G.C.E. VECTREX SERVICE MANUAL

TABLE OF CONTENTS

I. SPECIFICATIONS

The VECTREX is a self-contained, microprocessor based, Vector Display, portable home video game arcade with external game cartridge program capability.

MPU	68A09	8K X 8 BIT DATA — 16 BIT ADDRESS	
INTERNAL ROM	2363	8K X 8 BIT	PLEASE NOTE:
INTERNAL RAM	2114 (2)	1K X 4 BIT (ea.)	INTERNAL ROM 2114 (2) should read: INTERNAL RAM 2114 (2)
EXTERNAL ROM	(GAME CARTRIDGE)	8K X 8 BIT CAPABILITY	

CRT: SAMSUNG 240RB40 90 DEG. DEF. B&W VECTOR DEFLECTION

12 EXTERNAL GAME CARTRIDGES CURRENTLY
1 RESIDENT GAME

Second Controller available as an accessory

Game Cartridges include a screen overlay.

120V AC — 60Hz

DIMENSIONS: 9¾ X 11½ X 14½

WEIGHT: 15 Lbs.

V. CIRCUIT DESCRIPTION

As a general description, the HP3000 is a self-contained video game system intended for home use. The system includes its own 9" B&W monitor screen and 3" permanent magnet speaker. Four ROM type cartridges are available offering arcade type video and sound game play. No external TV receiver hookup is needed or provided for. A front panel storable controller allows control over the game via joystick and push button action switches. For two player operation a second controller identical to the single player controller is available as an accessory product. Both controllers attach to the main game console through nine wire coiled telephone style cables. There is a consumer power switch/volume control on the front panel as well as a game reset button. A consumer adjustable brightness control is located on the main console rear housing.

For the technical description which follows, the reader is encouraged to refer to the block diagram and schematic.

The HP3000 is a microprocessor based, vector scan system using a standard 9" black & white CRT as its video display device. The microprocessor (MPU) is the Motorola 68A09 device. The MPU operates at 1.6 MHz from a 6 MHz external Xtal. An internal divide by 4 circuit generates the MPU 1.6 MHz "E" clock signal used in the system. Program memory is stored in the 8K x 8 bit 2363 type ROM. This ROM contains common subroutines, the "executive" or assembler instructions plus one complete game.

Two 1K x 4 bit 2114 type static RAMS provide storage locations for data indicative of locations of objects, game status, and various other information needed by the microprocessor during game operation. Peripheral Interface Adaptor (PIA) Chip, has two 8 bit peripheral ports which interfaces the MPU with peripheral devices and external signals. One of the PIA ports interfaces the General Instrument AY-3-8912 sound-I.O. chip with the MPU and also drives the digital to analog converter chip MC1408. The other PIA port is used as control lines for the sound chip, selector control for the multiplex chip and as a means to read the A/D comparator that's used in the joystick successive approximation circuitry. Sound is either MPU generated directly or by use of the AY-3-8912 sound chip.

The AY-3-8912 sound chip is a programmable sound generator containing 3 tone generators and wave shaping circuitry. This chip also has a single 8 bit I.O. port used to read the status of each of the handcontrollers 4 action switches.

The standard TTL device types 74LS00 and 74LS32 are used as control line decoders to allow the MPU to select the appropriate circuit element to be addressed at any particular time.

The analog processing section includes digital to analog converter (DAC) chip type MC1408, dual 4 channel multiplexer/demultiplexer chip type CD4052, and dual channel op-amps types LF353 and LF347.

DAC chip MC1408 receives an 8 bit word at data terminals D0-D7. DAC output (pin 4) is current source. One section of IC LF353 is used to change this current to a voltage representative of the 8 bit digital word received by the DAC chip. The LF353 voltage is applied to an input of the dual 4 channel multiplexer (MUX) chip CD4052. This same voltage (designated "DAC" on the schematic) is the X-axis drive signal.

The CD4052 MUX chip serves two purposes: it selectively couples, under MPU control, the output of the DAC current/voltage converter to one of 4 places and is used to selectively couple the inputs from the joystick pots to the voltage comparator IC LF353.

The 4 places to which the "DAC" signal is coupled by the MUX are:

1) The Y-axis sample and hold IC LF347
2) The "O" reference charge capacitor
3) The Z-axis (brightness signal) sample and hold IC LF347
4) MPU sound resistive matrix

Each of these 4 signals is a voltage value representative of the 8 bit DAC input word for that function.

The joystick pot positions are sensed by a successive approximation process. The MUX chip selects each joystick pot input line and applies it to the plus input of comparator IC LF353. At the same time the MPU generates digital words that are changed to voltages by the DAC and current/voltage converter mentioned previously. These voltages are successfully applied to the comparator's minus input until the MPU generated voltage is equal to the joystick voltage. The MPU then recognizes the digital word representative of the comparison voltage and is able to establish a location for the joystick pot. The present position for each joystick pot is sensed in this manner. The pot position information is updated on a regular basis by the MPU.

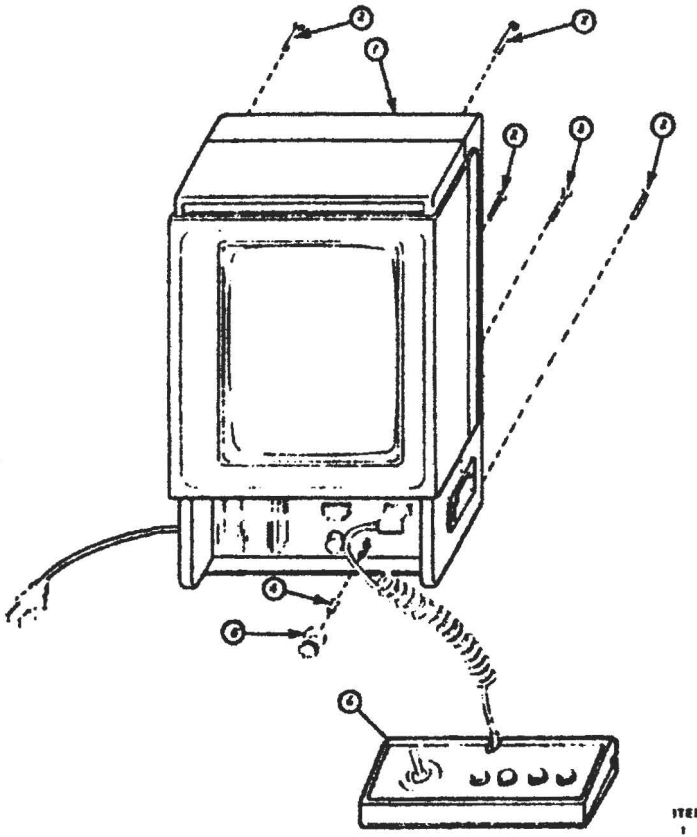
Returning to the X and Y axis drive signals, these signals are applied to X,Y integrator IC LF347 negative input pins through series analog switch types 4066B. The "zero" reference signal is applied to the positive inputs of the integrators. There are also analog switches across the integrator IC capacitors. The series analog switches are controlled by MPU signal RAMP and the parallel capacitor switches are controlled by MPU signal Zero 10. RAMP 10 determines when and for how long the X and Y axis voltage levels will be applied to the integrator amps. Zero 10 is used to discharge the X & Y axis integrator caps thus initializing them for the next signal to be integrated.

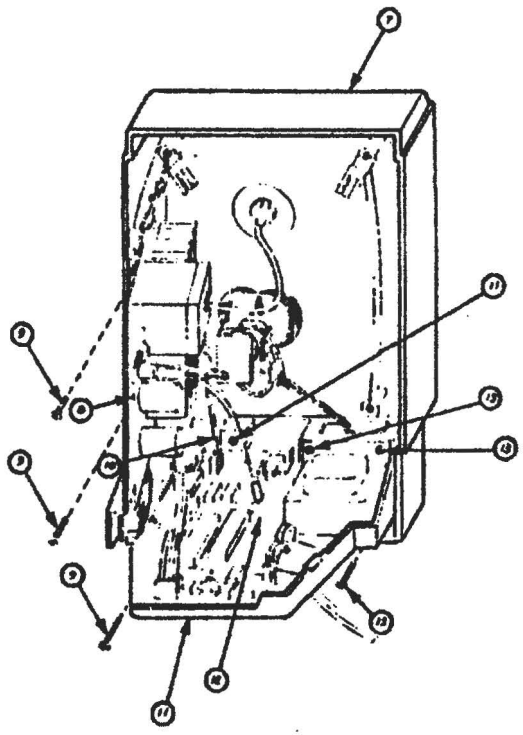
The outputs of the X,Y axis integrators are coupled through J-FET switches to IC LM379 deflection amplifiers. The LM379 operates as a voltage to current driver, the current through the deflection coils forming the electromagnetic field which deflects the CRT beam. To protect the CRT from spot burn in the event of a loss of deflection, the Y axis drive amplifiers output is detected and a deflection enable/disable signal generated. This signal controls the J-FET switches in series with the X,Y deflection amp inputs to reduce the scan drive signal in the event of a software or hardware failure plus discrete transistor type 2SC1921 operates to bias off the CRT.

Conventional full wave rectification and three terminal regulators are used in the low voltage power supply. A special negative DC source is generated by a voltage double-circuit which is used to supply a 13V to the DAC chip.

The high voltage is generated via an oscillator, drive transistor and flyback type transformer circuitry similar to what is commonly used in small black and white TV receivers.

Judicious use of bypass caps, RF filter chokes, ferrite beads, etc., has been used in the design to control RFI emissions.





ITEM	DESCRIPTION	PART NO.
1	RACK CASE ASSY	[illegible]
2	SCREW RACK TO FRONT CASE ASSY	[illegible]
3	SCREW RACK TO FRONT CASE ASSY	[illegible]
4	KNOB CLIP	[illegible]
5	KNOB, VOLUME ON OFF	[illegible]
6	[illegible]	[illegible]
7	FRONT CASE ASSY	[illegible]
8	PCB BRACKET ASSY	[illegible]
9	SCREW [illegible]	[illegible]
10	CABLE ASSY AUDIO	[illegible]
11	DIGITAL ANALOG PCB BRKT ASSY	[illegible]
12	CABLE ASSY, VIDEO	[illegible]
13	[illegible]	[illegible]

VECTREX EXPLODED VIEW

VI. DISASSEMBLY

A. Back Cover Removal

1. Lay the unit on a mat, CRT down.
2. Remove 5 screws from the back cover.
3. Remove the back cover.

B. Power Board Removal

1. Remove all connectors (5) and HV lead from the CRT.
2. Unsolder three leads (2 red, 1 white) from the bottom rear of the board at location EP104, 105 and 106. (Note: Two of these three leads go to the on/off volume control switch, the white lead goes to the power transformer (secondary C.T.)
3. Unsolder the Aquadag ground lead from the top rear of the board.
4. Unsolder ground jumper (braid) between the logic board and power board.
5. Remove two small Phillips head screws from the bottom of the board that secures it to the frame.
6. Slide board back and remove it from the frame.

C. Logic Board Removal

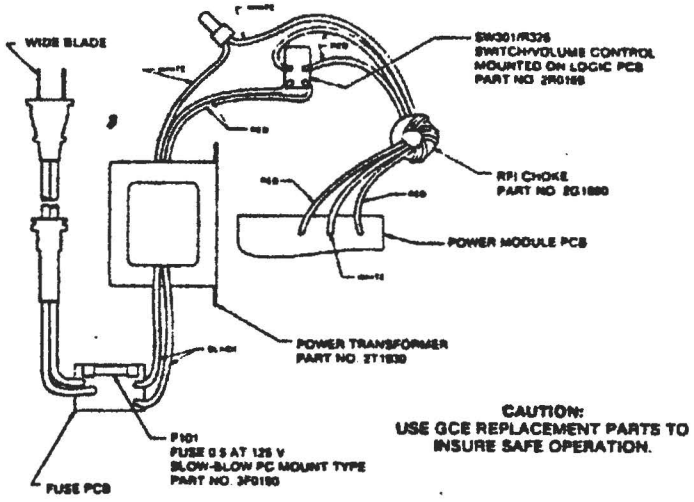
1. Remove all cable connectors from the top of the board (3).
2. Unsolder ground jumper between the logic board and power board at the logic board (left side).
3. Unsolder and remove the 3 power leads at the power board, EP104, 105 and 106. Unsolder 2 of these leads (red) plus 2 from the power transformer on the back of the on/off switch.
4. Remove the logic board mounting frame which includes the speaker, power transformer and reset button by removing retaining screws that hold the frame to the front cover. There are two screws located just above the power transformer bracket that must be removed also.
5. Remove the logic board mounting frame.
6. Unsolder the leads on the reset button.
7. Remove the retaining hardware on the front of the volume control, on/off switch.
8. Remove 4 small Phillips head screws on the top of the logic board that hold the board to the frame. One of the screws holds the plastic game cartridge guide to the logic board. Remove the guide.
9. Remove the logic board.

D. Power Transformer Removal

MAKE SURE A/C CORD IS UNPLUGGED FROM ALL POWER

1. Remove the small screw holding the fuse cover and remove the cover.
2. Remove the screw in the center of the Fuse PCB and remove the PCB.
3. Unsolder the 2 power and two primary leads from the fuse PCB.
4. Unsolder and remove 2 red leads from the on/off switch mounted behind the volume control.
5. Remove the splice on the white lead (secondary C.T.).
6. Remove the two screws holding the power transformer to the frame. Note the ground lead on the right hand screw (as viewed from the rear) has a ground lug on it.

(See Page 12 for Illustration)



POWER TRANSFORMER WIRING DIAGRAM

E. Speaker Removal

Follow steps 1 thru 7 under "Logic Board Removal."

1. After the frame is out, remove 2 small screws from the top of the speaker grill on the front of the frame. Lift up and out on the speaker grill. The speaker and grill will come out as one.
2. Unsolder speaker leads, note colors on + and – terminals and the position of the terminals in relation to the speaker grill and frame. It must be replaced the same way for lead routing.
3. Loosen retaining clip holding the speaker in.
4. Gently slide the speaker out of the two plastic retaining lips and remove. Rough handling at this point will break these two plastic retaining lips and cause problems in securing another speaker in the assembly.

VII. LOGIC BOARD ADJUSTMENTS

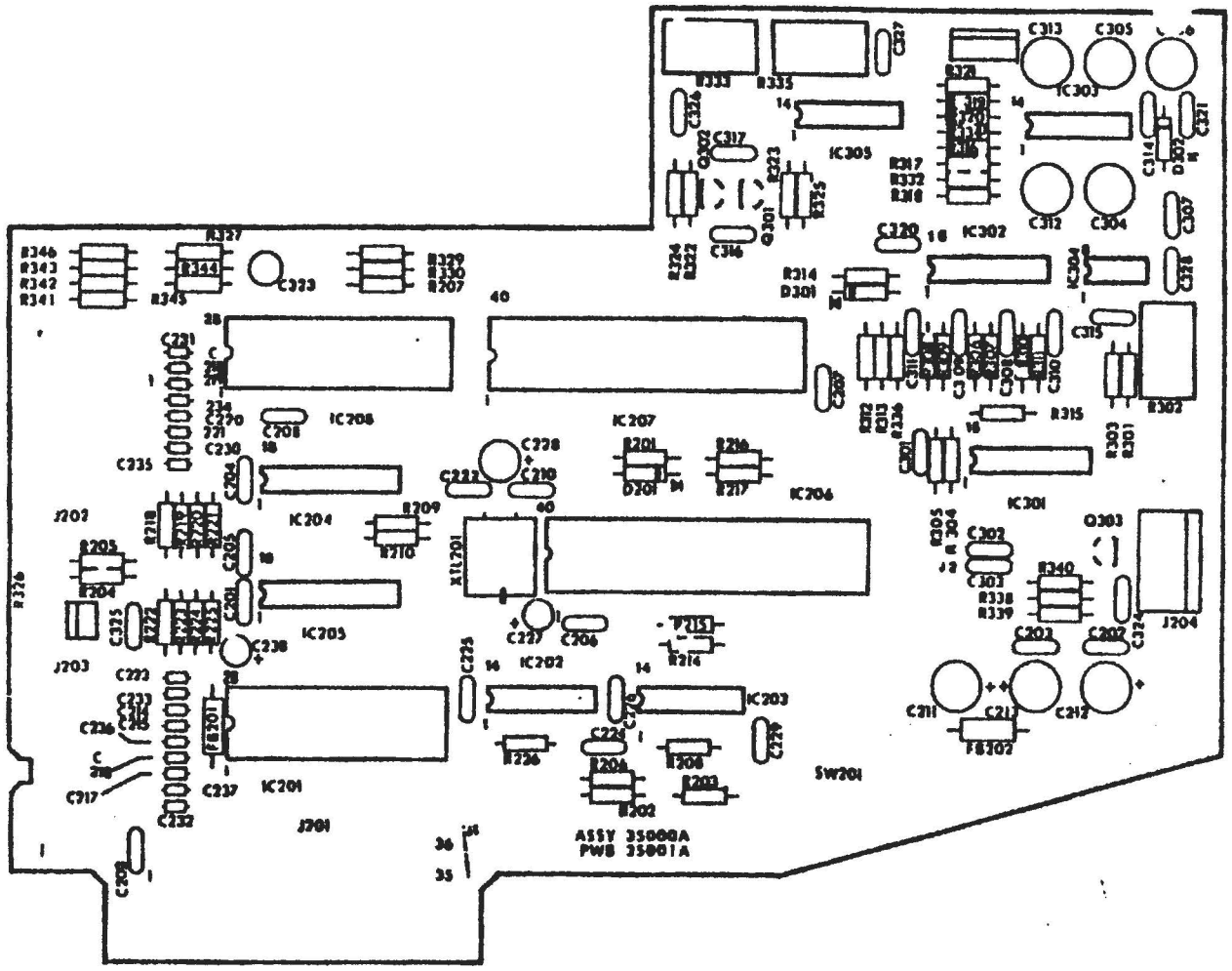
(See Test Cartridge Procedure, Page 18)

After the Logic Board has been replaced and installed in the VECTREX, the following adjustments must be made.

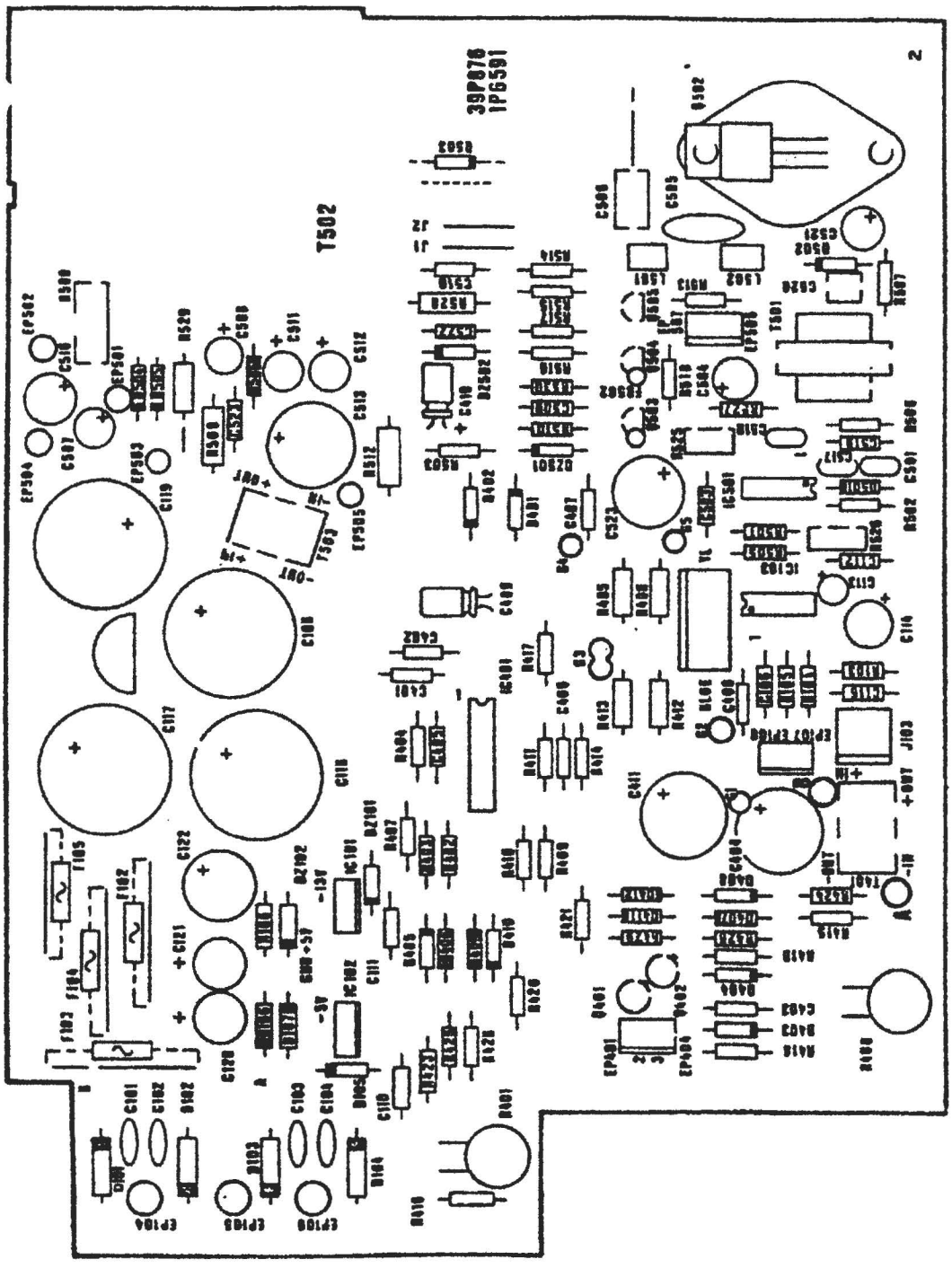
A. Initial Power-up — Install Test Cartridge

1. Plug the unit in and turn it on, volume as required. The CRT should display GCE title page and introductory tune should occur within fifteen (15) seconds of power-up. This should be followed by the test cartridge's title page.

2. Select "DAC Zero Test." These words will appear on the screen followed by a blank screen, the actual adjustment must be made during this blank screen interval. It will cycle back and forth between the word display and blank screen.



LOGIC BOARD COMPONENT LAYOUT



POWER BOARD COMPONENT LAYOUT

Set your scope on "DC" and the 5mv/div scale. Connect the ground lead to ground on the board and connect the probe to pin 1 of IC 304, adjust R302 "DAC OFFSET" POT for OVDC.

After the adjustment is completed, press the reset button.

It may now be necessary to recenter the picture as the DAC zeroing will affect it. Use the centering magnets on the rear of the deflection yoke and the "Linearity Pattern" in the test cartridge to set the centering. UNDER NO CIRCUMSTANCES IS R302 "DAC OFF-SET" TO BE USED TO HELP CENTER THE PICTURE.

C. Integrator Off-Set Test

Select the "Integrator Off-Set" test. Alternately adjust R333 "Y Rate Off-Set" and R335 "X Rate Off-Set" POTs to align the cross bars for intersection at the center of the diamond patterns. The bottom row of diamonds is the most critical and should be used to set these controls — all patterns should be within one (1) line width.

D. Sound Test

Select the "Sound Test." The display will say "CHANNEL A." You should then hear the sound start at a low frequency and increase in frequency. CHANNEL B and CHANNEL C will follow with identical tones.

The next title on the CRT will be "NOISE ALL CHANNELS" and this will be noise (static).

The screen will remain blank and two (2) tones will be heard. This is the "CPU SOUND" check. If any are missing the board must be repaired.

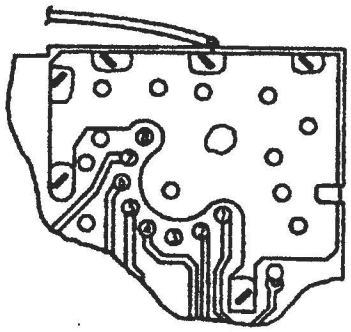
VIII. POWER BOARD ADJUSTMENTS

After installation of the Power Board Assembly make the following checks and, if necessary, adjustments.

A. Install the Test Cartridge and turn the VECTREX on. The GCE title page on the CRT and the introductory tune should occur within 15 seconds.

B. Turn the brightness to minimum (R509) and measure the high voltage; it should be 5.6KV+/- 150VDC.

C. To adjust the high voltage, connect an oscilloscope to T502 pin 7 and set the vertical at 20V/div and the horizontal at 10 usec/div.



D. Adjust R526 for minimum ringing in the displayed wave form.

E. Recheck the H.V. and adjust R525 to get proper reading (5.8KV+/- 105VDC).

F. Repeat the adjustments of these two controls until proper high voltage and minimum ringing are obtained.

G. **Deflection Protect Circuit Check (Beam Cut Off)**

Select the "Beam Cut Off" Test. Observe CRT monitor. The pattern will shrink in size, then disappear. In approximately 2-3 seconds, the pattern will reappear at about half-size and continues to increase in size and brightness until it is full-size, then the cycle will repeat. As the pattern decreases, the circuit time constraints prevent the protect switches from activating. When the circuit allows the switches to activate, they will not turn on until the brightness and deflection reach the design limits. This is when the pattern reappears at about half-size.

H. **Audio Amp. Check**

Select the "Sound Test" and with the volume control set at mid-range, monitor the audio. Sound chip channels A, B, C, and CPU sound test signals must be audible with no noticeable distortion.

IX. CRT/YOKE REMOVAL AND SET-UP

CRT removal, replacement and set-up is the same as most B/W TV tubes in most respects. The primary difference is in the centering technique.

After the CRT/yoke has been installed, do a preliminary centering using the test cartridge linearity pattern for a display and the centering ring magnets on the yoke.

The next step is to set the "DAC ZERO" as directed in Paragraph II of the LOGIC BOARD ADJUSTMENTS. After that is completed again recenter, if necessary, with the ring magnets on the yoke.

Adjust vertical and horizontal height so the linear lines are at the top and bottom, left and right edges of the CRT and front cover. Also see R401 and R408.

X. HAND CONTROL DISASSEMBLY

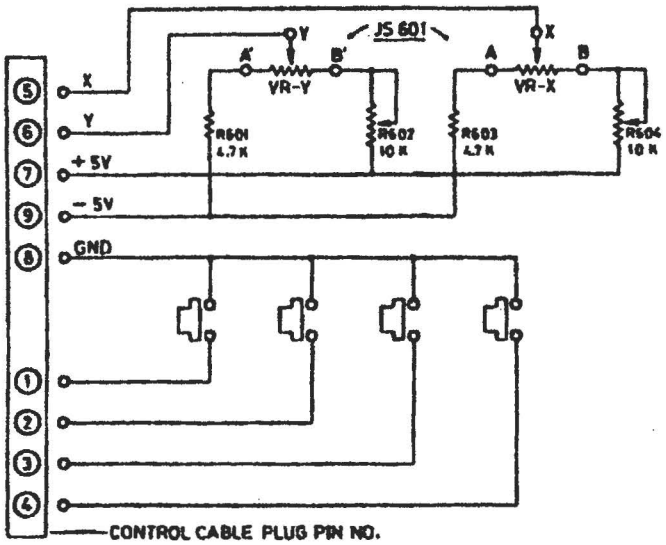
A. Remove the Hand Controller cord from the port in the Vectrex.

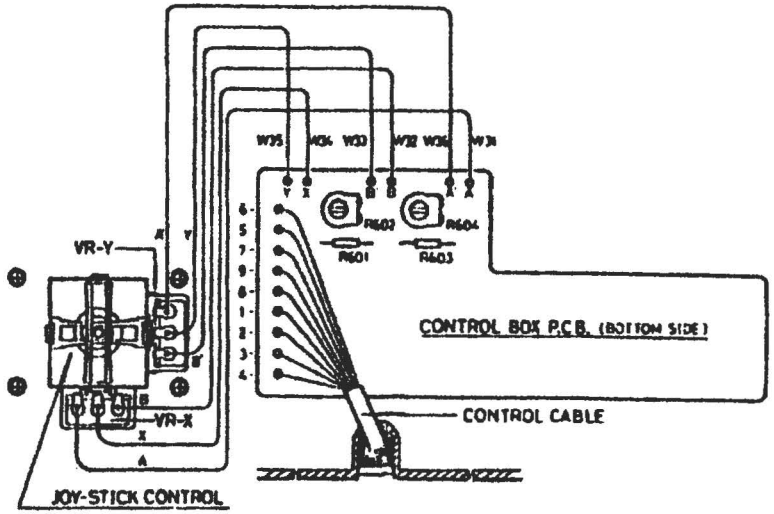
B. Remove the top inlay by inserting a tool between it and the case (use extreme caution not to injure yourself) and pry up one edge. Discard the tool and pull the inlay off.

C. Remove the five (5) screws and remove the top cover.

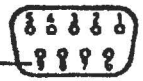
D. The P/C Board can be removed by taking out the screen in the center of the P/C Board.

The buttons and pad will then be easily removed.





PLUG CONNECTION



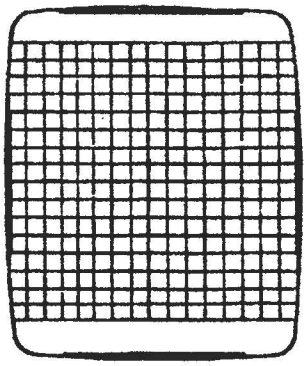
CONTROL BOX CONNECTION

CONTROL CABLE COLOR	TO CONTROL P.C.B. PIN NO.
WHITE	PIN 6
BROWN	PIN 5
GRAY	PIN 7
BLUE	PIN 9
BLACK	PIN 8
GREEN	PIN 1
YELLOW	PIN 2
ORANGE	PIN 3
RED	PIN 4

HAND CONTROL SCHEMATIC

XI. TEST CARTRIDGE PROCEDURE (REV 4)

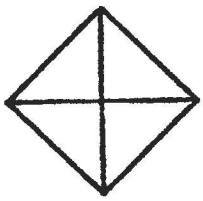
Install Test Cartridge
Turn unit on after VECTREX announcement
Title page
Then



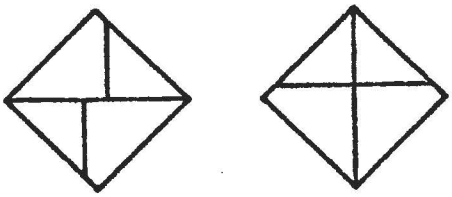
Linearity Pattern
Check for:
1. Pin Cushion
2. Barreling
3. Keystone
4. Vertical Size
5. Horizontal Size
6. Centering

The next test will have the words "Adjust DAC Offset" followed by a blank screen. This is an Oscilloscope Procedure and should be bypassed by pressing Button 3 or 4 twice in rapid succession.
NOTE: The DAC offset test cannot be escaped from via the controller keys unless the words "ADJUST DAC OFFSET" are on the screen. The words reappear for a short period every 6 seconds. Reset will allow escape at any time.

Good



Bad

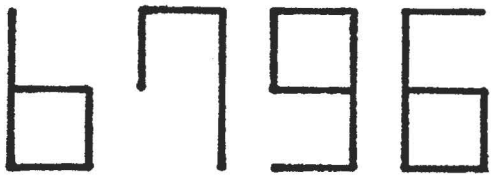


All lines must meet and be continuous.

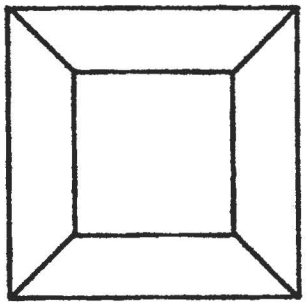
Press Button 3 or 4:

Wait a few seconds and the checksum will appear and read:

(B796) must appear.



Press Button 3 or 4
BEAM CUT OFF:



Pattern will decrease in size, then disappear, then reappear in about 2-3 seconds and be about half size and continue to increase until process repeats itself.

Press Button 3 or 4
SOUND TEST:

Words "CHANNEL A" will appear on bottom of the screen should have a one octave tone going from low to high smoothly and continuously.

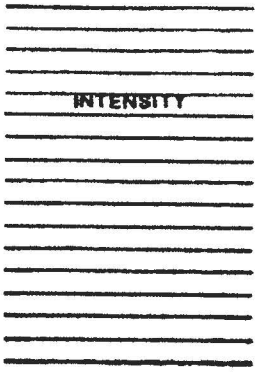
Words "CHANNEL B" will appear in the middle of the screen with same tone as above.

Words "CHANNEL C" will appear at the top of the screen with the same tone as above.

Words will appear in the center of the screen "NOISE ALL CHANNELS." There will be sound like static for a short duration. Screen will go blank, then you will hear two tones. This is the CPU sound check.

Press either Button 3 or 4:

The word "INTENSITY" will appear with 17 equally spaced lines running horizontally across the screen. The 2nd, 3rd and 4th line from the top should be extinguished; the 5th from the top sits right on top of the word "INTENSITY," and should be visible.



Press Button 3 or 4: **FOCUS TEST**

The focus pattern will appear with symbols.

These symbols should be clearly focused in the center of the screen with minimum unfocus on four (4) corner symbols.

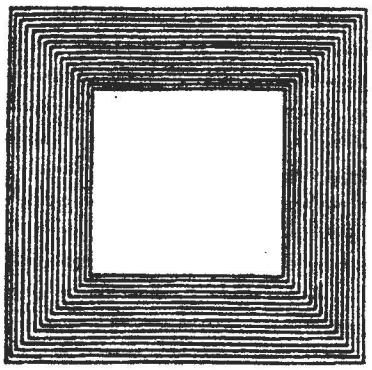
Press Button 3 or 4:

A border of triangles will appear on the screen with (DISTORTION) words in the center, if dissymmetry is apparent, make a note.

Press Button 3 or 4:

A rectangle will appear with the words (DISTORTION 2) in the center.

There are 16 rectangles traced around each other.



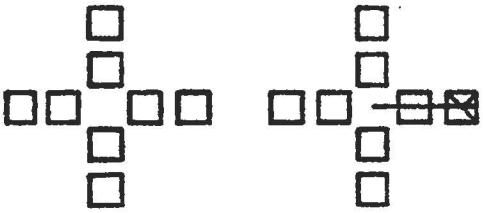
The spacing of each side must be the same for each succeeding rectangle and the overall pattern must be symmetrical.

Press Button 3 or 4.

The key and Joystick pattern will appear:

KEY

JOYSTICK



Press Buttons 1 thru 4 consecutively: The proper symbol must appear as each button is pushed in the appropriate square on the top row.

Displace the Joystick 90 degrees to the right slowly. The same symbol as above must appear first in the box closest to the center, then disappear and the outer box must indicate the symbol. There will also be a line that extends from the center of the diagram in the direction the Joystick is pushed. Check all four directions and check that the appropriate box lights up. Also slowly sweep the Joystick 360 degrees at its limits and make sure the line moves smoothly with no dropouts at any point.

Remove the hand control cord from its port and move it to the left port. Repeat the above test.

NOTE: If the left joystick is not plugged in, one of the inner boxes will light because of the 1MEG pull-up resistor internal to the VECTREX.

To escape from the controller test, hold Key 1 down while pressing either Key 3 (to back up to the distortion test) or Key 4 (to advance to the grid pattern).

XII. GENERAL CONSUMER ELECTRONICS PARTS LIST

ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
	Assembly List			
1	1	Console Assembly	4X5280	
2	1	Remote Control Box Assembly	4X5290	
3	1	Logic Board Assembly	4X5450	
4	1	Power Board Assembly	4X5460	
5	1	CRT Board Assembly	Beginning From 4X5470	
	Power Board Assembly (4X5460)			
		Carbon Film Resistor		
1	1	4.7 ohm	3R0091	R507
2	1	10 ohm	3R0151	R103, 508
3	1	39 ohm	3R0231	R530
4	3	100 ohm	3R0321	R417, 514, 527
5	1	47 ohm	3R0241	R505
6	1	150 ohm	3R0341	R515
7	2	220 ohm	3R0381	R513, 5106
8	1	1 K ohm	3R0511	R503
9	4	1.5K ohm	3R0531	R423, 424, 415, 416
10	2	2.2K ohm	3R0561	R501, 510
11	4	3.3K ohm	3R0691	R404, 411, 518, 502
12	2	4.7K ohm	3R0641	R428, 429
13	1	6.8K ohm	3R0681	R104
14	3	10K ohm	3R0711	R425, 426, 516
15	1	820K ohm	3R1131	R530
16	4	47K ohm	3R0861	R105, 420, 421, 517
17	2	3.3M ohm	3R1211	R418, 419
		CARBON FILM RESISTOR +5% 1/2W		
18	1	1K ohm	3R0514	R512
19	2	1.5K ohm	3R0035	R405, 412 NON-INDUCTIVE
20	1	10K ohm	3R0714	R528
21	1	470K ohm	3R1074	R508
		CARBON FILM RESISTOR +5% 1W		
22	1	3.3 ohm	3R0075	R529
		METAL FILM RESISTOR		
23	2	0.22 ohm +10% 1W	3R0005	R416, 413 NON-INDUCTIVE
24	2	3.3K ohm +1% 1/4W	3R0593	R407, 414
25	2	15K ohm +1% 1/4W	3R0743	R402, 409
26	2	1M ohm +1% 1/4W	3R1142	R403, 410

ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
/	2	2K ohm +10% 1/4W	3R2191	R401, 408 (H-TYPE)
28	1	250K ohm +10% 1/4W	3R2192	R509 (V-TYPE, LINEAR)
29	2	4K ohm +10% 1/8W	3R2200	R525, 526 (V-TYPE)
		CERAMIC DISC CAPACITOR +80 -20%		
30	1	0.01uF 500V	3C0840	C505
31	4	0.0047uF 500V	3C0821	C101-104
32	2	220pF 50V	3C2151	C106, 510
		MULTILAYER CERAMIC CAPACITOR		
33	15	0.1uF +20% 50V	3C0865	C110, 111, 112, 116, 401, 402, 403, 405, 406, 407, 408, 503, 509, 519, 522
		ALUMINUM ELECTROLYTIC CAPACITOR (+80 -20%)		
34	1	0.47uF 50V	3E0035	C410
35	1	0.47uF 50V	3E0036	C409
36	1	3.3uF 50V	3E0060	C516 (S.V.)
37	1	4.7uF 100V	3E0076	C507 (S.V.)
38	1	10uF 16V	3E0097	C113
39	2	15uF 63V	3E0100	C508 (S.V.)
40	2	22uF 16V	3E0102	C511, 512
41	3	47uF 16V	3E0132	C504, 521, 120
42	1	47uF 25V	3E0129	C121
43	1	220uF 16V	3E0152	C114 (S.V.)
44	1	220uF 25V	3E0153	C122 (S.V.)
45	1	470uF 16V	3E0172	C523 (S.V.)
46	3	1000uF 25V	3E0183	C404, 411, 513 (S.V.)
47	2	4700uF 25V	3E0262	C109, 119 (S.V.)
48	2	10000uF 25V	3E0266	C117, 118 (S.V.)
		POLYPROPYLENE CAPACITOR		
49	1	0.082uF +/- 10% 400V	3E0731	C506
		MYLAR CAPACITOR (50V Min.)		
50	1	0.0022uF +/- 10%	3C1031	C501
51	1	0.0033uF +/- 10%	3C1041	C517
52	2	0.033uF +/- 10%	3C1201	C518, 520
		RECTIFIER		
53	1	1N4001 1A 50 PIV	3M1032	D105
54	4	1N5624 3A 100 PIV	3M4550	D101, 102, 103, 104
55	3	1N4005 1A 600 PIV	3M4570	D503, 504, 506
		ZENER DIODE		
56	2	6.2V +/- 5% 1N753 400mW	3M1240	DZ501, 502
57	1	6.2V +/- 10% 5W 1N5341	3M1244	DZ101

ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
	1	13V +/- 1V @ 10mA 400mW	3M1322	DZ102
	16	1N4148 Signal Diode	3M1051	D405 thru 412 D106, 107 thru D404 D501, 502
60	1	MR 852 Damper Diode	3M4560	D503
		TRANSISTOR		
61	1	2N3904 NPN	3M3280	Q505
62	1	2N3905 PNP	3M3270	Q504
63	1	BU407 NPN	3M3280	Q502
64	1	2SC1921 NPN	3M3290	Q503
		N-CHANNEL JFET		
65	2	2N3824	3M3300	Q401, 402
66	1	DRIVING X'FORMER (EI-19)	2T1420	T501
67	1	LM379 Dual 6W AMP.	1V2970	IC401
68	1	LM386-3 1.5W AMP.	1V2980	IC103
69	1	LM555 Timer	1V1840	IC501
		VOLTAGE REGULATOR		
70	1	7805 +5V OR LM340T +5V	1V2533 1V6802	IC101
71	1	79005 - 5V	1V2532	IC102
72	1	Power Board (Solder Plate)	1P6592	PCB
73	2	Ferrite Cylinder 03 x 6mm	2B0530	FB501, 502
		R.F. Choke Coil (1/4" I.D. Air Core)		
74	2	7 Turn AWG#24 0.3 uH 0.8A	2G2030	L501, 502
75	2	Full Turn 2 x 1.5 mH 3A	2G1690	T401, 503
*76	1	Flyback Transformer	2T1410	T502
77	1	CRT Board Assembly	4X5440	
78	1	Video Cable Assembly	4Y0820	
79	1	Audio Cable Assembly	4Y0830	
80	1	DC Power Cable Assembly	4Y0790	
		MOLEX WAFER		
81	1	2 PIN (09-74-1021)	2J0390	
82	1	4 PIN (09-74-1041)	2J0360	
83	2	2 PIN (22-04-1021)	2J0430	
84	1	4 PIN (22-04-1041)	2J0380	
85	2	Insulation Spacer (for LM379)	4D4960	
86	2	Insulation Spacer (for Regulators)	4D4970	

*Critical Safety Component — Must Use Exact Replacement

ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
7	4	Heat Sink Supporter	4T8110	
88	1	Insulation Sheet (for LM79M05)	4K1161	
89	1	Insulation Sheet (for LM379)	4K1850	
90	AR	Silicone Compound (Dow Corning 340)		
91	1	30W Heat Sink	4T8090	
92	1	Shield Can	4T8070	
93	1	Shield Can Cover	4T8080	
94	1	Grommet	4D4840	
95	6	M3.0 x 16 x 0.5P CR-P/H	6S2440	
96	2	4-40NC-2B x 14 P/H	6S3091	
97	2	M2.6 x 6 x 0.45 P CR-P/H	6S1130	
98	6	Spring Washer 05.5 x 3.2 x 0.7	6W0020	
99	2	Spring Washer 02.8	6W0960	
100	6	03.2 x 06 x 0.6T	6W0190	
101	2	02.8 x 07 x 0.5 Plain Washer	6W0310	
102	1	M3.0 x 6 x 0.5P CR-P/H		
103	1	M3.0 x 0.5P Nut	6N0050	
104	1	03.2 Spring Washer		
105	6	Heat Sink Supporter	4T8110	
	Cable Assemblies			
1		DC Power Cable	4Y0790	
2		Deflection Coil Cable	4Y0800	
3		Speaker Cable	4Y0810	
4		Video Cable	4Y0820	
5		Audio Cable	4Y0830	
6		Ferrite Toroid Assembly	2G1680	
	Logic Board Assembly (4X5450)			
		Carbon Film Resistor +/- 5% 1/4W		
1	5	33 ohm	3R0221	R214-217, 226
2	2	75 ohm	3R0281	R318, 321
3	1	100 ohm	3R0321	R208
4	2	220 ohm	3R0381	R317, 320
5	12	680 ohm	3R0481	R218-225, 306, 308, 310, 312
6	3	1K ohm	3R0511	R341, 343
7	2	2.2K ohm	3R0561	R314, 315
8	11	3.3K ohm	3R0591	R202-207, 209, 322, 324, 210, 330
9	2	4.7K ohm	3R0641	R338, 339
10	1	6.8K ohm	3R0681	R329
11	1	10K ohm	3R0711	R340
12	2	15K ohm	3R0741	R323, 325
13	2	22K ohm	3R0771	R201
14	4	33K ohm	3R0821	R327, 344-346
15	4	330K ohm	3R1051	R307, 309, 311, 313
16	1	1M ohm	3R1141	R336
17	2	3.3M ohm	3R1211	R332, 334
		Metal Film Resistor +/- 1% 1/4W		
18	3	2.7K ohm	3R0580	R304, 305, 301
19	1	3.6K ohm	3R0610	R303

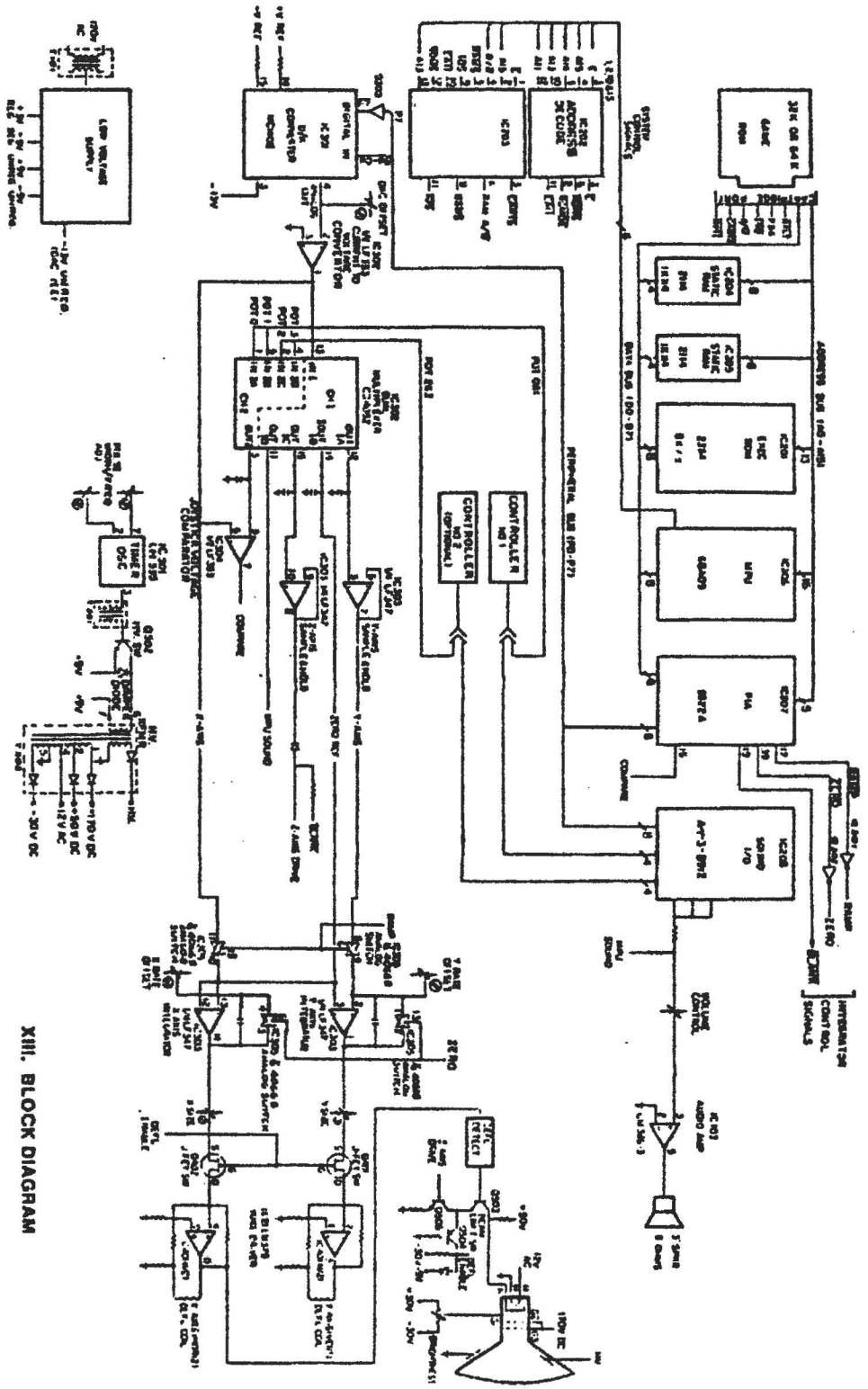
ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
1	2	10K ohm	3R0713	R316, 319
		Semi-Fixed Resistor (Vert. Type) 016MM		
21	3	10K ohm +/- 10% 1/4W	3R2210	R333, 335, 302
22	1	10K Volume Control W/Rotary SW	2R0169	R328
		Multilayer Ceramic Cap.		
23	13	0.001uF +/- 20% 50V	3C2120	C230-233, 214-221, 229
24	28	0.1uF +/- 20% 50V	3C0865	C201-209, 223-226, 234-237, 301, 302, 316, 317, 320, 321, 326-328, 324, 314
		Tantalum Capacitor		
25	1	22uF +/- 10% 16V	3C1640	C227
26	1	10uF +/- 10% 16V	3C1543	C238
		Aluminum Capacitor +80 -20%		
27	1	4.7uF 16V	3E0072	C323
28	1	220uF 16V	3E0152	C211
29	1	10uF 16V	3E0097	C228
30	2	100uF 16V	3E0142	C212, 213
		Ceramic Disc Capacitor		
31	2	20pF +/- 1pF 50V	3C0160	C210, 222
32	2	47pF +/- 20% 50V	3C0284	C303, 315
33	1	220pF +80 -20% 50V	3C2151	C325
		Polystyrene Capacitor		
34	5	0.01uF +/- 5% 50V	3E0710	C304-306, 312, 313
35	1	Logic Board (Solder Plate)	1P6600	PCB
		Mylar Capacitor (50V Min.)		
36	5	0.001uf +/- 10%	3C1012	C307-311
37	3	1N4148 Signal Diode	3M1051	D201, 301, 302
		Transistor		
38	1	2N3904 NPN	3M3260	Q303
39	2	2N3905 PNP	3M3270	Q301, 302
		Crystal (Without Coating)		
40	1	6.0000 MHz Crystal	3K0162	X'TAL 201
41	2	40 PIN I.C. Socket	2J0190	
42	2	28 PIN I.C. Socket	2J0010	
43	2	18 PIN I.C. Socket	2J0200	
44	2	16 PIN I.C. Socket	2J0011	
45	4	14 PIN I.C. Socket	2J0260	
46	1	8 PIN I.C. Socket	2J0440	

ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
7	1	Power Board Bracket	4D4540	
8	1	Speaker Frame	4D4530	
9	2	Fixing Stud	4D4570	
10	4	Rubber Stand (Console)	4D4600	
11	4	Rubber Stand Spacer	4T8030	
12	4	Rubber – CRT	4D4580	
13	1	Knob Clip	4T3660	
14	2	Grounding Plate, CRT	4T7861	
15	1	Board Compartment Cover	4F1520	
16	1	Speaker Clamp	4T7850	
17	1	Heat Sink	4T7960	
18	1	Toroid Cover	4F1600	
*19	1	CRT 240R84	1V4500	
*20	1	Deflection Yoke (450uH)	2G17002	
*21	1	Power Transformer (EI-66)	2T1930	T101
22	1	Ferrite Toroid Assembly	2G1680	T102
23	1	Close End Connector	5C0370	
24	1	Speaker 3" 8 ohm 1W	2L0101	SPKR 101
25	1	Reset Switch	2S0360	MS-029
26	1	Power Cord SPT-2	3W1690	
*27	1	Fuse 0.5A T 125V	3F0190	
28	1	Fuse P.C.B.	1P6582	
29	1	Deflection Coil Cable Assembly	4Y0800	
30	1	Speaker Cable Assembly	4Y0810	
31	1	Connecting Patch (LEG)	4T2390	
		CRT Board Assembly (4X5440)		
		Carbon Film Resistor +/- 5% 1/2W		
1	3	1.5K ohm	3R0534	R522-524
2	1	100K ohm	3R0954	R519
3	2	220K ohm	3R1004	R520, 521
		Ceramic Capacitor +80 -10%		
4	2	0.01uF 1 KV	3C0841	R514, 515
5	3	Spark Gap 1.5 KV	3F0180	SG501-503
6	1	CRT P.C.B.	1P4950	
		AWG #22 Strand Wire uL 1007	FR-1	
7	1	190mm Blue	3W7476	From CRT to Pwr Bd
8	1	190mm Gray	3W7478	From CRT to Pwr Bd
9	1	190mm Black	3W7470	From CRT to Pwr Bd
10	1	190mm Brown	3W7471	From CRT to Pwr Bd
11	1	7 PIN CRT Socket	2J0500	SMK P501 P.C. MOUNT

*Critical Safety Component — Must Use Exact Replacement

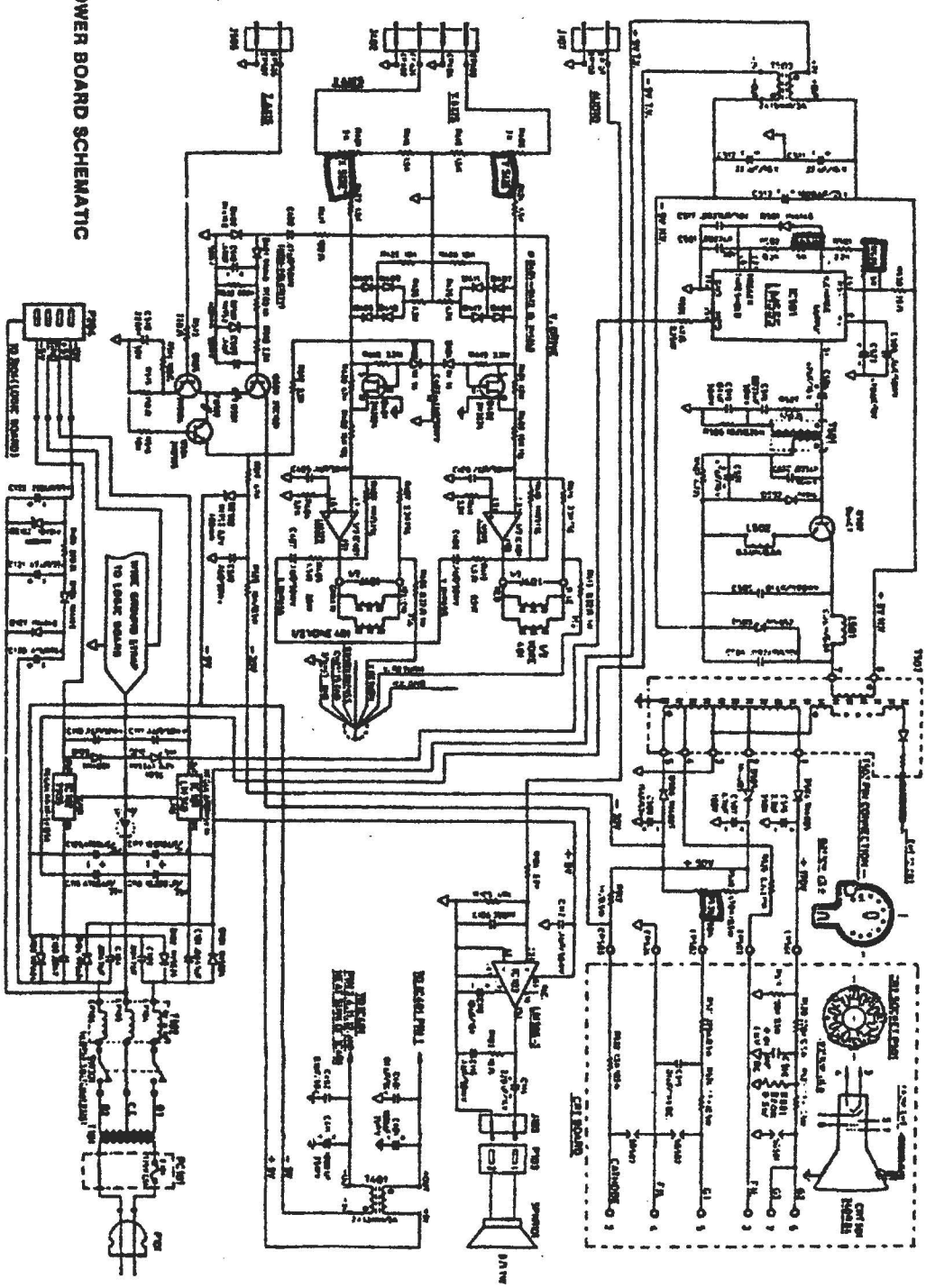
ITEM	QTY.	DESCRIPTION	PART NO.	CIRCUIT DESIGNATION
7	1	Power Board Bracket	4D4540	
8	1	Speaker Frame	4D4530	
9	2	Fixing Stud	4D4570	
10	4	Rubber Stand (Console)	4D4600	
11	4	Rubber Stand Spacer	4T8030	
12	4	Rubber – CRT	4D4580	
13	1	Knob Clip	4T3660	
14	2	Grounding Plate, CRT	4T7861	
15	1	Board Compartment Cover	4F1520	
16	1	Speaker Clamp	4T7850	
17	1	Heat Sink	4T7960	
18	1	Toroid Cover	4F1600	
*19	1	CRT 240R84	1V4500	
*20	1	Deflection Yoke (450uH)	2G17002	
*21	1	Power Transformer (EI-66)	2T1930	T101
22	1	Ferrite Toroid Assembly	2G1680	T102
23	1	Close End Connector	5C0370	
24	1	Speaker 3" 8 ohm 1W	2L0101	SPKR 101
25	1	Reset Switch	2S0360	MS-029
26	1	Power Cord SPT-2	3W1690	
*27	1	Fuse 0.5A T 125V	3F0190	
28	1	Fuse P.C.B.	1P6582	
29	1	Deflection Coil Cable Assembly	4Y0800	
30	1	Speaker Cable Assembly	4Y0810	
31	1	Connecting Patch (LEG)	4T2390	
		CRT Board Assembly (4X5440)		
		Carbon Film Resistor +/- 5% 1/2W		
1	3	1.5K ohm	3R0534	R522-524
2	1	100K ohm	3R0954	R519
3	2	220K ohm	3R1004	R520, 521
		Ceramic Capacitor +80 -10%		
4	2	0.01uF 1 KV	3C0841	R514, 515
5	3	Spark Gap 1.5 KV	3F0180	SG501-503
6	1	CRT P.C.B.	1P4950	
		AWG #22 Strand Wire uL 1007	FR-1	
7	1	190mm Blue	3W7476	From CRT to Pwr Bd
8	1	190mm Gray	3W7478	From CRT to Pwr Bd
9	1	190mm Black	3W7470	From CRT to Pwr Bd
10	1	190mm Brown	3W7471	From CRT to Pwr Bd
11	1	7 PIN CRT Socket	2J0500	SMK P501 P.C. MOUNT

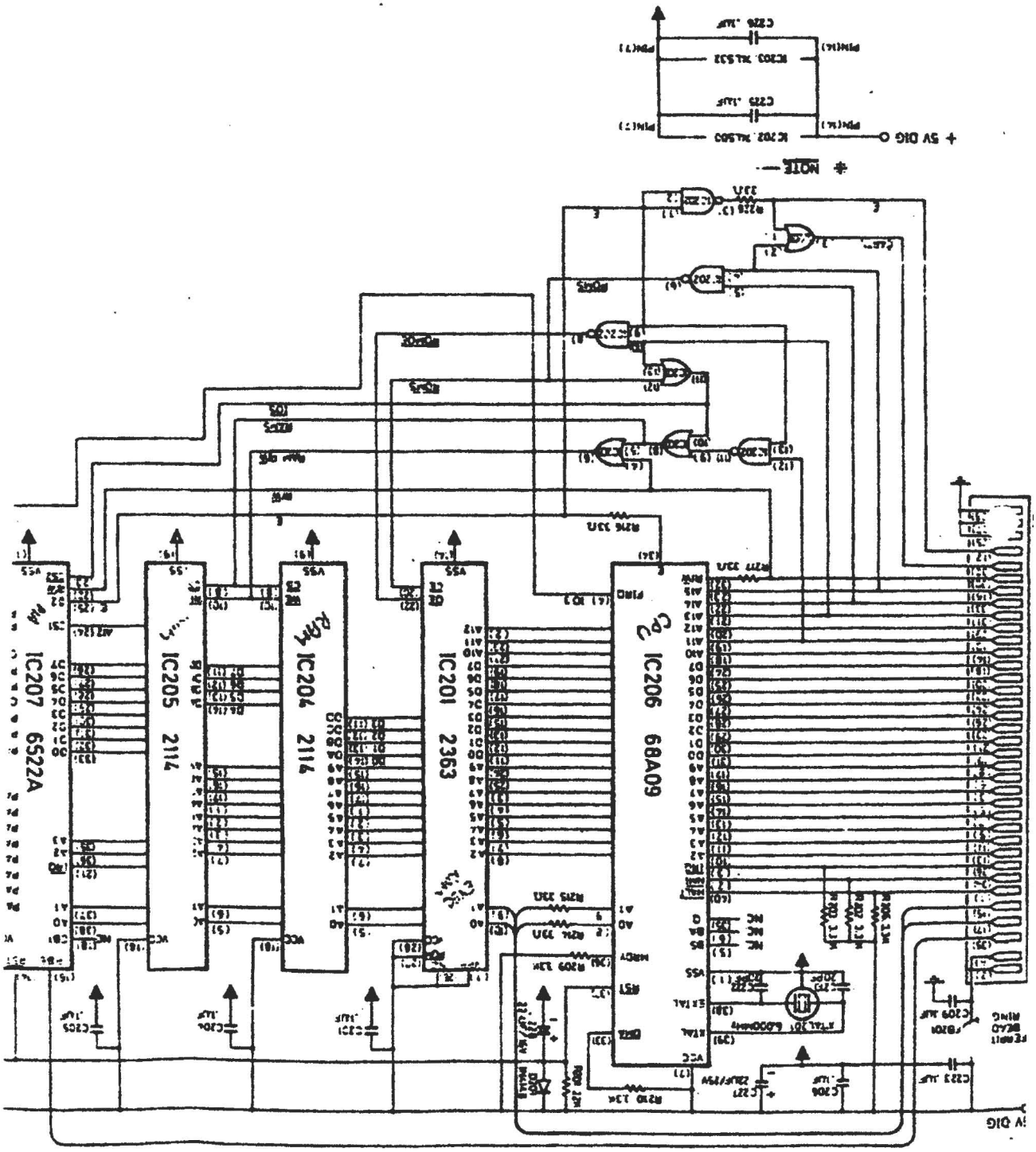
*Critical Safety Component — Must Use Exact Replacement

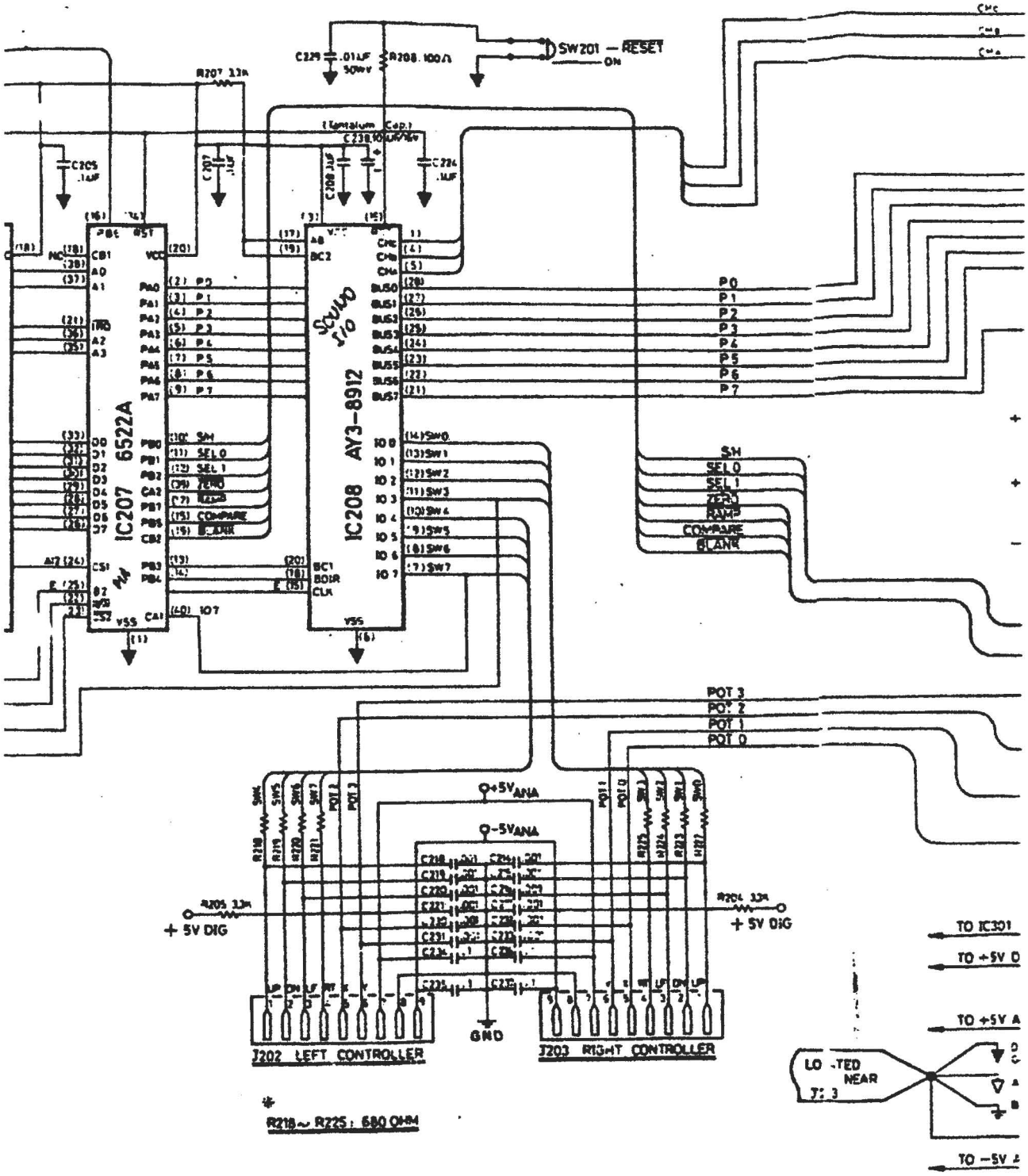


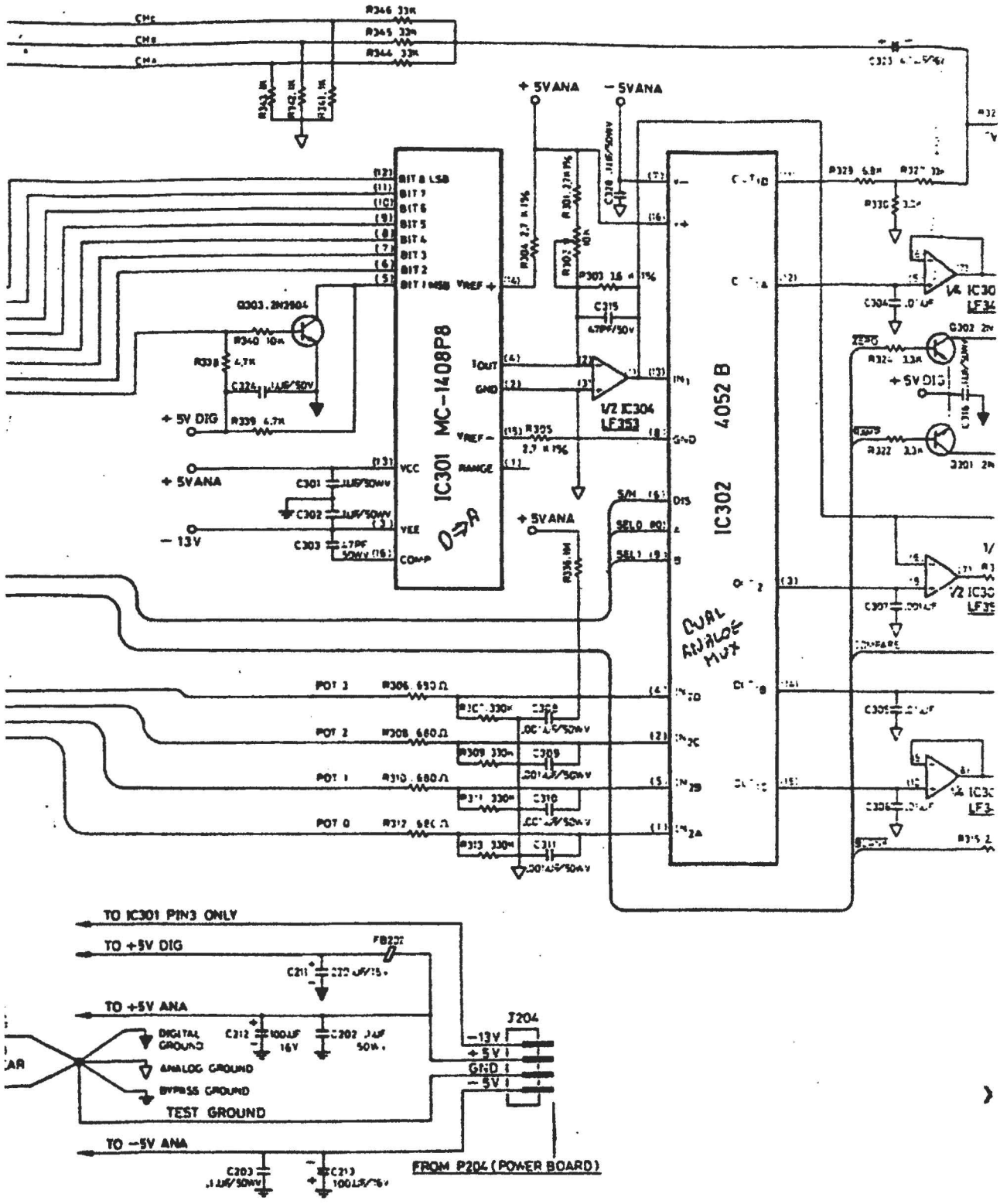
XIII. BLOCK DIAGRAM

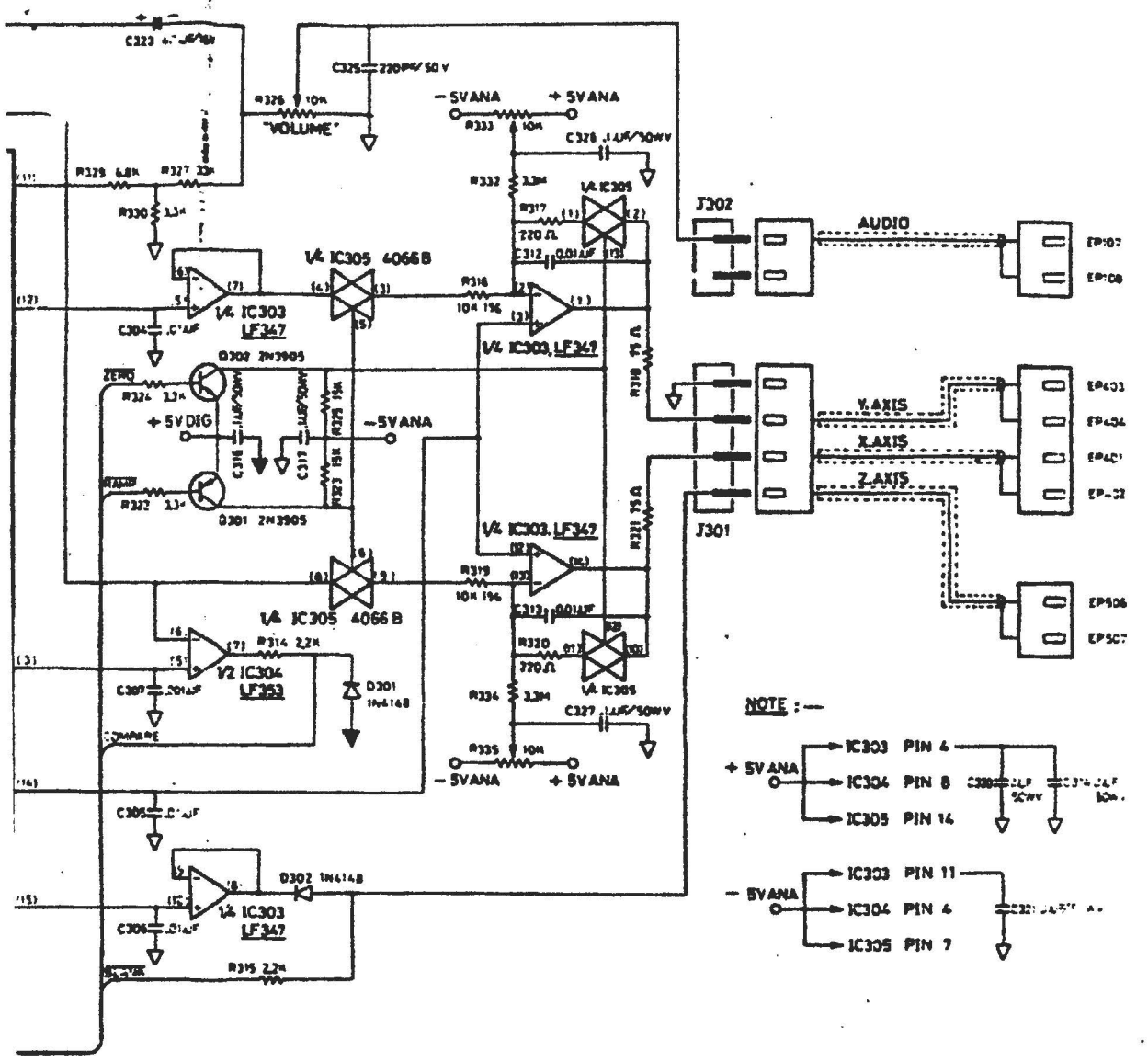
XIV. POWER BOARD SCHEMATIC




IC207 6522A
IC205 2114
IC204 2114
IC201 2363
IC206 68A09
CPU
+ 5V DIG
NOTE

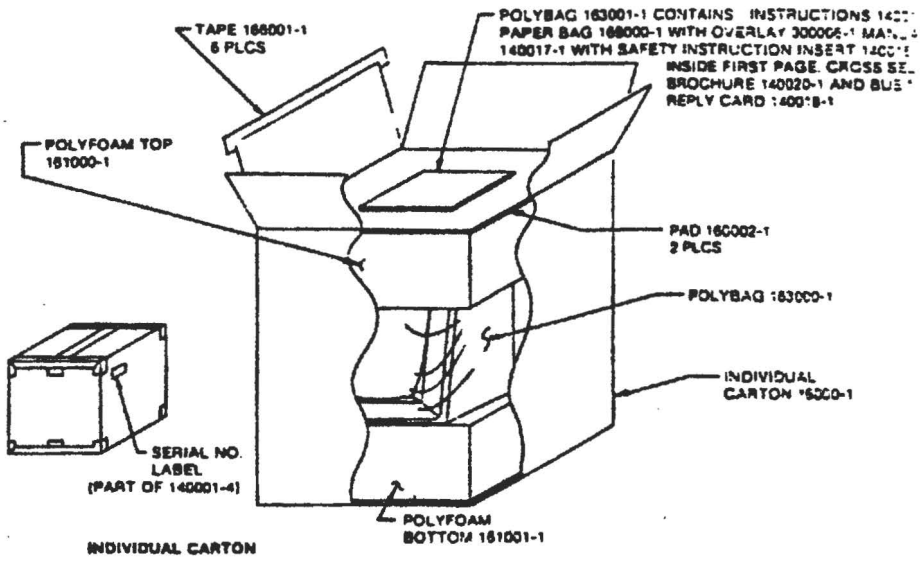

SW201 — RESET
ON
IC207 6522A
IC208 AY3-8912
Sound I/O
P0
P1
P2
P3
P4
P5
P6
P7
SH
SEL 0
SEL 1
ZERO
RAMP
COMPARE
BLANK
POT 3
POT 2
POT 1
POT 0
+5VANA
-5VANA
+5V DIG
+5V DIG
GND
J202 LEFT CONTROLLER
J203 RIGHT CONTROLLER
TO IC301
TO +5V D
TO +5V A
TO −5V
R218 ~ R225 : 680 OHM


+5VANA
-5VANA
IC301 MC-1408P8
D=A
IC302 4052 B
DUAL ANALOG MUX
Q303 2N3904
+5V DIG
+5VANA
-13V
POT 3
POT 2
POT 1
POT 0
TO IC301 PIN3 ONLY
TO +5V DIG
TO +5V ANA
DIGITAL GROUND
ANALOG GROUND
BYPASS GROUND
TEST GROUND
TO -5V ANA
J204
-13V
+5V
GND
-5V
FROM P204 (POWER BOARD)



XV. LOGIC BOARD SCHEMATIC

XVI. PACKAGING



VECTREX PACKAGING ASSEMBLY

VECTREX\TEXT\PINOUT.TXT

This information is empirical. Real information is available by scrutinzing the schematics in the nth-generation copy of the service manual...

The following is the Vectrex Cartridge connector pinout.

Pin #	info source	designation
1	Ohm	HALT* (MPU 40)
2	Ohm	Vcc (MPU 7)
3	cart	A7 (ROM 1)
4	Ohm	Vcc (MPU 7)
5	cart	A6 (ROM 2)
6	cart	A8 (ROM 23)
7	cart	A5 (ROM 3)
8	cart	A9 (ROM 22)
9	cart	A4 (ROM 4)
10	cart	A11 (ROM 21)
11	cart	A3 (ROM 5)
12	cart	OE*/Vpp (ROM 20) (LS32 1)
13	cart	A2 (ROM 6)
14	cart	A10 (ROM 19)
15	cart	A1 (ROM 7)
16	cart	CE* (ROM 18) A15 (MPU 23) (LS32 2)
17	cart	A0 (ROM 8)
18	cart	D7 (ROM 17)
19	cart	D0 (ROM 9)
20	cart	D6 (ROM 16)
21	cart	D1 (ROM 10)
22	cart	D5 (ROM 15)
23	cart	D2 (ROM 11)
24	cart	D4 (ROM 14)
25	cart	GND
26	cart	D3 (ROM 13)
27	cart	GND
28	cart	GND (ROM 12)
29	Ohm	A12 (MPU 20)
30	Ohm	(PIA 22) (35 Ohms to R/W* (MPU 32))
31	Ohm	A13 (MPU 21)
32	Ohm	(LS32 3)
33	Ohm	A14 (MPU 22)
34	Ohm	NMI* (MPU 2)
35	Ohm	(PIA 16)
36	Ohm	IRQ* (MPU 3)

VECTREX\TEXT\EPROM.TXT

From umn-cs!stolaf!ihnp4!harpo!seismo!hao!hplabs!oliveb!oliven!gnome Sun Feb
 6 00:28:16 206
Relay-Version: version B 2.10.1 6/24/83; site umn-ucc.UUCP
Posting-Version: version B 2.10 5/3/83; site oliven.UUCP
Path: umn-ucc!umn-cs!stolaf!ihnp4!harpo!seismo!hao!hplabs!oliveb!oliven!gnome
From: gnome@oliven.UUCP (Gary Traveis)
Newsgroups: net.micro,net.games.video
Subject: Vectrex ROM to EPROM
Message-ID: <116@oliven.UUCP>
Date: Fri, 23-Mar-84 16:03:08 CST
Date-Received: Mon, 2-Apr-84 20:05:05 CST
Organization: Olivetti ATC., Cupertino, Ca
Lines: 97

(sacrificial line)
The following is a first stab at documenting Vectrex
hacks that have been implemented. This is mainly for
encouraging the interchange of ideas and/or programs
for a unique and yet endangered species - the Vectrex.

****

Ok, so you want to run EPROM resident stuff on the
Vectrex! First, lets define the EPROM.

The EPROM I have been using in these tests have been
Intel 2732-2 4K x8 MOS EPROMs. Anything slower and
there may be problems.

1) Open the cartridge by taking the single screw out
of the bottom. Then, carefully unsnap the top and bottom
halves of the cartridge case.

2) You now have a little green and gold printed circuit board
with a 24 pin (Sharp or GI) ROM mounted on it.

You now have two routes possible. You can try to remove the
chip from the board unharmed (not really too bad with a good
solder-sucker tool) or simply cut the little guy out with a
sharp pair of diagonal cutters. I removed the chip, the leads
of the chip were formed such that they didn't really retain
too much solder. Ie: didn't cause any problems when removing
the chip from the holes.

3) Remove the chip. Remember that the PC board is the more
important part here.

    You will notice that all but two of the traces on the
component side hook up sequentially to the edge connector
pads (gold). The order of pins 10 and 16 (remember, the
component side of the edge connector has all even #s!) seem
to be reversed.

The reason for this is because GCE (the Vectrex manufacturer)
initially designed the PC board to accept a chip with an
Intel 2732 pinout. Later, they swapped 10 and 16 (to make

them out of order) to acomodate a different pinout for Sharp and/or GI mask-programmable ROMS.

4) So, in order to make it cozy for a 2732 again, one must re-reverse the wiring! The way it should look, when you finish, is -

Edge pad #	24pin EPROM pin #
10	21
16	18

If you are clever, you can cut the trace, peel it back with an exacto knife and, where the trace goes past the correct pin, scrape the coating off the copper conductor and solder it to the pad where the EPROM socket will be soldered in.

The best socket to put on it is either a ZIF (Zero Insertion Force) socket (by Textool) or a barrel-pin socket (from Augat). Remember, a cheap chip socket will be a real pain to replace later!

5) Solder the socket onto the pad where the ROM used to sit. Be careful to orient it properly (socket pin 1 to pad 1) and not to have solder bridges between pins. You may have to bend the socket pins to get them to sit well on the printed circuit pads.

Now, if you have no EPROM to test it with, there is a quick procedure that will not only test your board, but also make your old game ROM usable!

Get an extra Augat socket and mount the ROM in it after lifting up pins 21 and 18 of the ROM chip. Next, (tricky) wire pin 21 of the rom to pin 18 of the socket and visa-versa.
Now your old ROM has the same pinout as a 2732! Yeow!

Plug your Augat/ROM into the socket on the board and plug the whole assembly into the Vectrex.

Be sure to plug it in COMPONENT SIDE UP!!!

Pin one of the ROM/socket should be facing you (when looking at the front of the Vectrex.

Now you are ready to pop in EPROMs from new and exotic lands.

VECTREX\TEXT\GUTS.TXT

From umn-cs!stolaf!ihnp4!harpo!seismo!hao!hplabs!sri-unix!MKB@CMU-CS-C.ARPA
 Sun Feb 6 00:28:16 206
Relay-Version: version B 2.10.1 6/24/83; site umn-ucc.UUCP
Path: umn-ucc!umn-cs!stolaf!ihnp4!harpo!seismo!hao!hplabs!sri-unix!MKB@CMU-CS-C.ARPA
From: MKB@CMU-CS-C.ARPA
Newsgroups: net.micro
Subject: Vectrex guts
Message-ID: <17522@sri-arpa.UUCP>
Date: Tue, 13-Mar-84 17:48:00 CST
Date-Received: Sat, 17-Mar-84 15:48:10 CST
Lines: 122

From: Mike Blackwell <MKB@CMU-CS-C.ARPA>

>From the Vectrex Technical Description (my comments in {curly braces}):

The HP3000 {Vectrex} is a microprocessor based, vector scan system using a standard 9" black & white CRT as its video display device. The microprocessor (MPU) is the Motorola 68A09 device {hardware multiply and divide!}. The MPU operates at 1.6 MHz from a 6 MHz extrenal Xtal. An internal divide by 4 circuit generates the MPU 1.6 MHz "E" clock signal used in the system. Program memory is stored in the 8K x 8 2363 {really 2764} type ROM. This ROM contains common subroutines, the "executive" or assembler instructions plus one complete game {Mine Storm}.

Two 1K x 4 2114 type static RAMS provide storage locations for data indicative of locations of objects, game status, and various other information needed by the microprocessor during game operation. Peripheral Interface Adaptor (PIA) Chip {a 6522A}, has two 8 bit peripheral ports which interfaces the MPU with peripheral devices and external signals. One of the PIA ports interfaces the General Instruments AY-3-8912 sound-IO chip with the MPU and also drives the digital to analog converter chip MC1408. The other PIA port is used as control lines for the sound chip, selector control for the multiplex chip and as means to read the A/D comparator that's used in the joystick successive approximation circuitry. Sound is either MPU generated directly or by use of the AY-3-8912 sound chip.

The AY-3-8912 sound chip is a programmable sound generator containing 3 tone generators and wave shapping circuitry. This chip also has a single 8 bit IO port used to read the status of each of the handcontrollers 4 action switches.

The standard TTL device types 74LS00 and 74LS32 are usued as control line decoders to allow the MPU to select the appropriate circuit element to be addressed at any particular time.

The analog processing section includes digital to analog converter (DAC) chip type MC1408, dual 4 channel multiplexer/demultiplexer chip type CD4052, and dual op-amps types LF353 and LF347.

DAC chip M1408 receives an 8 bit word at data terminals D0-D7. DAC output (pin 4) is current source. One section of IC LF353 is used to change this current to a voltage representative of the 8 bit digital word received by the DAC chip. The LF353 voltage is applied to an input of the dual 4 channel multiplexer (MUX) chip CD4052. This same voltage (designated "DAC" on the schematic) is the X-axis drive signal.

The CD4052 MUX chip serves two purposes: it selectively couples, under MPU control, the output of the DAC current/voltage converter to one of 4 places and is used to selectively couple the inputs from the joystick pots to the voltage comparator IC LF353.

The 4 places to which the "DAC" signal is coupled by the MUX are:

1) The Y-axis sample and hold IC LF347
2) The "0" reference charge capacitor
3) The Z-axis (brightness signal) sample and hold IC LF347
4) MPU sound resistive netowrk

Each of these 4 signals is a voltage value representative of the 8 bit DAC input word for that function.

The joystick pot positions are sensed by a successive approximation process. The MUX chip selects each joystick pot input line and applies it to the plus input of comparator IC LF353. At the same time the MPU generates digital words that are changed to voltages by the DAC and current/voltage converter mentioned previously. These voltages are successfully {successively} applied to the comparator's minus input until the MPU generated voltage is equal to the joystick voltage. The MPU then recognizes the digital word representative of the comparison voltage and is able to establish a location for the joystick pot. The present position for each joystick is sensed in this manner. The pot position information is updated on a regular basis by the MPU.

Returning to the X and Y axis drive signals, these signals are applied to X,Y integrator IC LF347 negative input pins through series analog switch types 4066B. The "zero" reference signal is applied to the positive inputs of the integrators. There are also analog switches across integrator IC capacitors. The series analog switches are controlled by MPU signal RAMP and the parallel capacitor switches are controlled by MPU signal Zero 10. RAMP 10 determines when and for how long the X and Y axis voltage levels will be applied to the integrator amps. Zero 10 is used to discharge the X & Y axis integrator caps thus initializing them for the next signal to be integrated.

The outputs of the X,Y axis integrators are coupled through J-FET switches to IC LM379 deflection amplifiers. The LM379 operates as a voltage to current driver, the current through the deflection coils forming the electromagnetic field which deflects the CRT beam. To protect the CRT from spot burn in the event of a loss of deflection, the Y axis drive amplifiers output is detected and a deflection enable/disable signal is generated. This signal controls the J-FET switches in series with the X,Y deflection amp inputs to reduce the scan drive signal in the event of a software or hardware failure plus discrete transistor type 2SC1921 operates to bias off the CRT.

Conventional full wave rectification and three terminal regulators are used in the low voltage power supply. A special negative DC source is generated by a voltage double-circuit which is used to supply a 13V to the DAV chip.

The high voltage is generated via an oscillator, drive transistor and flyback type transformer circuitry similar to what is commonly used in small black and white TV receivecrs.

Judicious use of bypass caps, RF filter chokes, ferrite beads, etc., has been used in the design to control RFI emmisions.

{Me again}
The cartridge connector has the following lines from the MPU: A0-A15, D0-D7, R/~W, ~HALT, ~NMI, ~IRQ, ~E, a cartridge chip select, and one bit from PIA port B (PB6).

I've dumped the contents of the Exec ROM in to a file... One of these days somebody will get around to disassembling it... (any volunteers?)

If you'd like copies of the schematics and block diagrams, drop me your snail address.

There's a bunch of us here at CMU interested in hacking on our Vectrexes (Vectrex-People@CMUC). If we come up with anything good, I'll let the net know. If anybody out there has come up with anything, or wants to get involved, let me know... We could have a world wide effort!

cheers, -m- (mkb@cmu-cs-c)

VECTREX\TEXT\INDIC.TXT

Read Message:
R)eply K)ill E)nter N)ext P)revious
-)Read-Orig. +)Read-Reply Q)uit
[49] 1 - 53 R K E N P - + Q or ? for help:
#50 0 13 Dec 85 14:19:35 (PRIVATE)
From: Sysop
To: Jeff Woolsey
Subj: vectrex

Jeff -- just thought I'd let you know that I have about 200 DSDD 8" floppies full of old Vectrex stuff -- Source, hex, released, hacked, and unreleased -- including the original source to the executive ROM, and a few other goodies. I have two light pens, and a 3-d imager. Several prototype pieces of equipment. I also have a couple of original development systems (S-100), one with the full 64K address space as RAM. Since I wrote VID (Vectrex Interactive Debugger), I also have all of the source to it.

There is one catch... I had my S-100 system stolen about a year ago. I have no 8" drives, and therefore, all of this magnetic gold-pile of archaic information is slowly going the way of all magnetic flux -- bye-bye... Any ideas? I need to copy ALL of my 8" disks to other 8" disks, and then they will be ok for another 4 or 5 years. Just don't have the time, inclination, or capital to have it done, or buy another S-100 / CP/M system to do it on. Let me know your thoughts, and I'd love to talk Vectrex again someday.

Regards -- Mark Indictor, Sysop: Fiddlers_Fido 104/606 & Vectrex Programmer.

More ?

Kill this message now, right? (y,n) y
Message 50 deleted.

Read Message:
R)eply K)ill E)nter N)ext P)revious
-)Read-Orig. +)Read-Reply Q)uit
[50] 1 - 53 R K E N P - + Q or ? for help:

VECTREX\TEXT\INDICTOR.TXT

#2 100 14 Sep 85 00:11:47 (RECV'D)
From: Zhahai Stewart
To: Sysop
SEE ALSO #3
Subj: old days

Really, the vectrex? That was the system with a gen-u-ine vector display (ie: stroke graphics) right? I was interested in that at one time, soory to see it leave.
The 6809 was a neat processor; unfortunately it was late in the 8 bit game - if Motorola had introduced it a couple of years earlier, it might have had a more important impact on both the market and the micro architecture of today.
By the way, the punster in me has always thought that Motorola missed one of the best opportunities in naming within the micro world. There are 5 instructions which load an effecive address to a 16 bit register in the 6809 - all but 1 of which have the mnemonic LEA? where ? is th register name. The other one loads an effective address into the program counter (jmp) and would be logically called LEAP (without any stretching). Such logically fitting puns are hard to find, too bad they didn't accept LEAP as an alias for JMP/ . Oh well.
How did you get into horse ranching? How involved are you in music now-a-days, by the way?

Read Message:
N)ext P)revious -)Read-Orig. +)Read-Reply
Q)uit
[2] 1 - 48 N P - + Q or ? for help: N
#3 88 14 Sep 85 08:32:10 (RECV'D)
From: Sysop
To: Zhahai Stewart
REPLY TO #2
Subj: musical horses

I am pleased that you have heard of Vectrex. I have reason to believe that there is an underground Vectrex User's Group floating around in the higher parts of the academic sector. They are using the thing as a vector terminal to larger machines. They have completely reverse engineered it, and disassembled its ROM code. They have discovered several new games (one of which I wrote), that never were released. It is facinating how a machine that has so much potential can be scrapped for bad marketing in a badly timed market.

As for horses, I just love them. It has been a dream for me for many years to own horses, and live somewhere where there is enough room to keep them. I have this problem, though. I can't seem to justify doing anything (to myself only) if it is not a "business venture". I don't like this problem because it makes it very difficult for me to "play". At any rate, my wife and I decided that to breed horses was the PERFECT excuse for having them. So here we are...

As for music, "...I don't get around much any more...", but I would very much like to. I have met a few people over Mountain Bell's lines, but time prohibits me from being very sociable. I have my hands full with two computer clients, a horse ranch, and a house that is a "Fixerupper", and must be ready for winter in a month or less.

Are you a musician? Do you ride horses? If your knees bent the other way, what would chairs look like?

Good typin' at you. Until later then.... /mark/

P.S. I never thought of "LEAP"... That's very nice. I always liked the mnemonic "SEX" though. Bye

VECTREX\TEXT\INTERFA2.TXT

T85/08/13. 19.17.59. VECTREX. MYBASMT. Jeff Woolsey.

Blue-sky Vectrex Interface design notes.

The S100 Vectrex interface continues to have noise problems with its cable. The problems have gotten worse since I reworked the memory array. Unless I can correct the problems soon, I may be tempted to begin work on the second version of the Vectrex interface -- a more general purpose, but perhaps less versatile one.

I envision a board that plugs into the game port. Initially it would have had its own CPU, a serial port, and some dual-ported memory. I revised the design somewhat, considering that the Vectrex already had its own intelligence. What I have, then, is a board with a ROM, some 16K (or more) of RAM, a serial port, and some logic. The ROM contains an Interface Monitor which allows a terminal (or whatever) attached to the serial port to do normal monitor-type things, like read and write memory. It will also have a mode where it can load S-format files, and jump to them, and so on. Because part of the purpose here is to retain the capabilites of the S100 interface, the Mark II should behave in rather a different way.

When the board is plugged into the Vectrex, and the Vectrex is reset, it will jump to the ROM on board, which at 0 contains the Interface Monitor. It looks like an ordinary game cartridge. When it starts up, it monitors the serial port for commands (or possibly one of the game ports can be treated as a serial port?) The ROM will have code in it which will get it out of the way, and allow the interface to look like a pile of RAM (with some status bits that control write protect and so on.

Perhaps the ROM operation will be to copy itself into internal RAM, or to the high end of the itnerface memory. It will then set some bit that makes the ROM disappear from the address space, so that the RAM can be written (with, possibly, a game image). The monitor can then jump to a specific address, or to the Executive to have it start the game. The code in the ROM can be made position independent, facilitating a mechanism which might just allow relocation of the ROM somewhere else in the address space.

One thing that doesn't appear feasible in this implementation is to wrest control of the Vectrex from the running game to the Interface Monitor. In the S100 interface this is done by halting the Vectrex and writing all over the interface memory. In this version the only thing we could possibly hope for is that interrupts (from the serial port, however that's configured) still work. Most games, however, either trash the interrupt vectors, disable interrupts, or do something else with the ports (like run a light pen or an imager). This advances the case for offboard intelligence.

The problem I'm trying to eliminate here is noise on the long cable between the Vectrex and the interface, and one obvious thing to do is put the interface right outside the Vectrex. We lose some capability by doing this, however, as I have outlined above. However, one would not need any real sort of support system to do a few other things.

There is also the matter of the serial port (such as it is). Ideally, it is autobaud and everything. However, the Vectrex does not provide +/- 12 volts at the cartridge connector, and it appears that the pins on the two ports are only TTL levels. What's a designer to do, particularly when one of the goals is to work with an unmodified Vectrex?

VECTREX\TEXT\INTERFAC.TXT

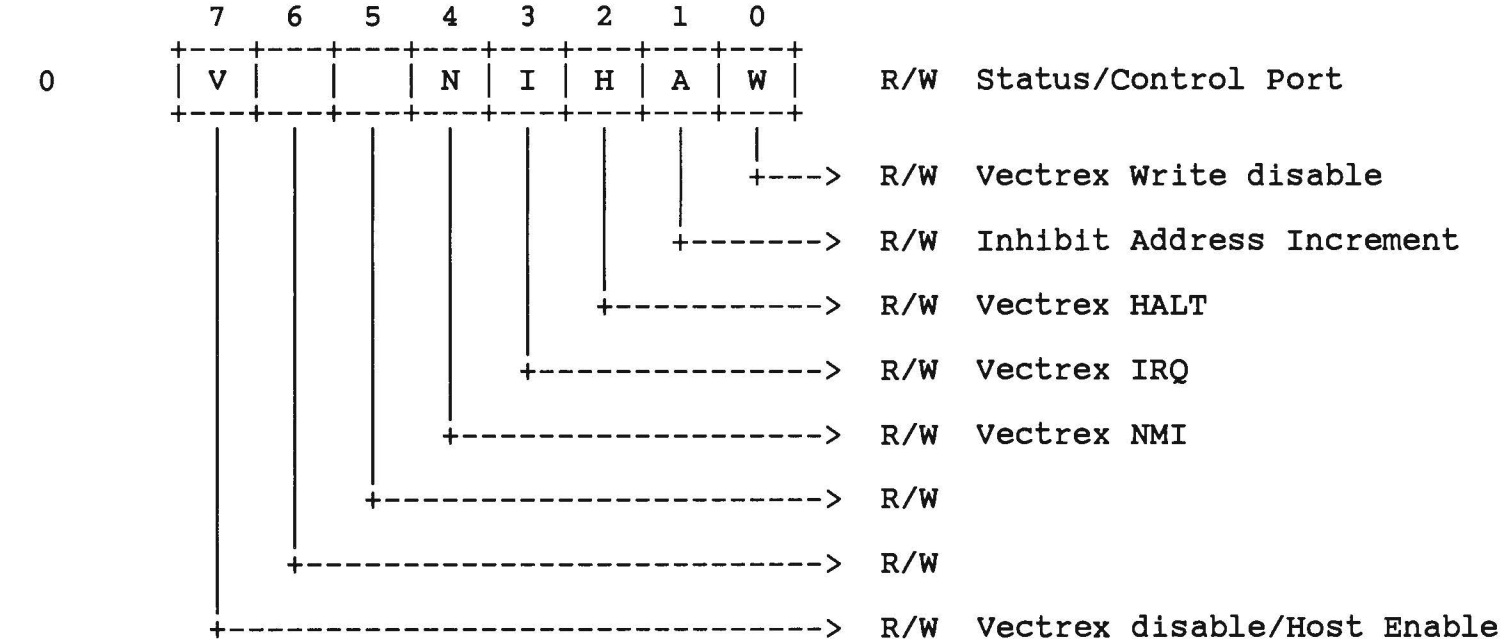
This pretty much describes the working interface as built.

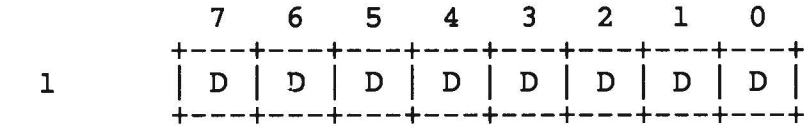
S84/06/23. 15.13.37. VECTREX. MYBASMT. Jeff Woolsey.
N85/03/31. 13.25.06. VECTREX. MYBASMT. Jeff Woolsey.

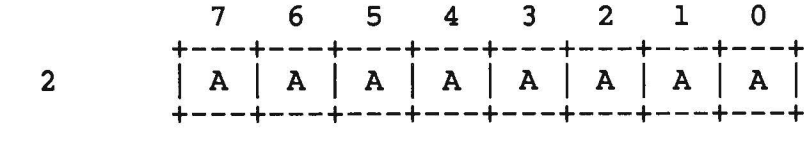
These proposals are being scaled down to obtain a functional unit from the previous prototype which was perhaps a bit ambitious. In doing so we shall remove anything that doesn't seem immediately attainable or necessary, without undoing too much work, and add only that which is needed to get the interface running usably.

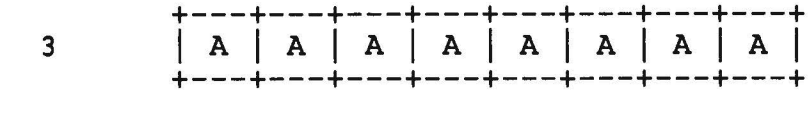
Proposed operational details for the S100-HP3000 (Vectrex) interface.

The interface will be accessed through 4 ports, defined as follows, relative to the base port.









The interface consists primarily of a chunk of memory which is accessible by both the Vectrex, and the S100 Host. It resides at address 0 on up the Vectrex address space (same as the ROM cartridges). In the S100 host, an onboard

address counter is used to give the same address range, although access is indirect through the use of I/O ports, described below.

Setting and reading bits in the Status/Control port is straightforward. The values of bits were chosen such that writing a 0 to the port restores normal Vectrex operation. Individual bit descriptions follow.

W - Vectrex Write Disable. Setting this bit makes the interface look like ROM to the Vectrex. Clearing it allows both read and write access to the memory. Host operation is not affected.

A - Inhibit Address Increment. Setting this bit prevents the address counters from being incremented when the data port is accessed. Handshaking is the intended use.

H - Vectrex HALT. Setting this bit halts the Vectrex in its tracks. This is useful if memory is not dual-ported, in that the Vectrex will not access memory when it is halted, thus the host can run amok without worrying about timing windows. Clearing this bit releases the Vectrex from the host's grip.

I - Vectrex IRQ. Setting this bit asserts IRQ at the Vectrex cartridge socket. Interrupt acknowledge is not directly available at the cartridge.

N - Vectrex NMI. Same as above, but asserts NMI.

V - Vectrex disable/Host enable. Delegates responsibility for memory access to the host. Clearing the bit enables Vectrex access to the interface memory and disables host access. Setting the bit reverses the situation, which generally means that the Vectrex will jump off into Never-never Land, so care must be taken. This area has not been fully researched yet.

Writing a byte to the data port causes the following:

1. The data are latched.
2. The address in the counters is presented to the memory.
3. The data are presented to memory.
4. WR is applied to memory.
5. If not inhibited, the address counter is incremented.

Reading the data port causes the following.

1. The address in the counters is presented to the memory.
2. RD is applied to memory.
3. The data are latched and presented to the host bus.
4. If not inhibited, the address counter is incremented.

Reading and writing data is arbitrated by the host setting or clearing the V bit in the status byte (see above). We repeat the concern that Vectrex control can be lost if care is not taken.

Writing to the address ports causes the data byte to be loaded into the corresponding address counter. Reading the port causes the value in the counter to be returned. Only 32K of memory is addressible externally to the Vectrex, but all 16 address lines are available.

One glaring difficulty in using this interface becomes apparent when one notices the lack of a Vectrex RESET capability. The line simply is not

present at the cartridge connector. This is not an insurmountable problem, however, partly because there IS a reset button on the Vectrex. It is inconvenient, though, to have to hit the button repeatedly.

The desired effect is to be able to have the Vectrex execute a loaded program from having been powered up, without having to touch it. This can be accomplished, although recovering from runaway programs is more difficult.

When the Vectrex is first powered up, it displays its title page and plays its music for about 5 seconds. This is ample time to load something into the interface, provided the host can do this within 5 seconds from power up as well, assuming that the host and the Vectrex are on the same mains. If not, be patient. The Vectrex then looks for a cartridge in the socket. This procedure is described in detail elsewhere. If it finds no cartridge, it runs the MINE STORM game. Unattended, MINE STORM will end in about 90 seconds, at which point it will wait a while for any joystick button to be pressed to run the game over. If this times out, it will jump back to the monitor, which displays the Vectrex title page and music again, and then looks once more for a cartridge (If you manage to plug in a cartridge while the machine is running, against all advice, it will work. Positive reinforcement.). If by this time you have managed to load a program into the interface, it will be executed. Less patient programmers will want to wrest control immediately. In looking over the executive ROM for clues, one finds that NMI and SWI both vector to $CBFB, and IRQ goes to $CBF8. One could try tugging on these control lines via the Status/Control Port.

Recovering from a runaway program is unexplored territory. Progress in this area depends upon having a working interface to experiment with.

Now that we have a working interface, here's an untried method of keeping the Vec's attention once we have it. Assume for the moment that we have some program running in the interface. The procedure should work as follows. First, HALT the Vectrex using the HALT* line on the interface. Wait long enough for whatever instruction was being executed to finish. This could involve as many and six additional memory references. Once the Vectrex is halted, fill the interface memory with three specific things: 1) Fill the middle of the interface address space with NOPs. 2) Put a JMP opcode in the last byte of the interface address space. The address that the JMP reads follows immediately, and will read FF since there is no memory there. At $FFFF there is a 00 byte. This is the opcode of a two-byte NEG instruction, the object of which is obtained by wrapping around to $0000. 3) Put a small bootstrap loader loader program at $0000. The first byte of which is the object of the NEG instruction. The loader loader must move a small bootstrap program into the internal RAM and jump to it. This small bootstrap program will montior $0000 for $67, at which point it jumps to $F000. While monitoring, it might put out some message on the screen saying that it is ready for a load. It might also want to set $0000 to something to indicate to the host that it has taken over and that the interface is free for the host to write all over. At this point, when the interface is done loading something, the little bootstrap program will notice and execute it. This will not work the first time the interface is loaded, as it will be executing MINE STORM, and a HALT will simply halt it, and when MINE STORM finishes, the bootstrap loader loader will not look like a cartridge header unless we're awful damn clever: The Vec jumps to $FFFF which does a NEG $67 (at 0) in direct mode, which negates byte $67 in whatever page DP points at. Then it hits a space ($20) which is a BRA, followed by the G in GCE. That's $47, so it branches forward $47 bytes, to $004A So, if we can put up a little message and music that fits, things might actually work....

VECTREX\TEXT\PLAY.TXT

Vectrex Music, as accepted by the PLAY.PAS program, is as follows.

PLAY Statement, additional information as specified for the Vectrex...
Purpose: Plays music as specified by *string*.

Format: PLAY *string*

Remarks:
PLAY implements a concept similar to DRAW by imbedding a "tune definition language" into a character string.
string is a string expression consisting of single- or double-character music commands.
The commands in PLAY are:

A to G with optional #, +, or -
: Plays the indicated notes in the current octave. A number sign (#) or plus sign (+) afterward indicates a sharp; a minus sign (-) indicates a flat. A #, +, or - is not allowed unless it corresponds to a black key on a piano. For example, B# is an invalid note. The Vectrex is not so picky.

H*string*
: Declare *string* to be the title of the piece. On the Vectrex, it will be centered in the middle of the screen. The string terminates at end-of-line. If this command is omitted (or late) the tune file name becomes the title. This command must occur before any notes or rests if it is to be displayed.

K*string*
: Declare *string* to be the author (Komposer?) of the piece. This string is centered above the title. If omitted or late, some suitable thing will be displayed.

O n
: Octave. Sets the current octave for the notes that follow. There are 7 octaves, numebred 0 to 6. Each octave goes from C to B. Octave 3 starts with middle C. Octave 4 is the default octave.

> n
: Go up to the next higher octave and play note n. Each time note n is played, the octave goes up until it reaches octave 6. For example, PLAY ">A" raises the octave and plays note A. Each time PLAY ">A" is executed, the octave goes up until it reaches octave 6; then each time PLAY ">A" executes, note A plays at octave 6. (For BASIC 2.0 and later releases.)

< n
: Go down one octave and play note n. [...until octave 0 per > -jxh]

N n
: Plays note n, which can range from 0 to 84. In the 7 possible octaves, there are 84 notes. n=0 means "rest." This is an alternative way of selecting notes besides specifying the octave (O n) and the note name (A-G). (The Vectrex range is narrower (0-62; an attempt was made to center it the same as PLAY), and a rest is 63.)

L n

Sets the length of the notes that follow. The actual length of the note is 1/n. n can range from 1 to 64. In the Vectrex implementation, the default is 4.

Length	Equivalent
L1	whole note
L2	half note
L3	one of a triplet of three half notes (1/3 of a 4-beat measure)
L4	quarter note
L5	one of a quintuplet (1/5 of a measure)
L6	one of a quarter-note triplet
.	
.	
.	
L64	sixty-fourth note

The length can also follow the note when you want to change only the length of the note. For example, A16 is equivalent to L16A.

P n

Pause (rest). n can range from 1 to 64, and figures the length of the pause in the same way as L (length).

.

(dot or period) When placed after a note, causes the note to be played as a dotted note. A dot increases the duration of a note by half the duration of the note. A note can have more than one dot. Each dot increases the total value by 1/2 the value of the previous dot. For example, a double-dotted halfnote is equivalent in duration to a half note plus a quarter note plus an eighth note. Dots can also appear after a pause (P) to scale the pause length in the same way.

T n

Tempo. Sets the number of quarter notes in a minute. n can range from 32 to 255. The default is 120. Under "SOUND Statement" is a table listing common tempos and the equivalent beats per minute.

MF

Music Foreground. Music (created by SOUND or PLAY) runs in foreground. Each subsequent note or sound will not start until the previous note or sound is finished. You can press Ctrl-Break to exit PLAY. Music foreground is the default state. Not relevant on the Vectrex.

MB

Music Background. Music (created by SOUND or PLAY) runs in background instead of foreground. Each note or sound is placed in a buffer, allowing the BASIC program to continue executing while music plays in the background. The music background buffer can hold up to 32 notes at one time. Not relevant on the Vectrex.

MN

Music Normal. Each note plays 7/8 of the time specified by L (length). This is the default setting of MN, ML, and MS.

MD

Old dot mode. One piece I received was written for a dot mode where d := d + d / 2, whereas the default mode is d := d + dot / 2. Automatic translation of the piece from the old mode into the new

is proving difficult.

ML
Music Legato. Each note plays the full period set by L (length).

MS
Music Staccato. Each note plays 3/4 of the time specified by L.

X variable;
Executes the specified string. The Vectrex implemenation gets angry if you try to use this.

R*string*
Remarks. Arbitrary text for commenting (or whatever). Ends at next newline.

W*string*
Intended to represent the lyrics (if any) for the tune being played. Not presently implemented on the Vectrex; behaves like R.

Y*date*
date is placed in the copyright message while the tune is playing. *date* generally should be the year the tune was composed, but it may be any 4 characters.

Z
Enter special polyphonic mode where all of the notes on one line are played for the duration of the shortest. This exists to support true polyphonic sound for the Vectrex.

In all these commands, the n argument can be a constant such as 12, or it can be *=variable;* where *variable* is the name of a variable. The semicolon (;) is required when you use a variable in this way, and when you use the X command. Otherwise a semicolon is not allowed after MF, MB, MN, ML, or MS. Also, any blanks in *string* are ignored. The Vectrex does not accept the =*variable*; syntax.

You can specify variables in the form VARPTR$(variable), instead of =variable;. The VARPTR$ form is the only one that can be used in compiled programs. For example:

```
One method       Alternative method
PLAY "XA$;"      PLAY "X"+VARPTR$(A$)
PLAY "O=I;"      PLAY "O="+VARPTR$(I)
```

You can use X to store a "subtune" in one string and call it repetitively with different tempos or octaves from another string.

Examples: The following example plays a tune.

```
10  'little lamb
20  MARY$="GFE-FGGG"
30  PLAY "MB T100 O3 L8;XMARY$;P8 FFF4"
40  PLAY "GB-B-4; XMARY$; GFFGFE-...."
```

The following example plays the scale from octave 0 to octave 6.

```
10  ' Play the scale using > octave
20  SCALE$="CDEFGAB"
30  PLAY "O0 XSCALE$;"
```

```
40  FOR I=1 TO 6
50  PLAY ">XSCALE;"
60  NEXT
70  ' Play the scale using < octave
80  PLAY "O6 XSCALE;"
90  FOR I=1 TO 6
100 PLAY "<XSCALE$;"
110 NEXT
```