WANG

2200

BASIC-2 Multiuser Operating System
Release 2.4
Software Bulletin

2200 BASIC-2 Multiuser Operating System

Release 2.4

Software Bulletin

First Edition -- August, 1982

700-7643

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 this software package was acquired, nor increases in any way Wang's liability to the customer. In no event shall Wang Laboratories, Inc., or its subsidiaries be liable for incidental or consequential damages in connection with or arising from the use of the software package, the accompanying manual, or any related materials.

NOTICE:

All Wang Program Products are licensed to customers in accordance with the terms and conditions of the Wang Laboratories, Inc. Standard Program Products License; no ownership of Wang Software is transferred and any use beyond the terms of the aforesaid License, without the written authorization of Wang Laboratories, Inc., is prohibited.

WANG

WANG LABORATORIES, INC., ONE INDUSTRIAL AVENUE, LOWELL, MA 01851 • TEL: 617/459-5000, TWX 710-343-6769, TELEX 94-7421

PREFACE

This bulletin describes the changes and features of Release 2.4 of the 2200 BASIC-2 Multiuser Operating System. The purpose of this Software Bulletin is to provide up-to-date documentation for software enhancements to the 2200 Series until the appropriate manuals are revised.

This documentation is intended to be used in conjunction with the following manuals:

- BASIC-2 Utilities Reference Manual (700-6855)
- Wang BASIC-2 Language Reference Manual (700-4080D)
- Wang BASIC-2 Disk Reference Manual (700-4081G)

CONTENTS

FIGURES

CHAPTER 1
SYSTEM PLATTER (RELEASE 2.4)

1.1 INTRODUCTION

Release 2.4 of the BASIC-2 Multiuser Operating System is issued primarily to provide support for the Model 2236MXE Terminal Processor and SVP Option-W. Release 2.4 includes the following new features and enhancements to earlier versions of the operating system:

- MXE Support

  Release 2.4 (or later) of the BASIC-2 Operating System is required for BASIC-2 systems with a Model 2236MXE Terminal Processor or SVP Option-W. The Model 2236MXE cannot be used with the VP BASIC-2 Operating System.

- DATE and TIME

  The Model 2236MXE Terminal Processor, SVP Option-W, and the current version of the Model 22C32 Triple Controller include a time-of-day clock that is accessible through the new BASIC-2 functions DATE and TIME.

- Screen Input

  Release 2.4, when used with a Model 2336DW Terminal and a Model 2236MXE Terminal Processor or SVP Option-W, allows the image displayed on the screen to be transmitted to a BASIC program for further processing. BASIC receives the screen data with an INPUT SCREEN statement.

- Terminal Disconnect

  The 2236MXE and SVP Option-W can alert the 2200 CPU when a local or remote terminal is connected or disconnected. With the BASIC-2 Release 2.4 Operating System, a programmer can use the terminal disconnect capability to initiate bookkeeping procedures, force a terminal to disconnect automatically after a specified period of time, or initiate log on and log off programs.

- Disk Write

  If a Data Error (Error 96) occurs during a disk write operation, the system will retry the operation up to three times. This is expected to reduce error rates for disk operations and should be particularly effective for multiplexed 2280 disk units.

NOTE

VP BASIC-2 will not be updated to support the Model 2236MXE and the SVP Option-W. Therefore, the DATE, TIME, $DISCONNECT, and INPUT SCREEN statements will generate syntax errors.

1.2 RELEASE INFORMATION

Release 2.4 of the BASIC-2 Multiuser Operating System is available on the following media:

Part Number	Type	Comment
701-2294Q	SSSD	2200 MVP, MVPC, LVP, LVPC (2270A Diskette Drive)
731-0058	DSDD	2200 LVP, LVPC, SVP (DSDD Diskette Drive)

1.3 CHANGES TO THE SYSTEM PLATTER

The system platter includes the MVP Operating System and BASIC-2 language processor, the system diagnostics, and the system utilities. The following system files have been revised or added since Release 2.3:

@MVP - MVP Operating System and BASIC-2 Language Processor (Rel. 2.4)
@SYSMVPB - System Menu
@MXE0 - MXE Microcode
@CLOC - Initialize Date & Time Utility
@DATE - File containing system default date
@BACKUP - Backup Platter Utility
@INSTALL - System Install Utility
@MOVEFIL - Move File Utility
@DAVFU - Vertical Format Control Utility

The Initialize Date & Time utility (@CLOC) has been added since Release 2.3. This utility, which allows an operator to set the date and time, is discussed in Chapter 3.

The name of the Install System Files utility has been changed to System Install utility. However, the purpose and operation of the System Install utility, as well as all other utilities, remains the same. Refer to the BASIC-2 Utilities Reference Manual (700-6855) for information on these system utilities.

1.4 CORRECTED ANOMALIES

The PACK statement now operates correctly when packing a numeric array with an exponential format. Previously, an incorrect exponent was occasionally stored.

In the Backup Platter utility (@BACKUP), the system displays the activity message "Creating @BADSCTR" instead of the message "Creating @BADSECTR". Additionally, if a disk error occurs when the utility attempts to scratch the destination platter, the system displays an error message instructing the user to correct the problem and then retries the scratch operation. Previously, the utility would not retry the scratch operation.

In the Move File utility (@MOVEFIL), it is now possible to input fields for source and destination disk drives that are case insensitive (i.e., an operator can enter an address in either uppercase or lowercase.)

In the Vertical Format Control utility (@DAVFU), the instructions displayed on the screen pertaining to the loading of a VFU data file were incorrect. The previous version displayed the following instruction:

```
DATA LOAD DC T$()
```

This instruction has been corrected to read as follows:

```
DATA LOAD DC x$, T$(): REM x$ is any scratch variable
```

CHAPTER 2
MVP BASIC-2 LANGUAGE ENHANCEMENTS

2.1 INTRODUCTION

Release 2.4 includes a system date and time, a connect/disconnect detection facility, and a screen input facility. The following BASIC-2 language enhancements are provided to support these new features:

- The DATE statement sets or changes the system date; the DATE function returns the value of the current date.
- The TIME statement sets or changes the system time; the TIME function returns the value of the current date.
- The $DISCONNECT statement enables or disables terminal disconnection detection.
- The INPUT SCREEN statement reads the screen of the terminal attached to the current partition and stores an image of the screen in an alpha-variable.

2.2 DATE AND TIME

BASIC-2 can maintain a system date and time. A programmer uses the DATE and TIME functions to access the clock. The clock is set with the DATE and TIME statements. Refer to Section 2.5 for a description of the DATE and TIME functions and statements.

To set or adjust the time and date, Wang Laboratories, Inc., provides an Initialize Date & Time utility. This utility resides on the system platter; it is discussed in Chapter 3.

ERR X79

To support the DATE and TIME statements, Release 2.4 includes the following new error message:

ERR X79 - Invalid Password

If an incorrect password is specified in either the DATE or TIME statement, the system returns an ERR X79. This error is recoverable (i.e., a programmer can intercept and respond to the error under program control). Refer to Chapter 9 of the BASIC-2 Language Reference Manual for more information on error control features.

2.3 TERMINAL CONNECT/DISCONNECT DETECTION FACILITY

The terminal connect/disconnect detection facility enables 2200 BASIC programs to be written to monitor system users and their use. With this facility, the operating system can detect when a terminal is connected or disconnected. Upon disconnection, the system can initiate bookkeeping procedures specified by the system programmer. The $DISCONNECT statement enables or disables disconnect detection. Refer to Section 2.5 for a description of the $DISCONNECT statement.

The following list summarizes the features of the terminal connect/disconnect detection facility:

- Connect detection for partition allocation and initiation of a user log on program.
- Forced disconnection if the user does not log on within the program specified time. Thus, a user is prevented from tying up a port without completing the log on procedure.
- Program control of the connect/disconnect facility through BASIC.
- Disconnect detection for initiation of user log off programs.
- BASIC TIME and DATE functions for logging system use.

System Requirements

The connect/disconnect facility requires the following hardware and software:

- 2200MVP, LVP, or SVP Central Processing Unit
- MVP BASIC-2 Release 2.4 (or later) or BASIC-3 Release 1.0 (or later)
- Model 2236MXE Terminal Processor (for MVP and LVP systems)
- Option-W (for SVP systems)

2.4 SCREEN INPUT FACILITY

The 2200 screen input facility enables a BASIC program to receive an image of a terminal screen display. This program would typically print the image on a system printer or save the image in a disk file for processing. A screen input can be requested by a BASIC program or by an operator. In the first case, a BASIC program executes an INPUT SCREEN statement that reads the screen of the terminal attached to the current partition. In the second case, the operator holds down the SHIFT and EDIT keys, thereby attaching the terminal to a partition that has previously been made available for screen input. Refer to Section 2.5 for a description of the INPUT SCREEN statement and the procedure for operator-initiated screen inputs.

System requirements

The screen input facility requires the following hardware and software:

- 2200MVP, LVP, or SVP Central Processing Unit
- MVP BASIC-2 Release 2.4 (or later) or BASIC-3 Release 1.0 (or later)
- Model 2336DW Terminal
- Model 2236MXE Terminal Processor (for MVP and LVP systems)
- Option-W (for SVP systems)

2.5 GENERAL FORMS

General forms of the DATE and TIME functions and statements, plus the $DISCONNECT and INPUT SCREEN statements are presented on the following pages.

DATE

Format (as a statement):

$$\text{DATE} = \begin{Bmatrix}\text{alpha-variable}\\ \text{literal-string}\end{Bmatrix} \quad \text{PASSWORD} \quad \begin{Bmatrix}\text{alpha-variable}\\ \text{literal-string}\end{Bmatrix}$$

Format (as a function):

DATE

DATE Statement

The DATE statement sets or changes the system date. The programmer can specify the new ASCII date in an alpha-variable or a literal-string. The date is specified as a six character alphanumeric value in the form YYMMDD (year, month, day).

The alpha-variable or literal-string following the keyword PASSWORD represents the system password. If the given password is correct (i.e., matches @GENPART password), the system date is updated. Otherwise, the system returns a recoverable error (ERR X79 - Invalid Password).

The system date can be set even if there is no system clock. This capability allows a clockless system to maintain the system date manually.

Examples of valid syntax:

```
DATE = "820801" PASSWORD "SYSTEM"

DATE = D$ PASSWORD P$
```

DATE Function

DATE is an alphanumeric function that returns a six character ASCII string containing the current date in the form YYMMDD (year, month, day). DATE is used as an operand in alphanumeric expressions.

Examples of valid syntax:

```
T$ = DATE & TIME

A$ = DATE
```

TIME

Format (as a statement):

$$\text{TIME} = \begin{Bmatrix}\text{alpha-variable}\\ \text{literal-string}\end{Bmatrix} \quad \text{PASSWORD} \quad \begin{Bmatrix}\text{alpha-variable}\\ \text{literal-string}\end{Bmatrix}$$

Format (as a function):

TIME

TIME Statement

The TIME statement sets or changes the system time. The programmer can specify the new ASCII time in an alpha-variable or a literal-string. The time is specified as a six character alphanumeric value in the form HHMMSS (hours, minutes, seconds).

The alpha-variable or literal-string following the keyword PASSWORD represents the system password. If the given password is correct (i.e., matches @GENPART password), the system time is updated. Otherwise, the system returns a recoverable error (ERR X79 - Invalid Password).

The system time can only be set if a system clock exists. If a clock does not exist and a user attempts to set the time, the system signals an error.

Examples of valid syntax:

```
TIME = "170001" PASSWORD "SYSTEM"

TIME = T$ PASSWORD P$
```

TIME Function

TIME is an alphanumeric function that returns an eight character ASCII string containing the current time in the form HHMMSSCC (hours, minutes, seconds, centiseconds). TIME is used as an operand in alphanumeric expressions.

The TIME function returns a value of 99999999 (i.e., invalid) if there is no system clock.

Examples of valid syntax:

```
A$ = TIME & " " & DATE

T$ = TIME
```

$DISCONNECT

Format:

```
                 {ON  [expression]}
$DISCONNECT      {                 }
                 {OFF              }
```

where:

0 ≤ value of expression ≤ 65534

The $DISCONNECT statement enables or disables terminal disconnection detection. Once enabled, the operating system can detect when a terminal is disconnected. A local terminal is considered to be disconnected when the terminal is powered off. A remote terminal is disconnected when the phone communication is terminated. The operating system cannot distinguish between disconnections caused by timeout, modem disconnection, or terminal power off.

A $DISCONNECT ON statement enables disconnect detection. A $DISCONNECT OFF statement disables disconnect detection; this is the default state of the system. Issuing a subsequent $DISCONNECT statement <u>always</u> overrides the previous command and sets a new state.

The optional expression in the $DISCONNECT ON statement represents the time in seconds after which the operating system forces a disconnection. If specified, the operating system forces a disconnection of the terminal when the value of the expression equals zero. The counter remains at zero until another $DISCONNECT ON statement sets the time expression. Therefore, after the terminal is disconnected, the next terminal at the same port is not disconnected until the execution of another $DISCONNECT ON statement.

$DISCONNECT is a command to the port attached to the partition issuing the statement and not to the partition. Therefore, changing partitions or terminals does not affect the way $DISCONNECT affects a specific port (i.e., $DISCONNECT remains in effect after the execution of a $RELEASE PART or $RELEASE TERMINAL statement).

Terminal Connect Detection

A local terminal is considered to be connected when it is powered on. A remote terminal is considered to be connected when the phone line link between the terminal and the terminal processor is established.

When the operating system detects a terminal connection or a RESET from a terminal that does not control any partitions, the operating system automatically assigns an available partition to that terminal. All partitions assigned to the null terminal (i.e., terminal 0) and waiting for a terminal (i.e., having accessed a statement that performs I/O to the terminal, such as PRINT or LINPUT) are available partitions. If the terminal already controls one or more partitions, or if no available partitions exist, the operating system does not perform any action.

Once a partition is assigned to a terminal, this partition can then execute a log on program that issues a $DISCONNECT ON statement and sets a time limit for the user to complete the log on sequence. (Refer to Example 1, "Sample Log On Program".) If the user does not complete the log on sequence within the program specified time, the operating system automatically disconnects the user. Since a user can initiate but cannot complete the log on procedure, forced disconnection prevents the user from tying up a terminal port. If, however, the user properly completes the log on procedure, the forced disconnection can be overridden by executing a $DISCONNECT ON statement with no specified disconnect time.

Example 1: Sample Log On Program

```
0010 REM @CONNECT -- Sample Log On Program
0020 REM To log system users, run this program (or similar
   : REM program) in all released partitions.  When a terminal is
   : REM connected, the operating system assigns a partition to the
   : REM terminal and begins execution of this program.
0025 REM For protection, the partition should be nonprogrammable.
0030 REM%

Disconnect terminal if not logged on within 2 minutes

   : $DISCONNECT ON 120
0040 REM%

Title

   : PRINT HEX(03); AT(0,30); "LOG ON"
0050 REM%

Obtain user's password

   : PRINT AT(10,20,8); "Enter password: ";
   : FOR I = 1 TO 8
   : KEYIN STR(P$,I,1)
   : PRINT HEX(8B);
   : NEXT I
0060 REM%

Verify user's password and log on user

   : REM Perform any required log on procedures

0070 REM%

Turn off the disconnect timeout

   : $DISCONNECT ON
0080 REM%

Load menu

   : LOAD RUN T "START"
   : REM START should check that user has logged on
```

Terminal Disconnect Detection

The operating system is informed of disconnections only if disconnect is enabled. Upon detection of a disconnect, the operating system forces all partitions assigned to that terminal to run a user-written BASIC program called @DISCNCT. (Refer to Example 2, "Sample Log Off Program".) The process is functionally equivalent to typing RESET followed by LOAD RUN "@DISCNCT". However, running the program occurs independently of the terminal.

The @DISCNCT program can perform any required log off or accounting procedures concerning system use. Typically, this program makes the partition available to other users by executing a $RELEASE PART statement. Thereafter, the released partition can execute a log on program that interacts with the user assigned this partition when terminal connections are detected.

If disconnect detection is not enabled for the terminal port, the operating system does not perform any action following the disconnection.

Example 2: Sample Log Off Program

```
0010 REM%

@DISCNCT -- Sample Log Off Program

0020 REM If terminal disconnect detection is enabled and a terminal
   : REM disconnects, the operating system automatically runs the
   : REM @DISCNCT program in all partitions assigned to that terminal
0030 REM%

Release the partition

   : $RELEASE PART
0040 REM%

Log off the user

   : REM Perform any required log off procedures
0050 REM%

Load a log on program

   : LOAD T "@CONNECT"
```

Examples of valid syntax:

```
$DISCONNECT ON

$DISCONNECT ON 60

$DISCONNECT OFF
```

INPUT SCREEN

Format:

INPUT SCREEN alpha-variable

The INPUT SCREEN statement reads the screen of the terminal attached to the current partition and stores an image of the screen in the alpha-variable. Each character on the screen and its associated display attributes are represented in the screen image.

INPUT SCREEN can only be used with controllers (e.g., Model 2236MXE and SVP Option-W) and terminals (e.g., Model 2336DW) that support the screen input facility. Attempting to execute an INPUT SCREEN statement with other terminals or controllers results in an error.

During a screen input, the terminal sends a total of 4080 bytes (characters) to the alpha-variable. Therefore, the alpha-variable should be at least 4080 characters in length in order to receive a full screen. The screen image consists of the following items:

- Terminal self-identification message
- Current cursor position
- Characters currently displayed
- Display attributes for each character

The first 78 bytes contain the self-identification message. The message identifies the type of terminal and its character set. The next 2 bytes identify the cursor location. The first byte identifies the row position; the second byte identifies the column position. Rows are numbered 0 to 24; columns are numbered 0 to 79.

The following 2000 bytes (25 rows by 80 columns) represent the characters currently displayed. The terminal uses the character codes of the Alternate Character Set (refer to Figure 2-1). Codes are sent row by row, starting at the first character in the first row. The 25th row of characters is all zeroes.

Each character has an associated attribute byte describing how the character is displayed. The 2000 bytes representing the characters are followed by 2000 bytes representing the display attributes. The 25th row of display attributes is only used for the box graphics under the characters of the 24th row. A display attribute byte has the following format:

bit 80 = 1 if character graphic
bit 40 = 1 if reverse video
bit 20 = 1 if blink
bit 10 = 1 if high intensity
bit 08 = 1 if underline
bit 04 = 1 if left horizontal box graphic segment
bit 02 = 1 if right horizontal box graphic segment
bit 01 = 1 if vertical box graphic segment

Only a foreground partition (i.e., a partition with a terminal attached) can execute an INPUT SCREEN statement. If a background partition with a terminal assigned to it issues INPUT SCREEN, execution is suspended until the terminal is attached to this partition.

Operator-Initiated Screen Input

Executing INPUT SCREEN in a partition with no terminal assigned to it (i.e., a $RELEASE PART statement is executed) notifies the operating system that this partition is available to receive screen input initiated by a terminal operator. Execution is suspended until a request for screen input is received.

To initiate a request for screen input, the operator must hold down the SHIFT and EDIT keys for at least two seconds. The operating system then temporarily attaches the terminal to a partition waiting to receive screen input. If no such partition is available, the screen input request remains pending until a partition becomes available, or until the operator aborts the request. On the Model 2336DW terminal, the operator aborts the request by pressing the RESET key or the SHIFT and RESET keys. Pressing the RESET key terminates the screen input request and leaves the CRT screen intact. Pressing the SHIFT and RESET keys terminates the request and clears the screen.

To handle more than one screen input concurrently, more than one partition can be set up to receive screen input. The operating system selects one of the available partitions for each screen input. Once the terminal is attached to a partition, the INPUT SCREEN statement resumes execution.

When INPUT SCREEN is finished, the partition remains assigned to the terminal, but becomes a background partition. The programmer can identify the terminal that executed the screen input request by using the #TERM function. As soon as the screen image is processed, the partition can be made available for another screen input by reexecuting the $RELEASE PART and INPUT SCREEN statements.

Example:

The following example outlines how to structure a program to receive screen input:

```
10 DIM A$(51)80
20 REM Release partition from any terminal
 : $RELEASE PART
30 REM Wait for screen input
 : INPUT SCREEN A$( )
40 REM Process screen input
50 GOTO 20
```

Example of valid syntax:

```
INPUT SCREEN A$( )
```

High-order HEX Digit

Low-order HEX Digit	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	â	Space	0	@	P	°	p	.	•						
1	ê	!	1	A	Q	a	q	♦	◊						
2	î	"	2	B	R	b	r	▶	▲						
3	ô	#	3	C	S	c	s	◀	▼						
4	û	$	4	D	T	d	t	→	↓						
5	ä	%	5	E	U	e	u	⌊	⌉						
6	ë	&	6	F	V	f	v	|	√						
7	ï	'	7	G	W	g	w	··	∘						
8	ö	(	8	H	X	h	x	'	{						
9	ü	)	9	I	Y	i	y	`	}						
A	à	*	:	J	Z	j	z	^	Δ						
B	è	+	;	K	[	k	§	■	□						
C	ù	,	<	L	\	l	£	!!							
D	Ä	-	=	M	]	m	é	↕							
E	Ö	.	>	N	↑	n	ç	ß							
F	Ü	/	?	O	←	o	¢	¶							

Figure 2-1. Alternate Character Set

CHAPTER 3
INITIALIZE DATE & TIME UTILITY

3.1 INTRODUCTION

The operator can select the Initialize Date & Time utility from the System Utilities menu (refer to Figure 3-1). The Initialize Date & Time utility (@CLOC) allows the operator to edit the system date and, if a system clock exists, the system time. A system equipped with a clock automatically advances the date at midnight. If the system does not have a clock, the operator must reset the date each day.

```
                    ***** SYSTEM UTILITIES *****
Select item with SPACE & BACKSPACE.                    Partition 1, 61 K
Key RUN to execute, CLEAR or PREV SCRN for previous screen.    Terminal 1

                    . Partition Generator
                    . Partition Status
                    . Format Disk Platter
                    . Move File
                    . Backup Platter
                    . Recover from Backup
                    . System Install
                    . Initialize Date & Time
                    . Vertical Format Control
                    . Game
```

Figure 3-1. System Utilities Menu

3.2 OPERATION

Whenever the system password is not SYSTEM, the utility first requests the system password (refer to Figure 3-2). The operator enters the system password and presses the RETURN key to proceed to edit the date and/or time.

```
                    *** Initialize Date & Time ***

08/01/82                                                  :  :

                    Password:________

                                        Press FN or TAB to return to menu
```

Figure 3-2. Password Screen

When the password requirement is satisfied, the utility displays the last entered date, the calendar page for the corresponding month, the calendar page for the succeeding month, and an enter date prompt (refer to Figure 3-3).

The operator can then enter any valid date (in the form MM/DD/YY). When the operator presses RETURN, the system updates the date, saves the date in a data file named @DATE, and displays the appropriate calendar pages.

If the system permits, the operator can enter any valid time (in the form HH:MM:SS). When the operator presses RETURN, the system updates the time.

At this point, the operator can reedit the date and/or time. Pressing FN or TAB accepts the date and/or time and returns the System Utilities menu to the screen.

```
                    *** Initialize Date & Time ***

08/02/82                                                  11:28:09

  Enter date: 08/02/82                        Enter time: 11:28:09

        AUGUST 1982                                 SEPTEMBER 1982
SUN MON TUE WED THU FRI SAT                 SUN MON TUE WED THU FRI SAT
  1 [ 2] 3   4   5   6   7                                1   2   3   4
  8   9 10  11  12  13  14                    5   6   7   8   9  10  11
 15  16 17  18  19  20  21                   12  13  14  15  16  17  18
 22  23 24  25  26  27  28                   19  20  21  22  23  24  25
 29  30 31                                   26  27  28  29  30

                                        Press FN or TAB to accept date & time
                                        Press RETURN to modify date & time
```

Figure 3-3. Edit Date & Time Screen

WANG

ONE INDUSTRIAL AVENUE
LOWELL, MASSACHUSETTS 01851
TEL. (617) 459-5000
TWX 710-343-6769, TELEX 94-7421

Printed in U.S.A.
700-7643
8-82-5M

TO: Gene Mantoni
copies of memo only to distribution
FROM: Pete Seymour
DATE: November 2, 1982

SUBJECT: BASIC-2 Release 2.4 (Release Date = 11/01/82)

The attached diskettes contain the official software for MVP BASIC-2 Release 2.4 (Release Date = 11/01/82). The Software Release Document (S.E. PACKAGE NO. NA) was distributed on July 27, 1982; please refer to it for details of the release. The attached document gives a detailed account of the known anomalies and maintenance history that apply to Release 2.4. This page should replace page 4 of your copy of the Software Release document. The software ECO that applies to this release is #24307. This ECO has already been completed and signed off.

Distribution List

Department	Name	Mail Stop
Customer Engineering	J. Belanger	8236A
	J. Proulx	8236A
ECO Group	J. Mulno	1137
Hardware R & D	P. Congo	1339
Integration Lab	Dan Morelle	2752
	R. Rosen	2752
Manufacturing	B. Patel	2212
	R. Jones	1329
Manufacturing Engineering	B. Ruggirello	2434
Product Marketing	G. Mantoni	14A7A
	J. Thibault	14A7A
Product Quality Assurance	E. Dilorenzo	1363
Software Distribution	P. Welch	5222
Small Systems Development	E. Kelley	1383
	B. Patterson	1383
	N. Sen	1383
	J. Sevigny	1383
Technical Systems Support	[redacted]	[redacted]
	F. Sullivan	1213
Tech Writing	D. Buttlar	1112
	T. DiPietro	1112
	R. Fox	1112
	J. Green	1112

doc=MVP BASIC-2 (2.4) 'RELEASE DOCUMENT' cover sheet (11/02/82 A)

2. @MOVEFIL:

   a. Previously, the operator input fields for source and destination disk drives were not case insensitive. This bug has now been fixed.
   b. Previously, if more than 30 files were not moved due to the 'overwrite = no' condition, the utility would halt on a subscript overflow error. This problem has now been corrected; the utility will stop recording the names of such files after the 30th one recorded.

3. @DAVFU: The instructions displayed on the screen pertaining to the loading of a VFU data file were incorrect. The previous version contained the displayed instruction:

   DATA LOAD DC T$().

   This instruction has been corrected to read:

   DATA LOAD DC x$,T$() :REM x$ is any scratch variable

NOTE: All these corrections except 1-b and 2-b have been implemented and released on the last release of the VP system platter (Rel 2.5). Numbers 1-b and 2-b will be described in a future issue of the CSS Newsletter for VP users.

G-3: MAINTENANCE HISTORY: Known Anomalies

The following operational limitations exist in the current release of @MXEO (the 2236MXE operating software):

1) The system can be powered up from the first terminal ONLY. (This is compatable with the MXD terminal controller) The capability to power up the system from other terminals will be added in the next release of the MXE operating code.

2) The LED on the MXE is not valid after the system has been configured. The LED should be checked after power up but <u>before</u> selecting an operating system. When the 2200 is loaded with an operating system the LED will be turned off REGARDLESS of errors on the board.

3) When using WP or IDEAS to print a document on the <u>terminal printer</u>, the system may think the printer is ready when it is not. Upcoming maintenance releases of WP and IDEAS should rectify this problem.

NOTE: The above printer problem is not a bug in the MXE. It is due to a faulty implementation in the WP and IDEAS software packages. <u>system printers</u> are <u>not affected</u>.