CS

CS Introductory Manual

1st Edition — May 1987

715-1213

WANG LABORATORIES, INC.
ONE INDUSTRIAL AVE., LOWELL, MA 01851 TEL. (617) 459-5000, TWX 710-343-6769, TELEX 94-7421

Disclaimer of Warranties and Limitation of Liabilities

The staff of Wang Laboratories, Inc., has taken due care in preparing this manual. However, nothing contained herein modifies or alters in any way the standard terms and conditions of the Wang purchase, lease, or license agreement by which the product was acquired, nor increases in any way Wang's liability to the customer. In no event shall Wang or its subsidiaries be liable for incidental or consequential damages in connection with or arising from the use of the product, the accompanying manual, or any related materials.

Software Notice

All Wang Program Products (software) are licensed to customers in accordance with the terms and conditions of the Wang Standard Software License. No title or ownership of Wang software is transferred, and any use of the software beyond the terms of the aforesaid license, without the written authorization of Wang, is prohibited.

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 protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

FIGURES

TABLES

PREFACE

This manual introduces the Wang CS. It describes the features of the basic system and its nominal components: the Central Processing Unit (CPU), an interactive D-type terminal such as the 2436DE Interactive Terminal with business-graphics capabilities, and the diskette drive. Master initializing the system, configuring the system (partition generation), backing up the system platter, executing the user diagnostics, and performing routine maintenance procedures are also discussed.

This manual should be used in conjunction with the following documentation supplied with every system.

- *Multiuser BASIC-2 Language Reference Manual*(700-4080) - describes the system's operational features, documents the set of system commands, and describes each statement in the BASIC-2 instruction set.

- *BASIC-2 Utilities Reference Manual*(700-6855) - describes the utilities provided with the BASIC-2 Operating System. It includes specifications for the utilities as well as installation and operating instructions.

- *Programming in BASIC*(700-3231) - introduces the beginner to BASIC programming and the utilization of peripherals that are frequently used with the CS series of computers.

Additionally, a separate reference manual is provided for each optional peripheral device attached to the system.

CHAPTER 1
INTRODUCTION

SYSTEM OVERVIEW

The Wang CS is a versatile, disk-based computer system that concurrently supports multiple users programming in the Wang BASIC-2 language. As many as 16 terminals can run up to 16 jobs with the same response time as a single-user interactive system. The following components constitute a basic CS system.

- A CS CPU with 128K or 512K bytes of user memory
- A disk system, DS or 2275
- One to sixteen Model 2436-type terminals with business-graphics capabilities

The CS also provides multiprogramming (a feature usually available only on larger systems), yet is as easy to use as small single-user systems. The operating system executes several programs concurrently because several users share a single CPU. A terminal can control more than one program, allowing foreground (interactive) and background (noninteractive) processing. Global (reentrant) programs and variables can be shared by users, greatly reducing the memory required to perform multiuser applications. The system overlaps processing and I/O to efficiently use the memory and peripherals that support a variety of applications.

The system is simple to program and operate by using BASIC-2, a variety of powerful system commands, and application software available from Wang and other vendors. Since the user at each terminal may access all system resources directly from the keyboard, no special personnel is required to oversee system operation. This direct access saves programming, job turnaround, and operating time.

The modular design of the system is economical and permits easy expansion and physical relocation of the system's components. Connections between the CPU and each terminal can be either local or remote (RS-232-C/CCITTV.24), allowing the system to be adapted to a wide variety of applications and configurations. Each terminal supports a local printer.

Allocation (configuration) of system resources can be redefined as needed by the primary system operator. When security requirements exist, the CS lets you selectively restrict access to the system's resources.

The features of this system are discussed in greater detail in the *Multiuser BASIC-2 Language Reference Manual.*

Typical System Configuration

A CS may be configured with a variety of peripheral devices and options. A typical system might include a CPU with 512K bytes of user memory, three Model 2436-type terminals for interactive communication with the CPU, a DS Diskette Drive unit for system interfacing, software loading, for program and data storage, and a Model PM017 Line Printer for hardcopy output.

Each unit is attractively designed for easy integration into the office environment. See Figure 1-1 for an illustration of a typical configuration.

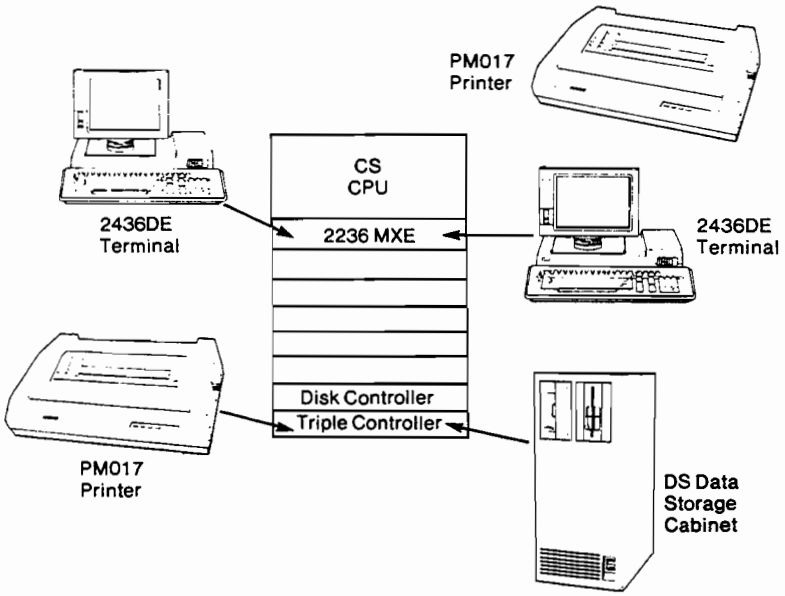

Figure 1-1. Typical CS Configuration

System Memory

The CS utilizes two distinct types of memory: control memory for system needs, and user memory for user programs and data. Control memory contains the operating system, Multiuser BASIC-2 interpreter, and other system data; it is inaccessible to the user or the user's programs and data. User memory contains user programs and data, as well as some system control information. The size of the user memory, which can be 128K bytes or 512K bytes, determines the "size" of the particular CS.

User memory is organized into banks. Bank 1 contains 61K of user memory, while Banks 2 through 8 contain 56K bytes each of user memory. User memory in each bank is distinct from memory in any other bank, and programs and data in one bank are not generally accessible to programs and data in another bank. However, the first 5K bytes of available memory in Bank 1 constitute a universal global area; programs stored in this area of memory can be accessed by any other program in the system.

User memory can be divided into as many as 16 separate partitions, each of which can be used for a separate program. A global partition, however, contains programs accessible to several or all users on the system. Up to 16 terminals can concurrently access the memory partitions. Each partition is assigned to either one terminal on the system or to the null terminal during configuration of the system by the operator. However, each partition is controlled by only one terminal at any time, giving each user essentially a standalone system. Each terminal may control more than one partition simultaneously, and ownership of these partitions can be changed during system operation without reconfiguration. See Chapter 3 for information on partitioning and terminal assignment.

The System Platter

The operating system, Multiuser BASIC-2 interpreter, system utilities, and other system programs are provided on two system diskettes. The files on these diskettes may be "installed" onto a single system platter. When the system is master initialized, the operating system and interpreter are loaded from one system platter into the control memory. The system diskettes shipped with each CS also contain a variety of hardware diagnostic routines as well as a utility program to generate, execute, and save system configurations.

Caution: *Because they contain so much configuration information, the system platters must be handled with extreme care. Make a backup copy at the earliest opportunity to protect against accidental loss or destruction.*

Chapter 3 describes how to load the operating system and the Partition Generator utility from the system platter. Suggested procedures for creating a backup copy of the system platter are also covered in Chapter 3. Chapter 5 discusses the procedure for loading and executing the diagnostic routines from the system platter.

BASIC-2 Language Features

The Wang CS, running either Multiuser BASIC-2 or VP BASIC-2, supports the BASIC-2 language, whose powerful and flexible capabilities provide a wide range of applications. Multiuser BASIC-2 is compatible with earlier versions of Wang BASIC-2, so all Wang software previously developed for 2200 series systems can be run on a CS.

Multiuser BASIC-2 contains instructions for data testing, manipulation, and logical operations at the bit and byte levels, comparable in scope to those of many assembler languages. Many data operation statements can support both numeric and alphanumeric expressions and can operate on all or part of alphanumeric arrays, thereby enhancing the system's data-handling capability.

System commands give the operator complete control over system operations from the keyboard and also serve as powerful debugging tools. Statements exist to facilitate common programming tasks such as decision-making, branching, looping, passing data to subroutines, overlaying program modules, and accepting and processing operator-entered data. A variety of instructions are provided for controlling disks and plotters and for formatting output to the printer and CRT displays. Several groups of special-purpose statements perform operations such as code conversion, sorting, matrix arithmetic, and customized I/O control. Additionally, in Multiuser BASIC-2, a number of statements are implemented to support interpartition sharing of programs and variables, temporary control of shared devices by one partition, specification of multiuser system-configuration parameters, and other multiprogramming features.

Numeric operations are carried to 13 digits of accuracy for most operations. The user may specify whether results are to be rounded or truncated at 13 digits. When full precision is not required, numeric data can be packed from eight bytes to as few as one byte. Trigonometric arguments can be specified in degrees, radians, or gradians.

Multiuser BASIC-2 supports numeric variables and alphanumeric string variables (both scalar and array). The maximum number of elements in one-dimensional arrays is 65,535, and 255 by 255 in two-dimensional arrays. Each alphanumeric array element or alpha-scalar-variable can be from 1 to 124 bytes in length. Alphanumeric arrays can be used in data manipulation statements as scalar variables (element boundaries are ignored), providing a convenient technique for manipulating extremely long character strings.

The Operating System

The CS runs Multiuser BASIC-2, which supports up to 16 concurrent programs.

The CS can also execute VP BASIC-2 previously used by the 2200VP and 2200SVP systems. BASIC-2, which configures the CS as a single-user, one partition system, is generally more restrictive than Multiuser BASIC-2 and therefore, is not normally used with the CS.

The general instruction set for the CS is discussed in the *Multiuser BASIC-2 Language Reference Manual*. Instructions used to control all other I/O devices are described in the reference manual for the particular device.

Interactive Terminals

The 2436D-type terminals let each user easily control and communicate with the system. The terminal consists of a 24-line by 80-character-per-line, Cathode Ray Tube (CRT) screen and displays a full 128-character set. Its keyboard contains a 10-key numeric keypad and 34 user-definable function keys. Up to 16 2436D-type terminals can be configured with the CS CPU.

The Model 2436DE and 2436DW Terminals are capable of both box and character graphics and can display bright, blinking, underlined, and reverse video (dark characters on a light background) characters. Special features of the Model 2436D-type Terminal also include an audio signal to indicate a variety of error conditions, repeating keys, and the capacity to support a screen dump to its own local printer. The features of the Model 2436DE and 2436DW Terminals are detailed in Chapter 2.

System Expansion

Because of the CS modular design, it is easy to add memory and peripheral devices. The CPU supports up to 512K bytes of user memory. The CPU can be configured with up to 16 terminals, several disk models, high-speed printers (up to 600 lines per minute), and industry-compatible, 9-track magnetic tape. Appendix C contains a list of peripheral devices that can be added to the system.

The CS also supports telecommunications. This option allows the system to share information, either locally or remotely, over phone lines and modems, with another Wang system or a host computer.

ENVIRONMENTAL CONSIDERATIONS

Although the CS is designed to operate in most office environments, the environment should be designed to encourage optimal system performance. An ideal environment is one in which the temperature and humidity are controlled and airborne contaminants are minimal. The AC power source should be grounded, adequate for present and proposed system needs, regulated, and noise free. The room should have space for future expansion and be easily accessible to operating personnel, yet sufficiently removed from the main traffic flow to permit the system to operate smoothly. See Appendix D for further information.

UNPACKING, INSPECTION, AND INSTALLATION

Special factory packing techniques require that the CS be unpacked, inspected, and installed by a Wang service representative. When the system arrives, call the Wang Customer Engineering Office and request that this service be performed. Failure to follow this procedure voids the warranty.

The Wang service representative checks to see that all equipment has been delivered, inspects each unit for shipping damage, connects each device in the system and performs tests to verify the proper operation of all system components.

If modems and telephone lines are going to be used, this equipment should be installed before the Wang equipment is delivered. The Wang service representative connects the Wang equipment to the modems. The modem vendor or phone company, however, must connect all modems to the proper transmission lines.

CHAPTER 2
SYSTEM COMPONENTS

THE CENTRAL PROCESSING UNIT

The CS Central Processing Unit contains the processor, interpreter, memory, and system logic necessary to resolve and execute programs, perform arithmetic operations, and control the input and output of data. The minimal CS CPU contains 128K bytes (1K = 1024 bytes) of user memory and nine I/O slots. (Controller boards for peripheral devices are plugged into the I/O slots.) User memory can be expanded to a maximum of 512K bytes.

Data memory is divided into 64K byte banks. The hardware maps the first 8K addresses in each bank onto the first bank. In this shared 8K byte area, 3K bytes of control information are stored. In the remaining 5K bytes, the user may create one or more partitions that have the special property of being accessible to all banks. Partitions created in the remaining 56K bytes of each bank may share data and program text within their bank, but are isolated from partitions in the other seven banks.

The CPU and operating system employ many techniques to maintain optimum memory integrity and efficiency. The CS provides overlapped I/O and program execution, significantly shortening the overall processing time of foreground (interactive) and background (noninteractive) jobs. All programs are stored in a condensed form to save memory. Multistatement lines can also be used to conserve memory, speed program execution, and logically group related statements.

Programmable device selection allows increased flexibility for I/O operations. One user can monopolize the devices allocated to all users during system configuration to secure exclusive access to those devices. (Refer to the *Multiuser BASIC-2 Language Reference Manual* for further information.)

System diagnostics automatically verify proper CPU operation whenever the system is master initialized. Additionally, hardware and software errors detected during any system operation are signalled by an audible tone and a code identifying the error and its approximate position. System response to many types of errors can be suppressed if handling errors through program control is preferred. Debugging facilities provide complete program, variable, and subroutine cross-reference listings and allow the programmer to step through the execution of a program one statement at a time, observing variable assignments and program transfers as they occur. The programmer can also interactively edit program lines, immediate mode lines, and input date values both during and after entry.

I/O Controllers

The standard CS CPU has nine I/O controller slots, each of which holds an I/O controller board used to manage one or more peripheral devices. Some controllers possess front-end intelligence, which allows the controller to perform I/O processing and CPU processing concurrently.
The controller for every peripheral device attached to the CPU is identified by a unique device address consisting of three hexadecimal digits. The first digit of the device address is used by the operating system to identify what type of I/O routines are to be used to control the device. The second and third digits of the address represent the hardware address set on the controller, which is used by the system to electronically identify that device when information is to be transmitted to or from it. A printer attached to the CPU is normally addressed 215, and plotters are normally addressed 413 or C13.

The 2236MXE Terminal Processor requires one I/O slot and can support up to four 2436D-type terminals. Unlike the device addresses set on most peripheral processors, the physical device address set on the 2236MXE Terminal Processor is not used to access the terminals. Instead, the program address of all CRTs, keyboards, and local printers is the same for all terminals on the system. The operating system automatically translates the program addresses into the appropriate physical address for the processor port being utilized. Thus, each program in each partition addresses its terminal and local printer as if they were the only ones on the system. The last two digits of the device address of the CRT are always 05. The device address of the keyboard is always 001. The last two digits of the device address of a local printer (i.e., a printer directly attached to a terminal) are always 04.

If a single disk unit is the only one on the system, the fixed drive normally has address D11, while the removable drive has address D10. The procedures for addressing a disk, when more than one drive unit is contained in the system, are discussed in the *Multiuser BASIC-2 Language Reference Manual*.

Each 2436D-type terminal is connected to a 2236MXE Terminal Processor or a 22C32 Triple Controller that controls data input and output at the terminals. Hardcopy printout to a local printer can be created concurrently with CRT displays at each terminal. The terminal processor and triple controller collect keystrokes into lines of text and buffer the CRT and local printer output to contend with the speed discrepancies between the asynchronous line, the CRTs, and local printers. Buffering allows the CPU to be more fully devoted to the execution of programs than would be possible if it had to pause frequently to test the ready status of I/O devices.

Line handling between the CPU and each terminal is asynchronous and full-duplex. Since the 2436D-type terminals and the Model 2236MXE Terminal Processors are RS-232-C/CCITTV.24 compatible, both local and long-distance terminal placement are possible. Local terminals can be placed up to 2,000 feet away from the CS CPU, while unlimited distance is available for remote terminals using telephone lines and modems.

Any combination of local, extended-local, and remote connection may be used. However, the terminal connected to 2236MXE Port 1 should reside in the same area as the CPU because it acts as the system console for master initialization when the system is turned on and for the reporting and correction of system errors. Allowable configurations of the terminals relative to the CPU and 2236MXE processor are as follows.

Local - For distances up to 25 feet (7.6 meters), transmission rates of 19.2K baud are available with a direct four-wire connection using a Wang-supplied cable. (This is the standard configuration.)

Extended Local - For distances from 25 feet (7.6 meters) to 2000 feet (610 meters), optional Wang-supplied cables provide direct extended-local connection at speeds of 19.2K baud. These cables are available in lengths up to 2000 feet (610 meters).

Remote - For distances greater than 2000 feet (610 meters), two asynchronous, full-duplex, RS-232-C compatible modems, operating at the same baud rate, and suitable transmission lines must be used to provide the communications link. One modem is plugged into the terminal and the other is plugged into the terminal processor. The following two types of remote configurations are possible, depending upon the distances involved.

> *Distances up to 5 miles (8 kilometers)* - Short-haul modems using private, voice-grade lines with a four-wire connection between modems may be employed.
>
> *Distances exceeding 5 miles (8 kilometers)* - Private (dedicated) or switched (dial-up), voice-grade telephone lines provide the connection between telecommunication modems. The modem selected determines whether these lines must be two-wire or four-wire.

Line speeds ranging from a minimum of 300 to a maximum of 19.2K baud may be selected (19.2K baud equals about 1900 characters per second). Baud rates of 300, 600, 1200, 2400, 4800, 9600, and 19,200 are supported. The Model 2236MXE Terminal Processor and the 2436D-type terminal each have a set of switches for setting the transmission speed. Both the Model 2236 MXE Terminal Processor and the terminal connected to it must be set at the same baud rate. The transmission character format is 11 bit (1 stop + 8 data + odd parity + 1 stop bit). Special cables for modem connections of Wang equipment are required; they are available in 12-foot (3.7-meter), 25-foot (7.6-meter), and 50-foot (15-2-meter) lengths. Modems must also be set at the same baud rate as the terminal and terminal processor.

Memory Partitions

Multiuser BASIC-2 supports multiuser operations utilizing a fixed-partition memory approach. In the first bank of user memory, approximately 3K bytes of memory are required for system overhead. In each subsequent bank, 8K bytes of memory are required. Thus, a 512K-byte CS contains 453K bytes of memory available for partitions. This memory can be divided into a maximum of 16 partitions. Each partition's size is specified in 256-byte (0.25K-byte) increments, with a minimum partition size of 1.25K bytes required and a maximum size consisting of the remaining available memory in any one bank. However, no partition may be specified that contains memory from more than one bank. Approximately 8K bytes are needed in each partition for overhead, leaving the remaining memory available for user programs and data.

The system configuration (i.e., number and sizes of partitions, the terminals and peripherals attached to each partition, etc.) is defined by a procedure called partition generation. The allocation of system resources (system configuration) can be changed whenever necessary by the primary system operator. Since a number of different system configurations can be generated, named, and stored on the system disk, partition generation can be made similar to loading a program and can even be performed automatically. The process of system configuration and the parameters which can be modified by the user during system configuration (i.e., background/foreground partitions, disabled programming mode, bootstrap programs, etc.) is described in Chapter 3.

Each user has a distinct memory area or partition for program and variable storage. There is, however, only one central processor. The central processor executes the program in each partition sequentially in a round-robin fashion. By bypassing inactive partitions, overlapping I/O with CPU processing, and switching between users frequently, the system assures good response time for each user. This interleaving of the execution of the programs in the system causes the user programs to appear to be executing simultaneously, creating the appearance of a single-user system for each user.

The CS partitioning scheme imposes some programming constraints on system users because timing considerations of the operating system must be considered. Unbuffered, time-dependent software and associated devices cannot be used with the operating system. For a discussion of programming considerations and BASIC-2 language enhancements for multiprogramming, refer to the *Multiuser BASIC-2 Language Reference Manual*. If time dependency is required, the CS can use the VP BASIC-2. The same hardware, at different times, supports both systems.

The system supports up to 16 partitions, and 16 terminals at the same time. A single terminal can also control more than one partition. Partitions that are currently attached to a terminal (running an interactive job, for example) are said to be running in the foreground. Those partitions that are not currently attached to a terminal (running noninteractive jobs, for example) are said to be running in the background.

A terminal that controls several partitions can be switched from one partition to the next to initiate program execution and to permit necessary user/program interaction. When a program running in a background partition attempts to communicate with a shared terminal but the terminal is attached to another partition, program execution is suspended until the terminal is released by the current partition. A terminal can be released to a specific partition or made available to any waiting partition either by the operator or under program control. For each partition, only the terminal assigned to the partition can list the program in the partition, release the terminal to that partition, or modify the program executing in the partition. A partition assigned to the null terminal, however, can be attached to any terminal upon request. Partitions may be assigned to the null terminal at any time during system operation.

Each partition functions independently of the others in a manner similar to a separate, single-user system. However, any partition may declare itself global, making its subroutines and specially labeled global variables accessible to other partitions within its bank. Since BASIC-2 program text in a global partition is re-entrant two or more partitions may simultaneously execute the same statement in a shared subroutine. Global variables can be interrogated and modified by several users, providing a convenient mechanism for interpartition communication. (Note that although other partitions can access and modify global variables, they cannot modify global programs.) In many cases, the special features of global partitions also speed throughout and allow several users to concurrently process several related applications at each of the terminal sites. This concurrent processing allows users to share the application programs, eliminating the necessity of duplicating those programs. Refer to the *Multiuser BASIC-2 Language Reference Manual* for a detailed discussion of global programming capabilities.

Programming and immediate mode operations can be inhibited for any partition(s). In the Disabled Programming mode, the operator is limited to BASIC-2 software, preventing inadvertent or unauthorized use of certain data files and programs. The operator can only load and execute a program called START, which can provide a menu of operations that can be performed and cause other software to be loaded.

TERMINALS

A variety of 2200 interactive terminals and terminal emulations may be used with the Wang CS. The interactive terminal is your principal means of communicating with the CS. The different terminals have different physical characteristics but offer similar functionality.

The terminals that may be attached to the CS include: 2236, 2336, 2436, and the 2436WP. There are 2200 terminal emulations that may be run on different personal computers.

KEYBOARDS

The 2236 is a single unit made up of the CRT and keyboard. The 2336 and 2436 models offer detached keyboards. Three different keyboards are available: DE, DW and expanded DW. Refer to the *2436 Series Interactive Terminal User's Manual* (715-0739) for more information on the 2436D-Type Interactive Terminal.

The DE-style keyboard was designed primarily to support data entry. It consists of a typewriter-style keypad along with a numeric keypad. In addition, certain BASIC-2 keywords are supported with single keystrokes. Additionally, the function keys are labelled to support the built-in edit functions.

The DW-style keyboard was designed to support word processing in addition to the data entry function. The the DW-style keyboard offers the typewriter keypad, the numeric keypad, as well as a cursor control pad. The cursor control pad allows you to move the cursor up, down, left, and right. In addition, single key screen scrolling (previous/next screen) and editing (insertion and deletion of text) are offered. Additionally, the function keys are labelled to support word processing functions such as INDENT, PAGE, CENTER, and etc.

The expanded DW keyboard offers the standard DW features with some additional keys that are used to support foreign character sets.

The 2436WP is a multi-tasking standalone word processor that offers the added functionality of 2200 Terminal Emulation. It provides local word processing, printing, forms, and merge print capabilities. When connected to a Wang 2200 minicomputer system, it becomes a remote terminal with the same basic functionality as a Wang 2236DE/DW terminal. A file transfer and archive system is provided to allow 2436 documents to be stored in an archive file on the 2200 system. A complete description and set of instructions are included with the 2436WP Terminal Workstation.

Additional 2200 terminal emulators, available through Wang and other vendors, allow a Wang PC, as well as IBM and IBM clones, to act as a 2200 terminal.

DISK UNITS

The principle disk unit for the Wang CS is the Wang DS Data Storage Cabinet that offers a wide range of storage and backup capabilities. In addition, all the current 2200 disk peripherals may be used with the Wang CS. The list of available units include: the Model 2260 and 2270A Series Disk Drives, the Model 2275 Disk Drive, the Model 2280 Disk Drive.

These drives offer various amounts of storage on combinations of floppy drives and fixed/removable hard drives (e.g. Winchester drives). The disk units are divided into platters. A platter is made up of a number of sectors and may contain up to 16 MB of storage depending upon the disk model.

Up to three disk units may be attached to any one CPU. This includes a DS, a 2275, or pairs of 2280. In addition to those three units, other 2260 and 2270 disk drives may be attached. The disk unit may also contain up to 32 disk platters.

Some disk units may be "shared" by more than one CS or 2200 via a Disk Multiplexer. The Models DS, 2280, and 2275 may be shared by up to a total of 16 CPUs via the 2275MUX and 2275MUXE controllers.

Multiuser BASIC-2 provides the user with a number of statements that are designed to allow the user to store and retrieve data and programs on any of the disk units available.

PRINTERS

The Model CS may utilize any Wang 2200 printer as a hardcopy output device. The 2200 printers and some Wang PC printers are configured as system printers and terminal printers. All printers are accessed through BASIC-2 statements such as PRINT, PRINT USING, and LIST.

User application programs may be written for a specific printer or may be written to be used with the Generalized Printer Driver (for Wang PC printers). The CS system platter contains a number of printer driver tables that may be used in conjunction with the Generalized Printer Driver feature of the BASIC-2 Operating System.

Printers are treated as either a system printer or a terminal printer. A system printer is attached to a printer controller installed in the CS and is accessible by all users. A terminal printer is attached to one of the terminals and is accessible by that terminal only.

Note: *Only 2200 printers can be attached to terminals.*

BASIC-2 provides statements that allow the application program to direct hardcopy output to either type of printer.

Wang offers a range of printers that includes matrix printers for draft quality output, daisy wheel printers for high-quality, letter output and band printers for high-speed output. All printers must be attached to a 22C02 controller, a 22C11 Dual Controller, a 22C32 Triple Controller, or directly to a terminal.

COMMUNICATIONS CONTROLLERS

Communication is handled by a software package working in conjunction with a communications controller board. Packages are available for both asynchronous and synchronous communications.

CHAPTER 3
MASTER INITIALIZING THE SYSTEM

MASTER INITIALIZATION

Master initialization is the process of powering up components of the system, loading the operating system and BASIC-2 interpreter, and exercising the CPU to determine if any malfunctions exist. In the CS, the bootstrap proms contain the only resident (hardwired) microcode. These proms represent the only area that is functional as soon as the CPU is turned on. Neither user memory nor control memory contains any information when the system is first powered up. Before any operations can be performed by the system, the operating system and BASIC-2 interpreter must be loaded from the system platter into the control memory to manage the processes of interpreting and executing the BASIC text, overseeing variable and program storage in user memory, and controlling I/O.

The CS can be master initialized only at Terminal 1 (the terminal attached to the primary 2236MXE Terminal Processor at Port 1). For this reason, Terminal 1 should be located at the central system site as near as possible to the CPU and disk drive.

The bootstrap routines are invoked automatically whenever the system is powered up or whenever the operator at Terminal 1 executes the Immediate mode $INIT password command (in order to reconfigure the system or whenever certain CPU malfunctions are detected). The operator need only verify that the system disk is properly mounted and inform the bootstrap routines (via the RESET button and function keys) of which disk address to access and which system program to load.

Both user memory and control memory are cleared when the system is powered down. Therefore, the system must be master initialized following any power-down operation or whenever the system resources are to be reallocated. In rare instances, it may be necessary to power off and master initialize to recover from an error condition. Always save any desired programs or data residing in memory before powering off the system.

Perform the following procedures to master initialize the CS.

1. Power up the CPU, the system disk, each terminal, and any other device to be used. (Always power the system up before mounting the system platter.)

   If the system is already running and master initialization is required to reconfigure the system, it is not necessary to power the system components down and power them up again. The Immediate mode form of $INIT, when executed at Terminal 1, causes the system to initiate the bootstrap routines as if the system had just been powered up.

2. Mount the system platter, containing the BASIC-2 interpreter and operating system, as well as a variety of hardware diagnostics and other system programs.

3. Use the RESET and function keys to load the operating system and BASIC-2 interpreter from the system platter into control memory.

Each of these procedures is described in detail in the following paragraphs.

System Power Up

To power up the system, first place the CPU's Power switch in the ON position. Then turn on the CRT and printers. As soon as the system warms up (15 seconds), the Terminal 1 CRT displays a self-identification message, such as the following one.

```
2436DE   R0101  19200BPS    8 + O  (USA)
```

where:

2436DE	is the model number.
R0101	is the revision number of terminal firmware, preceded by R.
19200BPS	is the data rate, followed by BPS.
8 + O	is the number of data bits (7 or 8); E is even parity, O is odd parity, and N is no parity.
(USA)	is the version of the keyboard and CRT character set, enclosed in parentheses.

This indicates that the terminal has passed its internal diagnostics.

Next press the RESET key. The following message should then appear on the CRT display of Terminal 1.

```
MOUNT SYSTEM PLATTER
PRESS RESET
```

If the MOUNT SYSTEM PLATTER PRESS RESET message does not occur, or if it is incomplete, an error has been detected by the bootstrap diagnostics. Master initialize the system again. If the second attempt fails, see Master Initialization Error Recovery in Chapter 5. If master initialization still cannot be performed, call your Wang service representative.

Mounting the System Platter

The following procedure applies to the DS Diskette Drive and should be followed when the system platter is first mounted. It is recommended that the system files be copied onto another disk/diskette and mounted from there at each master initialization and that the system platter be used solely as backup. (See the section in this chapter that discusses the backup of system platters.)

A write-protect slot is provided on each diskette to prohibit writing over information already stored. Valuable programs or data can be protected from inadvertent loss by using the write-protect tabs. If the slot on a 5 1/4 inch diskette is covered, information can be read from the diskette but no information can be written onto the diskette. To write-protect the diskette, place one of the adhesive tabs supplied with the diskettes over the write-protect slot.

All blank diskettes are manufactured without the tabs covering the write-protect slot. Diskettes can accept data only when the write-protect slot is uncovered.

Note: *The procedure for write protecting 5 1/4" 2275 diskettes is exactly opposite of what is used for 8-inch diskettes on 2200 systems. An open slot on an 8-inch diskette protects it from being overwritten; writing is allowed when the slot is covered.*

1. Remove the system platter from its envelope, and check to ensure that the platter moves freely in the jacket and that it is not write-protected.

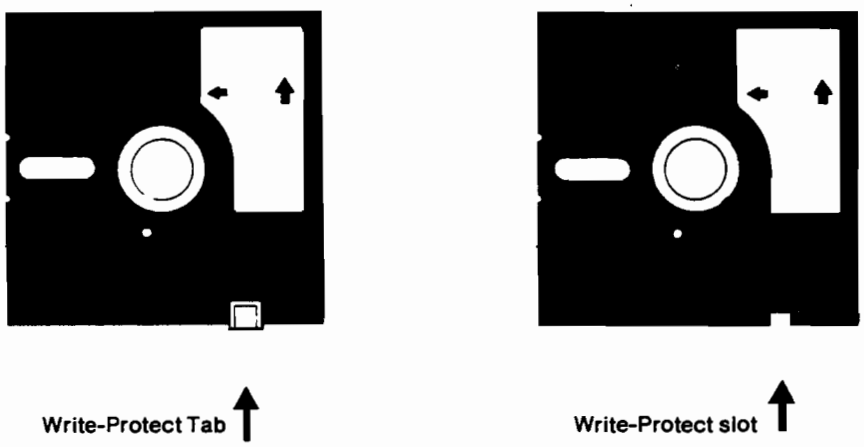

Figure 3-1. A 5 1/4 inch Double Sided Double Density Diskette Showing the Write-Protect Feature

2. Open the door of the diskette drive by raising the floppy door lever to its vertical position.

3. Grasp the diskette as indicated in Figure 3-2 and slip it into the slot in the drive. Be sure the edge of the diskette closest to the recording slot goes into the drive first. Use the Insert and Up arrows on the orientation label to confirm that the position of the diskette is correct as you insert it in the drive. Slip the diskette completely into the drive.

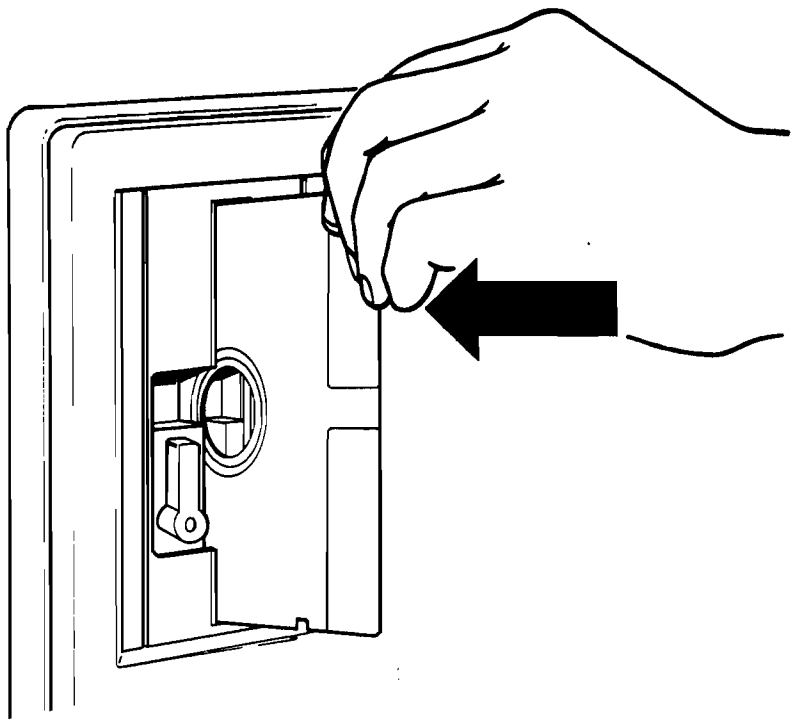

Figure 3-2. Inserting a Diskette

4. Close the diskette drive door by lowering the lever to the horizontal position.

5. Once loaded, the system platter can be removed by raising the drive door over to the vertical position and removing the diskette. The system platter should immediately be placed in its protective envelope.

Loading the System Programs

When the system platter is properly mounted, press the RESET key, which is located in the upper-left corner of the keyboard. The following prompt is displayed at Terminal 1.

```
KEY SF'?
```

You must now press a function key to specify the disk address where the system files are located. (The function keys are arrayed across the top of the keyboard. See Chapter 2.) The following options are available for loading BASIC-2 from the system platter.

SF KEY '00 = Load from disk address D11 (or 310)
SF KEY '01 = Load from disk address D10 (or B10)
SF KEY '02 = Load from disk address D21 (or 320)
SF KEY '03 = Load from disk address D20 (or B20)
SF KEY '04 = Load from disk address D31 (or 330)
SF KEY '05 = Load from disk address D30 (or B30)

The bootstrap also allows the default file name @@ to be overridden by entering the file name immediately before pressing the function key. The name of the file to load must begin with @ and can be no more than four characters in length. This feature allows more than one operating system to functionally reside on a platter.

Normally, the removable drive in the primary or default diskette unit is assigned address B10, and the first fixed drive is assigned address 310. If there is an additional disk unit on the system, the disks on the second drive are usually addressed B20 and 320.

Approximately 15 seconds are required for the BASIC-2 interpreter and operating system to be loaded into control memory from diskette. While they are loading, a BASIC-2 or diagnostics menu appears. After making that selection, the following message appears.

```
Loading:  Multiuser BASIC-2 Release X.X
```

Where X.X is the number of the release being used.

If you press the wrong function key (e.g., if the system platter is mounted at address 310, but you press SF key '01), an error message is displayed in the following form.

```
***SYSTEM ERROR (DISK 00XX)***

PRESS RESET
```

It is generally easy to recover by simply pressing RESET and pressing the correct function key. If RESET fails, turn the system off, wait at least 5 seconds, turn the system on again, and repeat the master initialization sequence. If this approach fails, try the master initialization error recovery procedures discussed in Chapter 5. If the master initialization still cannot be completed, call your Wang service representative.

When the operating system is loaded, the system loads and runs the @GENPART program from the system platter. @GENPART allows you to interactively create a system configuration or automatically execute any configuration stored on the system platter. @GENPART also allows other programs resident on the system platter to be loaded into the partitions and executed automatically when the system is configured, with no further operator intervention.

If @GENPART is not on the system platter, the system is configured into a limited system in which Terminal 1, the default disk drive, and any local printer attached to Terminal 1 are ready for limited use and the READY (BASIC-2) message is displayed. The other terminals and system resources can be utilized (i.e., are enabled) only after configuring the system.

PARTITION GENERATION (SYSTEM CONFIGURATION)

Partition generation (system configuration) divides the resources of the system among the various users. This section discusses the use of the @GENPART utility program to create, save, and execute system configurations. @GENPART is supplied with every system on the system platter.

User-Specified System Configuration Parameters

You can specify the following ten parameters when configuring a system.

1. The number of partitions
2. The number of terminals
3. The size of each partition
4. The terminal associated with each partition
5. The programmability of each partition
6. The bootstrap program for each partition
7. The addresses of the peripherals attached to the system
8. The access to peripherals
9. The system message
10. The system reconfiguration password

System Configuration Using @GENPART

The process of master initialization, discussed previously, creates a limited BASIC-2 system having a single partition with all of the user memory controlled by Terminal 1. Only Terminal 1, any local printer attached to it, and the system disk drive are operative at this time. No other system devices are available until a configuration is executed. As a part of master initialization, the system microcode automatically loads and runs the BASIC program file @GENPART from the system platter if such a file exists. (The system always assumes @GENPART to be the name of the system configuration program, whether Wang-supplied or user-written.) If @GENPART is not on the system platter, the READY message is displayed at Terminal 1. When @GENPART is first executed, the parameters from the previously loaded configuration (called current) are loaded; a list of previously saved configurations is displayed, along with a prompt inquiring if a different configuration is to be loaded; and the list of @GENPART options is displayed.

You then enter responses to the prompts displayed by each option and use the function keys to advance from option to option. Refer to the *BASIC-2 Utilities Reference Manual* for a complete description of Generating a Sample Configuration.

The standard Wang @GENPART program allows you the following two basic options.

1. Creating configurations to be executed or stored for future use.

   If you are creating a configuration for the first time or wish to modify a previously defined one, the function keys can be used to load and modify the old configuration or to create a new definition, execute it, and/or store the configuration for future use.

2. Loading and executing previously defined configurations

   If you wish to execute a system configuration (previously defined and stored in the configuration file on the system platter) without making modifications, you can select a configuration from the list of configuration names displayed on the screen and manually execute it. (It is also possible to *modify* the @GENPART utility program so that a specified configuration is loaded and executed automatically when the system is master initialized. Refer to the *BASIC-2 Utilities Reference Manual* for a discussion on Customizing @GENPART.)

Refer to the *BASIC-2 Utilities Reference Manual* for a complete discussion on @GENPART operations.

USING THE DISK/DISKETTE DRIVE

After the system has been initialized, a disk/diskette drive may be used to store programs or data files. An unused disk/diskette, however, must be formatted before it can be used. Ordinarily, a used disk/diskette is not formatted before storing additional files since formatting destroys any information previously recorded on the diskette. For this reason, disks/diskettes containing packaged programs must never be formatted or scratched.

Formatting Disks/Diskettes

You can format fixed disks as well as removable diskettes by using the $FORMAT command in the BASIC-2 language or by running the Wang-supplied Format Disk Platter utility. The Format Disk Platter utility produces the same results as executing a $FORMAT command, but prompts you to be sure that the proper disk is being formatted. This feature enables you to be sure you are not formatting a disk or diskette that has information on it that you want to keep. Refer to the *Multiuser BASIC-2 Language Reference Manual* for instructions on how to use the $FORMAT command. For instructions on how to use the Format Disk Platter utility, refer to the *BASIC-2 Utilities Reference Manual*.

Defining a Catalog Index

Before the first program or data file can be stored on a disk/diskette in the Automatic Cataloging mode, you must create a catalog. The process of creating a catalog is called scratching the disk/diskette because the SCRATCH DISK statement is used to perform the operation. Take care not to scratch a disk/diskette with packaged programs or desired files since a SCRATCH DISK statement overwrites a previous catalog index.

In a SCRATCH DISK statement, you must specify how many sectors are to be reserved for both the Catalog Index (where the file names are stored), and the Catalog Area (where the contents of the files are actually stored). The Catalog Index always begins at the first sector on a disk/diskette (sector numbering starts with zero) and occupies a number of sequential sectors specified by the user. The Catalog Area begins immediately after the Catalog Index and occupies all sequential sectors up to and including the user-specified last, or ending, sector. The end of the Catalog Area is usually specified as the last available sector on the disk/diskette.

The size of the Catalog Index is defined with the LS parameter in a SCRATCH DISK statement. For example, LS = 10 indicates that 10 sectors are to be reserved for the Catalog Index. If no value is specified, the system assigned (default) value is 24. (The first sector, sector 0, can hold 15 file names, and each subsequent sector can hold 16 file names.)

The last sector in the Catalog Area is specified with the END parameter in a SCRATCH DISK statement. For example, END = 1023 indicates that sector 1023 is the last sector to be used for the catalog area. You must specify the END parameter.

To scratch a disk/diskette, perform the following steps.

1. Mount a formatted disk/diskette in the drive.

2. Enter a statement such as:

   ```
   SCRATCH DISK T/D10, LS = 20, END = 1023
   ```

   Press the RETURN key. LS = 20 specifies that 20 sectors be reserved for the Catalog Index; END = 1023 specifies that sector 1023 is the last sector to be used by the catalog. The number of sectors allocated for the catalog index and data storage may vary depending on the user's system and programming requirements. (Refer to the *Multiuser BASIC-2 Language Reference Manual* for further information.)

3. Repeat Steps 1 and 2 for any other disk/diskette that must be scratched. After a disk/diskette is formatted and scratched, it is ready for data and program storage.

Note: *A LIST DC T [/disk address] statement can be used to check the size of the Catalog Index. Entering the statement LIST DC T/D10 after scratching disk D10 in the manner shown above would cause the following to be displayed on the CRT.*

```
INDEX SECTORS = 00020
END CAT.AREA  = 01023
CURRENT END   = 00019
```

BACKING UP THE SYSTEM PLATTER(S)

After loading the system programs, first make a backup of the system platter(s). *At least one copy should be made of each system platter* on another diskette or on a disk in the fixed/removable disk drive in case the original is accidentally damaged or destroyed. You have a choice between using the BASIC-2 disk commands and the operating system utilities. The BASIC-2 commands and the system utilities provide the flexibility of backing up an entire platter or backing up the platter file by file.

BASIC-2 Statements

The following statement copies an entire platter (including scratched files).

```
COPY T/xxx, TO T/yyy,
```

The following statement moves an entire platter (removing scratched files).

```
MOVE T/xxx, TO T/yyy,
```

The disk address of the system platter is xxx and yyy is the address of the new disk/diskette on which the copy is made. This statement copies the entire catalog of the original platter to the destination platter. An example of valid syntax follows.

```
MOVE T/B10, TO T/310,
```

To move a platter one file at a time, use the following statement.

```
MOVE T/xxx, "filename" TO T/yyy,
```

This statement causes only the *named file* to be copied from the Catalog Area of the original platter to the Catalog Area of the destination platter. Two examples of valid syntax follow.

```
MOVE T/D11, "Bonds" TO T/D10,
MOVE T/310, "@@" TO T/B10,
```

The BASIC-2 interpreter and operating system are contained in a single file named @@ on the system platter. Other programs on the system platter may be copied, individually, in a similar manner. (Refer to the *Multiuser BASIC-2 Language Reference Manual* for a complete explanation of the MOVE statement.)

When the system software is provided on multiple diskettes, the above procedures should be used to make a backup of each diskette.

Utilities

The system platter contains a variety of system programs. (The file names of these programs begin with the @ (at) sign.) System programs are stored on the disk in Automatic File Catalog mode. (Refer to the *Multiuser BASIC-2 Language Reference Manual* for an explanation of Automatic File Catalog mode and to the *BASIC-2 Utilities Reference Manual* for a detailed description of these programs.) There are several utilities that accomplish the task of backing up the system platter.

1. Move System Files (@MOVE)

   This utility moves specified system files from one platter to another. You must verify each move. Existing files on the destination platter can be overwritten in order to update system programs. To use this utility, select the system platter and press RETURN. Type

   ```
   SELECT DISK hhh and press RETURN
   LOAD RUN "@MOVE" and press RETURN
   ```

   Another way to access this utility is to type

   ```
   LOAD RUN and press RETURN
   ```

   Press the Space Bar to move the cursor to MOVE SYSTEM FILES and press RUN.

   The program provides prompts that require you to specify the details of the file transfer. If the system platter is larger than the destination disk, disk error D85 appears, "Catalog Index Full". To continue, you should mount a new disk platter and create a new catalog.

2. Backup Disk Platter (@BACKUP)

   This utility copies the contents of a single disk platter to another platter or to a series of smaller platters. This utility moves the entire contents of the origin platter including the Catalog Index, if one is present. To retrieve the contents of the backup platter(s), you must use the @RECOVER utility.

3. Move File (@MOVEFIL)

   This utility provides a general file move capability for both system and user files. Specified files are moved from one platter to another. If necessary, a file can span from one output platter to another.

CHAPTER 4
OPERATING THE CS

INTRODUCTION

The CS is a versatile system that can be used as both a programmable data processor and a convenient and powerful calculator using the Wang BASIC-2 language. Wang BASIC-2 is an interactive programming language that uses many English words (such as PRINT, READ, and STOP). Although such words are given a special and clearly defined meaning when used in a BASIC-2 program, their usage and meaning is similar to that of commonly used English. Detailed information about the Wang BASIC-2 instruction set is presented in the *Multiuser BASIC-2 Language Reference Manual*. Programming techniques are discussed in detail in *Programming in BASIC*.

This chapter introduces you to the operation of the CS. It includes general instructions for running preprogrammed software modules and provides information on using the CS workstation as a calculator. Additionally, this chapter presents a sample BASIC-2 program, along with an explanation of the program itself and the procedures for entering, running, and saving the program.

RUNNING SOFTWARE PACKAGES

Preprogrammed software packages may be run easily on the CS. First, insert the program diskette in the Model DS diskette drive. Most Wang software packages contain starting modules named START, although some have starting modules with other names. Check the manual supplies with your software package for the required module name.

Loading the Starting Module

When software packages contain the file START, enter the following command.

Press SHIFT and RESET

SELECT DISK hhh

The diskette containing the software package is hhh. When the following statement is entered, it clears memory, loads the starting module, and begins program execution.

LOAD RUN (RETURN)

If the software package does not contain a program module named START, enter the following commands.

```
CLEAR (RETURN)
LOAD RUN [/disk address], "xxxxx" (RETURN)
```

The name of the starting module is "xxxxx".

Entering Data

During program execution, operator prompts usually appear on the CRT. Remember the following facts when responding to these prompts.

- In general, a displayed question mark indicates that the system expects a keyboard entry.
- If a numeric value is requested, the system permits a maximum of 13 digits, a decimal point, a sign, and a signed 2-digit exponent to be entered. (The program itself may impose more restrictive limits on a response.) If entering a sign, it must precede the digits; the letter E is used to mark the beginning of the exponent. The following are examples of system-acceptable numeric entries.

```
 25.15
-79.5
  4.56E4
+23.2437E-12
```

- If an alphanumeric response is requested, the system accepts any keyboard characters except some sequences containing commas and double quotes. (Refer to INPUT statement in the *Multiuser BASIC-2 Language Reference Manual*.) Most software packages remove the restrictions imposed by the INPUT statement by using the KEYIN statement or the LINPUT verb.

- The Edit mode keys may be used to correct a response before entering a response to a prompt (see Chapter 2).

- When you have typed in a response and it appears on the display in the desired form, press the RETURN key to enter the response and terminate the keyboard entry.

- After a response is entered, the system may signal an error condition with a message in the form of ERR Yxx, where Y represents a letter prefix of the error class and xx represents an error number (see Appendix E for a discussion of system execution error messages). An English description of the error is also displayed. The question mark reappears on the screen; this indicates that the entered response was not acceptable to the system. To continue, check the form of the response and enter another.

USING THE CS IN IMMEDIATE MODE

The CS can perform as a calculator when a BASIC-2 statement is entered without a preceding line number. When such a statement is entered and executed, the CS is in *Immediate mode*. To perform quick calculations, you can enter unnumbered, single, or multiple statement lines in Immediate mode, using the PRINT statement to display the results. If the system detects no syntax errors, it executes an Immediate mode statement when the operator presses the RETURN key, as in the following example.

```
:PRINT 25 + 273 (RETURN)
:298
```

In this case, when you press the RETURN key, the line is immediately executed, the expression 25 + 273 is evaluated, and the result is displayed. The line is not saved in memory because it was not preceded by a line number when entered.

Multiple statement lines (individual statements separated by colons) are acceptable in the Immediate mode, as in the following example.

```
:PRINT 3 + 8:  PRINT 15/9:  PRINT 144/4.2
```

Upon execution, the following is displayed on the screen.

```
11
1.666666666667
34.28571428571
```

The function keys along the top of the 2236DE Terminal can be used to edit an Immediate mode statement. To activate the editing capabilities of these keys, first press the EDIT key, located in the top right corner of the terminal keyboard. For example, pressing the EDIT key then pressing the RECALL key causes the most recently entered Immediate mode statement to appear on the screen where it can be edited and re-executed. The capabilities of these keys are explained in the *BASIC-2 Utilities Reference manual*.

Arithmetic Operations

The CS can operate on numeric quantities as large as 10^{100} and as close to zero as 10^{-99}. The quantities may be positive or negative. Quantities within this range can be represented by a maximum of 13 digits, a decimal point and sign, and a two-digit positive or negative integer exponent. The letter E in a quantity means times ten raised to the power of.

The arithmetic symbols, or operators, of BASIC-2 are the following.

+ addition
- subtraction
* multiplication
/ division (5/4 reads 5 divided by 4)
↑ exponentiation (5 ↑ 4 reads 5 raised to the 4th power)

Any number of variables and constants in an expression can be linked together by arithmetic symbols. Some simple expressions, using arithmetic symbols, are underlined as follows.

```
PRINT 12/7
PRINT 23*110000*32.1*4
PRINT 63/22.5E23
PRINT 4↑23
```

Expressions using the same arithmetic operators are evaluated left to right. For expressions with mixed arithmetic operators, the following priorities of evaluation are observed.

All exponentiation (↑) is performed left to right.
All multiplication and division is performed left to right.
All addition and subtraction is performed left to right.

Not all combinations of constants and variables connected by arithmetic operators are valid expressions. In order for an expression to be valid, it must be capable of being evaluated in the stipulated sequence. At each state of evaluation, the operation to be performed must be defined for the given values, and must yield a valid numeric quantity. For example, the expressions in Table 4-1 are invalid for the reasons shown.

Table 4-1. Invalid Arithmetic Expressions

Invalid Expression	Reason
(3.4E26 4) /9.7E17	3.4E26 4 yields an invalid numeric quantity of 10^{100}.
17/((4*5)-20)	After evaluating ((4*5-20), the system attempts to divide 17 by 0, an undefined operation.
(-3) ↑ 3.5	The exponentiation (↑) operation is undefined for non-real results.

If the result of evaluating an expression or a portion of an expression yields a quantity Q in the following range, the value Q is zero.

-10^{-99} Q 10^{-99}

BASIC-2 Functions

A BASIC-2 function is a special type of instruction that accepts a given value as an argument and returns a unique value as a result. The result of a function can be displayed in Immediate mode if the function is included in a PRINT statement. PRINT SQR(55), for instance, prints the square root of 55.

Some BASIC-2 functions that may be useful in Immediate mode are listed in Table 4-2.

Table 4-2. BASIC-2 Functions

Function	Result
ARCCOS	Returns the arccosine of an expression.
ARCSIN	Returns the arcsine of an expression.
ARCTAN (or ATN)	Returns the arctangent of an expression.
EXP	Finds the value of e raised to the value of the expression.
LOG	Returns the natural logarithm of an expression.
SIN	Returns the sine of an expression.
SQR	Finds the square root of an expression.
TAN	Returns the tangent of an expression.

Other BASIC-2 functions and keywords are explained in detail in the *Multiuser BASIC-2 Language Reference Manual*.

Numeric Variables

When used in Immediate mode, the CS can assign values to variables and perform operations on those variables. A numeric variable in BASIC is designated by a letter of the alphabet A-Z or a letter followed by a single digit such as A0, A1-A9, B0, B1-B9, C0, C1-Z7, Z8, and Z9. The letters in variable designations must be uppercase. Variables such as A and A0 are distinct. There are 286 numeric variables available for use in BASIC-2. A numeric variable may be assigned any legal numeric value.

Variables retain their values until they are explicitly cleared from memory or assigned new values. The commands CLEAR, RUN, and LOAD RUN are some of the commands that cause variables to be cleared from memory.

Care should be taken when assigning variables in Immediate mode since Immediate mode commands can change the values stored in variables created by a program. Immediate mode operations involving variables should be performed only at times when it is known that disturbing the values stored in the program-created variables causes no harm. While running a program, it is best to refrain from performing Immediate mode calculations that involve variables, unless the author of the program has specifically documented that Immediate mode operations do not adversely affect the operation of the program.

For further information on BASIC-2 variables, refer to the *Multiuser BASIC-2 Language Reference Manual* or *Programming in BASIC*.

A SAMPLE PROGRAM

It is not difficult to program using BASIC-2 on the CS. The simple syntactical rules and most frequently used commands can be learned very quickly from reading *Programming in BASIC*. With this knowledge, even a beginner can write simple, functional programs. Further reading and practice provides facility with the more advanced BASIC-2 commands and features. The *Multiuser BASIC-2 Language Reference Manual*, the prime source of the wide variety of commands, functions, and statements to which the CS responds, discusses the many BASIC-2 programming features that can make programs more powerful and efficient.

This section discusses how to enter and run a simple program, list it on the CRT screen and printer, save it on disk, retrieve it from disk and modify it, and resave it on disk.

An Adding Machine Program

A program is a set of instructions enabling a computer to receive data, manipulate it, and return it to the operator in a desired form. The following program has been written to make the system function like a simple calculator: when the program is executing, the user enters two numbers on the keyboard and indicates the operation he wishes to perform upon them. The system displays the answer on the CRT screen.

A program line is entered by typing in a line number, followed by one or more BASIC-2 statements and their operands. Each line is terminated by pressing the RETURN key. When the RETURN key is pressed, the entire line is saved in memory.

It is customary to number program lines in multiples of 10. This enables a programmer to insert new lines between already existing lines by assigning the new line an intermediate statement number. When the operator enters such a line on the keyboard, the computer places it in its proper place in the program. The computer normally executes these statements in sequential order, though several BASIC-2 statements allow the operator to change the order of execution.

Type in the following program on the keyboard, pressing the RETURN key after each line is entered.

```
10  DIM A$1
20  INPUT "WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM", A$
30  INPUT "ENTER FIRST NUMBER", A
40  PRINT A; A$; "";
50  INPUT B
60  IF A$= "+" THEN GOTO 170
70  IF A$= "-" THEN GOTO 180
80  IF A$= "X" THEN GOTO 190
90  IF A$= "/" THEN GOTO 200
100 GOTO 20
110 PRINT A; A$; B; "EQUALS"; C
120 INPUT "DO YOU WANT ANOTHER OPERATION", B$
130 IF STR(B$,1,1) = "Y" THEN 20
140 IF STR(B$,1,1) = "N" THEN 160
150 GOTO 120
160 END
170 REM ADDITION:  C=A+B: GOTO 110
180 REM SUBTRACTION:  C=A-B: GOTO 110
190 REM MULTIPLICATION:  C=A*B: GOTO 110
200 REM DIVISION:  IF B< >0 THEN 220
210 PRINT "DIVISION BY ZERO IS AN ILLEGAL OPERATION, START AGAIN
    PLEASE": GOTO 20
220 C=A/B: GOTO 110
```

Listing the Program

To review the program entered in memory, enter the LIST command.

```
LIST
```

This statement lists the entire program on the CRT a screen at a time. When output is directed to some device other than the CRT, the LIST S command lists the program a screen at a time. You must press the RETURN key when ready to review each subsequent screen. LIST [line number] lists the specified line only; LIST 10, for example, lists only line 10.

To obtain a hardcopy listing of the program on a printer, turn on and manually select the printer (press its SELECT switch). (When the printer is selected, its SELECT switch indicator lamp is illuminated.) Then, enter the following statement from the keyboard to select the printer for listing.

```
SELECT LIST 215
```

This statement changes the default output address for the I/O class parameter LIST from the primary address 005 (the CRT) to the address 215 (the system printer). Next, enter LIST and press the RETURN key. Program listings formerly displayed on the CRT are now listed on the printer. To again obtain program listings on the CRT, enter the following statement.

```
SELECT LIST 005
```

Next, enter LIST and press the RETURN key. If a local printer is attached to the rear of the 2436D-type interactive terminal, it may be accessed at address 204. Further discussion of the SELECT statement and its various parameters can be found in the *Multiuser BASIC-2 Language Reference Manual* and in the individual printer manuals.

Running the Program

Once the program is saved in memory, it can be executed by pressing the RUN key. Note that the program text is saved in memory each time that RETURN is pressed. When you press the RUN key, followed by the RETURN key, the screen displays the following prompt.

```
WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM?
```

As an example, enter +. The screen displays the following.

```
ENTER FIRST NUMBER?
```

Enter 25. The following is displayed.

```
25 + ?
```

Enter 75. The following appears on the screen.

```
25 + 75 EQUALS 100
DO YOU WANT ANOTHER OPERATION?
```

Enter YES. The program once again asks, WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM?

The program has been designed to recognize an attempt to divide by zero. Enter /. When the screen displays ENTER FIRST NUMBER?, type in 100. The following appears on the screen.

```
100 /?
```

Enter 0. The screen displays the following information.

```
DIVISION BY ZERO IS AN ILLEGAL OPERATION.  START AGAIN PLEASE.
WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM?
```

The program continues processing until you enter NO or N to the question, DO YOU WANT ANOTHER OPERATION?

Explanation of the Program

The adding machine program instructs the computer how and where to store information entered by the operator or produced by processing, sends prompts to the operator on the CRT screen, accepts operator input from the keyboard, processes entered data according to a defined formula, and displays results on the screen. The following section examines the program line by line and discusses the use and purpose of each BASIC-2 command.

Line 10:

```
10 DIM A$1
```

The DIM statement in line 10 is a dimension statement, limiting the length of the information which can be stored in variable A$. The only reason the program includes this statement is to guarantee that, if an operator entered an inappropriate response such as + + + + + + + + to the question WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM?, the adding machine program would not generate a prompt on the screen such as the following.

```
44 + + + + + + + + ?
```

Because of the dimension statement, the system cuts off the six extra plus signs and stores a variable of only one character in length. As a result, the screen shows the following.

```
44 + ?
```

There are many other uses of the DIM statement. Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the DIM statement.

Lines 20 and 30:

```
20 INPUT "WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM",A$
30 INPUT "ENTER FIRST NUMBER",A
```

The INPUT statements in Lines 20 and 30 are the computer's way of receiving information from an operator. The computer stores the information entered by the operator in variables, such as A$ and A in this program. A$ is called an alphanumeric variable. Alphanumeric variable names can be distinguished from numeric variable names by the presence of a dollar sign ($) immediately following the variable name. Alphanumeric variables may contain any kind of character: letters, digits, or special characters, but may not be used in mathematical computations, because they do not necessarily contain numbers. A$ in this program is used to store the character +, -, X, or /. (Refer to the *Multiuser BASIC-2 Language Reference Manual* for more information on variables.)

The phrase in quotes is called a literal. Here it is used as a prompt to the operator to enter the first number of the arithmetical operation. The INPUT statement automatically displays a question mark following the prompt. BASIC-2 does not require the literal in the INPUT statement. If the literal is omitted, the INPUT statement simply displays a question mark. A lone question mark is of little use to an operator who does not know the program, so it is a good idea to always provide prompts. The literal in the INPUT statement is one means of providing prompts.

Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the INPUT statement.

Lines 40 and 50:

```
40 PRINT A; A$;" ";
50 INPUT B
```

The PRINT statement causes values and literals to appear on the CRT screen. Additionally, its punctuation can determine the spacing of the printed material. A comma causes elements to be displayed in zones, each 16 characters wide. This is ideal for charts, where figures must be arranged in columns and rows, but it is not convenient for this program. A semicolon is used in this program, because it causes no additional spaces to be placed between printed elements.

In line 40, the PRINT statement instructs the computer to display the current value of variables A and A$. As in the INPUT statement, literals (those characters and phrases contained in quotation marks) are displayed on the screen as written in the program line, even if the literal is a blank space, as in line 40.

Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the PRINT statement.

The INPUT statement in line 50, an example of an INPUT statement not containing a literal, causes the question mark to appear on the screen. For example, when the program is run as in the previous section above, 25 + ? appears. The value of A is 25, and + is the value of A$. The space after the addition sign results from the coding of the literal space, " ", in line 40. The question mark results from the INPUT statement and is placed immediately after the literal space due to the semicolon in line 40.

Lines 60-90:

```
60 IF A$ = "+" THEN GOTO 170
70 IF A$ = "-" THEN GOTO 200
80 IF A$ = "X" THEN GOTO 210
90 IF A$ = "/" THEN GOTO 220
```

It is not always desirable to have the computer execute program instructions in statement number sequence. Two ways of altering the sequence of program execution are demonstrated in the adding machine program. Lines 60, 70, 80, and 90 use the IF ... THEN statement to cause execution to be diverted to the portion of the program that handles the appropriate arithmetic operation specified in variable A$. Another statement, the GOTO statement, is used later in the program to redirect the order of program execution.

Lines 60, 70, 80, and 90 test the value entered for the alphanumeric variable, A$, that is, they determine what operation you wish to perform on the values assigned to variables A and B. This test enables the computer to branch to a special routine for each operation. In the special routine, the computer can compute the solution of the operation, store it in a variable, and return it to be processed in line 110 for display upon the screen. When A$ has a value of +, the program goes to the addition routine at line 170. When A$ has a value of -, the program goes to the subtraction routine at line 180. When A$ has a value of X, the program goes to the multiplication routine at line 190. When A$ has a value of /, the program goes to the division routine at line 200.

Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the GOTO statement.

Line 100:

```
100 GOTO 20
```

If you enter any value other than +, -, X, or / for A$, the computer falls through statements 60, 70, 80, and 90 to line 100, where the GOTO statement sends the program back to line 20. This GOTO statement ensures that an operational symbol has been entered.

Lines 170-190:

```
170 REM ADDITION:  C=A+B:  GOTO 110
180 REM SUBTRACTION:  C=A-B:  GOTO 110
190 REM MULTIPLICATION:  C=A*B:  GOTO 110
```

These three lines are examples of multistatement lines. In each, three short statements are grouped together, separated by colons. Short statements are grouped together to keep the program from growing too large and to aid in program clarity. The purpose of the REM statement is to document the program. Here, REM ADDITION labels the portion of the program that computes addition problems. A REM statement is a nonexecutable statement ignored by the computer when the program is run, but displayed whenever a program is listed.

Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the REM statement.

Lines 200-220:

```
200 REM DIVISION:  IF B< >0 then 220
210 PRINT "DIVISION BY ZERO IS AN ILLEGAL OPERATION.  START AGAIN
PLEASE":  GOTO 20
220 C=A/B:  GOTO 110
```

The division routine is given three lines to ensure that you do not enter zero as a divisor. If Lines 200 and 210 did not exist, the computer would cause a Computational error (C62 or C63) to appear on the screen and the program would immediately terminate.

The IF ... THEN statement is another way by you can instruct the program to process statements out of sequence. Here, line 200 tests to determine whether the divisor, B, is greater than or less than zero. If the divisor is not zero, this statement is true, and the program skips to line 220, where the program computes the quotient and is sent back to line 110 to display it. If the divisor is zero, this statement is false, and the program falls through to line 210, the message is displayed, and control is sent back to line 20. As result, the division computation is not performed.

Refer to *Programming in BASIC* and the *Multiuser BASIC-2 Language Reference Manual* for more information on the IF ... THEN statement.

line 110:

```
110 PRINT A; A$; B; "EQUALS "; C
```

In line 110, as in line 40, variable values and a literal are included in a PRINT statement. Here, there are three numeric variables (A, B, and C), an alphanumeric (A$), and a literal ("EQUALS").

line 120:

```
120 INPUT "DO YOU WANT ANOTHER OPERATION", B$
```

line 120 requests a value for alphanumeric variable B$.

lines 130-160:

```
130 IF STR(B$,1,1) = "Y" THEN 20
140 IF STR(B$,1,1) = "N" THEN 20
150 GOTO 120
160 END
```

Lines 130 and 140 introduce another function of BASIC-2 that is, isolating a particular character from an alphanumeric string. In this case, the program examines the value of the first character in variables B$ for the character "Y" or the character "N". By examining the first character of the string instead of every character, lines 130 and 140 enable you to continue the program by entering the abbreviated answer Y or N as well as the answer YES or NO to the question asked in line 120.

In short, line 130 means: if the first character from the first character of alphanumeric variable B$ is equal to Y, then go to line 20. If you enter Y or YES (or, because of first character isolation, any word beginning with Y), the first character is Y, and sends the program back to line 20. A similar logic is exhibited in line 140. In this case, if the first character of the response is N, the program is sent to statement 160, an END statement, causing the program to end.

However, if the first character of B$ is other than Y or N, the program falls through to line 150. There, the program is sent back to line 120. This procedure guarantees that the operator responds correctly, at least insofar as the first character of the response is either Y or N.

Refer to the *Multiuser BASIC-2 Language Reference Manual* for more information on the STR function.

Saving the Program on Diskette

To save this program on diskette, load a new diskette into the diskette slot of the diskette drive, format it, and establish a catalog as detailed in Chapter 3 of this manual. Enter the following command.

```
SAVE T/xxx, "ADDING"
```

The device address of the diskette where the program is to be saved is xxx. Then enter the command, LIST DCT. The following appears on the screen.

```
INDEX SECTORS = 00030
END CAT.AREA  = 03977
CURRENT END   = 00029

NAME    TYPE    START    END    USED    FREE
ADDING   P      00029    00033  00005   00000
```

Listing the Catalog Index ensures that the program you saved is in fact contained on the diskette.

Refer to the *Multiuser BASIC-2 Language Reference Manual* for more information on saving programs on disk.

Retrieving the Program from Diskette

First, type in the following command.

```
SELECT DISK xxx
```

The device address of the disk containing the program to be retrieved is xxx. The system selects the system diskette drive. Then type the following command.

```
LOAD RUN "ADDING"
```

The adding machine program is first loaded, and then run.

Modifying the Program

There are several ways in which the adding machine program can be improved. One of the program's major difficulties occurs when you enter something other than +, -, X, or / in response to the request to enter the type of operation to be performed. If, for instance, you entered 999 in response to WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM?, two things occur. Firstly, the last two nines are cut off, due to the DIM statement in Line 10. Secondly, when line 40 is executed, 9 actually appears in place of an operational sign on the CRT screen. The program continues processing until it reaches line 100. At that point, since the value of A$ is neither +, -, X, nor /, the program loops back to line 20 to request the type of operation again.

The LINPUT statement is often used to avoid the first situation, the POS function can rectify the second.

First, replace line 20 with

```
:20 LINPUT "WHAT TYPE OF OPERATION (+,-,X,/) DO YOU WANT TO PERFORM"-A$
```

Then, insert line 25.

```
:25 IF POS("+-X/"=A$) = 0 THEN GOTO 20
```

Finally, delete line 100 by typing the following.

```
:100
```

When this version of the program is run, the LINPUT statement in line 20 causes an underscore to follow the prompt to enter the operational sign. The LINPUT statement also ensures that you only enter one character in response to this prompt.

The POS function causes the system to locate the position of the variable, A$, in the character string "+-X/". If you enter + for the variable A$, POS ("+-X/"=A$) equals 1, since + is in the first position of the character string in the POS function. Likewise, -, X, and / equals 2, 3, and 4, respectively. A character other than one of these has no position in the character string and is equal to 0. If you enter such a character, the IF ... THEN statement in Line 25 causes the program to go back immediately to line 20 and request the operational sign to be entered again.

Refer to the *Multiuser BASIC-2 Language Reference Manual* for more information on the LINPUT statement. Refer to the *Multiuser BASIC-2 Language Reference Manual* for more information on the POS function.

Resaving the Modified Program

To resave the modified program,

```
RESAVE T/D10, "ADDING"
```

T/xxx is the device address of the modified program. This command copies the new version of the program called ADDING over the old version, saving it under the same name. Subsequently, when LOAD RUN DCT/xxx "ADDING" is executed, the new version of the adding program is loaded.

CHAPTER 5
SYSTEM ERRORS AND ERROR-RECOVERY PROCEDURES

TYPES OF SYSTEM ERRORS

There are two different types of system errors that may occur on the CS: CPU hardware errors that indicate malfunctions in the CPU hardware (registers, user and control memory, etc.), and general system errors that indicate that an illegal operation has been attempted or a desired operation cannot be performed. Correspondingly, the CS CPU performs two different types of system error diagnostics: the CPU/hardware error diagnostics and the general system error diagnostics. The CPU/hardware errors are discussed in this chapter.

If a CPU/hardware error occurs write down the error display that always appears on Terminal 1. Then, if the error cannot be corrected by the general recovery procedures outlined in this chapter, or if the error recurs intermittently, the error information enables the Wang service representative to quickly locate the malfunctioning components. When the general procedures fail, try the specific procedures discussed in Appendix E. If the procedures in Appendix E fail or error conditions recur, call your Wang service representative.

The CPU hardware error diagnostics are designed to detect and report any malfunction that occurs in the CPU hardware (e.g., registers, and user and control memory). The CPU error-checking routines of the CS treat the entire CPU as a single unit since memory errors can affect all users of the system. The current configuration scheme does not affect the error-checking routines or error-recovery procedures. Whenever a location in memory is accessed, the system automatically performs a parity check to ensure that the accessed memory contents are valid and the accessed location is not faulty. Additionally, whenever RESET or CLEAR is executed, the entire memory is automatically verified, and you are alerted to any memory error condition. If a hardware error is discovered, all the terminals sound an audible tone and a system error message is displayed at Terminal 1. Although there are several courses of action, the hardware errors require that a corrective procedure be performed at Terminal 1.

General error messages, however, are designed to alert you that an illegal operation has been attempted or a desired operation cannot be performed. Those general system errors which are not hardware related can be handled under program control in a user-application program. The general error routines are handled separately for each partition. The error message is displayed on the terminal currently attached to the partition in which the error occurred. If the error occurs while the partition is running a background job, the error-message display is suppressed until the terminal is released to the background partition. General error messages are signaled by an audible tone and a display on the CRT of the terminal attached to the partition in which they occur. Error messages consist of a two-digit number with a letter prefix accompanied by an arrow which points to the approximate location of the error in an Immediate mode or Program mode line; it also includes an English description of the error. The types of general error diagnostics performed by the system and the techniques for recovering from such errors are described in detail in the *Multiuser BASIC-2 Language Reference Manual*.

CPU Hardware Errors

Several possible system error messages can be reported when CPU/hardware errors occur during master initialization or once the operating system has assumed control of program execution. These hardware errors cannot be handled under program control. Since a hardware error represents a system hardware malfunction that may be transient, any information that the system furnishes or receives after the hardware error has occurred may be invalid. The procedures for recovery from these errors are detailed in the following section.

Master Initialization Error Recovery

The general procedure to recover from master initialization errors appears below. When these general procedures fail, call your Wang service representative.

1. When the system's power is turned on, the bootstrap fails to display the complete MOUNT SYSTEM PLATTER PRESS RESET message on the CRT. This condition usually indicates a CPU-related or I/O-related error. Turn the system's power off and check the cabling and device addresses. Wait five seconds, then power the system up again. Do not turn the terminal off when performing this procedure.

2. Respond to the KEY SF'? message by pressing a function key. The following conditions may exist.

   a. The hex-digit display of the function key did not appear on the CRT, implying that the function key was not pressed sufficiently or that the wrong function key was pressed. Press RESET and then press the desired function key again.

   b. The KEY SF'? message reappears upon the CRT. This error implies that the specified system file could not be located on the specified disk. Verify that the system platter is properly mounted at the specified address, and press the desired function key again.

   c. The ***SYSTEM ERROR (DISK 00XX)*** PRESS RESET messages appears. This error message implies that the wrong function key was pressed or that a disk hardware error occurred while the bootstrap was trying to load the disk file specified by the function key. Make sure the platter is properly mounted, and press RESET and then the desired function key. If the general procedure fails, see the table of disk errors in Appendix E for a specific recovery procedure.

General Hardware Error Recovery

A system error message is reported by an audible tone at each terminal and a display on the CRT of Terminal 1 whenever the operating system detects a memory error during its normal operation. A memory failure causes the following message to appear at Terminal 1.

```
***SYSTEM ERROR (MMMM XXXX)***
PRESS RESET

                AECM - Addressing Error Control Memory
                AEDM - Addressing Error Data Memory
                BECM - Bit Error Control Memory
                BEDM - Bit Error Data Memory
where MMMM =    PECM - Parity Error Control Memory
                PEDM - Parity Error Data Memory
                REDM - Read Error Data Memory
                VECM - Verify Error Control Memory
                VEDM - Verify Error Data Memory

and   XXXX = Location and Nature of Error
```

The general procedure used to recover from system errors in user and/or control memory is shown below. When the general procedure outlined here fails, see the table of system and disk errors in Appendix E for more detailed recovery procedures. If the detailed recovery procedures fail or a system error condition recurs, call your Wang service representative.

Press RESET in response to the PRESS RESET message on line 1 of the CRT, and then choose one of the following three procedures in response to the SF Key message on the CRT.

1. Select the diagnostic from the start up menu, or press SF'16-'19 to load the diagnostic menu from which you may choose the applicable diagnostic (see the section in this chapter that discusses Hardware Diagnostics). Although this procedure is recommended, execute Procedure Number 3 if the current contents of user memory cannot be reproduced (i.e., must be salvaged). If no errors are discovered during the execution of the hardware diagnostics, the error condition can be presumed transient. Proceed with Procedure Number 2. Note that with Procedure Number 2, the contents of user memory are erased.

2. First, power down, then power up, then press SF'00-'03 to load BASIC-2 from Disk D11 (310), D10 (B10), D21 (320), or D20 (B20), respectively, in order to start over again. This is the recommended procedure when the system validity has been compromised by an unknown system error, but the condition at fault is suspected to be temporary and does not require execution of a diagnostic routine. For example, the CPU was jarred or a cable was stepped on or shaken loose. This option should be exercised when there is no need to save the current contents of user memory. The system must now be reconfigured with the @GENPART utility to resume multiuser operation.

3. Press SF'15 to resume normal operation, using the operating system and application program currently loaded. This option should be exercised when the user-memory contents cannot be duplicated and must therefore be salvaged. The system will return (if possible) to Console Input mode and the READY BASIC-2 message will be displayed. Since user memory was not erased, you can now determine the point of program execution, when the system error was encountered in the current application program, by printing out the key variables. However, since a reset has been performed on one of the partitions, program execution may not be able to be continued from that point, but must be restarted with a Run command. This procedure is undertaken at your risk, however, since the hardware error may recur and the operating system may not function properly.

USING THE HARDWARE DIAGNOSTICS

As with master initialization, the hardware diagnostics can only be executed at Terminal 1. The hardware diagnostics are a set of programs that exhaustively test the CPU hardware components and attempt to identify any malfunctions. The diagnostics can be run only after master initialization or a system error has occurred. The diagnostics run continuously until RESET is pressed or an error is detected. When an error is detected, diagnostic processing stops, or an error message is displayed. For a discussion of possible error causes, see Appendix E. If a hardware error recurs, call your Wang service representative.

A hardcopy listing of the diagnostic messages can be obtained by turning on the printer attached to the CPU during diagnostic execution. The printer attached to the CPU must be assigned address 215 or 204 and SELECTed (press SELECT on the printer). Obtain a hardcopy trace of the diagnostic run should be obtained because it will help your Wang service representative locate and correct any problems. If no errors occur, they will print under the appropriate diagnostic test title.

Since the diagnostic programs destroy the contents of user memory, save all valuable programs and data on disk prior to running the diagnostics.

The diagnostic programs of this system should be executed

- At least once every 60 to 100 hours of light-to-normal operation
- Whenever errors cause a hardware malfunction to be suspected
- Whenever a Wang service representative requests their execution

Execute the diagnostic routines as quickly as possible when you observe a suspected problem, to ensure that the environmental conditions of the test most closely resemble those under which the problem occurred.

Random, intermittent problems and permanent component failure are often due to static electricity, EMI, temperature and humidity extremes, and excessive airborne dust and dirt. Prolonged operation in a poor environment produces permanent failure. Every effort should be made to secure an environment which, if not optimal, at least provides the operating conditions required for satisfactory system performance. Appendix D gives the required environmental conditions.

Table 5-1 provides a functional description of each diagnostic. Press RESET and then press the appropriate function key (SF'16 - '19) after each to select another diagnostic. To reselect BASIC-2, press SF'00 - '05, or select the diagnostic from the start up menu and use menu control for the diagnostic selection.

Table 5-1. Functions of the User Diagnostic

Diagnostic	Function
CPU	Tests the CS processor
Control Memory	Tests control memory
Data Memory	Tests data memory

Loading the Diagnostics Menu

If the system is on, turn it off. Then, turn the system on, press RESET, and then press one of the following function keys.

SF key '16 - Load diagnostics from Disk D11 (or 310)
SF key '17 - Load diagnostics from Disk D10 (or B10)
SF key '18 - Load diagnostics from Disk D21 (or 320)
SF key '19 - Load diagnostics from Disk D20 (or B20)

The following display appears upon selection of the Diagnostic menu (SF key '16, '17, '18, or '19).

```
KEY SF'
USER DIAGNOSTIC MENU
'00 CPU DIAGNOSTIC              '02 DATA MEMORY DIAGNOSTIC
'01 CONTROL MEMORY DIAGNOSTIC
```

Press the appropriate function key to load the desired diagnostic.

The CPU Diagnostic

This diagnostic tests the CS processor. This test runs continuously until either an error occurs (the pass number stops incrementing) or RESET is pressed. If an error occurs, call your Wang service representative. When a sufficient number of successful test passes have occurred (at least five to ten), press RESET. The Diagnostics menu can be restored to the screen, or BASIC-2 can be reloaded by pressing the appropriate function key.

The Control Memory Diagnostic

This diagnostic is designed to test the control memory. These control memory diagnostic tests are repeated until RESET is pressed. Error messages are displayed or printed whenever memory failures are discovered. When a sufficient number of successful test passes have occurred (at least five to ten passes), press RESET. The Diagnostics menu can then be restored to the screen, or BASIC-2 can be loaded by pressing the appropriate SF key.

The Data (User) Memory Diagnostic

This diagnostic is designed to test data (user) memory. The user memory diagnostic tests are also repeated until RESET is pressed. Error messages are displayed or printed whenever memory failures are discovered. When a sufficient number of successful test passes have occurred (at least five to ten passes), press RESET. At the completion of the diagnostic tests, you can pass control to the BASIC-2 operating system by pressing RESET, then pressing the appropriate function key to load the operating system.

APPENDIX A
SYSTEM SPECIFICATIONS

CS CPU SPECIFICATIONS

CPU Size

Height

23.0 in. (58.4 cm)

Width

15.0 in. (38.1 cm)

Depth

15.8 in. (40.0 cm)
9 I/O Ports

Weight

50 lb (22.5 kg) -- No I/O options
70 lb (31.5 kg) -- Fully loaded

Memory Size

32K bytes -- Control memory
128K or 512K -- User memory

Power Requirements

115 V ± 10%, 50 or 60 Hz ± 1 Hz - 3.0 A
230 V ± 10%, 50 or 60 Hz ± 1 Hz - 1.5 A
345 W

APPENDIX B
TERMINAL CABLES

TERMINAL/CPU CABLE

One 8-ft (2.4 m) cable to power source. One 25-ft (7.6 m) direct connection cable is provided with each terminal, unless an optional direct connection cable is ordered for a terminal. Nonextendable cables (see Table B-1) are available for direct connection up to 2,000 ft (606.1 M).

Table B-1. Direct Connection Cables

Length in Feet	Length in Meters	Part Number
25	7.6	120-2236-25
50	15.2	120-2236-50
100	30.3	120-2236-1
200	60.6	120-2236-2
300	90.9	120-2236-3
400	121.5	120-2236-4
500	151.5	120-2236-5
600	181.8	120-2236-6
700	212.1	120-2236-7
800	242.4	120-2236-8
900	272.7	120-2236-9
1,000	303.0	120-2236-10
1,250	378.8	120-2236-11
1,500	454.5	120-2236-12
1,750	530.3	120-2236-13
2,000	606.1	120-2236-14

Modem cables are available optionally in lengths of 12 ft (3.7 m), with extensions of 25 ft (7.6 m) and 50 ft (15.2\m). Combined cable distance however, from Wang equipment to a modem is 50 ft (15.2 m) maximum according to EIA standards.

Table B-2. Modem Cables

Length in Feet	Length in Meters	Part Number
12	3.7	120-2227-12
25	7.6	220-0219
50	15.2	220-0220

APPENDIX C
AVAILABLE PERIPHERALS

The CS supports all CS peripherals and most 2200 peripheral devices currently listed on the standard 2200 products price list, except unbuffered and time-dependent devices. For example, some exclusions include certain instrument interfaces, the Model 2209 9-Track Tape Drive and the Disk Controller 22C31 (Triple Controller). A list of supported devices (as of the publication date of this manual) follows.

Model Number	Description
DS	*Data Storage Cabinet*
DS-320	320 KB Diskette
DS-1.2	1 MB Diskette
DS-10R	10 MB Removable Hard Disk
DS-TS	Streaming Tape Diskette
DS-20	20 MB Hard Disk
DS-32	32 MB Hard Disk
DS-64	64 MB Hard Disk
	Disk Drives
2270A	Diskette Drive (IBM 3740 Compatible)
2270A-1	.25 mb
2270A-2	.50 mb
2270A-3	.75 mb
2275	Fixed/Removable Disk Drives
2275-10	
2275-20	
2275-30	
2275-60	

Model Number	Description
2280	Fixed/Removable Disk Drives
2280-1	26.8 mb
2280-2	53.6 mb
2280-3	80.5 mb
	Tape Drives
2209A	Nine-Track Tape Drive - 1600 bits per inch (bpi)
2229	Four-track, 6400 BTI
	Terminals
2436DE	Interactive Terminal
2436DW	Interactive Terminal
2426DW	Interactive Terminal with Expanded Keyboard
2236MXE	Interactive Terminal Processor
2436WP	WOA 2436 Emulation
	Printers
2273-1	Band Printer - 250 lpm
2273-2	Band Printer - 600 lpm
2200/PM018	Daisy Printer - 60 cps
2200/PM017	Matrix Line Printer
2200/PM010	Matrix Character Printer
2200/PM016	Matrix Character Printer
LCS15	Laser Printer
BFT-1	Bidirectional Forms Tractor for 2281W
BFT-2	Bidirectional Forms Tractor for 2281WC
	Communications Controllers
2227B	Buffered Asynchronous Communications Controller
2228B	Synchronous/Asynchronous Communications Controller
2228C	Synchronous/Asynchronous Communications Controller
2228D	Data Communications Controller
2275MUX, 2275 MUXE	Disk Multiplexer Controllers

APPENDIX D
PREVENTIVE MAINTENANCE AND ENVIRONMENTAL CONSIDERATIONS

PREVENTIVE MAINTENANCE

Wang recommends that the equipment be serviced twice a year. A maintenance agreement is available to assure this servicing automatically. If no maintenance agreement is acquired, any servicing must be arranged for by the customer. A maintenance agreement protects the customer's investment and offers the following benefits.

- Preventive maintenance - The equipment is inspected semi-annually for worn parts, adjusted, lubricated, cleaned, and updated with any engineering changes. Preventive maintenance minimizes the chances of unexpected downtime.

- Fixed annual cost - When a maintenance agreement is bought, only one purchase order for service is issued for an entire year and one annual billing is received. Other billing arrangements can be made if desired.

Your local Wang Sales/Service Office can provide further information regarding maintenance agreements.

Note: *Wang Laboratories, Inc., neither honors maintenance agreements for, nor guarantees any equipment modified by the user. Damage to equipment incurred as a result of such modification is the financial responsibility of the user.*

ENVIRONMENTAL CONSIDERATIONS

When the recommended temperature range is exceeded, both component failure rates and the loss of data through distortion of data storage materials are likely to increase.

Airborne comtaminants can accumulate rapidly on the circuit boards and their components, forming a film which not only prevents adequate heat dissipation from the electronic elements, but also creates leakage paths, causing errors in the system signals. Additionally, dust will cause excessive wear in the disk read/write heads and the oxide coatings of storage media. The filters of all HVAC (heating, ventilating, air conditioning) equipment should be cleaned or replaced regularly. In areas where these filters do not sufficiently remove airborne contaminants, an electrostatic air filter should be installed.

Low humidity increases the probability of static buildup, causes oxide shed in data storage materials, and increases the static charge imparted to carpets and clothing. When the operator comes in contact with the system, the resultant static discharge, which could be several thousand volts, will also cause system errors or destruction of data. High humidity rapidly deteriorates paper stock and magnetic disks and tape. Humidifiers or dehumidifiers should be installed in the environment's heating, ventilating, and air conditioning systems as required.

Carpeting to be installed should be a nonstatic variety. Static carpeting that is already installed must be treated with nonstatic spray, or an electrically conductive mat should be installed under the system operating area and be properly connected to an earth ground to prevent static buildup.

Computers and peripherals are susceptible to malfunction due to electromagnetic interference (EMI) from devices such as radio transmitters and industrial motors. EMI can enter the system by conduction through wiring and cabling or by direct radiation. An illustration of EMI is a television which becomes full of "snow" when a car with a poorly tuned engine idles outside (radiated EMI) or someone turns on a hair dryer or vacuum cleaner in the next room (conducted EMI). To minimize such interference, the three-prong AC power line should be dedicated to the system, grounded, properly installed in a steel conduit, and isolated from interference-generating devices like office machines, fluorescent lighting, motors, and HVAC units. If these devices are located near the system area, they must be relocated, repaired, or filtered to ensure that they do not disturb the system. (EMI filters, isolation transformers, and line conditioners should be installed on the system's AC power line.) In cases of high residual EMI, it may also be necessary to shield all peripheral cables.

The recommended operating environment is defined by the following parameters.

Temperature: 60° F to 80° F (15° C to 27° C).

Relative Humidity: 40% to 60%, noncondensing.

Dust: No obvious accumulation in a 24-hour period.

Power: Grounded, noise-free, dedicated 115 or 230 VAC ± 10%, 50 or 60 Hz ± 1.0 Hz.

Interference: All sources of static electricity, extreme magnetism, and EMI shall be controlled.

A list of additional suggestions follows.

- Keep an accurate service history on all equipment.
- Avoid consuming food and beverages, smoking, and performing any other activity not related to operation of the system while near the equipment.
- Avoid touching magnetic disks and tape or exposing them to direct sunlight, strong magnetic fields, or freezing temperatures. If subjected to extremes, allow the media to return to normal operating conditions before use (24 hours is recommended), and test the media thoroughly.
- Return the diskette to its storage envelope whenever it is removed from the drive. Replace the storage envelopes when they become worn, cracked, or distorted. The envelopes are designed to protect the diskette.
- Date the media when first used, since media over two years old should be replaced. Use felt-tip pens to write on the media label. Never use an eraser to alter a label; instead, use a new label.
- Always make a duplicate copy (backup) of tapes and diskettes, and then store the backup safely in a different area.

Note: *Most Wang electrical devices are equipped with an exhaust fan and entry vents. These vents and fan should not be obstructed. Therefore, never place system equipment immediately adjacent to file cabinets or other surfaces that might impede proper air flow.*

APPENDIX E
SYSTEM HARDWARE ERROR MESSAGES AND RECOVERY PROCEDURES

DISK ERROR MESSAGES AND RECOVERY

The following discussion explains the significance of each system error and suggests possible recovery procedures. If these procedures fail, call your Wang service representative. A system error is usually serious enough to warrant executing a control memory diagnostic. However, it may be possible to resume execution of the currently loaded application program by pressing RESET and then SF'15. If the error is reported again, a memory diagnostic should be run to locate the defective memory location.

Several possible disk errors may occur while trying to load disk information. The recommended recovery procedure involves consulting the following description of each possbile disk error to determine if the problem can be corrected and then attempting to reload. Should successive failures occur, call your Wang service representative. All disk errors are more fully documented in the *Multiuser BASIC-2 Language Reference Manual*.

D80 File Not Open

The file operation cannot be performed upon a closed file.

D81 File Full

No more information can be written into the indicated file. Correct the program, or transfer the file to another platter, reserving additional space on the new platter for this file.

D82 File Not Found

The file name does not exist, or a data file was loaded as a program file or a program file as a data file. Ensure that the file name is entered correctly; ensure that the proper disk is mounted; and ensure that the correct disk drive is being accessed.

D83 File Already Exists

The file name already exists in the Catalog Index. Use a different name, or catalog the file on a different platter.

D84 File Not Scratched

A file must be scratched before it can be renamed or written over.

D85 Index Full

The Catalog Index contains no space for new names. Scratch unwanted files and compress the catalog using a MOVE statement, or mount a new disk platter and create a new catalog.

D86 Catalog End Error

The defined Catalog Area ends within the Catalog Index or has no more available space to store information. This usually occurs because a MOVE END statement tries to move the end of the Catalog Area to the area already occupied by cataloged files. Correct the SCRATCH DISK or MOVE END statement, or increase the size of the Catalog Area by executing a MOVE END statement. Alternately, scratch unwanted files and compress the catalog using a MOVE statement, or mount a new disk platter and create a new catalog.

D87 No End-of-File

Because neither a DATASAVE DC END nor a DATASAVE DA END statement recorded an end-of-file record in the file, the DSKIP END statement cannot locate an end-of-file record. Write an end-of-file trailer after the last data record in the file.

D88 Wrong Record Type

The system encountered a program record when a data record was expected or vice versa. Ensure that the proper drive and file is being accessed.

D89 Sector Address Beyond End-of-File

A DATALOAD DC or DATASAVE DC statement accesses a sector address beyond the end-of-file. This error can be caused by a bad disk platter. Press RESET and run the program again. If the error persists, use a different platter or reformat the platter. If the error still exists, inform a Wang Service Representative.

I/O ERRORS

I90 Disk Controller Error

The system aborts the disk operation because the controller responded improperly at the beginning of the operation. Press RESET and rerun the program. If the error recurs, make certain that the disk unit is on and all cables are properly connected. If the error persists, contact a Wang Service Representative.

I91 Disk Drive Not Ready

The disk unit is not ready for access. Make certain that the program addresses the correct disk. Also, make sure that the disk unit is on and in run mode, and all cables are properly connected. Press RESET and rerun the program. If the error recurs, power the disk unit off and then back on and rerun the program. If the error persists, call a Wang Service Representative.

I92 Timeout Error

A device did not respond to the system. If the device is a disk, the system aborts the disk operation. Press RESET and run the program again. If the error recurs, ensure that the disk has been formatted. If the error persists, inform a Wang Service Representative.

I93 Format Error

The system detects invalid sector control information of the disk platter. If a disk operation is in progress, the platter may need to be reformatted. If the formatting is in progress, the surface of the platter may be flawed. Reformat the disk platter; if the error recurs, replace the platter. If the error persists, contact a Wang Service Representative. The error can also occur if the user attempts to access a disk formatted for use on a different type of system.

I94 Disk Controller Error

The system aborts the disk operation because the controller did not receive the disk command correctly. Press RESET and rerun the program. If the error recurs, make certain that the disk unit is on and all cables are properly connected. If the error persists, contact a Wang Service Representative.

For those disks with a format key, this error message also can indicate that the format key is engaged. Disk operations cannot be performed until formatting is turned off with the format key.

I95 Device Error

The disk cannot perform the requested operation. Repeat the operation. If performing a write operation, make certain that the disk is not write-protected. If the error recurs, power the disk off and back on, and again perform the operation. If the error persists, contact a Wang Service Representative.

I96 Data Error

For read operations, the checksum calculations (CRC or ECC) indicate that the data read is incorrect. For disk drives that perform ECC, the attempt to correct errors was unsuccessful. Rewrite the data; the read sector may have been written incorrectly. If read errors recur, reformat the platter.

For write operations, the LRC calculation indicates that the data sent to the disk is incorrect. The data has not been written. Repeat the write operation. If write errors recur, make certain that all disk cables are properly connected.

If either error persists, inform a Wang Service Representative.

I97 LRC Error

A longitudinal redundancy check error occurred while a sector was being written or read. An LRC error usually indicates a transmission error between the disk and the CPU. Press RESET and rerun the program. If the error recurs, rewrite the flawed sector; the sector may have been previously written incorrectly. If the error persists, contact a Wang Service Representative.

I98 Illegal Sector Address or No Platter

The indicated sector is not on the disk platter, or the specified drive contains no platter. Ensure that the correct drive is being accessed. Correct the program statement, or ensure that the diskette is inserted into the drive.

I99 Read-After-Write-Error

The comparison of read-after-write to a disk sector failed, usually indicating a defective platter. Rewrite the information; the data may have been previously written incorrectly. If the error recurs, replace the platter. If the error persists, contact a Wang Service Representative.

SYSTEM (EXECUTION) ERROR MESSAGES

The system initially checks each text line for various types of errors as the line is entered by the programmer during program resolution and execution. The system responds to an error condition by producing an audible tone, terminating the current operation immediately, displaying the erroneous line, and presenting beneath it the message ERR followed by an error code, with an arrow pointing to the approximate position of the error. Note that the system stops error scanning when the first error is detected. Thus, if a line contains more than one error, only the first is detected and reported by the system. Some errors can be recovered under program control.

The error codes with a two-digit number preceded by a letter prefix (for example, A04) are the type that occur once the system program has been given control. The letter identifies the particular class of errors to which the error belongs, while the two-digit number identifies the specific error condition. (For example, an error commonly encountered during text entry is S13, Missing Comma. The S indicates a syntax error, and the 13 identifies the error uniquely as Missing Comma.) There are seven classes of error conditions, each identified by a unique letter prefix in the error code. These error classes and the letter prefixes which identify them follow.

Class of Errors	Letter Prefix
Miscellaneous errors	A
Syntax errors	S
Program errors	P
Computation errors	C
Execution errors	X
Disk errors	D
I/O errors	I

See the complete list of the errors included in each class and their specific recovery procedures in the *Multiuser BASIC-2 Language Reference Manual*.

Errors in the first three classes listed (Miscellaneous errors, Syntax errors, and Program errors) are detected during text entry or program resolution and cause the system to terminate the current operation and display an error message. The operator then must correct the error before proceeding with further operations. Errors of this kind are called nonrecoverable, because they cannot be recovered under program control.

Errors in the four remaining classes (Computation errors, Execution errors, Disk errors, and I/O errors) typically occur during program execution and can be recovered under program control without aborting the program or distrupting the display with an error message. Errors which can be intercepted by the program before the operating system intervenes are called recoverable errors. Three BASIC-2 instructions are provided for intercepting and responding to recoverable errors: the SELECT ERROR statement, the ERR function, and the ERROR statement.

See the *Multiuser BASIC-2 Language Reference Manual* for a discussion of error recovery under program control.

INDEX

F

G

H

I

K

L

M

N

O

P

R

WANG

Customer Comment Form

Publication Number 715-1213

Title CS INTRODUCTORY MANUAL

Help Us Help You . . .

We've worked hard to make this document useful, readable, and technically accurate. Did we succeed? Only you can tell us! Your comments and suggestions will help us improve our technical communications. Please take a few minutes to let us know how you feel.

How did you receive this publication?

- ☐ Support or Sales Rep
- ☐ Wang Supplies Division
- ☐ From another user
- ☐ Enclosed with equipment
- ☐ Don't know
- ☐ Other _____

How did you use this Publication?

- ☐ Introduction to the subject
- ☐ Classroom text (student)
- ☐ Classroom text (teacher)
- ☐ Self-study text
- ☐ Aid to advanced knowledge
- ☐ Guide to operating instructions
- ☐ As a reference manual
- ☐ Other _____

Please rate the quality of this publication in each of the following areas.

	EXCELLENT	GOOD	FAIR	POOR	VERY POOR
Technical Accuracy — Does the system work the way the manual says it does?	☐	☐	☐	☐	☐
Readability — Is the manual easy to read and understand?	☐	☐	☐	☐	☐
Clarity — Are the instructions easy to follow?	☐	☐	☐	☐	☐
Examples — Were they helpful, realistic? Were there enough of them?	☐	☐	☐	☐	☐
Organization — Was it logical? Was it easy to find what you needed to know?	☐	☐	☐	☐	☐
Illustrations — Were they clear and useful?	☐	☐	☐	☐	☐
Physical Attractiveness — What did you think of the printing, binding, etc?	☐	☐	☐	☐	☐

Were there any terms or concepts that were not defined properly? ☐ Y ☐ N If so, what were they? _____

After reading this document do you feel that you will be able to operate the equipment/software? ☐ Yes ☐ No ☐ Yes, with practice

What errors or faults did you find in the manual? (Please include page numbers) _____

Do you have any other comments or suggestions? _____

Name _____ **Street** _____

Title _____ **City** _____

Dept/Mail Stop _____ **State/Country** _____

Company _____ **Zip Code** _____ **Telephone** _____

Thank you for your help.

Printed in U.S.A. 14-3140 7-83

WANG

Cut along dotted line

Fold

NO POSTAGE
NECESSARY IF
MAILED IN THE
UNITED STATES

WANG

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 16 LOWELL, MA

POSTAGE WILL BE PAID BY ADDRESSEE

Wang Laboratories, Inc.
Technical Publications Dept.
M/S 012-260
One Industrial Avenue
Lowell, Massachusetts 01851-9971

Fold

WANG

To Order by Phone, Call:
1-(800)225-0234
Telex 172108

Order Form for Wang Manuals and Documentation

① Customer Number (If Known)

② Bill To: Ship To:

③ Customer Contact: ( ) ( ) Phone Name

④ Date Purchase Order Number

⑤ Taxable Yes ☐ No ☐ ⑥ Tax Exempt Number

⑦ Credit This Order to A Wang Salesperson Please Complete Salesperson's Name Employee No. RDB No.

⑧ Document Number	Description	Quantity	⑨ Unit Price	Total Price

⑩ Authorized Signature Date

Sub Total	
Less Any Applicable Discount	
Sub Total	
LocalState Tax	
Total Amount	

☐ Check this box if you would like a free copy of
WangDirect Software & Literature Catalog
(711-0888A)

Ordering Instructions

1. If you have purchased supplies from Wang before, and know your Customer Number, please write it here.
2. Provide appropriate Billing Address and Shipping Address.
3. Please provide a phone number and name, should it be necessary for WANG to contact you about your order.
4. Your purchase order number and date.
5. Show whether order is taxable or not.
6. If tax exempt, please provide your exemption number.
7. If you wish credit for this order to be given to a WANG salesperson, please complete.
8. Show part numbers, description and quantity for each product ordered.
9. *Pricing extensions and totaling can be completed at your option; Wang will refigure these prices and add freight on your invoice.*
10. Signature of authorized buyer and date.

Wang Terms and Conditions

1. **TAXES** — Prices are exclusive of all sales, use, and like taxes.
2. **DELIVERY** — Delivery will be F.O.B. Wang's plant. Customer will be billed for freight charges; and unless customer specifies otherwise, all shipments will go best way surface as determined by Wang. Wang shall not assume any liability in connection with the shipment nor shall the carrier be construed to be an agent of Wang. If the customer requests that Wang arrange for insurance the customer will be billed for the insurance charges.
3. **PAYMENT** — Terms are net 30 days from date of invoice. Unless otherwise stated by customer, partial shipments will generate partial invoices.
4. **PRICES** — The prices shown are subject to change without notice. Individual document prices may be found in the WangDirect Software & Literature Catalog (711-0888A)
5. **LIMITATION OF LIABILITY** — In no event shall Wang be liable for loss of data or for special, incidental or consequential damages in connection with or arising out of the use of or information contained in any manuals or documentation furnished hereunder.

Printed in U.S.A. 14-3141A 2-88

WANG

Cut along dotted line

Fold

NO POSTAGE
NECESSARY IF
MAILED IN THE
UNITED STATES

WANG

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 16 LOWELL, MA

POSTAGE WILL BE PAID BY ADDRESSEE

WangDirect
Wang Laboratories, Inc.
M/S 017-110
800 Chelmsford Street
Lowell, Massachusetts 01851-9972

Fold