

OHIO SCIENTIFIC, INC.

TECH NEWSLETTER #25

December 28, 1979

Copyright 1979
OHIO SCIENTIFIC, INC.
All Rights Reserved.

OS-65U NMHZ AND THE NEC SPINWRITER

The following POKES modify the NEC driver contained in the program "NECDRV". The modification allows the "PRINTER STALLED" timeout routine to work at NMHZ.

To insert the changes:

1) TYPE: LOAD"NECDRV" <CR>

2) Enter the following lines:

POKE 25012,162: POKE 25013,217: POKE 25014,165 <CR>

POKE 25015,224: POKE 25016,45: POKE 25017,0 <CR>

POKE 25018,245: POKE 25019,73: POKE 25020,16 <CR>

POKE 25021,240: POKE 25022,15: POKE 25023,41 <CR>

POKE 25024,16 : POKE 25025,208: POKE 25026,9 <CR>

POKE 25027,169: POKE 25028,6: POKE 25029,32 <CR>

POKE 25030,184: POKE 25031,62: POKE 25032,202 <CR>

POKE 25033,208: POKE 25034,235: POKE 25035,234 <CR>

POKE 25036,169: POKE 25037,128: POKE 25038,96 <CR>

POKE 25396,162: POKE 25397,217: POKE 25398,165 <CR>

POKE 25399,224: POKE 25400,45: POKE 25401,0 <CR>

POKE 25402,245: POKE 25403,73: POKE 25404,16 <CR>

POKE 25405,240: POKE 25406,15: POKE 25407,41 <CR>

POKE 25408,16: POKE 25409,208: POKE 25410,9<CR>
POKE 25411,169: POKE 25412,6: POKE 25413,32<CR>
POKE 25414,184: POKE 25415,62: POKE 25416,202<CR>
POKE 25417,208: POKE 25418,235: POKE 25419,234<CR>
POKE 25420,169: POKE 25421,128: POKE 25422,96<CR>

3) Now type:

SAVE "NECDRV", "PASS"<CR>

4) The changes have been saved to disc.

NOTE: The program must be RUN to make the changes effective.

Applicability:

ALL NMHZ VERSIONS OF OS-65U (V1.2)



TROUBLESHOOTING MEMORY BOARDS TO CHIP LEVEL

If an error occurs when using the Memory Test Program (MTP), you may be able to save yourself the time and expense of sending back an entire board for repair. If you are not already familiar with troubleshooting to memory chip level, please take the time to read the following article....

520 (CM-3)

Refer to Diagram 1 for the physical layout of the 4200 chip array. While looking at the board's component side with the molex connectors to the right, the top row of 4200's contain the first hexadecimal "K" of RAM (\$0000 through \$0FFF). Note the second row contains the second "K" and so on. Since each 4200 memory chip is 4K (Decimal) by 1 bit, there are 8 chips in this row (one for each data bit). The leftmost chip in each row contains the least significant bit of a byte, while the right most chip contains the most significant.

Hence, if the MTP delivers the message, "location \$23F7 is 08 should be 00" it has been determined that the most significant bit of the least significant Hex digit at location \$2F37 has failed (see Diagram #2). This would be chip #20 in Diagram #1. If this error message was delivered, "Location \$15D5 is BF should be FF", then the "4 Bit" of the most significant Hex digit at location \$15D5 is defective (See Diagram #3). This would be chip #15 of Diagram #1.

Keep in mind that the CM-3 in the first 16K slot contains Hex addresses \$0000 through \$3FFF, the second slot contains \$4000 through \$7FFF, and the third contains \$8000 through \$BFFF. (Refer to Tech Newsletter #8, Pages 4-7 for information on addressing 520 CM-3 Boards.)

Unlike 4200 chips that are 4K by 1 bit, the 2114 is a 1K by 4 bit chip. Therefore, to cover 8 bits of data only 2 chips are required.

Refer to Diagram #4. Notice the example is strapped so that the top 8K bank is at 0 dec or \$0000 Hex, the middle 8K is at 8192 dec or \$2000 Hex, and the bottom 8K is at 16384 dec or \$4000 Hex. The lowest byte is stored in the chips at the upper left corner of the array and the highest byte is stored in the lower right. (Refer to Tech Newsletter #8 for 527 addressing information).

Rows A, C, and E of the diagram handle the lower 4 bits (D0 through D3) while Rows B, D, and F handle the upper 4 bits (D4 through D7). So, as an example, chips A1 and B1 of Diagram #4 carry the 8 bits of locations \$0000 through \$03FF, with D0 through D3 stored in Chip A1 and D4 through D7 in chip B1. Likewise, chips A2 and B2 are used to store addresses \$0400 to \$07FF, chips A3 and B3 have locations \$0800 to \$0BFF, etc.

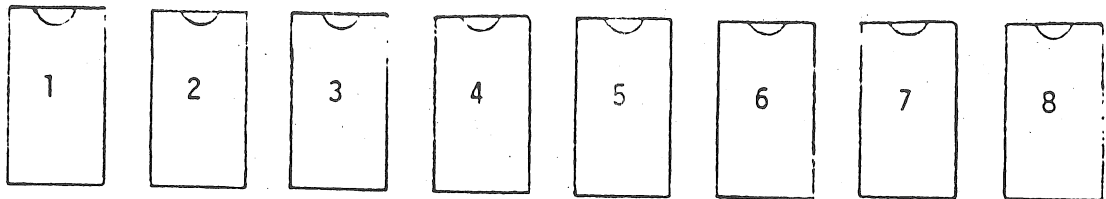
To find out what locations are handled in Rows C and D add \$2000 to the corresponding addresses at the top of Diagram #4. The same is true with Rows E and F, but add \$4000. For example, the data at locations \$3000 through \$33FF are stored in chips C5 and D5. Another example is that locations \$4800 through \$4BFF are handled by chips E3 and F3.

Now, here are a few possible MTP error reports:

"Location \$193B is F7 should FF". (This error report indicates the lower data of \$193B has changed. This would be chip A7 of Diagram #4). See Diagram #5.

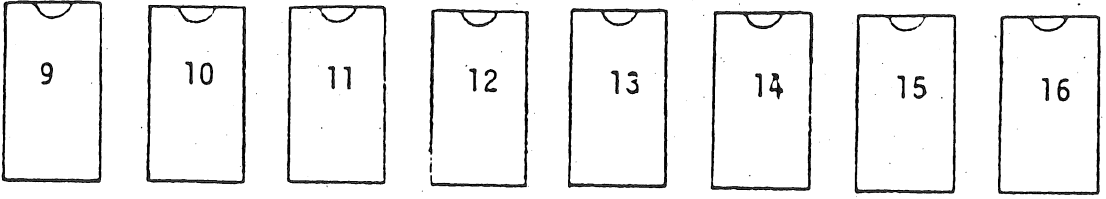
"Location \$3655 is 40 should be 00". (This error is reporting a change of data has been detected at chip D6 of Diagram #4) See Diagram #6.

\$0000
↓
\$0FFF



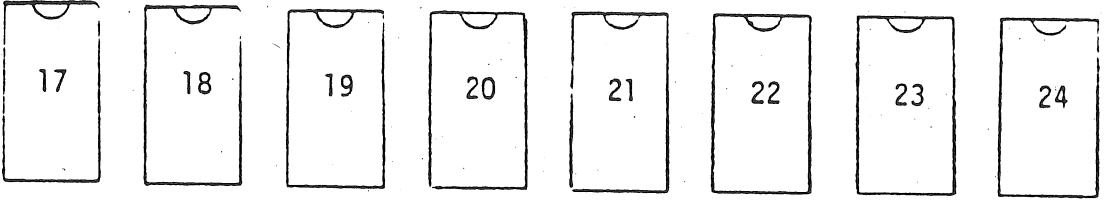
6

\$1000
↓
\$1FFF



COMPONENT
SIDE
WITH MOLEX
CONNECTORS
TO THE
RIGHT
→

\$2000
↓
\$2FFF



\$3000
↓
\$3FFF

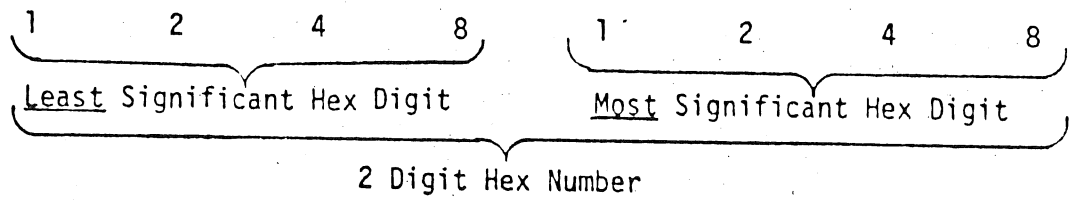
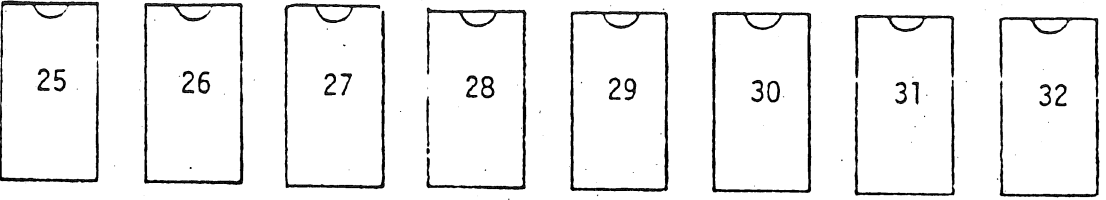


DIAGRAM #1

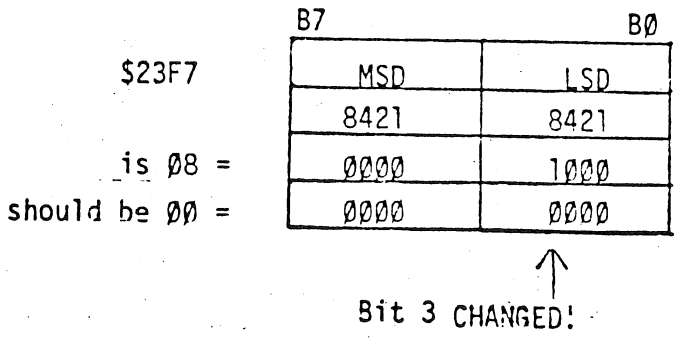


DIAGRAM 2

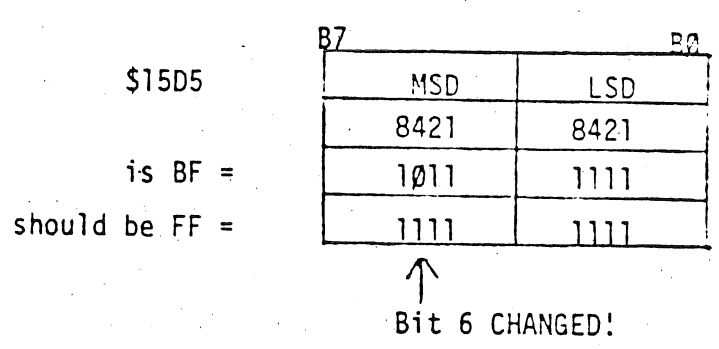
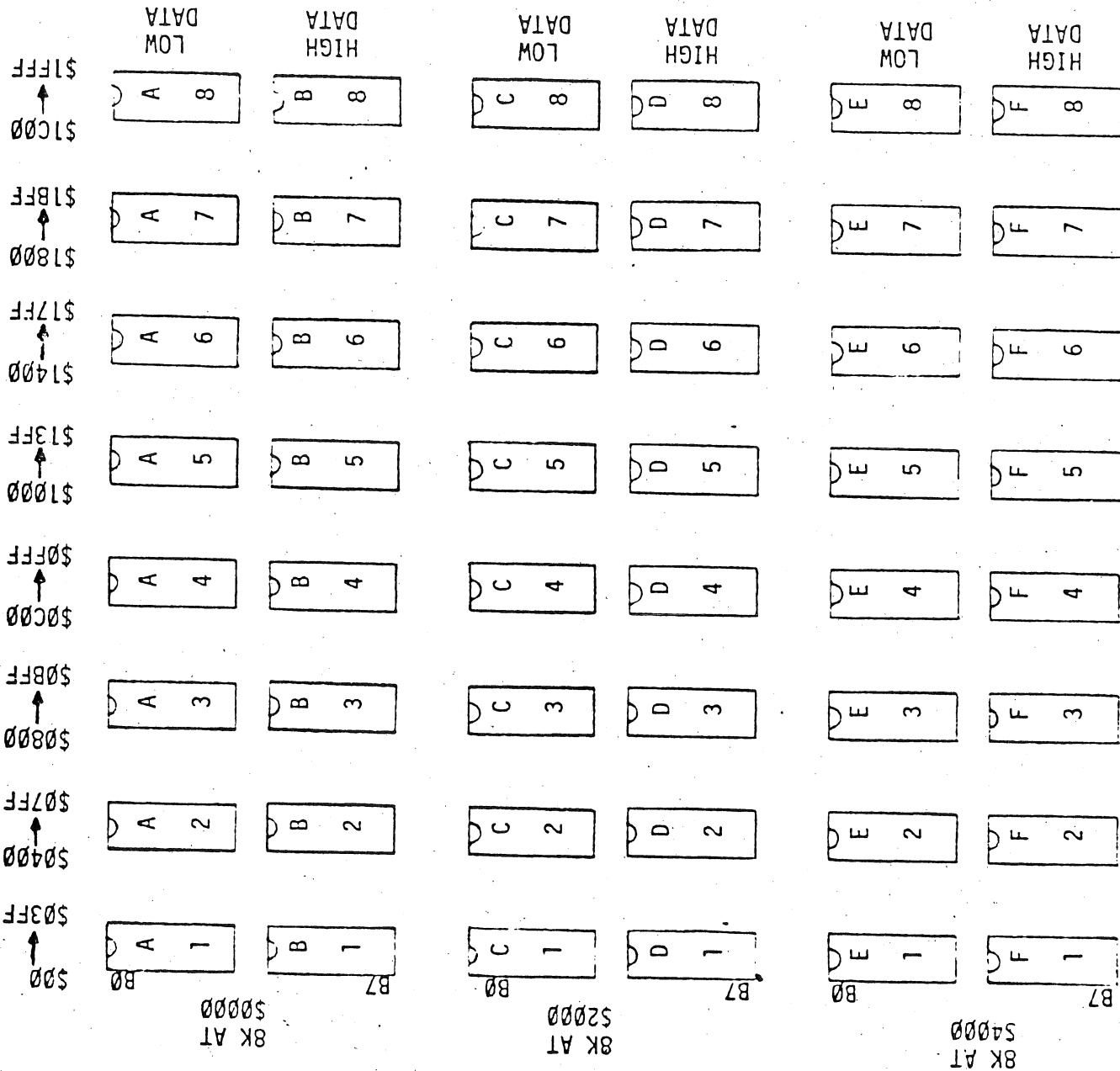


DIAGRAM 3

COMPONENT SIDE WITH
MOLEX CONNECTORS

TO RIGHT
↑



NOTE
STRAPPING

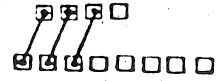


DIAGRAM 4

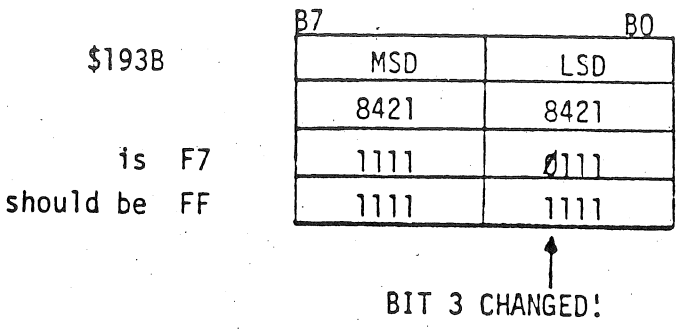


DIAGRAM 5

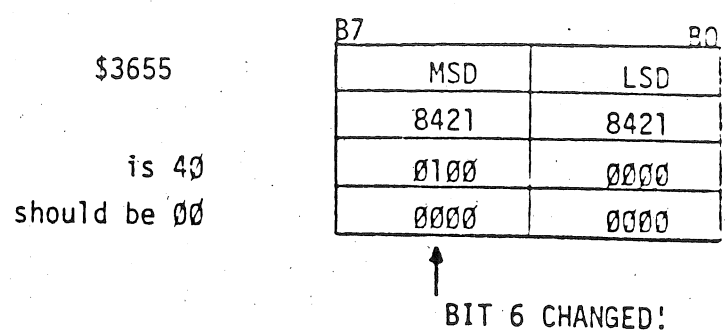
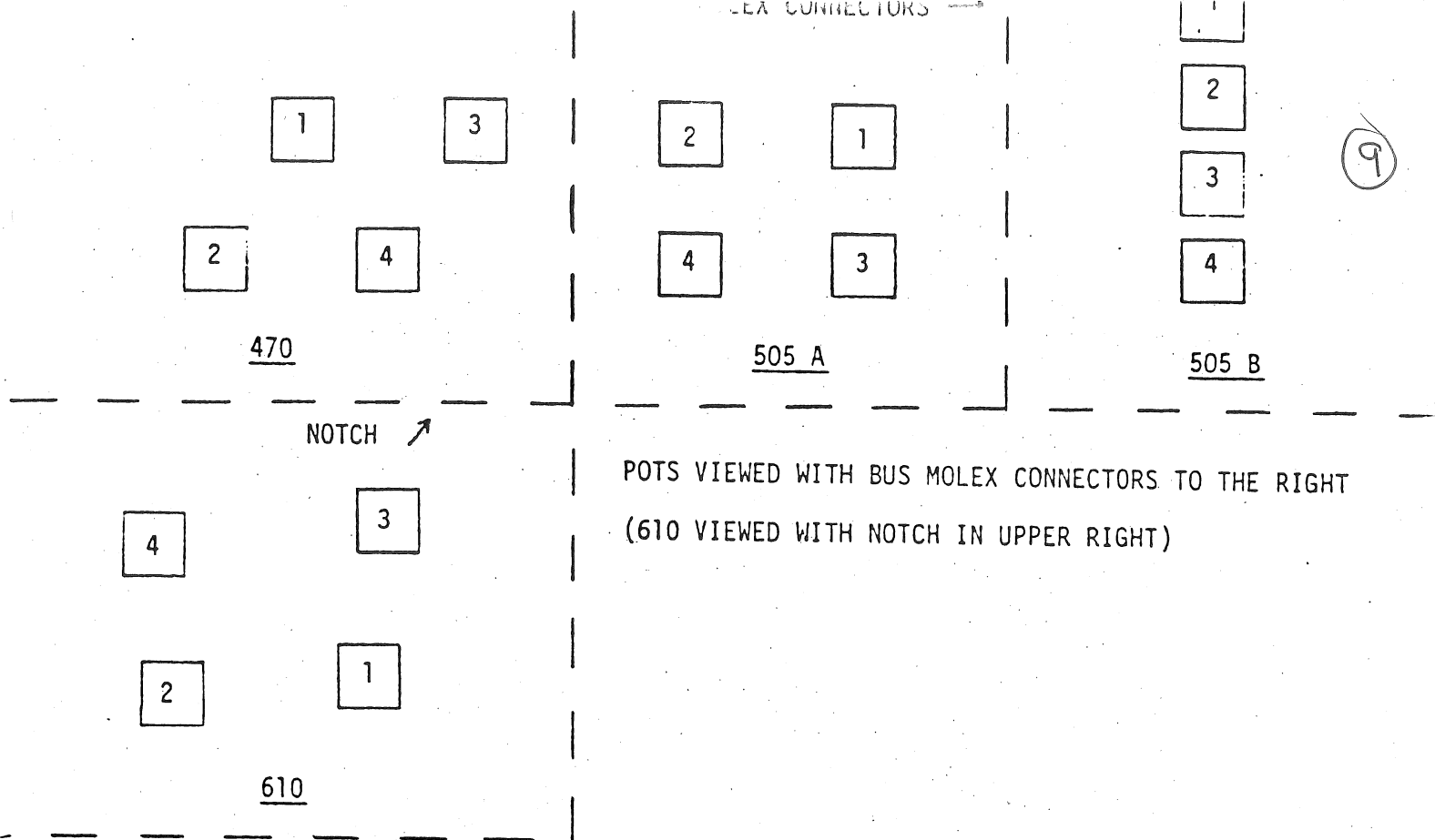


DIAGRAM 6

530 (CM-4)

The array of 4027 memory chips on the 530 board is laid out similar to the 520 chip array shown in diagram (1). That is, each row contains 4K of memory with the least significant bit to the left and the most significant bit to the right.

If you have not already done so, read the section of this article concerning the 520 (CM-3) for troubleshooting this type of array. (Refer to tech newsletter #9 for 530 addressing.)



FLOPPY TIMING ADJUSTMENTS
REMOVE FLOPPY ACIA & PIA BEFORE ALL ADJUSTMENTS!
(NOTE: 505 B MUST HAVE FLOPPY CABLE AND DRIVE CONNECTED)

POT #	INSTRUCTIONS	5"	8"	NOTES
1	SCOPE PIN #9 OF BOARD'S MALE MOLEX CONNECTOR	350 NS NEGATIVE	250 NS NEGATIVE	
2	GROUND PIN #6 OF THE ACIA SOCKET, SLIGHTLY OFF-TRIGGER SCOPE AND CONTINUE SCOPING PIN #9	350 NS NEGATIVE	250 NS NEGATIVE	IMAGE IN PHASE AND SUPERIMPOSED ABOVE SETTING
3	TIE MALE MOLEX CONNECTORS #9 AND #10 TOGETHER, AND SCOPE PIN #3 OF ACIA SOCKET	1 μ S POSITIVE	1 μ S POSITIVE	
4	TIE MALE MOLEX CONNECTORS #9 AND #11 TOGETHER, AND SCOPE PIN #2 OF ACIA SOCKET	6 μ S NEGATIVE	2.5 μ S NEGATIVE	5" ONE SHOT RESISTOR = 10 8" ONE SHOT RESISTOR = 4.7

Many of you have asked for the differences of a 590 board for the CD-23 vs a 590 for the CD-74. The following describes the modifications to the 590 for a CD-23.

(10)

N.C. - NON-COMPONENT SIDE
C.S. - COMPONENT SIDE

<u>ACTIVITY</u>	<u>LOCATION</u>	<u>VALUE</u>
CUT FOIL	N.C. IC-B6, PIN 3 TO IC-B4, PIN 8	-----
ADD JUMPER	N.C. IC-B6, PIN 3 TO IC-B3, PIN 6	-----
CUT FOIL	C.S. IC-E7, PIN 13 TO IC-C8, PIN 1	-----
ADD DIODE	C.S. CATHODE TO IC-E7, PIN 13/ANODE TO IC-C8, PIN 1	1N914
ADD DIODE	N.C. CATHODE TO IC-C8, PIN 13/ANODE TO IC-C8, PIN 1	1N914
ADD RESISTOR	N.C. IC-C8, PIN 1 TO +5V	4.7K OHM

The following pages list numerous memory locations and their use in OS-65U.

Ohio Scientific extends its appreciation to T. L. Wallis and The Computer Shop/
Texas for their contribution of this list.

MEMORY MAP OF OS65U AND LOCATION OF VARIOUS PARAMETERS

DEVELOPED 12-10-79

REVISION #0 - 12-10-79

BY T.L. WALLIS

THE COMPUTER SHOP

6812 San Pedro

FROM OSI LITERATURE, NEWSLETTERS, ETC.

San Antonio, Texas 78216

828-0553

LOCATION		INITIAL CONTENTS		NAME OR DESCRIPTION
DEC	HEX	DEC	HEX	
23	0017			TERMINAL WIDTH
24	0018			INT(TERM. WIDTH/14)*14+1
132	0084			MEM SIZE - INT(MEM SIZE/256)*256
133	0085			INT(MEM SIZE/256)
1390	056E	64	40	BASIC LINE DELETE CODE - '@'
1394	0572	95	5F	BASIC CHAR DELETE CODE - '-'
1632	0660			ORG "NEW"
1658	067A			ORG "CLEAR"
1862	0746			ORG "FOR"
2056	0808			ORG "RESTORE"
2073	0819			CONTROL 'O' - ENABLE OR DISABLE LOC.
2086	0826			ORG "STOP"
2088	0828			ORG "END"
2129	0851			ORG "CONT"
2155	086B			ORG "NULL"
2183	0887			ORG "GOSUB"
2212	08A4			ORG "GOTO"
2257	08D1			ORG "RETURN"
2295	08F7			ORG "DATA"
2343	0927			ORG "IF"
2362	093A			ORG "REM"
2378	094A			ORG "ON X GOTO"
2468	09A4			ORG "LET"
2611	0A33			ORG "PRINT"
2676	0A74	13	0D	LOC OF 'CR' INPUT ECHO (POKE 0-INHIB.)
2683	0A7B	10	0A	LOC OF 'LF' INPUT ECHO (POKE 0-INHIB.)
2797	0AED			ASCII ECHO TO INPUT STATEMENT - '?'
2858	0B2A			ORG "INPUT"
2888	0B48	27		POKE 0 TO ALLOW 'CR' ON INPUT
2902	0B56			ORG "READ"
2972	0B9C	58		POKE 13 TO ALLOW ',' ON INPUT
2976	0BA0	44		POKE 13 TO ALLOW ':' ON INPUT
3145	0C49			ORG "NEXT"
3561	0DE9			ORG "COMPLIMENT NUMBER"
3720	0E88			ORG "LOGICAL OR"

3723	0E8B	ORG "LOGICAL AND"	
3768	0EB8	ORG "RELATIONAL OPERATOR"	
3374	0F23	ORG "DIM"	
4304	10D0	ORG "USR"	
4612	1204	ORG "FRE"	
4645	1225	ORG "POS"	
4659	1233	ORG "DEF"	
4841	12E9	ORG "STR\$"	
5478	1566	ORG "CHR\$"	
5498	157A	ORG "LEFT\$"	
5542	15A6	ORG "RIGHT\$"	
5553	15B1	ORG "MID\$"	
5622	15F6	ORG "LEN"	
5637	1605	ORG "ASC"	
5671	1627	ORG "VAL"	
5768	1688	ORG "PEEK"	
5777	1691	ORG "POKE"	
5786	169A	ORG "WAIT"	
5816	16B8	FLOATING POINT ADD & SUB	SIZE=461
5925	16C1	ORG "SUBTRACTION"	
5864	16D8	ORG "ADDITION"	
6277	1885	NATURAL LOG	SIZE=108
6323	18B3	ORG "LOG"	
6335	18F1	FLOAT'G POINT MULT & DIV	SIZE=565
6387	18F3	ORG "MULTIPLICATION"	
6668	1A0C	ORG "DIVISION"	
6950	1B26	SIGN, SGN, FLOAT, NEG, ABS	SIZE=43
6964	1B34	ORG "SGN"	
6995	1B53	ORG "ABS"	
6998	1B56	COMPARE	SIZE=64
7062	1B96	GREATEST INTEGER	SIZE=88
7111	1BC7	ORG "INT"	
7150	1BEE	FLOAT'G POINT INPUT	SIZE=212
7362	1CC2	FLOAT'G POINT OUTPUT	SIZE=387
7749	1E45	ORG "SQR" (SQR+EXP:SIZE=78)	
7758	1E4E	ORG "EXPONENTIATION"	
7815	1E87	ORG "COMPLIMENT SIGN"	
7827	1E93	EXPONENTIATION	SIZE=129
7873	1EC1	ORG "EXP"	
7956	1F14	POLYNOMIAL	SIZE=74
8030	1F5E	RANDOM	SIZE=68
8038	1F66	ORG "RND"	
8098	1FA2	ORG "COS" (COS, SIN, TAN:SIZE=180)	
8105	1FA9	ORG "SIN"	
8178	1FF2	ORG "TAN"	
8278	2056	ORG "ATN"	
8704	2200	DTBLE 'END'	\$0929-1=2088 D
8706	2202	DTBLE 'FOR'	\$0747-1=1362 D
8708	2204	DTBLE 'NEXT'	\$0C4A-1=3145 D
8710	2206	DTBLE 'DATA'	\$08F8-1=2295 D
8712	2208	DTBLE 'INPUT'	\$0B2B-1=2858 D
8714	220A	DTBLE 'DIM'	\$0F23-1=3874 D
8716	220C	DTBLE 'READ'	\$0B57-1=2902 D
8718	220E	DTBLE 'LET'	\$09A5-1=2469 D
8720	2210	DTBLE 'GOTO'	\$08A5-1=2212 D

V/S

8722	2212			DTBLE	'RUN'	\$2423-1=9250	D
8724	2214			DTELE	'IF'	\$0928-1=2343	D
8726	2216			DTBLE	'RESTORE'	\$0809-1=2056	D
8728	2218			DTBLE	'GOSUB'	\$0838-1=2183	D
8730	221A			DTBLE	'RETURN'	\$08D2-1=2257	D
8732	221C			DTBLE	'REM'	\$093B-1=2362	D
8734	221E			DTBLE	'STOP'	\$0827-1=2086	D
8736	2220			DTBLE	'ON X GOTO'	\$094B-1=2378	D
8738	2222			DTBLE	'NULL'	\$086C-1=2155	D
8740	2224			DTBLE	'WAIT'	\$169B-1=5786	D
8742	2226			DTBLE	'LOAD'	\$23FD-1=10492	D
8744	2228			DTBLE	'SAVE'	\$2997-1=10646	D
8746	222A			DTBLE	'DEF'	\$1234-1=4659	D
8748	222C			DTBLE	'POKE'	\$1692-1=5777	D
8750	222E			DTBLE	'PRINT'	\$0A34-1=2611	D
8752	2230			DTBLE	'CONT'	\$0852-1=2129	D
8754	2232			DTBLE	'LIST'	\$4BDA-1=19417	D
8756	2234			DTBLE	'CLEAR'	\$067B-1=1658	D
8758	2236			DTBLE	'INDEX<'	\$2C33-1=11314	D
8760	2238			DTBLE	'OPEN'	\$2AD2-1=10961	D
8762	223A			DTBLE	'CLOSE'	\$2BC7-1=11206	D
8764	223C			DTBLE	'FIND'	\$4A4F-1=19022	D
8766	223E			DTBLE	'DEV'	\$2C05-1=11268	D
8768	2240			DTBLE	'FLAG'	\$497F-1=18914	D
8770	2242			DTBLE	'NEW'	\$0661-1=1632	D
8772	2244			DTBLE	'SGN'	\$1B34 =6964	D
8774	2246			DTBLE	'INT'	\$1BC7 =7111	D
8776	2248			DTBLE	'ABS'	\$1B53 =6995	D
8778	224A			DTBLE	'USR'	\$10D0 =4304	D
8780	224C			DTBLE	'FRE'	\$1204 =4612	D
8782	224E			DTBLE	'POS'	\$1225 =4645	D
8784	2250			DTBLE	'SQR'	\$1E45 =7749	D
8786	2252			DTBLE	'RND'	\$1F66 =8038	D
8788	2254			DTBLE	'LOG'	\$18B3 =6323	D
8790	2256			DTBLE	'EXP'	\$1EC1 =7873	D
8792	2258			DTBLE	'COS'	\$1FA2 =8098	D
8794	225A			DTBLE	'SIN'	\$1FA9 =8105	D
8796	225C			DTBLE	'TAN'	\$1FF2 =8178	D
8798	225E			DTBLE	'ATN'	\$2056 =8278	D
8800	2260			DTBLE	'PEEK'	\$1688 =5768	D
8802	2262			DTBLE	'LEN'	\$15F6 =5622	D
8804	2264			DTBLE	'STR\$'	\$12E9 =4841	D
8806	2266			DTBLE	'VAL'	\$1627 =5671	D
8808	2268			DTBLE	'ASC'	\$1605 =5637	D
8810	226A			DTBLE	'CHR\$'	\$1566 =5478	D
8812	226C			DTBLE	'INDEX'	\$2C80 =11392	D
8814	226E			DTBLE	'LEFT\$'	\$157A =5493	D
8816	2270			DTBLE	'RIGHT\$'	\$15A6 =5542	D
8818	2272			DTBLE	'MID\$'	\$15B1 =5553	D
8820	2274	121	79		FPOT PRECEDENCE FOR ADDITION		
8821	2275				FPOT DISP. FOR ADDITION	\$16D8=5864	DEC.
8823	2277	121	79		FPOT PRECEDENCE FOR SUBTRACTION		
8824	2278				FPOT DISP. FOR SUBTRACT	\$16C1=5825	DEC.
8826	227A	123	7B		FPOT PRECEDENCE FOR MULTIPLY		
8827	227B				FPOT DISP. FOR MULTIPLY	\$18F3=6387	DEC.

8829	227D	123	7B	FPOT PRECEDENCE FOR DIVISION	
9330	227E			FPOT DISP. FOR DIVISION S1A0C=6668 DEC.	
8832	2280	126	7F	FPOT PRECEDENCE FOR EXPONENTIATE	
8833	2281			FPOT DISP. FOR EXPONEN. \$1E4E=7758 DEC.	
8835	2283	80	50	FPOT PRECEDENCE FOR LOGICAL AND	
8836	2284			FPOT DISP. FOR LOGIC AND \$0E8B=3723 D	
8838	2286	70	46	FPOT PRECEDENCE FOR LOGICAL OR	
8839	2297			FPOT DISP. FOR LOGIC OR \$0E88=3720 DEC.	
8841	2289	125	7D	FPOT PRECEDENCE FOR COMPLEMENT SIGN	
8842	228A			FPOT DISP. FOR COMPL'T SGN \$1E87=7315 D	
8844	228C	90	58	FPOT PRECEDENCE FOR COMPLIMENT NUMBER	
8845	228D			FPOT DISP. FOR COMPL'T NUM \$0DE9=3561 D	
8847	228F	100	64	FPOT PRECEDENCE FOR RELATIONAL OPERATOR	
8848	2290			FPOT DISP. FOR RELATL OPER \$0EB8=3763 D	
8850	2292			END OF DISPATCH TABLE	
8960	2300			RW LIST 'END'	TOKEN=\$90=128 D.
8963	2303			RW LIST 'FOR'	TOKEN=\$91=129 D.
8966	2306			RW LIST 'NEXT'	TOKEN=\$92=130 D.
8970	230A			RW LIST 'DATA'	TOKEN=\$93=131 D.
8974	230E			RW LIST 'INPUT'	TOKEN=\$94=132 D.
8979	2313			RW LIST 'DIM'	TOKEN=\$95=133 D.
8982	2316			RW LIST 'READ'	TOKEN=\$96=134 D.
8986	231A			RW LIST 'LET'	TOKEN=\$97=135 D.
8989	231D			RW LIST 'GOTO'	TOKEN=\$98=136 D.
8993	2321			RW LIST 'RUN'	TOKEN=\$99=137 D.
8996	2324			RW LIST 'IF'	TOKEN=\$9A=138 D.
8998	2326			RW LIST 'RESTORE'	TOKEN=\$9B=139 D.
9005	232D			RW LIST 'GOSUB'	TOKEN=\$9C=140 D.
9010	2332			RW LIST 'RETURN'	TOKEN=\$9D=141 D.
9016	2338			RW LIST 'REM'	TOKEN=\$9E=142 D.
9019	233B			RW LIST 'STOP'	TOKEN=\$9F=143 D.
9023	233F			RW LIST 'ON'	TOKEN=\$90=144 D.
9025	2341			RW LIST 'NULL'	TOKEN=\$91=145 D.
9029	2345			RW LIST 'WAIT'	TOKEN=\$92=146 D.
9033	2349			RW LIST 'LOAD'	TOKEN=\$93=147 D.
9037	234D			RW LIST 'SAVE'	TOKEN=\$94=148 D.
9041	2351			RW LIST 'DEF'	TOKEN=\$95=149 D.
9044	2354			RW LIST 'POKE'	TOKEN=\$96=150 D.
9048	2358			RW LIST 'PRINT'	TOKEN=\$97=151 D.
9053	235D			RW LIST 'CONT'	TOKEN=\$98=152 D.
9057	2361			RW LIST 'LIST'	TOKEN=\$99=153 D.
9061	2365			RW LIST 'CLEAR'	TOKEN=\$9A=154 D.
9066	236A			RW LIST 'INDEX<'	TOKEN=\$9B=155 D.
9072	2370			RW LIST 'OPEN'	TOKEN=\$9C=156 D.
9076	2374			RW LIST 'CLOSE'	TOKEN=\$9D=157 D.
9081	2379			RW LIST 'FIND'	TOKEN=\$9E=158 D.
9085	237D			RW LIST 'DEV'	TOKEN=\$9F=159 D.
9088	2380			RW LIST 'FLAG'	TOKEN=\$A0=160 D.
9092	2384			RW LIST 'NEW'	TOKEN=\$A1=161 D.
9095	2387			RW LIST 'TAB'	TOKEN=\$A2=162 D.
9099	238B			RW LIST 'TO'	TOKEN=\$A3=163 D.
9101	238D			RW LIST 'FN'	TOKEN=\$A4=164 D.
9103	238F			RW LIST 'SPC('	TOKEN=\$A5=165 D.
9107	2393			RW LIST 'THEN'	TOKEN=\$A6=166 D.
9111	2397			RW LIST 'NOT'	TOKEN=\$A7=167 D.

(14)

9114	239A	RW LIST 'STEP'	TOKEN=\$A9=168 D.
9118	239E	RW LIST '+'	TOKEN=\$A9=169 D.
9119	239F	RW LIST '-'	TOKEN=\$AA=170 D.
9120	23A0	RW LIST '*'	TOKEN=\$AB=171 D.
9121	23A1	RW LIST '/'	TOKEN=\$AC=172 D.
9122	23A2	RW LIST '^'	TOKEN=\$AD=173 D.
9123	23A3	RW LIST 'AND'	TOKEN=\$AE=174 D.
9126	23A6	RW LIST 'OR'	TOKEN=\$AF=175 D.
9128	23AB	RW LIST '>'	TOKEN=\$B0=176 D.
9129	23A9	RW LIST '='	TOKEN=\$B1=177 D.
9130	23AA	RW LIST '<'	TOKEN=\$B2=178 D.
9131	23AB	RW LIST 'SGN'	TOKEN=\$B3=179 D.
9134	23AE	RW LIST 'INT'	TOKEN=\$B4=180 D.
9137	23B1	RW LIST 'ABS'	TOKEN=\$B5=181 D.
9140	23B4	RW LIST 'USR'	TOKEN=\$B6=182 D.
9143	23B7	RW LIST 'FRE'	TOKEN=\$B7=183 D.
9146	23BA	RW LIST 'POS'	TOKEN=\$B8=184 D.
9149	23BD	RW LIST 'SQR'	TOKEN=\$B9=185 D.
9152	23C0	RW LIST 'RND'	TOKEN=\$BA=186 D.
9155	23C3	RW LIST 'LOG'	TOKEN=\$BB=187 D.
9158	23C6	RW LIST 'EXP'	TOKEN=\$BC=188 D.
9161	23C9	RW LIST 'COS'	TOKEN=\$BD=189 D.
9164	23CC	RW LIST 'SIN'	TOKEN=\$BE=190 D.
9167	23CF	RW LIST 'TAN'	TOKEN=\$BF=191 D.
9170	23D2	RW LIST 'ATN'	TOKEN=\$C0=192 D.
9173	23D5	RW LIST 'PEEK'	TOKEN=\$C1=193 D.
9177	23D9	RW LIST 'LEN'	TOKEN=\$C2=194 D.
9180	23DC	RW LIST 'STR\$'	TOKEN=\$C3=195 D.
9184	23E0	RW LIST 'VAL'	TOKEN=\$C4=196 D.
9187	23E3	RW LIST 'ASC'	TOKEN=\$C5=197 D.
9190	23E6	RW LIST 'CHR\$'	TOKEN=\$C6=198 D.
9194	23EA	RW LIST 'INDEX'	TOKEN=\$C7=199 D.
9199	23EF	RW LIST 'LEFT\$'	TOKEN=\$C8=200 D.
9204	23F4	RW LIST 'RIGHT\$'	TOKEN=\$C9=201 D.
9210	23FA	RW LIST 'MID\$'	TOKEN=\$CA=202 D.
9214	23FE	RW LIST '0' MARKS END OF TABLE	
9250	2422	ORG "RUN"	
9473	2501	BASOD= LOCATION OF DEVICE # FOR OUTPUT	
9478	2506	ARC TAN	SIZE=109
10226	27F1	DISK ERROR NUMBER	
10229	27F5	BASIC INPUT	BASIC I/O HOOK
10248	2808	BASIC OUTPUT	BASIC I/O HOOK
10492	23FC	ORG "LOAD"	
10646	2996	ORG "SAVE"	
10961	2AD1	ORG "OPEN"	
11206	2BC6	ORG "CLOSE"	
11268	2C04	ORG "DEV"	
11314	2C32	ORG "INDEX<"	
11392	2C80	ORG "INDEX"	
11535	2D0F	ADDR. OF INPDN (CONSOLE INPUT)	
11556	2D24	ADDR. OF INPSDN (SPECIFIC DEV INPUT)	
11571	2D33	ADDR. OF OUTD (CONSOLE OUTPUT)	
11577	2D39	ADDR. OF OUTSD (SPECIFIC DEV OUTPUT)	
11657	2D89	LSB MEMORY INPUT POINTER	
11658	2D8A	MSB MEMORY INPUT POINTER	

18

11661	2D8D			LSB MEMORY OUTPUT POINTER
11662	2D8E			MSB MEMORY OUTPUT POINTER
11664	2D90			CONSOLE INPUT DEVICE NUMBERS
11665	2D91			CONSOLE OUTPUT DEVICE NUMBERS
11666	2D92			LSB MEM. I/O PNTR FOR INDIRECT FILES
11667	2D93			MSB MEM. I/O PNTR FOR INDIRECT FILES
11668	2D94			ACTIVE DEVICE FLAG (INPUT)
11686	2DA6			ACTIVE DEVICE FLAG (OUTPUT)
11702	2DB6			LSB CONSTANT MEM. SIZE (0=USE ALL MEM)
11703	2DB7			MSB CONSTANT MEM. SIZE (0=USE ALL MEM)
11774	2DFE			PEEK(11774)+PEEK(11775)*256=
11775	2DFE			LINE # WHERE DISC ERROR OCCURRED
11802	2E1A			NUMBER FLOPPY DRIVES ON SYSTEM
12644	3164	2	02	SEEK RETRY COUNT (0=NO RETRIES)
12796	31FC	6	06	READ RETRY COUNT (0=NO RETRIES)
13002	32CA	2	02	WRITE RETRY COUNT (0=NO RETRIES)
13322	340A			NUMBER OF CD-74 HARD DISC ON SYSTEM
14387	3833			# LINES/PAGE FOR PRINTER ON PORT 5
14457	3879			DESIRED # OF PRINTED LINES/PAGE (ON#5)
14639	392F			CONTROL C ENABLE OR DISABLE FLAG
14646	3936			POKE DECIMAL 91 TO ENABLE INDIRECT FILE
14721	3981			POKE DECIMAL 24 TO ENABLE INDIRECT FILE
14942	3A5E			CONSOLE PORT INPUT I/O DISTR.
14948	3A64			\$A9=169>LEVEL I \$4C= 76>LEVEL III
14949	3A65			\$C8=200>LEVEL I \$03= 3>LEVEL III
14950	3A66			\$8D=141>LEVEL I \$D8=216>LEVEL III
14989	3A8D			CONSOLE PORT OUTPUT I/O DISTR.
15908	3E24			CURRENT # OF PRINTED LINES (PORT 5)
15297	3BC1			440/540 OUTPUT I/O DISTR.
15448	3C58			ASCII/POLLED KEYBD. INPUT I/O DISTR.
15577	3CD9			430 BOARD INPUT I/O DISTR.
15717	3D65			430 BOARD OUTPUT I/O DISTR.
15751	3D87			MEMORY INPUT I/O DISTR.
15792	3DB0			MEMORY OUTPUT I/O DISTR.
15868	3DFC			LINE PRINTER #5 OUTPUT I/O DISTR.
16036	3ED6	19	13	LOCATION OF CONTROL S
16096	3EE0	17	11	LOCATION OF CONTROL Q
16127	3EFF	03	03	LOCATION OF CONTROL C
16131	3F03	15	0F	LOCATION OF CONTROL O
16137	3F09	07	07	LOCATION OF CONTROL D
16139	3F0B	23	17	LOCATION OF CONTROL W
18915	497E			ORG "FLAG"
19022	4A4E			ORG "FIND"
19418	4BD9			ORG "LIST"
19798	4D56			CA-10X DEV INDEX (0,2,4..30)=(1,2..16)
19803	4D5B			CA-10X OUTPUT
19862	4D96			CA-10X INPUT
24559	5FF9			DAY STORAGE LOCATION
24570	5FFA			MONTH STORAGE LOCATION
24571	5FFB			YEAR STORAGE LOCATION
24576	6000			START OF BASIC WORK SPACE
48658	BE12			OS65U (48K) WARM START
61438	EFEE			HARD DISC CYLINDER ADDR.
61439	EFEE			HARD DISC CYLINDER ADDR.

(17)

OS-65U NMHZ AND THE NEC SPINWRITER

The following POKES modify the NEC driver contained in the program "NECDRV". The modification allows the "PRINTER STALLED" timeout routine to work at NMHZ.

To insert the changes:

1) TYPE: LOAD"NECDRV" <CR>

2) Enter the following lines:

POKE 25012,162: POKE 25013,217: POKE 25014,165 <CR>

POKE 25015,224: POKE 25016,45: POKE 25017,0 <CR>

POKE 25018,245: POKE 25019,73: POKE 25020,16 <CR>

POKE 25021,240: POKE 25022,15: POKE 25023,41 <CR>

POKE 25024,16 : POKE 25025,208: POKE 25026,9 <CR>

POKE 25027,169: POKE 25028,6: POKE 25029,32 <CR>

POKE 25030,184: POKE 25031,62: POKE 25032,202 <CR>

POKE 25033,208: POKE 25034,235: POKE 25035,234 <CR>

POKE 25036,169: POKE 25037,128: POKE 25038,96 <CR>

POKE 25396,162: POKE 25397,217: POKE 25398,165 <CR>

POKE 25399,224: POKE 25400,45: POKE 25401,0 <CR>

POKE 25402,245: POKE 25403,73: POKE 25404,16 <CR>

POKE 25405,240: POKE 25406,15: POKE 25407,41 <CR>

POKE 25408,16: POKE 25409,208: POKE 25410,9<CR>

POKE 25411,169: POKE 25412,6: POKE 25413,32<CR>

POKE 25414,184: POKE 25415,62: POKE 25416,202<CR>

POKE 25417,208: POKE 25418,235: POKE 25419,234<CR>

POKE 25420,169: POKE 25421,128: POKE 25422,96<CR>

3) Now type:

SAVE "NECDRV", "PASS"<CR>

4) The changes have been saved to disc.

NOTE: The program must be RUN to make the changes effective.

Applicability:

ALL NMHZ VERSIONS OF OS-65U (V1.2)