MAX-80

OPERATION MANUAL

358 SOUTH FAIRVIEW AVE., GOLETA, CALIFORNIA 93117 (805) 683-1576 TELEX 658 492
TRADEMARKS

MAX-80 is a trademark of Lobo Systems, Inc.

CP/M is a registered trademark and CP/M 2.2, CP/M 3 and CP/M Plus are trademarks of Digital Research.

GENERAL MILLS is a registered trademark of General Mills Corp.

Z-80 is a registered trademark of Zilog Corporation.

LDOS is a trademark of Logical Systems, Inc.

Model I, Model III, Model 4 and TRS-80 are registered trademarks of Tandy Corporation.

PUBLISHED BY

LOBO SYSTEMS
358 South Fairview
Goleta, California 93117

MAX-80™ OPERATION MANUAL

Copyright (c) 1984 by Lobo Systems. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a data base or retrieval system, without the prior written permission of the publisher.
THE MAX-80 IS A VERIFIED CLASS A COMPUTING DEVICE

The FCC requires that the following be included in the users' manual for this computing device:

WARNING: This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions manual, may cause interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the owner, at his or her own expense, will be required to take whatever measures may be required to correct interference.
LOBO SYSTEMS STANDARD HARDWARE WARRANTY

MAX-80

Lobo Systems warrants all of its hardware products against defects in material and workmanship for ONE YEAR from the date of purchase. During the ONE YEAR period, Lobo Systems will repair or, at its election, replace a defective Lobo Systems hardware product or part at NO CHARGE to the original purchaser, PROVIDED that the product is returned freight prepaid to Lobo Systems or to an authorized repair station. Lobo Systems' mailing address is 358 South Fairview Avenue, Goleta, California 93117. Telephone (805) 683-1576 for the name and address of your nearest authorized repair station. In order to obtain warranty performance, the Lobo Systems hardware product must be returned in its original packing material and must be accompanied by both a written description of the defect and proof of the date of purchase. (If a purchaser needs replacement packing materials, Lobo Systems will provide them at cost.)

Products available through Lobo Systems, but which are not manufactured by Lobo Systems, carry the manufacturer's warranty only.

LIMITATION OF LIABILITIES

The above warranty does not apply if the product has been damaged by accident, abuse, misuse, misapplication, or if the product has been subjected to unusual physical or electrical stress or has been modified without Lobo Systems' express approval. The purchaser's remedy is limited to repair or replacement.

In no event shall Lobo Systems be liable for incidental or consequential damages, including loss of profits, arising from the breach of any warranty. THERE ARE NO WARRANTIES WHICH EXTEND BEYOND THE DESCRIPTION ON THE FACE HEREOF.

DISCLAIMER OF IMPLIED WARRANTIES

THE FOREGOING WARRANTY IS EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, EITHER WRITTEN, ORAL OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR PURPOSE.

No representative of the company may change any of the foregoing and the buyer hereby accepts the product subject to all terms expressed here.

LOBO SYSTEMS VERSION 2.1
DISCLAIMER OF SOFTWARE WARRANTY

Lobo Systems makes no warranties, express or implied, with respect to its manuals or the software sold by Lobo Systems or described in its manuals (collectively, the "software"). Lobo Systems DOES NOT WARRANT the quality, performance, merchantability, or fitness for purpose of its software, all of which is sold or licensed strictly AS IS. The entire risk as to quality and performance of software is with the buyer. Should the software prove defective or unsatisfactory following purchase, the buyer (and NOT the manufacturer, distributor or retailer) assumes the entire cost of all necessary modification, service or repair.

Lobo Systems shall have no liability or responsibility to the buyer or any other person or entity for any liability, loss or damage caused or alleged to be caused directly or indirectly by the software -- including, but not limited to, any interruption of service, loss of business, loss of anticipated profits, or consequential damages resulting from the use or operation of the software.
INTRODUCTION

TABLE OF CONTENTS

INTRODUCTION ..... INTRO-1

Standard Features of the MAX-80 ..... INTRO-3
The MAX-80 Philosophy of Documentation ..... INTRO-6
About This Manual ..... INTRO-7
And If? ..... INTRO-8
Manual Overview ..... INTRO-9

GLOSSARY ..... INTRO-13
INTRODUCTION

The MAX-80 business computer is an exceptional small computer.

It works much faster than the vast majority of microcomputer systems (including many of the much more expensive 8 and 16 bit computers).

It can accommodate a complete range of disk drives, from 3-1/2 inch microfloppies to large 10 Megabyte hard drives. It can even run up to nine drives at a time -- that is, it can keep a very large amount of information on hand, ready for instant access.

It can operate a huge number of different software packages. With its standard operating system, the MAX-80 can run programs from the world's largest collection of software, the CP/M software base. With the other operating systems available for it (DOSPLUS, LDOS and MULTIDOS), the MAX-80 can also use programs from one of the other extremely large bases of software.

It includes, standard, all of the hardware necessary for controlling a variety of computer accessories -- letter quality and high speed printers, graphics plotters, telephone communications modems, speech synthesizers, scientific and engineering instrumentation, industrial equipment controllers, even appliance control modules for turning appliances on and off.

It is fully reconfigurable. The shapes of the letters and graphics characters used on its video screen may be completely changed; all of the keys on its keyboard may be redefined; its commands may be shortened to a single keystroke by the use of its function keys.

And, to top it all, the MAX-80 is one of the least expensive business computers on the market.
TO MAKE THE BEST USE OF THE MAX-80, YOU MUST KNOW HOW TO RUN IT.

This manual is the MAX-80 Operations Manual. It describes how to set up your MAX-80 hardware, how to install its standard CP/M Plus (CP/M 3) operating system, how to use the Lobo utilities that were written especially for the MAX-80, and how to install applications software (the kind you need to really do things) on CP/M Plus.

This section of the manual presents overall descriptions of the MAX-80 and its standard CP/M operating system. It also gives a brief glossary of computer terms that are all too often bandied about without adequate definition. (Please note: we make no claim to having the official definitions of any of these terms. These definitions are only what we mean when we bandy these terms about.)
STANDARD FEATURES OF THE MAX-80

MAXIMUM SPEED: The 80B microprocessor (that is the heart of the MAX-80) runs at 5 MHz (5 million cycles per second). This is over double the speed of most of the popular microcomputers sold today, and significantly faster than most of the rest. The MAX-80's working output (determined in terms of the speed at which a program processes data) is actually greater than some of the so-called 16 bit microcomputers selling for hundreds of dollars more.

What does this mean to you? It means that word processors respond faster to your editing commands, spreadsheets recalculate more quickly, programs run with less of your waiting around, and less time is required for practically everything else. In short, since time is money, it means money saved.

MAXIMUM INTERNAL MEMORY: 128K RAM (Random Access Memory) is a standard feature of the MAX-80. This is at least double (and frequently more than double) the amount of internal memory that is provided with most other microcomputers, including business microcomputers. (Even many of the 16 bit computers come standard with only 64K RAM.)

Meaning? Meaning that programs can be made to run faster than they do with just a fast microprocessor. Meaning even more money saved.

MAXIMUM FLOPPY DISK DRIVE EXPANSION: The standard MAX-80 contains controller interfaces for four 5-1/4 inch and four 8 inch floppy disk drives. The built-in floppy disk controller is capable of reading and writing in single- or double-sided and single- or double-density formats in all of the standard microcomputer floppy disk sizes. This produces both maximum compatibility with other computer systems and maximum storage capacity.

Meaning? Compatibility with other computer systems means that more software is available to do what you want to do with the MAX-80. More software means a greater variety of software. And a greater variety means that it is more likely you will find the software that does exactly what you want. Storage capacity means
that you can have more information at hand -- so either you will have to go searching for the right information less often, or your searches will be finished faster, or both. Again, money saved.

FIXED DISK EXPANSION: In addition to the floppy drive interfaces, the MAX-80 includes (again standard) a Winchester interface for the attachment of a fixed disk drive. Either a Lobo Systems UVC or a SASI fixed disk system can be run from this interface. (For definitions of this jargon, see the glossary at the end of this section.) To connect a non-Lobo drive that is SASI compatible, it is necessary to modify the Lobo disk controller software. (Because, even though SASI is a standard, there still is a lot of variation both in Winchester drives and in Winchester drive controllers.)

Fixed disks cost more than floppies, but they store much more information and retrieve that information faster. So an initial investment pays off in dividends. (Many other computers can't even talk to large 8 inch floppy drives, much less Winchesters.)

MAXIMUM COMMUNICATIONS: Two RS-232C serial ports with independent baud rates are standard on the MAX-80, as is a Centronics-compatible parallel printer port. These three ports allow maximum interfacing to most printers, plotters, modems and speech synthesizers.

A computer specializes in communications. Cars specialize in transportation, houses specialize in shelter, governments specialize in taxes. But for computers it's communications. So the more ways a computer has to communicate, the more ways you have to communicate: words on the screen (CRTs), words on paper (printers), pictures on paper (plotters), words and pictures over telephone lines (modems), words in the air (speech synthesizers). The MAX-80 communicates with a very large variety of these devices. Meaning that you can more easily find the exact device that says what you want to say -- or tells you what you want to hear. With the MAX-80 you can frequently get more than what you pay for.

MAXIMUM REDEFINABILITY: The MAX-80 is capable of displaying on a video screen 192 different character shapes at a time. Any or all of these shapes can be redefined. The MAX-80 has 76 keys on its keyboard. The letters associated with any or all of these keys can be rearranged.
Don't like the characters you see on the screen? Change them. Don't like the locations of the keys on the keyboard? Change them. Dissatisfaction slows things down. Satisfaction speeds things up. The MAX-80 can't guarantee satisfaction. But it can make the chances a lot better.

MAXIMUM RELIABILITY: No extra boards or hardware modifications are required in order to make the MAX-80 do what the small business needs. The MAX-80 requires no special video boards, no extra RAM boards, no extra controller boards, no extra processor boards, no extra keyboard enhancement boards. Once a compatible disk drive (and most disk drives are compatible) has been plugged in, the MAX-80 is ready to run any of the variety of operating systems set up for it (including CP/M Plus, DOSPLUS, LDOS, and MULTIDOS). You won't have to worry about multiple hardware manufacturers -- or multiple hardware warranties. There is just one hardware warranty -- the Lobo Systems one year, all parts, all labor warranty.

We could not afford to warrant the MAX for a full year if the reliability weren't built in. It is interesting that other manufacturers don't have full year warranties. But it is more interesting that you can expect your MAX to keep running, without downtimes, for much more than a single year. That means more effective work time. What else could a small business want?

WELL, HOW ABOUT DOCUMENTATION?

Glad you asked, since that's the subject of the next section.
THE MAX-80 PHILOSOPHY OF DOCUMENTATION

It is one of Lobo's goals to provide the MAX-80 owner with as much information as possible about the MAX-80 system. We therefore provide descriptions not only of the computer's startup procedures and system utilities, but also complete documentation of every major chip on its circuit boards and of every facet of its primary operating system CP/M Plus. The documentation is in every case at least equal to the documentation used by Lobo's logic designers when they designed the MAX-80 hardware and by Lobo's software engineers when they installed the CP/M Plus operating system. Frequently this documentation is better than what they had to work with -- the manuals having often been rewritten and reprinted in the meantime.

Why all this support? Two reasons: beliefs and objectives.

Beliefs. We believe that we should be as open as possible, consistent with our survival. We do not either expect or intend to make a lot of money by discovering some special means of doing something and then keeping it secret so that everyone has to come to us in order be able to use it. The MAX-80 is no secret. It uses high technology hardware and high quality software, but the techniques and knowledge involved are not privy to us. Our income from the MAX comes because it is a genuinely good value. We therefore believe that we can survive in this competitive market even though we fully publish its hardware design and software sources (at least, all of the ones that we authored).

Objectives. We are not interested in selling a lot of computers that are not used -- especially if they aren't used because no one can get the necessary information. Conversely, we frequently are happily surprised to find that someone has figured a new and better way to use one of our computers. One of the pleasures of being a computer manufacturer. But we can attain those objectives only if we give the person who is going to do the work all the information we can.

So, for both reasons, we provide very extensive documentation of the MAX-80.
INTRODUCTION

ABOUT THIS MANUAL

The MAX-80 is a powerful machine. It can do more than most microcomputers. So there are more things that can be done with it than with most microcomputers.

That is, there must be more description of what it can do. There must be more description of what you can do with it. There must be a huge manual.


This manual is large -- not because anything described in it is complicated, and not because great detail is required to do every job -- but because the MAX-80 does so much.

Take a computer that can send and receive information from four 3-/12 inch or 5-1/4 inch drives. And from four 8 inch drives. And from a 5 or 10 Megabyte Winchester drive. Plus connections to parallel printers. And to serial devices (more printers or modems). And to miscellaneous (unknown) controllers. Plus total reconfigurability of the characters put up on the video screen. And reconfigurability of the keyboard. And 12 function keys (in CP/M). My word, even the cursor is changeable.

Describe all that, will you? Well, we tried. And all these words are the result.

In truth, it is easy to set up any one of the things the MAX can do. It even is easy to describe how to set them up. But by the time all of those descriptions are put together, you have what you see before you now -- a manual that is several hundred pages long.

YOU DO NOT HAVE TO READ ALL THESE PAGES. Thankfully.

But you do have to read some of them. Which ones, depends on what you want to do -- and how much you already know.
INTRODUCTION

Even if you do know a lot about computers, we STRONGLY advise that you review the beginning setup sections of the manual. You might have run computers for many years, but it is doubtful that you have spent that much time setting up computers.

Remember, the MAX is not a simple all-in-one computer box. It can be set up in many different ways; therefore, it can be set up in many different wrong ways. You can have a time-wasting and frustrating experience trying to track down just one little thing you missed when you set your system up. Consequently, you can have a time-saving (and frustration preventing) experience to set things up the right way the first time.

Take your time, follow the procedures in this manual carefully, and you will find that your MAX-80 will be up and running in very little time.

AND IF?

However, if you have read and followed the instructions in this manual and still have questions or problems, then call Lobo's Technical Information Office at (805) 683-1576. Our technical support staff is happy to assist you. But, please, first make every attempt to become familiar with this manual and with your new computer.

(One of our chief frustrations is helping someone who has not even attempted to read the manual, while someone else who has genuine problems is kept waiting. So, believe us, we really appreciate it when you have read the relevant parts of the manual, and know what we're talking about when we ask you diagnostic questions. Thanks.)
INTRODUCTION

MANUAL OVERVIEW


Hardware Setup Manual

The hardware setup starts with the unpacking instructions and continues through to the final hardware installation and diskette care. We know the temptation is great to set this manual aside and just have a go at it, but therein lies the path to much frustration. Experienced and inexperienced computer users alike still need to learn the details of the MAX-80.

If you do not intend to use CP/M, but plan to work exclusively on one or more of the other operating systems that are available for the MAX, then the Hardware Setup manual is the only one of the manuals that you will have to read. (Eventually the Technical Reference Manual may become a necessity for some future project, but that will be some time after you have mastered the other operating system.) You can now turn to the respective operating system manual.

CP/M Startup Manual

Once your MAX-80 hardware is set up, you can begin with the operating system. This manual first introduces you to the proper "boot up" procedures for CP/M, and then, when your system software diskette is booted, it shows you how to make a backup copy of your system diskette.

Lobo Systems has written special installation utilities that are not on standard CP/M diskettes. This manual will introduce you to the utilities that are necessary if you want to change the operating system -- whether for other equipment that you are attaching to your MAX, or just for your own style (keyboard reconfiguration, character set redefinition, even cursor redesign).
System Utilities and Software Installation Manual

This manual describes two additional utilities, COPIER and TIMESET, then turns to a function (for the lack of a better name) that is a convenience to many programmers, and finally presents a set of guidelines for installing software packages on the MAX.

You will already, by the time you read this section, have used COPIER to create the backups to your Original System Diskettes. The description in this section will provide complete reference documentation to the COPIER program itself.

TIMESET is a utility that allows you to change the time on the standard MAX-80 clock/calendar. The time is set before the MAX is shipped, but you will still have to reset it for your own time zone. TIMESET is a very quick, and convenient, utility.

The third section describes a random number generator. Many programmers need a random "seed" for a program (they might, for instance, want to demonstrate that, even with an arbitrary starting point, you always get the same result). But there is no way to generate a random seed, except by one's own guesswork -- which is not very random. Lobo's Pseudo-Random Number Generator won't generate a truly random number, but it will generate one that is pretty close -- it certainly is a lot better than guesswork.

And, finally, this manual discusses a subject near and dear to no one's heart, software installation. To run most software on the MAX, you are going to have to tell the program about the MAX's characteristics. It's hard for a program to clear the video screen, when it doesn't know what commands are needed in the MAX-80. It's hard for a program to even write a simple message on the screen, when it doesn't know where the cursor is. The commands for these functions are different in every computer. So your software must know what the commands are in your computer, your MAX-80. This section describes all of the specialized MAX-80 commands, and gives advice on installing software on the MAX.

The technical documentation of the MAX-80 is split along the traditional line dividing the computer world -- hardware and software. The hardware section describes the requirements of all of the input/output ports on the MAX-80, and the Appendices (which are all hardware oriented) provide copies of the original manufacturers' documentation on all of the major chips in the MAX-80. (The other chips are standard items that are described in most modern logic design reference manuals.)

The System Programming section describes, in detail, all of the dedicated hardware addresses in the MAX. With this information, one could install a complete operating system on the computer (in fact, many have been already).

All That?

You do not have to read everything in this manual in order to be able to run the MAX-80. It is a large manual because we try to describe all of the important aspects of the MAX -- so that you have the least difficulty in using it. Your best approach, therefore, will be to pick and choose what you need from all of the manuals.

But first, the hardware setup. Everyone needs that.
GLOSSARY

APPLICATION PROGRAM: Software which is written to accomplish a particular task, usually in the business environment. For instance, many companies use programs called "accounting packages". An accounting package is a set of application programs which together are capable of computerizing the accounts receivable, accounts payable, general ledger, payroll etc. of a business. Other applications are given by mail list programs, statistics programs, fortune teller programs, all the necessary software for running a small business. Now that you own a MAX-80, you can purchase and use application programs which are distributed in the Xerox 5-1/4 inch CP/M format or the standard CP/M single-sided, single density 8 inch diskette format. In addition, with the other operating systems available for the MAX, you will be able to use most of the applications programs which run on the TRS-80 Models I and III.

BACK UP: To make copies of. To back up a floppy diskette or a hard disk is to duplicate the contents of the disk. This is necessary because the information on the disk is readable only by your computer -- and only then if everything on the media is in order. If anything goes wrong with that copy, it will be next to impossible to get all of the rest of the information off that disk. And, of course, the time that you have only one copy of valuable information is the time that something will go wrong. Thus the absolute necessity of backups.
BAUD: One signal change per second (applied to serial communications devices). This term is usually taken to mean the same as "bits per second", but that is inaccurate. In slow speed serial communications, the baud rates (such as 300 baud) do coincide with the data rates given in bits per second. But in higher speed serial communications advanced techniques are used to transfer a number of bits of information in one signal change -- thus making the bits per second data rate many times that of the baud rate. Because every communication requires some extra bits to be included to doublecheck the accuracy of the transmission, a rate of 300 bits per second amounts to an actual transmission of 30 characters (letters, symbols, control codes) per second.

BELL 103, 202, 212, and many other numbers. The standards put forward for telephone transmission. In the microcomputer world, 103 is for modems that run only at 300 baud, 202 for modems that run only at 1200 baud, and 212 for modems that run at either 300 or 1200 baud. The 202 and 212 modems are considerably more expensive than the 103 modems, but the 103 modems are so slow that they cost quite a bit more in long distance phone bills and in aggravation (since many people can read faster than that). The 202 modems have largely been replaced the the 212 modems, since they're nearly the same price and the latter can communicate with a greater variety of other modems.

BIOS: Basic Input/Output System. The part of CP/M which is created by the hardware manufacturer for a particular configuration of hardware. Lobo Systems has created the BIOS for the MAX-80. This BIOS is located in special system files on your MAX-80 system disks; it is loaded into RAM each time your system is turned on or the RESET button is pushed. The program that Lobo wrote to create this BIOS (called its "source code") is also included on the CP/M System Diskettes. The availability of the source code allows a systems programmer to completely reconfigure the operating system to his or her own desires.

BIT (BI_nary digit): A 0 or 1. The computer responds to 0's and 1's and combinations of 0's and 1's. It takes eight of these (called a "byte") to uniquely represent a character. If you never write computer programs, then your only involvement with bits is that you may want to get an idea how many words or pages of English you can fit into your MAX-80 or its disk drives. Check the definitions of Byte, Kilobyte, Megabyte and Permanent Storage Device.
BOOT: When the system is powered up or the reset button is pushed, the boot process is started. This begins with the execution (running) of the program that is contained in the boot ROM. This is a very small program (512 bytes, to be exact). It is only big enough to sound the speaker beep (warning you that your system is not really running yet) and to go out to a disk drive (whichever one is indicated by the boot drive selection switch on the back of the MAX) to get the operating system. No matter which operating system is being used, the boot diskette in the designated boot drive must be turned on and have an appropriate version of the operating system in place. These versions are constructed specifically to operate with the MAX-80. That is, the boot diskette must contain a special MAX-80 version of the operating system. If the appropriate data are found on the boot disk the speaker tone will cease, the system will successfully boot (start up) and the user prompt for the operating system being used will appear. The term "boot", by the way, comes from some jargon of yesteryear: "lifting oneself up by one's own bootstraps". Fortunately, starting up your MAX is a little easier than that.

BOOT DRIVE SELECT SWITCH: This switch is accessible from the rear of the MAX-80 and is used to designate the drive that the MAX-80 will expect to be the permanent boot drive. The switch settings are given in the Hardware Setup section of this manual.

BOOT SECTOR: The initial chunk of code which the boot ROM looks for (on the diskette or hard disk) and which is responsible for causing the rest of the operating system to be properly loaded into memory.

BUG: A problem in either hardware or software (though usually applied to software). At times it is debatable whether some occurrence is desirable or not -- which leads to the ancient saying "One person's bug is another person's feature" (Pliny the Elder, circa 50 AD). Odd happenings in software are universally regarded by software users to be bugs, but by software authors to be features.

BYTE: Eight bits. This is the smallest unit of information that can be used to represent a single character or microprocessor instruction.
CAPS LOCK (SHIFT LOCK) KEY: The key that puts all of the letters you type into upper case. In CP/M Plus the initial caps lock key is the "unshifted" F1 key, but you can change it to any other function key by using the program MAKESYS. The MAX-80 CP/M caps lock is a computer caps lock -- that is, it holds all the letters at uppercase, but the numbers and all other symbols remain at their unshifted values. (As opposed to the typical typewriter, which gives you all the symbols over the numbers when you have the shift lock on.) The program MAKESYS also allows you to specify that your computer start off each day in the caps lock mode. (F1 can then be used to delete the caps lock.) This is useful if you do a lot of BASIC programming, since BASIC only understands capital letters.

CHARACTER MATRIX: Each character that you see on your video screen is made up of discrete units of light (pixels). The MAX-80 can put 192 different symbols (all redefinable by the user) on the screen at one time. There are two types of characters that make up these symbols. 128 of the characters are each composed of a matrix that is 8 pixels wide and 8 pixels tall. 64 of the characters are made up of matrices that are 8 pixels wide and up to 16 pixels tall. To understand what "user-defineable characters" means, see the definition of Graphics Resolution below.

COMMUNICATIONS: The RS-232 serial and the Centronics-type parallel interface ports are often called communications ports because they usually are used to interface to communications devices, such as modems and printers.

COMPATABILITY: Two computer systems are compatible when they can run each other's programs. But to run each other's programs, they usually have to be able to read each other's diskettes. (We say "usually" here, because you can also get programs into your MAX through a modem attached to your phone line. In that case, you MAX might not even have the same diskette size as your host computer. Because your MAX-80 supports CP/M, you have compatibility with many other business computer systems. CP/M's most standard format is one that is single sided, single density, 8-inch, 77 track and soft-sectored. If you have a Lobo Systems 8 inch drive attached to your MAX-80, your MAX-80 is capable of reading this standard release format for CP/M. Your MAX-80 can then be said to be compatible with all other machines which support that format.
The 5-1/4 inch world is not as simple as the 8 inch world. There are too many 5-1/4 inch diskette formats for any one of them to be called a "standard". MAX-80 CP/M will both read and write the same format as is used by the Xerox Corporation 820 computer, one of the largest selling CP/M computers that uses 5-1/4 inch drives.

CONTROLLER: An electronic circuit board (or box containing an electronic circuit board) that governs some device. If a box plugged into the expansion bus connector of your MAX-80 turns your appliances on and off under certain conditions, then that box is a controller. A controller usually works in both directions: it takes commands from the computer, but it also feeds data back to the computer (so that the computer can make intelligent decisions based on the facts -- or, at least, based on the facts as the controller sees them).

CP/M: Control Program for Microcomputers. (The CP/M anagram originally stood for "Control Program/Monitor", which even fewer people understood.) CP/M was created in 1974 by Dr. Gary Kildall of Digital Research. This operating system is so popular in the microcomputer industry that it has become known as "a standard microcomputer operating system". Not all microcomputers are capable of supporting CP/M, but many do. The widespread use of CP/M has encouraged the development of impressive quantities of application software which run under CP/M. The MAX-80 is sold with Version 3 of Digital Research's CP/M and can support all correctly written applications which run under that version (as well as most of the programs that run under earlier versions) of CP/M.

CRT: A Cathode Ray Tube, also referred to as a video monitor or a video screen. The instructions or text which you input (type in) through the keyboard appear on the screen, as do the computer's responses to your instructions. When you boot up (turn the power switch on or push the RESET button when the power is already on) the MAX-80 begins executing its boot ROM instructions. You will see evidence of this because the computer will start to write messages on the CRT. (If the boot up is successful, it show you the system prompt for the operating system you are using.) When you ask the MAX-80 to run a particular program, what you see on the screen will be the results of the computer's following the instructions provided to it when that program is executed (run).
CURSOR: The shape on a video screen that shows where the next changes will be made. If you type a key, the appropriate character will show up at the location of the cursor, and the cursor will jump to the next appropriate location. Usually the next location is just to the right of the last one, or on the next line if you're at the end of the one you're on. This is just a result of Western tradition, the Western cultures typically writing from left to right and top to bottom. But there is no necessity for the cursor to follow tradition. In a database program the cursor can jump from location to location, depending on where the author wanted you to fill in the next blank. In a game the cursor usually jumps all over the place — though usually all the wrong places are preferred.

DATA: Information. In computer systems this is either on disk, in RAM, in ROM, or en route between hardware components. Fundamentally, computer hardware knows no data but 1's and 0's, the on and off electrical states of its circuits. All of the information that you are interested in is only known to the computer by translation, by a program translating what you tell the computer into those 1's and 0's — and back again when you want the answers to your problems. Thus you need both hardware and software to collect, analyze and disseminate data with a computer.

DATABASE: A structured collection of data that is kept on disk and which is accessible and updatable by applications programs. The database of a business is the primary factor in how much disk storage the business needs.

DISKETTE: The diskette is the medium used for data storage in floppy (flexible) disk drives. Most business microcomputers have at least one floppy disk drive, even if they also have fixed disk drives. Diskettes are convenient because, relative to other types of storage media, they are inexpensive and portable. Your operating system arrives on diskette, as does every applications program you will use. You will be purchasing blank diskettes both for data storage and for backups for the system and applications diskettes which you be purchasing.

The diskette is a circular piece of mylar that is coated with a magnetic film. It is permanently enclosed in a square, protective jacket. The diskette is inserted in a certain orientation into the disk drive and then is read by a special mechanism inside the disk drive. The diskette can contain operating system data, data that represent text, data that
represent runnable programs, and data that represent a company's data base. In a system which has only flexible disk drives the diskette is the only means of permanent (non-volatile) storage. In a system which has one or more fixed disk drives, the diskette is used to bring programs and data to and from other computers. Applications programs are almost always purchased on diskettes.

DISK OPERATING SYSTEM (DOS): The software which tells the computer how to make use of all of its hardware components, including disk drives. If the MAX is to successfully start up, this software must be present in on a diskette or hard disk of the chosen boot drive. The machine must be booted before any other software can be executed (run). Once the machine has been booted, the disk operating system has been loaded into memory (RAM) where it resides while the machine is on. Usually the system disk must reside in the boot drive during operation, even after boot up, because the parts of the operating system that are loaded into memory may require that other parts of the operating system be read from the boot disk.

DISK SYSTEM: Your MAX-80 is a disk system. It requires a disk drive for operation. It supports a number of popular disk operating systems -- CP/M, DOSPLUS, LDOS and MULTIDOS. The diskette or diskettes that contain the operating system are called the system diskettes. When the machine is booted, the operating system on the diskette in the boot drive is loaded into memory and run. The terms "disk system", "disk based system" and "RAM based system" are often contrasted to "ROM based system". In a ROM based system the operating system (and usually one or more programs, such as BASIC) is stored on a ROM inside the computer, so no disk drives are necessary to run the computer (though they are a practical necessity, anyway, for doing serious work). See DISK OPERATING SYSTEM and ROM.

DOSPLUS: TRS-80 compatible operating system that runs on the MAX-80. The MAX-80 version is closest to that of the Radio Shack Model 4 operating system, and runs most of the Model 4 software. DOSPLUS was created and installed on the MAX-80 by Micro-Systems Software, Inc. The complete DOSPLUS documentation is provided in a manual that comes with the operating system.

EXPANSION: Addition of peripherals to a computer through expansion interfaces such as the MAX-80 expansion bus port, disk drive connectors, or RS-232C serial ports.
EXPANSION BUS: A set of lines coming out of the MAX-80 which may be used for attaching specialized controllers, data acquisition devices, speech synthesizers, etc. Most computers that do have expansion capability are expandable only through a unique bus -- i.e., equipment that works on other computers must be adapted to work on this one. The MAX-80 is no exception, though the MAX-80 expansion bus was designed to be as close as possible to the Radio Shack Model I expansion bus (given the constraints of the other circuits in the MAX). Controllers designed for the Model I will usually have to be adapted, but that adaptation should be a lot easier than if you had to start from scratch. Why the Model I? We chose that bus because a very large number of inexpensive controllers are available for the Model I, and an even larger number of articles have been written describing how to build your own controllers for the Model I.

FIXED DISK DRIVE: A permanent storage device which has non-removable media capable of storing large quantities of data very reliably. The access time (the time taken to find specific data) for the fixed disk is very fast. High access speed, coupled with very large storage capacities, make fixed disks desirable for business systems. Businesses usually require that very large data bases be readily accessible to application programs.

FIXED MEDIA: The media in a fixed disk drive. Unlike the media in a flexible disk drive, these media are permanent. (They are usually solid aluminum platters coated with a magnetic material and held in place so that they can rotate, but cannot be removed from the drive mechanism.) Since they are capable of storing far more information than a floppy diskette, the fact that the media are permanent is not a problem. Floppy disk drives are also desirable in a system which has a fixed disk drive because the flexible media can be used either to transfer information to other computers or to back up (make duplicates of the information on) the fixed disk drive.

FLEXIBLE OR FLOPPY DISK DRIVE: The device which reads and writes on flexible or "floppy" media. (See Diskette.)

FORMAT: The pattern for recording data on the storage media. The type of drive containing the media and the disk operating system being used determine the format of the media. In order for a diskette to be read under a particular operating system on
LDOS: TRS-80 compatible operating system that runs on the MAX-80. The MAX-80 version is closest to those of the Radio Shack Model I and III operating systems, and runs most of the Model I and III software. LDOS was written and installed on the MAX-80 by Logical Systems, Inc. The complete LDOS documentation is provided in a manual that comes with the operating system.

LOGIC BOARD: Printed circuit board which houses a microprocessor, video controller, communications, and memory chips and their associated circuitry. The MAX-80 contains two logic boards which are described in detail in the MAX-80 Technical Reference Manual.

MEGABYTE (MB): 1,048,576 bytes. There are 1024 Kilobytes in a Megabyte.

MICROPROCESSOR: See Processor.

MEGAHERTZ (MHz): One million cycles per second. The Z80B in the MAX-80 processes at 5 MHz or 5 million cycles per second.

MODEM: Short for MODulator DEModulator. A device that connects the MAX to a telephone line so that messages, files, etc. can be sent or received over the telephone network. For various types of modems, see the definition of Bell, above.

MULTIDOS: TRS-80 compatible operating system that runs on the MAX-80. The MAX-80 version is closest to those of the Radio Shack Model I and III operating systems, and runs most of the Model I and III software. MULTIDOS was written and installed on the MAX-80 by Cosmopolitan Electronics Corporation. The complete MULTIDOS documentation is provided in a manual that comes with the operating system.

ON LINE: "Accessible to the computer." You could say that there are 3 disk drives on line, or that there are 5 MB of storage on line.
OPERATING SYSTEM: See Disk Operating System, CP/M, DOSPLUS, LDOS and MULTIDOS.

PARALLEL PORT: The parallel port is the communications port used to interface to parallel (Centronics-type) printers.

PC BOARD (PRINTED CIRCUIT): See Logic Board.

PERIPHERALS: Any hardware device which is interfaced to a computer -- such as fixed and flexible disk drives, printers, modems, CRTs, voice synthesizers, and plotters.

PIXEL: A single cell on a screen. See Graphics Resolution.

PORT: An Outlet for a cable that connects one or more peripheral devices to the MAX-80. The MAX-80 has 8 ports (not counting the power cord inlet) -- for up to four 8 inch floppy drives, up to four 3-1/2 inch or 5-1/4 inch floppy drives, a SASI Winchester drive, a Centronics-compatible printer, one or two RS-232C serial devices (two ports), another expansion device (controller, electronics project board, etc.), and the CRT. Most of the ports require edge connectors, 34 pin for the 3 and 5 inch drives and for the parallel printer, 40 pin for the expansion bus, and 50 pin for the 8 inch and Winchester drives. The RS-232C connectors each require a DB25 (25 pin) connector, and the video requires a "phono" plug. The technical details of all of these ports are given in the Hardware Part of the Technical Reference Manual.

PROCESSOR: The device located in a computer which performs calculations and controls the flow of data to and from the different components of the system. The MAX-80 has a Z80B processor. The speed with which the computer can schedule and carry out its activities depends on the speed of the processor as well as the speed with which it can send data across communications lines.

RAM: Random Access Memory. A RAM chip provides volatile (non permanent -- i.e., when the computer is turned off, the information is gone) memory capability. The MAX-80 has 16 64 kilo-bit RAM chips which together provide 128K byte RAM. In order to function correctly, the processor needs to know where in RAM to look for a specific type of information. Therefore RAM
a particular computer, it must be in a format which is readable to that system. One advantage to all of the MAX-80 operating systems is that floppies can be formatted and written by a non-MAX-80 computer and then can be read and written in the MAX-80. For example, single-sided, single-density 8-inch, flexible, soft-sector diskettes which are formatted under the CP/M operating system on another computer system are compatible with a MAX-80 which has a Lobo Systems 8 inch floppy disk system attached.

**FUNCTION KEY:** A keyboard key that may be used for various functions -- such as commands for the operating system, typing out a full string of letters at a single keystroke, slowing down the screen display, etc. MAX-80 CP/M Plus actually provides for 12 function keys: keys F1 through F4, CTRL-F1 through CTRL-F4, and SHIFT-F1 through SHIFT-F4, each of which can have a unique function.

**GRAPHICS RESOLUTION:** The number of points of light (pixels) that can be seen on the screen, and the way in which these pixels are controlled in order to create shapes. The MAX-80 has the capability of displaying 640 pixels horizontally and 225 pixels vertically and allows a great degree of control over the images you can create on your screen. There are 192 character shapes which the MAX-80 is capable of displaying at one time. You may re-define any or all of these character shapes. Although you can not independently control each dot (pixel) of light (a method of graphics resolution control known as bit-mapping) you can achieve the following results:

You can change characters to accommodate the needs of foreign language alphabets.

You can create a small, truly bit-mapped area of adjacent characters, as long as this area is no larger than 192 character positions. This is done by providing a different character shape for each character position in the area.

You can design a special set of character shapes which, when placed in adjacent character areas, can be used to draw larger but simpler diagrams, than those that are represented by bit-mapped graphics. We call these "simpler" diagrams because in order to cover the entire screen area it is necessary to repeat some characters many times. This capability allows creation of graphics shapes such as bar charts and block diagrams, which are often useful in the business environment.
The creative system programmer can achieve very effective display images with the MAX-80's redefineable characters. The Technical Reference Manual provides the technical details about this feature.

HARD DISK: See Fixed Disk.

HARDWARE: All components of the MAX-80 except the documentation, storage media and the contents of the storage media. The first part of this manual tells how to set up your MAX-80 hardware.

HEADER: A device for connecting two sets of lines. One of the primary causes of trouble with serial devices is that they put their signal lines in different places. These devices are almost always compatible with the MAX-80. But they almost always have their lines in different places than the MAX expects. There are three ways of connecting the correct lines together: changing the lines in the peripheral device (rarely feasible), changing the locations of the lines at either end of the cable (possible, but sometimes expensive and troublesome -- because the easiest cable to use is the molded one that already has all its lines in place), and changing the lines in the MAX-80. The last method has been made quite easy, because the MAX-80 includes two removable headers (one for each serial port) whose lines may be moved with a minimum of effort. See the description in the Hardware Part of the Technical Reference Manual.

INTERFACE: In the computer industry the word "interface" is used to refer to hundreds of different types of instances in which two pieces of equipment must be able to work together or to send data to one another. This term is often used to refer to the software needed to make a particular computer system perform a particular function. And is more often used by people who like to throw computer jargon around. In this manual we will try to minimize our use of this term. However, when we do use it, we are talking about the connector or port which accepts the cable from a peripheral.

KEYBOARD: The top surface of the basic MAX-80 unit contains the key pads on which the user types.

KILOBYTE: 1024 bytes. There are 1024 kilobytes in one Megabyte. The capital K indicates the number 1024; the small k indicates good old 1000.
is divided into separate locations, each with its own address. When we speak of 128K of RAM, we mean that there are 128 Kilobytes of volatile memory (or 131,072 different addresses which the processor can keep track of. The 280 processor chip is only capable of keeping track of 64K (65,536) different addresses at one time. This means that the 128K RAM in the MAX-80 is twice as much as the processor can keep track of at one time. In order for the extra 64K of RAM to be useful, a method for trading portions of RAM in and out of 280 address range was designed. The operating system being run must know how to tell the processor which area of RAM to activate so that a particular location of memory can be accessed. Also see ROM.

RASTER-SCAN LINE: One of the lines which run horizontally across the CRT screen and which contains the pixels which are used to form shapes. There are eight raster-scan lines in an alphabetic character that is represented on the MAX-80 CRT.

RIBBON CABLE: A type of cable used to connect peripherals to the computer. Your disk system must connect to the appropriate disk system interface via the appropriate ribbon cable.

ROM: Read Only Memory. When a program is installed in it, the ROM is also called Firmware. ROMs are chips which contain programs that are executed by a computer. Usually the ROMs are activated upon boot up of the system and their code is placed into an area of memory. For instance, the MAX-80 contains a boot ROM with code that executes every time the MAX-80 is powered up or the reset button is pressed. This code is placed into a particular area of memory. It tells the MAX-80 to look for the disk operating system on the boot disk, and to load it into another location (address) in memory. Some computers have ROMs with BASIC in them. Children's toys have ROMs which are capable of activating voice synthesizers and video displays. The MAX-80 does not have a high level language in ROM. The MAX-80 ROM has been kept very simple (it only serves boot functions) so that the usefulness of the computer can be maximized. If the MAX-80 did have BASIC in ROM, the BASIC would be occupying a particular area of memory every time the system is turned on. This would limit the MAX-80's ability to use different disk operating systems that might require that the same area of memory be used for a different purpose. The MAX-80 depends on disk storage devices rather than ROM's to provide it with executable code. That code all runs from the 128K RAM in the computer. For that reason, the MAX-80 is what is termed a "RAM based system". Also see RAM.
RS-232C: A type of serial communications port commonly used to interface to printers, plotters, modems and mainframe computers. The MAX-80 contains two RS-232C ports.

SERIAL PORT: See RS-232C.

SOFTWARE: Programs which the computer can execute (load into memory and run). Programmers write series of instructions in high or low level languages. High level languages include C, BASIC, COBOL and other applications programming languages which must be interpreted or compiled before they can be executed (run as programs). Low level languages are those which consist of instructions which have a one to one relationship with instructions that the machine's processor can understand. Disk operating systems are software, and so are applications programs.

SPREADSHEET: A type of program used for doing multiple interconnected calculations. These programs were created in order to replace the accounting worksheets that contain many rows and columns of calculations, but today have much wider uses (engineering and scientific calculations, modeling in production management, etc.). They are also called Calcs, Calcsheets, Financial Modelers, etc.

SPEECH SYNTHESIZER: A device for turning computer signals (from either a parallel or serial port) into noise. Sometimes the noise even sounds like something recognizable. But never believe a fast talking computer. This device is the opposite of the SPEECH ANALYZER, which converts noise into computer signals. If you don't believe what the computer tells you, what do you think the computer believes about what you're telling it?

UTILITY: A program that accomplishes a particular narrow task, usually a task associated with the operating system or the MAX-80 itself. Thus the program TIMESET is a utility, because it allows you to easily set the clock/calender. The same is true of MAKESYS, which allows you to set the operating system up for various disk drives, printers, and modems, as well as changing the MAX-80 for your own proclivities (keyboard layout, function keys, even the cursor shape).
WINCHESTER DISK DRIVE: A specific type of hard disk drive invented by the little known IBM Corporation. The first versions of the drive had two plates (disks), each with 30 MBytes of storage (and were physically quite a bit larger than the drives we today know as Winchesters), and were called 30-30s. But soon the name changed to Winchesters. So if you run into anyone who says they used to know Mr. Winchester himself, you now know that you would be better off to get computer advice elsewhere.

WORD: Inside the computer: two bytes (16 bits) in succession. In the MAX-80's Z-80B microprocessor, each address is given by a Word. The term is still used in the normal way when it comes to human words.

WORD PROCESSOR: A program for writing, editing, rewriting, and deleting the whole mess. Since the primary reason for computers is communication, word processors are by far the most popular programs purchased for computers, especially for microcomputers.

Z-80B: A microprocessor designed and produced by the Zilog Corporation. The Z-80B is a very fast (maximum clock speed is 6 Megahertz -- 6 million cycles per second) member of Zilog's Z-80 family of microprocessors. Slower members of the family can be found in other CP/M and TRS-80 compatible microcomputers.
HARDWARE SETUP
This PDF document contains bookmarks and hot links for the Table of Contents (TOC) and Index. The bookmarks contain additional links not listed in the TOC.

To use Index links, click on the page number rather than the subject. Index listings with a page range such as '12-14' will link to the first page in the range. Index listings with 'See [subject]' (such as 'See Video') will link to the index page where the subject is located if it is not on the current index page.

Please report errors and omissions to: max-80@d30.info

PDF document created by John Ray - 02/06/2008 - V1R0
# HARDWARE SETUP

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>UNPACKING</td>
<td>HS-1</td>
</tr>
<tr>
<td></td>
<td>Save Your Packing Instructions!</td>
<td>HS-1</td>
</tr>
<tr>
<td></td>
<td>Unpacking Instructions</td>
<td>HS-2</td>
</tr>
<tr>
<td>2</td>
<td>HARDWARE ASSEMBLY</td>
<td>HS-5</td>
</tr>
<tr>
<td></td>
<td>Before You Start</td>
<td>HS-5</td>
</tr>
<tr>
<td></td>
<td>MAX-80 Back Panel</td>
<td>HS-7</td>
</tr>
<tr>
<td>3</td>
<td>DRIVE INSTALLATION</td>
<td>HS-9</td>
</tr>
<tr>
<td></td>
<td>Minifloppy (5-1/4 Inch) Disk Drives</td>
<td>HS-9</td>
</tr>
<tr>
<td></td>
<td>8 Inch Floppy Disk Drives</td>
<td>HS-13</td>
</tr>
<tr>
<td></td>
<td>Winchester Hard Drives</td>
<td>HS-17</td>
</tr>
<tr>
<td></td>
<td>Boot Drive Selection Switch</td>
<td>HS-18</td>
</tr>
<tr>
<td>4</td>
<td>OTHER INTERFACES</td>
<td>HS-21</td>
</tr>
<tr>
<td></td>
<td>Serial Ports</td>
<td>HS-21</td>
</tr>
<tr>
<td></td>
<td>Printers With Parallel Interfaces</td>
<td>HS-23</td>
</tr>
<tr>
<td></td>
<td>Video Hookup</td>
<td>HS-24</td>
</tr>
<tr>
<td>5</td>
<td>PLUGGING IN</td>
<td>HS-25</td>
</tr>
<tr>
<td>6</td>
<td>DISKETTE CARE</td>
<td>HS-29</td>
</tr>
</tbody>
</table>

LOBO SYSTEMS

HS-i

VERSION 1.1
SAVE YOUR PACKING MATERIALS!

In order to remain covered under Lobo's full year warranty, you must save all original packing materials -- and use them if your equipment is being transported for ANY reason. If the original packaging is lost or damaged, replacement materials can be purchased from Lobo Systems at cost.

The major components of your system are packaged together with the appropriate manuals, cables and software. The major components are:

* MAX-80 with CP/M and manuals
* LDOS with manual
* Monitor
* Dual 5-1/4 inch floppy system (with either 40 or 80 track drives)
* Dual 8 inch floppy system
* 5-1/4 inch hard drive
* Printer
* Modem.

If you have ordered more than one of these components and it does not arrive within two days of the other equipment that you have ordered (and we have not sent you a notification of delay), then contact us immediately. Call Lobo Systems' Sales Department. In the Continental U.S., except for California, the toll free phone number is (800) 235-1245; in California the number is (800) 322-6103.
UNPACKING INSTRUCTIONS

First, and most important, do not throw out either the shipping list or the packing instructions from ANY of the boxes that contain your system.

The list tells you not only what has been packed, but also what our shipping department was told should be included in your order. If there is any discrepancy, notify us immediately. Again, call Lobo Systems' Sales Office: in the Continental U.S., except for California, (800) 235-1245; in California, (800) 322-6103.

The packing instructions tell you how to repack the components, should you ever move or (perish the thought) have to return a component for repair. As with any complex machinery, the components of a computer system must be carefully packed (and well insured) before shipping.

BEFORE you remove your system from its shipping container, inspect it for shipping damage or mishandling (bent corners, scarred sides, etc.). If there is damage, immediately file a report with the carrier. Do not remove the component if there is any evidence of damage.

Next remove the equipment from the carton and check it thoroughly. Again, if any damage is evident, contact Lobo and the carrier immediately. When you remove each component, shake it lightly. If it rattles, it probably has been damaged in shipment.

If the system appears to be in good condition, carefully place it in an area large enough to hold all the components of your system. CAUTION: because they are ruggedly constructed, Lobo systems are quite heavy. Be careful.
(NOTE: The RMA number is necessary if any component is to be returned to Lobo systems.)
BEFORE YOU START

Be sure you have enough AC power outlets to run the components (the MAX-80, floppy disk drive(s), hard disk drive, monitor, printer, and any other peripherals in your system). Since all of these units require 115 Volt AC power, we recommend purchasing a switched power distribution box (usually a six outlet box will do). A switched connector eliminates (or at least lessens) the chore of remembering to turn on all of the separate units. (The power distribution box can be used because the MAX-80 does not require the peripherals to be turned on in any particular order -- so they might as well all be turned on at once.)
There should be no problems with current loading of the AC electrical system, as the requirements of each of these components should be quite low. The Max-80 draws about 1/4 Amp, the video monitor slightly more, a pair of 8 inch floppy disk drives draw about 1 Amp, a hard disk is usually in the same ballpark (check the specifications of the particular drive you're using), and the only serious drain is associated with the printer (again, check the specs). Consequently, the total current required by the system should only be critical on already heavily loaded circuits. If you are having problems, the only solution is to take some of the other equipment off the line.

You might also have problems with power surges and brownouts (power dropouts) on your AC lines. Power surges can damage any electronic equipment, so if you have found such problems in the past, make sure that you have a surge suppressor (available at most electronics stores) on the line. Brownouts on the line will not damage the equipment, but can cause your MAX-80 system to reset -- i.e., you lose everything you have been working on since the last time you stored your work on disk! If such brownouts are a problem with that power line, you should either try to find a cleaner line (perhaps without your electric dryer, refrigerator, etc. on it), or should invest in an "uninterruptable power supply" (UPS). The chief advantage of an UPS, by the way, is that the power can go out altogether and it still allows you enough time to save what you are doing before the system shuts down.

CAUTION: FOR SAFETY, WHILE YOU ARE CONNECTING ANY PART OF YOUR SYSTEM, MAKE SURE THE ENTIRE SYSTEM IS TURNED OFF AND EVERYTHING IS UNPLUGGED FROM THE AC POWER LINE. DO NOT PLUG IN ANY OF THE COMPONENTS UNTIL THE ENTIRE SYSTEM IS ASSEMBLED AND YOU HAVE ASSURED (see the instructions below) THAT IT IS ASSEMBLED CORRECTLY.

Before you set up your hardware, note that the MAX-80 requires at least one floppy disk drive and a video monitor. The drive is necessary because the MAX-80's operating system must be loaded from floppy disk (which is included with your MAX-80) -- that is, the operating system is not already implanted in the hardware of the machine. If you bought a complete package from Lobo, then you purchased drives that have been configured for use with the MAX-80.
HARDWARE SETUP

MAX-80 BACK PANEL

The connections on the back panel are all clearly labeled, as are the Number 1 pins of each connector. The connections are:

* Power In: connector for the MAX-80's 115 VAC cord
* Parallel Printer: parallel connector for a Centronics-type printer
* Winchester Interface: parallel connector for a Lobo-compatible hard disk drive that has a controller
* Video Out: RCA/phono jack for a video monitor
* Serial A and Serial B: RS-232C connector for a serial printer, modem, etc.
* 8 Inch Floppy: 50 pin parallel connector for one to four Lobo compatible 8 inch floppy disk drives (controller is in the MAX-80)
* 5-1/4 Inch Floppy: 34 pin parallel connector for one to four Lobo compatible 5-1/4 inch floppy disk drives (controller is in the MAX-80)

The equipment can be connected in any order, as long as the power is off. For safety's sake, do not plug in the power cord until all of your peripherals have been connected.

You may connect up to four 5-1/4 inch drives, four 8 inch drives, and a hard disk drive at any one time. The CP/M software adjustments necessary for the various configurations are described in the manual CP/M Startup.
MINIFLOPPY (5-1/4 INCH) DISK DRIVES

One or Two Lobo Dual Minifloppy Systems

The MAX-80 directly supports Lobo's single and dual minifloppy drive systems.

The first dual drive unit plugs directly into the MAX-80's 5 inch floppy connector. A (polarizing) key in the connector prevents the cable from being connected upside down. This dual drive includes a cable terminator resistor.

If you order a second dual drive with your MAX-80, or specify that the drives you order will be your third and forth drives, then your second drive configuration will have a special tongue connector. This connector taps into the middle of the first drive's cable. The tongue connector may not have a polarizing key, so be sure that the cable's dark colored edges are situated close together. This dual drive does not have a cable terminator resistor.

A two-position switch sits just above the MAX-80 minifloppy connector (inside the MAX-80 itself). Make sure this switch is switched AWAY from the numeric keypad.
Lobo Minifloppy Drives Originally Installed on TRS-80s

In these drives the selection request depends only on which cable connector the drive is plugged into. The fourth drive responds to Pin 32 of the cable for drive selection. Unfortunately, Pin 32 goes to the standard side selection line — so double sided drives are not supported under this scheme. If you now have double sided Lobo drives, be sure to follow the instructions in the next subsection.

One and only one of the drives must have an internal electrical terminator resistor. TRS-80 compatible drives from Lobo Systems have a sticker saying "NO T.R." if they are NOT terminated. If you need to check, some typical terminator locations are shown in the figure on the next page.

A TRS-80 compatible cable also is unique. It has specific pins removed from each of the connectors. Plug the computer end of this cable (the connector furthest from the other four) into the MAX-80. The edge with the dark stripe goes closest to the Pin 1 label on the back panel of the MAX-80. The next connector on the cable plugs into Drive 1 (which may or not be Drive A: in MAX-80 CP/M). The dark stripe on the cable must be closest to the slot in the drive connector. The following connectors are for Drives 2, 3 and 4, in succession. The last drive on the cable should be the one with the terminator resistor.

If you are not installing four drives, leave the unused connectors at the end of the cable free. If you are installing four drives, make sure that the two position selector switch that sits just above the MAX-80 connector (inside the MAX-80 itself) is moved TOWARD the numeric keypad, or the fourth drive won't work.
Lobo Minifloppy Drives Originally Intended for Other Systems

Use a straight through cable (a cable with all its pins intact) and configure each drive as a different drive number. To comply with this scheme, Drive 1 on the cable must respond to Pin 10 for drive selection. Drives 2, 3 and 4 must respond to Pins 12, 14, and 06 (NOT 16, as they would for a TRS-80), respectively. If a drive is double sided, it must respond to Pin 32 for side select.

Each drive must be opened up and checked to be sure that it has a unique drive select number. Locate the DIP SHUNT (which looks like an integrated circuit, but which has several metal strips across its top) and verify either that one and only one of the drive selection strips is intact or that the rest of the lines are somehow disconnected. (Perhaps the legs are bent up or broken off.) Adjust (break the strips or bend the legs) each drive's dip shunt so that it has a unique drive select number. Also check the dip shunt to be sure that MH (or HM) is intact, and that the other head load request pins (MX, MS, HS, HL, etc.) are broken away. Typical dip shunt locations are shown above.

If a Lobo drive was originally intended for a TRS-80, it has a piece of wire attaching Pin 32 of the drive's connector to some other place in the drive. Remove this wire.

Again, one and only one of the drives must have a terminator resistor, and Lobo attaches the label "NO T.R." to its unterminated drives. You must locate the terminator (a 14 or 16 pin chip plugged into a socket near the dip shunt) within each drive and remove any extras you may have.

A straight through cable (i.e., no pins removed) can be obtained from Lobo. You can also use a TRS-80 compatible cable by simply reversing (plugging in upside down) all five (or however many you are using) of the connectors. The connector furthest from the others still goes to the MAX-80. Unless you are reversing a TRS-80 compatible cable, plug the computer end of the cable in so that the dark stripe goes nearest the Pin 1 label on the back panel of the MAX-80.

The figure on the next page shows the connections for the 5 inch floppy, 8 inch floppy, and expansion bus. Note the locations of the dark stripes on each cable. Plug the drives into the other connectors; order does not matter, but make sure that the terminated drive is furthest out on the cable. Orient each connector so that the dark stripe is closest to the slot in the drive connector, UNLESS you are reversing a TRS-80 compatible cable. Make sure that the MAX-80 selector switch (located immediately above the 5 inch connector) is moved AWAY from the numeric keypad.
FLOPPY DRIVE AND EXPANSION BUS CONNECTIONS

WARNING: IF THE CABLE TO A DRIVE UNIT IS UPSIDE DOWN, ITS DRIVES WILL TURN ON AND ERASE ITS DISKETTES. TEST FOR THIS CONDITION BEFORE INSERTING DISKETTES CONTAINING VALUABLE DATA. INSERT BLANK DISKETTES AND TURN ON THE DRIVES, BUT LEAVE THE MAX-80 OFF. IF BOTH LIGHTS COME ON AND STAY ON, YOU'VE MISCONNECTED THE CABLE.

8 INCH FLOPPY DISK DRIVES

Drives Purchased with the MAX-80

The MAX-80 directly supports one or two Lobo dual 8 inch floppy drives.

The first dual drive system connects to the MAX-80's 8 inch floppy connector through its 50 line ribbon cable. The edge with the dark stripe goes closest to the Pin 1 label on the back panel of the MAX-80. (See the figure above.) A (polarizing) key in the connector should prevent the cable from being connected upside down. This first dual drive unit will include a terminating resistor.

If you order a second dual drive with your MAX-80, or specify that the drives you order will be your third and forth drives, then your second drive configuration will have a special tongue connector. This connector taps into the middle of the first drive's cable. The tongue connector may not have a polarizing key, so be sure that the cable's dark colored edges are situated close together. This dual drive does not have a cable terminator resistor.
Lobo Drives That Were Not Intended for the MAX-80

If you have purchased Lobo 8 inch floppy drives that were not packaged specifically for the MAX-80, you probably will have to jumper these drives before the setup instructions above can be followed. This jumpering involves disassembling the drive cabinet and moving pin jumper plugs on the drive's printed circuit board.

BEFORE PROCEEDING, MAKE SURE YOU DISCONNECT ALL AC POWER TO THE DRIVES.

Reconfiguring Single Sided 8 Inch Lobo Drives

Do the following to all single sided 8 inch Lobo drives:

-- Install jumpers A,B,X,Z,HL,800 and T1.

-- Place the L jumper (if your drive has one) to the -5 position, parallel to the drive's side. To avoid confusion, REMOVE ALL OTHER JUMPERS!

If you have a single 8 inch drive:

-- Add jumpers DS1,T3,T4,T5 and T6.

If you have a dual 8 inch drive system:

-- On Drive 1, add DS1. Remove T3, T4, T5 and T6.

-- On Drive 2, add DS2, T3, T4, T5 and T6.

If you have a three drive system:

-- On Drive 1, add DS1. Remove T3, T4, T5 and T6.


If you have a four drive system:

-- On Drive 1, add DS1. Remove T3, T4, T5 and T6.


Reconfiguring Double Sided 8 Inch Lobo Drives

The 150 Ohm resistor network mentioned below is a device that looks like an integrated circuit, is installed into a socket, and probably reads "Beckman" on top.

Do the following to all double sided 8 inch Lobo drives:

-- Install jumpers RS, FS, IT, 850, S2, 2S, -5, M and AF. Some drives do not have some of these jumpers. To avoid confusion, REMOVE ALL OTHER JUMPERS.

-- Add jumper IW. If your IW position has three pins, position the jumper so it connects to the large, heavy circuit foil (ground).

-- Install a new Dual Inline Package (DIP) shunt, so all eight jumpers are intact (Z, HL, A, B, X, I, R and S).

If you have a single 8 inch drive:

-- Add DS1 and the 150 Ohm resistor network.

If you have a dual 8 inch drive system:

-- On Drive 1, install DS1. Remove the 150 Ohm resistor network.

-- On Drive 2, install DS2 and a 150 Ohm resistor network.

If you have a three drive system:

-- On Drive 1, install DS1. Remove the 150 Ohm resistor network.

-- On Drive 2, install DS2. Remove the 150 Ohm resistor network.

-- On Drive 3, install DS3 and a 150 Ohm resistor network.

If you have a four drive system:

-- On Drive 1, install DS1. Remove the 150 Ohm resistor.

-- On Drive 2, install DS2. Remove the 150 Ohm resistor.

-- On Drive 3, install DS3. Remove the 150 Ohm resistor.

-- On Drive 4, install DS4 and the 150 Ohm resistor network.
Requirements for Interfacing Other Drives

Note that, if you are installing non-Lobo drives, the interfacing is entirely your responsibility. The precise pinout and timing data for the MAX-80 are given in the Technical Reference Manual. The list below is a quick summary of these requirements. The requirements of your drives must be obtained from your drive supplier.

-- When Drive Select (DSn) is asserted, Drive n is selected, its activity light is on, its head is loaded, its head positioner is energized, and its door is locked.

-- Two-headed drives respond to the HEAD0 signal for head selection.

-- The drive should reduce its write current in response to LOWI.

-- The drive should return only one index pulse per diskette revolution.

-- The drive returns two-sided status when a two sided diskette is inserted.

-- The -5 -15 jumper should be set to match the power supply.
WINCHESTER HARD DRIVES

The MAX-80 operating system software can communicate with either a Lobo 5-1/4 inch or a Lobo 8 inch hard drive. All Lobo hard drives come complete with controllers, so they can be connected through parallel cables (included with the systems) directly to the Winchester Interface on the MAX-80.

PARALLEL PRINTER AND HARD DRIVE CONNECTIONS

The cable connections follow the same basic guidelines as the floppy drive cables. A 50 line, two connector cable is used for all Lobo hard drives. Plug one end into the socket on the hard drive so that the dark stripe on the edge of the cable is nearest the notch in the hard drive edge connector. Plug the other end of the cable into the MAX-80 so that its dark edge is nearest the Pin 1 indicator on the back panel.

The controllers of Winchester disk drives have radically varying interfacing requirements, so we cannot possibly give you installation instructions for all the makes and models. The Technical Reference Manual does, however, give complete MAX-80 interfacing requirements for a hard disk controller. Unfortunately, if you are planning to connect a non-Lobo hard drive, you (or a computer technician) are going to have to figure out the requirements of your hard disk drive controller and the way to interface it to the MAX-80.
BOOT DRIVE SELECTION SWITCH

After the drives have all been connected, you still have to tell the MAX-80 which drive will be your boot drive. The location of the selection switch and the possible choices are shown on the next page.

Of the eight levers on the boot selection switch, only the three marked "6", "7" and "8" are related to the boot drive. (The others are reserved for use in some of the operating systems.) Using the figure above as a guide, push up the appropriate rockers with a small screwdriver, pen tip, unbent paper clip, or whatever -- just be sure not to harm the switches themselves. The MAX-80 is shipped with all of the rockers pointed down (so, if the MAX-80 is turned on without a boot drive being chosen, it will beep at you repeatedly).

Obviously, at first you won't be choosing either of the hard drive boot options, as the operating system is supplied to you on a floppy diskette. After you have copied the operating system over to your hard drive or to another floppy, you may want to change the boot drive selection. With the MAX-80 turned off, you can then change the switches to boot off any drive you choose.
HARDWARE SETUP

DRIVE INSTALLATION

BOOT DRIVE SELECTION

NO DRIVE. MAX-80 BEEPS REPEATEDLY

FIRST 5 INCH FLOPPY CONNECTED TO MAX-80

FIRST 8 INCH FLOPPY CONNECTED TO MAX-80

5 INCH HARD DRIVE CONNECTED THROUGH UVC

8 INCH HARD DRIVE CONNECTED THROUGH UVC

5 INCH FLOPPY CONNECTED THROUGH UVC

8 INCH FLOPPY CONNECTED THROUGH UVC

LOBÓ HARD DRIVE CONNECTED THROUGH SASI

NOTE: This is the Boot drive selection switch. It does NOT determine which other drives are on your system.
SERIAL PORTS

Serial ports A and B are standard RS-232C ports that can be used for modems, serial printers and some kinds of controller boards. The connectors are one-way, so you don't have to worry about the location of Pin 1. The standard connector used for the RS-232C interface is called the "DB25" connector. In order to attach to the MAX-80, the device's cable must have a male DB25 connector at its MAX-80 end. The MAX-80 connectors are shown below.
NOTE: printers may require either serial or parallel communications. Parallel connections are usually Centronics compatible. If your printer has Centronics compatible parallel interfacing, turn to the subsection (below) "Parallel Printer Installation". If your printer has a parallel connection, but it is not Centronics compatible, you'll have to turn to the technical description of the parallel printer port (in the Technical Reference Manual) and will have to figure out a lot of the interfacing for yourself. Obviously, life is a lot easier if your printer has one of the two "standard" interfaces.

Unfortunately, the RS-232C "standard" is infamous for being violated. For instance, the connectors of two serial printers usually match, but the lines in the connectors do different things. Your best procedure is simply to plug in the printer, modem, or whatever, and output something to it.

The software adjustments necessary for configuring the MAX-80 to recognize and talk civilly to your equipment are described in the manual, CP/M Startup.

If you have problems with your equipment, then compare the description of the MAX-80 serial ports (in the MAX-80 Technical Reference Manual) to the literature provided with your printer, modem, etc. If things just don't seem to match, we suggest that you read the excellent article "The Input/Output Primer, Part 4" by Steve Liebson in the May 1982 Byte Magazine. This article should help you figure out how to make the MAX-80 talk to your peripherals. (Or at least it will console you with the fact that the rest of the world is as bad off as you are.)
PRINTERS WITH PARALLEL INTERFACES

If your printer uses a parallel interface that is Centronics compatible, it may be directly connected to the MAX-80. If its parallel interface is not Centronics compatible, you'll have to check the specifications of the MAX-80 parallel printer port given in the Technical Reference Manual and compare them to the requirements of your printer. In that case, interfacing the two can take a bit of work.

NOTE: printers can use either parallel or serial interfacing to a computer. For serial interfacing, see the section above on serial ports.

The parallel printer connection is shown at the end of Section 3.

To connect a Centronics compatible printer to the MAX-80, you'll need a 34 line cable that has an edge connector on its MAX-80 end. Make sure Line 1 of this connector is attached to Pin 1 of the MAX-80 connector (indicated next to the connector). Most parallel cables are flat, and so an edge connector (available from any electronics store) can easily be attached to them (just make sure to keep track of Line 1). Still, plugging the connector in upside down is not dangerous -- your printer won't work, but that should be correctible simply by turning the connector over.

If you are fortunate enough to have a cable that has Line 1 clearly marked (usually by a dark stripe), then that line must be connected to the end of the port nearest the "1" marker. Once again, you can double-check the location of the Pin 1 end of the connector by checking on the notch in the connector itself; the notch is nearest the Pin 1 end.
VIDEO HOOKUP

The MAX-80 requires a standard composite video monitor. Most monitors produced these days have composite video output.

Both the MAX-80 connector and the connector on the back of the Lobo monitor require a male phono plug. The appropriate cable is included with the Lobo monitor.
Finally, after all of the hardware is connected, you can plug in the power cord(s) of your system, turn the power on and check to see that all is in order.

NOTE: Make sure that the ground prongs of the power cords of the whole system are actually plugged in (no cheating with 2 wire extension cords) and that all are plugged into the same ground line (another advantage of having a single power switch box). In many cases the MAX-80 will not boot up if the ground is not adequate. In older houses, for instance, there frequently is no ground line, and sometimes even the "groundable" adapter plugs do not work because the outlet box itself is not grounded. In such cases you should have an electrician install a decent ground to that line.
To check out your system, follow these steps:

1. Make sure all drive doors are open and that there are NO diskettes in the drives.

2. Turn on the separate units in any order. If you have all of them connected to one switched power connector, you only need to throw the single switch. Make sure all of the separate units are turned on before proceeding. The MAX-80 will beep at a regular rate.

3. If the drives have been properly connected, then the drive lights of Drive B:, Drive C:, etc. must NOT be on.

   Note: Drive A: is the drive that you set up as Drive 1 on your main drive cable. If you are using more than one size drive, the main cable is the one for which you have a system diskette (and also, hopefully, which you chose by setting the appropriate boot select switch on the back of the MAX).

4. Insert a blank diskette (or one which has no valuable information on it) into Drive A: and snap the door or latch shut. Whether or not the Drive A: light and motor were on previously, they should now turn on. If the diskette has nothing on it at all (it was last passed under a bulk eraser), the light and motor (as well as the MAX-80 beeper) will stay on. But if the diskette has been formatted, the light will soon go off and the computer will stop beeping -- though, of course, you will only get static on the monitor.

   A new diskette, by the way, may come formatted from the factory. So, when you insert a new diskette, don't be surprised if the drive light goes on, then off again, and the MAX-80 stops beeping. Frequently the formatting is nothing that either CP/M or LDOS can read, but the MAX-80 will still accept it as sensible data and shut off its beeper.

   Remember, the system diskette provided with your MAX-80 is write protected, but, for safety's sake, do NOT use it in these tests. Cables that are upside down can cause your drives to ERASE even a write protected diskette. AND LOBO SYSTEMS IS NOT RESPONSIBLE FOR DESTROYED DATA, INCLUDING THE SYSTEM DISKETTE.

   DO pay careful attention to the lights on all of the other drives. Under NO circumstances should they be on before the operating system has actually booted up.
NOTE: the indicator lights on disk drives (either hard or floppy) are ACCESS lights. They should be on ONLY when the MAX-80 is attempting to access a diskette in the drive. Thus they should be off when there is a diskette in the drive but the MAX-80 is not attempting to communicate with it.

All drives sharing a cable are affected by one incorrectly connected drive; when one drive is misconnected, you will usually find that most of the drive lights are on and the motors are running. This is why the drive doors MUST be open during first time power up. INCORRECTLY CONNECTED DRIVES CAN COMPLETELY ERASE YOUR DISKETTES!

5. If any of the drive lights are on or the wrong motors are running, either one (or more) of your connectors is upside down, or you have left an unused connector at the wrong end of the connector cable. Turn off (or disconnect the plugs to) your entire system and re-read the drive hook up instructions above. After you think you have corrected the error, turn on the system again and run the above tests. (Remember to leave the drive doors open and to use only write protected or blank diskettes.)

6. If the drive lights and motors go on and off according to the guidelines given above, you're ready to start thinking about your diskettes. Turn your system off until you're ready to start backing up the CP/M system diskette (after you go through the next section).
Before you insert your MAX-80 system diskette in Drive A: (the drive you configured to be your system boot up drive), you should become very familiar with the following diskette care instructions:

1. Keep the diskette in its storage envelope (the easily removable paper folder that covers half of the diskette) whenever it is not in use. The end of the diskette that has the long open slot should be inside the envelope.

2. Never handle the unprotected parts of the diskette. Even a fingerprint on the glossy surface can destroy data. Do not try to wipe the diskette surface; microscratches can destroy even more data.

3. Keep the diskette away from magnetic fields (TV sets, speakers, AC motors, transformers, magnets, etc.). Strong magnetic fields will destroy data on the diskettes. Weaker magnetic fields will cause the records on diskettes to become unreliable.

NOTE: the utility COPIER can be used to strengthen the magnetic records on some diskettes that have been exposed to weaker magnetic fields. If the diskette is COPIERED to itself, any data that can be read will be given a stronger impression on the surface of the diskette. But if you have reason to doubt the reliability of the records, the only way to check is to go through all of the files on the diskette, line by line.
4. Keep the diskette away from heat and sunlight. Both can warp the diskette and thus cause it to be unreadable in your drives.

5. Avoid contamination of the diskette by cigarette smoke, dust or other particles.

The heads of your drive ride directly on the surface of the diskette. It happens that the particles that make up cigarette smoke are the ideal size to catch under the heads and to score the diskette surface. Just smoke in the air can be enough for you to lose random bits of data. It is advisable not to smoke near the computer at all -- but at least to avoid smoking near the drives.

6. Use only felt tip pens to write on diskette labels. Pressing on the label with a pen or pencil will cause slight indentations in the diskette surface. So the next time the drive head passes over that area, it can skip over the data in the indentations altogether. And you've just lost your data.

7. Insert the diskette into the system drive carefully. Never force the door or latch closed. If the door resists closing, the diskette is not seated correctly; gently remove and re-insert the diskette.
8. You can prevent further recording on the diskette by "write protecting" it. This is especially good for preserving data that you definitely do not want to lose.

Unfortunately, the conventions for write protecting diskettes are exact opposites. To write protect a 5-1/4 inch diskette, PLACE a silvered (or other opaque) stick-on tab over the write protect notch. To write protect an 8 inch diskette, REMOVE the silvered stick-on tab. See the figure on the next page for the locations of these notches.

If a diskette is write protected, the system will not be able to write information onto the diskette. All masters should be write protected.

WRITE ENABLE NOTCH

WRITE PROTECT NOTCH

WRITE ENABLE/PROTECT NOTCHES
9. You can identify a single sided from a double sided 8 inch diskette by the locations of their sector holes. Note the figure below.

Single and double sided 5-1/4 inch diskettes look exactly the same. The only way to tell the difference is to read their labels.
HARDWARE SETUP

INDEX

C
Centronics.
   See Printer, parallel.

Composite video, 24

Connector, edge.
   See Edge connector.

Controller
   Board, 21
   Winchester drive, 7

COPIER Utility, 29

CP/M, inclusive, 1

CRT. See Video.

D
DB25 connector, 21

Dip shunt, 10, 11

Diskette, 29-32

Drive. See the appropriate type:
   Boot, 8 inch floppy,
   Minifloppy, Winchester.

5-1/4 inch floppy drive.
   See Minifloppy drive.

5-1/4 inch hard drive.
   See Winchester.

8 inch floppy drive, 1, 7, 13
   Jumpers, 14-16

AC power.
   See Power, electrical.

Back panel, 7, 8

Boot drive
   Selection switch, 18-19

Brownouts, 6

LOBO SYSTEMS
HARDWARE SETUP

E

Edge connector, 23
Expansion bus, 12

F

Floppy drive.
   See 8 inch, Minifloppy.

Ground line.
   See Power, electrical.

Hard drive.
   See Winchester.

Interfacing drives.
   See appropriate drive type

L

LDOS, 1

M

Minifloppy drive, 1, 7
   Jumpering, 9-12
   Installation, 9

Modem, 21

Monitor.
   See Video.

P

Packing.
   See Unpacking advice.

Panel, back, 7, 8

Parallel printer.
   See Printer, parallel.

Phono plug, 24

Port, serial, 21

Power distribution box, 5

Power, electrical, 5, 7, 25

Power surges, 6

Printers
   Parallel, 7, 17, 23
   Serial, 21, 22

R

RS-232C
   Port, 21
   "Standard", 22
HARDWARE SETUP

S
Serial port, 7, 22
Serial printer.
   See Printer, serial.
System check, 25-27

T
Terminator resistor, 9-12, 15
Test, system, 25-27

U
Unpacking advice, 1-3

V
Video
   Connector, 7, 24

W
Winchester drive, 17
Write enable, protect notches, 31
CP/M STARTUP
This PDF document contains bookmarks and hot links for the Table of Contents (TOC) and Index. The bookmarks contain additional links not listed in the TOC.

To use Index links, click on the page number rather than the subject. Index listings with a page range such as '12-14' will link to the first page in the range. Index listings with 'See [subject]' (such as 'See Video') will link to the index page where the subject is located if it is not on the current index page.

Please report errors and omissions to: max-80@d30.info

PDF document created by John Ray - 02/06/2008 - V1R0
# TABLE OF CONTENTS

1 **BACKUP** ................................................. CS-1
   - The Routine ........................................ CS-2
   - Single Drive Copying ............................. CS-5
   - What's Left To Do? ............................... CS-7

2 **CP/M PLUS** ............................................ CS-9
   - Bank On It .......................................... CS-12
   - As Usual in the Software World, No Guarantees  CS-13
   - A Brief Outline of What Lies Ahead ........... CS-16

3 **SYSTEM INSTALLATION** ............................... CS-17
   - Basic Steps ........................................ CS-18
   - The Necessities ................................... CS-19
TABLE OF CONTENTS, Continued

4 MAKESYS .................................................. CS-21
   Drive A: vs. Drive 1 ...................................... CS-25
   Installing a Floppy Drive ................................ CS-27
   Installing a Hard Drive or a Backup Floppy .......... CS-29
   Installing More Drives ................................... CS-30
   Redefining Function Keys ................................ CS-31
   Port Modification ........................................ CS-36
   Check Your Printer ....................................... CS-37
   General (Mills?) Serial Instructions .................. CS-40
   Keyboard Reconfiguration ................................ CS-45
   Default Devices .......................................... CS-47
   Cursor Shape ............................................. CS-52
   Shift Lock ................................................ CS-55
   The End of Makesys ........................................ CS-56

5 GENCPM .................................................... CS-57
   Be Cautious with This Program ......................... CS-57
   The Simple Procedure ................................... CS-59
   Notes on Manual Installation ............................. CS-61

6 CPLD ........................................................ CS-69
   Altering the Character Set ............................... CS-72
   And, Finally, the End of CPLD, Too ..................... CS-74
Even though several diskettes are required for your CP/M 3, there is only one copy of the operating system in all of that material. You, therefore, possess only one copy of a valuable commodity. You must first make sure you back it up (make copies of it) -- many times, in fact.

Start off by collecting a number of clean diskettes, all with the exact same specifications as the Lobo Original System Diskettes (all double density, single sided and the same disk size as the Original). By "clean" we mean either with no information at all on them -- or, at least, with no valuable information on them. After you are finished with this backup procedure, all of the old information on these backup diskettes will be completely gone. SO MAKE SURE THEY DO NOT CONTAIN ANYTHING YOU WANT TO KEEP.

With the backup diskettes close at hand, check that your boot drive selection switch is set to the correct drive, turn the system on, put the Lobo Original System Diskette 1 in Drive A:, and close the door. Your system will boot up with:

* Drive A: must be a floppy drive that is not controlled by a hard disk controller. For example, the backup floppy in a Lobo Systems 950 or 1850 hard drive cannot be used as a boot drive.

** This is just an example. The exact numbers given here will probably be different from the ones shown for your system; this list depends directly on the installation of the system -- 8 inch drive versions being different from 5-1/4 inch 40 track versions, which are different from 5-1/4 inch 80 track versions, etc.
Then, after a few seconds:

MAX-80 60K Banked CP/M V3.0 - [01/27/83]
Copyright (c) 1983, 84 Lobo Systems, Inc

Together with the present date and time, and, of course, the familiar A> prompt.

THE ROUTINE

Type DIR <cr> and you'll see a list of more system files and system utilities than you ever wanted. Yep, these are all part of the system -- and this is only your first system diskette. CP/M 3 is, indeed, more operating system.

Since the first task is backup (if you only have a single drive system, see the next section), type

COPIER B:=A:<cr>

and you'll get back

*******************************************************************************
* MAX-80 Format/Copy Program [12/23/83] *
* Copyright (c) 1983, Lobo Systems, Inc. *
*******************************************************************************

Insert source disk in drive A: and hit return...

Since your source already is in Drive A:, you can go directly to <cr>. Your reward will be
CP/M STARTUP

Reading - SS/DD/512
Insert destination disk in drive B: and hit return...

When you insert the disk and press <cr> you'll get back an ongoing report of the copying progress. Three messages will overlay each other on the bottom line, starting with

   Now reading...... track - 0000
   Now formatting... track - 0000
   Now writing...... track - 0000

You'll notice that the drives go through a regular pattern: first reading from the source, then formatting the destination, and finally writing to the destination. The tracks shown will jump back and forth because COPIER takes in information from a set of tracks and then goes back and formats those same tracks on the destination diskette before it writes the information out. The final message will be

   Now writing...... track - xxxx

where xxxx is one number less than the actual number of tracks on the diskette (remember that the copying started with track number 0000 ?) and the message:

   Done.
   Identical operation again? (Y/N):

We are into heavy production here, so type Y and you'll immediately get back

   Insert new diskette(s) and hit return...

Remove the destination diskette, put in another clean diskette, and press <cr> to make your second system copy. While you are waiting for the second copy to finish, label the first copy "MASTER" and make up a second label for your new "BACKUP MASTER".
When this copy is complete, you'll again find

Done.
Identical operation again? (Y/N):

And, once again, you should type Y and get the message

Insert new diskette(s) and hit return...

Note that this message says nothing about keeping the same source diskette in place. In fact, the whole of the program COPIER.COM is now in the memory of your MAX-80 and you can switch both the source and destination diskettes. So now's the time. Remove the main operating system diskette and your new copy of it -- and replace them with System Diskette 2 in Drive A: and another clean diskette in Drive B:. Press <cr> to start the process again.

WARNING: Do not reverse these diskettes; you could be writing over (completely destroying) your Original System Diskette.

After you make a second copy of that system diskette, go through the same procedure with each of the other Lobo Original diskettes. All of them.

When you have finished with all of the copies, at the question

Done.
Identical operation again? (Y/N):

answer N and the A> prompt will return. Now place your new Master diskette in Drive A: and hit reset. It should boot up in the exact same way as the Lobo Original. Type DIR <cr> to make sure. Then try the Backup Master. And, when that boots, place each of the other copies in Drive B:, in succession, and type DIR <cr>.

When you are through with all of the Lobo Original Diskettes, and everything checks out, you should seal your Original diskettes in a container and store them in a secure place. You do not need the pain and aggravation of replacing them.

LOBO SYSTEMS VERSION 2.1
CP/M STARTUP

SINGLE DRIVE COPYING

Even if you have only one drive that is the size of your Lobo Original System Diskettes, you can still make direct backup copies of the Originals. We do, however, firmly recommend that you find a second drive of that type, as the procedure necessary for single drive copying is long and tedious (and therefore more prone to error -- i.e., DISASTER).

If you insist, here is the basic single drive backup procedure: Boot up your Original System Diskette 1, at the A> prompt type COPIER A:=A: <cr>, and you will get back

*******************************************************************************
* MAX-80 Format/Copy Program [12/23/83] *
* Copyright (c) 1983, Lobo Systems, Inc. *
*******************************************************************************

Insert source disk in drive A: and hit return...

Since you are backing up the present system diskette, press <cr> without changing diskettes:

  Now reading...... track - 0000

But after just a few tracks (depending on the drive size you are using), you will be confronted with

  Insert destination disk in drive A: and hit return...

When you do, the messages

  Now formatting... track - 0000

and

  Now writing...... track - 0000

will alternate until you see

LOBO SYSTEMS

VERSION 2.1
Insert source disk in drive A: and hit return...

You will have to continue this process until the program finally reports

Done.
Identical operation again? (Y/N):

But that message is a long way away from the beginning of the process. (Single drive copying an 8 inch single sided diskette, for instance, requires 16 passes. That's 32 (!!) diskette interchanges for just one copy. And you really are willing to go through that for all those system diskettes? Twice for each? Surely you have something better to do with your life. PLEASE get another drive and do the job right.
CP/M STARTUP

WHAT'S LEFT TO DO?

Now that you have copies of your system disks, you can proceed in either of two directions.

If you have more hardware (other disk drives, a printer, a modem) to install in the operating system, or you want to make changes to the keyboard layout, the character set, etc., then you should continue on through the rest of this manual.

If you do not have other hardware to install or other operating system changes to make, then you should proceed directly to the Digital Research manuals and to the installation manuals that accompany whatever software you wish to use on the MAX. (For hints on installing software, see the manual System Utilities and Software Installation, Chapter 4, "Software Installation".) That is, if you do not want to make any other changes to the operating system, then you don't have to bother with the rest of this manual. Hooray, school's out!

Suggestion: even if you have no intention of changing anything, do make an attempt to skim through this manual, just to see what you could be changing if you desired. Lobo MAX-80 CP/M is quite a useful tool, and the ways you can customize it can make the difference between using this machine just a little bit and using it for everything you want to do. Check out the possibilities.
CP/M STARTUP
The standard operating system for the MAX-80 is CP/M 3 (CP/M Plus), the full banked version.

MAX-80 CP/M 3 is called "banked" because it has a special feature that is not found in most microcomputer operating systems.

One of the major limitations of 8 bit microprocessors (such as the Z80B in the MAX-80) is that they can only use 65,536 (called "64K") characters of information at any one time. Now, 64K is a great deal of information (equivalent to approximately 40 typewritten pages). But it still is not enough for many of today's advanced applications (especially the large databases, spreadsheets and word processors that can be used to run entire companies). Thus it is important to have even more information available immediately -- that is, available in very fast memory (RAM, random access memory) inside the computer.

*Note: If you have never seen CP/M before, the present section may be more confusing than helpful. The intent here is to introduce CP/M 3 to those who already have some familiarity with CP/M 2.2. If you only wish to install CP/M on your hardware and immediately start running some software, you should skip directly to the next section. Return here only when you have some feeling for what's going on in CP/M. (Computers are best learned "hands on". You should only bother with deep explanations when you start to get nagging worries about why the computer does those odd things in those strange ways. If you never get those feelings, all the better. Perish the thought that you'd ever become one of those insidious programmers.)
The MAX-80 was designed with double the normal amount of memory (128K) usually found in 8 bit microcomputers. But the fact that this memory is physically mounted in the machine does not mean that it is usable (by most of us, anyway). It is only usable if the operating system (CP/M, for instance) takes advantage of it.

That is what CP/M 3 is all about. Beyond all the well advertised features of CP/M 3 (the help utilities, password protection schemes, time and date stamping, advanced programming tools, etc.) CP/M 3 provides one great advantage: it automatically uses the entire 128K available in the MAX-80.

The key to much of what you are going to be doing with your MAX-80 is the amount of space your software has to work in. When the alternative methods of operation (moving information back and forth to disk drives, breaking up your files into smaller parts, using paper and pencils) are all slower, then space is speed -- and speed is the essential ingredient of computers. Computers don't do anything humans can't do; but what they can do, they do faster -- which is, after all, why you own a computer.

The trick is to somehow get the microprocessor to use more memory than it is designed to use. This requires, to put it mildly, special techniques. The most popular of such special techniques is the one called bank switching -- a procedure in which blocks (banks) of memory are moved in and out of the "active" information area of the microprocessor.

In CP/M Plus the term "banked" means that the operating system automatically (with no special instructions required of either the MAX-80 user or the software writer) switches blocks (banks) of memory into and out of the area being used by the microprocessor. When you, the owner, are using that information, CP/M has it in hand. When you are not using that information, CP/M has it in an "inactive" bank. And all switched automatically. It's as if the Z80B suddenly learned how to access more information than it ever had before. Surprising.

There is no need for the user of the MAX to know where anything is in memory at any one time. There is no need for even the applications programmer (the person who writes WonderWriter, ConnoisseurCalc and DazzleBase) to pay attention to the different banks. CP/M 3 automatically loads the additional parts of the program into the banks and switches them in and out as needed. Even very old programs can "take advantage" of the extra banks -- without the very old program writers even knowing about them. Slick.

LOBO SYSTEMS   CS-10   VERSION 2.1
Unfortunately ..., there always is an "unfortunately".

Unfortunately, the banking operation of this new CP/M makes the operating system significantly more complicated than its predecessors. Many of the system utilities, therefore, are also more complicated than their predecessors.

So, unfortunately, you, the MAX owner, are going to have to do a significant amount of installation -- at least, if you want to install anything above and beyond the minimal system (i.e., two identical drives that can run your Original Lobo Diskettes). And, if you have an elaborate hardware system, you may have to do quite a bit more installation.

The rewards, on the other hand, are much greater. You will now have automatic bank switching to run complete programs much more quickly than you've ever seen them run before. You will also have all of the other (heavily advertised) advantages of CP/M Plus: on-line help manual, disk drive chaining, passwords, time and date stamping, advanced assemblers and debuggers, and on and on. For the description of all of that, see the Digital Research manuals. For the description of the installation of CP/3 into your system (or your system into CP/M 3), here is where you begin.

* The greatest amount of work is involved in operating the Digital Research program GENCPM manually; hopefully you will be able to run this program automatically until you are much more familiar with the inner workings of CP/M 3. The complexity of what you are trying to do with the operating system is the determining factor of whether GENCPM must be run manually or automatically. If all of this is as clear as mud now, don't worry. The entire procedure is described completely below.
The Z80B central processing unit (CPU) can address (talk to) 65,536 (called "64K") characters at any one time. This is the CPU's "address space" (the amount of memory that the CPU can use).

The MAX-80's memory, on the other hand, can hold 131,072 (128K) characters. So some special procedures must be employed in order for the Z80B to make use of the full memory available.

To do this, CP/M 3 uses the procedure called "bank switching". Memory (RAM -- Random Access Memory) is divided into sections called banks, and these banks are switched into and out of the "active" (64K) address space. (During the time that one or more banks are actually usable by the CPU, they are said to be the "active memory".)

Just switching banks of memory is the easy part. The complicated part comes when the system programmer tries to make the operating system run as well as possible. If the operating system is going to do that, it must keep as much free space as possible in the active memory. The free space available to programs is called, in typical CP/M jargon, the Transient Program Area (TPA). When the operating system provides the largest TPA possible, it is using the least amount of memory for itself.

There are only three basic kinds of software that go into the memory of your machine: the data necessary for running the parts of the machine itself; the code that provides the basic environment (keyboard recognition, disk accessing, video control, etc.) for the applications programs (word processors, calcsheets, etc.); and the applications programs themselves. Much of the hardware data (especially the video memory) has to be in active memory at all times; there's no way to move it somewhere else and bring it back when you need it.

So the only way for CP/M to open up as much TPA as possible is to move as much of the "environment" as possible into the other banks of memory. That way, the system environment is not getting in the way of the programs you want to run; the programs have more machine memory (fast memory) to fill up before they have to go out to the disk drives (slow memory). Net result, much faster operation. But much more complication in the operating system. A tradeoff.
CP/M STARTUP

AS USUAL IN THE SOFTWARE WORLD, NO GUARANTEES

CP/M 3 is NOT guaranteed to work with whatever other software you have.

But what if the software works with CP/M 2.2?

That still is no guarantee. And it is not Digital Research's fault. (Nor, Heavens to Betsy, Lobo's.)

There are two sorts of programs that won't work with CP/M 3: programs whose whole intent is to go around the operating system and programs that have no business trying to go around the operating system.

The Ones That Must Avoid the Correct Procedures

There are many programs that, by their nature, must skip over parts of the operating system -- programs that "repair" damaged disks, do memory tests, examine code in memory, etc. The authors of these programs have to ignore the "correct" CP/M procedures to get what they want done. However, they usually don't go out and write all their own routines (which would make the program completely dependent on this one brand of computer). Instead, they employ whatever parts of CP/M they find useful. They make "unauthorized system calls". They create programs that depend explicitly on CP/M 2.2.

And what happens when a new version of the operating system appears, one that doesn't support those unauthorized calls? The programs don't work, that's what. They can't work. Part of the operating system software that they depend on is gone.

The only remedies here are new versions of the programs. Since these programs must depend on the exact material in a particular version of CP/M, then a CP/M 2.2 version must remain with CP/M 2.2. And a new version, specific to CP/M 3, must be written.

Before you purchase any of these programs for CP/M 3, therefore, make certain that they are written specifically for CP/M 3. Caution is the better part of safety.
The Ones That Shouldn't Have Avoided the Correct Procedures

In the beginning, Digital Research, Inc. set up The System. And, when it did, it stated clearly and loudly, for all programmers to hear, that certain functions (calls, locations, whatever you like to call them) would be supported for all time. But other functions, while they were definitely in the present version, would not necessarily be in any future version of CP/M.

Aha, so you know the plot already.

It is easy to take a shortcut. If it weren't, why would you call it a shortcut? And people take shortcuts. So, since the parent of CP/M 3 (good old 2.2) was so standard, and had been around for so long, the temptation to take shortcuts became, for some, unbearable. They skipped the official calls. They talked directly to the "almost standard" functions in CP/M 2.2. They didn't close the data files they opened. They didn't follow The System.

Then came CP/M 3.

Digital supported every call that they said they would.

But the system is now bank switched, and most of the operating system is not in active memory at any one time. So, even if an "almost standard" 2.2 function still exists, guess where it is? Right, in another bank somewhere. So where do shortcut programs go when they try to jump to those functions? What do you bet they don't go where the functions are? What do you bet they don't work?

You'd be making a very safe bet.

In truth, you will rarely find these defects in popular commercial software. Because the makers of popular commercial software have to play it safe. They have to be sure that they will still have a product to sell after the next version of the operating system comes out.

On the other hand, a great deal of less popular, non-commercial software can be expected to do just the opposite. If you are creating software that you practically (or really) give away, why go to all the effort of following all the rules?
In summary: Beware. Even well-running CP/M 2.2 software is not guaranteed to run on CP/M 3. Try it out before you spend a lot of time, effort and money on something that might not work. And, when you find something you own that definitely does not work, check with the software manufacturer to see if they have an update available.
A BRIEF OUTLINE OF WHAT LIES AHEAD

If you are familiar with CP/M 2.2, you will notice that CP/M 3 is much larger than 2.2. Most of what you will find in this additional material, however, is new utilities and features, ones which you may or may not choose to use.

Approximately half of the material, for instance, is composed of huge, complex assembly language files that give the source code of the Lobo-generated programs. If you are not an expert assembly language programmer, this is NOT the place to begin. It is included on disk only because Lobo has a policy of supporting programmers who intend to develop applications on Lobo computers.

The vast majority of CP/M 3 is located in system files (i.e., files which must be kept on disk). The two most important system files are CPM3.SYS and CCP.COM. Without those two files, your system won't even boot, much less do anything useful.

To change your CP/M 3 operating system (for instance, to install more disk drives than the two identical ones assumed in your Lobo Original Diskettes), you must use the program MAKESYS to change the CP/M BIOS (Basic Input / Output System), then run the GENCPM program (a Digital Research utility) to install the modified BIOS in the CPM3.SYS file. But that still does not give you a bootable disk. To create a boot disk you must either copy the system tracks from an identical disk (if you are fortunate enough to already have one) or run the program CPLD to install the boot routines (called the "CP/M loader") onto the system tracks of the disk.

Those are the basic steps of installation. Most of the work in installing CP/M 3 comes in the execution of each of these basic steps. All details. All described in the present manual. Hopefully not all things that you have to do.

But, any way it goes, it's the old story: do your homework and get the rewards -- but first you have to do your homework.
The Lobo CP/M 3 operating system knows quite a bit about the hardware you might attach to the MAX. But it cannot know what options you intend to use -- until you tell it what you're up to. So you will now be setting up some of the operating system parameters yourself; that is, you will be setting up CP/M for your own MAX-80 system.

Note: the procedure described below does not commit you to these changes once and for all time. You can go back through the procedure at any time to reset CP/M. Perhaps you want to install new hardware, or just want to change the keyboard layout or the shapes of the letters on the screen).

If you just want to make temporary changes in some of the parameters, some of the other CP/M utilities would probably be better (at least, easier) than altering your whole operating system. Say, for instance, you want to change the active printer from the one on the serial port to the one on the parallel port. If you have the program DEVICE.COM on the current drive, then the command

```
DEVICE LST:=PPRT <cr>
```
will do the job a lot quicker than will the procedures we are going to describe below. On the other hand, this assignment will disappear the moment you turn the MAX-80 off. If you wish a more permanent change, then you must revert to the complete system reinstallation. (Note: CP/M 2.2's STAT program is gone completely, and its functions are now divided up into a number of other programs.)

Before starting to reconfigure CP/M, you should have available as much information as you can about the devices you are planning to attach to the MAX-80. The Lobo CP/M installation programs will ask you specific questions about your equipment; you will find it very helpful to have on hand at least the literature that came with the devices. Review the questions described below to make sure that you have all the relevant answers before you start the whole process.

You should also, before beginning, either already have a good knowledge of CP/M or should read, quite thoroughly, Chapters 1 through 4 of the CP/M 3 Operating System User's Guide.

The description in the next section gives a rundown of the steps necessary in order to change CP/M. The following sections describe the procedure in more detail.

BASIC STEPS

The reconfiguration of CP/M 3 takes three basic steps:

1. Run the program MAKESYS.COM and answer all of the configuration questions it asks.

2. Run the program GENCPM.COM in the most automatic way possible (so that you have to answer as few questions as possible). WARNING: Do not fool with this program; follow exactly the directions below and in Section 5 of Digital Research's CP/M Operating System System Guide.

3. If you need to have changes made on the system tracks of a disk, or if you need to have the system tracks put on a new size of disk (i.e., you can't use COPIER to copy the tracks), then you should run the program CPLD.COM and answer the questions it asks.
Note: contrary to the practice with earlier versions of CP/M, in CP/M 3 you do not have to actually change the system tracks to make changes to the operating system. This is true because much of the CP/M 3 operating system is loaded in from the data tracks of the diskette -- from the files CCP.COM and CPM3.SYS. You can, for instance, add all sorts of new drives without ever touching the system tracks (just run MAKESYS and GENCPM, and you're done).

To find out whether or not you need to run CPLD, see the beginning of the chapter below aptly titled "CPLD".

When you are through with this basic procedure, your operating system will be reconfigured and installed -- with one exception (there's always at least one). If you are going to change your boot drive to a new kind of drive (one which is not already installed in your system), you will have to run through the first two steps of the procedure twice -- once to set up the operating system to communicate with the drive, the second time to move the boot tracks to the new drive (which is possible, of course, only after CP/M already knows what the drive is).

If you are familiar with CP/M 2.2, you'll notice a slight change in the way Plus does things. Because CP/M 3 is file-based, only the system file CPM3.SYS needs to be altered for the MAX to be able to talk to your new drives. So in CP/M 3 you do have to run MAKESYS and GENCPM, then reboot and run MAKESYS and GENCPM again. But CPLD (the program to load the system tracks) only needs to be run once -- to load the system tracks on the new size disk.

THE NECESSITIES

To create a new version of your operating system (i.e., to install new drives, new printer default, etc.) you will need the following files on the currently logged disk:

MAKESYS.COM
GENCPM.COM
CPLD.COM
GENCPM.DAT
BNKBIOS3.SPR
BNKBDOS3.SPR
RESBDOS3.SPR
The first three are the utilities that (1) change a data file that holds part of the operating system, (2) make the output from that file and two others into a full operating system data file, and (3) install the CP/M "loader" routines into the system tracks. The last four files are the data files that hold all the necessary information.

BNKBIOS3.SPR is adjusted, according to your input, by the MAKESYS.COM program. GENCPM.COM then uses the data in the adjusted BNKBIOS3.SPR, together with the data from BNKBDO3.SPR, RESBDOS3.SPR and GENCPM.DAT, to create the full operating system file CPM3.SYS. If you are planning to go through the full GENCPM program (answering all of the questions), then the GENCPM.DAT file is not necessary. That file is one of the outputs of the GENCPM program itself; it is basically a record of the parameters of the last installation of your system.

The CPM3.SYS file is used by the operating system when it boots up (remember that it is one of the files that is necessary for your system to start off in the morning). But it cannot be held accountable for the parts of your system that must be on the boot tracks of your disk.

CPLD.COM changes the boot tracks of a disk (places the system "loader" code on the boot tracks). It basically is the replacement of Digital Research's program COPYSYS.COM that is mentioned in Section 5 of their CP/M Plus Operating System System Guide. (CPLD and COPYSYS differ in a number of functions; the other functions of COPYSYS are accomplished in the other Lobo system utilities.)
After checking that you have the files MAKESYS.COM and BNKBIO3.SPR on your currently logged drive (or another drive that is already installed in your system), type

```
MAKESYS <cr>
```

and MAX will reply

```
*************************************************
* MAX-80 CPM3 Installation Program [12/23/83] *
* Copyright (c) 1983, Lobo Systems, Inc. *
*************************************************

Read BNKBIO3.SPR from which drive?
```

(NOTE: if, for any reason, you decide that you want to leave this program, just press the BREAK key or type CTRL-C. The session will be aborted and you will be returned to the operating system, with no changes made to the data file BNKBIO3.SPR.)
Typing the single letter of the correct drive ("A" for Drive A:, "B" for Drive B:, and so forth) brings back

Reading BNKBIO3.SPR...

How many drives?

Answer with the number of drives you wish to install on your system. The number can be any integer from 0 to 16 (decimal, of course). Yes, you can install 16 drives on CP/M 3. But see the comments in the Box "16 Drives".

If you answer this question with 0 <cr>, MAKESYS will default the drive configuration to the last configuration you have installed. It will then skip over all of the questions on drives and proceed directly with other modifications. (See the section "Port Modifications" and following sections below.) This is a great convenience, since if you enter the drive configuration section (by answering 1 <cr> to 16 <cr> for this question), you have to go through all of the questions on all of the drives. Quite a price to pay if you only wanted to change the baud rate on a serial port.

Recommendation:

MAKESYS runs serially (that is, the topics are brought up, one at a time, in a set order). However, you do not have to make changes in any one section just in order to make changes in another. (You can skip any or all of the sections but the one you want to change.)

We recommend, therefore, that you make changes only in one section of MAKESYS at a time, install those changes, and test them out before going on to another section. This allows you not only to see more quickly what havoc you hath wrought, but allows you to save a great deal more time when you decide that those changes really weren't what you wanted after all.
If the number you put down is in the range 1 to 16, then after you press <cr> you will see

**CONFIGURATION FOR DRIVE A:**
Hard disk or Floppy? (1/2):

Be careful how you answer this. If you are planning, right now, to follow this process (all three steps mentioned above) through to the end, then the drive you list for Drive A: must be one of the drives that is already installed in CP/M 3.

Specifically, if this is your first reconfiguration of CP/M 3, then this drive must be of the same type as your original boot drive. To install a different Drive A: you will have to go through this part of the configuration process twice, the first time to install the drive in the system (so that the MAX-80 can talk to it), and the second to change the bootup information (the information that goes on the system tracks), so that the new type of drive can boot. (You will have to run MAKESYS and GENCPM twice, then CPLD once to load the system tracks of the new size diskette or hard disk.)

**WARNING:** If you wish simply to prepare another version of the operating system as a first step to later changes, MAKESYS will let you save all sorts of versions (but they all will be called "BNKBIOS3.SPR" -- and so, of course, they will all have to be on different disks). But this is a dangerous policy. You can save an 8 inch boot drive version of BNKBIOS3.SPR on a 5 inch drive, for instance, but there is no way (short of object code disassembly by an expert) to tell what you have done. That is, you could, at any time, run the GENCPM program on this file and create a totally unbootable version of CPM3.SYS. (Symptom: when you hit reset, the boot messages come up, but the cursor freezes just before putting the A> prompt on the screen.) So be careful; it is possible to get yourself into trouble if you are not paying attention.
16 DRIVES

Can't afford 16 drives? Don't worry, the MAX-80 hardware can only talk to 9 drives at a time, anyway. (Which is still, as far as we know, the most that any 8 bit microcomputer can handle at one time.) The full complement of 16 drives has been left in the operating system primarily because of future expansion possibilities. (For instance, a local area network based on CP/M 3 might make use of the full 16 drives — and provide a means of extending the MAX-80's hardware to actually talk to all of them.)

The choice of up to 16 drives is of some convenience to people who have always wanted to have Drives 0: and P: on their systems. You can choose many more drives than you have physically attached to your computer, and fill in the questions for the other drives with phony data. Warning: CP/M 3 will work even through have a plethora of phony drives, but allocating room in the operating system for all of them takes a lot of space, and can cause more (and slower) disk accesses when you run a program. This a trick that is not advised, though certainly possible, to do.

If this is your first reconfiguration of CP/M 3 and you are presently installing Drive A:, then answer this and the following questions with a description of your present Drive A:. If you have already reconfigured for other drives, then you can describe any of them to be your new Drive A:.*

If you are installing a hard drive, answer 1 to this question and turn to the subsection below that is (appropriately) titled "Installing a Hard Drive or a Backup Floppy".

* Any drive installed must be Lobo-compatible. This means that, while it is possible to install drives from other manufacturers on the MAX-80, it is the responsibility of the purchaser to do all of the information gathering, testing, wiring, etc. The MAX-80 Technical Reference Manual gives complete interfacing details on the MAX-80, and the source code for the Lobo CP/M BIOS is included in the Lobo CP/M system diskettes. From there on, you're on your own.
CP/M STARTUP

DRIVE A: vs. DRIVE 1

New concept. We said that any of your drives can be your new Drive A:. Note that we did not say that just any drive can be your boot drive. You still are allowed to boot only from the "first" drive on each cable (that is, the drive jumpered with Drive Select 1).

NOTE: CP/M 3 cannot boot from the backup floppy in the UVC hard drive system. This is a limitation imposed not because of some incapacity of the MAX, but because of the sheer weight of operating system code. For the few people who would actually use this feature, everybody's operating systems would be made larger and slower. Sorry if this is an inconvenience, but we decided that the price was too much to pay for the small amount of advantage.

For a discussion of drive select codes, see the next subsection.

In CP/M 3 the boot drive does not necessarily have to be Drive A:. In fact, it can be any of the Drives A: through P:. Conversely, Drive A: does not have to be Drive 1 (jumpered Drive Select 1). However, the boot drive does have to be Drive 1 on one of the cables. (That is a fact dictated by the hardware, not by the operating system).

This new separation of church and state (hardware and software -- opinions differ on which is which) leads to some interesting possibilities (and some less than interesting problems). Say, for instance, you have two drives, but you set up CP/M Plus so that it has only one drive (Drive A:, of course) and that single drive is Drive Select 2. If you go through all of the steps in MAKESYS, and run GENCMP, then you will have created a system that will, indeed, boot on Drive 1, but will refer to Drive 2 as Drive A: and will have no other drive in it! That is, you will not be able to read or write to your boot drive (Drive 1) at all.

To most of us this situation might seem rather disadvantageous.

But imagine a program that is written for novices. This setup is perfect for making sure that the novice users can't go out and ruin the operating system on Drive 1. They can't read or write anything on that drive!
Obviously, however, that situation is rare, since most of us want to access all of our drives (why else have them?).

There is one basic diagnostic clue for determining whether your operating system has this "problem". When it boots up, everything goes along normally until suddenly you see a different drive light go on, just before the CP/M prompt A> comes up on the screen.

Solution? Get a new copy of your system files and reconfigure the system again.
CP/M STARTUP

INSTALLING A FLOPPY DRIVE

The present question is still

Hard disk or floppy? (1/2):

If this is a floppy drive, answer 2 and you will immediately get back

3, 5 or 8 inch? (1/2/3):

Answer 1 if you are installing a microfloppy, 2 if a minifloppy, and 3 if an 8 inch drive. No matter which you answer, the next question is

Slow or fast seek? (1/2)

Now you have to know something about your drives. The general rule of thumb is to choose slow (Number 1) if you are not sure, but fast (2) if you are installing a double sided Lobo drive. (All double sided Lobo drives are fast seek; all single sided are slow seek.) If you choose slow seek and the drive sounds raucous when it is trying to work, you can come back later and reinstall it. If you choose fast seek and the drive can't handle it, you won't be able to access diskettes on the drives -- and won't be able to boot at all, if it is Drive A:.

And the next question for this drive shows up as:

Drive select code is? (1/2/3/4):

This is asking you for the placement of the drive select jumper in this particular drive. If it is your only Lobo drive of that size, then Lobo has (unless you specifically requested otherwise) jumpered it as Drive 1 (left hand or upper drive) or Drive 2 (right hand or lower drive) in a dual drive cabinet -- or only as Drive 1 in a single drive cabinet. If this drive is a Lobo drive originally set up for a different system, the necessary jumpering
information can be found in the Hardware Setup Manual, above. If you are installing a drive from another manufacturer, check with the manufacturer for the jumpering information.

NOTE: The drive select code is the code for that particular cable. That is, if you have both minifloppies and microfloppies on the same cable, only one of them can be Drive Select 1, one Drive Select 2, etc. The 8 inch and Winchester drives, because they are on separate cables, must be treated separately; Drive Select 1 on an 8 inch cable is independent of Drive Select 1 on a 5 inch (both minifloppy and microfloppy) cable, and both are independent of Drive Select 1 on a Winchester cable. Thus, for instance, if you have two 5 inch and two 8 inch drives on your system, you will (unless specifically jumpered to be otherwise) almost always have 5 inch drives with Drive Select Codes 1 and 2 and 8 inch drives with Drive Select Codes 1 and 2.

If you are installing an 8 inch drive, the drive select code is the last question. If you are installing a 5-1/4 inch drive, one question remains:

40T or 80T? (1/2):

Answer that, and you're done -- for this drive, at least.
CP/M STARTUP

INSTALLING A HARD DRIVE OR A BACKUP FLOPPY

If you answer the first question about the drive

   Hard disk or Floppy? (1/2):

with a 1, the next question will be

   5 inch or 8 inch? (1/2):

Answer for the appropriate size and you'll get back

   Is controller UVC or SASI? (1/2):

If you have a Lobo hard drive with a backup floppy in the same cabinet, then answer 1. If you have a Lobo hard drive in a single drive cabinet, then answer 2. If you have another hard drive with a SASI (Shugart Associates System Interface) output, you still are going to have to make some changes to CP/M itself. Answering 2 here will get you closer to what you need, but it will not be close enough to work. (This drive "standard", by the way, does not guarantee complete communications, only that the drives are accessible by the same hardware. A lot of software setup must be done for each specific controller and hard drive.)

If you answer 1 (UVC) above, your reward will be

   Hard drive or backup? (1/2):

That is, is the present drive the hard drive or the backup floppy of your UVC drive system? If you answer 1, the remainder of the questions will be the same as for the SASI controller (see below). If you answer 2, the next (but last) question about this drive will be on the drive select code. See the discussion above.
If the present drive you are installing is a hard drive (whether a UVC or a SASI), the next question is

5 or 10 megabytes? (1/2):

Lobo UVC hard drives hold 8.8 Megabytes if they are 8 inch drives, 5 Megabytes if they are 5-1/4 inch. Lobo SASI drives are 5-1/4 inch drives that may hold either 5 or 10 Megabytes. The answer you give here brings up the final question for this drive -- the standard question about the drive select code. Always select 1 for the hard drive and, if you have a Lobo UVC system, 2 for the backup floppy.

INSTALLING MORE DRIVES

When you choose a drive select code for Drive A:, the questions move on to the next drive.

You will see the same questions again for Drive B:. Answer them with the appropriate information for that drive -- remembering, of course, that, if this drive is on the same cable as Drive A:, then it must have a different drive select code. The same follows for all of the remaining drives that you wish to install, whether or not they presently are attached to your MAX.
REDEFINING FUNCTION KEYS

When you have answered all of the questions about all of your drives, the questions move on to other topics. If you wish simply to install new drives in your system, or just to move the boot drive, answer N to each of the next six questions, until you arrive at the question "Drive to write to?". (See the end of this chapter for a discussion of that question.)

The next question comes up the moment you finish describing your drives:

Do you wish to redefine function keys? (Y/N):

Answer Y to this if you want to install specific functions that can be called by the function keys (F1 through F4 on your keypad).

Each of the four function keys can be used in three different ways (press CTRL-function, SHIFT-function, or just the function key itself), for a total of 12 different functions. To see the complete list of functions, simply answer the question above with a Y -- and you will be treated to the elaborate display:

FUNCTION KEY REDEFINITION SECTION

| A. CURSOR HOME | H. POSITION CURSOR | O. NORMAL |
| B. CURSOR UP   | I. CLEAR SCREEN    | P. INVERSE |
| C. CURSOR DOWN | J. CLEAR TO END-OF-LINE | Q. BELL |
| D. CURSOR RIGHT| K. CLEAR TO END-OF-SCREEN | R. INSERT LINE |
| E. CURSOR LEFT | L. CAPS LOCK TOGGLE| S. DELETE LINE |
| F. CURSOR ON   | M. WIDE CHAR TOGGLE| T. COMMAND |
| G. CURSOR OFF  | N. SLOW SCROLL TOGGLE| U. NO ACTION |

NORMAL       | SHIFT               | CONTROL             |
F1 KEY:      | (L)                 | No function         | No function |
F2 KEY:      | (M)                 | No function         | No function |
F3 KEY:      | DIR                 | No function         | No function |
F4 KEY:      | (N)                 | No function         | No function |

Field contains contents of function key. Screen functions are shown with their letter in parentheses, commands are shown in ASCII. To create a command, use SPACE to place (T) in box and hit RETURN. Command keys follow: Arrow keys position the cursor, SPACE alters values and ESCAPE exits.
The list gives all of the possible functions. Most of these are the standard "terminal" functions of the MAX-80. If, at the system prompt, for instance, you press ESC, then the asterisk, followed by <cr>, the screen will be cleared. That is just the function (1) listed here.

WARNING: This list of functions includes every terminal function that it is possible for the MAX to do. That does not mean that you will ever want to use all of them. Nor does it mean that it would ever be wise for you to use all of them. In fact, some of them are dangerous to install at all. As innocuous as the cursor functions seem to be, you must be aware that these functions are the ones that actually move the cursor on the screen. They are not the ones that are detected by your software. Thus, if you define the function keys to move the cursor, you will be able to move the cursor around the screen of your word processor, all right, but your word processor won't know where it is. Tell the program to delete something where the cursor is, and it'll delete something somewhere else. Hardly a pretty thought.

The basic commands for working in this Redefinition Section are given on the screen. When you reach the position you want with the cursor keys, use the space bar to change the functions. For each of the functions, the appropriate letter will be displayed in parentheses. To activitate most of them, you only need to leave that letter in that location when you leave this menu and install this version of your operating system.

**Position Cursor**

If you choose the Position Cursor (H) function, the function requires further input. To fully install this function, therefore, you must press <cr> after the (H) is at the correct location. This will bring up the question

**LINE? 0-23:**

which is asking you what line you want the cursor to land on when you press this function key. Note that the lines are numbered 0 through 23, the 0 being at the top of the screen and the 23 at the bottom.
Even if you do not intend to use this function, you still must answer this question to leave this mode. Even pressing the BREAK key will not help you. Don't worry. There is no harm done if you put in the most phony of numbers -- and even if you install them in your operating system. Any number larger than 23 lines and 79 columns will be accepted by MAKESYS, but totally ignored by the BIOS. So if you put in a huge number for this function key, the key simply won't work to position the cursor.

Why, pray tell, will MAKESYS accept numbers here that the BIOS can't possibly use? MAKESYS accepts those numbers because they might be needed by programmers who change the BIOS. Since we do ship the source to the BIOS with the operating system, people might want to do such things as reprogram the video controller chip for more lines. If MAKESYS only recognized 24 by 80 line screens, the programmer would be left without an installation utility. So we made MAKESYS flexible enough to accept any strange number (two digits, of course) of lines. Who knows, maybe you're about to come up with the next great video game -- that just has to have 87 lines by 94 columns?

When you answer the "Lines?" question with two digits (you must put in 01 for line 1, etc.), you will be granted another inquiry:

   COLM? 0-79:

That is, which column do you want the cursor to land on when you press this key? Again, the numbering begins at 0, so 0 is the first column and 79 is the 80th column. When you type in another two digit number, the display will immediately revert to the inverse field and you may move to another function key.

If you make a mistake with these numbers, simply press <cr> and you will be presented the "Line?" question again.
Command Line

The other function that requires additional input is Choice T, the command function.

With this function each of the twelve keys may output as many as 15 characters apiece — i.e., a command line of code. The 15 characters must include all spaces, punctuation marks, symbols, etc. — everything except for the <cr> at the end of the line. Thus, if you are tired of typing

COPIER B:=A: <cr>

you can now put the whole thing on one function key.

To move to the COMMAND option, simply SPACE your way through the alphabet until you get to T, press <cr>, any you will find the cursor at the beginning of the 15 character field allowed for that function key. If you make a mistake in your command, you won't be able to use CLEAR (CTRL-H), but you will be able to try again before you finish with this function key.

When you're finished, press <cr> and, if you have not used up the full 15 characters, you'll be presented:

CR/LF? (Y/N):

This is asking you whether you wish to have a Carraige Return (RETURN) and Line Feed placed at the end of the command. Remember that all CP/M commands (and many in other programs, also) finish with a CR/LF pair. This question is asking you, therefore, whether you want the characters you've put into the function to be a command or just a line to be printed on the screen.

Note three things here:

(1) If you want a command line longer than 15 characters, you can create it with two or more function keys. Simply do not put a CR/LF at the end of any but the last function key to be pressed.
CP/M STARTUP

Say, for instance, you want to "functionalize" your standard backup command:

    PIP A:BACKUP.BAK=B:UPDATE.DAT <cr>

Then all you have to do is put

    PIP A:BACKUP.BA

(no, the "BA" is not a typo) as one function key, and

    K=B:UPDATE.DAT

with a CR/LF as another function key -- and you have your regular backup command all done with two strokes of the function keys.

(2) You are limited to 15 characters, total, for each key. That includes the CR/LF, which counts as one character. If you reach 15 characters on the line without the CR/LF, the program won't even ask you if you want a CR/LF. If you do want one, you have to either shorten the beginning of the command, or continue the rest of the command on another key.

(3) Control keys are not allowed in the command sequence. We apologize to all the word processor and spreadsheet fans, but we had to have some keys left to operate the function key redefinition program -- and it required too much code to decide whether you really meant that control key to be in the sequence or meant it as a command in the redefinition program.

After you are finished with the configuration for each function key, use the arrow keys to move to the other keys. When you are finished with all of the keys you wish to install, press ESC and MAKESYS will move on to its next section, port modification.
CP/M STARTUP

PORT MODIFICATION

The next question is

Do you wish to modify ports? (Y/N):

This is asking whether you want to change the parameters of either of the serial ports or of the parallel printer port. If you answer Y, you will find yet another menu:

PORT MODIFICATION SECTION

<table>
<thead>
<tr>
<th></th>
<th>Serial A:</th>
<th>Serial B:</th>
<th>Parallel Port:</th>
</tr>
</thead>
<tbody>
<tr>
<td>baud rate</td>
<td>9600</td>
<td>9600</td>
<td>N/A</td>
</tr>
<tr>
<td>stop bits</td>
<td>1.5</td>
<td>1.5</td>
<td>N/A</td>
</tr>
<tr>
<td>word size</td>
<td>8 BITS</td>
<td>7 BITS</td>
<td>8 BITS</td>
</tr>
<tr>
<td>handshake</td>
<td>YES</td>
<td>YES</td>
<td>N/A</td>
</tr>
<tr>
<td>parity</td>
<td>NONE</td>
<td>ODD</td>
<td>N/A</td>
</tr>
<tr>
<td>protocol</td>
<td>NONE</td>
<td>NONE</td>
<td>N/A</td>
</tr>
<tr>
<td>auto LF</td>
<td>'OFF</td>
<td>OFF</td>
<td>OFF</td>
</tr>
</tbody>
</table>

Arrow keys position the cursor, SPACE alters values, and ESCAPE exits.

The control instructions are on the screen. If you don't have anything valuable in the other changes that you have made earlier (i.e., you won't lose anything important if you have to hit BREAK), then play around with the changes. If you do have something valuable there, perhaps you should skip this phase for now (just press RETURN without making any changes), install your other changes, and then come back later to look this section over. (Remember, to move directly to this section without making the other changes, just type 0 when MAKESYS asks you how many disk drives you have and N when it asks you about function keys.)

If you are going to make any changes to these values, first review the literature on the devices you are going to attach. If the device is a parallel printer, plotter, etc., then review the next section below. If you are working with the serial ports, then make sure that you at least skim over the section, "General Serial Instructions", contributed by our chief design engineer.
When you are finished with your changes, press RETURN and you'll get back an "Are You Sure?" question:

Port modifications correct? (Y/N):

If you answer N, this question will disappear and you will remain in the port modification routine. If you have just been playing around and don't remember how the ports were set up to begin with, it's probably best for you to press BREAK and drop out of MAKESYS altogether right now.

Note: You don't want to take the chance of saving a useless version of the operating system. If you do, it'll certainly come back and haunt you later. You wouldn't believe how much time you can spend looking for an electrical problem in your printer or modem when it is "only" a problem with the way the ports are set up. In general, don't let any changes go through unless you're sure the device you're setting up needs them.

When you are happy with the configuration of the ports, answer Y to the confirmation question and you'll be moved on to the next section of MAKESYS, keyboard reconfiguration.

CHECK YOUR PRINTER (OR OTHER PARALLEL PERIPHERAL)

If the device you're installing is a parallel printer, you will probably not have to make any changes at all. (And will botch things up if you do make some.) Note that very few of the categories in the Port Modification menu even apply to a parallel port. That's the advantage of a true standard (or, at least, something closer to a true standard) -- no work required.

On the other hand, before reconfiguring the operating system to talk to any peripheral device, you should first check that the MAX-80 actually can talk to it. After you have connected the device to the MAX-80 port, you should write a brief CP/M instruction to see if communication actually takes place.

For instance, if you are interfacing a parallel printer, your test would go like this:

1. Attach the printer cable to the printer and to the MAX-80 parallel printer printer port.
2. Boot up the MAX-80 or press the RESET button with a system diskette in Drive A:.

3. At the system prompt A> type

\[\text{DEVICE LST:=PPRT}\]

followed by a carraige return. This is the CP/M Plus instruction that directs the MAX-80 to output all printing instructions and data to the parallel printer port. (If you are familiar with CP/M 2.2, note two changes: there is no colon at the end of PPRT, and the name for the parallel printer port is PPRT, not 2.2's LPT. The first change is a facit of CP/M Plus itself; the second is limited to MAX-80 CP/M Plus -- that is, you will find no reference to PPRT in the Digital Research documentation of CP/M Plus.)

4. You will receive back another system prompt A> . Now type CTRL-P. You'll notice that the cursor stops blinking for a second, but otherwise nothing seems to happen. This is the CP/M instruction to output to the printer everything that you type. Type CONTROL P only once; typing it again will cancel the instruction.

5. Now type anything, followed by a carraige return. Try something intelligent:

\[\text{ljoriuoawj4raohawd9xzo8yal}\]

(our apologies to Shakespeare.). Of course, your operating system doesn't understand this, so when you hit <cr>, it echoes back

\[\text{LJORIUOAWJ4RAOHAWD9XZO8YAL?}\]

AND your printer should immediately start printing both sides of this engrossing conversation.

\textbf{Should}, that is. If it actually does, then you're in business, and you can go on to reconfigure the operating system.

But, as always, there are a few things that can go wrong. Here are some of the problem symptoms:

A. After typing CTRL-P, you can only type one or two letters before the MAX-80 "hangs up" -- and your MAX can only be reawakened by pressing the RESET button.
CAUSE: Your printer is not fully Centronics compatible. To interface it to the MAX-80, you are going to have to carefully check all of its interfacing requirements (pinouts, voltages, handshaking protocol, etc.) and compare them to those of the MAX-80. See the section in the Technical Reference Manual on the Parallel Printer Connector Interface.

B. Your printer insists on putting out two carriage returns (line feeds) for every one you put in.

CAUSE: There are two possibilities. Some printers have internal hardware that generates its own line feeds every time it gets a carriage return. Check your printer's documentation. There should be a way of turning that "feature" off. If there is, do so at once. If there isn't, turn to the section "Parallel Printer Connector Interface" in the Technical Reference Manual. A special switch on the MAX-80 can be used to accommodate some printers.

However, if that switch does not help, then, at best, you're going to have to do some research into cables and connector pinouts -- because your printer is not truly Centronics compatible. If you're not successful at that, then you're stuck.

But why do I have this problem now, when I know the printer works on some other computers? Because CP/M, the operating system itself, sends out every carriage return with a line feed. Your printer, which probably was built to interfact to another operating system, is not compatible with CP/M. (Unless you like double carriage returns. One person's bug is another's feature.)

The other possibility is that you presently are using a word processor that is creating the double line feeds. If you are, run the CTRL-P test we outlined above. If the problem goes away, then check your word processor's documentation for a way to turn off the extra line feeds. If it doesn't have anything, then it's time to check your printer's documentation.
CP/M STARTUP

GENERAL (MILLS?) SERIAL INSTRUCTIONS

QUICKER OATS TEST

If you are installing a serial device, it is best to check for compatibility before you change the operating system. If the device does not work right away, solve the problem(s) first, and then use MAKESYS to modify the operating system.

To see whether your serial device will work without further changes, attach its cable directly to Serial Port A and, at the system prompt A>, type

DEVICE LST:=SIOA[300 <cr>

This causes the standard CP/M printer output (list) to go out Serial Port A at the rate of 300 baud. The speed is probably wrong for your device, but it will work well enough to give you an idea whether or not the device is configured correctly.

Next, type CTRL-P (which will appear on the screen as ^P ), followed by a carriage return. The operating system will reply with a question mark, but now everything put on the screen will also be sent to Serial Port A.

If you are installing a serial printer, for instance, it should now (if it is turned on) be mimicking everything that you type on the screen. If it is not, then it is time for you to start digging through both the MAX-80 Technical Reference Manual and your printer’s manual.

A plotter or modem will act in random ways (usually they will not understand direct commands in English), but you should at least be able to find out whether they are operating. If you have a program to control them, that program can also be run, and, if it has been configured (see the program’s documentation) to communicate with the CP/M LST: device, then it should be controlling your serial device correctly (as long as you don’t hit RESET; the DEVICE setting lasts only until your system reboots -- see the discussion below on default devices).
If you do not alter the configuration headers (see the Technical Reference Manual for a description of these), then the DB-25S connectors on the back of the MAX have the following pin definitions:

![Diagram showing pin definitions of DB-25S connector]

- **1 (Ground)**
- **(8) Data Carrier Detect (DCD)**
- **(7) Ground**
- **(5) Clear To Send (CTS)**
- **(4) Request To Send (RTS)**
- **(3) Receive Data (RxD)**
- **(2) Transmit Data (TxD)**
- **(17) Receive Clock** (tied to Transmit Clock)
- **(20) Data Terminal Ready (DTR)**
- **(24) Transmit Clock** (16 times the baud rate)
### Typical Signal Uses

<table>
<thead>
<tr>
<th>Signal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmit Data</td>
<td>Serial data leave the MAX and travel to the peripheral.</td>
</tr>
<tr>
<td>Receive Data</td>
<td>Serial data come into the MAX from the peripheral. Printers that use ETX/ACK or XON/XOFF protocol also use this signal to transmit data.</td>
</tr>
<tr>
<td>Request To Send</td>
<td>Many peripherals require a signal before they will accept any data transmitted to them. The MAX turns this signal on before transmitting data.</td>
</tr>
<tr>
<td>Clear To Send</td>
<td>Many peripherals, especially printers, must temporarily block the MAX's transmitted data so their mechanisms can &quot;catch up&quot; with the faster data. If hardware handshaking is enabled in the MAX, the peripheral may turn off this signal to block the MAX's transmitter.</td>
</tr>
<tr>
<td>Data Carrier Detect</td>
<td>Some devices can tell the computer that valid receive data are coming soon. One example is a modem that detects when the telephone connection has been established. If hardware handshaking is enabled in the MAX, this signal must be on before the MAX will accept any received data.</td>
</tr>
<tr>
<td>Data Terminal Ready</td>
<td>Many peripherals must be told to send data to the MAX. The MAX turns on this signal when it expects to receive data.</td>
</tr>
<tr>
<td>Transmit and Receive Clocks</td>
<td>Very few peripherals require these signals. Both clocks run sixteen times the baud rate when the MAX is set to communicate asynchronously.</td>
</tr>
<tr>
<td>Grounds</td>
<td>Both ground lines are tied to the MAX's circuit ground.</td>
</tr>
</tbody>
</table>
Answers to common complaints.

My peripheral doesn't do ANYTHING.

Make sure that the transmit and receive data signal are not crossed. Data must travel from the MAX's TxD output to the peripheral's RxD input. For received data, the signal travels from the peripheral's TxD output to the MAX's RxD input.

If the peripheral uses explicit handshake signals to enable and block the flow of data, make sure that CTS and DCD are connected correctly. Also make sure that the MAX's software is set to accept this hardware handshaking.

Many peripherals require some of their inputs to be turned on before anything works. The RTS and DTR outputs from the MAX are used to turn on these signals.

My peripheral seems to be LOSING SOME DATA.

If the MAX isn't getting CTS, it will continue pouring out data that the peripheral isn't ready to accept. Connect CTS correctly.

My peripheral gets nothing but GARBAGE (strange data).

The MAX's baud rate probably doesn't match that of the peripheral. Also, the parity or word length of the MAX may not match what the peripheral is set to accept. Use the software configuration utility to set the MAX's options to match those required by the peripheral.

My peripheral indicates an ERROR CONDITION, but keeps working.

The parity or word length of the MAX does not match what the peripheral is set to accept. Use the software configuration utility to set the MAX's options to match those required by the peripheral.
My peripheral fouls up once in a while, but the problem IS NOT REPEATABLE.

External electromagnetic noise may be getting into the communication cable. Be sure that the cable doesn't run beside high-powered electrical equipment. Do not make the serial cable longer than sixty feet. Check (use a meter) for excessive ground loop current.

My printer locks up only when I am using a WORD PROCESSOR.

If the printer uses ETX/ACK protocol, as do many daisy wheel printers, the problem is rather awkward to overcome. If at all possible, convert the printer to accept either hardware handshaking or XON/XOFF protocol. The trouble is that ETX/ACK printers also use escape code sequences that violate the ETX/ACK protocol. The solution is to modify the word processor to communicate directly with the serial ports, instead of letting the operating system do it. The operating system simply can't anticipate these escape code sequences. See the next item for help.

I need to write a program to COMMUNICATE DIRECTLY with the serial ports.

The serial ports in the MAX are memory mapped. (They appear as if they were memory locations.)

The following procedures are written under the assumption that the baud rates and SIO chip have already been initialized by the operating system.

To transmit a byte to Port A, first wait until Bit 2 of address 0F7E5H goes on. Then write the data byte to address 0F7E4H. To receive a byte from Port A, wait until Bit 0 of address 0F7E5H goes on. Then read the data byte from address 0F7E4H.

To transmit a byte to Port B, first wait until Bit 2 of address 0F7E7H goes on. Then write the data byte to address 0F7E6H. To receive a byte from Port B, wait until Bit 0 of address 0F7E7H goes on. Then read the data byte from address 0F7E6H.

For more sophisticated control of errors and handshake signals, see the SIO data sheet section in Appendix C of the Technical Reference Manual.
CP/M STARTUP

KEYBOARD RECONFIGURATION

Do you wish to modify the keyboard? (Y/N):

Answer Y and you'll be presented with

<table>
<thead>
<tr>
<th>STRIKE KEY TO BE CHANGED:</th>
</tr>
</thead>
<tbody>
<tr>
<td>STANDARD DEFINITION</td>
</tr>
<tr>
<td>--------------</td>
</tr>
</tbody>
</table>

If you press the space bar, you will be dropped out of this section and will move on to the next. Then the only way to get back is to either finish or drop out of MAKESYS altogether and to run it all over again.

Press any other key and you will receive a report of that key's standard definition (what it began as, both in face "value" and in hexadecimal value), its existing definition (which will be the same as the standard until you reconfigure it), and the program will be ready to receive the new hex value you want. If you want to skip this key for a moment, just press RETURN and you can move on to another character (which gives you an easy way to check the hex values of any of the keys).

When you're happy with the changes you've made, press the space bar and MAKESYS will move on to the next topic. Remember, though, that these changes are not in your MAX until you save them at the end of the MAKESYS program, run GENCMP on them, and finally press the RESET button (to load your new version of CP/M into memory). You won't see the results of any of the changes until then.
Remapping Restrictions

Because the space bar serves a function in the program, you cannot remap the space bar itself. That is the first restriction on keyboard reconfiguration.

The second restriction is on upper case characters. They can't be mapped separately from the accompanying lower case characters. This is true even if the upper and lower "cases" have nothing to do with each other (such as "DEL" and the underline, or "=" and "-" ). The fact that they are on the same key means that the operating system can't separate them.

There are, by the way, commercial programs that "trap" each keyboard character you type and reinterpret it as you wish. These programs do take up memory space, do slow the MAX-80's response down at times, and do not necessarily work with CP/M 3 (check before you buy). Most of these are limitations that are acceptable in an application program, because if you don't like the program you can always throw it away. They obviously are not acceptable for an operating system, so the operating system does not do that kind of trapping.

The third restriction is with most control characters and all multiple character sequences (ESC *, ^X ^S, etc.). MAKESYS can't remap them. It can remap the few control characters that are available on the MAX-80 keyboard as single stroke characters -- ESC, the cursor arrows, CLEAR, RETURN and BREAK.
DEFAULT DEVICES

Either answering N to the question about modifying the keyboard, or striking the space bar while you are in that section, brings you the question

Do you wish to modify default devices? (Y/N):

Answer Y if you want to change which input and output devices (other than disk drives) that your MAX normally talks to. This applies to the monitor, serial and parallel port inputs and outputs.

These are called "default" devices because you can temporarily (as long as the MAX is not reset or shut off) change any of these defaults. In CP/M 3 the command is (at the system prompt)

DEVICE logicalname:=physicalname

on any drive that holds the program DEVICE.COM. For instance,

DEVICE LST:=SIOB <cr>

sends the printer output to Serial Port B. But, if you reset the MAX, this setting will disappear and the output will go to the default setting (whatever it was before you gave this command).

Before you use MAKESYS to reset these defaults for a new device, first set up the port for that device (use the port modification section above and install the necessary changes by completing MAKESYS, running GENCPM and rebooting), and then temporarily change the default ports by running DEVICE. This is especially important if you are changing the console (monitor and/or keyboard) for your system. If you set the defaults incorrectly, the operating system simply won't run. (It'll be waiting forever for inputs from devices that aren't there.)

If you just want to check what the present device settings are, at the system prompt type
CP/M STARTUP

DEVICE <cr>

and you will get back a list of the present default devices.

For more information on how to run DEVICE.COM (or just to read the list of default devices), see Section 5 of Digital Research's CP/M 3 Operating System User's Guide.

The physical device names you'll need for the MAX-80 are:

- CRT -- The terminal (including video and keyboard)
- SIOA -- Serial Port A
  (closest to the outside edge of the Max)
- SIOB -- Serial Port B
  (closest to the center of the Max)
- PPRT -- Parallel Printer Port.

If you type Y to the question about modifying default devices, you'll receive back

DEFAULT DEVICE SELECTION

NOTE: More than one physical device may be selected for each logical device. Choose from the logical devices shown and terminate input with [ESC]. Input only devices will not be accepted for output vectors and vice versa.

PHYSICAL DEVICES AVAILABLE ARE:

A. CRT I/O  B. SIOA I/O  C. SIOB I/O  D. PPRT 0

Console input:

The physical devices which may be connected, interestingly enough, are the same as those that you can use with the DEVICE command. Note the "I/O" after all but one of the devices. This means that all of the devices except for PPRT can be used for
either input or output (or both). PPRT is just the parallel printer port, which is set up only to send data, never receive data.

The most troublesome of these devices to understand is the CRT. This name (which usually stands for "Cathode Ray Tube" — i.e., the video monitor) here refers to all of the “terminal characteristics” that the MAX has. For instance, the MAX-80 uses certain codes (independently of the operating system) to clear the screen, sound a beep, etc. Thus it is a terminal. If you wish to replace this terminal with another (say, attached through Serial Port A), then CP/M allows you to do so. Simply set up the "Console" input and output to be that serial port, and you're off and running on another terminal (after this version of the operating system is installed, of course).

The screen above is presenting you with one of the "logical" devices and asking which "physical" device or devices are to be correlated with it. If you wish to leave this device set to whatever it was set to before, simply press ESC and the next logical device will come up. If you wish to change the physical device from what it was before, choose the letter corresponding to one of the other physical devices on the list.

The term "logical device" is just CP/M's way of saying that it has, in its operating system, a function that does certain things in certain ways. However, taken alone, the function is just that — a function in an operating system. A function is a tool of logic (mathematics being a branch of logic), and so a function is a "logical device". Now this logical device can work with a physical (real) device, such as a port on the MAX. But, until you tell it which port (and attach something, say, a printer, to the port) this function is just a small section of CP/M, sitting there with nothing to do.

The logical device that comes up first is Console Input (CONIN: in CP/M 3 jargon). If you wish to change the input device (which right now is probably the keyboard on the MAX-80), choose A, B or C. You can't choose D because that is only for output, and you are now being asked about an input logical device.

NOTE: choices here can't be reversed. If you make the wrong choice, the only way out is to press BREAK and start MAKESYS all over again.
If you don't want to choose a new Console Input, press the ESC key and the Console Input device will remain with whatever physical device it had before. If you are presently typing on the MAX keyboard, then that device (the physical device "CRT") is the one you will still have.

If you do choose one of the options, notice that the appropriate name is displayed with a comma after it. The comma is to indicate that you can have more than one input device on the system at any one time.

No, this does not make CP/M a "multi-user" system. All of the terminals here have to be doing the same thing. The primary advantage of this setup is that you can have multiple inputs (say, for a classroom in which more than one person should be responding to a question) to the same program, without everyone crowded around the same keyboard.

Warning: if you want to have multiple input devices, remember that you do have to list the MAX-80's input device (CRT) explicitly. Otherwise, the MAX keyboard itself will not be able to input.

When you are finished with the Console Input devices, press ESC and the prompt line will be replaced with

Console output:

That is, where do you want the output to go? If you want it to go to some other terminal, then press the key corresponding to that port. If you want multiple outputs, make certain that, if you want one of them to be the MAX's video monitor, then you also choose CRT. Pressing ESC brings in the next logical device in order.
In all, there are five logical devices (listed here with their official CP/M 3 names):

- Console Input  CONIN:
- Console Output  CONOUT:
- List Output  LST:
- Auxiliary Input  AUXIN:
- Auxiliary Output  AUXOUT:

The List Output is usually the printer output, though it could also be to a plotter or other output device.

The Auxiliary Input and Output are for inputs and outputs separate from those that go to or leave from the console. You might want, for instance, a different output going to another terminal, one that does not show up on the home (MAX-80) terminal. And you might want a program to accept different inputs, and not have those inputs interfered with by the inputs from the MAX. So you can set up the second terminal to be the Auxiliary Input and Output.

No, this still is not either multi-using (different terminals running distinct programs), nor even multi-tasking (a single terminal running more than one program). The MAX is still only running one program at a time and responding to only one terminal at a time. But a clever programmer can certainly make it seem as if more than one person is doing more than one job at a time. We wouldn't be suprised to see a few of these programs around before long.
Some people want to have an "underline" cursor -- just like the terminals back at the office. Others just like to have a personalized cursor -- just unlike the terminals back at the office. Choose your poison. The MAX-80 cursor is now fully configurable.

Prefix: it is quite difficult, while you are changing the cursor, to imagine what the result will really look like. It is best, therefore, to install all of the other changes you want to make before going into this section. That way, when you are happy with everything else, you can simply enter MAKESYS, move directly to this section (by answering N to all of the other sections), make your changes, and move directly out to install the revised version of the operating system. (Yes, even changing the cursor means revising the operating system.) And remember, you have to reboot (press the reset button) to see the changes you've made.

The cursor is just two images that alternate. One is always present at first -- until a couple of seconds lapse, when the second image starts alternating with it. If you want to change the shapes of either or both of these images, just type Y in answer to the question

Do you wish to alter the shape of the cursor? (Y/N):

and your comeuppance will be

CURSOR SHAPE MODIFICATION SECTION

This section allows you to alter the shape of your cursor. The arrow keys move around the shape, SPACE alters the values, and ESCAPE exits. A 1 means the pixel will be lit; a 0 means it will be dark.

Here is the main cursor shape:

```
11111110
11111110
11111110
11111110
11111110
11111110
11111110
11111110
```

LOBO SYSTEMS

CS-52

VERSION 2.1
The main cursor shape is the one that is held on the screen for a couple of seconds after your last input. It is not the first boot up cursor shape. (That shape is inherent to the operating system.) But it is the shape that you will see most often. You can now make it anything that you can put on an 8 by 8 matrix of points.

In your designs, remember a couple facts from the good old days of changing your character set: lit pixels are stronger than unlit ones, but significant lines are always created with two or more pixels side by side.

The cursor has one characteristic that distinguishes it from all of the other characters on your screen: it is composed of two shapes that alternate regularly. This feature can present interesting visual images, if not illusions. Make each of the images a straight diagonal across the character area, for instance, and people would swear that the images are curving while they are alternating (the old bent pencil trick).

When you are finished with this shape (at least, for the moment), press ESC and the question will come back

Finished? (Y/N):

This is just asking if you hit ESC by mistake. If you did, just type N. If you didn't, a typed Y will bring you the next question:

Save this fine shape? (Y/N):

NOTE: the blinking cursor you are looking at is not the shape that you will be saving. The cursor that is presently on the screen is just the one that is installed in the operating system. The question above is asking if you want to save the shape you have just created. Whether you do or don't, the topic will move on to the second shape (that alternates with the first). Never fear, you can return to the first shape without even leaving the cursor modification section of MAKESYS. But answering N to this question will lose the work you have just done, so type Y if you have done anything you want to save.
Type either Y or N to the question above and the program will move to the second cursor shape:

Here is the alternate shape:  

```
 11111110
11000110
11000110
11000110
11000110
11000110
11000110
11111110
```

Again, you can make this anything you can fit into an 8 by 8 matrix. (Always wanted to learn matrix mechanics? Here's your chance.)

When you are finished with this section you are treated to the same two "Are you sure?" questions:

Finished? (Y/N):

Save this fine shape? (Y/N):

No matter which way you answer the second question, the next line is

Now are you finished? (Y/N):

If you answer N, you'll return to the first shape for more changes. If you answer Y, you'll move on to the next question. If you just want to return to the alternate shape, you'll have to go back to the main shape, then press ESC, Y and Y to move up to the alternate shape again.

But don't spend too much time theorizing what the two shapes will look like together. The best procedure is simply to finish up MAKESYS, run GENCPM AUTO, and press the reset button to see what you've done to yourself now.
SHIFT LOCK

And so you finally arrive at the last installation question. Finally.

Caps lock upon boot?

This question is primarily a convenience to BASIC (and a few other computer language) programmers and to people who use a lot of BASIC programs. It is asking whether, when you turn on the MAX in the morning, it should start off with all capital letters (the alphabetical keys only -- not the numbers or other symbols). This is convenient to BASIC programmers because most versions of BASIC recognize only caps, and will do odd things when you use lower case letters.

Ever see a lowercase file in your directory? Can't read it or even erase it, can you? That's because it was created by a BASIC programmer who forgot to press the shift lock key. The only way you can read that file (without special machine language programming tricks) is to run BASIC, load the file (using its lower case name, of course), and list it. A directive to the operating system to start off in upper case letters can save a lot of hassle.

Want to get your lowercase letters again? Simple. Press the F1 key; it still works as a caps lock toggle (unless you have, in an earlier session, reconfigured it with the function key redefinition described above).
THE END OF MAKESYS

Whether you answer the caps lock question Y or N, you will get back the message

File modifications complete...
Ready to write new file...
Drive to write to?

The moment you type in a drive letter, MAKESYS will go out to write the revised information on the disk (no <cr> required). If you choose either a non-existent drive or one whose disk is full, MAKESYS will abort and return you to the operating system. You will lose all of the work you have done!

So, before typing in a drive letter (A, B, etc.), make sure that you choose a drive that CP/M already knows by that letter. Remember that any disk drive changes you have made in this session are unknown to the operating system until they are put into CPM3.SYS (by the program GENCPM; see the next chapter) and you have rebooted the system. Next, make sure that the disk in the drive (whether a diskette or a hard disk) is one that has enough space to accept an 8K file. Again, if it can't, you will lose everything.

When the new version of BNKBIOS3.SPR is successfully saved, you can move on to the second step of the process, generating a system file.
Fortunately, the second step in this process is usually much easier than the first. Unfortunately, it is easier only as long as the standard (default) answers to its questions are adequate for your purposes. The simplest way to see whether or not they are adequate is to accept the default values and see whether the resulting system meets your expectations.

The defaults will **not** cause problems talking to disk drives, reading your keystrokes, or interfacing with your modem. The primary problem will be in the speed of operation (some disk accesses, screen writes, etc. not being as fast as they could be).

**BE CAUTIOUS WITH THIS PROGRAM**

On the other hand, not knowing what you are doing when you answer the questions in GENCPM can be disastrous to your operating system. If you decide to answer some of the questions yourself, make absolutely certain that you understand every step of the program, and what it is doing with the memory allocations, hashing, and so forth. Read Section 5 of the CP/M 3 Operating System System Guide thoroughly. If you don't make the correct decisions in GENCPM, you can very easily have part of the operating system overwriting other parts -- and nothing works that way.
Another Word of Caution

An important warning: when it starts up, GENCPM.COM destroys the file named CPM3.SYS that it finds on the logged in drive. That is your system file! Without it you won't be able to boot. Theoretically, you're still all right, as long as you go through the whole GENCPM program, because at the end of the program it generates a new CPM3.SYS on the logged drive.

However, say, for instance, that in the middle of GENCPM you decide you don't like some of your earlier answers. You can't go backwards, so you decide to drop out of the program. But now CPM3.SYS is gone! (This is especially deceptive, as CPM3.SYS still appears in your directory.) If you make the mistake of pressing RESET, you'll get the nice message

CPMLDR error: failed to read CPM3.SYS

and nothing more. Your operating system is no more. The CPM3.SYS that remains on the disk is unusable (destroyed).

So, the first thing to do in running GENCPM is to make sure you have a complete backup of your present system disk, with a recent version of CPM3.SYS on it (so you don't have to go through the whole reconfiguration process again just to get back to where you were when you started this session).

On the other hand, even after you have dropped out of GENCPM, you can still run GENCPM (and even MAKESYS) again and this time follow it through until you create a new system file. Just don't try to reboot until the process is complete.
CP/M STARTUP

THE SIMPLE PROCEDURE

First make sure you have the files GENCPM.DAT, BNKBIOS3.SPR, BNKBDSOS3.SPR and RESBDOS3.SPR (as well as, of course, the program GENCPM.COM) on the logged in drive.

Then the simplest (and most foolproof) way of generating a complete operating system is to type, at the system prompt,

GENCPM AUTO <cr>

This will deliver the message

CP/M 3.0 System Generation
Copyright (C) 1982, Digital Research

and you'll notice a lot of disk drive whirring and clicking going on. After a minute or so of this activity, the program will announce

*** CP/M 3.0 SYSTEM GENERATION DONE ***

and return you to the system prompt. You're done. That's all that's you need to have a new CPM3.SYS generated.

Now, if you're the sort that doesn't like programs that do things when you don't even get a chance of understanding what's going on, you can safely type

GENCPM AUTO DISPLAY <cr>

and you will be presented with a display of the questions GENCPM asks and its default answers to those questions. Going by a little fast for you? You can always rerun it; it'll only do
everything all over again. And, this time, you can slow things
down with the F4 key, though that's usually not slow enough.
Better, you can stop the screen altogether by pressing CTRL-S
(CONTROL-S). But remember (you've read your Digital Research
CP/M Operating System Guides, haven't you?) that CTRL-S is no
longer a toggle switch (which it was in CP/M 2.2). To start the
display again you're going to have to press CTRL-Q.

If even auto display is not enough for your inquisitiveness, you
can simply type

GENCPM <cr>

and "default your way" through the program (i.e., you keep
hitting RETURN -- taking the default values -- until the program
finishes).

But that's as far as you should go without really understanding
what's going on in CP/M 3. If you really want to optimize your
system, now is the time to start reading those Digital Research
manuals.
NOTES ON MANUAL INSTALLATION

If you haven't read Digital Research's instructions on GENCPM, then don't read this. It'll only make the instructions seem more complicated than they really are. GENCPM, as most things, is quite easy to run once you know how. It's getting to know how that's the problem.

(1) Until you understand GENCPM, use GENCPM AUTO as much as possible. When you do understand what the GENCPM program is doing, then you will probably want to run GENCPM just to optimize the system for your own use.

(2) Make sure the file GENCPM.DAT is on the logged in drive and use it in the GENCPM program. (Answer Y to the GENCPM question whether you want to use GENCPM.DAT for defaults.) This file will not only save you a lot of time, it will also give you a lot of default answers to questions that you won't know how to answer otherwise.

(3) The MAX has four 32K banks of memory. The memory segment table includes only two of these banks, because Bank 1 is the area used for most of the TPA and Common is the bank that is always resident in memory. You must, therefore, tell GENCPM that there are two memory segments. Fortunately, that is a default value given by GENCPM.DAT. Because one bank (Common) stays in active memory at all times, it always retains the same addresses: 8000H through FFFFH. The other three banks are switched in and out "under" this bank (i.e., in and out of the area with the addresses 0000H through 7FFFFH).

(4) The cryptic message

    CP/M 3 Base, size, bank (3B, 45, 00)

means that you are presently talking about Bank 0, that the base address of CP/M in this area is 3B00H, and that CP/M extends for 4500H bytes. Add 3B00H and 4500H and what do you get? Right, 8000H. But remember that Address 1 of CP/M in this area is 3B00H, so the last address is really 7FFFFH. Exactly as it should be.

LOBO SYSTEMS

VERSION 2.1
When GENCPM asks

Enter memory segment table:
Base, size, bank (00, 3B, 00) ?

it is again asking you about Bank 0, only now about the part that remains after CP/M has taken its share. Note that this area starts at 0000H and continues for 3B00H bytes -- or to address 3AFFH. That works out to the whole bank. Perfect.

Not so perfect, however, if the suggested base and size do not add up this way (i.e., if they don't cover the appropriate area in memory), which they just might not do. If you installed a number of drives in MAKESYS, for instance, you're likely to get back in GENCPM:

```
CP/M 3 Base, size, bank (36, 4A, 00)
```

Enter memory segment table:
Base, size, bank (00, 3B, 00) ?

Note that CP/M 3 is now bigger (4A00H vs. 4500H), and, because of that, its base address has been extended down further -- down past the top of the suggested memory segment. What are you going to do now?

Why do they inflict such questions on me? Me, of all people?

GENCPM asks these questions because they allow people (programmers usually are people, after all) to change things for their own use. Say, for instance, you were one of those assembly language types who liked to write directly to memory. Then setting the base at 05, say, would give you 5000H bytes (12.5K) that are completely protected. No matter what else CP/M does with its fancy bank switching, it will never go into that area, because you have just told it that that area doesn't even exist.

Neat. For programmers, anyway. If they don't want to protect any space, they just enter

```
00, 36, 00 <CR>
```
and go on to the next question, knowing that they have made maximum use of the available memory.

For the rest of us, we just have to press the RETURN and hope for the best. Fortunately, GENCPM is quite forgiving here. If the numbers overlap with the CP/M segment listed earlier, GENCPM will just chop off this area until it matches. And your report will be:

```
ERROR: Memory conflict - segment trimmed.
Base, size, bank (00, 36, 00) ?
```

Aha, GENCPM has adjusted the values to the optimal values. Quick, accept this by pressing RETURN.

(6) GENCPM almost always brings up the next segment as

```
Base, size, bank (10, 70, 02) ?
```

We are now talking about Bank 2 (getting the hang of it?) and the base and size do, indeed, total up to 8000H bytes -- the base address being 1000H and the region extending from 1000H up 7000H bytes to address 7FFFH.

But, wait, aren't we being gypped 1000H bytes down there at the bottom?

Right you are. That's where the memory image of the CCP goes. DO NOT CHANGE THIS BASE ADDRESS. CP/M frequently goes straight to this area and tries to execute whatever is there (it's supposed to be the CCP, after all). Overwrite this area and your whole system will crash. (Talk about threats...)

(7) Enable hashing only for drives that you actually have. Every hashing table takes space, and that space comes out of the buffer space of Bank 2. If you put in a lot of drives and have hashing for all of them, you can severely limit the space available for drive buffering (and thus restrict much of the utility of your whole CP/M 3 system). Hashing, by the way, is a procedure for sorting directories for much quicker access -- much quicker both when you say DIR and when you or one of your programs tries to access a file on the drive.
The trickiest part of GENCPM is the allocation of buffers for the drives. These buffers are the way (the only way) that CP/M allows standard programs to use the other banks of memory. So they are very important to the overall operation of your system.

A buffer is an area in memory where CP/M may place programs or parts of programs. Since machine memory (RAM) is much faster than disk memory, a program will run faster when more of its parts are available in machine memory.

How do you get more of a program's parts into machine memory? By setting up the buffers so that they are available to the right drives at the right times. And you do that by answering the GENCPM questions about the buffers for each drive.

But those questions are just what makes the buffer allocation the toughest part of GENCPM. They are the toughest because there are no cut and dried answers to them. (There never are universal answers to hard questions.) The answers all depend on your own use of the equipment.

Basically, you want to keep as much as possible of each of your programs in machine memory. But this depends on which drive you use to bring in those programs. You have to allocate the most buffers to the drives that you use the most. So the decision is easy if you tend to use one drive almost exclusively, but hard if you tend to use programs from several drives.

The typical report telling you how much space you have to allocate looks like

Setting up Blocking/Deblocking buffers:

The physical record size is 200H:

Available space in 256 byte pages:
TPA = 00ECH, Bank 0 = 0036H, Other banks = 0044H

Number of directory buffers for Drive A: (#4) ?

Note first that the physical record size is 512 bytes, but the space is given in 256 byte pages. It doesn't do you a lot of good, therefore, to allocate odd numbers of pages; the last page would only be seen by the operating system as a partial record, which it can't use.
This report tells you that the TPA is ECH pages -- which is 14 x 16 + 12 = 236 pages or 59K. Pretty good.

You can allocate some of this memory to drive buffering, but you should never do it. The TPA is the most used memory in the machine. If some of it is given up to drive buffering, then the program itself will have less space to work and either won't work at all anymore, or will have to do more disk or other bank accesses. Net result: loss.

The report also tells you that Bank 0 is 36H pages. Exactly right. That is what Bank 0 was trimmed down to just a few moments ago.

And the other banks hold 44H pages. Now the only other bank is Bank 2, and that used to have 70H pages. What happened? Well, remember all those drives you enabled hashing on? Those hashing tables had to go somewhere. And there they are, all 26H of them, on Bank 2. Doesn't leave you a lot of space, does it?

Note: if you now decide that you have made a mistake somewhere earlier, you can press BREAK to get out of GENCPM. But remember that your CPM3.SYS file is destroyed, so you can't boot up off this diskette. You can, however, run GENCPM (and even MAKESYS) again, as long as you don't try to reboot.

Now the goal is to use up all of the remaining space for drive buffering, and to put the most buffering where you have the most need.

Directory buffers and data buffers come from the same areas of memory, so a lot of one means fewer of the other. You almost always will want to have many more data than directory buffers.

Directory buffers always come from Bank 0; data buffers come predominantly from Bank 2, but can be placed in Bank 0 when space is available.

You can attempt to accept the default values as much as possible, but you usually will get into a loop in which GENCPM retains the same values for each drive and is unable to allocate either the directory or data buffers. The only way out of this is to reduce some of the buffer space somewhere until GENCPM can allocate the space demanded.
(10) A UVC Winchester (Lobo Winchester with backup floppy in the same cabinet) must be set up with 256 byte sectors. When you install a UVC drive in MAKESYS, this record size is written into the file BNKBIO3.SPR. So, when you run GENCPM, this is the record size that shows up for both the UVC Winchester and its backup floppy. Take this difference into account when you are allocating buffers. You'll notice, for instance, that allocating, say, four buffers of 200H records takes eight pages from the available banks; but allocating four buffers of the 100H records associated with the UVC takes only four pages from the available banks. (Obviously, since that is all the information that is present in four 100H pages.) Thus you are only allocating half as much information space when you allocate the four UVC buffers.

(11) You'll note that when you allocate buffers, sometimes one extra page seems to disappear from the available space in memory. The amount of space (in 256 byte pages) that should disappear is two times the number of 200H buffers you allocate and one times the number of 100H buffers you allocate. But, at times (though not always), a spare page (100H) seems to disappear. The page disappears because of the overhead required to control the buffers (that's what overhead is for, after all). But the page does not disappear every time you allocate buffers because GENCPM (mercifully) tries to optimize the space it takes up. Thus sometimes it can stuff more than one control block (its official name) into one page of memory; so no extra space is taken. And sometimes it can't; so you lose more than you thought you would.

For an explanation of all of this, see the description that accompanies Figure 3.1 of Digital Research's CP/M Plus Operating System Guide.

This extra page of overhead, by the way, makes it impossible to predict exactly how much space you yourself can allocate (since some of the available space disappears at random times). It is best, therefore, to always plan (you do plan, don't you) to leave three or four pages extra at the end. You'll find that two or three of them will disappear. But at least you won't come up short.
(12) Whenever you are asked

Allocate buffers outside of Common (Y) ?

ALWAYS answer Y. If you answer N, not only will your TPA will be reduced, but files can actually start to overwrite some of the system code in upper memory. VERY dangerous.

(12) Review the Digital Research description of buffer sharing. In larger MAX-80 systems (more and larger drives) that is usually the only way to go. To share buffers, assign buffers to one of the first drives that GENCPM lists, then, on some later drive, assign 0 buffers. A question will immediately come back

Share buffer(s) with which drive (A:) ?

You will only be allowed to specify a drive to which you have already allocated buffers. If possible, choose a drive that you think might presently have too much buffer space.

Sharing buffers is just what the name implies. The same buffer space is used for two or more drives. When you access the second drive, you frequently are replacing (destroying) the buffers set up for the first. So, once again, you have to be careful. If buffers are shared between two very active drives, each time you (or your program) read from one of the drives, you'll be wiping out the buffers of the other. This can actually make the operation slower than if you had no buffers at all.

(13) When you have reached the end of GENCPM, it gives you the load map that can be put up on the screen at the beginning of every reboot. Here's one possible report:

<table>
<thead>
<tr>
<th>Module</th>
<th>SPR</th>
<th>Offset</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>BNKBIOS3</td>
<td>SPR</td>
<td>F200H</td>
<td>0500H</td>
</tr>
<tr>
<td>BNKBIOS3</td>
<td>SPR</td>
<td>5D00H</td>
<td>2300H</td>
</tr>
<tr>
<td>RESBDOS3</td>
<td>SPR</td>
<td>E00H</td>
<td>0600H</td>
</tr>
<tr>
<td>BNKBDO3</td>
<td>SPR</td>
<td>3000H</td>
<td>2D00H</td>
</tr>
</tbody>
</table>

LOBO SYSTEMS

CS-67

VERSION 2.1
CP/M STARTUP

This is telling you that the BIOS (called "BNKBIOS" here to signify that this is the banked version of CP/M 3) is split into two parts. The first part begins at F200H (i.e., in Common) and extends to F6FFH.

If you turn now to the Technical Reference Manual, you'll find, (in Chapter 2, Memory Organization) a section titled "Memory Map of Movable 4K Block". This section covers part of SWAP1K and all of RAM400. That is, the "free system RAM" mentioned in the Technical Reference Manual may be free as far as the hardware is concerned, but is definitely NOT free in the CP/M 3 operating system. It now holds part of the operating system itself. (The tighter the fit up here, the more room for your application programs in the TPA.)

The second part of the BIOS is switched in and out (note the low base address) just like any other program. This section is in Bank 0 and extends for 2300H bytes from 5000H up to (guess where?) 7FFFH, the top of that bank. A good fit.

The next item is called RESBDOS because it's the part of BDOS that's always present in memory. Thus it has to be in the upper (Common) portion. Note that it starts at, of course, the top of the TPA and extends up to the bottom of BNKBIOS3. Use every byte, we always say.

And the last item on the list is the banked part of BDOS, which starts at 3000H of Bank 0 and extends 2D00H to, suprise, the bottom of the banked portion of the BIOS.

And, if you chose the load-map-at-cold-boot option at the beginning of the GENCFCM program, this is what you will see every time you reboot. What you always wanted, but never could afford.
Depending on exactly what you want to do with the operating system, you may not even have to run CPLD.COM.

This program is built to perform basically just one task. It places on the system tracks of a floppy or hard disk all of the information that is needed there. And what is needed there? The "loader" routines, some basic information about the boot drive itself, and the character set (it's hard to write to the screen when you don't know how to write).

The loader routines are necessary because all of the operating system is in the disk files CPM3.SYS (the file created by GENCPM) and CCP.COM. These routines place those files into the appropriate parts of the MAX's memory.

The information about the boot drive is necessary because it is software that interprets how to read the information coming from the drive. The boot ROM (the Read Only Memory that starts the MAX off looking for the correct drive) does not know exactly what sort of drive you have on the end of the cable. It just goes out to the drive in a very generic fashion. The information on the drive itself (i.e., the system tracks) must determine how the MAX reads that information.
CPMLDR is responsible for writing some messages on the screen even before CPM3.SYS is loaded. (Remember that the system comes up with an error even when the operating system file CPM3.SYS can't be read. That message had to come from somewhere.)

To run most of CPLD you only need one file on your logged in disk: CPLD.COM. If you wish to change the character set, you will need the file CHRSET.COM, but that is the only other file that might be needed. That is because CPLD's business is just installing loader information on the system tracks of disks. Your whole active operating system, except for the character set, is in the file CPM3.SYS that the system tracks (once installed) load into the MAX-80's memory.

Type

CPLD <cr>

and you will get your just desserts:

*******************************
* MAX-80 CPM3 LOADER PROGRAM [12/23/83] *
* Copyright (c) 1983 Lobo Systems, Inc. *
*******************************

Hard disk or Floppy? (1/2):

That is, are you going to be installing the system tracks on a hard or floppy disk?

If you type 1, you will get back, in succession, a recognizable series of questions:

5 inch or 8 inch? (1/2):
Controller is UVC or SASI? (1/2):
5 or 10 Megabyte? (1/2):

Note that if you have a UVC controller, then you do not receive a question about the floppy backup. You cannot create a CP/M 3 floppy backup boot diskette.
The answers to these questions are all things you should know offhand by now (or, at least, you should re-read the description in the discussion of MAKESYS).

If you type 2 in response to the hard disk or floppy question, you will find yourself in another series of all too familiar questions:

3 inch, 5 inch or 8 inch? (1/2):
1 or 2 sided? (1/2):
Single or Double Density? (1/2):
Slow or Fast seek? (1/2):

And, if you chose 5 inch,

40 or 80 Tracks? (1/2):

Note that some configurations are not allowed. For instance, single density 80 track drives (whether 3 inch or 5 inch) simply cannot be used. Fortunately, such drives are practically nonexistent (which is the primary reason for them not being accommodated in MAX-80 CP/M). If you do try to make up a single density 80 track boot disk, you'll just get back an error message:

No 80 track SD !! - [ESC] to continue

And, when you press ESC, you'll be back to the beginning of CPLD for another chance.
CP/M STARTUP

ALTERING THE CHARACTER SET

No matter which boot drive you are intending, the next question from CPLD will be

Do you wish to change character set? (y/n):

The character set is the group of symbols that are placed on the screen when keys are typed. If you want to change which letter goes with which key, you should use MAKESYS to reconfigure the keyboard. But if you want to change the shapes of the characters on the video screen, then you will have to change the character set.

There is a bit of a tradeoff here. You could configure your keyboard (using MAKESYS) so that the key for T produces the letter K, but then configure your character set so that the letter K looks like the letter T. But there is a catch: the character set does not influence the computer, nor any peripheral devices other than the video monitor). So the key T you type that is really a letter K, but that looks like a T on the screen, will still come out as a K on a printer, and act like a K in a response to a program's questions. You have a lot of freedom with a MAX-80, but there are limits.

Character Set Redefinition

If you say Y to changing the character set, you will be returned the message

CHARACTER SET INSTALLATION

NOTE: This section requires a special character set file of an exact length and type, named CHRSET.COM, on the logged drive. Refer to your manual for details.

*** DO NOT EXPERIMENT WITH THIS CONFIGURATION ! ***

Are you still sure? (y/n):
The special character set file is an object file that is directly usable by the operating system. To obtain this file for your new character set, start with the file INVERSE.MAC that is on your backup copy of Lobo System Diskette 1. The object file CHRSET.COM (which was created from INVERSE.MAC) is the file that was used to make the characters on your screen.

INVERSE.MAC is an normal text (ASCII) file, so most editors can be used to edit it. (Beware of editors that install non-ASCII characters, however. They can ruin this file for further use as a .MAC file.) WARNING: DO NOT LOSE THIS FILE. It is easy to alter the character set file sufficiently that you can't understand anything written on the screen. Make sure you have backup copies handy.

The special character set object file must be exactly 2048 bytes long. In addition, you must keep the structure and length of every line in the source file exactly as you see it. MAKE SURE THAT YOU ONLY CHANGE THE SPECIFIC VALUES (substitute a 0 for a 1 or vice versa) AND NEVER DELETE A LINE OR A VALUE.

As you can see when you examine INVERSE.MAC, each character is made up of an 8 by 8 matrix, and each point in the matrix is governed by an individual bit. To actually see what the character looks like, you should take a piece of graph paper and fill in the squares. THEN go back and assign the 1's and 0's according to the pattern on the graph paper.

NOTE: The ninth row that passes between the character lines is available for reconfiguration, but only for the characters 80H through BFH.

Once you are through editing INVERSE.MAC, you can then assemble it using the CP/M macro assembler MAC.COM, just as you would assemble any other source code in CP/M 3. Make sure that you call the output file CHRSET.COM, as that is the file that CPLD needs. For the operation of this assembler, see the Digital Research manual Programmer's Utilities Guide for the CP/M Family of Operating Systems. (You will soon see why your changes had better be accurate; you don't need to spend your life re-assembling, loading and installing versions of your character set.)
Character Set Installation

Finally, once you have the .COM file, you can run CPLD to install the character set on the system tracks. When you receive the message given above, press Y and CPLD will immediately go out to look for CHRSET.COM. If there are obvious problems with the file (such as, it isn't on the disk), then you'll immediately get back the message

FILE NOT PRESENT OR FAULTY! [ESC] TO CLEAR

One try is all you get. If you press ESC, CPLD will go on to the next question. If, indeed, the file was there all the time, then you need to go back to the .MAC file, check it over, and reassemble it.

AND, FINALLY, THE END OF CPLD, TOO

The last question of the whole installation series is

Ready to write to system tracks...
Which drive? (A-P) (R=Abort)

Remember, just one more time, that the drive you call out must be one that CP/M already knows about (it must already be installed in the file CPM3.SYS that was used when you booted the system up). If it is, then the moment you hit the appropriate letter, CPLD will start writing on the system tracks. If it isn't, you'll get back another favorite message and will be back in the operating system:

CP/M Error on x: Invalid Drive
BDOS Function = 14
A>

Ah well, try again.
INDEX

3-1/2 inch floppy drives.
   See Microfloppy drives.

5-1/4 inch floppy drives.
   See Minifloppy drives.

5-1/4 inch hard drives.
   See Winchester.

8 inch floppy drives, 27, 70
40, 80 track drives.
   See Minifloppy drives.

A
AUXIN: 51

AUXOUT: 51

B
Backup copying, 1-6
Backup floppy drive, 1, 25, 29, 70
Banked memory, 9-12, 61-68
BASIC, 55
BIOS, 16, 24, 33, 68
Blocking, 64
BNKBOS3.SPR, 2, 59, 67
BNKBOS3.SPR, 2, 20, 21, 23, 56, 59, 67-69
BREAK key, 21, 33, 46
Buffers, 64-68
   Sharing, 67

LOBO SYSTEMS

VERSION 2.1
Carriage return, 34-35
CCP.COM, 16, 19, 63, 69
Centronics.
   See Printer, parallel.
Character set, 72-74
CHRSET, 70, 73-74
CLEAR function, 34, 46
Columns, 33
Command line, 34-35
Compatibility, software, 13-15
CONIN:, 48-51
CONOUT:, 50-51
Control block, 66
COPIER, 2-6, 18
Copying
   Backup, 1-6
   Single drive, 5-6
COPYSYS, 20
Controller
   Board,
   Winchester drive,

COPIER
CPLD, 16, 18-20, 69-74
CP/M 2.2, 13-19, 38, 60
CP/M Plus (CP/M 3), 9-16
CPM3.SYS, 16, 18, 20, 23, 69-70
   Problems with, 23, 58, 65
CPMLDR, 70
CRT, 48
CTRL keys, 21, 31, 34-35, 38-40, 60
Cursor
   Control, 32-33
   Shape, 52-54
DB25 connector, 41
Default devices, 47-51
DEL key, 46
DEVICE, 17, 38, 40, 47-48
Diagnosis, 26, 37-39, 40, 43-44
DIR, 63
Diskette tracks, 3
Drive.
   See appropriate type:
   Boot, 8 inch floppy,
   Minifloppy, Winchester.
Drive 1:, 25-27
Drive A:, 1, 23, 25-26, 30
Drive select code, 27-28
Drives, total number, 22, 24
CP/M STARTUP

E
ESC, 53, 74

K
Keyboard, 45-46

L
Line feed, 34-35
Loader, 16, 20, 70
Lobo Original Diskettes, 1-5, 73
LST:, 17, 38, 40, 51

M
MAC, 73
MAKESYS, 16, 18-20, 21-56, 58, 62, 65, 72
Master diskette, 3
MAX-80, 1-74
Memory allocation, 64-68
Microfloppy drive, 27-28
Minifloppy drive, 27-28, 70
Installation,
Modem
See Port, serial.
Multit-user, multi-tasking, 50-51

I
INVERSE.MAC, 73

LOBO SYSTEMS
CS-77
LOBO SYSTEMS
VERSION 2.1
CP/M STARTUP

P
Parallel printer.
See Printer, parallel.

PIP, 35

Port, 36-44
  Parallel, 36-39
  Serial, 36, 40-44

PPRT, 17, 38, 48

Printer
  Parallel, 17, 37-39
  Serial. See Port, serial.

R
RAM, 12, 64
RESBDOS3.SPR, 2, 20, 59, 67-68
RESET, 45
ROM, 69
Rows, 33

RS-232C
  See Port, serial.

S
SASI, 29-30, 70
Seek, 27, 71

Serial port.
  See Port, serial.

Serial printer.
  See Port, serial.

SHIFT
  Function keys, 31
  Lock 31, 55

SIOA, SIOB, 40, 47-48
SN Slow seek, 27, 71
Software compatibility, 13-15
STAT, 18
System calls, 13
System files, 16, 19
System tracks, 70

T
Terminal functions, 32
TPA, 12, 65, 67-68

U
UVC, 25, 29-30, 66, 70

Winchester, 27, 29-30
  Lobo 950, Lobo 1850, 1

LOBO SYSTEMS
CS-78
VERSION 2.1