



VIP EXPANSION KEYPAD	VP580
VIP EXPANSION KEYPAD INTERFACE BOARD	VP585
VIP COLOR BOARD	VP590
VIP SIMPLE SOUND BOARD	VP595

INSTRUCTION MANUAL INCLUDING CHIP-8X

RCA COSMAC VIP MARKETING
New Holland Avenue
Lancaster, PA 17604

© Copyright RCA Corporation
All Rights Reserved

71L233

Information furnished by RCA is believed to be accurate and reliable. However, no responsibility is assumed by RCA for its use; nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of RCA.

CONTENTS

I. Introduction	I-1
A Note on Power Supplies	I-2
Help	I-2
II. CHIP-8X	II-1
CHIP-8X Listing	II-5
Table of CHIP-8X Instructions	II-7
III. The VP-590 Color Board	III-1
Installation	III-1
A Color Test Program	III-2
CHIP-8X Color Display Format	III-3
CHIP-8X Color Instructions	III-3
VIP Component Location Map	III-9
U4 Jumper Placement	III-10
Color Board Component Location Map	III-11
IV. The VP-595 Simple Sound Board	IV-1
Installation	IV-1
Testing	IV-1
Programming	IV-2
Miscellaneous	IV-3
Speaker Switch Connection	IV-5
Simple Sound Block Diagram	IV-6
V. The VP-580 VIP Expansion Keypads	V-1
Installation	V-1
Programming	V-3
VI. Circuit Description and Schematic Diagrams	VI-1
Color Board Description	VI-1
Color Board Schematic	VI-3
Simple Sound Circuit Description	VI-4
Simple Sound Schematic	VI-5
Table of Simple Sound Frequencies	VI-6
Expansion Keypad Interface Card Circuit Description	VI-11
Expansion Keypad Interface Card Schematic	VI-11
Expansion Keypad Circuit Description	VI-12
Expansion Keypad Schematic	VI-12
Appendix	
A. CHIP-8X Programs	A-1
Color Kaleidoscope	A-1
VIP Blockout	A-2

Chapter I

Introduction

Welcome to an expanded world of VIP fun!

This manual will show you how to install and use three expansion modules for your VIP: the VP-590 Color board, the VP-595 Simple Sound board, and the VP-580 Expansion Keypad. This manual also covers the use of the VP-585 Expansion Keypad Interface card, which you will need if you are using one or two Expansion Keypads and do not have the Color board.

All three boards are covered in one manual because they are designed to complement one another. A single game or other application can be programmed to use color, extended sound and a second keypad, so we have provided information on all three in the same manual. In Appendix A to this manual we have provided some games which use several of these features together.

To use these units together this manual contains a revised version of the CHIP-8 interpreter originally furnished with your VIP. Rather than relying on the use of specialized machine language subroutines, the new version of CHIP-8, which we have named CHIP-8X, allows you to program these new VIP capabilities just as you have programmed applications using the VIP alone.

Chapter II of this manual will show you how to modify your present version of CHIP-8 to create CHIP-8X, and will give you a brief description of the new commands. Chapter III explains how to install and use the Color board, Chapter IV covers the Simple Sound board, and Chapter V explains the use of the Expansion Keypad. Chapter VI gives schematics, parts lists, and a brief theory of operation for each module. Appendix A provides two games which use the features of these expansion modules.

Please take time to read this manual and try some of the sample programs we have provided. As always, the best way to learn is to do, so when you have read

the material on a particular module, try writing a simple program which uses the new features.

All of the programming information in this manual assumes that you already know how to use the CHIP-8 language and are familiar with the hexadecimal notation used in VIP programming. If this is not the case, we suggest that you first read the VIP User Guide and try some programming in CHIP-8 before attempting to use the extended features provided in CHIP-8X.

A Note on Power Supplies

All three modules described in this manual can safely be used simultaneously when powered from the regulated power supply provided with your VIP. If you have expanded the memory used on your VIP, you should check that your VIP, with all modules installed, draws less than .6A from the power supply. The CMOS circuits used in the VIP and its expansion modules are very tolerant of power supply variations but the memory circuits require a voltage of $5.0V \pm 5\%$ for proper operation. Excessive current drawn from the power supply will also cause it to overheat, thus shortening its operational life.

Help

We have tried to make sure that the information in this manual is as accurate as possible, but nobody is perfect. If you think you have found an error in the manual, or if the information we have provided is unclear, contact the Product Manager for VIP Products, New Holland Pike, Lancaster, PA 17604. RCA cannot provide programs for specific applications, nor can we provide "debugging" assistance on user programs.

Chapter II

CHIP-8X

CHIP-8X is an expanded version of the original CHIP-8 interpreter which allows easy control over RCA's new options for the VIP: color card, simple sound, and expansion hex keyboard. Programs written in CHIP-8X will run on a standard VIP with none of the options installed, but for full feature use the options will greatly enhance your enjoyment.

All of the instructions from the original CHIP-8, except BMMM, are valid for CHIP-8X, and CHIP-8X offers these new instructions:

Instruction	Function
FXF8	VX → output port (used to program simple sound)
FXFB	Input port → VX (waits for EF4=1)
EXF2	Skip next instruction if VX= hex keypad 2
EXF5	Skip next instruction if VX≠ hex keypad 2
BXYO	Set VY color @ VX(#H), VX+1(NV) (provides low resolution color 8x8)
BXYN	N≠0, set VY color @ VX, VX+1 byte N bytes vertically (provides high resolution color 8x32)
02A0	Steps background 1 color (→ blue → black → green → red →)
5XYI	Let VX= VX+VY (hex digits 00 to 77) (useful for manipulating the NH, NV parameters for low resolution color)

We will cover the use of these instructions in more detail in the subsequent chapters. Your first task is to enter the modifications which transform CHIP-8 into CHIP-8X. You begin this process by loading a copy of CHIP-8 into memory from tape. Loading the first two pages of any CHIP-8 game will give you the proper memory contents to begin. The new version of CHIP-8X will require three pages of memory (0000-02FF).

CHIP-8X can be created by modifying the original CHIP-8 interpreter as follows:

- 1) Load the original CHIP-8 into M(0000) - M(01FF).
- 2) Replace original codes with new codes shown below:
(Those memory addresses which must be modified if you relocate this code to another memory area are flagged with an asterisk.)

II-2

Memory Address	New Code	Original Code
0015	02	01
0018	FA	FC
0055	00	01
*005B	02	01
0065	FE	95
*006B	00	A4
00FE	05F6	0000
0100	33A4	0000
0102	3095	0000
01A4	E606	F8F0
01A6	FA77	A7E7
01A8	5607	45F4
01AA	FA77	A586
01AC	F4FA	FA0F
01AE	7756	3BB2
01B0	15D4	FC01
01B2	0000	B5D4
01F2	3788	0000
01F4	D43F	0000
01F6	88D4	0000
01F8	E663	0000
01FA	D4E6	0000
01FC	3FFC	00E0
01FE	6BD4	004B

The following color subroutine must be entered starting at M(0200).

Memory Address	Codes
0200	92BD
0202	F89FAD
0205	0D
*0206	<u>320A</u>
0208	45D4
020A	0045
020C	FA0FAF
*020F	<u>3231</u>
0211	46FA3F
0214	F6F6F6
0217	2252E2
021A	06FA1F
021D	FEFEFE
0220	F1AC12
0223	F8D0BC
0226	075C8C
0229	FC08AC
022C	2F8F
*022E	<u>3A26</u>
0230	D4
0231	07BD
0233	46AC
0235	06BC
0237	8CFA07
023A	2252E2
023D	F8C0BE
0240	9C
0241	F6F6F6F6
0245	FA07AD9C
0249	FEFEFEFEFE
024E	F4AE8C
0251	F6F6F6F6
0255	FA07AF
0258	9EB7
025A	8EA7
025C	8DBC
025E	F804AC
0261	9D5787
0264	FC08A7
0267	2C8C
*0269	<u>3A61</u>
026B	9C
*026C	<u>3273</u>
026E	FF01BC
*0271	<u>305E</u>
0273	1E8E
0275	FAE7AE
0278	8F
*0279	<u>327E</u>
027B	2F
027C	<u>3058</u>
027E	<u>12D4</u>

II-4

The following code begins at M(0280) and is a routine that scans for the color map at M(C000). If the color map is found, a "00" is stored at M(0Y9F). If no color map exists then an "01" is placed in storage and color instructions DO NOT EXECUTE. This allows color programs to be run on a standard B&W VIP. This subroutine is automatically called during CHIP-8X initialization; you need not call it from your program.

Memory Address	Code
0280	92BD
0282	F89FAD
0285	F8C0BC
0288	94AC
028A	F8AA5C
028D	94BCAC
0290	0CFB91
0293	329C
0295	F8915C
0298	F8015D
029B	D4
029C	F8005D
029F	D4

At M(02A0) there is a 4 byte subroutine called from CHIP-8X programs to switch background colors.

0240	E265
02A2	22D4

The space between M(02A5) and M(02FA) is available for other machine language subroutines and is planned to contain routines for future expansions (i.e. Joy Sticks etc.).

Finally, these CHIP-8 instructions must be placed as follows:

02FA	0280scans for color map
02FC	00E0erase display page
02FE	004BTV on

The listing below is a hex dump of CHIP-8X including the unmodified portions of CHIP-8. You can use this table as a master listing of CHIP-8X. Memory

locations 02A4 through 02F9 can have any value without affecting the operation of CHIP-8X.

CHIP-8X Listing

0000	91	66	FF	01	B2	B6	F8	0F	0100	0F	85	04	45	E6	F3	3A	82
0008	A2	F8	81	B1	F8	46	A1	90	0108	15	15	04	45	E6	F3	3A	88
0010	B4	F8	1B	A4	F8	02	B5	F8	0110	04	45	07	00	80	45	07	30
0018	FA	A5	D4	96	E7	E2	94	BC	0118	84	E6	62	26	45	A3	36	83
0020	45	AF	F6	F6	F6	F6	32	44	01A0	04	3E	88	04	E6	06	FA	77
0028	F9	50	AC	8F	FA	0F	F9	F0	01A8	56	07	FA	77	F4	FA	77	56
0030	A6	05	F6	F6	F6	F6	F9	F0	01B0	15	04	00	00	45	56	04	45
0038	A7	4C	B3	8C	FC	0F	AC	0C	01B8	E6	F4	56	04	45	FA	0F	3A
0040	A3	03	30	1B	8F	FA	0F	B3	01C0	04	07	56	04	AF	22	F8	D3
0048	45	30	40	22	69	12	04	00	01C8	73	8F	F9	F0	52	E6	07	D2
0050	00	01	01	01	01	00	01	01	01D0	56	F8	FF	A6	F8	00	7E	56
0058	01	01	01	02	01	00	01	01	01D8	04	19	89	AE	93	BE	99	EE
0060	00	7C	75	83	8B	FE	B4	B7	01E0	F4	56	76	E6	F4	B9	56	45
0068	BC	91	EB	00	D9	70	99	05	01E8	F2	56	04	45	AA	86	FA	0F
0070	06	FA	07	BE	06	FA	3F	F6	01F0	BA	04	37	88	04	3F	88	04
0078	F6	F6	22	52	07	FA	1F	FE	01F8	E6	63	04	E6	3F	FC	6B	04
0080	FE	FE	F1	AC	9B	BC	45	FA	0200	92	BD	F8	9F	AD	0D	32	0A
0088	0F	AD	A7	F8	D0	A6	93	AF	0208	45	04	00	45	FA	0F	AF	32
0090	87	32	F3	27	4A	BD	9E	AE	0210	31	46	FA	3F	F6	F6	F6	22
0098	8E	32	A4	90	F6	BD	8F	76	0218	52	E2	06	FA	1F	FE	FE	FE
00A0	AF	2E	30	98	9D	56	16	8F	0220	F1	AC	12	F8	D0	BC	07	5C
00A8	56	16	30	8E	00	EC	F8	D0	0228	8C	FC	08	AC	2F	8F	3A	26
00B0	A6	93	A7	8D	32	D9	06	F2	0230	04	07	BD	46	AC	06	BC	8C
00B8	2D	32	BE	F8	01	A7	46	F3	0238	FA	07	22	52	E2	F8	C0	BE
00C0	5C	02	FB	07	32	D2	1C	06	0240	9C	F6	F6	F6	F6	FA	07	AD
00C8	F2	32	CE	F8	01	A7	06	F3	0248	9C	FE	FE	FE	FE	FE	F4	AE
00D0	5C	2C	16	8C	FC	08	AC	3B	0250	8C	F6	F6	F6	F6	FA	07	AF
00D8	B3	F8	FF	A6	87	56	12	D4	0258	9E	B7	8E	A7	8D	BC	F8	04
00E0	9B	BF	F8	FF	AF	93	5F	8F	0260	AC	9D	57	87	FC	08	A7	2C
00E8	32	DF	2F	30	E5	00	42	B5	0268	8C	3A	61	9C	32	73	FF	01
00F0	42	A5	04	8D	A7	87	32	AC	0270	BC	30	5E	1E	8E	FA	E7	AE
00F8	2A	27	30	F5	00	00	05	F6	0278	8F	32	7E	2F	30	58	12	D4
0100	33	A4	30	95	00	45	A3	98	0280	92	BD	F8	9F	AD	F8	C0	BC
0108	56	04	F8	81	BC	F8	95	AC	0288	94	AC	F8	AA	5C	94	BC	AC
0110	22	DC	12	56	04	06	B8	D4	0290	0C	FB	91	32	9C	F8	91	5C
0118	06	A8	04	64	0A	01	E6	8A	0298	F8	01	5D	04	F8	00	5D	D4
0120	F4	AA	3B	28	9A	FC	01	8A	02A0	E2	65	22	04	00	00	00	00
0128	04	F8	81	BA	06	FA	0F	AA	02A8	00	00	00	00	00	00	00	00
0130	0A	AA	04	E6	06	BF	93	BE	02B0	00	00	00	00	00	00	00	00
0138	F8	1B	AE	2A	1A	F8	00	5A	02B8	00	00	00	00	00	00	00	00
0140	0E	F5	3B	4B	56	0A	FC	01	02C0	00	00	00	00	00	00	00	00
0148	5A	30	40	4E	F6	3B	3C	9F	02C8	00	00	00	00	00	00	00	00
0150	56	2A	2A	04	00	22	86	52	02D0	00	00	00	00	00	00	00	00
0158	F8	F0	A7	07	5A	87	F3	17	02D8	00	00	00	00	00	00	00	00
0160	1A	3A	5B	12	04	22	86	52	02E0	00	00	00	00	00	00	00	00
0168	F8	F0	A7	0A	57	87	F3	17	02E8	00	00	00	00	00	00	00	00
0170	1A	3A	6B	12	04	15	85	22	02F0	00	00	00	00	00	00	00	00
0178	73	95	52	25	45	A5	86	FA	02F8	00	00	02	80	00	E0	00	4B

II-6

After you have loaded CHIP-8X and verified your entries, save a copy of CHIP-8X on tape. Remember to save three pages, not two.

Table I - CHIP-8X Instructions

1MMM	Go to OMMM
2MMM	Do subroutine at OMMM (must end with 00EE)
00EE	Return from subroutine
3XKK	Skip next instruction if $VX=KK$
4XKK	Skip next instruction if $VX \neq KK$
5XY0	Skip next instruction if $VX=VY$
5XY1	Let $VX=VX+VY$ (hex digits 00 to 77)
9XY0	Skip next instruction if $VX \neq VY$
EX9E	Skip next instruction if $VX=$ Hex key #1 (LSD)
EXA1	Skip next instruction if $VX \neq$ Hex key #1 (LSD)
EXF2	Skip next instruction if $VX=$ Hex key #2 (LSD)
EXF5	Skip next instruction if $VX \neq$ Hex key #2 (LSD)
6XKK	Let $VX=KK$
CXKK	Let $VX=$ Random Byte ($KK=$ Mask)
7XKK	Let $VX=VX+KK$
8XY0	Let $VX=VY$
8XY1	Let $VX=VX/VY$ (VF changed)
8XY2	Let $VX=VX \& VY$ (VF changed)
8XY4	Let $VX=VX+VY$ (VF=00 if $VX+VY \leq FF$, VF=01 if $VX+VY > FF$)
8XY5	Let $VX=VX-VY$ (VF=00 if $VX < VY$, VF=01 if $VX > XY$)
FX07	Let $VX=$ current timer value
FX0A	Let $VX=$ hex key digit (waits for any key pressed #1)
FX15	Set timer= VX (01=1/60 second)
FX18	Set tone duration= VX (01=1/60 second)
FXF8	VX to output port
FXFB	Input port to VX (waits for EF4=1)
AMMM	Let $I=OMMM$
FX1E	Let $I=I+VX$
FX29	Let $I=$ 5-byte display pattern for LSD of VX
FX33	Let $MI=$ 3-decimal digit equivalent of VX (I unchanged)
FX55	Let $MI=V0 : VX$ ($I=I+X+1$)
FX65	Let $V0 : VX=MI$ ($I=I+1$)
00E0	Erase display (all 0's)
DXYN	Show n-byte MI pattern at $VX-VY$ coordinates. I unchanged. MI pattern is combined with existing display via EXCLUSIVE-OR function. VF=01 if a 1 in MI pattern matches 1 in existing display.
OMMM	Do machine language subroutine at OMMM (subroutine must end with D4 byte)
BXY0	Set VY color @ $VX(NH)$, $VX+1(NV)$
BXYN	$N \neq 0$, set VY color @ VX , $VX+1$ byte, N bytes vertically
02A0	Step background 1 color

Chapter III

The VP-590 Color Board

Installation

To connect your Color board, three steps are necessary. First, several components must be removed from your VIP. Second, the Color board must be installed on the VIP. Third, the Color board must be attached to your color display unit (monitor or TV set with a RF modulator).

1. Modifying your VIP - Disconnect all power from the VIP. Remove the dust cover from your VIP by loosening and removing the retaining nut on the RESET/RUN toggle switch. Remove the cover by gently prying the lower edge of the cover from one side of the VIP circuit board. The cover will then lift off. Locate the two integrated circuit chips mounted in sockets marked U3 and U4 (see Figure 1-III). Remove the integrated circuits by gently prying them up. Also remove the integrated circuit marked on the board as U2 (type 1861). Install a short wire jumper between pins 3 and 5 of the socket of U4. Figure 1 shows the location of U2, U3, and U4. Figure 2 shows the placement of the jumper. Before replacing the dust cover, you may wish to modify the speaker wiring if you also intend to use the Simple Sound board. See Chapter IV. Replace the dust cover by snapping it back over the VIP circuit board. Screw the retaining nut back onto the RESET/RUN switch. The 1861 integrated circuit removed from the VIP should now be inserted in the socket on the Color board - refer to diagram figure 3-III and gently snap the circuit into the socket. Make sure the notch on the IC is correctly oriented.
2. Place the Color board in the left socket on the rear of the VIP circuit board (the expansion connector). The components on the

III-2

Color board should face toward the front of the VIP. The color board should be pushed down firmly until it is completely inserted in the socket.

3. If you are using a color monitor for your video display, connect the coaxial cable from the Color board to the monitor input jack. If you are using an external RF modulator unit with a color TV set, follow the modulator manufacturer's instruction. You can now test your installation by applying power to the VIP and turning on your color monitor. Hold down the "C" key and move the toggle switch from RESET to RUN. The usual pattern of white dots should appear, but they should now appear on a blue background. Adjust the controls on the monitor or TV set for a satisfactory display. Note that the Color board will work satisfactorily with a black and white monitor providing gray levels instead of colors. There will be no output from the video cable attached to the VIP circuit board unless the Color board is disconnected, the jumper is removed and U2, 3, and 4 are replaced. Black and white CHIP-8 programs will operate normally on a VIP with the Color board installed, except that the display will now be white on blue. Only programs written using CHIP-8X color commands will appear in color.

A Color Test Program - load your copy of CHIP-8X from tape and enter the following color test program:

0300	A32E	0316	C577	0326	F807
0302	6200	0318	CC07	0328	3800
0304	6100	031A	B4C0	032A	1326
0306	D124	031C	C707	032C	1314
0308	7108	031E	4701	032E	FFFF
030A	3140	0320	02A0	0330	FFFF
030C	1306	0322	6820		
030E	7204	0324	F815		
0310	3220				
0312	1304				
0314	C477				

Table 1-III

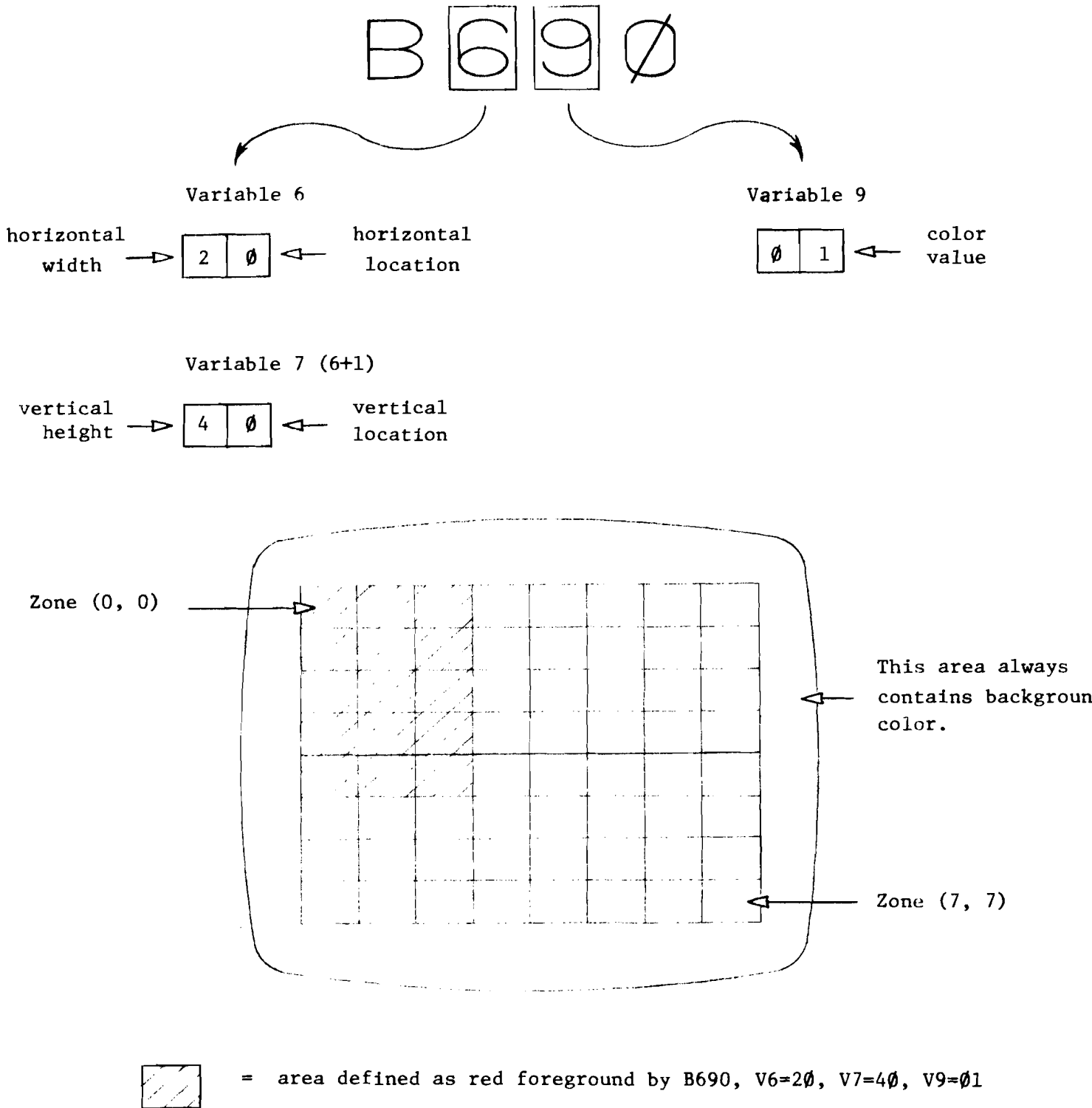
Color Value	Foreground Color
0	Black
1	Red
2	Blue
3	Violet
4	Green
5	Yellow
6	Aqua
7	White

The X in BXY0 represents the number of the variable which contains both the horizontal position and horizontal size of the area. The most significant digit of VX specifies the width (1 to 8 zones) of the area. The least significant digit of VX specifies the horizontal position (0-7) of the left-most zone in the area.

The vertical position and size of the area is specified by V(X+1), the next higher numbered variable. The size of the area defined is always one greater than the number specified (thus an area defined with a width of 0 will actually be 1 zone wide).

As an example, suppose we wanted to define a red area in the upper-left corner of the screen which was three zones wide and 5 zones high (24 dots x 20 dots). We will store this information in variables V9, V6 and V7. Store a value of 01 in V9 to define the color as red. V6 should be set to 20 (2+1=3 for the area width, zero for the horizontal coordinate). V7 should be set to 40 (4+1=5 for the area height, zero for the vertical coordinate).

The area would be defined by the instruction B690.



III-7

As another example, suppose you wanted to set yellow as foreground color for the entire display.

<u>Address</u>	<u>Code</u>	
0300	6205	V2=05 (yellow color value)
0302	6670	V6=70 (7=height-1, 0=horizontal coordinate)
0304	6770	V7=70 (7=width-1, 0=vertical coordinate)
0306	B620	Define 8 x 8 yellow area with upper left-hand corner at 0,0.

5XY1 - special addition routine for low-resolution color definition.

In the BXY0 instruction the variables VX and VX+1 are each treated as two single-digit hex numbers rather than a single two-digit number. In addition only the numbers 0-7 are valid for each digit. To assist in manipulating these variables, the 5XY1 instruction is provided in CHIP-8X. This instruction adds each digit in VX to each digit in VY, stores the result in VX, and translates the result to modulus 8. Thus if V7=36 and VA=22 then an instruction 57A1 would leave a value of 50 in V7. This occurs because 2+3=5 for the most significant digit and 6+2=8 for the least significant digit. Since the result of the addition of the two least significant digits is greater than seven only the value of the 3 least significant bits, which in this case is 0, is retained.

BXYN - set high resolution color foreground area.

For some applications it is useful to be able to divide the display screen into color zones smaller than the 8 x 4 dot area controlled by the BXY0 instruction. The BXYN does this by dividing the display area into zones which are each 8 x 1 dots. Thus each byte displayed on the screen can be defined separately. BXYN allows you to define a foreground color area which is one byte (eight dots) wide and one to fifteen dots high. Variable VY defines the color of the area, just as with the BXY0. The

instruction works similarly to the DXYN instruction of CHIP-8X. It turns the color block containing the VX, VX+1 coordinates to the VY color, N bytes high. Thus the same coordinates used to show a pattern can be used to define color at the same location.* Variable VX defines the horizontal column (0-3F) and VX+1 defines the starting row (0-1F) for the area. N is a hex number (1-F) defining the height of the area. Thus to define a green foreground area for the third dot over from the left edge of the display area, starting at the ninth row of dots and extending downward for ten (decimal) rows you would:

1. Define a variable which contains 04 (green).
2. Define a variable which contains 02 (column number).
3. Define the next variable to contain 08 (the starting row of the area).
4. Use an N value of A (hexidecimal 10) for the height of the display.

For instance:

<u>Instruction</u>	<u>Comments</u>
6304	V3=04
6702	V7=02
6808	V8=08
B73A	set foreground to green for 10 vertical color blocks starting at (2,8)

*NOTE: Because of the CHIP-8X display format routines, if the pattern is more than 1 bit wide it is necessary to also change the next higher horizontal color blocks. This can be easily done by incrementing the VX variable by 8 (7X08), executing another set color instruction (BXYN) and decrementing VX to its original value (7XF8).

VIP Component Location Map

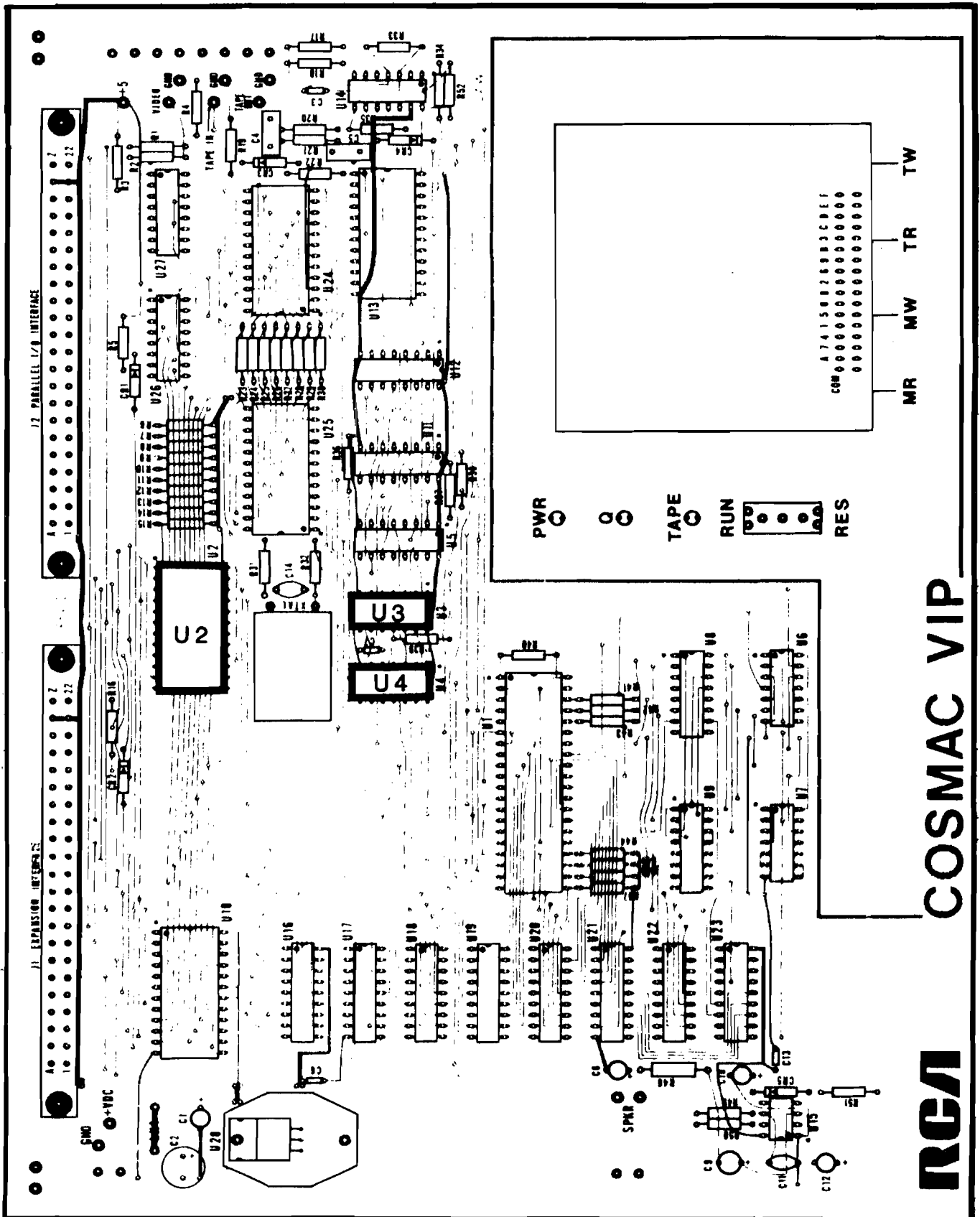


Figure 1-III

III-10

U4 Jumper Placement

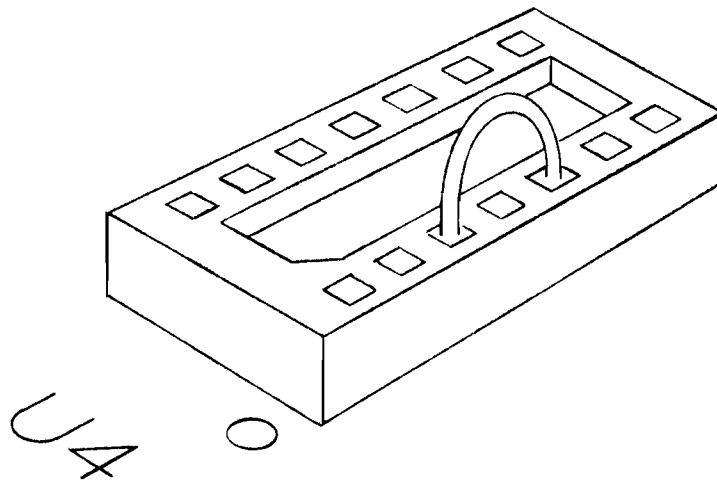
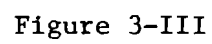


Figure 2-III



Chapter IV

The VP-595 Simple Sound Board

Installation

The Simple Sound board uses the processor Q line to turn the sound on and off. The Q line is also connected to the "beeper" circuitry of the VIP, so the VIP speaker must be disabled when the Simple Sound board is in use. The most flexible arrangement is to place a switch in series with one of the VIP speaker leads. This allows you to easily switch between the speaker and the Simple Sound output. If you intend to always leave the Simple Sound board connected to your VIP, just cutting a speaker lead is sufficient.

In any case, to get access to the VIP speaker leads you must remove the VIP dust cover, as described in Chapter II. You can then use a pair of wire cutters or even scissors to cut one of the speaker leads. Figure 1-IV shows the speaker location. If you wish to install a switch in the speaker lead, just unsolder one of the speaker leads from the printed circuit card, attach the free lead to a switch, and solder a wire from the other side of the switch to the point on the printed circuit card from which you removed the speaker wire. Replace the dust cover and your VIP modification is complete.

To install the Simple Sound board, insert it in the right-hand VIP connector (the I/O connector).

Testing

Once the board is installed, apply power to the VIP and move the toggle switch from RESET to RUN while holding down the "C" key. You should get a tone from the Simple Sound speaker as long as the "C" key is depressed. If you also get a tone from the VIP speaker, the switch you installed to cut off the speaker is in the wrong position.

IV-2

To test CHIP-8X capabilities with Simple Sound, load CHIP-8X from tape and then enter the following program:

<u>address</u>	<u>contents</u>	<u>remarks</u>
0300	6008	V0=08
0302	6100	V1=00
0304	F1F8	V1 → out
0306	F015	V0 → Timer
0308	F018	V0 → Tone
030A	F207	Timer → V2
030C	3200	Skip if V2=0
030E	130A	Go back 2 instructions
0310	7101	V1=V1 + 01
0312	1304	Go to third instruction

When you run this program you should hear a tone starting at a high pitch which descends in steps to a low pitch and then repeats.

Programming

There is only one instruction in CHIP-8X which is used with the Simple Sound board: FXF8. This instruction takes the byte stored in variable VX and puts it in the output port. This instruction can be used for general control of any device connected to the output port, but here we will only consider its use with the Simple Sound board.

The Simple Sound board (see Figure 2-IV for block diagram) is composed of a high frequency oscillator, whose frequency is about 440 Khz. This frequency may be varied slightly by the trimming resistor R2 on the Simple Sound board. This high frequency is fed into a programmable divider - a circuit whose output frequency is a function of the input frequency divided by the binary number present on the VIP output port. The general form of the equation defining frequency generated is:

$$\text{Frequency Generated} = \frac{\text{Input Frequency}}{(\text{Hex Code} + 1)_{10} (16)} \text{ Hz}$$

The only exception to this rule occurs when a value of 00 is sent to the VIP output port. In this case special Circuitry on the Simple Sound board forces a divide ratio of

IV-3

129X16 which results in an output tone of about 213 cycles per second. This feature was added to the Simple Sound board because the VIP puts a value of 00 in the output port every time the VIP is initialized. Normally this would result in no tone being generated, which would interfere with the operation of the monitor program (no tone would be heard when you depressed a key on the keypad).

Regardless of the value stored in the output port, no tone will be generated unless the Q line is at logic one (high). Thus you are able to program an output frequency and then determine the length of the output tone by controlling how long the Q line is on. After passing through the switch controlled by the Q line, the tone is passed through an amplifier. The gain (volume) of the amplifier is controlled by potentiometer R4 on the right side of the Simple Sound board. The amplifier output is then connected to the speaker.

To put out a burst of sound from the Simple Sound speaker, you simply:

1. Put the desired divide ratio into a variable. (See Table 1-IV for frequency vs hex code data out.)
2. Transfer the contents of that variable to the output port through a FXF8 instruction.
3. Set a variable to the desired length of tone (in 1/60th of a second increments).
4. Set the tone output length with a FX18 instruction.

For example, to generate a one-second "beep" of a 2300 cps tone:

0300	603A	V0=3A (60_{10})
0302	6112	V1=12 (divide ratio)
0304	F1F8	V1 output port
0306	F018	V0 Tone + timer
0308	1308	loop forever here

Miscellaneous

The Q line is used to put out the pulse sequence used when saving memory contents on tape. This pulse train will be heard as a buzzing or rasping sound on the Simple

Sound output whenever data is saved.

If you wish to record the Simple Sound output or use an earphone for private listening, a jack is provided on the board. This jack can also be connected to a larger speaker or the auxiliary input of a larger amplifier. Whenever a plug is inserted in the jack, the on-board speaker is disconnected.

Table 1-VI provides a quick reference of hex code vs frequency along with the corresponding musical note and octave. You will find this convenient for composing tunes that compliment your CHIP-8X programs.

Speaker Switch Connection

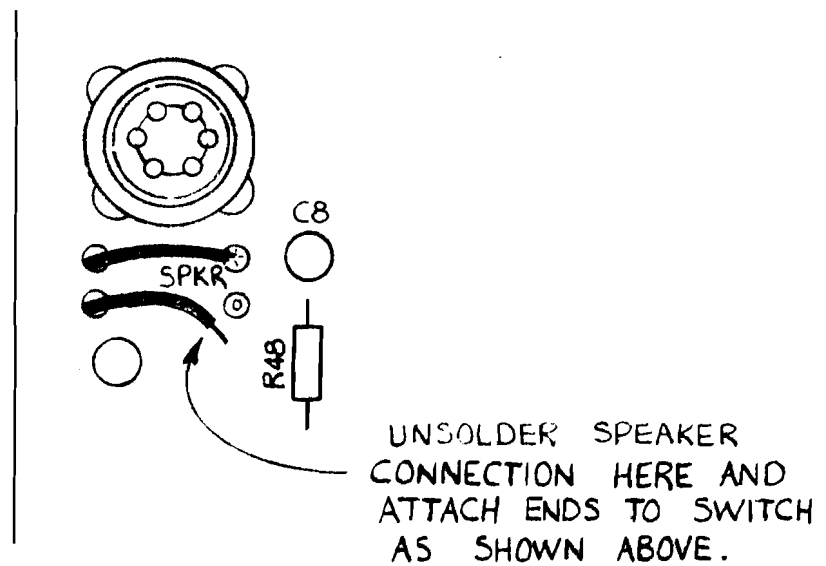
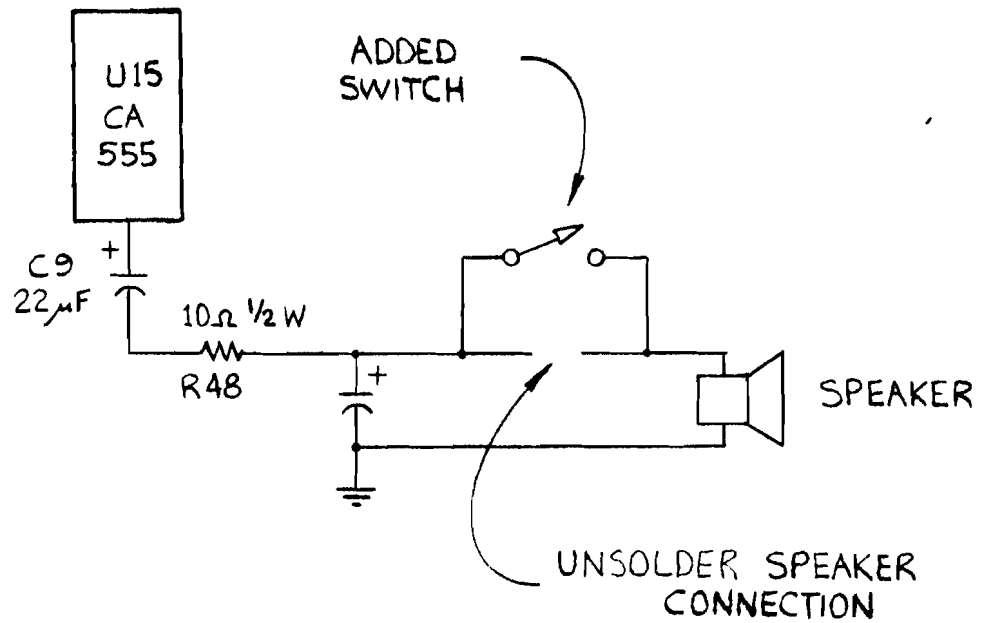


Figure 1-IV

IV-6

Simple Sound Block Diagram

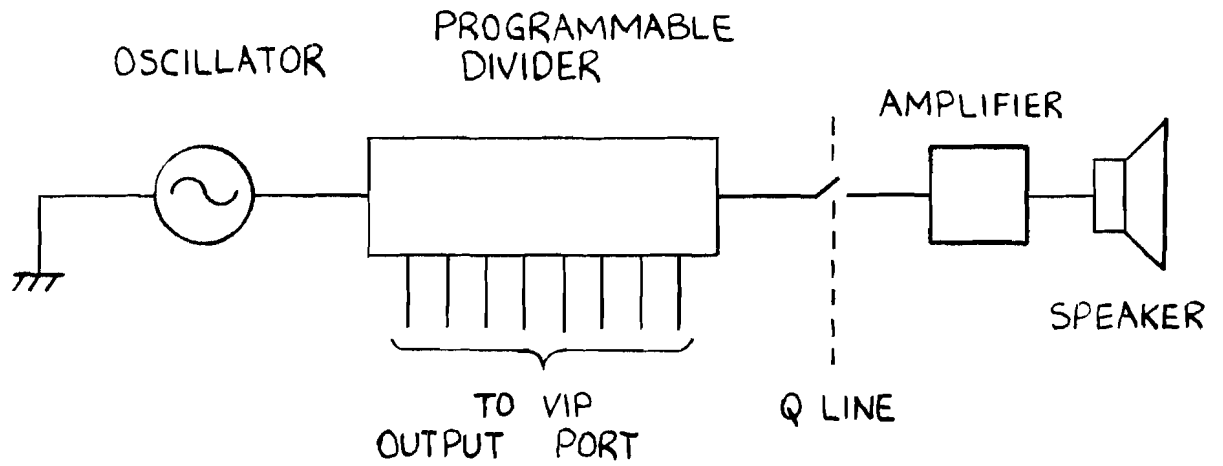


Figure 2-IV

Chapter V

VP-580 VIP Expansion Keypads

One or two Expansion Keypads may be added to your VIP. Sockets for connecting the Expansion Keypads are located on the VP-590 Color board. If you wish to use the keypads without a Color board, you will need a VP-585 Keypad Interface card.

Adding one keypad gives you a second input device for two-player games. CHIP-8X contains two new instructions for accessing this external keypad. If a second keypad is added, it duplicates the function of the VIP's on-board keypad. This keypad is accessed using the normal CHIP-8 keypad instructions. Through this text we will refer to the keypad which uses the new instructions as Keypad 2, and the keypad which duplicates the on-board as Keypad 1.

Installation

Prior to installing the external keypads, either a VP-590 Color board or a VP-585 Hex Keyboard Interface board must be installed in the expansion (left) socket.

To install either Keypad, insert the keypad cable through the appropriate slot in the Color board or keypad Interface card from the rear of the board. Loop the cable as shown in Figure 1-V and insert it in the appropriate socket. Keypad 1 should be inserted in the socket marked J1 on the Hex Keyboard Interface board or the socket marked #1 on the Color board. Keypad 2 is connected to socket J2 on the Hex Keyboard Interface board or socket #2 on the Color board.

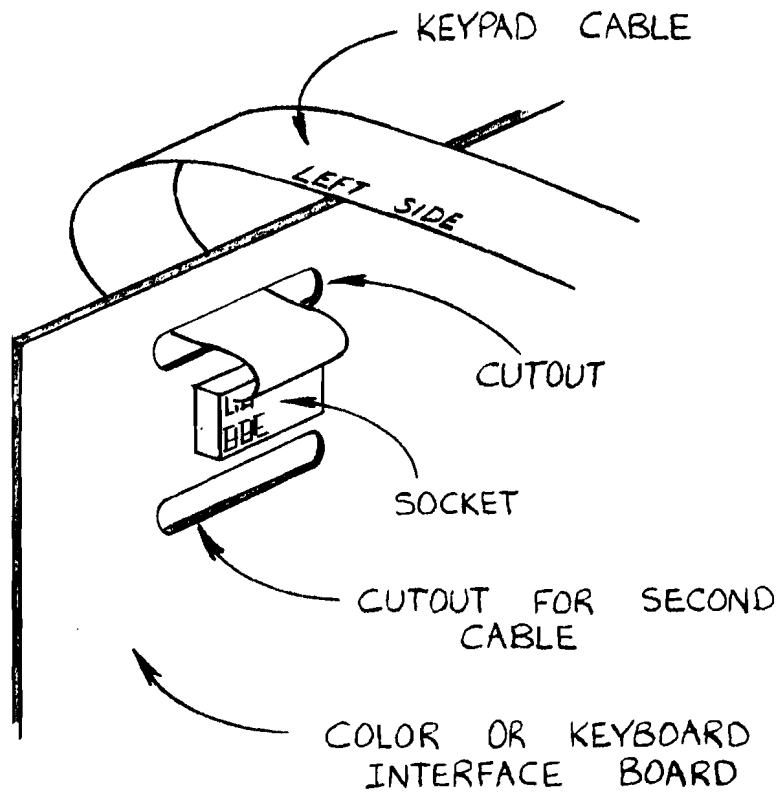


Figure 1-V

Programming

CHIP-8X has two instructions for accessing Keypad 2:

EXF2 Skip next instruction if VX=Keypad 2
 EXF5 Skip next instruction if VX≠Keypad 2

Their operation can be summarized as:

Instruction Executed	Condition		
	Key=VX	Key≠VX	No Key Depressed
EXF2	Skip	No Skip	No Skip
EXF5	No Skip	Skip	Skip

Programming Examples:

1. Wait for key #3 to be depressed:

```

0300      6503      V5=3
0302      E5F2      Skip next if key=3
0304      1302      go back and check again
              .
              .
              .
              program continues
  
```

2. Wait for any key to be depressed and return its value in V6:

```

0300      7601      V6=V6+1
0302      E6F2      KEY=V6?
0304      1300      No, go back
0306      6A0F      yes, mask off MSD
0308      86A2      by AND ing with VA
                  continue
              .
              .
              .
  
```

3. Go to subroutine at 0400 if key "F" is pressed, otherwise continue.

```

0300      640F      V4=0F
0302      E4F2      Skip if key=4V
0304      2400      go sub 400
0306                      continue
  
```

4. Wait for any key to be pressed, then wait for key to be released. Return key value in V6. (This routine is useful when entering data from the keypad, since a key will continue to be read as long as it is depressed, causing possible multiple entry of the same key.)

0300	7601	V6=V6+1
0302	E6F2	Key=V6
0304	1300	No, go back
0306	6A0F	Yes, mask off MSD
0308	86A2	by ANDing with VA
030A	7303	V3=04
030C	F315	V3 → Timer
030E	F307	Timer → V3
0310	3300	Test if V3=0
0312	130E	No, go back
0314	E6F5	Key≠V6
0316	1314	No, loop until key released
0318		continue

Instructions 030A through 0312 form a brief delay for key debouncing.

Chapter VI

Circuit Description and Schematic Diagrams

Color Board Circuit Description:

U1 and U2 combine to generate the necessary NTSC compatible color signals. U1, a CDP1861, generates the sync and spot signals while U2, a CDP1862, generates the color burst frequency and phases. The luminous, color, and sync signals are combined through resistors and capacitors and fed to a 75 Ω video driver formed by the transistors of U12.

The color data is stored in U3, a CDP1822 (256 x 4 Bit RAM), and read at the appropriate time by U2. When reset the CDP1862 is locked into a mode in which all spots are white and the background color is blue. This mode continues during program execution until a memory write to any memory location in the range C000 to DFFF occurs. From this point U2 receives color information from the color memory, U3.

The most significant 4 bits of the memory address are latched by U4. These bits are then decoded by U6 and U8 to generate appropriate enable signals. U8 also acts as a one bit memory for selecting the high or low resolution mode for the color display.

The address of the color memory is assigned as follows:

C000-C007 : Low Resolution (8x8) Color Overlay
C020-C027
C040-C047
C060-C067
C080-C087
C0A0-C0A7
C0C0-C0C7
C0E0-C0E7

D000-D007 : High Resolution (32x8) Color Overlay
D008-D00F
D010-D017

.
.
.
.

D0F8-D0FF

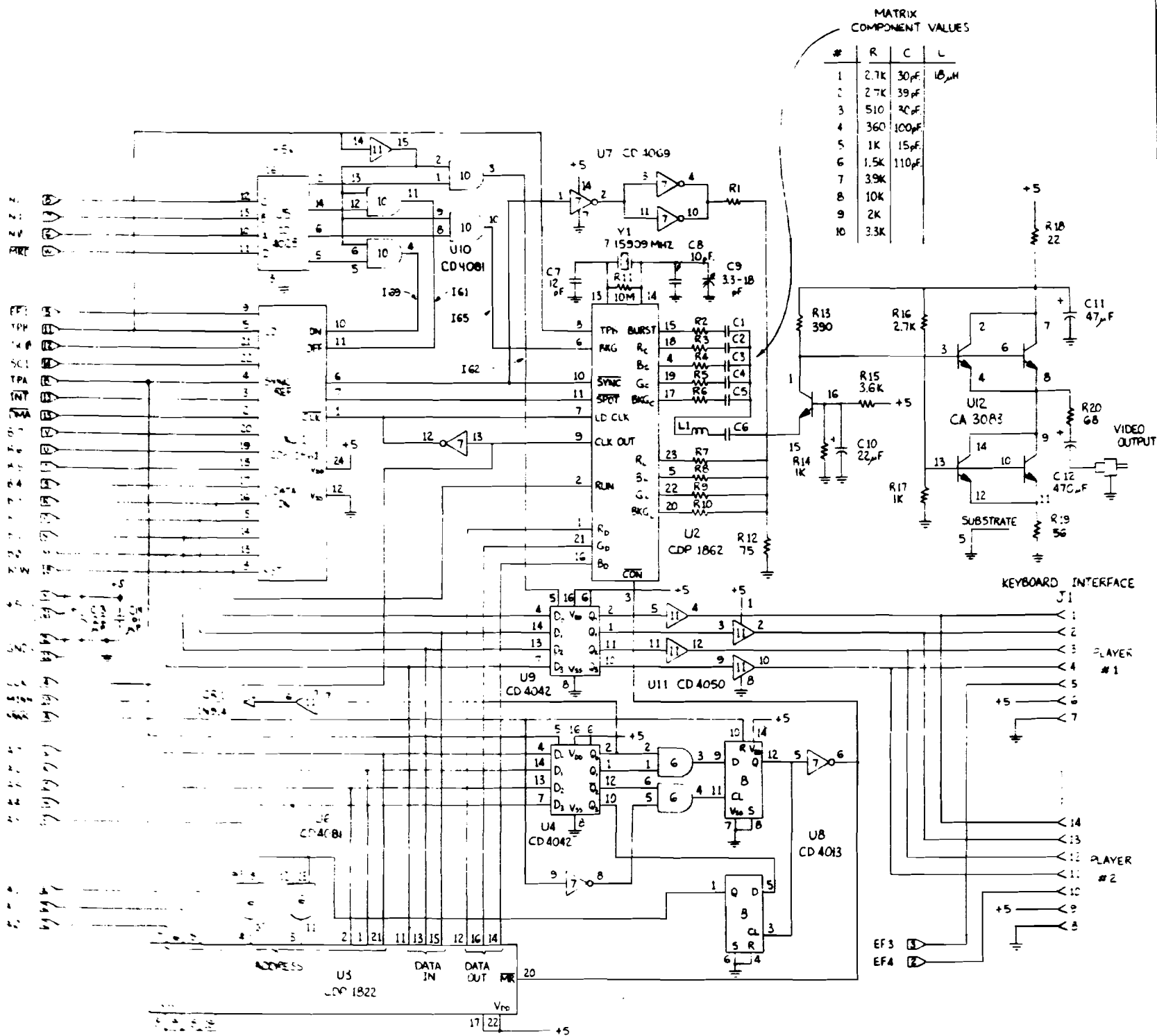
U5 and U10 form a decoder to generate input/output signals. I/O instructions

are assigned as follows:

- 61 - Turn Display Off
- 62 - Output to Keyboard
- 65 - Switch Background Color
- 69 - Turn Display On

The external keyboard interface is implemented by U9, a 4-bit latch, and four drivers of U11.

VP590 Color Board Schematic



Simple Sound Circuit Description:

U1, a CDP1863, is a programmable divide-by-n counter. It accepts a clock input, generated by two inverters at U2, on pin 2 and outputs the divided frequency on pin 14. The divided frequency is then buffered by 4 parallel inverters (U2) driving a transistor amplifier. The n factor for the divide is determined by the data on Bus 0 through Bus 7 (output port on VIP). U3 and U4 decode 00 and inject 80 into U1. This is necessary since the output port latches 00 when the VIP is reset (00 produces an inaudible frequency).

The frequency adjust pot is factory adjusted such that:

$$\text{Frequency Generated} = \frac{27535}{(\text{Hex Code} + 1)_{10}} \text{ Hz}$$

The general form of the equation defining frequency generated is:

$$\text{Frequency Generated} = \frac{\text{Input Frequency}}{(\text{Hex Code} + 1)_{10} (16)} \text{ Hz}$$

Refer to Table 1 for hex code vs frequency generated information.

VP550 Simple Sound Schematic

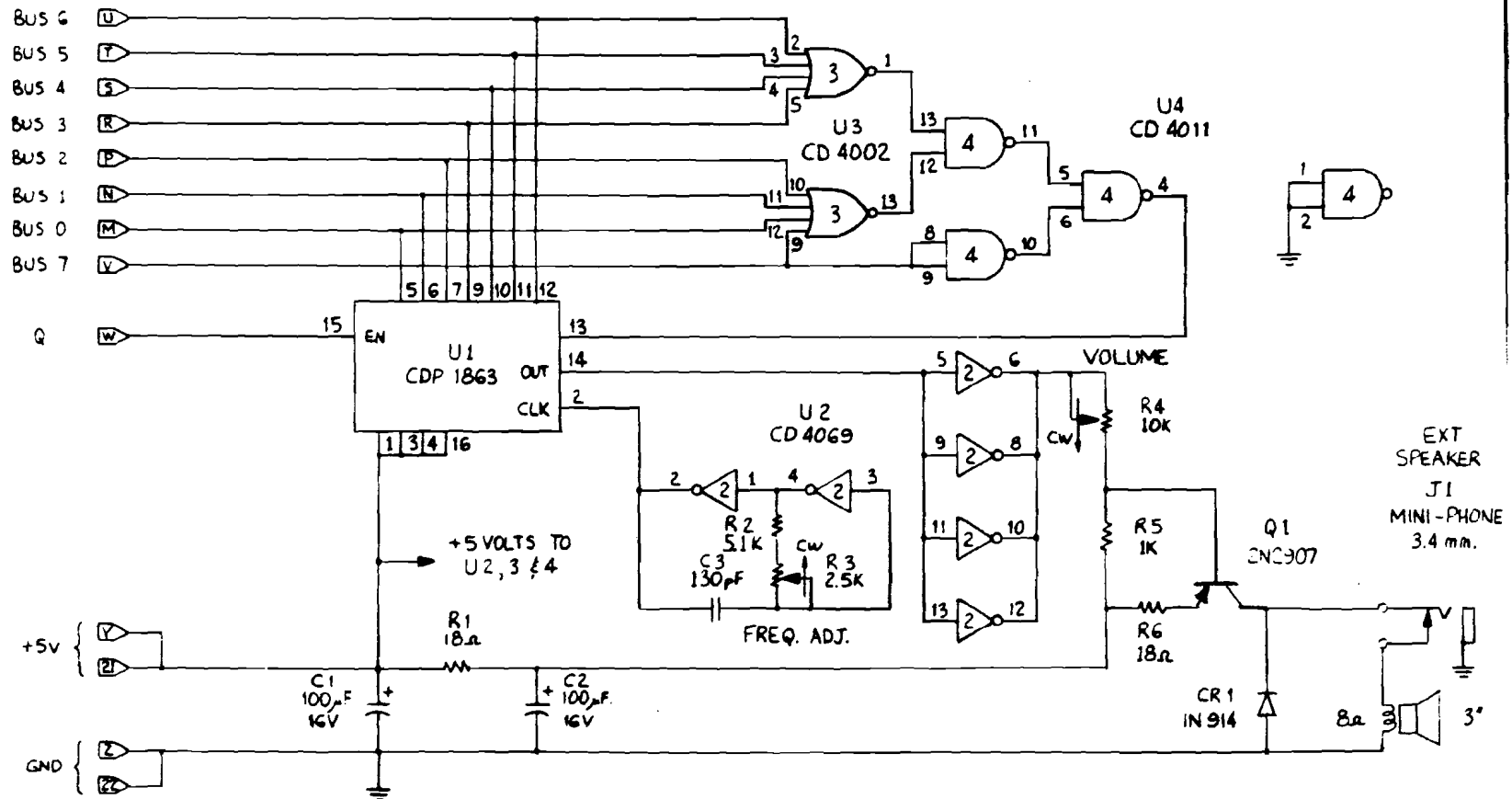


Table 1-VI Simple Sound Frequencies

<u>Hex Code</u>	<u>Divide-By-N</u> (Hex + 1) ₁₀	<u>Frequency Generated</u> Hz	<u>Note</u> (and Octave)	<u>Note Frequency</u> Hz
FF	256	107.56		
FE	255	107.98		
FD	254	108.41		
FC	253	108.83		
FB	252	109.27		
FA	251	109.70		
F9	250	110.14	A2	110.00
F8	249	110.58		
F7	248	111.03		
F6	247	111.48		
F5	246	111.93		
F4	245	112.39		
F3	244	112.85		
F2	243	113.31		
F1	242	113.78		
F0	241	114.25		
EF	240	114.73		
EE	239	115.21		
ED	238	115.69		
EC	237	116.18		
EB	236	116.61	A#2	116.54
EA	235	117.17		
E9	234	117.67		
E8	233	118.18		
E7	232	118.69		
E6	231	119.20		
E5	230	119.72		
E4	229	120.24		
E3	228	120.77		
E2	227	121.30		
E1	226	121.84		
E0	225	122.38		
DF	224	122.92		
DE	223	123.48	B2	127.47
DD	222	124.03		
DC	221	124.59		
DB	220	125.16		
DA	219	125.73		
D9	218	126.31		
D8	217	126.89		
D7	216	127.48		
D6	215	128.07		
D5	214	128.67		
D4	213	129.27		
D3	212	129.88		
D2	211	130.50	C3	130.81
D1	210	131.12		
D0	209	131.75		
CF	208	132.38		
CE	207	133.02		
CD	206	133.67		
CC	205	134.32		
CB	204	134.98		

<u>Hex Code</u>	<u>Divide-By-N</u> <u>(Hex + 1)₁₀</u>	<u>Frequency Generated</u> <u>Hz</u>	<u>Note</u> <u>(and Octave)</u>	<u>Note Frequency</u> <u>Hz</u>
CA	203	135.64		
C9	202	136.31		
C8	201	136.99		
C7	200	137.68		
C6	199	138.37	C#3	138.59
C5	198	139.07		
C4	197	139.77		
C3	196	140.48		
C2	195	141.21		
C1	194	141.93		
C0	193	142.67		
BF	192	143.41		
BE	191	144.16		
BD	190	144.92		
BC	189	145.69		
BB	188	146.46	D3	146.83
BA	187	147.25		
B9	186	148.04		
B8	185	148.84		
B7	184	149.65		
B6	183	150.46		
B5	182	151.29		
B4	181	152.13		
B3	180	152.97		
B2	179	153.83		
B1	178	154.69		
B0	177	155.56	D#3	155.56
AF	176	156.45		
AE	175	157.34		
AD	174	158.25		
AC	173	159.16		
AB	172	160.09		
AA	171	161.02		
A9	170	161.97		
A8	169	162.93		
A7	168	163.90		
A6	167	164.88	E3	164.81
A5	166	165.87		
A4	165	166.88		
A3	164	167.90		
A2	163	168.93		
A1	162	169.97		
A0	161	171.02		
9F	160	172.09		
9E	159	173.18		
9D	158	174.27	F3	174.61
9C	157	175.38		
9B	156	176.51		
9A	155	177.65		
99	154	178.80		
98	153	179.97		
97	152	181.15		
96	151	182.35		
95	150	183.57		

<u>Hex Code</u>	<u>Divide-By-N</u> (Hex + 1) ₁₀	<u>Frequency Generated</u> Hz	<u>Note</u> (and Octave)	<u>Note Frequency</u> Hz
94	149	184.80	F#3	185.00
93	148	186.05		
92	147	187.31		
91	146	188.60		
90	145	189.90		
8F	144	191.22		
8E	143	192.55		
8D	142	193.91		
8C	141	195.28	G3	196.00
8B	140	196.68		
8A	139	198.09		
89	138	199.53		
88	137	200.99		
87	136	202.46		
86	135	203.96		
85	134	205.49	G#3	207.65
84	133	207.03		
83	132	208.60		
82	131	210.19		
81	130	211.81		
80	129	213.45		
7F	128	215.12		
7E	127	216.81	A3	220.00
7D	126	218.53		
7C	125	220.28		
7B	124	222.06		
7A	123	223.86		
79	122	225.70		
78	121	227.56		
77	120	229.46		
76	119	231.39	A#3	233.08
75	118	233.35		
74	117	235.34		
73	116	237.37		
72	115	239.43		
71	114	241.54		
70	113	243.67		
6F	112	245.85	B3	246.94
6E	111	248.06		
6D	110	250.32		
6C	109	252.61		
6B	108	254.95		
6A	107	257.34		
69	106	259.76		
68	105	262.24	C4	261.63
67	104	264.76		
66	103	267.33		
65	102	269.95		
64	101	272.62		
63	100	275.35		
62	99	278.13	C#4	277.18
61	98	280.97		
60	97	283.87		

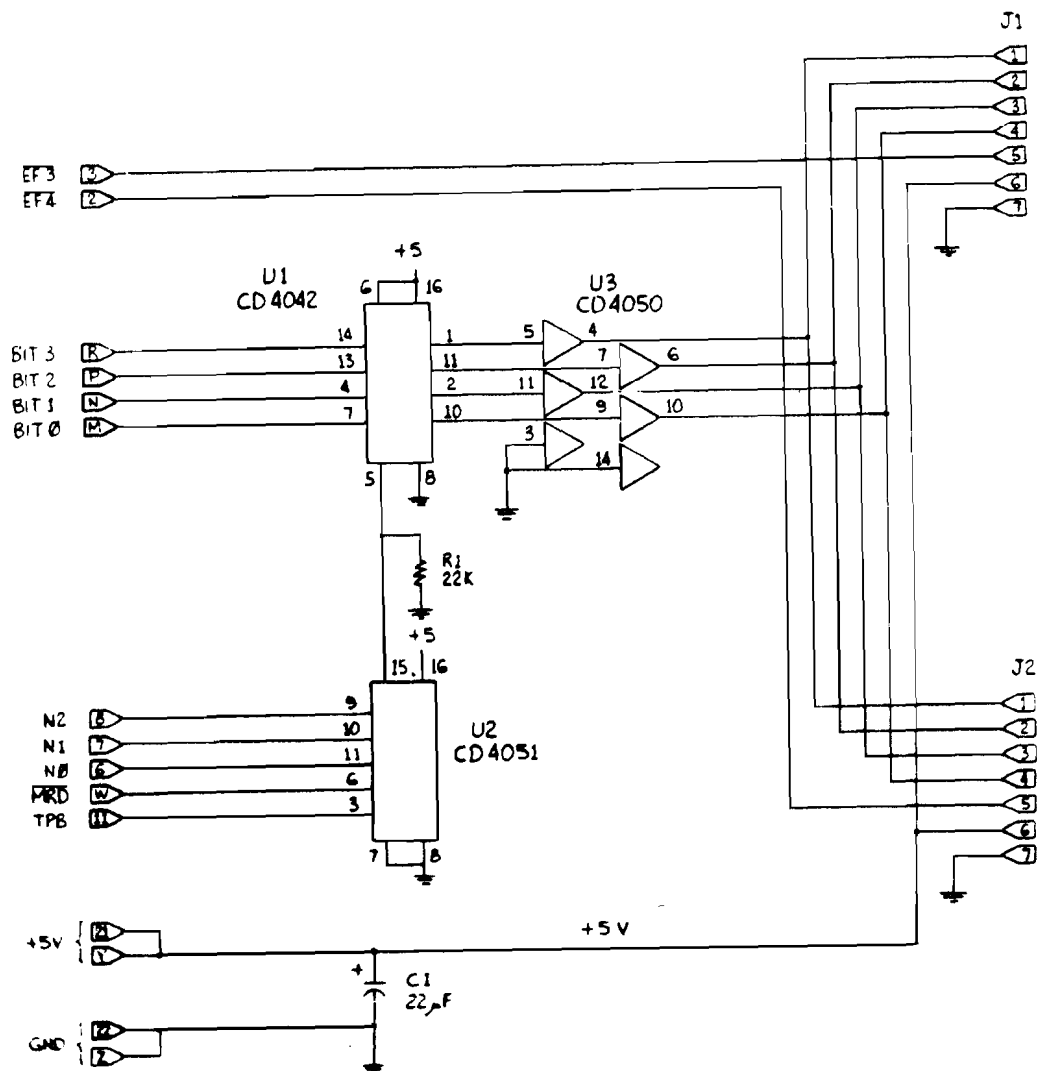
<u>Hex Code</u>	<u>Divide-By-N</u> <u>(Hex + 1)₁₀</u>	<u>Frequency Generated</u> <u>Hz</u>	<u>Note</u> <u>(and Octave)</u>	<u>Note Frequency</u> <u>Hz</u>
5F	96	286.82		
5E	95	289.84		
5D	94	292.93	D4	293.66
5C	93	296.08		
5B	92	299.29		
5A	91	302.58		
59	90	305.94		
58	89	309.38	D#4	311.13
57	88	312.90		
56	87	316.49		
55	88	320.17		
54	85	323.94		
53	84	327.80	E4	329.63
52	83	331.75		
51	82	335.79		
50	81	339.94		
4F	80	344.19		
4E	79	348.54	F4	349.23
4D	78	353.01		
4C	77	357.60		
4B	76	362.30		
4A	75	367.13		
49	74	372.09	F#3	369.99
48	73	377.19		
47	72	382.43		
46	71	387.82		
45	70	393.36	G4	392.00
44	69	399.06		
43	68	404.93		
42	67	410.97		
41	66	417.20	G#4	415.30
40	65	423.62		
3F	64	430.23		
3E	63	437.06	A4	440.00
3D	62	444.11		
3C	61	451.39		
3B	60	458.92		
3A	59	466.69	A#4	466.16
39	58	474.74		
38	57	483.07		
37	56	491.70	B4	493.88
36	55	500.64		
35	54	509.91		
34	53	519.53	C5	523.25
33	52	529.52		
32	51	539.91		
31	50	550.70	C#5	554.37
30	49	561.94		
2F	48	583.65		
2E	47	585.85	D5	587.33
2D	46	598.59		
2C	45	611.89		
2B	44	625.80	D#5	622.25

<u>Hex Code</u>	<u>Divide-By-N</u> (Hex + 1) ₁₀	<u>Frequency Generated</u> Hz	<u>Note</u> (and Octave)	<u>Note Frequency</u> Hz
2A	43	640.35		
29	42	655.60	E5	659.26
28	41	671.59		
27	40	688.38		
26	39	706.03	F5	698.46
25	38	724.61		
24	37	744.19	F#5	739.99
23	36	764.86		
22	35	785.71	G5	783.99
21	34	809.85		
20	33	834.39	G#5	830.61
1F	32	860.47		
1E	31	888.23	A5	880.00
1D	30	917.83	A#5	932.33
1C	29	949.48		
1B	28	983.39	B5	987.77
1A	27	1019.81		
19	26	1059.04	C6	1046.5
18	25	1101.40	C#6	1108.7
17	24	1147.29		
16	23	1197.17	D6	1174.7
15	22	1251.59	D#6	1244.5
14	21	1311.19	E6	1318.5
13	20	1376.75	F6	1396.9
12	19	1449.21	F#6	1480.0
11	18	1529.72	G6	1661.2
10	17	1619.71	G#6	1661.2
0F	16	1720.94	A6	1760.0
0E	15	1835.67	A#6	1864.7
0D	14	1966.79	B6	1975.5
0C	13	2118.08	C7	2093.0
0B	12	2294.58	C#7	2217.5
0A	11	2503.18	D#7	2489.0
09	10	2753.50	F7	2793.8
08	9	3059.44	F#7	2960.0
07	8	3441.88	A7	3520.0
06	7	3933.57	B7	3951.1
05	6	4589.17	D8	4698.6
04	5	5507.00	F8	5587.7
03	4	6883.75	G#8	6644.9
02	3	9178.33		
01	2	13767.50		
00	1	27535.00		

Expansion Keypad Interface Card Circuit Description:

U1 is a 4-bit latch. A "62" output instruction, decoded by U2 causes the least significant 4 bits of the data bus to be strobed into U1. These latched bits are then buffered to the expansion hex keypad via U3.

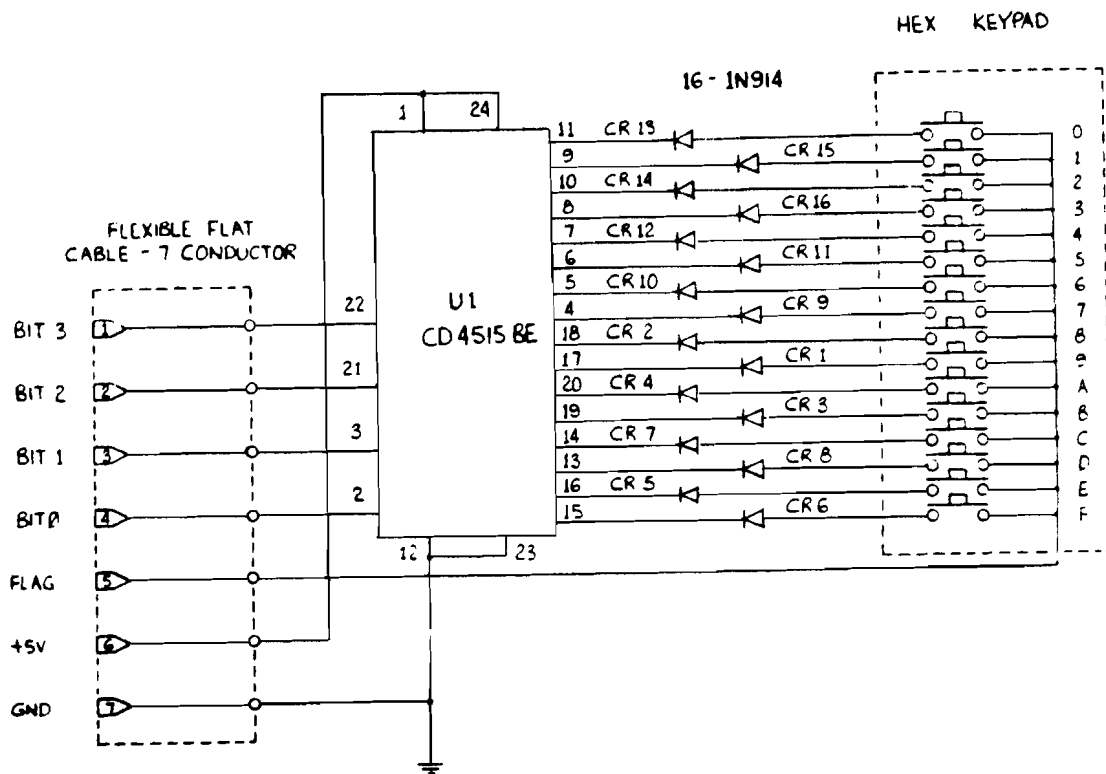
VP585 Expansion Keypad Interface Card Schematic



Expansion Keypad Circuit Description:

The Expansion Keypad consists of a 4 to 16 decoder (U1), 16 diodes, and 16 switches. The microprocessor outputs data to a 4-bit latch on the interface card. The latched 4 bits are decoded by U1 and a signal fed to one of the 16 switches. If that switch is closed the signal is transferred back to the microprocessor's data flag (EF3=Player #1 and EF4=Player #2). The flag may then be tested by the microprocessor's program. The 16 diodes prevent an undefined logic level should two or more switches be pressed simultaneously.

VP580 Expansion Keypad Schematic



APPENDIX A
CHIP-8X PROGRAMS

Color Kaleidoscope

This program uses the CHIP-8X Interpreter at 0000-02FF. Keys 2, 4, 6 and 8 are used to program patterns of up to 125 entries until the Kaleidoscope automatically starts. However, you may begin execution of the Kaleidoscope any time during the programming sequence by pressing key 0. The colors are randomly selected by the program.

Requirements:

1 VP-590 Color Board

0300 237C D0 037C	0342 A382 I=0382
0302 6383 V3=83	0344 6AE0 VA=E0
0304 611F V1=1F	0346 8A12 VA=VA&V1
0306 620F V2=0F	0348 681F VB=1F
0308 2332 D0 0332	034A 81B2 V1=V1&VB
030A A300 I=0300	034C 3A00 SKIP; VA EQ 00
030C F31E I=I+V3	034E 7201 V2+01
030E F00A V0=KEY	0350 6AF0 VA=F0
0310 F055 MI=V0: V0	0352 8A22 VA=VA&V2
0312 4000 SKIP; V0 NE 00	0354 680F VB=0F
0314 131C G0 031C	0356 82B2 V2=V2&VB
0316 7301 V3+01	0358 3A00 SKIP; VA EQ 00
0318 3300 SKIP; V3 EQ 00	035A 7101 V1+01
031A 1308 G0 0308	035C 681F VB=1F
031C 6383 V3=83	035E 81B2 V1=V1&VB
031E A300 I=0300	0360 CC07 VC=RND
0320 F31E I=I+V3	0362 B1C1 1 BLOCK VC COLOR @V1V2
0322 F065 V0: V0=MI	0364 D121 SHOW 1MI@V1V2
0324 4000 SKIP; V0 NE 00	0366 8A10 VA=V1
0326 131C G0 031C	0368 681F VB=1F
0328 7301 V3+01	036A 8825 VB=VB-V2
032A 4300 SKIP; V3 NE 00	036C 2376 D0 0376
032C 131C G0 031C	036E 6A3F VA=3F
032E 2332 D0 0332	0370 8A15 VA=VA-V1
0330 131E G0 031E	0372 2376 D0 0376
0332 4002 SKIP; V0 NE 02	0374 8820 VB=V2
0334 72FF V2+FF	0376 BAC1 1 BLOCK VC COLOR @VAVB
0336 4004 SKIP; V0 NE 04	0378 DAB1 SHOW 1MI@VAVB
0338 71FF V1+FF	037A 00EE RET
033A 4006 SKIP; V0 NE 06	037C 02A0 SWITCH BAKGRND
033C 7101 V1+01	037E 6000 V0=00
033E 4008 SKIP; V0 NE 08	0380 00EE RET
0340 7201 V2+01	0382 8002

Appendix A-2

VIP BLOCKOUT

This is a fast action 2-player game of concentration, coordination, skill and luck!

Two players each using their own keypad control the movement of their respective lines. Key 2 moves the line up, 8-down, 4-left, and 6-right.

The objective is for each player to maneuver his/her line avoiding collision while, at the same time, trying to force the opponent to collide. As the play continues the speed of movement increases, demanding faster decisions and responses.

Running the program in the usual fashion (by flipping the toggle switch to run), surrounds the playing field by a border. However, if key Ø on either keypad is held while the run switch is flipped, no border is created and the lines can wrap around the screen. Watch out for yourself!

A match consists of 9 games the winner getting the most points (the least number of collisions).

Have fun!

Requirements:

A minimum of 1 VP-580 Hex Keypads. (2 VP-580 keypads are necessary for totally independent key action.)

Optional:

- a) VP-590 - Color Board
- b) VP-595 - Simple Sound Board

Appendix A-3

CHIP-8X INTERPRETER: 0000-02FF

0300 6609 V6=09	0364 3E00 SKIP; VE EQ 00
0302 6700 V7=00	0366 1362 GO 0362
0304 1486 GO 0486	0368 00EE RET :
0306 6701 V7=01	036A 3A00 SKIP; VA EQ 00
0308 A490 I=0490	036C 1372 GO 0372
030A 6000 V0=00	036E 6A01 VA=01
030C 6100 V1=00	0370 00EE RET :
030E F155 MI=V0: V1	0372 6A00 VA=00
0310 2424 DO 0424	0374 00EE RET
0312 6EFF VE=FF	0376 6B02 VB=02
0314 2360 DO 0360	0378 EBA1 SKIP; VB NE KEY #1
0316 2424 DO 0424	037A 1384 GO 0384
0318 6C05 VC=05	037C 7B02 VB+02
031A 6170 V1=70	037E 3B0A SKIP; VB EQ 0A
031C 6270 V2=70	0380 1378 GO 0378
031E B1C0 VC COLOR @V1(NH), V2(NV)	0382 8BC0 VB=VC
0320 6880 V8=80	0384 8CB0 VC=VB
0322 6920 V9=20	0386 4B02 SKIP; VB NE 02
0324 3701 SKIP; V7 EQ 01	0388 1398 GO 0398
0326 244C DO 044C	038A 4B04 SKIP; VB NE 04
0328 6104 V1=04	038C 139E GO 039E
032A 6204 V2=04	038E 4B06 SKIP; VB NE 06
032C 6C08 VC=08	0390 13A4 GO 03A4
032E 633A V3=3A	0392 7100 V1+00
0330 641A V4=1A	0394 7202 V2+02
0332 6D02 V0=02	0396 13A8 GO 03A8
0334 CA01 VA=RND	0398 7100 V1+00
0336 6500 V5=00	039A 72FE V2+FE
0338 4A00 SKIP; VA NE 00	039C 13A8 GO 03A8
033A 2376 DO 0376	039E 71FE V1+FE
033C 4A01 SKIP; VA NE 01	03A0 7200 V2+00
033E 23AE DO 03AE	03A2 13A8 GO 03A8
0340 3F00 SKIP; VF EQ 00	03A4 7102 V1+02
0342 13E6 GO 03E6	03A6 7200 V2+00
0344 236A DO 036A	03A8 A481 I=0481
0346 7501 V5+01	03AA D122 SHOW 2MI@V1V2
0348 3502 SKIP; V5 EQ 02	03AC 00EE RET
034A 1338 GO 0338	03AE 6B02 VB=02
034C 2350 DO 0350	03B0 EB F5 SKIP; VB NE KEY #2
034E 1336 GO 0336	03B2 13BC GO 03BC
0350 F8F8 OUTPUT=V8	03B4 7B02 VB+02
0352 6E02 VE=02	03B6 3B0A SKIP; VB EQ 0A
0354 FE18 TONE=VE	03B8 13B0 GO 03B0
0356 4902 SKIP; V9 NE 02	03BA 8BD0 VB=VD
0358 135E GO 035E	03BC 8DB0 VD=VB
035A 79FF V9+FF	03BE 4B02 SKIP; VB NE 02
035C 78FE V8+FE	03C0 13D0 GO 03D0
035E 8E90 VE=V9	03C2 4B04 SKIP; VB NE 04
0360 FE15 TIME=VE	03C4 13D6 GO 03D6
0362 FE07 VE=TIME	03C6 4B06 SKIP; VB NE 06

(continued on next page)

Appendix A-4

03C8 13DC GO 03DC	042C 6C06 VC=06
03CA 7300 V3+00	042E 6134 V1=34
03CC 7402 V4+02	0430 6220 V2=20
03CE 13E0 GO 03E0	0432 B1C0 VC COLOR @V1(NH), V2(NV)
03D0 7300 V3+00	0434 A490 I=0490
03D2 74FE V4+FE	0436 6110 V1=10
03D4 13E0 GO 03E0	0438 6200 V2=00
03D6 73FE V3+FE	043A 2444 D0 0444
03D8 7400 V4+00	043C A491 I=0491
03DA 13E0 GO 03E0	043E 6128 V1=28
03DC 7302 V3+02	0440 2444 D0 0444
03DE 7400 V4+00	0442 00EE RET
03E0 A483 I=0483	0444 F065 V0: V0=MI
03E2 D342 SHOW 2MI@V3V4	0446 F029 I=V0(LSDP)
03E4 00EE RET	0448 D125 SHOW 5MI@V1V2
03E6 6EFF VE=FF	044A 00EE RET
03E8 FE15 TIME=VE	044C A483 I=0483
03EA FE18 TONE=VE	044E 6100 V1=00
03EC 4A00 SKIP; VA NE 00	0450 6200 V2=00
03EE D122 SHOW 2MI@V1V2	0452 D121 SHOW 1MI@V1V2
03F0 4A01 SKIP; VA NE 01	0454 3F00 SKIP; VF EQ 00
03F2 D342 SHOW 2MI@V3V4	0456 D121 SHOW 1MI@V1V2
03F4 FE07 VE=TIME	0458 62FF V2=FF
03F6 3E00 SKIP; VE EQ 00	045A D121 SHOW 1MI@V1V2
03F8 13EC GO 03EC	045C 7101 V1+01
03FA A490 I=0490	045E 3F01 SKIP; VF EQ 01
03FC F165 V0: V1=MI	0460 1450 GO 0450
03FE A490 I=0490	0462 71FF V1+FF
0400 3A00 SKIP; VA EQ 00	0464 D121 SHOW 1MI@V1V2
0402 7001 V0+01	0466 6201 V2=01
0404 3A01 SKIP; VA EQ 01	0468 6100 V1=00
0406 7101 V1+01	046A D121 SHOW 1MI@V1V2
0408 F155 MI=V0: V1	046C 3F00 SKIP; VF EQ 00
040A 00E0 ERASE	046E D121 SHOW 1MI@V1V2
040C 2424 D0 0424	0470 61FF V1=FF
040E 76FF V6+FF	0472 D121 SHOW 1MI@V1V2
0410 3600 SKIP; V6 EQ 00	0474 7201 V2+01
0412 1312 GO 0312	0476 3F01 SKIP; VF EQ 01
0414 6E02 VE=02	0478 1468 GO 0468
0416 FE18 TONE=VE	047A 72FF V2+FF
0418 6E04 VE=04	047C D121 SHOW 1MI@V1V2
041A FE15 TIME=VE	047E 00EE RET
041C FE07 VE=TIME	0480 01C0
041E 3E00 SKIP; VE EQ 00	0482 C080
0420 141C GO 041C	0484 48D4
0422 1414 GO 0414	0486 E7A1 SKIP; V7 NE KEY #1
0424 6C04 VC=04	0488 1306 GO 0306
0426 6130 V1=30	048A E7F5 SKIP; V7 NE KEY #2
0428 6220 V2=20	048C 1306 GO 0306
042A B1C0 VC COLOR @V1(NH), V2(NV)	048E 1308 GO 0308