB 01560 LDX #RPTFLD
01570
01580 * COMMAND NYI
01590 * MESSAGE
27E5 86 41 01600 LDA #65 A
27E7 A7 80 01610 STA ,X+
27E9 86 53 01620 LDA #83 S
27EB A7 80 01630 STA ,X+
27ED 86 4B 01640 LDA #75 K
27EF A7 80 01650 STA ,X+
01660
01670 * BRANCH TO NYI
01680 * MESSAGE
27F1 16 FE8C 01690 LBRA NYIMSG
0000 32767 END

```

00000 TOTAL ERRORS

=====

CLRBD: Clears the Board

```
00100 *****
00110 *
00120 * CLRBD.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * CLEARS THE BOARD
00160 *
00170 *****
00180
00190 * LOW RAM CURSOR ADDRESS
0088 00200 CURPOS EQU $0088
00210
00220 * SCREEN ADDRESSES
00230 * START OF VIDRAM
0400 00240 VIDRAM EQU $0400
00250
00260 * COMMAND PROMPT
00270 * SCREEN LOCATION
05AE 00280 CMDPMT EQU $05AE
00290
00300 * SCREEN REPORTING FIELD
05EB 00310 RPTFLD EQU $05EB
00320
00330 * ONE BYTE PAST THE
00340 * END OF VIDRAM
0600 00350 VIDEND EQU $0600
00360
00370 * ML FOUNDATION
00380 * CORE ADDRESSES
1C00 00390 REGXFR EQU $1C00
1C0E 00400 VIDCLS EQU $1C0E
1C1F 00410 PUTCHR EQU $1C1F
1CD2 00420 GETCHR EQU $1CD2
1CD5 00430 PUTBYT EQU $1CD5
1D17 00440 SCROLL EQU $1D17
1D36 00450 PUTCHA EQU $1D36
1D57 00460 PUTBYA EQU $1D57
1D8C 00470 CRLF EQU $1D8C
1DBD 00480 PK2PRT EQU $1DBD
1DDA 00490 PRT2PK EQU $1DDA
1DF7 00500 POLCAT EQU $1DF7
1E06 00510 PUTWRA EQU $1E06
1E11 00520 PUTWRD EQU $1E11
```

1E1C	00530	BKSPCE	EQU	\$1E1C
1E34	00540	DU1616	EQU	\$1E34
1E56	00550	FUDWRD	EQU	\$1E56
1E70	00560	PTDWRD	EQU	\$1E70
	00570			
	00580	* YACH ADDRESSES		
	00590	* SYSTEM FLAGS		
2000	00600	FLAGS	EQU	\$2000
2000	00610	SCFLAG	EQU	\$2000
2001	00620	STFLAG	EQU	\$2001
2002	00630	BFFLAG	EQU	\$2002
	00640			
	00650	* SYSTEM TEMPORARY		
	00660	* VARIABLES		
2040	00670	TMPVAR	EQU	\$2040
2040	00680	CMDCNT	EQU	\$2040
2041	00690	CMDVAR	EQU	\$2041
2046	00700	CMDFRF	EQU	\$2046
2047	00710	CMDFRR	EQU	\$2047
2048	00720	CMDTOF	EQU	\$2048
2049	00730	CMDTOR	EQU	\$2049
204A	00740	CMDFRI	EQU	\$204A
204B	00750	CMDFRJ	EQU	\$204B
204C	00760	CMDTOI	EQU	\$204C
204D	00770	CMDTOJ	EQU	\$204D
204E	00780	MCHVAR	EQU	\$204E
	00790			
	00800	* SCREEN MANAGEMENT		
20C0	00810	CIDGEN	EQU	\$20C0
24C0	00820	B2SARY	EQU	\$24C0
2540	00830	B2DARY	EQU	\$2540
2580	00840	B2LARY	EQU	\$2580
25C0	00850	BSCARY	EQU	\$25C0
	00860			
	00870	* GENERAL REPORTING		
2600	00880	NOTREC	EQU	\$2600
2680	00890	NYIMSG	EQU	\$2680
	00900			
	00910	* GENERAL COMMANDS		
26C0	00920	ZERO2	EQU	\$26C0
26D0	00930	ZERO3	EQU	\$26D0
26E0	00940	ANLYZ	EQU	\$26E0
27E0	00950	ASKFR	EQU	\$27E0
	00960	*CLRBD	EQU	\$28E0
29E0	00970	CKMOV	EQU	\$29E0
2AE0	00980	CKPOS	EQU	\$2AE0
2BE0	00990	CSTLK	EQU	\$2BE0

2CE0	01000	CSTLQ	EQU	\$2CE0
2DE0	01010	DSPBD	EQU	\$2DE0
2EE0	01020	FLIP	EQU	\$2EE0
2FE0	01030	GSTRT	EQU	\$2FE0
30E0	01040	GSTOP	EQU	\$30E0
31E0	01050	LDGAM	EQU	\$31E0
32E0	01060	LDPOS	EQU	\$32E0
33E0	01070	NEW	EQU	\$33E0
33F0	01080	NEWPS	EQU	\$33F0
34F0	01090	LTRO2	EQU	\$34F0
3500	01100	LTRO3	EQU	\$3500
3510	01110	QUIT	EQU	\$3510
3520	01120	SCOFF	EQU	\$3520
3620	01130	SCON	EQU	\$3620
3720	01140	START	EQU	\$3720
3730	01150	STOP	EQU	\$3730
3740	01160	SVGAM	EQU	\$3740
3840	01170	SVPOS	EQU	\$3840
3940	01180	TKBAK	EQU	\$3940
	01190			
	01200	* MOVEMENT COMMANDS		
	01210	* AND CALCULATIONS		
3A40	01220	FILCVT	EQU	\$3A40
3A60	01230	RNKCVT	EQU	\$3A60
3A80	01240	FRE0	EQU	\$3A80
3A90	01250	FRECP	EQU	\$3A90
3AA0	01260	FRFR	EQU	\$3AA0
3EA0	01270	FRFRE	EQU	\$3EA0
3FA0	01280	FRFRP	EQU	\$3FA0
40A0	01290	FRQ0	EQU	\$40A0
41A0	01300	FRQCP	EQU	\$41A0
	01310			
	01320	* COMMAND MANAGEMENT		
42A0	01330	GETCMD	EQU	\$42A0
43A0	01340	MATCH3	EQU	\$43A0
4420	01350	MATCH4	EQU	\$4420
44A0	01360	XLTCMD	EQU	\$44A0
48A0	01370	XLTCM2	EQU	\$48A0
	01380			
	01390	* YACH SUPERVISORY		
4CA0	01400	GMLOOP	EQU	\$4CA0
4CE0	01410	YACH	EQU	\$4CE0
	01420			
28E0		01430	ORG	\$28E0
		01440		
28E0	34	52	01450	CLRBD PSHS A,X,U
			01460	

```

01470 * CLEAR THE
01480 * BOARD SURFACE CONTENTS
01490 * ARRAY
01500
01510 * POINT TO THE
01520 * BOARD SURFACE CONTENTS
01530 * ARRAY
28E2 8E 25C0 01540          LDX          #BSCARY
01550
01560 * ALL SQUARES' CONTENTS
01570 *   = NONE = SPACE = 096
28E5 86 60 01580          LDA          #96
01590
01600 * CLEAR A SQUARE
28E7 A7 80 01610 L28E7  STA          ,X+
01620
01630 * ARE WE DONE?
28E9 8C 2600 01640          CMPX         #BSCARY+64
01650
01660 * GO IF NO
28EC 25 F9 01670          BLO          L28E7
01680
01690 * CLEAR THE DISPLAY SCREEN
01700 * TO ALL SPACES
01710
01720 * POINT TO THE
01730 * BOARD TO SCREEN
01740 * TRANSLATION ARRAY
28EE CE 24C0 01750          LDU          #B2SARY
01760
01770 * ALL DISPLAY SQUARES
01780 *   = SPACE = 096
01790 * REGISTER A STILL = #96
01800
01810 * GET A SQUARE'S
01820 * SCREEN ADDRESS
28F1 AE C1 01830 L28F1  LDX          ,U++
01840
01850 * SET THE SQUARE
01860 * TO A SPACE
28F3 A7 84 01870          STA          ,X
01880
01890 * ARE WE DONE?
28F5 1183 2540 01900          CMPU         #B2SARY+128
01910
01920 * GO IF NO
28F9 25 F6 01930          BLO          L28F1

```

			01940		
			01950	* GET THE	
			01960	* SCREEN COLOR FLAG	
28FB	B6	2000	01970	LDA	SCFLAG
			01980		
			01990	* GO IF CLEAR	
28FE	27	05	02000	BEQ	L2905
			02010		
			02020	* SCREEN COLORS ON	
2900	17	0D1D	02030	LBSR	SCON
2903	20	03	02040	BRA	L2908
			02050		
			02060	* SCREEN COLORS OFF	
2905	17	0C18	02070	L2905 LBSR	SCOFF
			02080		
			02090	* EXIT	
2908	35	52	02100	L2908 PULS	A,X,U
290A	39		02110	RTS	
		0000	32767	END	

00000 TOTAL ERRORS

=====

CKMOV: Check the Legality Of a Proposed Move (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * CKMOV.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * CHECK THE LEGALITY
00170 * OF A PROPOSED MOVE
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00
1C0E 00450 VIDCLS EQU $1C0E
1C1F 00460 PUTCHR EQU $1C1F
1CD2 00470 GETCHR EQU $1CD2
1CD5 00480 PUTBYT EQU $1CD5

```

1D17	00490	SCROLL	EQU	\$1D17
1D36	00500	PUTCHA	EQU	\$1D36
1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			
	00920	* GENERAL REPORTING		
2600	00930	NOTREC	EQU	\$2600
2680	00940	NYIMSG	EQU	\$2680
	00950			

	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2	EQU	\$26C0
26D0	00980	ZERO3	EQU	\$26D0
26E0	00990	ANLYZ	EQU	\$26E0
27E0	01000	ASKFR	EQU	\$27E0
28E0	01010	CLRBD	EQU	\$28E0
	01020	*CKMOV	EQU	\$29E0
2AE0	01030	CKPOS	EQU	\$2AE0
2BE0	01040	CSTLK	EQU	\$2BE0
2CE0	01050	CSTLQ	EQU	\$2CE0
2DE0	01060	DSPBD	EQU	\$2DE0
2EE0	01070	FLIP	EQU	\$2EE0
2FE0	01080	GSTRT	EQU	\$2FE0
30E0	01090	GSTOP	EQU	\$30E0
31E0	01100	LDGAM	EQU	\$31E0
32E0	01110	LDPOS	EQU	\$32E0
33E0	01120	NEW	EQU	\$33E0
33F0	01130	NEWPS	EQU	\$33F0
34F0	01140	LTRO2	EQU	\$34F0
3500	01150	LTRO3	EQU	\$3500
3510	01160	QUIT	EQU	\$3510
3520	01170	SCOFF	EQU	\$3520
3620	01180	SCON	EQU	\$3620
3720	01190	START	EQU	\$3720
3730	01200	STOP	EQU	\$3730
3740	01210	SVGAM	EQU	\$3740
3840	01220	SVPOS	EQU	\$3840
3940	01230	TKBAK	EQU	\$3940
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT	EQU	\$3A40
3A60	01280	RNKCVT	EQU	\$3A60
3A80	01290	FRE0	EQU	\$3A80
3A90	01300	FRECP	EQU	\$3A90
3AA0	01310	FRFR	EQU	\$3AA0
3EA0	01320	FRFRE	EQU	\$3EA0
3FA0	01330	FRFRP	EQU	\$3FA0
40A0	01340	FRQ0	EQU	\$40A0
41A0	01350	FRQCP	EQU	\$41A0
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD	EQU	\$42A0
43A0	01390	MATCH3	EQU	\$43A0
4420	01400	MATCH4	EQU	\$4420
44A0	01410	XLTCMD	EQU	\$44A0
48A0	01420	XLTCM2	EQU	\$48A0

```

01430
01440 * YACH SUPERVISORY
      4CA0 01450 GMLOOP EQU $4CA0
      4CE0 01460 YACH EQU $4CE0
01470
29E0 01480 ORG $29E0
01490
29E0 34 36 01500 CKMOV PSHS A,B,X,Y
01510
01520 * POINT TO THE
01530 * REPORTING FIELD
29E2 8E 05EB 01540 LDX #RPTFLD
01550
01560 * COMMAND NYI
01570 * MESSAGE
29E5 86 43 01580 LDA #67 C
29E7 A7 80 01590 STA ,X+
29E9 86 4B 01600 LDA #75 K
29EB A7 80 01610 STA ,X+
29ED 86 4D 01620 LDA #77 M
29EF A7 80 01630 STA ,X+
29F1 86 4F 01640 LDA #79 O
29F3 A7 80 01650 STA ,X+
29F5 86 56 01660 LDA #86 V
29F7 A7 80 01670 STA ,X+
01680
01690 * BRANCH TO NYI
01700 * MESSAGE
29F9 16 FC84 01710 LBRA NYIMSG
      0000 32767 END

```

00000 TOTAL ERRORS

=====

CKPOS: Check the Legality Of the Displayed Position (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * CKPOS.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * CHECK THE LEGALITY
00170 * OF A DISPLAYED
00180 * POSITION
00190 *
00200 * NOT YET
00210 * IMPLEMENTED
00220 *
00230 *****
00240
00250 * LOW RAM CURSOR ADDRESS
0088 00260 CURPOS EQU $0088
00270
00280 * SCREEN ADDRESSES
00290 * START OF VIDRAM
0400 00300 VIDRAM EQU $0400
00310
00320 * COMMAND PROMPT
00330 * SCREEN LOCATION
05AE 00340 CMDPMT EQU $05AE
00350
00360 * SCREEN REPORTING FIELD
05EB 00370 RPTFLD EQU $05EB
00380
00390 * ONE BYTE PAST THE
00400 * END OF VIDRAM
0600 00410 VIDEND EQU $0600
00420
00430 * ML FOUNDATION
00440 * CORE ADDRESSES
1C00 00450 REGXFR EQU $1C00
1C0E 00460 VIDCLS EQU $1C0E
1C1F 00470 PUTCHR EQU $1C1F
1CD2 00480 GETCHR EQU $1CD2

```


1CD5	00490	PUTBYT	EQU	\$1CD5
1D17	00500	SCROLL	EQU	\$1D17
1D36	00510	PUTCHA	EQU	\$1D36
1D57	00520	PUTBYA	EQU	\$1D57
1D8C	00530	CRLF	EQU	\$1D8C
1DBD	00540	PK2PRT	EQU	\$1DBD
1DDA	00550	PRT2PK	EQU	\$1DDA
1DF7	00560	POLCAT	EQU	\$1DF7
1E06	00570	PUTWRA	EQU	\$1E06
1E11	00580	PUTWRD	EQU	\$1E11
1E1C	00590	BKSPCE	EQU	\$1E1C
1E34	00600	DU1616	EQU	\$1E34
1E56	00610	FUDWRD	EQU	\$1E56
1E70	00620	PTDWRD	EQU	\$1E70
	00630			
	00640	* YACH ADDRESSES		
	00650	* SYSTEM FLAGS		
2000	00660	FLAGS	EQU	\$2000
2000	00670	SCFLAG	EQU	\$2000
2001	00680	STFLAG	EQU	\$2001
2002	00690	BFFLAG	EQU	\$2002
	00700			
	00710	* SYSTEM TEMPORARY		
	00720	* VARIABLES		
2040	00730	TMPVAR	EQU	\$2040
2040	00740	CMDCNT	EQU	\$2040
2041	00750	CMDVAR	EQU	\$2041
2046	00760	CMDFRF	EQU	\$2046
2047	00770	CMDFRR	EQU	\$2047
2048	00780	CMDTOF	EQU	\$2048
2049	00790	CMDTOR	EQU	\$2049
204A	00800	CMDFRI	EQU	\$204A
204B	00810	CMDFRJ	EQU	\$204B
204C	00820	CMDTOI	EQU	\$204C
204D	00830	CMDTOJ	EQU	\$204D
204E	00840	MCHVAR	EQU	\$204E
	00850			
	00860	* SCREEN MANAGEMENT		
20C0	00870	CIDGEN	EQU	\$20C0
24C0	00880	B2SARY	EQU	\$24C0
2540	00890	B2DARY	EQU	\$2540
2580	00900	B2LARY	EQU	\$2580
25C0	00910	BSCARY	EQU	\$25C0
	00920			
	00930	* GENERAL REPORTING		
2600	00940	NOTREC	EQU	\$2600
2680	00950	NYIMSG	EQU	\$2680

```

00960
00970 * GENERAL COMMANDS
26C0 00980 ZERO2 EQU $26C0
26D0 00990 ZERO3 EQU $26D0
26E0 01000 ANLYZ EQU $26E0
27E0 01010 ASKFR EQU $27E0
28E0 01020 CLRBD EQU $28E0
29E0 01030 CKMOV EQU $29E0
      01040 *CKPOS EQU $2AE0
2BE0 01050 CSTLK EQU $2BE0
2CE0 01060 CSTLQ EQU $2CE0
2DE0 01070 DSPBD EQU $2DE0
2EE0 01080 FLIP EQU $2EE0
2FE0 01090 GSTRT EQU $2FE0
30E0 01100 GSTOP EQU $30E0
31E0 01110 LDGAM EQU $31E0
32E0 01120 LDPOS EQU $32E0
33E0 01130 NEW EQU $33E0
33F0 01140 NEWPS EQU $33F0
34F0 01150 LTRO2 EQU $34F0
3500 01160 LTRO3 EQU $3500
3510 01170 QUIT EQU $3510
3520 01180 SCOFF EQU $3520
3620 01190 SCON EQU $3620
3720 01200 START EQU $3720
3730 01210 STOP EQU $3730
3740 01220 SVGAM EQU $3740
3840 01230 SVPOS EQU $3840
3940 01240 TKBAK EQU $3940
01250
01260 * MOVEMENT COMMANDS
01270 * AND CALCULATIONS
3A40 01280 FILCVT EQU $3A40
3A60 01290 RNKCVT EQU $3A60
3A80 01300 FRE0 EQU $3A80
3A90 01310 FRECP EQU $3A90
3AA0 01320 FRFR EQU $3AA0
3EA0 01330 FRFRE EQU $3EA0
3FA0 01340 FRFRP EQU $3FA0
40A0 01350 FRQ0 EQU $40A0
41A0 01360 FRQCP EQU $41A0
01370
01380 * COMMAND MANAGEMENT
42A0 01390 GETCMD EQU $42A0
43A0 01400 MATCH3 EQU $43A0
4420 01410 MATCH4 EQU $4420
44A0 01420 XLTCMD EQU $44A0

```

	48A0		01430	XLTCM2	EQU	\$48A0	
			01440				
			01450	* YACH SUPERVISORY			
	4CA0		01460	GMLOOP	EQU	\$4CA0	
	4CE0		01470	YACH	EQU	\$4CE0	
			01480				
2AE0			01490		ORG	\$2AE0	
			01500				
2AE0	34	36	01510	CKPOS	PSHS	A,B,X,Y	
			01520				
			01530	* POINT TO THE			
			01540	* REPORTING FIELD			
2AE2	8E	05EB	01550		LDX	#RPTFLD	
			01560				
			01570	* COMMAND NYI			
			01580	* MESSAGE			
2AE5	86	43	01590		LDA	#67	C
2AE7	A7	80	01600		STA	,X+	
2AE9	86	4B	01610		LDA	#75	K
2AEB	A7	80	01620		STA	,X+	
2AED	86	50	01630		LDA	#80	P
2AEF	A7	80	01640		STA	,X+	
2AF1	86	4F	01650		LDA	#79	O
2AF3	A7	80	01660		STA	,X+	
2AF5	86	53	01670		LDA	#83	S
2AF7	A7	80	01680		STA	,X+	
			01690				
			01700	* BRANCH TO NYI			
			01710	* MESSAGE			
2AF9	16	FB84	01720		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

CSTLK: Castle Kingside Action Command (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * CSTLK.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * CASTLE KINGSIDE
00170 * ACTION COMMAND
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00
1C0E 00450 VIDCLS EQU $1C0E
1C1F 00460 PUTCHR EQU $1C1F
1CD2 00470 GETCHR EQU $1CD2
1CD5 00480 PUTBYT EQU $1CD5

```

1D17	00490	SCROLL	EQU	\$1D17
1D36	00500	PUTCHA	EQU	\$1D36
1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			
	00920	* GENERAL REPORTING		
2600	00930	NOTREC	EQU	\$2600
2680	00940	NYIMSG	EQU	\$2680
	00950			

	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2	EQU	\$26C0
26D0	00980	ZERO3	EQU	\$26D0
26E0	00990	ANLYZ	EQU	\$26E0
27E0	01000	ASKFR	EQU	\$27E0
28E0	01010	CLRBD	EQU	\$28E0
29E0	01020	CKMOV	EQU	\$29E0
2AE0	01030	CKPOS	EQU	\$2AE0
	01040	*CSTLK	EQU	\$2BE0
2CE0	01050	CSTLQ	EQU	\$2CE0
2DE0	01060	DSPBD	EQU	\$2DE0
2EE0	01070	FLIP	EQU	\$2EE0
2FE0	01080	GSTRT	EQU	\$2FE0
30E0	01090	GSTOP	EQU	\$30E0
31E0	01100	LDGAM	EQU	\$31E0
32E0	01110	LDPOS	EQU	\$32E0
33E0	01120	NEW	EQU	\$33E0
33F0	01130	NEWPS	EQU	\$33F0
34F0	01140	LTRO2	EQU	\$34F0
3500	01150	LTRO3	EQU	\$3500
3510	01160	QUIT	EQU	\$3510
3520	01170	SCOFF	EQU	\$3520
3620	01180	SCON	EQU	\$3620
3720	01190	START	EQU	\$3720
3730	01200	STOP	EQU	\$3730
3740	01210	SVGAM	EQU	\$3740
3840	01220	SVPOS	EQU	\$3840
3940	01230	TKBAK	EQU	\$3940
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT	EQU	\$3A40
3A60	01280	RNKCVT	EQU	\$3A60
3A80	01290	FRE0	EQU	\$3A80
3A90	01300	FRECP	EQU	\$3A90
3AA0	01310	FRFR	EQU	\$3AA0
3EA0	01320	FRFRE	EQU	\$3EA0
3FA0	01330	FRFRP	EQU	\$3FA0
40A0	01340	FRQ0	EQU	\$40A0
41A0	01350	FRQCP	EQU	\$41A0
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD	EQU	\$42A0
43A0	01390	MATCH3	EQU	\$43A0
4420	01400	MATCH4	EQU	\$4420
44A0	01410	XLTCMD	EQU	\$44A0
48A0	01420	XLTCM2	EQU	\$48A0

```

01430
01440 * YACH SUPERVISORY
      4CA0 01450 GMLOOP EQU $4CA0
      4CE0 01460 YACH EQU $4CE0
01470
2BE0 01480 ORG $2BE0
01490
2BE0 34 36 01500 CASTLK PSHS A,B,X,Y
01510
01520 * POINT TO THE
01530 * REPORTING FIELD
2BE2 8E 05EB 01540 LDX #RPTFLD
01550
01560 * COMMAND NYI
01570 * MESSAGE
2BE5 86 43 01580 LDA #67 C
2BE7 A7 80 01590 STA ,X+
2BE9 86 41 01600 LDA #65 A
2BEB A7 80 01610 STA ,X+
2BED 86 53 01620 LDA #83 S
2BEF A7 80 01630 STA ,X+
2BF1 86 54 01640 LDA #84 T
2BF3 A7 80 01650 STA ,X+
2BF5 86 4C 01660 LDA #76 L
2BF7 A7 80 01670 STA ,X+
2BF9 86 49 01680 LDA #73 I
2BFB A7 80 01690 STA ,X+
2BFD 86 4E 01700 LDA #78 N
2BFF A7 80 01710 STA ,X+
2C01 86 47 01720 LDA #71 G
2C03 A7 80 01730 STA ,X+
01760 * MESSAGE
2C05 16 FA78 01770 LBRA NYIMSG
      0000 32767 END

```

00000 TOTAL ERRORS

=====

CSTLQ: Castle Queenside Action Command (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * CSTLQ.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * CASTLE QUEENSIDE
00170 * ACTION COMMAND
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00
1C0E 00450 VIDCLS EQU $1C0E
1C1F 00460 PUTCHR EQU $1C1F
1CD2 00470 GETCHR EQU $1CD2
1CD5 00480 PUTBYT EQU $1CD5

```


1D17	00490	SCROLL	EQU	\$1D17
1D36	00500	PUTCHA	EQU	\$1D36
1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			
	00920	* GENERAL REPORTING		
2600	00930	NOTREC	EQU	\$2600
2680	00940	NYIMSG	EQU	\$2680
	00950			

	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2	EQU	\$26C0
26D0	00980	ZERO3	EQU	\$26D0
26E0	00990	ANLYZ	EQU	\$26E0
27E0	01000	ASKFR	EQU	\$27E0
28E0	01010	CLRBD	EQU	\$28E0
29E0	01020	CKMOV	EQU	\$29E0
2AE0	01030	CKPOS	EQU	\$2AE0
2BE0	01040	CSTLK	EQU	\$2BE0
	01050	*CSTLQ	EQU	\$2CE0
2DE0	01060	DSPBD	EQU	\$2DE0
2EE0	01070	FLIP	EQU	\$2EE0
2FE0	01080	GSTRT	EQU	\$2FE0
30E0	01090	GSTOP	EQU	\$30E0
31E0	01100	LDGAM	EQU	\$31E0
32E0	01110	LDPOS	EQU	\$32E0
33E0	01120	NEW	EQU	\$33E0
33F0	01130	NEWPS	EQU	\$33F0
34F0	01140	LTRO2	EQU	\$34F0
3500	01150	LTRO3	EQU	\$3500
3510	01160	QUIT	EQU	\$3510
3520	01170	SCOFF	EQU	\$3520
3620	01180	SCON	EQU	\$3620
3720	01190	START	EQU	\$3720
3730	01200	STOP	EQU	\$3730
3740	01210	SVGAM	EQU	\$3740
3840	01220	SVPOS	EQU	\$3840
3940	01230	TKBAK	EQU	\$3940
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT	EQU	\$3A40
3A60	01280	RNKCVT	EQU	\$3A60
3A80	01290	FRE0	EQU	\$3A80
3A90	01300	FRECP	EQU	\$3A90
3AA0	01310	FRFR	EQU	\$3AA0
3EA0	01320	FRFRE	EQU	\$3EA0
3FA0	01330	FRFRP	EQU	\$3FA0
40A0	01340	FRQ0	EQU	\$40A0
41A0	01350	FRQCP	EQU	\$41A0
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD	EQU	\$42A0
43A0	01390	MATCH3	EQU	\$43A0
4420	01400	MATCH4	EQU	\$4420
44A0	01410	XLTCMD	EQU	\$44A0
48A0	01420	XLTCM2	EQU	\$48A0

```

01430
01440 * YACH SUPERVISORY
      4CA0 01450 GMLOOP EQU $4CA0
      4CE0 01460 YACH EQU $4CE0
01470
2CE0 01480 ORG $2CE0
01490
2CE0 34 36 01500 CASTLQ PSHS A,B,X,Y
01510
01520 * POINT TO THE
01530 * REPORTING FIELD
2CE2 8E 05EB 01540 LDX #RPTFLD
01550
01560 * COMMAND NYI
01570 * MESSAGE
2CE5 86 43 01580 LDA #67 C
2CE7 A7 80 01590 STA ,X+
2CE9 86 41 01600 LDA #65 A
2CEB A7 80 01610 STA ,X+
2CED 86 53 01620 LDA #83 S
2CEF A7 80 01630 STA ,X+
2CF1 86 54 01640 LDA #84 T
2CF3 A7 80 01650 STA ,X+
2CF5 86 4C 01660 LDA #76 L
2CF7 A7 80 01670 STA ,X+
2CF9 86 49 01680 LDA #73 I
2CFB A7 80 01690 STA ,X+
2CFD 86 4E 01700 LDA #78 N
2CFF A7 80 01710 STA ,X+
2D01 86 47 01720 LDA #71 G
2D03 A7 80 01730 STA ,X+
01740
01750 * BRANCH TO NYI
01760 * MESSAGE
2D05 16 F978 01770 LBRA NYIMSG
      0000 32767 END

```

00000 TOTAL ERRORS

=====

DSPBD: Copies the Board Surface Contents Array to the Display Screen

```

00100 *****
00110 *
00120 * DSPBD.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * COPIES THE
00160 * BOARD SURFACE CONTENTS
00170 * ARRAY TO THE
00180 * DISPLAY SCREEN
00190 *
00200 *****
00210
00220 * LOW RAM CURSOR ADDRESS
0088 00230 CURPOS EQU $0088
00240
00250 * SCREEN ADDRESSES
00260 * START OF VIDRAM
0400 00270 VIDRAM EQU $0400
00280
00290 * COMMAND PROMPT
00300 * SCREEN LOCATION
05AE 00310 CMDPMT EQU $05AE
00320
00330 * SCREEN REPORTING FIELD
05EB 00340 RPTFLD EQU $05EB
00350
00360 * ONE BYTE PAST THE
00370 * END OF VIDRAM
0600 00380 VIDEND EQU $0600
00390
00400 * ML FOUNDATION
00410 * CORE ADDRESSES
1C00 00420 REGXFR EQU $1C00
1C0E 00430 VIDCLS EQU $1C0E
1C1F 00440 PUTCHR EQU $1C1F
1CD2 00450 GETCHR EQU $1CD2
1CD5 00460 PUTBYT EQU $1CD5
1D17 00470 SCROLL EQU $1D17
1D36 00480 PUTCHA EQU $1D36

```

1D57	00490	PUTBYA	EQU	\$1D57
1D8C	00500	CRLF	EQU	\$1D8C
1DBD	00510	PK2PRT	EQU	\$1DBD
1DDA	00520	PRT2PK	EQU	\$1DDA
1DF7	00530	POLCAT	EQU	\$1DF7
1E06	00540	PUTWRA	EQU	\$1E06
1E11	00550	PUTWRD	EQU	\$1E11
1E1C	00560	BKSPCE	EQU	\$1E1C
1E34	00570	DU1616	EQU	\$1E34
1E56	00580	FUDWRD	EQU	\$1E56
1E70	00590	PTDWRD	EQU	\$1E70
	00600			
	00610	* YACH ADDRESSES		
	00620	* SYSTEM FLAGS		
2000	00630	FLAGS	EQU	\$2000
2000	00640	SCFLAG	EQU	\$2000
2001	00650	STFLAG	EQU	\$2001
2002	00660	BFFLAG	EQU	\$2002
	00670			
	00680	* SYSTEM TEMPORARY		
	00690	* VARIABLES		
2040	00700	TMPVAR	EQU	\$2040
2040	00710	CMDCNT	EQU	\$2040
2041	00720	CMDVAR	EQU	\$2041
2046	00730	CMDFRF	EQU	\$2046
2047	00740	CMDFRR	EQU	\$2047
2048	00750	CMDTOF	EQU	\$2048
2049	00760	CMDTOR	EQU	\$2049
204A	00770	CMDFRI	EQU	\$204A
204B	00780	CMDFRJ	EQU	\$204B
204C	00790	CMDTOI	EQU	\$204C
204D	00800	CMDTOJ	EQU	\$204D
204E	00810	MCHVAR	EQU	\$204E
	00820			
	00830	* SCREEN MANAGEMENT		
20C0	00840	CIDGEN	EQU	\$20C0
24C0	00850	B2SARY	EQU	\$24C0
2540	00860	B2DARY	EQU	\$2540
2580	00870	B2LARY	EQU	\$2580
25C0	00880	BSCARY	EQU	\$25C0
	00890			
	00900	* GENERAL REPORTING		
2600	00910	NOTREC	EQU	\$2600
2680	00920	NYIMSG	EQU	\$2680
	00930			
	00940	* GENERAL COMMANDS		
26C0	00950	ZERO2	EQU	\$26C0

26D0	00960	ZERO3	EQU	\$26D0
26E0	00970	ANLYZ	EQU	\$26E0
27E0	00980	ASKFR	EQU	\$27E0
28E0	00990	CLRBD	EQU	\$28E0
29E0	01000	CKMOV	EQU	\$29E0
2AE0	01010	CKPOS	EQU	\$2AE0
2BE0	01020	CSTLK	EQU	\$2BE0
2CE0	01030	CSTLQ	EQU	\$2CE0
	01040	*DSPBD	EQU	\$2DE0
2EE0	01050	FLIP	EQU	\$2EE0
2FE0	01060	GSTRT	EQU	\$2FE0
30E0	01070	GSTOP	EQU	\$30E0
31E0	01080	LDGAM	EQU	\$31E0
32E0	01090	LDPOS	EQU	\$32E0
33E0	01100	NEW	EQU	\$33E0
33F0	01110	NEWPS	EQU	\$33F0
34F0	01120	LTRO2	EQU	\$34F0
3500	01130	LTRO3	EQU	\$3500
3510	01140	QUIT	EQU	\$3510
3520	01150	SCOFF	EQU	\$3520
3620	01160	SCON	EQU	\$3620
3720	01170	START	EQU	\$3720
3730	01180	STOP	EQU	\$3730
3740	01190	SVGAM	EQU	\$3740
3840	01200	SVPOS	EQU	\$3840
3940	01210	TKBAK	EQU	\$3940
	01220			
	01230	* MOVEMENT COMMANDS		
	01240	* AND CALCULATIONS		
3A40	01250	FILCVT	EQU	\$3A40
3A60	01260	RNKCVT	EQU	\$3A60
3A80	01270	FRE0	EQU	\$3A80
3A90	01280	FRECP	EQU	\$3A90
3AA0	01290	FRFR	EQU	\$3AA0
3EA0	01300	FRFRE	EQU	\$3EA0
3FA0	01310	FRFRP	EQU	\$3FA0
40A0	01320	FRQ0	EQU	\$40A0
41A0	01330	FRQCP	EQU	\$41A0
	01340			
	01350	* COMMAND MANAGEMENT		
42A0	01360	GETCMD	EQU	\$42A0
43A0	01370	MATCH3	EQU	\$43A0
4420	01380	MATCH4	EQU	\$4420
44A0	01390	XLTCMD	EQU	\$44A0
48A0	01400	XLTCM2	EQU	\$48A0
	01410			
	01420	* YACH SUPERVISORY		

	4CA0		01430	GMLOOP	EQU	\$4CA0
	4CE0		01440	YACH	EQU	\$4CE0
			01450			
2DE0			01460		ORG	\$2DE0
			01470			
2DE0	34	72	01480	DSPBD	PSHS	A,X,Y,U
			01490			
			01500	* POINT TO THE		
			01510	* BOARD TO SCREEN		
			01520	* TRANSLATION ARRAY		
2DE2	CE	24C0	01530		LDU	#B2SARY
			01540			
			01550	* POINT TO THE		
			01560	* BOARD SURFACE CONTENTS		
			01570	* ARRAY		
2DE5	108E	25C0	01580		LDY	#BSCARY
			01590			
			01600	* GET A SQUARE'S CONTENTS		
2DE9	A6	A0	01610	L0001	LDA	,Y+
			01620			
			01630	* GET THE SQUARE'S		
			01640	* SCREEN ADDRESS		
2DEB	AE	C1	01650		LDX	,U++
			01660			
			01670	* DISPLAY THE CONTENTS		
			01680	* ON THE SCREEN		
2DED	A7	84	01690		STA	,X
			01700			
			01710	* ARE WE DONE?		
2DEF	1183	2540	01720		CMPU	#B2SARY+128
			01730			
			01740	* GO IF NO		
2DF3	25	F4	01750		BLO	L0001
			01760			
			01770	* GET THE		
			01780	* SCREEN COLOR FLAG		
2DF5	B6	2000	01790		LDA	SCFLAG
			01800			
			01810	* GO IF CLEAR		
2DF8	27	05	01820		BEQ	L0002
			01830			
			01840	* SCREEN COLORS ON		
2DFA	17	0823	01850		LBSR	SCON
2DFD	20	03	01860		BRA	L0003
			01870			
			01880	* SCREEN COLORS OFF		
2DFE	17	071E	01890	L0002	LBSR	SCOFF

```
01900
01910 * EXIT
2E02 35 72 01920 L0003 PULS A,X,Y,U
2E04 39 0000 01930 RTS
32767 END
```

00000 TOTAL ERRORS

=====

FLIP: Flip the Board

(N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * FLIP.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * FLIP THE BOARD
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57

```

1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0

26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
	01060	*FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0

	4CE0		01450	YACH	EQU	\$4CE0	
			01460				
2EE0			01470		ORG	\$2EE0	
			01480				
2EE0	34	36	01490	FLIP	PSHS	A,B,X,Y	
			01500				
			01510	* POINT TO THE			
			01520	* REPORTING FIELD			
2EE2	8E	05EB	01530		LDX	#RPTFLD	
			01540				
			01550	* COMMAND NYI			
			01560	* MESSAGE			
2EE5	86	46	01570		LDA	#70	F
2EE7	A7	80	01580		STA	,X+	
2EE9	86	4C	01590		LDA	#76	L
2EEB	A7	80	01600		STA	,X+	
2EED	86	49	01610		LDA	#73	I
2EEF	A7	80	01620		STA	,X+	
2EF1	86	50	01630		LDA	#80	P
2EF3	A7	80	01640		STA	,X+	
			01650				
			01660	* BRANCH TO NYI			
			01670	* MESSAGE			
2EF5	16	F788	01680		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

GSTRT: Restart Accepting Commands After a GSTOP Action Command

```

00100 *****
00110 *
00120 * GSTRT.ASM
00130 * MDJ 2021/09/21
00140 *
00150 * RESTART ACCEPTING
00160 * COMMANDS
00170 * AFTER A STOP
00180 * ACTION COMMAND
00190 *
00200 *****
00210
00220 * STOPPED FLAG
          2001 00230 STFLAG EQU $2001
00240
2FE0      00250          ORG $2FE0
00260
2FE0 34   02   00270 GSTRT PSHS A
00280
00290 * SET THE STOPPED FLAG
2FE2 86   01   00300          LDA #1
2FE4 B7   2001 00310          STA STFLAG
00320
00330 * EXIT
2FE7 35   02   00340          PULS A
2FE9 39           00350          RTS
          0000 32767          END

```

00000 TOTAL ERRORS

=====

GSTOP: Ignore All Subsequent Commands Except GSTRT Action Command

```
00100 *****
00110 *
00120 * GSTOP.ASM
00130 * MDJ 2021/09/21
00140 *
00150 * IGNORE ALL
00160 * SUBSEQUENT
00170 * COMMANDS
00180 * EXCEPT START
00190 * ACTION COMMAND
00200 *
00210 *****
00220
00230 * STOPPED FLAG
2001 00240 STFLAG EQU $2001
00250
30E0 00260 ORG $30E0
00270
00280 * CLEAR THE STOPPED FLAG
30E0 7F 2001 00290 GSTOP CLR STFLAG
00300
00310 * EXIT
30E3 39 00320 RTS
0000 32767 END

00000 TOTAL ERRORS
```

=====

MAKEGC1: Make the General Commands Group One GC1.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEGC1.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "ZERO2.BIN"
1130 LOADM "ZERO3.BIN"
1140 LOADM "ANLYZ.BIN"
1150 LOADM "ASKFR.BIN"
1160 LOADM "CLRBD.BIN"
1170 LOADM "CKMOV.BIN"
1180 LOADM "CKPOS.BIN"
1190 LOADM "CSTLK.BIN"
1200 LOADM "CSTLQ.BIN"
1210 LOADM "DSPBD.BIN"
1220 LOADM "FLIP.BIN"
1230 LOADM "GSTRT.BIN"
1240 LOADM "GSTOP.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "GC1.BIN", &H26C0, &H31DF, &H26C0
1330 '
32767 END
```

=====

LDGAM: Load a Game (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * LDGAM.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * LOAD A GAME
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57

```


1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0

26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
	01090	*LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0

	4CE0		01450	YACH	EQU	\$4CE0	
			01460				
31E0			01470		ORG	\$31E0	
			01480				
31E0	34	36	01490	LDGAM	PSHS	A,B,X,Y	
			01500				
			01510	* POINT TO THE			
			01520	* REPORTING FIELD			
31E2	8E	05EB	01530		LDX	#RPTFLD	
			01540				
			01550	* COMMAND NYI			
			01560	* MESSAGE			
31E5	86	4C	01570		LDA	#76	L
31E7	A7	80	01580		STA	,X+	
31E9	86	44	01590		LDA	#68	D
31EB	A7	80	01600		STA	,X+	
31ED	86	47	01610		LDA	#71	G
31EF	A7	80	01620		STA	,X+	
31F1	86	41	01630		LDA	#65	A
31F3	A7	80	01640		STA	,X+	
31F5	86	4D	01650		LDA	#77	M
31F7	A7	80	01660		STA	,X+	
			01670				
			01680	* BRANCH TO NYI			
			01690	* MESSAGE			
31F9	16	F484	01700		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

LDPOS: Load a Position (N.Y.I. - Not Yet Implemented)

```
00100 *****
00110 *
00120 * LDPOS.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * LOAD A POSITION
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57
```

1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0

26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
	01100	*LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0

		4CE0	01450	YACH	EQU	\$4CE0	
			01460				
32E0			01470		ORG	\$32E0	
			01480				
32E0	34	36	01490	LDPOS	PSHS	A,B,X,Y	
			01500				
			01510	* POINT TO THE			
			01520	* REPORTING FIELD			
32E2	8E	05EB	01530	LDX		#RPTFLD	
			01540				
			01550	* COMMAND NYI			
			01560	* MESSAGE			
32E5	86	4C	01570	LDA	#76		L
32E7	A7	80	01580	STA	,X+		
32E9	86	44	01590	LDA	#68		D
32EB	A7	80	01600	STA	,X+		
32ED	86	50	01610	LDA	#80		P
32EF	A7	80	01620	STA	,X+		
32F1	86	4F	01630	LDA	#79		O
32F3	A7	80	01640	STA	,X+		
32F5	86	53	01650	LDA	#83		S
32F7	A7	80	01660	STA	,X+		
			01670				
			01680	* BRANCH TO NYI			
			01690	* MESSAGE			
32F9	16	F384	01700	LBRA		NYIMSG	
		0000	32767	END			

00000 TOTAL ERRORS

=====

NEW: Sets a New Game Position Transfer Command

```
00100 *****
00110 *
00120 * NEW.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * SETS A NEW GAME
00160 * POSITION
00170 * TRANSFER COMMAND
00180 *
00190 *****
00200
          33F0 00210 NEWPS   EQU      $33F0
00220
33E0          00230          ORG      $33E0
00240
00250 * BRANCH TO THE
00260 * ACTION COMMAND
33E0 16      000D 00270          LBRA   NEWPS
00280
          0000 32767          END
```

00000 TOTAL ERRORS

=====

NEWPS: Sets a New Game Position Action Command

```

00100 *****
00110 *
00120 * NEWPS.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * SETS A NEW GAME
00160 * POSITION
00170 * ACTION COMMAND
00180 *
00190 *****
00200
00210 * LOW RAM CURSOR ADDRESS
0088 00220 CURPOS EQU $0088
00230
00240 * SCREEN ADDRESSES
00250 * START OF VIDRAM
0400 00260 VIDRAM EQU $0400
00270
00280 * COMMAND PROMPT
00290 * SCREEN LOCATION
05AE 00300 CMDPMT EQU $05AE
00310
00320 * SCREEN REPORTING FIELD
05EB 00330 RPTFLD EQU $05EB
00340
00350 * ONE BYTE PAST THE
00360 * END OF VIDRAM
0600 00370 VIDEND EQU $0600
00380
00390 * ML FOUNDATION
00400 * CORE ADDRESSES
1C00 00410 REGXFR EQU $1C00
1C0E 00420 VIDCLS EQU $1C0E
1C1F 00430 PUTCHR EQU $1C1F
1CD2 00440 GETCHR EQU $1CD2
1CD5 00450 PUTBYT EQU $1CD5
1D17 00460 SCROLL EQU $1D17
1D36 00470 PUTCHA EQU $1D36
1D57 00480 PUTBYA EQU $1D57

```

1D8C	00490	CRLF	EQU	\$1D8C
1DBD	00500	PK2PRT	EQU	\$1DBD
1DDA	00510	PRT2PK	EQU	\$1DDA
1DF7	00520	POLCAT	EQU	\$1DF7
1E06	00530	PUTWRA	EQU	\$1E06
1E11	00540	PUTWRD	EQU	\$1E11
1E1C	00550	BKSPCE	EQU	\$1E1C
1E34	00560	DU1616	EQU	\$1E34
1E56	00570	FUDWRD	EQU	\$1E56
1E70	00580	PTDWRD	EQU	\$1E70
	00590			
	00600	* YACH ADDRESSES		
	00610	* SYSTEM FLAGS		
2000	00620	FLAGS	EQU	\$2000
2000	00630	SCFLAG	EQU	\$2000
2001	00640	STFLAG	EQU	\$2001
2002	00650	BFFLAG	EQU	\$2002
	00660			
	00670	* SYSTEM TEMPORARY		
	00680	* VARIABLES		
2040	00690	TMPVAR	EQU	\$2040
2040	00700	CMDCNT	EQU	\$2040
2041	00710	CMDVAR	EQU	\$2041
2046	00720	CMDFRF	EQU	\$2046
2047	00730	CMDFRR	EQU	\$2047
2048	00740	CMDTOF	EQU	\$2048
2049	00750	CMDTOR	EQU	\$2049
204A	00760	CMDFRI	EQU	\$204A
204B	00770	CMDFRJ	EQU	\$204B
204C	00780	CMDTOI	EQU	\$204C
204D	00790	CMDTOJ	EQU	\$204D
204E	00800	MCHVAR	EQU	\$204E
	00810			
	00820	* SCREEN MANAGEMENT		
20C0	00830	CIDGEN	EQU	\$20C0
24C0	00840	B2SARY	EQU	\$24C0
2540	00850	B2DARY	EQU	\$2540
2580	00860	B2LARY	EQU	\$2580
25C0	00870	BSCARY	EQU	\$25C0
	00880			
	00890	* GENERAL REPORTING		
2600	00900	NOTREC	EQU	\$2600
2680	00910	NYIMSG	EQU	\$2680
	00920			
	00930	* GENERAL COMMANDS		
26C0	00940	ZERO2	EQU	\$26C0
26D0	00950	ZERO3	EQU	\$26D0

26E0	00960	ANLYZ	EQU	\$26E0
27E0	00970	ASKFR	EQU	\$27E0
28E0	00980	CLRBD	EQU	\$28E0
29E0	00990	CKMOV	EQU	\$29E0
2AE0	01000	CKPOS	EQU	\$2AE0
2BE0	01010	CSTLK	EQU	\$2BE0
2CE0	01020	CSTLQ	EQU	\$2CE0
2DE0	01030	DSPBD	EQU	\$2DE0
2EE0	01040	FLIP	EQU	\$2EE0
2FE0	01050	GSTRT	EQU	\$2FE0
30E0	01060	GSTOP	EQU	\$30E0
31E0	01070	LDGAM	EQU	\$31E0
32E0	01080	LDPOS	EQU	\$32E0
33E0	01090	NEW	EQU	\$33E0
	01100	*NEWPS	EQU	\$33F0
34F0	01110	LTRO2	EQU	\$34F0
3500	01120	LTRO3	EQU	\$3500
3510	01130	QUIT	EQU	\$3510
3520	01140	SCOFF	EQU	\$3520
3620	01150	SCON	EQU	\$3620
3720	01160	START	EQU	\$3720
3730	01170	STOP	EQU	\$3730
3740	01180	SVGAM	EQU	\$3740
3840	01190	SVPOS	EQU	\$3840
3940	01200	TKBAK	EQU	\$3940
	01210			
	01220	* MOVEMENT COMMANDS		
	01230	* AND CALCULATIONS		
3A40	01240	FILCVT	EQU	\$3A40
3A60	01250	RNKCVT	EQU	\$3A60
3A80	01260	FRE0	EQU	\$3A80
3A90	01270	FRECP	EQU	\$3A90
3AA0	01280	FRFR	EQU	\$3AA0
3EA0	01290	FRFRE	EQU	\$3EA0
3FA0	01300	FRFRP	EQU	\$3FA0
40A0	01310	FRQ0	EQU	\$40A0
41A0	01320	FRQCP	EQU	\$41A0
	01330			
	01340	* COMMAND MANAGEMENT		
42A0	01350	GETCMD	EQU	\$42A0
43A0	01360	MATCH3	EQU	\$43A0
4420	01370	MATCH4	EQU	\$4420
44A0	01380	XLTCMD	EQU	\$44A0
48A0	01390	XLTCM2	EQU	\$48A0
	01400			
	01410	* YACH SUPERVISORY		
4CA0	01420	GMLOOP	EQU	\$4CA0

		4CE0	01430	YACH	EQU	\$4CE0	
			01440				
33F0			01450		ORG	\$33F0	
			01460				
33F0	34	12	01470	NEWPOS	PSHS	A,X	
			01480				
			01490	* CLEAR THE BOARD			
33F2	17	F4EB	01500		LBSR	CLRBD	
			01510				
			01520	* POINT TO THE			
			01530	* BOARD SURFACE CONTENTS			
			01540	* ARRAY			
33F5	8E	25C0	01550		LDX	#BSCARY	
			01560				
			01570	* SET THE FIRST RANK			
33F8	86	52	01580		LDA	#82	WR
33FA	A7	80	01590		STA	,X+	
33FC	86	4E	01600		LDA	#78	WN
33FE	A7	80	01610		STA	,X+	
3400	86	42	01620		LDA	#66	WB
3402	A7	80	01630		STA	,X+	
3404	86	51	01640		LDA	#81	WQ
3406	A7	80	01650		STA	,X+	
3408	86	4B	01660		LDA	#75	WK
340A	A7	80	01670		STA	,X+	
340C	86	42	01680		LDA	#66	WB
340E	A7	80	01690		STA	,X+	
3410	86	4E	01700		LDA	#78	WN
3412	A7	80	01710		STA	,X+	
3414	86	52	01720		LDA	#82	WR
3416	A7	80	01730		STA	,X+	
			01740				
			01750	* SET THE SECOND RANK			
3418	86	50	01760		LDA	#80	WP
341A	A7	80	01770		STA	,X+	
341C	A7	80	01780		STA	,X+	
341E	A7	80	01790		STA	,X+	
3420	A7	80	01800		STA	,X+	
3422	A7	80	01810		STA	,X+	
3424	A7	80	01820		STA	,X+	
3426	A7	80	01830		STA	,X+	
3428	A7	80	01840		STA	,X+	
			01850				
			01860	* POINT TO THE			
			01870	* SEVENTH RANK			
342A	30	88 20	01880		LEAX	32,X	
			01890				

```

01900 * SET THE SEVENTH RANK
342D 86 10 01910 LDA #16 BP
342F A7 80 01920 STA ,X+
3431 A7 80 01930 STA ,X+
3433 A7 80 01940 STA ,X+
3435 A7 80 01950 STA ,X+
3437 A7 80 01960 STA ,X+
3439 A7 80 01970 STA ,X+
343B A7 80 01980 STA ,X+
343D A7 80 01990 STA ,X+
02000
02010 * SET THE EIGHTH RANK
343F 86 12 02020 LDA #18 BR
3441 A7 80 02030 STA ,X+
3443 86 0E 02040 LDA #14 BN
3445 A7 80 02050 STA ,X+
3447 86 02 02060 LDA #2 BB
3449 A7 80 02070 STA ,X+
344B 86 11 02080 LDA #17 BQ
344D A7 80 02090 STA ,X+
344F 86 0B 02100 LDA #11 BK
3451 A7 80 02110 STA ,X+
3453 86 02 02120 LDA #2 BB
3455 A7 80 02130 STA ,X+
3457 86 0E 02140 LDA #14 BN
3459 A7 80 02150 STA ,X+
345B 86 12 02160 LDA #18 BR
345D A7 80 02170 STA ,X+
02180
02190 * COPY THE BOARD
02200 * TO THE SCREEN
345F 17 F97E 02210 LBSR DSPBD
02220
02230 * EXIT
3462 35 12 02240 PULS A,X
3464 39 0000 02250 RTS
32767 END

```

00000 TOTAL ERRORS

=====

LTRO2: Castle Kingside Actual Command = O-O Transfer Command

```
00100 *****
00110 *
00120 * LTRO2.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * CASTLE KINGSIDE
00170 * ACTUAL COMMAND = O-O
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
          2BE0 00220 CSTLK   EQU   $2BE0
00230
34F0          00240          ORG   $34F0
00250
00260 * BRANCH TO THE
00270 * ACTION COMMAND
34F0 16      F6ED 00280          LBRA  CSTLK
00290
          0000 32767          END
```

00000 TOTAL ERRORS

=====

LTRO3: Castle Queenside

Actual Command = O-O-O

Transfer Command

```
00100 *****
00110 *
00120 * LTRO3.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * CASTLE QUEENSIDE
00170 * ACTUAL COMMAND = O-O-O
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
          2CE0 00220 CSTLQ   EQU   $2CE0
3500          00230
          00240          ORG   $3500
          00250
          00260 * BRANCH TO THE
          00270 * ACTION COMMAND
3500 16   F7DD 00280          LBRA   CSTLQ
          00290
          0000 32767          END

00000 TOTAL ERRORS
```

=====

QUIT: Quit the Game and Return to the Command Prompt

```
00100 *****
00110 *
00120 * QUIT.ASM
00130 * MDJ 2021/09/21
00140 *
00150 * QUIT THE GAME AND
00160 * RETURN TO THE
00170 * COMMAND PROMPT
00180 *
00190 * I.E. DO WARM START
00200 *
00210 *****
00220
3510 00230          ORG          $3510
00240
00250 * UNCOMMENT THE
00260 * FOLLOWING CODE LINES
00270 * AND COMMENT-OUT THE
00280 * ORIGINAL CODE LINE
00290 * WHEN REBUILDING AND
00300 * RENUMBERING AFTER
00310 * COMPLETION OF THE
00320 * MLF INTERRUPTS
00330 * ADDITION PROJECT
00340
00350 * SET RAMROM MODE
00360 * AND QUIT
00370
00380 * QUIT      STA      $FFDE
00390 *           JMP      [$FFFE]
00400
3510 6E      9F FFFE 00410 QUIT      JMP      [$FFFE]
          0000      32767  END
```

00000 TOTAL ERRORS

=====

SCOFF: Turns Screen Colors Off

```

00100 *****
00110 *
00120 * SCOFF.ASM
00130 * MDJ 2021/09/18
00140 *
00150 * TURNS SCREEN
00160 * COLORS OFF
00170 *
00180 * I.E. REMOVES
00190 * THE ASTERISKS
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57
1D8C 00510 CRLF EQU $1D8C
1DBD 00520 PK2PRT EQU $1DBD

```

1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0
26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0

28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
	01160	*SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0
4CE0	01450	YACH	EQU	\$4CE0
	01460			

3520			01470	ORG	\$3520
			01480		
3520	34	52	01490	SCOFF PSHS	A,X,U
			01500		
			01510	* POINT TO B2D ARRAY	
3522	CE	2540	01520	LDU	#B2DARY
			01530		
			01540	* GET A DARK SQUARE	
			01550	* ADDRESS	
3525	AE	C1	01560	L0001 LDX	,U++
			01570		
			01580	* GET THE SCREEN VALUE	
			01590	* AT THAT ADDRESS	
3527	A6	84	01600	LDA	,X
			01610		
			01620	* IS THE SQUARE EMPTY,	
			01630	* I.E. IS IT AN ASTERISK?	
3529	81	6A	01640	CMPA	#106
			01650		
			01660	* GO IF NO	
352B	26	F8	01670	BNE	L0001
			01680		
			01690	* TURN OFF THE SCREEN	
			01700	* COLOR, I.E. REPLACE THE	
			01710	* ASTERISK WITH A SPACE	
352D	86	60	01720	LDA	#96
352F	A7	84	01730	STA	,X
			01740		
			01750	* ARE WE DONE, I.E.	
			01760	* ARE WE PAST THE	
			01770	* LAST DARK SQUARE?	
3531	1183	257E	01780	CMPU	#B2DARY+62
			01790		
			01800	* GO IF YES	
3535	22	02	01810	BHI	L0002
			01820		
			01830	* GO DO NEXT SQUARE	
3537	20	EC	01840	BRA	L0001
			01850		
			01860	* CLEAR SCREEN COLOR FLAG	
3539	7F	2000	01870	L0002 CLR	SCFLAG
			01880		
			01890	* EXIT	
353C	35	52	01900	PULS	A,X,U
353E	39		01910	RTS	
		0000	32767	END	
00000	TOTAL ERRORS				

SCON: Turns Screen Colors On

```
00100 *****
00110 *
00120 * SCON.ASM
00130 * MDJ 2021/09/18
00140 *
00150 * TURNS SCREEN
00160 * COLORS ON
00170 *
00180 * I.E. ADDS
00190 * THE ASTERISKS
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57
1D8C 00510 CRLF EQU $1D8C
1DBD 00520 PK2PRT EQU $1DBD
```

1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0
26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0

28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
	01170	*SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0
4CE0	01450	YACH	EQU	\$4CE0
	01460			

3620			01470	ORG	\$3620
			01480		
3620	34	52	01490	SCON PSHS	A,X,U
			01500		
			01510	* POINT TO B2D ARRAY	
3622	CE	2540	01520	LDU	#B2DARY
			01530		
			01540	* GET A DARK SQUARE	
			01550	* ADDRESS	
3625	AE	C1	01560	L0001 LDX	,U++
			01570		
			01580	* GET THE SCREEN VALUE	
			01590	* AT THAT ADDRESS	
3627	A6	84	01600	LDA	,X
			01610		
			01620	* IS THE SQUARE EMPTY,	
			01630	* I.E. IS IT A SPACE?	
3629	81	60	01640	CMPA	#96
			01650		
			01660	* GO IF NO	
362B	26	F8	01670	BNE	L0001
			01680		
			01690	* TURN ON THE SCREEN	
			01700	* COLOR, I.E. REPLACE THE	
			01710	* SPACE WITH AN ASTERISK	
362D	86	6A	01720	LDA	#106
362F	A7	84	01730	STA	,X
			01740		
			01750	* ARE WE DONE, I.E.	
			01760	* ARE WE PAST THE	
			01770	* LAST DARK SQUARE?	
3631	1183	257E	01780	CMPU	#B2DARY+62
			01790		
			01800	* GO IF YES	
3635	22	02	01810	BHI	L0002
			01820		
			01830	* GO DO NEXT SQUARE	
3637	20	EC	01840	BRA	L0001
			01850		
			01860	* SET SCREEN COLOR FLAG	
3639	86	01	01870	L0002 LDA	#1
363B	B7	2000	01880	STA	SCFLAG
			01890		
			01900	* EXIT	
363E	35	52	01910	PULS	A,X,U
3640	39		01920	RTS	
		0000	32767	END	

00000 TOTAL ERRORS

The Assembly Language Test Routine:

```
00100 *****
00110 *
00120 * CTST0003.ASM
00130 * MDJ 2021/09/18
00140 *
00150 * TEST OF
00160 * SCOFF.ASM AND
00170 * SCON.ASM
00180 *
00190 *****
00200
00210 * LOW RAM CURSOR ADDRESS
0088 00220 CURPOS EQU $0088
00230
00240 * SCREEN ADDRESSES
00250 * START OF VIDRAM
0400 00260 VIDRAM EQU $0400
00270
00280 * ONE BYTE PAST THE
00290 * END OF VIDRAM
0600 00300 VIDEND EQU $0600
00310
00320 * ML FOUNDATION
00330 * CORE ADDRESSES
1C00 00340 REGXFR EQU $1C00
1C0E 00350 VIDCLS EQU $1C0E
1C1F 00360 PUTCHR EQU $1C1F
1CD2 00370 GETCHR EQU $1CD2
1CD5 00380 PUTBYT EQU $1CD5
1D17 00390 SCROLL EQU $1D17
1D36 00400 PUTCHA EQU $1D36
1D57 00410 PUTBYA EQU $1D57
1D8C 00420 CRLF EQU $1D8C
1DBD 00430 PK2PRT EQU $1DBD
1DDA 00440 PRT2PK EQU $1DDA
1DF7 00450 POLCAT EQU $1DF7
1E06 00460 PUTWRA EQU $1E06
1E11 00470 PUTWRD EQU $1E11
1E1C 00480 BKSPCE EQU $1E1C
1E34 00490 DU1616 EQU $1E34
```

	1E56		00500	FUDWRD	EQU	\$1E56
	1E70		00510	PTDWRD	EQU	\$1E70
	1E8A		00520	CIDGEN	EQU	\$1E8A
	20A5		00530	B2SARY	EQU	\$20A5
	2125		00540	B2DARY	EQU	\$2125
	2165		00550	B2LARY	EQU	\$2165
	21A5		00560	BSCARY	EQU	\$21A5
	21E5		00570	FLAGS	EQU	\$21E5
	21E5		00580	SCFLAG	EQU	\$21E5
			00590			
			00600	* PROGRAMS BEING TESTED		
			00610	* EQUATE - CHANGE AS		
			00620	* NEEDED		
	2225		00630	SCOFF	EQU	\$2225
	2244		00640	SCON	EQU	\$2244
			00650			
			00660	* TEST ROUTINE		
			00670	* PROGRAM ORIGIN:		
			00680	* (CHANGE AS NEEDED)		
7000			00690	ORG		\$7000
			00700			
			00710	* DISPLAY THE SCREEN		
7000	17	AE87	00720	LBSR	CIDGEN	
			00730			
			00740	* WAIT FOR A KEY PRESS		
7003	17	ADF1	00750	L0001	LBSR	POLCAT
7006	27	FB	00760	BEQ	L0001	
			00770			
			00780	* TURN SCREEN COLORS OFF		
7008	17	B21A	00790	LBSR	SCOFF	
			00800			
			00810	* WAIT FOR A KEY PRESS		
700B	17	ADE9	00820	L0002	LBSR	POLCAT
700E	27	FB	00830	BEQ	L0002	
			00840			
			00850	* TURN SCREEN COLORS ON		
7010	17	B231	00860	LBSR	SCON	
			00870			
			00880	* ENDLESS LOOP		
7013	20	EE	00890	BRA	L0001	
			00900			
			00910	* EXIT		
7015	39		00920	RTS		
	0000		32767	END		

00000 TOTAL ERRORS

The BASIC Language Control Program:

```
1000 '*****
1010 '*
1020 '* CTST0003.BAS
1030 '* MDJ 2021/09/18
1040 '*
1050 '* SCOFF AND SCON TEST
1060 '*
1070 '*****
1080 '
1090 'SETUP MEMORY
1100 PCLEAR 1
1110 CLEAR 200, &H1C00
1120 '
1130 CLS
1140 PRINT @ 105, "Y. A. C. H."
1150 PRINT @ 162, "(YET ANOTHER CHESS HANDLER)"
1160 PRINT @ 201, "VERSION 0.0.1"
1170 PRINT @ 304, "BY"
1180 PRINT @ 328, "M. DAVID JOHNSON"
1190 PRINT @ 360, "INFO@BDS-SOFT.COM"
1200 PRINT @ 395, "2021/09/29"
1210 PRINT @ 490, "PLEASE WAIT"
1220 '
1230 'LOAD ML ROUTINES
1240 LOADM "REGXFR.BIN"
1250 LOADM "VIDCLS.BIN"
1260 LOADM "PUTCHR.BIN"
1270 LOADM "GETCHR.BIN"
1280 LOADM "PUTBYT.BIN"
1290 LOADM "SCROLL.BIN"
1300 LOADM "PUTCHA.BIN"
1310 LOADM "PUTBYA.BIN"
1320 LOADM "CRLF.BIN"
1330 LOADM "PK2PRT.BIN"
1340 LOADM "PRT2PK.BIN"
1350 LOADM "POLCAT.BIN"
1360 LOADM "CIDGEN.BIN"
1370 LOADM "B2SARY.BIN"
1380 LOADM "B2DARY.BIN"
1390 LOADM "B2LARY.BIN"
1400 LOADM "SCOFF.BIN"
1410 LOADM "SCON.BIN"
1420 LOADM "CTST0003.BIN"
1430 '
1440 PRINT @ 451, "PRESS ANY KEY TO CONTINUE"
```

```
1450 A$ = INKEY$
1460 IF A$ = "" GOTO 1450
1470 '
1480 'GO DO THE TEST
1490 EXEC &H7000
1500 '
32767 END
```

=====

START: Restart Accepting Commands After a STOP Transfer Command

```
00100 *****
00110 *
00120 * START.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * RESTART ACCEPTING
00160 * COMMANDS
00170 * AFTER A STOP
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
          2FE0 00220 GSTRT   EQU      $2FE0
3720          00230
          00240          ORG      $3720
          00250
          00260 * BRANCH TO THE
          00270 * ACTION COMMAND
3720 16      F8BD 00280          LBRA    GSTRT
          00290
          0000 32767          END
```

00000 TOTAL ERRORS

=====

STOP: Ignore All Subsequent Commands Except START Transfer Command

```
00100 *****
00110 *
00120 * STOP.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * IGNORE ALL
00160 * SUBSEQUENT
00170 * COMMANDS
00180 * EXCEPT START
00190 * TRANSFER COMMAND
00200 *
00210 *****
00220
          30E0 00230 GSTOP EQU $30E0
3730          00240
          00250          ORG $3730
          00260
          00270 * BRANCH TO THE
          00280 * ACTION COMMAND
3730 16  F9AD 00290          LBRA GSTOP
          00300
          0000 32767          END

00000 TOTAL ERRORS
```

=====

SVGAM: Save the Game (N.Y.I. - Not Yet Implemented)

```
00100 *****
00110 *
00120 * SVGAM.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * SAVE THE GAME
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57
1D8C 00510 CRLF EQU $1D8C
```

1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0
26E0	00980	ANLYZ	EQU	\$26E0

27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
	01200	*SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0
4CE0	01450	YACH	EQU	\$4CE0

			01460			
3740			01470	ORG	\$3740	
			01480			
3740	34	36	01490	SVGAM	PSHS	A,B,X,Y
			01500			
			01510	* POINT TO THE		
			01520	* REPORTING FIELD		
3742	8E	05EB	01530	LDX	#RPTFLD	
			01540			
			01550	* COMMAND NYI		
			01560	* MESSAGE		
3745	86	53	01570	LDA	#83	S
3747	A7	80	01580	STA	,X+	
3749	86	56	01590	LDA	#86	V
374B	A7	80	01600	STA	,X+	
374D	86	47	01610	LDA	#71	G
374F	A7	80	01620	STA	,X+	
3751	86	41	01630	LDA	#65	A
3753	A7	80	01640	STA	,X+	
3755	86	4D	01650	LDA	#77	M
3757	A7	80	01660	STA	,X+	
			01670			
			01680	* BRANCH TO NYI		
			01690	* MESSAGE		
3759	16	EF24	01700	LBRA	NYIMSG	
		0000	32767	END		

00000 TOTAL ERRORS

=====

SVPOS: Save the Position (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * SVPOS.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * SAVE THE POSITION
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57

```

1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0

26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
	01210	*SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0

		4CE0	01450	YACH	EQU	\$4CE0	
			01460				
3840			01470		ORG	\$3840	
			01480				
3840	34	36	01490	SVPOS	PSHS	A,B,X,Y	
			01500				
			01510	* POINT TO THE			
			01520	* REPORTING FIELD			
3842	8E	05EB	01530		LDX	#RPTFLD	
			01540				
			01550	* COMMAND NYI			
			01560	* MESSAGE			
3845	86	53	01570		LDA	#83	S
3847	A7	80	01580		STA	,X+	
3849	86	56	01590		LDA	#86	V
384B	A7	80	01600		STA	,X+	
384D	86	50	01610		LDA	#80	P
384F	A7	80	01620		STA	,X+	
3851	86	4F	01630		LDA	#79	O
3853	A7	80	01640		STA	,X+	
3855	86	53	01650		LDA	#83	S
3857	A7	80	01660		STA	,X+	
			01670				
			01680	* BRANCH TO NYI			
			01690	* MESSAGE			
3859	16	EE24	01700		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

TKBAK: Take Back the Last Move (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * TKBAK.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * TAKE BACK THE LAST MOVE
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400
00410 * ML FOUNDATION
00420 * CORE ADDRESSES
1C00 00430 REGXFR EQU $1C00
1C0E 00440 VIDCLS EQU $1C0E
1C1F 00450 PUTCHR EQU $1C1F
1CD2 00460 GETCHR EQU $1CD2
1CD5 00470 PUTBYT EQU $1CD5
1D17 00480 SCROLL EQU $1D17
1D36 00490 PUTCHA EQU $1D36
1D57 00500 PUTBYA EQU $1D57

```

1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540
2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0

26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
	01220	*TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
3AA0	01300	FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0
	01350			
	01360	* COMMAND MANAGEMENT		
42A0	01370	GETCMD	EQU	\$42A0
43A0	01380	MATCH3	EQU	\$43A0
4420	01390	MATCH4	EQU	\$4420
44A0	01400	XLTCMD	EQU	\$44A0
48A0	01410	XLTCM2	EQU	\$48A0
	01420			
	01430	* YACH SUPERVISORY		
4CA0	01440	GMLOOP	EQU	\$4CA0

	4CE0		01450	YACH	EQU	\$4CE0	
			01460				
3940			01470		ORG	\$3940	
			01480				
3940	34	36	01490	TKBAK	PSHS	A,B,X,Y	
			01500				
			01510	* POINT TO THE			
			01520	* REPORTING FIELD			
3942	8E	05EB	01530		LDX	#RPTFLD	
			01540				
			01550	* COMMAND NYI			
			01560	* MESSAGE			
3945	86	54	01570		LDA	#84	T
3947	A7	80	01580		STA	,X+	
3949	86	4B	01590		LDA	#75	K
394B	A7	80	01600		STA	,X+	
394D	86	42	01610		LDA	#66	B
394F	A7	80	01620		STA	,X+	
3951	86	41	01630		LDA	#65	A
3953	A7	80	01640		STA	,X+	
3955	86	4B	01650		LDA	#75	K
3957	A7	80	01660		STA	,X+	
			01670				
			01680	* BRANCH TO NYI			
			01690	* MESSAGE			
3959	16	ED24	01700		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

MAKEGC2: Make the General Commands Group Two GC2.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEGC2.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "LDGAM.BIN"
1130 LOADM "LDPOS.BIN"
1140 LOADM "NEW.BIN"
1150 LOADM "NEWPS.BIN"
1160 LOADM "LTRO2.BIN"
1170 LOADM "LTRO3.BIN"
1180 LOADM "QUIT.BIN"
1190 LOADM "SCOFF.BIN"
1200 LOADM "SCON.BIN"
1210 LOADM "START.BIN"
1220 LOADM "STOP.BIN"
1230 LOADM "SVGAM.BIN"
1240 LOADM "SVPOS.BIN"
1250 LOADM "TKBAK.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "GC2.BIN", &H31E0, &H3A3F, &H31E0
1330 '
32767 END
```

=====

Movement Commands And Calculations

This is Section Five: Movement Commands And Calculations

1. FILCVT: Converts a Chess Board File Character ("A" to "H") to an Array I-Value (0 to 7) i.e. an x-coordinate
2. RNKCVT: Converts a Chess Board Rank Character ("1" to "8") to an Array J-Value (0 to 7) i.e. a y-coordinate
3. FRE0: Empty the Square at F = File Character R = Rank Character Actual Command = FR=0 Transfer Command
4. FRECP: Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character Actual Command = FR=CP Transfer Command
5. FRFR: Standard Move From F = File Character R = Rank Character To F = File Character R = Rank Character (e.g. E2E4)
6. FRFRE: Standard Move Plus Capture En Passant (N.Y.I. - Not Yet Implemented)
7. FRFRP: Standard Move Plus Pawn Promotion (N.Y.I. - Not Yet Implemented)
8. FRQ0: Empty the Square at F = File Character R = Rank Character Action Command (N.Y.I. - Not Yet Implemented)
9. FRQCP: Place a C = Color (W,B) P = Piece (K,Q,R,B,N,P) at F = File Character R = Rank Character Action Command (N.Y.I. - Not Yet Implemented)

=====

FILCVT: Converts a Chess Board File Character ("A" to "H") to an Array I-Value (0 to 7) i.e. an x-coordinate

Note that the Chess Board (and thus the **BSCARY** Array) is an 8 x 8, 64-cell array, with I = x-coordinates running horizontally from left-to-right, and J = y-coordinates running vertically from bottom-to-top. This routine converts the Chess Board's traditional File (i.e. column) Letters ("A" to "H") to Array I-Values (0 to 7). For example, square "E3" converts to (4,2).

```
00100 *****
00110 *
00120 * FILCVT.ASM
00130 * MDJ 2021/09/21
00140 *
00150 * CONVERTS A CHESS BOARD
00160 * FILE CHARACTER
00170 * ("A" TO "H")
00180 * TO AN ARRAY I-VALUE
00190 * (0 TO 7)
00200 * I.E. AN X-COORDINATE
00210 *
00220 * ENTRY CONDITIONS:
00230 *   A = THE CHARACTER
00240 *
00250 * EXIT CONDITIONS:
00260 *   CC Z-BIT = 1
00270 *   A = THE I-VALUE
00280 *
00290 * ERROR CONDITIONS:
00300 *   CC Z-BIT = 0
00310 *   A = 255
00320 *
00330 *****
00340
00350 * LOW RAM CURSOR ADDRESS
0088 00360 CURPOS EQU    $0088
00370
00380 * SCREEN ADDRESSES
00390 * START OF VIDRAM
0400 00400 VIDRAM EQU    $0400
00410
00420 * COMMAND PROMPT
```

	00430	* SCREEN LOCATION		
05AE	00440	CMDPMT EQU	\$05AE	
	00450			
	00460	* SCREEN REPORTING FIELD		
05EB	00470	RPTFLD EQU	\$05EB	
	00480			
	00490	* ONE BYTE PAST THE		
	00500	* END OF VIDRAM		
0600	00510	VIDEND EQU	\$0600	
	00520			
	00530	* ML FOUNDATION		
	00540	* CORE ADDRESSES		
1C00	00550	REGXFR EQU	\$1C00	
1C0E	00560	VIDCLS EQU	\$1C0E	
1C1F	00570	PUTCHR EQU	\$1C1F	
1CD2	00580	GETCHR EQU	\$1CD2	
1CD5	00590	PUTBYT EQU	\$1CD5	
1D17	00600	SCROLL EQU	\$1D17	
1D36	00610	PUTCHA EQU	\$1D36	
1D57	00620	PUTBYA EQU	\$1D57	
1D8C	00630	CRLF EQU	\$1D8C	
1DBD	00640	PK2PRT EQU	\$1DBD	
1DDA	00650	PRT2PK EQU	\$1DDA	
1DF7	00660	POLCAT EQU	\$1DF7	
1E06	00670	PUTWRA EQU	\$1E06	
1E11	00680	PUTWRD EQU	\$1E11	
1E1C	00690	BKSPCE EQU	\$1E1C	
1E34	00700	DU1616 EQU	\$1E34	
1E56	00710	FUDWRD EQU	\$1E56	
1E70	00720	PTDWRD EQU	\$1E70	
	00730			
	00740	* YACH ADDRESSES		
	00750	* SYSTEM FLAGS		
2000	00760	FLAGS EQU	\$2000	
2000	00770	SCFLAG EQU	\$2000	
2001	00780	STFLAG EQU	\$2001	
2002	00790	BFFLAG EQU	\$2002	
	00800			
	00810	* SYSTEM TEMPORARY		
	00820	* VARIABLES		
2040	00830	TMPVAR EQU	\$2040	
2040	00840	CMDCNT EQU	\$2040	
2041	00850	CMDVAR EQU	\$2041	
2046	00860	CMDFRF EQU	\$2046	
2047	00870	CMDFRR EQU	\$2047	
2048	00880	CMDTOF EQU	\$2048	
2049	00890	CMDTOR EQU	\$2049	

204A	00900	CMDFRI	EQU	\$204A
204B	00910	CMDFRJ	EQU	\$204B
204C	00920	CMDTOI	EQU	\$204C
204D	00930	CMDTOJ	EQU	\$204D
204E	00940	MCHVAR	EQU	\$204E
	00950			
	00960	* SCREEN MANAGEMENT		
20C0	00970	CIDGEN	EQU	\$20C0
24C0	00980	B2SARY	EQU	\$24C0
2540	00990	B2DARY	EQU	\$2540
2580	01000	B2LARY	EQU	\$2580
25C0	01010	BSCARY	EQU	\$25C0
	01020			
	01030	* GENERAL REPORTING		
2600	01040	NOTREC	EQU	\$2600
2680	01050	NYIMSG	EQU	\$2680
	01060			
	01070	* GENERAL COMMANDS		
26C0	01080	ZERO2	EQU	\$26C0
26D0	01090	ZERO3	EQU	\$26D0
26E0	01100	ANLYZ	EQU	\$26E0
27E0	01110	ASKFR	EQU	\$27E0
28E0	01120	CLRBD	EQU	\$28E0
29E0	01130	CKMOV	EQU	\$29E0
2AE0	01140	CKPOS	EQU	\$2AE0
2BE0	01150	CSTLK	EQU	\$2BE0
2CE0	01160	CSTLQ	EQU	\$2CE0
2DE0	01170	DSPBD	EQU	\$2DE0
2EE0	01180	FLIP	EQU	\$2EE0
2FE0	01190	GSTRT	EQU	\$2FE0
30E0	01200	GSTOP	EQU	\$30E0
31E0	01210	LDGAM	EQU	\$31E0
32E0	01220	LDPOS	EQU	\$32E0
33E0	01230	NEW	EQU	\$33E0
33F0	01240	NEWPS	EQU	\$33F0
34F0	01250	LTRO2	EQU	\$34F0
3500	01260	LTRO3	EQU	\$3500
3510	01270	QUIT	EQU	\$3510
3520	01280	SCOFF	EQU	\$3520
3620	01290	SCON	EQU	\$3620
3720	01300	START	EQU	\$3720
3730	01310	STOP	EQU	\$3730
3740	01320	SVGAM	EQU	\$3740
3840	01330	SVPOS	EQU	\$3840
3940	01340	TKBAK	EQU	\$3940
	01350			
	01360	* MOVEMENT COMMANDS		

```

01370 * AND CALCULATIONS
01380 *FILCVT EQU $3A40
3A60 01390 RNKCVT EQU $3A60
3A80 01400 FRE0 EQU $3A80
3A90 01410 FRECP EQU $3A90
3AA0 01420 FRFR EQU $3AA0
3EA0 01430 FRFRE EQU $3EA0
3FA0 01440 FRFRP EQU $3FA0
40A0 01450 FRQ0 EQU $40A0
41A0 01460 FRQCP EQU $41A0
01470
01480 * COMMAND MANAGEMENT
42A0 01490 GETCMD EQU $42A0
43A0 01500 MATCH3 EQU $43A0
4420 01510 MATCH4 EQU $4420
44A0 01520 XLTCMD EQU $44A0
48A0 01530 XLTCM2 EQU $48A0
01540
01550 * YACH SUPERVISORY
4CA0 01560 GMLOOP EQU $4CA0
4CE0 01570 YACH EQU $4CE0
01580
3A40 01590 ORG $3A40
01600
01610 * VALID FILE
01620 * CHARACTERS
01630 * = "A" THROUGH "H"
01640 * = CODES 65 TO 72
01650
01660 * DO THE CONVERSION
3A40 81 41 01670 FILCVT CMPA #65 A
3A42 25 04 01680 BLO L0001
3A44 81 48 01690 CMPA #72 H
3A46 23 06 01700 BLS L0002
01710
01720 * ERROR: A = 255
01730 * CLEAR THE Z-BIT
3A48 86 FF 01740 L0001 LDA #255
3A4A 1C FB 01750 ANDCC #$FB
3A4C 20 04 01760 BRA L0003
01770
01780 * VALID: I = CHAR - 65
01790 * SET THE Z-BIT
3A4E 80 41 01800 L0002 SUBA #65
3A50 1A 04 01810 ORCC #$04
01820
01830 * EXIT

```


3A52 39 01840 L0003 RTS
 32767 END

00000 TOTAL ERRORS

=====

RNKCVT: Converts a Chess Board Rank Character ("1" to "8") to an Array J-Value (0 to 7) i.e. a y-coordinate

Note that the Chess Board (and thus the **BSCARY** Array) is an 8 x 8, 64-cell array, with I = x-coordinates running horizontally from left-to-right, and J = y-coordinates running vertically from bottom-to-top. This routine converts the Chess Board's traditional Rank (i.e. row) Characters ("1" to "8") to Array J-Values (0 to 7). For example, square "G7" converts to (6,6).

```
00100 *****
00110 *
00120 * RNKCVT.ASM
00130 * MDJ 2021/09/21
00140 *
00150 * CONVERTS A CHESS BOARD
00160 * RANK CHARACTER
00170 * ("1" TO "8")
00180 * TO AN ARRAY J-VALUE
00190 * (0 TO 7)
00200 * I.E. A Y-COORDINATE
00210 *
00220 * ENTRY CONDITIONS:
00230 *   A = THE CHARACTER
00240 *   (PRINT MECHANISM)
00250 *
00260 * EXIT CONDITIONS:
00270 *   CC Z-BIT = 1
00280 *   A = THE J-VALUE
00290 *
00300 * ERROR CONDITIONS:
00310 *   CC Z-BIT = 0
00320 *   A = 255
00330 *
00340 *****
00350
00360 * LOW RAM CURSOR ADDRESS
0088 00370 CURPOS EQU $0088
00380
00390 * SCREEN ADDRESSES
00400 * START OF VIDRAM
0400 00410 VIDRAM EQU $0400
00420
```

```

00430 * COMMAND PROMPT
00440 * SCREEN LOCATION
05AE 00450 CMDPMT EQU      $05AE
00460
00470 * SCREEN REPORTING FIELD
05EB 00480 RPTFLD EQU      $05EB
00490
00500 * ONE BYTE PAST THE
00510 * END OF VIDRAM
0600 00520 VIDEND EQU      $0600
00530
00540 * ML FOUNDATION
00550 * CORE ADDRESSES
1C00 00560 REGXFR EQU      $1C00
1C0E 00570 VIDCLS EQU      $1C0E
1C1F 00580 PUTCHR EQU      $1C1F
1CD2 00590 GETCHR EQU      $1CD2
1CD5 00600 PUTBYT EQU      $1CD5
1D17 00610 SCROLL EQU      $1D17
1D36 00620 PUTCHA EQU      $1D36
1D57 00630 PUTBYA EQU      $1D57
1D8C 00640 CRLF EQU      $1D8C
1DBD 00650 PK2PRT EQU      $1DBD
1DDA 00660 PRT2PK EQU      $1DDA
1DF7 00670 POLCAT EQU      $1DF7
1E06 00680 PUTWRA EQU      $1E06
1E11 00690 PUTWRD EQU      $1E11
1E1C 00700 BKSPCE EQU      $1E1C
1E34 00710 DU1616 EQU      $1E34
1E56 00720 FUDWRD EQU      $1E56
1E70 00730 PTDWRD EQU      $1E70
00740
00750 * YACH ADDRESSES
00760 * SYSTEM FLAGS
2000 00770 FLAGS EQU      $2000
2000 00780 SCFLAG EQU      $2000
2001 00790 STFLAG EQU      $2001
2002 00800 BFFLAG EQU      $2002
00810
00820 * SYSTEM TEMPORARY
00830 * VARIABLES
2040 00840 TMPVAR EQU      $2040
2040 00850 CMDCNT EQU      $2040
2041 00860 CMDVAR EQU      $2041
2046 00870 CMDFRF EQU      $2046
2047 00880 CMDFRR EQU      $2047
2048 00890 CMDTOF EQU      $2048

```

2049	00900	CMDTOR	EQU	\$2049
204A	00910	CMDFRI	EQU	\$204A
204B	00920	CMDFRJ	EQU	\$204B
204C	00930	CMDTOI	EQU	\$204C
204D	00940	CMDTOJ	EQU	\$204D
204E	00950	MCHVAR	EQU	\$204E
	00960			
	00970	* SCREEN MANAGEMENT		
20C0	00980	CIDGEN	EQU	\$20C0
24C0	00990	B2SARY	EQU	\$24C0
2540	01000	B2DARY	EQU	\$2540
2580	01010	B2LARY	EQU	\$2580
25C0	01020	BSCARY	EQU	\$25C0
	01030			
	01040	* GENERAL REPORTING		
2600	01050	NOTREC	EQU	\$2600
2680	01060	NYIMSG	EQU	\$2680
	01070			
	01080	* GENERAL COMMANDS		
26C0	01090	ZERO2	EQU	\$26C0
26D0	01100	ZERO3	EQU	\$26D0
26E0	01110	ANLYZ	EQU	\$26E0
27E0	01120	ASKFR	EQU	\$27E0
28E0	01130	CLRBD	EQU	\$28E0
29E0	01140	CKMOV	EQU	\$29E0
2AE0	01150	CKPOS	EQU	\$2AE0
2BE0	01160	CSTLK	EQU	\$2BE0
2CE0	01170	CSTLQ	EQU	\$2CE0
2DE0	01180	DSPBD	EQU	\$2DE0
2EE0	01190	FLIP	EQU	\$2EE0
2FE0	01200	GSTRT	EQU	\$2FE0
30E0	01210	GSTOP	EQU	\$30E0
31E0	01220	LDGAM	EQU	\$31E0
32E0	01230	LDPOS	EQU	\$32E0
33E0	01240	NEW	EQU	\$33E0
33F0	01250	NEWPS	EQU	\$33F0
34F0	01260	LTRO2	EQU	\$34F0
3500	01270	LTRO3	EQU	\$3500
3510	01280	QUIT	EQU	\$3510
3520	01290	SCOFF	EQU	\$3520
3620	01300	SCON	EQU	\$3620
3720	01310	START	EQU	\$3720
3730	01320	STOP	EQU	\$3730
3740	01330	SVGAM	EQU	\$3740
3840	01340	SVPOS	EQU	\$3840
3940	01350	TKBAK	EQU	\$3940
	01360			

```

01370 * MOVEMENT COMMANDS
01380 * AND CALCULATIONS
3A40 01390 FILCVT EQU $3A40
01400 *RNKCVT EQU $3A60
3A80 01410 FRE0 EQU $3A80
3A90 01420 FRECP EQU $3A90
3AA0 01430 FRFR EQU $3AA0
3EA0 01440 FRFRE EQU $3EA0
3FA0 01450 FRFRP EQU $3FA0
40A0 01460 FRQ0 EQU $40A0
41A0 01470 FRQCP EQU $41A0
01480
01490 * COMMAND MANAGEMENT
42A0 01500 GETCMD EQU $42A0
43A0 01510 MATCH3 EQU $43A0
4420 01520 MATCH4 EQU $4420
44A0 01530 XLTCMD EQU $44A0
48A0 01540 XLTCM2 EQU $48A0
01550
01560 * YACH SUPERVISORY
4CA0 01570 GMLOOP EQU $4CA0
4CE0 01580 YACH EQU $4CE0
01590
3A60 01600 ORG $3A60
01610
01620 * VALID FILE
01630 * CHARACTERS
01640 * (INPUT -->
01650 * PRINT MECHANISM)
01660 * = "1" THROUGH "8"
01670 * = CODES 49 TO 56
01680
01690 * DO THE CONVERSION
3A60 81 31 01700 RNKCVT CMPA #49 1
3A62 25 04 01710 BLO L0001
3A64 81 38 01720 CMPA #56 8
3A66 23 06 01730 BLS L0002
01740
01750 * ERROR: A = 255
01760 * CLEAR THE Z-BIT
3A68 86 FF 01770 L0001 LDA #255
3A6A 1C FB 01780 ANDCC #$FB
3A6C 20 04 01790 BRA L0003
01800
01810 * VALID: J = CHAR - 49
01820 * SET THE Z-BIT
3A6E 80 31 01830 L0002 SUBA #49

```

```
3A70 1A 04 01840 ORCC #S04
01850
01860 * EXIT
3A72 39 01870 L0003 RTS
0000 32767 END
```

00000 TOTAL ERRORS

=====

FRE0: Empty the Square at
F = File Character
R = Rank Character
Actual Command = FR=0
Transfer Command

```

00100 *****
00110 *
00120 * FRE0.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * EMPTY A SQUARE
00170 * ACTUAL COMMAND = FR=0
00180 * TRANSFER COMMAND
00190 *
00200 *****
00210
00220 FRQ0 EQU $40A0
00230
00240 ORG $3A80
00250
00260 * BRANCH TO THE
00270 * ACTION COMMAND
00280 LBRA FRQ0
00290
0000 32767 END

00000 TOTAL ERRORS

```

=====

**FRECP: Place a
 C = Color (W,B)
 P = Piece (K,Q,R,B,N,P) at
 F = File Character
 R = Rank Character
 Actual Command = FR=CP
 Transfer Command**

```

00100 *****
00110 *
00120 * FRECP.ASM
00130 * MDJ 2021/09/25
00140 *
00150 * COMMAND TO
00160 * PLACE A PIECE
00170 * ON A SQUARE
00180 * ACTUAL COMMAND = FR=CP
00190 * TRANSFER COMMAND
00200 *
00210 *****
00220
          41A0 00230 FRQCP   EQU      $41A0
00240
3A90          00250          ORG      $3A90
00260
00270 * BRANCH TO THE
00280 * ACTION COMMAND
3A90 16      070D 00290          LBRA   FRQCP
00300
          0000 32767          END

```

00000 TOTAL ERRORS

=====

FRFR: Standard Move From
F = File Character
R = Rank Character
To
F = File Character
R = Rank Character
(e.g. E2E4)

```

00100 *****
00110 *
00120 * FRFR.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * DO STANDARD MOVE
00170 *
00180 * NOT YET
00190 * IMPLEMENTED
00200 *
00210 *****
00220
00230 * LOW RAM CURSOR ADDRESS
0088 00240 CURPOS EQU $0088
00250
00260 * SCREEN ADDRESSES
00270 * START OF VIDRAM
0400 00280 VIDRAM EQU $0400
00290
00300 * COMMAND PROMPT
00310 * SCREEN LOCATION
05AE 00320 CMDPMT EQU $05AE
00330
00340 * SCREEN REPORTING FIELD
05EB 00350 RPTFLD EQU $05EB
00360
00370 * ONE BYTE PAST THE
00380 * END OF VIDRAM
0600 00390 VIDEND EQU $0600
00400

```

	00410	* ML FOUNDATION		
	00420	* CORE ADDRESSES		
1C00	00430	REGXFR	EQU	\$1C00
1C0E	00440	VIDCLS	EQU	\$1C0E
1C1F	00450	PUTCHR	EQU	\$1C1F
1CD2	00460	GETCHR	EQU	\$1CD2
1CD5	00470	PUTBYT	EQU	\$1CD5
1D17	00480	SCROLL	EQU	\$1D17
1D36	00490	PUTCHA	EQU	\$1D36
1D57	00500	PUTBYA	EQU	\$1D57
1D8C	00510	CRLF	EQU	\$1D8C
1DBD	00520	PK2PRT	EQU	\$1DBD
1DDA	00530	PRT2PK	EQU	\$1DDA
1DF7	00540	POLCAT	EQU	\$1DF7
1E06	00550	PUTWRA	EQU	\$1E06
1E11	00560	PUTWRD	EQU	\$1E11
1E1C	00570	BKSPCE	EQU	\$1E1C
1E34	00580	DU1616	EQU	\$1E34
1E56	00590	FUDWRD	EQU	\$1E56
1E70	00600	PTDWRD	EQU	\$1E70
	00610			
	00620	* YACH ADDRESSES		
	00630	* SYSTEM FLAGS		
2000	00640	FLAGS	EQU	\$2000
2000	00650	SCFLAG	EQU	\$2000
2001	00660	STFLAG	EQU	\$2001
2002	00670	BFFLAG	EQU	\$2002
	00680			
	00690	* SYSTEM TEMPORARY		
	00700	* VARIABLES		
2040	00710	TMPVAR	EQU	\$2040
2040	00720	CMDCNT	EQU	\$2040
2041	00730	CMDVAR	EQU	\$2041
2046	00740	CMDFRF	EQU	\$2046
2047	00750	CMDFRR	EQU	\$2047
2048	00760	CMDTOF	EQU	\$2048
2049	00770	CMDTOR	EQU	\$2049
204A	00780	CMDFRI	EQU	\$204A
204B	00790	CMDFRJ	EQU	\$204B
204C	00800	CMDTOI	EQU	\$204C
204D	00810	CMDTOJ	EQU	\$204D
204E	00820	MCHVAR	EQU	\$204E
	00830			
	00840	* SCREEN MANAGEMENT		
20C0	00850	CIDGEN	EQU	\$20C0
24C0	00860	B2SARY	EQU	\$24C0
2540	00870	B2DARY	EQU	\$2540

2580	00880	B2LARY	EQU	\$2580
25C0	00890	BSCARY	EQU	\$25C0
	00900			
	00910	* GENERAL REPORTING		
2600	00920	NOTREC	EQU	\$2600
2680	00930	NYIMSG	EQU	\$2680
	00940			
	00950	* GENERAL COMMANDS		
26C0	00960	ZERO2	EQU	\$26C0
26D0	00970	ZERO3	EQU	\$26D0
26E0	00980	ANLYZ	EQU	\$26E0
27E0	00990	ASKFR	EQU	\$27E0
28E0	01000	CLRBD	EQU	\$28E0
29E0	01010	CKMOV	EQU	\$29E0
2AE0	01020	CKPOS	EQU	\$2AE0
2BE0	01030	CSTLK	EQU	\$2BE0
2CE0	01040	CSTLQ	EQU	\$2CE0
2DE0	01050	DSPBD	EQU	\$2DE0
2EE0	01060	FLIP	EQU	\$2EE0
2FE0	01070	GSTRT	EQU	\$2FE0
30E0	01080	GSTOP	EQU	\$30E0
31E0	01090	LDGAM	EQU	\$31E0
32E0	01100	LDPOS	EQU	\$32E0
33E0	01110	NEW	EQU	\$33E0
33F0	01120	NEWPS	EQU	\$33F0
34F0	01130	LTRO2	EQU	\$34F0
3500	01140	LTRO3	EQU	\$3500
3510	01150	QUIT	EQU	\$3510
3520	01160	SCOFF	EQU	\$3520
3620	01170	SCON	EQU	\$3620
3720	01180	START	EQU	\$3720
3730	01190	STOP	EQU	\$3730
3740	01200	SVGAM	EQU	\$3740
3840	01210	SVPOS	EQU	\$3840
3940	01220	TKBAK	EQU	\$3940
	01230			
	01240	* MOVEMENT COMMANDS		
	01250	* AND CALCULATIONS		
3A40	01260	FILCVT	EQU	\$3A40
3A60	01270	RNKCVT	EQU	\$3A60
3A80	01280	FRE0	EQU	\$3A80
3A90	01290	FRECP	EQU	\$3A90
	01300	*FRFR	EQU	\$3AA0
3EA0	01310	FRFRE	EQU	\$3EA0
3FA0	01320	FRFRP	EQU	\$3FA0
40A0	01330	FRQ0	EQU	\$40A0
41A0	01340	FRQCP	EQU	\$41A0

			01350			
			01360	* COMMAND MANAGEMENT		
	42A0		01370	GETCMD EQU	\$42A0	
	43A0		01380	MATCH3 EQU	\$43A0	
	4420		01390	MATCH4 EQU	\$4420	
	44A0		01400	XLTCMD EQU	\$44A0	
	48A0		01410	XLTCM2 EQU	\$48A0	
			01420			
			01430	* YACH SUPERVISORY		
	4CA0		01440	GMLOOP EQU	\$4CA0	
	4CE0		01450	YACH EQU	\$4CE0	
			01460			
3AA0			01470	ORG	\$3AA0	
			01480			
3AA0	34	36	01490	FRFR PSHS	A,B,X,Y	
			01500			
			01510	* POINT TO THE		
			01520	* REPORTING FIELD		
3AA2	8E	05EB	01530	LDX	#RPTFLD	
			01540			
			01550	* COMMAND NYI		
			01560	* MESSAGE		
3AA5	86	53	01570	LDA	#83	S
3AA7	A7	80	01580	STA	,X+	
3AA9	86	54	01590	LDA	#84	T
3AAB	A7	80	01600	STA	,X+	
3AAD	86	44	01610	LDA	#68	D
3AAF	A7	80	01620	STA	,X+	
3AB1	86	6E	01630	LDA	#110	.
3AB3	A7	80	01640	STA	,X+	
3AB5	86	60	01650	LDA	#96	SP
3AB7	A7	80	01660	STA	,X+	
3AB9	86	4D	01670	LDA	#77	M
3ABB	A7	80	01680	STA	,X+	
3ABD	86	4F	01690	LDA	#79	O
3ABF	A7	80	01700	STA	,X+	
3AC1	86	56	01710	LDA	#86	V
3AC3	A7	80	01720	STA	,X+	
3AC5	86	45	01730	LDA	#69	E
3AC7	A7	80	01740	STA	,X+	
3AC9	86	53	01750	LDA	#83	S
3ACB	A7	80	01760	STA	,X+	
			01770			
			01780	* BRANCH TO NYI		
			01790	* MESSAGE		
3ACD	16	EBB0	01800	LBRA	NYIMSG	
		0000	32767	END		

00000 TOTAL ERRORS

=====

FRFRE: Standard Move Plus Capture En Passant (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * FRFRE.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * DO STANDARD MOVE
00170 * PLUS EN PASSANT
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00
1C0E 00450 VIDCLS EQU $1C0E
1C1F 00460 PUTCHR EQU $1C1F
1CD2 00470 GETCHR EQU $1CD2
1CD5 00480 PUTBYT EQU $1CD5

```

1D17	00490	SCROLL	EQU	\$1D17
1D36	00500	PUTCHA	EQU	\$1D36
1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			
	00920	* GENERAL REPORTING		
2600	00930	NOTREC	EQU	\$2600
2680	00940	NYIMSG	EQU	\$2680
	00950			

	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2	EQU	\$26C0
26D0	00980	ZERO3	EQU	\$26D0
26E0	00990	ANLYZ	EQU	\$26E0
27E0	01000	ASKFR	EQU	\$27E0
28E0	01010	CLRBD	EQU	\$28E0
29E0	01020	CKMOV	EQU	\$29E0
2AE0	01030	CKPOS	EQU	\$2AE0
2BE0	01040	CSTLK	EQU	\$2BE0
2CE0	01050	CSTLQ	EQU	\$2CE0
2DE0	01060	DSPBD	EQU	\$2DE0
2EE0	01070	FLIP	EQU	\$2EE0
2FE0	01080	GSTRT	EQU	\$2FE0
30E0	01090	GSTOP	EQU	\$30E0
31E0	01100	LDGAM	EQU	\$31E0
32E0	01110	LDPOS	EQU	\$32E0
33E0	01120	NEW	EQU	\$33E0
33F0	01130	NEWPS	EQU	\$33F0
34F0	01140	LTRO2	EQU	\$34F0
3500	01150	LTRO3	EQU	\$3500
3510	01160	QUIT	EQU	\$3510
3520	01170	SCOFF	EQU	\$3520
3620	01180	SCON	EQU	\$3620
3720	01190	START	EQU	\$3720
3730	01200	STOP	EQU	\$3730
3740	01210	SVGAM	EQU	\$3740
3840	01220	SVPOS	EQU	\$3840
3940	01230	TKBAK	EQU	\$3940
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT	EQU	\$3A40
3A60	01280	RNKCVT	EQU	\$3A60
3A80	01290	FRE0	EQU	\$3A80
3A90	01300	FRECP	EQU	\$3A90
3AA0	01310	FRFR	EQU	\$3AA0
	01320	*FRFRE	EQU	\$3EA0
3FA0	01330	FRFRP	EQU	\$3FA0
40A0	01340	FRQ0	EQU	\$40A0
41A0	01350	FRQCP	EQU	\$41A0
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD	EQU	\$42A0
43A0	01390	MATCH3	EQU	\$43A0
4420	01400	MATCH4	EQU	\$4420
44A0	01410	XLTCMD	EQU	\$44A0
48A0	01420	XLTCM2	EQU	\$48A0


```

01430
01440 * YACH SUPERVISORY
4CA0 01450 GMLOOP EQU $4CA0
4CE0 01460 YACH EQU $4CE0
01470
3EA0 01480 ORG $3EA0
01490
3EA0 34 36 01500 FRFRE PSHS A,B,X,Y
01510
01520 * POINT TO THE
01530 * REPORTING FIELD
3EA2 8E 05EB 01540 LDX #RPTFLD
01550
01560 * COMMAND NYI
01570 * MESSAGE
3EA5 86 45 01580 LDA #69 E
3EA7 A7 80 01590 STA ,X+
3EA9 86 4E 01600 LDA #78 N
3EAB A7 80 01610 STA ,X+
3EAD 86 60 01620 LDA #96 SP
3EAF A7 80 01630 STA ,X+
3EB1 86 50 01640 LDA #80 P
3EB3 A7 80 01650 STA ,X+
3EB5 86 41 01660 LDA #65 A
3EB7 A7 80 01670 STA ,X+
3EB9 86 53 01680 LDA #83 S
3EBB A7 80 01690 STA ,X+
3EBD 86 53 01700 LDA #83 S
3EBF A7 80 01710 STA ,X+
3EC1 86 41 01720 LDA #65 A
3EC3 A7 80 01730 STA ,X+
3EC5 86 4E 01740 LDA #78 N
3EC7 A7 80 01750 STA ,X+
3EC9 86 54 01760 LDA #84 T
3ECB A7 80 01770 STA ,X+
01780
01790 * BRANCH TO NYI
01800 * MESSAGE
3ECD 16 E7B0 01810 LBRA NYIMSG
0000 32767 END

```

00000 TOTAL ERRORS

=====

FRFRP: Standard Move Plus Pawn Promotion (N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * FRFRP.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * DO STANDARD MOVE
00170 * PLUS PAWN
00180 * PROMOTION
00190 *
00200 * NOT YET
00210 * IMPLEMENTED
00220 *
00230 *****
00240
00250 * LOW RAM CURSOR ADDRESS
0088 00260 CURPOS EQU $0088
00270
00280 * SCREEN ADDRESSES
00290 * START OF VIDRAM
0400 00300 VIDRAM EQU $0400
00310
00320 * COMMAND PROMPT
00330 * SCREEN LOCATION
05AE 00340 CMDPMT EQU $05AE
00350
00360 * SCREEN REPORTING FIELD
05EB 00370 RPTFLD EQU $05EB
00380
00390 * ONE BYTE PAST THE
00400 * END OF VIDRAM
0600 00410 VIDEND EQU $0600
00420
00430 * ML FOUNDATION
00440 * CORE ADDRESSES
1C00 00450 REGXFR EQU $1C00
1C0E 00460 VIDCLS EQU $1C0E
1C1F 00470 PUTCHR EQU $1C1F
1CD2 00480 GETCHR EQU $1CD2

```

1CD5	00490	PUTBYT	EQU	\$1CD5
1D17	00500	SCROLL	EQU	\$1D17
1D36	00510	PUTCHA	EQU	\$1D36
1D57	00520	PUTBYA	EQU	\$1D57
1D8C	00530	CRLF	EQU	\$1D8C
1DBD	00540	PK2PRT	EQU	\$1DBD
1DDA	00550	PRT2PK	EQU	\$1DDA
1DF7	00560	POLCAT	EQU	\$1DF7
1E06	00570	PUTWRA	EQU	\$1E06
1E11	00580	PUTWRD	EQU	\$1E11
1E1C	00590	BKSPCE	EQU	\$1E1C
1E34	00600	DU1616	EQU	\$1E34
1E56	00610	FUDWRD	EQU	\$1E56
1E70	00620	PTDWRD	EQU	\$1E70
	00630			
	00640	* YACH ADDRESSES		
	00650	* SYSTEM FLAGS		
2000	00660	FLAGS	EQU	\$2000
2000	00670	SCFLAG	EQU	\$2000
2001	00680	STFLAG	EQU	\$2001
2002	00690	BFFLAG	EQU	\$2002
	00700			
	00710	* SYSTEM TEMPORARY		
	00720	* VARIABLES		
2040	00730	TMPVAR	EQU	\$2040
2040	00740	CMDCNT	EQU	\$2040
2041	00750	CMDVAR	EQU	\$2041
2046	00760	CMDFRF	EQU	\$2046
2047	00770	CMDFRR	EQU	\$2047
2048	00780	CMDTOF	EQU	\$2048
2049	00790	CMDTOR	EQU	\$2049
204A	00800	CMDFRI	EQU	\$204A
204B	00810	CMDFRJ	EQU	\$204B
204C	00820	CMDTOI	EQU	\$204C
204D	00830	CMDTOJ	EQU	\$204D
204E	00840	MCHVAR	EQU	\$204E
	00850			
	00860	* SCREEN MANAGEMENT		
20C0	00870	CIDGEN	EQU	\$20C0
24C0	00880	B2SARY	EQU	\$24C0
2540	00890	B2DARY	EQU	\$2540
2580	00900	B2LARY	EQU	\$2580
25C0	00910	BSCARY	EQU	\$25C0
	00920			
	00930	* GENERAL REPORTING		
2600	00940	NOTREC	EQU	\$2600
2680	00950	NYIMSG	EQU	\$2680

```

00960
00970 * GENERAL COMMANDS
26C0 00980 ZERO2 EQU $26C0
26D0 00990 ZERO3 EQU $26D0
26E0 01000 ANLYZ EQU $26E0
27E0 01010 ASKFR EQU $27E0
28E0 01020 CLRBD EQU $28E0
29E0 01030 CKMOV EQU $29E0
2AE0 01040 CKPOS EQU $2AE0
2BE0 01050 CSTLK EQU $2BE0
2CE0 01060 CSTLQ EQU $2CE0
2DE0 01070 DSPBD EQU $2DE0
2EE0 01080 FLIP EQU $2EE0
2FE0 01090 GSTRT EQU $2FE0
30E0 01100 GSTOP EQU $30E0
31E0 01110 LDGAM EQU $31E0
32E0 01120 LDPOS EQU $32E0
33E0 01130 NEW EQU $33E0
33F0 01140 NEWPS EQU $33F0
34F0 01150 LTRO2 EQU $34F0
3500 01160 LTRO3 EQU $3500
3510 01170 QUIT EQU $3510
3520 01180 SCOFF EQU $3520
3620 01190 SCON EQU $3620
3720 01200 START EQU $3720
3730 01210 STOP EQU $3730
3740 01220 SVGAM EQU $3740
3840 01230 SVPOS EQU $3840
3940 01240 TKBAK EQU $3940
01250
01260 * MOVEMENT COMMANDS
01270 * AND CALCULATIONS
3A40 01280 FILCVT EQU $3A40
3A60 01290 RNKCVT EQU $3A60
3A80 01300 FRE0 EQU $3A80
3A90 01310 FRECP EQU $3A90
3AA0 01320 FRFR EQU $3AA0
3EA0 01330 FRFRE EQU $3EA0
01340 *FRFRP EQU $3FA0
40A0 01350 FRQ0 EQU $40A0
41A0 01360 FRQCP EQU $41A0
01370
01380 * COMMAND MANAGEMENT
42A0 01390 GETCMD EQU $42A0
43A0 01400 MATCH3 EQU $43A0
4420 01410 MATCH4 EQU $4420
44A0 01420 XLTCMD EQU $44A0

```

	48A0		01430	XLTCM2	EQU	\$48A0	
			01440				
			01450	* YACH SUPERVISORY			
	4CA0		01460	GMLOOP	EQU	\$4CA0	
	4CE0		01470	YACH	EQU	\$4CE0	
			01480				
3FA0			01490		ORG	\$3FA0	
			01500				
3FA0	34	36	01510	FRFRP	PSHS	A,B,X,Y	
			01520				
			01530	* POINT TO THE			
			01540	* REPORTING FIELD			
3FA2	8E	05EB	01550		LDX	#RPTFLD	
			01560				
			01570	* COMMAND NYI			
			01580	* MESSAGE			
3FA5	86	50	01590		LDA	#80	P
3FA7	A7	80	01600		STA	,X+	
3FA9	86	52	01610		LDA	#82	R
3FAB	A7	80	01620		STA	,X+	
3FAD	86	4F	01630		LDA	#79	O
3FAF	A7	80	01640		STA	,X+	
3FB1	86	4D	01650		LDA	#77	M
3FB3	A7	80	01660		STA	,X+	
3FB5	86	4F	01670		LDA	#79	O
3FB7	A7	80	01680		STA	,X+	
3FB9	86	54	01690		LDA	#84	T
3FBB	A7	80	01700		STA	,X+	
3FBD	86	49	01710		LDA	#73	I
3FBF	A7	80	01720		STA	,X+	
3FC1	86	4F	01730		LDA	#79	O
3FC3	A7	80	01740		STA	,X+	
3FC5	86	4E	01750		LDA	#78	N
3FC7	A7	80	01760		STA	,X+	
			01770				
			01780	* BRANCH TO NYI			
			01790	* MESSAGE			
3FC9	16	E6B4	01800		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

FRQ0: Empty the Square at
F = File Character
R = Rank Character
Action Command
(N.Y.I. - Not Yet Implemented)

```

00100 *****
00110 *
00120 * FRQ0.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * EMPTY A SQUARE
00170 * ACTION COMMAND
00180 *
00190 * NOT YET
00200 * IMPLEMENTED
00210 *
00220 *****
00230
00240 * LOW RAM CURSOR ADDRESS
0088 00250 CURPOS EQU $0088
00260
00270 * SCREEN ADDRESSES
00280 * START OF VIDRAM
0400 00290 VIDRAM EQU $0400
00300
00310 * COMMAND PROMPT
00320 * SCREEN LOCATION
05AE 00330 CMDPMT EQU $05AE
00340
00350 * SCREEN REPORTING FIELD
05EB 00360 RPTFLD EQU $05EB
00370
00380 * ONE BYTE PAST THE
00390 * END OF VIDRAM
0600 00400 VIDEND EQU $0600
00410
00420 * ML FOUNDATION
00430 * CORE ADDRESSES
1C00 00440 REGXFR EQU $1C00

```

1C0E	00450	VIDCLS	EQU	\$1C0E
1C1F	00460	PUTCHR	EQU	\$1C1F
1CD2	00470	GETCHR	EQU	\$1CD2
1CD5	00480	PUTBYT	EQU	\$1CD5
1D17	00490	SCROLL	EQU	\$1D17
1D36	00500	PUTCHA	EQU	\$1D36
1D57	00510	PUTBYA	EQU	\$1D57
1D8C	00520	CRLF	EQU	\$1D8C
1DBD	00530	PK2PRT	EQU	\$1DBD
1DDA	00540	PRT2PK	EQU	\$1DDA
1DF7	00550	POLCAT	EQU	\$1DF7
1E06	00560	PUTWRA	EQU	\$1E06
1E11	00570	PUTWRD	EQU	\$1E11
1E1C	00580	BKSPCE	EQU	\$1E1C
1E34	00590	DU1616	EQU	\$1E34
1E56	00600	FUDWRD	EQU	\$1E56
1E70	00610	PTDWRD	EQU	\$1E70
	00620			
	00630	* YACH ADDRESSES		
	00640	* SYSTEM FLAGS		
2000	00650	FLAGS	EQU	\$2000
2000	00660	SCFLAG	EQU	\$2000
2001	00670	STFLAG	EQU	\$2001
2002	00680	BFFLAG	EQU	\$2002
	00690			
	00700	* SYSTEM TEMPORARY		
	00710	* VARIABLES		
2040	00720	TMPVAR	EQU	\$2040
2040	00730	CMDCNT	EQU	\$2040
2041	00740	CMDVAR	EQU	\$2041
2046	00750	CMDFRF	EQU	\$2046
2047	00760	CMDFRR	EQU	\$2047
2048	00770	CMDTOF	EQU	\$2048
2049	00780	CMDTOR	EQU	\$2049
204A	00790	CMDFRI	EQU	\$204A
204B	00800	CMDFRJ	EQU	\$204B
204C	00810	CMDTOI	EQU	\$204C
204D	00820	CMDTOJ	EQU	\$204D
204E	00830	MCHVAR	EQU	\$204E
	00840			
	00850	* SCREEN MANAGEMENT		
20C0	00860	CIDGEN	EQU	\$20C0
24C0	00870	B2SARY	EQU	\$24C0
2540	00880	B2DARY	EQU	\$2540
2580	00890	B2LARY	EQU	\$2580
25C0	00900	BSCARY	EQU	\$25C0
	00910			

	00920	* GENERAL REPORTING		
2600	00930	NOTREC EQU	\$2600	
2680	00940	NYIMSG EQU	\$2680	
	00950			
	00960	* GENERAL COMMANDS		
26C0	00970	ZERO2 EQU	\$26C0	
26D0	00980	ZERO3 EQU	\$26D0	
26E0	00990	ANLYZ EQU	\$26E0	
27E0	01000	ASKFR EQU	\$27E0	
28E0	01010	CLRBD EQU	\$28E0	
29E0	01020	CKMOV EQU	\$29E0	
2AE0	01030	CKPOS EQU	\$2AE0	
2BE0	01040	CSTLK EQU	\$2BE0	
2CE0	01050	CSTLQ EQU	\$2CE0	
2DE0	01060	DSPBD EQU	\$2DE0	
2EE0	01070	FLIP EQU	\$2EE0	
2FE0	01080	GSTRT EQU	\$2FE0	
30E0	01090	GSTOP EQU	\$30E0	
31E0	01100	LDGAM EQU	\$31E0	
32E0	01110	LDPOS EQU	\$32E0	
33E0	01120	NEW EQU	\$33E0	
33F0	01130	NEWPS EQU	\$33F0	
34F0	01140	LTRO2 EQU	\$34F0	
3500	01150	LTRO3 EQU	\$3500	
3510	01160	QUIT EQU	\$3510	
3520	01170	SCOFF EQU	\$3520	
3620	01180	SCON EQU	\$3620	
3720	01190	START EQU	\$3720	
3730	01200	STOP EQU	\$3730	
3740	01210	SVGAM EQU	\$3740	
3840	01220	SVPOS EQU	\$3840	
3940	01230	TKBAK EQU	\$3940	
	01240			
	01250	* MOVEMENT COMMANDS		
	01260	* AND CALCULATIONS		
3A40	01270	FILCVT EQU	\$3A40	
3A60	01280	RNKCVT EQU	\$3A60	
3A80	01290	FRE0 EQU	\$3A80	
3A90	01300	FRECP EQU	\$3A90	
3AA0	01310	FRFR EQU	\$3AA0	
3EA0	01320	FRFRE EQU	\$3EA0	
3FA0	01330	FRFRP EQU	\$3FA0	
	01340	*FRQ0 EQU	\$40A0	
41A0	01350	FRQCP EQU	\$41A0	
	01360			
	01370	* COMMAND MANAGEMENT		
42A0	01380	GETCMD EQU	\$42A0	

	43A0		01390	MATCH3	EQU	\$43A0	
	4420		01400	MATCH4	EQU	\$4420	
	44A0		01410	XLTCMD	EQU	\$44A0	
	48A0		01420	XLTCM2	EQU	\$48A0	
			01430				
			01440	* YACH SUPERVISORY			
	4CA0		01450	GMLOOP	EQU	\$4CA0	
	4CE0		01460	YACH	EQU	\$4CE0	
			01470				
40A0			01480		ORG	\$40A0	
			01490				
40A0	34	36	01500	FRQ0	PSHS	A,B,X,Y	
			01510				
			01520	* POINT TO THE			
			01530	* REPORTING FIELD			
40A2	8E	05EB	01540		LDX	#RPTFLD	
			01550				
			01560	* COMMAND NYI			
			01570	* MESSAGE			
40A5	86	45	01580		LDA	#69	E
40A7	A7	80	01590		STA	,X+	
40A9	86	4D	01600		LDA	#77	M
40AB	A7	80	01610		STA	,X+	
40AD	86	50	01620		LDA	#80	P
40AF	A7	80	01630		STA	,X+	
40B1	86	54	01640		LDA	#84	T
40B3	A7	80	01650		STA	,X+	
40B5	86	59	01660		LDA	#89	Y
40B7	A7	80	01670		STA	,X+	
40B9	86	60	01680		LDA	#96	SP
40BB	A7	80	01690		STA	,X+	
40BD	86	53	01700		LDA	#83	S
40BF	A7	80	01710		STA	,X+	
40C1	86	51	01720		LDA	#81	Q
40C3	A7	80	01730		STA	,X+	
40C5	86	52	01740		LDA	#82	R
40C7	A7	80	01750		STA	,X+	
			01760				
			01770	* BRANCH TO NYI			
			01780	* MESSAGE			
40C9	16	E5B4	01790		LBRA	NYIMSG	
		0000	32767		END		

00000 TOTAL ERRORS

=====

**FRQCP: Place a
 C = Color (W,B)
 P = Piece (K,Q,R,B,N,P) at
 F = File Character
 R = Rank Character
 Action Command
 (N.Y.I. - Not Yet Implemented)**

```

00100 *****
00110 *
00120 * FRQCP.ASM
00130 * MDJ 2021/09/20
00140 *
00150 * COMMAND TO
00160 * PLACE A PIECE
00170 * ON A SQUARE
00180 * ACTION COMMAND
00190 *
00200 * NOT YET
00210 * IMPLEMENTED
00220 *
00230 *****
00240
00250 * LOW RAM CURSOR ADDRESS
0088 00260 CURPOS EQU $0088
00270
00280 * SCREEN ADDRESSES
00290 * START OF VIDRAM
0400 00300 VIDRAM EQU $0400
00310
00320 * COMMAND PROMPT
00330 * SCREEN LOCATION
05AE 00340 CMDPMT EQU $05AE
00350
00360 * SCREEN REPORTING FIELD
05EB 00370 RPTFLD EQU $05EB
00380
00390 * ONE BYTE PAST THE
00400 * END OF VIDRAM

```

0600	00410	VIDEND	EQU	\$0600
	00420			
	00430	* ML FOUNDATION		
	00440	* CORE ADDRESSES		
1C00	00450	REGXFR	EQU	\$1C00
1C0E	00460	VIDCLS	EQU	\$1C0E
1C1F	00470	PUTCHR	EQU	\$1C1F
1CD2	00480	GETCHR	EQU	\$1CD2
1CD5	00490	PUTBYT	EQU	\$1CD5
1D17	00500	SCROLL	EQU	\$1D17
1D36	00510	PUTCHA	EQU	\$1D36
1D57	00520	PUTBYA	EQU	\$1D57
1D8C	00530	CRLF	EQU	\$1D8C
1DBD	00540	PK2PRT	EQU	\$1DBD
1DDA	00550	PRT2PK	EQU	\$1DDA
1DF7	00560	POLCAT	EQU	\$1DF7
1E06	00570	PUTWRA	EQU	\$1E06
1E11	00580	PUTWRD	EQU	\$1E11
1E1C	00590	BKSPCE	EQU	\$1E1C
1E34	00600	DU1616	EQU	\$1E34
1E56	00610	FUDWRD	EQU	\$1E56
1E70	00620	PTDWRD	EQU	\$1E70
	00630			
	00640	* YACH ADDRESSES		
	00650	* SYSTEM FLAGS		
2000	00660	FLAGS	EQU	\$2000
2000	00670	SCFLAG	EQU	\$2000
2001	00680	STFLAG	EQU	\$2001
2002	00690	BFFLAG	EQU	\$2002
	00700			
	00710	* SYSTEM TEMPORARY		
	00720	* VARIABLES		
2040	00730	TMPVAR	EQU	\$2040
2040	00740	CMDCNT	EQU	\$2040
2041	00750	CMDVAR	EQU	\$2041
2046	00760	CMDFRF	EQU	\$2046
2047	00770	CMDFRR	EQU	\$2047
2048	00780	CMDTOF	EQU	\$2048
2049	00790	CMDTOR	EQU	\$2049
204A	00800	CMDFRI	EQU	\$204A
204B	00810	CMDFRJ	EQU	\$204B
204C	00820	CMDTOI	EQU	\$204C
204D	00830	CMDTOJ	EQU	\$204D
204E	00840	MCHVAR	EQU	\$204E
	00850			
	00860	* SCREEN MANAGEMENT		
20C0	00870	CIDGEN	EQU	\$20C0

24C0	00880	B2SARY	EQU	\$24C0
2540	00890	B2DARY	EQU	\$2540
2580	00900	B2LARY	EQU	\$2580
25C0	00910	BSCARY	EQU	\$25C0
	00920			
	00930	* GENERAL REPORTING		
2600	00940	NOTREC	EQU	\$2600
2680	00950	NYIMSG	EQU	\$2680
	00960			
	00970	* GENERAL COMMANDS		
26C0	00980	ZERO2	EQU	\$26C0
26D0	00990	ZERO3	EQU	\$26D0
26E0	01000	ANLYZ	EQU	\$26E0
27E0	01010	ASKFR	EQU	\$27E0
28E0	01020	CLRBD	EQU	\$28E0
29E0	01030	CKMOV	EQU	\$29E0
2AE0	01040	CKPOS	EQU	\$2AE0
2BE0	01050	CSTLK	EQU	\$2BE0
2CE0	01060	CSTLQ	EQU	\$2CE0
2DE0	01070	DSPBD	EQU	\$2DE0
2EE0	01080	FLIP	EQU	\$2EE0
2FE0	01090	GSTRT	EQU	\$2FE0
30E0	01100	GSTOP	EQU	\$30E0
31E0	01110	LDGAM	EQU	\$31E0
32E0	01120	LDPOS	EQU	\$32E0
33E0	01130	NEW	EQU	\$33E0
33F0	01140	NEWPS	EQU	\$33F0
34F0	01150	LTRO2	EQU	\$34F0
3500	01160	LTRO3	EQU	\$3500
3510	01170	QUIT	EQU	\$3510
3520	01180	SCOFF	EQU	\$3520
3620	01190	SCON	EQU	\$3620
3720	01200	START	EQU	\$3720
3730	01210	STOP	EQU	\$3730
3740	01220	SVGAM	EQU	\$3740
3840	01230	SVPOS	EQU	\$3840
3940	01240	TKBAK	EQU	\$3940
	01250			
	01260	* MOVEMENT COMMANDS		
	01270	* AND CALCULATIONS		
3A40	01280	FILCVT	EQU	\$3A40
3A60	01290	RNKCVT	EQU	\$3A60
3A80	01300	FRE0	EQU	\$3A80
3A90	01310	FRECP	EQU	\$3A90
3AA0	01320	FRFR	EQU	\$3AA0
3EA0	01330	FRFRE	EQU	\$3EA0
3FA0	01340	FRFRP	EQU	\$3FA0

	40A0		01350	FRQ0	EQU	\$40A0	
			01360	*FRQCP	EQU	\$41A0	
			01370				
			01380	* COMMAND MANAGEMENT			
	42A0		01390	GETCMD	EQU	\$42A0	
	43A0		01400	MATCH3	EQU	\$43A0	
	4420		01410	MATCH4	EQU	\$4420	
	44A0		01420	XLTCMD	EQU	\$44A0	
	48A0		01430	XLTCM2	EQU	\$48A0	
			01440				
			01450	* YACH SUPERVISORY			
	4CA0		01460	GMLOOP	EQU	\$4CA0	
	4CE0		01470	YACH	EQU	\$4CE0	
			01480				
	41A0		01490		ORG	\$41A0	
			01500				
	41A0	34	36	01510	FRQCP	PSHS	A,B,X,Y
				01520			
				01530	* POINT TO THE		
				01540	* REPORTING FIELD		
	41A2	8E	05EB	01550		LDX	#RPTFLD
				01560			
				01570	* COMMAND NYI		
				01580	* MESSAGE		
	41A5	86	50	01590		LDA	#80 P
	41A7	A7	80	01600		STA	,X+
	41A9	86	4C	01610		LDA	#76 L
	41AB	A7	80	01620		STA	,X+
	41AD	86	41	01630		LDA	#65 A
	41AF	A7	80	01640		STA	,X+
	41B1	86	43	01650		LDA	#67 C
	41B3	A7	80	01660		STA	,X+
	41B5	86	45	01670		LDA	#69 E
	41B7	A7	80	01680		STA	,X+
	41B9	86	60	01690		LDA	#96 SP
	41BB	A7	80	01700		STA	,X+
	41BD	86	50	01710		LDA	#80 P
	41BF	A7	80	01720		STA	,X+
	41C1	86	43	01730		LDA	#67 C
	41C3	A7	80	01740		STA	,X+
	41C5	86	45	01750		LDA	#69 E
	41C7	A7	80	01760		STA	,X+
				01770			
				01780	* BRANCH TO NYI		
				01790	* MESSAGE		
	41C9	16	E4B4	01800		LBRA	NYIMSG
			0000	32767		END	

00000 TOTAL ERRORS

=====

MAKEMC: Make the Movement Commands and Calculations MC.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEMC.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "FILCVT.BIN"
1130 LOADM "RNKCVT.BIN"
1140 LOADM "FRE0.BIN"
1150 LOADM "FRECP.BIN"
1160 LOADM "FRFR.BIN"
1170 LOADM "FRFRE.BIN"
1180 LOADM "FRFRP.BIN"
1190 LOADM "FRQ0.BIN"
1200 LOADM "FRQCP.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "MC.BIN", &H3A40, &H429F, &H3A40
1330 '
32767 END
```

=====

Command Management

This is Section Six: Command Management

1. GETCMD: Get a Command
2. MATCH3: Checks for a Three-Character Command Match
3. MATCH4 Checks for a Four-Character Command Match
4. MATCH5 Checks for a Five-Character Command Match
5. XLTCMD: Translate and Dispatch a Command The First Third
6. XLTCM2: Translate and Dispatch a Command The Second Third
7. XLTCM3: Translate and Dispatch a Command The Final Third

=====

GETCMD: Get a Command

In this routine, you might want to take particular notice of Lines 2220-2240. Characters received from the keyboard are in the form of what I refer to as Print Mechanism Codes. Meanwhile, characters placed on the screen are in the form of what I refer to as Poke Mechanism Codes. (MDJ02).

The PRT2PK ML Foundation Routine is being used here to convert Print Mechanism Codes to Poke Mechanism Codes. (MDJ01).

```
00100 *****
00110 *
00120 * GETCMD.ASM
00130 * MDJ 2021/09/19
00140 *
00150 * GET A COMMAND
00160 * FROM THE CONSOLE
00170 *
00180 *****
00190
00200 * LOW RAM CURSOR ADDRESS
0088 00210 CURPOS EQU $0088
00220
00230 * SCREEN ADDRESSES
00240 * START OF VIDRAM
0400 00250 VIDRAM EQU $0400
00260
00270 * COMMAND PROMPT
00280 * SCREEN LOCATION
05AE 00290 CMDPMT EQU $05AE
00300
00310 * SCREEN REPORTING FIELD
05EB 00320 RPTFLD EQU $05EB
00330
00340 * ONE BYTE PAST THE
00350 * END OF VIDRAM
0600 00360 VIDEND EQU $0600
00370
00380 * ML FOUNDATION
00390 * CORE ADDRESSES
1C00 00400 REGXFR EQU $1C00
1C0E 00410 VIDCLS EQU $1C0E
1C1F 00420 PUTCHR EQU $1C1F
1CD2 00430 GETCHR EQU $1CD2
1CD5 00440 PUTBYT EQU $1CD5
1D17 00450 SCROLL EQU $1D17
```

1D36	00460	PUTCHA	EQU	\$1D36
1D57	00470	PUTBYA	EQU	\$1D57
1D8C	00480	CRLF	EQU	\$1D8C
1DBD	00490	PK2PRT	EQU	\$1DBD
1DDA	00500	PRT2PK	EQU	\$1DDA
1DF7	00510	POLCAT	EQU	\$1DF7
1E06	00520	PUTWRA	EQU	\$1E06
1E11	00530	PUTWRD	EQU	\$1E11
1E1C	00540	BKSPCE	EQU	\$1E1C
1E34	00550	DU1616	EQU	\$1E34
1E56	00560	FUDWRD	EQU	\$1E56
1E70	00570	PTDWRD	EQU	\$1E70
	00580			
	00590	* YACH ADDRESSES		
	00600	* SYSTEM FLAGS		
2000	00610	FLAGS	EQU	\$2000
2000	00620	SCFLAG	EQU	\$2000
2001	00630	STFLAG	EQU	\$2001
2002	00640	BFFLAG	EQU	\$2002
	00650			
	00660	* SYSTEM TEMPORARY		
	00670	* VARIABLES		
2040	00680	TMPVAR	EQU	\$2040
2040	00690	CMDCNT	EQU	\$2040
2041	00700	CMDVAR	EQU	\$2041
2046	00710	CMDFRF	EQU	\$2046
2047	00720	CMDFRR	EQU	\$2047
2048	00730	CMDTOF	EQU	\$2048
2049	00740	CMDTOR	EQU	\$2049
204A	00750	CMDFRI	EQU	\$204A
204B	00760	CMDFRJ	EQU	\$204B
204C	00770	CMDTOI	EQU	\$204C
204D	00780	CMDTOJ	EQU	\$204D
204E	00790	MCHVAR	EQU	\$204E
	00800			
	00810	* SCREEN MANAGEMENT		
20C0	00820	CIDGEN	EQU	\$20C0
24C0	00830	B2SARY	EQU	\$24C0
2540	00840	B2DARY	EQU	\$2540
2580	00850	B2LARY	EQU	\$2580
25C0	00860	BSCARY	EQU	\$25C0
	00870			
	00880	* GENERAL REPORTING		
2600	00890	NOTREC	EQU	\$2600
2680	00900	NYIMSG	EQU	\$2680
	00910			
	00920	* GENERAL COMMANDS		

26C0	00930	ZERO2	EQU	\$26C0
26D0	00940	ZERO3	EQU	\$26D0
26E0	00950	ANLYZ	EQU	\$26E0
27E0	00960	ASKFR	EQU	\$27E0
28E0	00970	CLRBD	EQU	\$28E0
29E0	00980	CKMOV	EQU	\$29E0
2AE0	00990	CKPOS	EQU	\$2AE0
2BE0	01000	CSTLK	EQU	\$2BE0
2CE0	01010	CSTLQ	EQU	\$2CE0
2DE0	01020	DSPBD	EQU	\$2DE0
2EE0	01030	FLIP	EQU	\$2EE0
2FE0	01040	GSTRT	EQU	\$2FE0
30E0	01050	GSTOP	EQU	\$30E0
31E0	01060	LDGAM	EQU	\$31E0
32E0	01070	LDPOS	EQU	\$32E0
33E0	01080	NEW	EQU	\$33E0
33F0	01090	NEWPS	EQU	\$33F0
34F0	01100	LTRO2	EQU	\$34F0
3500	01110	LTRO3	EQU	\$3500
3510	01120	QUIT	EQU	\$3510
3520	01130	SCOFF	EQU	\$3520
3620	01140	SCON	EQU	\$3620
3720	01150	START	EQU	\$3720
3730	01160	STOP	EQU	\$3730
3740	01170	SVGAM	EQU	\$3740
3840	01180	SVPOS	EQU	\$3840
3940	01190	TKBAK	EQU	\$3940
	01200			
	01210	* MOVEMENT COMMANDS		
	01220	* AND CALCULATIONS		
3A40	01230	FILCVT	EQU	\$3A40
3A60	01240	RNKCVT	EQU	\$3A60
3A80	01250	FRE0	EQU	\$3A80
3A90	01260	FRECP	EQU	\$3A90
3AA0	01270	FRFR	EQU	\$3AA0
3EA0	01280	FRFRE	EQU	\$3EA0
3FA0	01290	FRFRP	EQU	\$3FA0
40A0	01300	FRQ0	EQU	\$40A0
41A0	01310	FRQCP	EQU	\$41A0
	01320			
	01330	* COMMAND MANAGEMENT		
	01340	*GETCMD	EQU	\$42A0
43A0	01350	MATCH3	EQU	\$43A0
4420	01360	MATCH4	EQU	\$4420
4460	01370	MATCH5	EQU	\$4460
44A0	01380	XLTCMD	EQU	\$44A0
48A0	01390	XLTCM2	EQU	\$48A0

			01400	
			01410	* YACH SUPERVISORY
	4CA0		01420	GMLOOP EQU \$4CA0
	4CE0		01430	YACH EQU \$4CE0
			01440	
42A0			01450	ORG \$42A0
			01460	
42A0	34	37	01470	GETCMD PSHS A,B,X,Y,CC
			01480	
			01490	* ZERO THE COUNT
42A2	7F	2040	01500	CLR CMDCNT
			01510	
			01520	* POINT TO THE
			01530	* COMMAND VARIABLE
42A5	108E	2041	01540	LDY #CMDVAR
			01550	
			01560	* EMPTY THE
			01570	* COMMAND VARIABLE
42A9	86	60	01580	LDA #96
42AB	A7	A0	01590	STA ,Y+
42AD	A7	A0	01600	STA ,Y+
42AF	A7	A0	01610	STA ,Y+
42B1	A7	A0	01620	STA ,Y+
42B3	A7	A0	01630	STA ,Y+
42B5	108E	2041	01640	LDY #CMDVAR
			01650	
			01660	* DISPLAY THE COMMAND
			01670	* FIELD AS FIVE WHITE
			01680	* CHARACTERS AT THE
			01690	* COMMAND PROMPT
			01700	* SCREEN LOCATION
42B9	86	CF	01710	LDA #207
42BB	8E	05AE	01720	LDX #CMDPMT
42BE	A7	80	01730	STA ,X+
42C0	A7	80	01740	STA ,X+
42C2	A7	80	01750	STA ,X+
42C4	A7	80	01760	STA ,X+
42C6	A7	80	01770	STA ,X+
42C8	8E	05AE	01780	LDX #CMDPMT
			01790	
			01800	* PUT THE CURSOR AT THE
			01810	* COMMAND PROMPT
			01820	* LOCATION
42CB	9F	88	01830	STX CURPOS
			01840	
			01850	* WAIT FOR A KEY PRESS
42CD	17	DB27	01860	L0001 LBSR POLCAT

42D0	27	FB	01870	BEQ	L0001
			01880		
			01890	* WAS IT A	
			01900	* BACKSPACE?	
42D2	81	08	01910	CMPA	#8
			01920		
			01930	* GO IF YES	
42D4	27	21	01940	BEQ	L0002
			01950		
			01960	* WAS IT A	
			01970	* CARRIAGE RETURN?	
42D6	81	0D	01980	CMPA	#13
			01990		
			02000	* GO IF YES	
42D8	27	3D	02010	BEQ	L0003
			02020		
			02030	* IS IT LOWER THAN THE	
			02040	* "STANDARD" CHARACTERS	
			02050	* RANGE?	
42DA	81	20	02060	CMPA	#32
			02070		
			02080	* GO IF YES (IGNORE)	
42DC	25	EF	02090	BLO	L0001
			02100		
			02110	* IS IT ABOVE THE	
			02120	* "STANDARD" CHARACTERS	
			02130	* RANGE?	
42DE	81	5F	02140	CMPA	#95
			02150		
			02160	* GO IF YES (IGNORE)	
42E0	22	EB	02170	BHI	L0001
			02180		
			02190	* IT IS A	
			02200	* "STANDARD" CHARACTER	
			02210		
			02220	* CONVERT IT TO	
			02230	* POKE MECHANISM CODE	
42E2	17	DAF5	02240	LBSR	PRT2PK
			02250		
			02260	* GET THE COUNT	
42E5	F6	2040	02270	LDB	CMDCNT
			02280		
			02290	* IS IT FIVE?	
			02300	* I.E. "FULL"	
42E8	C1	05	02310	CMPB	#5
			02320		
			02330	* GO IF YES (IGNORE)	

42EA	24	E1	02340	BHS	L0001
			02350		
			02360	* INCREMENT THE COUNT	
42EC	7C	2040	02370	INC	CMDCNT
			02380		
			02390	* PUT CHARACTER TO	
			02400	* COMMAND VARIABLE	
42EF	A7	A0	02410	STA	,Y+
			02420		
			02430	* PUT CHARACTER TO	
			02440	* COMMAND FIELD ON	
			02450	* SCREEN	
42F1	A7	80	02460	STA	,X+
			02470		
			02480	* INCREMENT THE CURSOR	
42F3	9F	88	02490	STX	CURPOS
			02500		
			02510	* GO FOR THE	
			02520	* NEXT CHARACTER	
42F5	20	D6	02530	BRA	L0001
			02540		
			02550	* IT IS A	
			02560	* BACKSPACE	
			02570		
			02580	* GET THE COUNT, AND	
			02590	* IS IT ZERO?	
			02600	* I.E. "EMPTY"	
42F7	F6	2040	02610	L0002 LDB	CMDCNT
			02620		
			02630	* GO IF YES (IGNORE)	
42FA	27	D1	02640	BEQ	L0001
			02650		
			02660	* DECREMENT THE COUNT	
42FC	7A	2040	02670	DEC	CMDCNT
			02680		
			02690	* BACKSPACE THE	
			02700	* COMMAND VARIABLE	
42FF	31	3F	02710	LEAY	-1,Y
4301	34	02	02720	PSHS	A
4303	86	60	02730	LDA	#96
4305	A7	A4	02740	STA	,Y
4307	35	02	02750	PULS	A
			02760		
			02770	* DECREMENT THE CURSOR	
4309	30	1F	02780	LEAX	-1,X
430B	9F	88	02790	STX	CURPOS
			02800		

			02810	*	REMOVE THE CHARACTER	
			02820	*	FROM THE COMMAND	
			02830	*	FIELD	
430D	34	02	02840		PSHS	A
430F	86	CF	02850		LDA	#207
4311	A7	84	02860		STA	,X
4313	35	02	02870		PULS	A
			02880			
			02890	*	GO FOR THE	
			02900	*	NEXT CHARACTER	
4315	20	B6	02910		BRA	L0001
			02920			
			02930	*	IT IS A	
			02940	*	CARRIAGE RETURN	
			02950			
			02960	*	GET THE COUNT, AND	
			02970	*	IS IT ZERO?	
			02980	*	I.E. "EMPTY"	
4317	F6	2040	02990	L0003	LDB	CMDCNT
			03000			
			03010	*	GO IF YES (IGNORE)	
431A	27	B1	03020		BEQ	L0001
			03030			
			03040	*	EXIT	
431C	35	37	03050		PULS	A,B,X,Y,CC
431E	39		03060		RTS	
		0000	32767		END	

00000 TOTAL ERRORS

=====

MATCH3: Checks for a Three-Character Command Match

```
00100 *****
00110 *
00120 * MATCH3.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * CHECKS FOR A
00160 * THREE-CHARACTER
00170 * COMMAND MATCH
00180 *
00190 * ENTRY CONDITIONS
00200 * IN THE TEMPORARY
00210 * VARIABLES:
00220 *
00230 * CMDVAR =
00240 * THE COMMAND
00250 * CHARACTERS
00260 *
00270 * MCHVAR =
00280 * THE CHARACTERS
00290 * TO MATCH
00300 *
00310 * EXIT CONDITIONS:
00320 *
00330 * NO CHANGE IN
00340 *   CMDVAR OR
00350 *   MCHVAR
00360 *
00370 * IF MATCH:
00380 *   CC Z-BIT = 1
00390 *
00400 * IF NO MATCH:
00410 *   CC Z-BIT = 0
00420 *
00430 *****
00440
00450 * LOW RAM CURSOR ADDRESS
0088 00460 CURPOS EQU    $0088
00470
00480 * SCREEN ADDRESSES
00490 * START OF VIDRAM
0400 00500 VIDRAM EQU    $0400
```


	00510		
	00520	* COMMAND PROMPT	
	00530	* SCREEN LOCATION	
05AE	00540	CMDPMT EQU	\$05AE
	00550		
	00560	* SCREEN REPORTING FIELD	
05EB	00570	RPTFLD EQU	\$05EB
	00580		
	00590	* ONE BYTE PAST THE	
	00600	* END OF VIDRAM	
0600	00610	VIDEND EQU	\$0600
	00620		
	00630	* ML FOUNDATION	
	00640	* CORE ADDRESSES	
1C00	00650	REGXFR EQU	\$1C00
1C0E	00660	VIDCLS EQU	\$1C0E
1C1F	00670	PUTCHR EQU	\$1C1F
1CD2	00680	GETCHR EQU	\$1CD2
1CD5	00690	PUTBYT EQU	\$1CD5
1D17	00700	SCROLL EQU	\$1D17
1D36	00710	PUTCHA EQU	\$1D36
1D57	00720	PUTBYA EQU	\$1D57
1D8C	00730	CRLF EQU	\$1D8C
1DBD	00740	PK2PRT EQU	\$1DBD
1DDA	00750	PRT2PK EQU	\$1DDA
1DF7	00760	POLCAT EQU	\$1DF7
1E06	00770	PUTWRA EQU	\$1E06
1E11	00780	PUTWRD EQU	\$1E11
1E1C	00790	BKSPCE EQU	\$1E1C
1E34	00800	DU1616 EQU	\$1E34
1E56	00810	FUDWRD EQU	\$1E56
1E70	00820	PTDWRD EQU	\$1E70
	00830		
	00840	* YACH ADDRESSES	
	00850	* SYSTEM FLAGS	
2000	00860	FLAGS EQU	\$2000
2000	00870	SCFLAG EQU	\$2000
2001	00880	STFLAG EQU	\$2001
2002	00890	BFFLAG EQU	\$2002
	00900		
	00910	* SYSTEM TEMPORARY	
	00920	* VARIABLES	
2040	00930	TMPVAR EQU	\$2040
2040	00940	CMDCNT EQU	\$2040
2041	00950	CMDVAR EQU	\$2041
2046	00960	CMDFRF EQU	\$2046
2047	00970	CMDFRR EQU	\$2047

2048	00980	CMDTOF	EQU	\$2048
2049	00990	CMDTOR	EQU	\$2049
204A	01000	CMDFRI	EQU	\$204A
204B	01010	CMDFRJ	EQU	\$204B
204C	01020	CMDTOI	EQU	\$204C
204D	01030	CMDTOJ	EQU	\$204D
204E	01040	MCHVAR	EQU	\$204E
	01050			
	01060	* SCREEN MANAGEMENT		
20C0	01070	CIDGEN	EQU	\$20C0
24C0	01080	B2SARY	EQU	\$24C0
2540	01090	B2DARY	EQU	\$2540
2580	01100	B2LARY	EQU	\$2580
25C0	01110	BSCARY	EQU	\$25C0
	01120			
	01130	* GENERAL REPORTING		
2600	01140	NOTREC	EQU	\$2600
2680	01150	NYIMSG	EQU	\$2680
	01160			
	01170	* GENERAL COMMANDS		
26C0	01180	ZERO2	EQU	\$26C0
26D0	01190	ZERO3	EQU	\$26D0
26E0	01200	ANLYZ	EQU	\$26E0
27E0	01210	ASKFR	EQU	\$27E0
28E0	01220	CLRBD	EQU	\$28E0
29E0	01230	CKMOV	EQU	\$29E0
2AE0	01240	CKPOS	EQU	\$2AE0
2BE0	01250	CSTLK	EQU	\$2BE0
2CE0	01260	CSTLQ	EQU	\$2CE0
2DE0	01270	DSPBD	EQU	\$2DE0
2EE0	01280	FLIP	EQU	\$2EE0
2FE0	01290	GSTRT	EQU	\$2FE0
30E0	01300	GSTOP	EQU	\$30E0
31E0	01310	LDGAM	EQU	\$31E0
32E0	01320	LDPOS	EQU	\$32E0
33E0	01330	NEW	EQU	\$33E0
33F0	01340	NEWPS	EQU	\$33F0
34F0	01350	LTRO2	EQU	\$34F0
3500	01360	LTRO3	EQU	\$3500
3510	01370	QUIT	EQU	\$3510
3520	01380	SCOFF	EQU	\$3520
3620	01390	SCON	EQU	\$3620
3720	01400	START	EQU	\$3720
3730	01410	STOP	EQU	\$3730
3740	01420	SVGAM	EQU	\$3740
3840	01430	SVPOS	EQU	\$3840
3940	01440	TKBAK	EQU	\$3940

```

01450
01460 * MOVEMENT COMMANDS
01470 * AND CALCULATIONS
3A40 01480 FILCVT EQU $3A40
3A60 01490 RNKCVT EQU $3A60
3A80 01500 FRE0 EQU $3A80
3A90 01510 FRECP EQU $3A90
3AA0 01520 FRFR EQU $3AA0
3EA0 01530 FRFRE EQU $3EA0
3FA0 01540 FRFRP EQU $3FA0
40A0 01550 FRQ0 EQU $40A0
41A0 01560 FRQCP EQU $41A0
01570
01580 * COMMAND MANAGEMENT
42A0 01590 GETCMD EQU $42A0
01600 *MATCH3 EQU $43A0
4420 01610 MATCH4 EQU $4420
44A0 01620 XLTCMD EQU $44A0
48A0 01630 XLTCM2 EQU $48A0
01640
01650 * YACH SUPERVISORY
4CA0 01660 GMLOOP EQU $4CA0
4CE0 01670 YACH EQU $4CE0
01680
43A0 01690 ORG $43A0
01700
43A0 34 02 01710 MATCH3 PSHS A
01720
01730 * DO THE FIRST
01740 * CHARACTER
43A2 B6 2041 01750 LDA CMDVAR
43A5 B1 204E 01760 CMPA MCHVAR
01770
01780 * GO IF NO MATCH
43A8 26 14 01790 BNE L0001
01800
01810 * SECOND CHARACTER
43AA B6 2042 01820 LDA CMDVAR+1
43AD B1 204F 01830 CMPA MCHVAR+1
01840
01850 * GO IF NO MATCH
43B0 26 0C 01860 BNE L0001
01870
01880 * THIRD CHARACTER
43B2 B6 2043 01890 LDA CMDVAR+2
43B5 B1 2050 01900 CMPA MCHVAR+2
01910

```

			01920	*	GO IF NO MATCH	
43B8	26	04	01930		BNE	L0001
			01940			
			01950	*	IT IS A MATCH3	
			01960	*	SET THE Z-BIT	
43BA	1A	04	01970		ORCC	#\$04
43BC	20	02	01980		BRA	L0002
			01990			
			02000	*	IT IS NOT A MATCH3	
			02010	*	CLEAR THE Z-BIT	
43BE	1C	FB	02020	L0001	ANDCC	#\$FB
			02030			
			02040	*	EXIT	
43C0	35	02	02050	L0002	PULS	A
43C2	39		02060		RTS	
		0000	32767		END	

00000 TOTAL ERRORS

=====

MATCH4 Checks for a Four-Character Command Match

```
00100 *****
00110 *
00120 * MATCH4.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * CHECKS FOR A
00160 * FOUR-CHARACTER
00170 * COMMAND MATCH
00180 *
00190 * ENTRY CONDITIONS
00200 * IN THE TEMPORARY
00210 * VARIABLES:
00220 *
00230 * CMDVAR =
00240 * THE COMMAND
00250 * CHARACTERS
00260 *
00270 * MCHVAR =
00280 * THE CHARACTERS
00290 * TO MATCH
00300 *
00310 * EXIT CONDITIONS:
00320 *
00330 * NO CHANGE IN
00340 *   CMDVAR OR
00350 *   MCHVAR
00360 *
00370 * IF MATCH:
00380 *   CC Z-BIT = 1
00390 *
00400 * IF NO MATCH:
00410 *   CC Z-BIT = 0
00420 *
00430 *****
00440
00450 * LOW RAM CURSOR ADDRESS
0088 00460 CURPOS EQU    $0088
00470
00480 * SCREEN ADDRESSES
00490 * START OF VIDRAM
0400 00500 VIDRAM EQU    $0400
```

	00510		
	00520	* COMMAND PROMPT	
	00530	* SCREEN LOCATION	
05AE	00540	CMDPMT EQU	\$05AE
	00550		
	00560	* SCREEN REPORTING FIELD	
05EB	00570	RPTFLD EQU	\$05EB
	00580		
	00590	* ONE BYTE PAST THE	
	00600	* END OF VIDRAM	
0600	00610	VIDEND EQU	\$0600
	00620		
	00630	* ML FOUNDATION	
	00640	* CORE ADDRESSES	
1C00	00650	REGXFR EQU	\$1C00
1C0E	00660	VIDCLS EQU	\$1C0E
1C1F	00670	PUTCHR EQU	\$1C1F
1CD2	00680	GETCHR EQU	\$1CD2
1CD5	00690	PUTBYT EQU	\$1CD5
1D17	00700	SCROLL EQU	\$1D17
1D36	00710	PUTCHA EQU	\$1D36
1D57	00720	PUTBYA EQU	\$1D57
1D8C	00730	CRLF EQU	\$1D8C
1DBD	00740	PK2PRT EQU	\$1DBD
1DDA	00750	PRT2PK EQU	\$1DDA
1DF7	00760	POLCAT EQU	\$1DF7
1E06	00770	PUTWRA EQU	\$1E06
1E11	00780	PUTWRD EQU	\$1E11
1E1C	00790	BKSPCE EQU	\$1E1C
1E34	00800	DU1616 EQU	\$1E34
1E56	00810	FUDWRD EQU	\$1E56
1E70	00820	PTDWRD EQU	\$1E70
	00830		
	00840	* YACH ADDRESSES	
	00850	* SYSTEM FLAGS	
2000	00860	FLAGS EQU	\$2000
2000	00870	SCFLAG EQU	\$2000
2001	00880	STFLAG EQU	\$2001
2002	00890	BFFLAG EQU	\$2002
	00900		
	00910	* SYSTEM TEMPORARY	
	00920	* VARIABLES	
2040	00930	TMPVAR EQU	\$2040
2040	00940	CMDCNT EQU	\$2040
2041	00950	CMDVAR EQU	\$2041
2046	00960	CMDFRF EQU	\$2046
2047	00970	CMDFRR EQU	\$2047

2048	00980	CMDTOF	EQU	\$2048
2049	00990	CMDTOR	EQU	\$2049
204A	01000	CMDFRI	EQU	\$204A
204B	01010	CMDFRJ	EQU	\$204B
204C	01020	CMDTOI	EQU	\$204C
204D	01030	CMDTOJ	EQU	\$204D
204E	01040	MCHVAR	EQU	\$204E
	01050			
	01060	* SCREEN MANAGEMENT		
20C0	01070	CIDGEN	EQU	\$20C0
24C0	01080	B2SARY	EQU	\$24C0
2540	01090	B2DARY	EQU	\$2540
2580	01100	B2LARY	EQU	\$2580
25C0	01110	BSCARY	EQU	\$25C0
	01120			
	01130	* GENERAL REPORTING		
2600	01140	NOTREC	EQU	\$2600
2680	01150	NYIMSG	EQU	\$2680
	01160			
	01170	* GENERAL COMMANDS		
26C0	01180	ZERO2	EQU	\$26C0
26D0	01190	ZERO3	EQU	\$26D0
26E0	01200	ANLYZ	EQU	\$26E0
27E0	01210	ASKFR	EQU	\$27E0
28E0	01220	CLRBD	EQU	\$28E0
29E0	01230	CKMOV	EQU	\$29E0
2AE0	01240	CKPOS	EQU	\$2AE0
2BE0	01250	CSTLK	EQU	\$2BE0
2CE0	01260	CSTLQ	EQU	\$2CE0
2DE0	01270	DSPBD	EQU	\$2DE0
2EE0	01280	FLIP	EQU	\$2EE0
2FE0	01290	GSTRT	EQU	\$2FE0
30E0	01300	GSTOP	EQU	\$30E0
31E0	01310	LDGAM	EQU	\$31E0
32E0	01320	LDPOS	EQU	\$32E0
33E0	01330	NEW	EQU	\$33E0
33F0	01340	NEWPS	EQU	\$33F0
34F0	01350	LTRO2	EQU	\$34F0
3500	01360	LTRO3	EQU	\$3500
3510	01370	QUIT	EQU	\$3510
3520	01380	SCOFF	EQU	\$3520
3620	01390	SCON	EQU	\$3620
3720	01400	START	EQU	\$3720
3730	01410	STOP	EQU	\$3730
3740	01420	SVGAM	EQU	\$3740
3840	01430	SVPOS	EQU	\$3840
3940	01440	TKBAK	EQU	\$3940

			01450		
			01460	* MOVEMENT COMMANDS	
			01470	* AND CALCULATIONS	
3A40			01480	FILCVT EQU	\$3A40
3A60			01490	RNKCVT EQU	\$3A60
3A80			01500	FREQ EQU	\$3A80
3A90			01510	FRECP EQU	\$3A90
3AA0			01520	FRFR EQU	\$3AA0
3EA0			01530	FRFRE EQU	\$3EA0
3FA0			01540	FRFRP EQU	\$3FA0
40A0			01550	FRQ0 EQU	\$40A0
41A0			01560	FRQCP EQU	\$41A0
			01570		
			01580	* COMMAND MANAGEMENT	
42A0			01590	GETCMD EQU	\$42A0
43A0			01600	MATCH3 EQU	\$43A0
			01610	*MATCH4 EQU	\$4420
44A0			01620	XLTCMD EQU	\$44A0
48A0			01630	XLTCM2 EQU	\$48A0
			01640		
			01650	* YACH SUPERVISORY	
4CA0			01660	GMLOOP EQU	\$4CA0
4CE0			01670	YACH EQU	\$4CE0
			01680		
4420			01690	ORG	\$4420
			01700		
4420	34	02	01710	MATCH4 PSHS	A
			01720		
			01730	* DO THE FIRST	
			01740	* CHARACTER	
4422	B6	2041	01750	LDA	CMDVAR
4425	B1	204E	01760	CMPA	MCHVAR
			01770		
			01780	* GO IF NO MATCH	
4428	26	1C	01790	BNE	L0001
			01800		
			01810	* SECOND CHARACTER	
442A	B6	2042	01820	LDA	CMDVAR+1
442D	B1	204F	01830	CMPA	MCHVAR+1
			01840		
			01850	* GO IF NO MATCH	
4430	26	14	01860	BNE	L0001
			01870		
			01880	* THIRD CHARACTER	
4432	B6	2043	01890	LDA	CMDVAR+2
4435	B1	2050	01900	CMPA	MCHVAR+2
			01910		


```

01920 * GO IF NO MATCH
4438 26 0C 01930          BNE      L0001
01940
01950 * FOURTH CHARACTER
443A B6 2044 01960          LDA      CMDVAR+3
443D B1 2051 01970          CMPA    MCHVAR+3
01980
01990 * GO IF NO MATCH
4440 26 04 02000          BNE      L0001
02010
02020 * IT IS A MATCH4
02030 * SET THE Z-BIT
4442 1A 04 02040          ORCC    #$04
4444 20 02 02050          BRA      L0002
02060
02070 * IT IS NOT A MATCH4
02080 * CLEAR THE Z-BIT
4446 1C FB 02090 L0001  ANDCC   #$FB
02100
02110 * EXIT
4448 35 02 02120 L0002  PULS    A
444A 39 0000 02130          RTS
32767          END

```

00000 TOTAL ERRORS

=====

MATCH5 Checks for a FIVE-Character Command Match

```
00100 *****
00110 *
00120 * MATCH5.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * CHECKS FOR A
00160 * FIVE-CHARACTER
00170 * COMMAND MATCH
00180 *
00190 * ENTRY CONDITIONS
00200 * IN THE TEMPORARY
00210 * VARIABLES:
00220 *
00230 * CMDVAR =
00240 * THE COMMAND
00250 * CHARACTERS
00260 *
00270 * MCHVAR =
00280 * THE CHARACTERS
00290 * TO MATCH
00300 *
00310 * EXIT CONDITIONS:
00320 *
00330 * NO CHANGE IN
00340 *   CMDVAR OR
00350 *   MCHVAR
00360 *
00370 * IF MATCH:
00380 *   CC Z-BIT = 1
00390 *
00400 * IF NO MATCH:
00410 *   CC Z-BIT = 0
00420 *
00430 *****
00440
00450 * LOW RAM CURSOR ADDRESS
0088 00460 CURPOS EQU    $0088
00470
00480 * SCREEN ADDRESSES
00490 * START OF VIDRAM
0400 00500 VIDRAM EQU    $0400
```

	00510		
	00520	* COMMAND PROMPT	
	00530	* SCREEN LOCATION	
05AE	00540	CMDPMT EQU	\$05AE
	00550		
	00560	* SCREEN REPORTING FIELD	
05EB	00570	RPTFLD EQU	\$05EB
	00580		
	00590	* ONE BYTE PAST THE	
	00600	* END OF VIDRAM	
0600	00610	VIDEND EQU	\$0600
	00620		
	00630	* ML FOUNDATION	
	00640	* CORE ADDRESSES	
1C00	00650	REGXFR EQU	\$1C00
1C0E	00660	VIDCLS EQU	\$1C0E
1C1F	00670	PUTCHR EQU	\$1C1F
1CD2	00680	GETCHR EQU	\$1CD2
1CD5	00690	PUTBYT EQU	\$1CD5
1D17	00700	SCROLL EQU	\$1D17
1D36	00710	PUTCHA EQU	\$1D36
1D57	00720	PUTBYA EQU	\$1D57
1D8C	00730	CRLF EQU	\$1D8C
1DBD	00740	PK2PRT EQU	\$1DBD
1DDA	00750	PRT2PK EQU	\$1DDA
1DF7	00760	POLCAT EQU	\$1DF7
1E06	00770	PUTWRA EQU	\$1E06
1E11	00780	PUTWRD EQU	\$1E11
1E1C	00790	BKSPCE EQU	\$1E1C
1E34	00800	DU1616 EQU	\$1E34
1E56	00810	FUDWRD EQU	\$1E56
1E70	00820	PTDWRD EQU	\$1E70
	00830		
	00840	* YACH ADDRESSES	
	00850	* SYSTEM FLAGS	
2000	00860	FLAGS EQU	\$2000
2000	00870	SCFLAG EQU	\$2000
2001	00880	STFLAG EQU	\$2001
2002	00890	BFFLAG EQU	\$2002
	00900		
	00910	* SYSTEM TEMPORARY	
	00920	* VARIABLES	
2040	00930	TMPVAR EQU	\$2040
2040	00940	CMDCNT EQU	\$2040
2041	00950	CMDVAR EQU	\$2041
2046	00960	CMDFRF EQU	\$2046
2047	00970	CMDFRR EQU	\$2047

2048	00980	CMDTOF	EQU	\$2048
2049	00990	CMDTOR	EQU	\$2049
204A	01000	CMDFRI	EQU	\$204A
204B	01010	CMDFRJ	EQU	\$204B
204C	01020	CMDTOI	EQU	\$204C
204D	01030	CMDTOJ	EQU	\$204D
204E	01040	MCHVAR	EQU	\$204E
	01050			
	01060	* SCREEN MANAGEMENT		
20C0	01070	CIDGEN	EQU	\$20C0
24C0	01080	B2SARY	EQU	\$24C0
2540	01090	B2DARY	EQU	\$2540
2580	01100	B2LARY	EQU	\$2580
25C0	01110	BSCARY	EQU	\$25C0
	01120			
	01130	* GENERAL REPORTING		
2600	01140	NOTREC	EQU	\$2600
2680	01150	NYIMSG	EQU	\$2680
	01160			
	01170	* GENERAL COMMANDS		
26C0	01180	ZERO2	EQU	\$26C0
26D0	01190	ZERO3	EQU	\$26D0
26E0	01200	ANLYZ	EQU	\$26E0
27E0	01210	ASKFR	EQU	\$27E0
28E0	01220	CLRBD	EQU	\$28E0
29E0	01230	CKMOV	EQU	\$29E0
2AE0	01240	CKPOS	EQU	\$2AE0
2BE0	01250	CSTLK	EQU	\$2BE0
2CE0	01260	CSTLQ	EQU	\$2CE0
2DE0	01270	DSPBD	EQU	\$2DE0
2EE0	01280	FLIP	EQU	\$2EE0
2FE0	01290	GSTRT	EQU	\$2FE0
30E0	01300	GSTOP	EQU	\$30E0
31E0	01310	LDGAM	EQU	\$31E0
32E0	01320	LDPOS	EQU	\$32E0
33E0	01330	NEW	EQU	\$33E0
33F0	01340	NEWPS	EQU	\$33F0
34F0	01350	LTRO2	EQU	\$34F0
3500	01360	LTRO3	EQU	\$3500
3510	01370	QUIT	EQU	\$3510
3520	01380	SCOFF	EQU	\$3520
3620	01390	SCON	EQU	\$3620
3720	01400	START	EQU	\$3720
3730	01410	STOP	EQU	\$3730
3740	01420	SVGAM	EQU	\$3740
3840	01430	SVPOS	EQU	\$3840
3940	01440	TKBAK	EQU	\$3940

```

01450
01460 * MOVEMENT COMMANDS
01470 * AND CALCULATIONS
3A40 01480 FILCVT EQU $3A40
3A60 01490 RNKCVT EQU $3A60
3A80 01500 FRE0 EQU $3A80
3A90 01510 FRECP EQU $3A90
3AA0 01520 FRFR EQU $3AA0
3EA0 01530 FRFRE EQU $3EA0
3FA0 01540 FRFRP EQU $3FA0
40A0 01550 FRQ0 EQU $40A0
41A0 01560 FRQCP EQU $41A0
01570
01580 * COMMAND MANAGEMENT
42A0 01590 GETCMD EQU $42A0
43A0 01600 MATCH3 EQU $43A0
4420 01610 MATCH4 EQU $4420
01620 *MATCH5 EQU $4460
44A0 01630 XLTCMD EQU $44A0
48A0 01640 XLTCM2 EQU $48A0
01650
01660 * YACH SUPERVISORY
4CA0 01670 GMLOOP EQU $4CA0
4CE0 01680 YACH EQU $4CE0
01690
4460 01700 ORG $4460
01710
4460 34 02 01720 MATCH5 PSHS A
01730
01740 * DO THE FIRST
01750 * CHARACTER
4462 B6 2041 01760 LDA CMDVAR
4465 B1 204E 01770 CMPA MCHVAR
01780
01790 * GO IF NO MATCH
4468 26 24 01800 BNE L0001
01810
01820 * SECOND CHARACTER
446A B6 2042 01830 LDA CMDVAR+1
446D B1 204F 01840 CMPA MCHVAR+1
01850
01860 * GO IF NO MATCH
4470 26 1C 01870 BNE L0001
01880
01890 * THIRD CHARACTER
4472 B6 2043 01900 LDA CMDVAR+2
4475 B1 2050 01910 CMPA MCHVAR+2

```

			01920	
			01930	* GO IF NO MATCH
4478	26	14	01940	BNE L0001
			01950	
			01960	* FOURTH CHARACTER
447A	B6	2044	01970	LDA CMDVAR+3
447D	B1	2051	01980	CMPA MCHVAR+3
			01990	
			02000	* GO IF NO MATCH
4480	26	0C	02010	BNE L0001
			02020	
			02030	* FIFTH CHARACTER
4482	B6	2045	02040	LDA CMDVAR+4
4485	B1	2052	02050	CMPA MCHVAR+4
			02060	
			02070	* GO IF NO MATCH
4488	26	04	02080	BNE L0001
			02090	
			02100	* IT IS A MATCH5
			02110	* SET THE Z-BIT
448A	1A	04	02120	ORCC #\$04
448C	20	02	02130	BRA L0002
			02140	
			02150	* IT IS NOT A MATCH5
			02160	* CLEAR THE Z-BIT
448E	1C	FB	02170	L0001 ANDCC #\$FB
			02180	
			02190	* EXIT
4490	35	02	02200	L0002 PULS A
4492	39		02210	RTS
		0000	32767	END

00000 TOTAL ERRORS

=====

XLTCMD: Translate and Dispatch a Command The First Third

Originally, XLTCMD, XLTCM2, and XLTCM3 were a single file: XLTCMD.ASM. But the file was so large that it caused a "Buffer Full" error in EDTASM. It was fun figuring how and where to break the file inTO THREE PARTS.

```
00100 *****
00110 *
00120 * XLTCMD.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * TRANSLATE AND
00160 * DISPATCH A
00170 * COMMAND
00180 *
00190 * THIS IS THE FIRST
00200 * HALF OF XLTCMD -
00210 * SEE XLTCM2.ASM
00220 * FOR THE SECOND
00230 * HALF
00240 *
00250 * ENTRY CONDITIONS
00260 * IN THE TEMPORARY
00270 * VARIABLES:
00280 *
00290 * CMDCNT =
00300 *   THE NUMBER
00310 *   OF CHARACTERS
00320 *   IN THE COMMAND
00330 *
00340 * CMDVAR =
00350 *   THE COMMAND
00360 *   CHARACTERS
00370 *
00380 * EXIT CONDITIONS:
00390 *   NO CHANGE
00400 *
00410 *****
00420
00430 * LOW RAM CURSOR ADDRESS
0088 00440 CURPOS EQU $0088
00450
```

	00460	* SCREEN ADDRESSES	
	00470	* START OF VIDRAM	
0400	00480	VIDRAM EQU	\$0400
	00490		
	00500	* COMMAND PROMPT	
	00510	* SCREEN LOCATION	
05AE	00520	CMDPMT EQU	\$05AE
	00530		
	00540	* SCREEN REPORTING FIELD	
05EB	00550	RPTFLD EQU	\$05EB
	00560		
	00570	* ONE BYTE PAST THE	
	00580	* END OF VIDRAM	
0600	00590	VIDEND EQU	\$0600
	00600		
	00610	* ML FOUNDATION	
	00620	* CORE ADDRESSES	
1C00	00630	REGXFR EQU	\$1C00
1C0E	00640	VIDCLS EQU	\$1C0E
1C1F	00650	PUTCHR EQU	\$1C1F
1CD2	00660	GETCHR EQU	\$1CD2
1CD5	00670	PUTBYT EQU	\$1CD5
1D17	00680	SCROLL EQU	\$1D17
1D36	00690	PUTCHA EQU	\$1D36
1D57	00700	PUTBYA EQU	\$1D57
1D8C	00710	CRLF EQU	\$1D8C
1DBD	00720	PK2PRT EQU	\$1DBD
1DDA	00730	PRT2PK EQU	\$1DDA
1DF7	00740	POLCAT EQU	\$1DF7
1E06	00750	PUTWRA EQU	\$1E06
1E11	00760	PUTWRD EQU	\$1E11
1E1C	00770	BKSPCE EQU	\$1E1C
1E34	00780	DU1616 EQU	\$1E34
1E56	00790	FUDWRD EQU	\$1E56
1E70	00800	PTDWRD EQU	\$1E70
	00810		
	00820	* YACH ADDRESSES	
	00830	* SYSTEM FLAGS	
2000	00840	FLAGS EQU	\$2000
2000	00850	SCFLAG EQU	\$2000
2001	00860	STFLAG EQU	\$2001
2002	00870	BFFLAG EQU	\$2002
	00880		
	00890	* SYSTEM TEMPORARY	
	00900	* VARIABLES	
2040	00910	TMPVAR EQU	\$2040
2040	00920	CMDCNT EQU	\$2040

2041	00930	CMDVAR	EQU	\$2041
2046	00940	CMDFRF	EQU	\$2046
2047	00950	CMDFRR	EQU	\$2047
2048	00960	CMDTOF	EQU	\$2048
2049	00970	CMDTOR	EQU	\$2049
204A	00980	CMDFRI	EQU	\$204A
204B	00990	CMDFRJ	EQU	\$204B
204C	01000	CMDTOI	EQU	\$204C
204D	01010	CMDTOJ	EQU	\$204D
204E	01020	MCHVAR	EQU	\$204E
	01030			
	01040	* SCREEN MANAGEMENT		
20C0	01050	CIDGEN	EQU	\$20C0
24C0	01060	B2SARY	EQU	\$24C0
2540	01070	B2DARY	EQU	\$2540
2580	01080	B2LARY	EQU	\$2580
25C0	01090	BSCARY	EQU	\$25C0
	01100			
	01110	* GENERAL REPORTING		
2600	01120	NOTREC	EQU	\$2600
2680	01130	NYIMSG	EQU	\$2680
	01140			
	01150	* GENERAL COMMANDS		
26C0	01160	ZERO2	EQU	\$26C0
26D0	01170	ZERO3	EQU	\$26D0
26E0	01180	ANLYZ	EQU	\$26E0
27E0	01190	ASKFR	EQU	\$27E0
28E0	01200	CLRBD	EQU	\$28E0
29E0	01210	CKMOV	EQU	\$29E0
2AE0	01220	CKPOS	EQU	\$2AE0
2BE0	01230	CSTLK	EQU	\$2BE0
2CE0	01240	CSTLQ	EQU	\$2CE0
2DE0	01250	DSPBD	EQU	\$2DE0
2EE0	01260	FLIP	EQU	\$2EE0
2FE0	01270	GSTRT	EQU	\$2FE0
30E0	01280	GSTOP	EQU	\$30E0
31E0	01290	LDGAM	EQU	\$31E0
32E0	01300	LDPOS	EQU	\$32E0
33E0	01310	NEW	EQU	\$33E0
33F0	01320	NEWPS	EQU	\$33F0
34F0	01330	LTRO2	EQU	\$34F0
3500	01340	LTRO3	EQU	\$3500
3510	01350	QUIT	EQU	\$3510
3520	01360	SCOFF	EQU	\$3520
3620	01370	SCON	EQU	\$3620
3720	01380	START	EQU	\$3720
3730	01390	STOP	EQU	\$3730

3740	01400	SVGAM	EQU	\$3740
3840	01410	SVPOS	EQU	\$3840
3940	01420	TKBAK	EQU	\$3940
	01430			
	01440	* MOVEMENT COMMANDS		
	01450	* AND CALCULATIONS		
3A40	01460	FILCVT	EQU	\$3A40
3A60	01470	RNKCVT	EQU	\$3A60
3A80	01480	FRE0	EQU	\$3A80
3A90	01490	FRECP	EQU	\$3A90
3AA0	01500	FRFR	EQU	\$3AA0
3EA0	01510	FRFRE	EQU	\$3EA0
3FA0	01520	FRFRP	EQU	\$3FA0
40A0	01530	FRQ0	EQU	\$40A0
41A0	01540	FRQCP	EQU	\$41A0
	01550			
	01560	* COMMAND MANAGEMENT		
42A0	01570	GETCMD	EQU	\$42A0
43A0	01580	MATCH3	EQU	\$43A0
4420	01590	MATCH4	EQU	\$4420
4460	01600	MATCH5	EQU	\$4460
	01610	*XLTCMD	EQU	\$44A0
46A0	01620	XLTCM2	EQU	\$46A0
48A0	01630	XLTCM3	EQU	\$48A0
	01640			
	01650	* YACH SUPERVISORY		
4CA0	01660	GMLOOP	EQU	\$4CA0
4CE0	01670	YACH	EQU	\$4CE0
	01680			
44A0	01690		ORG	\$44A0
	01700			
44A0	34	12	01710	XLTCMD PSHS A,X
			01720	
			01730	* CLEAR THE SCREEN
			01740	* REPORTING FIELD
			01750	
			01760	* LOAD A BLANK SPACE
44A2	86	60	01770	LDA #96
			01780	
			01790	* POINT TO THE SCREEN
			01800	* REPORTING FIELD
44A4	8E	05EB	01810	LDX #RPTFLD
			01820	
			01830	* CLEAR THE FIELD
44A7	A7	80	01840	LC001 STA ,X+
			01850	
			01860	* ARE WE DONE?

44A9	8C	0600	01870	CMPX	#VIDEND	
			01880			
			01890	* GO IF NO		
44AC	25	F9	01900	BLO	LC001	
			01910			
			01920	* BEGIN TRANSLATING		
			01930	* COMMANDS		
			01940			
			01950	* L00NR = NOT RECOGNIZED		
			01960	* L00CT = CONTINUE		
			01970			
			01980	* IS IT "0-0"?		
44AE	86	30	01990	L0001 LDA	#48	0
44B0	B7	204E	02000	STA	MCHVAR	
44B3	86	2D	02010	LDA	#45	-
44B5	B7	204F	02020	STA	MCHVAR+1	
44B8	86	30	02030	LDA	#48	0
44BA	B7	2050	02040	STA	MCHVAR+2	
44BD	86	60	02050	LDA	#96	SP
44BF	B7	2051	02060	STA	MCHVAR+3	
44C2	86	60	02070	LDA	#96	SP
44C4	B7	2052	02080	STA	MCHVAR+4	
44C7	17	FF96	02090	LBSR	MATCH5	
			02100			
			02110	* GO IF NO		
44CA	26	06	02120	BNE	L0002	
			02130			
			02140	* DO IF YES		
44CC	17	E1F1	02150	LBSR	ZERO2	
44CF	16	01A3	02160	LBRA	L00CT	
			02170			
			02180	* IS IT "0-0-0"?		
44D2	86	30	02190	L0002 LDA	#48	0
44D4	B7	204E	02200	STA	MCHVAR	
44D7	86	2D	02210	LDA	#45	-
44D9	B7	204F	02220	STA	MCHVAR+1	
44DC	86	30	02230	LDA	#48	0
44DE	B7	2050	02240	STA	MCHVAR+2	
44E1	86	2D	02250	LDA	#45	-
44E3	B7	2051	02260	STA	MCHVAR+3	
44E6	86	30	02270	LDA	#48	0
44E8	B7	2052	02280	STA	MCHVAR+4	
44EB	17	FF72	02290	LBSR	MATCH5	
			02300			
			02310	* GO IF NO		
44EE	26	06	02320	BNE	L0003	
			02330			

			02340	* DO IF YES		
44F0	17	E1DD	02350	LBSR	ZERO3	
44F3	16	017F	02360	LBRA	L00CT	
			02370			
			02380	* IS IT "ANLYZ"?		
44F6	86	41	02390	L0003 LDA	#65 A	
44F8	B7	204E	02400	STA	MCHVAR	
44FB	86	4E	02410	LDA	#78 N	
44FD	B7	204F	02420	STA	MCHVAR+1	
4500	86	4C	02430	LDA	#76 L	
4502	B7	2050	02440	STA	MCHVAR+2	
4505	86	59	02450	LDA	#89 Y	
4507	B7	2051	02460	STA	MCHVAR+3	
450A	86	5A	02470	LDA	#90 Z	
450C	B7	2052	02480	STA	MCHVAR+4	
450F	17	FF4E	02490	LBSR	MATCH5	
			02500			
			02510	* GO IF NO		
4512	26	06	02520	BNE	L0004	
			02530			
			02540	* DO IF YES		
4514	17	E1C9	02550	LBSR	ANLYZ	
4517	16	015B	02560	LBRA	L00CT	
			02570			
			02580	* IS IT "ASKFR"?		
			02590	* ONLY CHECK FIRST		
			02600	* THREE CHARACTERS		
451A	86	41	02610	L0004 LDA	#65 A	
451C	B7	204E	02620	STA	MCHVAR	
451F	86	53	02630	LDA	#83 S	
4521	B7	204F	02640	STA	MCHVAR+1	
4524	86	4B	02650	LDA	#75 K	
4526	B7	2050	02660	STA	MCHVAR+2	
4529	17	FE74	02670	LBSR	MATCH3	
			02680			
			02690	* GO IF NO		
452C	26	06	02700	BNE	L0005	
			02710			
			02720	* DO IF YES		
452E	17	E2AF	02730	LBSR	ASKFR	
4531	16	0141	02740	LBRA	L00CT	
			02750			
			02760	* IS IT "CLRBD"?		
4534	86	43	02770	L0005 LDA	#67 C	
4536	B7	204E	02780	STA	MCHVAR	
4539	86	4C	02790	LDA	#76 L	
453B	B7	204F	02800	STA	MCHVAR+1	

453E	86	52	02810	LDA	#82	R
4540	B7	2050	02820	STA	MCHVAR+2	
4543	86	42	02830	LDA	#66	B
4545	B7	2051	02840	STA	MCHVAR+3	
4548	86	44	02850	LDA	#68	D
454A	B7	2052	02860	STA	MCHVAR+4	
454D	17	FF10	02870	LBSR	MATCH5	
			02880			
			02890	* GO IF NO		
4550	26	06	02900	BNE	L0006	
			02910			
			02920	* DO IF YES		
4552	17	E38B	02930	LBSR	CLRBD	
4555	16	011D	02940	LBRA	L00CT	
			02950			
			02960	* IS IT "CKMOV"?		
4558	86	43	02970	L0006 LDA	#67	C
455A	B7	204E	02980	STA	MCHVAR	
455D	86	4B	02990	LDA	#75	K
455F	B7	204F	03000	STA	MCHVAR+1	
4562	86	4D	03010	LDA	#77	M
4564	B7	2050	03020	STA	MCHVAR+2	
4567	86	4F	03030	LDA	#79	O
4569	B7	2051	03040	STA	MCHVAR+3	
456C	86	56	03050	LDA	#86	V
456E	B7	2052	03060	STA	MCHVAR+4	
4571	17	FEEC	03070	LBSR	MATCH5	
			03080			
			03090	* GO IF NO		
4574	26	06	03100	BNE	L0007	
			03110			
			03120	* DO IF YES		
4576	17	E467	03130	LBSR	CKMOV	
4579	16	00F9	03140	LBRA	L00CT	
			03150			
			03160	* IS IT "CKPOS"?		
457C	86	43	03170	L0007 LDA	#67	C
457E	B7	204E	03180	STA	MCHVAR	
4581	86	4B	03190	LDA	#75	K
4583	B7	204F	03200	STA	MCHVAR+1	
4586	86	50	03210	LDA	#80	P
4588	B7	2050	03220	STA	MCHVAR+2	
458B	86	4F	03230	LDA	#79	O
458D	B7	2051	03240	STA	MCHVAR+3	
4590	86	53	03250	LDA	#83	S
4592	B7	2052	03260	STA	MCHVAR+4	
4595	17	FEC8	03270	LBSR	MATCH5	

			03280		
			03290	* GO IF NO	
4598	26	06	03300	BNE	L0008
			03310		
			03320	* DO IF YES	
459A	17	E543	03330	LBSR	CKPOS
459D	16	00D5	03340	LBRA	L00CT
			03350		
			03360	* IS IT "CSTLK"?	
45A0	86	43	03370	L0008 LDA	#67 C
45A2	B7	204E	03380	STA	MCHVAR
45A5	86	53	03390	LDA	#83 S
45A7	B7	204F	03400	STA	MCHVAR+1
45AA	86	54	03410	LDA	#84 T
45AC	B7	2050	03420	STA	MCHVAR+2
45AF	86	4C	03430	LDA	#76 L
45B1	B7	2051	03440	STA	MCHVAR+3
45B4	86	4B	03450	LDA	#75 K
45B6	B7	2052	03460	STA	MCHVAR+4
45B9	17	FEA4	03470	LBSR	MATCH5
			03480		
			03490	* GO IF NO	
45BC	26	06	03500	BNE	L0009
			03510		
			03520	* DO IF YES	
45BE	17	E61F	03530	LBSR	CSTLK
45C1	16	00B1	03540	LBRA	L00CT
			03550		
			03560	* IS IT "CSTLQ"?	
45C4	86	43	03570	L0009 LDA	#67 C
45C6	B7	204E	03580	STA	MCHVAR
45C9	86	53	03590	LDA	#83 S
45CB	B7	204F	03600	STA	MCHVAR+1
45CE	86	54	03610	LDA	#84 T
45D0	B7	2050	03620	STA	MCHVAR+2
45D3	86	4C	03630	LDA	#76 L
45D5	B7	2051	03640	STA	MCHVAR+3
45D8	86	51	03650	LDA	#81 Q
45DA	B7	2052	03660	STA	MCHVAR+4
45DD	17	FE80	03670	LBSR	MATCH5
			03680		
			03690	* GO IF NO	
45E0	26	06	03700	BNE	L000A
			03710		
			03720	* DO IF YES	
45E2	17	E6FB	03730	LBSR	CSTLQ
45E5	16	008D	03740	LBRA	L00CT

			03750			
			03760	* IS IT "DSPBD"?		
45E8	86	44	03770	L000A LDA #68	D	
45EA	B7	204E	03780	STA MCHVAR		
45ED	86	53	03790	LDA #83	S	
45EF	B7	204F	03800	STA MCHVAR+1		
45F2	86	50	03810	LDA #80	P	
45F4	B7	2050	03820	STA MCHVAR+2		
45F7	86	42	03830	LDA #66	B	
45F9	B7	2051	03840	STA MCHVAR+3		
45FC	86	44	03850	LDA #68	D	
45FE	B7	2052	03860	STA MCHVAR+4		
4601	17	FE5C	03870	LBSR MATCH5		
			03880			
			03890	* GO IF NO		
4604	26	06	03900	BNE L000B		
			03910			
			03920	* DO IF YES		
4606	17	E7D7	03930	LBSR DSPBD		
4609	16	0069	03940	LBRA L00CT		
			03950			
			03960	* IS IT "FLIP"?		
460C	86	46	03970	L000B LDA #70	F	
460E	B7	204E	03980	STA MCHVAR		
4611	86	4C	03990	LDA #76	L	
4613	B7	204F	04000	STA MCHVAR+1		
4616	86	49	04010	LDA #73	I	
4618	B7	2050	04020	STA MCHVAR+2		
461B	86	50	04030	LDA #80	P	
461D	B7	2051	04040	STA MCHVAR+3		
4620	86	60	04050	LDA #96	SP	
4622	B7	2052	04060	STA MCHVAR+4		
4625	17	FE38	04070	LBSR MATCH5		
			04080			
			04090	* GO IF NO		
4628	26	06	04100	BNE L000C		
			04110			
			04120	* DO IF YES		
462A	17	E8B3	04130	LBSR FLIP		
462D	16	0045	04140	LBRA L00CT		
			04150			
			04160	* IS IT "GSTRT"?		
4630	86	47	04170	L000C LDA #71	G	
4632	B7	204E	04180	STA MCHVAR		
4635	86	53	04190	LDA #83	S	
4637	B7	204F	04200	STA MCHVAR+1		
463A	86	54	04210	LDA #84	T	

```

463C B7 2050 04220 STA MCHVAR+2
463F 86 52 04230 LDA #82 R
4641 B7 2051 04240 STA MCHVAR+3
4644 86 54 04250 LDA #84 T
4646 B7 2052 04260 STA MCHVAR+4
4649 17 FE14 04270 LBSR MATCH5
04280
04290 * GO IF NO
464C 26 06 04300 BNE L000D
04310
04320 * DO IF YES
464E 17 E98F 04330 LBSR GSTRT
4651 16 0021 04340 LBRA L00CT
04350
04360 * IS IT "GSTOP"?
4654 86 47 04370 L000D LDA #71 G
4656 B7 204E 04380 STA MCHVAR
4659 86 53 04390 LDA #83 S
465B B7 204F 04400 STA MCHVAR+1
465E 86 54 04410 LDA #84 T
4660 B7 2050 04420 STA MCHVAR+2
4663 86 4F 04430 LDA #79 O
4665 B7 2051 04440 STA MCHVAR+3
4668 86 50 04450 LDA #80 P
466A B7 2052 04460 STA MCHVAR+4
466D 17 FDF0 04470 LBSR MATCH5
04480
04490 * GO TO SECOND THIRD IF NO
04500 * (ALIAS L000E)
4670 26 2E 04510 BNE XLTCM2
04520
04530 * DO IF YES
4672 17 EA6B 04540 LBSR GSTOP
04550 * FALL THROUGH
04560
04570 * EXIT AND CONTINUE
4675 35 12 04580 L00CT PULS A,X
4677 39 0000 04590 RTS
32767 END

```

00000 TOTAL ERRORS

=====

XLTCM2: Translate and Dispatch a Command The Second Third

```

00100 *****
00110 *
00120 * XLTCM2.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * THIS IS THE
00160 * SECOND THIRD
00170 * OF XLTCMD
00180 *
00190 *****
00200
00210 * LOW RAM CURSOR ADDRESS
0088 00220 CURPOS EQU $0088
00230
00240 * SCREEN ADDRESSES
00250 * START OF VIDRAM
0400 00260 VIDRAM EQU $0400
00270
00280 * COMMAND PROMPT
00290 * SCREEN LOCATION
05AE 00300 CMDPMT EQU $05AE
00310
00320 * SCREEN REPORTING FIELD
05EB 00330 RPTFLD EQU $05EB
00340
00350 * ONE BYTE PAST THE
00360 * END OF VIDRAM
0600 00370 VIDEND EQU $0600
00380
00390 * ML FOUNDATION
00400 * CORE ADDRESSES
1C00 00410 REGXFR EQU $1C00
1C0E 00420 VIDCLS EQU $1C0E
1C1F 00430 PUTCHR EQU $1C1F
1CD2 00440 GETCHR EQU $1CD2
1CD5 00450 PUTBYT EQU $1CD5
1D17 00460 SCROLL EQU $1D17
1D36 00470 PUTCHA EQU $1D36
1D57 00480 PUTBYA EQU $1D57

```

1D8C	00490	CRLF	EQU	\$1D8C
1DBD	00500	PK2PRT	EQU	\$1DBD
1DDA	00510	PRT2PK	EQU	\$1DDA
1DF7	00520	POLCAT	EQU	\$1DF7
1E06	00530	PUTWRA	EQU	\$1E06
1E11	00540	PUTWRD	EQU	\$1E11
1E1C	00550	BKSPCE	EQU	\$1E1C
1E34	00560	DU1616	EQU	\$1E34
1E56	00570	FUDWRD	EQU	\$1E56
1E70	00580	PTDWRD	EQU	\$1E70
	00590			
	00600	* YACH ADDRESSES		
	00610	* SYSTEM FLAGS		
2000	00620	FLAGS	EQU	\$2000
2000	00630	SCFLAG	EQU	\$2000
2001	00640	STFLAG	EQU	\$2001
2002	00650	BFFLAG	EQU	\$2002
	00660			
	00670	* SYSTEM TEMPORARY		
	00680	* VARIABLES		
2040	00690	TMPVAR	EQU	\$2040
2040	00700	CMDCNT	EQU	\$2040
2041	00710	CMDVAR	EQU	\$2041
2046	00720	CMDFRF	EQU	\$2046
2047	00730	CMDFRR	EQU	\$2047
2048	00740	CMDTOF	EQU	\$2048
2049	00750	CMDTOR	EQU	\$2049
204A	00760	CMDFRI	EQU	\$204A
204B	00770	CMDFRJ	EQU	\$204B
204C	00780	CMDTOI	EQU	\$204C
204D	00790	CMDTOJ	EQU	\$204D
204E	00800	MCHVAR	EQU	\$204E
	00810			
	00820	* SCREEN MANAGEMENT		
20C0	00830	CIDGEN	EQU	\$20C0
24C0	00840	B2SARY	EQU	\$24C0
2540	00850	B2DARY	EQU	\$2540
2580	00860	B2LARY	EQU	\$2580
25C0	00870	BSCARY	EQU	\$25C0
	00880			
	00890	* GENERAL REPORTING		
2600	00900	NOTREC	EQU	\$2600
2680	00910	NYIMSG	EQU	\$2680
	00920			
	00930	* GENERAL COMMANDS		
26C0	00940	ZERO2	EQU	\$26C0
26D0	00950	ZERO3	EQU	\$26D0

26E0	00960	ANLYZ	EQU	\$26E0
27E0	00970	ASKFR	EQU	\$27E0
28E0	00980	CLRBD	EQU	\$28E0
29E0	00990	CKMOV	EQU	\$29E0
2AE0	01000	CKPOS	EQU	\$2AE0
2BE0	01010	CSTLK	EQU	\$2BE0
2CE0	01020	CSTLQ	EQU	\$2CE0
2DE0	01030	DSPBD	EQU	\$2DE0
2EE0	01040	FLIP	EQU	\$2EE0
2FE0	01050	GSTRT	EQU	\$2FE0
30E0	01060	GSTOP	EQU	\$30E0
31E0	01070	LDGAM	EQU	\$31E0
32E0	01080	LDPOS	EQU	\$32E0
33E0	01090	NEW	EQU	\$33E0
33F0	01100	NEWPS	EQU	\$33F0
34F0	01110	LTRO2	EQU	\$34F0
3500	01120	LTRO3	EQU	\$3500
3510	01130	QUIT	EQU	\$3510
3520	01140	SCOFF	EQU	\$3520
3620	01150	SCON	EQU	\$3620
3720	01160	START	EQU	\$3720
3730	01170	STOP	EQU	\$3730
3740	01180	SVGAM	EQU	\$3740
3840	01190	SVPOS	EQU	\$3840
3940	01200	TKBAK	EQU	\$3940
	01210			
	01220	* MOVEMENT COMMANDS		
	01230	* AND CALCULATIONS		
3A40	01240	FILCVT	EQU	\$3A40
3A60	01250	RNKCVT	EQU	\$3A60
3A80	01260	FRE0	EQU	\$3A80
3A90	01270	FRECP	EQU	\$3A90
3AA0	01280	FRFR	EQU	\$3AA0
3EA0	01290	FRFRE	EQU	\$3EA0
3FA0	01300	FRFRP	EQU	\$3FA0
40A0	01310	FRQ0	EQU	\$40A0
41A0	01320	FRQCP	EQU	\$41A0
	01330			
	01340	* COMMAND MANAGEMENT		
42A0	01350	GETCMD	EQU	\$42A0
43A0	01360	MATCH3	EQU	\$43A0
4420	01370	MATCH4	EQU	\$4420
4460	01380	MATCH5	EQU	\$4460
44A0	01390	XLTCMD	EQU	\$44A0
	01400	*XLTCM2	EQU	\$46A0
48A0	01410	XLTCM3	EQU	\$48A0
	01420			

			01430	* YACH SUPERVISORY		
	4CA0		01440	GMLOOP EQU	\$4CA0	
	4CE0		01450	YACH EQU	\$4CE0	
			01460			
46A0			01470	ORG	\$46A0	
			01480			
			01490	* IS IT "LDGAM"?		
			01500	* (ALIAS L000E)		
46A0	86	4C	01510	XLTCM2 LDA	#76	L
46A2	B7	204E	01520	STA	MCHVAR	
46A5	86	44	01530	LDA	#68	D
46A7	B7	204F	01540	STA	MCHVAR+1	
46AA	86	47	01550	LDA	#71	G
46AC	B7	2050	01560	STA	MCHVAR+2	
46AF	86	41	01570	LDA	#65	A
46B1	B7	2051	01580	STA	MCHVAR+3	
46B4	86	4D	01590	LDA	#77	M
46B6	B7	2052	01600	STA	MCHVAR+4	
46B9	17	FDA4	01610	LBSR	MATCH5	
			01620			
			01630	* GO IF NO		
46BC	26	06	01640	BNE	L000F	
			01650			
			01660	* DO IF YES		
46BE	17	EB1F	01670	LBSR	LDGAM	
46C1	16	01D1	01680	LBRA	L00CT	
			01690			
			01700	* IS IT "LDPOS"?		
46C4	86	4C	01710	L000F LDA	#76	L
46C6	B7	204E	01720	STA	MCHVAR	
46C9	86	44	01730	LDA	#68	D
46CB	B7	204F	01740	STA	MCHVAR+1	
46CE	86	50	01750	LDA	#80	P
46D0	B7	2050	01760	STA	MCHVAR+2	
46D3	86	4F	01770	LDA	#79	O
46D5	B7	2051	01780	STA	MCHVAR+3	
46D8	86	53	01790	LDA	#83	S
46DA	B7	2052	01800	STA	MCHVAR+4	
46DD	17	FD80	01810	LBSR	MATCH5	
			01820			
			01830	* GO IF NO		
46E0	26	06	01840	BNE	L0010	
			01850			
			01860	* DO IF YES		
46E2	17	EBFB	01870	LBSR	LDPOS	
46E5	16	01AD	01880	LBRA	L00CT	
			01890			

			01900	* IS IT "NEW"?		
46E8	86	4E	01910	L0010 LDA	#78	N
46EA	B7	204E	01920	STA	MCHVAR	
46ED	86	45	01930	LDA	#69	E
46EF	B7	204F	01940	STA	MCHVAR+1	
46F2	86	57	01950	LDA	#87	W
46F4	B7	2050	01960	STA	MCHVAR+2	
46F7	86	60	01970	LDA	#96	SP
46F9	B7	2051	01980	STA	MCHVAR+3	
46FC	86	60	01990	LDA	#96	SP
46FE	B7	2052	02000	STA	MCHVAR+4	
4701	17	FD5C	02010	LBSR	MATCH5	
			02020			
			02030	* GO IF NO		
4704	26	06	02040	BNE	L0011	
			02050			
			02060	* DO IF YES		
4706	17	ECD7	02070	LBSR	NEW	
4709	16	0189	02080	LBRA	L00CT	
			02090			
			02100	* IS IT "NEWPS"?		
470C	86	4E	02110	L0011 LDA	#78	N
470E	B7	204E	02120	STA	MCHVAR	
4711	86	45	02130	LDA	#69	E
4713	B7	204F	02140	STA	MCHVAR+1	
4716	86	57	02150	LDA	#87	W
4718	B7	2050	02160	STA	MCHVAR+2	
471B	86	50	02170	LDA	#80	P
471D	B7	2051	02180	STA	MCHVAR+3	
4720	86	53	02190	LDA	#83	S
4722	B7	2052	02200	STA	MCHVAR+4	
4725	17	FD38	02210	LBSR	MATCH5	
			02220			
			02230	* GO IF NO		
4728	26	06	02240	BNE	L0012	
			02250			
			02260	* DO IF YES		
472A	17	ECC3	02270	LBSR	NEWPS	
472D	16	0165	02280	LBRA	L00CT	
			02290			
			02300	* IS IT "O-O"?		
4730	86	4F	02310	L0012 LDA	#79	O
4732	B7	204E	02320	STA	MCHVAR	
4735	86	2D	02330	LDA	#45	-
4737	B7	204F	02340	STA	MCHVAR+1	
473A	86	4F	02350	LDA	#79	O
473C	B7	2050	02360	STA	MCHVAR+2	

473F	86	60	02370	LDA	#96	SP
4741	B7	2051	02380	STA	MCHVAR+3	
4744	86	60	02390	LDA	#96	SP
4746	B7	2052	02400	STA	MCHVAR+4	
4749	17	FD14	02410	LBSR	MATCH5	
			02420			
			02430	* GO IF NO		
474C	26	06	02440	BNE	L0013	
			02450			
			02460	* DO IF YES		
474E	17	ED9F	02470	LBSR	LTRO2	
4751	16	0141	02480	LBRA	L00CT	
			02490			
			02500	* IS IT "O-O-O"?		
4754	86	4F	02510	L0013 LDA	#79	O
4756	B7	204E	02520	STA	MCHVAR	
4759	86	2D	02530	LDA	#45	-
475B	B7	204F	02540	STA	MCHVAR+1	
475E	86	4F	02550	LDA	#79	O
4760	B7	2050	02560	STA	MCHVAR+2	
4763	86	2D	02570	LDA	#45	-
4765	B7	2051	02580	STA	MCHVAR+3	
4768	86	4F	02590	LDA	#79	O
476A	B7	2052	02600	STA	MCHVAR+4	
476D	17	FCF0	02610	LBSR	MATCH5	
			02620			
			02630	* GO IF NO		
4770	26	06	02640	BNE	L0014	
			02650			
			02660	* DO IF YES		
4772	17	ED8B	02670	LBSR	LTRO3	
4775	16	011D	02680	LBRA	L00CT	
			02690			
			02700	* IS IT "QUIT"?		
4778	86	51	02710	L0014 LDA	#81	Q
477A	B7	204E	02720	STA	MCHVAR	
477D	86	55	02730	LDA	#85	U
477F	B7	204F	02740	STA	MCHVAR+1	
4782	86	49	02750	LDA	#73	I
4784	B7	2050	02760	STA	MCHVAR+2	
4787	86	54	02770	LDA	#84	T
4789	B7	2051	02780	STA	MCHVAR+3	
478C	86	60	02790	LDA	#96	SP
478E	B7	2052	02800	STA	MCHVAR+4	
4791	17	FCCC	02810	LBSR	MATCH5	
			02820			
			02830	* GO IF NO		

4794	26	06	02840	BNE	L0015	
			02850			
			02860	* DO IF YES		
4796	17	ED77	02870	LBSR	QUIT	
4799	16	00F9	02880	LBRA	L00CT	
			02890			
			02900	* IS IT "SCOFF"?		
479C	86	53	02910	L0015 LDA	#83	S
479E	B7	204E	02920	STA	MCHVAR	
47A1	86	43	02930	LDA	#67	C
47A3	B7	204F	02940	STA	MCHVAR+1	
47A6	86	4F	02950	LDA	#79	O
47A8	B7	2050	02960	STA	MCHVAR+2	
47AB	86	46	02970	LDA	#70	F
47AD	B7	2051	02980	STA	MCHVAR+3	
47B0	86	46	02990	LDA	#70	F
47B2	B7	2052	03000	STA	MCHVAR+4	
47B5	17	FCA8	03010	LBSR	MATCH5	
			03020			
			03030	* GO IF NO		
47B8	26	06	03040	BNE	L0016	
			03050			
			03060	* DO IF YES		
47BA	17	ED63	03070	LBSR	SCOFF	
47BD	16	00D5	03080	LBRA	L00CT	
			03090			
			03100	* IS IT "SCON"?		
47C0	86	53	03110	L0016 LDA	#83	S
47C2	B7	204E	03120	STA	MCHVAR	
47C5	86	43	03130	LDA	#67	C
47C7	B7	204F	03140	STA	MCHVAR+1	
47CA	86	4F	03150	LDA	#79	O
47CC	B7	2050	03160	STA	MCHVAR+2	
47CF	86	4E	03170	LDA	#78	N
47D1	B7	2051	03180	STA	MCHVAR+3	
47D4	86	60	03190	LDA	#96	SP
47D6	B7	2052	03200	STA	MCHVAR+4	
47D9	17	FC84	03210	LBSR	MATCH5	
			03220			
			03230	* GO IF NO		
47DC	26	06	03240	BNE	L0017	
			03250			
			03260	* DO IF YES		
47DE	17	EE3F	03270	LBSR	SCON	
47E1	16	00B1	03280	LBRA	L00CT	
			03290			
			03300	* IS IT "START"?		

47E4	86	53	03310	L0017	LDA	#83	S
47E6	B7	204E	03320		STA	MCHVAR	
47E9	86	54	03330		LDA	#84	T
47EB	B7	204F	03340		STA	MCHVAR+1	
47EE	86	41	03350		LDA	#65	A
47F0	B7	2050	03360		STA	MCHVAR+2	
47F3	86	52	03370		LDA	#82	R
47F5	B7	2051	03380		STA	MCHVAR+3	
47F8	86	54	03390		LDA	#84	T
47FA	B7	2052	03400		STA	MCHVAR+4	
47FD	17	FC60	03410		LBSR	MATCH5	
			03420				
			03430		* GO IF NO		
4800	26	06	03440		BNE	L0018	
			03450				
			03460		* DO IF YES		
4802	17	EF1B	03470		LBSR	START	
4805	16	008D	03480		LBRA	L00CT	
			03490				
			03500		* IS IT "STOP"?		
4808	86	53	03510	L0018	LDA	#83	S
480A	B7	204E	03520		STA	MCHVAR	
480D	86	54	03530		LDA	#84	T
480F	B7	204F	03540		STA	MCHVAR+1	
4812	86	4F	03550		LDA	#79	O
4814	B7	2050	03560		STA	MCHVAR+2	
4817	86	50	03570		LDA	#80	P
4819	B7	2051	03580		STA	MCHVAR+3	
481C	86	60	03590		LDA	#96	SP
481E	B7	2052	03600		STA	MCHVAR+4	
4821	17	FC3C	03610		LBSR	MATCH5	
			03620				
			03630		* GO IF NO		
4824	26	06	03640		BNE	L0019	
			03650				
			03660		* DO IF YES		
4826	17	EF07	03670		LBSR	STOP	
4829	16	0069	03680		LBRA	L00CT	
			03690				
			03700		* IS IT "SVGAM"?		
482C	86	53	03710	L0019	LDA	#83	S
482E	B7	204E	03720		STA	MCHVAR	
4831	86	56	03730		LDA	#86	V
4833	B7	204F	03740		STA	MCHVAR+1	
4836	86	47	03750		LDA	#71	G
4838	B7	2050	03760		STA	MCHVAR+2	
483B	86	41	03770		LDA	#65	A

483D	B7	2051	03780	STA	MCHVAR+3	
4840	86	4D	03790	LDA	#77	M
4842	B7	2052	03800	STA	MCHVAR+4	
4845	17	FC18	03810	LBSR	MATCH5	
			03820			
			03830	* GO IF NO		
4848	26	06	03840	BNE	L001A	
			03850			
			03860	* DO IF YES		
484A	17	EEF3	03870	LBSR	SVGAM	
484D	16	0045	03880	LBRA	L00CT	
			03890			
			03900	* IS IT "SVPOS"?		
4850	86	53	03910	L001A LDA	#83	S
4852	B7	204E	03920	STA	MCHVAR	
4855	86	56	03930	LDA	#86	V
4857	B7	204F	03940	STA	MCHVAR+1	
485A	86	50	03950	LDA	#80	P
485C	B7	2050	03960	STA	MCHVAR+2	
485F	86	4F	03970	LDA	#79	O
4861	B7	2051	03980	STA	MCHVAR+3	
4864	86	53	03990	LDA	#83	S
4866	B7	2052	04000	STA	MCHVAR+4	
4869	17	FBF4	04010	LBSR	MATCH5	
			04020			
			04030	* GO IF NO		
486C	26	06	04040	BNE	L001B	
			04050			
			04060	* DO IF YES		
486E	17	EFCF	04070	LBSR	SVPOS	
4871	16	0021	04080	LBRA	L00CT	
			04090			
			04100	* IS IT "TKBAK"?		
4874	86	54	04110	L001B LDA	#84	T
4876	B7	204E	04120	STA	MCHVAR	
4879	86	4B	04130	LDA	#75	K
487B	B7	204F	04140	STA	MCHVAR+1	
487E	86	42	04150	LDA	#66	B
4880	B7	2050	04160	STA	MCHVAR+2	
4883	86	41	04170	LDA	#65	A
4885	B7	2051	04180	STA	MCHVAR+3	
4888	86	4B	04190	LDA	#75	K
488A	B7	2052	04200	STA	MCHVAR+4	
488D	17	FBD0	04210	LBSR	MATCH5	
			04220			
			04230	* GO TO THIRD THIRD IF NO		
			04240	* (ALIAS L001C)		

```
4890 26 0E      04250      BNE      XLTCM3
                04260
                04270 * DO IF YES
4892 17 F0AB    04280      LBSR      TKBAK
                04290 * FALL THROUGH
                04300
                04310 * EXIT AND CONTINUE
4895 35 12      04320 L00CT    PULS      A,X
4897 39          04330      RTS
                0000      32767      END
```

00000 TOTAL ERRORS

=====

XLTCM3: Translate and Dispatch a Command The Final Third

```

00100 *****
00110 *
00120 * XLTCM3.ASM
00130 * MDJ 2021/09/22
00140 *
00150 * THIS IS THE
00160 * THIRD THIRD
00170 * OF XLTCMD
00180 *
00190 *****
00200
00210 * LOW RAM CURSOR ADDRESS
0088 00220 CURPOS EQU $0088
00230
00240 * SCREEN ADDRESSES
00250 * START OF VIDRAM
0400 00260 VIDRAM EQU $0400
00270
00280 * COMMAND PROMPT
00290 * SCREEN LOCATION
05AE 00300 CMDPMT EQU $05AE
00310
00320 * SCREEN REPORTING FIELD
05EB 00330 RPTFLD EQU $05EB
00340
00350 * ONE BYTE PAST THE
00360 * END OF VIDRAM
0600 00370 VIDEND EQU $0600
00380
00390 * ML FOUNDATION
00400 * CORE ADDRESSES
1C00 00410 REGXFR EQU $1C00
1C0E 00420 VIDCLS EQU $1C0E
1C1F 00430 PUTCHR EQU $1C1F
1CD2 00440 GETCHR EQU $1CD2
1CD5 00450 PUTBYT EQU $1CD5
1D17 00460 SCROLL EQU $1D17
1D36 00470 PUTCHA EQU $1D36
1D57 00480 PUTBYA EQU $1D57

```

1D8C	00490	CRLF	EQU	\$1D8C
1DBD	00500	PK2PRT	EQU	\$1DBD
1DDA	00510	PRT2PK	EQU	\$1DDA
1DF7	00520	POLCAT	EQU	\$1DF7
1E06	00530	PUTWRA	EQU	\$1E06
1E11	00540	PUTWRD	EQU	\$1E11
1E1C	00550	BKSPCE	EQU	\$1E1C
1E34	00560	DU1616	EQU	\$1E34
1E56	00570	FUDWRD	EQU	\$1E56
1E70	00580	PTDWRD	EQU	\$1E70
	00590			
	00600	* YACH ADDRESSES		
	00610	* SYSTEM FLAGS		
2000	00620	FLAGS	EQU	\$2000
2000	00630	SCFLAG	EQU	\$2000
2001	00640	STFLAG	EQU	\$2001
2002	00650	BFFLAG	EQU	\$2002
	00660			
	00670	* SYSTEM TEMPORARY		
	00680	* VARIABLES		
2040	00690	TMPVAR	EQU	\$2040
2040	00700	CMDCNT	EQU	\$2040
2041	00710	CMDVAR	EQU	\$2041
2046	00720	CMDFRF	EQU	\$2046
2047	00730	CMDFRR	EQU	\$2047
2048	00740	CMDTOF	EQU	\$2048
2049	00750	CMDTOR	EQU	\$2049
204A	00760	CMDFRI	EQU	\$204A
204B	00770	CMDFRJ	EQU	\$204B
204C	00780	CMDTOI	EQU	\$204C
204D	00790	CMDTOJ	EQU	\$204D
204E	00800	MCHVAR	EQU	\$204E
	00810			
	00820	* SCREEN MANAGEMENT		
20C0	00830	CIDGEN	EQU	\$20C0
24C0	00840	B2SARY	EQU	\$24C0
2540	00850	B2DARY	EQU	\$2540
2580	00860	B2LARY	EQU	\$2580
25C0	00870	BSCARY	EQU	\$25C0
	00880			
	00890	* GENERAL REPORTING		
2600	00900	NOTREC	EQU	\$2600
2680	00910	NYIMSG	EQU	\$2680
	00920			
	00930	* GENERAL COMMANDS		
26C0	00940	ZERO2	EQU	\$26C0
26D0	00950	ZERO3	EQU	\$26D0

26E0	00960	ANLYZ	EQU	\$26E0
27E0	00970	ASKFR	EQU	\$27E0
28E0	00980	CLRBD	EQU	\$28E0
29E0	00990	CKMOV	EQU	\$29E0
2AE0	01000	CKPOS	EQU	\$2AE0
2BE0	01010	CSTLK	EQU	\$2BE0
2CE0	01020	CSTLQ	EQU	\$2CE0
2DE0	01030	DSPBD	EQU	\$2DE0
2EE0	01040	FLIP	EQU	\$2EE0
2FE0	01050	GSTRT	EQU	\$2FE0
30E0	01060	GSTOP	EQU	\$30E0
31E0	01070	LDGAM	EQU	\$31E0
32E0	01080	LDPOS	EQU	\$32E0
33E0	01090	NEW	EQU	\$33E0
33F0	01100	NEWPS	EQU	\$33F0
34F0	01110	LTRO2	EQU	\$34F0
3500	01120	LTRO3	EQU	\$3500
3510	01130	QUIT	EQU	\$3510
3520	01140	SCOFF	EQU	\$3520
3620	01150	SCON	EQU	\$3620
3720	01160	START	EQU	\$3720
3730	01170	STOP	EQU	\$3730
3740	01180	SVGAM	EQU	\$3740
3840	01190	SVPOS	EQU	\$3840
3940	01200	TKBAK	EQU	\$3940
	01210			
	01220	* MOVEMENT COMMANDS		
	01230	* AND CALCULATIONS		
3A40	01240	FILCVT	EQU	\$3A40
3A60	01250	RNKCVT	EQU	\$3A60
3A80	01260	FRE0	EQU	\$3A80
3A90	01270	FRECP	EQU	\$3A90
3AA0	01280	FRFR	EQU	\$3AA0
3EA0	01290	FRFRE	EQU	\$3EA0
3FA0	01300	FRFRP	EQU	\$3FA0
40A0	01310	FRQ0	EQU	\$40A0
41A0	01320	FRQCP	EQU	\$41A0
	01330			
	01340	* COMMAND MANAGEMENT		
42A0	01350	GETCMD	EQU	\$42A0
43A0	01360	MATCH3	EQU	\$43A0
4420	01370	MATCH4	EQU	\$4420
4460	01380	MATCH5	EQU	\$4460
44A0	01390	XLTCMD	EQU	\$44A0
46A0	01400	XLTCM2	EQU	\$46A0
	01410	*XLTCM3	EQU	\$48A0
	01420			

	01430	* YACH SUPERVISORY	
4CA0	01440	GMLOOP EQU	\$4CA0
4CE0	01450	YACH EQU	\$4CE0
	01460		
48A0	01470	ORG	\$48A0
	01480		
	01490	* STANDARD MOVE	
	01500	* ANALYSIS	
	01510		
	01520	* REFERENCE:	
	01530		
	01540	* FILCVT = FILE	
	01550	* CHARACTER	
	01560	* CONVERTER	
	01570		
	01580	* RNKCVT = RANK	
	01590	* CHARACTER	
	01600	* CONVERTER	
	01610		
	01620	* CMDFRF = "FROM"	
	01630	* FILE CHARACTER	
	01640		
	01650	* CMDFRR = "FROM"	
	01660	* RANK CHARACTER	
	01670		
	01680	* CMDTOF = "TO"	
	01690	* FILE CHARACTER	
	01700		
	01710	* CMDTOR = "TO"	
	01720	* RANK CHARACTER	
	01730		
	01740	* CMDFRI = "FROM"	
	01750	* I-VALUE	
	01760		
	01770	* CMDFRJ = "FROM"	
	01780	* J-VALUE	
	01790		
	01800	* CMDTOI = "TO"	
	01810	* I-VALUE	
	01820		
	01830	* CMDTOJ = "TO"	
	01840	* J-VALUE	
	01850		
	01860	* IS THE FIRST	
	01870	* CHARACTER A VALID	
	01880	* FILE CHARACTER?	
	01890	* (ALIAS L001C)	

48A0	B6	2041	01900	XLTCM3	LDA	CMDVAR	
48A3	B7	2046	01910		STA	CMDFRF	
48A6	17	F197	01920		LBSR	FILCVT	
48A9	34	01	01930		PSHS	CC	
48AB	B7	204A	01940		STA	CMDFRI	
48AE	35	01	01950		PULS	CC	
			01960				
			01970	* GO IF YES			
48B0	27	03	01980		BEQ	L001D	
			01990				
			02000	* GO IF NO (NOTREC)			
48B2	16	0104	02010		LBRA	L00NR	
			02020				
			02030	* IS THE SECOND			
			02040	* CHARACTER A VALID			
			02050	* RANK CHARACTER?			
48B5	B6	2042	02060	L001D	LDA	CMDVAR+1	
48B8	B7	2047	02070		STA	CMDFRF	
48BB	17	F1A2	02080		LBSR	RNKCVT	
48BE	34	01	02090		PSHS	CC	
48C0	B7	204B	02100		STA	CMDFRJ	
48C3	35	01	02110		PULS	CC	
			02120				
			02130	* GO IF YES			
48C5	1027	0003	02140		LBEQ	L001E	
			02150				
			02160	* GO IF NO (NOTREC)			
48C9	16	00ED	02170		LBRA	L00NR	
			02180				
			02190	* FR= CHECK			
			02200	* IS THE THIRD			
			02210	* CHARACTER			
			02220	* AN EQUALS SIGN?			
48CC	B6	2043	02230	L001E	LDA	CMDVAR+2	
48CF	81	3D	02240		CMPA	#61	=
			02250				
			02260	* GO IF NO			
			02270	* (CHECK FOR FRQ VERSION)			
48D1	1026	003F	02280		LBNE	L0022	
			02290				
			02300	* FR=0 CHECK			
			02310	* IS THE FOURTH			
			02320	* CHARACTER			
			02330	* A ZERO?			
48D5	B6	2043	02340		LDA	CMDVAR+2	
48D8	81	30	02350		CMPA	#48	0
			02360				

			02370	*	GO IF NO			
			02380	*	(CHECK FOR CP)			
48DA	26	06	02390		BNE	L001F		
			02400					
			02410	*	DO IF YES			
			02420	*	IT IS FR=0			
48DC	17	F1A1	02430		LBSR	FRE0		
48DF	16	00DA	02440		LBRA	L00CT		
			02450					
			02460	*	FR=CP CHECK			
			02470	*	IS THE FOURTH			
			02480	*	CHARACTER			
			02490	*	A "W"			
48E2	B6	2043	02500	L001F	LDA	CMDVAR+2		
48E5	81	57	02510		CMPA	#87	W	
			02520					
			02530	*	GO IF YES			
48E7	27	07	02540		BEQ	L0020		
			02550					
			02560	*	IS THE FOURTH			
			02570	*	CHARACTER			
			02580	*	A "B"			
48E9	81	42	02590		CMPA	#66	B	
			02600					
			02610	*	GO IF YES			
48EB	27	03	02620		BEQ	L0020		
			02630					
			02640	*	GO IF NO (NOTREC)			
48ED	16	00C9	02650		LBRA	L00NR		
			02660					
			02670	*	IS THE FIFTH			
			02680	*	CHARACTER			
			02690	*	A "K"			
48F0	B6	2043	02700	L0020	LDA	CMDVAR+2		
48F3	81	4B	02710		CMPA	#75	K	
			02720					
			02730	*	GO IF YES			
48F5	27	17	02740		BEQ	L0021		
			02750					
			02760	*	IS THE FIFTH			
			02770	*	CHARACTER			
			02780	*	A "Q"			
48F7	81	51	02790		CMPA	#81	Q	
			02800					
			02810	*	GO IF YES			
48F9	27	13	02820		BEQ	L0021		
			02830					

			02840	*	IS THE FIFTH			
			02850	*	CHARACTER			
			02860	*	A "R"			
48FB	81	52	02870		CMPA	#82		R
			02880					
			02890	*	GO IF YES			
48FD	27	0F	02900		BEQ	L0021		
			02910					
			02920	*	IS THE FIFTH			
			02930	*	CHARACTER			
			02940	*	A "B"			
48FF	81	42	02950		CMPA	#66		B
			02960					
			02970	*	GO IF YES			
4901	27	0B	02980		BEQ	L0021		
			02990					
			03000	*	IS THE FIFTH			
			03010	*	CHARACTER			
			03020	*	A "N"			
4903	81	4E	03030		CMPA	#78		N
			03040					
			03050	*	GO IF YES			
4905	27	07	03060		BEQ	L0021		
			03070					
			03080	*	IS THE FIFTH			
			03090	*	CHARACTER			
			03100	*	A "P"			
4907	81	50	03110		CMPA	#80		P
			03120					
			03130	*	GO IF YES			
4909	27	03	03140		BEQ	L0021		
			03150					
			03160	*	GO IF NO (NOTREC)			
490B	16	00AB	03170		LBRA	L00NR		
			03180					
			03190	*	DO IF YES			
			03200	*	IT IS FR=CP			
490E	17	F17F	03210	L0021	LBSR	FRECP		
4911	16	00A8	03220		LBRA	L00CT		
			03230					
			03240	*	FRQ CHECK			
			03250	*	IS THE THIRD			
			03260	*	CHARACTER A "Q"?			
4914	B6	2043	03270	L0022	LDA	CMDVAR+2		
4917	81	51	03280		CMPA	#81		Q
			03290					
			03300	*	GO IF NO			

			03310	*	(RETURN TO STANDARD			
			03320	*	MOVE CHECKING)			
4919	1026	003F	03330		LBNE	L0026		
			03340					
			03350	*	FRQ0 CHECK			
			03360	*	IS THE FOURTH			
			03370	*	CHARACTER			
			03380	*	A ZERO?			
491D	B6	2043	03390		LDA	CMDVAR+2		
4920	81	30	03400		CMPA	#48	0	
			03410					
			03420	*	GO IF NO			
			03430	*	(CHECK FOR CP)			
4922	26	06	03440		BNE	L0023		
			03450					
			03460	*	DO IF YES			
			03470	*	IT IS FRQ0			
4924	17	F779	03480		LBSR	FRQ0		
4927	16	0092	03490		LBRA	L00CT		
			03500					
			03510	*	FRQCP CHECK			
			03520	*	IS THE FOURTH			
			03530	*	CHARACTER			
			03540	*	A "W" (I.E. WHITE)			
492A	B6	2043	03550	L0023	LDA	CMDVAR+2		
492D	81	57	03560		CMPA	#87	W	
			03570					
			03580	*	GO IF YES			
492F	27	07	03590		BEQ	L0024		
			03600					
			03610	*	IS THE FOURTH			
			03620	*	CHARACTER			
			03630	*	A "B" (I.E. BLACK)			
4931	81	42	03640		CMPA	#66	B	
			03650					
			03660	*	GO IF YES			
4933	27	03	03670		BEQ	L0024		
			03680					
			03690	*	GO IF NO (NOTREC)			
4935	16	0081	03700		LBRA	L00NR		
			03710					
			03720	*	IS THE FIFTH			
			03730	*	CHARACTER A VALID			
			03740	*	PIECE IDENTIFIER?			
			03750	*	IS THE FIFTH			
			03760	*	CHARACTER			
			03770	*	A "K"			

4938	B6	2043	03780	L0024	LDA	CMDVAR+2		
493B	81	4B	03790		CMPA	#75	K	
			03800					
			03810		* GO IF YES			
493D	27	17	03820		BEQ	L0025		
			03830					
			03840		* IS THE FIFTH			
			03850		* CHARACTER			
			03860		* A "Q"			
493F	81	51	03870		CMPA	#81	Q	
			03880					
			03890		* GO IF YES			
4941	27	13	03900		BEQ	L0025		
			03910					
			03920		* IS THE FIFTH			
			03930		* CHARACTER			
			03940		* AN "R"			
4943	81	52	03950		CMPA	#82	R	
			03960					
			03970		* GO IF YES			
4945	27	0F	03980		BEQ	L0025		
			03990					
			04000		* IS THE FIFTH			
			04010		* CHARACTER			
			04020		* A "B"			
4947	81	42	04030		CMPA	#66	B	
			04040					
			04050		* GO IF YES			
4949	27	0B	04060		BEQ	L0025		
			04070					
			04080		* IS THE FIFTH			
			04090		* CHARACTER			
			04100		* AN "N"			
494B	81	4E	04110		CMPA	#78	N	
			04120					
			04130		* GO IF YES			
494D	27	07	04140		BEQ	L0025		
			04150					
			04160		* IS THE FIFTH			
			04170		* CHARACTER			
			04180		* A "P"			
494F	81	50	04190		CMPA	#80	P	
			04200					
			04210		* GO IF YES			
4951	27	03	04220		BEQ	L0025		
			04230					
			04240		* GO IF NO (NOTREC)			

4953	16	0063	04250	LBRA	L00NR
			04260		
			04270	* DO IF YES	
			04280	* IT IS FRQCP	
4956	17	F847	04290	L0025 LBSR	FRQCP
4959	16	0060	04300	LBRA	L00CT
			04310		
			04320	* CONTINUE CHECKING	
			04330	* STANDARD MOVES	
			04340	* IS THE THIRD	
			04350	* CHARACTER A VALID	
			04360	* FILE CHARACTER?	
495C	B6	2043	04370	L0026 LDA	CMDVAR+2
495F	B7	2048	04380	STA	CMDTOF
4962	17	F0DB	04390	LBSR	FILCVT
4965	34	01	04400	PSHS	CC
4967	B7	204C	04410	STA	CMDTOI
496A	35	01	04420	PULS	CC
			04430		
			04440	* GO IF YES	
496C	27	03	04450	BEQ	L0027
			04460		
			04470	* GO IF NO (NOTREC)	
496E	16	0048	04480	LBRA	L00NR
			04490		
			04500	* IS THE FOURTH	
			04510	* CHARACTER A VALID	
			04520	* RANK CHARACTER?	
4971	B6	2044	04530	L0027 LDA	CMDVAR+3
4974	B7	2049	04540	STA	CMDTOR
4977	17	F0E6	04550	LBSR	RNKCVT
497A	34	01	04560	PSHS	CC
497C	B7	204D	04570	STA	CMDTOJ
497F	35	01	04580	PULS	CC
			04590		
			04600	* GO IF YES	
4981	27	03	04610	BEQ	L0028
			04620		
			04630	* GO IF NO (NOTREC)	
4983	16	0033	04640	LBRA	L00NR
			04650		
			04660	* IS THE FIFTH	
			04670	* CHARACTER A SPACE	
4986	B6	2045	04680	L0028 LDA	CMDVAR+4
4989	81	60	04690	CMPA	#96 SP
			04700		
			04710	* GO IF NO	

498B	26	06	04720	BNE	L0029		
			04730				
			04740	* DO IF YES			
			04750	* IT IS A STANDARD MOVE			
498D	17	F110	04760	LBSR	FRFR		
4990	16	0029	04770	LBRA	L00CT		
			04780				
			04790	* IS THE FIFTH			
			04800	* CHARACTER			
			04810	* AN "E"			
			04820	* (EN PASSANT)			
4993	B6	2045	04830	L0029 LDA	CMDVAR+4		
4996	81	45	04840	CMPA	#69	E	
			04850				
			04860	* GO IF NO			
4998	26	06	04870	BNE	L002A		
			04880				
			04890	* DO IF YES			
			04900	* IT IS EN PASSANT			
499A	17	F503	04910	LBSR	FRFRE		
499D	16	001C	04920	LBRA	L00CT		
			04930				
			04940	* IS THE FIFTH			
			04950	* CHARACTER			
			04960	* A "Q", AN "R", A "B",			
			04970	* OR AN "N"?			
			04980	* (PAWN PROMOTION)			
			04990	* IS THE FIFTH			
			05000	* CHARACTER			
			05010	* A "Q"			
49A0	81	51	05020	L002A CMPA	#81	Q	
			05030				
			05040	* GO IF YES			
49A2	27	0F	05050	BEQ	L002B		
			05060				
			05070	* IS THE FIFTH			
			05080	* CHARACTER			
			05090	* AN "R"			
49A4	81	52	05100	CMPA	#82	R	
			05110				
			05120	* GO IF YES			
49A6	27	0B	05130	BEQ	L002B		
			05140				
			05150	* IS THE FIFTH			
			05160	* CHARACTER			
			05170	* A "B"			
49A8	81	42	05180	CMPA	#66	B	

```

05190
49AA 27 07 05200 * GO IF YES
05210          BEQ          L002B
05220
05230 * IS THE FIFTH
05240 * CHARACTER
05250 * AN "N"
49AC 81 4E 05260          CMPA          #78          N
05270
05280 * GO IF YES
49AE 27 03 05290          BEQ          L002B
05300
05310 * GO IF NO (NOTREC)
49B0 16 0006 05320          LBRA          L00NR
05330
05340 * DO IF YES
05350 * IT IS A
05360 * PAWN PROMOTION
49B3 17 F5EA 05370 L002B  LBSR          FRFRP
49B6 16 0003 05380          LBRA          L00CT
05390
05400 * DO THE
05410 * NOT RECOGNIZED
05420 * COMMAND
49B9 17 DC44 05430 L00NR  LBSR NOTREC
05440
05450 * EXIT AND CONTINUE
49BC 35 12 05460 L00CT  PULS          A,X
49BE 39
0000 05470          RTS
32767          END

```

00000 TOTAL ERRORS

=====

MAKECM: Make the Command Management CM.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKECM.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "GETCMD.BIN"
1130 LOADM "MATCH3.BIN"
1140 LOADM "MATCH4.BIN"
1150 LOADM "XLTCMD.BIN"
1160 LOADM "XLTCM2.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "CM.BIN", &H42A0, &H4C9F, &H42A0
1330 '
32767 END
```

=====

YACH Supervisory

This is Section Seven: YACH Supervisory

1. GMLOOP: The Game Loop
2. YACH.ASM: Game Entry Routine Set Up the Board and Enter the Game Loop
3. YACH.BAS: The Control Program

=====

GMLOOP: The Game Loop

```
00100 *****
00110 *
00120 * GMLOOP.ASM
00130 * MDJ 2021/09/23
00140 *
00150 * YACH
00160 * MAIN GAME LOOP
00170 *
00180 *****
00190
00200 * LOW RAM CURSOR ADDRESS
0088 00210 CURPOS EQU $0088
00220
00230 * SCREEN ADDRESSES
00240 * START OF VIDRAM
0400 00250 VIDRAM EQU $0400
00260
00270 * COMMAND PROMPT
00280 * SCREEN LOCATION
05AE 00290 CMDPMT EQU $05AE
00300
00310 * SCREEN REPORTING FIELD
05EB 00320 RPTFLD EQU $05EB
00330
00340 * ONE BYTE PAST THE
00350 * END OF VIDRAM
0600 00360 VIDEND EQU $0600
00370
00380 * ML FOUNDATION
00390 * CORE ADDRESSES
1C00 00400 REGXFR EQU $1C00
1C0E 00410 VIDCLS EQU $1C0E
1C1F 00420 PUTCHR EQU $1C1F
1CD2 00430 GETCHR EQU $1CD2
1CD5 00440 PUTBYT EQU $1CD5
1D17 00450 SCROLL EQU $1D17
1D36 00460 PUTCHA EQU $1D36
1D57 00470 PUTBYA EQU $1D57
1D8C 00480 CRLF EQU $1D8C
1DBD 00490 PK2PRT EQU $1DBD
1DDA 00500 PRT2PK EQU $1DDA
1DF7 00510 POLCAT EQU $1DF7
1E06 00520 PUTWRA EQU $1E06
1E11 00530 PUTWRD EQU $1E11
```

1E1C	00540	BKSPCE	EQU	\$1E1C
1E34	00550	DU1616	EQU	\$1E34
1E56	00560	FUDWRD	EQU	\$1E56
1E70	00570	PTDWRD	EQU	\$1E70
	00580			
	00590	* YACH ADDRESSES		
	00600	* SYSTEM FLAGS		
2000	00610	FLAGS	EQU	\$2000
2000	00620	SCFLAG	EQU	\$2000
2001	00630	STFLAG	EQU	\$2001
2002	00640	BFFLAG	EQU	\$2002
	00650			
	00660	* SYSTEM TEMPORARY		
	00670	* VARIABLES		
2040	00680	TMPVAR	EQU	\$2040
2040	00690	CMDCNT	EQU	\$2040
2041	00700	CMDVAR	EQU	\$2041
2046	00710	CMDFRF	EQU	\$2046
2047	00720	CMDFRR	EQU	\$2047
2048	00730	CMDTOF	EQU	\$2048
2049	00740	CMDTOR	EQU	\$2049
204A	00750	CMDFRI	EQU	\$204A
204B	00760	CMDFRJ	EQU	\$204B
204C	00770	CMDTOI	EQU	\$204C
204D	00780	CMDTOJ	EQU	\$204D
204E	00790	MCHVAR	EQU	\$204E
	00800			
	00810	* SCREEN MANAGEMENT		
20C0	00820	CIDGEN	EQU	\$20C0
24C0	00830	B2SARY	EQU	\$24C0
2540	00840	B2DARY	EQU	\$2540
2580	00850	B2LARY	EQU	\$2580
25C0	00860	BSCARY	EQU	\$25C0
	00870			
	00880	* GENERAL REPORTING		
2600	00890	NOTREC	EQU	\$2600
2680	00900	NYIMSG	EQU	\$2680
	00910			
	00920	* GENERAL COMMANDS		
26C0	00930	ZERO2	EQU	\$26C0
26D0	00940	ZERO3	EQU	\$26D0
26E0	00950	ANLYZ	EQU	\$26E0
27E0	00960	ASKFR	EQU	\$27E0
28E0	00970	CLRBD	EQU	\$28E0
29E0	00980	CKMOV	EQU	\$29E0
2AE0	00990	CKPOS	EQU	\$2AE0
2BE0	01000	CSTLK	EQU	\$2BE0

2CE0	01010	CSTLQ	EQU	\$2CE0
2DE0	01020	DSPBD	EQU	\$2DE0
2EE0	01030	FLIP	EQU	\$2EE0
2FE0	01040	GSTRT	EQU	\$2FE0
30E0	01050	GSTOP	EQU	\$30E0
31E0	01060	LDGAM	EQU	\$31E0
32E0	01070	LDPOS	EQU	\$32E0
33E0	01080	NEW	EQU	\$33E0
33F0	01090	NEWPS	EQU	\$33F0
34F0	01100	LTRO2	EQU	\$34F0
3500	01110	LTRO3	EQU	\$3500
3510	01120	QUIT	EQU	\$3510
3520	01130	SCOFF	EQU	\$3520
3620	01140	SCON	EQU	\$3620
3720	01150	START	EQU	\$3720
3730	01160	STOP	EQU	\$3730
3740	01170	SVGAM	EQU	\$3740
3840	01180	SVPOS	EQU	\$3840
3940	01190	TKBAK	EQU	\$3940
	01200			
	01210	* MOVEMENT COMMANDS		
	01220	* AND CALCULATIONS		
3A40	01230	FILCVT	EQU	\$3A40
3A60	01240	RNKCVT	EQU	\$3A60
3A80	01250	FRE0	EQU	\$3A80
3A90	01260	FRECP	EQU	\$3A90
3AA0	01270	FRFR	EQU	\$3AA0
3EA0	01280	FRFRE	EQU	\$3EA0
3FA0	01290	FRFRP	EQU	\$3FA0
40A0	01300	FRQ0	EQU	\$40A0
41A0	01310	FRQCP	EQU	\$41A0
	01320			
	01330	* COMMAND MANAGEMENT		
42A0	01340	GETCMD	EQU	\$42A0
43A0	01350	MATCH3	EQU	\$43A0
4420	01360	MATCH4	EQU	\$4420
44A0	01370	XLTCMD	EQU	\$44A0
48A0	01380	XLTCM2	EQU	\$48A0
	01390			
	01400	* YACH SUPERVISORY		
	01410	*GMLOOP	EQU	\$4CA0
4CE0	01420	YACH	EQU	\$4CE0
	01430			
4CA0	01440		ORG	\$4CA0
	01450			
	01460	* WASH		
4CA0 17	F5FD	01470	GMLOOP LBSR	GETCMD

```
01480
01490 * RINSE
4CA3 17 F7FA 01500 LBSR XLTCMD
01510
01520 * REPEAT :-)
4CA6 20 F8 01530 BRA GMLOOP
01540
01550 * EXIT
0000 32767 END
```

00000 TOTAL ERRORS

=====

YACH.ASM: Game Entry Routine

Set Up the Board and Enter the Game Loop

```
00100 *****
00110 *
00120 * YACH.ASM
00130 * VERSION 0.0.1
00140 *
00150 * MDJ 2021/09/22
00160 *
00170 * YACH
00180 * SETUP ROUTINE
00190 *
00200 *****
00210
00220 * LOW RAM CURSOR ADDRESS
0088 00230 CURPOS EQU $0088
00240
00250 * SCREEN ADDRESSES
00260 * START OF VIDRAM
0400 00270 VIDRAM EQU $0400
00280
00290 * COMMAND PROMPT
00300 * SCREEN LOCATION
05AE 00310 CMDPMT EQU $05AE
00320
00330 * SCREEN REPORTING FIELD
05EB 00340 RPTFLD EQU $05EB
00350
00360 * ONE BYTE PAST THE
00370 * END OF VIDRAM
0600 00380 VIDEND EQU $0600
00390
00400 * ML FOUNDATION
00410 * CORE ADDRESSES
1C00 00420 REGXFR EQU $1C00
1C0E 00430 VIDCLS EQU $1C0E
1C1F 00440 PUTCHR EQU $1C1F
1CD2 00450 GETCHR EQU $1CD2
1CD5 00460 PUTBYT EQU $1CD5
1D17 00470 SCROLL EQU $1D17
1D36 00480 PUTCHA EQU $1D36
```

1D57	00490	PUTBYA	EQU	\$1D57
1D8C	00500	CRLF	EQU	\$1D8C
1DBD	00510	PK2PRT	EQU	\$1DBD
1DDA	00520	PRT2PK	EQU	\$1DDA
1DF7	00530	POLCAT	EQU	\$1DF7
1E06	00540	PUTWRA	EQU	\$1E06
1E11	00550	PUTWRD	EQU	\$1E11
1E1C	00560	BKSPCE	EQU	\$1E1C
1E34	00570	DU1616	EQU	\$1E34
1E56	00580	FUDWRD	EQU	\$1E56
1E70	00590	PTDWRD	EQU	\$1E70
	00600			
	00610	* YACH ADDRESSES		
	00620	* SYSTEM FLAGS		
2000	00630	FLAGS	EQU	\$2000
2000	00640	SCFLAG	EQU	\$2000
2001	00650	STFLAG	EQU	\$2001
2002	00660	BFFLAG	EQU	\$2002
	00670			
	00680	* SYSTEM TEMPORARY		
	00690	* VARIABLES		
2040	00700	TMPVAR	EQU	\$2040
2040	00710	CMDCNT	EQU	\$2040
2041	00720	CMDVAR	EQU	\$2041
2046	00730	CMDFRF	EQU	\$2046
2047	00740	CMDFRR	EQU	\$2047
2048	00750	CMDTOF	EQU	\$2048
2049	00760	CMDTOR	EQU	\$2049
204A	00770	CMDFRI	EQU	\$204A
204B	00780	CMDFRJ	EQU	\$204B
204C	00790	CMDTOI	EQU	\$204C
204D	00800	CMDTOJ	EQU	\$204D
204E	00810	MCHVAR	EQU	\$204E
	00820			
	00830	* SCREEN MANAGEMENT		
20C0	00840	CIDGEN	EQU	\$20C0
24C0	00850	B2SARY	EQU	\$24C0
2540	00860	B2DARY	EQU	\$2540
2580	00870	B2LARY	EQU	\$2580
25C0	00880	BSCARY	EQU	\$25C0
	00890			
	00900	* GENERAL REPORTING		
2600	00910	NOTREC	EQU	\$2600
2680	00920	NYIMSG	EQU	\$2680
	00930			
	00940	* GENERAL COMMANDS		
26C0	00950	ZERO2	EQU	\$26C0

26D0	00960	ZERO3	EQU	\$26D0
26E0	00970	ANLYZ	EQU	\$26E0
27E0	00980	ASKFR	EQU	\$27E0
28E0	00990	CLRBD	EQU	\$28E0
29E0	01000	CKMOV	EQU	\$29E0
2AE0	01010	CKPOS	EQU	\$2AE0
2BE0	01020	CSTLK	EQU	\$2BE0
2CE0	01030	CSTLQ	EQU	\$2CE0
2DE0	01040	DSPBD	EQU	\$2DE0
2EE0	01050	FLIP	EQU	\$2EE0
2FE0	01060	GSTRT	EQU	\$2FE0
30E0	01070	GSTOP	EQU	\$30E0
31E0	01080	LDGAM	EQU	\$31E0
32E0	01090	LDPOS	EQU	\$32E0
33E0	01100	NEW	EQU	\$33E0
33F0	01110	NEWPS	EQU	\$33F0
34F0	01120	LTRO2	EQU	\$34F0
3500	01130	LTRO3	EQU	\$3500
3510	01140	QUIT	EQU	\$3510
3520	01150	SCOFF	EQU	\$3520
3620	01160	SCON	EQU	\$3620
3720	01170	START	EQU	\$3720
3730	01180	STOP	EQU	\$3730
3740	01190	SVGAM	EQU	\$3740
3840	01200	SVPOS	EQU	\$3840
3940	01210	TKBAK	EQU	\$3940
	01220			
	01230	* MOVEMENT COMMANDS		
	01240	* AND CALCULATIONS		
3A40	01250	FILCVT	EQU	\$3A40
3A60	01260	RNKCVT	EQU	\$3A60
3A80	01270	FRE0	EQU	\$3A80
3A90	01280	FRECP	EQU	\$3A90
3AA0	01290	FRFR	EQU	\$3AA0
3EA0	01300	FRFRE	EQU	\$3EA0
3FA0	01310	FRFRP	EQU	\$3FA0
40A0	01320	FRQ0	EQU	\$40A0
41A0	01330	FRQCP	EQU	\$41A0
	01340			
	01350	* COMMAND MANAGEMENT		
42A0	01360	GETCMD	EQU	\$42A0
43A0	01370	MATCH3	EQU	\$43A0
4420	01380	MATCH4	EQU	\$4420
44A0	01390	XLTCMD	EQU	\$44A0
48A0	01400	XLTCM2	EQU	\$48A0
	01410			
	01420	* YACH SUPERVISORY		

```

          4CA0      01430 GMLOOP  EQU      $4CA0
          01440 *YACH    EQU      $4CE0
          01450
4CE0      01460          ORG      $4CE0
          01470
4CE0 34      12      01480 YACH    PSHS    A,X
          01490
          01500 * MAIN  SCREEN  SETUP
4CE2 17      D3DB    01510          LBSR    CIDGEN
          01520
          01530 * VERSION 0.0.1
          01540 * SCREEN MODIFICATIONS
          01550 * LOAD  BLANK  SPACE
4CE5 86      60      01560          LDA    #96
          01570
          01580 * POINT TO THE SCREEN'S
          01590 * "MOVES" FIELD
4CE7 8E      041B    01600          LDX    #$041B
          01610
          01620 * CLEAR IT
4CEA A7      80      01630 L0001  STA    ,X+
4CEC 8C      0420    01640          CMPX   #$0420
4CEF 25      F9      01650          BLO    L0001
          01660
          01670 * POINT TO THE SCREEN'S
          01680 * MOVE NUMBER FIELD
4CF1 8E      0565    01690          LDX    #$0565
          01700
          01710 * CLEAR IT
4CF4 A7      80      01720 L0002  STA    ,X+
4CF6 8C      0573    01730          CMPX   #$0573
4CF9 25      F9      01740          BLO    L0002
          01750
          01760 * UNCOMMENT THE
          01770 * FOLLOWING CODE LINE
          01780 * WHEN REBUILDING AND
          01790 * RENUMBERING AFTER
          01800 * COMPLETION OF THE
          01810 * MLF INTERRUPTS
          01820 * ADDITION PROJECT
          01830
          01840 * SET ALLRAM MODE
          01850 *          STA    $FFDF
          01860
          01870 * ENTER THE
          01880 * MAIN GAME LOOP
4CFB 17      FFA2    01890          LBSR    GMLOOP

```



```
01900
01910 * EXIT
4CFE 35 12 01920 PULS A,X
4D00 39 0000 01930 RTS
32767 END
```

00000 TOTAL ERRORS

=====

MAKEYS: Make the YACH Supervisory YS.BIN Combined File

```
1000 '*****
1010 '*
1020 '* MAKEYS.BAS
1030 '* MDJ 2021/09/27
1040 '*
1050 '*****
1060 '
1070 'SETUP MEMORY
1080 PCLEAR 1
1090 CLEAR 200, &H1C00
1100 '
1110 'LOAD YACH ROUTINES
1120 LOADM "GMLOOP.BIN"
1130 LOADM "YACH.BIN"
1300 '
1310 'SAVE COMBO ROUTINE
1320 SAVEM "YS.BIN", &H4CA0, &H4D5F, &H4CA0
1330 '
32767 END
```

=====

YACH.BAS: The Game Control Program

```
1000 '*****
1010 '*
1020 '* YACH.BAS
1025 '* VERSION 0.0.1
1026 '*
1030 '* MDJ 2021/09/18
1040 '*
1050 '* YET ANOTHER
1051 '* CHESS HANDLER
1060 '*
1070 '*****
1080 '
1090 'SETUP MEMORY
1100 PCLEAR 1
1110 CLEAR 200, &H1C00
1120 '
1130 CLS
1140 PRINT @ 105, "Y. A. C. H."
1150 PRINT @ 162, "(YET ANOTHER CHESS HANDLER)"
1160 PRINT @ 201, "VERSION 0.0.1"
1170 PRINT @ 304, "BY"
1180 PRINT @ 328, "M. DAVID JOHNSON"
1190 PRINT @ 360, "INFO@BDS-SOFT.COM"
1200 PRINT @ 395, "2021/09/29"
1210 PRINT @ 490, "PLEASE WAIT"
1220 '
1230 'LOAD ML
1231 'FOUNDATION ROUTINES
1240 LOADM "MLFCOMB1.BIN"
1250 LOADM "MLFCOMB2.BIN"
1260 '
1270 'LOAD YACH ROUTINES
1280 LOADM "FT.BIN"
1290 LOADM "SM.BIN"
1300 LOADM "GR.BIN"
1310 LOADM "GC1.BIN"
1320 LOADM "GC2.BIN"
1330 LOADM "MC.BIN"
1340 LOADM "CM.BIN"
1350 LOADM "YS.BIN"
1360 '
1370 PRINT @ 451, "PRESS ANY KEY TO CONTINUE"
1380 A$ = INKEY$
```

```
1390 IF A$ = "" GOTO 1380
1400 '
1410 'START THE GAME
1420 EXEC &H4CE0
1430 '
32767 END
```

=====

Appendix A

Decimal to Hexadecimal Conversions

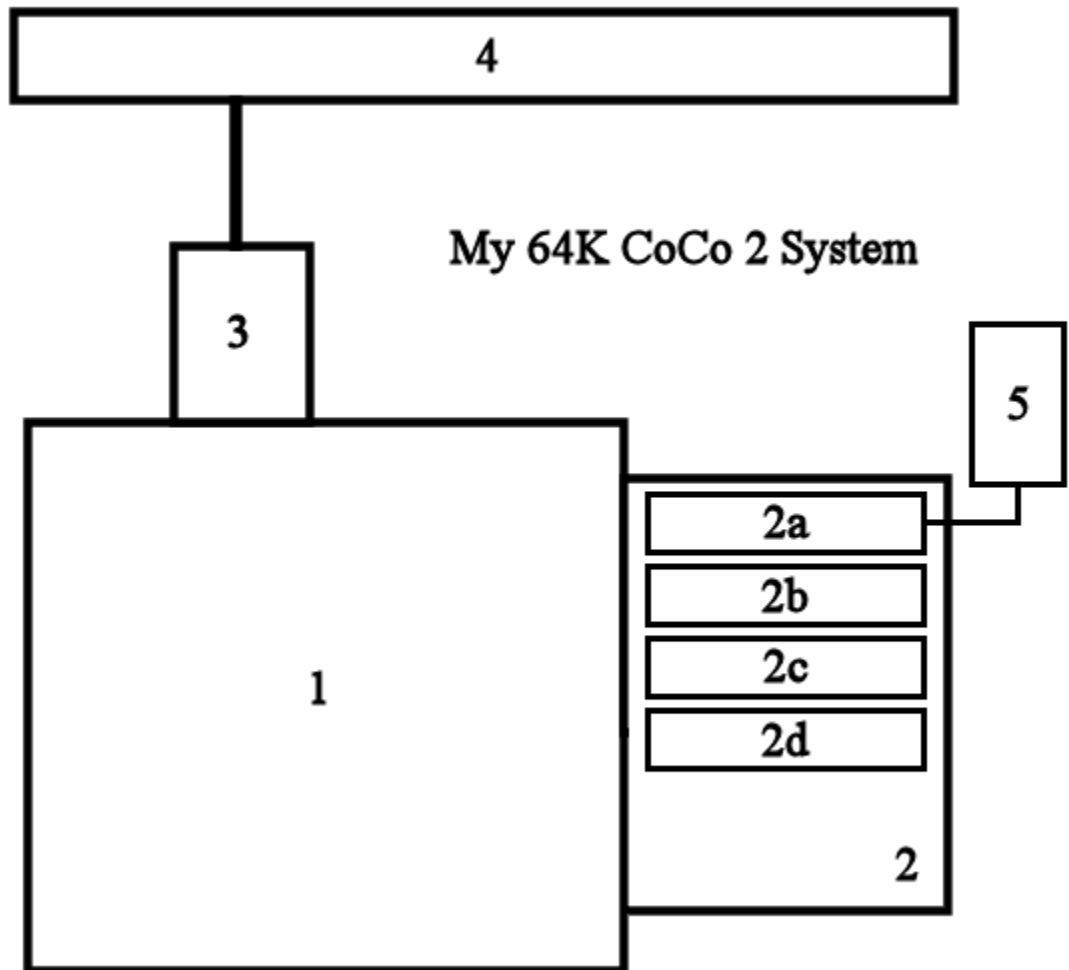
<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>
000	00	032	20	064	40	096	60
001	01	033	21	065	41	097	61
002	02	034	22	066	42	098	62
003	03	035	23	067	43	099	63
004	04	036	24	068	44	100	64
005	05	037	25	069	45	101	65
006	06	038	26	070	46	102	66
007	07	039	27	071	47	103	67
008	08	040	28	072	48	104	68
009	09	041	29	073	49	105	69
010	0A	042	2A	074	4A	106	6A
011	0B	043	2B	075	4B	107	6B
012	0C	044	2C	076	4C	108	6C
013	0D	045	2D	077	4D	109	6D
014	0E	046	2E	078	4E	110	6E
015	0F	047	2F	079	4F	111	6F
016	10	048	30	080	50	112	70
017	11	049	31	081	51	113	71
018	12	050	32	082	52	114	72
019	13	051	33	083	53	115	73
020	14	052	34	084	54	116	74
021	15	053	35	085	55	117	75
022	16	054	36	086	56	118	76
023	17	055	37	087	57	119	77
024	18	056	38	088	58	120	78
025	19	057	39	089	59	121	79
026	1A	058	3A	090	5A	122	7A
027	1B	059	3B	091	5B	123	7B
028	1C	060	3C	092	5C	124	7C
029	1D	061	3D	093	5D	125	7D
030	1E	062	3E	094	5E	126	7E
031	1F	063	3F	095	5F	127	7F

<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>	<u>DEC</u>	<u>HEX</u>
128	80	160	A0	192	C0	224	E0
129	81	161	A1	193	C1	225	E1
130	82	162	A2	194	C2	226	E2
131	83	163	A3	195	C3	227	E3
132	84	164	A4	196	C4	228	E4
133	85	165	A5	197	C5	229	E5
134	86	166	A6	198	C6	230	E6
135	87	167	A7	199	C7	231	E7
136	88	168	A8	200	C8	232	E8
137	89	169	A9	201	C9	233	E9
138	8A	170	AA	202	CA	234	EA
139	8B	171	AB	203	CB	235	EB
140	8C	172	AC	204	CC	236	EC
141	8D	173	AD	205	CD	237	ED
142	8E	174	AE	206	CE	238	EE
143	8F	175	AF	207	CF	239	EF
144	90	176	B0	208	D0	240	F0
145	91	177	B1	209	D1	241	F1
146	92	178	B2	210	D2	242	F2
147	93	179	B3	211	D3	243	F3
148	94	180	B4	212	D4	244	F4
149	95	181	B5	213	D5	245	F5
150	96	182	B6	214	D6	246	F6
151	97	183	B7	215	D7	247	F7
152	98	184	B8	216	D8	248	F8
153	99	185	B9	217	D9	249	F9
154	9A	186	BA	218	DA	250	FA
155	9B	187	BB	219	DB	251	FB
156	9C	188	BC	220	DC	252	FC
157	9D	189	BD	221	DD	253	FD
158	9E	190	BE	222	DE	254	FE
159	9F	191	BF	223	DF	255	FF

=====

Appendix B: My 64K CoCo 2 System

As of 2021/09/08, my physical 64K CoCo 2 System is depicted in the block diagram below.



See following page for System Legend.

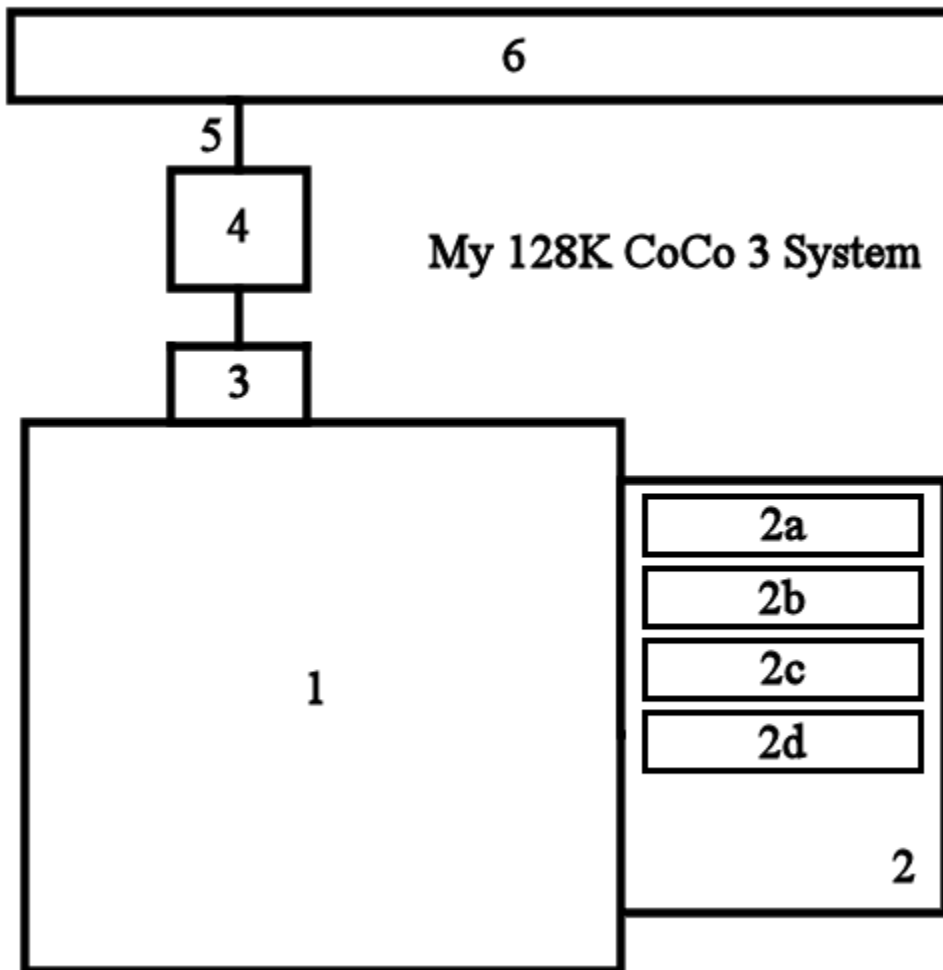
64K System Legend:

1. 64K CoCo 2, Model Number 26-3127, Serial Number 007601.
2. Multi-Pak Interface, Model Number 26-3124, Serial Number 2005259
 - 2a. Floppy Disk Controller, Model Number 26-3029
 - 2b. CoCo SDC, running SDC-DOS 1.6 CC2.
 - 2c. RS-232 Pak
 - 2d. Empty slot
3. CoCo VGA.
4. RCA 19" VGA/HDMI Monitor, Model RT1970.
5. 5.25" Disk Drive, Model Number 26-3022

=====

Appendix C: My 128K CoCo 3 System

As of 2021/09/08, my physical 128K CoCo 3 System is depicted in the block diagram below.



See following page for System Legend.

128K System Legend:

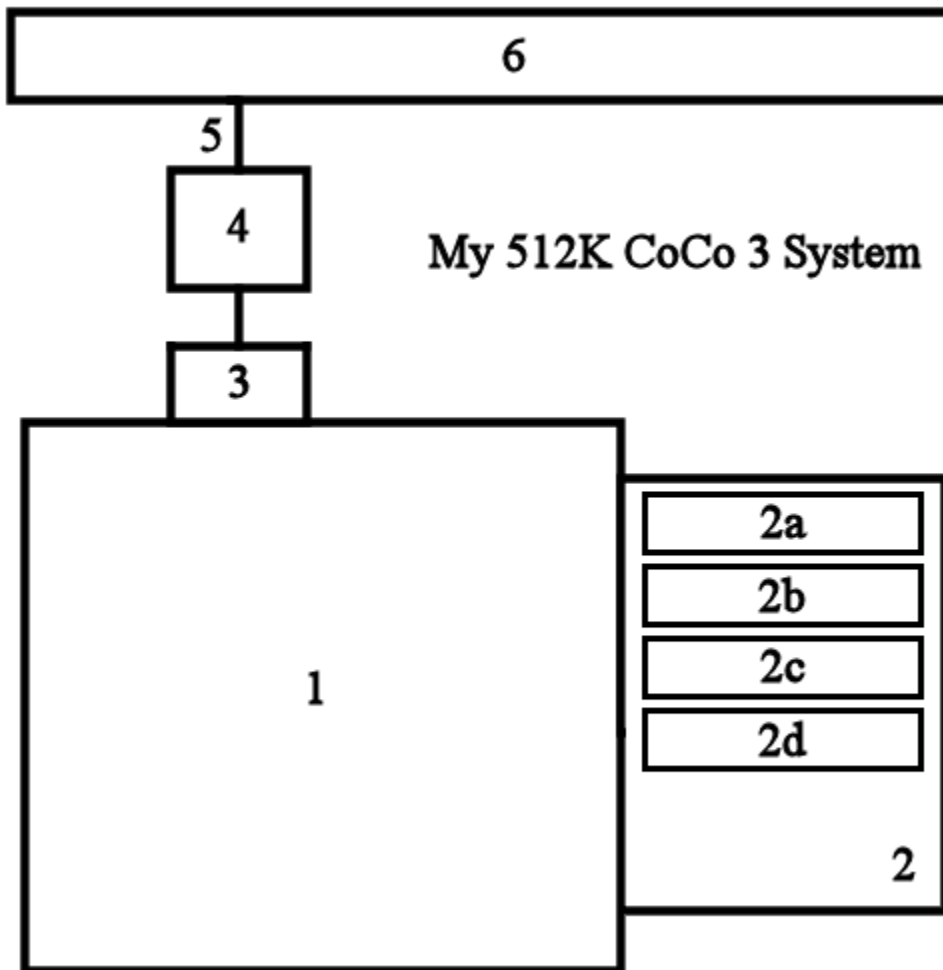
1. 128K CoCo 3, Model Number 26-3334, Serial Number 1110785
2. Mega-Mini MPI
 - 2a. CoCo SDC, running SDC-DOS 1.6 CC3
 - 2b. RS-232 Pak
 - 2c. Empty slot
 - 2d. Empty slot
3. CoCo Switch-a-Roo*
4. TNP SCART to HDMI Converter
5. HDMI Cable
6. RCA 19" VGA/HDMI Monitor, Model RT1970

*Not Shown: The Switch-a-Roo is powered from a ByEasy powered 7-port USB 2.0 hub.

=====

Appendix D: My 512K CoCo 3 System

As of 2021/09/08, my physical 512K CoCo 3 System is depicted in the block diagram below.



See following page for System Legend.

512K System Legend:

1. 512K CoCo 3, Model Number 26-3334, Serial Number 1037654
2. Mega-Mini MPI
 - 2a. CoCo SDC, running SDC-DOS 1.6 CC3
 - 2b. RS-232 Pak
 - 2c. Empty slot
 - 2d. Empty slot
3. CoCo Switch-a-Roo*
4. TNP SCART to HDMI Converter
5. HDMI Cable
6. RCA 19" VGA/HDMI Monitor, Model RT1970

*Not Shown: The Switch-a-Roo is powered from a ByEasy powered 7-port USB 2.0 hub.

=====

Appendix E: My CoCo Philosophy

The CoCo community enjoys a great diversity of interests.

Some choose to concentrate on hardware innovations and modifications such as interfacing with VGA and HDMI monitors, SD Card data storage, and 104-key keyboards. This interest is at least partly born of necessity, since composite monitors, floppy diskettes, and CoCo spare parts are no longer manufactured and are in increasingly short supply.

Others concentrate on expanding the software horizons of the CoCo 3, using NitrOS-9 and other operating systems to make the multitasking CoCo behave ever closer to modern Windows, Mac, and Linux machines.

Still others are devoted to emulating the CoCo on other platforms by developing emulators such as VCC, OVCC, MAME, and XRoar.

And some just love retro gaming.

My personal interest is twofold:

1. To see VCC increasingly used as a learning tool for budding software developers.
2. To see just how much I can cram into a 64K CoCo 2.

First, VCC: Today's Grade School, Junior High, and High School students have a wealth of available learning tools. Micro-bits, Arduinos, and Raspberry Pi supermicro devices provide highly affordable entry-level introductions to computer programming and interfacing. Maker-Spaces and Innovation Centers in our schools and libraries help foster growth and experience.

But these devices do have limitations. Even these simple(?) computers can have rather steep learning curves, and their low initial cost can quickly expand as new peripherals and experimental equipment and supplies are added.

VCC is free, and can be used on any Windows computer: just download it, install it, and it runs. If you don't own a Windows computer, your school, library, or a friend probably does. The included BASIC language is easy to learn and can readily serve as a stepping-stone towards more complex programming languages. (And, no, learning structured programming does not require a language that enforces structure. In fact, I think learning to structure your programs is actually more effective when you do so on your own.)

I prefer VCC to the other emulators for these purposes because its setup is trivial: Again, just download it, install it, and it runs. OVCC, MAME, and XRoar have their advantages, but ease of setup is not one of them. Even with their available Windows binary packages, they require pre-installation of other bits and pieces of software before they can be downloaded,

installed, and run. This may not be a major problem for a reasonably adept aficionado, but it forms a significant barrier for the newbie. And, it's the newbie whom we're trying to reach, interest, and encourage here; the newbie who may not yet recognize even the tiniest awakening of interest in things computational.

But, for these purposes, VCC has one glaring weakness: its instruction manual is woefully terse. I would like to see VCC bundled with a selection of tutorials, manuals, and examples suited to guiding even the most newbie of newbies into the wonders of computing.

Second, The Stuffed CoCo: I'm simply fascinated by the challenge of seeing how much functional capability I can sandwich into the nooks and crannies of the 64K space. Whether it's working in the available RAM left by the 32K ROM and the dedicated RAM that supports that ROM, or whether it's jumping right into ALLRAM mode and just filling the entire 64K to near-overflowing; it's an investigative gauntlet which goes right to the heart of my enchantment with puzzles in general.

It's great fun!

M.D.J. 2021/08/29

=====

Appendix F: New BDS Software License

This New Software License applies to all software found on the BDS Software site, and supersedes all previous copyright notices and licensing provisions which may appear in the software itself or in any documentation therefor.

All software which has previously been placed in the public domain remains in the public domain.

All other software, programs, experiments and reports, documentation, and any other material on this site (other than that attributed to outside sources) is hereby copyright © 2018 (or later if so marked) by M. David Johnson.

All software, documentation, and other information on the BDS Software site is available for you to freely download without cost.

Whether you downloaded such items directly from this site, or you obtained them by any other means, you are hereby licensed to copy them, to sell or give away such copies, to use them, and to excerpt from them, in any way whatsoever, so long as nothing you do with them would denigrate the name of our Lord and Savior, Jesus Christ.

I make absolutely no warranty whatsoever for any of these items. You use them entirely at your own risk.

If they don't work for you, I commiserate.

If they crash your system, I sympathize.

But I accept no responsibility whatsoever for any such consequences. Under no circumstances will BDS Software or M. David Johnson be liable for any negative results of any kind which you may experience from downloading or using these items.

BDS Software's former mail address at P.O. Box 485 in Glenview, IL is no longer valid. Any mail sent to that address will be rejected by the U.S. Postal Service. See my Contact page.

M.D.J. 2018/06/08

=====

Works Cited

- [MDJ01] Johnson, M. David. *Back To (Almost) Bare-Metal Programming*. Glenview, IL: BDS Software, 2021. Web. <http://www.bds-soft.com/cocoPapers.php> . Last Accessed 2021/09/29.
- [MDJ02] Johnson, M. David. *Key Codes and VIDRAM*. Glenview, IL: BDS Software, 2021. Web. <http://www.bds-soft.com/cocoPapers.php> . Last Accessed 2021/09/29.
- [MDJ03] Johnson, M. David. *Towards a VCC Bundle*. Glenview, IL: BDS Software, 2021. Web. <http://www.bds-soft.com/cocoPapers.php> . Last Accessed 2021/09/29.

=====

END