THE OMIKRON MAPPER

OWNERS MANUAL

09/01/80

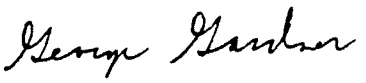
DEAR CUSTOMER,

THE OMIKRON MAPPERS HAVE BEEN IN PRODUCTION FOR OVER ONE YEAR NOW AND THEY HAVE PROVED THEMSELVES TOTALLY RELIABLE IN A WIDE VARIETY OF APPLICATIONS. OMIKRON HAS ENDEAVORED TO INCLUDE THE FEATURES AND UTILITIES NECESSARY TO SUPPORT THE FULL RANGE OF CP/M SOFTWARE CURRENTLY AVAILABLE. OMIKRON HAS RECENTLY ANNOUNCED ADVANCED SUPPORT FOR WORD PROCESSING AND BUSINESS APPLICATIONS AS WELL AS CP/M VERSION 2.2 FOR TOTAL COMPATIBILITY WITH THE CP/M FOR THE MODEL II.

RECENTLY, APPARAT (DENVER, CO.) ANNOUNCED NEWDOS/80. THIS IS A TOTALLY REWRITTEN AND GREATLY ENHANCED VERSION OF THEIR EARLIER NEWDOS. WHEN USED IN CONJUNCTION WITH THE MAPPER II, NEWDOS/80 ALLOWS THE EXECUTION OF TRS80-DOS COMPATIBLE SOFTWARE ON 8" DISK DRIVES. OMIKRON IS PLEASED TO ANNOUNCE THAT THIS OPERATING SYSTEM IS NOW AVAILABLE TO OUR CUSTOMERS.

OMIKRON IS DETERMINED TO OFFER PRODUCTS THAT OFFER THE GREATEST VALUE IN THE HIGHLY COMPETITIVE MICROCOMPUTER INDUSTRY. WE FEEL THE MAPPERS ARE AN EXCELLENT EXAMPLE OF THIS PHILOSOPHY AND WE ARE CONFIDENT THAT THEY WILL REMAIN THE MOST COST EFFECTIVE PATH TO UPGRADING YOUR TRS-80 FOR BUSINESS AND PROFESSIONAL APPLICATIONS. WE ARE ALWAYS OPEN TO SUGGESTIONS FOR IMPROVEMENTS AND REMAIN READY TO ASSIST YOU IN CASE OF DIFFICULTY.

SINCERELY YOURS,



GEORGE GARDNER

REVISED CP/M VERSION 2.2

Preliminary Documentation

Omikron has finished the development of the enhanced version of CP/M 2.2. This new version is a major rewrite of the previous software and contains many features not previously available. This document is intended to give more specific details on the system, however, all of the information in the Omikron owner's manual still applies.

This new revision is designed to provide maximum flexibility to Mapper I owners. All of the drive parameters are contained in a table. There is a separate area in the table for each drive so each drive can have the software optimized for it. The software now supports options for setting the step rates, setting the track numbers, and allowing the use of two-sided drives.

In order to use the new software, the prom on the Mapper I must be replaced. To replace the prom, open the keyboard according to the original installation instructions. It is not necessary to unplug the Mapper I. Simply remove the prom from its socket and substitute the new one. Remove the prom by prying it out with a screwdriver. Notice that all of the notches on all of the chips lie on the left side of the board. Be absolutely certain that the notch on the end of the new prom is oriented on the same side as the notch was on the old prom.

Omikron has tried to maintain compatibility with previous releases of the software. All of the software previously released will run with the new prom. All of the utilities are still useable but the new utilities, in many cases, have additional features. The new Disktest will allow the user to specify the number of tracks to test, and the test may be aborted by pressing any key. The new DDT program has the display corrected for the 64 character screen. The new format programs support the drive re-addressing features and the new Mformat, Copy, and Disktest programs support 80 track drives. All of the utilities have been updated to support lower case characters.

In addition to supporting the drive options, Omikron has added special features to enhance word processing on the TRS-80. The new software has an interrupt driven keyboard routine that corrects many problems found with the use of word processing software. It is almost impossible to drop characters when the interrupt option is specified. The blinking cursor makes screen editing much easier, and the auto repeat is quickly habit forming.

In order to customize the new software for each user, Omikron has developed a new program called "Options". This program is designed to customize the software by asking a series of questions in plain English. The program is invoked by simply typing the word "Options". After running the options program, the answers are stored permanently on the CP/M disk. Although extensive error checking has been designed into the program, serious mistakes can

arise if the wrong answers are given.

Always run the Options program on a copy of the original CP/M disk. This way the original is always perfect if problems arise.

When the word "Options" is typed, the program signon message appears with the message: "Prom Version 'A' Checksum Tests OK". If there is ever a problem with the Mapper I, run the Options program and verify that the checksum still tests OK. Some Eproms may lose data after a period of time.

At this point, the program is divided into two major sections. The first sets the drive parameters and the second sets the configuration options. Either section may be upgraded without entering the other section. Note that the old set-up program duplicates many of the questions. Use the Options program to set the default values, then use the set-up if temporary changes are required.

If you answer "yes" to the question on updating the Options table, the program will ask a series of ten questions. Most of these questions are explained in the Mapper manual. Refer to the page numbers given if you have questions. There are also several new questions which are explained here in greater detail:

1) Video graphics P.18 #5.
2) Enable lower case P.18 #3.
3) Blink-cursor
4) Deleted characters to echo P.17 #1.

Note that under CP/M version 2.2 the Break key(08H) is used to backspace. In most instances, it is required to allow the deleted characters to echo.

5) Printer line counter P.18 #4.
6) Printer linefeeds P.18 #2.
7) Swap "5B with 5EH".

Radio Shack uses two versions of the character generator prom in the video circuitry. This question is best answered by experimenting. The correct display will show an "up arrow" for control characters, and the right and left brackets should appear as mirror images of each other.

8) RS 232 in place of keyboard.

A few Mapper owners also own video terminals such as the Soroc or ADM 3. When a terminal is wired to the Radio Shack RS 232 board, this option allows the use of the terminal in place of the TRS-80 keyboard. Note that the Serial program is used to set up the RS 232 board. Use control "P" to send video output to the RS 232 board.

9) The RS 232 input routine is normally addressed through the CP/M reader function. There are two options. The CP/M protocol requires the computer to wait for the next character. This is useful when transferring files from one computer to another with the PIP command. Omikron supplies a brief program called "term". This converts the TRS-80 to a dumb terminal. If the computer waits to receive a character, no characters can be sent from the keyboard. Omikron allows the Reader to return 0 in "A" if no character is ready or with the character in the "A" register if a character was found.

10) Disable interrupt input.
The repeat function can be disabled by disabling the interrupts.

Drive Options

Most of the questions are self-explanatory, however, there are several areas that need clarification.

Double sided drives present special problems. The Omikron software is designed to use a separate CP/M drive assignment for each side of the drive. For example, "A" could be the top side and "B" the bottom side. The software is designed to use drive select "4" for the side select on the mini drives. Normally the cable connectors have pins removed to address to drive. Note that a double sided drive requires pins in locations 32 and 34, as well as the normal choice between 10,12, or 14 to select 0,1, or 2. Eight inch double sided drives do not require the side select line. The Omikron software requires the 8" drives to be jumpered to select each side with a different drive select line. A potential problem arises when 5" and 8" drives are mixed. Because a 5" double sided drive uses DS4 to select the side, it is not possible to use DS4 to select the 8" drive. An example is: use the two sided mini as drive 0, address the two sided 8" as 1 and 2.

One section of the Options configuration program is devoted to double sided drives. CP/M must keep a record of the track position on each drive. Double sided drives present a problem because both sides use the same head positioning mechanism. When the track counter is is changed for one side, the other side must also be updated. One question asks if the top or bottom side is being configured. If the bottom side is being specified, a question asks: indicate the number of the top side. This refers to the drive number that was used to address the drive. The top side is selected when the drive select line goes low; the bottom side is selected when both the drive and side select line go low.

When setting the step rate on mini drives, there are four choices. Normally, the TRS-80 is limited to 12 ms. per track. If a Mapper II is installed, it is possible to step the mini at 6 ms. if a "0" is specified in answer to the question.

Omikron has developed a method of translating CP/M drive references to any of the drives present. For instance, drive 0 can be referenced with the letters A,B,C, or D. This allows total freedom to set the drives up for optimizing the applications software. Refer to page 20 in the owner's manual. The new revision is actually much more flexible than page 20 implies. However, certain precautions are necessary. Don't use the same CP/M letter for two different drives. Be particularly careful with drive A. The warm boot is always on drive A, independent of the physical drive addressing. Refer to page 20 and double check that a CP/M system has been sysgened on every disk used in drive A.

After all of the questions are answered, the final question asks if you choose to write all of the answers to the disk. This is the last chance to correct mistakes! Once the disk is updated, check the system very carefully. Use the CP/M Stat command to verify that

each drive shows the correct storage; a blank 35 track shows 70K, 40 track shows 81K, 77 track shows 164K, and 80 track shows 171K. See that each drive steps properly and the directory reads correctly. A note of caution is required when mixing 35 and 40 track drives. Note that it is possible to switch disks between the drives, however, the 35 track drive cannot read files stored on the outer 5 tracks of a 40 track disk. Note also that a disk formatted for 35 tracks cannot be used in a 40 track drive without errors when the outer 5 tracks are accessed. You can use OMCOPY to copy a 35 to a 40 track disk but not a 40 to a 35 track disk. The PIP command must be used to copy the files from a 40 to a 35 track disk.

The video driver now contains several new features in addition to those listed in the Mapper manual. These new features duplicate the ones in the Radio Shack driver so software conversion to CP/M is much simpler.

17H = Cursor up, restore old character. Note that the old prom used 1BH for this function.

1DH = Home cursor, don't clear screen.

1EH = Clear to end of line -- also called with the 2 character sequence: "escape" "T".

1FH = Clear to end of field.

This ends our discussion of the new software. We have tried to anticipate your needs, but feel free to write us if you have a specific feature in mind that we haven't included. Good luck.

INSTALLATION INSTRUCTIONS FOR THE MAPPER 1

THE MAPPER PRINTED CIRCUIT BOARD IS DESIGNED TO FIT NEATLY INSIDE THE KEYBOARD UNIT. HOWEVER THIS REQUIRES THAT YOU DISASSEMBLE THE KEYBOARD UNIT. THIS CAN BE DONE EASILY AND EFFICIENTLY IF YOU FOLLOW THESE INSTRUCTIONS CAREFULLY.

FIRST REMOVE ALL CABLES TO THE KEYBOARD UNIT. PLACE THE KEYBOARD UNIT UPSIDE DOWN ON A SOFT SURFACE SUCH AS A TOWELL OR BED. ORIENT THE UNIT SO THAT THE SPACE BAR ON THE KEYBOARD IS NEAREST YOURSELF. YOU WILL HAVE TO REMOVE THE SIX PHILLIPS HEAD SCREWS HOLDING THE TWO HALVES TOGETHER. ONE OF THEM MAY HAVE A SEALANT AROUND IT. IF YOU FIRST HEAT THE SCREWDRIVER THIS SEALANT WILL MELT ENOUGH TO REMOVE THE SCREW. REMOVE THE OTHER FIVE SCREWS AND SET ALL SIX ASIDE.

USE YOUR HANDS TO HOLD THE TWO HALVES TOGETHER WHILE YOU TURN THE UNIT UPRIGHT. THE SPACE BAR SHOULD STILL BE ON THE SIDE NEAREST YOU. LIFT OFF THE TOP COVER. YOU WILL SEE THE KEYBOARD PRINTED CIRCUIT BOARD MOUNTED ABOVE THE LARGER MAIN BOARD. THERE ARE FIVE PLASTIC SPACERS HOLDING THE TWO BOARDS APART. CAREFULLY LIFT THE KEYBOARD OFF THE SPACERS, ROTATING IT UPWARDS SO THAT NO STRAIN IS PLACED ON THE FLAT JUMPER STRIP JOINING THE TWO BOARDS. REMOVE THE SPACERS AND SET THEM ASIDE. SET THE KEYBOARD LOOSELY DOWN ON TOP OF THE MAIN BOARD. NOW USE YOUR HANDS TO PRY THE TWO BOARDS OUT OF THE BOTTOM SECTION. CAREFULLY TURN THE WHOLE ASSEMBLY UPSIDE DOWN AND SET IT DOWN. BE CAREFUL THAT THE TWO BOARDS DON'T SEPARATE AND STRAIN THE FLAT RIBBON CONNECTOR.

ON THE RIGHT SIDE OF THE MAIN BOARD IS A FORTY PIN INTEGRATED CIRCUIT. THIS IS THE (Z80) MICROPROCESSOR. YOU WILL HAVE TO REMOVE THIS CHIP FROM ITS SOCKET AND PLACE IT IN THE SOCKET ON THE MAPPER BOARD. AT THIS POINT NOTE THAT THIS CHIP CAN BE DAMAGED BY STATIC ELECTRIC CHARGES. DO NOT TOUCH THE LEADS UNNECESSARILY. DO NOT WEAR SYNTHETIC CLOTHING OR WORK ON SYNTHETIC RUGS.

REMOVE THE (Z80) CHIP FROM ITS SOCKET BY PRYING IT OUT WITH A FLAT BLADE SCREWDRIVER (ALTERNATELY PRY ONE END THEN THE OTHER) AND SET IT ASIDE. PICK UP THE MAPPER BOARD AND ALIGN IT SO THE PINS UNDERNEATH THE BOARD ARE ORIENTED IN TWO VERTICAL ROWS (FROM FRONT TO BACK). SET THE BOARD ON THE EDGE OF A TABLE WITH THE PINS SAFELY OVERHANGING THE EDGE. PICK UP THE (Z80) CHIP AND ALIGN IT WITH THE NOTCH TOWARDS THE LEFT (OBSERVE THAT THE NOTCHES IN ALL THE CHIPS ON THE MAPPER ARE LOCATED ON THE LEFT). CAREFULLY SET THE (Z80) IN THE FORTY PIN SOCKET AND PRESS IT DOWN FIRMLY UNTIL FULLY SEATED. CHECK FOR BENT PINS VERY CAREFULLY.

INSTALLATION OF THE MAPPER I CONTINUED

ORIENT THE MAPPER SO IT FITS CORRECTLY OVER THE MAIN BOARD (THE MAPPER IS INSTALLED WITH THE (Z80) CHIP NEAREST THE BACK). CAREFULLY ALIGN THE PINS ON THE MAPPER BOARD TO FIT INSIDE THE (Z80) SOCKET. CAREFULLY BUT FIRMLY ROCK THE MAPPER BOARD FROM LEFT TO RIGHT WHILE PRESSING IT INTO THE SOCKET. DO THIS UNTIL THE MAPPER BOARD IS COMPLETELY SEATED IN THE (Z80) SOCKET.

YOU HAVE NOW COMPLETED THE INSTALLATION OF THE MAPPER UNIT. TO REASSEMBLE YOUR TRS80 PLACE THE BOTTOM HALF IN FRONT OF YOU WITH THE LONGER PAIR OF STUDS FARTHEST FROM YOU. CAREFULLY TURN THE MAIN BOARD AND KEYBOARD ASSEMBLY OVER AND PLACE IT INSIDE THE BOTTOM SHELL. THREAD THE SINGLE WIRE OUT THROUGH THE EXPANSION INTERFACE CUTOUT. (THIS WIRE MUST BE CONNECTED TO THE DISK CONTROLLER BOARD WHEN EIGHT INCH DRIVES ARE USED). WHEN THE STUDS ARE PROPERLY INSIDE THE HOLES IN THE MAIN BOARD ROTATE THE KEYBOARD ASSEMBLY UP AND REPLACE THE FIVE PLASTIC SPACERS. THEY GO AROUND THE STUDS WITH THE HOLES IN THEM. SET THE KEYBOARD BACK DOWN AND CHECK THAT EVERYTHING LINES UP PROPERLY. PRESS DOWN ON THE KEYBOARD AND MAKE SURE THE BOARD SETS ON THE LIPS ON THE FIVE STUDS. PLACE THE TOP COVER OVER THE KEYBOARD WHILE CHECKING THAT THE ON OFF LIGHT FITS THROUGH THE HOLE IN THE TOP COVER. TURN THE ENTIRE UNIT OVER WHILE SQUEEZING THE TWO HALVES TOGETHER WITH YOUR HANDS. SEPARATE THE SIX SCREWS INTO THREE PAIRS ACCORDING TO LENGTH. THE LONGEST TWO GO IN THE HOLES IN THE UPPER MIDDLE OF THE BACK. THE MIDDLE TWO GO IN THE HOLES IN THE MIDDLE OF THE BACK AND THE SHORTEST TWO GO IN THE TWO HOLES NEAREST THE EDGE. IF ANY OF THESE SIX SCREWS ARE DIFFICULT TO REPLACE BACK THEM OUT AND TRY AGAIN. THEY MAY NOT BE THREADING PROPERLY.THIS COMPLETES THE INSTALLATION OF THE MAPPER. TURN THE UNIT OVER AND REPLACE THE CABLES.

INSTALLATION NOTICE

MAPPER I: AFTER COMPLETING THE INSTALLATION OF THE MAPPER CIRCUIT BOARD, RECONNECT THE VIDEO AND POWER CONNECTORS. DO NOT CONNECT THE EXPANSION INTERFACE FOR THIS TEST. TURN ON THE POWER AND OBSERVE THE VIDEO DISPLAY. THE MESSAGE "T = TRS80 C = CPM" SHOULD APPEAR ON THE SCREEN. IF THE MESSAGE IS NOT PRESENT, IMMEDIATELY TURN OFF THE POWER AND RECHECK THE INSTALLATION. IF YOU CAN'T FIND ANYTHING WRONG REMOVE THE MAPPER BOARD AND VERIFY THAT YOUR TRS80 IS STILL OPERATING. CONTACT OMIKROM IMMEDIATELY! IF THE SIGNON MESSAGE HAS APPEARED TYPE "T" WHILE HOLDING THE BREAK KEY DOWN. THE TRS80 BASIC SHOULD TAKE CONTROL. IF EVERYTHING SEEMS NORMAL AND YOU HAVE PURCHASED A "MAPPER II" PROCEED TO THE "DISK ADAPTER INSTALLATION" SECTION.

MAPPER II: AFTER COMPLETING THE INSTALLATION OF THE DISK ADAPTER, CONNECT THE UNIT TO THE TRS80 KEYBOARD AND TURN ON THE POWER TO BOTH THE UNITS. AFTER VERIFYING THAT THE SIGNON MESSAGE STILL APPEARS, INSERT THE TRS80 DOS DISKETTE OR THE CPM DISKETTE INSIDE DRIVE 0 (OR "A") AND TYPE "T" OR "C" RESPECTIVELY. VERIFY THAT THE SYSTEM WILL BOOT. IF YOU ENCOUNTER ANY PROBLEMS CONTACT OMIKRON IMMEDIATELY.

SYSTEM TESTING

OMIKRON RECOMENDS THAT THE FOLLOWING TWO TESTS BE RUN ON YOUR TRS80 IN ORDER TO VERIFY CORRECT FUNCTIONING. THIS SHORT PROGRAM, WRITTEN IN PROM BASIC, TESTS THE NORMAL OPERATION OF THE TRS80.

```
10 X=X+1:?X;
20 GOTO 10
RUN
```

WITH THE EXPANSION INTERFACE CONNECTED, TYPE "T" WHILE HOLDING THE BREAK KEY DOWN. AFTER ENTERING THE TRS80 BASIC, ENTER THE ABOVE PROGRAM AND RUN IT FOR AT LEAST 15 MINUTES. IF THERE IS ANY PROBLEM EXECUTING THE PROGRAM CONTACT OMIKRON FOR ADVICE. IF NO PROBLEMS ARE ENCOUNTERED, BOOT UP THE CPM SYSTEM AND TYPE "MEMTEST" TO LOAD THE MEMORY TEST PROGRAM. RUN THIS PROGRAM FOR AT LEAST 15 MINUTES IN EACH BLOCK OF MEMORY. IF THE MEMORY TESTS PROVE THE MEMORY IS FUNCTIONING CORRECTLY YOUR SYSTEM SHOULD BE RUNNING CORRECTLY.

IF YOUR TRS80 HAS NEVER WORKED RELIABLY, RADIO SHACK HAS SEVERAL MODIFICATIONS WHICH THEY ARE CURRENTLY INSTALLING IN ALL MODELS. IF YOUR TRS80 WORKS PERFECTLY IN THE EVENING BUT NOT DURING THE DAY, YOU PROBABLY NEED AN AC LINE FILTER.

DISK ADAPTER INSTALLATION 1

THE SQUARE CIRCUIT BOARD WITH EIGHT CHIPS ON IT IS THE "MAPPER II" DISK ADAPTER BOARD. THIS BOARD FEATURES AN EXTERNAL DATA SEPARATOR THAT SWITCHES BETWEEN THE MINI AND EIGHT INCH DRIVES. THIS DATA SEPARATOR ALSO WORKS WHEN RUNNING IN THE TRSDOS MODE. THIS SEPARATOR WILL SIGNIFICANTLY REDUCE ALL DISK ERRORS. THE BOARD INSTALLS INSIDE THE EXPANSION INTERFACE. TO INSTALL THIS BOARD PROCEED AS FOLLOWS.

FIRST REMOVE ALL CABLES TO THE EXPANSION INTERFACE AND UNPLUG THE POWER SUPPLY CABLE FROM THE WALL. TURN THE UNIT UPSIDE DOWN. YOU WILL HAVE TO REMOVE THE SIX PHILLIPS HEAD SCREWS HOLDING THE TWO HALVES TOGETHER. SOME OF THEM MAY HAVE A SEALANT AROUND THEM. IF YOU FIRST HEAT THE SCREWDRIVER THIS SEALANT WILL MELT ENOUGH TO LOOSEN THE SCREWS. REMOVE THE SCREWS AND SET THEM ASIDE. REMOVE THE BOTTOM COVER. ORIENT THE UNIT SO THAT THE POWER SUPPLY CAVITY IS ON YOUR LEFT. LOCATE THE WESTERN DIGITAL (1771) INTEGRATED CIRCUIT. THIS IS THE LARGE CHIP INSERTED IN A FORTY PIN SOCKET. NOTE THE NOTCH IN THE CENTER OF ONE END OF THE CHIP. OBSERVING THE STATIC ELECTRICTY PRECAUTIONS PREVIOUSLY MENTIONED, REMOVE THE FORTY PIN CHIP BY PRYING IT LOOSE WITH A FLAT BLADE SCREWDRIVER (FIRST ONE END THEN THE OTHER) AND SET IT SAFELY ASIDE.

PICK UP THE MAPPER II CIRCUIT BOARD AND ALIGN IT WITH THE FORTY PIN SOCKET TOWARDS THE BACK LEFT CORNER. PUT THE MAPPER II DOWN FLAT ON THE TABLE, WITH THE PINS ON THE BACK SIDE OF THE BOARD OVERHANGING THE EDGE OF THE TABLE AND THE FORTY PIN SOCKET SEATED FIRMLY ON THE TABLE. TAKE THE WESTERN DIGITAL CHIP AND ALIGN IT WITH THE NOTCH ON THE LEFT SIDE OVER THE FORTY PIN SOCKET. NOW PRESS THE WESTERN DIGITAL CHIP DOWN INTO THE SOCKET ON THE MAPPER II CHECKING FOR PROPER ALIGNMENT. CHECK THAT ALL OF THE IC'S ON THE MAPPER II ARE ALIGNED IN THE SAME DIRECTION.

THE DISK ADAPTER BOARD PLUGS INTO THE WESTERN DIGITAL CHIP SOCKET ON THE EXPANSION INTERFACE BOARD. THERE ARE TWO DIFFERENT EXPANSION INTERFACES. IT IS NECESSARY TO IDENTIFY YOUR MODEL BEFORE INSTALLING THE MAPPER II BOARD. THE EARLY MODEL HAS TWO VERTICAL ROWS (RUNNING FROM FRONT TO BACK) ON THE LEFT SIDE OF THE EXPANSION INTERFACE. THE NEW REVISION HAS THE TWO ROWS OF MEMORY CHIPS RUNNING IN HORIZONTAL ROWS (FROM LEFT TO RIGHT) IN THE BACK CENTER OF THE INTERFACE BOARD. PROCEED TO THE APPROPRIATE SECTION ON THE NEXT PAGE.

DISK ADAPTER INSTALLATION 2

EARLY REVISION: ORIENT THE DISK ADAPTER BOARD SO THE TWO HORIZONTAL ROWS OF IC'S ARE NEAREST THE FRONT (SIDE NEAREST YOU). THE TWO WIRES ATTACHED TO THE BOARD SHOULD BE NEAR THE RIGHT EDGE OF THE INTERFACE CASE. CAREFULLY SET THE ADAPTER BOARD INTO THE FORTY PIN SOCKET. DOUBLE CHECK FOR CORRECT INSERTION OF ALL THE PINS. PRESS THE BOARD DOWN WHILE ROCKING IT FROM FRONT TO BACK. CHECK AGAIN TO SEE THAT IT IS FULLY SEATED. BE ABSOLUTELY CERTAIN THAT THE NOTCH IN THE END OF THE CHIP LIES ON THE LEFT SIDE OF THE BOARD. ALL OF THE NOTCHES IN ALL OF THE CHIPS ON BOTH THE MAPPER II AND THE EXPANSION INTERFACE SHOULD BE ON THE LEFT SIDE. CHECK CAREFULLY FOR BENT UNDER PINS. THE WIRE WITH THE EZHOOK TIP MUST BE FASTENED TO IC (Z27) PIN (14). (Z27) IS THE IC ON THE MAIN BOARD IN THE BACK ROW OF CHIPS DIRECTLY IN FRONT OF THE CENTER DIN CONNECTOR SOCKET. CHECK THAT (Z27) IS A (74LS90). PIN (14) IS THE PIN IN THE BACK LEFT CORNER. PROCEED TO "FINAL INSTRUCTIONS".

NEW REVISION: ALIGN THE DISK ADAPTER BOARD OVER THE EMPTY SOCKET SO THE TWO VERTICAL ROWS OF IC'S ARE ON THE LEFT SIDE AND THE TWO EXTERNAL WIRES ARE ON THE FRONT SIDE. CAREFULLY SET THE BOARD INTO THE SOCKET AND CHECK FOR PROPER ALIGNMENT. PRESS THE BOARD INTO THE SOCKET WHILE ROCKING IT FROM LEFT TO RIGHT. ALL OF THE NOTCHES ON THE MAPPER II SHOULD BE TOWARDS THE BACK. THE WIRE WITH THE EZ HOOK TIP MUST BE FASTENED TO IC (Z22) PIN (14) . (Z22) IS A (74LS90) LOCATED TO THE LEFT OF THE BACK ROW OF MEMORY CHIPS. PIN (14) IS THE PIN IN THE BACK LEFT CORNER.

FINAL INSTRUCTIONS: PLACE THE LOOSE WIRE IN THE CUTOUT FOR THE TRS80 CABLE AND REPLACE THE COVER. THE TWO SHORTEST SCREWS GO IN THE TWO BACK HOLES. THE MEDIUM LENGTH SCREWS GO IN THE TWO MIDDLE HOLES AND THE TWO LONGEST GO IN THE FRONT HOLES. TURN THE UNIT OVER AND REPLACE ALL CABLES. CONNECT THE LOOSE WIRE WITH THE MOLEX CONNECTOR ON THE END TO THE SIMILAR WIRE CONNECTED TO THE MAPPER I BOARD. PROCEED TO THE "SYSTEM TESTING" SECTION FOR FINAL CHECKOUT.

IMPORTANT NOTE: IF YOU HAVE PURCHASED THE MAPPER II ALONE FOR USE WITH NEWDOS/80 ONLY, THE MAPPER II HAS A 40 PIN EDGE CONNECTOR CONNECTED TO IT BY A SINGLE WIRE. THIS EDGE CONNECTOR MUST BE CONNECTED TO THE SCREEN PRINTER PORT. MOUNT THE CONNECTOR WITH THE SINGLE WIRE NEAREST THE FRONT OF THE INTERFACE. IF THE SCREEN PRINTER PORT IS USED FOR ANOTHER PURPOSE, THE WIRE MUST BE SOLDERED DIRECTLY TO THE INTERFACE BOARD. CONTACT OMIKRON FOR DIRECTIONS.

HARDWARE NOTES

DRIVE ADDRESSING: BOTH THE TRS80 MINI DRIVES AND STANDARD 8" DRIVES HAVE FOUR DRIVE SELECT LINES. RADIO SHACK USES THE CABLE TO ADDRESS THE MINI DRIVES BY TERMINATING ONLY ONE OF THE DRIVE SELECT LINES IN EACH CABLE CONNECTOR. THE OMIKRON CABLE USES THE SAME TECHNIQUE TO ADDRESS THE MINI DRIVES HOWEVER THE USER WILL HAVE TO ADDRESS THE 8" DRIVES IF THEY WERE NOT PURCHASED THROUGH OMIKRON. THE PROCEDURE TO ADDRESS A DRIVE WILL VARY DEPENDING ON THE DRIVE MANUFACTURER. SHUGART USES SHORTING PLUGS ON THEIR 8" DRIVES. REFER TO THE SPECIFIC MANUFACTURES DRIVE MANUAL. THE FOUR DRIVE SELECT LINES ARE LABELED 1,2,3,AND 4. THESE CORRESPOND TO CPM DRIVE REFERENCES A,B,C,AND D AND TRSDOS DRIVE REFERENCES 0,1,2, AND 3 RESPECTIVELY.

THE OMIKRON CPM MINI DISK REQUIRES DRIVES 1 AND 2 TO BE MINI DRIVES AND DRIVES 3 AND 4 TO BE 8". THE TRS80 DOS ALSO REQUIRES DRIVE 0 TO BE A MINI DRIVE. IN THIS CONFIGURATION BOTH CPM AND TRS80 DOS CAN BE RUN WITH THE SAME DRIVE ADDRESSING.

THE OMIKRON 8" DISK REQUIRES DRIVES 1 AND 2 TO BE 8" AND DRIVES 3 AND 4 TO BE MINI. THIS ALLOWS THE OPERATOR TO BOOT THE SYSTEM FROM AN 8" DRIVE, PERMITTING OPERATION WITHOUT MINI DRIVES. HOWEVER THE TRS80 DOS CANNOT BE LOADED WITHOUT READDRESSING THE DRIVES.

TERMINATOR RESISTORS: ALL DISK DRIVES ARE SUPPLIED WITH LINE TERMINATING RESISTORS AT THE INTERFACE CONNECTOR. SOME MANUFACTURES USE A REMOVABLE DIP RESISTOR NETWORK WHILE OTHERS USE REMOVABLE SHORTING PLUGS. ONLY ONE DRIVE ON A CABLE SHOULD HAVE THE TERMINATOR CONNECTED. THE OTHER DRIVES MUST HAVE THEIR TERMINATORS REMOVED OR DISCONNECTED. THE BEST LOCATION FOR THE TERMINATOR IS ON THE DRIVE LOCATED FARTHEST FROM THE TRS80. HOWEVER MOST ANY DRIVE WILL PROBABLY WORK AS LONG AS THE OTHER TERMINATORS ARE DISABLED.

LOCATING MINI DRIVE TERMINATORS: RADIO SHACK SELLS TWO DIFFERENT MINI DISK DRIVES. THE ONLY DIFFERENCE BETWEEN THE TWO IS THE PRESENCE OR ABSENCE OF THE TERMINATOR. THE TERMINATOR ON MOST ALL MINI DRIVES IS A (14) PIN PACKAGE THAT RESEMBLES AN INTEGRATED CIRCUIT. IT IS ALWAYS SEATED IN A SOCKET FOR EASY REMOVAL. MOST COMPANIES USE A BECKMAN PART WITH (R150) WRITTEN ON IT. THIS PART IS SIMPLY A GROUP OF (7) RESISTORS CONNECTED TO THE PINS. IT DOESN'T MATTER WHICH WAY IT IS INSTALLED. TO LOCATE THIS PART, REMOVE THE COVER OF THE DRIVE AND LOOK AT THE LOGIC BOARD NEAR THE CABLE EDGE CONNECTOR. IT IS NEXT TO THE ADDRESSING SHUNT.

OMIKRON DISK DRIVES

OMIKRON 8" DRIVE SYSTEMS USE SHUGART MODEL 800 DISK DRIVES. OMIKRON ADDRESSES AND TERMINATES THE DRIVES ACCORDING TO THE CUSTOMERS SYSTEM CONFIGURATION. THE DRIVES ARE ADDRESSED IN THE FOLLOWING MANNER:

SINGLE DRIVE SYSTEMS: THE SINGLE 8" DRIVE SYSTEM HAS THE DRIVE ADDRESSED AS DRIVE "C" OR "2". THE TERMINATOR JUMPERS ARE INSTALLED.

DUAL DRIVE SYSTEMS: IF THE CUSTOMER HAS ORDERED THE CP/M ON AN 8" DISKETTE, THE 8" DRIVES ARE ADDRESSED AS "A" AND "B". THE TERMINATOR JUMPERS ARE INSTALLED ON THE "B" DRIVE. IF THE CP/M HAS BEEN SUPPLIED ON A 5" DISKETTE OR THE SYSTEM HAS BEEN ORDERED FOR NEWDOS/80, THE DRIVES ARE SUPPLIED ADDRESSED AS "C" AND "D" OR "2" AND "3". THE TERMINATOR JUMPERS ARE INSTALLED ON THE "D" OR "3" DRIVE.

THE SHUGART OEM DRIVE MANUAL IS SUPPLIED WITH THE OMIKRON DRIVE UNIT. THIS MANUAL DESCRIBES THE VARIOUS DRIVE JUMPER OPTIONS IN GREAT DETAIL. THERE IS A TABLE ON PAGE 25 THAT LISTS THE WAY THE DRIVE IS ORIGINALLY JUMPERED BY THE FACTORY. WITH THE EXCEPTION OF THE DRIVE ADDRESSING AND TERMINATION, THE OMIKRON MAPPER II IS DESIGNED TO RUN WITH THE DRIVE JUMPERED IN THE "STANDARD" FACTORY MODE. IF YOU HAVE OBTAINED THE DRIVE FROM ANOTHER SOURCE CHECK THAT THE JUMPERS ARE INSTALLED ACCORDING TO THE TABLE.

DRIVE PLUGS: THE SHUGART DRIVE HAS SETS OF PINS PROTRUDING FROM THE PRINTED CIRCUIT BOARD. THESE PINS ARE DESIGNED TO BE JUMPERED TOGETHER WITH SMALL JUMPER PLUGS. MOST OF THE CONFIGURATION OPTIONS USE THESE PLUGS TO AVOID CUTTING OR SOLDERING TRACES.

DRIVE ADDRESSING: THE SHUGART DRIVE HAS FOUR DRIVE SELECT LINES. THESE ARE LABELED DS1, DS2, DS3, AND DS4. ONLY ONE OF THESE LINES SHOULD BE PLUGGED ON A GIVEN DRIVE. WHEN RUNNING UNDER CP/M THE DRIVES ARE REFERENCED AS "A", "B", "C" AND "D". DRIVE "A" IS ADDRESSED BY PLUGGING DS1, "B" WITH DS2, "C" WITH DS3, AND "D" WITH DS4. WHEN RUNNING UNDER NEWDOS/80 THE DRIVES ARE REFERENCED AS "0", "1", "2", AND "3". DRIVE "0" IS ADDRESSED BY PLUGGING DS1, "1" WITH DS2, "2" WITH DS3 AND "3" WITH DS4.

DRIVE TERMINATING: THE SHUGART 8" DRIVE HAS FOUR TERMINATOR JUMPERS. THESE ARE TREATED AS A GROUP; ALL FOUR SHOULD BE JUMPERED OR OPEN. THE FOUR LINES ARE LABELED T3, T4, T5, AND T6. ONLY THE LAST DRIVE ON THE CABLE SHOULD BE TERMINATED. THE OTHER DRIVES SHOULD HAVE THE TERMINATOR RESISTOR REMOVED OR THE PLUGS REMOVED.

MISCELLANEOUS COMMENTS 1

RESERVED RAM: THE CPM OPERATING SYSTEM RESERVES THE RAM BETWEEN 40H AND 50H FOR USE BY THE CBIOS. THE CPM SYSTEM ALTERATION GUIDE CLEARLY STATES THAT THIS AREA IS RESERVED HOWEVER SOME OF THE SOFTWARE THAT CLAIMS TO BE CPM COMPATABLE USES THIS AREA FOR PROGRAM STORAGE. THESE PROGRAMS WILL NOT RUN WITH THE MAPPER BECAUSE THE KEYBOARD DRIVER ROUTINE USES THIS AREA EVERY TIME THE KEYBOARD IS SCANNED. EARLY VERSIONS OF MICROSOFT BASIC WILL NOT RUN HOWEVER THE PROBLEM WAS CORRECTED WITH VERSION 4.51. IF YOU FIND A PROGRAM THAT RUNS UNDER CPM BUT WHICH WILL NOT RUN WITH THE MAPPER CHECK WITH THE AUTHOR TO SEE IF THE PROGRAM USES THIS AREA OF RAM.

NON-SHUGART DRIVES: SHUGART DRIVES ARE SET UP TO LOAD THE HEAD WHEN THE DRIVE IS SELECTED. IF YOU ARE NOT USING SHUGART DRIVES MAKE SURE THE DRIVE IS JUMPERED TO LOAD THE HEAD WHEN SELECTED. THE TRS80 DOES NOT SUPPORT THE HEADLOAD SIGNAL.

MEMORY MAPPING: WHEN OPERATING UNDER CP/M WITH THE OMIKRON MAPPER, THE MEMORY MAP OF THE TRS80 IS DIFFERENT FROM THAT GIVEN IN THE RADIO SHACK MANUALS. ALL OF THE MEMORY MAPPED PORTS ARE ADDRESSED AT NEW ADDRESSES. THIS INCLUDES THE KEYBOARD, THE VIDEO MEMORY, THE CASETTE PORTS, AND THE DISK CONTROLLER PORTS. THE NEW ADDRESSES ARE CALCULATED BY ADDING (0C000H) TO THE OLD ADDRESSES. FOR EXAMPLE, THE SCREEN MEMORY IS NOW MAPPED AT (0FC00H) TO (0FFFFH). PROGRAMS THAT WRITE TO THE SCREEN DIRECTLY SHOULD BE MODIFIED ACCORDINGLY.

CABLE INSTALLATION: THE RED STRIPE OF THE CABLE IS USED TO POLARIZE THE CABLE CONNECTORS. THIS STRIPE IS CONNECTED TO PIN ONE ON EACH OF THE MINI DRIVE CONNECTORS. THE SIDE WITHOUT THE STRIPE IS CONNECTED TO PIN (50) OF EACH EIGHT INCH DRIVE CONNECTOR. THE CABLE IS INSTALLED WITH THE RED STRIPE CLOSEST TO THE OUTSIDE EDGE ON BOTH THE EXPANSION INTERFACE AND THE MINI DRIVES. EIGHT INCH DRIVES VARY BUT THE PIN NUMBERS ARE USUALLY WRITTED ON THE DRIVE NEAR THE CONNECTOR. MANY DRIVES HAVE A SLOT CUT IN THE EDGE CONNECTOR. THE RED STRIPE SHOULD BE ON THE SAME SIDE AS THE SLOT. IF ALL OF THE LIGHTS ON THE DRIVES COME ON WHEN A CONNECTOR IS CONNECTED, THAT CONNECTOR IS PROBABLY INSTALLED BACKWARDS.

MISCELLANEOUS COMMENTS 2

DISK INITIALIZATION: THE OMIKRON MAPPER AND THE TRS80 USE SOFT SECTORED DISKS. SOFT SECTORING REQUIRES THAT ADDRESS HEADERS BE WRITTEN ON A DISK BEFORE IT CAN BE USED. MOST 8" DISKS ARE SUPPLIED CORRECTLY FORMATTED. HOWEVER THE OMIKRON MAPPER MINI DISK FORMAT IS DIFFERENT FROM THAT OF THE TRS80 FORMAT. ALL MINI DISKS USED WITH CPM MUST BE INITIALIZED BEFORE USE WITH THE MFORMAT.COM PROGRAM.

SYSTEM DUPLICATION: THE FIRST TWO TRACKS OF AN 8" CPM DISK AND THE FIRST THREE TRACKS OF A CPM MINI DISK CONTAIN THE CPM OPERATING SYSTEM. ALL DISKS USED WITH CPM ON DRIVE "A" SHOULD HAVE THIS CPM SYSTEM WRITTEN ON THEM. USE THE LSYSGEN.COM PROGRAM FOR 8" DRIVES AND THE MSYSGEN.COM PROGRAM FOR MINI DRIVES TO COPY THE CPM SYSTEM FROM THE DISTRIBUTION DISK TO YOUR WORKING DISKS.

WRITEPROTECT: MOST DISK DRIVES ARE EQUIPPED WITH A PHOTOCELL SENSOR TO INDICATE THE PRESENCE OR ABSENCE OF A WRITEPROTECT TAB ON THE DISKETTE. UNFORTUNATELY THE SPECIFICATION FOR THE MINI DRIVE IS OPPOSITE THAT OF THE 8" DRIVE. IN THE CASE OF THE MINI DRIVE YOU WILL HAVE TO REMOVE THE TAB TO WRITE ON THE DISK. THE 8" DRIVES REQUIRE YOU TO COVER THE NOTCH, IF PRESENT, TO WRITE ON THE DISKS. IF YOU ATTEMPT TO WRITE ON A PROTECTED DISK THE MESSAGE "PROTECTED TYPE C" WILL APPEAR ON YOUR SCREEN. AT THIS POINT YOU MAY UN-PROTECT THE DISK AND TYPE "C" TO CONTINUE OR (CONTROL C) TO REBOOT THE SYSTEM IF THE WRITE WAS UNINTENTIONAL. ALL OMIKRON DISKS ARE SHIPPED WRITE PROTECTED TO PREVENT ACCIDENTAL ERRORS FROM DESTROYING THE FILES. IT IS RECOMMENDED THAT YOU NEVER WRITE ON THIS DISK. BACK IT UP WITH THE PIP AND SYSGEN COMMANDS IF YOU HAVE DIFFERENT SIZE DRIVES OR "OMCOPY ALL" IF YOU HAVE TWO SIMILAR SIZE DRIVES. WHEN READING A LARGE FILE UNDER CPM THE WRITE PROTECT MESSAGE MAY APPEAR. THIS HAPPENS WHEN CPM OPENS A NEW EXTENT. THE ONLY SOLUTION IS TO UNPROTECT THE DISK TEMPORARILY.

NOT READY: EVERY TIME CPM SELECTS A NEW DRIVE THE DRIVE IS TESTED FOR A PROPERLY INSERTED DISK. IF THE DISK IS NOT SPINNING THE MESSAGE "NOT READY TYPE C" WILL APPEAR. AT THIS POINT YOU MAY CORRECTLY INSERT THE DISK AND TYPE "C" TO CONTINUE OR TYPE (CONTROL C) TO REBOOT THE SYSTEM IF A NONEXISTANT DRIVE WAS SELECTED.

KEYBOARD DRIVER

OMIKRON HAS DEVELOPED A SOPHISTICATED KEYBOARD ROUTINE TO MAXIMIZE THE UTILITY OF THE CP/M PACKAGE. IT FEATURES FULL DEBOUNCING, A SINGLE KEY "CONTROL" KEY, A CAPS-LOC FUNCTION, AND THE ABILITY TO PRODUCE THE FULL 128 CHARACTERS IN THE ASCII SET. SOME OF THE CONTROL CHARACTERS USED FOR CP/M HAVE BEEN ASSIGNED TO SEPARATE KEYS FOR ONE STROKE OPERATION. IN ADDITION, THE SHIFT KEY FUNCTIONS CORRECTLY, PRODUCING LOWER CASE CHARACTERS UNLESS THE SHIFT KEY IS PRESSED.

THE DOWN ARROW FUNCTIONS AS THE CONTROL KEY. WITH THIS KEY DEPRESSED, CONTROL CHARACTERS A THROUGH Z ARE FORMED BY PRESSING THE APPROPRIATE KEY. THE CONTROL KEY IS ALSO USED WITH NUMBERS 1 THROUGH 9 TO GENERATE CHARACTERS NOT NORMALLY AVAILABLE WITH THE TRS80 KEYBOARD. THESE CHARACTERS ARE FOUND BY REFERENCING THE TABLE.

CONTROL (0) HAS A SPECIAL FUNCTION AND DOESN'T GENERATE A CHARACTER: IT FUNCTIONS AS A CAPS-LOC KEY. THE KEYBOARD IS NORMALLY LOCKED IN THE UPPER CASE MODE. THE SETUP PROGRAM IS USED TO ENABLE THE LOWER CASE FUNCTION. WITH THE LOWER CASE ENABLED, A CONTROL (0) WILL TOGGLE A BYTE THAT LOCKS AND UNLOCKS THE KEYBOARD IN THE UPPER CASE ONLY MODE. THE SETUP PROGRAM ALSO CAN BE USED TO DISABLE THE LOWER CASE FUNCTION ONCE IT HAS BEEN ENABLED. WHENEVER THE SYSTEM IS REBOOTED, THE DRIVER DEFAULTS TO UPPER CASE ONLY.

SPECIAL KEYBOARD CHARACTERS

CONTROL KEY THE DOWN ARROW FUNCTIONS AS THE CONTROL KEY. CONTROL CHARACTERS ARE FORMED BY FIRST PRESSING THE DOWN ARROW KEY AND HOLDING IT DOWN WHILE THE APPROPRIATE LETTER OR NUMBER IS PRESSED.

TABS KEY A TAB KEY GENERATES A (CONTROL I) OR (09H). THE RIGHT ARROW KEY PRODUCES A (CONTROL I).

ESCAPE KEY AN ESCAPE KEY GENERATES A (1BH). A SHIFT-(BREAK) WILL PRODUCE AN ESCAPE CHARACTER.

DELETE KEY THE DELETE KEY GENERATES A (7FH). THE LEFT ARROW KEY PRODUCES THE DELETE CHARACTER.

LINEFEED KEY A LINEFEED KEY GENERATES A (0AH) OR (CONTROL J). THE SHIFT-(ENTER) KEY WILL PRODUCE A LINEFEED CHARACTER.

SPECIAL KEYBOARD CHARACTERS

BACKSPACE KEY A BACKSPACE KEY GENERATES A (08H) OR (CONTROL H). CP/M 2.0 USES THE BACKSPACE CHARACTER INSTEAD OF THE DELETE CHARACTER. THE BREAK KEY PRODUCES THE BACKSPACE CHARACTER.

BRACKETS KEYS THE LEFT BRACKET IS A (5BH). THE RIGHT BRACKET IS A (5DH). THE PIP UTILITY USES THESE CHARACTERS FOR SPECIAL FUNCTIONS. THE SHIFT-(RIGHT ARROW) KEY PRODUCES A RIGHT BRACKET. THE SHIFT-(LEFT ARROW) KEY PRODUCES A LEFT BRACKET.

UP ARROW KEY AN UP ARROW KEY GENERATES A (5EH). MOST BASIC PROGRAMS USE THIS CHARACTER AS AN EXPONENT SYMBOL. THE SHIFT-(CLEAR) KEY WILL PRODUCE AN UP ARROW CHARACTER.

BACKSLASH KEY THE BACKSLASH CHARACTER IS A (5CH). MOST BASIC LANGUAGES USE THIS CHARACTER. THE SHIFT-(UP ARROW) KEY WILL PRODUCE THE BACKSLASH CHARACTER. THE SCREEN WILL DISPLAY A DOWN ARROW.

CONTROL S CP/M USES A (CONTROL S) TO START AND STOP THE TYPING OF A FILE. THE CLEAR KEY PRODUCES A (CONTROL S) WITH ONE KEY.

CONTROL Z THE CP/M EDITOR USES A (CONTROL Z) FOR TERMINATING COMMANDS. THE UP ARROW KEY PRODUCES A (CONTROL Z) WITH ONE KEY.

CAPS-LOC KEY THE CAPS-LOC FUNCTION IS PRODUCED WITH A (CONTROL 0).

CONTROL NUMBERS TABLE

(5FH) = (CONTROL 1)	(1CH) = (CONTROL 2)
(1DH) = (CONTROL 3)	(1EH) = (CONTROL 4)
(1FH) = (CONTROL 5)	(7BH) = (CONTROL 6)
(7CH) = (CONTROL 7)	(7DH) = (CONTROL 8)
(7EH) = (CONTROL 9)	CAPS-LOC = (CONTROL 0)

PRINTER DRIVERS

THE OMIKRON CP/M SOFTWARE CONTAINS DRIVERS FOR BOTH THE PARALLEL PORT AND THE RS232C SERIAL PORT. WHEN FIRST BOOTED UP OR RESET THE PARALLEL DRIVER IS ENABLED. THE DRIVER IS SET UP FOR PRINTERS (SUCH AS THE CENTRONICS 779) THAT EXECUTE BOTH A CARRIAGE RETURN AND LINEFEED WHEN SENT A CARRIAGE RETURN CHARACTER (0DH). THE SETUP PROGRAM ALLOWS THE OPERATOR TO RECONFIGURE THE DRIVER FOR PRINTERS THAT REQUIRE A LINEFEED (0AH) CHARACTER AFTER A CARRIAGE RETURN.

THE PARALLEL DRIVER ALSO CONTAINS A FORM FEED ROUTINE THAT INTERCEPTS THE FORM FEED CHARACTER (0CH) AND OUTPUTS LINEFEEDS UNTIL THE TOP OF THE NEXT PAGE. PRINTERS LIKE THE CENTRONICS MODEL 779 CAN MAKE USE OF THIS FEATURE BECAUSE THEY HAVE NO INTERNAL LOGIC TO PERFORM THIS FUNCTION. THE SETUP PROGRAM MUST BE USED TO ENABLE THIS FUNCTION. THE DRIVER IS SET UP FOR 66 LINES PER PAGE. THIS CAN BE ALTERED BY CHANGING TWO BYTES IN THE CBIOS TO THE DESIRED VALUE. THESE BYTES ARE CURRENTLY LOCATED AT THE BASE OF THE CBIOS (0BE00H FOR A 48K SYSTEM) PLUS (1C7H AND 1C8H). FUTURE VERSIONS WILL HAVE THESE BYTES LOCATED AT THE BASE OF THE CBIOS PLUS (30H AND 31H). THESE BYTES ARE CURRENTLY EQUAL TO (42H OR 66D).

THE SERIAL DRIVER IS ENABLED WITH THE SERIAL PROGRAM. THE SERIAL PROGRAM READS THE SENSE SWITCHES ON THE RADIO SHACK RS232C BOARD AND INITIALIZES THE UART TO THE VALUES LISTED IN THE RADIO SHACK RS232 MANUAL. ALL OF THE UART PARAMETERS ARE PRINTED ON THE SCREEN FOR VERIFICATION.

THE SERIAL DRIVER OVERLAYS THE PARALLEL DRIVER IN MEMORY. BECAUSE OF SPACE LIMITATIONS THE FORM FEED FEATURE IS NOT SUPPORTED IN THE SERIAL MODE. THE LINEFEED OPTION CAN STILL BE ENABLED WITH THE SETUP PROGRAM. THE SERIAL DRIVER MONITORS THE CTS INPUT LINE FOR A PRINTER READY SIGNAL. THIS SIGNAL IS FOUND ON PIN 5 OF THE RS232 CABLE. THE ONLY WAY TO SWITCH BACK TO THE PARALLEL PORT IS TO REBOOT THE SYSTEM.

THE PARALLEL DRIVER IS PLACED AT THE VERY END OF THE CBIOS SO PATCHES CAN BE DONE IN MEMORY IF REQUIRED. THE SERIAL DRIVER CHECKS TO SEE IF THE PARALLEL DRIVER HAS BEEN MODIFIED. IF THE PARALLEL DRIVER HAS BEEN MODIFIED (EITHER BY HAVING PREVIOUSLY CALLED THE SERIAL PROGRAM OR BECAUSE OF PROGRAM MODIFICATIONS) THE OPERATOR HAS THE OPTION OF PREVENTING THE SERIAL DRIVER FROM OVERLAYING THE PARALLEL DRIVER. THE UART WILL STILL BE INITIALIZED CORRECTLY. IF THE OPERATOR CHOOSES TO MODIFY THE PRINTER DRIVER IN MEMORY INSTEAD OF REASSEMBLING THE CBIOS, THE CBIOS CAN BE SAVED IN A COM FILE BY USING THE SYSGEN PROGRAM AND TYPING "SAVE 32 CPM.COM". THE CBIOS CAN BE FOUND IN THIS COM FILE STARTING AT (1E80H). CP/M VERSION 2.0 HAS A MUCH LARGER CBIOS AND THERE IS NO NEED TO HAVE THE SERIAL DRIVER OVERLAY THE PARALLEL DRIVER. PATCHES ARE STILL POSSIBLE . USE THE SYSGEN PROGRAM AND "SAVE 34 CPM.COM" FOR CP/M 2.0. PLEASE NOTE THAT THE PREVIOUSLY MENTIONED ADDRESSES WILL HAVE TO BE MODIFIED FOR CP/M 2.0.

VIDEO DRIVER

THE OMIKRON VIDEO DRIVER IS DESIGNED TO SUPPORT PROGRAMS WRITTEN FOR THE CP/M OPERATING SYSTEM. THIS INCLUDES PROGRAMS REQUIRING LOWER CASE DISPLAY, GRAPHICS DISPLAY, CURSOR POSITIONING COMMANDS, AND AN ADDRESSABLE CURSOR MODE.

THERE IS A ROUTINE IN THE VIDEO DRIVER THAT CONVERTS LOWER CASE CHARACTERS TO UPPER CASE. THIS ALLOWS THE OPERATOR TO READ FILES WITH LOWER CASE CHARACTERS IN THEM. IF THE TRS80 HAS A LOWER CASE MODIFICATION INSTALLED, THE LOWER CASE CONVERSION CAN BE DISABLED WITH THE SETUP PROGRAM. WITH THE CONVERSION DISABLED, THE ENTIRE ASCII SET MAY BE DISPLAYED.

THE DRIVER ALSO ALLOWS THE DISPLAY OF THE TRS80 GRAPHICS SET. WHEN THE DISPLAY IS CALLED WITH A CHARACTER BETWEEN (80H) AND (0C0H), THE STANDARD GRAPHICS CHARACTER WILL BE DISPLAYED. WHEN THE DRIVER IS CALLED WITH A CHARACTER BETWEEN (0C0H) AND (0FFH), THE CHARACTER IS CONVERTED TO A VALUE BETWEEN (0) AND (40H). IF THE LOWER CASE "MOD" IS ENABLED, THE DISPLAY WILL CONTAIN THE CONTROL CHARACTERS BETWEEN (0) AND (20H) WHICH ARE NORMALLY USED FOR COMMANDING THE VIDEO ROUTINE.

THE SETUP PROGRAM ALLOWS YOU TO ELIMINATE THE GRAPHICS SET AND CONVERT ALL GRAPHICS CHARACTERS TO THEIR ASCII EQUIVALENTS. THIS IS USEFUL WITH PROGRAMS THAT SET BIT (7) HIGH TO FLAG SPECIAL FUNCTIONS (SUCH AS LINE NUMBERS).

THE VIDEO DRIVER ALSO CONTAINS A ROUTINE TO ADDRESS THE CURSOR. THIS ASSURES COMPATABILITY WITH A WIDE RANGE OF BUSINESS APPLICATION AND WORD PROCESSOR PACKAGES WHICH WOULD NOT NORMALLY RUN ON THE TRS80. THE OMIKRON DRIVER IS SET UP TO EMULATE THE SOROC "IQ120" TERMINAL. THE CURSOR IS ADDRESSED WITH A FOUR CHARACTER STRING. THE FIRST CHARACTER IS AN ASCII ESCAPE (1BH). THE SECOND IS AN ASCII "=" OR (3DH). THE THIRD CHARACTER DETERMINES THE LINE NUMBER, NUMBERED (0) THROUGH (15). THE CHARACTER REQUIRED TO ADDRESS A GIVEN LINE IS CALCULATED BY ADDING (20H) OR (32D) TO THE DESIRED LINE NUMBER. THE FOURTH CHARACTER DETERMINES THE COLUMN NUMBER, NUMBERED (0) THROUGH (63). IT IS CALCULATED IN THE SAME MANNER AS THE LINE NUMBER. THE SEQUENCE "ESCAPE, EQUALS, SPACE, SPACE" WILL HOME THE CURSOR. NOTE THAT THE TRS80 HAS A SCREEN SIZE OF (16) ROWS BY (64) COLUMNS. THE ROUTINE TO CLEAR THE SCREEN AND HOME THE CURSOR IS CALLED WITH A TWO CHARACTER SEQUENCE. THE FIRST IS AN ESCAPE. THE SECOND IS AN ASCII (*) OR (2AH).

THIS CURSOR ADDRESSING ROUTINE HAS BEEN USED WITH THE OSBORNE ACCOUNTING PACKAGES AS WELL AS WITH WORDSTAR AND WORDMASTER BY MICROPRO INTERNATIONAL. IT IS ALSO VERY USEFUL FOR CONVERTING PRINT@ STATEMENTS TO EQUIVALENT PRINT STATEMENTS.

IN ADDITION TO THE PREVIOUSLY MENTIONED FEATURES THE VIDEO DRIVER SUPPORTS MANY OF THE TRADITIONAL DISPLAY FUNCTIONS. THESE ARE LISTED IN THE FOLLOWING TABLE.

STANDARD VIDEO FUNCTIONS

(08H) = BACKSPACE AND DELETE CHARACTER

(0AH) = LINEFEED TO THE NEXT LINE

(0DH) = RETURN CURSOR TO START OF LINE

(18H) = MOVE CURSOR LEFT: RESTORE OLD CHARACTER

(19H) = MOVE CURSOR RIGHT: RESTORE OLD CHARACTER

(1AH) = MOVE CURSOR DOWN: RESTORE OLD CHARACTER

(1BH) = MOVE CURSOR UP: RESTORE OLD CHARACTER

(1CH) = HOME CURSOR AND CLEAR SCREEN

SOFTWARE COMMENTS 1

MOVCPM.COM: THIS FILE IS USED IN CREATING A CPM SYSTEM FOR A SPECIFIED MEMORY SIZE. IT IS ALSO NECESSARY TO REASSEMBLE THE CBIOS AND BOOT PROGRAMS FOR THE NEW MEMORY SIZE. THE PROCEDURE IS DESCRIBED IN THE CPM DOCUMENTATION BUT OMIKRON RECOMMENDS THAT ONLY PEOPLE EXPERIENCED IN ASSEMBLY LANGUAGE PROGRAMMING ATTEMPT IT. OMIKRON CAN SUPPLY CPM DISKS FOR ALL THREE STANDARD MEMORY SIZES.

PIP.COM: THE USE OF THIS COMMAND IS DESCRIBED IN THE CPM DOCUMENTATION. HOWEVER THE FOLLOWING COMMENTS SHOULD BE REMEMBERED. THIS COMMAND IS USED TO TRANSFER FILES BETWEEN DISKS. WHEN USED WITH THE OMIKRON MAPPER, FILES CAN BE TRANSFERRED BETWEEN MINI AND 8" DISKS WITHOUT SYNTAX MODIFICATION . ALL REQUIRED PARAMETER CHANGES ARE HANDLED AUTOMATICALLY. PIP IS ALSO USED TO COPY DISKS AND BACK UP FILES. THE COMMAND PIP C:=A:*.*[V] WILL TRANSFER ALL FILES FROM DRIVE A TO DRIVE C AND VERIFY ALL TRANSFERS. WHEN A FILE IS TOO LARGE TO FIT IN THE SPACE LEFT ON A DISK THE MESSAGE "DISK WRITE ERROR" WILL WARN THE OPERATOR. WHEN TRANSFERRING A LARGE FILE THE MESSAGE "PROTECTED" MAY APPEAR. THIS IS NORMAL UNDER CPM. UN-WRITEPROTECT THE SOURCE DISK AND TYPE C TO PROCEED.

STAT.COM: THE STAT COMMAND IS USED TO INDICATE DISK STORAGE USEAGE. IT WILL AUTOMATICALLY COMPENSATE FOR THE DIFFERENT DRIVE SIZES.

DDT.COM: THE DDT FILE SUPPLIED ON THE OMIKRON DISK IS DIFFERENT FROM THE STANDARD DISTRIBUTION DDT PROGRAM. THE STANDARD DDT VERSION WILL ENABLE THE INTERRUPTS AND CAUSE A SYSTEM CRASH.

LSYSGEN.COM: THIS IS THE SYSGEN PROGRAM USED FOR 8" DISKS. IT WILL NOT WORK WITH MINI DISKS.

MSYSGEN.COM: THIS IS THE SYSGEN PROGRAM USED FOR MINI DISKS. IT WILL NOT WORK WITH 8" DISKS.

MFORMAT.COM: THIS IS THE FORMAT PROGRAM USED TO FORMAT MINI DISKS. THE COMMAND MFORMAT @ WHERE @ = A,B,C, OR D WILL FORMAT THE DISK ON THE SELECTED DRIVE. OMIKRON USES A SOFT SECTOR FORMAT WITH EIGHTEEN 128 BYTE SECTORS PER TRACK. THE OPERATOR CAN CHOOSE EITHER 35 OR 40 TRACKS TO BE FORMATTED HOWEVER THE EXTRA STORAGE A 40 TRACK DRIVE PROVIDES WILL NOT BE USED WITH THE STANDARD SOFTWARE. A SPECIAL DISK MUST BE ORDERED FROM OMIKRON IF THIS EXTRA SPACE IS TO BE UTILIZED.

SOFTWARE COMMENTS 2

LFORMAT.COM: THIS IS THE FORMAT PROGRAM FOR 8" DISKS. THE OMIKRON FORMAT IS COMPATABLE WITH THE IBM 3740 SOFT SECTOR SPECIFICATION. THIS HAS TWENTY SIX 128 BYTE SECTORS PER TRACK. THERE ARE 77 TRACKS PER DISK. MOST 8" SOFT SECTORED DISKS ARE SUPPLIED FROM THE MANUFACTURER WITH THIS FORMAT WRITTEN ON THEM. A SPECIAL FORMAT PROGRAM THAT ALLOWS SKEW FACTORS OTHER THAN SIX IS AVAILABLE FOR SPECIAL APPLICATIONS.

L48BIOS.ASM: THIS IS THE SOURCE FILE FOR THE CBIOS FOR A 48K SYSTEM ON AN 8" DISK. THIS PROGRAM IS SUPPLIED SO THAT THE USER CAN MODIFY IT OR MAKE ADDITIONS IF NECESSARY. THIS PROGRAM CAN ALSO BE USED AS THE SOURCE FILE FOR 16K AND 32K MINI AND LARGE DISKETTE SYSTEMS. WHEN ASSEMBLED WITH THE CPM MACRO ASSEMBLER THE USER CAN CHANGE MEMSIZ AND MINISWITCH TO PRODUCE THE DESIRED SYSTEM CONFIGURATION. THE LIST ROUTINE IS PLACED AT THE VERY END SO THAT THE KMOWLEDGABLE USER CAN MAKE AN ASSEMBLY LANGUAGE MEMORY PATCH.

L48BOOT.ASM: THIS PROGRAM IS THE SOURCE FILE FOR A 48K SYSTEM SUPPLIED ON AN 8" DISK. IT IS ALSO WRITTEN WITH CONDITIONAL ASSEMBLY SWITCHES SO IT CAN BE USED FOR ALL OTHER CONFIGURATIONS AS MENTIONED ABOVE.

OMCOPY.COM: THIS PROGRAM IS USED TO MAKE BACKUP COPIES OF YOUR CPM DISKS. IT COPIES THE DISK BY TRACK AND SECTOR SO IT WILL RUN MUCH MORE QUICKLY THAN THE PIP PROGRAM, WHICH COPIES THE DISK FILE BY FILE. IT WILL NOT COPY A MINI DISK TO AN 8" OR AN 8" TO A MINI. THE PIP PROGRAM MUST BE USED FOR THIS. THE COPY PROGRAM REREADS EVERY SECTOR TO INSURE THAT THE DATA HAS BEEN TRANSFERED PROPERLY. THERE ARE THREE DIFFERENT COMMANDS FOR THIS PROGRAM. THE COMMAND "OMCOPY ALL" WILL COPY THE ENTIRE DISK. THE COMMAND "OMCOPY DATA" WILL COPY THE ENTIRE DISK EXCEPT FOR THE FIRST TWO TRACKS OF AN 8" DISK OR THE FIRST THREE TRACKS OF A MINI DISK. THE COMMAND "OMCOPY SYSTEM" WILL COPY ONLY THE FIRST TWO TRACKS OF AN 8" DISK OR THE FIRST 3 TRACKS OF A MINI.

MEMTEST.COM: THIS PROGRAM IS AN EXTENSIVE MEMORY TEST. IT CAN BE USED TO VERIFY THAT YOUR TRS80 IS RUNNING PROPERLY WITH THE MAPPER. THE PROGRAM IS SELF DOCUMENTING. IT WILL NOT RUN IN A 16K CPM SYSTEM BECAUSE THE PROGRAM WOULD STEP ON ITSELF.

SOFTWARE COMMENTS 3

DSKTEST.COM: THIS PROGRAM IS USED TO EVALUATE THE CONDITION OF YOUR DRIVES. IT WILL TEST BOTH MINI AND 8" DRIVES. IT IS SELF DOCUMENTING HOWEVER A NEWLY FORMATTED DISK SHOULD BE USED IN THE OBJECT DRIVE. THE PROGRAM WRITES A SERIES OF PATTERNS ON THE DISKETTE AND THEN READS IT BACK WHILE COUNTING SECTOR RETRIES. SECTOR RETRIES OCCUR WHEN A SECTOR IS READ INCORRECTLY AT FIRST BUT READS CORRECTLY ON A SUCCESSIVE TRY. THE MOST COMMON CAUSES FOR THIS ARE: AN OUT OF ALIGNMENT DRIVE, A WORN OR DIRTY HEAD, A WORN HEADLOAD PRESSURE PAD, OR A WORN OR DEFECTIVE DISKETTE. A NEW DRIVE WITH A NEW DISKETTE SHOULD GET LESS THAN 20 RETRIES PER COMPLETE PASS. IF MORE THAN FIFTY RETRIES OCCUR THE SYSTEM SHOULD BE CLEANED AND ADJUSTED.

XDIR.COM: THIS PROGRAM READS THE DIRECTORY AND PRINTS IT OUT THREE COLUMNS ACROSS WITH THE NUMBER OF SECTORS USED IN EACH FILE. IT IS MUCH MORE CONVENIENT THAN THE CPM "DIR" COMMAND.

SERIAL.COM: THIS PROGRAM IS DESCRIBED IN MORE DETAIL ON THE PAGE TITLED "PRINTER DRIVERS". WHEN THE WORD "SERIAL" IS TYPED THE SERIAL DRIVER IN THE CBIOS IS ENABLED AND THE UART IS INITIALIZED ACCORDING TO THE SENSE SWITCHES AS DESCRIBED IN THE RADIO SHACK RS232 MANUAL. ALL OF THE PARAMETERS ARE PRINTED ON THE SCREEN.

SETUP.COM: THIS PROGRAM IS USED TO CUSTOMIZE THE OMIKRON SOFTWARE FOR SPECIFIC APPLICATIONS. THE PROGRAM ASKS THE OPERATOR SEVEN QUESTIONS. AFTER ANSWERING THESE SEVEN QUESTIONS, THE USER HAS THE OPTION OF SAVING THEM FOR FUTURE CONVENIENCE. THE COMMAND "SETUP D" WILL EXECUTE THE SETUP PROGRAM AND INITIALIZE THE SYSTEM ACCORDING TO THE STORED DEFAULT VALUES. THIS SAVES THE USER FROM STEPPING THROUGH THE SETUP PROGRAM EACH TIME THE SYSTEM IS BOOTED. THE SEVEN QUESTIONS ARE DESCRIBED IN THE FOLLOWING PARAGRAPHS.

1) THE CPM OPERATING SYSTEM WAS WRITTEN FOR USE WITH TERMINALS THAT COULD NOT BACKSPACE. WHEN THE DELETE KEY IS PRESSED THE DELETED CHARACTER IS DUPLICATED ON THE DISPLAY. OMIKRON HAS ADDED A ROUTINE THAT SENDS A BACKSPACE TO THE SCREEN WHEN A CHARACTER IS DELETED. IN MOST CASES THE DISPLAY WILL READ CORRECTLY. HOWEVER SOME CPM COMPATABLE SOFTWARE PACKAGES (SUCH AS MICROSOFT MBASIC AND WORDSTAR BY MICROPRO) HAVE THEIR OWN ROUTINES FOR CORRECTING THE DISPLAY. THESE PROGRAMS WILL PRODUCE A CONFUSING DISPLAY BECAUSE OF INTERACTION WITH THE OMIKRON ROUTINE. THE OPERATOR CAN DISABLE THE OMIKRON ROUTINE WITH THIS PROGRAM.

SOFTWARE COMMENTS 4

2) SOME PRINTERS WILL LINEFEED AUTOMATICALLY WHEN SENT A CARRIAGE RETURN. OTHERS REQUIRE A SEPARATE LINEFEED CHARACTER. CPM NORMALLY INSERTS A LINEFEED AFTER A CARRIAGE RETURN. THIS WILL CAUSE A PRINTER THAT LINEFEEDS AUTOMATICALLY TO DOUBLE SPACE. OMIKRON HAS ADDED A ROUTINE THAT DELETES LINEFEEDS THAT FOLLOW CARRIAGE RETURNS. THIS ALLOWS MOST PRINTERS TO PRINT CORRECTLY. HOWEVER SOME PRINTERS REQUIRE THE LINEFEEDS AND SOME SOFTWARE PACKAGES REQUIRE A PRINTER THAT DOESN'T AUTO LINEFEED. THE OPERATOR CAN DISABLE THE LINEFEED SUPRESSION ROUTINE WITH THIS PROGRAM.

3) THE TRS80 WILL NOT DISPLAY LOWER CASE CHARACTERS UNLESS MODIFIED. THE VIDEO DRIVER SOFTWARE CONTAINS A ROUTINE THAT CONVERTS LOWER CASE LETTERS TO UPPER CASE SO THE MESSAGE CAN STILL BE RECOGNIZED.

THE OMIKRON KEYBOARD DRIVER LOCKS THE KEYBOARD IN UPPER CASE. OMIKRON RECOMMENDS THAT YOU USE ONLY UPPER CASE UNLESS THE TRS80 IS MODIFIED TO DISPLAY LOWER CASE. THIS WILL PREVENT LOWER CASE CHARACTERS FROM MYSTERIOUSLY APPEARING IN YOUR FILES. HOWEVER IF YOU REQUIRE LOWER CASE CHARACTERS OR YOUR TRS80 IS MODIFIED TO DISPLAY LOWER CASE, THE SETUP PROGRAM WILL UNLOCK THE KEYBOARD AND ALLOW THE VIDEO TO DISPLAY LOWER CASE CHARACTERS. THE SHIFT KEY WILL FUNCTION CORRECTLY. LOWER CASE CHARACTERS WILL BE SENT UNLESS THE SHIFT KEY IS PRESSED.

4) MOST PRINTERS HAVE A TOP OF FORM FEATURE HOWEVER A FEW (SUCH AS THE CENTRONICS 779) REQUIRE EXTERNAL SOFTWARE SUPPORT FOR THIS FUNCTION. THE OMIKRON PARALLEL PRINTER DRIVER HAS A ROUTINE THAT SIMULATES THE TOP OF FORM FEATURE. THE SETUP PROGRAM MUST BE USED TO ENABLE THIS FUNCTION.

5) THE VIDEO DRIVER SOFTWARE IS NORMALLY SET TO DISPLAY GRAPHICS CHARACTERS WHEN CALLED WITH VALUES LARGER THAN (80H). THE SETUP PROGRAM ALLOWS THE USER TO DISABLE THE GRAPHICS AND CONVERT THE CHARACTERS TO THEIR ASCII EQUIVALENTS.

SOFTWARE COMMENTS 5

6) MANY OF OUR CUSTOMERS HAVE EXPRESSED INTEREST IN RUNNING WORD PROCESSORS LIKE THE MAGIC WAND AND WORDSTAR. ALTHOUGH THE ADDRESSABLE CURSOR IS ALL THAT'S REALLY REQUIRED TO RUN THESE PROGRAMS, SOME PEOPLE NOTED THAT ADDITIONAL FEATURES WOULD BE HELPFUL. IN ORDER TO DEVELOP THE BEST POSSIBLE PACKAGE, A SPECIAL VERSION OF THE CP/M SYSTEM WAS DEVELOPED. THIS VERSION REQUIRES MORE MEMORY THAN THE STANDARD VERSION SO IT IS SUPPLIED ON REQUEST ONLY. IN THE FUTURE IT IS EXPECTED THAT A NEW LARGER ROM WILL BE SUBSTITUTED FOR THE CURRENT ONE. AT THIS TIME THE ADDITIONAL ROUTINES WILL BE PLACED IN ROM TO FREE THE RAM FOR PROGRAM USE.

THE WORD PROCESSING SOFTWARE HAS SEVERAL ADDITIONAL FEATURES TO OPTIMIZE THE CP/M SYSTEM FOR WORD PROCESSING USE. IN ORDER TO PREVENT DROPPING CHARACTERS WHEN TYPING AT MAXIMUM SPEED, AN INTERRUPT DRIVEN INPUT BUFFER HAS BEEN ADDED TO STORE CHARACTERS WHILE THEY ARE WAITING FOR PROCESSING. THE KEYBOARD ROUTINE HAS BEEN CHANGED TO INCORPORATE A REPEAT FUNCTION. WHEN A KEY IS DEPRESSED CONTINOUSLY, THE CHARACTER WILL BE REPEATED. THIS IS VERY USEFUL FOR RAPID CURSOR POSITIONING AND SCROLLING THROUGH TEXT. ALSO THE CURSOR IS SET TO BLINK. THIS SAVES TIME LOOKING FOR THE CURSOR WHEN A COMMAND IS EXECUTED.

IF YOU ARE USING THE WORD PROCESSING CP/M SYSTEM (IDENTIFIED BY A 47K SIGN ON MESSAGE), THE SETUP PROGRAM WILL ASK YOU ONE OR MORE ADDITIONAL QUESTIONS. THE FIRST OPTION WILL ALLOW YOU TO TURN THE BLINKING CURSOR ON AND OFF. IN SOME INSTANCES THE BLINKING MAY BE DISTRACTING. WHEN THE BLINKING IS TURNED OFF WITH THIS QUESTION, THE INPUT BUFFER AND REPEAT FUNCTIONS WILL STILL BE FUNCTIONAL. ALSO YOU WILL HAVE THE OPTION OF DISABLING THE INTERRUPTS. THIS WILL DISABLE ALL THE WORD PROCESSING FEATURES. THIS OPTION IS NECESSARY TO AVOID DEPENDENCY ON THE REAL TIME CLOCK. WHEN THE INTERRUPTS ARE DISABLED THE SYSTEM WILL STILL FUNCTION WITH A DEFECTIVE REAL TIME CLOCK.

SOFTWARE COMMENTS 6

7) THE OMIKRON SOFTWARE NOW INCLUDES A ROUTINE THAT READDRESSES THE DRIVES UNDER SOFTWARE CONTROL. MOST CUSTOMERS CHOOSE TO HAVE THE CP/M SUPPLIED ON A MINI DISK, ALLOWING BOTH TRSDOS AND CP/M TO BOOT FROM THE SAME DRIVE. THIS ELIMINATES THE NEED TO READDRESS THE DISK DRIVES WHEN SWITCHING OPERATING SYSTEMS. IN THIS CONFIGURATION DRIVES "A" AND "B" ARE MINI AND "C" AND "D" ARE 8".

IN SOME INSTANCES, IT IS DESIRABLE TO HAVE THE 8" DRIVES ADDRESSED AS "A" AND "B". FOR EXAMPLE, BOTH THE "PEACHTREE SOFTWARE" AND THE "STRUCTURED SYSTEMS GROUP" SOFTWARE REQUIRE THAT DRIVE "A" BE USED FOR PROGRAM STORAGE AND DRIVE "B" FOR DATA FILES. NORMALLY THIS WOULD REQUIRE THAT THE 8" DRIVES BE REMOVED FROM THEIR CABINET AND PHYSICALLY READDRESSED.

OMIKRON HAS SOLVED THIS PROBLEM BY INCLUDING AN OPTION IN THE "SETUP" PROGRAM TO "SWAP" THE DRIVE ADDRESSES. WHEN THE ADDRESSES ARE SWAPPED, THE MINI DRIVES, PHYSICALLY ADDRESSED AS "A" AND "B", ARE LOGICALLY ADDRESSED AS "C" AND "D". THE 8" DRIVES, PHYSICALLY ADDRESSED AS "C" AND "D", ARE ACCESSED AS "A" AND "B". WHEN THE SYSTEM IS "WARM-BOOTED" (CONTROL C), THE SOFTWARE WILL READ THE SYSTEM FROM THE 8" DRIVE.

IN ORDER TO WARM BOOT FROM THE 8" DRIVE, THE 8" DISK MUST HAVE THE CP/M SYSTEM "SYSGENED" ON IT. FIRST USE THE "MSYSGEN" PROGRAM TO READ THE SYSTEM FROM THE DISK IN DRIVE "A" (A MINI DISK) INTO MEMORY. REBOOT AND EXECUTE THE "LSYSGEN" PROGRAM. FOR THE SOURCE DESTINATION, TYPE "RET" TO SKIP, AND THEN TYPE "C" TO WRITE THE SYSTEM (FROM MEMORY) TO THE 8" DISK IN DRIVE "C". ONCE THE CP/M SYSTEM IS PLACED ON THE 8" DISK, THE "LSYSGEN" PROGRAM IS USED TO TRANSFER THE SYSTEM TO OTHER 8" DISKS.

TERM.ASM: WHEN THE SERIAL PROGRAM IS EXECUTED, THE UART IS INITIALIZED TO THE PROPER VALUES AND THE SERIAL PRINTER DRIVER IS MOVED INTO MEMORY ON TOP OF THE PARALLEL DRIVER. THE SERIAL PRINTER DRIVER INCLUDES AN RS232 INPUT ROUTINE. THIS INPUT ROUTINE IS CALLED THROUGH THE CP/M READER I/O FUNCTION. THE TERM PROGRAM IS A SAMPLE PROGRAM THAT CAN BE USED TO CONFIGURE THE TRS80 AS A DUMB TERMINAL. THIS TERMINAL MAY BE CONNECTED TO A MODEM FOR REMOTE USE. THIS PROGRAM IS GIVEN IN SOURCE FORM SO THE USER MAY ADD ADDITIONAL ROUTINES FOR HANDSHAKING OR DISK ACCESS.

SOFTWARE COMMENTS 7

TRSCPM.COM: THIS PROGRAM READS A FILE FROM A TRSDOS MINI DISK AND TRANSFERS IT TO A CP/M DISK. FILES ARE TRANSFERED BY FILE NAME AND THE NEWLY CREATED CP/M FILE HAS THE SAME NAME AS THE TRSDOS FILE. SEVERAL FILE CONVERSION OPTIONS ARE INCLUDED TO ASSURE COMPATABILITY WITH THE CP/M FILE STRUCTURE.

MOST PEOPLE FIND IT TROUBLESOME TO MAINTAIN TWO PROGRAM LIBRARIES FOR THE DIFFERENT OPERATING SYSTEMS. CP/M IS THE LOGICAL CHOICE BECAUSE FUTURE EXPANSION IS PAINLESS AND THE LONG TERM SOFTWARE INVESTMENT IS PROTECTED. "TRSCPM" SOLVES THE PROBLEM OF SAVING LONG PROGRAMS DEVELOPED UNDER TRSDOS. WHILE MANY TRSDOS PROGRAMS ARE EASILY CONVERTED TO RUN UNDER CP/M, SOME PRESENT SPECIAL PROBLEMS. THE FOLLOWING PARAGRAPHS CONSIDER PROGRAM CONVERSION IN MORE DETAIL.

TRSDOS FILES AND CP/M FILES HAVE A FEW IMPORTANT DIFFERENCES. TRSDOS FILES ARE STORED IN (256) BYTE SECTORS WHILE CP/M FILES USE (128) BYTE SECTORS. IN ADDITION, LINES ARE SEPARATED IN TRSDOS ASCII FILES WITH A SINGLE CARRIAGE RETURN (0DH), BUT WITH A CARRIAGE RETURN - LINEFEED PAIR (0DH,0AH) IN CP/M. THE TRSCPM PROGRAM HAS PROVISIONS TO COMPENSATE FOR THESE TWO DIFFERENCES, HOWEVER TRSDOS COMMAND FILES ARE MORE COMPLICATED BECAUSE THEY CONTAIN BYTES WHICH TELL THE TRSDOS SYSTEM WHERE TO LOAD THE PROGRAM. THIS INFORMATION IS DISCUSSED IN GREAT DETAIL IN H. PENNINGTON'S BOOK "TRS-80 DISK" . IN GENERAL TRSDOS ASCII FILES, INCLUDING BASIC PROGRAMS, CAN BE CONVERTED TO CP/M OPERATION WITH LITTLE OR NO TROUBLE, WHILE TRSDOS MACHINE CODE PROGRAMS REQUIRE EXTENSIVE MODIFICATION.

MICROSOFT HAS WRITTEN MOST OF THE BASIC INTERPRETERS AVAILABLE FOR MICROCOMPUTERS, INCLUDING VERSIONS FOR BOTH THE CP/M OPERATING SYSTEM AND THE TRSDOS SYSTEM. THESE VERSIONS ARE VERY SIMILAR AND PROGRAM CONVERSION IS NOT COMPLICATED. MICROSOFT CP/M BASIC VERSION (4.51) IS THE CLOSEST VERSION TO THE TRSDOS VERSION. THE DIFFERENCES ARE CAREFULLY EXPLAINED IN KEN KNECHT'S BOOK "MICROSOFT BASIC". THE "COMPRESSED" BASIC FORMAT (TOKENS) NORMALLY STORED ON DISK IS NOT INTERCHANGABLE BETWEEN VERSIONS: THE FILES MUST FIRST BE SAVED IN ASCII FORMAT. IF THE CARRIAGE RETURNS IN THE ASCII FILES ARE "EXPANDED" WHEN THE FILE IS CONVERTED TO CP/M, THE NEW FILES NEED SIMPLE STATEMENT SUBSTITUTIONS TO RUN UNDER THE CP/M BASIC. THE TWO MOST COMMON PROBLEMS ARE THE "PRINT@" STATEMENTS AND THE SCREEN ADDRESSING STATEMENTS. THE "PRINT@" STATEMENTS MAY BE CONVERTED TO PRINT STATEMENTS WITH THE ADDRESSABLE CURSOR ROUTINE AND THE SCREEN PEEK AND POKE STATEMENTS MUST HAVE (0C000H) ADDED TO THEM.

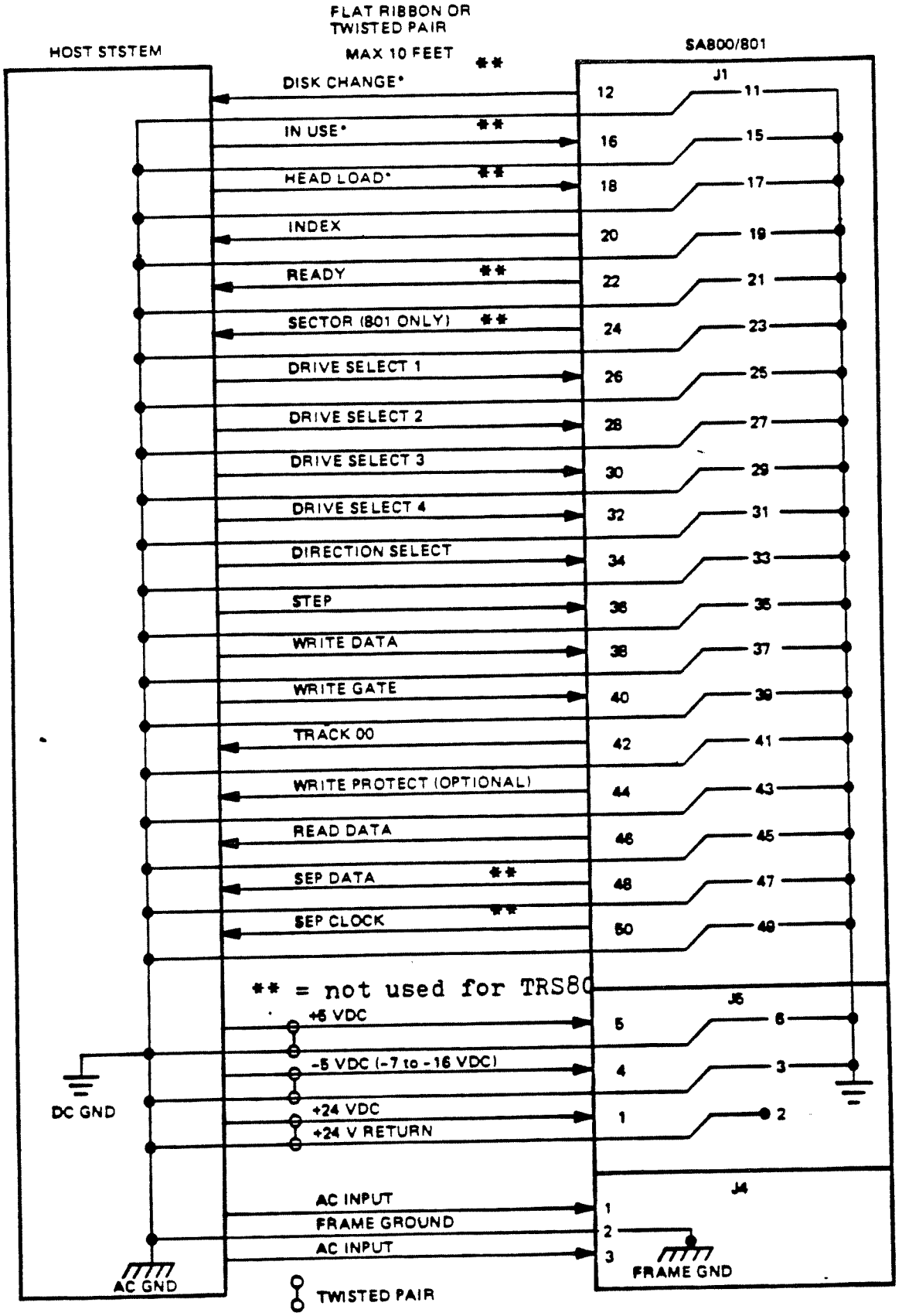
SOFTWARE COMMENTS 8

ASSEMBLY LANGUAGE SOURCE FILES DEVELOPED UNDER TRSDOS MAY BE CONVERTED WITH THE ASCII CONVERSION ROUTINE. THERE ARE SEVERAL COMMON CHANGES REQUIRED TO COMPLETE THE CONVERSION. IN MOST CASES THE "ORG" ADDRESS MUST BE CHANGED TO (100H) AND THE "IO" CALLS TO THE KEYBOARD, PRINTER, AND SCREEN MUST BE CHANGED TO THEIR CP/M EQUIVALENTS. PROGRAMS WHICH DIRECTLY ADDRESS THE SCREEN (OR ANY MEMORY MAPPED PORT) MUST HAVE THE ADDRESSES CORRECTED BY ADDING (0C000H) TO THEM. IF THE MICROSOFT CP/M ASSEMBLER IS USED, COLONS MAY HAVE TO BE ADDED TO STATEMENT LABELS.

WHEN A "YES" ANSWER IS GIVEN TO THE QUESTION ON EXPANDING LINEFEED CHARACTERS, A SECOND QUESTION MUST BE ANSWERED ("DO YOU WANT TO SET BIT (7) TO (0) ?"). SOME PROGRAMS, SUCH AS THE MICROSOFT EDITOR AND INTERPRETER, USE BIT (7) TO FLAG SPECIAL FUNCTIONS (SUCH AS LINE NUMBERS). THE OPERATOR HAS THE OPTION OF SETTING BIT (7) IN EVERY BYTE TO (0). IF THE PROGRAM IS GOING TO BE ASSEMBLED WITH THE MICROSOFT CP/M MACRO ASSEMBLER OR RUN UNDER THE MICROSOFT CP/M BASIC INTERPRETER, LEAVE BIT (7) ALONE. MOST CP/M PROGRAMS DO NOT USE BIT (7) FOR SPECIAL FUNCTIONS. IF ANOTHER EDITOR, ASSEMBLER OR INTERPRETER IS USED, IT MAY BE DESIREABLE TO SET BIT (7) TO (0). IN MANY CASES THE LINE NUMBERS WILL HAVE TO BE ELIMINATED USING THE CP/M EDITOR (THIS IS EASILY DONE WITH A MACRO COMMAND).

THE TRSCPM PROGRAM DOES NOT LOOK FOR AN "END OF FILE" MARKER TO STOP THE TRANSFER. IN MOST CASES EXTRA "GARBAGE" WILL BE FOUND AT THE END OF THE CP/M FILE. THIS WILL NOT INTERFERE WITH THE PROGRAM DATA IN ANY WAY. IT IS EASILY ELIMINATED BY USING THE CP/M EDITOR TO ERASE EVERYTHING PAST THE LAST STATEMENT IN THE PROGRAM.

CABLE CONNECTIONS FOR 8" DRIVE



NOTE: Not shown are 6 of the 9 Alternate I/O connections. The connections for these lines are on pins 2, 4, 6, 8, 10, and 14. Signal return for these lines are on pins 1, 3, 5, 7, 9 and 13 respectively. Reference section 7 for uses of these lines.

*These lines are alternate input/output lines and they are enabled by plugs. Reference section 7 for uses of these lines.

Figure 9. Interface Connections

7.0 CUSTOMER INSTALLABLE OPTIONS

The SA800/801 can be modified by the user to function differently than the standard method as outlined in sections 3 and 4. These modifications can be implemented by adding or deleting traces and by use of the Alternate I/O pins. Some traces are capable of being connected by use of a shorting plug, Shugart P/N 15648 or AMP P/N 530153-2. This section will discuss a few examples of modifications and how to install them. The examples are:

1. Drive Select one to eight drives.
2. Select drive without loading head or enabling stepper.
3. Select drive and enable stepper without loading head.
4. Load head without selecting drive or enabling stepper.
5. Radial Ready.
6. Radial Index/Sector.
7. Eight, 16, or 32 Sector option.
8. In Use (Activity L.E.D.) optional input.
9. Write Protect options.

Tabulated below are the trace options with the condition of the trace as it is shipped from the factory. Figure 21 shows the location of these traces on the PCB.

CUSTOMER CUT/ADD TRACE OPTIONS

TRACE DESIGNATOR	DESCRIPTION	SHIPPED FROM FACTORY	
		OPEN	SHORT
T3,T4,T5,T6	Terminations for Multiplexed Inputs		Plugged
T1	Terminator for Drive Select		Plugged
T2	Spare Terminator for Radial Head Load	X	
DS1,DS2,DS3,DS4	Drive Select Input Pins	X	DS1 is Plugged
RR	Radial Ready		X
RI	Radial Index and Sector		X
R,I,S,	Ready, Index, Sector Alternate Output Pads		X
HL	Stepper Power From Head Load		Plugged
DS	Stepper Power From Drive Select	X	
WP	Inhibit Write When Write Protected		X
NP	Allow Write When Write Protected	X	
8,16,32,	8, 16, 32 Sectors (SA801 Only)	8 & 16	32
D	Alternate Input-In Use	X	
2,4,6,8,10.12,14,16,18	Nine Alternate I/O Pins	X	
D1,D2,D4,DDS	Customer Installable Decode Drive Select Option	X	
A,B,X	Radial Head Load		Plugged
C	Alternate Input-Head Load	X	
Z	In Use from Drive Slect		Plugged
Y	In Use from HD LD	X	
DC	Alternate Output-Disk Change	X	

4.1.1.1 Input Line Termination

The SA800/801 has been provided with the capability of terminating the four input lines, which are meant to be multiplexed, by jumpering traces. The four lines and their respective jumpering traces are:

1. Direction Select . . . . . Trace "T3"
2. Step. . . . . Trace "T4"
3. Write Data . . . . . Trace "T5"
4. Write Gate . . . . . Trace "T6"

In order for the drive to function properly, the last drive on the interface must have these four lines terminated. Termination of these four lines can be accomplished by either of two methods.

1. As shipped from the factory, jumpers are installed on the terminator posts T3, T4, T5, and T6. Remove these shorting plugs from all drives except the last one on the Interface.

2. External termination may be used provided the terminator is beyond the last drive. Each of the four lines should be terminated by using a 150 ohm, ¼ watt resistor, pulled up to +5 VDC.

4.1.1.2 Drive Select 1 - 4

Drive Select when activated to a logical zero level, activates the multiplexed I/O lines and loads the R/W head. In this mode of operation only the drive with this line active will respond to the input lines and gate the output lines.

Four separate input lines, Drive Select 1, Drive Select 2, Drive Select 3, and Drive Select 4, are provided so that up to four drives may be multiplexed together in a system and have separate Drive Select lines. Traces 'DS1', 'DS2', 'DS3', and 'DS4' have been provided to select which Drive Select line will activate the interface signals for a unique drive. As shipped from the factory, a shorting plug is installed on 'DS1'. To select another Drive Select line, this plug should be moved to the appropriate 'DS' pin. For additional methods of selecting drives, see section 7.1.

4.1.1.3 Direction Select

This interface line is a control signal which defines direction of motion the R/W head will take when the Step line is pulsed. An open circuit or logical one defines the direction as "out" and if a pulse is applied to the Step line the R/W head will move away from the center of the disk. Conversely, if this input is shorted to ground or a logical zero level, the direction of motion is defined as "in" and if a pulse is applied to the step line, the R/W head will move towards the center of the disk.

4.1.1.4 Step

This interface line is a control signal which causes the R/W head to move with the direction of motion as defined by the Direction Select line.

The access motion is initiated on each logical zero to logical one transition, or the trailing edge of the signal pulse. Any change in the Direction Select line must be made at least 1μs before the trailing edge of the Step pulse. Refer to Figure 3 for these timings.

4.1.1.5 Write Gate

The active state of this signal, or logical zero, enables Write Data to be written on the diskette. The inactive state, or logical one, enables the read data logic (Separated Data, Separated Clock, and Read Data) and stepper logic. Refer to Figure 6 for timings.

4.1.1.6 Write Data

This interface line provides the data to be written on the diskette. Each transition from a logical one level to a logical zero level, will cause the current through the R/W head to be reversed thereby writing a data bit. This line is enabled by Write Gate being active. Refer to Figure 7 for timings.

4.1.1.7 Head Load (Alternate Input)

This customer installable option, when enabled by jumpering Trace 'C' and activated to a logical zero level and the diskette access door is closed, will load the R/W head load pad against the diskette. Refer to section 7 for uses and method of installation.

4.1.1.8 In Use (Alternate Input)

This customer installable option, when enabled by jumpering Trace 'D' and activated to a logical zero level will turn on the Activity LED in the door push button. This signal is an "OR" function with Drive Select. Refer to section 7.8 for uses and method of installation.

CP/M: An 8080 Disk Operating System

With Editor, Assembler, and Debugger

Reviewed by Alan R. Miller, Contributing Editor

This article describes the features of a marvelous disk operating system called CP/M. The combination of CP/M and an 8080 or Z-80 microprocessor produces a system that approaches the versatility of large computers. I have not written this article to be an operating manual, but rather to be a complement to the extensive documentation that accompanies CP/M. In particular, I will try to help you avoid the frustrating errors I made at the beginning. Part of the problem is that there is so much to learn and there is so much information in the manuals. I hope that this article will smooth the way for you. If you still have a problem, you might contact the people either at Digital Research or Lifeboat Associates. I have found both to be very helpful. If all else fails, contact me and I'll try to help.

INTRODUCTION

Every computer needs a system monitor which, at the very least, will coordinate communication between the computer and the peripherals (keyboard, printer, video screen, tape recorder, floppy disks, etc.). This monitor should be able to perform such housekeeping tasks as displaying memory in hexadecimal and ASCII, moving a block of memory, filling memory with a constant, comparing one block of memory to another, and the ability to jump to another program in memory.

Over the past several years, INTERFACE AGE has published articles addressed to one or another of these tasks. By combining the ideas in these articles, it is possible to produce a decent, non-disk monitor. If you have a floppy disk, you need something more.

Floppy disks usually come with a minimal disk operating system (DOS) that will perform only a few of these tasks. For example, the jump command is the only non-disk operation available in the North Star DOS. The PerSci intelligent controller has only disk-oriented commands. The user, therefore, will have to write the necessary non-disk monitor commands and integrate them into the DOS. This task will be much easier if you have an assembler available. You will also want an editor to change the assembler file, and maybe even a debugger to find the problem in the resulting machine code when it doesn't do what you expected.

The people at Digital Research[1] have nicely combined just about all of the features that you could possibly want into an 8080 disk operating system called CP/M. The basic system is available on an 8-inch soft-sectored floppy diskette for $100. The price includes a comprehensive set of operating manuals. A version of CP/M is also available for the North Star 5-inch floppy from Lifeboat Associates[2] for $145.

The basic system includes a monitor, text editor, assembler, and debugger. A minimum of 16K bytes of memory (20K for the Lifeboat version) are required. Additional programs are available separately: MAC, a more powerful assembler with macro capabilities; SID, a more powerful debugger; FORTRAN, and three versions of BASIC. The prices for these additional programs, which will be discussed near the end of this article are given in Table 1.

Table 1. The cost of CP/M programs.

Program	8" Disk	5" Disk
CP/M	$100	$145
MAC	90	100
SID	75	85
BASIC-E	32	30
CBASIC	100	95
BASIC (Microsoft)	300	300
FORTRAN	400	400

THE CP/M MONITOR

CP/M can be divided into several subsystems:

BIOS — The basic I/O (except disk)
BDOS — The basic disk operating system
CCP — The console command processor
TPA — The transient program area (user programs)

BIOS is the program that interfaces your non-disk peripherals to CP/M. Your first task is to write your own customized version of BIOS and copy it onto the CP/M diskette. This will be discussed in detail in a later section.

[1] Digital Research, P.O. Box 579, Pacific Grove, CA 93950
[2] Lifeboat Associates, 164 West 83rd St., New York, NY 10024

The BDOS section takes care of all disk operations for up to four floppy disks. It maintains the file directories and the allocation of sectors. Floppy diskettes are divided into a number of concentric tracks, and each track is further divided into a number of sectors. All but the smallest programs require more than one sector. An operating system that stored data on contiguous sectors would be very inefficient. This is because the overhead in switching from one sector to the next causes the system to just miss the next sector. The system then has to wait until the desired sector comes around again. CP/M solves the problem by staggering the data (from multiple-sector files) throughout the diskette. This gives the system a little more time between the end of one sector and the beginning of the next. Of course, all this manipulation is not apparent to the user.

STARTUP

After you have customized your version of CP/M and loaded it into memory (as discussed later in this article), it can be started up by branching to address zero. The prompt A> appears at the console, indicating that CP/M is ready for commands and that drive A is the default drive.

Some commands will also accept ambiguous file names as a reference to a group of files. The asterisk and question mark are used for this purpose. The * stands for any number of characters; the ? for only one. Thus the ambiguous file name *,* refers to all files on the disk.

THE BUILT-IN COMMANDS

Several commands, built into CP/M, can now be given:

DIR	List directory of files
ERA	Erase file(s) from disk
REN	Rename a file
SAVE	Copy a file from memory to disk
TYPE	Print out a file

Keyboard entries can be edited with the following commands ('X means Control-X):

'U	Cancel the current line
DEL	Delete the last character
'R	Retype the current line
'E	Local carriage return (don't send it)

Other special commands are:

'C	Restart CP/M (warm start from address zero)
'P	Output to list device (toggle on and off)
'S	Freeze/scroll the display (toggle)

The DIR command, without arguments, lists the entire directory on the default drive. An argument may also be given to list only a portion of the directory or to specify the disk drive.

FILE NAMES

File names may be ambiguous or unambiguous. Unambiguous file names may have three parts: the disk name, the primary name, and the file type. The following are examples of unambiguous file names:

B:SOLVIT.ASM
A:LINEAR.HEX
LOGGING.DAT
B:MAR2878

The drive name appears first, followed by a colon. If the file is on the default drive, then the drive name is optional. The primary name must start with a letter and may contain up to eight characters. The file type is used to designate the nature of information in the file. It is preceded by a decimal point.

Some file-types are reserved by the system:

ASM	Assembler source file
HEX	Intel HEX object file
PRN	Print file of assembly listing
COM	Command file (binary executable file)
BAS	BASIC source file
INT	Intermediate BASIC file
BAK	Backup file from prior edit
LIB	Library source file that can be inserted during editing
SUB	A list of commands to be executed
DAT	Data file
$$$	Temporary work file

The user can also invent other names, e.g., TXT for text.

Some commands will also accept ambiguous file names as a reference to a group of files. The asterisk and question mark are used for this purpose. The * stands for any number of characters; the ? for only one. Thus the ambiguous file name *.* refers to all files on the disk. The name INVER.* refers to all files with the primary name INVER, while *.HEX refers to all HEX files. Files named

NEWTON1.HEX
NEWTON2.HEX
NEWTON3.HEX

can be collectively referred to by

NEWTON?.HEX

THE DIR AND ERA COMMANDS

The commands DIR and ERA both accept ambiguous file names. Examples are:

DIR	List entire directory on default drive
DIR B:	List entire directory on drive B
DIR *.ASM	List all assembler source files
DIR B:PROM.*	List all files on drive B with primary name PROM
DIR *.*	Same as DIR

The ERA command is used to erase files and takes the same arguments as the DIR command. As a precaution against a potentially catastrophic error, the command ERA *.* requires verification. The user must answer the question: Y/N?. Of course, unambiguous file names can be used with the DIR and ERA commands to reference a single file.

THE REN, TYPE, AND SAVE COMMANDS

The REN, TYPE, and SAVE commands require the use of unambiguous file names. To rename a file, type the new file name, an equals sign, and the original file name (i.e., the names appear to be reversed). As we shall see in the next section, the editor gives the original file name to the new file. It then saves the original version, but changes its file type to .BAK. If this file were edited a second time, the original file would be lost. This occurs when the editor renames the intermediate file to type .BAK, and the newest version to .ASM. All three versions can be saved if the .BAK file is renamed before the second edit:

```
REN B:LIFE.BK2 = LIFE.BAK
```

Now after the second edit, there will be three LIFE files of types .ASM, .BAK, and .BK2.

The TYPE command is used to list files on the console or list device. (Typing a ^P alternately turns on and off the output to the list device. It always appears at the console.)

The SAVE command is used to copy programs to disk from memory. The command:

```
SAVE 10 B:TABL.COM
```

will save on disk B a binary file ten pages (256-byte blocks) long. The program must start at 100 HEX. the beginning of the transient program area (TPA). The SAVE command is useful for saving an altered version of a binary program. It is also useful for transferring your old source files to ASM. (More about this later.)

CHANGING DISKS

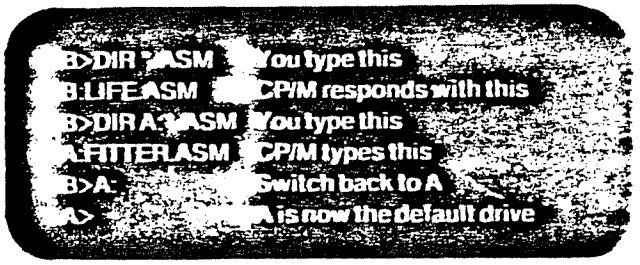
References to files stored on the default drive need not include the drive name:

```
A>TYPE SOLVIT.ASM
A>TYPE A:LINFIT.FOR
```

The default drive can be changed by typing the new drive name followed by a colon:

```
A>B:    You type this
B>      The system responds with this
```

Now commands that refer to disk B need not be preceded with the drive name. Of course, references to files on drive A must be preceded with an A:.



CP/M maintains a storage map of each diskette. So when you change diskettes, you must type a ^C to restart CP/M. Otherwise you won't be able to write on the new diskette. The commands we have just considered are built into the CCP part of CP/M. The other commands we will consider are separate programs. Their names appear separately in the directory listing, and can be removed if they are not needed.

STAT	SUBMIT
ED	DUMP
ASM	LOAD
DDT	PIP

These transient programs will be discussed in the following sections.

DISK STATUS

The program STAT.COM can be used to find the free area of each diskette. The command:

```
A>STAT B:
```

will print the message:

```
BYTES REMAINING ON B: 27 K
```

The STAT command will produce a directory listing in alphabetical order and give the size of each file. The command is:

```
A>STAT B:*.*
```

Disk drives can be software write protected with the command:

```
STAT B: = R/O
```

A warm start (^C) will cancel the protection. Also, the default drive cannot be write protected.

THE TEXT EDITOR

Disk files are created and altered with the command:

```
A>ED filename.type
```

A * prompt will appear, indicating that the editor is ready for commands. If you are creating a new file, type an I (for insert) and a carriage return <CR>. All of the next lines will be entered into the edit buffer in memory. At the end of the test, type a Control-Z (^Z) and the * prompt will reappear. The text you have written can now be displayed or altered. The commands you can use make reference to an imaginary character pointer (CP) that is positioned between two adjacent characters in the edit buffer. Some of the commands move the CP, others manipulate the text with respect to the CP.

MOVING THE CHARACTER POINTER

The pointer is set to the beginning of the buffer with a B and to the end with a —B. A command of C moves the CP one character forward (towards the end of the buffer). The CP can be moved to the beginning of the next line with an L. The pointer-movement routines don't produce any output, so you can't see what is actually happening. In the next section we will introduce additional commands for viewing the text. Either the C or L commands can be preceded by a decimal number to move the CP more than one location. If the number is negative, the pointer moves backwards. A 0L command sets the CP to the beginning of the current line. These commands are summarized in Table 2.

Table 2. Editor commands that move the pointer.

Command	moves the pointer
B	to the beginning of the edit buffer
—B	to the end of the buffer
nC	n characters forward
—nC	n characters backwards
nL	to the beginning of the nth line
—nL	backwards n lines
0L	to the beginning of the current line

VIEWING THE TEXT

The T command can be used to view (Type) lines of the edit buffer. Typing a single T, displays the text from the CP to the end of the line. A 0T, prints text from the beginning of the line to the CP. Since commands can generally be combined, the command 0TT types the entire line that the CP is on. A positive or negative decimal number may precede T as with the C and L commands. The editor can be made to simulate a line-oriented editor. Typing just a carriage return will move the CP to the beginning of the next line and print that line. This is equivalent to entering an LT command. Of course successive lines can be viewed by repeatedly typing a carriage return <CR>.

Text can be altered, inserted, or replaced. The n D command deletes n characters after the CP if n is positive. If n is negative, the n characters before the CP are deleted.

Typing a decimal number n and a <CR> will move the CP n lines and print out the new line. Each prior line can be viewed by typing a —1<CR>.

If scrolling is too fast during execution of the T command, you can type a Z or two. This slows down the display (sleep Z-Z-Z).

ALTERING THE TEXT

Text can be altered, inserted, or replaced. The n D command deletes n characters after the CP if n is positive. If n is negative, the n characters before the CP are deleted. The nK command is similar, except that it deletes (kills) n lines of text starting the the CP. Characters are inserted after the CP with the I command. The insert mode is terminated either with a ^Z or a <CR>. The latter is used if both a <CR> and line feed are to be inserted into the edit buffer. Otherwise the ^Z is used. These additional commands are summarized in Table 3.

Table 3. Edit commands that operate on the text

T	Display from CP to end of line
0T	Display from beginning of line to CP
Z	Slow down scroll speed (sleep)
nD	Delete n characters after the CP
—nD	Delete n characters before the CP
nK	Delete n lines (kill)
—nK	Delete n lines before the CP
—I	Insert text after the CP
<CR>	Move the CP to next line and print it
n	Move the CP n lines and print the last

THE FANCY COMMANDS

There are several fancy commands. Typing an F<string> will move the CP to the end of the string. A decimal number may precede the F to find the nth occurrence of the string. Notice that the F command doesn't change the text. The search/substitute command is often the best method of changing a passage. The format is nS<old string>^Z<new string> (the angle brackets are not actually entered). This command will change the next n occurrences of the old string to the new. Carriage return/line feed pairs are represented in the string by a Control-L.

The X command can be used to block move a portion of text. Typing an 8X, for example, will copy the eight lines after the CP to a temporary disk file. You then move the CP to the new position and type the R command. The eight lines are then copied from the disk file to the new location in the edit buffer. Finally, there is an M (macro) command that can be used to repeatedly execute the rest of the command line. Examples of these commands are given in Table 4.

Table 4. Fancy edit commands.

2FCALL^Z0TT	Find the 2nd occurrence of CALL and print the line it is on
MS"^Z'^Z0TT	Change every occurrence of " to ' and print the new lines

When you have finished editing the text, type an E <CR> (End). The edit buffer will be copied to disk and control will return to CP/M.

EDITING EXISTING FILES

In the previous section, we were talking about creating and editing a file. The editing of an existing file (on disk) is a little different. Give the same initial command.

```
ED filename.type
```

But if you then give the BT commands to move the pointer to the beginning of the file and print the first line, all you get is the * prompt. In fact, any of the above commands will only produce a * prompt. The problem is that the edit buffer is empty. (This problem didn't occur before, because we were creating the file directly in the edit buffer.) The answer is to append some text from the old disk file.

Files small enough to be contained in the edit buffer can be loaded with the command #A. The # sign stands for 65535 and can be used for any command that needs a decimal number. Now that the entire file is in memory, it can be displayed with the command B#T. Edit the text, then type an E to end the edit. The new version will be copied back to the disk and given the original name. The original version is saved too; its file type is changed to .BAK.

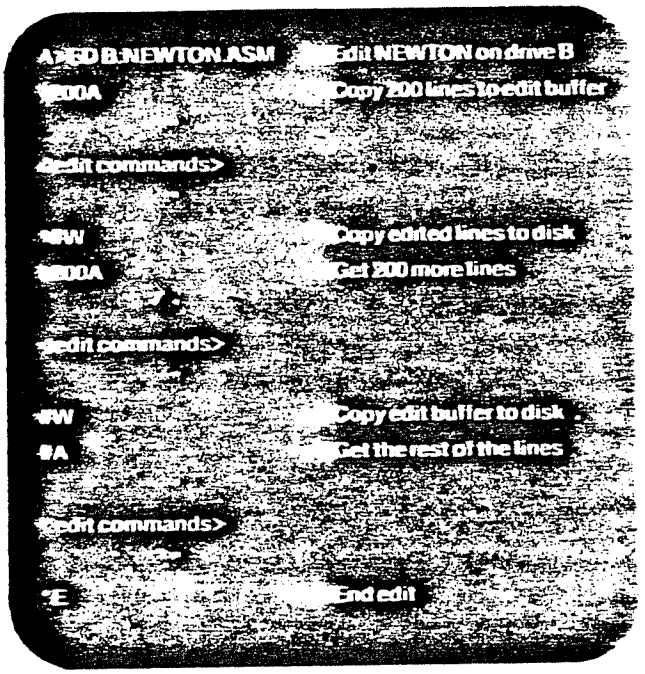
Disk names can be used in the ED command to specify where to get the editor, where to set the original file, and where to place the final version. The command

```
A>B:ED A:FORCE.ASM C:
```

will load the editor from drive B, the original file from disk A and put the edited version on disk C. In this case, C will become the default drive at the end of the edit

EDITING LONG FILES

Files that are too long to fit into the edit buffer can be edited in parts. (You may wonder how such a long file can be used. As we shall see shortly, the assembler operates on a disk source file and puts its results back onto the disk. Thus it is possible, in principle, to assemble huge source files.) The command 200A will copy 200 lines from the source file into the edit buffer. After editing this part, copy it back to disk with the command #W. (There may be more or less than 200 lines after the edit, so we use the # symbol.) The sequence of 200A/#W commands can be repeated as often as necessary.



The E command is then used to end the edit. Any text remaining in the edit buffer is copied to the new disk file (thus it is not necessary to give a #W command prior to the E command). Then any lines left in the original disk file are copied to the new file. The new file is given the old file name and the old file type is changed to type .BAK.

LIBRARY FILES

You probably have sections of assembly language code that you frequently use, such as I/O routines and HEX-binary/binary-HEX conversions. These routines can be placed into a disk file of type .LIB and subsequently inserted into another file that is being edited. The command:

```
*B HEXBIN
```

copies the file HEXBIN.LIB from disk into the edit buffer, starting at the CP.

THE ASSEMBLER

The CP/M assembler operates on disk files of type .ASM. It can produce an assembly listing file (a combination of ASCII HEX code and the original source listing) with the file type .PRN (print). It is called a print file, because it is useful only for humans. Computers have no use for such a listing. A checksummed hexadecimal file in the Intel format can also be produced (of type .HEX). This file contains the assembled machine code. The assembler can be directed to save the .PRN and .HEX files on any available disk, or to output them to the printer, or to not generate them at all.

The assembler uses the standard Intel op codes and the usual pseudo-op codes DB, DS, DW and ORG. Both the DB and DW directives allow multiple arguments which may be mixtures of ASCII and constants.

```
DB      'Checksum error',CR,LF,'at '.0
```

The ORR pseudo-op has no meaning, since the assembler doesn't actually put the resulting machine code into memory.

There is also a conditional assembly pseudo-op:

```
IF   NAME
.
.
ENDIF
```

which allows you to maintain one master source file containing several versions. For example, my main monitor can be assembled with or without VDM capabilities and with either North Star or PerSci disk commands. The portion of text between the IF/ENDIF pairs is assembled if NAME is true (not zero). Near the top of the program, the statement:

```
FALSE   EQU     0
TRUE    EQU     NOT FALSE
```

could be given. Then the statement:

```
NORTH   EQU     TRUE
```

would cause assembly of the conditional parts headed by:

```
IF      NORTH
```

but skip assembly of parts headed with:

```
IF      NOT NORTH
```

Labels can contain up to 16 characters and need not be terminated with a colon. Imbedded $ symbols are ignored, and so can be used for legibility. The assembler is designed to read free-formatted code, so commands can start in any field. More than one statement may appear on each line if separated by an exclamation point.

```
RAR ! RAR ! RAR ! RAR   :MOVE TO LOWER HALF
```

Numerous operations are possible in operands:

+	add	—	subtract
*	multiply	/	divide
AND	logical AND	OR	logical OR
NOT	complement	XOR	exclusive OR
SHR	shift right	SHL	shift left

Parentheses can be used to insure the desired hierarchy of operations.

ASSEMBLY-TIME OPTIONS

Several options can be selected with the assembly command:

command		source on	HEX on	PRN on
A>ASM	WORK	A	A	A
A>ASM	WORK.BBB	B	B	B
A>ASM	B:WORK	B	B	B
A>ASM	WORK.ABC	A	B	C
A>ASM	WORK.BXZ	B	printer	none

The source file must be of type .ASM, but the .ASM must not be entered in the command. If all action is on the same disk, the file name can be preceded with the disk name. An alternate format is also available. The location of the source file, and the disposition of the HEX and PRN files can be specified by three letters in the file-type location. The letters A,B,C and D designate

a disk dive. X sends the output to the console/list device, and Z indicates that generation is to be omitted. The first letter designates the source, the second refers to the HEX file, and the third letter is for PRN file.

A DEBUGGER CALLED DDT

CP/M comes with a powerful dynamic debugging tool (DDT). It is a transient program that is loaded with the command:

A>DDT

After the prompt — (minus) appears, DDT can be used to perform such common monitor functions as: dump memory in HEX and ASCII, fill memory with a constant, and move a block of memory. DDT can disassemble machine code into the 8080 mnemonics and directly assemble individual instructions. For example, there is a Go command which sets the enable interrupt, and then jumps to an absolute address. Before I jump out of CP/M to my PROM monitor, I enter a short program with DDT.

DDT jumps to F000 where the instructions will disable the interrupts, then jump to my monitor at F800.

An assembled program can be debugged with the command:

A>DDT CERAM.COM<CR>

The executable program CERAM is loaded at 100 HEX. The program counter will initially be set to 100 HEX so that a G command (without arguments) will start execution of the program. One or two breakpoints can be set with the G command, so that the test program can be terminated prematurely. All of the 8080 registers can be inspected and altered with the X command. There are additional commands which allow a program operation to be fully traced.

HEX files are more conveniently loaded with the commands:

—I filename.HEX
—Ra

where the argument a is an optional load offset. (HEX files contain the load address at the start of each record.) After assembling my monitor for F800 HEX, I can load the resulting HEX file into the beginning of the user's area (100 HEX) with the command:

—I PRM.HEX <CR>
—R 900 <CR>

FILE TRANSFER

Files can be transferred from disk to disk or from any peripheral to any other by using the transient program PIP. For example:

A> PIP B:PENCIL.COM = A:

will copy the binary file PENCIL from drive A to B. Notice that the order appears reversed, as with the REN command. All files on drive A can be copied to drive B with the command:

A>PIP B:= *.*

PIP can be used to combine several files into a master file. Thus:

A>PIP B:NEW.FOR = B:SORT.FOR,PLOT.FOR

will produce a file on drive B with the name NEW.FOR from the two files SORT on drive B and PLOT on drive A. With this PIP command, you have to be patient. There will be disk activity, then nothing for a while, then more disk activity. But finally, just when you think the system died, the CP/M prompt will appear indicating that all is well.

GETTING UP AND RUNNING

It's easier to get up and running the first time if you already have a disk operating system and an assembler. You have to write the routines that will interface CP/M to your I/O devices. CP/M has the ability to use several peripherals, such as a video screen, printer, punch, tape reader, etc. But it is better to get started with a minimum system using only console I/O. Then when you have CP/M running, you can use it to generate a more sophisticated system (BIOS).

The first thing you should do is make a copy of the CP/M diskette, and save the original for a backup. If you have only one disk drive, you can make a copy by alternately transferring portions of the original diskette into memory, switching to the new diskette, and copying back. If you don't have a disk operating system, it may be difficult to get CP/M running.

GETTING STARTED WITH THE LIFEBOAT VERSION

If you have a North Star system, you can get the Lifeboat Associates version of CP/M running in a matter of minutes. Assemble your BIOS for 4400 HEX. Load North Star DOS in the usual way, and then switch to the new CP/M diskette. Type the command:

WR 28 4400 2 <CR>

The North Star DOS will copy your BIOS onto the CP/M diskette. Now give the command:

JP E900 <CR>

DOS will jump to the bootstrap loader in PROM and start up CP/M. From now on, you can start CP/M by addressing E900.

FULL-SIZE FLOPPIES

With an eight-inch floppy disk, you may have a bit more work to do. You will need some routines that interface CP/M to your disk as well as to the other peripherals. Assemble your BIOS for the address 3E00 HEX and copy it onto Track 1, Sectors 18-21.

The disk interface programs took me a while to write. The problem was with the track- and sector-seek routines. CP/M sends its request in binary, and I was using a PerSci intelligent interface that wanted ASCII decimal input. I therefore had to include a binary-to-ASCII conversion routine. Furthermore, both CP/M and the PerSci controller perform cyclical-redundancy checks. I suppose that these factors are partly responsible for the noticably slow operation. The controller comes with a serial option that also causes a slowdown. (I understand that PerSci has a new version of its DOS available that accepts binary input and is designed for parallel operation only. This should make interfacing easier and perhaps increase the speed.)

It looks like one of the dumb-interface boards available from Tarbell or S.D. Sales is a good way to go. These apparently come with software (or firmware) that should make it much easier to get CP/M going. I will have the Tarbell unit shortly, and will devote an article to it in a forthcoming issue.

ENLARGING THE SYSTEM

CP/M does its work in the transient program area (TPA) between 100 and 28FF HEX. If you have more than the minimum memory, you should enlarge CP/M so that it will have more work space. This is easily accomplished with the routines on the CP/M diskette (if you don't follow the confusing instructions in the manual). There are several ways of accomplishing the expansion; the more sophisticated, use CP/M entirely.

I did the enlarging before I was familiar with all the CP/M commands, so it isn't the most efficient. Nevertheless, since I was able to accomplish the task, I will show you how it can be done on the North Star system. You will need an initialized diskette, and the best way to get one is to make another copy of CP/M.

To generate a nominal 46K system (which requires 50K of memory), type:

A>NRELOC 46 *

The system responds with:

CONSTRUCTING 46 K CP/M
READY FOR "SYSGEN" OR
"SAVE 38 CPM46.COM"

Now type:

A>NSYSGEN

and you get:

SYSGEN VERSION 1.1
GET SYSTEM? (answer N)
PUT SYSTEM?

Answer Y if you have more than one disk and put an initialized diskette into drive B. If you have only one drive, answer A and replace the 16K diskette with an initialized diskette. A verification message appears next:

SOURCE ON B [or A], THEN TYPE RETURN (do it)
FUNCTION COMPLETE
REBOOTING, TYPE RETURN.

If you typed an A to the PUT SYSTEM? command, switch back to the 16 K CP/M diskette before typing return.

The new version of CP/M won't run yet because it doesn't have your customized BIOS. Reassemble your I/O routines to run at the new address:

MSIZE* 1024 + 400H (Lifeboat version)

where MSIZE is the new nominal size in decimal K (46 in the above example). Copy the reassembled BIOS to address 28 as you did at the very beginning. Now a cold start (jump to E900) will load the new, larger system.

If you have an 8-inch floppy, give the commands:

A>CPM 46 *
A>SYSGEN

rather than NRELOC and NSYSGEN. Reassemble your BIOS to:

MSIZE* 1024 — 200H (8-inch disk)

and copy it to Track 1, Sectors 18-21.

CONVERTING YOUR OLD SOURCE PROGRAMS

You can copy your old assembly language source files to CP/M disk files, and then assemble them after making a few alterations. The CP/M editor stores both a carriage return and line feed at the end of each line. My MITS source files contain only a carriage return. Furthermore, CP/M expects a pair of apostrophes to delimit ASCII characters, but my old source files use quotation marks.

The CP/M editor can insert a carriage return/line combination, but it can't insert just a line feed. Therefore, I had to preprocess my source files. The easiest way to make the conversion is to replace all line feeds with something else not used in the text (e.g., a # symbol). Then copy the file to CP/M and use the S command to replace the # with the CR/LF pair (where ^L stands for CR/LF). I couldn't do this at first, since my monitor uses ^L to switch output to the list device.

The program shown below will add a line feed after each carriage return and will also change each quote to an apostrophe. I used my North Star DOS to load the file into memory starting at 4000 HEX, then jumped to my monitor and block moved the source program to 2C00 HEX. I started up CP/M and executed my ADDLF.

A>ADDLF

```
                ;  PROGRAM TO ADD LF TO MITS SOURCE PROGRAMS
                ;
                ;  APRIL 14, 1978
                ;
0100            ORG 100H
                ;
2C00 =          OLD     EQU     2C00H   ;START OF ORIGINAL
2200 =          NEW     EQU     2200H   ;START OF NEW VERSION
                ;
0100 11002C     START:  LXI     D,OLD   ;OLD POINTER
0103 210022             LXI     H,NEW   ;NEW POINTER
0106 1A         NEXT:   LDAX    D       ;GET A BYTE
0107 FE22               CPI     '"'     ;CHECK FOR QUOTE
0109 C20E01             JNZ     SKIP
010C 3E27               MVI     A,39    ;CHANGE TO '
010E 77         SKIP:   MOV     M,A     ;PUT INTO NEW FILE
010F 13                 INX     D       ;INCREMENT POINTERS
0110 23                 INX     H
0111 FE00               CPI     0       ;CHECK FOR 0 OR 1
0113 DA2101             JC      DONE
0116 23                 INX     H
0117 FE01               CPI     1
0119 C20601             JNZ     NEXT    ;NO
011C 3E0A               MVI     A,10    ;PUT LINE FEED IN
011E 23                 INX     H       ;INCREMENT POINTER
011F C30601             JMP     NEXT    ;NEXT BYTE
                ;
0122 361A       DONE:   MVI     M,1AH   ;PUT ^Z AT END
0124 C30000             JMP     0       ;JUMP TO CPM
0127                    END
```

ADDLF will copy the source file starting at 2C00 to 2200, making the necessary changes until either a binary one or zero is found. (I use a binary one for an end-of-file (EOF) marker). ADDLF puts a 1A HEX (^Z), the EOF marker for CP/M, at the end. When ADDLF is done (it takes only seconds), it jumps to my monitor so that I can inspect the new file and block move it down to 100 HEX. A jump to address zero will restart CP/M so that the converted source file can be saved. The command: SAVE 20 filename.ASM will copy a 20-block file to the default drive. There must be an easier way, but at least this method works.

PROGRAMS THAT RUN ON CP/M

There are several intersting programs designed to run with CP/M. No modification is necessary; just put the diskette in and type the program name. Three BASICs are available: BASIC-E, CBASIC, and Microsoft BASIC. The first two compile your source program from disk and put the resulting code back onto disk. The compiled code should run much faster since instructions in loops aren't continually being re-interpreted. Also there is no need for a 12K or 16K BASIC to be taking up memory. The disadvantage of a compiled code is that it may be harder to debug.

The Microsoft BASIC appears to be a CP/M version of MITS disk BASIC. Microsoft also has a CP/M version of FORTRAN that sounds yummy. It would certainly make data reduction in our thermodynamics lab easier.

I expect to have BASIC-E, a macro assembler called MAC and a debugger called SID running shortly; I will give you a report on them in the near future.

Next month I will report on the CP/M version of Michael Shrayer's Electric Pencil, a very versatile paragraph-oriented text editor. I used Electric Pencil to compose, edit and format this CP/M article. □